

# OBJECT ORIENTED FINITE ELEMENT IMPLEMENTATION

## Jean Carlo Ferreira de Oliveira

Departamento de Engenharia Mecânica-DEM/CT-UFES, Universidade Federal do Espírito Santo, Campus Universitário de Goiabeiras, Vitória, ES.  
jean@ucl.br

## Fernando César Meira Menandro

Departamento de Engenharia Mecânica-DEM/CT-UFES, Universidade Federal do Espírito Santo, Campus Universitário de Goiabeiras, Vitória, ES.  
fcmm@npd.ufes.br

**Abstract.** The objective of this work was to study the finite element method implementation techniques using object-oriented analysis. This paper presents a software for design of spatial metal structures by means of the finite element method, using the admissible stress method and the Limit States method, joint with the development of a Computer Aided Design environment for structural drawing input and output, and also with the possibility of consideration of semi-rigid joints for the beam element. Among the main characteristics of this system, named CAE 2002, there is the concept of Object Oriented Analysis, which was applied to the finite element method through Object Oriented Programming, implemented in Borland's C++ Builder. Finally, one example of a structure designed using CAE 2002 is shown, demonstrating the developed interface. Through object oriented analysis, a high level of flexibility in the development of a finite element software was achieved, allowing for other studies to be implemented in CAE 2002.

*Keywords: object oriented programming - metal structures - finite element method.*

## 1. Introduction

The structural project is a common job on the engineering field. Structural design put the engineers in front of the physical problems; some are simple and others more complex. The designers have solved the problems in a satisfactory way. In the resolution of physical problems we have to calculate the dimensions of structural components, both for security and cost. However, most of the structures are very complex to be analytically solved. Despite knowing the analytical solution of some standard problems, for a structure of different shape the analytical solution does not hold true and in practice the design problem needs a lot of simplification, resulting in less precision on the calculations.

The use of the FEM – Finite Element Method is a good way to solve complex structural problems. This is the reason why the method is useful to simulate some classes of engineering problems, where the solution is hard to obtain. Together with the utilization of computers the FEM became one of the most utilized engineering tools today.

In the 1950's, when the FEM first appeared, and continuing until the 1980's, the engineer used to write all geometry, mesh, boundary condition, etc., in text file for the preparation of data inputs in cards for processing on a mainframe computer. The method was not more than a highly mathematical algorithm of structural analysis. One of the first systems of public use was the ICES (Integrate Civil Engineering) developed in MIT (Massachusetts Institute of Technology). On next decade, research on the FEM increased, with the development of new technical solutions and with the appearance of SAP (Structural Analysis Program), developed in Berkeley in 1962, which source code was distributed around the world, and that influenced the path of the development of the method.

In the 1970's decade, there were no graphic resources to help the interpretation and the analyses of problem results, and the engineer used the FEM knowing that its utilization was leaning in a conceptual base. Because of the computer graphics evolution since the 1980's, the FEM use has expanded very easily, and on the next decades new elements were created. Numerical methods became more efficient and user-friendlier. Today, large FEM packages use computer graphics routines for pre and post-processing, to create the geometric model and FEM mesh and to analyze the results.

Although there are finite elements programs advanced in some innovative concepts, what consequently implies a higher cost, the majority of them do not perform the calculation of section dimensions of a structure, or structural design, following the Brazilian standard. There are a few studies and programs of calculation of section dimensions in Brazil. However, we feel the need for development of a basis for the creation of a larger finite element package, which should start with a few elements and subsequently improved with the addition of new features. Therefore, the program must be totally flexible for ease of implementation. This can be achieved with the use of software engineering concepts such as structured technical and data modeling with the use of object oriented analysis. Thus, a finite element program with a graphics platform for three-dimensional framed structures and with semi automatic steel structural design was created, which makes use of commercial section catalogs stored as data banks.

In the calculation of metal structures is normal the consideration of them as framed systems. Generally, the finite element analyses of metal structures represents them by truss and beam elements. Therefore, only these two kinds of elements were implemented. Two and three-dimensional elements should be implemented in subsequent work.

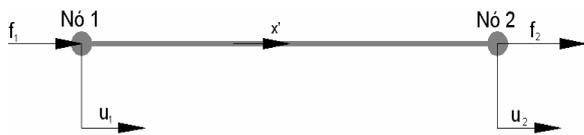
## 2. Finite Elements utilized

The Finite Element Method is an energy-based numerical method for the solution of differential equations applied particularly to boundary value problems, and for this reason useful in partial differential equation resolution. This paper will work with elements for structural analysis, but similar procedures can be applied in other areas of study such as fluid mechanics, heat propagation, electromagnetism and many others.

### 2.1. Truss Element

It is known that a truss consists of a series of linear structural elements, of length a lot longer than the dimensions of its cross section, and connected on its endpoint to compose a framed structure. The important particularity of a truss is the joint, where the discrete joint of two or more members is considered to be articulated. This articulation is obtained physically by making the extension of the line drawn by the center of gravity of each cross section of one element to meet the correspondent center of gravity line of all the other elements that meet at that particular joint. The external forces applied in truss structures act only on structural joints. As a result of the adopted hypotheses about the element joints, and based on the fact of the loading only applied on the nodes, a basic aspect is fundamental for study this kind of element: the truss only transmit axial forces of tension or compression in the truss element direction.

Analyzing the truss element in local coordinate system:



$$K = \frac{AE}{L}, \text{ Hooke's Law}$$

Where

A is the section area,

E is the elastic (Young's) modulus, and

L is the length of the truss element.

Also,

$$\begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = \begin{bmatrix} K & -K \\ -K & K \end{bmatrix} \begin{Bmatrix} u_1 \\ u_2 \end{Bmatrix}$$

$$\{f\} = [k]^e \{u\}$$

Analyzing the truss element in global coordinate we take a truss axial force and decomposed on the following:

$$F_{1x} = f_{1x} \cos \alpha_1 + f_{1y} \cos \beta_1 + f_{1z} \cos \gamma_1$$

$$F_{1y} = f_{1x} \cos \alpha_2 + f_{1y} \cos \beta_2 + f_{1z} \cos \gamma_2$$

$$F_{1z} = f_{1x} \cos \alpha_3 + f_{1y} \cos \beta_3 + f_{1z} \cos \gamma_3$$

$$\begin{bmatrix} \cos \alpha_1 = i.i' & \cos \beta_1 = i.j' & \cos \gamma_1 = i.k' \\ \cos \alpha_2 = j.i' & \cos \beta_2 = j.j' & \cos \gamma_2 = j.k' \\ \cos \alpha_3 = k.i' & \cos \beta_3 = k.j' & \cos \gamma_3 = k.k' \end{bmatrix}$$

$$F_{1i} = e_{i'}.f_{1i'}$$

Where

F is the force in the global system, and

f is the force in the local system.

The transformation for node 1 and 2 of the truss element:

$$\begin{Bmatrix} F_{1x} \\ F_{1y} \\ F_{1z} \\ F_{2x} \\ F_{2y} \\ F_{2z} \end{Bmatrix} = \begin{bmatrix} (i.i') & (i.j') & (i.k') & 0 & 0 & 0 \\ (j.i') & (j.j') & (j.k') & 0 & 0 & 0 \\ (k.i') & (k.j') & (k.k') & 0 & 0 & 0 \\ 0 & 0 & 0 & (i.i') & (i.j') & (i.k') \\ 0 & 0 & 0 & (j.i') & (j.j') & (j.k') \\ 0 & 0 & 0 & (k.i') & (k.j') & (k.k') \end{bmatrix} \begin{Bmatrix} f_{1x} \\ f_{1y} \\ f_{1z} \\ f_{2x} \\ f_{2y} \\ f_{2z} \end{Bmatrix} \quad F = T f$$

Angle Defined:

$\alpha_1$ =angle between axes X and x;

$\beta_1$ =angle between axes X and y;

$\gamma_1$ =angle between axes X and z;

$\alpha_2$ =angle between axes Y and x;

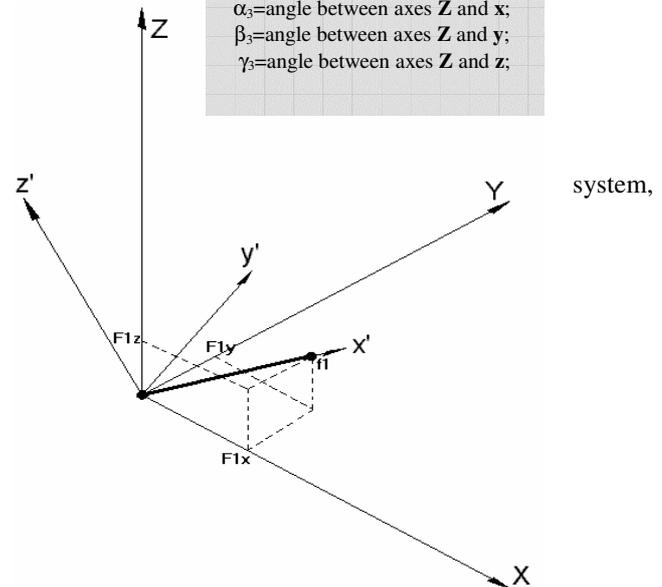
$\beta_2$ =angle between axes Y and y;

$\gamma_2$ =angle between axes Y and z;

$\alpha_3$ =angle between axes Z and x;

$\beta_3$ =angle between axes Z and y;

$\gamma_3$ =angle between axes Z and z;



It is possible to show that  $T^{-1} = T^T$ , by writing  $\mathbf{F}$  in function of  $\mathbf{f}$ . The opposite could be done and then it can be proved that the inverse matrix is equal to the transpose matrix.

Returning to the truss equation and increasing the equation system order of the local coordinate system, to be equal as the order of the global coordinate system:

$$\begin{Bmatrix} f_{1x} \\ f_{1y} \\ f_{1z} \\ f_{2x} \\ f_{2y} \\ f_{2z} \end{Bmatrix} = K \cdot \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{Bmatrix} u_{1x} \\ u_{1y} \\ u_{1z} \\ u_{2x} \\ u_{2y} \\ u_{2z} \end{Bmatrix}$$

Operating the equations, we have:

$$\begin{aligned} \{f\} &= [k]\{u\} \\ [T]\{f\} &= [T][k]\{u\} \\ \{F\} &= [T][k][T^T]\{\Delta\} \\ \{F\} &= [K^e]\{\Delta\} \end{aligned} \quad [K^e] = [T][k][T^T]$$

But,

$$(i.i') = a = \Delta x / L$$

$$(j.i') = b = \Delta y / L$$

$$(k.i') = c = \Delta z / L$$

Where:

L is the truss length,

$\Delta x$  is the truss Projection in axis x,

$\Delta Y$  is the truss Projection in axis y, and

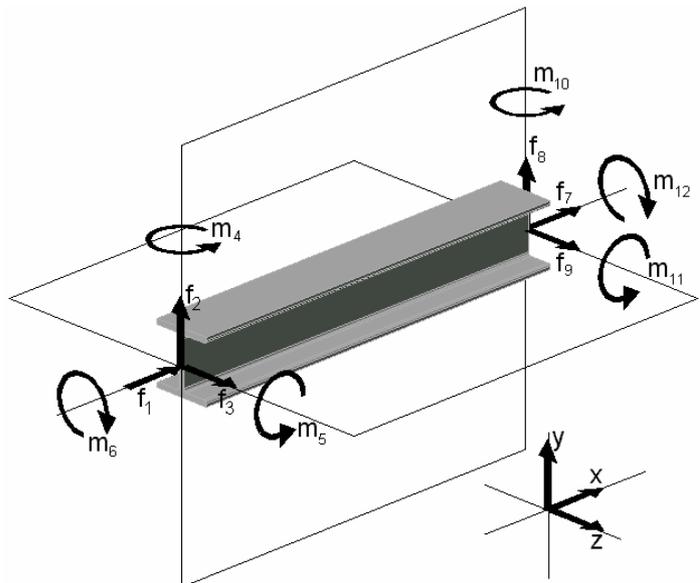
$\Delta Z$  is the truss Projection in axis z.

The Stiffness Matrix in the global coordinate system is:

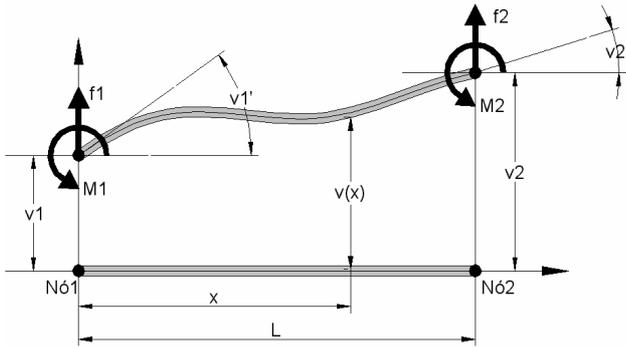
$$K^e = K \cdot \begin{bmatrix} a^2 & ab & ac & -a^2 & -ab & -ac \\ ab & b^2 & bc & -ab & -b^2 & -bc \\ ac & bc & c^2 & -ac & -bc & -c^2 \\ -a^2 & -ab & -ac & a^2 & ab & ac \\ -ab & -b^2 & -bc & ab & b^2 & bc \\ -ac & -bc & -c^2 & ac & bc & c^2 \end{bmatrix}$$

## 2.2. Beam Element

The beam study will be based in the idea of the superposition of the independent physical behavior in the same element and considering them independently as well. The beam can transmit axial forces, bending moments about the two principal axes in the plane of its cross section, and twisting moments about its center axis. The following forces are acting on the beam: axial forces  $f_1$  and  $f_7$ ; shearing forces  $f_2, f_3, f_8$  and  $f_9$ ; bending moments  $m_4, m_5, m_{10}$  and  $m_{11}$ ; and twisting moments  $m_6$  and  $m_{12}$ .

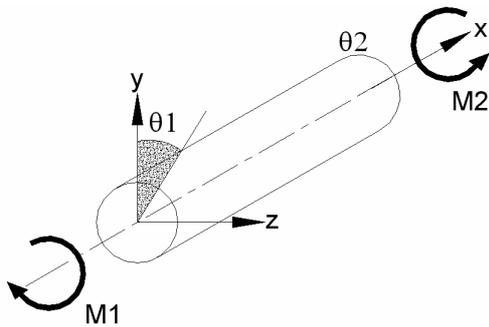


BENDING MOMENTS:



$$\begin{Bmatrix} f_1 \\ M_1 \\ f_2 \\ M_2 \end{Bmatrix} = \begin{bmatrix} \frac{12EI}{L^3} & \frac{6EI}{L^2} & -\frac{12EI}{L^3} & \frac{6EI}{L^2} \\ \frac{L^2}{6EI} & \frac{L}{4EI} & -\frac{L^2}{6EI} & \frac{L}{4EI} \\ -\frac{12EI}{L^3} & -\frac{6EI}{L^2} & \frac{12EI}{L^3} & -\frac{6EI}{L^2} \\ \frac{L^2}{6EI} & \frac{L}{4EI} & -\frac{L^2}{6EI} & \frac{L}{4EI} \end{bmatrix} \begin{Bmatrix} v_1 \\ v_1' \\ v_2 \\ v_2' \end{Bmatrix}$$

TWISTING MOMENTS:



$$M = \frac{GJ}{L} \theta$$

$$\begin{Bmatrix} M_1 \\ M_2 \end{Bmatrix} = \begin{bmatrix} \frac{GJ}{L} & -\frac{GJ}{L} \\ -\frac{GJ}{L} & \frac{GJ}{L} \end{bmatrix} \begin{Bmatrix} \theta_1 \\ \theta_2 \end{Bmatrix}$$

Where

I is the cross section's moment of inertia,

A is the area of the cross section,

J is the polar moment of inertia,

E is the elastic (Young's) modulus,

G is the shear modulus, and

L is the truss length.

BEAM 3D

The stiffness matrix of the beam element was formulated by examining the axial force, bending moments and twisting moments independently. The beam in a general case with twelve degrees of freedom behaves as follows:

$$[k]^e = \begin{bmatrix} a & & & & & & & & & & & \\ 0 & 12b_z & & & & & & & & & & \\ 0 & 0 & 12b_y & & & & & & & & & \\ 0 & 0 & 0 & t & & & & & & & & \\ 0 & 0 & -6b_yL & 0 & 4b_yL^2 & & & & & & & \\ 0 & 6b_zL & 0 & 0 & 0 & 4b_zL^2 & & & & & & \\ -a & 0 & 0 & 0 & 0 & 0 & a & & & & & \\ 0 & -12b_z & 0 & 0 & 0 & -6b_zL & 0 & 12b_z & & & & \\ 0 & 0 & -12b_y & 0 & -6b_yL & 0 & 0 & 0 & 12b_y & & & \\ 0 & 0 & 0 & -t & 0 & 0 & 0 & 0 & 0 & t & & \\ 0 & 0 & -6b_yL & 0 & 2b_yL^2 & 0 & 0 & 0 & 6b_yL & 0 & 4b_yL^2 & \\ 0 & 6b_zL & 0 & 0 & 0 & 2b_zL^2 & 0 & -6b_zL & 0 & 0 & 0 & 4b_zL^2 \end{bmatrix}$$

*symmetric*



$$\begin{Bmatrix} F_{Node1} \\ M_{Node1} \\ F_{Node2} \\ M_{Node2} \end{Bmatrix} = \begin{bmatrix} T1 & 0 & 0 & 0 \\ 0 & T1 & 0 & 0 \\ 0 & 0 & T1 & 0 \\ 0 & 0 & 0 & T1 \end{bmatrix} \begin{Bmatrix} \Delta_{Node1}^{Linear} \\ \Delta_{Node1}^{Angle} \\ \Delta_{Node2}^{Linear} \\ \Delta_{Node2}^{Angle} \end{Bmatrix}$$

### 3. Object Oriented Analysis of Finite Element Programs

For the last thirty years, the finite element method has achieved a place of consensus as a tool for modeling and simulation of physical systems of areas such as structural mechanics, fluid mechanics, electromagnetism and many others. However, it is known that a finite element program has to be more flexible and easier to development for better amplification and modification, not to mention code maintenance.

#### 3.1. Problem

The finite element programmer has, thus, faced a problem: normally the finite element programs are built with many source code lines, which were written in FORTRAN. The source code has many complex data structures that are accessed through the program, being from the start and becoming more and more with its growth an inflexible system. It is easy to observe the difficulty to modify existent source codes and enlarge source codes to adapt new features, new models and new solution procedures. The inflexibility of the programs is demonstrated in some cases: a high degree of knowledge of the totality of the program is necessary in order to work with a little part of the source code, reusability of source code is very hard, small changes of data structures cause big perturbations in the system, and data structures are not reliably integrated.

Another problem derived from the inflexibility is the access to source code of finite element programs to realize modifications and include new packages. The majority of finite element programs are available as closed packages, not permitting the access to its source code.

#### 3.2. Objective of Analysis

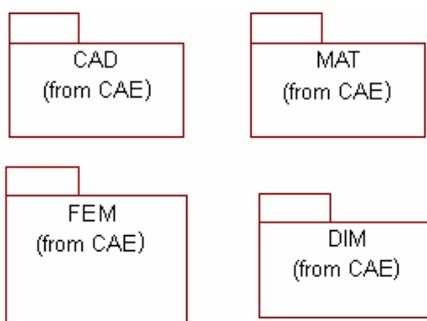
Currently a finite element analysis program is inflexible and shows barriers to some engineering and research applications. Rewriting this system in a new language will not remove its inflexibility. To remove this problem, it's needed a project of a new kind of analysis.

#### 3.3. Object oriented design of a finite element system

Our objective was to develop a finite element software with simple elements, three-dimensional truss and beam elements. However, the base of this project will be quite flexible to allow easy comprehension and development of other elements, other solution procedures, etc.

This work will be divided in four parts:

- 1 – System CAD – Computer Aided Designer
- 2 – System MAT – Numerical Math
- 3 – System FEM – Finite Element Method
- 4 – System DIM – Dimension Design Methods



##### 3.3.1. CAD

The CAD system is integrated with the graphics library called OpenGL. Therefore the components and events of main form will interact with CAE CAD Class and OpenGL. This will allow the creation of linear objects, surface objects and solid objects.

Develop a production CAD system and at the same time integrate it to a FEM system is a hard task. Usually a screen of CAD of a Finite Element Package is not easy to the average user to draw. Small Finite Element Packages have a pleasing desktop to CAD. Our objective is to create a complete platform of CAD using the OpenGL.

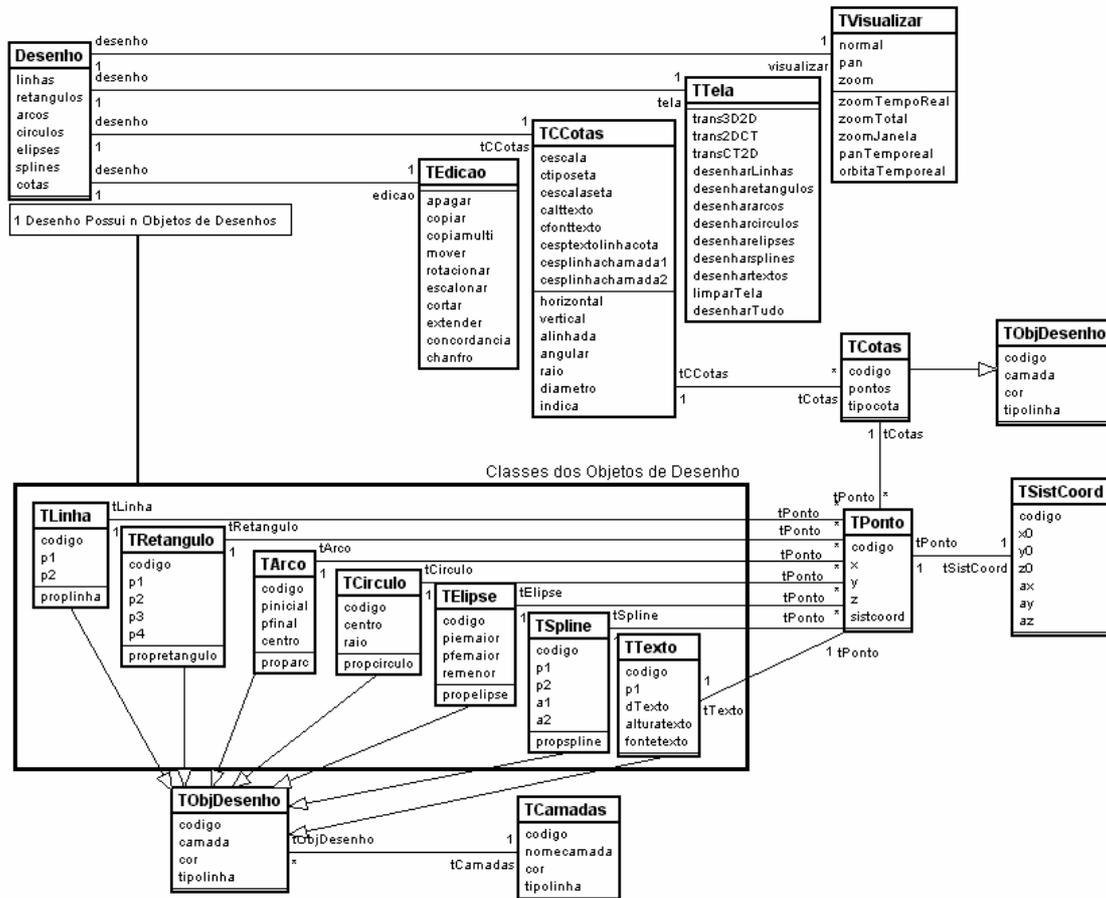


Diagram Class - CAD

### 3.3.2. MAT

This package is compound of classes about numerical methods utilized to solve the finite element problems as integral numerical methods, resolution of linear system, and definition of variable types as arrays, matrices, and others functionality to increase performance.



Diagram Class - MAT

### 3.3.3. FEM

This module will treat of the preprocessor, processor and postprocessor. The packages of CAE (CAD, FEM, MAT and DIM) are integrated and they work in group. However the types utilized and others relationship between classes of the others packages are hidden in the class diagram. The element library classes, analysis class, mesh class and others are shown in the diagram.

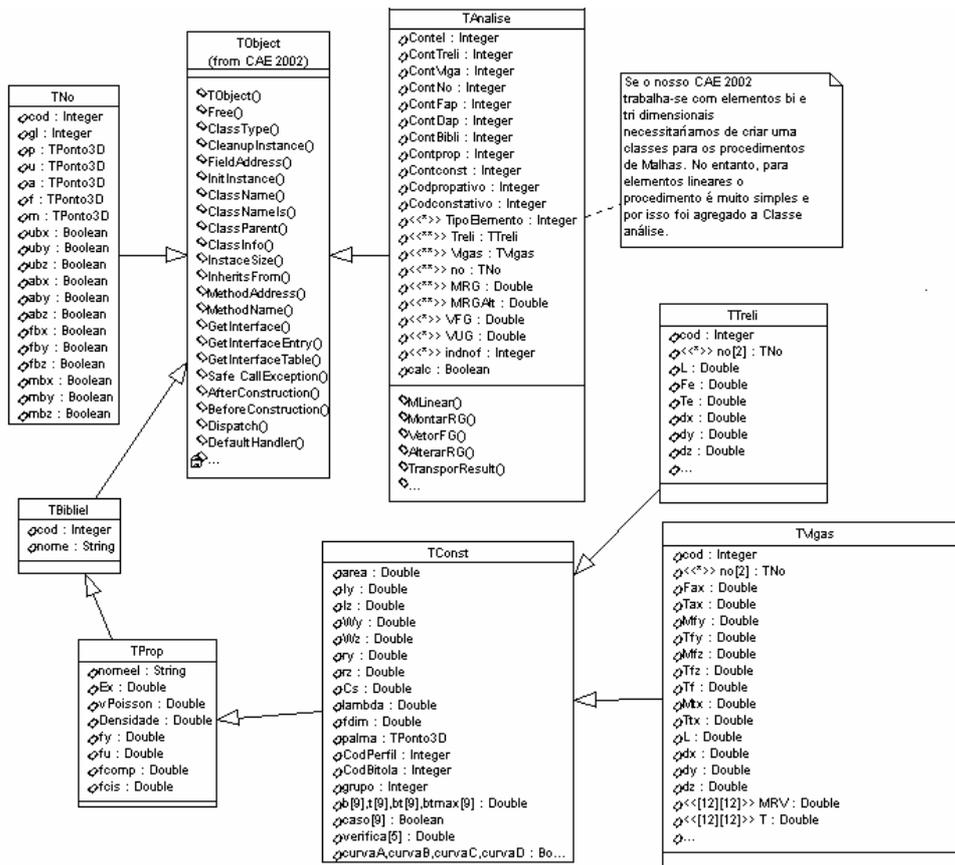


Diagram Class - FEM

### 3.3.4. DIM

This package about the design of steel structures performs the calculation of the section dimensions and consists in the creation of the TDIM Class together with the utilization of the TConst Class attributes. The definition of the TConst attributes starts from the choice of the section until post-processing. After that, we can activate the dimension design tools. For this system the Brazilian Steel Structural Design Standard NBR-8800 is used.



Diagram Class - DIM

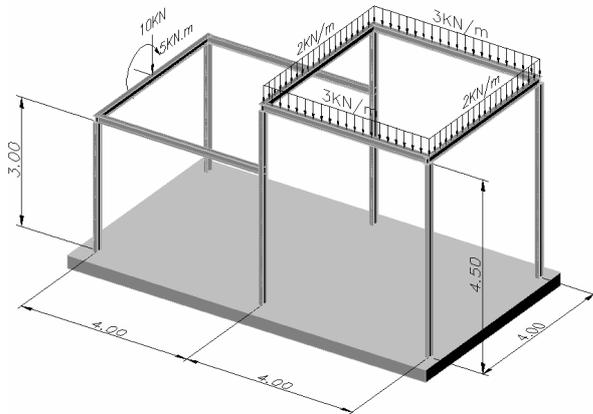
### 3.4 Choice of Program Language

The choice of the program language will depend on the verification of some fundamental requisites such as: execution efficiency, lower level resources, object orientation capabilities, and easy visual programming.

Among the languages studied, the one that best fits the proposed requisites is C++ language. However, as we want to develop a finite element system for desktop, for Windows and for Linux, we have to choose a visual C++ language. The first option was the Microsoft Visual C++, but when studied, that tool did not have visual components pre-programmed, but needed still a lower level of visual code generation, and this was not our focus. Then the choice turned to Borland C++ Builder, since it possesses all efficiency of the C++, and a large range of visual component classes, and integration with many libraries such as MFC, OpenGL, etc.

## 4. Example utilizing the CAE 2003

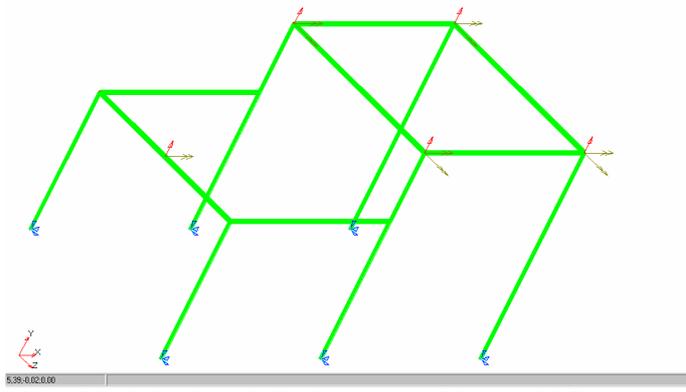
### 4.1. Three-dimensional beam



Spatial frame

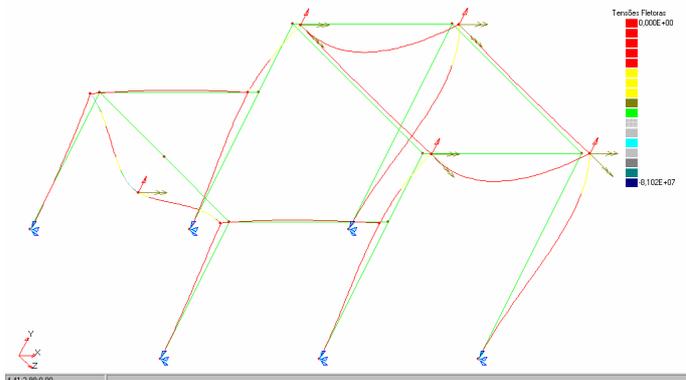
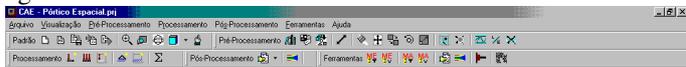
Creating the mesh of the problem above, we will inform the perpendicular plan of section for the beam element, the distributed loads, and the displacements.

Then let us start with the section I – S310 x 74 Kg/m.



Linear Mesh, loads and displacements of the spatial frame

Viewing the deformed structure, we can choose to observe either the axial stress, or bending stress, or the twisting stress.



Visualization of the deformed spatial frame

Then the design of the frame utilizing the admissible stress method is performed. It was verified that all beams support well the applied forces, except the beam 3. It exceeds the value in 21%. A new cross section can then be chosen in the steel laminate library.

Therefore, two examples of metallic structural design utilizing CAE 2002 were shown, which demonstrate the user friendliness and efficiency of the proposed algorithm. It is convenient to remember that although the examples utilize the admissible stress method, the limit state method of NBR-8800 is also within the scope of the program, but was not shown for lack of space.

## 5. Conclusion

This paper shows a new conceptual basis in finite element programming. The basis of a large system was laid down with a complete formulation of truss and beam elements, and a concurrent design module. Also shown is a new approach to the construction of a finite element system, utilizing the concept of object oriented analysis. A complete description of procedure utilized in a finite element program and a commercial section library in its database are also part of the proposed system. This research had as final objective a finite element system integrated in a CAD desktop environment, in its essence object oriented and of easy manipulation. This software realizes the design according with the admissible stress method and the limit state method, and also with the possibility of consideration of semi-rigid joints for the beam element.

Beyond the simple implementation of a finite element program, this work included special characteristics in the model, as for example the question of the deflection due to shearing force, the design criterion, the real stiffness of the joints in the frame. The implementation of these characteristics involved little programming work in the program because of the flexibility of the object-oriented system.

The success obtained in this implementation shows the flexibility and easy maintenance obtained in the system because of the adoption of the object-oriented paradigm. This was the motivation of the present work and is the main conclusion that we can obtain.

## 6. References

- Archer, G. C., 1996, "Object Oriented Finite Element Analysis", Doctor of Philosophy in Engineering-Civil, University of California at Berkeley.
- Associação Brasileira de Normas Técnicas - ABNT, 1986, "NBR-8800, Projeto e execução de estruturas de aço de edifícios: método dos estados limites", 2<sup>nd</sup> Ed., Rio de Janeiro, RJ, Brazil.
- Baloupos V., John F. A., 2002, "Use of shallow class hierarchies to facilitate object-oriented nonlinear structural simulations", Cornell University.
- Bathe, K.-J., 1996, "Finite Element Procedures", Prentice Hall.
- Booch, G., 1994, "Object-Oriented Analysis and Design with Applications", 2Ed., Benjamin/Cummings Publishing Company, Inc.
- Ellis, M. A.; Stroustrup, B., 1993, "C++ Manual de Referência Comentado", Rio de Janeiro, Campus.
- Jacobson, I., 1992, "Object Oriented Software Engineering", Addison-Wesley.
- Lichao Yu, Ashock V. Kumar, 2000, "An object-oriented modular framework for implementing the finite element method", University of Florida.
- Mackerle J., 1999, "Object Oriented techniques in FEM and BEM, A bibliography", Linköping Institute of Technology.
- Nickamp R, Stein E., 2001, "An object-oriented approach for parallel two-and three-dimensional adaptive finite element computations", University of Hannover.
- Oliveira, J. C. F. de, 2002, "Dimensionamento de estruturas metálicas espaciais com o desenvolvimento de um software de elementos finitos orientado a objeto", Master's Thesis, Universidade Federal do Espírito Santo, Vitória, ES, Brazil.
- Rumbaugh, J., 1994, "Modelagem e Projetos Baseados em Objetos", Editora Campus.