ADAPTING GID FOR ELECTROMAGNETICS

Diego Sánchez-Repila and Luis E. García-Castillo

[1] 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

Key words: Finite Element Method, Electromagnetics, Curl-Conforming Finite Elements, Self-Adaptive Mesh, .

Abstract. *The objective of this paper is to show and share the experiences on the use of GiD as a pre- and post- processor for some of the electromagnetic FEM codes developed in the Research Group that the authors belong to. GiD has advantageous features but important limitations for electromagnetic problems, mainly due to the unavailability of non-standard (H(curl) and H(div) —low and higher orders—) finite elements. At the present time, several FEM codes have been integrated, specifically, those for the full wave analysis of arbitrary waveguides and multiconductor transmission lines and the eigenvalue analysis of three dimensional cavities. Due to the process standardization, new FEM codes are easily included.*

1 INTRODUCTION

The objective of this paper is to show and share the experiences on the use of GiD as a pre- and post- processor for some of the electromagnetic FEM 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 2D 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 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 postprocess 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. For this last deficiency, and its architecture that allows an easy integration of stand-alone codes, we have resorted to GiD [3].

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

The use of GiD, has reported us great advantages for its well know features. However, for an electromagnetic analysis, GiD has important limitations, mainly due due to the unavailability of non-standard (H(curl) and H(div) —low and higher orders—) finite elements. These limitations have been overcome with an appropriate *mesh export module*, which transforms the second order Lagrange elements provided by GiD, to non-standard elements (H(curl) and H(div)) needed for the analysis.

At the present time, the software tool developed making use of GiD may be used for the full wave analysis of arbitrary waveguiding structures (including computation of frequency-dependent parameters of multiconductor transmission lines [4] and self-adaptive mesh algorithms [5]) and the resonance analysis of 3D cavities [1]. 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 2D and 3D radiation and scattering problems is also planned to be 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 preprocessing stage where the geometry, materials(*mat extension*), boundary conditions (*cnd extension*) and specific data of the problem to be solved are defined (*prb extension*), and the FEM mesh is generated. Obviously, the definition of boundary conditions and materials depends on the kind of analysis performed.

Next stage is the FEM calculation of the electromagnetic solution of the problem (*Calculate* option is selected). In electromagnetics, the finite elements appropriate for the discretization of the electric and magnetic fields and the electric and magnetic inductions are the curl and div conforming finite elements, respectively. These elements impose among elements the continuity of the tangential (curl-conforming) or normal (div-conforming) component of the vector unknown and allow the jump of the other components. However GiD only allows to generate first or second order Lagrange element meshes. This limitations of GiD mesh generator may be overcome with the coding of an appropriate *mesh export module* (see the diagram of Fig. 1). The input to this module is an ASCII file generated by the *problem_type.bas* with geometrical, topological and connectivity information about the mesh. Next, a C language program, places this data into an ASCII MODULEF format file [2]. This C coded program has to determine the references (boundary conditions) to each point, edge and face of each element as demanded by MODULEF format. In order to facilitate this process and to reduce the number of ambiguities, the GiD mesh generator always works with second order Lagrange elements. Thus, for example, each edge reference is the reference of each second order node.

The next stage is the use of some MODULEF utilities to convert from ASCII format to the mesh binary MODULEF format (called NOPO). In this last process, the mesh is modified including an arbitrary type of finite element (in terms of number and position of nodes) according to the *Finite Element Definition* made in the *problem type choice* stage (see again diagram of Fig. 1). Thus, new connectivity information has to be generated. This is typically used for the generation of curl and/or div conforming elements. The NOPO file from the mesh export module output is read by the FEM program (written in Fortran). The other input data to the FEM program is an ASCII file containing information about materials and boundary conditions, and also several other parameters needed for the calculations. This data file is created automatically from information written by *problem_type.bas*. It is worth mentioning at this point that an stand-alone mesh export (from GiD to NOPO format) utility has been created as a *problem type* in order to facilitate the FEM kernel development.

The 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 (Fig. 1). An export module, which is run when the postprocessing 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

The use of GiD as a pre- and post- processor for the FEM analysis of electromagnetic problems represents an attractive option. However, some work needs to be done, mainly in the preprocess stages, in order to adapt GiD for electromagnetics. This paper has shown the work done by the authors to integrate several electromagnetic FEM codes within the GiD interface.

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] M. C. Jiménez-González, L. E. García-Castillo, and M. Salazar-Palma, "Computation of characteristic impedanceand losses of multiconductor anisotropic transmission lines using the finite element method", in *IEEE Antennas and Propagation Society International Symposium (AP-S),* Salt Lake City, UTAH, USA, July 2000, pp. 1172–1175.

[5] M. Salazar-Palma and L. E. García-Castillo, *Finite Element Software for Microwave Engineering,* chapter "Self-Adaptive Procedures for Waveguiding Structures Analysis", pp. 401–432, Wiley Series in Microwave and Optical Engineering. John Wiley & Sons, Inc., 1996.

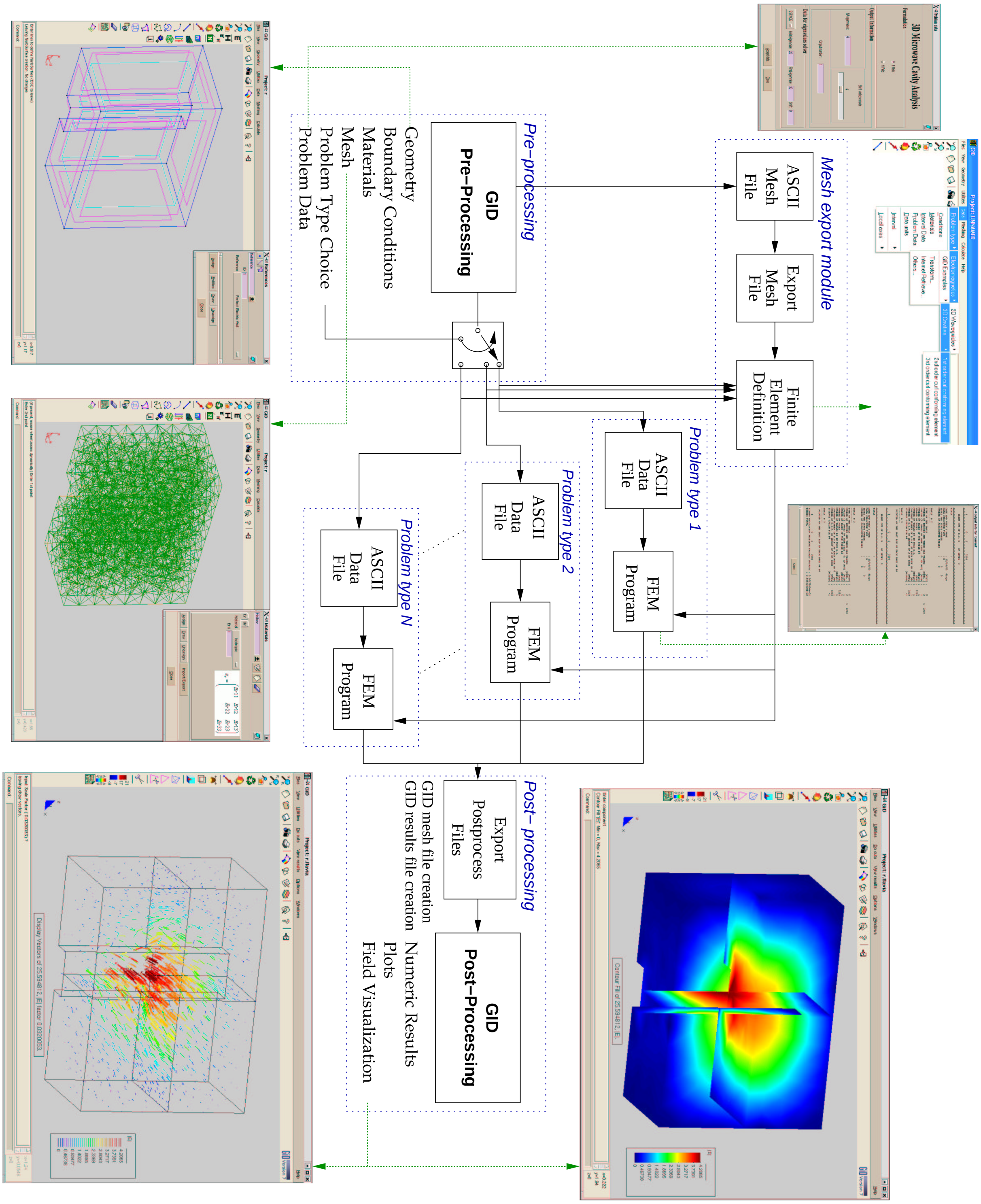

Figure 1: Implementation diagram of developed software