HOBBIES: ELECTROMAGNETIC SIMULATOR USING GID

Daniel García-Doñoro*, Yu Zhang†, Weixin Zhao†, Tapan K. Sarkar†, Luis E. Garcia-Castillo* and Magdalena Salazar Palma*

* Dpto. de Teoría de la Señal y Comunicaciones, Universidad Carlos III de Madrid. Escuela Politécnica Superior (Edificio Torres Quevedo), Avenida de la Universidad, 30, 28911 Leganés, Madrid, Spain
E-mail: dgdonoro@tsc.uc3m.es, luise@tsc.uc3m.es, salazar@tsc.uc3m.es

† Department of Electrical Engineering and Computer Science, Syracuse University. 335 Link Hall, Syracuse University, Syracuse, NY 13244 USA
Email: yzhang39@syr.edu, wzhao04@syr.edu, tksarkar@syr.edu

Key words: Electromagnetic, Simulator, Interface, GiD, HOBIES, MoM

Abstract. *This paper introduces the graphical user interface of a new electromagnetic solver called HOBBIES (Higher Order Basis Based Integral Equation Solver). This graphical user interface is based on the personal pre- and post- processor called GiD®. GiD® is used to create the geometrical and topological information of the model, set the electromagnetic variables needed by the solver (frequency, excitations, etc.) and define the run environment of HOBBIES. Then, HOBBIES is called through GiD® and results are returned by the solver. Finally, results are displayed using GiD®. In addition, this interface allows GiD® to support some important features such as design of models using symbolic variables called "symbols", automatic goal oriented optimization or the option to execute simulations on a high performance parallel computing cluster.*

1 INTRODUCTION

Nowadays, there is a large variety of electromagnetic simulators that are being used by researchers in various fields. These electromagnetic simulators are based on different numerical methods such as Method of Moment (MoM), Finite Element Method (FEM) or Finite Different Method (FD). Also, these simulators present different features such as multiplatform and multilanguage environments, friendly graphical user interfaces (GUI), compatibility with computer aided design (CAD) software package or/and automatic goal oriented optimization.

In this context, the use of GiD® to develop a new electromagnetic simulator allows to integrate and improve most of the above mentioned features. However, GiD® presents an important problem when the user wants to use an automatic goal oriented optimizer since doesn't support the use of symbolic variables or "*symbols*" to create a model. Thus, a framework that supports the use of these symbols to create a model is required to develop a new electromagnetic simulator based on GiD®.

This paper presents an interface between GiD® and a new electromagnetic solver called HOBBIES (High Order Basis Based Integral Equation Solver) [1-2]. A problemtype has been developed to provide a framework that supports the definitions of models using symbols enabling the use of an automatic goal oriented optimizer. In addition, this problemtype provides a simple and easy to use graphical interface using specific procedures created through TCL-TK and TKWidget [3]. Thus, a graphical and comfortable environment has been developed to configure and execute simulations on a high performance parallel computing cluster. Several tabs of the setup environment window are shown in Fig. 1 and 2. Using the hosts tab (see Fig. 1), the users may configure their high performance cluster where the solver is executed. Analogously, the users may configure the parallel environment of the solver using the MPI tab (see Fig. 2).

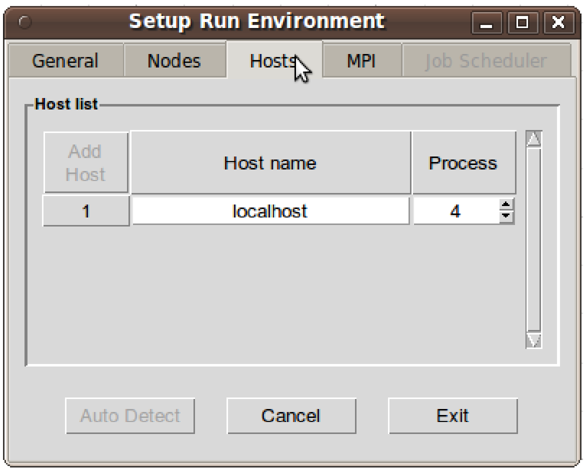

Fig. 1 Hosts tab of the setup environment window

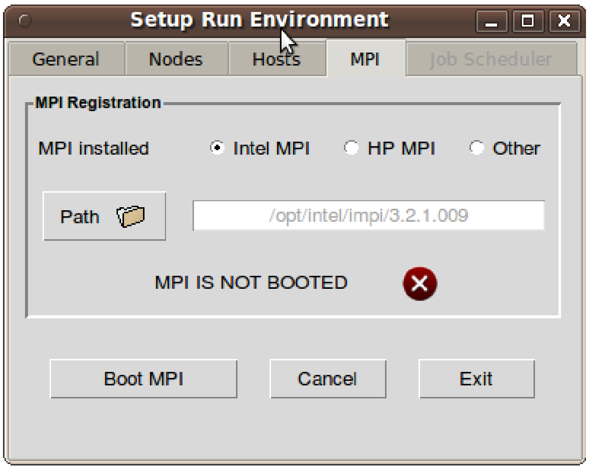

Fig. 2 MPI tab of the setup environment window

2 USE OF SYMBOLS WITHIN GID

As mentioned above, the problemtype provides new features to GiD® such as the definition of models using symbols enabling automatic goal oriented optimization. This section is focused on explain how GiD® may support the use of symbols to create a model. Also, an example of an optimization problem is shown in Fig. 3 to 9 to illustrate how we can run these kinds of problems within GiD®.

In general, the definition and/or modification of a model in GiD® must be performed using numbers. For instance, the user must use real numbers to define the coordinates of a point or to define the vector used to move a model from one point to another. In this context, one of the most important problems to use symbols in GiD® is that tools such as translation, rotation or scale, support the use of these symbols. We consider that is important to create a controlled environment where the users may define and use their symbols. Thus, the developer may know when the user is defining or using a symbol and what features of GiD® is going to employ. To create this controlled environment we have developed a window that allows the users to create, delete and/or modify symbols (See Fig. 4). Also, we have created several windows that allow the users modify the most important properties of nodes, lines or objects through numbers or symbols. These windows contain data entries that allow the users change the value of some properties of the model, such as the coordinates of the points, the length of a cylinder or the radius of a sphere. One example of these windows is shown in Fig. 3. This window allows the users define the coordinates of nodes through number or symbols. Once the symbol is defined and assigned to a certain property, the symbolic value of this symbol is exchanged for its real value before the required operation is executed by GiD®.

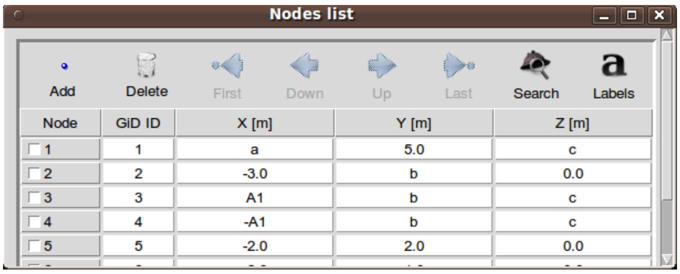

Fig. 3 Nodes list window

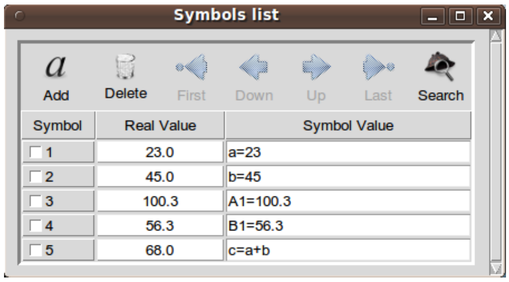

Fig. 4 Symbols list window

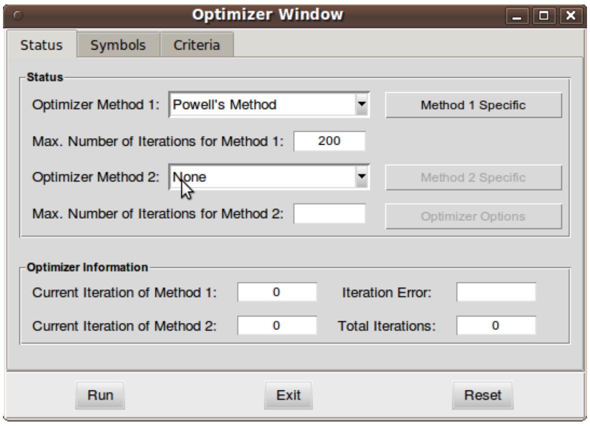

Fig. 5 Optimizer window

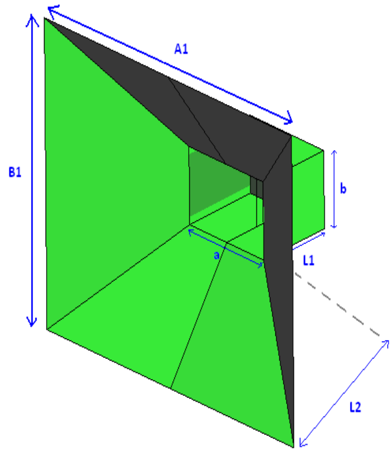

Fig. 6 Horn antenna (initial iteration)

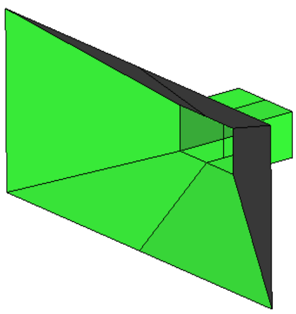

Fig. 7 Horn antenna (after 50 iterations)

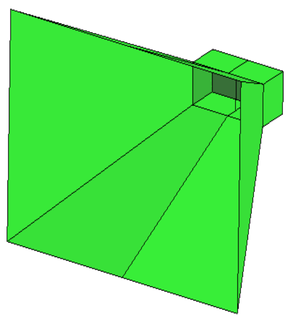

Fig. 8 Horn antenna (final iteration)

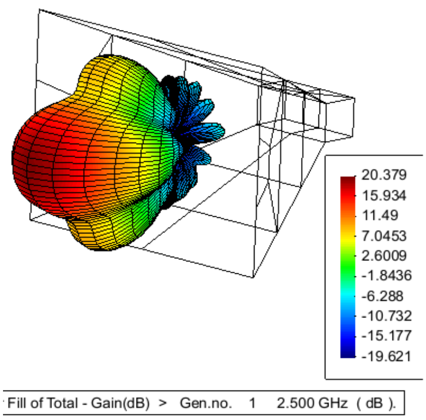

Fig. 9 Goal achieved (20 dB of gain)

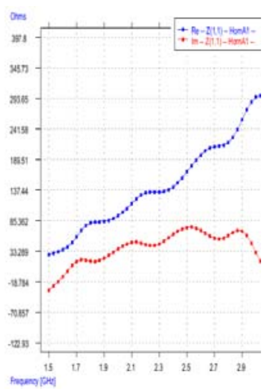

Fig. 10 Input impedance of horn antenna

To illustrate an automatic goal oriented optimization problem an example is shown in Fig. 5 to 9. The example is designed to optimize the dimensions of a horn antenna in order to obtain 20 dB of gain along the broadside direction. The first step in the optimization process is to create the structure of the horn antenna using symbols (see Fig. 6). The second step is to set the optimization method (Powell's method [4-6]), the number of iterations of the process (200 iterations), the maximum and minimum value of the symbols and the desired goal through the optimizer window (see Fig. 5). The last step is to execute the optimizer using the various buttons available in the optimizer window. Fig. 7 and 8 show the horn antenna after 50 iterations and 100 iterations respectively. Looking at these figures it is observed how the optimizer has changed the dimensions of the horn antenna until it reaches its desired objectives. To demonstrate that the planned goal has been reached, the radiation pattern is shown in Fig. 9.

Finally, an example of a graph in 2D is shown in Fig. 10. This graph shows the input impedance of the horn antenna shown in Fig. 8. Note that the input impedance of the antenna has not been used like criteria in the optimization process.

3 CONCLUSIONS

This paper presents interface between GiD® and a parallel In-Core and Out-of-Core integral-equation solver. This interface supports the use of symbolic variables to define the models that allows an automatic goal oriented optimization. In addition, it presents other important features such as multilanguage and multiplatform environment, friendly use, compatibility with CAD softwares package, and the possibility to execute simulations on a high performance parallel computing cluster. All these features make HOBBIES a very powerful electromagnetic simulator that is available from OHRN ENTERPRISES, INC.

REFERENCES

[1] Y. Zhang, T. Sarkar, D. García-Doñoro, H. Moon, M. Taylor, R. Van de Geijn, "*Parallel solution of integral equation-based EM problems in the frequency domain*". Wiley-IEEE Press, 2009.

[2] Y. Zhang, M. Taylor, T. Sarkar, H. Moon, and M. Yuan, "*Solving large complex problems using a higher-order basis: Parallel in-core and out-of-core integral-equation solvers*", IEEE Antennas and Propagation Magazine, vol. 50, pp. 13-30, Aug. 2008.

[3] B. Welch, K. Jones, J. Hobbs, Practical Programming in TCL and TK, fourth ed., Prentice-Hall, Englewood Cliffs, NJ, June 2003.

[4] M. J. D. Powell, "*The NEWUOA software for unconstrained optimization without derivatives*" vol. 83 of Nonconvex Optimization and Its Applications. Springer US, 2006.

[5] M. J. D. Powell, "*Developments of NEWUOA for unconstrained minimization without derivatives,*" Technical Report DAMTP 2007/NA05, pp. 255–297, 2007.

[6] W. Zhao, A. De, D. García-Doñoro, Y. Zhang and T. Sarkar, "*Antenna optimization by using NEWUOA*", IEEE Antennas and Propagation Society International Symposium, 10.1109/APS.2009.5171520, June 2009.