

Advances in BIM Technology for 3D Modeling and Meshing of Axi-symmetrical Pressure Vessels for the Computation of Limit Load

*José Ricardo Queiroz Franco¹⁾ and Alexandre Balabram²⁾

^{1), 2)} *Department of Structural Engineering, UFMG, Belo Horizonte, Brazil*

¹⁾ franco@dees.ufmg.br

ABSTRACT

Concepts of Building Information Modeling - BIM technologies were applied for modeling the geometry of pressure vessels, using thin axi-symmetrical shells elements. A BIM type application has being developed based upon the programming environment of the API ObjectARX (AutoCAD Runtime Extension). The API ObjectARX provides a programming interface that allows developers to use, customize and extend the capabilities of AutoCAD. The BIM application presented here performs geometrical modeling of pressure vessels and their discretization into axi-symmetrical finite elements for limit and shakedown analysis. AutoCAD is used as a graphical platform, taking advantage of its open architecture, which provides direct access to AutoCAD's graphical system, database structures and native command definition. New customized entities are generated by the BIM application developed using the concepts of BIM technologies, which require that all information, data and attributes of such entities are stored and managed through a single database and parameterized BIM digital models. The new entities, objects and commands of the BIM application can be ordered by command line, menu or toolbars of AutoCAD. The BIM application is loaded into AutoCAD as a plug-in where the new entities can be used to model pressure vessels in a practically indistinguishable way from the native built-in AutoCAD entities. Advances in a finite element formulation for the limit load problem of pressure vessels are also presented. An error estimator and adaptive strategies developed based on the energy norm is used to optimize the solutions.

1. INTRODUCTION

A finite element technique for limit analysis of pressure vessels has been developed by Franco and Ponter (1997, Parts 1, 2) and implemented in a BIM code using the paradigm of object oriented programming in C++. The code consists of three separate modules: *the modeler or pre-processor, the analysis code and the post-processor*. This

¹⁾ Professor

²⁾ Graduate Student

article is mainly concerned with the application of the concepts of BIM technologies to construct a modeler for pre and post-processing of geometries and collapse mechanisms of pressure vessels. For Nam et al. (2008) the concept of BIM (Building Information Modeling) is becoming important from the point of integrated management on information generated during the lifecycle of the building in advanced AEC (Architecture, Engineering, Construction) industries. This study aims to apply the concept of BIM as a tool supporting information to support the process of conception, limit analysis and pre-design of pressure vessels. The BIM models allows decisions-making based upon the integration of various types of structural data, information and knowledge generated by different professionals. In recent years, Building Information Modelling has become an active research area in the solutions of problems related to information integration and interoperability. (Isikdag 2010) stated that Building Information Models (BIMs) are promising to be the facilitators of integration, interoperability and collaboration for the future of the construction industry. Building Information Modelling (BIM) is an IT enabled approach that involves applying and maintaining an integral digital representation of all building information for different phases of the project lifecycle in the form of a data repository. The building information involved in the BIM approach can include both geometric data as well as non-geometric data. BIM is one of the important areas in current Virtual Reality (VR) research, (Ning Gu and London 2010). BIM is expected to envision efficient collaboration, improved data integrity, (Ellis 2006), intelligent documentation, distributed access and retrieval of building data (Popov 2006). A unique database of a BIM system facilitates the creation, capture and recovery of integrated information for design, construction and operation of pressure vessels.

As stated by Cerovsek (2010), any BIM application should be constantly improved and be developed as an evolutionary ontology. This first version of the geometrical modeler was constructed taking advantage of the open architecture of AutoCAD, which provides an object oriented C++ programming interface for developers to use, customize and extend AutoCAD. The API ObjectARX consists of a programming environment containing a set of classes and methods ready for use as a library for programming AutoCAD. The open architecture of AutoCAD allows a direct access to its graphical system, database and native proceedings. As a result the customized entities, objects and commands can be ordered as extensions of AutoCAD, through the command line, toolbars or menu without distinction from AutoCAD built-in functions. Another great advantage of using ObjectARX is that native and customized entities created by the modeler can be controlled and maintained through AutoCAD database. The application has been constructed to be independent of the analysis code, so that it not only became much lighter to run but also can be used as a pre-processor for other analysis software by simply developing appropriate interfaces.

2. OBJECT ORIENTED PROGRAMMING APPLIED TO BIM APPLICATION

Implementation of a BIM application code requires a great programming effort to build a good software containing some basic qualities such as good development speed, maintainability, reliability, robustness, legibility, extendibility, portability and reusability. As pointed out in (Dubois 1992 and Dubois 1993), object oriented codes

have all the merits of those features. An object is just a run-time instance of some class. A class works as a mold for the object. For example, the attributes and data structure representing cylindrical geometries is an instance of the class cylindrical shell.

2.1 Criteria and Advantages of O.O.P

The set of criteria adopted for O.O.P. has been divided by (Meyer 2007), in three categories: **Method and language**: covers programming language, notation, textual or graphical, used for analysis and design of the software. **Implementation and Environment** : describes the basic properties of the tools to apply object-oriented ideas. **Libraries** : cover the availability of basic libraries and the components needed to use libraries and produce new ones. Rely on the reuse of software components. Besides the definitions of object and classes the following set of features related to the first category, are very important to the paradigm of O.O.P.: *encapsulation, message, methods, single multiple and repeated inheritance, polymorphism, dynamic binding*,. The second category includes *automatic and fast update, persistence, documentation and browsing*. The third encompasses basic libraries, graphics and user libraries and library evolution and indexing mechanisms.

3. GRAPHICAL PROCESSING AND INTERFACES

Most software systems for structural analysis are interactive with the users through graphical and other friendly interface techniques. The major contribution of the present work is to provide a simple application for modeling and discretizing axi-symmetrical pressure vessels for stress analysis. The idea is to provide an application with an intuitive interface that can export data for different F.E. analysis packages, although it has been built for a specific system for limit and shakedown analysis of pressure vessels. The object-oriented BIM environment of the computer system here consists of two dynamically linked libraries(DLL), developed using ObjectARX, which shares the same memory space with AutoCAD. The first application, which has the name extension DLL replaced by ARX, implements an interface with the user, treating tasks ordered through command lines, dialog boxes and modified menus. The other, with extension DBX (Data Base eXtension), takes care of the creation, visualization and transformations of customized objects and entities. Such organization of the code was suggested in ObjectARX Wizard and ObjectARX Develop'sGuide (Autodesk, Inc., 2001).

The present BIM system should provide interactive means for the generation of pressure vessels geometries and collapse mechanism and for modeling and meshing of finite elements. The geometry will be graphically represented through a convenient two-dimensional representation of half of the longitudinal cut through the axis of revolution. Edition of the geometry will be possible using the grip points. Representation of a solid three-dimensional model of the vessel can be generated by using AutoCAD resources for rotating the longitudinal cut around the axis of revolution. The application will produce the Finite Element models using available tools to generate meshes. Editing and refining the mesh is also one of the facilities of the application. Although the BIM modeler can impose restrictions on any node, limitations of the analysis system for limit analysis, developed to interoperate with it, treats end conditions only at the far end

nodes. The loads implemented for the analysis are a ring load applied to any node of the discretized vessel and an internal pressure. The user can also input properties of the vessel isotropic material to the BIM application.

3.1 BIM Application for the definition of customized entities and objects

This is a DBX type library and was developed to implement entities for finite element modeling. The entities have been grouped as geometrical and those for the finite element modeling. The geometrical entities were developed to define some class of primitive shells, which by combination allow the user to reproduce a large range of pressure vessels geometries found in the industry. These primitive shells are the cylindrical, conical, spherical and toroidal shells. The finite element entities represent axi-symmetrical elements with two nodes encompassing the nodal restrictions and the loads. An special entity has been derived to store the shell material properties. Several AutoCAD layers has been created to distinguish the different types of entities or even to the separate the components of an entity. Thus, there are separate layers for the shells elements, for the mesh, for the node numbers, for element number, for the restrictions and for the loads. Users can take advantage of this separation to display information contained in layers according to their own convenience. One can easily change color, type and thickness of lines of any set of entities.

Concepts to derive personalized objects and entities There are two super-classes available in the API ObjectARX to derive specific types of objects. The pure object, with no graphical representation, has its definition and declaration done through the class named `AcDbObject`. The entities, however, are defined through the class `AcDbEntity`. The four initial letters indicate that both classes belong to the AutoCAD database. Within the hierarchy of classes `AcDbEntity` is derived from `AcDbObject`, since all entities are special types of objects with graphical representation. Customized entities contain specific characteristics determined by the programmer, which can have graphical, logical and functional behavior totally controlled by the application and by AutoCAD.

Geometrical Entities Figure 1 shows the tree of classes of the geometrical entities. The arrow points from base class to the derived class. Thus, the abstract classes `CCap` and `CShell` are derived directly from `AcDbEntity`. The primitive classes `CCylindrical`, `CCone`, `CSpherical` and `CToroidal` inherit the behavior of the native general entity of AutoCAD and can also use the attributes and methods of the abstract shell.

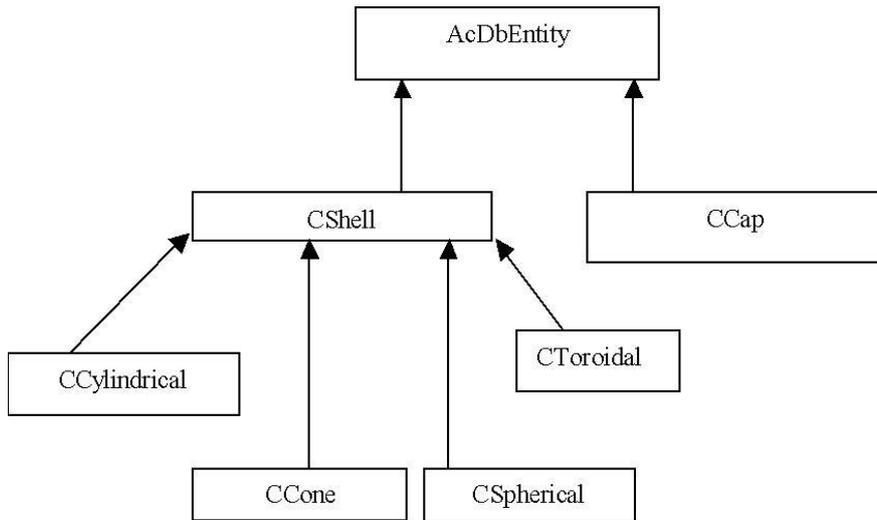


Fig. 1 Tree of Geometrical Entities Classes

The geometrical classes can be illustrated as shown next:

CShell Non-instantiated abstract class. Contains all common attributes and functions of all the other primitive shells derived from it as shown in Fig. 2.

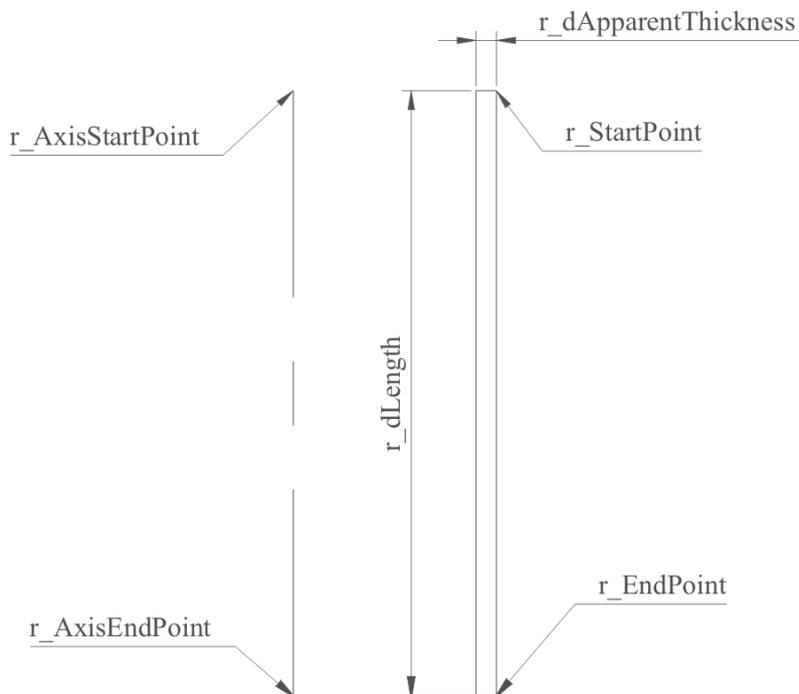


Fig. 2. Attributes of the Class CShell

CCylindrical This class implements the cylindrical shell and the corresponding attributes is shown in Fig. 3(a).

CCone This class represents the conical shells and conical trunks and the corresponding attributes is shown in Fig. 3(b).

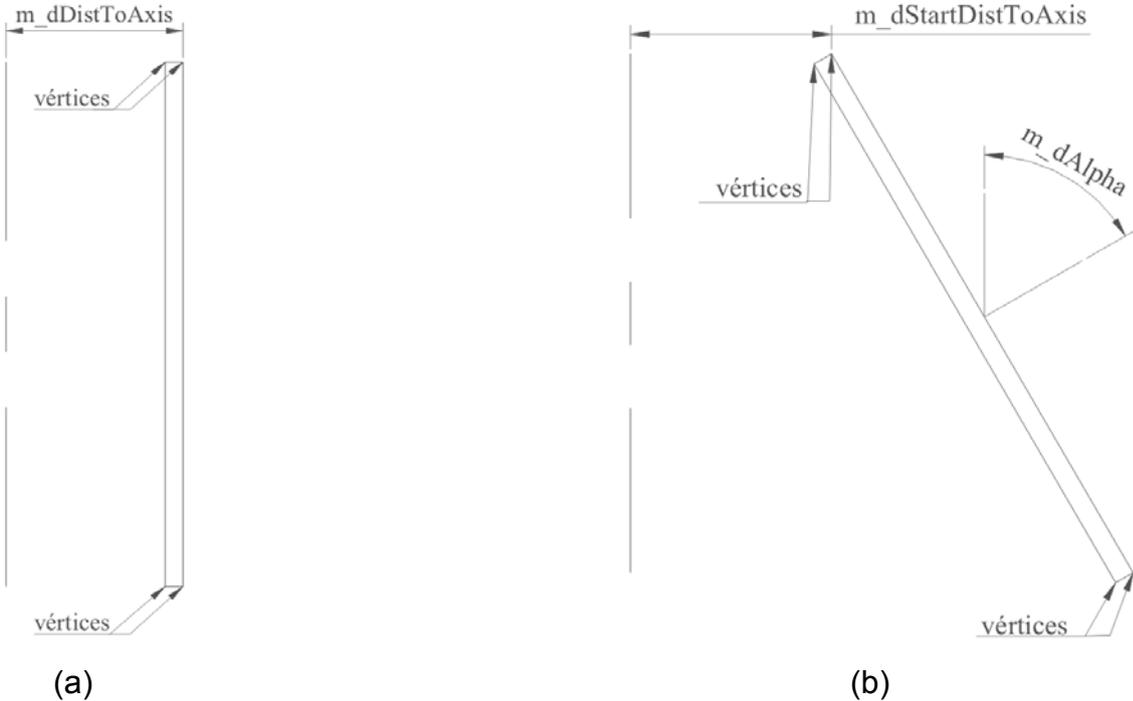


Fig. 3. Attributes of the Classe CCylindrical and CCone

CSpherical Class representing the spherical shells and the corresponding attributes is shown Fig. 4(a).

CToroidal Class that represents the toroidal shells and the corresponding attributes is shown Fig. 4(b).

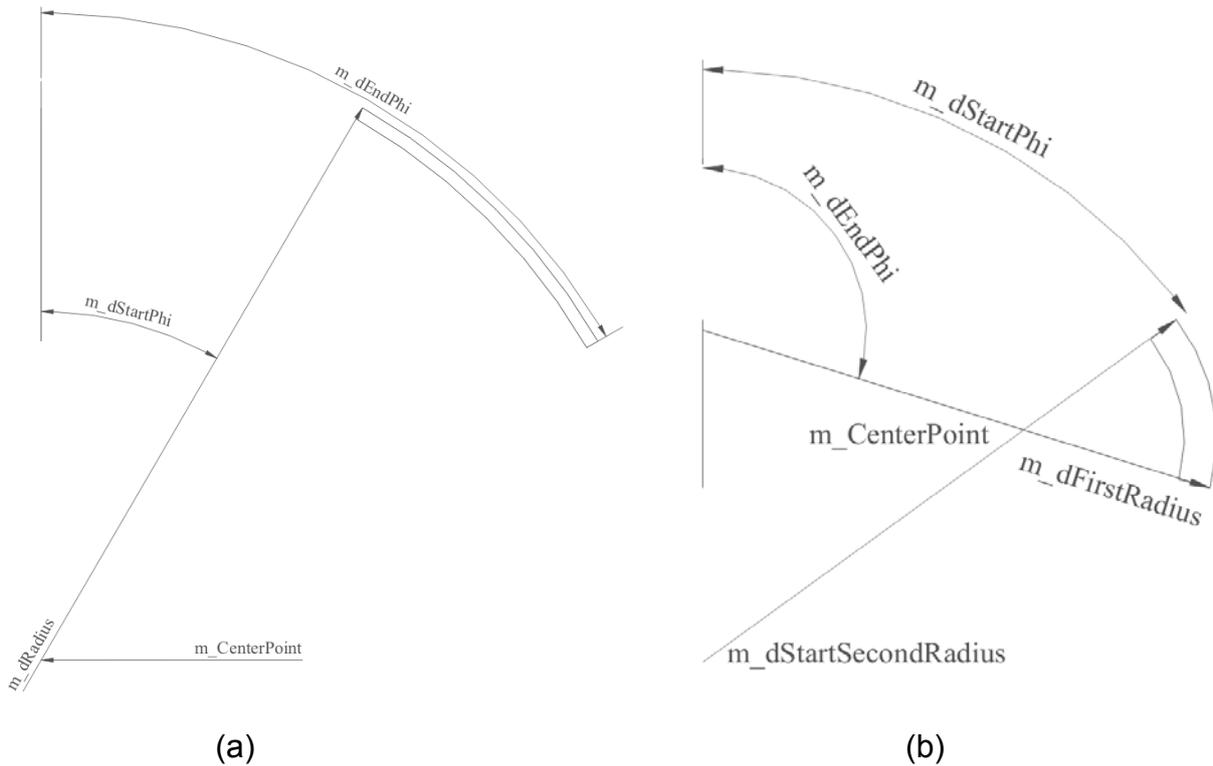


Fig. 4. Attributes of the Classe CSpherical and CToroidal

Entities and objects of the Finite Element Model Figure 5 shows the tree of useful classes created for modeling finite elements of axisymmetrical shells.

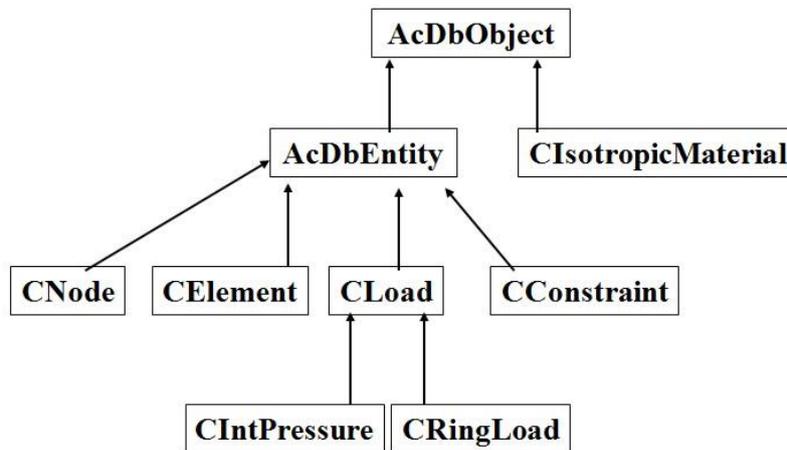


Fig. 5. Entities and objects tree for a Finite Element Model

Since the material is assumed to be an abstract type of object, with no graphical representation, the class CIsotropic Material is derived directly from the AutoCAD native object AcDbObject.

Nevertheless, the entities CNode and CElement have been derived directly from the AutoCAD native class AcDbEntity, since both need to be represented graphically.

CNode entity is the most primitive of a finite element model created for limit analysis of pressure vessels. Its graphical representation and attributes are shown in Fig. 6.

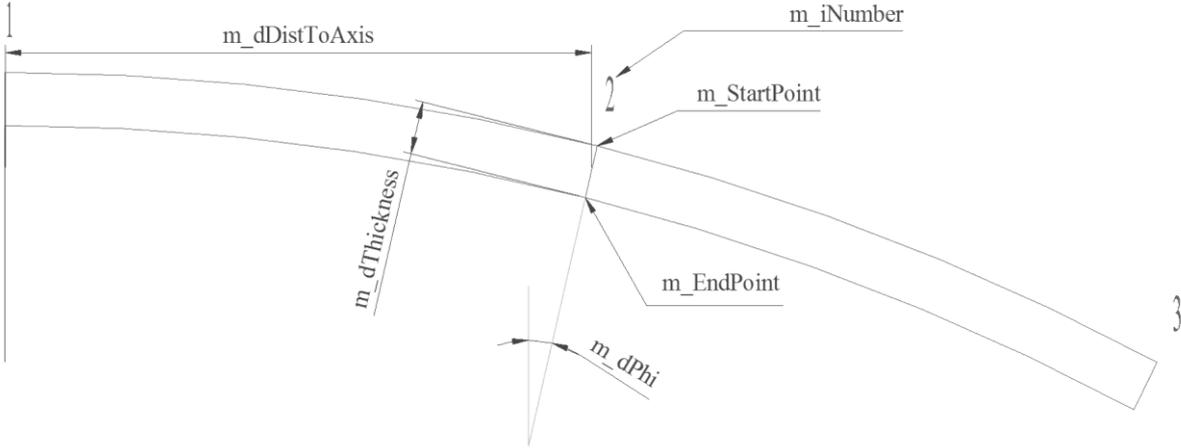


Fig. 6. Attributes of the Entity CNode

Differently from the abstract class CShell, where a customized entity was derived for each geometrical primitive, the entity CElement represents a general finite element containing attributes for different types of axi-symmetrical shell elements. Therefore, the axi-symmetric geometry of a thin pressure vessel, defined by the composition of shells of different geometries, can be discretized by this single entity.

The CElement entity was derived to discretize thin shells geometries into finite elements for limit analysis of pressure vessels. The entity CElement represents a two-node element, whose attributes and graphical representation are shown in Fig.7.

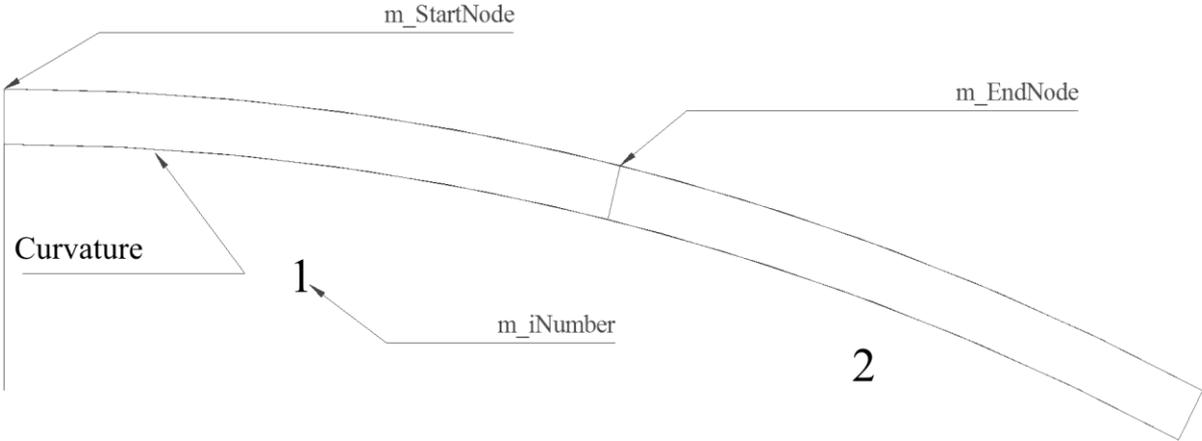


Fig. 7. Attributes of the Entity CElement

The class CConstraint responsible for the constraint symbols is also derived from AcDbEntity. Its attributes and graphical representation are shown in Fig. 8.

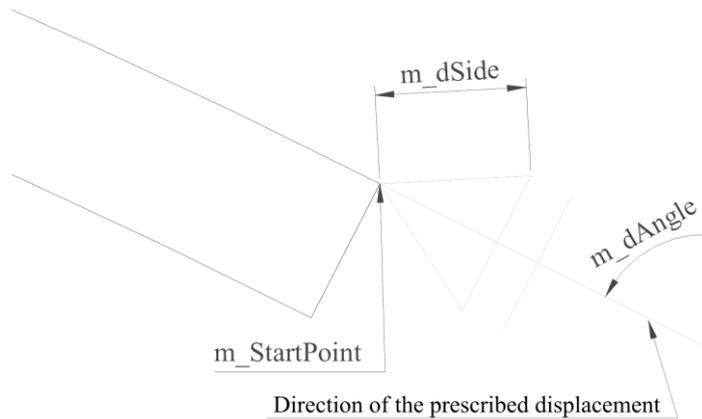


Fig. 8. Attributes of the Entity CConstraint

The BIM application deals with two types of loading for which the class CLoad was created. The class Cload stores the attributes and methods that define the loads of a ring load (CRingLoad) and of an internal pressure (CIntPressure). Their attributes and graphical representations are illustrated in Figs. 9.

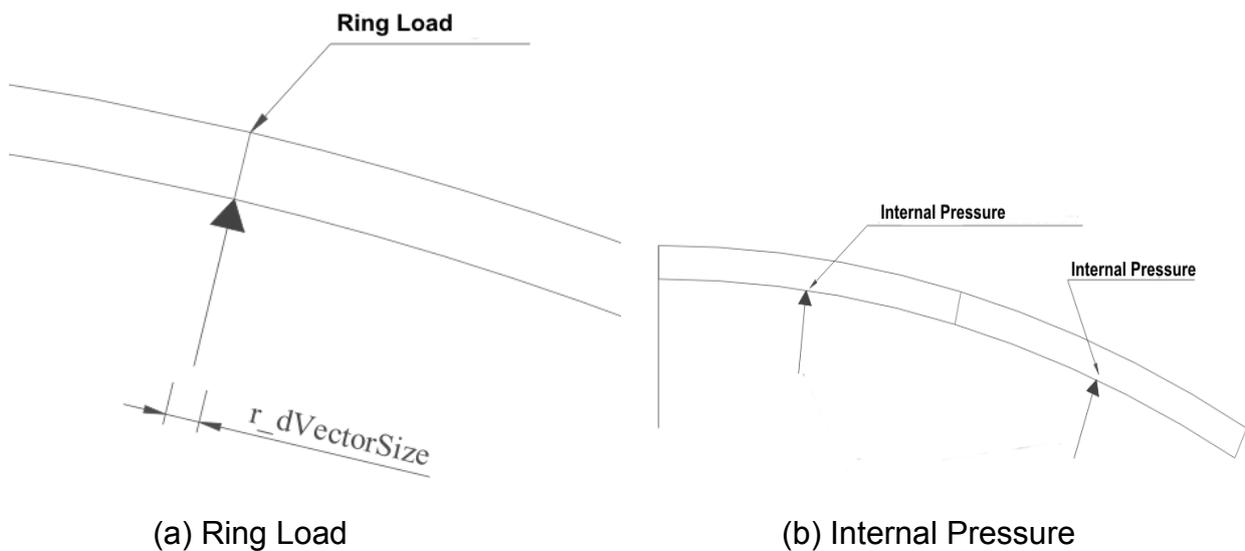


Fig. 9. Cases of Loading

3.2 Interface Application for the User

This ObjectARX BIM application implements modeling commands for representation of the vessel geometry and for post-processing results, which allows visualization of collapse mechanisms and the error estimator results. This work refers only to the pre-processing of the geometry model and the graphical post-processing of results. The application customized entities, objects and commands can be ordered by command line, toolbars or menu and will be virtually indistinguishable from built-in AutoCAD's related. The application menu and toolbars added to AutoCAD standard screen is shown in Fig. 10.

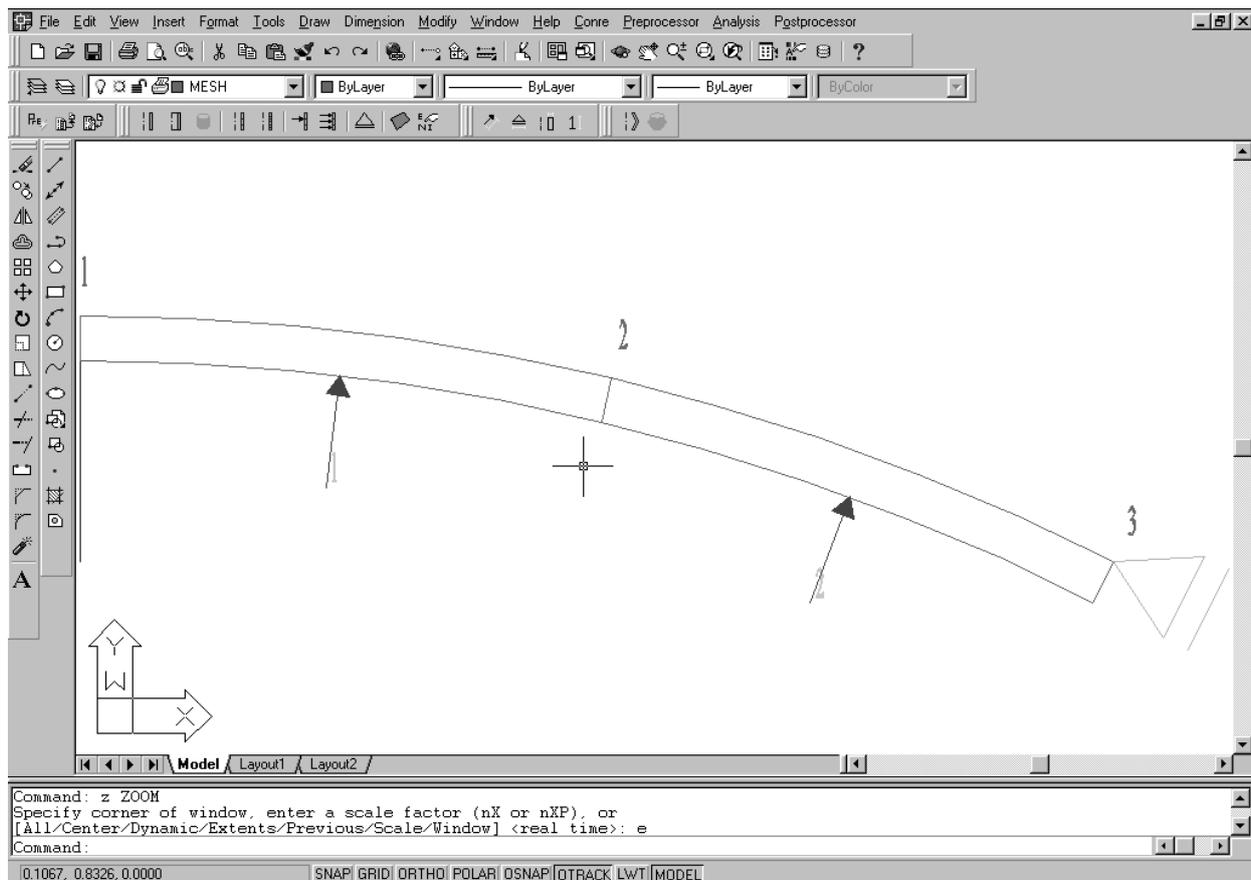


Fig. 10. AutoCAD Customized Screen

Geometric Commands The BIM application contains a list of commands created to generate geometries and collapse mechanisms of pressure vessels. The commands can be ordered by command line, toolbars or menu pulldown. The following icons on the toolbar of the application can be fired for BIM modeling geometries, to generate finite element meshes and view results graphically:

CreateShell Icon  Creates the vessel profile using one or more primitive shells including the axis of revolution. The axi-symmetry of the pressure vessel allows a plane representation of the its geometry by half of the longitudinal cut through the axis of revolution (Fig. 11b). The class CShellDialog implements a dialog box to complement this command.

CdTcLimRevolve Icon  This command generates a solid of revolution by rotating the shell profile around the axis of symmetry (Figs.11a,b).

CdTcLimCAP Icon  This command allows the user to close the top or bottom of a cylindrical vessel.

Finite Element Meshing Commands **MeshShell Icon**  This command discretizes the shell into finite elements. The actions of the command is to run through each type of shell composing the vessel, which is highlighted and then to open a dialog box where

the user can define the number of elements the highlighted shell will be subdivided into and the size ratio between the previous element and the subsequent.

CdTcLimRefine Icon  The user can order this command if he wants to refine any element.

CdTcConstraintMesh Icon  This command applies restrictions to any node of the element mesh, although the analysis considers restrictions to far end nodes only. A dialog box is also associated to the command.

CdTcLim IntPressure Icon  Create an entity for the class CIntPressure on layer LOADS and add it to the load group(LOADLIST).

CdTcLimRingLoad Icon  Applies a ring load to a prescribed node.

CdTcLimMaterial Icon  Define the shell material properties.

CdTcLimListMaterialProperties Icon  List properties and attributes of the material model.

Commands for viewing and editing the appearance of entities The application provides a few other commands to edit the appearance or shape of some entities such as the vectors representing the loads, the restriction symbols, the shell thickness, etc... These commands, shown next, helps to improve visualization of small entities when compared to the vessel scale.

CdTcLimSetSize Vector Icon  This command alters dimensions of load vectors.

CdTcLimSetSize Symbol Icon  This command alters dimensions of constraint symbols.

CdTcLimSetSize Thickness Icon  This command alters the thickness of the shell geometry and of the elements by using a scaling factor input by the user.

CdTcLimSetSize Number Icon  The visual appearance of nodes and elements numbers can be scaled by this command allowing a better representation.

Commands for files manipulation There are also commands create to manage files manipulation, which one can use to load, download, import and export files.

CdTcLimLimite Icon This command loads the two application of the BIM graphical processor into AutoCAD

CdTcLimExportDataFile Icon This command exports data files for a limit analysis finite element system using the extension .dat. The file is then saved to a specific directory.

CdTcLimOpenResultsFile Icon This command imports data from the finite element limit analysis for post-processing the results.

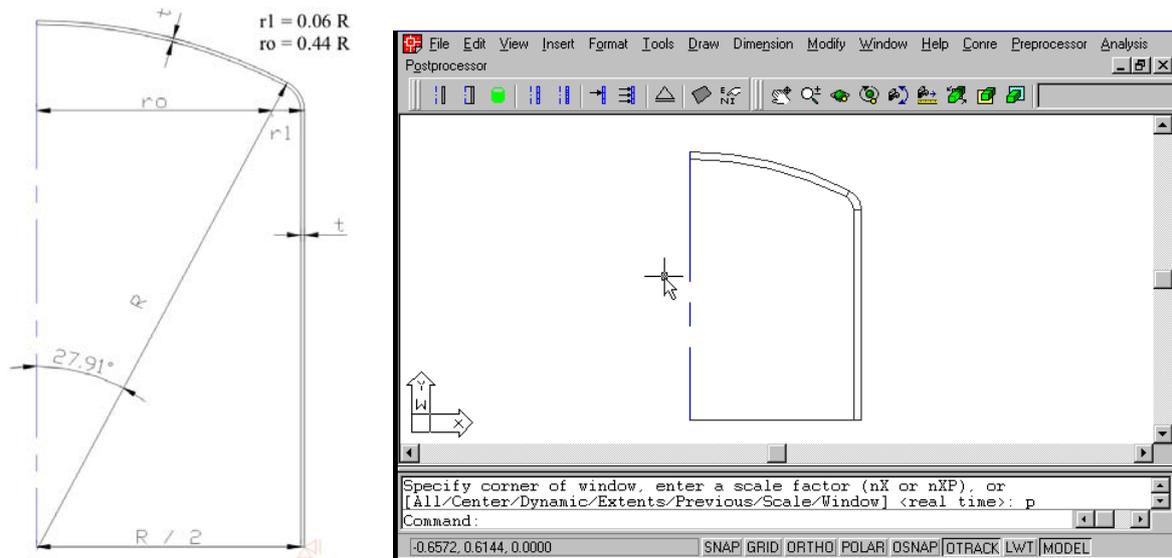
Commands for graphically post-processing results In order to make the use of this post-processing command easier, the class CFileReader created for the object oriented F. E. system was reused. The main attribute of this class is a file.

CdTcLimCollapseMechanism Icon  This command plots the collapse mechanism in two dimension, representing a plane representation of the mechanism by half of the longitudinal cut through the axis of revolution.

CdTcLimDeformSolid Icon  This command display the solid 3D geometry of the collapse mechanism of the pressure vessel.

4. EXAMPLE

The example presented here will illustrate the potential and quality of the BIM application developed to model axi-symmetrical pressure vessel subjected to an internal pressure. The geometry chosen is of the ASME standard torispherical vessel composed by three primitive shells; spherical, toroidal and cylindrical. The geometrical parameters and BIM model are shown in Figs.11(a,b).

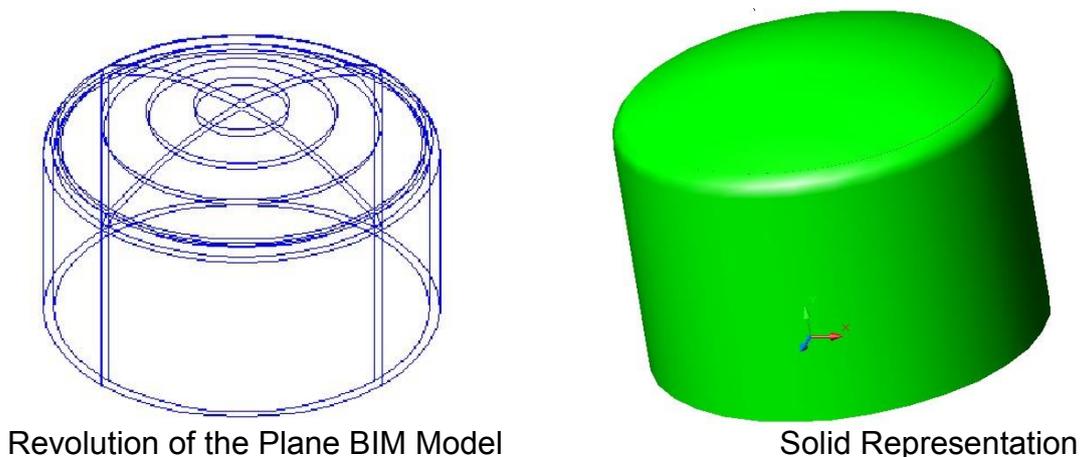


(a) Geometrical Parameters

(b) Geometrical BIM Model

Fig. 11. Standard ASME Torispherical Pressure Vessel

The BIM application uses the axi-symmetry of the pressure vessel to create the plane representation of the vessel geometry(BIM model) displayed as half of the longitudinal cut through the axis of revolution as in Fig. 11(b). The result of the revolution of such a cut around the axis is represented in Figs. 12.



Revolution of the Plane BIM Model

Solid Representation

Fig. 12. ASME Torispherical Pressure Vessel - Solid Representation

The command **Create Shell** associated to a dialog box shown in Fig. 13 were applied to construct the BIM model shown in Fig. 11(b). In this case the geometry of the ASME vessel is constructed starting with the spherical cap and subsequently adding the toroidal and cylindrical shells;

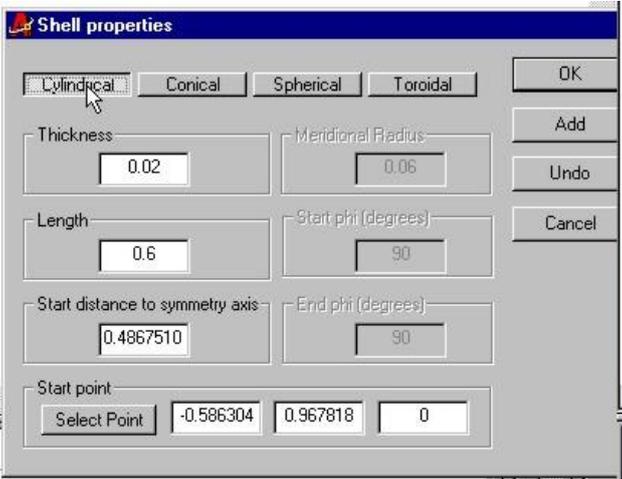


Fig. 13. Dialog Box for Constructing the Geometry Model of a Vessel

The command **MeshShell** and associated dialog box, described previously, discretize the geometry model in Fig. 11 to generate a broad mesh shown in Fig. 14a. Adaptive procedures (Franco 2009 and Franco 1997) and error estimation (Franco 1997 and Zienkiewicz 1987) produce refine finite element meshes for the limit analysis of a ASME Torispherical Pressure Vessel. The results are shown in Figs. 14.

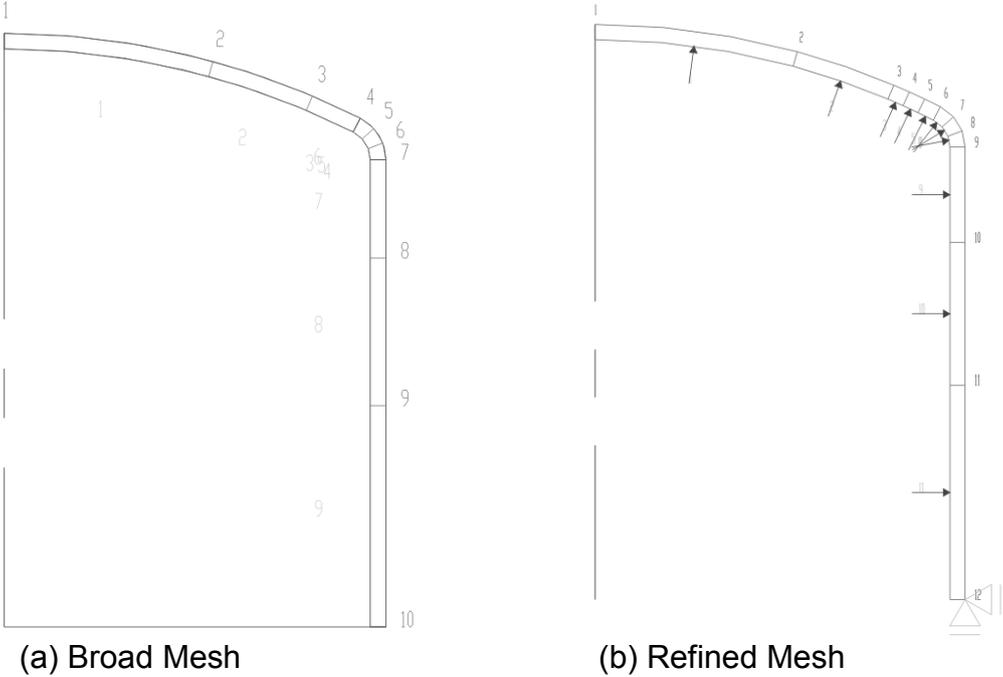
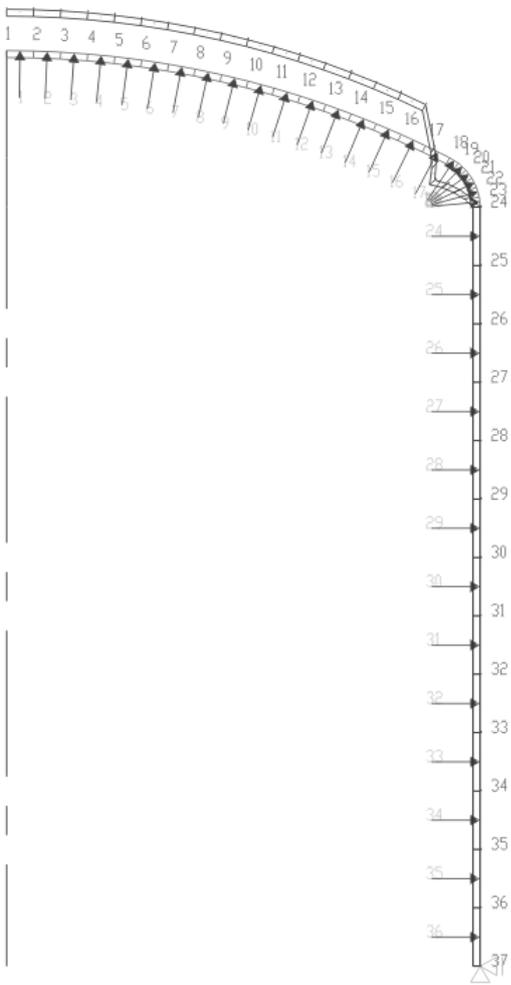


Fig. 14 - Discretization of the ASME Pressure Vessel

The commands **CdTcLim IntPressure** and **ConstraintMesh** apply an internal pressure and impose restrictions to the bottom node of the ASME vessel and the results are illustrated in Figs. 14b and 15. An optimized mesh of the finite element model, the applied internal pressure load and end conditions are displayed as a plane cut in Fig. 15. The BIM application can also post-process the results graphically to show the collapse mechanism by ordering the following commands: **CdTcLimCollapseMechanism** produces a plane representation of the collapse mechanism illustrated in Fig.15. The command **CdTcLimDeformSolid** generates a solid view of the ASME vessel collapse mechanism. The 3D representation of such mechanism is shown in Fig. 16.



(a) 2D Representation

```

Example.dat
3
TRSPHERICAL
6,9,12
12,5
SPHERICAL
NO
0.266587
0.133293
0.0222155
0.0222155
0.0222155
5
0.99,0.99
0
YES
0.02
TOROIDAL
YES
0.018326
3
0.05,-0.426751
27
YES
0.02
CYLINDRICAL
NO
0.126316
0.189474
0.284211
3
0.476751
90
YES
0.02
0.3
1
2
TF
BOCL
TFBC
0.102556,-0.233052,0.529308,0.604494
0.

```

(b) Data File

Fig. 15 - BIM Geometrical and Finite Element Model and Plane Representation of the Collapse Mechanism



Fig. 16. - 3D Collapse Mechanism of the ASME vessel

The BIM model depicted in Figure 15a stores geometric data of the pressure vessel, attributes and information about the finite element model, loading data, material data and boundary conditions. This data set is the basis of the input data required to run limit analysis of pressure vessels. Such a set of data can be exported to text files using the command **CdTeLimExportDataFile** in the same reading format (Fig. 15b) of the finite element system for limit analysis.

4.1 Numerical Results

A finite element technique to calculate limit load on pressure vessels was implemented in a BIM object-oriented programming system. An error estimator and adaptive strategies developed based on the energy norm were also incorporated to the BIM system to obtain optimized solutions.

The BIM system is now applied in the analysis of the ASME shell described in Section 4. Firstly, a coarse mesh is used to obtain two sequences of results for improved nested meshes. One sequence is the result of a uniform refinement and the other is obtained by applying the adaptive strategy referred above. Tables and graphs are then produced using data extracted from each sequence to show the efficiency of the error estimator. Basically the error estimator is evaluated by the effectivity index Θ , which are depicted in the Fig. 17 showing its convergence to the unity as expected. Table 1 shows the data base originated from the sequence of uniform meshes, which was used to calculate the norms of the error estimator $\|e_{\epsilon}^*\|$ and the error from the reference solution $\|e_{\epsilon}^r\|$.

Table 1. Uniform Mesh Refinement.

Mesh	NN	h	$\ \dot{\epsilon}^p\ ^2$	$\ \dot{\epsilon}\ ^2$	$\ \bar{\epsilon}^*\ ^2$	P_L/σ_y
0	24	0.06484	1.64851E-04	1.42820E-03	1.09698E-03	6.63028E-04
1	47	0.03242	1.41507E-04	2.92394E-04	1.36768E-04	6.34354E-04
2	93	0.01621	1.45390E-04	1.81889E-04	3.29304E-05	6.36398E-04

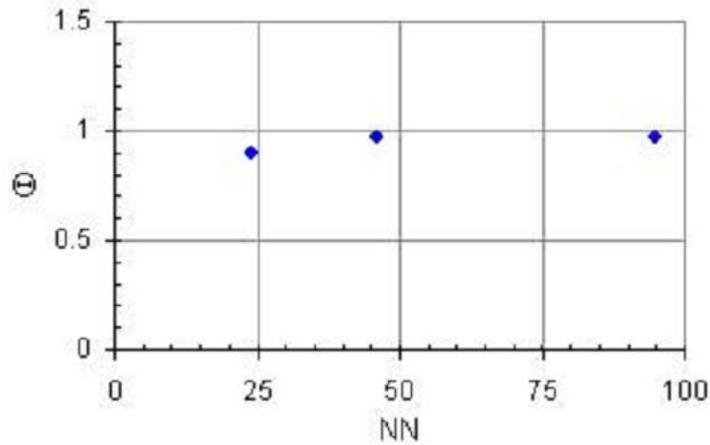
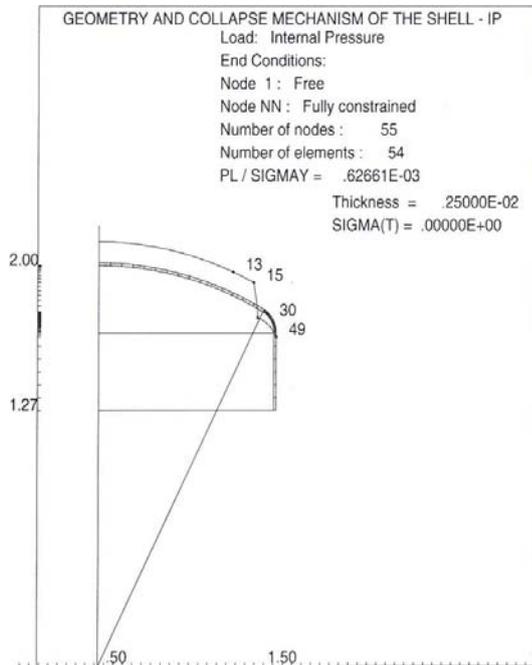
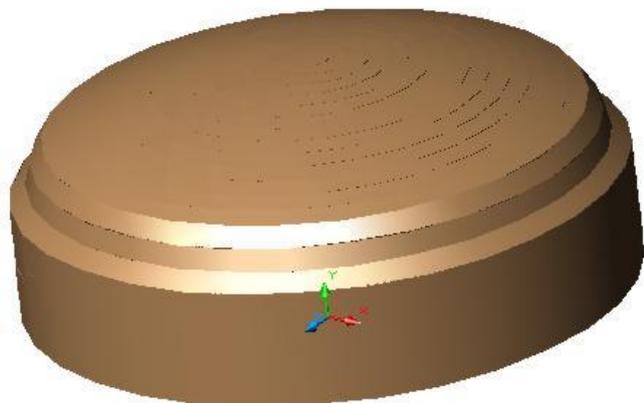


Figure 17. Convergence of the Effectivity Index to Unity

The collapse load and associated collapse mechanism for a refined mesh is shown in Figs. 18.



(a) Collapse Mechanism in 2D



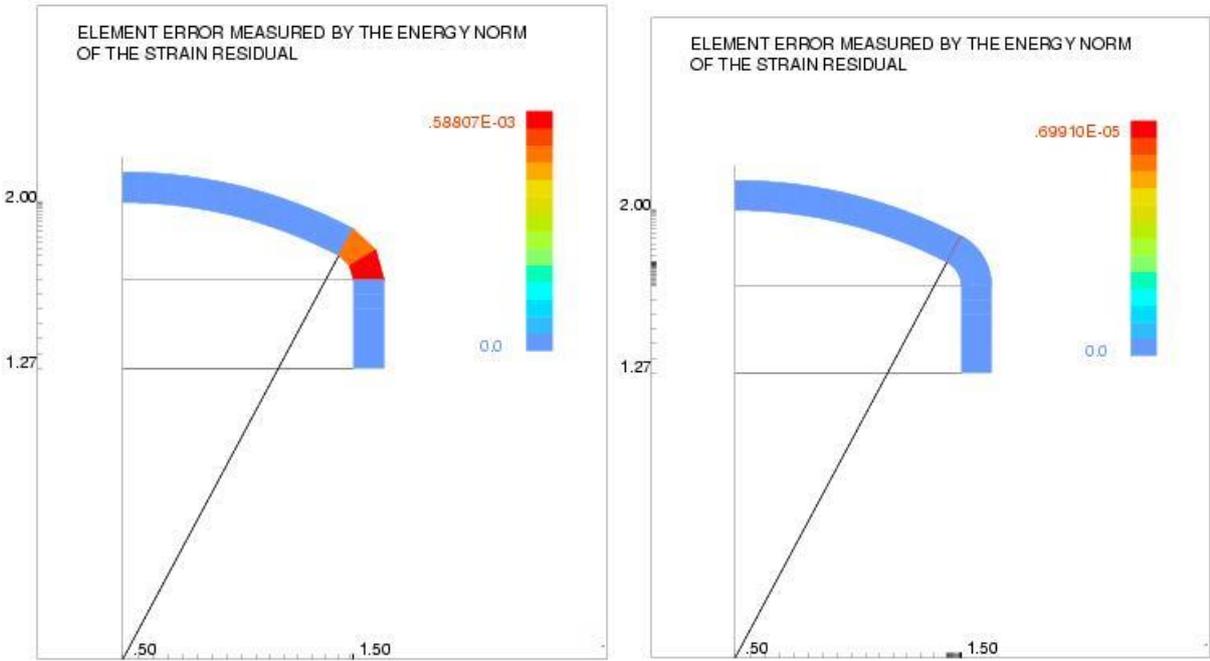
(b) Collapse Mechanism in 3D

Fig. 18. Limit Load and Collapse Mechanism for a Fine Mesh

The values of the parameters for the ASME torispherical shell and the limit load calculated by applying the adaptive procedure to obtain optimized meshes are presented in Table 2. The error distributions for a coarse mesh and for an optimized mesh are illustrated in Figs. 19.

Table 2. Adaptively Refined Meshes

Mesh	NN	$\ \dot{\epsilon}^p\ ^2$	$\ \dot{\epsilon}\ ^2$	$\ \bar{e}_\epsilon^*\ ^2$	P_L/σ_y
0	24	1.64851E-04	1.42820E-03	1.09698E-03	6.63028E-04
1	45	1.43365E-04	1.49459E-04	8.45073E-06	6.30268E-04



(a) Coarse mesh (b) Adaptively refined mesh
 Fig. 19. Elemental Error Distribution - ASME Torispherical Shell

6. CONCLUSIONS

Advances in a finite element formulation for the limit load problem of pressure vessels were implemented in a BIM object-oriented programming system through three separate modules defined as *modeler or pre-processor, the processor or code analysis and the post-processor*. An error estimator and adaptive strategies developed in Franco (1997), based on the energy norm as proposed by Zienkiewicz (1987) are used to optimize the solutions.

In this paper we considered only a specific issue of BIM concepts applied to a finite element formulation for limit analysis of pressure vessels. The paper main goal is to present the development of a BIM application for automating data management of the various information models of the limit analysis process. Concepts of BIM technologies

have been applied to integrate data generated by the system modules. The BIM system facilitates the integration, exchangeability, interoperability and reusability of information in all phases of the process of limit analysis of pressure vessels starting from its geometrical modeling till the digital representation of the results.

The open architecture of AutoCAD, allowed the construction of the BIM application by means of the API ObjectARX, a programming environment for using, customizing and extending AutoCAD. This work presents a first version of a BIM application used here to produce geometrical models, finite element models and collapse mechanisms models for pressure vessels, besides the error distributions in the elements. The information model built by the BIM application integrates all necessary data to perform a finite element limit analysis of pressure vessels. The application has an intuitive and friendly visual interface, providing tools for detecting and correcting geometrical inconsistencies and for optimizing meshes based on adaptive procedures. The customized entities, objects and commands can be ordered as extensions of AutoCAD, through command line, toolbars or menu without distinction from AutoCAD built-in functions. The customized entities, objects and commands inherit most of the features and attributes of their native counterpart, once the BIM application has direct access to AutoCAD graphical system, database and native proceedings. The development of BIM technologies provides tools for the automation of engineering processes in general and can also serve as means to improve education in engineering.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the support of the funding agencies CNPq-Conselho Nacional de Desenvolvimento Científico e Tecnológico and FAPEMIG-Fundação de Amparo a Pesquisa do Estado de Minas Gerais.

REFERENCES

- Autodesk, Inc. (2001) "ObjectARX developer's guide manual on line"
Autodesk, Inc. (2001) "ObjectARX reference manual on line"
Cerovsek, Tomo (2010)"A review and outlook for a 'Building Information Model' (BIM): multi-standpoint framework for technological development", Article history
Dubois-P`elerin Y., T. Zimmermann and P. Bomme. (1992) "Object-oriented finite element programming: I. governing principles", *Comput. Meth. Appl. Mech. Eng.*, 98 p. 291–303,.
Dubois-P`elerin Y., T. Zimmermann. Object-oriented finite element programming: III. an efficient implementation in c++. *Comput. Meth. Appl. Mech. Eng.*, 108 p. 165–183, 1993.
Ellis B.A (2006), "Building Information Modeling: An Informational Tool for Stakeholders"
Franco J.R.Q. and Ponter A.R.S. (1997) A general technique for the finite element shakedown and limit analysis of axisymmetrical shells - part 1 - theory and fundamental relations. *Int. J. Numer. Meth. Eng.*, 40 p. 3495–3514,.

- Franco J.R.Q. and Ponter A.R.S. (1997) A general technique for the finite element shakedown and limit analysis of axisymmetrical shells - part 2 - numerical algorithm. *International J. Num. Meth. in Eng.*, 40 p. 3515–3536.
- Franco, J.R.Q., Oden, J.T., Ponter, A.R.S. and Barros, (1997) F.B. "A posteriori error estimator and adaptive procedures for computation of shakedown and limit loads on pressure vessels", *Comput. Meth. Appl. Mech. Eng.*, 150 p. 155–171,.
- Isikdag, U. and Underwood, J. (2010) "Two design patterns for facilitating Building Information Model-based synchronous collaboration", *Automation in Construction*, 19, 544–553
- Meyer, B. (1997). *Objected-Oriented Software Construction*, 2nd Edition. Prentice Hall.
- Nam-Hyuk Ham, Kyung-Min Min, Ju-Hyung Kim, Yoon-Sun Lee, Jae-Jun Kim (2008), "A Study on Application of BIM(Building Information Modeling) to Pre-design in Construction Project", *Third 2008 International Conference on Convergence and Hybrid Information Technology*, 42-49.
- Ning Gu and Kerry London (2010), "Understanding and facilitating BIM adoption in the AEC industry", *Automation in Construction*, 19, 988–999.
- Popov V., Mikalauskas S., Migilinskas D. and Vainiunas P., (2006) "Complex Usage of 4D Information Modelling Concept for Building Design, Estimation, Scheduling and Determination of Effective Variant, Technological and Economic Development of Economy" 12, (2), 91–98.
- Zienkiewicz, O. C. and Zhu, J. Z. (1987) "Estimator and adaptive procedure for practical engineering analysis" *Int. J. Numer. Meth. Eng.*, 24 p. 337–357,