XLVI CILAMCE
VITÓRIA - ES - BRAZIL - 2025
XLVI IBERO-LATIN AMERICAN CONGRESS ON COMPUTATIONAL METHODS IN ENGENEERING

SFAP – Structural Frame Analysis Program: A Python-Based Educational Tool for First-Order Elastic Analysis of Planar Frame Structures

João Pedro Carneiro, Andréa R. D. da Silva, Walliston S. Fernandes, Ricardo A. M. Silveira

Programa de Pós-Graduação em Engenharia Civil (PROPEC/ DECIV/EM), Universidade de Ouro Preto
Campus Universitário, Morro do Cruzeiro, 35400-000, Ouro Preto, MG, Brasil
joao.carneiro@aluno.ufop.edu.br, andreardsilva@ufop.edu.br, walliston.fernandes@ufop.edu.br
ricardo@ufop.edu.br

Abstract. This paper presents the development of SFAP (Structural Frame Analysis Program), an educational tool implemented entirely in Python for the linear-elastic analysis of two-dimensional frame structures. SFAP is grounded in the Direct Stiffness Method and employs Euler-Bernoulli beam theory to evaluate structural responses under various loading and support conditions. The software features a user-friendly graphical interface built with Tkinter and CustomTkinter, allowing intuitive data input and efficient visualization of results, including displacements, internal forces, and support reactions. The program supports multiple load types (concentrated, distributed, thermal), diverse boundary conditions (rigid, elastic, prescribed displacements), and connection types (pinned, rigid, semi-rigid), with the latter modeled using rotational springs and fixity factors. Export capabilities include .xlsx, .docx, .pdf, and .txt formats, along with graphical outputs of the deformed configuration. Validation against benchmark results from the literature demonstrate the accuracy and robustness of SFAP. The program is intended as both a didactic resource for teaching structural analysis and a flexible foundation for future extensions, including nonlinear or 3D analyses.

Keywords: Structural Analysis, Direct Stiffness Method, Planar Frame Structures, Python-Based Implementation, Graphical User Interface

1 Introduction

Structural analysis is a fundamental phase in the design process, aimed at predicting the response of structures to various loading conditions. It enables the assessment of both ultimate and serviceability limit states, providing essential insights into internal forces, stress distributions, displacements, and overall structural behavior [1]. A solid understanding of these responses is indispensable for engineers, particularly in the context of educational environments where the foundational principles of structural mechanics are taught.

With this context in mind, the present work introduces SFAP (Structural Frame Analysis Program), a Python-based educational software developed to facilitate the teaching and learning of linear-elastic analysis of planar frame structures. The program is based on the Direct Stiffness Method and employs the Euler-Bernoulli beam theory to model the mechanical behavior of frame elements. One of its distinctive features is the incorporation of semi-rigid connections using rotational springs, enabling the analysis of a wider range of structural configurations.

The development of SFAP responds to the growing need for accessible and flexible tools in engineering education. While established programs such as Ftool [2] are widely used in Brazilian universities and offer excellent performance for 2D structural analysis, they are typically developed in compiled languages such as C, which may limit flexibility for educational purposes. In contrast, SFAP is entirely implemented in Python, a high-level programming language known for its readability, ease of use, and extensive package of scientific libraries. In addition to numerical robustness, SFAP offers enhanced report generation capabilities and a modern graphical user interface, making it a powerful platform for both students and instructors.

This paper is structured as follows: Section 2 presents an overview of the Direct Stiffness Method and the treatment of semi-rigid connections. Section 3 describes the architecture and features of the SFAP software, including its graphical interface and numerical routines. Section 4 presents a validation study based on a benchmark problem from the literature. Finally, Section 5 summarizes the main contributions and outlines directions for future work.

2 Direct Stiffness Method

The Structural Frame Analysis Program (SFAP) is based on the Direct Stiffness Method (DSM), a matrix formulation of the displacement method widely used in structural engineering. In this approach, the structure is discretized into elements with known stiffness, and nodal equilibrium leads to the global system

$$\mathbf{KU} = \mathbf{F}, \tag{1}$$

where $\mathbf{K}$ is the global stiffness matrix, $\mathbf{U}$ the nodal displacement vector, and $\mathbf{F}$ the external force vector. Once $\mathbf{U}$ is determined, internal forces and support reactions follow directly. The formulations implemented in SFAP are consistent with the classical developments presented by Martha [3] and Silva [4], ensuring reliability for educational and research purposes.

A distinctive feature of SFAP is the explicit treatment of semi-rigid connections, which represent intermediate conditions between pinned and rigid joints. As illustrated in Fig. 1, such connections are modeled by rotational springs placed at the ends of frame elements. Their behavior is assumed linear, following the relation

$$M(\theta) = S_c\theta, \tag{2}$$

where S_c denotes the rotational stiffness of the joint. To facilitate their incorporation, the fixity factor γ, introduced by Cunningham [5], is adopted. This parameter varies from 0 (pinned) to 1 (rigid) and allows a straightforward adjustment of joint stiffness within the element formulation.



Figure 1. Planar frame element with semi-rigid connections

The complete stiffness matrix for members with semi-rigid ends is derived according to the procedure of Chan and Chui [6] (see equations below), ensuring a consistent representation of intermediate joint conditions.

$$\mathbf{k}' = \begin{bmatrix} \frac{EA}{L} & 0 & 0 & -\frac{EA}{L} & 0 & 0 \\ 0 & a & b & 0 & -a & c \\ 0 & b & d & 0 & -b & e \\ -\frac{EA}{L} & 0 & 0 & \frac{EA}{L} & 0 & 0 \\ 0 & -a & -b & 0 & a & -c \\ 0 & c & e & 0 & -c & f \end{bmatrix}, \tag{3}$$

with:

$$\begin{aligned} a &= \frac{12EI}{L^3}\left(\frac{\gamma_i\gamma_j + \gamma_i + \gamma_j}{4 - \gamma_i\gamma_j}\right), \quad b = \frac{6EI}{L^2}\cdot\frac{\gamma_i(2+\gamma_j)}{(4-\gamma_i\gamma_j)}, \quad c = \frac{6EI}{L^2}\cdot\frac{\gamma_j(2+\gamma_i)}{(4-\gamma_i\gamma_j)}, \\ d &= \frac{12EI}{L}\left(\frac{\gamma_i}{4-\gamma_i\gamma_j}\right), \quad e = \frac{6EI}{L}\left(\frac{\gamma_i\gamma_j}{4-\gamma_i\gamma_j}\right), \quad f = \frac{12EI}{L}\left(\frac{\gamma_j}{4-\gamma_i\gamma_j}\right), \end{aligned} \tag{4}$$

where E represents the elastic modulus of the material; A and I represent the cross-sectional area and moment of inertia about the z-axis, respectively; and L represents the element's length. The constants γ_i and γ_j represent the stiffness fixity factors of the initial and final node, respectively.

3 The SFAP program

The Structural Frame Analysis Program (SFAP) is a Python-based educational tool designed for the linear-elastic analysis of planar frame structures using the Direct Stiffness Method. Its primary objective is to provide a robust yet accessible environment for teaching and learning structural analysis, while also serving as a flexible platform for further developments.

A key feature of SFAP is its ability to handle different joint conditions — rigid, pinned, and semi-rigid connections — the latter represented by rotational springs and the fixity factor, as described in Section 2. The program also accommodates a wide range of load types, including concentrated, distributed (partial or total), thermal (uniform and non-uniform), and nodal loads. In addition, it supports elastic restraints and prescribed displacements, extending its applicability to diverse structural configurations.

The software architecture integrates numerical routines with a Graphical User Interface (GUI) developed using Tkinter [7] and CustomTkinter [8]. Tkinter was selected because it is a built-in Python library, while CustomTkinter was adopted to allow greater customization of the interface. The interface is structured into three main areas (see Fig. 2): (1) the options menu, which provides general commands for calculations, file management, and unit settings; (2) the data visualization panel, which displays input and output data in real time; and (3) the editing panel, organized into tabs for the definition of geometry, material properties, restraints, loads, and export options (see Table 1).

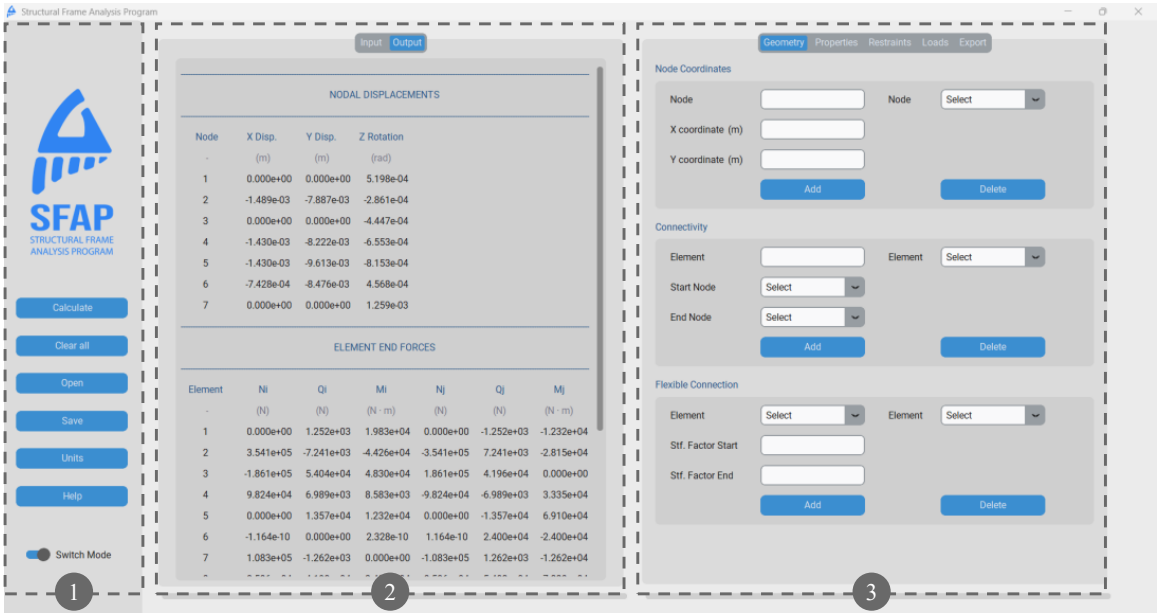


Figure 2. Main interface of the SFAP program

Table 1. Tabs available in the editor area

Tab	Description
Geometry	Input of node coordinates, connectivities and semi-rigid connections
Properties	Definition and assignment of sections and materials
Restraints	Definition and assignment of supports (fixed or elastic) and prescribed displacements
Loads	Definition and assignment of loads (distributed, thermal, concentrated and nodal)
Exports	Create report files in .xlsx (MS Excel), .docx (MS Word), .pdf (PDF) and .txt formats

Numerical computations are performed with the NumPy library, while export capabilities rely on OpenPyXL

(.xlsx), python-docx (.docx), and FPDF (.pdf). This combination enables both detailed numerical output and professional-quality reports, including graphical visualization of the deformed configuration of the structure.

4 Numerical example

To validate the implementation of SFAP, a benchmark problem originally proposed by Chan and Chui [6] was analyzed. The structure, illustrated in Fig. 3, is a sway portal frame composed of two prismatic columns and a horizontal beam. The system is subjected to a horizontal load of $10\,kN$ applied at the top of the left column and a vertical load of $100\,kN$ applied at midspan of the beam.

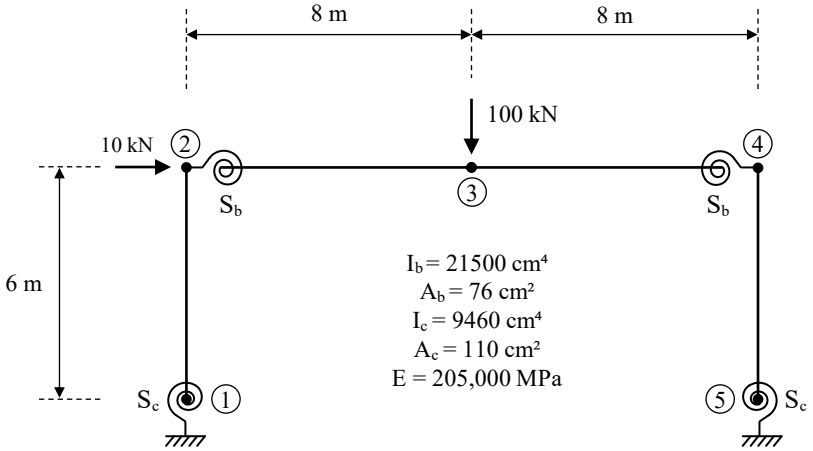


Figure 3. Semi-rigid sway portal

The mechanical properties of the members are also indicated in Fig. 3, where the subscripts "c" and "b" denote the columns and the beam, respectively. The semi-rigid behavior of the connections was modeled by assigning a rotational stiffness of $S_b = 4EI_b/L_b$ to the beam-to-column joints and $S_c = EI_c/L_c$ to the column bases, which yields the corresponding fixity factors $\gamma_b = 0.1176$ and $\gamma_c = 0.03226$.

The analysis focused on the evaluation of bending moments at five key nodes of the frame, highlighted in Fig. 3. The results computed with SFAP were compared to those reported by Chan and Chui [6]. As shown in Table 2, the differences are negligible within the reported precision for all cases studied, including both rigid and semi-rigid connections. This confirms the accuracy and reliability of the proposed implementation.

Overall, this validation confirms that SFAP is capable of reproducing benchmark solutions with high precision, ensuring its adequacy as both an educational and research-oriented tool for structural analysis.

Table 2. Bending moment (kNm) in the frame for different column and beam stiffness values

Stiffness	Node	Chan and Chui [6]	SFAP
$S_b = 4EI_b/L_b, \quad S_c = \infty$	1	31.7	31.7
	2	93.6	93.6
	3	296.3	296.3
	4	113.8	113.8
	5	71.5	71.5
$S_b = 4EI_b/L_b, \quad S_c = EI_c/L_c$	1	0.3	0.3
	2	80.3	80.3
	3	301.7	301.7
	4	116.4	116.4
	5	24.2	24.2

5 Conclusions

The development of the Structural Frame Analysis Program (SFAP) demonstrates how a Python-based implementation, grounded in the Direct Stiffness Method and Euler–Bernoulli beam theory, can successfully combine academic rigor with accessibility. By integrating the modeling of semi-rigid connections, diverse loading and boundary conditions, and an intuitive graphical interface, the program expands the scope of traditional educational tools while maintaining numerical robustness. Validation against benchmark problems confirmed its reliability, reinforcing SFAP's role as both a robust didactic resource and a methodological foundation for further research.

Beyond meeting immediate educational and academic needs, SFAP establishes a pathway for future advancements in structural analysis. The program is intended to be extended to encompass nonlinear and three-dimensional applications. In this sense, SFAP not only addresses existing gaps in teaching and practice but also positions itself as a strategic platform for advancing structural modeling and analysis.

Acknowledgements. The authors gratefully acknowledge the financial and institutional support that made this research possible. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) – Finance Code 001. The authors also acknowledge the support received from the Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq), the Fundação de Amparo à Pesquisa do Estado de Minas Gerais (FAPEMIG), and the Universidade Federal de Ouro Preto (UFOP), through the Programa de Pós-Graduação em Engenharia Civil (PROPEC/UFOP) and the Pró-Reitoria de Pesquisa, Pós-Graduação e Inovação (PROPPI/UFOP), as well as from the Fundação Gorceix, for the development of this research.

Authorship statement. The authors hereby confirm that they are the sole liable persons responsible for the authorship of this work, and that all material that has been herein included as part of the present paper is either the property (and authorship) of the authors, or has the permission of the owners to be included here.

References

[1] Associação Brasileira de Normas Técnicas. NBR 6118:2023 – Projeto de estruturas de concreto — Procedimento. Rio de Janeiro: ABNT, 2023.

[2] L. F. Martha and R.L.Rangel. Ftool: Three decades of success as an educational program for structural analysis. In *XLIII Ibero-Latin-American Congress on Computational Methods in Engineering*, Foz do Iguaçu, PR. ABMEC, 2020.

[3] L. F. Martha. *Análise de estruturas: conceitos e métodos básicos*. GEN LTC, Rio de Janeiro, 3rd edition, 2022.

[4] A. R. D. Silva. *Sistema computacional para análise avançada estática e dinâmica de estruturas metálicas.* PhD thesis, Universidade Federal de Ouro Preto, Programa de Pós-Graduação em Engenharia Civil, 2009.

[5] R. Cunningham. Some aspects of semi-rigid connections in structural steelwork. *The Structural Engineer*, vol. 68, n. 5, pp. 85–92, 1990.

[6] S. L. Chan and P. P. T. Chui. *Non-linear Static and Cyclic Analysis of Steel Frames with Semi-Rigid Connections*. Elsevier, Oxford, 2000.

[7] Python Software Foundation. Tkinter — python interface to tcl/tk. `https://docs.python.org/3/library/tkinter.html`. Accessed: 2025-09-23, 2025.

[8] T. Schimansky. CustomTkinter documentation. `https://customtkinter.tomschimansky.com/`. Accessed: 2025-09-23, 2025.