

SYMBOLIC MATRIX STRUCTURAL ANALYSIS OF TRUSSES: STATIC AND DYNAMIC APPLICATIONS

Vagelis Plevris, Abdulaziz Almutairi and Usama Ebead

Department of Civil and Environmental Engineering, College of Engineering, Qatar University
P.O. Box 2713, Doha–Qatar
e-mail: vplevris@qu.edu.qa, aa1903166@student.qu.edu.qa, uebead@qu.edu.qa

Abstract

Building on previous research that focused on deriving symbolic solutions for static loading scenarios using symbolic matrix structural analysis for plane trusses, this study expands the previous work by presenting an additional static analysis example and extending the methodology to free vibration dynamic analysis. Specifically, we investigate the feasibility of using symbolic computation to derive analytical expressions for natural frequencies and mode shapes in undamped free vibration 2d truss problems. The methodology is implemented in MATLAB using the Symbolic Math Toolbox. In the static analysis case, it enables closed-form solutions for displacements, support reactions, and internal axial forces, while in the dynamic analysis case it yields symbolic expressions for eigenfrequencies, eigenperiods, and vibration modes. Although the symbolic method performs well in static cases and yields elegant results for simple dynamic free vibration problems, it encounters significant scalability limitations in more complex dynamic scenarios, where expressions become excessively long and difficult to interpret. Nonetheless, the symbolic results remain mathematically valid and can be evaluated numerically in MATLAB, Excel, or other environments. All examples presented are fully validated using commercial finite element software and published analytical results.

Keywords: Symbolic Computation, Matrix Structural Analysis, Plane Trusses, Structural Dynamics, Free Vibration, MATLAB Symbolic Toolbox.

1 INTRODUCTION

The Finite Element Method (FEM) is a foundational tool in structural engineering, enabling the analysis of complex structural systems with high precision [1, 2]. It is widely used across academia and industry for evaluating the static and dynamic behavior of trusses, frames, and continuum structures. Traditionally, FEM is implemented numerically, offering flexibility and efficiency for large-scale problems. However, numerical solutions are tied to specific input values and typically obscure the explicit relationships between structural parameters and system response.

Matrix Structural Analysis (MatSA), a specialized form of FEM for linear-type structures such as trusses and frames [3], provides a systematic way to derive structural equations and assemble global stiffness matrices. When combined with symbolic computation, MatSA can be transformed from a purely numerical method into a tool that delivers closed-form analytical expressions [4, 5]. This symbolic approach has the potential to enhance understanding, support parametric exploration, and offer educational value by explicitly showing how input parameters influence the response of a structure.

MATLAB is a widely adopted platform in structural engineering, valued for its robust numerical capabilities in areas such as matrix structural analysis, finite element modeling [6-9], structural dynamics [10-13], and optimization tasks [14-18], among others [19, 20]. Beyond its well-established numerical strengths, MATLAB also offers symbolic computation through the Symbolic Math Toolbox [21, 22]. This toolbox enhances MATLAB's functionality by supporting algebraic simplification, symbolic differentiation and integration, equation solving, and matrix operations in symbolic form [23-25]. For both researchers and educators, it provides a practical means to derive exact analytical expressions and explore theoretical relationships, thereby deepening understanding of complex structural systems. Thanks to its seamless integration with MATLAB's numerical environment, users can easily transition between symbolic derivations and numerical evaluations. Similar symbolic capabilities are also available in other platforms such as Mathematica [26], Maple [27], and Python's SymPy library [28].

In previous works, we applied symbolic computation to the analysis of continuous beams [5, 29] and to 2D trusses [4, 30], under static loading conditions. Those works demonstrated how symbolic expressions for nodal displacements, axial forces, and support reactions can be derived directly from finite element formulations. The results were validated against closed-form solutions from the literature and commercial finite element software, showing excellent accuracy and confirming the practicality of the method for small to moderately sized beam and truss systems.

The present contribution extends the previous work on plane trusses [4, 30] in two directions. First, we revisit the static symbolic analysis and introduce an additional example to further demonstrate the methodology's flexibility and capability to handle more intricate truss configurations. Second, and more significantly, we extend the symbolic MatSA framework to dynamic analysis. We investigate the symbolic formulation and solution of the free vibration problem for plane trusses by constructing both the global stiffness and mass matrices symbolically and solving the resulting eigenvalue problem. To illustrate this extension, we present two dynamic examples involving trusses with increasing complexity.

This work contributes to the growing body of research that explores the interface between symbolic computation and structural analysis. It demonstrates that symbolic MatSA is not only viable for static problems but can also be extended to structural dynamics, albeit with practical constraints. The examples presented herein serve as benchmarks for assessing the scope and limitations of symbolic methods in structural engineering applications.

2 SUMMARY OF THE SYMBOLIC METHODOLOGY AND EXTENSION TO DYNAMIC ANALYSIS

The symbolic MatSA methodology for plane trusses enables the derivation of closed-form expressions for structural response by expressing all quantities—geometry, material properties, displacements, and forces—as symbolic variables. In the static analysis case [4], the global stiffness matrix $\mathbf{K}$ is assembled using standard finite element procedures, where the element stiffness matrices are derived symbolically and transformed into the global coordinate system. The global matrix is then constructed by assembling all element contributions based on the model's connectivity and geometry.

In extending this methodology to dynamic analysis, we consider the undamped free vibration of plane truss structures with lumped masses at the nodes. External static loads are not taken into account. The key extension involves introducing a global mass matrix $\mathbf{M}$, which is also constructed symbolically. To retain analytical tractability, a lumped mass approach is adopted: each node can carry a symbolic (or numeric) translational mass, and rotational inertia is neglected. Only the explicitly assigned nodal masses are considered—self-weight or distributed mass effects are not included.

Support conditions are incorporated by identifying constrained degrees of freedom and removing them from the global stiffness and mass matrices. This yields the reduced matrices $\mathbf{K}_{ff}$ for stiffness and $\mathbf{M}_{ff}$ for mass, corresponding to the active (free) degrees of freedom. The dynamic free vibration problem is then formulated as the standard generalized eigenvalue problem:

$$\mathbf{K}_{ff} \boldsymbol{\Phi}_i = \lambda_i \mathbf{M}_{ff} \boldsymbol{\Phi}_i \quad (1)$$

where $\lambda_i = \omega_i^2$ is the eigenvalue i representing the square of the natural circular frequency ω_i , and $\boldsymbol{\Phi}_i$ is the eigenvector corresponding to the vibration mode shape i . In the symbolic framework, this equation is solved by setting the determinant of Eq. (2) equal to zero:

$$|\mathbf{K}_{ff} - \lambda_i \mathbf{M}_{ff}| = 0 \quad (2)$$

and symbolically solving the resulting characteristic polynomial for λ . In MATLAB, symbolic eigenvalue analysis differs from the numerical case in that the symbolic `eig` function does not support the two-argument form `eig(Kff, Mff)`. Therefore, the generalized eigenvalue problem must be reformulated into a standard one using matrix inversion, as follows: $[V, D] = \text{eig}(\text{inv}(\mathbf{M}_{ff}) * \mathbf{K}_{ff})$. This approach allows for the symbolic computation of eigenvalues (stored in D) and corresponding eigenvectors (stored in V) for the free vibration problem.

For simple 2D truss configurations, this symbolic process yields explicit analytical expressions for natural frequencies and mode shapes. These expressions clearly show how the dynamic behavior of the structure depends on parameters such as stiffness, geometry, and nodal masses. As with the static symbolic formulation, this method provides valuable insight into parameter sensitivity and structural performance—particularly in educational settings and verification tasks.

3 DEFINITION OF THE MODEL SYMBOLICALLY

In the symbolic Matrix Structural Analysis framework, structural models are defined using symbolic variables that represent geometry, material properties, connectivity, and boundary conditions. In the static analysis of trusses, this was previously achieved using symbolic definitions for the following input arrays [4]: *NodeCoords*, *ElemMatSec*, *ElemCon*, *Supports*, *PointLoads*.

For free vibration dynamic analysis, we retain the same symbolic model definition with one important change: instead of defining external point loads, we introduce lumped masses at the nodes to capture the inertial properties of the structure. Since we are interested in free vibration behavior, external loads are not required—only the stiffness and mass characteristics of the structure matter. Additionally, self-weight is not considered; only the explicitly defined nodal masses are included in the model. The symbolic definition of the model is organized through the following five input variables:

- **NodeCoords**: A matrix of size $NumNodes \times 2$, where each row defines the x - and y -coordinates of a node. The first column represents the x -coordinate, and the second the y -coordinate. Coordinates can be either numeric values or symbolic variables (e.g. L).
- **ElemMatSec**: A column vector of size $NumElements$, where each entry specifies the axial stiffness of a truss element as the product $E \cdot A$, where E is Young's modulus and A is the cross-sectional area. Values can be numeric or symbolic (e.g. EA).
- **ElemCon**: A matrix of size $NumElements \times 2$, where each row defines the element connectivity. The two columns represent the start and end node IDs of each element, with node IDs given as numeric values.
- **Supports**: A matrix of size $NumNodes \times 2$, where each row defines the boundary conditions at a node. The first column corresponds to restraint in the x -direction, and the second column to the y -direction. A value of 1 indicates a fixed degree of freedom (DOF), while 0 indicates a free DOF.
- **Masses**: A column vector of size $NumNodes \times 1$, where each entry specifies the lumped mass at the corresponding node. Mass values can also be defined symbolically (e.g. m).

This flexible symbolic structure allows the formulation of both simple and moderately complex plane truss models, making it possible to carry out fully symbolic static and dynamic analyses. Once defined, the model serves as input to the symbolic finite element procedures that construct the global stiffness and mass matrices required for the eigenvalue problem.

4 LINEAR STATIC ANALYSIS EXAMPLE

We present a new numerical example for the static analysis case. In this example, the axial stiffness EA is treated as a symbolic parameter, as it appears consistently in the stiffness matrix and related analytical expressions for 2D trusses. While a common symbolic EA is used for simplicity, the methodology supports assigning different EA values to each member to model varying stiffness properties.

This numerical example can be found in the literature. In particular, it is the case No 3 truss model from the work of Tinkov [31], which is shown in Figure 1. The symbolic parameters are four: EA , a , b , and P . It consists of 14 nodes and 25 bar elements, forming a configuration composed of vertical panels of equal width a and height b . The truss is supported at nodes 1 and 13, which are assumed to be roller-supported (node 1) and pinned (node 13), providing a statically determinate support condition. A single vertical point load P is applied downward at the central bottom node 7. All members have a symbolic axial stiffness EA . The structure is ideal for symbolic analysis due to its regularity.

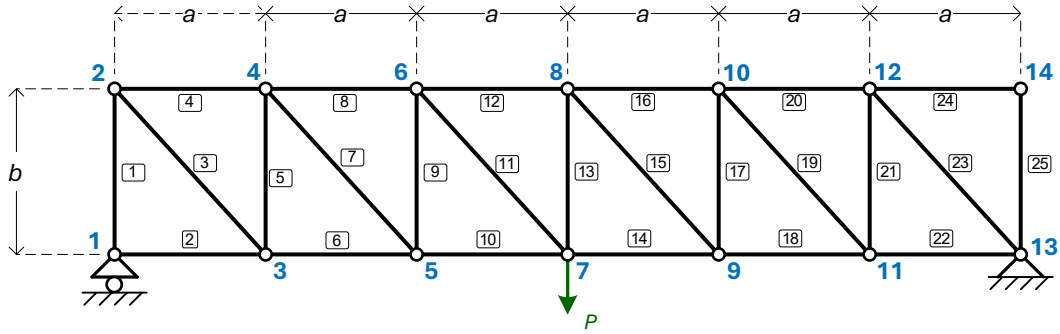


Figure 1. The truss of the numerical example of the static analysis case.

The results of the symbolic analysis are given in Table 1, Table 2 and Table 3, for Node displacements (for selected nodes), support reactions, and element axial forces, respectively.

Node #	x-Displacement (D_x)	y-Displacement (D_y)
Node 3	$-\frac{9Pa^2}{2EAb}$	$-\frac{P((a^2+b^2)^{3/2}+9a^3+b^3)}{2EAb^2}$
Node 5	$-\frac{4Pa^2}{EAb}$	$-\frac{P((a^2+b^2)^{3/2}+8a^3+b^3)}{EAb^2}$
Node 7	$-\frac{3Pa^2}{EAb}$	$-\frac{P(3(a^2+b^2)^{3/2}+19a^3+3b^3)}{2EAb^2}$
Node 9	$-\frac{3Pa^2}{2EAb}$	$-\frac{P((a^2+b^2)^{3/2}+8a^3+b^3)}{EAb^2}$
Node 11	$-\frac{Pa^2}{2EAb}$	$-\frac{P((a^2+b^2)^{3/2}+9a^3+b^3)}{2EAb^2}$

Table 1: Static analysis example – Node displacements (for selected nodes).

Node #	Force F_x	Force F_y
Node 1	0	$\frac{P}{2}$
Node 13	0	$\frac{P}{2}$

Table 2: Static analysis example – Support reactions.

Element #	Axial Force	Element #	Axial Force	Element #	Axial Force
1	$-\frac{P}{2}$	10	$\frac{a}{b}P$	19	$-\frac{P\sqrt{a^2+b^2}}{2b}$
2	0	11	$\frac{P\sqrt{a^2+b^2}}{2b}$	20	$-\frac{a}{2b}P$
3	$\frac{P\sqrt{a^2+b^2}}{2b}$	12	$-\frac{3a}{2b}P$	21	$\frac{P}{2}$
4	$-\frac{a}{2b}P$	13	$\frac{P}{2}$	22	$\frac{a}{2b}P$
5	$-\frac{P}{2}$	14	$\frac{3a}{2b}P$	23	$-\frac{P\sqrt{a^2+b^2}}{2b}$
6	$\frac{a}{2b}P$	15	$-\frac{P\sqrt{a^2+b^2}}{2b}$	24	0
7	$\frac{P\sqrt{a^2+b^2}}{2b}$	16	$-\frac{a}{b}P$	25	0
8	$-\frac{a}{b}P$	17	$\frac{P}{2}$		
9	$-\frac{P}{2}$	18	$\frac{a}{b}P$		

Table 3. Static analysis example: Element axial forces.

4.1 Validation against analytical results from the literature

As mentioned, this truss example is based on a case from the literature, specifically Case No. 3 from the work of Tinkov [31]. In this study, the author presents an analytical formula for calculating the vertical displacement of the central node (node 7 in our model). The formula is the one shown in Eq. (3):

$$\Delta_7 = \frac{P}{EF} \cdot \frac{Aa^3 + Bb^3 + Cc^3}{Db^2} \quad (3)$$

Tinkov's methodology provides analytical solutions for displacements in various truss configurations, using the parameter n , which defines the truss topology and the number of repetitions of the main truss pattern, for each truss configuration. For the specific truss model shown in Figure 1, it is $n = 3$. According to the study [31], the values of the parameters A , B , C , D and c for the specific truss are the following:

- $A = \frac{n(2n^2 + 1)}{3} = 19$
- $B = C = n = 3$
- $D = 2$
- $c = \sqrt{a^2 + b^2}$

It is important to note that in Eq. (3) the variable F represents the cross-sectional area. Substituting F with our variable A for the cross-sectional area (not to be confused with the above parameter $A=19$ of the work of Tinkov [31]), and taking the above values of the parameters into

account, the final formula for the vertical displacement of node 7, as derived in [31], is the one of Eq. (4):

$$\Delta_7 = \frac{P}{EA} \cdot \frac{Aa^3 + Bb^3 + Cc^3}{Db^2} = \frac{P}{EA} \cdot \frac{19a^3 + 3b^3 + 3(\sqrt{a^2 + b^2})^3}{2b^2} \quad (4)$$

Eq. (4) fully matches the symbolic formula derived by our program for the vertical displacement of node 7, as presented in Table 1. The negative sign in the equation in Table 1 indicates that the displacement is directed downwards.

4.2 Validation using numerical results from EngiLab Truss.2D Pro

The static analysis example was validated using EngiLab Truss 2D 2022 Pro, a specialized software for the static analysis of plane truss models. The following properties are used for the numerical calculations:

- Length (a): 2 m
- Height (b): 3 m
- Axial stiffness (EA): Derived using $E=200 \times 10^6$ kN/m² (200 GPa) and $A=4 \times 10^{-4}$ m² (4 cm²), resulting in $EA=8 \times 10^4$ kN.
- Point load (P): 50 kN

The truss was modeled in EngiLab Truss 2D Pro, as illustrated in Figure 2. Figure 3 shows the deformation of the truss, and Figure 4 shows the element forces with colors, where elements in tension are drawn with red color while elements in compression are shown with blue color.

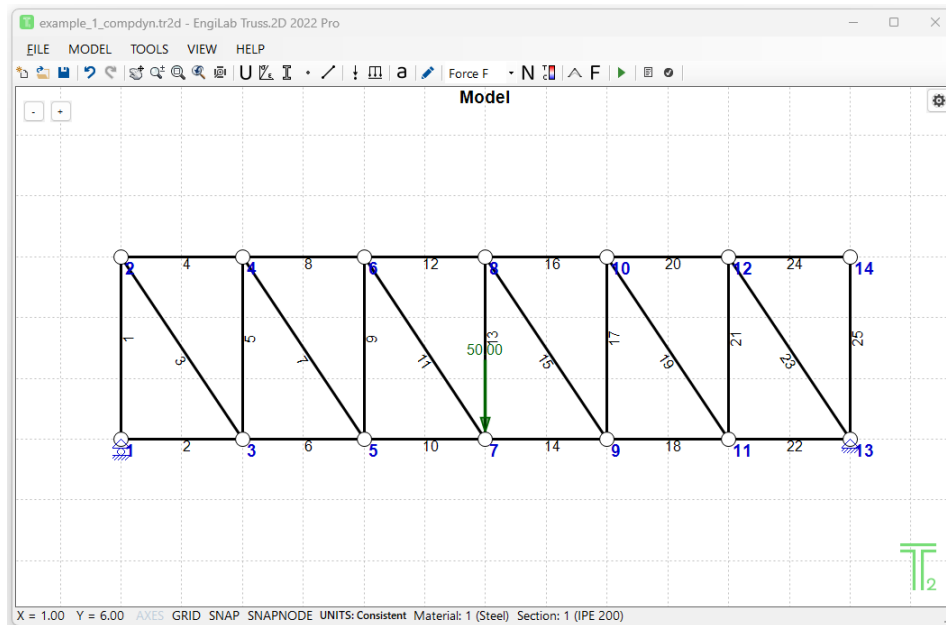


Figure 2. The truss of the numerical example of the static analysis case, modeled with EngiLab Truss.2D Pro.

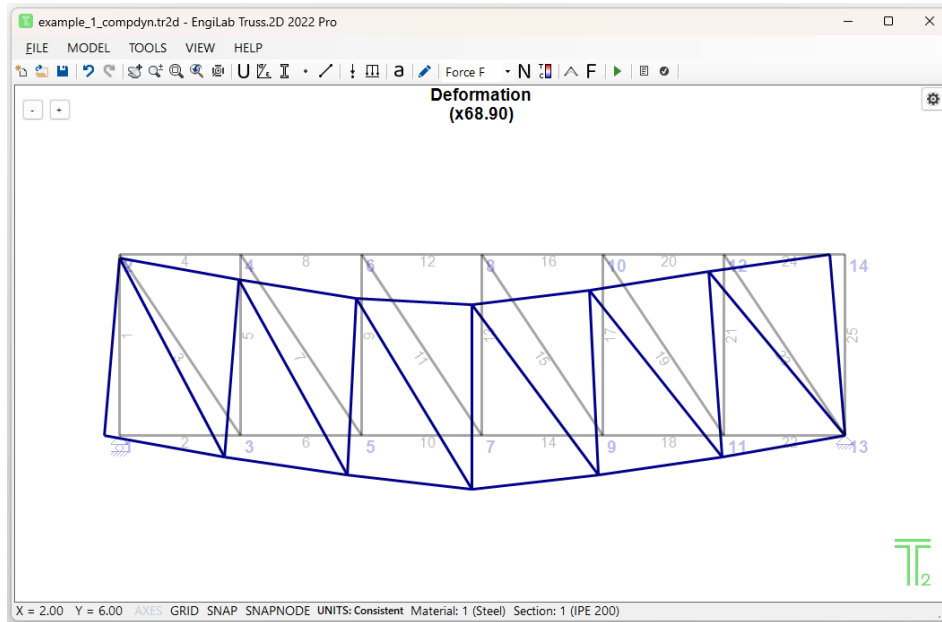


Figure 3. Deformation of the truss structure (exaggerated), as visualized using EngiLab Truss.2D Pro..

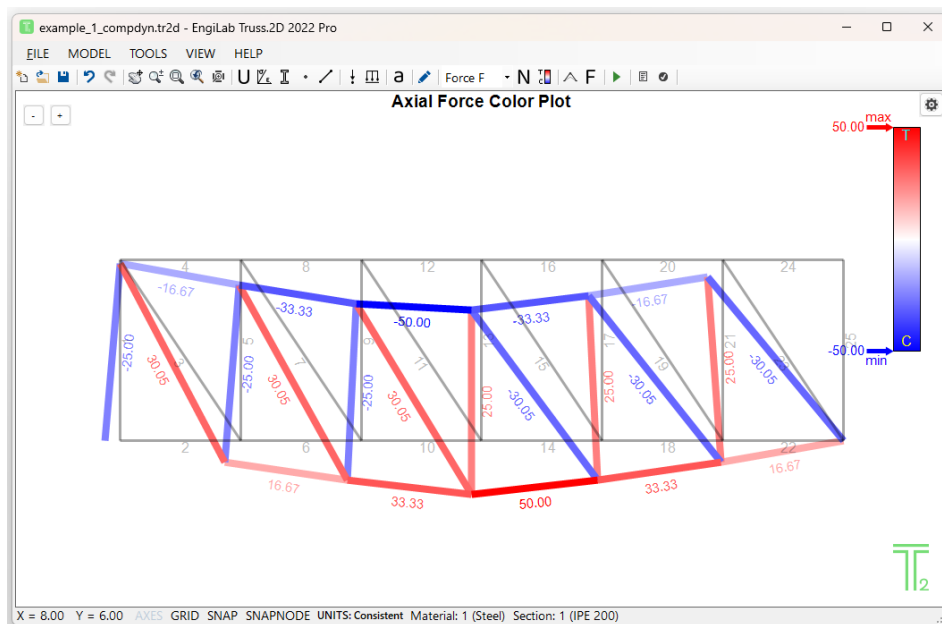


Figure 4. Axial force distribution in the truss, as computed by EngiLab Truss.2D Pro.
Compressive forces are shown in blue, tensile forces in red.

Table 4 compares the node displacements between our symbolic software and EngiLab Truss 2D Pro, showing identical results. This consistency also applies to support reactions and element axial forces, though these results are not included here.

Node #	Numeric value of symbolic expression		EngiLab Truss.2D Pro Result	
	x-Displacement (D_x)	y-Displacement (D_y)	x-Displacement (D_x)	y-Displacement (D_y)
Node 1	-0.00375	0	-0.00375	0.00000
Node 2	0	-0.00094	0.00000	-0.00094
Node 3	-0.00375	-0.00507	-0.00375	-0.00507
Node 4	-0.00042	-0.006	-0.00042	-0.00600
Node 5	-0.00333	-0.00957	-0.00333	-0.00957
Node 6	-0.00125	-0.01051	-0.00125	-0.01051
Node 7	-0.0025	-0.01297	-0.00250	-0.01297
Node 8	-0.0025	-0.01204	-0.00250	-0.01204
Node 9	-0.00125	-0.00957	-0.00125	-0.00957
Node 10	-0.00333	-0.00864	-0.00333	-0.00864
Node 11	-0.00042	-0.00507	-0.00042	-0.00507
Node 12	-0.00375	-0.00413	-0.00375	-0.00413
Node 13	0	0	0.00000	0.00000
Node 14	-0.00375	0	-0.00375	0.00000

Table 4. Node displacements for the static analysis example.
Comparison with EngiLab Truss.2D Pro (all values in m units for this example).

5 DYNAMIC ANALYSIS EXAMPLES

5.1 Dynamic analysis Example 1

The first dynamic analysis numerical example is a simple 2-bar truss with a single mass m , as shown in Figure 5. This example considers undamped free vibration, meaning that no damping or external forces are included—the analysis focuses purely on the inherent dynamic properties of the structure based on its stiffness and lumped masses. The symbolic parameters are three: EA , L , and m .

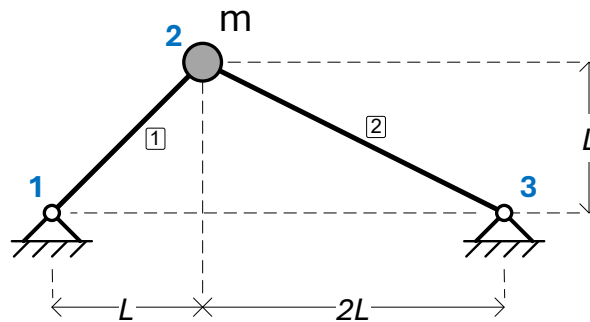


Figure 5. Dynamic analysis Example 1: The truss structure.

Table 5 shows the details of the model, as given in MATLAB. In every example, the number of nodes ($NumNodes$) is defined by the rows of the $NodeCoords$ matrix and the number of elements ($NumElements$) is defined by the rows of the $ElemMatSec$ matrix.

Table 6 presents the results of the dynamic symbolic analysis in terms of the symbolic parameters. It provides expressions for the squared angular frequencies ω_i^2 , natural frequencies f_i , periods T_i , and mode shapes Φ_i , all given in symbolic form as functions of the axial stiffness

EA , the length parameter L , and the mass m . These expressions offer exact analytical insight into the dynamic behavior of the system and clearly illustrate how each quantity depends on the fundamental physical parameters.

NodeCoords	$\begin{bmatrix} 0 & L & 3L \\ 0 & L & 0 \end{bmatrix}^T$
ElemMatSec	$[EA \quad EA]^T$
ElemCon	$\begin{bmatrix} 1 & 2 \\ 2 & 3 \end{bmatrix}^T$
Supports	$\begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \end{bmatrix}^T$
Masses	$[0 \quad m \quad 0]^T$

Table 5. Dynamic analysis Example 1: Details of the input parameters.

Quantity	1 st Eigenmode
ω_1^2	$\frac{EA(5\sqrt{2} - \sqrt{2}\sqrt{35-8\sqrt{10}} + 2\sqrt{5})}{20Lm}$
ω_2^2	$\frac{EA(\sqrt{2}\sqrt{35-8\sqrt{10}} + 5\sqrt{2} + 2\sqrt{5})}{20Lm}$
f_1	$-\frac{\sqrt{10EA}(\sqrt{5\sqrt{2}+2\sqrt{5}-6\sqrt[4]{10}} - \sqrt{5\sqrt{2}+2\sqrt{5}+6\sqrt[4]{10}})}{40\pi\sqrt{Lm}}$
f_2	$\frac{\sqrt{10EA}(\sqrt{5\sqrt{2}+2\sqrt{5}-6\sqrt[4]{10}} + \sqrt{5\sqrt{2}+2\sqrt{5}+6\sqrt[4]{10}})}{40\pi\sqrt{Lm}}$
T_1	$\frac{\pi\sqrt[4]{10}\sqrt{Lm}(\sqrt{5\sqrt{2}+2\sqrt{5}-6\sqrt[4]{10}} + \sqrt{5\sqrt{2}+2\sqrt{5}+6\sqrt[4]{10}})}{3\sqrt{EA}}$
T_2	$-\frac{\pi\sqrt[4]{10}\sqrt{Lm}(\sqrt{5\sqrt{2}+2\sqrt{5}-6\sqrt[4]{10}} - \sqrt{5\sqrt{2}+2\sqrt{5}+6\sqrt[4]{10}})}{3\sqrt{EA}}$
Φ_1	$\begin{bmatrix} \frac{5(5\sqrt{2}EA + 2\sqrt{5}EA - \sqrt{2}EA\sqrt{35-8\sqrt{2}\sqrt{5}})}{EA(25\sqrt{2}-8\sqrt{5})} - \frac{25\sqrt{2}+4\sqrt{5}}{25\sqrt{2}-8\sqrt{5}} \\ 1 \end{bmatrix}$
Φ_2	$\begin{bmatrix} \frac{5(5\sqrt{2}EA + 2\sqrt{5}EA + \sqrt{2}EA\sqrt{35-8\sqrt{2}\sqrt{5}})}{EA(25\sqrt{2}-8\sqrt{5})} - \frac{25\sqrt{2}+4\sqrt{5}}{25\sqrt{2}-8\sqrt{5}} \\ 1 \end{bmatrix}$

Table 6. Dynamic analysis Example 1: Analytical symbolic results.

Validation using numerical results from SAP2000

To validate the symbolic results for the first dynamic analysis example, a numerical comparison is performed between the outcomes of our symbolic analysis and those obtained from SAP2000 for the same truss configuration. Consistency in units is essential, as the symbolic program accepts any unit system, as long as it is internally consistent. For this validation, we adopt the following consistent unit system: meters (m) for length, seconds (s) for time, kilopascals (kPa) for Young's modulus, square meters (m²) for cross-sectional area, and metric tons (Mg) for mass. The symbolic parameters are substituted with the following numerical values:

- Length (L): 4 m
- Axial stiffness (EA): Computed using $E=200\times10^6$ kN/m² (i.e. 200 GPa) and $A=1\times10^{-3}$ m² (20 cm²), giving $EA=2\times10^5$ kN.
- Mass (m): 5 Mg (5000 kg)

Table 7 summarizes the numerical results obtained from the symbolic dynamic analysis of the truss system, following the substitution of symbolic parameters with the above numerical values. It presents key quantities for both the 1st and 2nd eigenmodes of the structure.

Mode	ω_i^2 (s ⁻²)	ω_i (s ⁻¹)	f_i (Hz)	T_i (s)	Φ_i
1 st	3569.1	59.7422	9.5083	0.1052	$\begin{bmatrix} -0.4928 \\ 1 \end{bmatrix}$
2 nd	7974.1	89.2977	14.2122	0.0704	$\begin{bmatrix} 2.0291 \\ 1 \end{bmatrix}$

Table 7. Dynamic analysis Example 1: Numerical results of the symbolic dynamic analysis, obtained by substituting the symbolic parameters with numerical values.

Figure 6 shows the results of the dynamic analysis performed in SAP2000 for the same truss configuration (with identical parameters) as the one analyzed symbolically in MATLAB. The figure presents the deformed shapes corresponding to the 1st and 2nd eigenmodes. As shown in the output window, the periods and natural frequencies match exactly with those reported in Table 7 from the symbolic analysis. This perfect agreement confirms the accuracy of the symbolic computation procedure and validates the correctness of the results obtained using the symbolic MATLAB program.

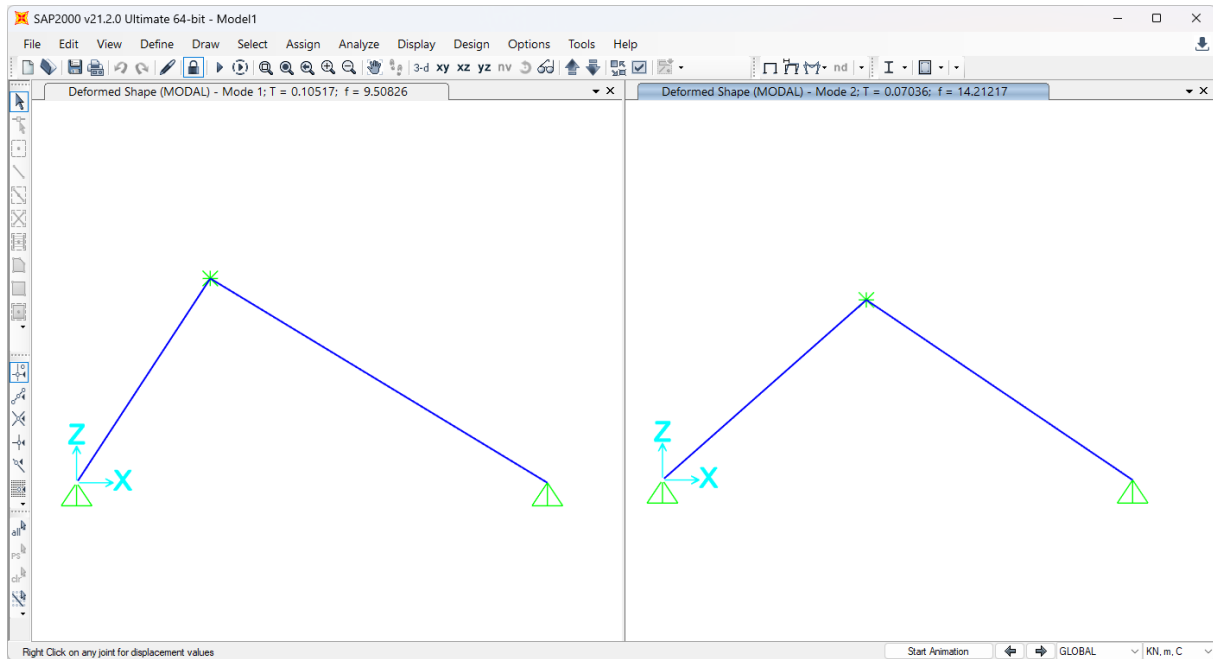


Figure 6. Dynamic analysis Example 1: Results from SAP2000 for the first ($T_1=0.10517$ s) and second ($T_2=0.07036$ s) eigenmodes.

5.2 Dynamic analysis Example 2

The second dynamic analysis example consists of a 5-bar truss with four nodes and three point masses (each of magnitude m), as illustrated in Figure 7. The structure is analyzed symbolically using three key parameters: EA , L , and m .

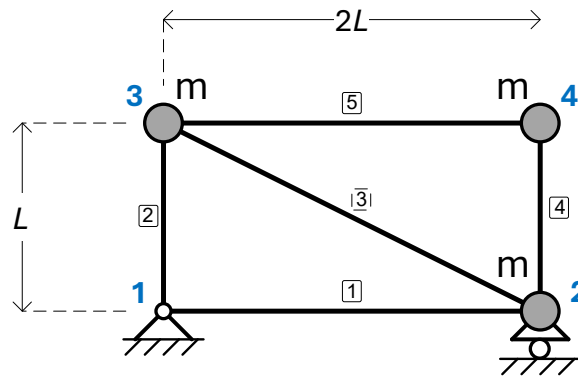


Figure 7. Dynamic analysis Example 2: The truss structure.

Table 8 shows the details of the model, as given in MATLAB.

NodeCoords	$\begin{bmatrix} 0 & 2L & 0 & 2L \\ 0 & 0 & L & L \end{bmatrix}^T$
ElemMatSec	$[EA \quad EA \quad EA \quad EA \quad EA]^T$
ElemCon	$\begin{bmatrix} 1 & 1 & 2 & 2 & 3 \\ 2 & 3 & 3 & 4 & 4 \end{bmatrix}^T$
Supports	$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \end{bmatrix}^T$
Masses	$[m \quad m \quad 0 \quad 0]^T$

Table 8. Dynamic analysis Example 2: Details of the input parameters.

Once again, the symbolic framework successfully computed exact analytical expressions for all key dynamic quantities, including squared angular frequencies ω_i^2 , angular frequencies ω_i , natural frequencies f_i , periods T_i , and normalized mode shapes Φ_i . However, unlike earlier static and simpler dynamic examples, the expressions obtained in this case—although mathematically valid—are exceptionally lengthy and complex. For instance, the symbolic expression for just ω_1^2 spans multiple lines and contains deeply nested radicals, trigonometric functions, and irrational numbers, as shown below:

$$\begin{aligned} \omega_1^2 = & -((15^{1/3}) * EA * (19 * 15^{1/6}) * \sin(\pi/6 - \operatorname{atan}((3 * (10123 - 1189 * 5^{1/2}))^{1/2} * (91 * 396426^{1/2} + \\ & 30 * 1982130^{1/2}))) / 125534900) / 3 * (12998837 - 1633670 * 5^{1/2}))^{1/6} * (233 * 447^{1/2} + 10 * 2235^{1/2})) + \\ & 53789 * 3^{2/3} * 5^{1/6}) * \sin(\operatorname{atan}((25 * 30^{1/2}) * (1189 * 5^{1/2} + 10123)^{1/2}) / (819 * 5^{1/2} + 1350)) / 3 * (1633670 * 5^{1/2} + \\ & 12998837)^{1/6}))) / 322734 - 6 * 5^{1/2} * EA - 25 * EA + (EA * \sin(\pi/3 + \operatorname{atan}((3 * (10123 - \\ & 1189 * 5^{1/2}))^{1/2} * (91 * 396426^{1/2} + 30 * 1982130^{1/2}))) / 125534900) / 3 * (12998837 - \\ & 1633670 * 5^{1/2}))^{1/6} * (50 * 447^{1/2} + 233 * 2235^{1/2})) / 16986 + (15^{1/2} * EA * \sin(\pi/3 + \operatorname{atan}((3 * (10123 - \\ & 1189 * 5^{1/2}))^{1/2} * (91 * 396426^{1/2} + 30 * 1982130^{1/2}))) / 125534900) / 3 * (1633670 * 5^{1/2} + \\ & 12998837)^{1/6}))) / 6) / (50 * L * m) \end{aligned} \quad (5)$$

Similar complexity appears in the expressions for ω_2^2 , ω_3^2 , and their corresponding periods and mode shapes. This illustrates a key limitation of symbolic dynamic analysis: while the symbolic approach works well for static problems of simple 2D trusses—it becomes increasingly impractical for dynamic problems, even in seemingly simple examples like the one considered here. This is not a flaw of the symbolic method itself, but rather a mathematical reality: the eigenvalue problem inherent in free vibration dynamic analysis is significantly more complex than solving a linear system or inverting a stiffness matrix. As the system size grows, or the expressions remain fully symbolic, the resulting formulas rapidly lose elegance and interpretability.

In this example (and the previous dynamic analysis example) we limited the number of symbolic variables to just three (EA , L , m), whereas previous successful static examples handled up to five symbolic variables without issue [4, 5, 29, 30]. Interestingly, some simplicity still emerged: the symbolic expressions for ω_4^2 and ω_5^2 were found to be elegant and compact, showcasing that clarity may still be achievable, but only in certain eigenmodes:

$$\omega_4^2 = \omega_5^2 = \frac{EA}{L} m \quad (6)$$

This example highlights both the potential and the boundaries of symbolic computation in structural dynamics and reinforces the need to balance symbolic generality with computational practicality.

Validation using numerical results from SAP2000

Despite the complexity of the symbolic expressions derived in this example, the symbolic computation procedure remains fully functional and mathematically valid. To verify the accuracy of the results, we performed a numerical validation by comparing the outcomes of our symbolic analysis with those obtained using SAP2000 for the same truss configuration.

For this example, we used the same structural properties as in the first dynamic example (i.e. $L=4$ m, $EA=2 \times 10^5$ kN, $m=5$ Mg). An equivalent truss model was created in SAP2000 and these values were used to define exactly the same truss configuration in both MATLAB and SAP2000.

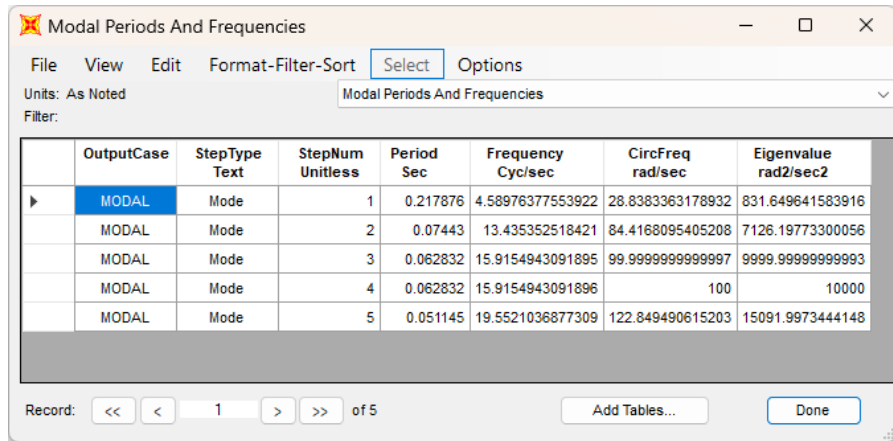
To evaluate numerical values from the symbolic expressions, we used MATLAB's `subs` command to replace the symbolic parameters with their assigned numerical values. We also replicated this process in Microsoft Excel. Specifically, we copied the complex symbolic expressions (e.g., Eq. (5)) from MATLAB and pasted them into Excel, after defining the value of each variable (such as L , EA , m) in separate cells. In doing so, we replaced MATLAB's `pi` with Excel's `PI()` function. This approach enabled straightforward evaluation of the symbolic formulas in Excel, providing fast and accurate numerical results. This also demonstrates that symbolic analysis is not limited to specialized programming environments like MATLAB—it can be extended to widely used tools like Excel, making the method accessible even to users without programming experience.

The results from MATLAB and Excel were identical, confirming that the symbolic expressions are correctly formulated and compatible across different computing environments. From the ω_i^2 values, we derived the corresponding angular frequencies ω_i , natural frequencies f_i , and eigenperiods T_i , as shown in Table 9.

Mode	ω_i^2 (s ⁻²)	ω_i (s ⁻¹)	f_i (Hz)	T_i (s)
1 st	831.6496	28.83834	4.5898	0.2179
2 nd	7126.1977	84.41681	13.4354	0.0744
3 rd	10000	100	15.9155	0.0628
4 th	10000	100	15.9155	0.0628
5 th	15092	122.8495	19.5521	0.0511

Table 9. Dynamic analysis Example 2: Numerical results of the symbolic analysis, obtained by substituting the symbolic parameters with numerical values (for ω_i^2).

Next, we compare these numerical results from symbolic computations with those obtained by SAP2000. The SAP2000 results, shown in Figure 8 and Figure 9 as program screenshots, demonstrate exact agreement for all five eigenmodes. Figure 8 shows the modal periods and frequencies, while Figure 9 shows the first four eigenmodes. These results confirm that, although the symbolic expressions are lengthy and not elegant, the methodology remains valid and robust, producing accurate and reliable symbolic solutions even for more involved dynamic cases.

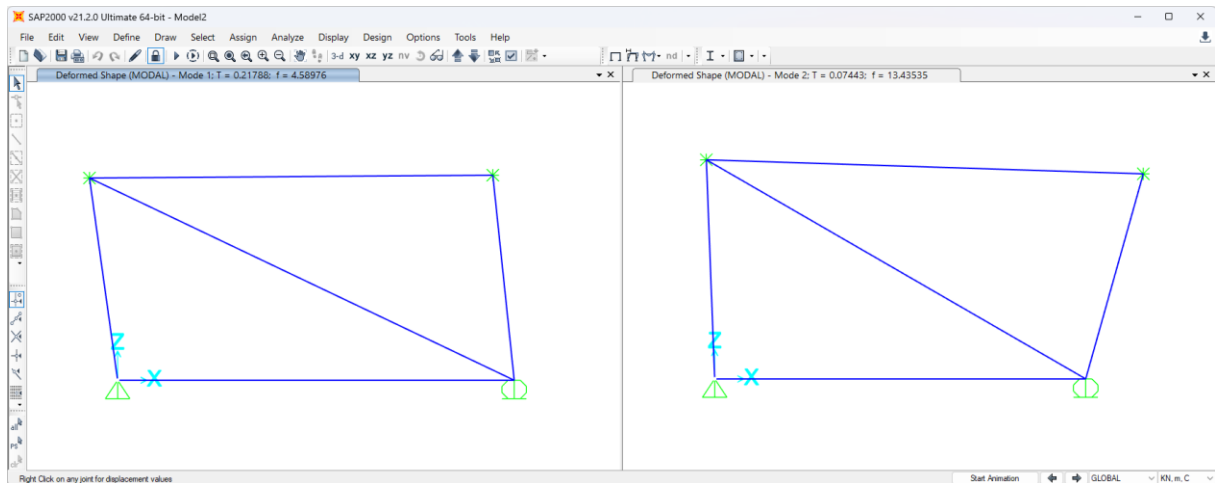


	OutputCase	StepType	StepNum	Unitless	Period Sec	Frequency Cyc/sec	CircFreq rad/sec	Eigenvalue rad2/sec2
▶	MODAL	Mode	1		0.217876	4.58976377553922	28.8383363178932	831.649641583916
	MODAL	Mode	2		0.07443	13.435352518421	84.4168095405208	7126.19773300056
	MODAL	Mode	3		0.062832	15.9154943091895	99.9999999999997	9999.99999999993
	MODAL	Mode	4		0.062832	15.9154943091896	100	10000
	MODAL	Mode	5		0.051145	19.5521036877309	122.849490615203	15091.9973444148

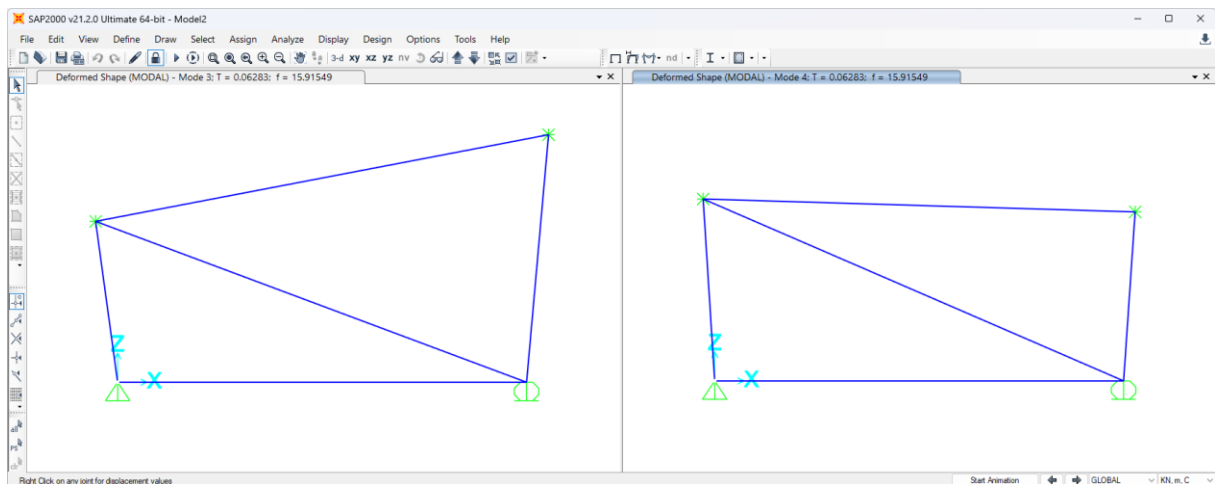
Record: << < 1 > >> of 5

Add Tables... Done

Figure 8. Dynamic analysis Example 2: Model periods and frequencies from SAP2000.



(a)



(b)

Figure 9. Dynamic analysis Example 2: Results from SAP2000.
(a) the 1st and 2nd eigenmodes, (b) the 3rd and 4th eigenmodes.

6 CONCLUSIONS

In previous work, we demonstrated that symbolic Matrix Structural Analysis can be successfully applied to continuous beams [5] and 2D trusses [4], producing analytical expressions using finite element procedures combined with symbolic computation capabilities in MATLAB. These results were both elegant and practical, showing that closed-form solutions can be derived for various static structural problems with impressive clarity and generality.

In the present study, we extended the methodology for plane trusses in two key directions. First, we introduced a new static analysis example. The symbolic procedure worked well, producing valid analytical expressions for displacements, reactions, and axial forces. The results were validated using closed-form solutions from the literature and numerical outputs from Engi-Lab Truss.2D Pro, confirming the accuracy and reliability of the symbolic computations.

Second, and more significantly, we explored the application of symbolic analysis to the dynamic behavior of 2D trusses—specifically the free vibration problem with the computation of eigenvalues, eigenfrequencies, periods, and mode shapes based on nodal masses. Two numerical examples were studied. The first involved a single nodal mass and two effective degrees of freedom. In this case, the symbolic method performed exceptionally well, producing closed-form solutions for all dynamic quantities. Although some expressions were lengthy, they remained interpretable and computationally usable.

The second dynamic example introduced greater complexity with five effective degrees of freedom. Again, the symbolic computation process completed successfully and provided mathematically valid results. However, the expressions became excessively long and complex, limiting their practical use. Nevertheless, we were still able to extract numerical values from these symbolic expressions using both MATLAB and Excel, confirming their consistency. These results were then compared with the corresponding dynamic modal analysis results obtained from a SAP2000 model of the same truss, showing perfect agreement across all five modes. This demonstrates that the symbolic methodology is correct and that the resulting expressions can be evaluated and used in a variety of computational environments.

It is important to note that in more complex dynamic cases, the symbolic expressions generated tend to lose the elegance and simplicity typically preferred for engineering interpretation or manual use. In additional experiments beyond the scope of this paper, we explored more intricate truss configurations and found that MATLAB often failed to produce results in a reasonable time or returned symbolic outputs so lengthy and complex that they offered very limited practical value.

This confirms that while the symbolic approach can produce exact results, its effective application in dynamic analysis is limited to relatively simple truss systems. Unlike static analysis—where symbolic MatSA scales reasonably well for larger models with more symbolic parameters (within certain limits)—the symbolic solution of the eigenvalue problem becomes increasingly infeasible as complexity grows. This is a natural consequence of the increased mathematical difficulty of eigenvalue problems compared to solving linear systems.

In summary, symbolic dynamic modal analysis of 2D trusses is viable and accurate for simple systems, but it faces practical limitations due to expression complexity as the problem scales. The method is most effective when applied to small trusses with a limited number of symbolic variables, where it can serve as both a computational tool and a valuable educational resource.

REFERENCES

- [1] Liu, W.K., S. Li, and H.S. Park, *Eighty Years of the Finite Element Method: Birth, Evolution, and Future*. Archives of Computational Methods in Engineering, 2022. **29**(6): p. 4431-4453 DOI: <https://doi.org/10.1007/s11831-022-09740-9>.
- [2] Plevris, V. and G. Tsiatas, *Computational Structural Engineering: Past Achievements and Future Challenges*. Frontiers in Built Environment, 2018. **4**(21): p. 1-5 DOI: <https://doi.org/10.3389/fbuil.2018.00021>.
- [3] Weaver, W. and J.M. Gere, *Finite-Element Method for Framed Structures*, in *Matrix Analysis of Framed Structures*. 1990, Springer US: Boston, MA. p. 447-467. DOI https://doi.org/10.1007/978-1-4684-7487-9_7.
- [4] Plevris, V. and A. Ahmad, *Deriving analytical solutions using symbolic matrix structural analysis: Part 2 – Plane trusses*. Heliyon, 2025. **11**(4) DOI: <https://doi.org/10.1016/j.heliyon.2025.e42372>.
- [5] Plevris, V. and A. Ahmad, *Deriving analytical solutions using symbolic matrix structural analysis for continuous beams*. Scientific Reports, 2025. **15** DOI: <https://doi.org/10.1038/s41598-025-98023-x>.
- [6] Farahmand-Tabar, S. and K. Aghani, *Chapter 2 - Matrix Analysis of Framed Structures*, in *Practical Programming of Finite Element Procedures for Solids and Structures with MATLAB®*, S. Farahmand-Tabar and K. Aghani, Editors. 2024, Elsevier. p. 29-116. DOI <https://doi.org/10.1016/B978-0-443-15338-9.00007-1>.
- [7] Geisel, A. and F. Svaricek, *A MATLAB Toolbox for Structural Analysis of Linear Systems*. IFAC-PapersOnLine, 2019. **52**(17): p. 7-12 DOI: <https://doi.org/10.1016/j.ifacol.2019.11.018>.
- [8] Yang, B., *Chapter 6 - Static analysis of constrained multispan beams*, in *Stress, Strain, and Structural Dynamics (Second Edition)*, B. Yang, Editor. 2023, Academic Press. p. 183-223. DOI <https://doi.org/10.1016/B978-0-12-818563-6.00018-3>.
- [9] Kattan, P.I., *MATLAB guide to finite elements: An interactive approach*. MATLAB Guide to Finite Elements: An Interactive Approach. 2008. 1-429. DOI: 10.1007/978-3-540-70698-4.
- [10] Papazafeiropoulos, G. and V. Plevris, *OpenSeismoMatlab: New Features, Verification and Charting Future Endeavors*. Buildings, 2024. **14**(1): p. 304 DOI: <https://doi.org/10.3390/buildings14010304>.
- [11] Yang, B., *Chapter 9 - Dynamics of particles and rigid bodies*, in *Stress, Strain, and Structural Dynamics (Second Edition)*, B. Yang, Editor. 2023, Academic Press. p. 351-430. DOI <https://doi.org/10.1016/B978-0-12-818563-6.00012-2>.
- [12] Yang, B., *Chapter 14 - Dynamic analysis of constrained, combined, and stepped beams*, in *Stress, Strain, and Structural Dynamics (Second Edition)*, B. Yang, Editor. 2023, Academic Press. p. 801-872. DOI <https://doi.org/10.1016/B978-0-12-818563-6.00019-5>.
- [13] François, S., et al., *Stabil: An educational Matlab toolbox for static and dynamic structural analysis*. Computer Applications in Engineering Education, 2021. **29**(5): p. 1372-1389 DOI: <https://doi.org/10.1002/cae.22391>.
- [14] Plevris, V. and G. Solorzano, *A Collection of 30 Multidimensional Functions for Global Optimization Benchmarking*. Data, 2022. **7**(4): p. 52 DOI: <https://doi.org/10.3390/data7040046>.
- [15] Andreassen, E., A. Clausen, M. Schevenels, B.S. Lazarov, and O. Sigmund, *Efficient topology optimization in MATLAB using 88 lines of code*. Structural and

- Multidisciplinary Optimization, 2011. **43**(1): p. 1-16 DOI: <https://doi.org/https://doi.org/10.1007/s00158-010-0594-7>.
- [16] Sigmund, O., *A 99 line topology optimization code written in Matlab*. Structural and Multidisciplinary Optimization, 2001. **21**(2): p. 120-127 DOI: <https://doi.org/10.1007/s001580050176>.
- [17] Solorzano, G. and V. Plevris, *Optimum Design of RC Footings with Genetic Algorithms According to ACI 318-19*. Buildings, 2020. **10**(6): p. 1-17 DOI: <https://doi.org/10.3390/buildings10060110>.
- [18] Arora, J.S., *Chapter 7 - Optimum Design with MATLAB®*, in *Introduction to Optimum Design (Third Edition)*, J.S. Arora, Editor. 2012, Academic Press: Boston. p. 275-298. DOI <https://doi.org/10.1016/B978-0-12-381375-6.00007-3>.
- [19] Ferreira, A.J.M. and N. Fantuzzi, *Bernoulli 2D Frames*, in *MATLAB Codes for Finite Element Analysis: Solids and Structures*. 2020, Springer International Publishing: Cham. p. 105-121. DOI https://doi.org/10.1007/978-3-030-47952-7_7.
- [20] Plevris, V., A. Jiménez Rios, and U.A. Ebead. *Exploring the Predictive Performance of Simple Regression Models and ANN in 2D Truss Analysis*. in *Proceedings of the The 1st International Conference on Net-Zero Built Environment: Innovations in Materials, Structures, and Management Practices (Net-Zero Future 2024)*. 2024. Oslo, Norway: Springer Nature Switzerland. p. 1473-1485. DOI: https://doi.org/10.1007/978-3-031-69626-8_123.
- [21] MathWorks, *Symbolic Math Toolbox™ User's Guide*. 2024 DOI. Available from: https://www.mathworks.com/help/pdf_doc/symbolic/symbolic_ug.pdf (Accessed
- [22] Lynch, S., *A Tutorial Introduction to MATLAB and the Symbolic Math Toolbox*, in *Dynamical Systems with Applications using MATLAB®*. 2004, Birkhäuser Boston: Boston, MA. p. 1-13. DOI https://doi.org/10.1007/978-0-8176-8156-2_1.
- [23] Attaway, S., *Chapter 14 - Advanced Mathematics*, in *MATLAB (Fifth Edition)*, S. Attaway, Editor. 2019, Butterworth-Heinemann. p. 527-581. DOI <https://doi.org/10.1016/B978-0-12-815479-3.00014-3>.
- [24] Valentine, D.T. and B.D. Hahn, *Chapter 17 - Symbolics toolbox*, in *Essential MATLAB for Engineers and Scientists (Eighth Edition)*, D.T. Valentine and B.D. Hahn, Editors. 2023, Academic Press. p. 351-368. DOI <https://doi.org/10.1016/B978-0-32-399548-1.00024-2>.
- [25] Asadi, F., *Symbolic Calculations in MATLAB®*, in *Applied Numerical Analysis with MATLAB®/Simulink®: For Engineers and Scientists*, F. Asadi, Editor. 2023, Springer Nature Switzerland: Cham. p. 55-71. DOI https://doi.org/10.1007/978-3-031-19366-8_2.
- [26] Mureşan, M., *Introduction to Mathematica® with Applications*. 2017: Springer Cham, ISBN: 978-3-319-52003-2. DOI: 10.1007/978-3-319-52003-2.
- [27] Smith, E., *Maple*, in *Introduction to the Tools of Scientific Computing*. 2022, Springer International Publishing: Cham. p. 219-244. DOI https://doi.org/10.1007/978-3-031-16972-4_10.
- [28] Cywiak, M. and D. Cywiak, *SymPy*, in *Multi-Platform Graphics Programming with Kivy: Basic Analytical Programming for 2D, 3D, and Stereoscopic Design*. 2021, Apress: Berkeley, CA. p. 173-190. DOI https://doi.org/10.1007/978-1-4842-7113-1_11.
- [29] Plevris, V. and A. Ahmad, *Deriving Analytical Solutions Using Symbolic Matrix Structural Analysis: Part I - Continuous Beams*. ArXiv e-prints, 2024(arXiv:2411.03514) DOI: <https://doi.org/10.48550/arXiv.2411.03514>.

- [30] Plevris, V. and A. Ahmad, *Deriving Analytical Solutions Using Symbolic Matrix Structural Analysis: Part 2 - Plane Trusses*. ArXiv e-prints, 2024(arXiv:2411.16573) DOI: <https://doi.org/10.48550/arXiv.2411.16573>.
- [31] Tinkov, D.V., *Comparative analysis of analytical solutions to the problem of truss structure deflection*. Magazine of Civil Engineering, 2015. **57**(5): p. 66-73 DOI: <https://doi.org/10.5862/MCE.57.6>.