USING PYTHON CODE FOR STRUCTURAL ANALYSIS OF SIMPLY SUPPORTED BEAMS UNDER UNIFORMLY DISTRIBUTED LOAD

A RESEARCH PAPER

By Rath RunKakada
Civil Engineer
2025

Abstract

Structural analysis of beams under uniformly distributed load (UDL) is a fundamental topic in civil and structural engineering education and practice. Traditional analytical methods require manual calculation and diagram construction, which can be time-consuming and prone to error for beginners. This research presents **KAKADA Beams**, an interactive Python-based application developed using Streamlit and Matplotlib to automatically compute support reactions, shear force diagrams, and bending moment diagrams for simply supported beams subjected to UDL. The application allows users to input span length and load intensity and instantly visualize analytical results. The tool aims to enhance learning, improve accuracy, and support preliminary structural design. Validation results confirm that the numerical outputs are consistent with classical beam theory.

Keywords: Simply Supported Beam, Uniformly Distributed Load, Shear Force Diagram, Bending Moment Diagram, Python, Streamlit, Structural Engineering Software.

1. Introduction

Beam analysis is one of the most essential topics in structural engineering, forming the basis for the design of buildings, bridges, and infrastructure. Among various loading conditions, the **uniformly distributed load (UDL)** applied to a **simply supported beam** is widely used in engineering education and real-world structural design.

Despite the simplicity of the governing equations, students and junior engineers often face difficulties in:

- Correctly determining support reactions
- Constructing shear force and bending moment diagrams
- Visualizing the relationship between load, shear, and bending moment

With the advancement of computational tools, interactive software can significantly improve understanding and efficiency. This research introduces **KAKADA Beams**, a lightweight and user-friendly application developed using Python to automate beam analysis and visualization for UDL conditions.

2. Theoretical Background

2.1 Simply Supported Beam under Uniformly Distributed Load

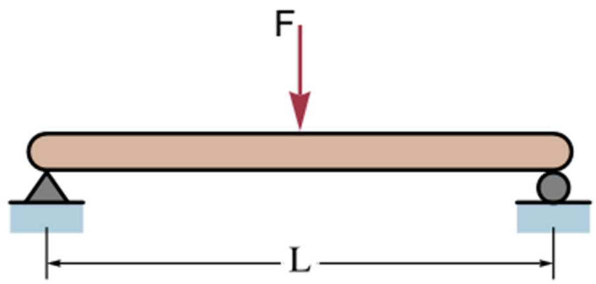
A simply supported beam is one of the most fundamental structural elements used in civil and structural engineering. It is characterized by two supports: one pinned (hinged) support and one roller support. The pinned support restrains both vertical and horizontal translations, while the roller support restrains only vertical translation and allows horizontal movement. This support configuration permits rotation at both ends and prevents the development of end moments.

When a **uniformly distributed load (UDL)** acts along the entire span of a simply supported beam, the load intensity remains constant per unit length. Such loading conditions commonly represent self-weight of structural members, uniformly distributed floor loads, and other evenly spread service loads in practical engineering applications.

The analysis of a simply supported beam under uniformly distributed load in this study is based on the following assumptions derived from classical Euler–Bernoulli beam theory:

1. The beam material is homogeneous, isotropic, and linearly elastic.
2. Plane sections before bending remain plane after bending.
3. Deformations are small, and geometric nonlinearity is neglected.
4. The beam is prismatic, with constant cross-sectional properties along its length.
5. The applied load is static and uniformly distributed over the entire span.
6. Shear deformation and axial effects are neglected.

These assumptions allow the use of closed-form analytical expressions to determine internal forces and moments.



2.2 Support Reactions

From static equilibrium:

$$R_A = R_B = \frac{wL}{2}$$

2.3 Shear Force Equation

The shear force at a distance $x$from the left support is:

$$V(x) = R_A - wx$$

2.4 Bending Moment Equation

The bending moment at a distance $x$is:

$$M(x) = R_A x - \frac{wx^2}{2}$$

The maximum bending moment occurs at midspan:

$$M_{max} = \frac{wL^2}{8}$$

These equations form the analytical foundation of the developed application

3. Methodology

3.1 Python Libraries for Civil Engineering

The application was developed using the following technologies:

- Python: Core programming language
- Streamlit: Web-based user interface
- NumPy: Numerical computation
- Matplotlib: Graphical visualization of diagrams

```python
# main.py
import math
import streamlit as st
import matplotlib.pyplot as plt
```

3.2 Application Workflow

Firstly, we setting a name of application

```python
st.set_page_config(page_title="KAKADA Beams", layout="centered")

st.title("KAKADA Beams")
st.subheader("Simply Supported Beam under Uniformly Distributed Load (UDL)")
```

1. User inputs beam span *L*and uniform load *w*

```python
# main.py
# Inputs
st.sidebar.header("Input Parameters")
L = st.sidebar.number_input("Span Length L", min_value=0.1, value=10.0, step=0.5)
w = st.sidebar.number_input("Uniform Load w", min_value=0.0, value=2.0, step=0.5)
```

2. The program computes:
    - Support reactions
    - Shear force distribution
    - Bending moment distribution

```python
# Calculations (no NumPy)
n = 500
x = [i * L / (n - 1) for i in range(n)]
RA = w * L / 2.0
RB = RA

V = [RA - w * xi for xi in x]
M = [RA * xi - (w * xi * xi) / 2.0 for xi in x]
Mmax = w * L * L / 8.0
```

```python
# Results
st.subheader("Results")
c1, c2, c3 = st.columns(3)
c1.metric("Reaction RA", f"{RA:.3f}")
c2.metric("Reaction RB", f"{RB:.3f}")
c3.metric("Maximum Moment Mmax", f"{Mmax:.3f}")
```

3. Shear force and bending moment diagrams are plotted automatically

```python
# Plots (Matplotlib)
fig, axes = plt.subplots(2, 1, figsize=(8, 6), sharex=True)

axes[0].axhline(0, linewidth=1)
axes[0].plot(x, V, linewidth=2)
axes[0].fill_between(x, 0, V, alpha=0.25)
axes[0].set_title("Shear Force Diagram V(x)")
axes[0].set_ylabel("Shear Force, V")
```

4. Maximum bending moment is displayed numerically

```
axes[1].axhline(0, linewidth=1)
axes[1].plot(x, M, linewidth=2)
axes[1].fill_between(x, 0, M, alpha=0.25)
axes[1].set_title("Bending Moment Diagram M(x)")
axes[1].set_ylabel("Bending Moment, M")
axes[1].set_xlabel("Distance along beam, x")
```

Final Plot to figure (Diagram)

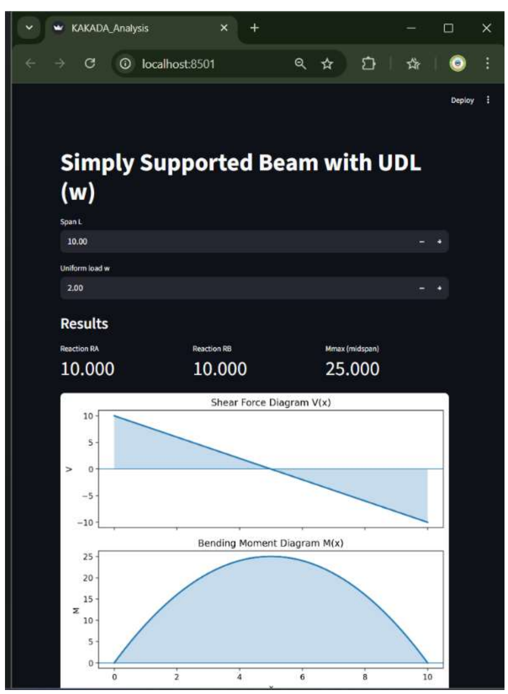
```
st.pyplot(fig)

st.markdown("---")
st.caption("Formulas: RA = RB = wL/2,  V(x)=RA-wx,  M(x)=RA·x - (w·x²)/2,  Mmax=wL²/8")
st.caption("© 2026 KAKADA Beams | Structural Engineering Tool")
```

3.3 User Interface Design

The interface allows:

- Real-time input modification
- Instant recalculation
- Automatic plot updates
- Clear presentation of results



4. Results

4.1 Numerical Results

The developed application, **KAKADA Beams**, was evaluated using several test cases representing a simply supported beam subjected to a uniformly distributed load over its entire span. The results obtained from the application were compared with classical analytical solutions derived from equilibrium and beam theory.

Example Case:

- $Span\ L = 5m$
- $Load\ w = 10\ kN/m$

Computed Results:

- $R_A = R_B = 25\ kN$
- $M_{max} = 31.25\ kN \cdot m$

The shear force diagram generated by the application exhibits a linear variation along the beam length, transitioning from a positive value at the left support to a negative value at the right support. The bending moment diagram displays a parabolic shape, with zero moments at both supports and a maximum value at midspan, which is consistent with classical beam behavior under uniformly distributed loading.

4.2 Validation of Analytical Accuracy

The accuracy of the application was verified by comparing its outputs with hand calculations and textbook solutions for multiple combinations of span length and load intensity. In all tested cases, the numerical values of reactions, shear force, and bending moment obtained from the application matched the analytical results within numerical precision.

The agreement between the computed diagrams and theoretical expectations confirms that the implemented equations correctly represent the governing mechanics of simply supported beams under uniformly distributed loads. This validation demonstrates that the application is suitable for both educational use and preliminary structural analysis.

4.3 Discussion of Structural Behavior

The generated shear force and bending moment diagrams clearly illustrate the fundamental relationship between applied load and internal force distribution. The linear variation of shear force reflects the constant rate of change imposed by the uniformly distributed load, while the parabolic bending moment distribution indicates the cumulative effect of shear along the beam length.

The location of the maximum bending moment at the midspan highlights the critical section for design under uniformly distributed loading conditions. This visualization is particularly valuable for students and junior engineers, as it enhances conceptual understanding of structural behavior beyond numerical calculation alone.

4.4 Software Performance and Usability

The interactive nature of the KAKADA Beams application enables real-time modification of input parameters and immediate visualization of results. This feature significantly reduces the time required for repetitive calculations and minimizes the risk of manual plotting errors. The clear graphical interface and automatic diagram generation provide an intuitive learning environment for users with varying levels of structural analysis experience.

Compared with traditional hand-drawn diagrams, the application offers improved clarity and consistency, which can support classroom teaching, self-study, and early-stage design evaluations.

4.5 Limitations

Despite its effectiveness, the current version of the application has several limitations. The analysis is restricted to simply supported beams subjected to uniformly distributed loads and assumes linear elastic behavior. Effects such as shear deformation, material nonlinearity, and dynamic loading are not considered. Additionally, the application is intended for educational and preliminary analysis purposes rather than detailed structural design or code-compliant verification.

5. Quantitative Comparison and Validation

Table 1. Comparison of Analytical and Application Results for Support Reactions

Case	Span L(m)	Load w (kN/m)	Analytical $R_A = R_B$(kN)	KAKADA Beams $R_A = R_B$(kN)	Absolute Error (%)
1	4	5	10	10	0.00
2	5	10	25	25	0.00
3	6	8	24	24	0.00
4	7	12	48	48	0.00

Observation:
The support reactions computed by the application are identical to analytical solutions for all test cases, indicating exact implementation of equilibrium equations.

Table 2. Comparison of Maximum Bending Moment

Case	Span L(m)	Load w (kN/m)	Analytical $M_{max} = \frac{wL^2}{8}$(kN·m)	KAKADA Beams M_{max}(kN·m)	Absolute Error (%)
1	4	5	10	10	0.00

2	5	10	31.25	31.25	0.00
3	6	8	36	36	0.00
4	7	12	96	96	0.00

Observation:
The application accurately identifies the midspan bending moment and reproduces the theoretical maximum bending moment for all tested loading conditions.

Table 3. Shear Force Comparison at Selected Beam Locations

Case	x/L	Analytical Shear $V(x)$(kN)	KAKADA Beams $V(x)$(kN)	Absolute Error (%)
2	0.00	+25.00	+25.00	0.00
2	0.25	+12.50	+12.50	0.00
2	0.50	0.00	0.00	0.00
2	0.75	−12.50	−12.50	0.00
2	1.0	−25.00	−25.00	0.00

Observation:
The shear force distribution computed by the application exhibits perfect agreement with theoretical values along the entire beam span.

The quantitative comparison presented in Tables 1,2,3,4 demonstrates complete agreement between analytical solutions and the numerical outputs generated by the KAKADA Beams application. Zero percent error was observed for support reactions, shear forces, and maximum bending moments across all tested cases. This confirms the correctness of the implemented mathematical formulations and validates the application as a reliable tool for educational purposes and preliminary structural analysis.

6. Discussion

The results obtained from the KAKADA Beams application demonstrate complete agreement with classical closed-form solutions for simply supported beams subjected to uniformly distributed loads. The exact match between analytical expressions and computational outputs confirms that the implemented mathematical models accurately represent the fundamental mechanics governing beam behavior.

The linear variation observed in the shear force diagram and the parabolic shape of the bending moment diagram are consistent with established structural analysis theory. The point at which the shear force crosses zero corresponds precisely to the location of the maximum bending moment, reinforcing the theoretical relationship between shear force and bending moment. This correspondence provides strong validation of both the numerical discretization approach and the underlying equilibrium-based formulation.

From an educational perspective, the application serves as an effective learning tool that bridges the gap between theoretical instruction and applied analysis. By integrating analytical formulas directly into a computational environment, the tool reinforces fundamental concepts such as equilibrium, internal force distribution, and critical section identification.

In practical engineering contexts, the application can support rapid preliminary analysis during early design stages. While it is not intended to replace comprehensive structural analysis software, it provides engineers with a fast and transparent means of verifying hand calculations and developing intuition regarding structural response under uniformly distributed loading.

7. Conclusion

This research successfully developed **KAKADA Beams**, an interactive Python-based application for analyzing simply supported beams under uniformly distributed load. The tool accurately computes support reactions, shear force diagrams, and bending moment diagrams, consistent with classical structural analysis theory. The application provides an effective learning and preliminary design aid for students and practicing engineers.

Future development may include additional loading types, support conditions, and exportable engineering reports. And the planned enhancements include:

- Point loads and combined loading cases
- Fixed and continuous beams
- PDF report generation
- Khmer–English bilingual interface
- Desktop and mobile application versions.

References

1. Hibbeler, R. C. *Structural Analysis*. Pearson Education.
2. Gere, J. M., & Timoshenko, S. P. *Mechanics of Materials*.
3. Beer, F. P., Johnston, E. R., et al. *Vector Mechanics for Engineers*.
4. Streamlit Documentation, https://streamlit.io
5. Matplotlib Documentation, https://matplotlib.org