

THE MEMEC SOFTWARE APPLIED TO FRACTURE MECHANICS PROBLEMS

Carlos Andrés Reyna Vera-Tudela, candres@ufrrj.br
Bruno de Souza Silva, brunosirj@yahoo.com.br
Universidade Federal Rural do Rio de Janeiro, Departamento de Matemática, Caixa Postal 74517, Seropédica, 23890-971, RJ, Brasil

Marlucio Barbosa, marluciobarbosa@gmail.com
Universidade Federal Rural do Rio de Janeiro, COINFO

Edivaldo Figueiredo Fontes Junior, edivaldofontes@gmail.com
José Claudio de Faria Telles, telles@coc.ufrj.br
Ana Ibis Abreu Rojas, anai@coc.ufrj.br
COPPE/UFRJ, Programa de Engenharia Civil

Abstract. *The software "Mecânica Elastostática – Método dos Elementos de Contorno", (MEMEC) has been developed based on a previous Fortran 77 program. The software solves Solid Mechanics problems numerically using the Boundary Element Method. In its current version the MEMEC code employs Java language. Though initially intended to solve only standard elastostatic problems, in this second stage of development the authors introduced the implementation for fracture mechanics problems using a so called numerical Green´s Function (NGF) procedure. To illustrate the new strategy implemented, examples of scientific visualization of fracture problems in a plate are discussed.*

Keywords*: Boundary Element Method, Scientific Visualization, Fracture Mechanics Problems.*

1. INTRODUCTION

Accurate crack analysis is important for health monitoring and quality control in many critical applications pertaining to civil, aeronautical, nuclear and mechanical engineering areas. The appropriate numerical modeling of linear elastic fracture problems, however, strongly depends on the appropriate simulation of the singular stress field in the vicinity of the crack tip. With regards to the boundary element method (BEM), another difficulty arises: the crack surfaces geometrically coincide in the numerical model, causing a degeneration of the boundary integral equation. This problem is usually avoided by the three alternative approaches: applying the sub-regions technique (Blandford, *et al.*, 1981), employing the mixed or dual formulation (Cruse, 1998; Guimarães and Telles, 1994; Portela *et al.*, 1992), or using the associated Green´s function (Cruse, 1998; Telles, *et al.*, 1995).

The evolution of computational tools has been fundamental for the development of competitive software to tackle more complex problems. The traditional algorithms must be revised and, sometimes, modified for use in high performance applications. For example, parallel processing has been growing and practically, there is no limitation in the use of CPU memory. Scientific visualization is a necessity in science and engineering applications because the observation of the physical behavior is fundamental for efficient problem analysis.

The authors are working in the development of a software to solve, initially, 2D elastostatic problems with the Boundary Element Method (BEM) (Brebbia, *et al.*, 1984). The software MEMEC was developed in its original version in Fortran, but for efficient visualization it is necessary to use an Object Oriented Language. Thus, the choice was Java with the Visualization Toolkit (VTK) (Schroeder, *et al.*, 2004). The traditional algorithms were modified to the new requirements of computational efficiency and visualization. New algorithms have recently been be implemented, for example, a mesh generator algorithm, interaction between BEM-FEM and fracture mechanics simulation problems.

First the initial version was made to be an efficient and friendly software, then the relative error has been made smaller and the visualization of the problem has also been emphasized: the initial physical problem, with load and boundary conditions are fist depicted; an independent graph of stress and displacements is then be obtained and the final deformed shape too. In addition, it is also possible to have a 3D visualization and move the body in all directions.

In this work are presented two examples of central straight cracks, taking advantage of the symmetry of the body. The main objective of the results is the scientific visualization of the solution. In an ongoing version the implementation of the numerical Green´s function is being carried out.

2. THE BOUNDARY ELEMENT METHOD

The conditions of equilibrium for elastostatic problem in the absence of body forces can be expressed by the well-known Navier equation which may be written in the form;

$$G u_{j,kk} + \frac{G}{(1-2\nu)} u_{k,kj} = 0 \tag{1}$$

where *u* is the displacement field, *G* is the Shear Modulus and ν is the Poisson´s Ratio. The relation between the displacement tensor and the traction tensor is given by

$$p_i = \sigma_{ij} n_j \tag{2}$$

where n_j represents the direction cosines of the outward normal to the surface.

The starting boundary integral equation can be obtained in a simple way based on considerations of the weighted residual technique and permits the extension of the method to the solution of more complex partial differential equations.

$$G \int_\Omega u_{j,kk} u_{ij}^* \, d\Omega + \frac{G}{(1-2\nu)} \int_\Omega u_{k,kj} u_{ij}^* \, d\Omega = 0 \tag{3}$$

In principle, the errors introduced in the Eq. (3) if the exact (but unknown) values of *u* and *p* are replaced by an approximating solution, can be minimized by orthogonalizing them with respect to a weighting function u_{ij}^*.

Integrating by parts and taking into consideration the reciprocity principle, the boundary integral equation is obtained as follows:

$$C_{ij}(\xi) u_j(\xi) + \int_\Gamma p_{ij}^* u_j \, d\Gamma = \int_\Gamma u_{ij}^* p_j \, d\Gamma \tag{4}$$

where C_{ij} is a coefficient related to the geometry of the domain (Brebbia, *et al.* 1984).

The fundamental solution for the plane strain problems in an infinite elastic medium, known as Kelvin´s solution, is of the form:

$$u_{ij}^*(\xi, x) = \frac{-1}{8\pi(1-\nu)G} \left\{ (3-4\nu) \ln(r) \delta_{ij} - r_{,i} r_{,j} \right\} \tag{5}$$

with the corresponding traction field

$$p_{ij}^*(\xi, x) = \frac{-1}{4\pi(1-\nu) r} \left\{ \left[(1-2\nu)\delta_{ij} + 2 r_{,i} r_{,j} \right] \frac{\partial r}{\partial n} - (1-2\nu)(r_{,i} n_j - r_{,j} n_i) \right\} \tag{6}$$

where *r* is the distance from the point of application of the load to the point under consideration.

2. NUMERICAL IMPLEMENTATION

We need to solve the Eq. (3):

$$G \int_\Omega u_{j,kk} u_{ij}^* \, d\Omega + \frac{G}{(1-2\nu)} \int_\Omega u_{k,kj} u_{ij}^* \, d\Omega = 0 \tag{3}$$

with the boundary conditions (Fig. 1)

$$\begin{aligned} u_j &= \bar{u}_j \qquad in \;\; \Gamma_1 \\ p_j &= \bar{p}_j \qquad in \;\; \Gamma_2 \end{aligned} \tag{7}$$

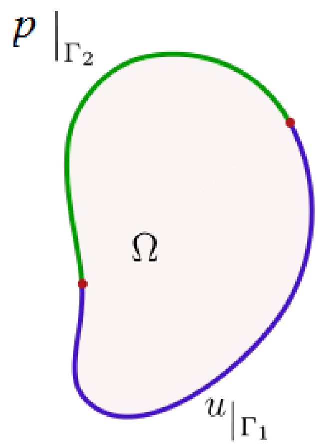

Figure 1: Boundary Conditions

The Cartesian coordinates $\underset{\sim}{x}^{(j)}$ of points located within each element Γ_j are expressed in terms of interpolation functions $\underset{\sim}{M}$ and the nodal coordinates $\underset{\sim}{x}^{(m)}$ of the element by the matrix relation

$$\underset{\sim}{x}^{(j)} = \underset{\sim}{M}\,\underset{\sim}{x}^{(m)} \tag{8}$$

In a similar way, boundary displacements and tractions are approximated over each element through interpolation functions $\underset{\sim}{N}$:

$$\begin{aligned} \underset{\sim}{u}^{(j)} &= \underset{\sim}{N}\,\underset{\sim}{u}^{(n)} \\ \underset{\sim}{p}^{(j)} &= \underset{\sim}{N}\,\underset{\sim}{p}^{(n)} \end{aligned} \tag{9}$$

where $\underset{\sim}{u}^{(n)}$ and $\underset{\sim}{p}^{(n)}$ contain the nodal displacements and tractions, respectively.

We will now assume that the boundary Γ is divided into L elements Γ_j, the Eq. (3) can be write as:

$$\underset{\sim}{C}_i(\xi_i)\underset{\sim}{u}(\xi_i) + \sum_{j=1}^{L}\left[\int_{\Gamma_j} \underset{\sim}{p}^{*}\,\underset{\sim}{N}\,d\Gamma\right]\underset{\sim}{u}^{(n)} = \sum_{j=1}^{L}\left[\int_{\Gamma_j} \underset{\sim}{u}^{*}\,\underset{\sim}{N}\,d\Gamma\right]\underset{\sim}{\overline{p}}^{(n)} \tag{10}$$

The integrals in Eq. (10) are usually solved numerically and the interpolation functions $\underset{\sim}{M}$ and $\underset{\sim}{N}$ are expressed in terms of the dimensionless coordinate η ; we must write $d\Gamma$ in terms of the intrinsic system:

$$d\Gamma = \left|\underline{J}\right| d\eta \tag{11}$$

where the Jacobian $\left|\underline{J}\right|$ of the transformation is defines as:

$$\left|\underset{-}{J}\right| = \sqrt{\left(\frac{dx_1}{d\eta}\right)^2 + \left(\frac{dx_2}{d\eta}\right)^2} \tag{12}$$

In a simple case, the integrals in Eq. (10) can be solved analytically. In general, numerical process (Gauss integration) are procedures more efficient and can be used in high order functions. In the special situation, when $\xi_i \in \Gamma_j$ exists a singularity in $\Omega = 0$.

In the normal situation, when $\xi_i \notin \Gamma_j$, the integrals can be obtained as follow

$$\int_{\Gamma_j} \underset{\sim}{p}^* \underset{\sim}{N}\, d\Gamma = \int_{-1}^{1} \underset{\sim}{p}^* \underset{\sim}{N} \left|\underset{-}{J}\right| d\eta \cong \sum_{k=1}^{K} \left|\underset{-}{J}\right|_k \omega_k \left(\underset{\sim}{p}^* \underset{\sim}{N}\right)_k \tag{13}$$

$$\int_{\Gamma_j} \underset{\sim}{u}^* \underset{\sim}{N}\, d\Gamma = \int_{-1}^{1} \underset{\sim}{u}^* \underset{\sim}{N} \left|\underset{-}{J}\right| d\eta \cong \sum_{k=1}^{K} \left|\underset{-}{J}\right|_k \omega_k \left(\underset{\sim}{u}^* \underset{\sim}{N}\right)_k \tag{14}$$

where K is the total number of integration points and ω_k is the associated weighting factor to the k point.

After the application of Eq.(10) to all the N nodal functional points, is formed a 2N system of equations as

$$\left(\underset{\sim}{C} + \underset{\sim}{\hat{H}}\right)\underset{\sim}{u} = \underset{\sim}{G}\,\underset{\sim}{p} \tag{15}$$

where the vectors $\underset{\sim}{u}$ and $\underset{\sim}{p}$ are the displacements and tractions in the total nodal points and the quasi-diagonal matrix $\underset{\sim}{C}$ can be added to the $\underset{\sim}{\hat{H}}$ matrix to form the $\underset{\sim}{H}$ matrix:

$$\underset{\sim}{H} = \underset{\sim}{C} + \underset{\sim}{\hat{H}} \tag{16}$$

Hence

$$\underset{\sim}{H}\,\underset{\sim}{u} = \underset{\sim}{G}\,\underset{\sim}{p} \tag{17}$$

We now have to apply the boundary conditions in this system of equations. Thus, can be reordered in such a way that all the unknowns are written on the left-hand side in an $\underset{\sim}{y}$ vector. The final result can be written

$$\underset{\sim}{A}\,\underset{\sim}{y} = \underset{\sim}{f} \tag{18}$$

where $\underset{\sim}{A}$ is a fully populated matrix of order 2N.

3. THE MEMEC SOFTWARE

The software MEMEC (Fontes Jr, *et al.*, 2007) was originally based on software developed in Fortran. The software developed in FORTRAN was used for a long time to obtain the numerical results presented in articles, events and theses. It can be stated that MEMEC is not a pure translation of the software for the Java (Deitel & Deitel, 2004). The MEMEC is an application based in a n-layers model. Figure 2 shows a simplified diagram of the software.

As shown in the diagram, the MEMEC was developed to be a software for the Boundary Element Method and its variations, regardless of the physical problem to be solved. The expansion of issues to be borne by MEMEC takes place by inserting a new processor features obeying the previous project. Thus, MEMEC is an environment that covers the basic requirements for an extension number, thus, having pre-processor, and postprocessor.

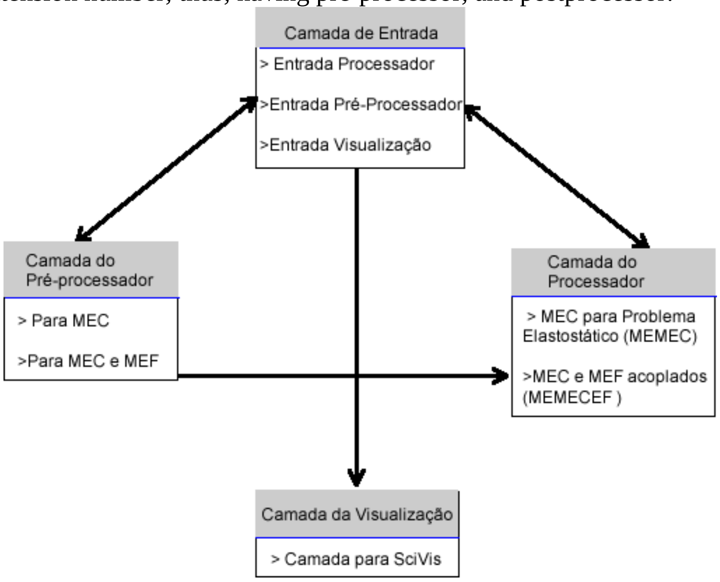

Figure 2. Diagram of the software MEMEC

They are currently implemented in the MEMEC software a processor to solve elastostatic problems and a processor BEM / FEM in order to solve problems of coupling between these two methods. In Figure 3 we can see a window in the MEMEC software with the various options available information.

Besides the classical approach to solving the BEM, two subjects were studied in depth and implemented to provide an efficient product and optimize the computational cost. The development of a graphical environment (Barbosa, 2010) is the differential for the end user and represents an important resource to analyze the results. To increase the efficiency of the software and make appropriate use of the resources of modern multicore processors have been implemented multithreading applications (Fontes Jr, 2010)

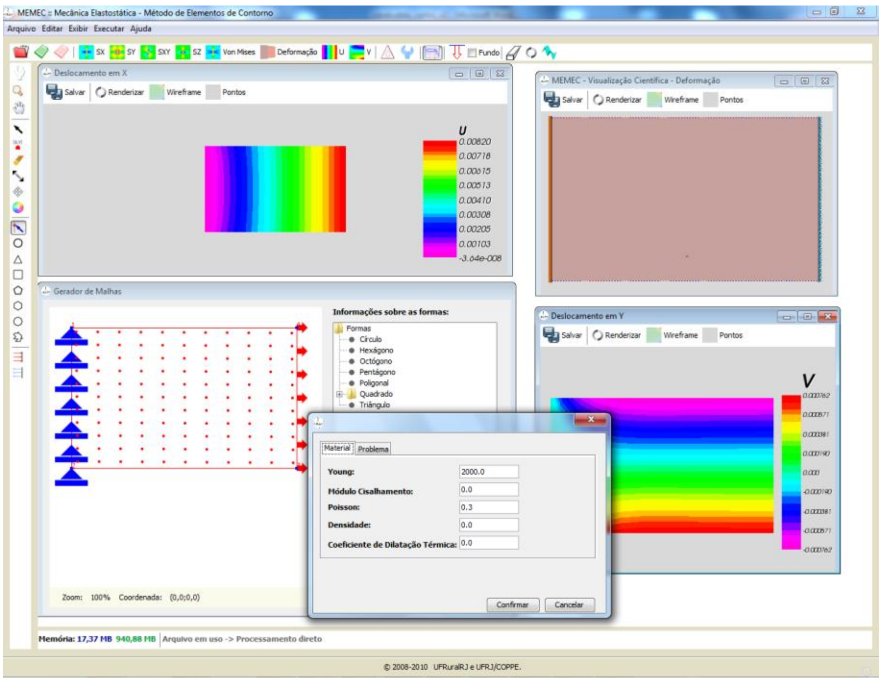

Figure 3. A example of the MEMEC presentation with five windows.

This work shown the numerical solution of fracture mechanics problems utilizing the symmetry of the body. In this way, we can to observe the visualization in the end crack that is a point of numerical singularity. The next stage in this research is the implementation of a computational model based on the numerical Green´s function. The numerical

Green´s function, corresponding to an embedded crack within the infinite medium, is introduced into a boundary element formulation, as the fundamental solution, to calculate the unknown external boundary displacements and tractions and in post-processing determine the crack opening displacements.

4. NUMERICAL EXAMPLES

Two numerical examples are presented to observe the scientific visualization of the problem.

Example 1: A rectangular plate is studied and the geometry and traction is shown in Fig. 4, the material properties are Young modulus *E=7.43*, Poisson ratio $\nu = 0.3$. The plate is loaded along opposite boundaries and the load is equal 1. The crack length is $2a = 5$. The gauss integration utilized 20 points.

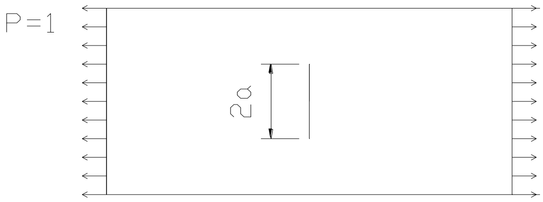

Figure 4. Central crack in a plate.

Figure 5 shown the geometry of the one quarter of plate region, the boundary conditions and the load in the extremes of the plate.

Many problems in science and mathematics exhibit symmetry phenomena which may be exploited to effect a significant cost reduction in their numerical treatment. In fracture mechanics problems, the symmetry is important to be able to solve the problem because there is a singularity between the surfaces of the plate that the classical boundary element formulation cannot solve.

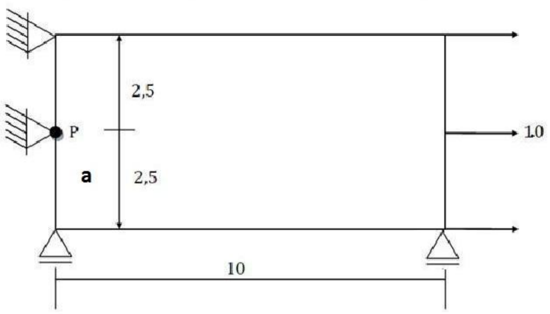

Figure 5. Geometry and boundary conditions of one-quarter of plate region.

The boundary discretization of one-quarter of plate region has a total of 75 linear element and the elements are distributed as can be shown in Fig. 6.

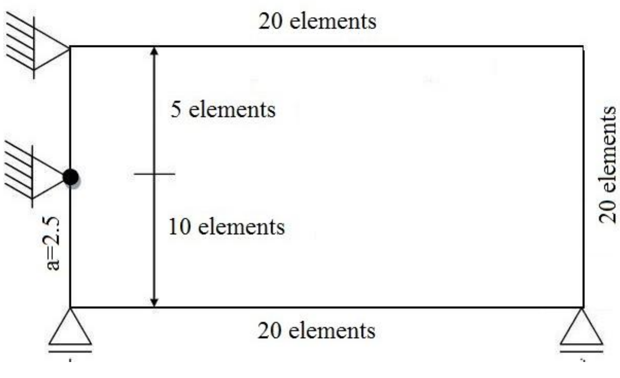

Figure 6. Discretization of one-quarter of plate region.

Figure 7 shown a window of the MEMEC with the boundary conditions and load in a one-quarter of plate.

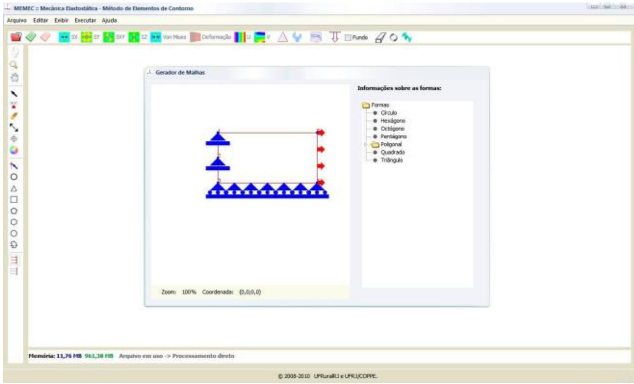

Figure 7: A window of software MEMEC

In Fig. 8 and 9 are presented a window with the field of displacements and stresses respectively in the one-quarter of plate.

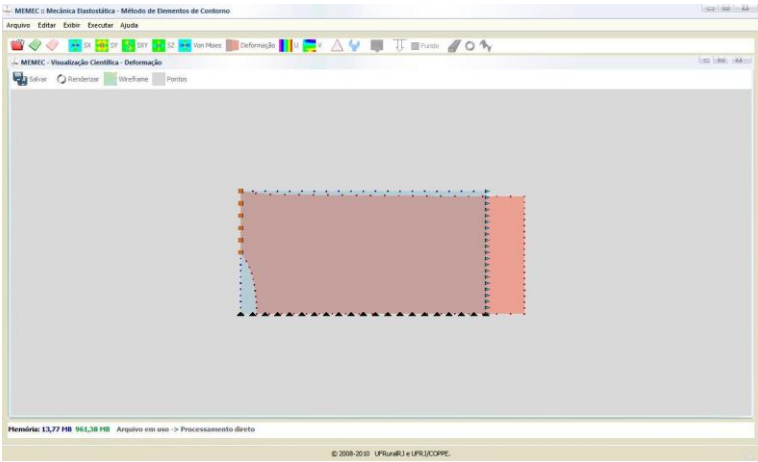

Figure 8. Displacements observed in the software MEMEC

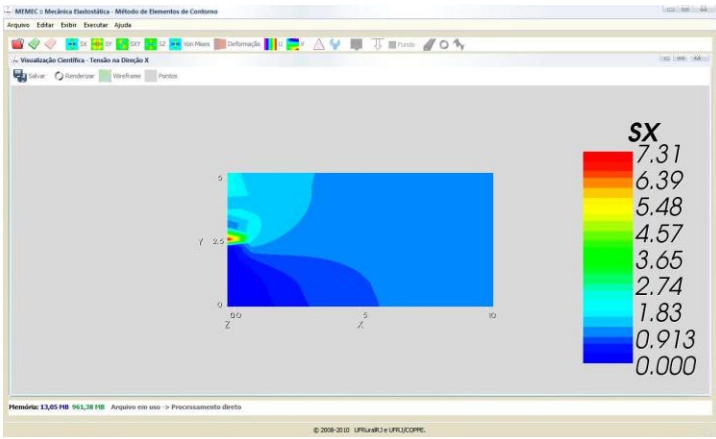

Figure 9. stresses in the one-quarter of plate region.

Example 2: This example is a rectangular plate where the distributed unitary load is applied in the surfaces of the crack (Castor, 1993) as shown in Fig.10. The material properties are Young modulus *E=7.43*, Poisson ratio $\nu = 0.3$. The crack length is $2a = 5$. The gauss integration utilized 20 points. For simple cases it is possible to consider that the external load, show in Fig. 12 produce the same results that the internal load (Mews, 1987), and Castor (1993) proved this. Then, in this example the load consider was the external load.

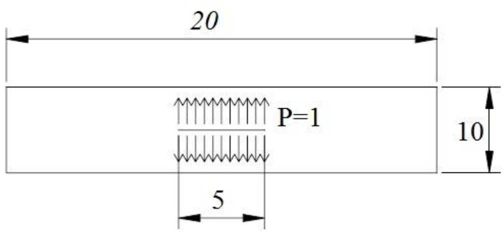

Figure 10. Rectangular plate with a central crack

The boundary discretization of one-quarter of plate region has a total of 300 linear element and the elements are distributed as can be shown in Fig. 11. The region on the surface of the crack has a more refined discretization.

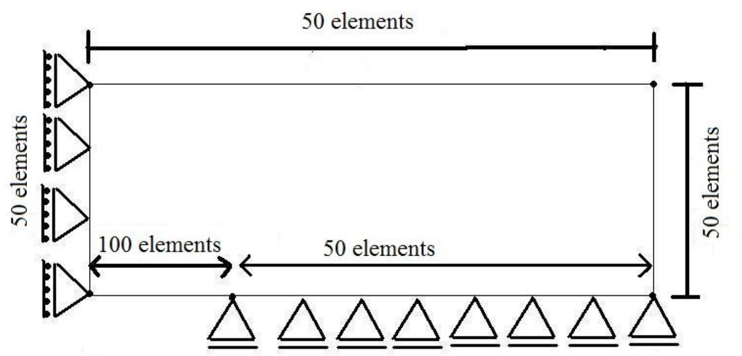

Figure 11: Discretization in the one-quarter rectangular plate

Figure 12 shown a window of the MEMEC with the boundary conditions and load in a one-quarter rectangular plate.

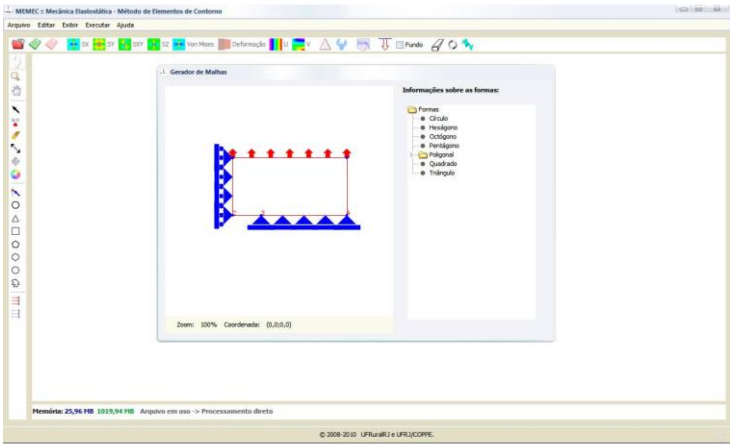

Figure 12: A window of software MEMEC

In Fig. 13 and 14 are presented a window with the field of displacements and stresses respectively in the one-quarter rectangular plate.

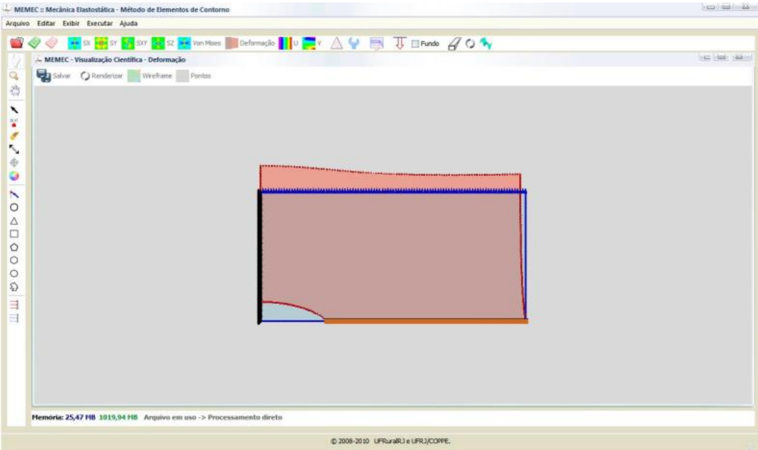

Figure 13. Displacements observed in the software MEMEC

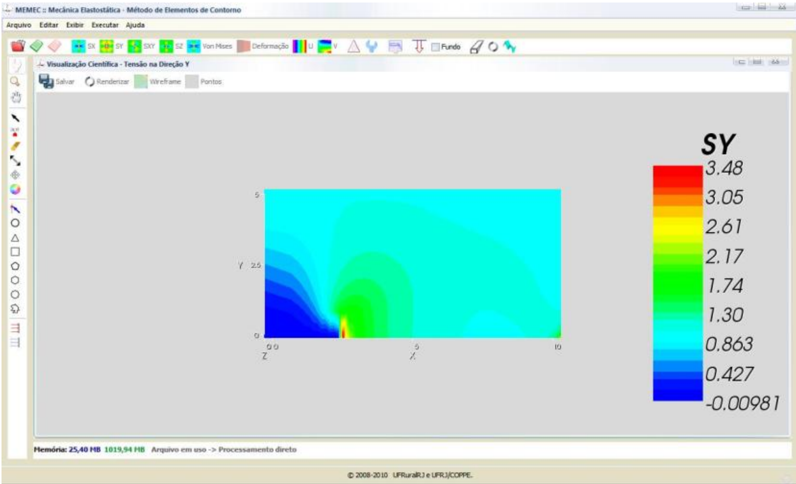

Figure 14. stresses in the one-quarter of plate region.

5. ACKNOWLEDGEMENTS

This work has been carried out with the support of the FAPERJ with the APQ1 program.

6. CONCLUSIONS

It is possible to verify that the software MEMEC has good numerical results, in addition to presenting these results allow a global and local problem. The scientific visualization of the problem and the results is an important point of this work and future possibilities are endless.

It is important to note that a mesh generator is in final implementation and for the near future we intend to implement other applications like the case of heat problems, elastodynamic problems, problems with the semi-infinite and fracture mechanics problems where static and dynamic by using the Green function Numerical. Other proposals will be included along with the development of the work, but the idea is to have a powerful tool for solving problems in science and engineering in general.

7. REFERENCES

Barbosa, M., 2010, *"Um ambiente gráfico interativo para o Método dos Elementos de Contorno"*, Dissertação de Mestrado, COPPE/UFRJ, Programa de Engenharia Civil, Rio de Janeiro.

Blandford, G.E., Ingraffea, A. R. and Ligget, J.A., 1981."Two-dimensional stress intensity factor computation using the boundary element method". International Journal of Numerical Methods in Engineering, Vol. 17, pp. 387-404.

Brebbia, C.A., Telles, J.C.F. and Wrobel, L.C., 1984, "Boundary elements techniques: theory and applications in engineering", Springer, Berlin, 464 p.

Castor, G.S., 1993. "Aplicação de funções de Green a problemas da mecânica da fratura com o método dos elementos de contorno", M.Sc Dissertação, COPPE/UFRJ, Rio de Janeiro, Brasil.

Deitel, H.M. and Deitel, P.J. 2004, Java: How to Program, 6th Edition, Prentice Hall.

Fontes Jr, E. F., 2010, *"Uma abordagem Multithreading para o Acoplamento entre os Métodos dos Elementos de Contorno e Finitos"*, Dissertação de Mestrado, COPPE/UFRJ, Programa de Engenharia Civil, Rio de Janeiro.

Fontes Jr, E. F., M. Barbosa, e C. A. R. Vera-Tudela; 2007. "A Visualização Científica em um problema resolvido com o Método dos Elementos de Contorno e o VTK". *In: Anais do X Encontro de Modelagem Computacional*. Nova Friburgo: UERJ.

Guimarães, S. and Telles, J.C.F., 1994. "On the hyper-singular boundary element formulation for fracture mechanics applications". Engineering Analysis with Boundary Element, Vol. 13, pp. 353-363.

Portela, A., Aliabadi, M.H. and Rooke, D.P., 1992. "Dual boundary element analysis of cracked plates: singularity subtraction technique". International Journal of Fracture, Vol. 55, pp. 17-28.

Schroeder, W., Ken, M. and Lorensen, B., 2004. "The Visualization Toolkit", Kitware, Inc. Publishers.

Telles, J.C.F., Castor, G.S. and Guimães, S., 1995. "A numerical Green´s function approach for boundary elements applied to fracture mechanics". International Journal of Numerical Methods in Engineering, Vol. 38, pp. 3259-3274.

8. RESPONSIBILITY NOTICE

The authors are the only responsible for the printed material included in this paper.