Online 3D Solid FEM Program
Utilizing a HPC Cluster

Kurt Gramoll[1] and Zachary Vick[1]

[1]University of Oklahoma, Norman, OK, USA

Abstract— **In an effort to facilitate the stress analysis of large 3D structures using finite elements in undergraduate solid mechanics courses, a remote, high-performance computing (HPC) cluster for solving was investigated. An online, interactive engineering tool was developed to explore the potential for remote versus local computing of complex geometry stress analysis using FEM. In this engineering tool, 3D models in a COLLADA (COLLAborative Design Activity) [1] (.DAE format) can be uploaded from local memory or from a pre-designed geometry library. The 3D solids tool then volumetrically meshes the geometry using tetrahedral elements to a user defined level of refinement. Loads and boundary conditions can be applied to the model, and the program solves for nodal displacements and stresses.**

It was found that the use of the 3D solids tool has many advantages in engineering education to traditional, locally installed FEM software packages. The engineering tool is accessible from any machine with internet access and requires no cost to use. The webpage-based nature of the 3D solids tool allows it to be integrated seamlessly into the course material for a basic mechanics of materials class. Namely, it has been included as part of the electronic book (eBook) Multimedia Engineering Mechanics of Materials [2] for use in demonstrating 3D elasticity and stress fields. The tool was also used for the basic solid mechanics course at the University of Oklahoma.

The use of a remote HPC cluster in solving computationally intensive problems was found to have numerous advantages to local computation. The HPC cluster has much greater computing power than most personal machines, greatly reducing the computational time. In this case, the computational time no longer depends on the processing power of the local machine as the most complex calculations are performed remotely. This opens up the possibility of performing FEM simulations with many degrees of freedom from less powerful hardware, including laptops, tablets, and smartphones. The only requirements are Internet accessibility and hardware support of Flash.

Index Terms—**FEM, Mechanics, High Performance Computing (HPC), Beam, Web-based**

I. PURPOSE OF THE 3D SOLIDS TOOL

Coursework in mechanics of materials (i.e. solid mechanics or strength of materials) is a key course in most Civil, Aerospace, and Mechanical undergraduate engineering curriculum. The concepts of stress, strain, and deformation are fundamental in a student's ability to design, improve, and/or predict failure conditions for structural and mechanical systems. Traditionally, a solid mechanics class introduces these concepts as applied to extremely fundamental cases. For example, most classes present beam theory to predict bending stress, shear stress, and deflection in a cantilevered or simply supported beam when subjected to idealized loading scenarios. Similarly, students will learn to determine the principal stresses from a provided stress state and compare these values to simple failure criteria. Due to the mathematical complexities of analyzing or transforming the 3D stress tensor, almost all examples seen by students are 2D approximations.

The average engineering student may complete a first course in solid mechanics with little conceptual understanding of a practical, 3D stress state and how that stress state is a direct result of loading seen on a mechanical system. Unfortunately, a student's only visualization of stress and strain may be a Mohr's Circle diagram or basic 2D stress plots of a static, single example. Providing interactive media which allows the student to visualize stress, strain, and deformation is one means to facilitate comprehension. By imposing the requirement that such media be interactive and widely accessible, and by noticing the average university student's almost continuous usage of portable electronic devices (i.e. smartphones, tablets, laptops, etc.), a webpage-based engineering tool represents one obvious vehicle to encourage student utilization of the media.

Electronic media has become extremely popular in engineering education where complex and abstract mathematical models can most be effectively visualized through the use of interactive media such as animations, graphics, and simulations [3]. As a result, many educators incorporate electronic media in their teaching. An increasingly popular form of electronic, internet-based media is the eBook. Here, eBook refers to an electronic book consisting of text, images, and other interactive media, specifically not a scanned or digitally reproduced copy of a print textbook. The benefits of an eBook as the primary or supplemental text for education are numerous. One major advantage of an eBook over traditional print material is that interactive media can be easily and cleanly incorporated with the content to be accessed by students as new material is presented. Interactive eBooks enable engineering educators to present course materials, concepts, and comprehensive visualizations to students in a compelling manner that can enhance education [4]. eBooks for many engineering courses are now readily available for use online, including Statics [5], Dynamics [6], Thermodynamics [7, 8] Solid Mechanics [3] and Fluid Mechanics [9]. Additionally, thanks to the work of Morales and others, eBooks may soon be easily accessible from a variety of devices including PCs, laptops, tablets, and mobile devices [4, 10]. In many cases the eBooks are less expensive than similar print version, or even free [2].

The primary goal of this work was to include a robust, FEM stress analysis 3D solids tool within the appropriate content of a solid mechanics eBook. Providing such a tool serves to enhance education through various functions. First, the tool allows students to visualize deflections and stress fields as applied to 3D components. Secondly, the tool serves to introduce students to the basic concepts of the finite element method and forces the student to recognize all of the major steps taken in the FEM process namely, pre-processing (discretization of the volume, application of boundary conditions and loading), processing (solving the simulation for nodal displacements), and post-processing (calculation of stress and strain from displacements).

The 3D solids tool is hosted on dedicated HPC cluster for engineering education at the University of Oklahoma and has been made available to use freely by anyone with an Internet connection. The tool is browser based and requires no program installation other than the Flash player. This make the tool easily accessible to educators wishing to demonstrate the concepts mentioned previously and prompt in-class discussion of the course material. Furthermore, it gives students access to the tool from any location. Many universities provide students with access to locally installed FEM packages through university managed computers. However, the developed engineering tool can be utilized from almost any machine, personal or public. The available option to perform all calculations on a remote HPC cluster allows less powerful devices to act as an interface for setting boundary conditions and viewing results without performing the computationally intensive calculations locally. Additionally, the use of the HPC cluster bypasses a timeout restraint imposed by Flash which will disallow any local calculations taking longer than 60 seconds to perform. (Note that in most cases, Flash will disallow any calculations requiring more than 15 seconds to perform. However, compiler options can be changed to lengthen this time to a maximum of 60 seconds.)

The 3D solids tool allows the user to import geometries in the COLLADA format, an open standard XML schema for exchanging digital assets among various graphic software applications [1]. The COLLADA file format was selected as it is one of the most widespread and popular file formats among free 3D modeling software packages, including Google SketchUp and Blender. When paired with one of these modeling programs, the 3D solids tool provides a costless alternative to commercially available, expensive, locally-installed FEM packages. It should be noted, however, the current version of the tool is limited to well posed structures due to automatic gridding difficulties. This means, complex objects with holes and convex surfaces may have gridding difficulties. New versions of the tool are being developed to overcome these issues. It was surprising to the authors the level of difficulty in developing robust automatic gridding routines.

Several free FEM applications have been created in the past, but there is a lack of web-based FEM tools created using Flash or HTML5 or FEM tools integrated into course material readily accessible to students. There are numerous free, open source FEM post- and pre-processors available for download and install. However, these examples require a local installation of the software and do not utilize a remote HPC cluster. Examples include CalculiX [11], Z88 Aurora [12], and Object Oriented Finite Element Solver (OOFEM) [13]. The requirement of a local software install limits the use of these programs to Windows or Linux based machines with enough computational power locally to complete the required calculations. Dhondt and Wittig have provided a free online demo of their FEM program CalculiX at http://www.onlinefeasolver.com/demo.php [11]. However this is just an online demo of a program that was written to be locally installed (freely) on a Linux machine. A full version of the Windows equivalent can be purchased under the name bConverged [14].

II. OPERATION OF THE 3D SOLIDS TOOL

A. Beam Example

This 3D solids tool was created using the Flash programming language, ActionScript, because of its advantages in creating a graphical user interface. The compiled form of a Flash project is easily embedded into a webpage, ensuring widespread accessibility. Flash also allows access to scripts on a remote server allowing a compiled executable to be easily accessed on the HPC cluster.

In order to illustrate the operation of the 3D solids tool, a simple problem will be analyzed for deflection using traditional beam theory and the 3D solids tool. The operation and available user functions of the 3D solids tool will be discussed as a sample simulation is presented. The example problem is given as follows. A 20 ft wooden beam (Young's modulus 1,300 ksi, Poisson's ratio 0.33) with a 1 ft by 1 ft cross-section is fixed on one end. A uniform load of 50 lb/ft is distributed across the length of the beam, as shown in Figure 1.
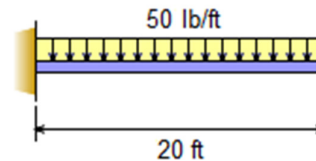

Figure 1: Sample problem description

It can be shown that the deflection, v, at any x-distance from the fixed end is a function of only the beam length, L, the distributed load, w, the material stiffness, E, and the cross-sectional moment of inertia, I,

$$v = \frac{-wx^2}{24EI}\left(x^2 - 4Lx + 6L^2\right) \qquad (1)$$

For this particular problem, Equation 1 becomes, as given in Equation 2.

$$v = 7.7284 \times 10^{-10}\left(x^4 - 960x^3 + 345{,}600x^2\right) \qquad (2)$$

where x and v are both in inches. The result obtained in Equation 2 is purely a mathematical expression for the deflection at every point in the beam based on elementary beam bending theory. In order to better visualize the

deformation and resulting stresses, the same problem can be analyzed using the 3D solids tool.

At this point, the authors would like to stress that the 3D solids tool is a new tool and still under development. Currently, there are many limitations to the tools use that are being addressed for the later versions. However, great strides have been made in order to progress the tool to this level of refinement. To access the tool, the user merely needs to open and direct a webpage to the solid mechanics eBook accessible from eCourses.ou.edu. The mechanics eBook Multimedia Engineering Mechanics of Materials Chapter 1: Stress and Strain contains a link to the engineering tool in the section covering 3D Hooke's Law [2]. The tool can also be accessed directly at www.eCluster.ou.edu/apps/solid3d. Once the tool appears in the webpage (see Figure 2), the user controls are located at the top of the window.
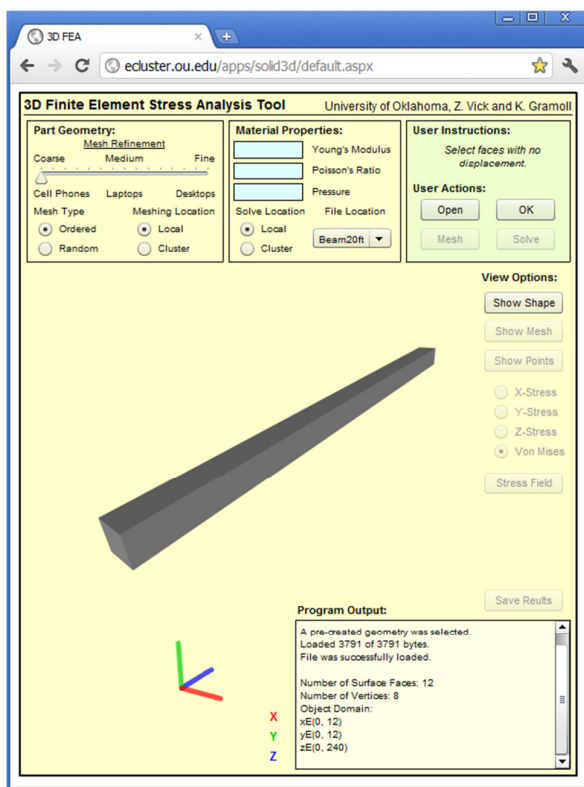

Figure 2: Tool interface with loaded geometry

B. Loading a geometry file

The user can choose to upload a geometry file from the local machine or download a pre-created file from the geometry library. Figure 2 depicts the tool interface once a geometry has been selected and loaded. The pictured object is a 20 ft beam with a 1 ft by 1 ft square cross-section. The geometry was created using the free version of Google SketchUp and has been made available in the geometry library, a pull down menu containing various pre-created geometry files.

Upon loading the geometry, the program reports the object domain to the user in order to verify that the correct dimensions have been loaded. This is done primarily for unit conversion purposes. Currently, it is assumed the user

will keep track of units to ensure they cancel correctly. As can be seen in Figure 2, the object domain is in the unit of inches. Note the Papervision3D library was used for 3D rendering purposes. As a result, all 3D rendering is done in a left-handed coordinate system as is default in the Away3D (previously Papervision3D) library [15].

C. Selecting boundary and loading conditions

Once the geometry is loaded, the user is able to select any face, or multiple faces, which are fixed and any faces which have a uniform pressure applied to them. Fixed faces are fixed in all three directions including all nodes on the selected face(s). Faces selected to have no displacement will be highlighted in red. The user is able to select multiple faces which are fixed or which are loaded.

Loads are applied as a uniform pressure to faces. The user only needs to input the pressure acting on the loaded face(s). The tool will calculate the loaded area and apply forces to nodes as required to approximate the uniform pressure acting on the selected face(s). All pressures act normal to the selected surface with positive values acting away from the object interior. Faces subjected to loading will be highlighted in green. While the current version restricts boundary conditions to fixed or uniform pressure, it is planned to include additional conditions in the future.

The user is able to select faces by clicking on them. The 3D solids tool allows the user to interact with the object using the mouse similar to other 3D modeling programs with options to rotate the object about the origin, pan the object, and change the camera zoom. In the case of the sample problem, one end face was selected to have no displacement and the top face was selected to have uniform pressure acting on it. The pressure was set to 0.34722 psi (equal to 50 lb/ft acting over 20 ft).

D. Meshing the object

After selecting the fixed and loaded faces, the user must then specify a level of mesh refinement and generate a mesh for the object. The user can chose to generate an ordered or random mesh depending on the geometry. In general, the ordered mesh is more efficient for rectangular objects such as beams or plates. The ordered meshing routine discretizes the volume inhabited by the geometry into rectangular prisms. Each rectangular prism can then be divided into six tetrahedral [16]. Any tetrahedron that is found to be outside the geometry is deleted.
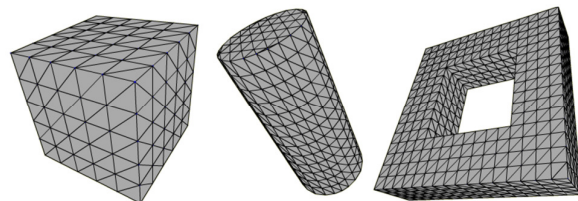

Figure 3: Three geometries meshed with the ordered meshing routine

The random mesh is more suited for irregular geometries. The random meshing routine will generate random points in the object interior and on the object surface. These points must be a minimum distance from all other generated points that depends on the geometry size and the mesh refinement. Using the generated points, a 3D Delaunay tetrahedralization is created using a simple

incremental method [17]. The surface points are added first, followed by the interior points. This is done to make sure that the overall object surface remains as close to the original as possible. Figure 4 shows two meshes generated by the random meshing routine.
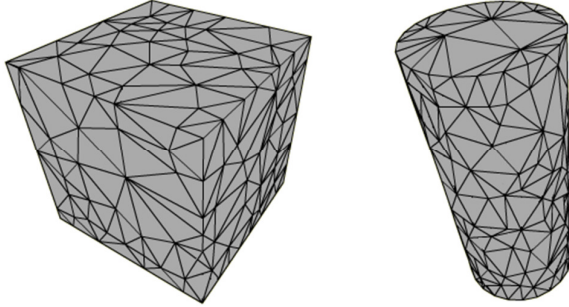

Figure 4: Two geometries meshed with
the random meshing routine

The user may also choose to run the meshing routine on the local machine or using the HPC cluster. Generating a random mesh for high mesh refinements often requires using the HPC cluster due to the local 60 second time restraint mentioned earlier. Similarly, generating an ordered or random mesh on complex geometries will also require using the HPC cluster.

Once the mesh is generated, a series of view options will become available to the user so that the mesh can be inspected. The user can choose to view any combination of the surface mesh, original object, and the generated nodal locations. This allows students and educators to inspect the mesh and can be used to demonstrate the importance of quality mesh generation. If the mesh is not satisfactory to the user, the user can then re-generate a mesh with a higher level of refinement. In the case of the sample problem, a relatively coarse, ordered mesh was generated.

E. Inputting material properties and solving the simulation

The user will be prompted to input the material stiffness, Poisson's ratio, and loading pressure (in force per unit area units). Once this information is entered, the simulation can be run. The user can choose to solve the simulation locally or using the HPC cluster. It will often be required to use the cluster for any simulation containing more than a few thousand degrees of freedom as this cannot be solved locally within the 60 second time limit.

The tetrahedral stress element was selected as the 3D element for use in the FEM solver. The tetrahedral element was selected for two primary reasons. The tetrahedral element has distinct advantages in mesh generation using 3D Delaunay tetrahedralization and orderly discretization, and the tetrahedron is a linear, constant-strain element which allows for closed form calculation of the elemental stiffness matrix without requiring numerical integration. It can be shown that the linear shape functions, N_1, N_2, N_3, and N_4 take the following form [18].

$$N_1 = \frac{\alpha_1 + \beta_1 x + \gamma_1 y + \delta_1 z}{6V}$$

$$N_2 = \frac{\alpha_2 + \beta_2 x + \gamma_2 y + \delta_2 z}{6V}$$

$$N_3 = \frac{\alpha_3 + \beta_3 x + \gamma_3 y + \delta_3 z}{6V}$$

$$N_4 = \frac{\alpha_4 + \beta_4 x + \gamma_4 y + \delta_4 z}{6V} \tag{3}$$

Here, the shape function coefficients α_i, β_i, γ_i, and δ_i are determined from the displacement function and are dependent on nodal coordinates [18]. The shape function matrix, [N], is given by

$$[N] = \begin{bmatrix} N_1 & 0 & 0 & N_2 & 0 & 0 & N_3 & 0 & 0 & N_4 & 0 & 0 \\ 0 & N_1 & 0 & 0 & N_2 & 0 & 0 & N_3 & 0 & 0 & N_4 & 0 \\ 0 & 0 & N_1 & 0 & 0 & N_2 & 0 & 0 & N_3 & 0 & 0 & N_4 \end{bmatrix} \tag{4}$$

It can then be shown that the element stiffness matrix, [k], is defined without numerical integration and takes the following form [18].

$$[k] = [B]^T [D][B] \tag{5}$$

Here, [D] is material constitutive matrix defined by the material Young's modulus, E, and Poisson's ratio, v.

$$[D] = \left(\frac{E}{(1+v)(1-2v)} \right)$$
$$\begin{bmatrix} 1-v & v & v & 0 & 0 & 0 \\ & 1-v & v & 0 & 0 & 0 \\ & & 1-v & 0 & 0 & 0 \\ & & & \frac{1-2v}{2} & 0 & 0 \\ & Symmetry & & & \frac{1-2v}{2} & 0 \\ & & & & & \frac{1-2v}{2} \end{bmatrix} \tag{6}$$

And the [B] matrix is defined as

$$[B] = \begin{bmatrix} \beta_1 & 0 & 0 & \beta_2 & 0 & 0 & \beta_3 & 0 & 0 & \beta_4 & 0 & 0 \\ 0 & \gamma_1 & 0 & 0 & \gamma_2 & 0 & 0 & \gamma_3 & 0 & 0 & \gamma_4 & 0 \\ 0 & 0 & \delta_1 & 0 & 0 & \delta_2 & 0 & 0 & \delta_3 & 0 & 0 & \delta_4 \\ \gamma_1 & \beta_1 & 0 & \gamma_2 & \beta_2 & 0 & \gamma_3 & \beta_3 & 0 & \gamma_4 & \beta_4 & 0 \\ 0 & \delta_1 & \gamma_1 & 0 & \delta_2 & \gamma_2 & 0 & \delta_3 & \gamma_3 & 0 & \delta_4 & \gamma_4 \\ \delta_1 & 0 & \beta_1 & \delta_2 & 0 & \beta_2 & \delta_3 & 0 & \beta_3 & \delta_4 & 0 & \beta_4 \end{bmatrix} \tag{7}$$

The element stiffness matrices can then be combined to create the global stiffness matrix, [K]. The global stiffness matrix is then inverted and multiplied by the forces

applied to each node, {F}, to find the nodal displacements, {d}.

$$\{F\} = [K]\{d\} \qquad (8)$$

However, due to the positive definite, banded form of the global stiffness matrix, the entire matrix does not need to be explicitly formed and saved in memory. Instead, only the non-zero terms of the upper triangle can be stored in memory using the modified Cholesky decomposition [19]. This technique greatly reduces the required memory for storing the global stiffness matrix. This method is employed by the 3D solids tool, and the inversion of global stiffness matrix is performed on this banded submatrix.

In the case of the sample problem, the stiffness was set to 1,300 ksi. The Poisson's ratio was set to 0.33, and the uniform pressure was set to -3.4722×10^{-4} ksi, signifying that the load is directed towards the beam. The FEM simulation was then run, finding displacements at every nodal location and the elemental strains and stresses.

F. Interacting with the results

Once the simulation has been solved, various options will become available to view and interact with the results. The user can choose to view the original object geometry, the deformed surface mesh, the elemental stresses in the x, y, and z-directions, the elemental von Mises stress invariant, and the displaced nodal locations scaled by two orders of magnitude. The user can select any combination of these options to display in the view field.

By default, the deformed surface mesh will be displayed on top of the original geometry. The deformed beam shape in the case of the sample problem is shown in Figure 5. The deformation has been scaled by two orders of magnitude to make the deflection more visible to the user. The deformed mesh and the original object geometry to help students to visualize deformation and identify locations of high stress concentration. Deformations of simple geometries such as beams can be compared to theory and discrepancies can be useful sources of class discussion.
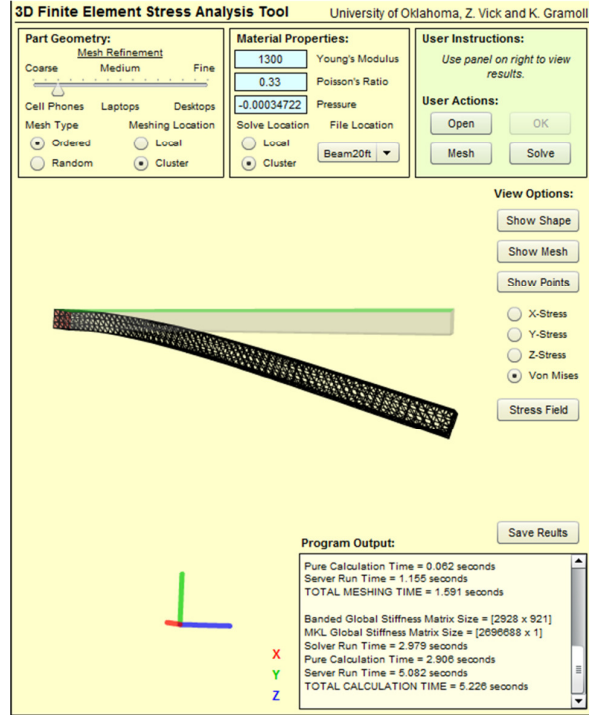

Figure 5: Deformed mesh, deformation scaled by 100

The elemental von Mises stresses can be plotted, as shown in Figure 6 for the fixed boundary section for the sample problem. Note that each colored point represents the stress seen in a tetrahedral element. As was discussed in the solution section, each tetrahedral is a constant-strain, constant-stress element. Each colored point is placed at the centroid of the element as it represents the most accurate stress location available instead of extrapolating elemental stresses to nodal locations. The areas of high stress can be easily compared to a recognizable color scale ranging linearly from green to red. This allows students to visualize how stress varies in all three dimensions as opposed to only surface stresses generally shown in many FEM software packages.
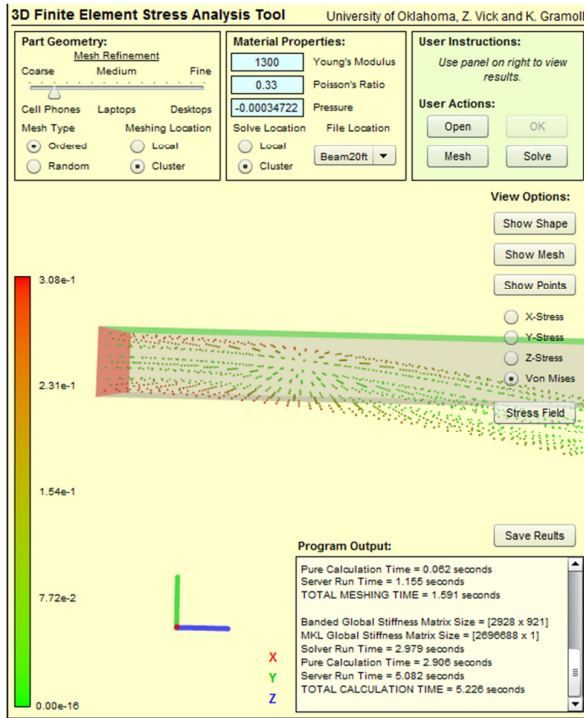
Figure 6: Von Mises elemental stress states

Once the simulation has been solved, the user is able to save the results in the form of a text file to the local machine. The savable results include the original node locations created by the meshing routing, the tetrahedral elements represented by four nodes, the nodal displacements corresponding to each node, the elemental strains and stresses (both normal and shear), and the von Mises stress invariant. The text file had been arranged so that it can be easily delimited and imported into standard spreadsheet software.

III. CLUSTER IMPLEMENTATION

A HPC cluster has greater computing power than most personal machines, which can reduce the computational time required for 3D FEM problems that commonly approach 50,000 degrees of freedom for even simple models used in education. All cluster computations are done on a 48 node (dual CPU blade servers, 12 cores), 576 core cluster (Intel CPUs running under windows 2008 HPC Server R2 system) at the University of Oklahoma dedicated to engineering education. All simulations are Internet-based, and are freely open to others to utilize at their institutions.

Currently, one instance of the engineering tool is run on a single node using all of the twelve cores available to that node. For more information on multicore utilization of the HPC cluster, see the conference paper "Development and Implementation of a High Performance Computer (HPC) Cluster for Engineering Education Simulations [20]." It is possible for multiple users to access the engineering tool simultaneously. The cluster job control will automatically locate and utilize an open node for each new user that accesses the tool.

The input and output parameters from the HPC cluster solving and meshing routines are written to/read from text files. It was found that the data conversion and string manipulation process requires a significant amount of time when transferring data to and from the server. However, this process became less significant for simulations with many degrees of freedom. A simple process map of cluster access is given in Figure 7.
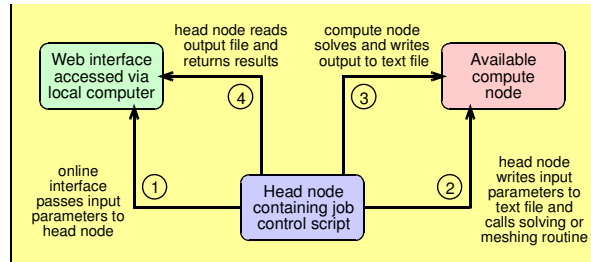

Figure 7: HPC cluster utilization process map

IV. RESULTS

The sample problem presented previously was solved for various mesh refinements both utilizing the HPC cluster and solving the simulation locally. The obtained results were compared to elementary beam theory, and calculation times were compared for locally and remotely performed meshing and solving.

The centerline deflections were found using the 3D solids tool and compared to elementary beam theory given as shown in Figure 8. It was found that, as the number of generated nodes was increased, the engineering tool converged near to the beam theory results with the highest mesh refinement used having an almost uniform 4-5% error (compared to theory) in the predicted displacement.
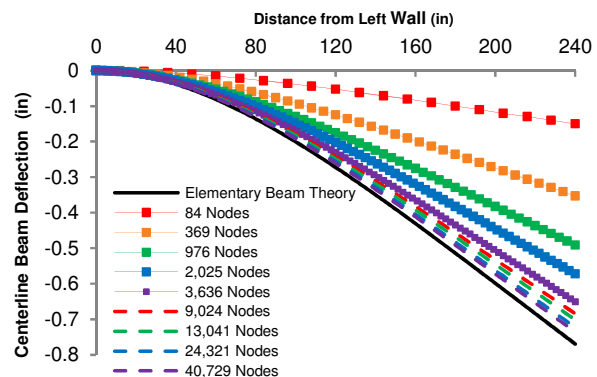

Figure 8: Engineering tool results compared to theory

Similarly, solution run times were compared for meshing and solving when the calculations were performed locally and remotely. All local calculations were performed on an Intel Core2 Duo CPU P8700 at 2.53 GHz and using 4.0 GB of RAM.

The times required to generate an ordered mesh for the 1x1x20 ft beam were measured. The results are summarized in Table 1. Notice that the total time required to mesh the geometry became significantly less for more refined meshes when the ordered meshing was performed

remotely. It is interesting to note that the "calculation time" from the remote routine section covers equivalent processes as the total meshing time locally. For example, for the finest mesh created locally, it required 40.27 seconds to mesh the object. These same calculations were performed remotely in 6.515 seconds. An additional 1.965 seconds were required for string manipulation, reading and writing to text files, and transferring the data to/from the server.

Table 1: Local vs. remote runtimes for ordered meshing

Mesh		Local Time (sec)	Remote Times (sec)			
Nodes	Elements	Total	Total	Calc.	Mesh Runtime	Data Transfer
84	120	0.046	0.671	0.001	0.003	0.668
369	960	0.072	0.682	0.008	0.016	0.666
976	3,240	0.393	0.808	0.066	0.086	0.722
2,025	7,680	1.713	1.138	0.363	0.408	0.730
3,636	15,000	5.857	1.780	1.041	1.094	0.686
5,929	25,920	16.42	3.572	2.729	2.821	0.751
9,024	41,160	40.27	7.263	6.515	6.655	0.608
13,041	61,440	>60	15.26	14.39	14.60	0.655
24,321	120,000	>60	54.16	51.42	51.87	2.289
40,729	207,360	>60	157.7	152.8	153.5	4.183

The times required to solve the FEM simulations for the meshes described in Table 1 were measured. The results are summarized in Table 2. Notice that the total time required to solve the simulation was faster utilizing the HPC cluster, even for relatively coarse mesh generation. It is also interesting to note that more finely generated meshes quickly require more time locally than is allowed by Flash. Therefore, for anything but the coarsest of meshes, the HPC cluster will be required to solve the simulation.

Table 2: Local vs. remote runtimes for FEM solver routine

Stiffness Matrix	Local Time (sec)	Remote Times (sec)			
	Total	Total	Calc.	Solver Runtime	Data Transfer
252 x 195	0.151	0.739	0.094	0.124	0.615
1,107 x 498	5.422	0.934	0.156	0.187	0.747
2,928 x 921	>60	1.223	0.281	0.436	0.787
6,075 x 1,464	>60	2.350	0.686	1.045	1.305
10,908 x 2,127	>60	4.234	1.467	1.996	2.238
17,787 x 2,910	>60	6.711	3.229	4.305	2.406
27,072 x 3,813	>60	12.62	6.895	8.673	3.948
39,123 x 4,836	>60	25.71	15.66	18.38	7.338
72,963 x 7,242	>60	88.64	60.95	66.28	22.36
122,187 x 10,128	>60	241.6	192.5	202.9	38.68

V. CLASSROOM USE

The main purpose of developing the tool was to enable students in a basic solid mechanics engineering course to better understand three dimensional deflection and stresses in basic beams. Of course, the tool can do more than just beam structures, but beam structures fits nicely into the standard first year solid mechanics curriculum. The tool also allowed students to experience finite element solution method for structural analysis without the large learning curve associated with modern FEM tools like ANSYS.

Implementation in the classroom was convenient since the first author teaches solid mechanics at the University of Oklahoma where all learning material (text and problems) are delivered through an online web site, eCourses.ou.edu. The students were required to use the 3D tool to compare the stress state in a cantilevered beam where two sides were loaded with a uniform distributed load, as shown in Figure 9. The results, Figure 10, where then compared to hand calculations to show how FEM can be used for stress analysis. In particular, the deflection of the end point was shown to be less than one percent different between basic beam theory and FEM. However, when the beam length-to-thickness ratio become small, less than 10, the solutions were shown to diverge. This was surprising to many students, even though they were told linear beam theory only applies to slender beams.
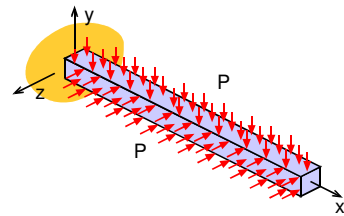

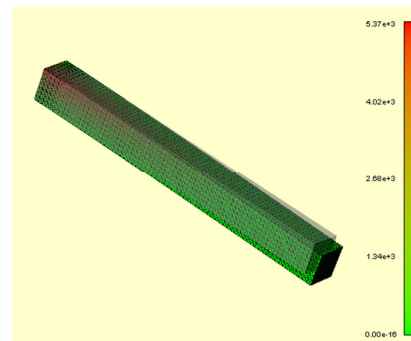Figure 9: Basic Beam Configuration Used in Solid Mechanics Course

Figure 10: Basic Beam Example Stress Result

Other beam configurations were also explored, such as non-rectangular cross sections. One homework problem was also assigned that required the use of the 3D tool to solve the problem. In the future, it is planned to integrate more 3D stress visualization for complex 3D structures into both the classroom and homework problems. While this tool is not a complete FEM analysis tool, it was useful in demonstrating 3D stresses and deflection for common structures in the basic sophomore or junior-level solid mechanics class.

VI. CONCLUSION

A web-page based 3D solids tool was developed to be included in a Multimedia Engineering Solid Mechanics eBook. The engineering tool allows the students and

educators to utilize a HPC cluster to solve FEM problems. The tool volumetrically meshes user defined geometry using tetrahedral elements to a user defined level of refinement. Loads and boundary conditions can be applied to the model, and the program solves for nodal displacements and the stresses seen in every element. The 3D solids tool is accessible from any machine with internet access and requires no cost to use. The use of a remote HPC cluster in solving computationally intensive problems was found to reduce the computational time in most cases and to be necessary for any but the coarsest of meshes. This opens up the possibility of performing FEM simulations with many degrees of freedom from less powerful hardware, including laptops, tablets, and smartphones.

REFERENCES

[1] Arnaud, R., and Barnes, M. C. (2008), COLLADA: Sailing the Gulf of 3D Digital Content Creation. A K Peters, Ltd.

[2] Gramoll, K. Multimedia Engineering Mechanics of Materials. Retrieved from www.ecourses.ou.edu/cgi-bin/ebook.cgi?topic=me

[3] Gramoll, K. (2007, June), "A Web-based Electronic Book (eBook) for Solid Mechanics," 2007 ASEE Annual Conference, Honolulu, HI. DOI: AC 2007-785

[4] Morales, C. (2011, June), "Development of a Multi-Platform (PC, iPad, Mobile) eBook Platform," 2011 ASEE Annual Conference, Vancouver, Canada. DOI: AC 2011-2352

[5] Gramoll, K. (1999, June), "Teaching Statics Online with only Electronic Media on Laptop Computers," 1999 ASEE Annual Conference, Charlotte, NC. DOI: AC 1999-1668

[6] Vikas, Y., Romanello, T., & Gramoll, K. (2000, June), "Teaching Dynamics Online with only Electronic Media on Laptop Computers," 2004 ASEE Annual Conference, St. Louis, MO. DOI: AC 2000-3666

[7] Ngo, C. and Lai, F. (2003). "An Online Thermodynamics Courseware," Computer Applications in Engineering Education, Vol. 11, pp. 75-82

[8] Huang, M. and Gramoll, K. (2004, June), "Online Interactive Multimedia for Engineering Thermodynamics," 2004 ASEE Annual Conference, Salt Lake City, UT. DOI: AC 2004-3166

[9] Gramoll, K. and Ngo, C. (2004, June), "A Web-based Electronic Book (eBook) for Fluid Mechanics," 2004 ASEE Annual Conference, Salt Lake City, UT. DOI: AC 2004-1793

[10] Morales, C. (2011, June), "Implications of Publishing eBooks on PCs and Mobile Devices for Engineering Technology Educators," Paper presented at 2011 ASEE Annual Conference, Vancouver, Canada. DOI: AC 2011-2345

[11] Dhondt, G., & Wittig, K. Calculix: A Free Software Three-Dimensional Structural Finite Element Program. Retrieved from http://www.calculix.de/

[12] Rieg, F. Z88 Aurora. Retrieved from www.z88.de/

[13] Patzák, B. (2000), OOFEM project home page. Retrieved from http://www.oofem.org

[14] Baylor, J. (2011), bConverged. Retrieved from www.bconverged.com/

[15] Winder, J., and Tondeur, P. (2011), Papervision3d Essentials. Birmingham, UK: Packt Pub Ltd

[16] Chandrupatla, T. R., and Belegundu, A. D. (2004), Introduction to Finite Elements in Engineering. Upper Saddle River, NJ: Prentice Hall

[17] George, P., and Borouchaki, H. (1998), Delaunay Triangulation and Meshing, Application To Finite Elements. Paris, France: Kogan Page Ltd.

[18] Logan, D. L. (1992), A First Course in the Finite Element Method. Boston, MA: PWS Publishing Company

[19] Weaver, W., & Gere, J. M. (1990), Matrix Analysis of Framed Structures. Kluwer Academic Publishers

[20] Gramoll, K. (2012, June), "Development and Implementation of a High Performance Computer (HPC) Cluster for Engineering Education Simulations," 2012 ASEE Annual Conference, San Antonio, TX

AUTHORS

Kurt Gramoll is with The University of Oklahoma, School of Aerospace and Mechanical Engineering, Norman, OK 73072 USA (e-mail: gramoll@ou.edu).

Zachary Vick was with The University of Oklahoma, School of Aerospace and Mechanical Engineering, Norman, OK 73072 USA (e-mail: zachary.vick@ou.edu). He is now employed with BP, Inc.