A Symbolic Derivation of the Equations of Motion for Rotor Blade Dynamic Stability Finite Element Analysis using *Mathematica*

Prof. Giuseppe Surace, Eng. Lorenzo Cardascia

Department of Aeronautical and Space Engineering
Politecnico di Torino - Italy

Eng. Viorel Anghel

Strength of Materials Department
University POLITEHNICA Bucharest-Romania

Abstract

The equations of motion of a hovering hingeless helicopter rotor blade are obtained using Hamilton's principle. Finite element analysis is used to discretise their spatial dependence. An element with two nodes and five degrees of freedom per node is developed in order to analyse the blade's flap-lag-torsion stability. All elements of inertia, damping and stiffness matrices and its load vector are obtained explicitly, in analytical form from the expressions for strain energy, kinetic energy and virtual work. The nonlinear steady state trim position in given flight conditions is solved iteratively starting from the linear solution. The aerodynamic loads in hover are considered to be quasi-steady and two-dimensional. The use of an ordering scheme makes it possible to retain all the important terms in the equations. High order term reduction and element matrix selection is accomplished automatically by routines written in *Mathematica*. Some numerical examples using MATLAB concerning natural frequencies of rotating beams, steady-state position and flutter analysis of a hingeless blade are presented and compared with the results of other studies.

Notation

a = *lift curve slope*
c = *blade chord*
C_{d_0} = *blade − section profile medium drag coefficient*
$C_{M_{ac}}$ = *blade − section pitching moment coefficient*
e_A = *tension center offset from elastic axis, positive forward*
e_d = *aerodynamic center offset from elastic axis, positive aft*
e_g = *center of mass offset from elastic axis, positive forward*
e_1 = *blade root offset*
EA = *blade axial stiffness*
F = *centrifugal force*
GJ = *blade elastic torsion stiffness*
I_b = *blade mass moment of inertia about flap axis at the root*
I_y, I_z = *blade cross section moments of inertia in the flap and lead − lag directions respectively*
k_A = *polar radius of gyration of blade cross section*
k_m = *mass radius of gyration of blade cross section*
k_{m1}, k_{m2} = *principal mass radii of gyration of blade cross section*
l_i = *length of ith element*
L = *blade length*
L_i = *distance between the rotor center and the beginning of the ith element blade*
m = *blade mass per unit length*

n	= *number of elements*
q_i	= *ith degree of freedom of the element or blade*
R	= *rotor radius*
u, v, w	= *elastic displacements in the x, y, z directions, respectively*
v_i	= *induced inflow*
V	= *blade section resultant air velocity*
x, y, z	= *undeformed blade coordinate*
x_i	= *local axial coordinate of the ith element*
β_p	= *blade precone angle*
γ	= *Lock number* , $\rho a c R^4 / I_b$
$\delta T, \delta U$	= *variation of kinetic and strain energies, respectively*
δW	= *virtual work done to the aerodynamic loads*
ε	= *nondimensional order of magnitude parameter*
ρ	= *air density*
θ_B	= *blade pretwist*
θ_c	= *control pitch angle of the blade*
ξ, η, ζ	= *deformed blade coordinate*
σ	= *solidity ratio*
ϕ	= *elastic torsion about elastic axis*
ω_v	= *fundamental coupled rotating lead – lag natural frequencies*
ω_w	= *fundamental coupled rotating flap natural frequencies*
ω_ϕ	= *fundamental coupled rotating torsion natural frequencies*
Ω	= *rotor blade angular velocity*
$(.)'$	$= \partial / \partial x$
$(.)''$	$= \partial^2 / \partial x^2$
$(.)_c$	= *circulatory aerodynamic term*
$(.)_{nc}$	= *noncirculatory aerodynamic term*
$(.)_A$	= *aerodynamic matrix*
$(.)_L$	= *linear matrix*
$(.)_{NL}$	= *nonlinear matrix*
$(.)_{SL}$	= *structural linear matrix*
$(.)^T$	= *transpose of a matrix*

Introduction

In the field of rotary-wing aeroelasticity, the problems involved in the study of nonlinear systems with constant or periodic coefficients are highly complex. Determining and solving rotating blade equations of motion is a difficult task, for which the computerised symbolic manipulation has shown itself to be a powerful analysis tool. Many symbolic codes are now available. General-purpose symbolic processors such as REDUCE and MACSYMA have been used to obtain the equations of motion of the helicopter blade [1], [2]. The latter paper also presents a special-purpose symbolic processor, DEHIM (Dynamic Equations for Helicopter Interpretative Models). A classic approach to flexible blade aeroelastic stability analysis is the Galerkin method [3], [4]. In recent years, the finite element method has been widely used because of its potential for solving complex nonuniform blade configurations which occur in modern hingeless or bearingless rotors. Both, the Galerkin-type finite element [5] and displacement finite element methods [6] are suitable procedures for rotor blade modeling. The finite element formulation has now reached a high level of sophistication [7] and has been adopted in complex helicopter dynamic analysis codes such as GRASP [8], UMARC [9], 2GCHAS [7]. The aim of the present study is to show the potential of *Mathematica* software in solving certain rotary-wing aeroelastic problems by applying MATLAB for numerical calculus. A finite element for blade discretisation with ten nodal degrees of freedom was developed on the basis of Hamilton's principle; this element was first presented in [6]. The exact symbolic relations for each element of the mass matrix, stiffness matrix and damping matrix were obtained from the expressions for kinetic energy, potential energy and virtual work. The numerical examples concern natural frequencies of the rotating blade, the nonlinear steady-state position of the blade in hover and flutter analysis under these flight conditions. Quasi-steady strip theory is used in evaluating aerodynamic loads.

Equations of motion

The formulation follows the steps presented in [6] and uses results published by Hodges and Dowell [10] and Hodges and Ormiston [3]. The rotor blade is a single load path isotropic beam

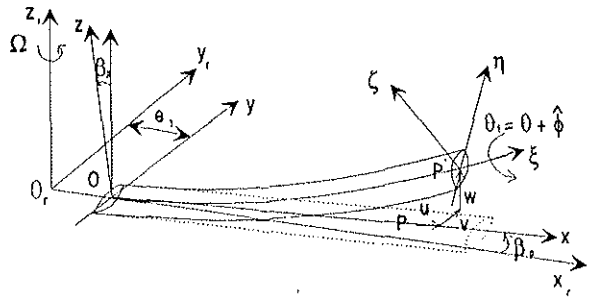

Fig. 1 Blade undeformed and deformed systems of coordinates

rotating with constant angular velocity Ω. A root offset e_1 and a precone angle β_p are also considered. The undeformed xyz coordinate system is shown in Fig. 1. A point P on the undeformed elastic axis occupies position P' after deformation (axial, flap-bending and lag-bending displacements). The cross section is then rotating with the angle θ_1 about the deformed elastic axis. The second coordinate system shown in Fig. 1 is obtained using the transformation:

$$\left\{ \begin{array}{c} i_\xi \\ i_\eta \\ i_\zeta \end{array} \right\} = [T] \left\{ \begin{array}{c} i_x \\ i_y \\ i_z \end{array} \right\} \qquad (1)$$

where $[T]$ is a transformation matrix given in Appendix A. In this matrix:

$$\begin{aligned} \theta_1 &= \theta_B + \theta_c + \hat{\phi} = \theta + \hat{\phi} \\ \hat{\phi} &= \phi - \int_0^x v'' w' dx \end{aligned} \qquad (2)$$

The equations of motion are derived using Hamilton's principle:

$$\int_{t_1}^{t_2} (\delta U - \delta T - \delta W) dt = 0 \qquad (3)$$

The expressions of variations of strain energy and kinetic energy are presented in [6], [10]. Virtual work of aerodynamic forces is given by:

$$\begin{aligned} \delta W &= \int_0^L (L_u \delta u + L_v \delta v + L_w \delta w \\ &+ L_w \delta w + M_\phi (\delta \hat{\phi} + w' \delta v')) dx \end{aligned} \qquad (4)$$

where L_u, L_v, L_w, M_ϕ are the external aerodynamic loads distributed along the length of

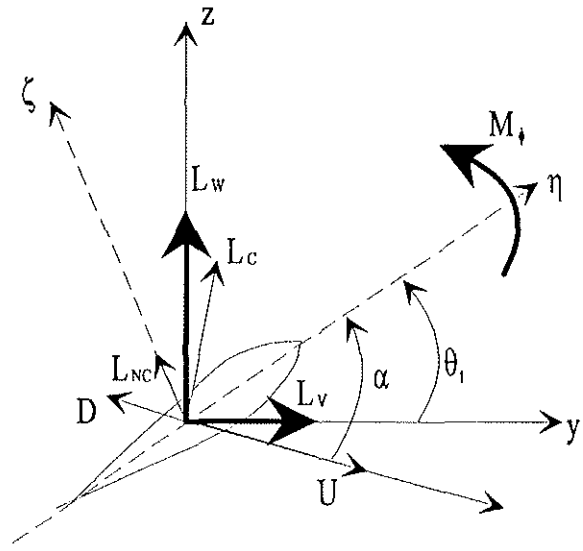

Fig. 2 The components of aerodynamic loading

the blade in axial, lag, flap and torsion directions. Equation (3) represents four equations of motion regarding δu, δv, δw and $\delta \hat{\phi}$. The centrifugal force:

$$F(x) = \int_x^L m(x)[\Omega^2(x + e_1) + 2\Omega \dot{v}] dx \qquad (5)$$

can also be written as:

$$\begin{aligned} F(x) &= EA[u' + \frac{1}{2}v'^2 + \frac{1}{2}w'^2 + {k_A}^2 \theta' \phi' \\ &- e_a(v'' cos\theta_1 + w'' sin\theta_1)] \end{aligned} \qquad (6)$$

In this case, the displacement u is eliminated from the expressions of δU, δT, δW. An ordering scheme is used in obtaining these expressions, the assumed orders of magnitude of each parameter being the same as used in [3]. Only θ was considered of order ε^0 for all structural and aerodynamics terms (ε is a small nondimensional parameter). All terms of order ε^2 are retained, as are some terms of order ε^3 [3], [6]. The aerodynamic loads in hover are based on Greenberg's model, using a quasi-steady approximation. Noncirculatory terms are also included. The force components and directions are shown in Fig. 2.

The complete relations for aerodynamic terms are given in [6], [10]. A special procedure was written in *Mathematica* in order to evaluate the order of magnitude of each term

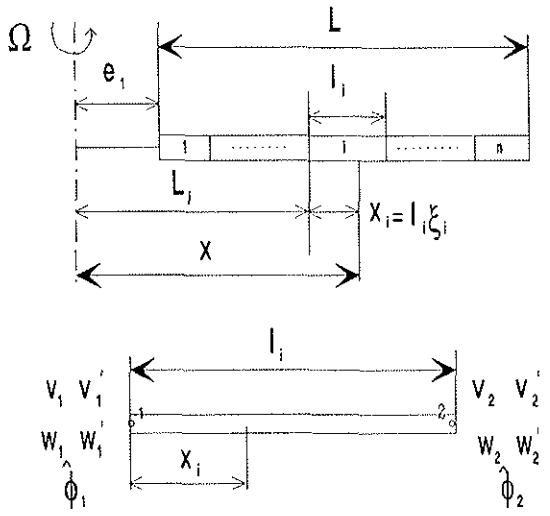

Fig. 3 *Finite element discretisation of the blade*

of a complex polynomial expression and to cancel all high terms which are bigger than the set limit. This routine is given in Appendix B. The constant inflow $\lambda_i(v_i)$ may be written as:

$$\lambda_i = 1.15\sqrt{C_T/2},\ v_i = \Omega R\lambda_i \qquad (7)$$

C_T being the thrust coefficient. The effective pitch of the blade at $0.75R$ is:

$$\theta_{0.75R} = 6C_T/\sigma a + 1.5\lambda_i \qquad (8)$$

which includes the torsion deformation.

Finite element formulation

The blade is divided into beam elements, each consisting of two nodes with five degrees of freedom: v, v', w, w' and $\hat{\phi}$ (Fig. 3). Hamilton's principle is discretised as:

$$\sum_{i=1}^{n} \Delta_i = \sum_{i=1}^{n} (\delta U_i - \delta T_i - \delta W_i) = 0 \qquad (9)$$

where the subscript i denotes the contributions of the ith element. The vector of element degrees of freedom $\{q_i\}$ is defined as:

$$[q_1, ..., q_{10}]^T = [v_1, v_1', w_1, w_1', \hat{\phi}_1, ..., \hat{\phi}_2]^T \qquad (10)$$

The distributions of the deflections v, w, $\hat{\phi}$ over an element are of the form:

$$\begin{aligned} v &= H_1 v_1 + H_2 v_1' + H_3 v_2 + H_4 v_2' \\ w &= H_1 w_1 + H_2 w_1' + H_3 w_2 + H_4 w_2' \\ \hat{\phi} &= H_5 \hat{\phi}_1 + H_6 \hat{\phi}_2 \end{aligned} \qquad (11)$$

where the shape functions are:

$$\begin{aligned} H_1(\xi_i) &= 2\xi_i^3 - 3\xi_i^2 + 1 \\ H_2(\xi_i) &= l_i(\xi_i^3 - 2\xi_i^2 + \xi_i) \\ H_3(\xi_i) &= -2\xi_i^3 + 3\xi_i^2 \\ H_4(\xi_i) &= l_i(\xi_i^3 - \xi_i^2) \\ H_5(\xi_i) &= 1 - \xi_i \\ H_6(\xi_i) &= \xi_i \end{aligned} \qquad (12)$$

and $\xi_i = x_i/l_i$. Virtual displacements δv, δw, $\delta\hat{\phi}$ are similar to equation (11), and are assumed to be of the same forms. With these relations, the contribution of the ith element is of the form:

$$\begin{aligned} \Delta_i &= \{\delta q_i\}^T ([M_i(q_i)]\{\ddot{q}_i\} + [C_i(q_i)]\{\dot{q}_i\} \\ &+ [K_i(q_i)]\{q_i\} + \{Q_i\}) \end{aligned} \qquad (13)$$

which highlighs the element inertia, damping and stiffness matrices respectively and the element load vector. These element matrices were obtained in *Mathematica* by performing an integration over the length of ith element in (9) and a systematic selection of the coefficients containing $\ddot{q}_i$, $\dot{q}_i$, q_i etc. The blade properties were assumed to be constant over each element. The steady centrifugal force in a given section of the ith element was written as:

$$F_i(\xi_i) = \Omega^2(a_i\xi_i^2 + b_i\xi_i + c_i) \qquad (14)$$

where a_i, b_i, c_i are functions of the position of the element:

$$\begin{aligned} a_i &= -\frac{m_i l_i^2}{2} \\ b_i &= m_i l_i L_i \\ c_i &= \sum_{k=1}^{n} m_k (L_{k+1}^2 - L_k^2) \end{aligned} \qquad (15)$$

The element matrices depend on the $\{q_i\}$ vector because of the retention of the nonlinear terms. In the Appendix C is illustrated the algorithm implemented in *Mathematica* for determining the linear parts of the element matrices and the Appendix D presents the symmetric stiffness matrix in this case. The global matrices of the blade are obtained by the assembly of the element matrices, and the global

load vector $\{Q\}$ is formed using the element load vectors $\{Q_i\}$. The equation (3) becomes

$$[M(q)]\{\ddot{q}\} + [C(q)]\{\dot{q}\} + [K(q)]\{q\} = \{Q\} \tag{16}$$

where q contains the $5(n+1)$ global degrees of freedom. The matrix $[M(q)]$ is banded and contains contribution of the inertial and aerodynamics terms. The $[C]$ matrix is treated in a special manner writing:

$$[C] = [C]_{NL} + [C]_L + [C]_A \tag{17}$$

where $[C]_{NL}$ is an asymmetric nonbanded matrix due of the presence of the term $\int_0^{l_i}\int_0^x(\dot{v}'v' + \dot{w}'w')dxdx_i$ in the expression of the kinetic energy, $[C]_L$ represents the gyroscopic linear terms and $[C]_A$ is the aerodynamic contribution. The stiffness matrix $[K]$ is banded asymmetric containing structural, inertial and aerodynamic terms.

Numerical Results

The first example is the calculation of the natural frequencies for coupled flapwise bending-chordwise bending and torsion for a fixed-free blade having the following properties [11]: $L = 40\ in$, $\theta_B = 45^o$, $EI_y = 25000\ lb\ in^2$, $EI_z = 75000\ lb\ in^2$, $GJ = 9000\ lb\ in^2$, $m = 0.0015\ slugs/in$, $k_{m1}^2 = 1\ in^2$, $k_{m2}^2 = 1\ in^2$, $e_g = \sqrt{2}\ in$. The results at $\Omega = 0$ are shown in Table 1, where they are compared with those obtained in [11] using the Transmission Matrix Method.

Table 1 Comparison of the natural frequencies [rad/s] at $\Omega = 0$

Mode	T.M.M., [11]	This study 6 elements	This study 12 elements
1	30.8295	30.8383	30.8370
2	53.8277	53.8407	53.8404
3	184.6175	185.0116	184.7434
4	337.3333	337.4953	337.4174
5	484.3373	490.5315	485.9276

The calculation uses only the linear structural part of the [K] matrix and the linear part of [M] matrix without aerodynamic terms. The solved eigenvalue problem is:

$$[K]_{SL} - \omega^2[M]_L = \{0\} \tag{18}$$

The other numerical example concerns flutter analysis of a hingeless rotor blade. The steady-state equations are of the form:

$$[K(q_0)]\{q_0\} = \{Q_0\} \tag{19}$$

These nonlinear equations are solved iteratively starting from a linear solution. No special algorithms are used for this. For a given collective pitch angle, the inflow velocity is calculated using the relations (7), (8). The iteration process also considers the torsional deflection $\phi_{0.75R}$ changes of blade pitch. The uniform blade properties used for testing the formulation are taken from [3] and are also given in Table 1 of [6]. They are: $EI_y/m\Omega^2R^4 = 0.014486$, $EI_z/m\Omega^2R^4 = 0.166908$, $GJ/m\Omega^2R^4 = 0.000925$ (0.005661), $k_{m1}/R = 0$, $k_{m2}/R = 0.025$, $k_A/k_m = 1.5$, $c/R = \pi/40$, $\sigma = 0.1$, $\gamma = 5$, $a = 6$, $C_{d_0} = 0.0095$, $C_{M_{ac}} = 0$, $\beta_p = -0.05, 0, 0.05$ or 0.1 rad. The center of mass, aerodynamic center, tension center and elastic center are considered to coincide. Because this model works with dimensional quantities, $m = 1$, $R = 1$, $\Omega = 1$ was adopted. The dimensionless rotating frequencies obtained for these data are given in Table 2.

Table 2 Nondimensional rotating frequencies of a hingeless blade

Mode	F.E.M. [6]	This study, 6 elements	This study, 12 elements
Flap ω_w	1.15	1.15	1.15
Lag ω_v	1.50	1.50	1.50
Torsion 1 ω_ϕ	2.5	2.460	2.456
Torsion 2 ω_ϕ	5.00	4.989	4.977

Figures 4, 5 and 6 show the equilibrium deflections of a rotor blade tip having $\omega_\phi = 5$, as a function of the pitch angle θ_c for various values of β_p. They are in good agreement with these of the first calculation in [3] using the Galerkin method and the results of [6] using the same finite element formulation. The flutter motion is assumed to be a small perturbation about the equilibrium position:

$$\{q\} = \{q_0\} + \{\Delta q\} \tag{20}$$

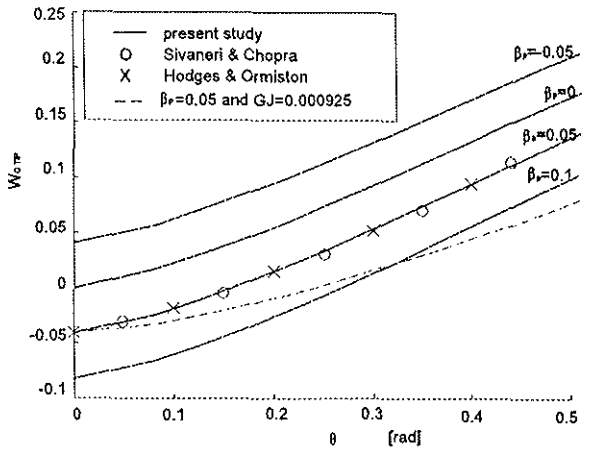

Fig. 4 Tip flap-bending equilibrium deflections for $\omega_v = 1.5$, $\omega_w = 1.15$, $\omega_\phi = 5.0$

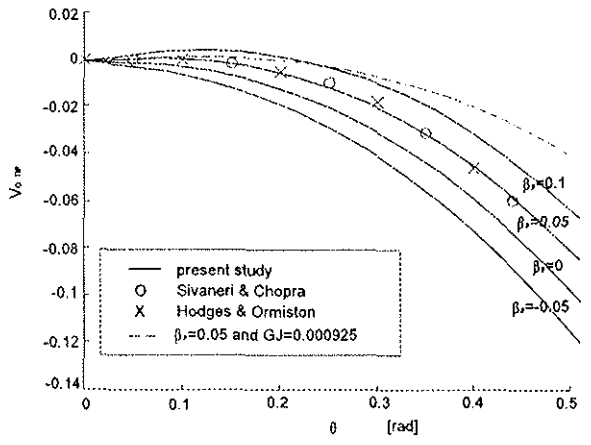

Fig. 5 Tip lag-bending equilibrium deflections for $\omega_v = 1.5$, $\omega_w = 1.15$, $\omega_\phi = 5.0$

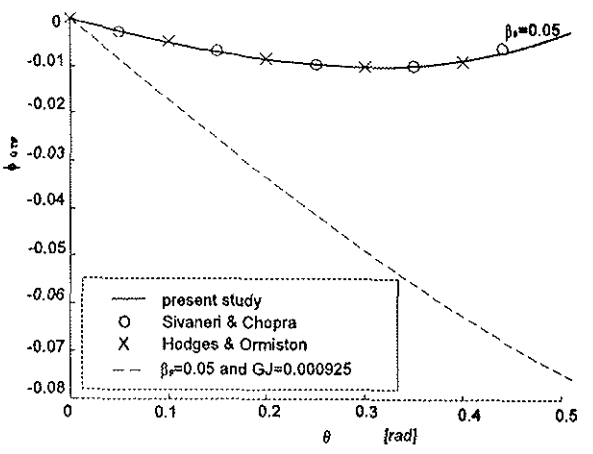

Fig. 6 Tip torsion equilibrium deflections for $\omega_v = 1.5$, $\omega_w = 1.15$, $\omega_\phi = 5.0$

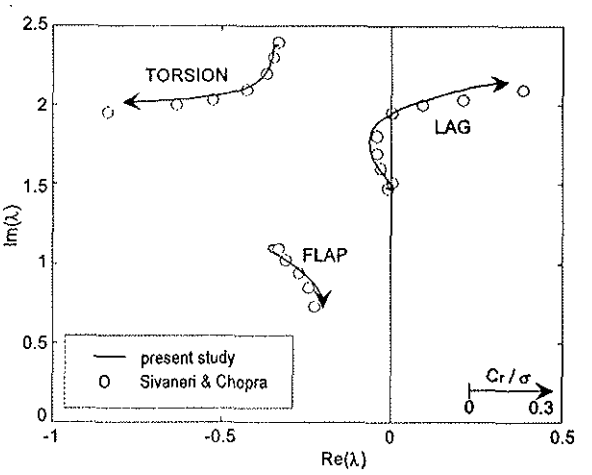

Fig. 7 Root locus plot for a hingeless blade

and the obtained flutter equations are:

$$[M(q_0)]\{\ddot{\Delta q}\}+[C(q_0]\{\dot{\Delta q}\}+[K(q_0)]\{\Delta q\} = \{0\} \quad (21)$$

First, the natural modes are obtained by solving the equation:

$$[M]\{\ddot{\Delta q}\} + [K]_S\{\Delta q\} = \{0\} \quad (22)$$

which contains only the structural and inertial terms. The corresponding eigenvalues are real, and equation (23) is then transformed in modal space. The first five eigenfrequencies and eigenvectors are used in $[\Phi]$, and performing the substitution:

$$\{\Delta q\} = [\Phi]\{p\} \quad (23)$$

in (23) and premultiplying by $[\Phi]^T$ yields the normal mode equations:

$$[M^*]\{\ddot{p}\} + [C^*]\{\dot{p}\} + [K^*]\{p\} = \{0\} \quad (24)$$

The complex eigenvalues of (24) are analysed for a given data set. Figure 7 shows the root locus plot of the first modes of flap, lag and torsion for a blade having $\omega_v = 1.5$, $\omega_w = 1.15$, $\omega_\phi = 2.5$, $\gamma = 5$ and $\beta_p = 0.05$ rad, when C_T/σ is varied from 0 to 0.3. Figure 8 shows the region of instability in the domain $\omega_v - \theta$ for a blade having the following properties: $\omega_w = 1.15$, $\omega_\phi = 2.5$, $\gamma = 5$ and $\beta_p = 0.05$ rad. The results of [3] and [6] are also shown for comparison. There is good agreement, as the differences that occur are probably due to little

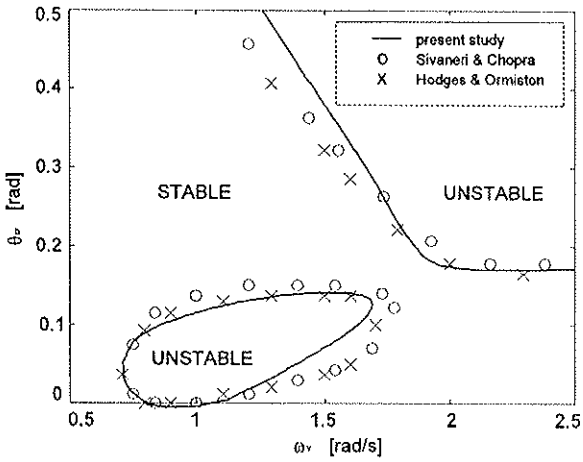

Fig. 8 Stability boundaries for a hingeless blade

differences in the simplifying assumptions or in the data considered.

Conclusions

A beam type finite element with ten degrees of freedom was developed using the capabilities of *Mathematica* software. The finite element formulation, first described in [6] was followed and the exact expressions of the nonlinear stiffness, inertia and damping matrices of the element were obtained. Quasi-steady two-dimensional airfoil theory of Greenberg's type was used to evaluate the aerodynamic loads. A special routine was written in *Mathematica* in order to retain in the equations only the important aerodynamic terms. The numerical applications presented concern the coupled natural frequency analysis of the rotating blade, the nonlinear steady-state position of the blade in hover at a set collective pitch, and flutter analysis about this equilibrium position. All these applications are written in MATLAB. The results are in good agreement with other similar analysis. The explicit relations of the element's matrices are an important advantage, as they make it possible to investigate the influences and contributions of each constructive or aerodynamic parameter of the blade. When the number of degrees of freedom of the element and the number of blade parameters is increased, however the obtained formulas may become very cumbersome as a result, this advantage is lost and numerical evaluation of the element's matrices is more appropiate [7], [12]. Another conclusion is that the general purpose symbolic code *Mathematica* makes it possible to readily perform the symbolic manipulations needed in problems of rotor blade dynamic and aeroelastic analysis.

Acknowledgements

This research has been funded by The Italian National Research Center (CNR) to which the authors wish to express their especial thanks.

References

[1] Crespo DaSilva, M. R. M., and Hodges, D. H., The role of Computerized Symbolic Manipulation in Rotorcraft Dynamic Analysis, Computers and Mathematics with Applications, Vol. 12A, 1986, pp. 161-172.

[2] Ravichandra, S., and Gaonkar, G., A Study of Symbolic Processing and Computational aspects in Helicopter Dynamics, Journal of Sound and Vibration, Vol. 137, No.3, 1990, pp. 495-507.

[3] Hodges, D. H., and Ormiston, R. A., Stability of Elastic Bending and Torsion of Uniform Cantilever Rotor Blade in Hover with Variable Structural Coupling, NASA TN D-8192, April, 1976.

[4] Friedmann, P.P., Formulation and Solution of Rotary-Wing Aeroelastic Stability and Response Problems, Vertica, Vol. 7, No. 2, 1983, pp.101-104.

[5] Straub, F. K., and Friedmann, P. P., Galerkin Type Finite Element Method for Rotary-Wing Aeroelasticity in Hover and Forward Flight, Vol. 5, No. 1, 1981, pp. 75-98.

[6] Sivaneri, N. T., and Chopra, I., Dynamic Stability of a Rotor Blade Using Finite

Element Analysis, A.I.A.A Journal, Vol. 20, No. 5, May, 1982, pp. 716-723.

[7] Straub, F. K., Sangha, K. B. and Panda, B., Advanced Finite Element Modeling of Rotor Blade Aeroelasticity, Journal of the American Helicopter Society, Vol. 39, No. 2, April, 1994, pp. 56-68.

[8] Hodges, D. H., Hopkins, A. S., Kunz, D. L., and Hinant, H. E., Introduction to GRASP-General Rotorcraft Aeromechanical Stability Program- A Modern Approach to Rotorcraft Modeling, Journal of the American Helicopter Society, Vol. 32, No. 2, April, 1987, pp. 78-90.

[9] Bir, G.S., Chopra, I., and Nguyen, K., Development of UMARC (University of Maryland Advanced Rotorcraft Code), Proceedings of *46th Annual Forum of the American Helicopter Society*, Vol. I, Washington, DC, May, 1990, pp. 55-78.

[10] Hodges, D. H., and Dowell, E. H., Nonlinear Equations of Motion for the Elastic Bending and Torsion of Twisted Nonuniform Rotor Blades, NASA TN D-7818, December, 1974.

[11] Murthy, V.R., Dynamic Characteristics of Rotor Blades, Journal of Sound and Vibration, Vol.49, No. 4, 1976, pp. 483-500.

[12] Rand, O., and Barkai, S. M., Numerical Evaluation of the Equation of Motion of Helicopter Blades with Symbolic Exactness, Journal of the American Helicopter Society, Vol. 40, No. 1, January, 1995, pp. 59-71.

Appendix A

$$[T] = \begin{bmatrix} 1 - \frac{v'^2}{2} - \frac{w'^2}{2} & v' & w' \\ -v'cos\theta_1 - w'sin\theta_1 & (1 - \frac{v'^2}{2})cos\theta_1 - v'w'sin\theta_1 & (1 - \frac{w'^2}{2})sin\theta_1 \\ v'sin\theta_1 - w'cos\theta_1 & -(1 - \frac{v'^2}{2})sin\theta_1 - v'w'cos\theta_1 & (1 - \frac{w'^2}{2})cos\theta_1 \end{bmatrix}$$

Appendix B

```
BeginPackage["Reduce'"]
Reduce::usage = "Reduce[expres, ordmax] reduces all the terms of the
                 expression 'expres' whose order is bigger than 'ordmax' "

Reduce[expres_, ordmax_] :=
  Module[{mag,factor,ordine,var,expout},
    row=5;              (* number of the parameters in the table mag  *)
    mag=Table[0,{i,1,row},{j,1,2}];
    mag[[1,1]]=A;  mag[[1,2]]=0;      (* parameter A; order of magnitude 0   *)
    mag[[2,1]]=B;  mag[[2,2]]=1;      (* parameter B; order of magnitude 1   *)
    mag[[3,1]]=C;  mag[[3,2]]=2;      (* parameter C; order of magnitude 2   *)
    mag[[4,1]]=D;  mag[[4,2]]=1.5;    (* parameter D; order of magnitude 1.5 *)
    mag[[5,1]]=E;  mag[[5,2]]=1;      (* parameter E; order of magnitude 1   *)
    expout=0;
    Do[{
         factor=Part[expres,i];
         ordtot=0;
         include=1;
         var=Variables[factor];
         Do[{
            here=0;
            Do[ If[ TrueQ[var[[j]]==mag[[k,1]] ],{
                    ordtot=ordtot+mag[[k,2]]*Exponent[factor,var[[j]]];
                    here=1}],
            {k,1,riga}];
            If[ here==0,{
               WriteString[{"stdout"},Error in the term,i];
               include=0}];
         },{j,1,Length[var]}];
         If[ ((ordtot <= ordmax) && (include == 1)), expout=expout+factor];
    },{i,1,Length[expres]}];
    expout
]
End[]
EndPackage[]
```

For example, if expres=$AC + A^3BD + 3ABE + 2DE^2 + D$ and ordmax=2
the output expression is AC+3ABE+D.
The message "Error in the term i" is given in the case of an unknown parameter (uncontained in Table 'mag')

Appendix C

The following symbols are used in *Mathematica*:

ai, bi, ci = the coefficients of the centrifugal force
eg = e_g; EA, EIy, EIz, GJ = stiffness EA, EI_y, EI_z, GJ
km, km1, km2 = k_m, k_{m1}, k_{m2}; l = l_i
Li = L_i; m = m; Om = Ω; s = $\xi_i = x_i/l_i$
st, ct = $sin(\theta_B + \theta_c)$, $cos(\theta_B + \theta_c)$
$v[s\_], w[s\_], f[s\_]$ = the displacements v, w, $\hat{\phi}$
$vd[s\_], wd[s\_], fd[s\_]$ = the derivatives of v, w, $\hat{\phi}$ with respect to x
$vdd[s\_], wdd[s\_]$ = the double derivatives of v, w with respect to x
$vp[s\_], wp[s\_], fp[s\_]$ = the derivatives of v, w, $\hat{\phi}$ with respect to time
$vpp[s\_], wpp[s\_], fpp[s\_]$ = the double derivatives of v, w, $\hat{\phi}$ with respect to time
$vdp[s\_], wdp[s\_]$ = the derivatives of v', w' with respect to time
$H1[s\_]...H6[s\_]$ = the shape functions $H_1...H_6$
$H1d[s\_]...H6d[s\_]$ = the derivatives with respect to x of the shape functions
$H1dd[s\_]...H6dd[s\_]$ = the double derivative with respect to x of the shape functions
q[i], qp[i], qpp[i] = q_i, $\dot{q}_i$, $\ddot{q}_i$, $i = 1..10$; dq[i] = δq_i
$dv[s\_]$ =, $dw[s\_], df[s\_] = \delta v$, δw, $\delta\hat{\phi}$ respectively
$dvd[s\_]$ =, $dwd[s\_], dfd[s\_] = \delta v'$, $\delta w'$, $\delta\hat{\phi}'$ respectively
$dvdd[s\_], dwdd[s\_] = \delta v''$, $\delta w''$
First the shape functions and the centrifugal force are defined:

```
H1[s_]:=2*s^3-3*s^2+1;
H2[s_]:=l*(s^3-2*s^2+s);
H3[s_]:=-2*s^3+3*s^2;
H4[s_]:=l*(s^3-s^2);
H5[s_]:=1-s;
H6[s_]:=s;
F[s_]:=ai*s^2+bi*s+ci;
```

The first and the second derivatives with respect to x of the shape functions are:

```
H1d[s_]:=D[H1[s],s]/l;
```

⋮

```
H6d[s_]:=D[H6[s],s]/l;

H1dd[s_]:=D[D[H1[s],s]/l,s]/l;
```

.

```
H6dd[s_]:=D[D[H6[s],s]/l,s]/l;
```

The expressions of the displacement field and of the derivatives with respect to x and t are:

```
v[s_]:=H1[s]*q[1]+H2[s]*q[2]+H3[s]*q[6]+H4[s]*q[7];
w[s_]:=H1[s]*q[3]+H2[s]*q[4]+H3[s]*q[8]+H4[s]*q[9];
f[s_]:=H5[s]*q[5]+H6[s]*q[10];

vd[s_]:=H1d[s]*q[1]+H2d[s]*q[2]+H3d[s]*q[6]+H4d[s]*q[7];
wd[s_]:=H1d[s]*q[3]+H2d[s]*q[4]+H3d[s]*q[8]+H4d[s]*q[9];
fd[s_]:=H5d[s]*q[5]+H6d[s]*q[10];

vdd[s_]:=H1dd[s]*q[1]+H2dd[s]*q[2]+H3dd[s]*q[6]+H4dd[s]*q[7];
wdd[s_]:=H1dd[s]*q[3]+H2dd[s]*q[4]+H3dd[s]*q[8]+H4dd[s]*q[9];

vp[s_]:=H1[s]*qp[1]+H2[s]*qp[2]+H3[s]*qp[6]+H4[s]*qp[7];
wp[s_]:=H1[s]*qp[3]+H2[s]*qp[4]+H3[s]*qp[8]+H4[s]*qp[9];
fp[s_]:=H5[s]*qp[5]+H6[s]*qp[10];

vdp[s_]:=H1d[s]*qp[1]+H2d[s]*qp[2]+H3d[s]*qp[6]+H4d[s]*qp[7];
wdp[s_]:=H1d[s]*qp[3]+H2d[s]*qp[4]+H3d[s]*qp[8]+H4d[s]*qp[9];

vpp[s_]:=H1[s]*qpp[1]+H2[s]*qpp[2]+H3[s]*qpp[6]+H4[s]*qpp[7];
wpp[s_]:=H1[s]*qpp[3]+H2[s]*qpp[4]+H3[s]*qpp[8]+H4[s]*qpp[9];
fpp[s_]:=H5[s]*qpp[5]+H6[s]*qpp[10];
```

The displacements variations are of the form:

```
dv[s_]:=H1[s]*dq[1]+H2[s]*dq[2]+H3[s]*dq[6]+H4[s]*dq[7];
dw[s_]:=H1[s]*dq[3]+H2[s]*dq[4]+H3[s]*dq[8]+H4[s]*dq[9];
df[s_]:=H5[s]*dq[5]+H6[s]*dq[10];

dvd[s_]:=H1d[s]*dq[1]+H2d[s]*dq[2]+H3d[s]*dq[6]+H4d[s]*dq[7];
dwd[s_]:=H1d[s]*dq[3]+H2d[s]*dq[4]+H3d[s]*dq[8]+H4d[s]*dq[9];
dfd[s_]:=H5d[s]*dq[5]+H6d[s]*dq[10];

dvdd[s_]:=H1dd[s]*dq[1]+H2dd[s]*dq[2]+H3dd[s]*dq[6]+H4dd[s]*dq[7];
dwdd[s_]:=H1dd[s]*dq[3]+H2dd[s]*dq[4]+H3dd[s]*dq[8]+H4dd[s]*dq[9];
```

This is followed by the expressions for strain energy, kinetic energy and virtual work:

```
(* only for linear terms in [M], [C] and [K]:
   EIzy=(EIz-EIy)*st*ct
   EIz1=EIz*ct^2+EIy*st^2
   EIy1=EIz*st^2+EIy*ct^2   *)

dUintlin[s_]:=F[s]*(vd[s]*dvd[s]
  +wd[s]*dwd[s])+GJ*fd[s]*dfd[s]+
  (EIz1*vdd[s]+EIzy*wdd[s])*dvdd[s]+
  (EIy1*wdd[s]+EIzy*vdd[s])*dwdd[s];

dTintlin[s_]:=(Om^2*(v[s]+eg*ct)+2*eg*Om*(vdp[s]*ct+
  wdp[s]*st)-vpp[s]+eg*fpp[s]*st)*dv[s]-(wpp[s]+
```

```
  eg*fpp[s]*ct)*dw[s]-(km^2*fpp[s]+Om^2*(km2^2-km1^2)*st*ct+
  Om^2*eg*(Li+s*l)*(wd[s]*ct-vd[s]*st)+eg*Om^2*v[s]*st-
  eg*(vpp[s]*st-wpp[s]*ct))*df[s]-eg*(Om^2*(Li+s*l)*ct+
  2*Om*vp[s]*ct)*dvd[s]-eg*(Om^2*(Li+s*l)*st+
  2*Om*vp[s]*st)*dwd[s]);

dWintlin[s_]:=0;

elint[s]=dUintlin[s]-dTintlin[s]-dWintlin[s];
elint1=Expand[elint[s]];
```

as well as by integration along the length of the element,

```
sumterms=0;
Do[{
term=Part[elint1,i];
sumterms=sumterms+l*Integrate[term,{s,0,1}];
},{i,1,Length[elint1]}]
```

and the systematic selection of each element of the mass, stiffness and damping matrices:

```
K=Table[0,{i,1,10},{j,1,10}];
M=Table[0,{i,1,10},{j,1,10}];
C=Table[0,{i,1,10},{j,1,10}];
Do[{
K[[i,j]]=Coefficient[sumterms,dq[i]*q[j]];
M[[i,j]]=Coefficient[sumterms,dq[i]*qpp[j]];
C[[i,j]]=Coefficient[sumterms,dq[i]*qp[j]];
},{i,1,10},{j,1,10}]
```

The symbolic expressions of these matrices are written in MATLAB as files mas10lin.m, sti10lin.m and dam10lin.m.

```
Do[{ t=ToString[StringForm["K(``,``)=``;",i,j,InputForm[K[[i,j]]]]];
    t >>> "\symb\sti10lin.m" }, {i,1,10}, {j,1,10}]
Do[{ t=ToString[StringForm["M(``,``)=``;",i,j,InputForm[M[[i,j]]]]];
    t >>> "\symb\mas10lin.m" }, {i,1,10}, {j,1,10}]
Do[{ t=ToString[StringForm["C(``,``)=``;",i,j,InputForm[C[[i,j]]]]];
    t >>> "\symb\dam10lin.m" }, {i,1,10}, {j,1,10}]
```

The nonzero terms of the symmetric stiffness matrix are listed in Appendix D. With this method, it is also possible to obtain these matrices in other cases starting from the more complete expressions of U, T and W. For example, β_p, k_A, e_A can be included. First the linear contributions in these matrices are obtained separately, as presented above. The nonlinear and aerodynamic contributions are then obtained using the ordering scheme to reduce the expanded expressions for the strain or kinetic energy and virtual work. In particular, the expression of the virtual work of the quasisteady aerodynamic loads contains several thousands terms, and the reduction procedure must be used.

Appendix D

For $e_g = 0$ the symmetric stiffness matrix of the element is:

```
K(1,1)=(12*EIz1)/l^3+ (12*ai)/(35*l) +(3*bi)/(5*l) + (6*ci)/(5*l) -
       (13*l*m*Om^2)/35;
K(1,2)=ai/14 + bi/10 + ci/10 + (6*EIz1)/l^2 - (11*l^2*m*Om^2)/210;
K(1,3)=(12*EIzy)/l^3;
K(1,4)=(6*EIzy)/l^2;
K(1,6)=(-12*EIz1)/l^3 - (12*ai)/(35*l) - (3*bi)/(5*l) - (6*ci)/(5*l) -
       (9*l*m*Om^2)/70;
K(1,7)=-ai/35 + ci/10 + (6*EIz1)/l^2 + (13*l^2*m*Om^2)/420;
K(1,8)=(-12*EIzy)/l^3;
K(1,9)=(6*EIzy)/l^2;
K(2,2)=(4*EIz1)/l + (2*ai*l)/105 + (bi*l)/30 + (2*ci*l)/15 -
       (l^3*m*Om^2)/105;
K(2,3)=(6*EIzy)/l^2;
K(2,4)=(4*EIzy)/l;
K(2,6)=-ai/14 - bi/10 - ci/10 - (6*EIz1)/l^2 - (13*l^2*m*Om^2)/420;
K(2,7)=(2*EIz1)/l - (ai*l)/70 - (bi*l)/60 - (ci*l)/30 + (l^3*m*Om^2)/140;
K(2,8)=(-6*EIzy)/l^2;
K(2,9)=(2*EIzy)/l;
K(3,3)=(12*EIy1)/l^3 + (12*ai)/(35*l) + (3*bi)/(5*l) + (6*ci)/(5*l);
K(3,4)=ai/14 + bi/10 + ci/10 + (6*EIy1)/l^2;
K(3,6)=(-12*EIzy)/l^3;
K(3,7)=(6*EIzy)/l^2;
K(3,8)=(-12*EIy1)/l^3 - (12*ai)/(35*l) - (3*bi)/(5*l) - (6*ci)/(5*l);
K(3,9)=-ai/35 + ci/10 + (6*EIy1)/l^2;
K(4,4)=(4*EIy1)/l + (2*ai*l)/105 + (bi*l)/30 + (2*ci*l)/15;
K(4,6)=(-6*EIzy)/l^2;
K(4,7)=(2*EIzy)/l;
K(4,8)=-ai/14 - bi/10 - ci/10 - (6*EIy1)/l^2;
K(4,9)=(2*EIy1)/l - (ai*l)/70 - (bi*l)/60 - (ci*l)/30;
K(5,5)=GJ/l;
K(5,10)=-(GJ/l);
K(6,6)=(12*EIz1)/l^3 + (12*ai)/(35*l) + (3*bi)/(5*l) + (6*ci)/(5*l) -
       (13*l*m*Om^2)/35;
K(6,7)=ai/35 - ci/10 - (6*EIz1)/l^2 + (11*l^2*m*Om^2)/210;
K(6,8)=(12*EIzy)/l^3;
K(6,9)=(-6*EIzy)/l^2;
K(7,7)=(4*EIz1)/l + (3*ai*l)/35 + (bi*l)/10 + (2*ci*l)/15 -
       (l^3*m*Om^2)/105;
K(7,8)=(-6*EIzy)/l^2;
K(7,9)=(4*EIzy)/l;
K(8,8)=(12*EIy1)/l^3 + (12*ai)/(35*l) + (3*bi)/(5*l) + (6*ci)/(5*l);
K(8,9)=ai/35 - ci/10 - (6*EIy1)/l^2;
K(9,9)=(4*EIy1)/l + (3*ai*l)/35 + (bi*l)/10 + (2*ci*l)/15;
K(10,10)=GJ/l;
```