Abstract

Numerical methods are used within the field of power engineering to solve a large number of problems. In this context, this paper presents a software library of numerical methods. It focuses on solving several applications in the field of electrical power engineering. The educational software library is designed for students within the field of electrical power engineering at BSc and MSc levels.

Keywords: numerical methods, educational software, power engineering.

1. Introduction

Numerical methods are used within the field of power engineering to solve a large number of problems. In this context, this paper presents a software library of numerical methods. It focuses on solving several applications in the electrical field of power engineering. The educational software library is designed for students within the field of electrical power engineering at BSc and MSc levels. The programs use classic numerical methods dedicated to numerically solve transcendental, algebraic and differential equations, linear and non-linear equation systems, eigenvalues and eigenvectors computing, and numerical processing functions and approximation. These programs synthesize the experience gathered in this field at the Power Systems Department of the Politehnica University Timisoara. The power engineering numerical methods library 4th version contains 10 programs. All of them will be briefly discussed within the full paper. For each of the computing programs several algorithms were implemented and are currently studied by students. They are able to realize a comparison (computing effort, convergence, iterative or direct method, etc.). The
students are able to understand (to see) methods of numerical method applications within the field of power engineering. The current version of the numerical methods software library is version 4. The first version was designed in the Fortran language for Felix C256 and C512 computers in 1975-1986 (Kilyeni, 1985). The second version was designed for Sinclair Spectrum compatible computers (1987-92), using Basic programming language (Kilyeni, 1991). The third version was designed by IBM-PC, for the DOS operating system (1992-1998), using Turbo Pascal and facilities provided by Turbo Vision environment (Kilyeni, 1997). The latest version, designed in the last few years, for the Windows operating system, also uses Turbo Pascal language, but with facilities offered by the Delphi environment (Borland, 1997).

2. Numerical Methods Library General Overview

The optimization library 4th version includes the following computing programs:
- COKEN – used for Kennelly's coefficients computing;
- ECUATIE, POLBAR and ECDIF – used for solving transcendental, algebraic and differential equations;
- MATINFA – used for inverse matrix and matrix factorization computing;
- SISLIN and SISNELIN – used for solving linear and nonlinear equation systems;
- REGRESIE – used for processing numerical functions
- INTEGRARE – used for integrating numerical functions
- VALPRO – computing eigenvalues and eigenvectors.

All computing programs have been developed in a professional way, with a powerful didactic accent. These programs try to respond in a better way to the following requests:
- to assure a uniform character for the structure and utilization of these programs, and efficiently use the facilities offered by the Delphi environment;
- user-friendly interface with an efficient help system;
- professional character regarding the optimization problems solving and elaboration of specialized software;
- to assure some powerful didactic qualities regarding user interface allowing the students to follow the optimization algorithms mechanism;
- uniform structure of user interface for all programs;
- to assure maximum flexibility regarding the visual results: from final results or computing evolution to the most detailed aspects of the computing process;
- to assure easy access to the desired results (final or intermediary);
- to assure the possibilities of creating, actualizing, saving and loading databases specific to applications;
- to intercept and diagnose errors of any nature by the program, avoiding "program crash” due to data errors, or due to those by non-allowed operations or entering an infinite cycle.

Next, we will discuss the essential elements of the optimization software library and user interface.

3. REGRESIE Software

3.1. Program Goal and Solving Methods

The REGRESIE software is designed for the numerical approximation of \( y = f(x) \) function defined by points. The main window is presented in Fig. 1.

It can be also used for situations when the analytical expression of the \( f(x) \) function is known, especially when it is complicated. In this case, it is recommended to replace it with a relatively simple approximation function, but sufficiently "accurate" within the interest domain. The program addresses both the problem itself, and related matters: creation, update, saving and loading of the database.

The approximation of function values is carried-out using the least square method, and considers various forms of approximation for the function \( g(x) \): polynomial of various degrees, exponential and logarithmic (Kilyeni, 2011), (Thangaraj, 2013).
3.2. Power Engineering Domain Application

Let us consider the experimentally determined unloaded characteristic for a synchronous generator (Fig. 2). The aim is to numerically approximate it, and is defined by points. To achieve this goal, different approximation functions (Fig. 3) will be used based on the least square method.
3.3. Initial Data

REGRESIE software requests the following initial data:
- definition of the f(x) function, number of known points and discretization step;
- initial point (x0) value;
- x and y values corresponding to the known data.

3.4. Results

REGRESIE software provides the following results (Fig. 3):
- approximation function expression;
- least square sum;
- approximation function value for the known points and other points;
- intermediary results represented by the coefficients used within the computing process.

3.5. Possible Problems

The program checks the correctness and compatibility of the initial data. The user is informed through specific messages about possible data errors, or problems that can occur during analysis. Possible remedial solutions are offered and the work can be continued or cancelled. The possible messages refer to:
- errors of initial data introduction (with the possibility of immediate correction);
- "Null pivot to solve linear system!" – during the process of solving the linear equation system the module of the current diagonal element is less than 1e-10;
- "There are Y(i) values negative or zero! The exponential approximation is not suitable" – the message may appear only if an exponential approximation function is requested;
- "The logarithmic approximation is not suitable!" – the message may appear only if a logarithmic approximation function is requested.

4. INTEGRARE Software

4.1. Program Goal and Solving Methods

The INTEGRARE software is designed to compute the numerical value of a definite integral; y = f(x) function, defined by points. It can be also used for situations when the analytical expression of the f(x) function is known, especially when it is complicated.

Numerical integration of f(x) function defined by points is carried-out using traditional methods, such as Newton-Cotes (order 1-8) and Gauss-type (order 1-8), or using Newton-type generalized formula (order 1-3) and Romberg method (order 1-2) (Fig. 4) (Kilyeni, 2011), (Thangaraj, 2013).
4.2. Power Engineering Domain Application

Let us consider the load curve for a specific electrical energy consumer (Fig. 5). It is requested to compute real energy consumption based on numerical load curve processing. To achieve this goal, different order Newton-Côtes relations (simple and generalized) will be used, as well as Gauss type relations.

![Daily real load curve for a specific consumer](image)

4.3. Initial Data

INTEGRARE software requests the following initial data:
- definition of the f(x) function, number of known points and discretization step;
- the initial point (x0) value;
- x and y values corresponding to the known data.

4.4. Results

INTEGRARE software provides the following results (Fig. 4):
- integration value;
- integration step;
- interval used for numerical integration;
- known points defining the function.
4.5. Possible Problems

The program checks the correctness and compatibility of the initial data. The user is informed through different messages about possible data errors, or problems that can occur during analysis. Possible solutions for improvement are offered and the work can be continued or cancelled. Possible messages refer to:

- errors of initial data introduction (with the possibility of immediate correction);
- "Null pivot to solve linear system!" – during the process of solving the linear equation system the module of the current diagonal element is less than 1e-10;
- "There are Y(i) values negative or zero! The exponential approximation is not suitable" – the message may appear only if an exponential approximation function is requested;
- "The logarithmic approximation is not suitable!" – the message may appear only if a logarithmic approximation function is requested.

5. ECUATIE Software

5.1. Program Goal and Solving Methods

Solutions of nonlinear, transcendent or algebraic equations are represented within several applications in technical fields, particularly in power engineering.

Nonlinear equations represent the most frequent applications in the field of power engineering. In the great majority of these types of applications the expression of the f(x) function is complicated, or its coefficients are not "exactly" known (they are determined based on experimental nature, considering different simplified hypotheses). Thus, "approximate" equation solving is discussed. To achieve this goal numerical equation solving methods are used, such as Newton type methods, numerical approximation methods, or methods that operate on the interval containing the equation solution. Thus, a previous solution has to be separated within a specific interval (Kilyeni, 2011), (Thangaraj, 2013).
5.2. Initial Data

ECUATIE software requests the following initial data (Fig. 6):
- definition of the \( f(x) \) function and its 1st and 2nd order derivatives;
- definition of the \( g(x) \) approximation function;
- definition of the interval containing the solution;
- maximum number of iterations and precision.

5.3. Results

ECUATIE software provides the following results (Fig. 7):
- equation solution and function value for that value;
- intermediary results represented by the solution value for each iteration, function value for these points, other numerical values corresponding to different coefficients specific to the used numerical solving method.

6. Conclusions

The computing programs described above are part of a software library that represents the background to laboratory classes of the course "Numerical methods and data structures in power engineering". This software tool has an intuitive graphic interface and is very easy to use, and fully benefits from the facilities offered by the Delphi visual development environment (visual programming). The user is guided step-by-step to solve the problem for which the program is intended.

References

Borland Delphi 7. (1997). Object Pascal Language Guide, Visual Component Library Reference, Borland International.
Kilyeni, S. (1985). Optimization techniques and computers within the electrical power engineering, Timișoara: Politehnica University.
Kilyeni, S. (2011). Numerical methods in power engineering. Applicative lectures, Timișoara: Orizonturi Universitare.
Kilyeni, S., Fülöp, Zs., & Dumitrescu, A. (1997). Numerical methods in power engineering. Applicative lectures, Timișoara: Orizonturi Universitare.
Kilyeni, S., Negru, C. (1991). Numerical methods in power engineering. Applicative lectures, Timișoara: Politehnica University.
Thangaraj P. (2008, 2013). Computer Oriented Numerical Methods, Prentice Hall of India Pvt Ltd.