Transactions on Engineering Sciences vol 22 © 1999 WIT Press, www.witpress.com, ISSN 1743-3533

An overview of the ELFIN code for finite element research in electrical engineering

G. Aiello, [(1)] S. Alfonzetti, [(1)] G. Borzì, [(2)] and N. Salerno [(1)]
[(1)] *Dipartimento Elettrico, Elettronico e Sistemistico, Università di Catania, Viale A. Doria 6, I-95125 Catania, Italy*
[(2)] *Dipartimento di Fisica della Materia e Tecnologie Fisiche Avanzate, Università di Messina, Salita Sperone 31, I-98166 Messina, Italy*

Abstract

This paper gives a general overview of the basic features and latest enhancements of the finite element code ELFIN, developed by the authors for research in the electrical engineering field. The basic features are N-dimensional (1D, 2D and 3D) geometrical discretization, wide application area, easy treatment of coupled problems, and generalised iterative procedures. The latest enhancements of the code refer to the computation of non-linear transient eddy currents in open boundary domains, to the solution of electromagnetic scattering problems from inhomogeneous bodies (in 2D and 3D), to the automatic generation of meshes of tetrahedra based on an artificial neural network and to the stochastic optimization of electromagnetic devices.

1 Introduction

Starting in 1984, the authors have developed a finite element code in order to perform research in the field of computational electromagnetics for electrical engineering [1]. The code is called ELFIN (from the Italian words ELementi FINiti) [2,3]. The reasons for such an effort are mainly due to the difficulties which arise when using commercial codes in a research environment. In fact, such codes seldom exhibit the following characteristics: i) openness of the source code (in order to operate modifications and add new capabilities easily); ii) modularity (new problems should be implementable mainly by performing simple operations of recombining existing functions); N-dimensionality (1D, 2D and 3D problems should be dealt with by means of the same algorithms); iii) mesh

freedom (higher-order and curved-side elements should be individually defined); iv) FEM/BEM integration. Many commercial finite element codes, based on well-known and efficient procedures, are developed to solve a specific class of problems and therefore they generally have limitations such as a fixed dimensionality, a fixed kind and order of elements, and a fixed number and kind of unknowns. These limitations are aimed at optimising performance, but they practically prevent the use of such codes in a research environment.
This paper briefly describes the basic characteristics of ELFIN and outlines the research conducted by means of it.

2 The ELFIN code

The ELFIN code is written in Fortran-77 and is now running on several computers under the OpenVMS, Unix, Linux and DOS/Windows operating systems. The code is structured according to the classical scheme of FE codes, in which three distinct main programs are devoted respectively to pre-processing, processing and post-processing. These programs recall a set of more than 600 routines, which can be grouped into three main categories: a) specific subroutines, which refer to the code data structure, including: I/O subroutines; geometrical discretization and material catalogue subroutines; geometrical window and list subroutines; shape function subroutines; universal matrix subroutines; b) non-specific subroutines, which do not refer to the code data structure, including: matricial and polynomial ordering; combinatorial calculus; polynomial algebra and calculus; polynomial matrix algebra; N-dimensional integration of polynomial rational functions; geometrical computations; the solution of linear algebraic equation systems; c) graphic subroutines, which interface the external graphic library used (Regis, GKS, Xlib, Display Postscript).

2.1 Pre-processing

In ELFIN four geometrical entities of the space discretization are defined: nodes, elements, boundary sides and, optionally, edges. Nodes are the primary entities of the discretization. A node is identified by its absolute Cartesian coordinates X_k ($k=1,...,N$), in an N-dimensional space. The other entities of the discretization (elements, boundary sides and edges) are derived from nodes in terms of ordered sets of them. An element is an ordered set of nodes, the ordering having a geometrical meaning. The following data are individually specified for each finite element: the element family (e. g. isoparametric Lagrangian simplex and N-rectangular); the element dimensionality N' (with $N' \leq N$); the element order m; the element curved-side indicator (for orders $m > 1$). The local sides of an element are implicitly defined as ordered subsets of its nodes. Boundary sides are globally defined local sides, which generally lie on the domain boundary, even if internal boundary sides are allowed. Optionally edge elements can be used, based on the definition of edge as an ordered pair of nodes. Region "colours" can be

Transactions on Engineering Sciences vol 22 © 1999 WIT Press, www.witpress.com, ISSN 1743-3533

independently attributed to nodes, elements, boundary sides and edges in order to treat particular subsets differently during the pre-processing and/or processing phases. These definitions of the geometrical entities imply hierarchy: nodes are more important than elements and edges, elements more important than boundary sides. This structure is clearly independent of the space dimensionality N (which only affects the cardinality of the co-ordinates of nodes) but has a cost in terms of larger memory occupation and computing time with respect to codes employing only elements of a fixed kind and order.
The ELFIN code, although developed in an electrical engineering department, has been designed as general-purpose environment to be used in any other engineering area (mechanical, civil, etc.). For this reason great freedom has been reserved in the definition of the field variables and sources. A set of M_S scalar variables S_i ($i=1,...,M_S$) and a set of M_V vector variables V_j ($j=1,...,M_V$) are associated to each node and to each edge, respectively, of the discretization. These variables can be all real or all complex, according to the real/complex indicator I_{RC}, which is set to 1 for real problems and to 2 for complex ones. For the field variables, boundary conditions (Dirichlet, Neumann or mixed) are imposed on the nodes or edges of the boundary sides. Four kinds of sources are defined: node sources, element sources, boundary-side sources and edge sources. The real/complex indicator also applies to sources. Note that parameterization with respect to the number of field variables and sources is the basis to allow unified treatment of coupled problems. Moreover, no semantics need to be preliminarily attributed to each variable in the pre-processing: a great variety of problems (scalar, vector, coupled, real or complex) with different field variables and sources can be implemented with no modifications of data structures.
Due to the fact that the code has been implemented by a small research group, the pre-processor has medium capabilities for solid modeling, but quite sophisticated features for defining boundary conditions and sources and for managing high-order and/or curved-side elements; however, functions are expressly foreseen to interface more powerful external solid modeling programs that may be available. The pre-processor exhibits a mixed interactive-batch user interface: before a pre-processing session the user can edit a text file, containing the sequence of pre-processing commands (a command is constituted by a 7-character identifier followed by some mumerical parameters), which can be recalled (once or more) at any time in the interactive session. The pre-processing commands refer to the following items: a) the relative frame (Cartesian, polar or cylindrical), optionally defined for more simple reference to the absolute spatial co-ordinates; b) the geometrical window, defined in the relative frame to restrict the application of some pre-processing commands to this domain; c) the geometrical list, defined as a set of nodes, elements, boundary sides or edges to restrict the application of some commands to this set; d) the geometrical functions, which are real functions of the spatial co-ordinates in the relative frame, defined to impose boundary conditions and sources in an automatic way; e) the mesh generation, which is relative to the current geometrical window and allows the generation of nodes, elements and boundary sides in an N-dimensional way. These features allow a

very efficient imposition of sources and boundary conditions. This operation is divided in two phases: 1) selection of the boundary sides on which the boundary conditions are to be imposed; geometrical (volume, surface, line), indexed and mixed selections are performed by using the geometrical window and list concepts; 2) assignment of the pertaining nodal numerical values, either by using the internal geometrical functions or by reading them from external files.

2.2 Processing

The ELFIN processing program works in a batch way, starting from the data prepared in the pre-processing session and stored in a suitable file. In the following some aspects of the program will be highlighted.

- Generalised iteration structures: a general iteration structure was implemented in order to treat the various iterative procedures independently; four kinds of iteration are implemented: non-linearity, adaptive mesh refinement, boundary conditions (see DBCI and RBCI in Sect. 3), and time discretization [4];
- Numerical techniques: classical variational and weighted residual (Galerkin) approaches are used. For each equation type a routine exists which builds the global algebraic system, by resorting to a set of lower-level routines dealing with the most common differential, integral and algebraic operators. This structure allows a simple treatment of equations by splitting them into elementary differential, integral and algebraic terms. Great ease in composing new problems is given: the addition of new problem-solving capabilities only involves the construction of a new limited-size routine, addressing the existing basic ones.
- Universal matrices: in addition to numerical (Gauss) integration, great care has been devoted to the universal matrix technique for simplex Lagrangian elements with straight sides. Since universal matrices, initially introduced by Silvester, are continuously growing in number and kind, a set of basic routines was developed to allow on-line computation of universal values in integer form. At present about twenty kinds of universal matrices are available [5,6].
- Coupled problems: the treatment of coupled problems is based on parameterization with respect to M_S and M_V and is dealt with in a generalised way: the ordered set of field variables (whose semantics is unknown to the preprocessor) is correctly interpreted only by the routine which builds the global algebraic system for the specific coupled problem. In this way great flexibility exists to implement new coupled problems into the code, because only one limited-size routine has to be developed and added to the processor.

2.3 Post-processing

The post-processing program exhibits the same interactive-batch user interface as the pre-processing one, with the provision of interfacing to more powerful commercial post-processing programs. The user can define up to three (real or complex, scalar or vector) variables V_R, related to the computed field variables by means of algebraic or differential operations. After one of them has been selected (that is, the token has been assigned to it) all the subsequent restitutions will refer to the variable currently selected. With respect to discretization four types of V_R are possible: node, element, boundary-side and edge variables. Restitutions are available such as: displaying/printing of nodal values, 2D contour line plotting, axonometric 3D contour line plotting [7], 2D and 3D vector plotting. Global quantities (energy, flux, current, etc.) can also be evaluated by means of integrals on elements, boundary (or local) sides or edges.

3 Research conducted by means of ELFIN

In this section an overview is given of the research conducted by means of the ELFIN code. For the sake of conciseness, only a brief outline is given; more details can be found in the papers referenced.

3.1 Static and quasi-static problems in open boundaries

These problems have been dealt with by means of a hybrid method, now called Dirichlet Boundary Condition Iteration (DBCI) and formerly referred to as charge iteration or current iteration. Referring to a simple electrostatic problem, in which a system of voltaged conductors is embedded in an unbounded homogeneous dielectric medium, DBCI is applied by introducing a fictitious boundary B_F which includes all the conductors and truncates the unbounded domain to a bounded one D [8-18]. The method is based on the following equations:

$$\nabla \cdot (\varepsilon \nabla v) = 0 \qquad \text{in } D \tag{1}$$

$$v(\mathbf{r}) = -\int_{B_C} G(\mathbf{r}, \mathbf{r}') \frac{\partial v(\mathbf{r}')}{\partial n'} d\sigma' \qquad \mathbf{r} \in B_F \tag{2}$$

where B_C is the conductor surface and G is the free-space Green's function. By discretising D by means of nodal finite elements, the following linear system of equations is obtained:

$$\mathbf{AV} = \mathbf{B} - \mathbf{A}_F \mathbf{V}_F \tag{3}$$

$$\mathbf{V}_F = \mathbf{V}_{F0} + \mathbf{HV} \tag{4}$$

This system can be efficiently solved iteratively: initially guessing the Dirichlet condition $\mathbf{V}_F$ on B_F, equation (3) is solved for $\mathbf{V}$, which is used in (4) to improve $\mathbf{V}_F$. The procedure is iterated until convergence takes place. A better convergence can be obtained by means of a relaxation coefficient in (4) or by means of GMRES. Similar procedures have been successfully applied to the solution of the integro-differential equation [19] of the two-dimensional time-harmonic skin effect [20-24] and of transient non-linear eddy current problems [25-26].

Further research in this context was reported on in [27] in which it was shown that the Pefectly Matched Layer (PML), recently proposed for static fields, is equivalent to the utilization of a trivial truncation followed by simple co-ordinate transformations. For this reason no advantage with respect to truncation derives from the use of such a method.

3.2 Electromagnetic scattering problems

Let us consider a system of conductors and/or dielectrics infinitely extended in the z-direction, embedded in a homogeneous dielectric medium, lit up by an incident monochromatic wave ϕ_0, E- or H-polarised along the z-axis. An unbounded scattering problem is set up in terms of the total field ϕ. To deal with this problem, the hybrid method illustrated in the previous Section has been modified into the RBCI (Robin Boundary Condition Iteration) method [28-33], in which the Dirichlet condition on the fictitious boundary B_F has been replaced by a Robin (mixed) one of the type:

$$\Re\phi = \frac{\partial\phi}{\partial n} + jk_0\phi = \psi \quad \text{on} \quad B_F \tag{5}$$

where k_0 is the free-space wavenumber and ψ is an unknown function on Γ_F. Then the working equations are:

$$\nabla\cdot\alpha^{-1}\nabla\phi + k_0^2\eta\phi = 0 \qquad \text{in } D \tag{6}$$

$$\psi(\mathbf{r}) = \Re\phi_0(\mathbf{r}) - \int_{B_S}\left[\Re G(\mathbf{r},\mathbf{r}')\frac{\partial\phi(\mathbf{r}')}{\partial n'} - \phi(\mathbf{r}')\frac{\partial\Re G(\mathbf{r},\mathbf{r}')}{\partial n'}\right]ds \qquad \mathbf{r}\in B_F \tag{7}$$

where α and η are relative constitutive parameters, B_S is the scatterer surface and G is the two-dimensional free-space Green's function

$$G(\mathbf{r},\mathbf{r}') = -\tfrac{1}{4}jH_0^{(2)}(k_0|\mathbf{r}-\mathbf{r}'|) \tag{8}$$

Equations (6) and (7), taking into account the boundary condition (5), are discretised as:

$$\mathbf{A\Phi} = \mathbf{C\Psi} \tag{9}$$

$$\mathbf{\Psi} = \mathbf{\Psi}_0 + \mathbf{M\Phi} \tag{10}$$

and solved iteratively as before. Initially guessing Ψ (a good guess is Ψ_0), equation (9) is solved for Φ, which is used in (10) to improve Ψ. This procedure is iterated until convergence is obtained. Note that if one tries to use Dirichlet conditions on the fictitious boundary B_F, as in standard FE-BI methods, possible singularities may arise in (6), due to the resonance of the domain D and also in (7), due to the existence of non-vanishing incident fields ϕ_0 which may vanish on the boundary B_F. On the contrary, by using the Robin boundary condition (5) resonances are completely avoided whatever the frequency of the incident wave. This very good property is due to the fact that the boundary condition (5) works as an absorbing-like one. Another important property is that the air gap in between the scatterer surface B_S and the fictitious boundary B_F can be very thin (two or three layers of elements are enough) still obtaining accurate results with an acceptable computing time (normally from 4 to 8 iterations are sufficient to obtain convergence). By means of a simple indicator, the accuracy of the solution can be tested in post-processing when the bistatic radar cross section is computed.

The RBCI method has been successfully adapted to the computation of scattering from cavity-backed apertures on a perfectly-conducting plane or wedge, and also to the solution of three-dimensional scattering problems, employing tetrahedral edge elements.

3.3 Mesh generation by neural networks

Starting from a rough initial mesh of triangles or tetrahedra with a small number of nodes the neural network algorithm increases the number of nodes until a user-selected value is reached [34-37]. Both internal and boundary nodes are separately increased according to two distinct probability density functions, specified by the user using the initial mesh as support. New nodes are inserted in the middle of triangle or tetrahedra edges. After an example point is generated, the nearest nodes are moved to it, taking into account the mesh quality; moreover the classical Delaunay triangulation algorithm is inserted in the growing process to further increase the quality of the final mesh. The main advantages of this neural approach are the lack of a need for classical deterministic programming, the simplicity of use (the user only needs to specify an initial rough mesh and the probability density functions) and the very good qualities of the simplex elements obtained.

3.4 Optimization by means of stochastic methods

Recently the ELFIN code has been adapted to deal with stochastic optimization of electromagnetic devices. This goal has been reached by allowing the use of symbolic variables in the pre- and post-processing command parameters instead of fixed numeric values. In this way both the pre- and post-processing batch sessions (and then the whole code) can be made parametric with respect to a set of variables, typically representing geometrical or constitutive data of the device to be optimised. Two stochastic optimisation methods have been implemented: genetic algorithms and simulated annealing. They have been applied to the optimization of some electromagnetic devices, ranging from magnetostatic to wave-propagation ones [38-40].

4 Conclusions

In this paper the finite element code ELFIN, developed for research purposes in the field of computational electromagnetics in electrical engineering, has been presented. A general overview of the code has been given, briefly describing some non-standard implementation aspects as well as the main research conducted by means of it.

The main merit of ELFIN is the facility it offers in building, testing and verifying new numerical procedures devised for the analysis and design of electromagnetic devices.

At present the code is composed of about 600 subroutines (for 60,000 statements). It will be expanded according to research subjects rather than widening the range of standard problem-solving capabilities.

Further information on ELFIN can be found on the Internet at the following Web address: http://wwwelfin.dees.unict.it/elfin/.

Acknowledgements

This work was partially supported by the MURST (the Italian Ministry for University and Scientific and Technological Research).

References

1. Silvester, P. P., Ferrari, R. L., *Finite Elements for Electrical Engineers*, Cambridge University Press, Cambridge (U.K.), 1990.
2. Alfonzetti, S., Coco, S., ELFIN: an N-dimensional finite-element code for the computation of electromagnetic fields, *IEEE Trans. Magn.*, vol. 24, pp. 362-365, 1988,
3. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., An FEM code for electrical engineering research: ELFIN, *Electrosoft*, S. Miniato (Italy), 28-30 may 1996.
4. Alfonzetti, S., Salerno, N., A generalised scheme to deal with time, boundary-condition and non-linearity iterations in an FEM code, *ISTET*, Palermo (Italy), 9-11 june 1997.
5. Alfonzetti, S., Construction of universal matrices with conventional program-ming languages, *Electrosoft*, S. Miniato (Italy), 28-30 may 1996.
6. Alfonzetti, S., Helmholtz equation FEM matrices for 2nd-order triangles with one curved side, ISTET, Thessaloniki (Greece), 22-23 september 1995.
7. Alfonzetti, S., An n-dimensional algorithm to draw contour lines over triangular elements, *IEEE Trans. Magn.*, vol. 32, pp. 1473-1476, 1996.
8. Aiello, G., Alfonzetti, S., Coco, S., Charge iteration for n-dimensional unbounded elecrical field computations, *IEEE Trans. Magn.*, vol. 28, pp. 1682-1685, 1992.
9. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Axisymmetric unbounded elecrical field computation by charge iteration, *IEEE Trans. Magn.*, vol. 29, pp. 2043-2046, 1993.
10. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Convergence analysis of the charge iteration procedure for unbounded electrical fields, *IEEE Trans. Magn.*, vol. 30, pp. 2873-2876, 1994.
11. Aiello, G., Alfonzetti, S., Coco, S., Capacitance computation in symmetric multiconductor systems, *IEEE Trans. Magn.*, vol. 30, pp. 2952-2955, 1994.
12. Aiello, G., Alfonzetti, S., Coco, S., Charge iteration: a procedure for the finite-element computation of unbounded electrical fields, *Int. J. Numer. Meth. Engng*, vol. 37, pp. 4147-4166, 1994.
13. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Placement of the fictitious boundary in the charge iteration procedure for unbounded electrical fields, *IEEE Trans. Magn.*, vol. 31, pp. 1392-1395, 1995.
14. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Overrelaxing the charge iteration procedure, *IEEE Trans. Magn.*, vol. 32, pp. 694-697, 1996.
15. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., A theoretical study of charge iteration, *COMPEL*, vol. 15, pp. 22-46, 1966.

Transactions on Engineering Sciences vol 22 © 1999 WIT Press, www.witpress.com, ISSN 1743-3533

16. Aiello,G., Alfonzetti,S., Borzì,G., Coco,S., Salerno,N., A generalization of the charge iteration procedure, *IEEE Trans. Magn.,* vol. 33, pp. 1204-1207, 1997.
17. Alfonzetti, S., Borzì, G., Salerno, N. A generalised minimal residual acceler-ation of the charge iteration procedure, *J. de Phys. III,* vol.7,pp.1955-66, 1997.
18. Aiello, G., Alfonzetti, S., Borzì, G., Salerno, N., Computing spatially-periodic electrical fields by charge iteration, *IEEE Trans. Magn.,* vol. 34, 1998.
19. Konrad, A. Integrodifferential finite element formulation of two-dimensional steady-state skin effect problems, *IEEE Trans. Magn.*, vol 18, pp. 284-292, 1982.
20. Aiello,G., Alfonzetti,S., Borzì,G., Coco,S., Salerno,N., Current iteration for unbounded skin effect problems, *COMPEL,* vol.13,suppl.A,pp.299-304, 1994.
21. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Treatment of unbounded skin-effect problems in the presence of material inhomogeneities, *IEEE Trans. Magn.*, vol. 31, pp. 1504-1507, 1995.
22. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Finite-element iterative solution of skin effect problems in open boundaries, *Int. J. Numer. Modelling: Electron. Netw., Dev. and Fields,* special issue on 'Computational Magnetics', vol. 9, pp. 125-143, 1996.
23. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Combining non-linearity and current iterations for the solution of boundless skin effect problems, *IEEE Trans. Magn.,* vcl. 33, pp. 1291-1294, 1997.
24. Aiello, G., Alfonzetti, S., Borzì, G., Salerno, N., Applying GMRES to the solution of FEM/DBCI systems for unbounded skin effect problems, *CEFC,* Tucson (USA), 1-3 June 1998.
25. Aiello, G., Alfonzetti, S., Coco, S., Salerno, N., Finite element analysis of unbounded non-linerar transient magnetic fields, *IEEE Trans. Magn.,* vol. 33, pp. 1318-21, 1997.
26. Aiello, G., Alfonzetti, S., Borzì, G., Salerno, N., A predictor-corrector scheme for open boundary problems, *IEEE Trans. Magn.,* vol. 34, 1998.
27. G., Alfonzetti, S., Borzì, G., Salerno, N., Some considerations about the Perfectly Matched Layer for static fields, *IGTE Symp.,* Graz (Austria), 21-23 September1998
28. Alfonzetti, S., Borzì, G., Salerno, N., FEM analysis of unbounded electromagnetic scattering by the Robin iteration procedure, *Electronic Letters*, vol. 32, pp. 1768-1769, 1996.
29. Alfonzetti, S., Borzì, G., Salerno, N., Accelerating the Robin iteration procedure by means of GMRES, *COMPEL*, vol. 17, pp. 49-54, 1998.
30. Alfonzetti, S., Borzì, G., Salerno, N., Iteratively-improved Robin boundary conditions for the finite element solution of scattering problems in unbounded domains, *Int. J. Numer. Meth. Engng*, vol. 42, pp. 601-629, 1998.

31. Alfonzetti, S., Borzì, G., Salerno, N., An iterative solution to scattering from cavity-backed apertures in a perfectly-conducting wedge, *IEEE Trans. Magn.,* vol. 34, 1998.
32. Alfonzetti, S., Borzì, G., Three-dimensional electromagnetic scattering computation by means of the Robin Boundary Condition Iteration method, *CEFC,* Tucson (USA), 1-3 June 1998.
33. Alfonzetti, S., Borzì, G., Accuracy of the Robin boundary condition iteration method for the finite element solution of scattering problems, *Finite Elements for Microwave Engng.,* Poitiers (France), 10-11 July 1998.
34. Alfonzetti,S., Cavalieri,S.,Coco,S.,Malgeri,M., Automatic mesh generation by the Let-It-Grow neural network, *IEEE Trans.Magn.,*vol.32,pp.1349-52, 1996.
35. Alfonzetti, S., High-quality finite element mesh generation by means of an artificial neural network, *IGTE Symp.*, Graz (Austria), 23-26 September 1996
36. Alfonzetti, S., A finite element mesh generator based on an adaptive neural network, *IEEE Trans. Magn.,* vol. 34, 1998.
37. Alfonzetti, S., A neural network generator for tetrahedral meshes, *CEFC,* Tucson (USA), 1-3 June 1998.
38. Aiello, G., Alfonzetti, S., Borzì, G., Coco, S., Salerno, N., Shape optimization of the magnetic channel of a superconducting cyclotron, *COMPEL*, vol. 17, pp. 123-127, 1998.
39. Aiello, G., Alfonzetti, S., Borzì, G., Salerno, N., Shape optimization of open-boundary axisymmetric induction devices, *CEFC,* Tucson (USA), 1-3 June 1998.
40. Alfonzetti, S., Borzì, G., Salerno, N., Optimization of the coating layer of a scattering target, *IGTE Symp.,* Graz (Austria), 21-23 September 1998