https://doi.org/10.32762/eygec.2025.16

DEVELOPMENT AND ADVANCEMENT OF A GEOTECHNICAL SOFTWARE BASED ON THE FINITE ELEMENT METHOD

Viacheslav POLUNIN[1]

ABSTRACT

This paper examines the application of the Finite Element Method (FEM) for the numerical modeling of soil foundations, which has become the most widely used method in geotechnical engineering practice. The work emphasizes the importance of selecting the appropriate soil model in the context of weak soils, particularly in the central part of Saint Petersburg, where buildings sensitive to uneven deformations are frequently found. It investigates the effectiveness of existing nonlinear soil models, such as Modified Cam-Clay, Soft Soil, and Hardening Soil, among others, as well as the necessity of adapting custom models for FEM applications. The study systematizes and expands upon previous research in this field, focusing on key aspects of finite element analysis, including the assembly of local stiffness matrices and the consideration of boundary conditions. Based on theoretical principles and developed algorithms, an "alpha" version of a computational program enabling efficient linear, transient, and nonlinear analyses using the Finite Element Method was introduced. The findings aim to enhance both education and practical applications in geotechnical design, contributing to the advancement of this area of scientific and engineering research.

Keywords: numerical simulation in geotechnical engineering, dynamic problem, finite element method, transient problem.

INTRODUCTION

Numerical modeling is indispensable for designing structures with significant underground elements or in complex geotechnical conditions, where accurate prediction of soil-structure interaction is critical for safety and cost-effectiveness. While commercial packages (e.g., Plaxis, Midas GTS NX) offer robust solutions, their accessibility is often hindered by high licensing costs and regional restrictions. Open-source alternatives, conversely, demand extensive expertise beyond core geotechnical engineering, including advanced programming and applied mathematics, presenting a significant barrier for practitioners.

To address this gap, this paper presents the development and key advancements of specialized geotechnical software based on the Finite Element Method (FEM). The core innovations and contributions of this research are as follows: First, an integrated computational framework specifically designed for geotechnical analysis has been developed. This solution bridges the gap between the complex requirements of open-source tools and the limitations of commercial software. Second, optimized algorithms have been implemented to enhance computational efficiency for typical geotechnical problems. Third, a modular architecture has been created, enabling future customization and extension by engineers without advanced programming skills.

As the developed software comprises numerous interconnected modules, this paper focuses primarily on presenting and detailing its dynamic analysis module. Details regarding the development of other software components are provided in associated publications. Validation of the dynamic analysis module against benchmark problems and comparison with established solutions demonstrate its accuracy and practical utility.

The developed software offers distinct advantages: it provides a cost-effective alternative to high-priced commercial software while significantly lowering the technical barrier compared to raw open-source projects.

Due to its modular architecture and purpose-built design, the system enables specialists to not only perform advanced FEM computations (including dynamic simulations) but also customize it for specific applications. For instance, users can integrate custom soil models or modify solution algorithms. These capabilities prove particularly critical when addressing non-standard geotechnical scenarios where commercial software often lacks sufficient flexibility.

1 PhD in Engineering, Associate Professor, SPbGASU, Saint-Petersburg, Russia, Senior Researcher, Russian Scientific Research Institute of Building Physics of the Russian Academy of Sciences , Moscow, Moscow, Russia, n1ce2u@yandex.ru

This work advances accessible, adaptable computational tools for the geotechnical engineering community, with a specific emphasis on dynamic modeling capabilities.

METHODS

FEM's construction applications originated with Galerkin (1915), Zienkiewicz (1967), and Bathe (1982). For geotechnics, key contributions came from Fadeev (1987) and Potts & Zdravković's theoretical work (2001). The literature mentioned above describes physical equations, their transition to matrix form, and subsequently to systems of linear algebraic equations. However, such literature often lacks sufficient detail, resulting in significant effort required to practically implement the developments presented. It is particularly important to describe in detail, with examples, how variables and matrix expressions are obtained, boundary conditions are accounted for, and other aspects.

It is important to note that geotechnics encounters almost all types of physical processes and mathematical methods for their description: linear problems, stationary, non-linear, non-stationary, and their combinations. A detailed description of solving filtration problems is provided in (Polunin et al., 2023). The solution of non-stationary temperature problems is presented in (Sakharov et al., 2023). The solution of linear-elastic stress-strain state problems is described in (Polunin, 2023). The methodology and algorithm for solving non-linear problems are presented in (Polunin et al., 2023).

This paper considers the solution of dynamics problems in a different formulation. The general view of the computational scheme is presented in Figure 1.

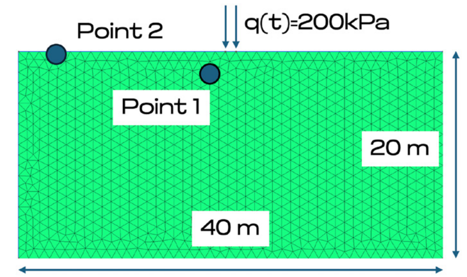

Figure 1 *Calculation scheme of the test problem*

A plane strain problem with dimensions of 40 by 20 m is considered, with a dynamic load of 200 kPa applied over a 2 m width at the center of the computational scheme. The frequency of impact is set at 1 Hz with a sinusoidal pattern, and the calculation time is 1 second. System damping and "viscous" boundaries were not considered; the problem is solved in its basic formulation. The static boundary conditions of the problem are standard: the lateral boundaries are fixed along the X-axis, and the lower boundary is fixed along both X and Y axes. Additional dynamic boundary conditions are not applied. The soil density is 2 t/m^3. The elastic modulus is 10 MPa; Poisson's ratio is 0.2. The numerical model incorporates two monitoring points for comparative analysis. Monitor Point 1 is situated at coordinates (x,y) = (2m,2m) relative to the load boundary. Monitor Point 2 is positioned 18.5m horizontally from the load boundary at ground level. These strategic locations serve as benchmark positions for validating displacement calculations against equivalent simulations performed using the Plaxis 2D finite element software package, thereby enabling quantitative verification of the developed computational framework.

The fundamental equation of dynamics is represented under position 1:

$$M \cdot \frac{d^2u}{dt^2} + C \cdot \frac{du}{dt} + K \cdot u = F(t) \tag{1}$$

Where M is a mass matrix; u is a vector displacement; C is a damping matrix; K is a stiffness matrix; F(t) is a- time-dependent external force vector .

Since a simple case without damping is being considered, equation 1 takes the form:

$$M \cdot \frac{d^2u}{dt^2} + K \cdot u = F(t) \tag{2}$$

The mass matrix for a three-node reference finite element (unit triangle), utilizing a data structure where the displacement vector's first row corresponds to nodal displacement along the x-axis and the second row to displacement along the y-axis, is formulated as follows:

$$\mathrm{M} = \frac{\rho \cdot \mathrm{S_e}}{3} \cdot \begin{bmatrix} 0.5 & 0 & 0.25 & 0 & 0.25 & 0 \\ 0 & 0.5 & 0 & 0.25 & 0 & 0.25 \\ 0.25 & 0 & 0.5 & 0 & 0.25 & 0 \\ 0 & 0.25 & 0 & 0.5 & 0 & 0.25 \\ 0.25 & 0 & 0.25 & 0 & 0.5 & 0 \\ 0 & 0.25 & 0 & 0.25 & 0 & 0.5 \end{bmatrix} \tag{3}$$

ρ is a density, t/m^3; Se is a finite element area, m^2 .

The formulation of the stiffness matrix and right-hand side vector is presented in detail, while the assembly procedure for the global stiffness matrix and global right-hand side vector is thoroughly documented in the research paper (Polunin, 2023).

The second derivative of the displacement vector can be approximated using the central finite difference scheme:

$$\frac{d^2u}{dt^2} = \frac{u_{i+1} - 2u_i + u_{i-1}}{\Delta t^2} \tag{4}$$

The system is assumed to be at rest at the initial time point (t=0), meaning the displacement vector at step i and at the preceding step equal zero. Following a series of mathematical transformations, the computation of the displacement vector can be expressed as:

$$u_{i+1} = 2u_i - u_{i-1} + \Delta t^2 M^{-1}(F_i - K_{u_i}) \tag{5}$$

Python was chosen as the development language for computational modules. This is because the language is relatively easy to learn and offers many ready-made libraries for solving systems of linear algebraic equations and visualizing data and calculation results. NumPy was used for matrix operations; Matplotlib for visualization; and the open-source package GMSH for finite element mesh generation.

RESULTS

Comparative analyses were conducted using both the Plaxis 2D commercial software package and the algorithm developed in this research. The following figures present displacement isoline contours at various time intervals, generated by Plaxis 2D and the Python implementation respectively. Figure 2 shows the contours of the total displacement vector in the computational model at time 0.25 s; Figure 3 shows them at time 0.625 s. Figure 4 illustrates comparative time-displacement curves for monitoring points 1 and 2 along the Y-axis, as computed by both Plaxis 2D and the developed Python algorithm.

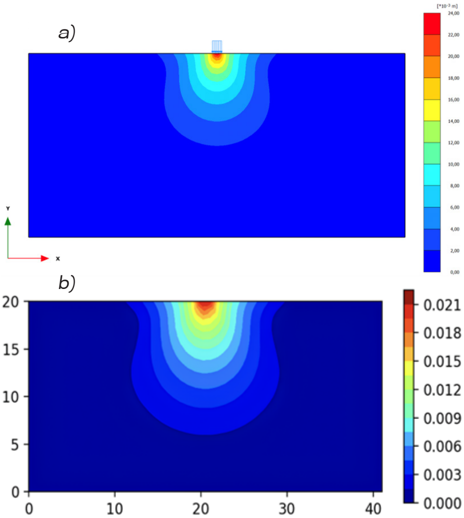

Figure 2 *Displacement contours at time 0.25 s. a) Plaxis 2D; b) Python.*

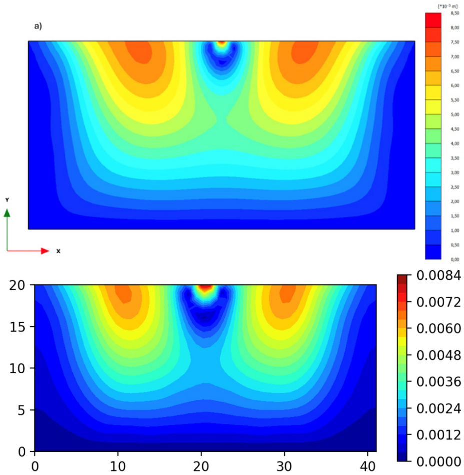

Figure 3 *Displacement contours at time 0.625 s. a) Plaxis 2D; b) Python.*

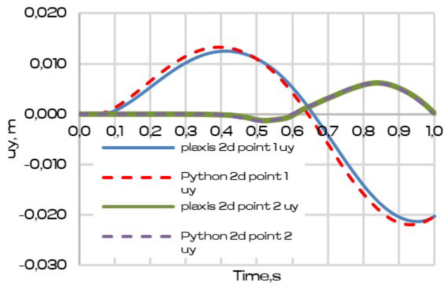

Figure 4 *Displacements of points 1 and 2 along the Y-axis as a function of time*

Figure 5 presents the corresponding temporal evolution of horizontal (X-axis) displacements for the same monitoring points throughout the simulation period.

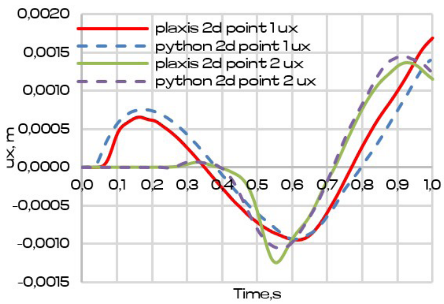

Figure 5 *displacements of points 1 and 2 along the X-axis as a function of time.*

ANALYSIS AND DISCUSSION

Upon critical examination of the obtained computational outcomes, the following conclusions can be drawn:

The results demonstrate both qualitative and quantitative convergence between the developed algorithm and the reference software implementation.

Quantitative discrepancies between maximum and minimum displacement values are summarized in Table 1, providing a statistical basis for validation assessment.

Table 1 *Discrepancy between the calculated results*

Point	1y	1x	2y	2x
Pl_max	0.0105	0.00047	0.00122	0.00062
Py_max	0.0104	0.0005	0.0012	0.00062
Δ	-0.6	0.7	2.0	-1.7
Pl_min	-0.012	-0.0004	-0.0007	-0.0004
Py_min	-0.012	0.0006	-0.0007	-0.0004
Δ	1.3	2.0	0.0	-0.2

The observed deviations in computational results may be attributed to the implementation of lower-order finite elements (three-node triangular elements versus six-node triangular elements utilized in Plaxis). Additionally, temporal discretization methodology differs between implementations: while Plaxis employs the Newmark integration scheme, the present work utilizes a simplified explicit finite-difference time-stepping algorithm.

Subsequent development phases for this computational module will incorporate Rayleigh damping characteristics of soil materials and implement specialized boundary conditions specifically designed for geotechnical dynamic analysis problems .

REFERENCES

Bathe, K. J. (1982). Finite element procedures in engineering analysis, Prentice-Hall, Englewood Cliffs, USA.

Fadeev, A. B. (1987). Finite element method in geomechanics, Nedra, Moscow, Russia (in Russian).

Galerkin, B. G. (1915). Rods and Plates, Series Occurring in Various Questions Concerning the Elastic Equilibrium of Rods and Plates, Vestnik Inzhenerov i Tekhnikov (Engineers and Technologists Bulletin), 19, pp. 897-908 (in Russian), (English Translation: 63-18925, Clearinghouse Fed. Sci. Tech. Info.1963).

Polunin, V. M. (2023). Analytical and numerical methods for determining the stress state of a soil mass when solving a plane problem, Housing Construction, 9, pp. 27-40. https://doi.org/10.31659/0044-4472-2023-9-27-40 (in Russian).

Polunin, V. M., Dyakonov, I. P., Bashmakov, I. B., Bolotov, D. A. (2023). Solution of the problem of steady-state filtration of groundwater by the finite element method, Geotechnics, 15(4), pp. 6-17. https://doi.org/10.25296/2221-5514-2023-15-4-6-17 (in Russian).

Polunin, V. M., Kolyukaev, I. S., Korablev, D. S., Paskacheva, D. A. (2023). Implementation of an ideal elastoplastic soil model in a deformation formulation, Geotechnics, 15(3), pp. 26-42. https://doi.org/10.25296/2221-5514-2023-15-3-26-42 (in Russian).

Potts, D. M., Zdravković, L. (2001). Finite element analysis in geotechnical engineering: Theory, Imperial College Press, London, UK.

Sakharov, I. I., Polunin, V. M., Litvinov, P. V. (2023). Mathematical modeling of changes in soil temperature in the Python programming language, Housing Construction, 1-2, pp. 65-70. https://doi.org/10.31659/0044-4472-2023-1-2-65-70 (in Russian).

Zienkiewicz, O. C. (1967). The Finite Element Method in Structural and Continuum Mechanics, McGraw-Hill, London, UK.

Gmsh (2023). An open software package for generating a finite element grid, [online] Available at: https://gmsh.info/, accessed: 15/04/2024.

Matplotlib (2023). A comprehensive library for creating static, animated, and interactive visualizations in Python, [online] Available at: https://matplotlib.org/, accessed: 15/04/2024.

NumPy (2023). The fundamental package for scientific computing with Python, [online] Available at: https://numpy.org/, accessed: 15/04/2024.

Python Software Foundation (2023). Python programming language, [online] Available at: https://www.python.org/, accessed: 15/04/2024.