

A Software Tool based on the Finite Element Method for Electromagnetic Education

*Diego Sánchez-Repila, Luis E. García-Castillo

Departamento de Teoría de la Señal y Comunicaciones, Universidad de Alcalá
Escuela Politécnica, Campus Universitario, Ctra. Madrid-Barcelona Km. 33.600
28806 Alcalá de Henares (Madrid), Spain. Phone: +34-1-8856720.
Fax: +34-1-8856699. Email: luise.garcia@uah.es

Abstract

A software tool for electromagnetic education is presented. It allows the analysis of several electromagnetic problems at microwave and millimeter-wave frequencies. Portability has been taken into account in its development, being available at the present time for *Linux* and *Windows* platforms. The graphic user interface is based on a general purpose pre- and post-processor for computer analysis. Electromagnetic analysis is based on stand-alone Finite Element Method programs developed for research purposes. The flexible architecture of the software developed allows an easy inclusion of new FEM kernels and, thus, the expansion of the range of electromagnetic problems that may be analyzed.

1 Introduction

The objective of this work is to provide the students with a software tool that allows them to analyze a wide variety of electromagnetic problems at microwave and millimeter-wave frequencies. Thus, the students, through the visualization of the electromagnetic field inside/around the structures analyzed and the computation of the different quantities of interest (e.g., the propagation constants of the modes inside a waveguide, the resonance frequencies of a microwave cavity, the scattering parameters of a junction, the radar cross section of an object, radiation pattern of an antenna, and so on), are able to get a better insight into the different electromagnetic phenomena involved in the analysis (guiding, scattering, and radiation of waves, resonances, and so on).

Due to the variety and complexity (inhomogeneous and anisotropic media, non-canonical geometries, and so on) of the structures that are wanted to be analyzed, a numerical method must be used. The Finite Element Method (FEM) has been chosen for this purpose because its accuracy and capability to handle arbitrary geometries and media. The software tool presented here makes use of the FEM computer codes developed in the Research Group that the authors belong to. These codes cover a wide variety of electromagnetic problems including the full wave analysis of multiconductor transmission lines and general waveguiding structures, three-dimensional (3D) cavities, waveguide discontinuities, and 3D radiation and scattering problems [1]. The codes make use of advanced features as higher-order curl-conforming finite elements or novel FEM mesh truncation procedures. The codes are programmed in Fortran and they make use of some procedures from MODULEF Library [2] which source code is freely available. MODULEF has nice features as a modular and flexible approach to finite element programming, allowing, for instance, virtually any kind of finite element type or boundary conditions to be defined. However, the interaction with the user is far from friendly, specially in those tasks related to the pre- and post-process stages of the FEM analysis, i.e., the specification of the geometry, materials and the generation of the mesh, and the visualization of the results and fields, being its graphic capabilities seriously limited. Thus, a software tool made only of these codes, although powerful and flexible, is not suitable to be used by the students.

*This work has been supported by the Ministerio de Ciencia y Tecnología of Spain under project TIC2001-1019

With the purpose of providing a powerful but also an attractive and easy to use software tool to the students, a friendly graphic user interface has been developed. As most of the interaction time of a FEM program with the final user happens during the specification of the geometry and material of the structure to be analyzed, and also on the visualization of the results, the user interface has been developed based on a graphic pre- and post-processor for computer analysis. Specifically, GiD pre-postprocessor [3] has been chosen because, in addition to its graphics and mesh generation capabilities, it has been designed as a general purpose tool that can be easily adapted to any *problem type* allowing our own FEM codes to be run within the GiD interface. An academic version (limited to 700 and 3000 elements for 2D and 3D, respectively) is freely available. That is enough for many educational applications. However, the professional version is available with important discounts for educational organizations.

Portability has been an important issue in the development of the presented software. FEM kernels codes and software modules to interact with GiD for each problem type are written in standard programming languages available in a wide variety of computer platforms. Input and output data files are in ASCII format. With respect to GiD, it is available at the present time for Silicon Graphics, Win95/98/NT/2000/ME/XP, Digital alpha UNIX, and Linux, platforms. Specifically, the software tool presented here is available for Linux and Windows systems, which are the most common options among students. However, main development is done on Linux.

It is worth noting that, due to the features mentioned above, the FEM software tool presented here is not only limited to the computation/visualization of the electromagnetic field in simple structures for a better understanding of electromagnetics, but it is also a powerful computational tool that allows the analysis of quite complex structures. Thus, it is suitable for design tasks in advanced undergraduate and graduate courses.

At the present time, it may be used for the full wave analysis of arbitrary waveguiding structures (including computation of frequency-dependent parameters of multiconductor transmission lines and self-adaptive mesh algorithms) and the resonance analysis of 3D cavities. FEM solvers make use of higher-order curl-conforming elements: straight tetrahedra for 3D, straight and curved hybrid Lagrange/curl-conforming triangles for 2D. First and second order elements are being used; third-order element is planned to be introduced very soon. The analysis of general 3D radiation and scattering problems is also planned to be included in the near future.

2 Usage and Implementation Details

A diagram showing the different stages that happen in the FEM analysis of a particular problem with the software presented here may be seen in Fig. 1. To illustrate the features of the software, the analysis of a ridge cavity has been chosen as an example.

First, there is the pre-processing stage where the geometry, materials, and boundary conditions of the problem are defined, and the FEM mesh is generated. This is done within the GiD user interface (geometry/mesh may be also imported from several CAD format files). GiD mesh module only allows structured (quadrangles and hexaedra) and unstructured (triangles and tetrahedra) mesh types to be generated with first or second order Lagrange elements. However, as it is explained later in this section, the limitations of GiD mesh generator may be overcome with the coding of an appropriate mesh export module. Obviously, the definition of boundary conditions and materials depends on the kind of analysis performed, being the customization of GiD for each problem type done by means of the specification of a few ASCII files. The selection of the problem type is done as shown in Fig. 1 (upper left screenshot). An example of the interface for the assignation of boundary conditions on the structure is shown also in Fig. 1 (lower left screenshot).

Next stage is the FEM calculation of the electromagnetic solution of the problem. The calculation module is a set of stand-alone FEM programs, one for each problem type, that can be run within the pre-processor interface. These programs are coded using Fortran language. The input data to these programs consists on two files: one file containing the mesh and other data file containing

information about materials, boundary conditions of the problem, and several parameters needed for the analysis. The later is in ASCII format and usually is about a few ten lines long. The former is in binary format, specifically the binary format used by MODULEF Library. However, there are some utilities available to convert between binary and ASCII MODULEF formats.

The input data to the FEM programs is automatically generated from the pre-processing stage data. When the *Calculate* option is selected from the pre-processor menu, two modules are run. These two modules must be provided for each problem type to be solved. First module extracts, making use of its own pseudolanguage, the pre-processor data to a file. The second module is a shell script that is used to run the FEM program. But previously to that, the shell script runs the software codes that generate the data in the format needed by the FEM programs, i.e., the mesh file in MODULEF binary format (written in C and Fortran languages to speed up the process for large meshes) and the ASCII input data file mentioned in the above paragraph (written in TCL/TK [4], a scripting language available in many platforms). It also runs the *graphic data presentation* module (see Fig. 1) where the input data are presented to the user through a graphic interface, which may also be used to modify them. The mesh export code is common for every problem type. However, the ASCII data file generation and the *graphic data presentation* codes are dependent of the FEM program to be run. A common framework has been developed in order to make easy the inclusion of new FEM programs. It is worth mentioning here some features of the conversion between the GiD and MODULEF mesh formats. First, a binary MODULEF mesh file is generated from the GiD mesh. Some mesh modification modules (from the MODULEF Library) are further applied in order to include curl and/or div conforming elements of arbitrary order, and to divide quadrangles and hexaedra into triangles and tetrahedra, respectively. Thus, most of the limitations of GiD mesh generator are overcome.

Last stage of the analysis is the visualization of the results, typically scalar and vector field plots (power, electric and/or magnetic fields) and some graphs with numerical data. This is done within the GiD postprocess interface which allows a wide variety of graphic possibilities, e.g., contour fill and vector field plots are shown in Fig. 1. An export module, which is run when the post-processing interface is activated, has been developed to convert the output data to GiD format. Note that a conversion from internal MODULEF mesh file format to GiD mesh format is needed when the mesh that has been used in the FEM computation is different from the one generated by GiD (different element type and/or different geometry).

3 Conclusions

A portable software tool for electromagnetic education has been presented. Usage and implementation details have been described showing the main features of the software, i.e., portability, a friendly graphic interface based on a general purpose pre- and post-processor, powerful and flexible electromagnetic analysis based on stand-alone FEM programs (developed for research purposes) run from the user interface, and easy to expand by the inclusion of new FEM kernels. .

References

- [1] M. Salazar-Palma, T. K. Sarkar, L. E. García-Castillo, T. Roy, and A. R. Djordjevic, *Iterative and Self-Adaptive Finite-Elements in Electromagnetic Modeling*, Artech House Publishers, Inc., Norwood, MA, 1998.
- [2] “The MODULEF Library”, Institut National de Recherche en Informatique et en Automatique (INRIA), URL:<http://www-rocq.inria.fr/modulef/>.
- [3] GiD: The Personal Pre and Postprocessor, ”, International Center for Numerical Methods in Engineering (CIMNE), URL:<http://gid.cimne.upc.es/>.
- [4] John Ousterhout, *Tcl and the Tk Toolkit*, Addison-Wesley.

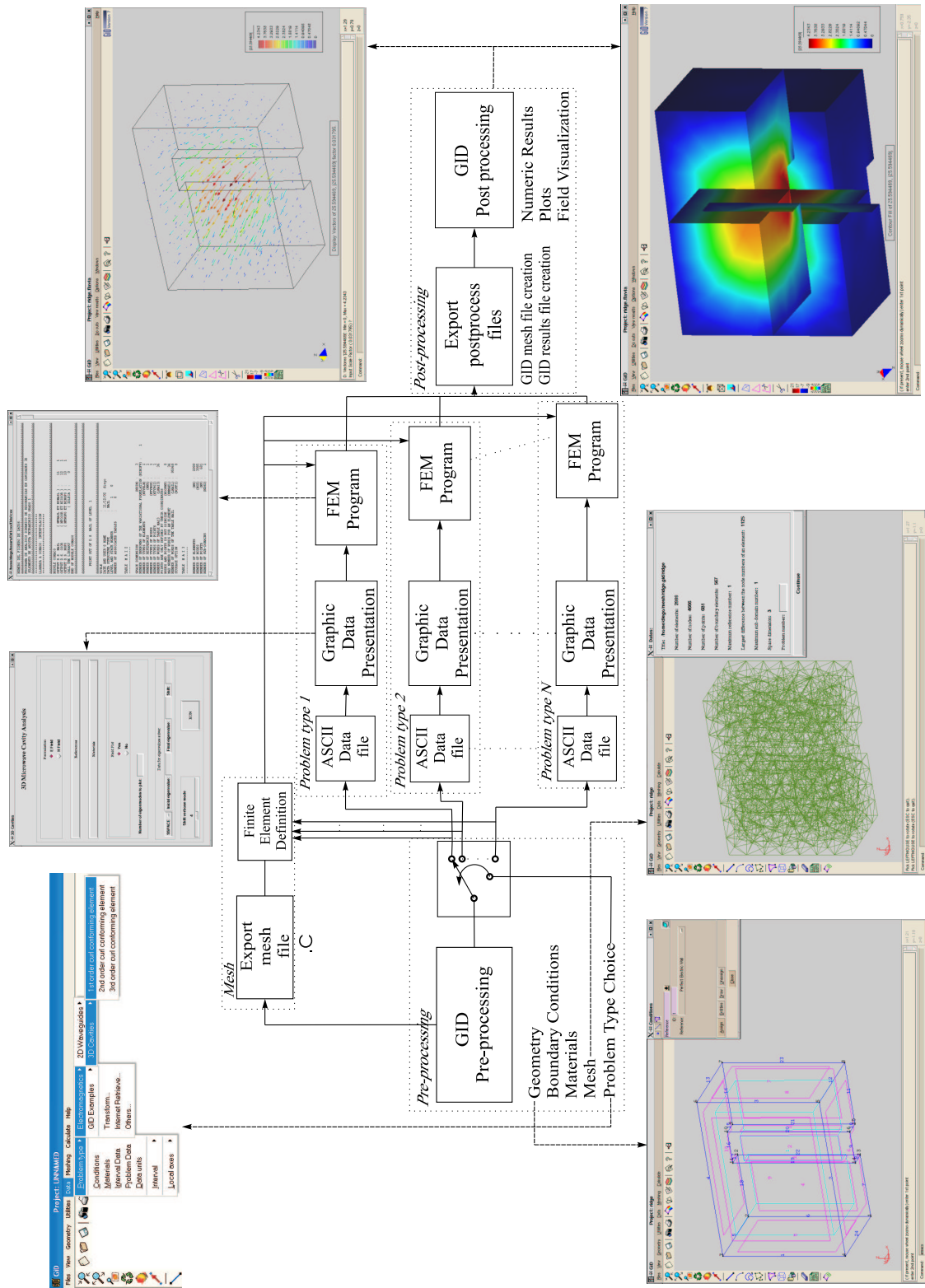


Fig. 1: Implementation diagram of developed software