

JACE

Journal Of Architecture and Civil Engineering

ISSN 2955-2451

Vol. 1 (1/2026)



JACE

JOURNAL OF ARCHITECTURE AND CIVIL ENGINEERING

ISSN 2955-2451

Vol.1 (1/2026)

JACE

JOURNAL OF ARCHITECTURE AND CIVIL ENGINEERING

ISSN 2955-2451

Vol.1 (1/2026)

Main Editor: Bekim Fetaji, PhD (North Macedonia)

Editor in Chief: Sani Demiri, PhD (North Macedonia)

Editors:

Iraj Hashi, PhD (England)

Robert Pichler, PhD (Austria)

Ozcan Asilkan, PhD (Germany)

Andrej Shkraba, PhD (Slovenia)

Maaruf Ali, PhD (England)

Quirico Migheli, PhD (Italy)

Bujar Krasniqi, PhD (Republic of Kosovo)

Zekirija Idrizi, PhD (North Macedonia)

Publisher: Mother Teresa University in Skopje, Republic of North Macedonia

Editorial Board

Alfonso Vargas Sanchez, PhD (Spain)	Nezir Kraki, PhD (France)
Isak Idrizi, PhD (North Macedonia)	Marc Hill, PhD (Austria)
Niko Qafoku, PhD (USA)	Inge Hutter, PhD (Netherland)
M. M. Sulphey, PhD (India)	Yavuz Emre Arslan, PhD (Turkey)
Dimitrios Karras, PhD (Greece)	Chih-Wen Lan, PhD (China)
Donald Elmazi, PhD (Albania)	Herman Maurerer, PhD (Austria)
Bujar Dugolli, PhD (Kosovo)	Ayhan Oral, PhD (Turkey)
Sermin Senturan, PhD (Turkey)	Zoran Gacovski, PhD (North Macedonia)
Mirko Perano, PhD (Italy)	Anton Stoilov, PhD (Bulgaria)
Salvatore Rubino, PhD (Italy)	Shpetim Rexhepi, PhD (North Macedonia)
Ruzhdi Sefa, PhD (Kosovo)	Mehmed Ganic, PhD (Bosnia and Hercegovina)
Vebi Fazliu, PhD (North Macedonia)	Andrej Shkraba, PhD (Slovenia)
Radmila Tomovska, PhD (North Macedonia)	Mesut Idriz, PhD (Bosnia and Hercegovina)
Kalman Mizsei, PhD (Hungary)	Monika Lutovska, PhD (North Macedonia)
Amir Ferati, PhD (North Macedonia)	Edmond Krusha, PhD (Croatia)
Fiona Todhri, PhD (Albania)	Blerta Nazarko, PhD (Albania)

Journal of Architecture and Civil Engineering

Main Editor: Bekim Fetaji, PhD

Editor in Chief: Sani Demiri, PhD

Secretary: Drenusha Kamberi, PhD

Technical Editing/Layout: Stojan Kitanov, PhD

Printing and Operations Council: Drenusha Kamberi, PhD, Stojan Kitanov, PhD, Drita Abazi, PhD; Teuta Jusufi Zenku, PhD; Daniel Pavlovski, PhD; Elisabeta Bajrami Ollogu, PhD; Isak Idrizi, PhD; Ariton Rama, MSc

ISSN (print) 2545-4463

ISSN (online) 2955-2451

Editorial Office: Journal of Architecture and Civil Engineering is published two times per year.

Journal of Architecture and Civil Engineering

Mother Teresa University in Skopje

Str. 1669 No. 11A

Skopje 1000

Republic of North Macedonia

Phone: +389 2 3161 004

E-mail: journals@unt.edu.mk

Web: <https://journals.unt.edu.mk/jace/>

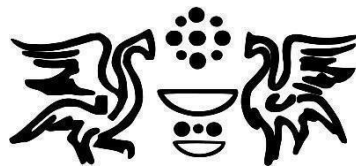
Account No.: 160016267778815

Code and program: 723019 - 45

Tax No.: 4080016561272

Bank: National Bank of the Republic of North Macedonia

The publication of the Journal is supported by:



Ministry of Culture of the Republic of North Macedonia

Editorial Foreword

Prof. Dr. Sani Demiri

Editor in Chief

Dear Readers, Authors and Collaborators,

Mother Teresa University in Skopje is pleased to present **Volume 1 (2026)** of the *Journal of Architecture and Civil Engineering (JACE)*. This edition builds on the momentum generated by the **TSD 2025** conference and features a selection of rigorously peer-reviewed papers addressing key sustainable development challenges in the Western Balkans and beyond.

The volume showcases advances in **green and renewable technologies, policy frameworks supporting, and AI-driven solutions for environmental sustainability**. Its interdisciplinary scope spans research on renewable energy systems, urban resilience and smart city strategies, cybersecurity as a pillar of digital sustainability, and the socio-economic dimensions of climate adaptation.

Under the leadership of **Editor-in-Chief Sani Demiri, PhD**, the contributions in this issue are closely aligned with the **United Nations Sustainable Development Goals (SDGs)** and propose practical, evidence-based pathways for regional development and long-term resilience.

We extend our sincere appreciation to the authors, reviewers, and institutional partners whose commitment and expertise made this volume possible. We invite the academic and professional community to engage with these works, contribute new research, and collaborate with us in future editions to further amplify their impact.

Sincerely,
Prof. Dr. Sani Demiri, PhD
Editor in Chief

CONTENTS

Author	Paper Title	Page No.
Isak S. Idrizi, Zekirija S. Idrizi	LEVERAGING MATLAB IN REINFORCED CONCRETE DESIGN STUDY CASE: PAD FOUNDATION	8
Agon Shaini, Isak S. Idrizi	DESIGN AND OPTIMIZATION OF RC BEAMS ACCORDING TO EUROCODE 2 STAGE 1: DEVELOPING THE CORE DESIGN ALGORITHM	19
Agon Shaini, Isak S. Idrizi	DESIGN AND OPTIMIZATION OF RC BEAMS ACCORDING TO EUROCODE 2 STAGE 2: DEVELOPING ADVANCED ROUTINE FOR OPTIMIZATION OF RC BEAMS IN COMPLIANCE WITH EC2 PROVISIONS	29

LEVERAGING MATLAB IN REINFORCED CONCRETE DESIGN STUDY CASE: PAD FOUNDATION

Isak S. Idrizi

Faculty of Civil Engineering, University Mother Teresa, Shkup, North Macedonia,
isak.idrizi@unt.edu.mk

Zekirija S. Idrizi

Faculty of Civil Engineering, University Mother Teresa, Shkup, North Macedonia,
zekirija.idrizi@unt.edu.mk

ABSTRACT

The design of reinforced concrete pad foundations in accordance with Eurocode 2 (EC2) is traditionally performed through manual calculations supported by engineering judgment and iterative trial-and-error procedures. While this manual approach ensures transparency, it is time-consuming and limits the ability to efficiently explore alternative design scenarios for optimization purposes. This paper presents the development and application of an interactive MATLAB-based software tool, padFoundationEC2 v.2025, created to support the preliminary design and optimization of reinforced concrete pad foundations in full compliance with EC2 provisions.

The developed application automates the complete design workflow, starting from the definition of permanent and variable loads, applied bending moment, material properties, column geometry, and designs the footing geometry within the interval of a specified soil bearing capacity range. For each soil pressure level, the required footing area is computed and translated into plan dimensions based on a user-defined aspect ratio. The footing thickness is then iteratively increased until all relevant EC2 ultimate limit state checks are satisfied. These include flexural resistance in two orthogonal directions, one-way shear verification at a distance equal to the effective depth from the column face, punching shear verification at the $2d$ control perimeter, and punching shear verification at the column face, incorporating the EC2 β -factor to account for load eccentricity and moment transfer.

In addition to the optimized EC2-compliant solution, the application introduces a reference footing thickness based solely on flexural considerations, enabling direct comparison between theoretical bending-controlled design and full code-compliant design. This dual approach highlights the governing role of shear and punching shear in pad footing design, particularly for higher soil bearing capacities.

The app features real-time parametric analysis, two-dimensional plots illustrating the

variation of footing volume, cost, thickness, and reinforcement demand with soil bearing capacity, and an interactive three-dimensional visualization of the footing–column–soil system. These visualization tools significantly enhance interpretability and provide immediate feedback on the influence of design parameters.

The results demonstrate that MATLAB can serve as a powerful and transparent platform for reinforced concrete foundation design, bridging the gap between manual calculations and closed commercial software. The proposed tool is particularly well suited for engineering education, preliminary design and parametric studies, offering an efficient environment for understanding and applying Eurocode 2 design principles.

KEYWORDS: MATLAB, RC design, pad foundation, Eurocode 2

1. INTRODUCTION AND PROBLEM DESCRIPTION

Reinforced concrete foundations are typically designed using manual Eurocode 2 (EC2) calculations, engineering judgment, and iterative adjustments to balance safety and material use. While structural analysis software can automate foundation sizing, these closed systems often lack transparency in intermediate steps and code checks, making it difficult for engineers to verify design conditions or quickly assess alternative options. To address these limitations, this research introduces a MATLAB-based interactive App specifically developed for the preliminary design of pad footings in accordance with Eurocode 2.

So, the structural problem addressed in this study concerns the design of a reinforced concrete pad footing supporting a centrally loaded column and subjected to combined vertical loading and bending moment. Pad footings represent one of the common shallow foundation types in building structures, and their design must comply with the safety and serviceability requirements defined by Eurocode 2 (EC2).

This App, named ‘padFoundationEC2 v.2025’ incorporates automated flexural design, one-way shear verification, punching shear verification at both the 2d perimeter and the column face, and cost/volume optimization over a user-defined soil bearing capacity range. It combines numerical methods, equation-based analysis, and a user-friendly graphical interface, thus streamlining manual tasks into an efficient visual environment that is ideal for engineering education, early design, and parametric studies.

2. MANUAL SOLUTION

The manual approach for designing the pad foundation consists of estimating the required area of the pad footing (so that the design loads acting over it are safely transmitted down the soil having a certain bearing capacity level) and evaluating the footing thickness such that all relevant EC2 limit states are satisfied.

The design process begins with the characterization of the applied loads: permanent load G_k , variable load Q_k , and the design bending moment M_z . The design value of the axial load N_{Ed} is calculated using the partial safety factors γ_G and γ_Q .

For a specific soil bearing capacity (allowable pressure), the theoretical area of the footing supporting the applied load is computed and used to obtain corresponding footing plan dimensions B and H based on the specified aspect ratio. Based on this geometry, Eurocode 2 flexural checks are carried out in both orthogonal directions.

Assuming a uniform design contact pressure $q = \frac{N_{Ed}}{B \cdot H}$, the cantilever moments at the column face are:

$$M_x = q \cdot H \cdot \frac{L_x^2}{2} \quad M_y = q \cdot B \cdot \frac{L_y^2}{2}$$

where L_x and L_y are the clear cantilever lengths. With effective depth $d = h \cdot 1000 - c_{nom} - 0.5 \Phi_{long}$ and internal lever arm $z = 0.9 \cdot d$, the required longitudinal reinforcement in each direction is obtained from the standard EC2 expression:

$$A_{s,req} = \frac{M_{Ed} \cdot 10^6}{0.87 \cdot f_{yd} \cdot z}$$

Shear verification follows EC2 clauses 6.2–6.4. One-way (beam) shear is evaluated at a distance d from the column face in both directions, while punching shear is checked on the control perimeter at approximately $2d$ from the column face and directly at the column perimeter.

Eccentric loading is included through the EC2 β -factor:

$$\beta = 1 + k_{\beta} \cdot \left(\frac{M_{Ed}}{V_{Ed}} \right) \cdot \left(\frac{u}{W_1} \right)$$

where u is the control parameter and W_1 is a geometric factor depending on the column dimensions. The design shear resistance is computed using the EC2 concrete shear strength $v_{Rd,c}$, which depends on the longitudinal reinforcement ratio ρ_l , the concrete strength f_{ck} , the safety factor γ_c and the size effect factor k .

All the Eurocode-based formulas utilized for design of the pad footing are outlined in detail below. They are organized into four tables, in terms of flexure and three distinct shear design conditions.

The sequential steps for performing the flexural resistance check are laid out in table 1. They involve computing the bending moments generated by soil reactions and comparing the required reinforcement area with allowable limits. The effective depth d , internal lever arm z , and resisting moment capacity are key quantities governing this verification.

Table 1: Sequential steps for “Flexural Design Check” (Eurocode 2, Section 6.1)

Step 1	Cantilever lengths	$L_x = \frac{B}{2} - \frac{h_c}{2} \quad ; \quad L_y = \frac{H}{2} - \frac{b_c}{2}$	
Step 2	Bending moments at the column face	$M_x = q H \frac{L_x^2}{2}, \quad M_y = q B \frac{L_y^2}{2}$	
Step 3	The effective depth	$d = h \times 1000 - c_{nom} - \frac{\Phi}{2}$	
Step 4	The lever arm	$z = 0.9d$	
Step 5	Required tensile	$A_{s,req,x} = \frac{M_x}{0.87 f_{yd} z}$	$A_{s,req,y} = \frac{M_y}{0.87 f_{yd} z}$

	reinforcement	
Step 6	Reinforcement ratios and limits	$\rho_{req,x} = \frac{A_{s,req,x}}{b_x * d}; \rho_{req,y} = \frac{A_{s,req,y}}{b_y * d}; \rho_{min} = 0.0013; \rho_{max} = 0.04$

The one-way shear check (table 2) evaluates vertical shear stresses along a critical section located at a distance d from the column face. The design shear stress v_{Ed} must not exceed the shear resistance of concrete $v_{Rd,c}$, which depends on parameters such as concrete strength, reinforcement ratio, and an enhancement factor k defined by EC2. The app uses EC2 Eq. (6.2a) and (6.2b) for $v_{Rd,c}$, with the design shear force based on the soil pressure acting outside the critical section.

Table 2: Sequential steps for “One-way Shear Design Check” (Eurocode 2, Section 6.1)

Step 1	Shear spans and design shear forces	$L_{x,shear} = L_x - d ; L_{y,shear} = L_y - d$ $V_{Ed,x} = q \cdot H \cdot L_{x,shear} ; V_{Ed,y} = q \cdot B \cdot L_{y,shear}$
Step 2	Design shear stress	$v_{Ed,x} = V_{Ed,x}/(H \cdot d) ; v_{Ed,y} = V_{Ed,y}/(B \cdot d)$
Step 3	Size factor k (EC2 6.2.2(1))	$k = \min \left\{ 2.0 ; 1 + \sqrt{\frac{200}{d}} \right\}$
Step 4	Minimum shear strength v_{min} (EC2 6.2a, b)	$v_{min} = 0.035 \cdot k^{1.5} \cdot \sqrt{f_{ck}}$
Step 5	Concrete shear resistance $v_{Rd,c}$ (EC2 6.2a)	$v_{Rd,c} = \max \left\{ C_{Rd,c} \cdot k \cdot (100 \cdot \rho_l \cdot f_{ck})^{\frac{1}{3}} ; v_{min} \right\}$ <p>with $C_{Rd,c} = 0.18/\gamma_c$ and ρ_l representing the average longitudinal reinforcement ratio</p>
Step 6	Shear Check	Shear satisfied if $v_{Ed} \leq v_{Rd,c}$

Punching shear checks are performed at two separate control perimeters: the perimeter directly around the column (referred to as u_0) and the outer perimeter situated at a distance $2d$ from the column faces (denoted as u_1). For each perimeter, the punching shear stress v_{Ed} must remain below the allowable punching resistance, considering the influence of unbalanced moments and geometric factors described in EC2.

When considering the control perimeter u_1 , at a distance $2d$ from the column face, the app

computes the perimeter, enclosed area, and the eccentricity factor β according to EC2 Clause 6.4 and Eq. (6.39), using explicit expressions for W_1 as suggested in design guides. The sequential steps for this type of shear check are laid out in the following table 3.

Table 3: Sequential steps of Shear Design Checks for “Punching Shear at perimeter $u_1 / 2d$ from Column faces” (Eurocode 2, Section 6.4) Control

Step 1	perimeter and enclosed area A_1	$u_1 = 2 \cdot [(c_1 + 2d) + (c_2 + 2d)]$; $A_1 = (c_1 + 2d) \cdot (c_2 + 2d)$
Step 2	Shear force outside u_1	$V_{Ed,\beta(u_1)} = q \cdot (B \cdot H - A_1)$
Step 3	First moment W_1 at u_1 (expanded, based on EC2 6.41)	$W_{1(u_1)} = 0.5 \cdot c_1^2 + c_1 \cdot c_2 + 4 \cdot c_2 \cdot d + 16 \cdot d^2 + 2 \cdot \pi \cdot d \cdot c_1$
Step 4	Eccentricity factor β at u_1 (EC2 Eq. 6.39)	$\beta(u_1) = 1 + k_\beta \cdot \left(\frac{M_{Ed}}{V_{Ed,\beta(u_1)}} \right) \cdot \left(\frac{u_1}{W_{1(u_1)}} \right)$ with k_β taken from EC2 Table 6.1 as a function of c_1/c_2 . $\beta \geq 1.0$
Step 5	Design punching shear stress at u_1	$v_{Ed,2d} = \beta(u_1) \cdot \frac{V_{Ed,\beta(u_1)}}{u_1 \cdot d}$
Step 6	Punching concrete resistance $v_{Rd,c,p}$ (EC2 6.4, analogous to 6.2a)	$v_{Rd,c,p} = \max\{ C_{Rd,c} \cdot k \cdot (100 \cdot \rho_1 \cdot f_{ck})^{\frac{1}{3}} ; v_{min,p} \}$ with $v_{min,p} = 0.035 \cdot \sqrt[3]{k^2} \cdot \sqrt{f_{ck}}$
Step 7	Shear Check	Punching shear satisfied if $v_{Ed,2d} \leq v_{Rd,c,p}$

When considering the control perimeter u_0 at the column face, the app checks punching against the maximum punching resistance $v_{Rd,max}$ as per EC2 Clause 6.4. The basic perimeter u_0 is the column perimeter; the shear force includes all soil pressure outside A_0 . The sequential steps for this type of shear check are laid out in the following table 4.

Table 4: Sequential steps of Shear Design Checks for “Punching Shear at Column face u_0 ” (Eurocode 2, Section 6.4, $v_{Rd,max}$)

Control		
Step 1	perimeter u_0 and enclosed area A_0	$u_0 = 2 \cdot (c_1 + c_2) ; A_0 = c_1 \cdot c_2$
Step 2	Shear force outside u_0	$V_{Ed,\beta(u_0)} = q \cdot (B \cdot H - A_0)$
Step 3	First moment W_1 at u_0 (expanded)	$W_{1(u_0)} = 0.5 \cdot c_1^2 + c_1 \cdot c_2$
Step 4	Eccentricity factor β at u_0 (EC2 Eq. 6.39)	$\beta(u_0) = 1 + k_\beta \cdot \left(\frac{M_{Ed}}{V_{Ed,\beta(u_0)}} \right) \cdot \left(\frac{u_0}{W_{1(u_0)}} \right)$ with k_β taken from EC2 Table 6.1 as a function of c_1/c_2 . $\beta \geq 1.0$
Step 5	Design punching shear stress at u_0	$v_{Ed,0} = \beta(u_0) \cdot \frac{V_{Ed,\beta(u_0)}}{u_0 \cdot d}$
Step 6	Maximum punching resistance $v_{Rd,max}$ (EC2 6.4)	$v_{Rd,max} = 0.5 \cdot \left[0.6 \cdot \left(1 - \frac{f_{ck}}{250} \right) \right] \cdot \frac{f_{ck}}{\gamma_c}$
Step 7	Shear Check	Punching shear satisfied if $v_{Ed,0} \leq v_{Rd,max}$

3. MATLAB APP ARCHITECTURE AND IMPLEMENTATION

The reinforced concrete pad footing design problem described previously is implemented in a dedicated MATLAB application developed using a fully programmatic approach. This choice allows complete control over the numerical procedures, graphical user interface elements, and data visualization routines, ensuring transparency and consistency with Eurocode 2 (EC2) provisions.

The app architecture is organized into three principal layers: the input layer, the computational core, and the visualization layer. The input layer collects geometric

parameters of the footing and column, material properties, load values, soil bearing capacity limits, and economic parameters such as unit concrete cost. These inputs are continuously monitored, and any modification triggers an automatic recalculation of the design solution.

The computational core implements the analytical expressions presented in Section 2. It evaluates the design axial load N_{Ed} , determines the required footing area for each soil bearing capacity level, and iteratively checks compliance with EC2 limit states. Flexural verification, one-way shear verification, punching shear at the 2d control perimeter, and punching shear at the column face are all implemented as independent checks. The modular structure of the code allows each verification to be updated without affecting the overall workflow.

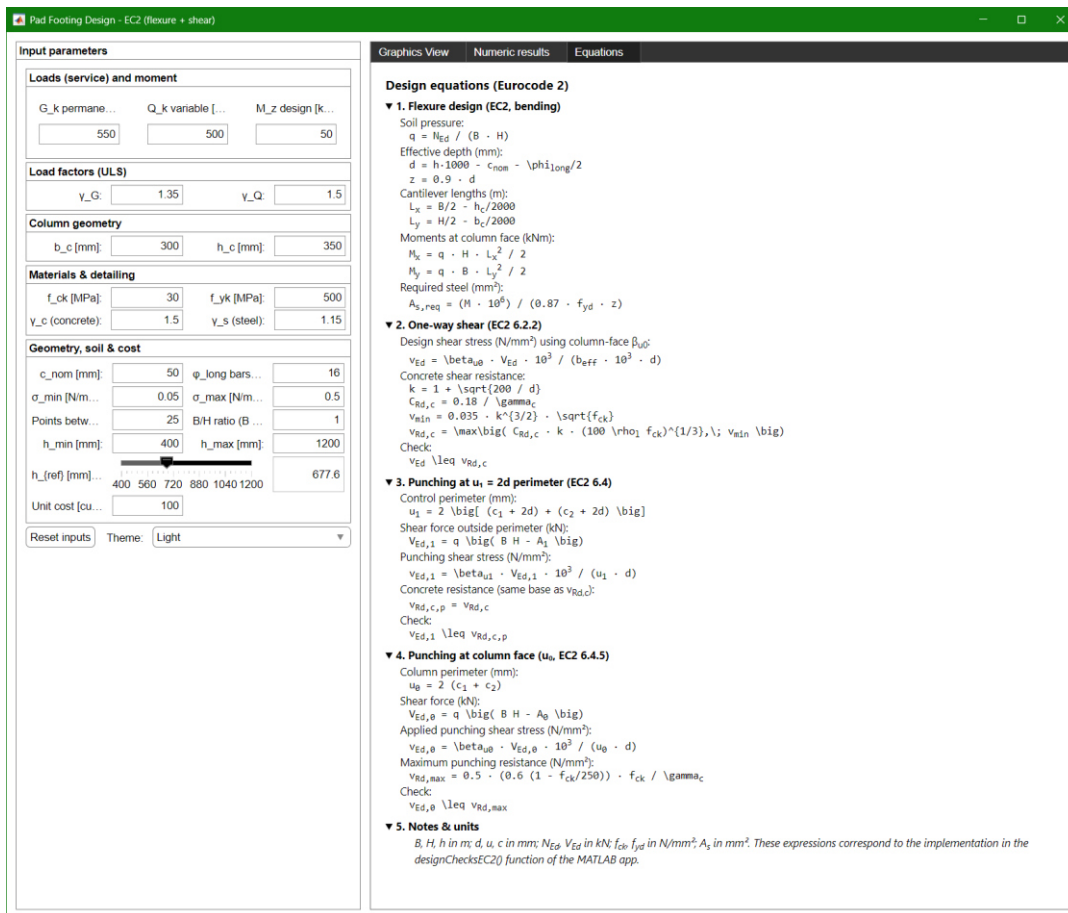


Figure 1: Number of courses on: a) 1st Cycle of studies; b) 2nd Cycle of studies

The visualization layer presents the results in both numerical and graphical form. Two-dimensional plots illustrate the variation of footing volume, cost, thickness, and reinforcement demand with soil bearing capacity, while a three-dimensional sketch provides an intuitive geometric representation of the footing–column–soil system.

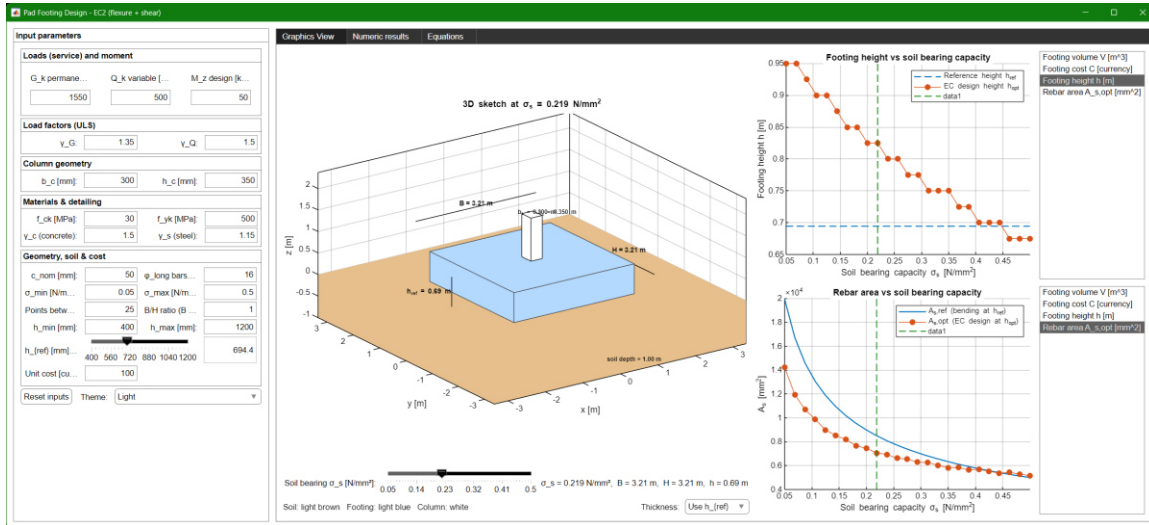


Figure 1: Number of courses on: a) 1st Cycle of studies; b) 2nd Cycle of studies

4. PARAMETRIC ANALYSIS AND OPTIMIZATION STRATEGY

A parametric analysis strategy is adopted to explore the influence of soil bearing capacity on the design of the pad footing. The soil bearing pressure is varied within a user-defined interval, representing realistic geotechnical conditions. For each soil pressure value, the theoretical footing area required to safely transmit the applied loads is calculated.

Once the required plan area is known, the footing plan dimensions are derived according to a prescribed aspect ratio. The footing thickness is then treated as the primary optimization variable. Starting from a minimum allowable thickness, the app incrementally increases the footing depth until all EC2 design checks are satisfied.

The optimal footing thickness h_{opt} is defined as the minimum thickness that simultaneously satisfies flexural resistance, one-way shear resistance, punching shear resistance at the 2d perimeter, and punching shear resistance at the column face. In parallel, a reference thickness h_{ref} is introduced to allow comparison between theoretical bending-based design and full EC2-compliant design.

This parametric framework enables designers to identify trends, critical soil pressure levels, and economically favorable design regions. This level of insight is difficult to achieve through traditional manual calculations.

5. VISUALIZATION AND INTERACTIVE FEATURES

Visualization plays a central role in the developed application. The app provides real-time graphical feedback that enhances the designer's understanding of the structural behavior of pad foundations under varying soil and loading conditions.

Two-dimensional plots depict the relationships between soil bearing capacity and key design outputs, including footing volume, construction cost, footing thickness, and reinforcement area. These plots clearly distinguish between theoretical solutions based on the reference thickness and optimized EC2-compliant solutions.

In addition to 2D plots, a three-dimensional graphical sketch illustrates the footing, supporting column, and underlying soil volume. The sketch updates dynamically as input parameters change and allows interactive rotation, providing spatial insight into the geometric proportions of the system. Color coding and dimension annotations improve clarity under both light and dark visualization modes.

Interactive controls, such as sliders and listboxes, allow users to explore the design space efficiently, making the app suitable not only for professional design tasks but also for educational purposes.

6. CONCLUSIONS AND FUTURE WORK

This study demonstrates that MATLAB can be effectively leveraged as a comprehensive platform for reinforced concrete foundation design in compliance with Eurocode 2. This developed pad footing application offers an efficient alternative to conventional spreadsheet-based approaches. The inclusion of real-time graphical feedback further enhances the designer's ability to interpret and validate results.

Future enhancements may involve broadening the application to encompass additional foundation types, such as strip footing, mat foundations, and pad foundations, as well as integrating seismic design criteria. The primary objective in developing similar applications is not to replace established professional tools widely utilized by the engineering community, but rather to provide designers with supplementary resources that accelerate the iterative processes of structural analysis and design, especially during the early stages of building development.

REFERENCES

- [1] EN 1992-1-1: Eurocode 2 – Design of Concrete Structures – Part 1-1: General Rules for Buildings.
- [2] Mosley, B., Bungey, J., and Hulse, R., Reinforced Concrete Design to Eurocode 2, Palgrave Macmillan.

[3] MathWorks, MATLAB Documentation, The MathWorks Inc.

DESIGN AND OPTIMIZATION OF RC BEAMS ACCORDING TO EUROCODE 2 STAGE 1: DEVELOPING THE CORE DESIGN ALGORITHM

Agon Shaini,
Doctoral Student,
The Institute of Earthquake Engineering and Engineering Seismology,
Shkup, North Macedonia,
shainiagon@gmail.com

Isak S. Idrizi
Faculty of Civil Engineering, University Mother Teresa,
Shkup, North Macedonia,
isak.idrizi@unt.edu.mk

ABSTRACT

Reinforced concrete is very common in building structures because it is strong, long lasting and flexible for different designs. Older methods of designing beams assumed materials behaved in a simple, linear way, which often gave safe but too conservative results.

Today, Eurocode 2 (EN 1992-1-1) uses limit state design, which looks at how concrete and steel really behave and checks both safety and service conditions. This study focuses on designing a reinforced concrete beam using Eurocode 2.

A VBA program in Excel was created to make calculations easier, finding the required reinforcement, bending capacity and shear checks based on loads, size, and materials. This tool can help to design beams more quickly and accurately.

KEYWORDS: Reinforced concrete, RC beam, Eurocode 2, Excel VBA, code-based design

1 Introduction

The design of reinforced concrete has been developed over time. Earlier approaches and methods based on the allowable stresses, assumed materials to behave in linear elastic manner. As these methods are more straightforward, they often have resulted in conservative designs for the real strength of concrete and steel. By the time that knowledge of the engineers have been improved, it became clearer that a more realistic approach is needed and the nonlinear behavior of both materials should be taken into consideration.

Reinforced concrete (RC) is one of the most widely used materials in modern construction due to its good properties to be used for structures. By using reinforced concrete, can build structures from simple ones to more complex high rise buildings and bridges. The material benefits from the well strength properties of the concrete in compression and steel reinforcement properties in tension. These properties make reinforced concrete to provide safe, economical and long lasting solutions for structures.

To design reinforced concrete beams requires calculations of bending and shear capacities, by checking the required amount of steel reinforcement needed and dimensions of cross sections to resist applied loads. Doing all these calculations manually can be time consuming, especially for larger projects or multiple design iterations. The Microsoft Excel tool using VBA (Visual Basic for Applications), is developed to design the beam according to Eurocode 2(EN 1992-1-1). In this paper only the design part is developed, where for the paper in the stage 2, this code will be used and improved more for the optimization of the cross section. The program allows to input beam dimensions, material properties and loads, then automatically calculates the required longitudinal and transversal reinforcement, by checking also the ductility of the cross section. By automating these calculations, this tool can improve efficiently , reduces human errors and provides quick preliminary design.

The focus of this study is to create a tool to design reinforced concrete structures and provide a practical computational tool for structural engineering students and help them understand the algorithm that needs to be followed according to Eurocode 2(EN 1992-1-1) to design the beam cross section. This work shows that using modern design codes with simple programming can make structural design faster and more accurate, by helping to understand better the design of reinforced concrete beam.

2 Algorithm

2.1 Creating the algorithm

To automate the design of RC beams according to EC2, firstly an algorithm is created. The algorithm shown in Fig.1 is made for study and practical design purposes and can help to understand the process that is followed to create the optimization tool. The program evaluates the beam's section by calculating the required reinforcement areas based on design

moments and material strengths.

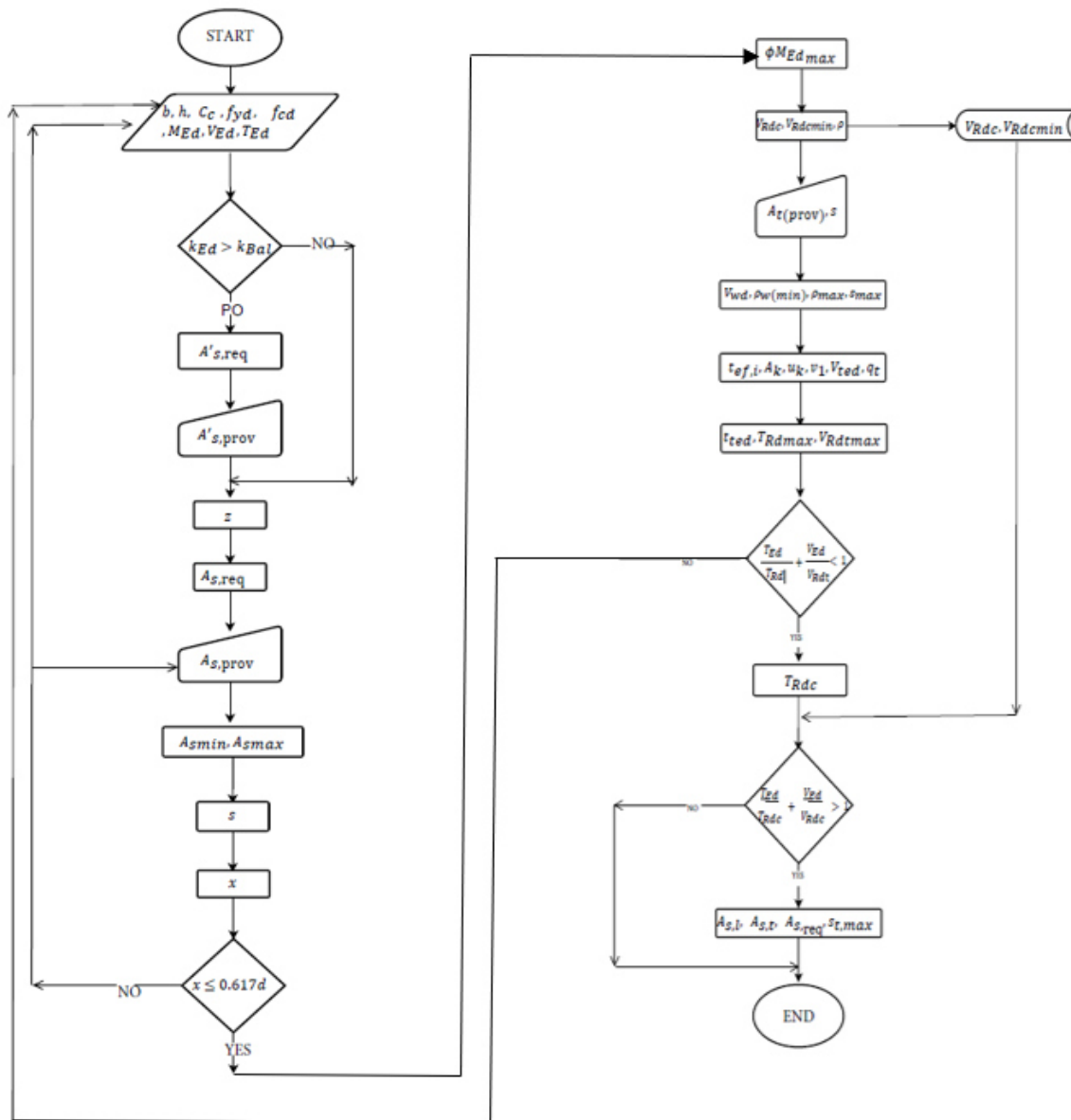


Figure 1. Algorithm for RC beam design according to EC2.

2.2 Developing the code in VBA

The code is designed so that we can input values such as the width and height of the cross section, clear cover, material properties of the concrete and steel, as well as the design moment and shear force. These inputs are shown below in the generated code:

```
b = Application.InputBox("Width of cross section b (cm) =", "Cross section properties",
Type:=1)
```

```
h = Application.InputBox("Height of cross section h (cm) =", " Cross section properties ",
Type:=1)
```

```
Cc = Application.InputBox("Clear cover Cc (cm) =", "Inputs", Type:=1)
```

```
fy = Application.InputBox("Yielding strength of reinforcement fy (MPa) =", "Material
properties", Type:=1)
```

```
fc = Application.InputBox("Compression strength of concrete f'c (MPa) =", " Material
properties ", Type:=1)
```

```
Med = Application.InputBox("Design bending moment Med (kNm) =", "Design loads",
Type:=1)
```

```
Ved = Application.InputBox("Design shear force Ved (kN) =", " Design loads ",
Type:=1)
```

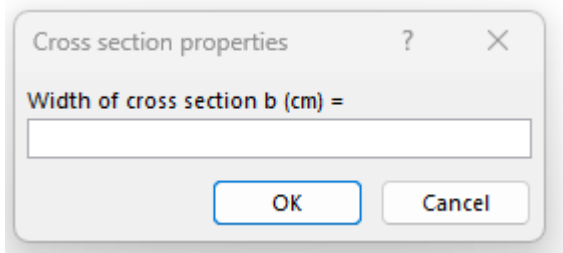


Figure. 2. InputBox created to enter the necessary parameters for calculations.

The beam is designed at the middle of the span, which clarifies that the reinforcement in the lower part of the cross section is in tension, while the upper part is in compression. After providing the inputs for material properties, cross section dimensions and loads, the tool will determine whether a single or double reinforcement layer is required. After checking if the compression reinforcement is required, it will display the results for the required longitudinal and transversal reinforcement. The code below is used to calculate the necessary reinforcement.

The code for longitudinal reinforcement:

```
kbal = 0.167
```

```
d = h - Cc - dia_t / 10 / 2- Effective height of cross section
```

```
ked = (Med * 100) / (b * d * d * fc * 0.1)
```

```
If ked <= kbal Then
```

```
‘Single Layer
```

```
dp = 0
```

```
z = 0.82 * d
```

```

As_req = (Med) / (z * fyd) * 1000
message = "Single Layer Reinforcement"
else
'Double Layer
dp = Cc + dia_c / 10 / 2
z = 0.82 * d
Asp_req = ((ked - kbal) * fc * b * d * d) / (0.87 * fy * (d - dp)) - Tension reinforcement
required
As_req = (kbal * d * d * b * fc) / (0.87 * fy * z) + Asp_req - Compression
reinforcement required
message = "Double Layer Reinforcement"
dia_c- Diameter of provided reinforcement in compression

```

The code for transversal reinforcement:

```

rho = (sl / (b * d)) * 100
Vrdc = (0.12 * (1 + Sqr(200 / (d * 10))) * ((100 * rho * 0.01 * fc) ^ (1 / 3)) * b * d) *
0.1
st = (fyd * As_pervetsuar_t) / (0.567 * fc * b)
z1 = d - st / 2
Vwd = 0.1 * (Av / s_st) * z1 * (fy / 1.15) / 0.83555
Vtot = Vrdc + Vwd

```

After it is determined the required longitudinal and transversal reinforcement, the tool allows the user to input the provided reinforcement in both compression and tension for longitudinal reinforcement, by specifying the number and diameter of rebars and by specifying the diameter and space of stirrups for the transversal reinforcement. It checks whether the provided reinforcement satisfies the required amount of steel and if it does then verifies whether the ductility condition for tension failure is satisfied. If any of these conditions, is not satisfied automatically a message it is displayed by indicating that the provided reinforcement is insufficient or the ductility condition is not achieved, so needs to be rechecked. The process of checking if these conditions are satisfied is done by using loop function.

If $ked \leq kbal$ Then

$d = h - Cc - dia_t / 10 / 2$ - Effective height of cross section

$$k_{ed} = (M_{ed} * 100) / (b * d * d * f_c * 0.1)$$

If $k_{ed} \leq k_{bal}$ Then

‘Single Layer

$$d_p = 0$$

$$z = 0.82 * d$$

$$A_{s_req} = (M_{ed}) / (z * f_{yd}) * 1000$$

message = "Single Layer Reinforcement"

else

' Now will check if the provided reinforcement satisfies the required amount

Do

$A_{s_prov_tens} = n_t * 3.14159 * (dia_t^2 / 4) * 0.01$ ' cm², where n_t and dia_t are the number and the diameter of reinforcements.

If $A_{s_prov_t} < A_{s_req}$ Then

MsgBox "Provided tension reinforcement is not enough!" & vbCrLf & _

"Please recheck the provided reinforcement.", vbExclamation

$dia_t = \text{Take_Diameter}(\text{"Diameter of the reinforcement"})$

$n_t = \text{Take_Number}(\text{"Number of rebars"})$

Else

Exit Do

End If

Loop

Else

$$d_p = C_c + dia_c / 10 / 2$$

$$z = 0.82 * d$$

$A_{sp_req} = ((k_{ed} - k_{bal}) * f_c * b * d * d) / (0.87 * f_y * (d - d_p))$ - Tension reinforcement required

$A_{s_req} = (k_{bal} * d * d * b * f_c) / (0.87 * f_y * z) + A_{sp_req}$ - Compression reinforcement required

message = "Double Layer Reinforcement"

dia_c - Diameter of provided reinforcement in compression

' Now will check if the provided tension reinforcement satisfies the required amount

Do

```

As_prov_t = n_t * 3.14159 * (dia_t ^ 2 / 4) * 0.01
If As_prov_t < As_req Then
    MsgBox " Provided tension reinforcement is not enough!" & vbCrLf & _
        " Please recheck the provided reinforcement.", vbExclamation
        dia_t = Take_Diameter("Diameter of the reinforcement")
        n_t = Take_Number("Number of rebars")
Else
    Exit Do
End If
Loop
' Now will check if the provided compression reinforcement satisfies the required
amount
Do
    As_prov_c = n_c * 3.14159 * (dia_c ^ 2 / 4) * 0.01
    If As_prov_c < Asp_req Then
        MsgBox " Provided compression reinforcement is not enough!" & vbCrLf & _
            " Please recheck the provided reinforcement.", vbExclamation
            dia_c = Take_Diameter("Diameter of the reinforcement")
            n_c = Take_Number("Number of rebars")
    Else
        Exit Do
    End If
Loop
End If

```

The part of code used to check if there is brittle or ductile failure:

```

If ked <= kbal Then
    s = (fyd * As_prov_t) / (0.567 * fc * b)
    x = s / 0.8
Else
    s = ((fyd * As_prov_t) - (fyd * As_prov_c)) / (0.567 * fc * b)
    x = s / 0.8
End If

```

```

If  $k_{ed} < k_{bal}$  Then
  If  $x < d * 0.617$  Then
    failure = "Tension failure"
  Else
    failure = "Compression failure"
  End If
Else
  If  $x < d * 0.617$  Then
    failure = "Tension failure"
  Else
    failure = "Compression failure"
  End If
End If

```

In other paper work named as stage 2, the code will be improved by adding new part for the optimization where it will be obtain the most cost effective dimensions and amount of reinforcement of cross section. By following the figure 1, where the algorithm is shown and part of the code attached here the students can understand better the workflow that needs to be followed to design the cross section according to Eurocode 2(EN 1992-1-1).

2.3 Discussion and conclusion

A case study is done and the results obtained were checked and confirmed to be correct, as it can be seen from graphic included in figure 3 the amount of reinforcement decrease by increasing the height of cross section. In figure 3 is shown the height and reinforcement area chart, while in figure 4 are shown the required and provided reinforcements.

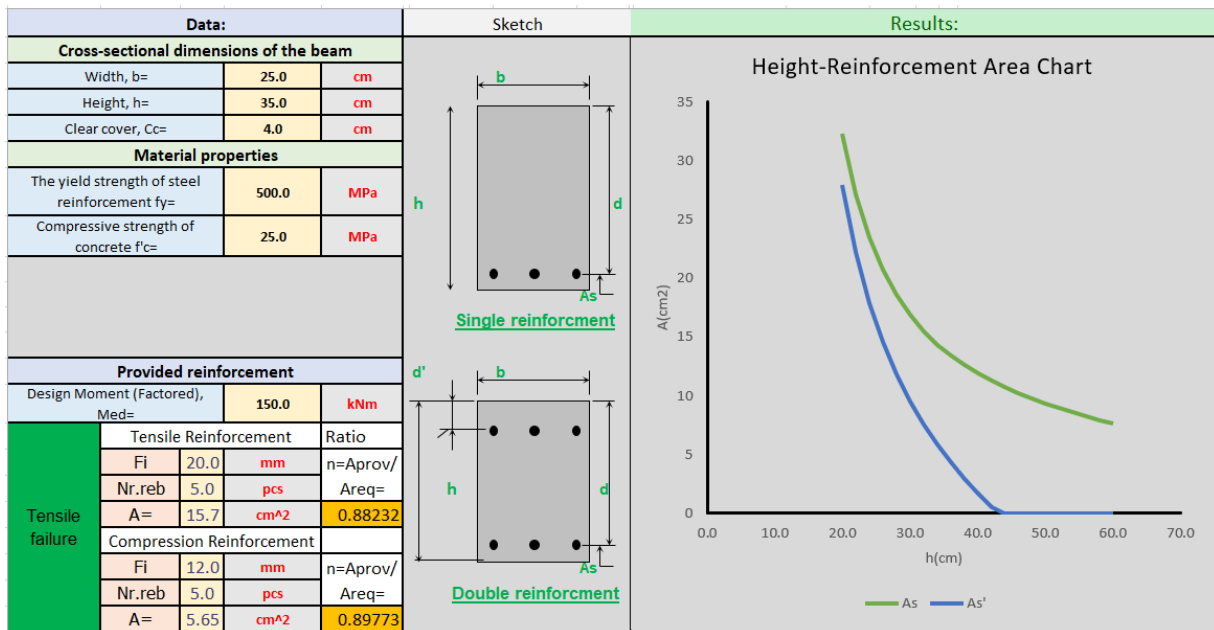


Figure 3. Practical example. Height – Reinforcement area chart

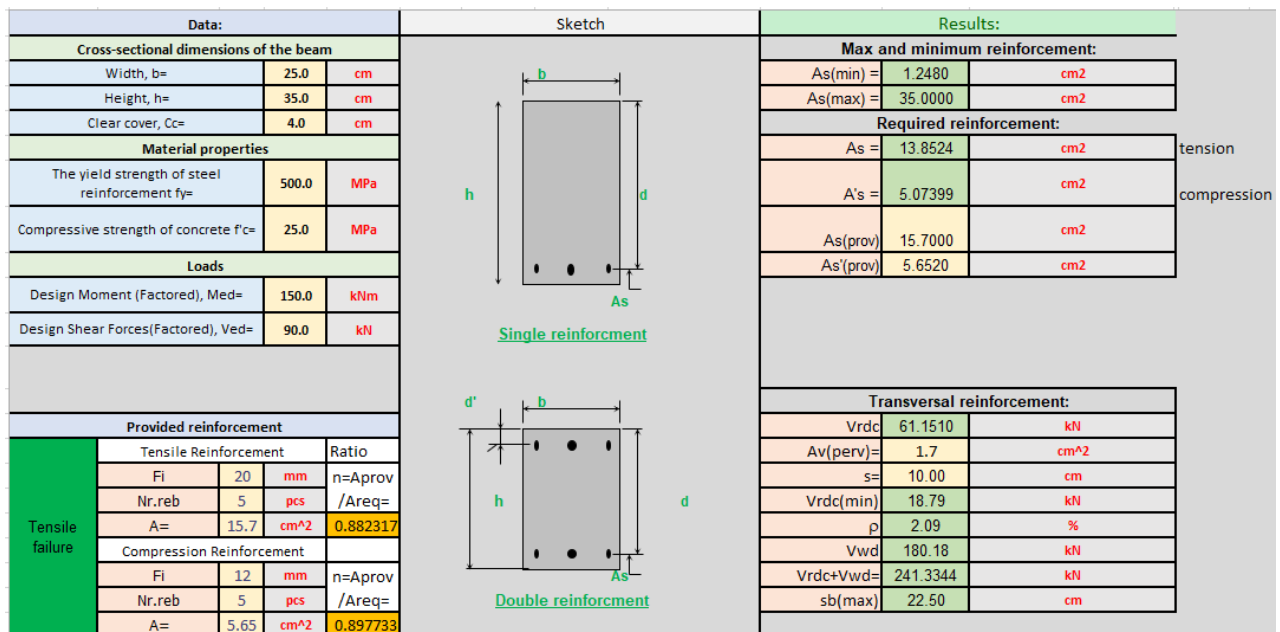


Figure 4. Practical example. Obtained results for required and provided reinforcement

The maximum, minimum, required and provided reinforcements by checking the ductility conditions are all shown in figure 4 automatically by using the algorithm and VBA code created to design the cross section. This paper can help for future steps of optimizing the most cost effective cross section. Students by using this practical tool and change the inputs, can understand better how much effect each parameter has on the capacity and ductility of

cross section.

References

- [1] EN 1992-1-1: *Eurocode 2 – Design of Concrete Structures – Part 1-1: General Rules and Rules for Buildings*, European Committee for Standardization (CEN), Brussels, 2004.
- [2] ACI Committee 318: *Building Code Requirements for Structural Concrete*, American Concrete Institute, Farmington Hills, MI,
- [3] Park, R., Paulay, T.: *Reinforced Concrete Structures*, John Wiley & Sons, New York, 1975.

**DESIGN AND OPTIMIZATION OF RC BEAMS ACCORDING TO EUROCODE 2
STAGE 2: DEVELOPING ADVANCED ROUTINE FOR OPTIMIZATION OF RC
BEAMS IN COMPLIANCE WITH EC2 PROVISIONS**

Agon Shaini
Doctoral Student,
The Institute of Earthquake Engineering and Engineering Seismology,
Shkup, North Macedonia,
shainiagon@gmail.com

Isak S. Idrizi
Faculty of Civil Engineering, University Mother Teresa,
Shkup, North Macedonia,
isak.idrizi@unt.edu.mk

ABSTRACT

This study focuses on developing a practical method for optimizing reinforced concrete (RC) beams using Excel and VBA programming, in compliance with Eurocode 2 (EN 1992-1-1). In stage 1 entitled “Developing core design algorithm” was developed an algorithm inside a VBA code for designing the RC beam cross section, by deriving the maximum, minimum, required and provided reinforcement. In determining these values, it was also checked the ductility condition, as means of assessing whether a tension or compression failure would ensue.

The code in stage 1 was enhanced to optimize cross sections by running hundreds of iterations to identify the most cost-effective design. The tool presents height, width, and reinforcement amounts in charts, allowing users to track how each parameter influences optimization. Ductility is considered by ensuring tension failure conditions are met before showing optimal results, with charts depicting ductility variations across parameters. This tool is intended for rapid preliminary design and optimization, not as a replacement for commercial software.

In the optimization charts, the price is also included, where the total price can be checked overall and compared for each iteration. Overall, this study demonstrates how Excel and VBA can be used to develop a practical, cost saving tool for optimizing RC beams. By combining optimization routines for evaluation of safety and cost simultaneously, a simple yet effective tool is provided to speed up the preliminary stage of analysis and design, thus improving the efficiency of RC beam design.

KEYWORDS: Reinforced concrete, RC beam design, Eurocode 2, VBA, optimization

1 Introduction

In stage 1 entitled “Developing core design algorithm” was developed an algorithm inside a VBA code for designing the RC beam cross section, by deriving the maximum, minimum, required and provided reinforcement. The code used in stage 1, it is developed and improved for optimization of cross sections by calculating hundreds of iterations of height h and width b , until it finds the most cost effective dimensions for the height and width. The height and width have direct impact in the required reinforcement, according to those values the code finds also the most cost effective amount of reinforcement. After obtaining the dimensions of cross sections and amount of reinforcement, the code is design to check if the ductility condition is satisfied, if it won't, then the iteration of calculations will continue until it finds the best results, by displaying these in a message box and generating also the plot.

An engineer's job is not only to design a beam but also to optimize it. Doing all these calculations to find which cross section can cost less manually can be time consuming, especially for larger projects or multiple design iterations. The Microsoft Excel tool using VBA (Visual Basic for Applications), is developed to optimize and design the beam according to Eurocode 2(EN 1992-1-1). The program is fully automated and considers height values ranging from 40 cm to 70 cm and width values from 20 cm to 60 cm. The diameters of the longitudinal reinforcement are 10 mm, 12 mm, 14 mm, 16 mm, 18 mm, and 20 mm. For stirrup reinforcement, diameters of 6 mm, 8 mm, and 10 mm are used. The stirrup spacing is taken as 10 cm, 15 cm, 20 cm, 25 cm, and 30 cm. All these values are selected based on practical considerations and are commonly used in engineering practice. By automating these calculations, this tool can improve efficiently, reduces human errors and provides quick preliminary design.

The focus of this study is to develop a tool for optimizing reinforced concrete beams and to provide a practical computational resource for structural engineering students and practicing engineers who wish to verify calculations. However, this tool is not intended to replace or serve as a commercial software package. This work shows that using modern design codes with simple programming can make structural design faster and more accurate.

2 VBA CODE

2.1 Input and design parameters in VBA

As it was explained in the introduction the code of stage 1 needs to be improved and redeveloped, not only to obtain the required amount of reinforcement but to optimize the cross section. Firstly, the variables are declared

```
Dim b As Double, h As Double, Cc As Double
```

```
Dim fy As Double, fc As Double
```

```
Dim fyd As Double, fcd As Double
```

```
Dim Med As Double, Ved As Double
```

```
Dim d As Double, dp As Double
```

```
Dim As_req As Double, Asp_req As Double
```

```
Dim z As Double
```

```
Dim ked As Double, kbal As Double
```

```
Dim x As Double
```

In this section of the program, several variables are declared to perform calculations for the design of a reinforced concrete beam. The variables are primarily related to the geometry of the section, material properties, and reinforcement requirements. Overall, these declarations set up the necessary variables to calculate the bending and shear resistance of a reinforced concrete beam, by being sure that the section is properly reinforced, safe and compliant with design standards.

```
' Provided reinforcement
```

```
Dim n_t As Integer, n_c As Integer
```

```
Dim dia_t As Double, dia_c As Double
```

```
Dim As_t As Double, As_c As Double
```

```
' Stirrup variables
```

```
Dim stirrupDia As Double, stirrupSpacing As Double
```

```
Dim StirrupBars() As Variant
```

```
Dim StirrupSpacings() As Variant
```

In this part of the program, the reinforcement for a concrete beam is defined. It specifies the number and diameter of the tensile and compression reinforcement, as well as their total areas, to check that the beam can resist bending. The program also includes stirrups, defining their diameter and spacing and allows for varying the number and spacing of

stirrups along the beam to resist shear forces and confine the concrete. These variables together provide a complete layout of the reinforcement needed for safe structural performance.

' Cost variables

Dim volConcrete As Double, weightSteel As Double

Dim costConcrete As Double, costSteel As Double, totalCost As Double

' Optimization tracking

Dim bestCost As Double

Dim bestB As Double, bestH As Double

Dim bestAs_t As Double, bestAs_c As Double

Dim bestDiaT As Double, bestDiaC As Double

Dim bestNT As Integer, bestNC As Integer

Dim bestX As Double, bestFailure As String

Dim bestMessage As String

Dim bestStirrupDia As Double, bestStirrupSpacing As Double

In this part, variables are defined to calculate and track the costs and optimization of the concrete beam design. The volume of concrete and the weight of steel are used to determine the cost of materials, which are then combined to give the total cost of the beam. Additional variables are included to track the best design configuration storing the dimensions of the section, the reinforcement details, stirrup specifications and the neutral axis depth that result in the most efficient and safe design. There are also variables to record any potential failure mode and a message describing the design outcome. This allows the program to not only design a safe beam but also identify the most cost effective and optimal solution.

Dim LongBars() As Variant

LongBars = Array(10, 12, 14, 16, 18, 20) ' mm

StirrupBars = Array(6, 8, 10) ' mm

StirrupSpacings = Array(10, 15, 20, 25, 30) ' cm

In this section, arrays are defined to represent the available reinforcement options. “LongBars” lists the diameters of main longitudinal rebars in millimeters, while “StirrupBars” lists the diameters of stirrup rebars. “StirrupSpacings” specifies the possible distances between stirrups in centimeters. These arrays allow the program to select from standard reinforcement sizes and spacings when designing the beam.

2.2 Developing the code for optimization in VBA

This part of the program runs an optimization loop that varies the beam’s height and width

to find the most efficient design. For each combination, it calculates the required transversal reinforcement, longitudinal tension and if needed, compression reinforcement, checking whether the section is singly or doubly reinforced. The loop is used to find that the beam design satisfies structural requirements while setting up the design for further optimization based on

```

For hi = 40 To 70 Step 1 ' height
  For bi = 20 To 60 Step 1 ' width
    h = hi      b = bi
    ' Compute required longitudinal reinforcement
    d = h - Cc
    ked = (Med * 100) / (b * d ^ 2 * fcd * 0.1)
    z = 0.82 * d
    ' Check if singly or doubly reinforced
    If ked <= kbal Then
      As_req = (Med) / (z * fyd) * 1000
      message = "Singly reinforced section"
      dp = 0    n_c = 0    As_c = 0
    Else
      Dim As_t_req As Double
      As_t_req = (kbal * b * d ^ 2 * fcd) / (0.87 * fyd * z)
      dp = 0
      As_req = As_t_req
      message = "Doubly reinforced section"
    End If
  
```

The tension and compression reinforcement is calculated by following the below code:

```

'Tension reinforcement selection
Dim feasibleT As Boolean
feasibleT = False
For Each it In LongBars
  dia_t = it
  n_t = WorksheetFunction.Ceiling(As_req / (3.14159 * (dia_t ^ 2 / 4) * 0.01), 1)
  If n_t = 1 Then spacingCheck = b - 2 * Cc

```

```

If n_t > 1 Then spacingCheck = (b - 2 * Cc - n_t * (dia_t / 10)) / (n_t - 1)
If spacingCheck >= 2 Then
  As_t = n_t * 3.14159 * (dia_t ^ 2 / 4) * 0.01
  feasibleT = True
  Exit For
End If
Next it
If feasibleT = False Then GoTo NextCrossSection
'Compression reinforcement selection
If ked > kbal Then
  Asp_req = (ked - kbal) * b * d ^ 2 * fcd / (0.87 * fyd * (d - Cc))
  Dim feasibleC As Boolean
  feasibleC = False
  For Each ic In LongBars
    dia_c = ic
    n_c = WorksheetFunction.Ceiling(Asp_req / (3.14159 * (dia_c ^ 2 / 4) * 0.01),
1)
    If n_c = 1 Then spacingCheck = b - 2 * Cc
    If n_c > 1 Then spacingCheck = (b - 2 * Cc - n_c * (dia_c / 10)) / (n_c - 1)
    If spacingCheck >= 2 Then
      As_c = n_c * 3.14159 * (dia_c ^ 2 / 4) * 0.01
      feasibleC = True
      Exit For
    End If
  Next ic
  If feasibleC = False Then GoTo NextCrossSection
End If

```

After checking the tension and compression longitudinal reinforcement, the shear design and ductility conditions checking done by using the part of the code below, where the code checks firstly the concrete contribution and then selects the stirrup diameter and spacing, for each iteration until it finds the most efficient amount.

' Shear resistance

```

sl = 15.7 ' constant
feasibleS = False
' Loop over stirrup diameters
For Each isb In StirrupBars
  ' Area of 2-leg stirrup in cm2
  Av_per_stirrup = 2 * (3.14159 * (isb ^ 2 / 4) * 0.01)
  ' Check spacings from largest to smallest (min steel)
  For i = UBound(StirrupSpacings) To 0 Step -1
    s_st = StirrupSpacings(i)
    rho = (sl / (b * d)) * 100
    Vrdc = (0.12 * (1 + Sqr(200 / (d * 10))) * ((100 * rho * 0.01 * fc) ^ (1 / 3)) * b
* d) * 0.1
    z1 = d - st / 2
    Vwd = 0.1 * (Av_per_stirrup / s_st) * z1 * fyd / 0.83555
    Vtot = Vrdc + Vwd
    If Vtot >= VEd Then
      stirrupDia = isb
      stirrupSpacing = s_st
      feasibleS = True
      Exit For
    End If
  Next i
  If feasibleS Then Exit For
Next isb
If Not feasibleS Then GoTo NextCrossSection
'Ductility check
If ked <= kbal Then
  st = (fyd * As_t) / (0.567 * fc * b)
Else
  st = ((fyd * As_t) - (fyd * As_c)) / (0.567 * fc * b)
End If
x = st / 0.8

```

```

If x < d * 0.617 Then
    failure = "Tension failure"
Else
    failure = "Compression failure"
End If

If failure <> "Tension failure" Then GoTo NextCrossSection

```

The code developed now systematically evaluates different beam geometries to determine the optimal combination of dimensions and reinforcement. Checks ductility, bending and shear resistance and while keeping material consumption low.

2.3 Discussion and conclusion

A case study is done with inputs of external loads $M_{ed} = 300 \text{ kNm}$ and $V_{ed} = 150 \text{ kNm}$. After giving the external loads, the program automatically will calculate if it is double or single reinforcement, optimize the geometry, by giving also the number and diameter of rebars to be used in compression or tension. The total cost is calculated for 1m length.

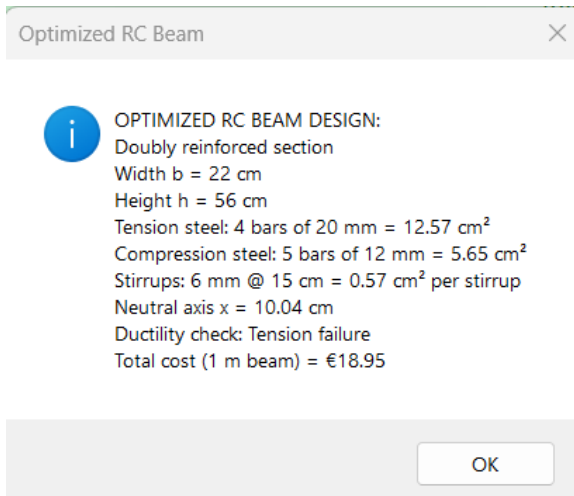


Figure. 1. Display the results in message box

The optimization process was used to find the most efficient beam cross section that can safely carry the applied loads. The final dimensions were chosen to use concrete and steel effectively without wasting material.

Steel reinforcement was placed both at the bottom and the top of the beam to increase its bending capacity. The amount and arrangement of steel were selected to provide enough strength while keeping the design economical. Stirrups were included to safely resist shear forces.

The behavior of the beam shows a safe and ductile response, meaning the steel yields before

the concrete fails, which is desirable in design.

Overall, the optimized cross section meets all strength and safety requirements while achieving the lowest possible cost, confirming that the optimization approach is effective.

References

- [4] EN 1992-1-1: *Eurocode 2 – Design of Concrete Structures – Part 1-1: General Rules and Rules for Buildings*, European Committee for Standardization (CEN), Brussels, 2004.
- [5] ACI Committee 318: *Building Code Requirements for Structural Concrete*, American Concrete Institute, Farmington Hills, MI,
- [6] Park, R., Paulay, T.: *Reinforced Concrete Structures*, John Wiley & Sons, New York, 1975.

JACE

Journal Of Architecture
and Civil Engineering

ISSN (print) 2545-4463

ISSN (online) 2955-2451

Volume 1, Issue 1, 2026