3-D GEOMETRIC MODELING FOR FINITE ELEMENT ANALYSIS USING MATLAB® C PROGRAMMING

Eyere.E and *Oluwole.O

Mechanical Engineering Department, University of Ibadan

Corresponding author: oluwoleo2@asme.org

ABSTRACT

Most Finite Element Analysts in Nigeria develop own codes for their analysis due to inability to purchase commercial finite element analysis software packages that come with GUI geometric modelers and difficulty in managing coordinate data from imported CAD files. General purpose computer-aided design software packages are good drafting tools but do not have imbedded facilities to carry out finite element analyses of descretized (mesh) models.

This paper demonstrates how MATLAB® programming language can be used to generate 3-D geometric models suitable for finite element analysis. In this work, model of a water storage tanker was drawn using AutoCAD®, and then the same model was generated using MATLAB®. The results showed that MATLAB® can be used to obtain meshed and surface model suitable for finite element analysis.

Keywords: Geometric modeling; MATLAB®; Finite Element Analysis

1.0 Introduction

According to Etas and Jones (1996), the engineering design process is a multi-step process which includes: the research, conceptualization, feasibility assessment, establishing design requirements, preliminary design, detailed design, production planning and tool design, and finally production. According to him, while the preliminary design focuses on the framework, the detailed design portion of the engineering design process is the task where the engineer can completely describe a product through solid modeling and drawings. Solid modeling simply is a mathematical technique for representing solid objects. Through solid modeling, designs are communicated, analyzed, explored and published (Wikipedia 1 and 2, 2012).

In the development of an advanced engineering system, engineers have to go through a very rigorous process of modeling, simulation, visualization, analysis, designing, prototyping, testing, and finally, fabrication/construction (Lui and Quek 2003). For this purpose, trial and error experimental procedure could be reliable but they are inefficient, relatively expensive and tedious to carry out. A better approach is to develop a mathematical model (set of equations describing the designed process and component) based on appropriate assumptions and scientific laws, equations governing the process, and then carry out numerical simulation.

Numerical simulation involves the use of numerical technique and computer to evaluate a mathematical model. There are a number of numerical methods; some common ones are the finite element method, the finite difference method, the finite volume method, Runge-Kutta, and Gauss-Seidel. In the past few decades, the Finite Element Method (FEM) has been developed into a key method in the modeling and simulation of various engineering systems (Lui and Quek, 2003).

According to Richard (1999), The Finite Element Method is a numerical technique ideally suited to digital computers in which a continuum is divided into smaller but finite well defined sub-structures that can be represented by simple equations. On applying the Finite element method, after deciding how best geometry can be simplified. The next issue an analyst needs to consider is what kind of element should be used. This requires a good understanding of the mechanics of the problem.

The process of breaking down the structure into simpler smaller well defined element is termed meshing. With developments in computer hardware and software, a FEM analysis can now be performed very easily. However, many structures are now designed using Computer Aided Design (CAD) packages. Most commercial preprocessors of FEM software come with GUI for geometric modeling and some packages can read certain formats of CAD files. However, many of the software are really expensive and are not available. Common software available with GUIs has ability for 2-D geometry modeling. However, if the designer can write his own codes, MATLAB programming can be used in geometry modeling after which can be used for the finite element analysis. MATLAB is a computer language as well as an interactive computing environment that enables numerical computation, analysis and data visualization (Patrick and Thomas, 2003).

This work demonstrates how MATLAB® variety of plotting functions, powerful computational tools and its other capabilities can be explored for 3-D geometry modeling and how the mesh can be use for the finite element analysis.

2.0 Methodology

2.1 Generation of Model using AutoCAD

A geometrical model of a water tank was generated using AUTOCAD®. The procedure is as follows:

1. Polyline command was used to the draw an outline of the tanker model.
2. The offset command was used to duplicate the line at a distance of 0.15m equivalent to the tank thickness. both lines were then joined
3. The revolved command was then used revolve the outline 360^0 to form the tank geometry.

2.2 Generation of Model using MATLAB

The model was then generated using MATLAB codes. The algorithm is stated as follows:

i. Sequential number (points) within the limit of tanker radius are generated for the X-axis using 'start:increment:end' command.

ii. Corresponding values for the Y-axis are then generated using the equation of a circle $Y = \sqrt{r^2 - \frac{X^2}{1}}$ where r is the radius of the tanker model.

iii. Sequential points are then generated for the Z-axis respectively. the points are equal in size to the x-array and all element in its array is equal to the tank elevation being considered.

iv. The variables are then plotted using suitable MATLAB command.

v. Using the 'hold on' command other section of the tanker are reproduced reaping I and ii above but with a different Z-axis array for the new elevation.

vi. Other effects are then added to the plot using the respective commands. Such effect includes color, lighting effect, view, grid lines, labels and text.

The flowchart for the MATLAB algorithm is shown in Fig 2.1 below

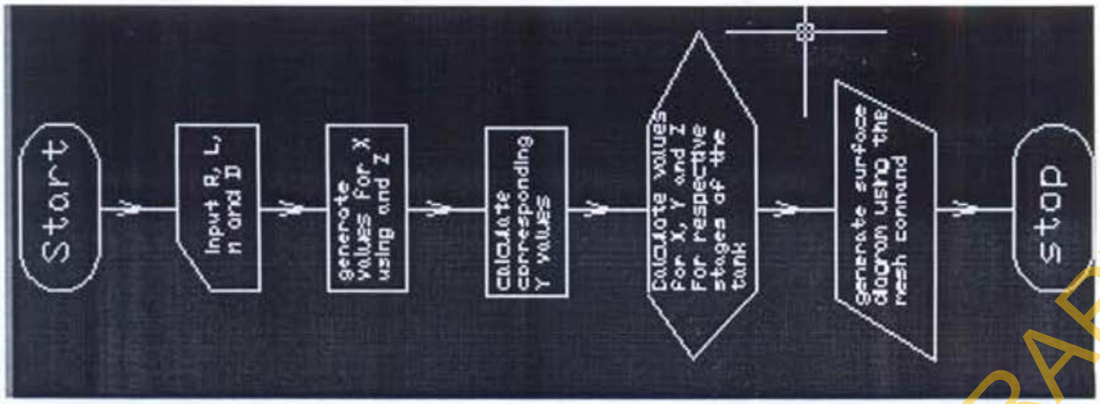

Fig2.1: flowchart for tank generation

The code was used to generate the tanker model. The meshes for the model, was refined continuously until a fine smoothly curved model was obtained. The model properties such as lines and shading were added to give the model a realistic look.

2.3 Extraction of Mesh Nodes for Finite Element Analysis

Each node in the mesh is a point corresponding to specific element of the variable say X(i), Y(i), and Z(i) in the X, Y and Z matrix respectively used to generate the geometric model. It is important to note that 'X' array contains arbitrary sequence of numbers between limits of the cylinder diameter generated using suitable MATLAB command. 'Y' array is a function of the 'X' array; related by the equation of a circle. 'Z' array is an array of size 'X' with all elements equal to the corresponding elevation. For finite element analysis, the matrices are read directly into own code.

3.0 Results and Discussions

3.1 Results

The figures below shows the tanker models generated using AutoCAD, Fig 2a shows the line drawn with the 'polyline' command, fig 2b shows the line duplicated with the 'offset' command and fig 2c shows the tanker model obtained using the 'revolve' command. Fig 2d is the tanker model shown at a S-E isometric view.

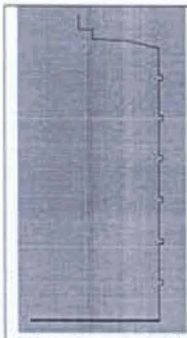

Fig. 2a: polyline trace

Fig.2b:offset by 0.15mm

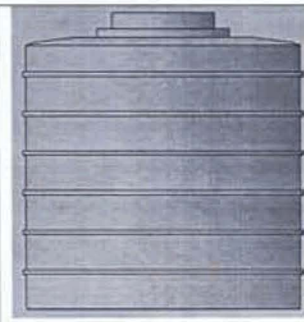

Fig.2c: front view of Tank

Fig.2d: NE Isometry of 3-DTank

Figures 3a-e show the tank geometry generated by MATLAB codes. Figures 3a-c shows three stages of the mesh refining to obtain a fine smoothly curved model as in Figure 3c. Lines were Figure 3d, and then special effect such as color and shadings were added to give it a more realistic looks (Fig3e).

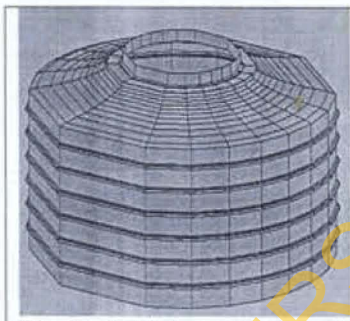

Fig.3a: Initial meshed model

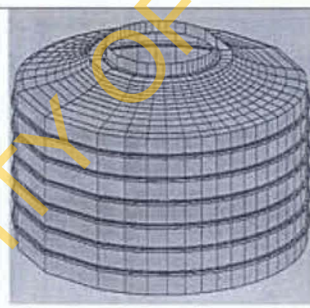

Fig.3b: more refined meshed

Fig.3c: Well meshed tank geometry

Fig.3d surface model

Fig.3e modified surface model

3.2 Discussion of Results

Figures 3a, b and c showed that the more refined the mesh, the better the curve obtained. The model, Figure 3a shows that the circumference of the model is not circular but polygonal. As the model is further refined in figures 3b and c respectively the model becomes more circular around its circumference. Also, just like other CAD software packages special effect can be added as to give the model a more realistic view.

4. Conclusion

It has been demonstrated that 3-D geometry modeling with meshing can be done using MATLAB® when the designer wants to code his finite element problem and also in the absence of a finite element software equipped with a geometric modeler. The mesh points array X(i), Y(i), Z(i) which have been derived by a mathematical relationship between the three coordinates become easy to read into the finite element program codes. On the other hand, this is not feasible with commercial CAD software packages where the solid model is generated by commands and matrices used to generate the model cannot be managed when using own codes because even if the solid geometry is meshed with a command, the array points will be difficult to pick-up for a finite element pre-processor input.

Reference

Ertas, A. and Jones, J. 1996. *The Engineering Design Process*. 2nd ed. John Wiley & Sons, Inc., New York

Liu, G. R. and Quek, S. S. 2003. *The Finite Element Method: A Practical Course*. Butterworth-Heinemann, Oxford

Richard D. B. 1999. *Advanced Stress and Applied Stress Analysis*. 2nd Edition, McGraw-Hill company Inc.London

Wikipedia 1. 2012. Solid modeling Retrieved 18th April, 2012 from http://en.wikipedia.org/wiki/Solid_modeling

Wikipedia 2. 2012. Engineering designed process retrieved 18th April 2012 from http://en.wikipedia.org/wiki/Engineering_design_process