Conf-9110344--8-Rev.1

WSRC-MS--91-230-Rev.1

DE92 013070

COMPUTER APPLICATIONS FOR ENGINEERING/STRUCTURAL ANALYSIS (U)

by

M. Zaslawsky
S. K. Samaddar

Westinghouse Savannah River Company
Savannah River Site
Aiken, SC 29808

A paper proposed for publication in the proceedings of the
1991 Westinghouse Computer Symposium
Monroeville, PA
October 21 - 22, 1991

This information was prepared in connection with work done under Contract No. DE-AC09-89SR18035 with the U. S. Department of Energy. By acceptance of this paper, the publisher and/or recipient acknowledges the U. S. Government's right to retain a nonexclusive, royalty-free license in and to any copyright covering this paper, along with the right to reproduce and to authorize others to reproduce all or part of the copyrighted paper.

COMPUTER APPLICATIONS FOR ENGINEERING/STRUCTURAL ANALYSIS

By:

M. Zaslawsky
S. K. Samaddar

Westinghouse Savannah River Company
Savannah River Site
Aiken, SC 29808

Westinghouse Computer Symposium
Radisson Hotel
Monroeville, PA
October 1991

This information in this article was developed during the course of work under Contract No. DE-AC09-89SR18035 with the U. S. Department of Energy.

COMPUTER APPLICATIONS FOR ENGINEERING/STRUCTURAL ANALYSIS

By:

M. Zaslawsky
S. K. Samaddar

Westinghouse Savannah River Company
Savannah River Site
Trailer 703-24C
Aiken, SC 29808
(803) 557-9070

Abstract

Analysts and organizations have a tendency to lock themselves into specific codes with the obvious consequences of not addressing the real problem and thus reaching the wrong conclusion. This paper discusses the role of the analyst in selecting computer codes. The participation and support of a computation division in modifying the source program, configuration management, and pre- and post-processing of codes are among the subjects discussed. Specific examples illustrating the computer code selection process are described in the following problem areas: soil structure interaction, structural analysis of nuclear reactors, analysis of waste tanks where fluid structure interaction is important, analysis of equipment, structure-structure interaction, analysis of the operation of the superconductor supercollider which includes friction and transient temperature, and 3D analysis of the 10-meter telescope being built in Hawaii. Validation and verification of computer codes and their impact on the selection process are also discussed.

This information in this article was developed during the course of work under Contract No. DE-AC09-89SR18035 with the U. S. Department of Energy.

INTRODUCTION

Computer solutions to engineering problems are now a way of life; from the university to industry to the think tanks, complex problems are addressed using a variety of hardware and software. The computer programs vary from small programs to mainframe programs which address "all" problems. As a consequence, numerous computer programs that solve a whole array of problems have become available. The results of numerous PhD theses are computer programs. A common practice is to obtain an "open" program, make a modification, and then rename the program and classify it as proprietary. Organizations as well as individuals have difficulty in keeping abreast of what programs are available and how appropriate the code is for the problem. Lack of adequate technical manuals and incomplete validation and verification have added to the complexity. As a consequence, individuals and organizations have tended to adopt one or two codes to use for all problems with an emphasis on the word "all". It must be recognized that each computer program has positive and negative attributes, and no single program is the panacea of the analyst.

DISCUSSION

High energy physics laboratories such as FERMI, LBL, BNL, and SSC Dallas Lab have universally adopted ANSYS[1] while others have adopted codes such as ADINA, ABAQUS[2], etc. Still others have written their own programs. ANSYS and ABAQUS have extensive documentation and are validated and verified programs.

ANSYS, like many other programs, is a proprietary program and a fee is charged to use it. In addition, while the manuals are complete, they are also expensive. As a consequence, all too often, it is the program in search of a problem rather than the problem in search of a solution.

An analyst must be able to defend the solution and the assumptions used to solve a problem. Assumptions made to accommodate a computer program are not acceptable. Sometimes arguments are made that the program is the state of the art, but in reality it is the state of the art only for that program. Ignoring friction between components, when in fact friction is important, is not acceptable. New codes without properly written manuals and adequate support or backup, or without documentation on verification and validation, present a formidable challenge to the analyst. The computation division must be able to control the executable program and must validate, verify, and document every change to the source program. In addition, the computation division must be capable of changing the source program to accommodate the analyst to the extent of adding material models, boundary conditions, and input or output data in a compatible format; those changes must be compatible with the organizations hardware.

The analyst must understand the workings of the code and ensure that the problem is running correctly and that the verification problems are appropriate to the applications anticipated. The analyst must understand the basis of the program in order to interpret the manual and the results correctly. A number of applications will be discussed from the analyst's perspective to illustrate the motivation and procedure used in selecting, evaluating, and running a computer program.

During the last decade, with the increase of bigger and faster computers and more sophisticated computer programs, the capability exists to more readily perform three-dimensional analysis. The three-dimensional mesh is difficult to generate and the results of a three-dimensional problem are significantly more difficult to evaluate. The formulation of the mesh and the running time are only two of the considerations required in determining the degree of difficulty of a three-dimensional (3D) problem over a two-dimensional (2D) problem. Debugging time, size limitations in terms of accuracy, and ability to preprocess or postprocess may also impact the problem. Experienced analysts tend to shy away from 3D problems.

A number of people are considered experts on a code because they have read the works of others. In view of the fact that not everything is written down, especially in a journal article or even in the manual, there is no substitute for experience; even then, since problems are not identical, expert advice must be acted on judiciously. These experts tend to want 3D solutions more, compared to the analysts who have to perform the work involved. In addition, both 2D and 3D codes have options which are subtle but have a large impact. The experienced analyst will want to evaluate the effect of the options, whereas the reviewer can only guess at the ramifications. For example, SASSI[3] is a 3D soil structure interaction code which has had wide acceptance, especially since it was used in the evaluation of the Diablo Canyon nuclear reactor by NRC. Hence, SASSI is recommended for the solution of numerous problems. What is not recognized by the community which reviews the problem rather than performs the analysis is that while SASSI is a 3D code, it is in a crude form and is currently unable to handle more than 1,000 underground nodes; 600 according to the leading author. Representation of an underground structure and excavated soil such as that at the waste tanks can only be obtained in a very rough approximation by 1,000 nodes. In addition, a 1,000-node SASSI can consume up to 1/2 time and/or space on a Cray. Therefore, the hands-on analyst will be more judicious in selecting what problems to run with SASSI and the range of frequencies. Clearly, if the problem is 3D, a 2D solution will not be appropriate.

Consider an axisymmetric structure. FLUSH[4], the 2D soil structure interaction program can only address plane strain or plane stress problems and therefore its solution to an axisymmetric problem would be significantly off. However, a version of FLUSH call AFLUSH, which addresses axisymmetric problems, would give fairly good results. SASSI, therefore would be required unless one had AFLUSH, but FLUSH would not be acceptable.

Another consideration is the degree of accuracy required. If differences in peak acceleration of the order of 30% are acceptable, then one could use FLUSH for an axisymmetric problem. If the differences need be significantly lower, then a more appropriate program is required. If the effect is unknown in the solution, such as friction in the superconductor supercollider project or sloshing of contents in waste tanks due to seismic loads, then programs have to be selected which can address these effects. ANSYS cannot address friction between components adequately (under large displacements the solution will not converge) and SASSI does not have fluid elements, so that they would not be appropriate to evaluate these effects.

APPLICATIONS

Different applications will now be reviewed in terms of the basis of selection of the computer programs: evaluation of a massive reinforced buried structure subjected to penetration blast effects, local adjustment of hexagonal spherical mirrors (the 10-meter Keck telescope being built on the island of Hawaii), evaluating the magnet of the superconductor supercollider project, and evaluating deeply embedded structures and soil structure interaction and sloshing/structure fluid interaction due to seismic effects. In each of these projects, it was the problem in search of the solution. The basis of code selection in each case will be presented.

The first case consists of a buried nuclear reactor subjected to a conventional penetration weapon; it is desired to obtain the floor response spectra at points a through h in Fig. 1. Another problem of the same type is the impact of a missile or the crash of a plane onto a critical massive structure, such as a reactor building, hardened missile site, or control command center. The details of Fig. 1 are fictitious.

The salient features of this problem are: (1) for the case of the detonation immediately over the axes of the dome, the problem is axis-symmetric and therefore two dimensional. For the other cases, the loading is unsymmetrical with respect to the structure and is therefore a three-dimensional problem even though the structure is symmetric. (2) the soil is clearly nonlinear which makes the problem nonlinear.

The absorption of the shock wave by the soil as well as the interface between the soil and the structure are important considerations. Rapid convergence of the solution is a necessity. A solution obtained by an A & E firm using a code inappropriate to this problem produced results which were inconsistent with hand calculations. Other codes had convergence problems because the soil deformations were large. The code requirements were: large deformations in some parts of the problem, small deformations in other parts of the problem, fast convergence, ability to use as a driving force a pressure-time history, a 3D loading condition, elastic models, orthotropic models, and a soil model, and as output - time history which can be converted to spectra. The code which was selected to address these issues is called DYNA[5] that was developed by John Hallquist. As the code was developed under government contract, both the source programs and manuals are available through LLNL. Today, DYNA has additional capabilities other than those it had when this problem was first addressed. The original pre- and postprocessors have been replaced by items with more efficient, accurate, and extensive capabilities than were available. The pre- and postprocessors available today that are compatible with DYNA are MAZE[6] and INGRID[7] for the 2D and 3D preprocessing, respectively, and ORION[8] and TAURUS[9] for 2D and 3D postprocessing, respectively. The source programs can be run remotely via a modem from the LLNL computer center. INGRID and MAZE produce the nodal coordinates and elements in the specified geometry. ORION and TAURUS produce the time–history plots of deflection and stress as well as contours of deflection, stress, etc. at a specified time during the application of the force.

The results of the DYNA calculation indicated that the capability of the hardening mechanism and the spectra obtained at the bottom of the reactor were comparable to seismic response spectra.

To achieve the final desired surface shape for the Keck telescope's primary mirror segments, a set of leaf springs are to be attached to the segment support structure. These springs will apply forces to the glass and remove aberrations remaining from the mirror polishing, the release of residual stress by the cutting into a hexagon, and the coring of the holes for the segment support. The set of leaf springs is collectively referred to as a "warping harness". The Keck Project is a joint effort of a number of institutions such as Cal Tech, U. C. Berkeley, LBL, University of Hawaii, and others. The problem is to determine the forces and their magnitude required to locally adjust the shape of the hexagonal mirror. The objective of the telescope is to observe new galaxies, and hence the accuracy required to view millions of miles must be comparable to distance. Here, a solution that is highly accurate such that small loads can be applied to deflect the mirror precisely, is required. Not only is the problem elastic, but the 2D solution does not have sufficient accuracy considering that the loading is unsymmetric. The results have to be magnified by the order of 10**7 power. If the mirror cannot be adjusted accurately, the project could not be successful.

A highly accurate linear elastic 3D code is required with a preprocessor that is capable of generating a spherical surface on one face of the mirror and hexagonal in shape and a postprocessor capable of magnifying the deflections. While a number of codes are capable of performing this calculation, GEMINI was selected because it is an updated version of SAPIV which is extremely well accepted for this type of calculation. In addition, a pre– and postprocessor compatible with GEMINI was available.

Papoulia et al.[10], using a commercial version of SAPIV[11] and modeling the structure with 2D, obtained results which were used until March 1989. Using the advanced version of SAPIV called GEMINI[12] and the postprocessor TAURUS and preprocessor INGRID, a three–dimensional model of the mirror using 3,500 nodes and 13,000 nodes, each, was developed for about a dozen load cases. The results from the two GEMINI models were identical and were slightly different than the 2D plate element calculation. The plate element considers a symmetric plate with a hole on both sides and a given thickness. In the GEMINI calculation, a hole exists only on one side and a spherical surface on the other. The 13,000–node calculation was performed to ensure that the 3,500–node calculation was sufficiently accurate. The hexagonal spherical mirror is shown in Fig. 2 and the results of one particular load case, where the z displacements are scaled by a factor of 10**7, shows that the deformation can be controlled locally, Fig. 3. GEMINI is now being used for controlling the 10–meter telescope mirrors.

In addition to the technical aspects just described, limited solution time was of the essence. The programs had to be on the computer, the input had to be written, and the problem had to be run by an experience analyst. Therefore, it was fortuitous to be able to meet all the requirements. Often it is recognized that another program can perform a calculation more accurately, faster, etc. but the time is not available to get the program and develop the expertise necessary to run and evaluate it. To this end, it is the responsibility of management and the senior analysts of an organization to have those computer programs which may be required available and to develop the expertise before a problem arises. Unless this capability is available, the selection process becomes a question of whether what you have in terms of programs is acceptable or subcontracting the work. Depending on the computation department and language in which the large frame programs are written, it will take a finite time to compile a source program compatible to your system. Sufficient lead time is generally not available. Getting a subcontractor depends on the time it takes to process the paperwork; in government circles this could jeopardize this option. Subcontracting analysis can be very expensive if commercial computers are used instead of government-owned and operated computers.

The magnet system of the superconductor supercollider is complex as shown in Fig. 4. The operation is as follows: first an assembly stress is placed on the magnet coils, inner and outer, the cryostat is cooled to 4.35° Kelvin, and then the magnet is energized to 6.5 tesla. Todays requirements and the design have somewhat changed since 1989 when the author was a member of the Central Design Group under Universities Research Association at LBL. To return to the magnet, as the magnet is energized, Lorentz forces are created which are opposite to the assembly stress. When the Lorentz forces exceed the assembly stress, the coils individually begin to move in a stick–slip abrupt motion and a quench occurs. The operation of the quench is described in Fig. 5. Experimental results indicate that the quench normally occurs in the coil ends. Modifications make the sections where the coil turns around more gradually to ensure that at the ends no components are permitted to move. The model is asymmetric in geometry with respect to the longitudinal axis but the transient temperatures are not axisymmetric nor are the forces at the ends. In selecting the analysis program applicable to this problem, ANSYS was preferred by physics laboratories as it is their main code. ANSYS is used successfully to calculate Lorentz forces. However, in evaluating the applicability of ANSYS is appeared that with friction included, ANSYS could not converge. Coupled with transient temperatures, this presented a difficulty in ANSYS as well as codes such as NASTRAN. Therefore, we opted for NIKE[13] with TOPAZ[14] which have both 2D and 3D versions. NIKE, it has been demonstrated, does not have any convergence problems. After several years, it is believed that friction under small displacements have been achievable with ANSYS. NIKE has a 2D preprocessor MAZE[6] and the postprocessor ORION[10], and the NIKE and TOPAZ 3D versions are compatible with INGRID and TAURUS. Convergence was not a problem and failure was predictable as compared to strain gage measurements (Figs. 6 and 7). Using INGRID, the complex 3D mesh was completed (Fig. 8). Much discussion has taken place about using ANSYS versus NIKE. Unlike NIKE, ANSYS is a proprietary code and it costs every time it is used. In addition, one does not have access to the source program. While the opposite is true for NIKE, the manuals and the other documentation do not compare in completeness or detail to those of ANSYS.

An attempt was made by the Dallas Lab to use NIKE. Instructors were brought in and courses taught. However, in view of the difficulty in coming to speed and the ability of ANSYS to finally address friction with small displacements, as well as a redirection in downplaying the role of analysis, the use of NIKE was abandoned. Three dimensional analysis of the magnet end was never considered by the Dallas Lab. The analysis was then passed on to subcontractors who have opted to emphasize the manufacturer of equipment rather than the analysis. The cost has also escalated from $4 billion under CDG in Berkeley to at least $8 billion. Magnet cost increases contributed significantly to this cost escalation.

In evaluating the P Reactor at the Savannah River Site, FLUSH and SHAKE[15] were used as well as GEMINI. GEMINI addressed the difference in center of mass and and center of rigidity or geometrical center. FLUSH models and GEMINI models are given in Figs. 9 through 13, with results of difference in spectra due to rO component shown in Fig. 14 and Fig. 15 of the enveloped spectra. To use SASSI on a reactor that is deeply embedded requires a crude mesh to model the underground. The experience gained in this investigation relative to the topic is as follows: (1) computer codes were selected that were acceptable to both the analysis community and DOE. (2) while the codes were onsite at the time of the decision, FLUSH was not compatible with UNICOS Cray language. FLUSH has to be run on the Cray. (3) Vax codes require almost no modification and can be compiled almost immediately. However, the Vax is both slow and has limited storage. It took over two months to convert the CDC version of FLUSH to UNICOS. Two months which we could ill afford. SHAKE and GEMINI were run on the VAX. At that time, it was taking too much time to get GEMINI running on the Cray. Today, both FLUSH and GEMINI can be run on the Cray. (4) there is no preprocessor or postprocessor for FLUSH, so they had to be written. (5) none of the three codes were validated or verified, a project we had to also undertake to be in conformance with site QA and DOE. (6) since no one here had run FLUSH previously, there was a learning experience. The FLUSH manual leaves much to be desired in completeness. This was all performed under a time constraint. Now that the experience has been obtained, assuming the staff does not leave, the organization is better equipped to address the next problem.

A seismic problem of considerable importance is the one dealing with the waste tanks, Fig. 16. The first aspect of this problem is how to evaluate the response due to a seismic load. The following issues need to be addressed: (1) soil-structure interaction since the waste tanks are buried, (2) structure soil-structure interaction – the effect of adjacent waste tanks, (3) sloshing of the contents of the tanks, and (4) modeling the contents of the tanks. Codes considered were SASSI, FLUSH, ABAQUS, SHAKE, DYNA, AFLUSH, ANSYS and combinations of them. The only code that can address all the issues and the one for which there are no negative arguments is DYNA. When DYNA was first suggested, the author of SASSI claimed it would take too long to run and a DOE/Westinghouse consultant claimed it was overkill. However, SASSI does not address fluid–structure interaction nor interface pressure between the tank wall and soil, FLUSH or SUPERFLUSH do not address sloshing and are 2D codes and the tanks are axisymmetric but the loading is unsymmetrical, ABAQUS does not address soil–structure interaction, and neither does ANSYS. Combinations such as SASSI and ABAQUS or DYNA were then considered.

For the program undertaken at WSRC, it was decided to use SASSI to obtain the input motion to the tank when the design criteria are R. G. 160 response spectra at the surface, to use SASSI to evaluate tank to soil to tank interaction, and to use DYNA to evaluate the sloshing and the stresses in the tank and tank liner. SHAKE is used to obtain strain compatible values to be input to SASSI; DYNA has two fluid models, one more accurate than the other. The results show that the R. G. 1.60 spectra exceed all other results obtained by SASSI including upper and lower bounds on the best estimate of soil properties. Using a time history consistent with R. G. 1.60 spectra applied to the base of a DYNA model, sloshing was observed.

In order to use SASSI, the use of symmetric and antisymmetric planes is required, even though the mode shapes may be distorted. Separation occurs with and without gravity though it is very pronounced when there is no gravity. Because of the limitations of the number of nodes underground, the SASSI models are crude. A limitation of DYNA is that, in the actual WSRC model, the thickness of the liner dictates the time step, which is very small. However, the calculations were performed for as long as required to get 10–20 seconds of strong shaking. Without the liner, the duration of a problem is very reasonable. The effect of the soil around the tank, as far as interface pressure, remains to be evaluated, but the effect at the base is less than R. G. 1.60. Since WSRC is now performing its own calculations, the one Cray is being taxed.

The difference cited by DOE consultants and WSRC consultants is the modeling of the soil. In SHAKE, SASSI, and FLUSH strain–compatible shear modulus and damping values based on the work of Seed and Idriss for sand, clay, and rock are used. In DYNA, there is one soil model which is based on the shear modulus, bulk unloading modulus, and the volumetric strain versus pressure, the same model with failure included, and in addition, an inviscid two–invariant geologic cap model. Basically, the cap model refers to the failure envelope. The Seed and Idriss model does not address failure but has gained wide acceptance. In a study performed by Goudreau, Chen, and Johnson, it was shown that FLUSH, DYNA, and CLASSY (SASSI) produce equivalent results. The main advantage of DYNA is that DYNA, like all of Hallquists computer codes, contains either a slide line or slide surface. This concept permits two materials to either be locked (permits disjoint elements), slide freely, slide with friction, separate, or come together. Other codes such as ABAQUS and ANSYS use the concept of ficticious gap surfaces which are at best unrealistic. The lessons learned here is that many codes were already in place, so options were available. With time it was possible to evaluate these options without compromising the program. Structure–soil–structure interaction could have been just as well evaluated with DYNA rather then SASSI. However, DYNA is not as well known for soil structure interaction as SASSI.

A DYNA model is shown in Figs. 20 and 21. The waste tanks are filled with combinations of viscous fluids. A time history was developed based on R. G. 1.60 scaled to 0.2g at 7% damping. The time history was input to the tank base and the stresses in the tank were evaluated. To capture a more realistic geometry, a steel liner was modeled as 3/4-in.-thick shell elements, recognizing the havoc with the time step. The time step was adjusted to max allowable. However, since it was only a time problem and not a space problem it was possible to make a complete run of an entire earthquake record. A 16 mm movie of sloshing has been made which shows the effect of separation of the fluid from the tank; with gravity the separation is significantly reduced.

In conclusion, many issued are involved in the analysis process and not all of them are technical. The successful analyst must unfortunately consider all the issues. Competing issues must be resolved. The documentation and quality assurance requirements that are now imposed in most organizations and not to be ignored. Substantial effort in this area is required and is beneficial to the analyst. Both management and the analyst must plan in advance for codes that are to be available; the peripheral hardware, terminal, printer, etc.; whether the data are to be stored on fiche or hard copy; and is there available time and space on the VAX and or Cray.

Effort should be placed on expanding code availability; this could be addressed in different ways. Expertise should be available to answer questions during the entire process. Learning in a vacuum is not only difficult and costly in terms of time, but sometimes impossible since the decision on a code requires evaluation of the manual and the program simultaneously. One of the key issues is the time available for obtaining the solution. Normally, the client wants the answer yesterday. As a cautionary note, do not underestimate the time required to setup a code on the Cray or learning a new code and its nuances. As far as obtaining equipment, never, but never, underestimate the time required in purchasing equipment or software if you work at a government facility.

REFERENCES

1. ANSYS, Engineering Analysis System Users Manual, Issued 6/1/87. G. J. DeSalvo and R. Gorman, Sunnson Analysis System, Inc. Houston, PA 15342.
2. ABAQUS Users Manual, Copyright 1989. Hibbitt, Karlsson, and Sorenson, Inc.
3. SASSI, System for Analysis of Soil Structure Interaction. Prof Lysmer, Prime Author, U. C. Berkeley.
4. FLUSH, J. T. Lysmer, C. Tsae, and H. B. Seed, A Computer Program For Approximate 3-D Analysis of Soil Structure Interaction Problems, U. C. Berkeley.
5. DYNA2D, DYNA3D, An Explicit Two-Dimensional Hydrodynamic Finite Element Code With Interactive Rezoning and Graphical Display. J. O. Hallquist, LLNL.
6. MAZE, An Input Generator For DYNA2D and NIKE2D. J. O. Hallquist, LLNL.
7. INGRID, A Three-dimensional Mesh Generator For Modeling Nonlinear Systems. J. O. Hallquist and D. W. Stillman, LLNL.

8. ORION, An Interactive Color Postprocessor for Two-dimensional Finite Element Codes. J. O. Hallquist and J. L. Levatin, LLNL.

9. TAURUS, An Interactive Postprocessor For Analysis Codes NIKE3D, DYNA3D, TACO3D, and GEMINI. LLNL.

10. K. D. Papoulia, V. Panoskaltsis, and T. Mast, Keck Observatory Report 163. April 1987.

11. SAPIV, A structural Analysis Program For Static and Dynamic Response of Linear Systems. K. J. Bethe, E. Wilson, and F. E. Peterson, U. C. Berkeley.

12. GEMINI, A Computer Program For Two- and Three-dimensional Linear Static, and Seismic Structural Analysis. R. C. Murray, LLNL.

13. NIKE2D, NIKE3D, A Vectorized Implicit, Finite Deformation Finite Element Code For Analyzing The Static and Dynamic Response of 2D/3D Solids With Interactive Rezoning and Graphics. J. O. Hallquist, LLNL.

14. TOPAZ3D, A Three-dimensional Finite Element Heat Transfer Code.

15. SHAKE, A Computer Program For Earthquake Response Analysis of Horizontally Layered Sites. P. D. Schnabel, J. Lysmer, and H. B. Seed, U. C. Berkeley.

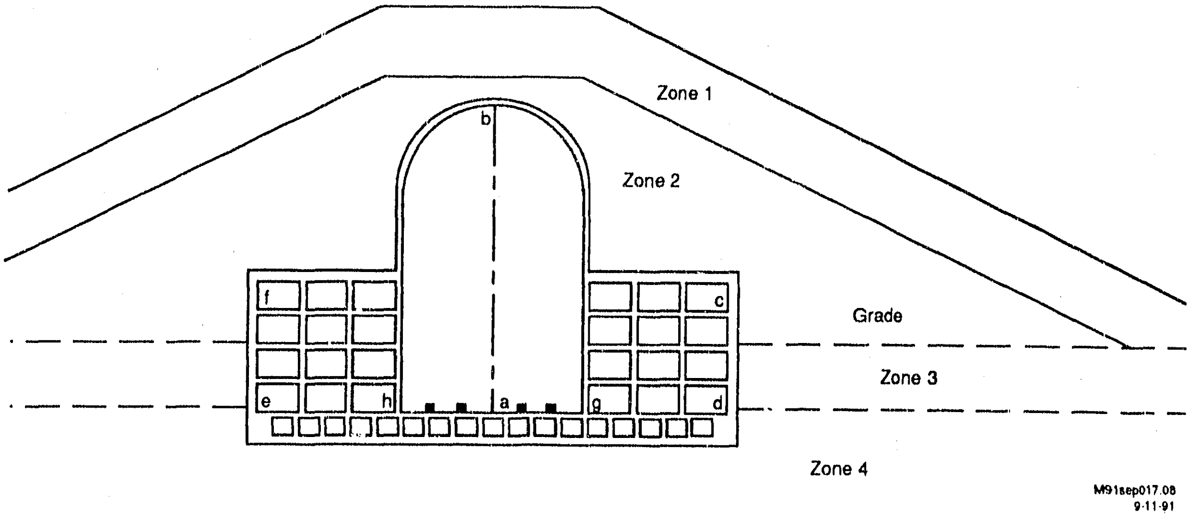


Figure 1. Buried Nuclear Reactor

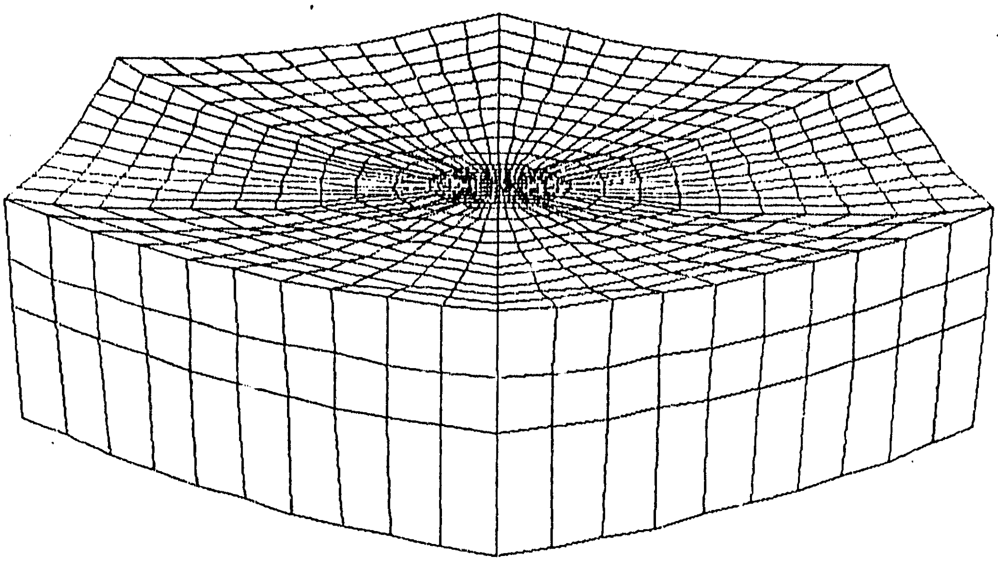


Figure 2. Hexagonal Spherical Mirror

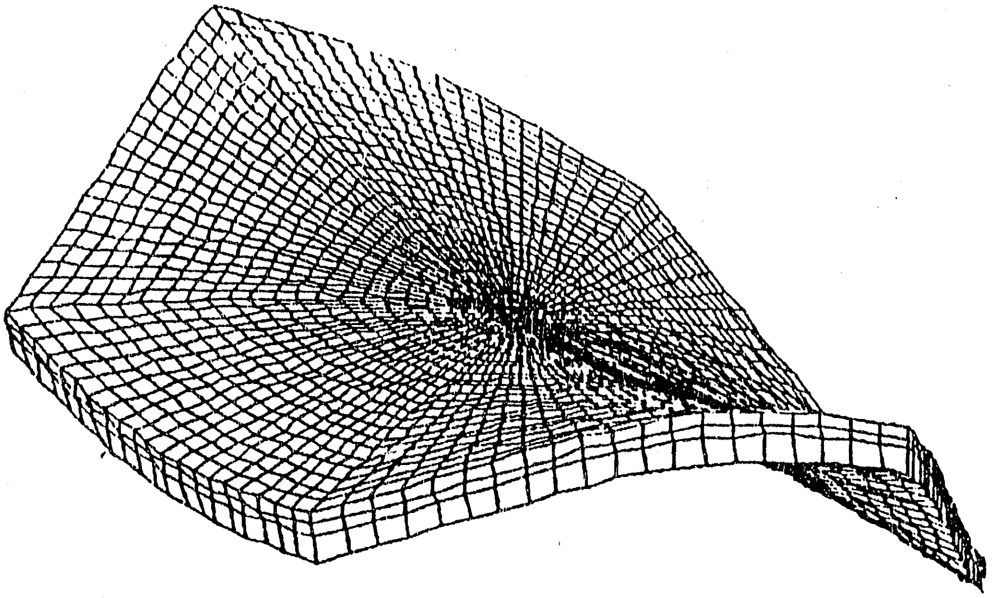


Figure 3. Displacements in the z Direction Scaled by 1×10^7

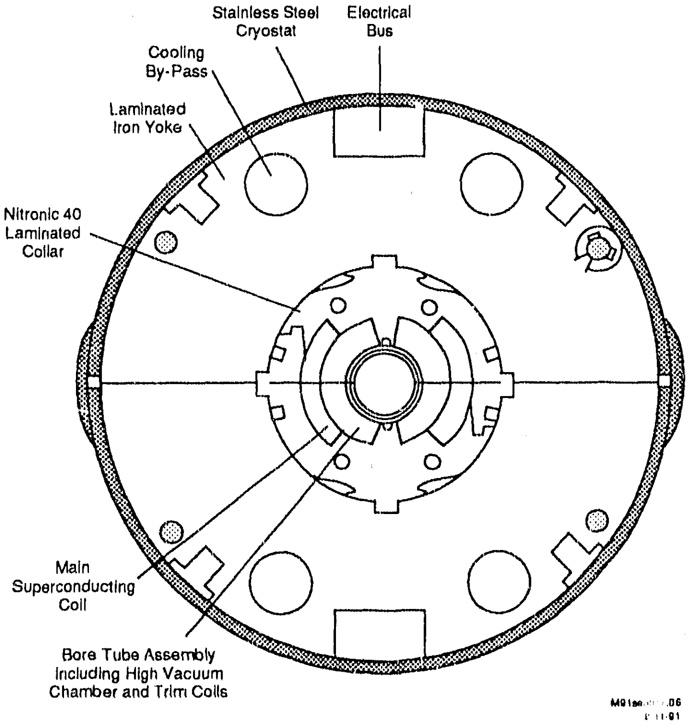
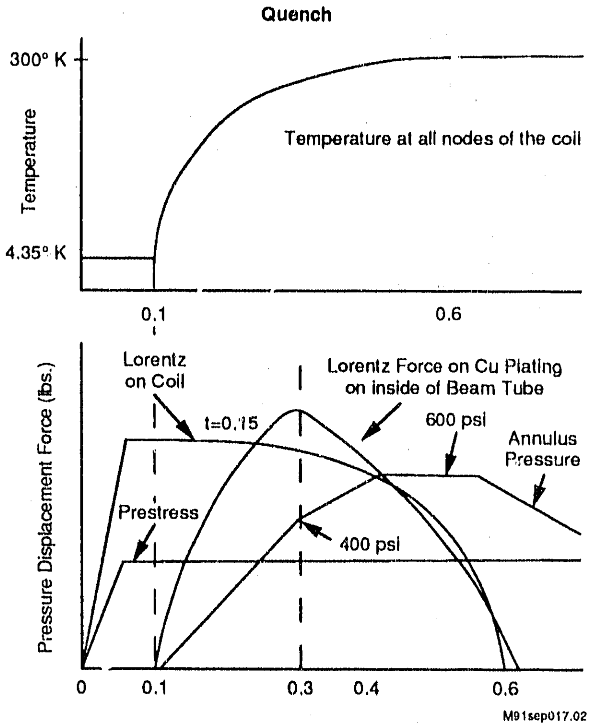
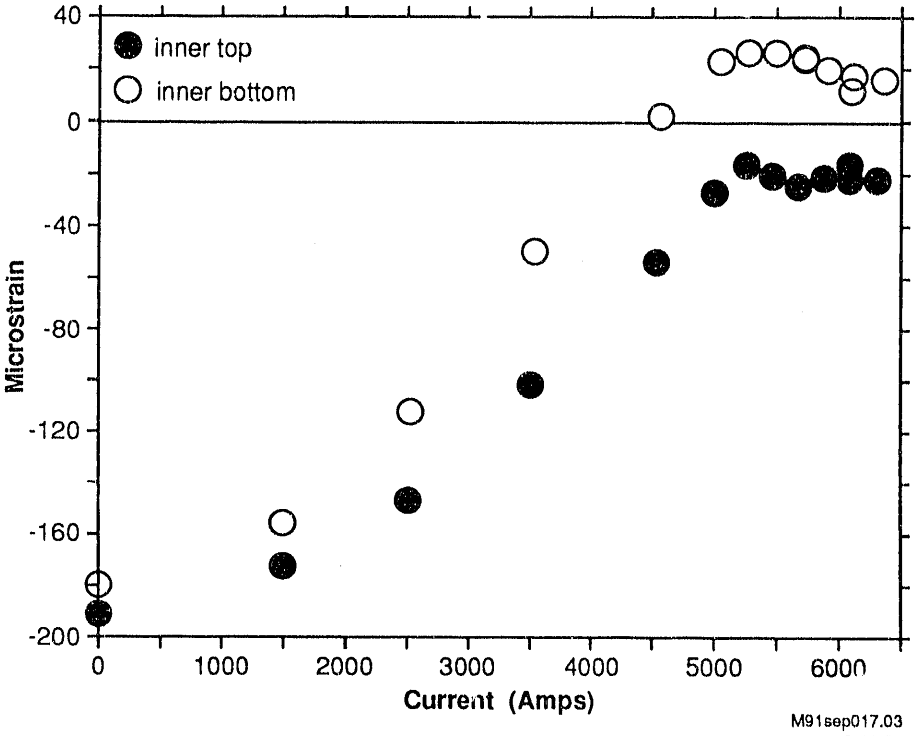


Figure 4. Completed Helium Containment Cross Section



M91sep017.02
9-11-91

Figure 5. Loading Conditions



M91sep017.03
9-11-91

Figure 6. Coil Strain Gages Versus Current D00002

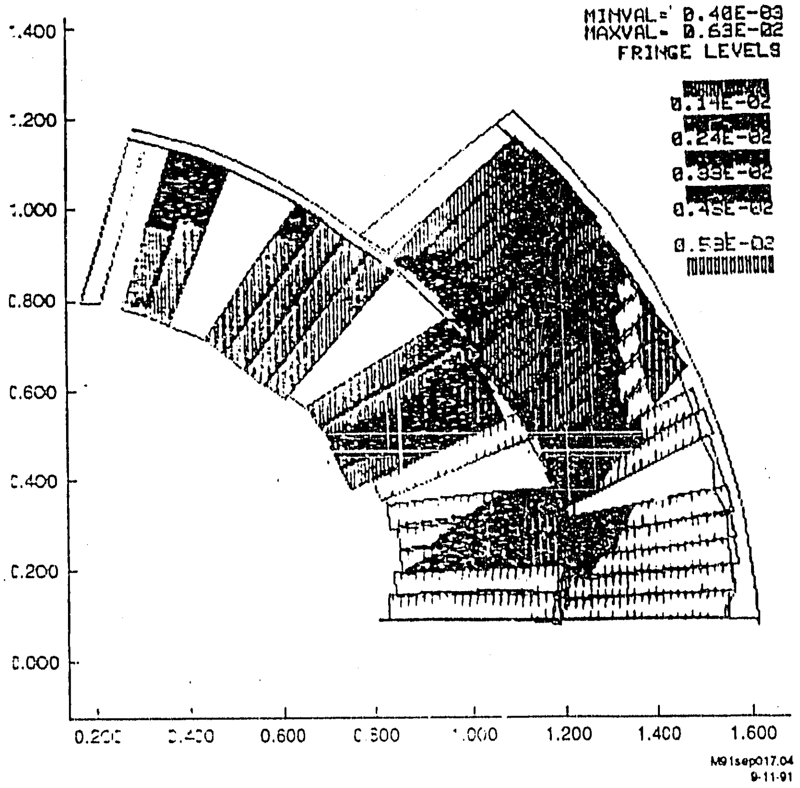


Figure 7. Fringes of Maximum Displacement

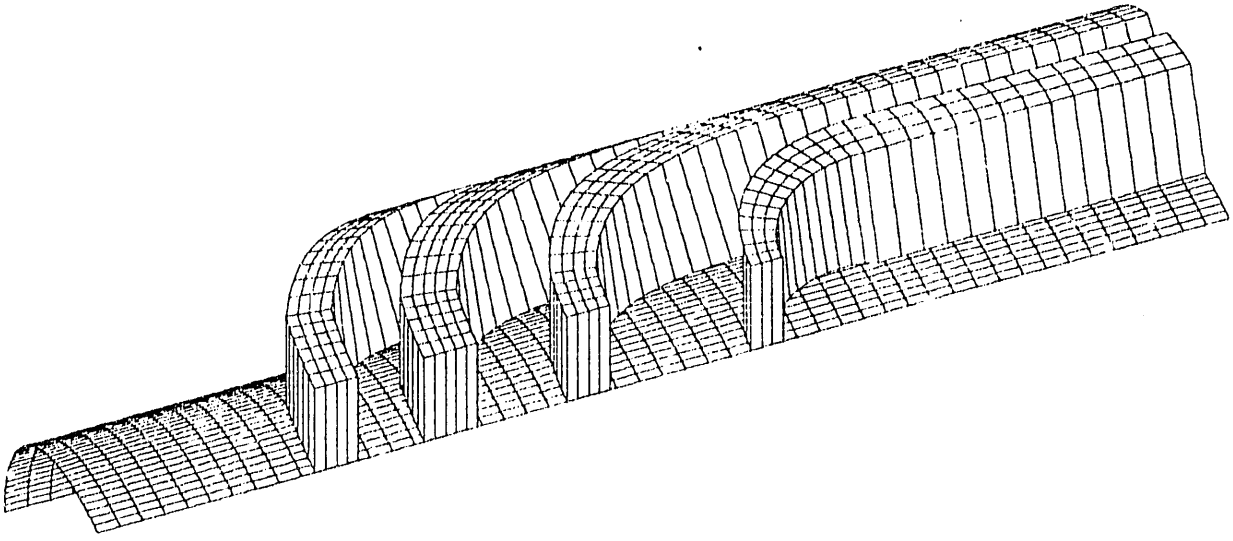


Figure 8. 3-D Mesh Using INGRID

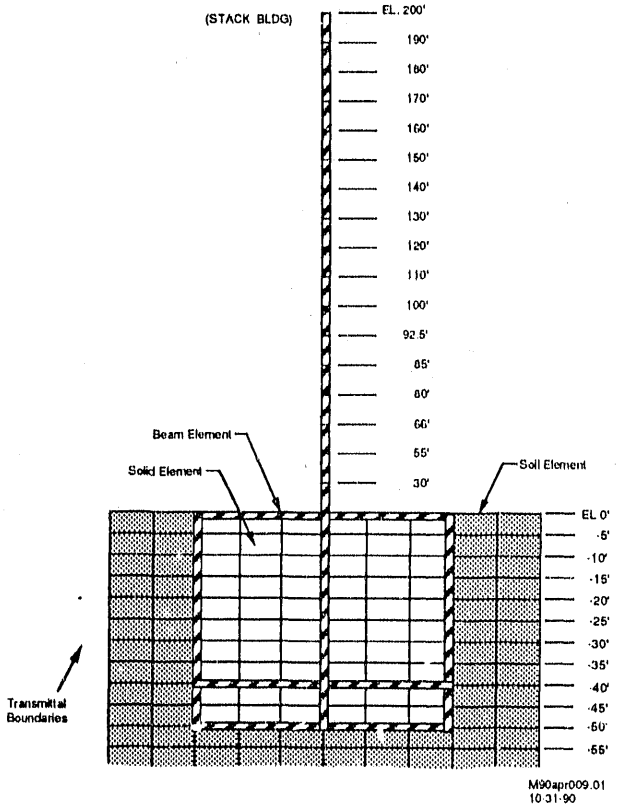


Figure 9. FLUSH Model (Looking West)

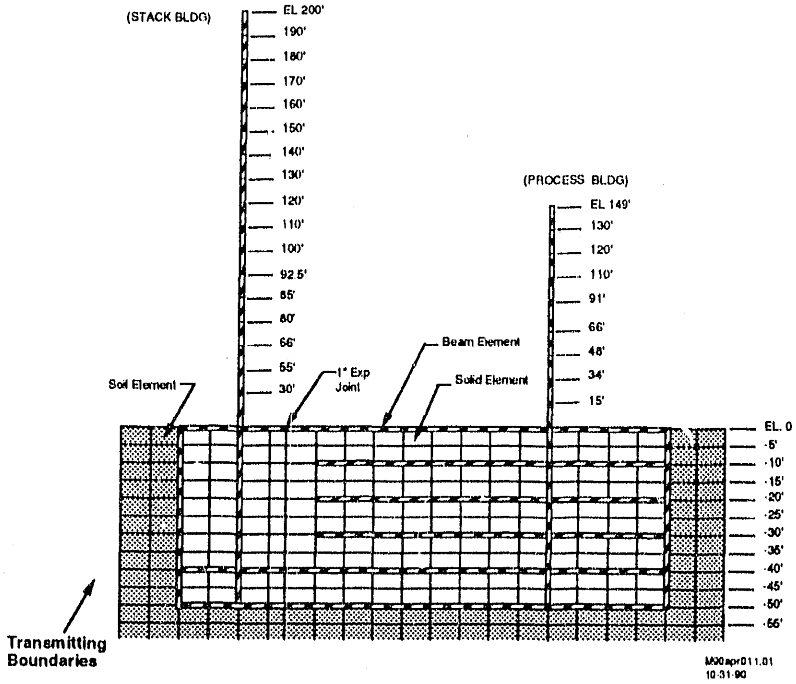


Figure 10. FLUSH Model (Looking South)

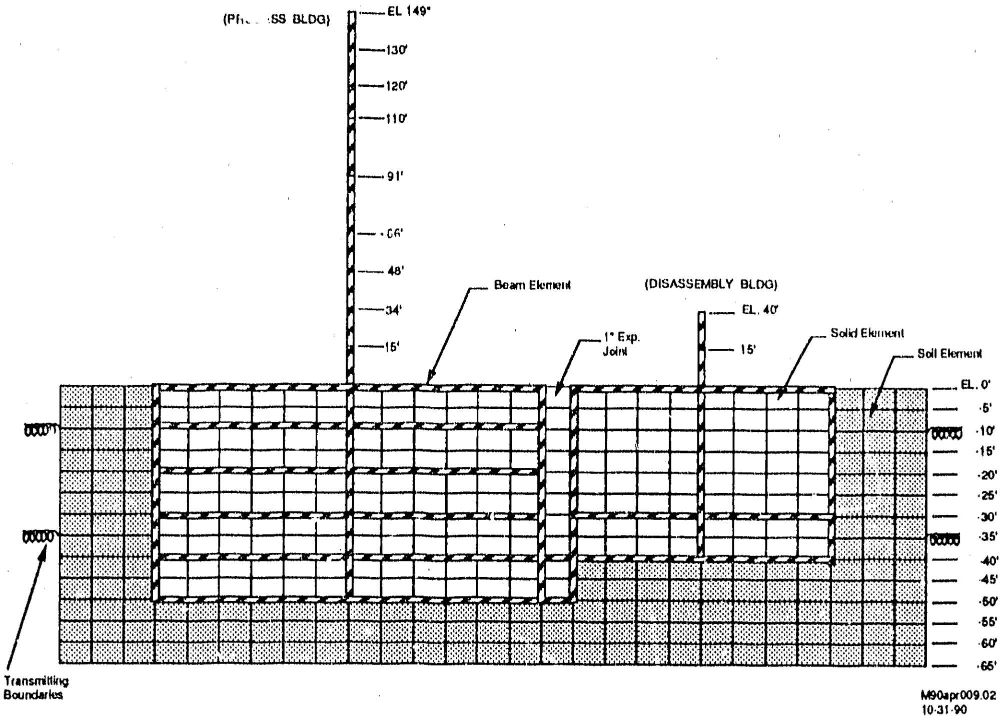


Figure 11. FLUSH Model (Looking East)

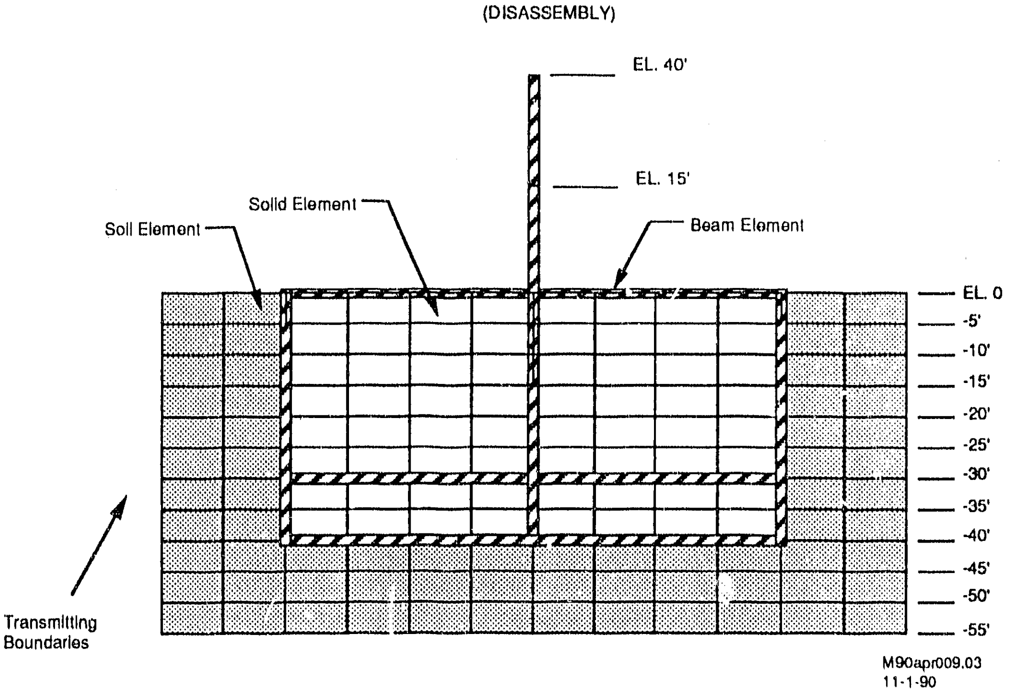


Figure 12. FLUSH Model (Looking North)

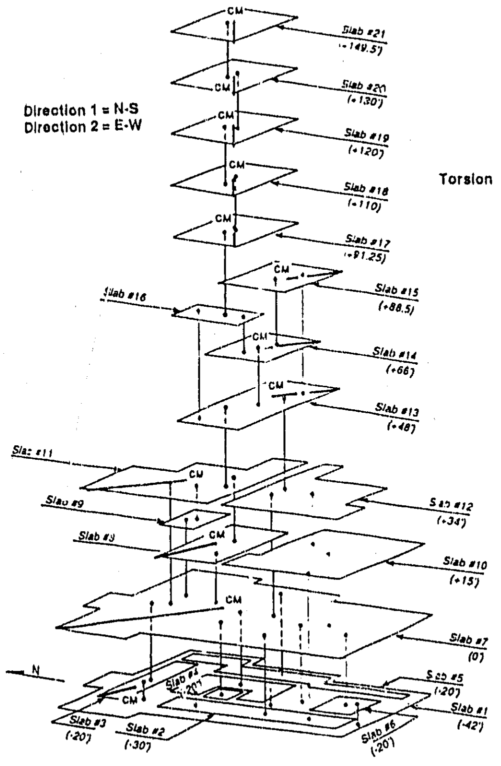


Figure 13. Three Dimensional Process Building Model with Rigid Links on Each Slab

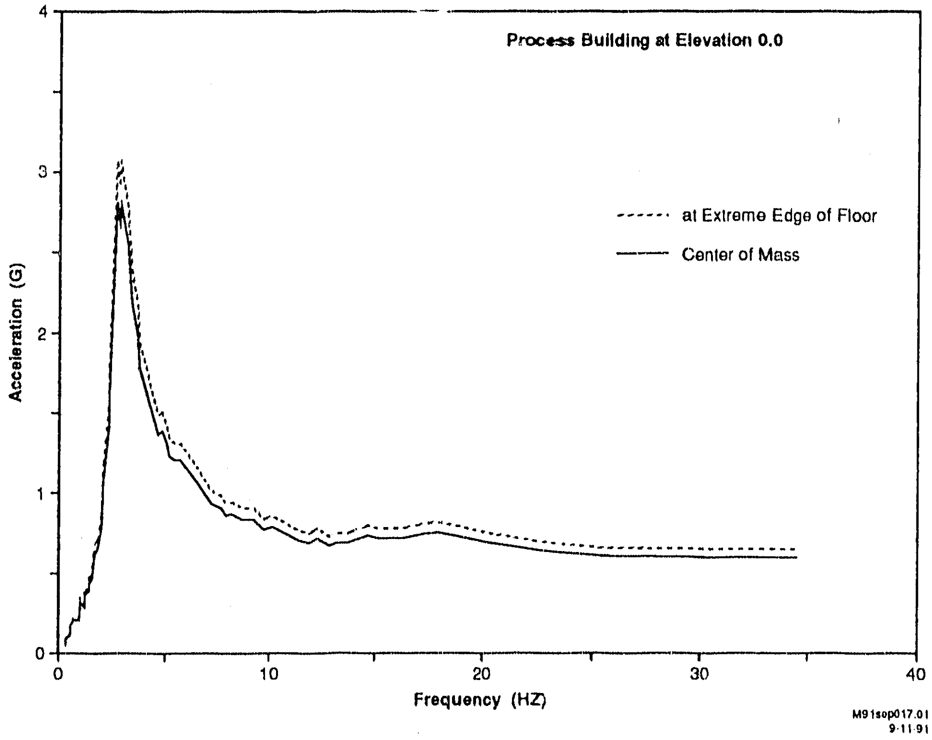


Figure 14. The Effect of the r0 Component on Spectra

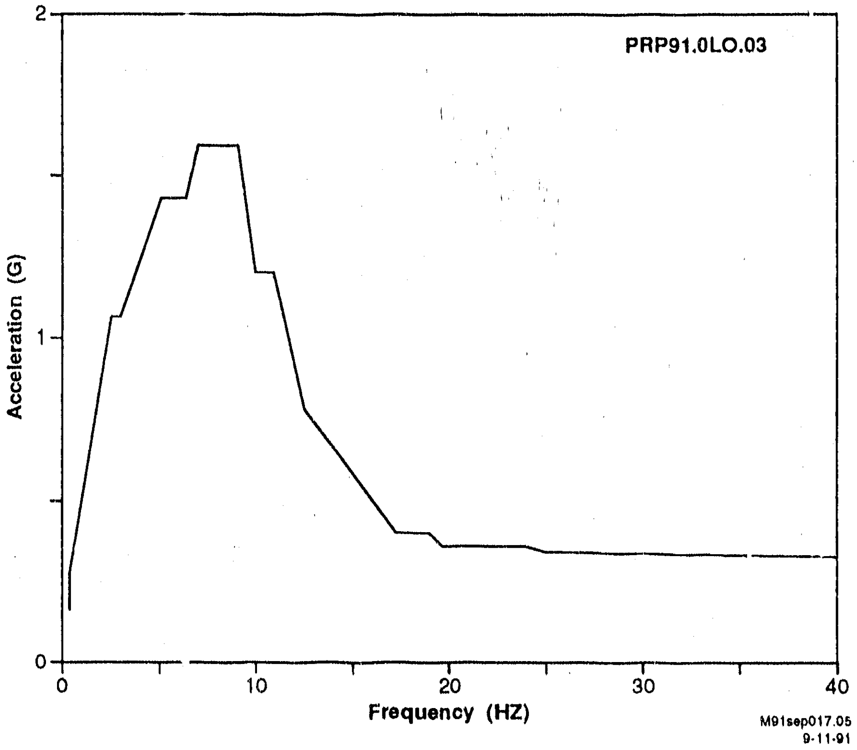


Figure 15. The Enveloped Spectra at 91 ft, Longitudinal Direction, Process Building, 3% Damping

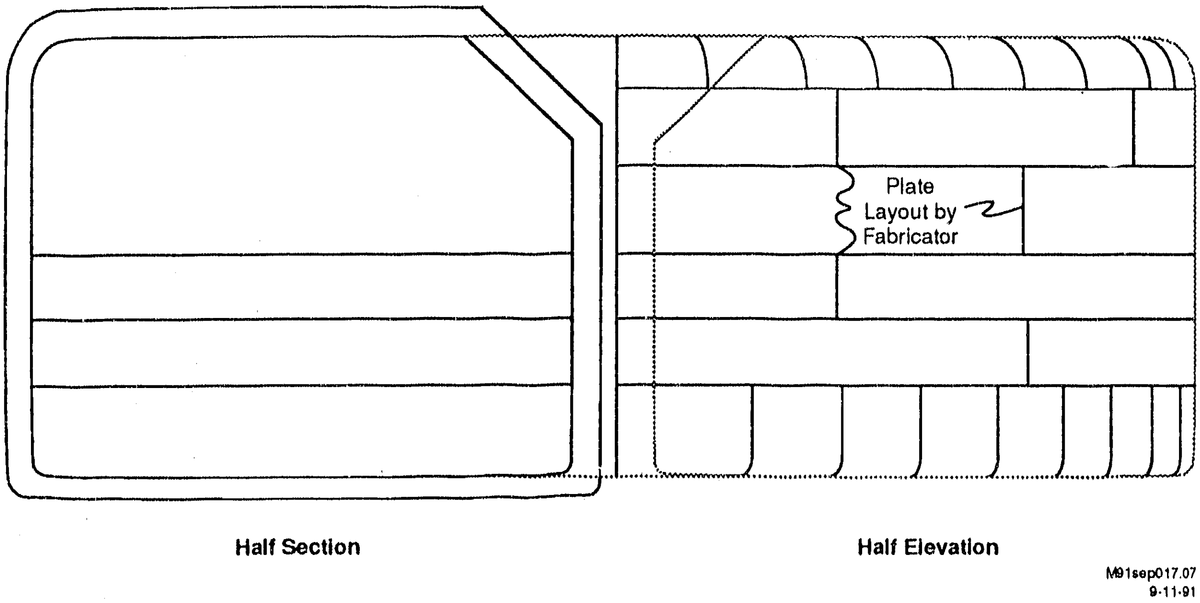


Figure 16. Waste Tanks

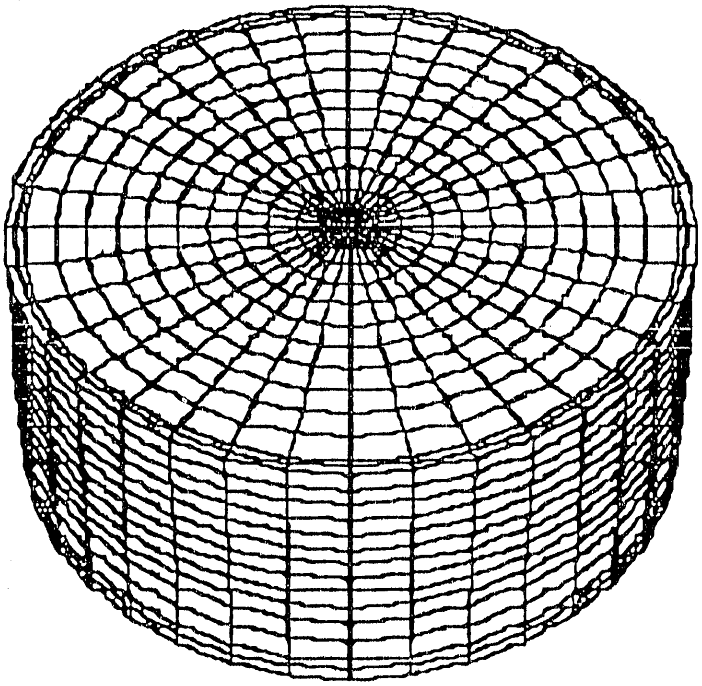
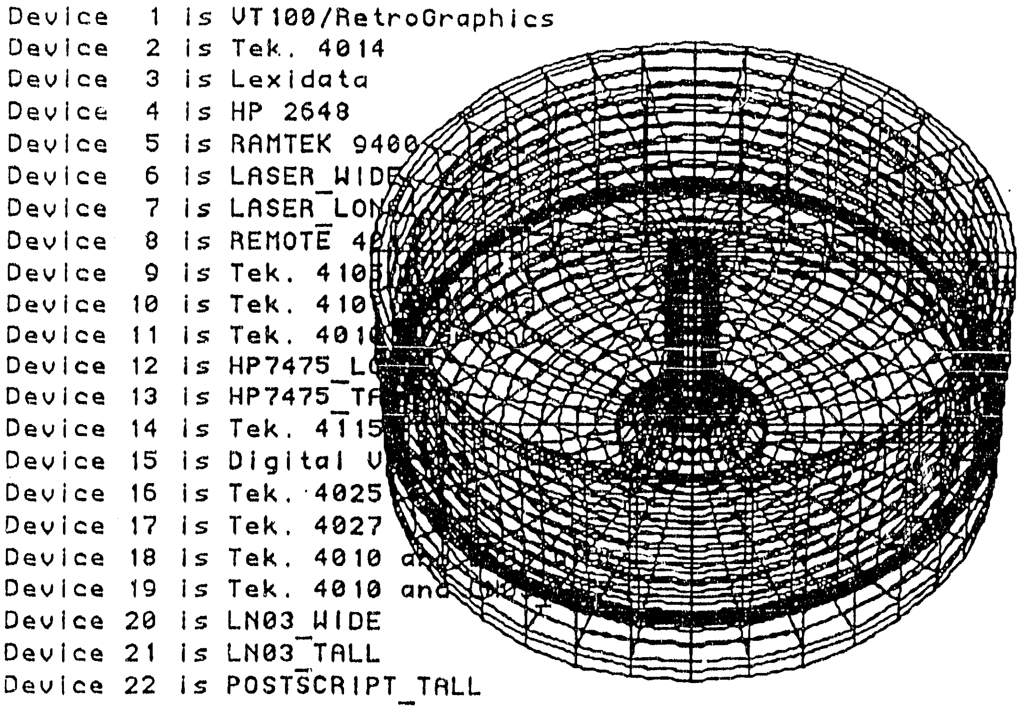


Figure 17. DYNA Model



```
Device  1 is VT100/RetroGraphics
Device  2 is Tek. 4014
Device  3 is Lexidata
Device  4 is HP 2648
Device  5 is RAMTEK 9400
Device  6 is LASER_WIDE
Device  7 is LASER_LON
Device  8 is REMOTE 4
Device  9 is Tek. 410
Device 10 is Tek. 410
Device 11 is Tek. 401
Device 12 is HP7475_L
Device 13 is HP7475_T
Device 14 is Tek. 4115
Device 15 is Digital V
Device 16 is Tek. 4025
Device 17 is Tek. 4027
Device 18 is Tek. 4010 a
Device 19 is Tek. 4010 and
Device 20 is LN03_WIDE
Device 21 is LN03_TALL
Device 22 is POSTSCRIPT_TALL
```

Figure 18. DYNA Model with Devices

END

DATE
FILMED

6/15/92