Aeroelastic Flutter of Continuous Systems: A Generalized Laplace Transform Method

Article *in* Journal of Applied Mechanics · May 2016

DOI: 10.1115/1.4033597

2 authors:

Arion Pons
University of Cambridge
16 PUBLICATIONS **47** CITATIONS

SEE PROFILE

Stefanie Gutschmidt
University of Canterbury
55 PUBLICATIONS **337** CITATIONS

SEE PROFILE

Aeroelastic flutter of continuous systems: a Generalised Laplace Transform Method

Arion Pons[1]
Department of Engineering, University of Cambridge
Trumpington street, Cambridge CB2 1PZ, United Kingdom
adp53@cam.ac.uk

Stefanie Gutschmidt
Department of Mechanical Engineering, University of Canterbury
Private Bag 4800, Christchurch 8140, New Zealand
stefanie.gutschmidt@canterbury.ac.nz

ABSTRACT

This paper presents a generalisation of the Laplace Transform Method (LTM) for determining the flutter points of a linear ordinary-differential aeroelastic system – a linear system involving a spatial derivative as well as a time-eigenvalue parameter. Current implementations of the LTM have two major problems: they are unable to solve systems of arbitrary size, order and boundary conditions, and they require certain key operations to be performed by hand or with symbolic manipulation libraries. Our generalised method overcomes both these problems. We also devise a new method for solving and visualising the algebraic system that arises from the LTM procedure. We validate our Generalised LTM and novel solution method against both the Goland wing model and a large system of high differential order, as a demonstration of their effectiveness for solving such systems.

[1] Corresponding author

1 INTRODUCTION

Consider a linear ordinary-differential aeroelastic system, involving a spatial derivative as well as a time-eigenvalue parameter. One method for finding the flutter points of such a system is to discretize it and solve the resulting discrete flutter problem by any one of a wide range of algebraic methods. These algebraic methods include the well-established p-method, k-method, and p-k method [1], as well as more novel developments such as the μ-type methods [2–4], the pp-method [5] and others [6,7]. The systems arising from the discretisation of real-life problems are typically very large, and for a given discretisation scheme must be made larger if increased accuracy is desired. The overall accuracy of the computed flutter points is affected by errors from two sources: the error arising from the discretisation process, and the error arising from the solution of the discrete system – unless very simple aerodynamics are used, the discrete flutter problem cannot usually be solved with a direct solver [1,8]. Despite these errors, discretisation is widely used in aeroelastic instability analysis [9,10].

The Laplace Transform Method (LTM) is a solution process for constant-coefficient ordinary-differential aeroelastic flutter problems that takes a different approach. A discrete system is devised which represents the continuous problem exactly. There is no variability in the size of this discrete system: for any given differential system there exists only one possible size, which is on the order of the size of the original continuous system. The LTM is often said to be *exact* [11,12], though the term must be used with care. As with standard discretisation methods, the LTM discrete

system must generally be solved via an inexact iterative method. The LTM as it stands is restricted to systems with spatially-independent ('constant') coefficients – in aeroelastic terms this corresponds to fixed structural moduli and strip theory aerodynamics. Note however that coefficients may have arbitrary frequency dependence, and moreover that there are techniques for the reduction of spatially-dependent systems to constant-coefficient ones, with or without approximation. There is also potential for variable-coefficient analysis to be built directly into the LTM. In Section 8 we discuss this, though in the bulk of this work we consider systems with constant coefficents. It should also be noted that the LTM that we are referring to is distinct from other 'Laplace Transform Methods' in aeroelasticity and structural mechanics: that is, methods based solely on Laplace transformation in time. Such methods are not applicable to undiscretised differential eigenvalue problems, and are unrelated to our LTM.

The LTM was introduced by Goland [13], who used it to make flutter speed calculations for a coupled Euler-Bernoulli / Saint-Venant beam wing model under Theodorsen aerodynamics. Since then, the LTM has been applied to a number of problems in aeroelasticity and structural mechanics, including variants of Goland's model [14,15] as well as higher-order models arising in the flutter analysis of composite wing structures [16], folding wings [17] and biplane wings [18,19]. The LTM has also seen use in other areas of structural dynamics, for the modal analysis of thin-walled and axially-moving beams [11,20]. However, the method itself has undergone very little development since Goland; the only innovations of note being the procedure used by

Liska and Dowell [17] to handle more complex boundary conditions, and the theoretical groundwork established by Balakrishnan [21] for the analysis of continuous aeroelastic systems (though this was not specifically concerned with the LTM). This is disappointing because, as noted by Karpouzian and Librescu [22], the LTM has the potential to become a useful tool in reduced-order modelling. The existence of a general but practical exact solution method for large high-order systems with complex boundary conditions would be significant for aeroelasticity and structural mechanics research. Moreover, with the ever-increasing use of numerical solution methods in engineering, the exactness of the method makes it an attractive tool for the verification of more widespread approximate methods.

However, existing implementations of the LTM are not suitable for solving large or high-order differential flutter problems [23]. There are two major problems:

- Existing implementations derive and use specific results for systems of specific size (typically two coupled ODEs up to fourth-order derivatives) and a limited class of boundary conditions. While the LTM is theoretically valid for a wide class of systems and boundary conditions, a solution procedure for systems of arbitrary size and conditions is completely lacking from the literature.

- Significant prior manipulation of the system is required (by hand or symbolic algebra software). This is not feasible for large systems where, for example, symbolic expressions for determinants and inverses become increasingly

expensive to determine. A fully numerical approach for an arbitrary system has never been developed before.

In the following paper, we present a generalisation of the LTM which solves both of the above problems: it is valid for constant-coefficient systems of arbitrary size and differential order, with an arbitrary set of constant-coefficient boundary conditions. The method is also completely numerical, while still remaining exact (within machine precision). We provide examples throughout the paper to facilitate understanding, and we apply the generalised method to both a standard aeroelastic test problem – the Goland wing [13] – and a more complex test problem to demonstrate its capability.

2 FORMULATION AND INITIAL MANIPULATION

Consider an aeroelastic constant-coefficient partial-differential system which has been subjected to a Fourier or Laplace transform in time, turning it into a continuous eigenvalue problem. Any such system, consisting of N linear homogeneous ordinary differential equations in N dependent variables, can be represented as

$$\sum_{k=0}^{M} \left(\mathrm{T}_{ijk}(\chi, U) \right)_{ij} \frac{d^k}{dx^k} \boldsymbol{\phi}(x) = \mathbf{0}, \tag{1}$$

where x is the independent (spatial) variable, M is the maximum differential order of the system, $\boldsymbol{\phi}$ is a column vector of the N dependent variables and $\mathbf{T} \in \mathbb{C}^{N \times N \times (M+1)}$ is a third-order tensor of coefficients. We define the system as over some domain $x \in [L_0, L_{m-1}]$ which will be specified fully when we consider the boundary conditions in Section 3. Note that we use zero-based array numbering throughout this paper, with

the exception of one algorithm written in MATLAB syntax which will be noted. Tensor $\mathbf{T}$ is a function of two parameters, χ and U, which together locate a flutter point of the system. The exact definition of these parameters is not relevant, however for convienience we take χ to represent modal frequency and U the free-stream airspeed, with χ defined such that $\breve{\boldsymbol{\phi}}(x,t) = \boldsymbol{\phi}(x)e^{i\chi t}$. That is, the unstable half-plane is the top half plane, and $\mathrm{Im}(\chi)$ represents modal damping. The objective is to find the flutter points of the system, i.e. the set of modal frequencies and airspeeds $\{[\chi_k, U_k]\} \in \mathbb{R}^2$ for which a solution exists to Eq. (1). The case $\chi_k \neq 0$ corresponds to a dynamic flutter point, and $\chi_k = 0$ to a divergence point [1].

Example

To put Eq. (1) in context, take the following system as an example:

$$\begin{aligned} &EI_b h'''' + (-m - L_h)\chi^2 h + (mx_\theta - L_\theta)\chi^2\theta = 0 \\ &-GJ\theta'' + (mx_\theta - M_h)\chi^2 h + (-I - M_\theta)\chi^2\theta = 0, \end{aligned} \tag{2}$$

which describes a beam with lateral deflection h and twist angle θ, under Euler-Bernoulli beam theory and Saint-Venant torsion theory. The system has been subjected to a Fourier transform in the time variable ($\breve{h}(x,t) = h(t)e^{i\chi t}$ etc.), turning it from a PDE to an ordinary differential eigenproblem. The aerodynamic coefficients $\{L_h, L_\theta, M_h, M_\theta\}$, functions of U and χ, may be defined via Theodorsen's theory [24], the finite-state theory of Peters et al. [25], or an unsteady panel method solver. For the system of Eq. (2), we have $\boldsymbol{\phi} = [h, \theta]^T$ (the sequence of variables is irrelevant provided it is consistent within the system definition) and the following coefficient matrices:

$$\begin{aligned}
&\left(\mathrm{T}_{ij0}\right)_{ij} = \chi^2 \begin{bmatrix} -m - L_h & mx_\theta - L_\theta \\ mx_\theta - M_h & -I - M_\theta \end{bmatrix}, \\
&\left(\mathrm{T}_{ij1}\right)_{ij} = \mathbf{0}_2, \left(\mathrm{T}_{ij3}\right)_{ij} = \mathbf{0}_2, \\
&\left(\mathrm{T}_{ij2}\right)_{ij} = \begin{bmatrix} 0 & 0 \\ 0 & -GJ \end{bmatrix}, \left(\mathrm{T}_{ij4}\right)_{ij} = \begin{bmatrix} EI_b & 0 \\ 0 & 0 \end{bmatrix}.
\end{aligned} \tag{3}$$

We then take the Laplace transform of the dependent variable vector $\boldsymbol{\phi}$ from the spatial variable x to a state-space variable p: $\mathcal{L}\{\boldsymbol{\phi}(x)\} = \widehat{\boldsymbol{\phi}}(p)$. An n-th derivative of $\boldsymbol{\phi}$ has the transform

$$\mathcal{L}\left\{\frac{\partial^n}{\partial x^n}\boldsymbol{\phi}\right\} = p^n\widehat{\boldsymbol{\phi}} - \sum_{i=1}^{n} p^{n-i}\frac{\partial^{i-1}}{\partial x^{i-1}}\boldsymbol{\phi}\bigg|_{x=0}. \tag{4}$$

Note that for $n = 0$ the empty sum $\sum_1^0$ is defined to be zero [26]. Then, taking the Laplace transform of Eq. (1) and using the identitiy in Eq. (4), we obtain

$$\sum_{k=0}^{M} p^k\left(\mathrm{T}_{ijk}\right)_{ij}\widehat{\boldsymbol{\phi}} = \sum_{k=1}^{M}\sum_{l=1}^{k} p^{k-l}\left(\mathrm{T}_{ijk}\right)_{ij}\frac{\partial^{l-1}}{\partial x^{l-1}}\boldsymbol{\phi}\bigg|_{x=0}, \tag{5}$$

where we write the lower bound of the k-summation on the right-hand side as $k = 1$ instead of $k = 0$ because the sum $\sum_{l=1}^{k=0}$ is empty. The left-hand side of Eq. (5) can be immediately simplified as $\mathbf{E}\widehat{\boldsymbol{\phi}}$, defining the matrix $\mathbf{E}$ as

$$\mathbf{E}(\chi, U, p) = \sum_{k=0}^{M} p^k\left(\mathrm{T}_{ijk}(\chi, U)\right)_{ij}. \tag{6}$$

To simplify the right-hand side of Eq. (5), we define an extended state vector, $\boldsymbol{\psi}$, which contains all the dependent variables and their derivatives less than the maximum

differential order of that dependent variable in the system, Eq. (1). We order this vector firstly by dependent variable (consistent with $\boldsymbol{\phi}$), and secondly by differential order (ascending). For example, Eq. (2) has an extended state vector of $\boldsymbol{\psi} = [h, h', h'', h''', \theta, \theta']^T$. For small systems $\boldsymbol{\psi}$ can be easily derived by hand, but for an arbitrary coefficient tensor $\mathbf{T}$ we must employ a more rigorous method.

To do this we introduce a vector $\mathbf{n} \in \mathbb{N}^N$, where the i-th element of $\mathbf{n}$ represents the maximum differential order of the i-th dependent variable in $\boldsymbol{\phi}$ that is present in the system. $\mathbf{n}$ may easily be computed from $\mathbf{T}$, but will often be known *a priori*. We then define the cumulative order of the system, σ, as the sum of $\mathbf{n}$. The extended state vector $\boldsymbol{\psi}$ is thus a $\sigma \times 1$ vector. For example, Eq. (2) has $\mathbf{n} = [4, 2]$ and thus $\sigma = 6$. We then note that $\boldsymbol{\psi}$ is of the form

$$\boldsymbol{\psi} = \left(\frac{d^{\mathrm{q}_h}}{dx^{\mathrm{q}_h}} \phi_{\mathrm{r}_h} \right)_h, \tag{7}$$

where $\mathbf{r} \in \mathbb{N}^\sigma$ and $\mathbf{q} \in \mathbb{N}^\sigma$ are index-contraction vectors such that the h-th element of $\mathbf{r}$ represents the independent variable number in the h-th element of $\boldsymbol{\psi}$, and the h-th element of $\mathbf{q}$ represents the order of the derivative in the h-th element of $\boldsymbol{\psi}$. Vectors $\mathbf{r}$ and $\mathbf{q}$ are functions solely of $\mathbf{n}$, and though it is difficult to express them as an equation it is easy to do so as an algorithm. Algorithm (1), in MATLAB® R2013b syntax, will compute $\mathbf{r}$ and $\mathbf{q}$ given $\mathbf{n}$. Note that this algorithm uses MATLAB's one-based indexing and that $\mathtt{n} = \mathbf{n}$, $\mathtt{p} = \mathbf{p}$ and $\mathtt{q} = \mathbf{q}$.

```
1    cum_n = cumsum(n);
2    r = zeros(sum(n), 1); q = zeros(sum(n), 1);
3    for i = 1:length(n)
4        st = (i~=1)*cum_n(i - (i~=1)) + 1;
5        en = cum_n(i);
6        r(st:en) = i;
7        q(st:en) = 0:(n(i) - 1);
8    end
```

Algorithm (1)

For a 0-based indexing system, lines 4-5 should be replaced with:

```
4*   st = (i~=1)*cum_n(i - (i~=1));
5*   en = cum_n(i) - 1;
```

Algorithm (2)

With Algorithms (1) and (2) we have thus fully defined the extended state vector, $\boldsymbol{\psi}$.

Using Eq. (7) we can rewrite the right-hand side of Eq. (5) in the form

$$\sum_{k=1}^{M}\sum_{l=1}^{k} p^{k-l}\left(\mathrm{T}_{ijk}\right)_{ij}\frac{\partial^{l-1}}{\partial x^{l-1}}\boldsymbol{\phi}\bigg|_{x=0} = \mathbf{B}\boldsymbol{\psi}|_{x=0}, \tag{8}$$

for some $\mathbf{B} \in \mathbb{C}^{N\times\sigma}$. To determine an expression for $\mathbf{B}$, we rewrite the matrix product of the left-hand side of Eq. (8) as a summation, and convert the variable-length summation over l into a fixed-length summation using the Iverson brackets. We can then reorder the sequence of summations. The result is

$$\left(\sum_{l=1}^{M}\sum_{j=1}^{N-1}\sum_{k=1}^{M}[l\le k]_{\mathrm{IVR}}p^{k-l}\mathrm{T}_{ijk}\frac{\partial^{l-1}}{\partial x^{l-1}}\phi_j\bigg|_{x=0}\right)_i = \mathbf{B}\boldsymbol{\psi}|_{x=0}, \tag{9}$$

where $[\cdot]_{\mathrm{IVR}}$ are the Iverson brackets: that is, $[P]_{\mathrm{IVR}} = 1$ when proposition P is true and $[P]_{\mathrm{IVR}} = 0$ otherwise [26]. A single coefficient of a derivative of ϕ_j in the l, j double sum of Eq. (9) can thus be written as

$$\sum_{k=1}^{M}[l \leq k]_{\mathrm{IVR}} p^{k-l}\mathrm{T}_{ijk} = \sum_{k=l}^{M} p^{k-l}\mathrm{T}_{ijk}. \tag{10}$$

We then introduce a new index, h, representing the index of $\boldsymbol{\psi}$ referenced in the derivative summation of Eq. (9). This allows us to replace the l, j summation indices by defining two vectors $\mathbf{r} \in \mathbb{N}^{\sigma}$, $\mathbf{q} \in \mathbb{N}^{\sigma}$ where $\mathrm{q}_h = l - 1$ and $\mathrm{r}_h = j$. That is, r_h represents the independent variable number at h and q_h represents the order of the derivative at h – these two vectors are identical to those in Eq. (7) and Algorithms (1) and (2). Substituting them into Eq. (9) yields an expression for $\mathbf{B}$:

$$\mathbf{B}(\chi, U, p) = \left(\sum_{k=\mathrm{q}_h+1}^{M} p^{k-\mathrm{q}_h-1}\mathrm{T}_{i\mathrm{r}_h k}\,(\chi, U) \right)_{ih}. \tag{11}$$

Note that coefficients in the space T_{ij0} can never appear in $\mathbf{B}$ because $\mathrm{q}_h \geq 0$ always. Thus in most aeroelastic systems $\mathbf{B}$ will be independent of χ and U, as the higher-order differential coefficients are seldom a function of these parameters. Using $\mathbf{E}$ and $\mathbf{B}$ we can thus write Eq. (5) as the state-space system

$$\mathbf{E}(\chi, U, p)\widehat{\boldsymbol{\phi}} = \mathbf{B}(\chi, U, p)\boldsymbol{\psi}|_{x=0}, \tag{12}$$

which is of fundamental importance to the LTM. This is an algebraic eigenproblem in χ, dependent on parameter U. However, the initial spatial condition, $\boldsymbol{\psi}|_{x=0}$ is still unknown. This is dealt with in Sections 3 and 4.

3 BOUNDARY CONDITIONS

To constrain the system we must enforce a set of boundary conditions. Implementations of the LTM by most previous authors have applied boundary conditions at only two points ($x = 0$ and $x = L$) where the conditions at $x = 0$ are of the form $d^n\phi_i/dx^n|_{x=0} = 0$. When this is the case, a simple but restrictive method is avilable for their enforcement [14–16]. A more general method of implementing boundary conditions was explored by Liska and Dowell [17], and we will develop it further. Suppose we have a set of σ linear boundary conditions, spread over m different x-locations, $\{L_i\}_{i=0}^{m-1} = \{L_0 \dots L_{m-1}\}$. We do not require that $L_0 = 0$, though it is always possible to make this the case (via a coordinate change) as the system is linear. Any such set of conditions can be represented as

$$\mathbf{R}(\chi, U)\begin{bmatrix} \boldsymbol{\psi}|_{x=L_0} \\ \vdots \\ \boldsymbol{\psi}|_{x=L_{m-1}} \end{bmatrix} = \mathbf{0}, \tag{13}$$

where $\mathbf{R} \in \mathbb{C}^{\sigma \times m\sigma}$. Note that this representation allows periodic and other complex conditions. The boundary condition matrix $\mathbf{R}$ and the set of boundary x-locations $\{L_i\}_{i=0}^{m-1}$ must be defined at the beginning of the analysis. However, to use Eq. (13) to define $\boldsymbol{\psi}|_{x=0}$ in Eq. (12), we require a relationship between $\boldsymbol{\psi}|_{x=L_i}$ and $\boldsymbol{\psi}|_{x=0}$. This is detailed in Section 4.

Example

Consider the Euler-Bernoulli beam system of Eq. (2) with a clamped condition at $x = 0$ and a free condition at $x = L$. These conditions correspond to

$$h|_{x=0} = h'|_{x=0} = \theta|_{x=0} = 0$$
$$h''|_{x=L} = h'''|_{x=L} = \theta'|_{x=L} = 0. \tag{14}$$

In the form of Eq. (13), they become

$$\mathbf{R}\begin{bmatrix}\boldsymbol{\psi}|_{x=0}\\ \boldsymbol{\psi}|_{x=L}\end{bmatrix} = 0 \tag{15}$$

with $\mathbf{R} \in \mathbb{R}^{6\times 12}$:

$$\mathrm{R}_{ij} = \begin{cases}1 & [i,j] = [1,1],[2,2],[3,5],[4,9],[5,10],[6,12]\\ 0 & \text{otherwise}\end{cases} \tag{16}$$

---

4 THE FLUTTER DETERMINANT

So far we have derived the state-space representation of our differential system, and constrained it with a general boundary condition. The objective is now to synthesise these two components into an algebraic system that can be solved with algebraic solvers. Consider again Eq. (12): the state-space representation of Eq. (1). We will omit writing the dependence of matrices $\mathbf{E}$ and $\mathbf{B}$ on χ and U for ease of readability. Solving Eq. (12) for the state-space vector $\boldsymbol{\phi}(x)$ using Cramer's rule, and taking the inverse Laplace Transform via Heaviside partial fraction expansion [27], we obtain

$$\boldsymbol{\phi}(x) = \left(\sum_{k=0}^{\sigma-1} e^{\lambda_k x}\,\frac{\mathrm{adj}(\mathbf{E})\,\mathbf{B}}{\det(\mathbf{E})'}\bigg|_{p=\lambda_k}\right)\boldsymbol{\psi}|_{x=0}, \tag{17}$$

where $\det(\mathbf{E})' = \partial\,\det\big(\mathbf{E}(p)\big)/\partial p$, and $\boldsymbol{\lambda} \in \mathbb{C}^{\sigma}$ is the set of roots of $\det(\mathbf{E})$ with respect to p, i.e. the eigenvalues of the polynomial eigenproblem $\mathbf{E}(p)\widehat{\boldsymbol{\phi}} = 0$. For Eq. (17) to be valid these roots must be distinct. For the case of repeated roots see Section 5.2. There

are many algorithms for computing these roots or eigenvalues exactly [28–30]; we use the algorithm of Holzel and Bernstein [29], which is based on the singular value decomposition. Our implmentation takes tensor $\mathbf{T}$ as an input and returns the zeros of polynomial matrix $\mathbf{E}$. Note also that Eq. (17) requires the evaluation of the derivative of the determinant of $\mathbf{E}$: we provide a direct and error-free method of doing this in the Section 5.1.

For ease of manipulation we define a 3rd-order tensor $\mathbf{A} \in \mathbb{R}^{N\times\sigma\times\sigma}$ such that

$$\mathbf{A} = \left(\frac{(\mathrm{adj}(\mathbf{E})\,\mathbf{B})_{ij}}{\det(\mathbf{E})'}\bigg|_{p=\lambda_k} \right)_{ijk}, \tag{18}$$

which enables us to write Eq. (17) as

$$\boldsymbol{\phi}(x) = \left(\sum_{k=0}^{\sigma-1} \mathrm{A}_{ijk} e^{\lambda_k x} \right)_{ij} \boldsymbol{\psi}|_{x=0}. \tag{19}$$

Our next goal is to determine $\boldsymbol{\psi}(x)$, because this will allow us to analyse the boundary conditions. By computing the derivatives of $\boldsymbol{\phi}$ with respect to x in Eq. (7) we have

$$\boldsymbol{\psi} = \left(\sum_{k=0}^{\sigma-1} \sum_{j=0}^{\sigma-1} \mathrm{A}_{\mathrm{r}_h jk} {\lambda_k}^{\mathrm{q}_h} e^{\lambda_k x} \psi_j \big|_{x=0} \right)_h. \tag{20}$$

We define a tensor $\mathbf{C} \in \mathbb{C}^{\sigma\times\sigma\times\sigma}$ which is independent of x:

$$\mathbf{C} = \left(\mathrm{A}_{\mathrm{r}_h jk} {\lambda_k}^{\mathrm{q}_h} \right)_{hjk}, \tag{21}$$

from which we can define a matrix $\mathbf{K} \in \mathbb{C}^{\sigma\times\sigma}$ which is dependent on x:

$$\mathbf{K}(x) = \left(\sum_{k=0}^{\sigma-1} \mathrm{C}_{hjk} e^{\lambda_k x} \right)_{hj}, \tag{22}$$

such that

$$\boldsymbol{\psi}(x) = \mathbf{K}(x)\boldsymbol{\psi}|_{x=0}. \tag{23}$$

Thus we have determined $\boldsymbol{\psi}(x)$. Then lastly we may express the boundary condition (Eq. 13) as

$$\mathbf{D}(\chi, U)\boldsymbol{\psi}|_{x=0} = 0 \tag{24}$$

with $\mathbf{D} \in \mathbb{C}^{\sigma\times\sigma}$:

$$\mathbf{D}(\chi, U) = \mathbf{R}(\chi, U) \begin{bmatrix} \mathbf{K}|_{x=L_0}(\chi, U) \\ \vdots \\ \mathbf{K}|_{x=L_{m-1}}(\chi, U) \end{bmatrix}, \tag{25}$$

Eq. (24) is a discrete eigenvalue problem (in χ), dependent on the airspeed U. This eigenvalue problem is an exact representation of the original continuous eigenvalue problem, Eq. (1) – all the operations we have performed so far can be perfomed by a computer without approximation. We have reduced a completely general formulation of a constant-coefficient continuous flutter problem into an algebraic problem. However Eq. (24) is a nonlinear algebraic eigenvalue problem; that is, the dependence of $\mathbf{D}$ on χ is not expressible in closed form, even if the dependence of $\mathbf{T}$ and $\mathbf{R}$ on χ is very simple (e.g. polynomial). The objective now is to find the set of flutter points $\{[\chi_k, U_k]\}$ such that Eq. (24) is satisfied. We discuss methods for doing this in Section 6.

5 COMPUTATIONAL ASPECTS

5.1 Derivative of $\det(\mathbf{E})$

The expression for $\mathbf{A}$, Eq. (18), requires the evaluation of a derivative of a determinant. For the lower-order systems in current LTM literature, the determinant is usually evaluated and differentiated symbolically. We propose a simple new approach; namely, the application of Jacobi's formula [31]:

$$\det(\mathbf{E})' = \mathrm{tr}(\mathrm{adj}(\mathbf{E})\,\mathbf{E}'), \tag{26}$$

which implies

$$\mathbf{A} = \left(\frac{(\mathrm{adj}(\mathbf{E})\,\mathbf{B})_{ij}}{\mathrm{tr}(\mathrm{adj}(\mathbf{E})\,\mathbf{E}')}\bigg|_{p=\lambda_k} \right)_{ijk}. \tag{27}$$

The derivative of $\mathbf{E}$ ($\mathbf{E}'$) is very easy to determine from Eq. (6); it is

$$\mathbf{E}' = \sum_{k=0}^{M} k p^{k-1} \left(\mathrm{T}_{ijk}\right)_{ij}. \tag{28}$$

This completely eliminates the need for symbolic manipulation in this operation.

5.2 Non-distinct zeros of $\mathbf{E}$

It should be noted that the Heaviside expansion formula implicit in Eq. (17) is a simplification of a more general procedure which allows for repeated roots [32]. It is not particularly difficult to implement this general formula: $\widehat{\boldsymbol{\phi}}$ then has an inverse Laplace transform

$$\boldsymbol{\phi}(x) = \sum_{i=1}^{n} \sum_{j=1}^{m_i} \frac{c_{i,j}}{(j-1)!} x^{j-1} e^{\lambda_i x} \boldsymbol{\psi}|_{x=0} \tag{29}$$

for some coefficients $c_{i,j}$, and then the method can proceed as normal. Repeated roots have not yet been a problem in existing implementations of the LTM and so there is no immediate reason to complicate the analysis for their sake. However, the generalised expansion formula does provide a solution to the problem of repeated roots if it arises.

5.3 Algorithm

Algorithm 3 presents the procedure for evaluating the eigenproblem matrix $\mathbf{D}$ corresponding to any system tensor $\mathbf{T}$ with the associated boundary condition matrix $\mathbf{R}$. This is a summary of our derivations and analysis (Sections 2-5), and can be used to implement the Generalised LTM in a computer code. The use of this algorithm for solving the flutter problem of Eq. (24) is discussed in Section 6.

Algorithm 3 – Generalised LTM evaluation

initialise N, M, $\{L_i\}_{i=0}^{m-1}$, $\mathbf{T}$ and $\mathbf{R}$
compute $\mathbf{n}$, σ, (Section 2) $\mathbf{q}$ and $\mathbf{r}$ (Algorithm 1 and/or 2)
compute $\boldsymbol{\lambda}$, the zeros of $\mathbf{E}$ (see Section 4)
compute $\mathbf{E}|_{\lambda_k}$ (Eq. (6)), $\mathbf{B}|_{\lambda_k}$ (Eq. (11)) and $\mathbf{E}'|_{\lambda_k}$ (Eq. (28))
compute $\mathbf{A}$ (Eq. (27))
compute $\mathbf{C}$ (Eq. (21))
compute $\mathbf{K}|_{x=L_i}$ for all $\{L_i\}_{i=0}^{m-1}$ (Eq. (22))
compute $\mathbf{D}$ (Eq. (25))
return $\mathbf{D}$

6 SOLVING THE ALGEBRAIC SYSTEM – THE CONTOUR PLOT METHOD

The traditional p-method of determining the flutter points of an aeroelastic system is to define a set of airspeed values, $\{U_k\} \subset \mathbb{R}$, possibly nondimensionalised, and then solve the eigenvalue problem for the reduced complex eigenvalue, $\{p_i\} \subset \mathbb{C}$ at each U_k. This p is entirely distinct from our spatial state-space p. The flutter points are the roots of the modal damping curves ($-\mathrm{Re}(p)$). The relation between p and our χ is $p = L\imath\chi/U$, with the lengthscale L commonly defined as the aerofoil semichord. It is thus equivalent to compute $\{\chi_i\} \subset \mathbb{C}$ for each U_k; the only differences are the dimensions of the eigenvalue (χ is in rad/s) and that, in the determination of the flutter point, $\mathrm{Im}(\chi)$ represents modal damping instead of $-\mathrm{Re}(p)$. Indeed any χ-U data may simply be postprocessed into p-U data and vice-versa.

For the p-method modal damping paths to have physical relevance away from the flutter point (subcritically and supercritically) the aerodynamic model involved must be physically accurate for damped motion, i.e. $p, \chi \in \mathbb{C}$. We should note the test models we present in Section 7 utilise the analytical continuation of Theodorsen's function proposed by Luke and Dengler [33] – the evaluation of Theodorsen's function with complex argument ($p, \chi \in \mathbb{C}$), rather than the traditional restriction to simple harmonic motion. This continuation is used in gust response analysis [9]. It is known to be physically inaccurate in some cases [9,34,35]; however this is not a problem as the model is still mathematically valid, and yields consistent modal frequencies which may be compared to the results of other methods applied to this system. The methods of

analysis we are discussing apply to all stability problems of suitable form, not only those which model aeroelastic phenomena. Theodorsen's theory is used for convenience; other models such as the finite-state theory of Peters et al. [25] may be substituted for greater aeroelastic physical accuracy.

The application of the p-method to Eq. (24) is, however, difficult: the eigenproblem in p or χ is strongly nonlinear, even for trivial aerodynamic and structural models. Solvers for the nonlinear eigenvalue problem do exist [36,37], but they are not well-suited for use with the p-method. They are necessarily iterative, and if convergent will only find one eigenvalue for a given initial guess. Multiple initial guesses are necessary in order to have a reasonable chance of identifying all the modes of interest. Moreover, the mode to which a given initial guess converges is likely to change as U is changed, and so some form of mode-tracking is necessary in order to compute modal damping paths and thus the flutter points. Implementing a p-method search with a nonlinear solver is time-consuming and requires manual tuning.

It is for these reasons that we devise a different method. Instead of searching over $\{U_k\} \subset \mathbb{R}$ and solving a nonlinear eigenvalue problem at each step, we search over two dimensions, $\{[\chi_k,\ U_k]\} \subset \mathbb{R}^2$ and evaluate $\det(\mathbf{D})$ at each step. In general, $\det(\mathbf{D}) \in \mathbb{C}$. Hence Eq. (24) is equivalent to specifying

$$\mathrm{Re}(\det(\mathbf{D})) = 0, \qquad \mathrm{Im}(\det(\mathbf{D})) = 0 \tag{30}$$

Eq. (30) defines two sets of contours over $[\chi, U] \in \mathbb{R}^2$. The flutter points are the intersections of these two sets of contours, where Eq. (24) holds, i.e. $\det(\mathbf{D}) = 0$. Dynamic flutter points are intersections at nonzero χ, and divergence points are those at $\chi = 0$ (though not exactly so due to inevitable interpolation error). The procedure for determining the flutter points is thus as follows: (**1**) Define a grid of points $\{[\chi_k, U_k]\} \subset \mathbb{R}^2$; (**2**) Evaluate $d_k = \det\big(\mathbf{D}(\chi_k, U_k)\big)$; (**3**) Interpolate the contours of $\mathrm{Re}(d_k) = 0$ and $\mathrm{Im}(d_k) = 0$; and (**4**) Interpolate the intersection of these contours.

The main advantage of the contour plot is its simplicity and applicability to eigenvalue stability problems that are strongly nonlinear. Clearly if a perfect and low-cost solver for nonlinear eigenvalue problems were to exist then the p-method would always be superior, due to its exact treatment of subcritical and supercritical modal behaviour. However, the contour plot can provide approximate information about this modal behaviour too: the real contours (lines of $\mathrm{Re}(d_k) = 0$) approximate the modal frequencies of the structure as a function of airspeed, with the approximation improving as the modal damping of a mode gets closer to zero. While the precise relation between the two has not yet been established, Section 7.2 presents representative numerical evidence which indicates that there is a strong correspondence. Further discussion of other features of the contour plot may be found in [8,19].

The contour plot retains the exactness of the LTM – while the use of interpolation will mean that the computed flutter points have a small associated

imprecision, the method nevertheless avoids all the problems associated with system discretisation and meshing. For example, higher eigenvalues can be found with the same precision (and no greater computational cost) as lower eigenvalues. However the contour plot is not specifically tied to the LTM, and may be applied to eigenvalue stability problems arising from any source – its strength, however, is small and strongly nonlinear problems. We should note that our contour plot method is not equivalent to the contour plot methods used by Liska and Dowell [17] and Wang [38] which involve computing contours over the plane of $\chi \in \mathbb{C}$ for a given U. Our method only involves computing contours over the plane defined by $U \in \mathbb{R}$ and $\chi \in \mathbb{R}$ which requires significantly less computational effort. However, we use the method of Liska and Dowell [17] in Section 7.2 to validate our method. Further details may be found in Pons [8].

7 NUMERICAL SIMULATIONS

7.1 Goland wing natural frequencies (*in vacuo*)

To validate the Generalised LTM at a basic level, we use it to determine the natural frequencies of the Goland wing in a vacuum. This is a simple undamped free vibration problem. The governing eigenvalue problem is identical to Eq. (2), with the aerodynamic coefficients $\{L_h, L_\theta, M_h, M_\theta\}$ equal to zero: that is,

$$\begin{aligned} EI_b h'''' - m\chi^2 h + mx_\theta \chi^2 \theta = 0 \\ -GJ\theta'' + mx_\theta \chi^2 h - I\chi^2 \theta = 0. \end{aligned} \tag{31}$$

This system has clamped-free boundary conditions identical to those of Eq. (14). For values of the structural parameters $\{EI_b, GJ, m, I, x_\theta, L\}$, we take the metric Goland wing parameters described in [39]. Note again χ is defined such that $\left[\check{h}(x,t), \check{\theta}(x,t)\right] =$

$[h(t), \theta(t)]e^{i\chi t}$. As the system is undamped and there is no airspeed parameter, we have no need of the contour plot method and can simply search along $\chi \in \mathbb{R}$ to locate the natural frequencies. Figure 1 shows $\det(\mathbf{D})$ over $\chi \in [0, 400]$ rad/s. The first two eigenvalues (roots) can be located at $\chi = 48.23$ rad/s and $\chi = 104.01$ rad/s. They agree very well with the values in [39], which are $\chi = 48.24$ rad/s and $\chi = 104.05$ rad/s.

7.2 Goland wing flutter points

As an extension of the analysis performed in Section 7.1, we now evaluate the flutter points of the Goland wing. The governing eigenvalue problem is that of Eq. (2), with clamped-free boundary conditions as per Eq. (14). The aerodynamic coefficients $\{L_h, L_\theta, M_h, M_\theta\}$ are given by Theodorsen's theory [1,24] and are complex-valued nonlinear functions of the airspeed U and the modal frequency χ. For this analysis we take parameter values from Wang [38]. Figure 2 shows a contour plot for the Goland wing over $\chi \in [0^+, 600]$ rad/s and $U \in [0, 700]$ m/s. The first flutter point can be seen in the lower left half of Figure 2. It can be located at $U = 138.11$ m/s and $\chi = 69.98$ rad/s. We note in passing that the divergence point exists at 253 m/s, but is not shown in Figure 2 as extending the contour down to exactly $\chi = 0$ as this results in numerical errors with the aerodynamic coefficents. Table 1 presents a comparison of our results with those of published literature. The agreement is very good, and the small scatter that is observed can be explained by the fact that most sources use approximate solution methods, as well as slightly different structural parameters – compare, for instance, [38,40,41]. Note also the good agreement of this flutter point with the

rigourous result of Balakrishnan [21], which is however presented only in a figure and not given in exact value.

Figure 2 also includes the modal frequencies of the Goland wing as a function of airspeed, over $U \in [50, 450]$ m/s. These modal frequencies were computed by solving Eq. (24) for $\chi \in \mathbb{C}$ for each given U. This was done with by evaluating $\det(\mathbf{D})$ over a grid of $\mathrm{Re}(\chi)$ and $\mathrm{Im}(\chi)$ values, and interpolating to its roots – a process which is computationally expensive but convenient to implement, and represents the contour plot of Liska and Dowell [17]. We can thus compare these frequencies to the lines of $\mathrm{Re}(\det(\mathbf{D})) = 0$ on the contour plot to assess the correspondence between the two. Figure 2 shows that there is indeed a strong relationship. Over the simulated airspeed range, the real contours approximate the third, fourth and fifth modes very well. The first and second modes are well-approximated up to the point were the real contour curves back on itself: these modes then break away from the contours, with the second mode rejoining another contour which rises up from the x-axis. The curve-back behaviour of the real contours is very interesting, and would appear to play an important part in determining the location of the flutter points [19]. The exact explanation for the discrepancy between the real contours and the modal frequencies in some locations is uncertain, but it is clear that the local modal damping plays a significant part. A more extensive discussion may be found in Pons [8].

7.3 Flutter points of an artificial higher-order system

The Goland wing has been solved via the LTM before: the system of Eq. (2) is sufficiently small and simple that the system may be manipulated by hand (just as we have done in the examples). However, a major strength of the Generalised LTM is its applicability to larger and higher-order systems. For this reason, we validate our method against our own higher-order test system. The system is

$$2\mathbf{I}_n\mathbf{h}^{(8)} + 10\mathbf{I}_n\mathbf{h}^{(6)} + 10(\mathbf{I}_n + \mathbf{1}_n)\mathbf{h}^{(4)} + 50(\mathbf{I}_n + \mathbf{1}_n)\mathbf{h}^{(3)} + \mathbf{A}(\chi, U)\mathbf{h} = \mathbf{0}, \qquad (32)$$

where

$$\begin{aligned}\mathbf{A}(\chi, U) = ((1+\imath)\chi^2 + 20\chi + 100 + 20\imath)\mathbf{I}_n \\ -\big((1+\imath)U^2 - 40U + 400\big)(\mathbf{I}_n + \mathbf{1}_n).\end{aligned} \qquad (33)$$

with boundary conditions

$$\mathbf{h}^{(i)}\big|_{x=0} = \mathbf{h}^{(i)}\big|_{x=L} = 0, \qquad i \in [0,3]. \qquad (34)$$

The eigenvalue χ is defined such that $\breve{\boldsymbol{\phi}}(x,t) = \boldsymbol{\phi}(x)e^{i\chi t}$ for some dimensionless time variable t. We treat all parameters in this system as being dimensionless. The elements of tensor $\mathbf{T}$ in the third index are the coefficients of Eq. (33), and the boundary condition matrix is:

$$\mathbf{R} = \begin{bmatrix} \mathbf{I}_4 & \mathbf{0}_4 & & & & \\ & & \mathbf{I}_4 & \mathbf{0}_4 & & \\ & & & & \ddots & \ddots \\ & & & & \mathbf{I}_4 & \mathbf{0}_4 \end{bmatrix} \Bigg\} 2n \text{ blocks.} \qquad (35)$$

The model thus has two parameters: the matrix size n and the domain length L. In our simulation we take $n = 10$ and $L = 4$. The tensor $\mathbf{T}$ is thus of size $10 \times 10 \times 9$,

and the cumulative differential order of the system is 80. This system is much larger than any continuous eigenvalue problems used in aeroelasticity to date, but it should be noted that the Generalised LTM can handle systems even more complex. For example, only the values at T_{ij1} are dependent on χ and U (the higher-order terms are constant), and in Eq. (35) the boundary conditions are independent of χ and U. The ability of the Generalised LTM to solve systems which are more strongly dependent on χ and U may become useful when it is applied to instability problems in other fields (e.g. micro- or nano-mechanics).

Figure 4 shows a contour plot for our system over $\chi \in [10, 50]$ and $U \in [0, 10]$. The single flutter point can be located at approximately $\chi = 19.2$ and $U = 5.95$. It is highlighted with a circle. To validate this solution, we compute the flutter point by performing a p-method analysis with a discretised system. We first linearise the system to the form $\mathbf{q}' = \mathbf{M}\mathbf{q}$, where $\mathbf{q} = \left[\mathbf{h};\ \mathbf{h}^{(1)};\ \ldots\ ;\ \mathbf{h}^{(7)}\right]$, and $\mathbf{M}$ is a block matrix which may be derived from Eq. (32). We solve this first-order system using MATLAB's built-in boundary-value problem solver (`bvp4c`), with U defined a priori and χ as a eigenvalue parameter. To constrain the eigenproblem we use the normalisation condition $\mathrm{h}_1^{(5)} = 1$. Figure 5 shows $\mathrm{Im}(\chi)$ and $\mathrm{Re}(\chi)$ from this solution method over $U \in [5, 6.5]$. Flutter occurs when the modal damping, $\mathrm{Im}(\chi)$, becomes zero. The circles represent the computed modal frequency points, and the solid line a piecewise linear spline. The flutter point can be located at approximately $\chi = 19.2$ and $U = 5.95$. This agrees very well with the results of the Generalised LTM.

8 EXTENSIONS TO WIDER CLASSES OF SYSTEM

The major limitation of the LTM (generalised or not) is its exclusion of systems with spatially-dependent coefficients. This is something of a fundamental limitation, as the Laplace transform itself cannot deal with many classes of variable-coefficient ODE – and indeed many of these not solvable by any analytical method. Nevertheless it should be emphasised that techniques for solving types of spatially-dependent systems with our LTM framework do exist. Systems in the form of the Cauchy-Euler equation, a restrictive class of polynomial system, may already be solved exactly as they can be analytically reduced into constant-coefficient systems [42]. With a small level of approximation (relative to full discretisation), a wide class of systems formulated as variational problems may be reduced into constant-coefficient ODEs via the Kantorovich / Partial Ritz method [43,44]. This approach is particularly useful in approximating chordwise deformation in a wing plate model. There are also avenues by which the method may be extended to wider classes of system. Most significantly, the introduction of exponentials (e.g. in a Fourier series) into the coefficients results in a linear functional difference equation in the spatial state-space domain. Preliminary work indicates that variable-coefficient systems of this form are still solvable exactly – even for a significant number of exponential terms – though whether the LTM is the best framework to do this in is unclear. This is a very promising area for future research.

9 CONCLUSION

In this paper we have presented a generalisation of the Laplace Transform Method for determining the flutter points of a linear ordinary-differential aeroelastic system. That the generalised method can handle systems of arbitrary size with constant-coefficient boundary conditions of arbitary complexity is itself a significant leap forward, and the elimination of all symbolic computations (while still remaining exact) is further progress. Though the method is only directly applicable to ordinary-differential systems with constant coefficients, the two deficiencies that we have eliminated will allow the LTM to be easily applied to a much wider range of systems than before. In parallel, we have also devised a new method for solving and visualising the algebraic flutter problem arising from the LTM procedure. This procedure is simpler and more convienient than existing exact methods (such as the p-method), and makes visualisation of flutter point locations and relationships easier. Both of these developments have the potential to pave the way for greater use of the Laplace Transform Method in aeroelasticity and in other disciplines. We have validated the Generalised LTM and the new solution method against both the Goland wing model and a large system of high differential order, as a demonstration of their effectiveness for solving such systems.

NOMENCLATURE

$(\cdot)^T$	matrix transposition
$(\cdot)', (\cdot)^{(n)}$	differentiation
$\mathrm{Re}(\cdot), \mathrm{Im}(\cdot)$	real, imaginary part
$\mathcal{L}\{\cdot\}$	Laplace transformation
$f\|_g$	evaluation of f at condition g
$\mathrm{T}_{abc\ldots}$	the element of tensor $\mathbf{T}$ located at the indices a, b, c …
$(\mathrm{F})_{abc\ldots}$	arrangement of scalar function $\mathrm{F}(a, b, c \ldots)$ into tensor $\mathbf{T}$: $\mathrm{T}_{abc\ldots} = \mathrm{F}(a, b, c \ldots)$
$[\cdot]_{\mathrm{IVR}}$	Iverson brackets
$\mathbf{0}_n, \mathbf{0}_{n\times m}$	zero matrix of size $n \times n$, $n \times m$
$\mathbf{I}_n$	identity matrix of size $n \times n$
$\mathbf{1}_n, \mathbf{1}_{n\times m}$	matrix with all entries $= 1$ of size $n \times n$, $n \times m$
$\imath$	imaginary unit

REFERENCES

[1] Hodges, D. H., and Pierce, G. A., 2014, Introduction to Structural Dynamics and Aeroelasticity, 2nd ed., Cambridge University Press, Cambridge, UK. ISBN: 978-1-1076-1709-4

[2] Lind, R., 2002, "Match-Point Solutions for Robust Flutter Analysis," Journal of Aircraft, **39**(1), pp. 91–99. DOI: 10.2514/2.2900

[3] Borglund, D., 2004, "The μ-k method for robust flutter solutions," Journal of Aircraft, **41**(5), pp. 1209–1216. DOI: 10.2514/1.3062

[4] Borglund, D., 2003, "Robust aeroelastic stability analysis considering frequency-domain aerodynamic uncertainty," Journal of Aircraft, **40**(1), pp. 189–193. DOI: 10.2514/2.3074

[5] Haddadpour, H., and Firouz-Abadi, R. D., 2009, "True damping and frequency prediction for aeroelastic systems: The PP method," Journal of Fluids and Structures, **25**(7), pp. 1177–1188. DOI: 10.1016/j.jfluidstructs.2009.06.006

[6] Afolabi, D., Pidaparti, R. M. V., and Yang, H. T. Y., 1998, "Flutter Prediction Using an Eigenvector Orientation Approach," AIAA Journal, **36**(1), pp. 69–74. DOI: 10.2514/2.353

[7] Irani, S., and Sazesh, S., 2013, "A new flutter speed analysis method using stochastic approach," Journal of Fluids and Structures, **40**, pp. 105–114. DOI: 10.1016/j.jfluidstructs.2013.03.018

[8] Pons, A., 2015, "Aeroelastic flutter as a multiparameter eigenvalue problem," Master's Thesis, Univ. of Canterbury, Christchurch, New Zealand.

[9] Bisplinghoff, R. L., Ashley, H., and Halfman, R. L., 1957, Aeroelasticity, Addison-Wesley, Reading, MA.

[10] Dowell, E. H., Cox, D., Curtiss, Jr., H. C., Edwards, J. W., Hall, K. C., Peters, D. A., Scanlan, R. H., Simiu, E., Sisto, F., and Strganac, T. W., 2004, A Modern Course in Aeroelasticity, Springer Netherlands, Dordrecht, The Netherlands. ISBN: 978-1-4020-2711-6

[11] Librescu, L., and Song, O., 2006, Thin-Walled Composite Beams, Springer Netherlands, Dordrecht, The Netherlands. ISBN: 978-1-4020-3457-2

[12] Yates, J. E., 1970, A study of Panel Flutter with the Exact Method of Zeydel, NASA, Washington, DC. DOI: 10.2514/3.48463

[13] Goland, M., 1945, "The Flutter of a Uniform Cantilever Wing," Journal of Applied Mechanics, **12**(4), pp. A197–A208.

[14] Runyan, H. L., and Watkins, C. E., 1950, Flutter of a Uniform Wing with an Arbitrarily Placed Mass According to a Differential-Equation Analysis and a Comparison with Experiment, NACA Report No. 966, Washington, DC.

[15] Graham, G. M., and Jenkins, J. E., 1997, "Aeroelasticity of an Airfoil Test Rig," Journal of Fluids and Structures, **11**(5), pp. 485–506. DOI: 10.1006/jfls.1997.0094

[16] Librescu, L., and Thangjitham, S., 1991, "Analytical Studies on Static Aeroelastic Behavior of Forward-Swept Composite Wing Structures," Journal of Aircraft, **28**(2), pp. 151–157. DOI: 10.2514/3.46004

[17] Liska, S., and Dowell, E. H., 2009, "Continuum Aeroelastic Model for a Folding-Wing Configuration," AIAA Journal, **47**(10), pp. 2350–2358. DOI: 10.2514/1.40475

[18] Pons, A., and Gutschmidt, S., 2014, "Lower-wing structural dynamics and flutter in V-strut biplanes," 17th U.S. National Congress on Theoretical & Applied Mechanics, East Lansing, USA.

[19] Pons, A., and Gutschmidt, S., 2014, "Lower-Wing Flutter in Biplanes," 8th European Nonlinear Dynamics Conference, Vienna, Austria.

[20] Chonan, S., 1986, “Steady state response of an axially moving strip subjected to a stationary lateral load,” Journal of Sound and Vibration, **107**(1), pp. 155–165. DOI: 10.1016/0022-460X(86)90290-7

[21] Balakrishnan, A. V., 2012, Aeroelasticity: The Continuum Theory, Springer Netherlands, Dordrecht, The Netherlands. ISBN: 978-1-4614-3608-9

[22] Karpouzian, G., and Librescu, L., 1996, “Nonclassical Effects on Divergence and Flutter of Anisotropic Swept Aircraft Wings,” AIAA Journal, **34**(4), pp. 786–794. DOI: 10.2514/3.13141

[23] Qin, Z., and Librescu, L., 2003, “Aeroelastic instability of aircraft wings modelled as anisotropic composite thin-walled beams in incompressible flow,” Journal of Fluids and Structures, **18**(1), pp. 43–61. DOI: 10.1016/S0889-9746(03)00082-3

[24] Theodorsen, T., 1935, General theory of aerodynamic instability and the mechanism of flutter, NACA Report No. 496, Washington, DC.

[25] Peters, D. A., Karunamoorthy, S., and Cao, W.-M., 1995, “Finite state induced flow models. I - Two-dimensional thin airfoil,” Journal of Aircraft, **32**(2), pp. 313–322. DOI: 10.2514/3.46718

[26] Graham, R. L., Knuth, D. E., and Patashnik, O., 1994, Concrete Mathematics, 2nd ed., Addison-Wesley, Reading, MA. ISBN: 978-0-2015-5802-9

[27] Tang, K. T., 2006, Mathematical Methods for Engineers and Scientists, **2**, Springer Netherlands, Dordrecht, The Netherlands. ISBN: 978-3-540-30268-1

[28] Aurentz, J. L., Vandebril, R., and Watkins, D. S., 2013, “Fast Computation of the Zeros of a Polynomial via Factorization of the Companion Matrix,” SIAM Journal on Scientific Computing, **35**(1), pp. A255–A269. DOI: 10.1137/120865392

[29] Holzel, M. S., and Bernstein, D. S., 2011, “SVD-based computation of zeros of polynomial matrices,” Proceedings of the 50th IEEE Conference on Decision and

Control and European Control Conference, Orlando, FL, pp. 6962–6966. ISBN: 978-1-6128-4800-6

[30] Berhanu, M., 2005, "The Polynomial Eigenvalue Problem," Doctoral Dissertation, Univ. of Manchester, Manchester, UK.

[31] Magnus, J. R., 1999, Matrix Differential Calculus with Applications in Statistics and Econometrics, 2nd ed., John Wiley & Sons, Chichester, UK. ISBN: 978-0-471-98633-1

[32] Duffy, D. G., 2010, Advanced Engineering Mathematics with MATLAB, 3rd ed., CRC Press, Boca Raton, FL. ISBN: 978-1-4398-1624-0

[33] Luke, Y. L., and Dengler, M. A., 1951, "Tables of the Theodorsen Circulation Function for Generalized Motion," Readers' Forum, Journal of the Aeronautical Sciences, **18**(7), pp. 478–483. DOI: 10.2514/8.2006

[34] Laitone, E. V., 1952, "Theodorsen's Circulation Function for Generalized Motion," Readers' Forum, Journal of the Aeronautical Sciences, **19**(3), pp. 211–213. DOI: 10.2514/8.2213

[35] van de Vooren, A. I., 1952, "Generalization of the Theodorsen Function to Stable Oscillations," Readers' Forum, Journal of the Aeronautical Sciences, **19**(3), pp. 209–211. DOI: 10.2514/8.2212

[36] Mehrmann, V., and Voss, H., 2004, "Nonlinear Eigenvalue Problems: A Challenge for Modern Eigenvalue Methods," GAMM-Mitteilungen, **27**(2), pp. 121–152. DOI: 10.1002/gamm.201490007

[37] Voss, H., 2013, "Nonlinear Eigenvalue Problems," Handbook of Linear Algebra, L. Hogben, ed., Chapman & Hall/CRC, FL. ISBN: 978-1-4665-0728-9

[38] Wang, I., 2011, "Component Modal Analysis of a Folding Wing," Master's Thesis, Duke Univ., Durham, NC.

[39] Borello, F., Cestino, E., and Frulla, G., 2010, "Structural Uncertainty Effect on Classical Wing Flutter Characteristics," Journal of Aerospace Engineering, **23**(4), pp. 327–338. DOI: 10.1061/(ASCE)AS.1943-5525.0000049

[40] Wang, Y., Wynn, A., and Palacios, R., 2013, "Robust Aeroelastic Control of Very Flexible Wings using Intrinsic Models," 54th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Boston, MA.

[41] Palacios, R., and Epureanu, B., 2011, "An Intrinsic Description of the Nonlinear Aeroelasticity of Very Flexible Wings," 52nd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, Denver, CO.

[42] Polyanin, A. D., and Manzhirov, A. V., 2007, Handbook of mathematics for engineers and scientists, Chapman & Hall/CRC, Boca Raton, FL. ISBN: 978-1-58488-502-3

[43] Jensen, D. W., and Crawley, E. F., 1984, "Frequency Determination Techniques for Cantilevered Plates with Bending-Torsion Coupling," AIAA Journal, **22**(3), pp. 415–420. DOI: 10.2514/3.48463

[44] Reissner, E., and Stein, M., 1951, Torsion and transverse bending of cantilever plates, NACA Technical Note No. 2369, Washington, DC.

[45] Marzocca, P., Librescu, L., and Silva, W. A., 2002, "Aeroelastic response and flutter of swept aircraft wings," AIAA Journal, **40**(5), pp. 801–812. DOI: 10.2514/2.1724

[46] Goland, M., and Luke, Y. L., 1948, "The Flutter of a Uniform Wing with Tip Weights," Journal of Applied Mechanics, **15**(1), pp. 13–20.

Figure Captions List

Fig. 1 det(**D**) against χ for the Goland wing, *in vacuo*

Fig. 2 Contour plot of the Goland wing system, with modal frequency paths superimposed

Fig. 3 Modal damping paths of the Goland wing

Fig. 4 Contour plot of the higher-order test system.

Fig. 5 Modal frequency and damping paths of the higher-order test system in the vicinity of the flutter point

Table Caption List

Table 1 Flutter point results for the Goland wing

FIGURES

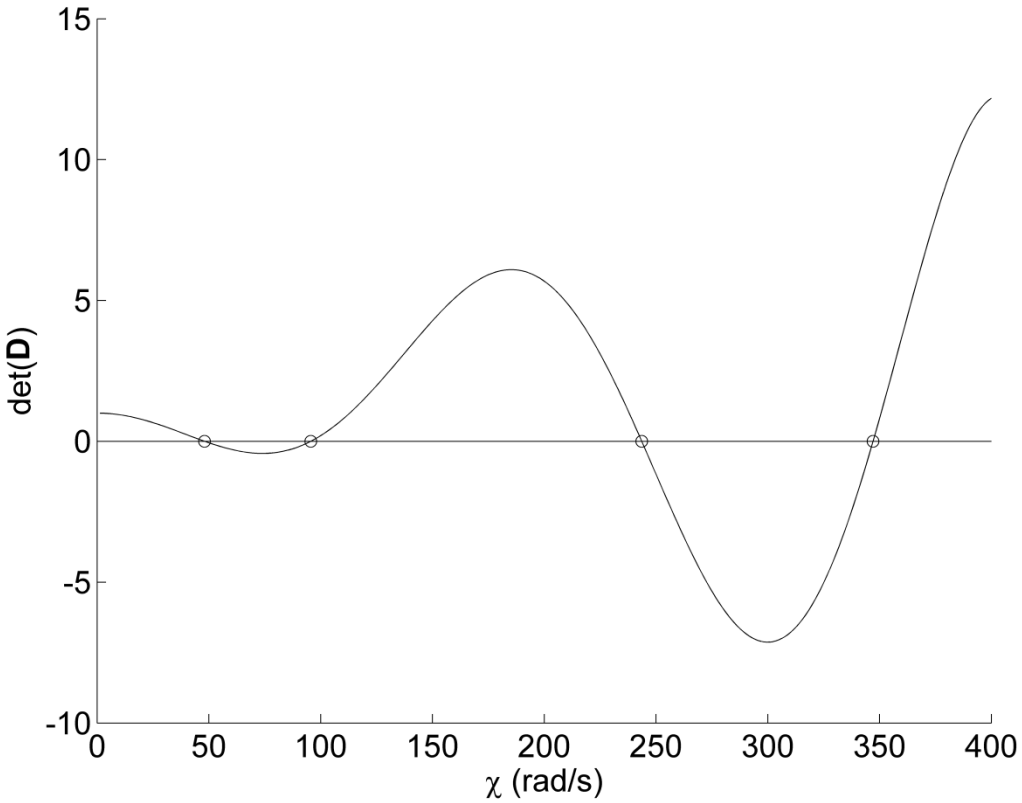

Figure 1

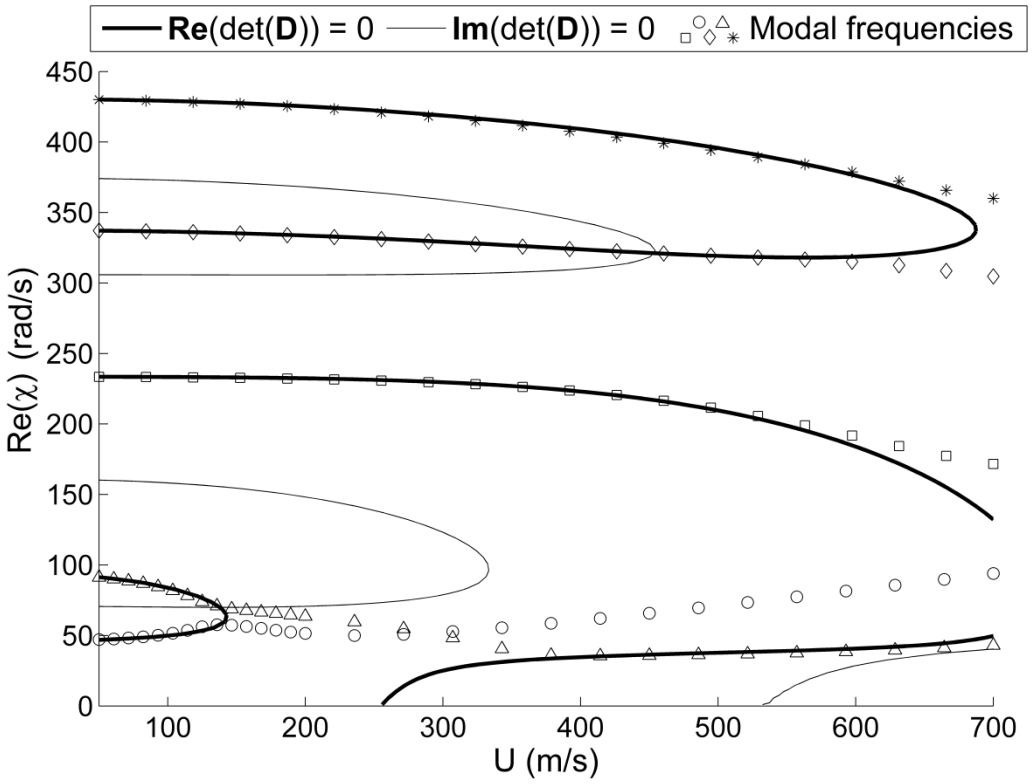
Re(det(D)) = 0
Im(det(D)) = 0
Modal frequencies
Re(χ) (rad/s)
U (m/s)
0
50
100
150
200
250
300
350
400
450
100
200
300
400
500
600
700

Figure 2

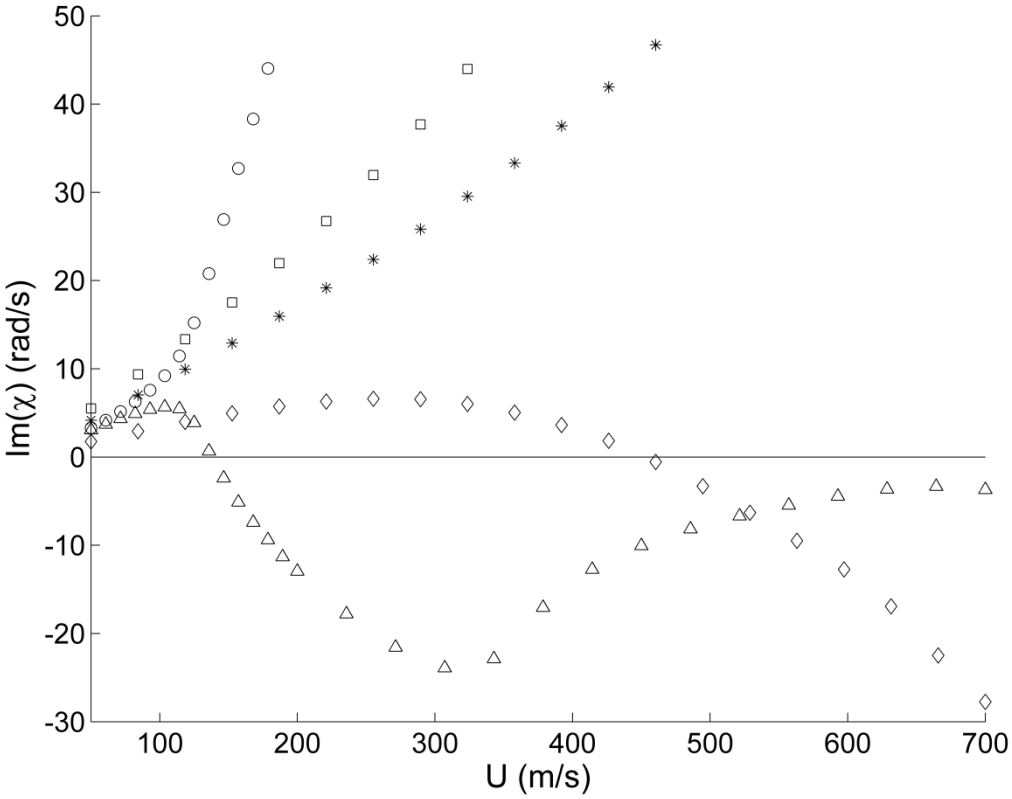

Figure 3

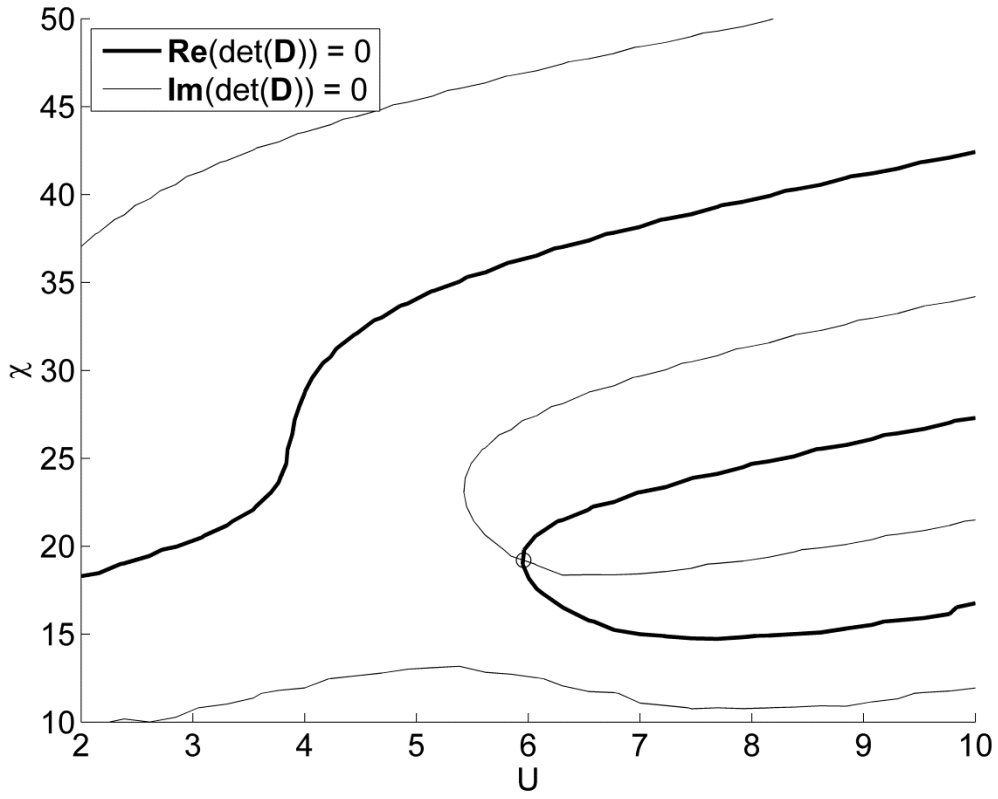

Figure 4

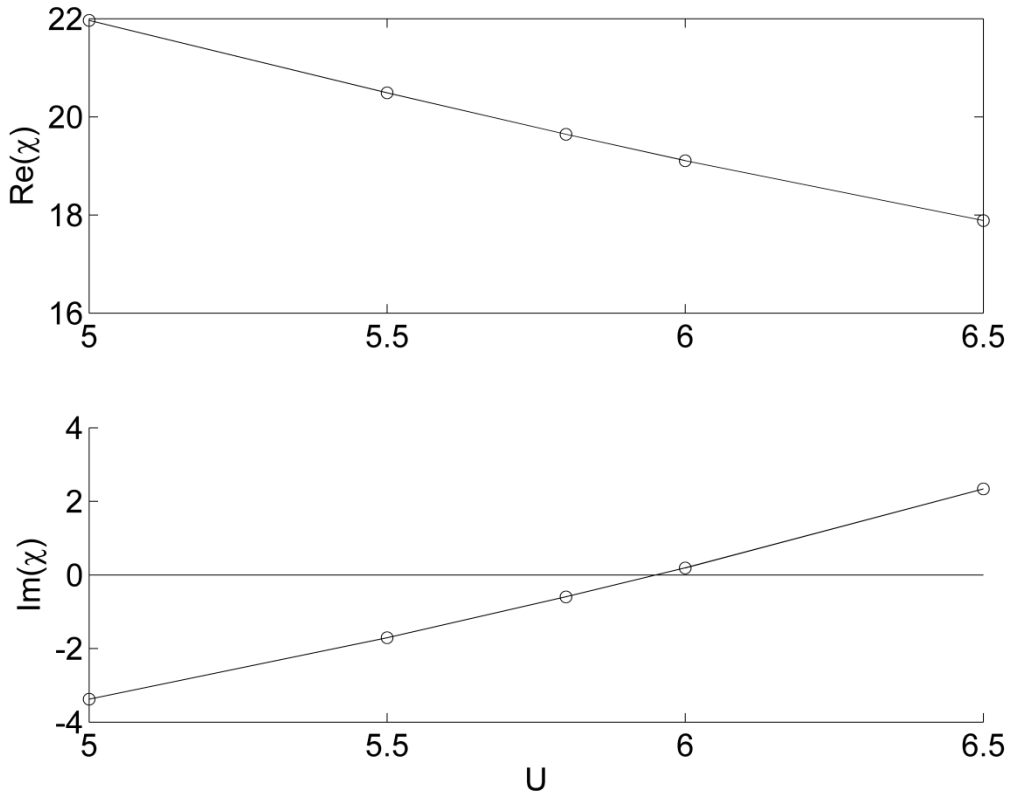

Figure 5

TABLES

Table 1

U (m/s)	χ (m/s)	**Reference**
138.11	69.98	Present work
139	70.7	Wang [38] (Source)
139	70.0	Wang et al. [40]
141	69.8	Palacios and Epureanu [41]
136.11	69.12	Marzocca et al. [45]
137.50	70.37	Goland and Luke [46]