FrameAnalysisGUI

User Manual

2D Frame Analysis

Matrix Structural Analysis Method

---

doc. Ing. Özgür Yurdakul, Ph.D.

Faculty of Transport Engineering

University of Pardubice

MATLAB-based 2D frame analysis software

Contents

1   Introduction

`FrameAnalysisGUI` is a MATLAB graphical user interface for linear–elastic static analysis of 2D frame structures using the **direct stiffness method** (matrix structural analysis). The program supports:

- Arbitrary 2D frame geometry (beams, frames, trusses)
- Four element end–release conditions (fixed–fixed, hinge at start/end, both hinged)
- Three member load types (concentrated force, concentrated moment, UDL)
- Nodal loads (forces and moments)
- Multiple material/section property sets
- Graphical output: deformed shape, reaction forces, axial/shear/moment diagrams
- Embedded structure preview with real–time visualization
- Export of all analysis variables to the MATLAB workspace

1.1   Requirements

- MATLAB R2020b or later (App Designer / `uifigure` support required)
- No additional toolboxes required

1.2   Getting Started

To launch the GUI, type in the MATLAB Command Window:

```
>> FrameAnalysisGUI
```

The main window (1400×720 pixels) will open with default input data for a cantilever beam example.

2   Sign Convention

> **Warning**
>
> Understanding the sign convention is essential for correct input and interpretation of results.

2.1   Global Coordinate System

- **X–axis**: Horizontal, **positive to the right**
- **Y–axis**: Vertical, **positive downward**
- **Moments**: **Clockwise (CW) positive**

2.2   Local Element Coordinate System

Each element has a local coordinate system:
- Local x–axis: from start node (i) to end node (j)
- Local y–axis: perpendicular to x, following the right–hand rule
- Local DOFs per node: u (axial), v (transverse), θ (rotation)

2.3   Applied Loads

- $F_x > 0$: Force in the positive X–direction (rightward)
- $F_y > 0$: Force in the positive Y–direction (downward)
- $M_z > 0$: Moment in the clockwise direction
- Member loads (Type 1 & 3): Positive magnitude = downward in local coordinates

3   Unit System

The program uses a consistent unit system throughout. All input must be provided in the following units:

Quantity	Unit	Symbol
Length / Coordinates	metre	m
Cross–sectional area	square metre	m^2
Moment of inertia	metre to the fourth	m^4
Elastic modulus	kilopascal	kPa $(= kN/m^2)$
Nodal forces	kilonewton	kN
Nodal moments	kilonewton–metre	kN·m
Concentrated member force	kilonewton	kN
Concentrated member moment	kilonewton–metre	kN·m
Distributed load (UDL)	kilonewton per metre	kN/m
Output quantities:		
Displacements	metre	m
Rotations	radian	rad
Internal forces (N, V)	kilonewton	kN
Internal moments (M)	kilonewton–metre	kN·m
Reactions (R_x, R_y)	kilonewton	kN
Reaction moments	kilonewton–metre	kN·m

4   GUI Layout Overview

The main window is divided into three areas:
1. **Left panel** (520×630 px): Six input tabs for defining the structural model
2. **Right panel** (850×630 px): Embedded structure preview plot
3. **Bottom panel**: Action buttons, deformation scale factor, and status bar

4.1   Bottom Panel Buttons

Button	Colour	Function
Update Plot	Blue	Refreshes the embedded structure preview with current input data
Analyze	Green	Runs the full structural analysis. Automatically closes any previous analysis figures before creating new ones
Clear Preview	Red	Clears the embedded preview plot
Close Figures	Grey	Closes all external analysis figure windows
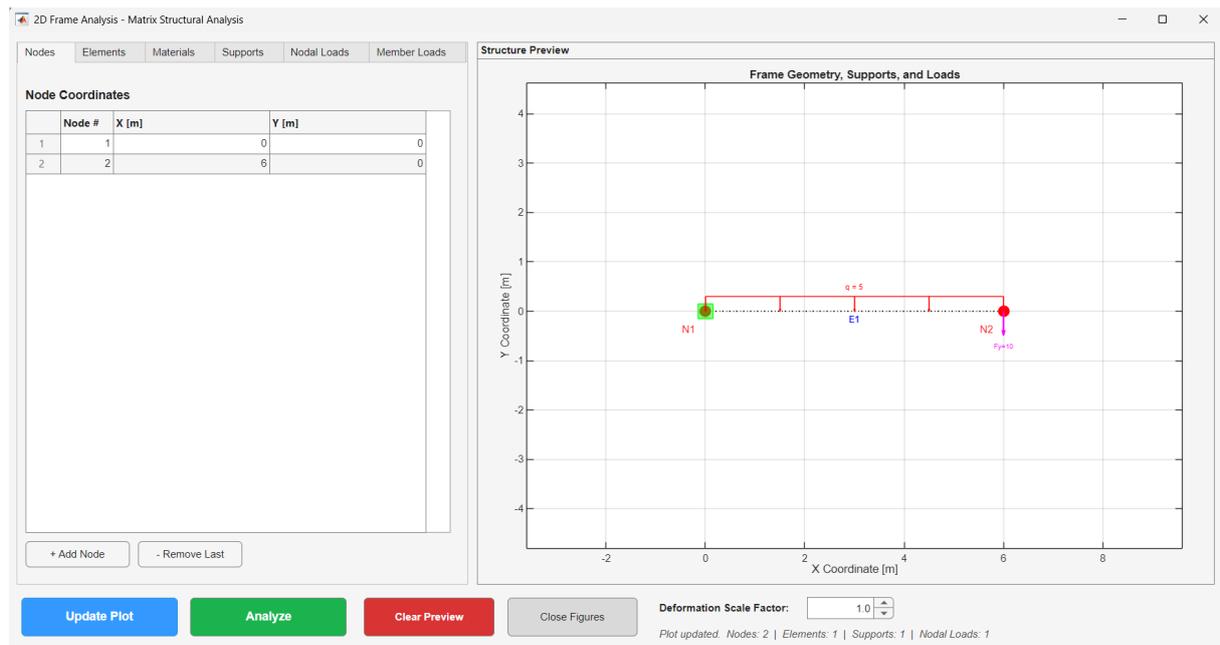
Figure 1: Main GUI window showing the Nodes tab (left) and structure preview (right) with the default cantilever beam example.

The **Deformation Scale Factor** spinner (default = 1.0) controls the magnification of the deformed shape plot.

5   Input Tabs

5.1   Tab 1: Nodes

Defines the node coordinates of the structure.

Column	Unit	Description
Node #	–	Auto–numbered (read–only)
X [m]	m	X–coordinate of the node
Y [m]	m	Y–coordinate of the node

- Use **+ Add Node** to add a new row (auto–numbered)
- Use **- Remove Last** to delete the last row

5.2   Tab 2: Elements

Defines the element connectivity, property assignment, and end release conditions.

Column	Values	Description
Elem #	–	Auto–numbered (read–only)
Start Node	integer	Node number at start (i–end)
End Node	integer	Node number at end (j–end)
Prop ID	integer	Material/section property set ID (from Tab 3)
Release (0/1/2/3)	0–3	End release condition (see below)
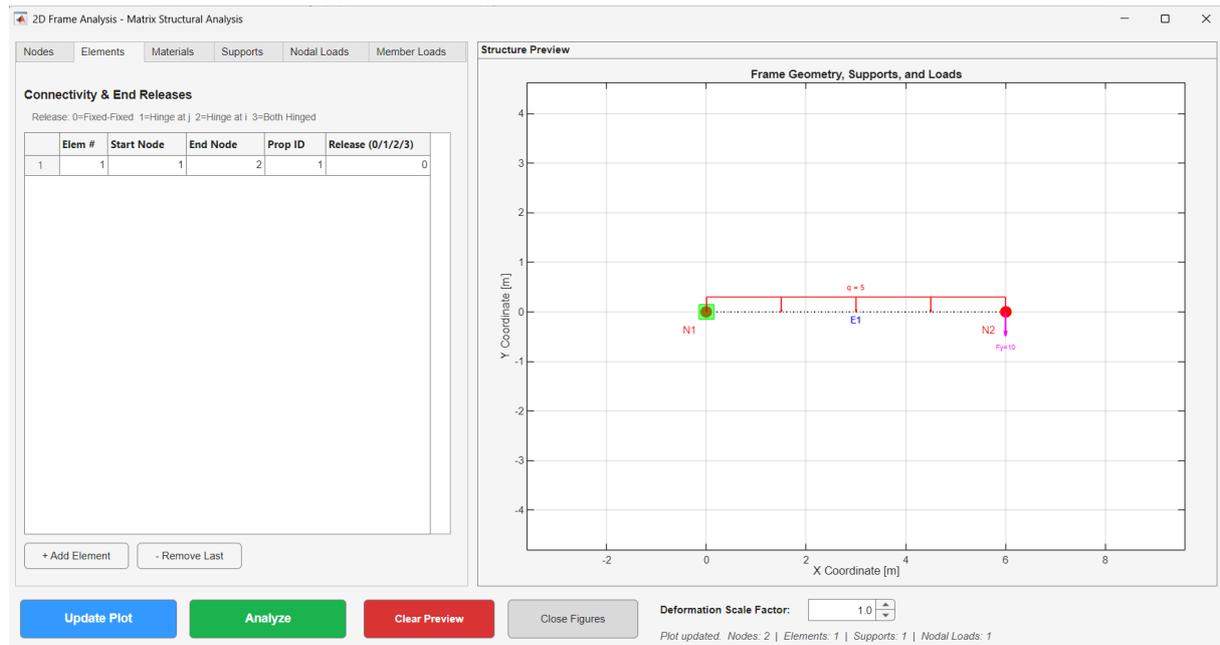
Figure 2: Elements tab showing connectivity, property assignment, and the Release column for end release conditions.

5.2.1 End Release Types

Code	Type	Description
0	Fixed–Fixed	Standard frame element. Both ends transfer axial force, shear, and moment.
1	Hinge at j	Moment release at the end node. No moment transfer at j; θ_j is free.
2	Hinge at i	Moment release at the start node. No moment transfer at i; θ_i is free.
3	Both Hinged	Truss–like element. No moment transfer at either end. Only axial force is transmitted.

> **Tip**
>
> Hinges are visualized as blue circles in the structure preview plot. When two elements share a node and both release rotation there (e.g., Element 1 has hinge at j and Element 2 has hinge at i, both at the same node), the program automatically detects and constrains the resulting zero–stiffness rotational DOF.

5.3 Tab 3: Materials

Defines material and cross–section properties. Multiple property sets can be defined and assigned to different elements via the `Prop ID` column in Tab 2.

Column	Unit	Description
Prop ID	–	Auto–numbered (read–only)
Area [m^2]	m^2	Cross–sectional area A
Inertia [m^4]	m^4	Second moment of area I
Modulus [kPa]	kPa	Elastic modulus E

5.4 Tab 4: Supports

Defines boundary conditions at specific nodes. Each DOF can be independently restrained (1) or free (0).

Column	Values	Description
Node	integer	Node number where support is applied
Dx (1/0)	0 or 1	Restrain horizontal displacement (D_x)
Dy (1/0)	0 or 1	Restrain vertical displacement (D_y)
Rz (1/0)	0 or 1	Restrain rotation (θ_z)

Common support types:

Support Type	Dx	Dy	Rz	Preview Symbol
Fixed (Encastré)	1	1	1	Green square
Pinned	1	1	0	Green triangle
Roller (vertical)	0	1	0	Blue triangle + circle
Roller (horizontal)	1	0	0	Cyan triangle + circle

5.5 Tab 5: Nodal Loads

Defines concentrated forces and moments applied directly at nodes.

Column	Unit	Description
Node	integer	Node where load is applied
Fx [kN]	kN	Horizontal force (positive = rightward)
Fy [kN]	kN	Vertical force (positive = downward)
Mz [kN·m]	kN·m	Moment (positive = clockwise)

Multiple loads at the same node are supported — simply add multiple rows with the same node number.

5.6 Tab 6: Member Loads

Defines loads applied along elements (between nodes). Three load types are supported.

Column	Unit	Description
Elem ID	integer	Element on which the load acts
Load Type (1/2/3)	1, 2, or 3	Type of member load (see below)
Magnitude	varies	Load value (see table below for units)
a_param [m]	m	Load position or loaded length

5.6.1 Load Type Details

Type	Description	Unit	a_param	Notes
1	Conc. vertical force	kN	Dist. from start [m]	Must satisfy $0 < a < L$
2	Conc. moment	kN·m	Dist. from start [m]	Must satisfy $0 < a < L$
3	Uniform dist. load	kN/m	Loaded length [m]	$a = 0 \Rightarrow$ full span
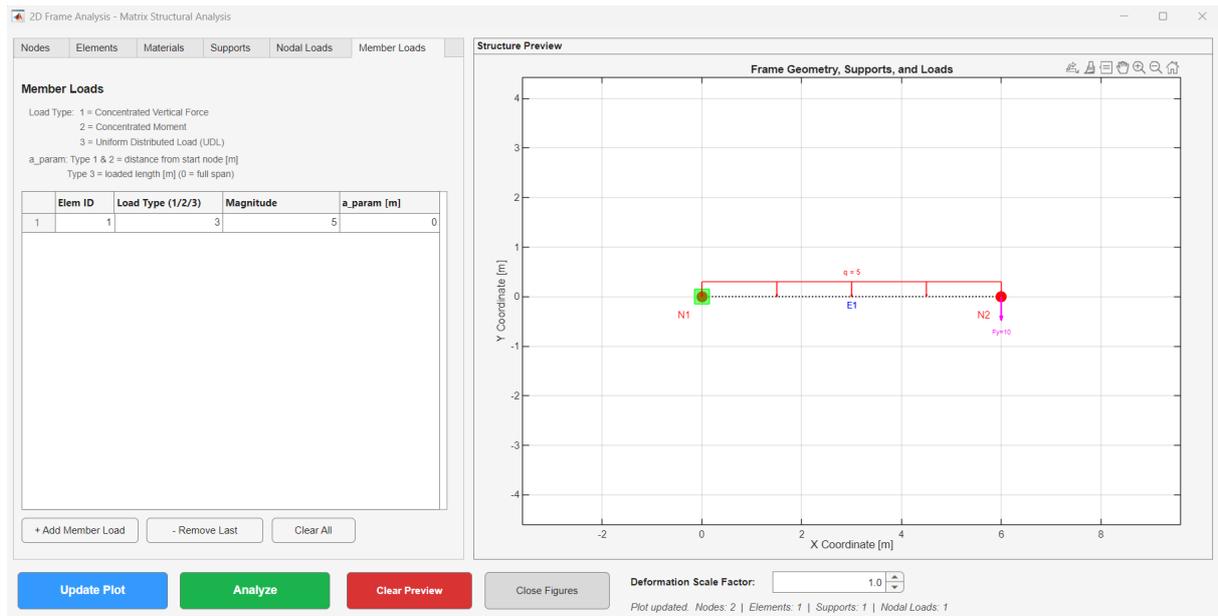
Figure 3: Member Loads tab showing load type descriptions and parameter definitions.

> **Warning**
>
> For load types 1 and 2, the `a_param` must be strictly between 0 and L (exclusive). Setting $a = 0$ or $a = L$ will produce a warning and skip the load.

6 Structure Preview

The right panel displays an embedded plot of the structural model. Click `Update Plot` to refresh. The preview shows:

- **Elements**: Black dotted lines with blue labels (E1, E2, …)
- **Nodes**: Red filled circles with red labels (N1, N2, …)
- **Supports**: Coloured symbols (see Tab 4)
- **Hinges**: Blue open circles at released element ends
- **Nodal loads**: Magenta arrows (forces) and arcs (moments)
- **Member loads**: Red arrows (point loads), red arcs (moments), or red distributed arrows with connecting line (UDL)

7 Analysis

7.1 Running the Analysis

Click the green `Analyze` button. The program will:

1. Close any previously opened analysis figures
2. Extract all input data from the GUI tables
3. Assemble the global stiffness matrix with appropriate end releases
4. Compute fixed–end moment vectors for each element and release type
5. Detect and constrain zero–stiffness DOFs (internal hinge nodes)
6. Solve $\mathbf{K}\,\mathbf{D} = \mathbf{F}$
7. Clean floating–point noise from results

8. Compute element local forces, reactions, and internal force distributions
9. Generate five analysis figure windows
10. Print all results to the MATLAB Command Window
11. Export all variables to the MATLAB base workspace

7.2 Output Figures

Five separate figure windows are generated:

Fig.	Title	Content
1	Deformed Shape	Original (dashed) and deformed (red) shape with cubic Hermite interpolation. Scale controlled by the Deformation Scale Factor spinner.
2	Reaction Forces	Green arrows showing support reactions (R_x, R_y, M_z) with numerical labels.
3	Axial Force Diagram (N)	Blue filled diagram showing axial force distribution along each element.
4	Shear Force Diagram (V)	Red filled diagram showing shear force variation, including parabolic distribution under UDL.
5	Bending Moment Diagram (M_z)	Green filled diagram showing bending moment distribution, including parabolic/cubic shapes under distributed/point loads.

7.3 Command Window Output

The following tables are printed to the MATLAB Command Window:
0. **Global Stiffness Matrix K** — Full reduced stiffness matrix with DOF labels
1. **Global Nodal Displacements** — D_x, D_y, θ_z for each node
2. **Member Local Displacements** — u, v, θ at start and end of each element
3. **Member Local Forces** — F_x, F_y, M_z at start and end of each element
4. **Support Reactions** — R_x, R_y, M_z at each support node

8   Workspace Variables

After analysis, the following variables are exported to the MATLAB base workspace via `assignin`:

Variable	Size	Description
K	$N_{eq} \times N_{eq}$	Reduced global stiffness matrix
D	$N_{eq} \times 1$	Displacement solution vector
F	$N_{eq} \times 1$	Load vector
E_map	$N_{node} \times 3$	Equation number map (0 = restrained)
z_ElementData	$1 \times N_{elem}$ struct	Element data (see below)
z_Results	struct	Analysis results (see below)
NodalDisp	$N_{node} \times 3$	Nodal displacements $[D_x, D_y, \theta_z]$
ForceTable	$N_{elem} \times 7$	Element local forces $[\text{ID}, f_1 \ldots f_6]$
DispTable	$N_{elem} \times 7$	Element local displacements $[\text{ID}, d_1 \ldots d_6]$
ReactionTable	$N_{sup} \times 4$	Reactions $[\text{Node}, R_x, R_y, M_z]$
XY	$N_{node} \times 2$	Node coordinates
C	$N_{elem} \times 4$	Connectivity + property + release

8.1 z_ElementData Structure Fields

Each element z_ElementData(i) contains:

Field	Size	Description
.k	6×6	Local stiffness matrix (with releases applied)
.T	6×6	Transformation matrix (local $\rightarrow$ global)
.P	1×6	DOF mapping vector (0 = restrained)
.L	scalar	Element length [m]
.ang	scalar	Element angle [rad]
.fem	6×1	Local fixed–end force vector
.fem_global	6×1	Global fixed–end force vector
.k_glob	6×6	Global stiffness matrix
.K_global	$N_{eq} \times N_{eq}$	Assembled global stiffness matrix
.releaseType	scalar	End release code (0/1/2/3)

9 Theory

9.1 Element Stiffness Matrices

Four stiffness matrices are implemented, corresponding to different end conditions. Let:

$$k_1 = \frac{EA}{L}, \qquad \alpha = \frac{EI}{L^3}, \qquad \beta = \frac{EI}{L^2}, \qquad \gamma = \frac{EI}{L}$$

9.1.1 Type 0: Fixed–Fixed (Standard Frame Element)

$$\mathbf{k}^{(0)} = \begin{bmatrix} k_1 & 0 & 0 & -k_1 & 0 & 0 \\ 0 & 12\alpha & -6\beta & 0 & -12\alpha & -6\beta \\ 0 & -6\beta & 4\gamma & 0 & 6\beta & 2\gamma \\ -k_1 & 0 & 0 & k_1 & 0 & 0 \\ 0 & -12\alpha & 6\beta & 0 & 12\alpha & 6\beta \\ 0 & -6\beta & 2\gamma & 0 & 6\beta & 4\gamma \end{bmatrix}$$

9.1.2 Type 1: Hinge at j (θ_j Released)

Obtained by static condensation of DOF 6 from the fixed–fixed matrix:

$$\mathbf{k}^{(1)} = \begin{bmatrix} k_1 & 0 & 0 & -k_1 & 0 & 0 \\ 0 & 3\alpha & -3\beta & 0 & -3\alpha & 0 \\ 0 & -3\beta & 3\gamma & 0 & 3\beta & 0 \\ -k_1 & 0 & 0 & k_1 & 0 & 0 \\ 0 & -3\alpha & 3\beta & 0 & 3\alpha & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

9.1.3 Type 2: Hinge at i (θ_i Released)

Obtained by static condensation of DOF 3:

$$\mathbf{k}^{(2)} = \begin{bmatrix} k_1 & 0 & 0 & -k_1 & 0 & 0 \\ 0 & 3\alpha & 0 & 0 & -3\alpha & -3\beta \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -k_1 & 0 & 0 & k_1 & 0 & 0 \\ 0 & -3\alpha & 0 & 0 & 3\alpha & 3\beta \\ 0 & -3\beta & 0 & 0 & 3\beta & 3\gamma \end{bmatrix}$$

9.1.4 Type 3: Both Hinged (Truss Element)

$$\mathbf{k}^{(3)} = \begin{bmatrix} k_1 & 0 & 0 & -k_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -k_1 & 0 & 0 & k_1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

9.1.5 Summary of Key Stiffness Coefficients

Release	$k_{22} = k_{55}$	k_{23} or k_{26}	k_{33} or k_{66}	k_{36}
0 (Fixed–Fixed)	12α	-6β	4γ	2γ
1 (Hinge at j)	3α	-3β (at i), 0 (at j)	3γ (at i), 0	0
2 (Hinge at i)	3α	0 (at i), -3β (at j)	0, 3γ (at j)	0
3 (Both Hinged)	0	0	0	0

DOF ordering: $[u_i,\ v_i,\ \theta_i,\ u_j,\ v_j,\ \theta_j]$

9.2 Fixed–End Forces

Fixed–end force (FEM) vectors are provided for each combination of release type (0–3) and load type (1–3). The vectors account for the sign convention (Y–down positive, CW moments positive).

For brevity, only the Type 0 (fixed–fixed) FEM vectors are listed here. Type 1, 2, and 3 vectors are derived by static condensation.

9.2.1 Load Type 1: Concentrated Vertical Force F_z at Distance a from Node i

Let $b = L - a$:

$$f_2 = -\frac{b^2(3L - 2b)}{L^3} F_z \qquad\qquad f_3 = \frac{ab^2}{L^2} F_z$$

$$f_5 = -\frac{a^2(3L - 2a)}{L^3} F_z \qquad\qquad f_6 = -\frac{a^2 b}{L^2} F_z$$

9.2.2 Load Type 2: Concentrated Moment M at Distance a from Node i

$$f_2 = -\frac{6ab}{L^3} M \qquad\qquad f_3 = \frac{b(2L - 3b)}{L^2} M$$

$$f_5 = \frac{6ab}{L^3} M \qquad\qquad f_6 = \frac{a(2L - 3a)}{L^2} M$$

9.2.3 Load Type 3: Uniform Distributed Load q over Length a from Node i

($a = L$ for full span; set `a_param = 0` in the GUI):

$$f_2 = -\frac{q(2aL^3 - 2a^3 L + a^4)}{2L^3} \qquad\qquad f_3 = \frac{q(6a^2 L^2 - 8a^3 L + 3a^4)}{12L^2}$$

$$f_5 = -\frac{q(2a^3 L - a^4)}{2L^3} \qquad\qquad f_6 = -\frac{q(4a^3 L - 3a^4)}{12L^2}$$

9.3 Internal Hinge Detection

When multiple elements share a node and all release their rotation at that node, the global stiffness matrix has a zero diagonal entry for that rotational DOF. The program detects this condition after assembly and applies a penalty constraint:

$$K_{ii} = 10^4 \cdot \max(\mathrm{diag}(\mathbf{K})), \qquad F_i = 0$$

This effectively constrains the free rotation DOF while maintaining numerical conditioning.

9.4 Floating–Point Noise Cleanup

After solving, the program applies relative–tolerance thresholding ($10^{-10} \times$ max value) to:
- Displacement vector $\mathbf{D}$
- Nodal displacements
- Element local forces and displacements
- Support reactions

9.5 Deformed Shape Computation

The deformed shape is plotted by interpolating the transverse displacement along each element using **cubic Hermite shape functions** combined with **particular solutions** for member loads. Each element is discretised into 500 evaluation points.

9.5.1 Homogeneous Solution (Hermite Interpolation)

Given the local end displacements v_i, θ_i (start) and v_j, θ_j (end), the transverse displacement at a local coordinate x along the element ($0 \leq x \leq L$) is:

$$v_{\mathrm{homo}}(x) = H_1(\eta)\, v_i + H_2(x, \eta)\, \theta_i + H_3(\eta)\, v_j + H_4(x, \eta)\, \theta_j$$

where $\eta = x/L$ and the four Hermite shape functions are:

$$H_1(\eta) = 1 - 3\eta^2 + 2\eta^3 \qquad\qquad H_2(x,\eta) = x\left(1 - 2\eta + \eta^2\right)$$
$$H_3(\eta) = 3\eta^2 - 2\eta^3 \qquad\qquad H_4(x,\eta) = x\left(\eta^2 - \eta\right)$$

The axial displacement is linearly interpolated: $u(x) = u_i + (u_j - u_i)\,x/L$.

9.5.2   Particular Solutions for Member Loads

For fixed–fixed elements (Release 0) carrying member loads, a particular solution $v_{\text{part}}(x)$ is added to capture the curved deformation between nodes.

Concentrated Force P at Distance a (Type 1):

$$v_{\text{part}}(x) = \begin{cases} \dfrac{Pb^2 x^2}{6EIL^3}\left(3aL - 3ax - bx\right) & \text{if } x \le a \\[2ex] \dfrac{Pa^2(L-x)^2}{6EIL^3}\left(3bL - 3b(L-x) - a(L-x)\right) & \text{if } x > a \end{cases}$$

where $b = L - a$.

Full–Span UDL q (Type 3):

$$v_{\text{part}}(x) = \frac{q\,x^2(L-x)^2}{24EI}$$

9.5.3   Transformation to Global Coordinates

The total displacement $v(x) = v_{\text{homo}}(x) + v_{\text{part}}(x)$ in local coordinates is transformed to global coordinates using the element angle α:

$$X_{\text{def}} = X_i + \left(x + u(x)\cdot s\right)\cos\alpha - v(x)\cdot s\cdot\sin\alpha$$
$$Y_{\text{def}} = Y_i + \left(x + u(x)\cdot s\right)\sin\alpha + v(x)\cdot s\cdot\cos\alpha$$

where s is the user–specified deformation scale factor. The endpoint coordinates are snapped to the exact displaced node positions to ensure continuity between elements.

10   Using MATLAB Online

Users who do not have MATLAB installed on their computer can run `FrameAnalysisGUI` entirely in a web browser using **MATLAB Online**. No installation or setup is required.

10.1   What is MATLAB Online?

MATLAB Online provides the full MATLAB desktop environment inside a web browser. It supports `uifigure`–based apps, the Command Window, the Workspace browser, and figure windows — everything needed to run `FrameAnalysisGUI`. Files are stored in MATLAB Drive (cloud storage linked to your MathWorks account).
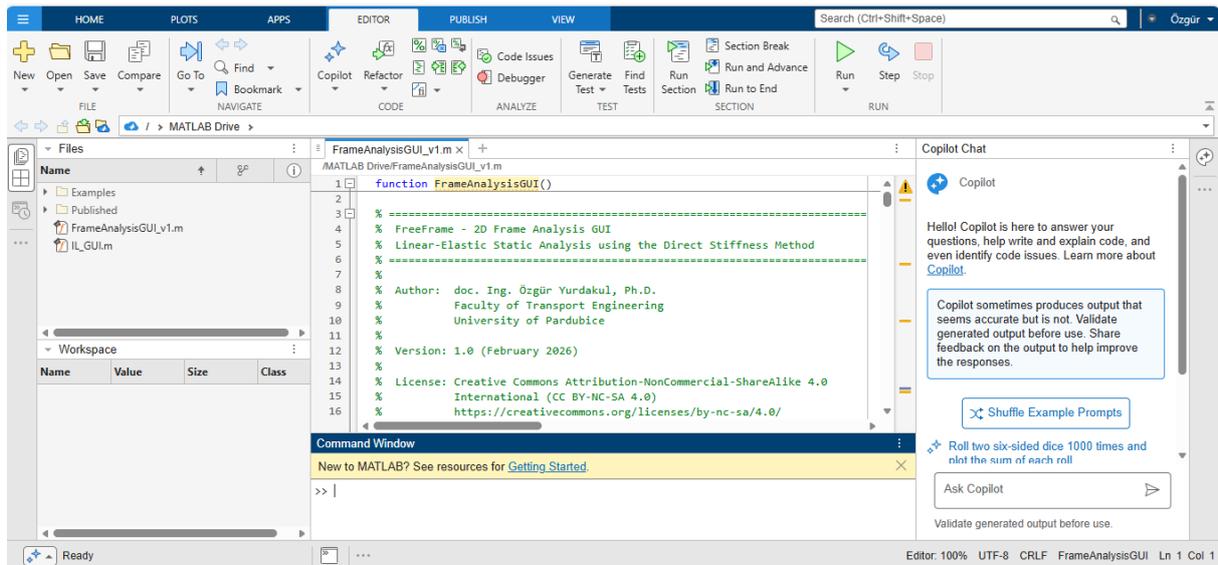
Figure 4: Layout of the MATLAB Online interface in a web browser, showing the main panels used when running `FrameAnalysisGUI`.

10.2   Accessing MATLAB Online

10.2.1   Option A: Institutional Campus–Wide License (Recommended)

Many universities and institutions provide campus–wide MATLAB licences that give their users **unlimited access** to MATLAB Online and all toolboxes.
1. Go to https://matlab.mathworks.com
2. Click **Sign in to get started**
3. Enter your **university or work e–mail address**
4. If prompted, sign in with your institutional credentials (SSO)
5. If you do not yet have a MathWorks account, create one using your institutional e–mail — the campus licence will be automatically linked
6. MATLAB Online will open in your browser — you are ready to work

> **Tip**
>
> Check whether your institution has a campus–wide licence at:
> https://www.mathworks.com/academia/tah-support-program/eligibility.html

10.2.2   Option B: Basic MATLAB Online

If your institution does not have a campus–wide licence, a basic tier is available:
1. Go to https://www.mathworks.com/products/matlab-online.html
2. Create a MathWorks account using your **work or school e–mail address**
3. Access MATLAB Online with **20 hours per month** of compute time and **5 GB** of MATLAB Drive storage

> **Note**
>
> The basic tier is sufficient for running `FrameAnalysisGUI`, as each analysis session typically takes only a few minutes.

10.3   Uploading and Running the Program

Once MATLAB Online is open in your browser:

1. **Upload the file:** In the *Current Folder* panel (left side), click the **Upload** button (cloud icon with an arrow) and select `FrameAnalysisGUI.m` from your computer. The file will appear in your MATLAB Drive.

2. **Run the program:** Type in the Command Window:

   ```
   >> FrameAnalysisGUI
   ```

   The GUI window will open inside the browser. All tabs, buttons, and the embedded preview plot work identically to the desktop version.

3. **Work with the GUI:** Enter your structural model data in the input tabs, click `Update Plot` to preview, and click `Analyze` to run the analysis. Results are printed in the Command Window and analysis figures open as separate tabs within the browser.

4. **View analysis figures:** The five output figures (deformed shape, reactions, N/V/M diagrams) open as figure tabs in MATLAB Online. You can switch between them using the tabs at the top of the figure area. To save a figure, right–click on it and select *Save As*.

5. **Access workspace variables:** After analysis, all exported variables (`K`, `D`, `F`, `NodalDisp`, etc.) are visible in the *Workspace* panel and can be inspected by double–clicking.

10.4   Tips

- **Save your work:** Files in MATLAB Drive persist between sessions. You can also download files to your computer at any time.
- **Browser compatibility:** MATLAB Online works best in Google Chrome, Mozilla Firefox, Microsoft Edge, or Safari. Use a recent version for the best experience.
- **Screen size:** The GUI is designed for a 1400×720 px window. On smaller screens, you may need to scroll or zoom out in the browser (e.g. `Ctrl+-`).
- **MATLAB Drive sync:** Install the free *MATLAB Drive Connector* on your computer to automatically synchronise files between your local machine and MATLAB Online.
- **Mobile access:** MATLAB Online also works on tablets, though the GUI layout is optimised for desktop–sized screens.
- **Sharing:** You can share your `.m` file or results with others by sharing your MATLAB Drive folder or by downloading and sending the files.

> **Warning**
>
> MATLAB Online requires a stable internet connection. If the connection drops during analysis, unsaved Command Window output may be lost. The GUI state and uploaded files are preserved in MATLAB Drive.

11   Examples

11.1   Example 1: Cantilever Beam (Default)

The default input defines a cantilever beam:
- Nodes: N1 $(0,0)$, N2 $(6,0)$ — span $L = 6$ m
- Element: E1 from N1 to N2, Release $= 0$ (fixed–fixed)

- Properties: $A = 0.15$ m^2, $I = 4.5 \times 10^{-3}$ m^4, $E = 20 \times 10^6$ kPa
- Support: Fixed at N1 ($D_x = D_y = R_z = 1$)
- Nodal load: $F_y = 10$ kN at N2 (downward)
- Member load: UDL $q = 5$ kN/m on E1 (full span)

Expected results (combined loading):

- Tip displacement D_y at N2 from point load: $\dfrac{FL^3}{3EI} = \dfrac{10 \times 6^3}{3 \times 90{,}000} = 8.0 \times 10^{-3}$ m

- Tip displacement D_y at N2 from UDL: $\dfrac{qL^4}{8EI} = \dfrac{5 \times 6^4}{8 \times 90{,}000} = 9.0 \times 10^{-3}$ m

- Fixed–end reaction R_y: $10 + 5 \times 6 = 40$ kN (upward $\to$ negative in output)
- Fixed–end moment M_z: $10 \times 6 + 5 \times 6^2/2 = 150$ kN·m

11.2   Example 2: Portal Frame with Internal Hinge

A two–member V–shaped frame with hinges at the apex:
- 3 nodes: N1 $(0,0)$, N2 $(5,-6)$, N3 $(10,0)$
- E1: N1$\to$N2 with Release = 1 (hinge at j)
- E2: N2$\to$N3 with Release = 2 (hinge at i)
- Supports: Pinned at N1 and N3
- Load: $F_y = 100$ kN at N2

The hinge at N2 creates a truss–like joint. The program will detect the zero–stiffness rotational DOF and constrain it automatically.

12   Limitations

- Linear–elastic analysis only (no geometric or material nonlinearity)
- 2D frames only (no out–of–plane behaviour)
- One member load per element (if multiple loads act on the same element, subdivide it with intermediate nodes)
- No self–weight calculation (must be manually applied as UDL)
- No load combinations (analyse each case separately)
- Particular solutions for the deformed shape are implemented for load types 1 and 3 on fixed–fixed elements only; for other release types the homogeneous Hermite interpolation is used
- No axial load member type (Type 1 is vertical force only)

13 Troubleshooting

Issue	Solution
Singular matrix error	Check that the structure is adequately supported (not a mechanism). Ensure at least 3 independent DOFs are restrained for stability.
Zero–stiffness warning	Expected when internal hinges create free rotation DOFs. The program handles this automatically. Verify the hinge configuration is intentional.
Workspace is empty	Make sure you clicked `Analyze`, not just `Update Plot`. Variables are only exported after a full analysis.
Previous figures persist	Click `Close Figures` or simply click `Analyze` again (it auto–closes previous figures).
Member load Type 1 or 2 ignored	Check that `a_param` is strictly between 0 and L (not equal to either).
Very small non-zero values	Expected floating–point artefacts. Values below $10^{-10} \times \max|x|$ are automatically set to zero.
Deformed shape too small/large	Adjust the Deformation Scale Factor spinner before clicking Analyze.