# A computational implementation of modal analysis of continuous dynamic systems

Ariel E. Matusevich and José A. Inaudi (corresponding author)
*Department of Aeronautical Engineering, National University of Córdoba, 5000 Córdoba, Argentina*
*E-mail: jinaudi@sirve.cl*

**Abstract**  A computational implementation of modal analysis of continuous systems is presented. Modal analysis of truss, beam and shaft structures is developed in Matlab®. A numerical and an analytical method are developed for the computation of mode shapes, natural frequencies and modal equations of continuous structures using the method of separation of variables. Time domain techniques are programmed in modular functions for structural analysis. The functions constitute a set of tools of the Structural Analysis Toolbox (SAT-Lab), developed recently for teaching modelling, analysis and design of structures and mechanical systems. Application examples of the computational tools are presented.

**Key words**  vibrations; modal analysis; continuous parameter; software

## Introduction

The vibration of continuous systems is often a subject of study in courses on dynamics of structures for engineering students. Because the method of separation of variables applied to modal analysis of continuous media requires the solution of ordinary differential equations for the mode shapes for a specified set of boundary conditions, textbooks typically cover simple one-dimensional and two-dimensional structural models, such as a simple bar, a beam, a shaft, a pre-stressed cable, a membrane or a plate. The computation of mode shapes and natural frequencies of truss structures or frames composed of several beams in flexure, torsion and axial deformations using continuous system dynamics remains beyond the scope of a course on dynamics of structures. Instead, these problems are solved by means of finite element (FE) analysis, through a domain discretisation and classical assembly of stiffness and mass matrices.

Although FE methods have widespread use and give versatility to structural modelling, they require significant computation to achieve reasonable accuracy. Exact mode shapes cannot be computed due to the limitation imposed by element shape functions, and accuracy in dynamic modelling can be achieved only using a refined mesh.

To introduce real structures as examples for modal analysis in continuous systems, a set of functions was programmed and appended to SAT-Lab, a Structural Analysis Toolbox developed for Matlab [1]. The scope of these functions is modal analysis of distributed parameter systems composed of constant-section truss, shaft, or beam elements, lumped masses or springs. The authors have no knowledge of similar developments elsewhere. Widespread use, ease of programming, connectivity, graphical capabilities and other tools readily available were the reasons for
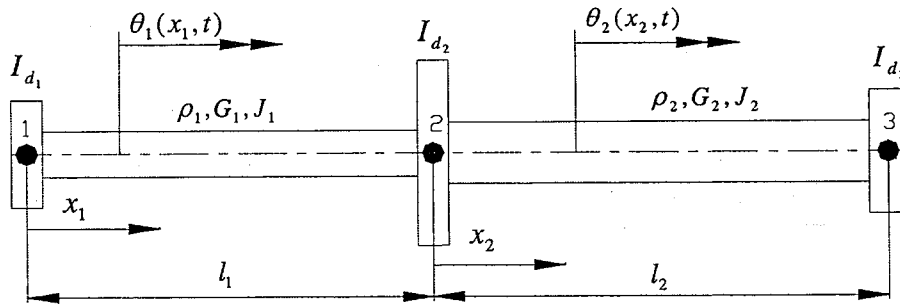
Fig. 1   *A non-uniform shaft.*

choosing the Matlab environment for the computer implementation of modal analysis of continuous systems.

This paper is organised as follows. First, the method of separation of variables for continuous system dynamics is revisited using a simple example. Boundary conditions are emphasised as the main tool for computing mode shapes. Next, the computer implementation is described; the subroutines or functions developed are presented along with their purpose, and the input and output variables. Two examples of modal analysis applied to a beam and a frame structure are described and analysis results are presented for these models.

## Analysis of continuous systems

As an example, consider the mechanical system shown in Fig. 1, consisting of three rigid discs connected by two continuous shafts of different cross-sections and materials. The modal analysis of torsional vibrations of this model can be developed using the method of separation of variables.

## Free vibration

The equations of motion in free vibration are:

$$G_1 J_1 \left( \frac{\partial^2 \theta_1(x_1, t)}{\partial x_1^2} \right) - \rho_1 J_1 \frac{\partial^2 \theta_1(x_1, t)}{\partial t^2} = 0 \tag{1}$$

$$G_2 J_2 \left( \frac{\partial^2 \theta_2(x_2, t)}{\partial x_2^2} \right) - \rho_2 J_2 \frac{\partial^2 \theta_2(x_2, t)}{\partial t^2} = 0 \tag{2}$$

where $\theta_1(x_1, t)$ and $\theta_2(x_2, t)$ are the displacement functions (angle of twist) and $G_i$, $\rho_i$ and $J_i$ represent the shear modulus, mass density and polar moment of inertia of element $i$, respectively. The mass polar moment of inertia of each disc is denoted by $I_{d_1}$, $I_{d_2}$ and $I_{d_3}$.

The boundary conditions of the problem are:

Moment equilibrium at $x_1 = 0$ (node 1):

$$G_1 J_1 \left[ \frac{\partial \theta_1(x_1, t)}{\partial x_1} \right]_{x_1 = 0} = I_{d_1} \frac{\partial^2 \theta_1(x_1, t)}{\partial t^2} \tag{3}$$

Moment equilibrium and compatibility of rotations at the joint (node 2):

$$-G_1 J_1 \left[ \frac{\partial \theta_1(x_1, t)}{\partial x_1} \right]_{x_1 = l_1} + G_2 J_2 \left[ \frac{\partial \theta_{21}(x_2, t)}{\partial x_2} \right]_{x_2 = 0} = I_{d_2} \frac{\partial^2 \theta_1(x_1, t)}{\partial t^2}$$

$$= I_{d_2} \frac{\partial^2 \theta_2(x_2, t)}{\partial t^2} \tag{4}$$

$$\theta_1(l_1, t) = \theta_2(0, t) \tag{5}$$

Moment equilibrium at $x_2 = l_2$ (node 3):

$$-G_2 J_2 \left[ \frac{\partial \theta_2(x_2, t)}{\partial x_2} \right]_{x_2 = l_2} = I_{d_3} \frac{\partial^2 \theta_1(x_1, t)}{\partial t^2} \tag{6}$$

The governing partial differential equations of motion, along with the prescribed boundary conditions, are usually referred to as a *boundary value problem.*

The equations of motion can be solved using the method of separation of variables, proposing:

$$\theta_1(x_1, t) = \gamma_1(x_1) q(t) \tag{7}$$

$$\theta_2(x_2, t) = \gamma_2(x_2) q(t) \tag{8}$$

Substitution of equations 7 and 8 into equations 1 and 2 yields ordinary differential equations for $\gamma_1(x_1)$, $\gamma_2(x_2)$ and $q(t)$. From these, we obtain the following solutions for the mode shape functions $\gamma_1(x_1)$ and $\gamma_2(x_2)$:

$$\gamma_1(x_1) = C_1 \sin(\eta_1 x_1) + C_2 \cos(\eta_1 x_1) \tag{9}$$

$$\gamma_2(x_2) = C_3 \sin(\eta_2 x_2) + C_4 \cos(\eta_2 x_2) \tag{10}$$

where:

$$\eta_1 = \sqrt{\frac{\rho_1}{G_1}} \omega \tag{11}$$

$$\eta_2 = \sqrt{\frac{\rho_2}{G_2}} \omega \tag{12}$$

The constants $C_1$, $C_2$, $C_3$, and $C_4$ must be evaluated so as to satisfy the specified boundary conditions.

Separating variables and adopting the prime and dot notations to indicate partial derivatives with respected to $x$ and $t$, respectively:

$$\theta_1(x_1, t) = \gamma_1(x_1)q(t)$$

$$\frac{\partial \theta_1}{\partial x_1} = \gamma_1'(x_1)q(t) \tag{13}$$

$$\frac{\partial^2 \theta_1}{\partial t^2} = \gamma_1(x_1)\ddot{q}(t) \tag{14}$$

Substituting equations 13 and 14 into equation 3:

$$(G_1 J_1)\gamma_1'(0)q(t) = I_{d_1}\gamma_1(0)\ddot{q}(t)$$

$$(G_1 J_1)\gamma_1'(0) + \omega^2 I_{d_1}\gamma_1(0) = 0 \tag{15}$$

In a similar manner, equation 4 leads to:

$$-(G_1 J_1)\gamma_1'(l_1) + (G_2 J_2)\gamma_2'(0) + \omega^2 I_{d_2}\gamma_2(0) = 0 \tag{16}$$

Equation 5 requires:

$$\gamma_1(l_1) - \gamma_2(0) = 0 \tag{17}$$

Finally, equation 6 imposes:

$$-G_2 J_2 \gamma_2'(l_2) + \omega^2 I_{d_3}\gamma_2(l_2) = 0 \tag{18}$$

Introducing equations 9 and 10 and their partial derivatives into equations 15–18, a set of linear equations on the coefficients $C_i$ is obtained:

$$[\mathbf{B}_c]\begin{bmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{19}$$

where $\mathbf{B}_c$ is the boundary condition matrix, composed of contributions $B_c^{(i)}$ from the $N_e$ elements of the structure (two shafts and three discs in this case):

$$\mathbf{B}_c = \sum_{i=1}^{N_e} B_c^{(i)} =$$

$$= \begin{bmatrix} G_1 J_1 \eta_1 & I_{d_1}\omega^2 & 0 & 0 \\ -G_1 J_1 \eta_1 \cos(\eta_1 l_1) & G_1 J_1 \eta_1 \sin(\eta_1 l_1) & G_2 J_2 \eta_2 & \omega^2 I_{d_2} \\ \sin(\eta_1 l_1) & \cos(\eta_1 l_1) & 0 & -1 \\ 0 & 0 & \begin{array}{c} -G_2 J_2 \eta_2 \cos(\eta_2 l_2) \\ + \omega^2 I_{d_3} \sin(\eta_2 l_2) \end{array} & \begin{array}{c} G_2 J_2 \eta_2 \sin(\eta_2 l_2) \\ + \omega^2 I_{d_3} \cos(\eta_2 l_2) \end{array} \end{bmatrix} \tag{20}$$

The existence of non-trivial solutions to equation 20 requires $\mathbf{B}_c$ to be singular. Therefore:

$$\det[\mathbf{B}_c(\omega)] = 0 \tag{21}$$

TABLE 1 *Shape functions for simple continuous systems*

| Element | Type of vibration | Mode shape function | |
|---------|-------------------|---------------------|---|
| Bar | Axial | $\phi(x) = C_1 \sin(\beta x) + C_2 \cos(\beta x)$ | $\beta = \sqrt{\dfrac{\rho}{E}}\,\omega$ |
| Shaft | Torsional | $\gamma(x) = C_1 \sin(\eta x) + C_2 \cos(\eta x)$ | $\eta = \sqrt{\dfrac{\rho}{G}}\,\omega$ |
| Bernoulli beam | Transverse | $\psi(x) = C_1 \sin(\alpha x) + C_2 \cos(\alpha x) + C_3 \sinh(\alpha x) + C_4 \cosh(\alpha x)$ | $\alpha = \left(\dfrac{\rho A}{EI}\right)^{\frac{1}{4}}\omega$ |

Equation 21 is the frequency equation of the continuous model. It has an infinite number of roots, $\omega_i$, and may be solved using numerical methods.

Mode shape functions, $\gamma_1(x)$ and $\gamma_2(x)$, can be computed, finding the non-zero coefficients $C_i$ that satisfy the linear homogeneous equation $\mathbf{B}_c|_{\omega_i} C = 0$ (equation 19) for each natural frequency. As this simple example illustrates, free vibration analysis of continuous systems reduces to the definition of shape functions and the assembly of the corresponding boundary condition matrix. Each boundary condition defines a row in $\mathbf{B}_c$, whose coefficients depend on the shape itself, its derivatives defined at the boundary and mass or stiffness parameters.

Shape functions for elements of constant cross-section and uniform material properties have analytical expressions that make the assembly of $\mathbf{B}_c$ relatively simple (see Table 1). The elements in Table 1 may be combined to build relatively complex three-dimensional models of structures. For example, consider the spatial frame shown in Fig. 2. Every member of the frame is subjected to flexural, axial and torsional deformations and can be modelled using the elements shown in Table 1. This approach requires the computation of shape functions for each kind of deformation. For instance, for the member connecting nodes 2 and 3, the shape functions correspond to axial vibration, torsional vibration, flexural vibration in the $x$–$y$ plane and flexural vibration in the $x$–$z$ plane.

These functions must satisfy given boundary conditions. For example, force equilibrium, moment equilibrium, nodal displacement compatibility and nodal rotation compatibility must be specified at node 2 of the structure. The specification of the boundary conditions at all nodes of the structure leads to the evaluation of $\mathbf{B}_c$.

The assembly of $\mathbf{B}_c$ is the most important operation of the present formulation and can be systematised through an assembly process, similar to the computation of mass or stiffness structural matrices in classical FE techniques.

The authors have developed a procedure for the assembly of $\mathbf{B}_c$ in structures made of bars, beams and shafts of constant section and uniform material properties, which may include additional lumped masses and springs. Once $\mathbf{B}_c$ is obtained, the process of computing natural frequencies and vibration modes is straightforward.
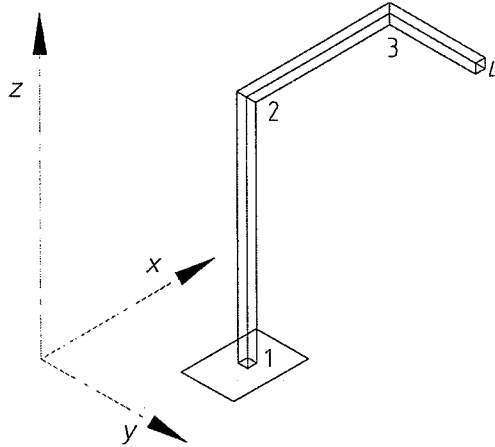
Fig. 2  *Spatial frame.*

## Forced vibration

Vibration modes can be used to solve forced vibration problems, by means of the modal superposition method, also known as modal analysis. Modal analysis relies on a transformation from displacement coordinates to normal or modal coordinates. In this transformation, we express a general response, $u(x,t)$, as a superposition of mode shapes $\phi_n(x)$, each multiplied by a generalised time-varying coordinate, $q_n(t)$:

$$u(x, t) = \sum_{n=0}^{\infty} \phi_n(x)q_n(t) \tag{22}$$

Although a continuous system has an infinite number of mode shapes, enough accuracy can be achieved using only a set of mode shapes associated with the lower natural frequencies.

It can be shown [2] that orthogonality relations of mode shapes in continuous systems imply a series of uncoupled equations of motion for each modal coordinate:

$$M_n\ddot{q}(t) + \omega^2 M_n q(t) = P_n(t) \tag{23}$$

where $M_n$ is the generalised mass associated with $\phi_n(x)$:

$$M_n = \int_0^l [\phi_n(x)]^2 \rho(x)A(x)\mathrm{d}x \tag{24}$$

and $P_n(t)$, is the generalised loading associated with $\phi_n(x)$:

$$P_n(t) = \int_0^l \phi_n(x)p(x, t)\mathrm{d}x \tag{25}$$

Equations 23–25 hold for longitudinal or transversal vibrations of a single-member structure of length $l$, cross-sectional area $A(x)$ and mass density $\rho(x)$. In structure models of several elements, including axial, torsional and flexural deformations, this computation requires an assembly process over each element of the structure [3].

## Computer implementation

Time domain tools for the analysis of continuous systems were implemented using the Matlab programming language [4]. The result was a set of subroutines or functions that can be used to solve free and forced problems of three-dimensional structures, and which have been included in the Structural Analysis Toolbox (SAT-Lab) [1].

These functions use the symbolic computation capabilities of Matlab's Symbolic Math Toolbox and implement the procedure outlined in Fig. 3.

The functions are Matlab programs, defined as Function M-files. These functions accept input arguments and return output arguments. For example, the function fun uses the arguments M, N, O, H, J and returns A and B:

```
[A, B] = fun (M, N, O, H, J)
```

In the remainder of this paper we will focus our attention on the software functionality, that is, the description of the purpose, input and output variables.

### Free vibration example
Consider a typical textbook example such as the uniform beam shown in Fig. 4. The SAT-Lab code required to obtain the first four natural frequencies and vibration modes of this model is described below. A detailed description of the variables and functions utilised is given afterwards.

```
% 1- Structural model definition:
1 = 6;                          % Beam length [m]
XYZ = [0 0 0; 1 0 0];           % Nodal coordinate matrix
EDICT.elname = 'csbeam';        % Element dictionary
EDICT.cstype = 'b';
EDICT.mode = 'phibeam';
E = 2e011;                      % Young Modulus [Pa]
A = 0.01;                       % Cross sectional area [m^2]
I = (0.1^4)/12;                 % Moment of inertia about
                                  local y-axis [m^4]
rho = 7800;                     % Mass density [kg/m3]
nc = 4; p = [0 1 0];
PROPERTIES = [nc E A rho I p];
ELEMENTS = [1 2 1 1];
DOF01 = [0 0 0 0 0 0; 0 0 1 0 0 0];  % Degrees of
                                        freedom matrix
% 2- Assembly of the boundary-condition matrix
[Bc, nc, ndofs, cpt] = csbc(XYZ, ELEMENTS, EDICT,
PROPERTIES, DOF01)
% 3- Computation of natural frequencies
po = 0.1; dp = 1; np = 4; tol = [1e-07 100];
om = csom(Bc, po, dp, np, tol)
% 4- C coefficients: the matrix CC
```
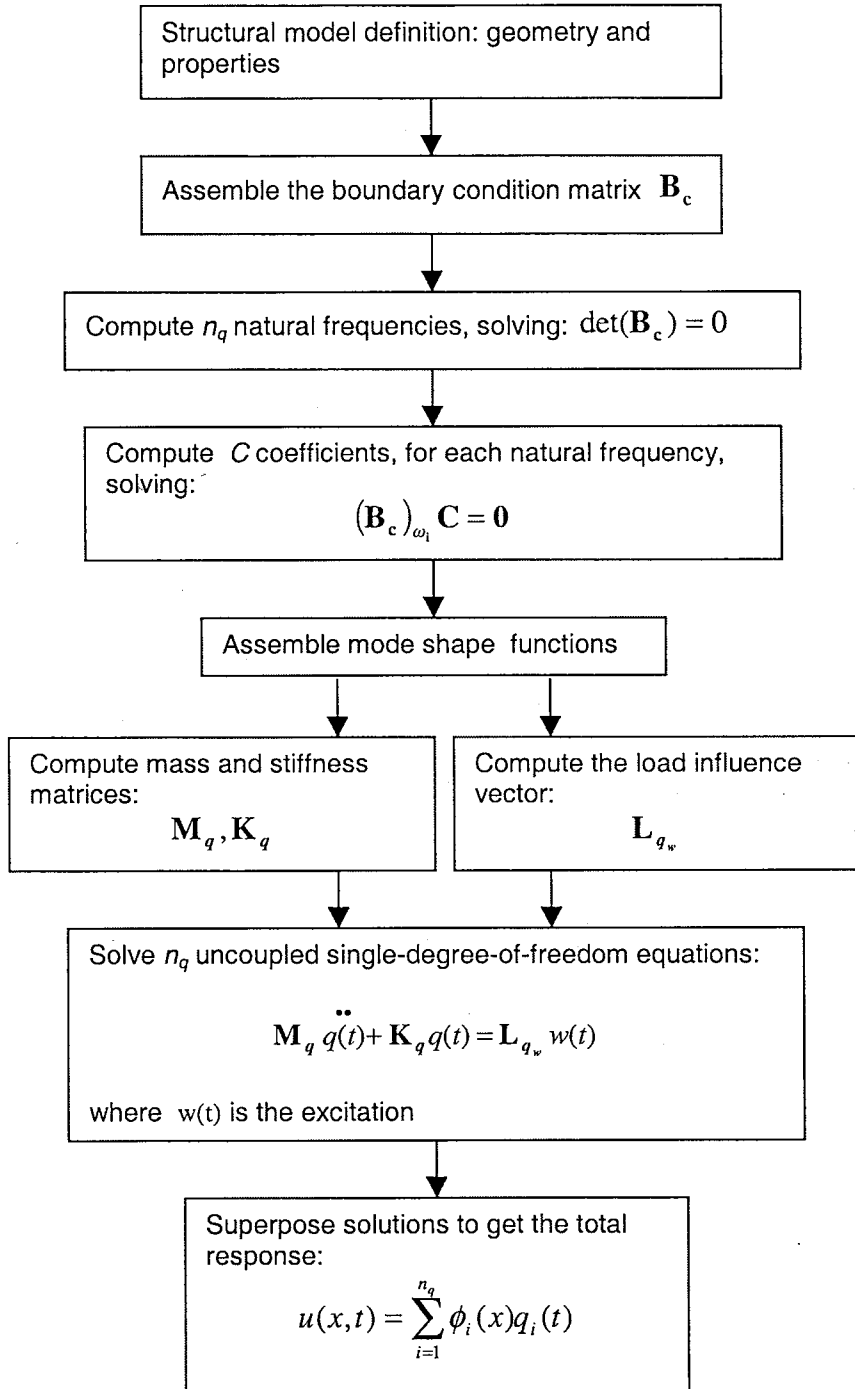
Structural model definition: geometry and properties

Assemble the boundary condition matrix $\mathbf{B}_c$

Compute $n_q$ natural frequencies, solving: $\det(\mathbf{B}_c) = 0$

Compute $C$ coefficients, for each natural frequency, solving:

$$\left(\mathbf{B}_c\right)_{\omega_i} \mathbf{C} = \mathbf{0}$$

Assemble mode shape functions

Compute mass and stiffness matrices:

$$\mathbf{M}_q, \mathbf{K}_q$$

Compute the load influence vector:

$$\mathbf{L}_{q_w}$$

Solve $n_q$ uncoupled single-degree-of-freedom equations:

$$\mathbf{M}_q \ddot{q}(t) + \mathbf{K}_q q(t) = \mathbf{L}_{q_w} w(t)$$

where $w(t)$ is the excitation

Superpose solutions to get the total response:

$$u(x,t) = \sum_{i=1}^{n_q} \phi_i(x) q_i(t)$$

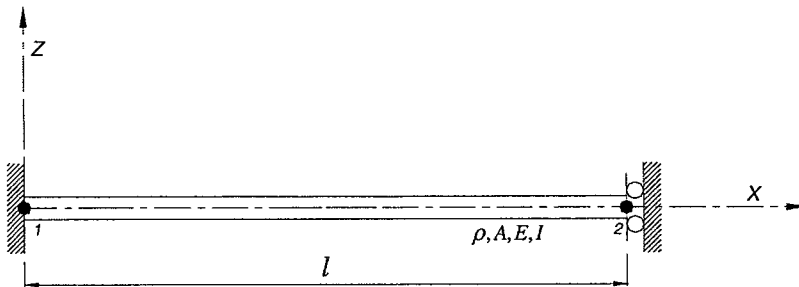Fig. 3   *Modal analysis procedure.*

Fig. 4 *Beam.*

```
CC = cscc(Bc,om);
% 5- Vibration modes
[phi] = csmodes (ELEMENTS, EDICT, PROPERTIES, CC, om,
        cpt);
```

*Structural model definition*

As we can see in the script presented in the previous section, the construction of a SAT-Lab structural model requires the specification of a set of variable, usually matrices and data structures, which provide information about the model geometry, element types, element properties, etc. These variables are input arguments of most SAT-Lab functions.

The Cartesian coordinates of the structure nodes are associated with rows of the nodal–coordinate matrix XYZ:

$$XYZ = \begin{bmatrix} \vdots & \vdots & \vdots \\ x_i & y_i & z_i \\ \vdots & \vdots & \vdots \end{bmatrix} \leftarrow \left(\text{row } i \text{ corresponds to element I}\right)$$

Information about the elements of the structural model is contained in the data structure EDICT, which has the fields shown in Table 2. Element names are stored in the field elname. They refer to SAT-Lab functions, which compute element contributions to **B**c as described in Table 3.

There are four possible types of boundary condition:

(1) force equilibrium;
(2) moment equilibrium;
(3) nodal displacement compatibility;
(4) nodal rotation compatibility.

The field cstype in EDICT refers to the kind of boundary condition that the element can contribute to **B**c, as described in Table 4.

The field mode in EDICT refers to functions that compute element mode shapes, as explained in Table 5.

TABLE 2   *Fields of* EDICT

| Field | Information |
|---|---|
| elname | Element name |
| cstype ' | Type of continuous element |
| mode | Mode-shape function name |

TABLE 3   *SAT-Lab continuous elements*

| Function name | Purpose |
|---|---|
| cstruss | Boundary conditions of a continuous truss element |
| csbeam | Boundary conditions of a continuous Bernoulli beam element |
| csshaft | Boundary conditions of a continuous shaft element |

TABLE 4   *Types of continuous elements*

| Field cstype | Possible boundary condition types | Elements of this type |
|---|---|---|
| t | 1–3 | Truss element |
| b | 1–2–3–4 | Bernoulli beam |
| s | 2–4 | Shaft |

TABLE 5   *Mode shape functions*

| Mode shape function | Corresponding element |
|---|---|
| phitruss | 'cstruss' |
| phibeam | 'csbeam' |
| phishaft | 'csshaft' |

Mechanical properties are specified in the matrix PROPERTIES. Each row of this matrix describes a different property type.

The first property to be specified in each continuous element is the number, $nc_e$, that indicates the quantity of coefficients $Ci$ of the element vibration mode. Mode shape coefficients determine the dimension of the boundary condition matrix $\mathbf{B}_c$.

In general, the matrix PROPERTIES has the following types of rows:

$$\text{PROPERTIES} = \begin{bmatrix} 2 & E & A & \rho & 0 & 0 & 0 & 0 \\ 2 & G & A & \rho & J & 0 & 0 & 0 \\ 4 & E & A & \rho & I_y & P_x & P_y & P_z \end{bmatrix} \begin{matrix} \leftarrow \text{(Continuous truss elements)} \\ \leftarrow \text{(Continuous shaft elements)} \\ \leftarrow \text{(Continuous beam elements)} \end{matrix}$$

where:

$E$ = Young's modulus;

$G$ = shear modulus;

$A$ = cross-sectional area;

$\rho$ = mass density;

$J$ = polar moment of inertia;

$I_y$ = cross-sectional moment of inertia about local $y$ axis;

$[P_x, P_y, P_z]$ = orientation vector of local $y$ axis in global coordinates.

A row of the matrix ELEMENTS defines the nodes I and J connected by the element, the type of element (pointer to an element of the data structure EDICT) and the property type (pointer to a row of the matrix PROPERTIES):

$$\text{ELEMENTS} = \begin{bmatrix} \vdots & \vdots & & \vdots & & \vdots \\ \text{I} & \text{J} & \text{Element type number} & \text{Property number} \\ \vdots & \vdots & & \vdots & & \vdots \end{bmatrix}$$

The kinematic conditions of nodal displacements are specified in the matrix DOF01. Row $i$ of DOF01 indicates the kinematic conditions of the displacements of node I in the six possible directions. Degrees of freedom are identified with a 1 (one) in DOF01, while restrained displacements are indicated with a 0 (zero).

$$\text{DOF01} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_i & y_i & z_i & \theta_{xi} & \theta_{yi} & \theta_{zi} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

The degree-of-freedom matrix (DOFS) is obtained by labelling or numbering degrees of freedom in DOF01.

If lumped masses and rotational inertias are present in some nodes of the structure, the matrix MASSES must be specified. In general:

$$\text{MASSES} = \begin{bmatrix} \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \text{I} & \text{M}_x & \text{M}_y & \text{M}_z & \text{I}_{xx} & \text{I}_{yy} & \text{I}_{zz} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{bmatrix}$$

where I indicates the node number in the structure in which the lumped mass is found. The remaining elements in a row of MASSES represent the mass properties.

*Assembly of $B_c$*

Once the model has been built, the symbolic boundary condition matrix is assembled. This task can be performed using csbc:

```
[Bc,nc,ndofs,cpt]=csbc(XYZ,ELEMENTS,EDICT,PROPERTIES,
DOF01,MASSES)
```

The input variables of this function define the structural model and were explained in the previous section. The output variables of csbc are:

(1)   the boundary condition matrix $\mathbf{B}_c$;
(2)   the number of mode shape coefficients of the structure (nc);
(3)   the number of degrees of freedom of the structure (ndofs);
(4)   pointers to element mode shape coefficients (cpt).

The authors have found, that in plane and spatial continuous structures, nodal displacements must be considered as problem unknowns [3]. For the sake of generality, nodal displacements are always taken as unknowns in the formulation. Thus, it follows from the preceding argument that the dimension of $\mathbf{B}_c$ is (nc+ndofs × nc+ndofs).

*Computation of natural frequencies*
Once $\mathbf{B}_c$ is obtained, natural frequencies can be computed using csom, which solves $\det(\mathbf{B}_c) = 0$, using the bisection method:

[om] =csom(Bc,po,dp,nw,tol)

This function returns a column vector om, which contains nw natural frequencies.

Given a point (po) at which to begin the iteration, the program evaluates $\det(\mathbf{B}_c)$ at dp intervals, until a change of sign is detected. In this way, a root is bracketed in an interval of length dp, where the bisection method can be used to obtain a precise value of the root. This process is repeated until the desired number of natural frequencies (nw), is computed.

The following convergence criterion is taken:

$$abs\left[\det(\mathbf{B}_c)\right] \le \varepsilon \tag{26}$$

where $\varepsilon$ is the absolute error. Since sometimes the condition given by equation 26 is unachievable, the program quits after a given number of iterations. The absolute error ($\varepsilon$) and the maximum number iterations are specified in the input variable tol.

*Computation of $C_i$ coefficients*
A set of $C_i$ coefficients includes the mode shape coefficients and nodal displacements associated with a vibration mode. The task of computing these coefficients can be performed using cscc:

[CC] =cscc(Bc,om)

The output of this function is a matrix CC, whose column i contains the set of coefficients associated with mode i.

*Assembly of vibration modes*
Once the $C_i$ coefficients are obtained, element mode shape functions can be assembled, using:

[phi] =csmodes(ELEMENTS,EDICT,PROPERTIES,CC,om,cpt)

This function returns, the modal function matrix phi. For a structure made of $k$ continuous elements, phi is of the following form:

$$\text{phi} = \begin{bmatrix} \text{phi}_1^{(1)} & \text{phi}_1^{(2)} & \cdots & \cdots & \text{phi}_1^{(n)} \\ \text{phi}_2^{(1)} & \text{phi}_2^{(2)} & & & \vdots \\ \text{phi}_3^{(1)} & \vdots & & & \vdots \\ \vdots & \vdots & & & \vdots \\ \text{phi}_k^{(1)} & \text{phi}_k^{(2)} & \cdots & \cdots & \text{phi}_k^{(n)} \end{bmatrix}$$

where the element $\text{phi}_j^{(i)}$ represents the mode shape function in element $j$ associated with the $i$th natural frequency.

*Example results*

The first four natural frequencies of the beam shown in Fig. 4 computed are:

$$\text{om} = \begin{bmatrix} 22.71 \\ 122.73 \\ 303.07 \\ 563.56 \end{bmatrix} [\text{rad/s}]$$

The following mode shape functions were obtained:

$$\text{phi}_1(x) = \text{phi}(1,1) = \sin(0.3942x) - 1.0178\cos(0.3942x)$$
$$- \sinh(0.3942x) + 1.0178\cosh(0.3942x)$$

$$\text{phi}_2(x) = \text{phi}(1,2) = \sin(0.9163x) - \cos(0.9163x) - \sinh(0.9163x) + \cosh(0.9163x)$$

$$\text{phi}_3(x) = \text{phi}(1,3) = \sin(1.4399x) - \cos(1.4399x) - \sinh(1.4399x) + \cosh(1.4399x)$$

$$\text{phi}_4(x) = \text{phi}(1,4) = \sin(1.9635x) - \cos(1.9635x) - \sinh(1.9635x) + \cosh(1.9635x)$$

A plot of these vibration modes is shown in Fig. 5.

### Forced vibration example

Suppose that the uniform beam analysed in the previous section is subjected to a central pulse function loading, as described in Fig. 6. To compute the response once modes and natural frequencies have been computed, we proceed as follows.

The following SAT-Lab code is a continuation of the code described above for the beam and shows how to a carry out a modal analysis of this model, using the first four vibration modes of the beam as normal coordinates, in order to get an approximation of the dynamic response.

```
% 6- Mass and stiffness matrices for modal analysis
[Mq, Kq] =csmk(om,phi,XYZ,ELEMENTS,PROPERTIES,EDICT);
% 7- Load-influence vector
LOAD.ldtype=2;            % Concentrated load
LOAD.elnum=1;
LOAD.ldparam=[-40 0.5]
[Lqw] =cslqw(LOAD,XYZ,ELEMENTS,phi,nc,CC)
% 8- Input Signal (SAT-Lab function sggen)
```
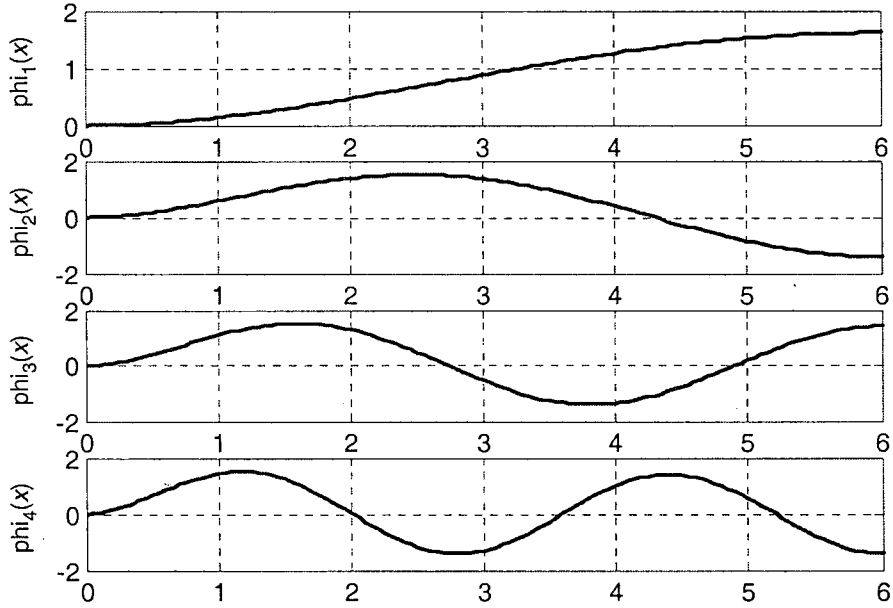
Fig. 5   *Beam vibration modes.*



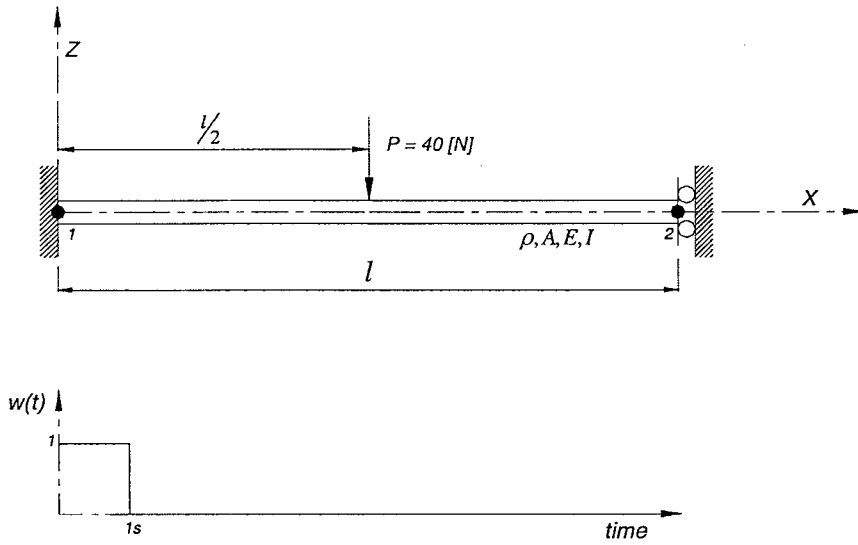Fig. 6   *Beam under a central pulse function loading.*

```
type=4                    % Rectangular pulse
W=1;                      % Intensity of pulse
tw=1                      % length of pulse (one second)
paramsg=[W tw];
to=0;                     % initial time
tf=3                      % final time
h=0.01                    % sampling time
tparam=[to tf h];
[w,t]=sggen(type,paramsg,tparam);
% 9- Numerical integration using the Newmark method
(SAT-Lab function: lvnewmk)
Cq=zeros(nw,nw)           % No damping
qo=zeros(nw,1);           % Displacement initial condition
qdo=zeros(nw,1);  % Velocity initial condition
h=0.01;                   % Sampling time
beta=1/4;
gamma=1/2;
param=[h beta gamma];
nsteps=301;
[q, qd]=lvnewmk(Mq,Cq,Kq,Lqw,w,qo,qdo,param,nsteps);  %
Integration using Newmark method
% 10- Superposition of modes coordinates
for t=1:nsteps
   u(t)=phi*q(:,t);
end
```

*Mass and stiffness matrices for modal analysis*
The elements of the mass matrix $\mathbf{M}_q$ can be evaluated using:

```
[Mij]=csmij(i,j,phi,XYZ,ELEMENTS,PROPERTIES,EDICT,
MASSES,DOF01,nc,CC)
```

Where i and j are the indices of the mass matrix element to be calculated. The remaining input variables have been computed in previous stages of this procedure (free-vibration analysis). The orthogonality of vibration modes with respect to mass distribution can be verified when different indices i and j are specified ($i \neq j \Rightarrow M(i,j) = 0$).

If $\mathbf{M}_q$ is the mass matrix and $\Omega$ is the natural-frequency matrix:

$$\mathbf{M}_q = \begin{bmatrix} M_{11} & & & \\ & M_{22} & & \\ & & \ddots & \\ & & & M_{n_q} \end{bmatrix}, \quad \Omega = \begin{bmatrix} \omega_1 & & & \\ & \omega_2 & & \\ & & \ddots & \\ & & & \omega_n \end{bmatrix}$$

the stiffness matrix $\mathbf{K}_q$ has the following expression:

$$\mathbf{K}_q = \mathbf{M}_q \Omega^2$$

Both $\mathbf{M}_q$ and $\mathbf{K}_q$ can be evaluated directly using:

```
[Mq,Kq]=csmk(om,phi,XYZ,ELEMENTS,PROPERTIES,EDICT,
MASSES,DOF01,nc,CC)
```

*Load influence vector*
The utilisation of modal coordinates involves a series of uncoupled single-degree-of-freedom equations of motion for each modal coordinate:

$$\mathbf{M}_q \ddot{q}(t) + \mathbf{K}_q q(t) = \mathbf{L}_{q_w} w(t)$$

Where $\mathbf{L}_{q_w}$ is the load influence vector and $w(t)$ the excitation. The load influence vector may be calculated for simple load patterns using `cslqw`:

```
[Lqw]=cslqw(LOAD,XYZ,ELEMENTS,phi,nc,CC)
```

where LOAD is a data structure, which contains information about the loads applied in the structural model.

*Example results*
The following single-degree-of-freedom equations, corresponding to each modal coordinate, were obtained:

$$\mathbf{M}_q \ddot{q}(t) + \mathbf{K}_q q(t) = \mathbf{L}_{q_w} w(t)$$

where:

$$\mathbf{M}_q = \begin{bmatrix} 484.82 & 0 & 0 & 0 \\ 0 & 468.0314 & 0 & 0 \\ 0 & 0 & 468.01 & 0 \\ 0 & 0 & 0 & 468 \end{bmatrix}$$

$$\mathbf{K}_q = 1.0e08 \begin{bmatrix} 0.0025 & 0 & 0 & 0 \\ 0 & 0.0705 & 0 & 0 \\ 0 & 0 & 0.4299 & 0 \\ 0 & 0 & 0 & 1.4863 \end{bmatrix}$$

$$\mathbf{L}_{q_w} = \begin{bmatrix} -35.1401 \\ -54.8340 \\ 21.1156 \\ 52.1519 \end{bmatrix}$$

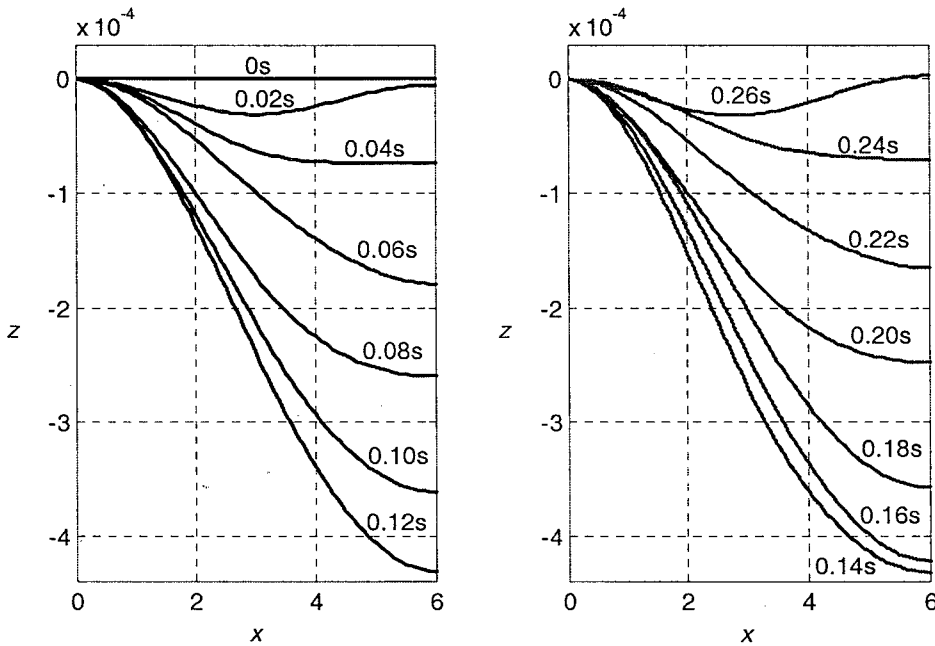These equations were numerically solved, using the SAT-Lab function `lvnewmk`, which uses the Newmark method [1].

Fig. 7   *Beam deformed configuration.*

Then, these solutions were superposed to obtain the total response. A plot of the deformed configuration from $t = 0$ s to $t = 0.26$ s, at 0.02 s intervals is shown in Fig. 7. The response can be visualised as a movie using functions available in the Toolbox.

### Frame example

Let us analyse the model of a three-dimensional frame subjected to a pulse loading, $F(t)$, applied at node 4, as described in Fig. 8 and Table 6.

As discussed above, in the section on free vibration, every member of this model includes axial, torsional and flexural deformations; therefore, a frame member can be modelled superposing a continuous truss element (cstruss) for axial deformation, a continuous shaft element (csshaft) for torsional deformation, a continuous beam element (csbeam) for flexural deformation in the $xz$ plane, and a continuous beam element (csbeam) for flexural deformation in the $xy$ plane.

The first four natural frequencies were computed and compared with those obtained using FE analysis tools available in SAT-Lab. As Table 7 shows, when the FE mesh is refined, the solution approaches to the result obtained using continuous elements.

Making use of the lower four vibration modes, an approximation of the forced vibration response was obtained. Vertical displacement, $z(t)$, of node 4 obtained is plotted in Fig. 9.
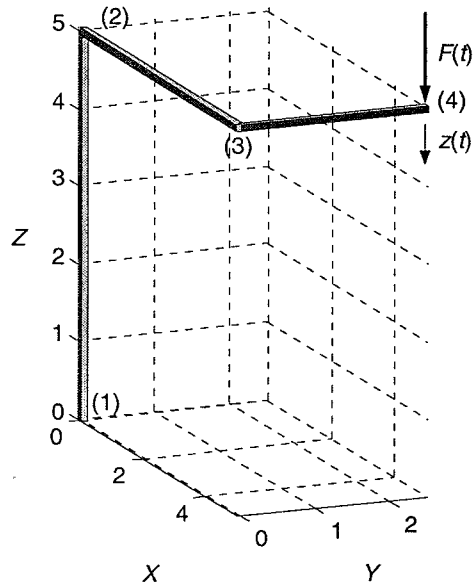
Fig. 8    *Frame under pulse loading.*

TABLE 6    *Frame properties*

| Mechanical properties | Value in example |
|---|---|
| Young's modulus ($E$) | 7.355$e$10 Pa |
| Cross-sectional area ($A$) | 0.0314 m$^2$ |
| Mass density ($\rho$) | 2700 kg/m$^3$ |
| Cross-sectional moment of inertial ($l_y = l_z$) | 4.9087$e$-066 m$^4$ |
| Polar moment of inertia | 8.33$e$-06 m$^4$ |
| Intensity of pulse loading | 40 N |
| Duration of pulse | 3 s |

TABLE 7    *Frame natural frequencies*

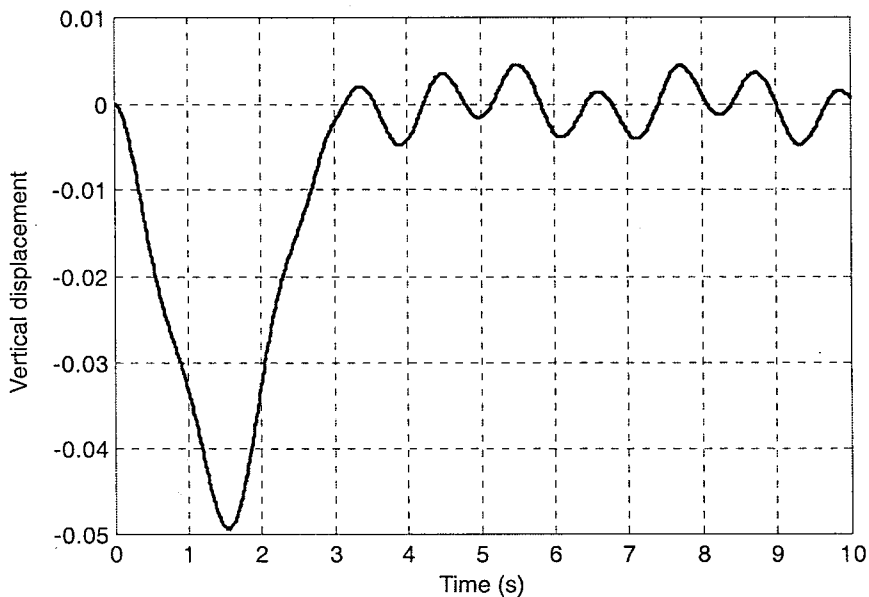| Method employed | $\omega_1$ (rad/s) | $\omega_2$ (rad/s) | $\omega_3$ (rad/s) | $\omega_4$ (rad/s) |
|---|---|---|---|---|
| Finite element mesh of 3 elements | 1.9349 | 2.0941 | 5.3228 | 5.8449 |
| Finite element mesh of 15 elements | 1.9317 | 2.1208 | 5.8413 | 6.2019 |
| Continuous elements (time domain) | 1.9314 | 2.1216 | 5.8389 | 6.2348 |

Fig. 9   *Displacement of node 4.*

## Numerical implementation

Modal analyses of large models lead to the assembly of boundary condition matrices of considerable dimensions. The manipulation of large symbolic objects is a time-consuming process in Matlab. This motivated the development of a numerical implementation of the modal analysis procedure. This implementation works with standard numerical Matlab commands and does not require the utilisation of Matlab's Symbolic Math Toolbox. The numerical functions programmed have similar input and output variables to those described in this paper [1].

These functions proved to be significantly better in computer efficiency than their symbolic counterparts. For example, the computation of natural frequencies in the frame example (Fig. 8) took 465 seconds on a personal computer (1200 MHz Athlon processor) using the analytical method and only 46 seconds using the numerical implementation.

## Concluding remarks

A set of tools developed for modal analysis of three-dimensional continuous structures has been presented. Two computer implementations were addressed. The analytical version was found to be very versatile, due to the handling of symbolic variables and functions in different stages of the modal analysis procedure. However, it proved to be less computer efficient (slower in computations) than the numerical version. The authors recommend the use of the analytical version for instructional

purposes, leaving the numerical version to the analysis of large structures, which require more computer time.

This software, used as a laboratory for improving the understanding of the dynamics of continuous systems, has proved to be a valuable instructional tool.

Future developments include the analysis of two-dimensional problems, such as plates or membranes, and a numerical implementation of variable parameter continuous elements using the technique presented here.

## References

[1]  J. A. Inaudi and J. C. De la Llera, *SAT-Lab Structural Analysis Toolbox, User Manual and Reference Manual*, www.sat-lab.com.
[2]  R. W. Clough and J. Penzien, *Dynamics of Structures* (MacGraw-Hill, New York, 1993).
[3]  A. E. Matusevich, *Computational Development for Modal Analysis of Continuous Dynamic Systems* (undergraduate thesis, in Spanish, National University of Córdoba, 2002).
[4]  The Mathworks, Inc., Matlab online documentation, www.matlab.com.