MSC/NASTRAN V66 OPTIMIZATION

as implemented in a

PRODUCTION ENVIRONMENT

using the capabilities of the

IBM 3090 VECTOR FACILITY

Mr. Scott R. Bird, Technical Marketing Manager
Dr. Curtis J. Hoff, Vice President

Automated Analysis Corporation
2001 Commonwealth Blvd, Suite 204
Ann Arbor, Michigan 48105
(313) 665-6300

ABSTRACT

Structural optimization is a tool that has been used in academic circles and research facilities for a number of years. It is now being recognized as a tool for use in production environments by design engineers and analysts. MSC/NASTRAN has included an optimization capability in Version 66 that claims to put this tool in the hands of the engineer. This paper will evaluate the tool from three different perspectives. The first is the accuracy and features contained in the code itself. Secondly, from a larger point of view, the capability is reviewed as a tool in a production environment. The last perspective is purely from a hardware point of view. The implementation on the IBM 3090 Vector Facility is reviewed in terms of CPU, disk access, and parallel processing.

Two problems are presented. The first is a simply supported square plate with a constant pressure load. This academic problem enabled the authors to learn and understand the capability and compare the results to a theoretical solution.

A previously developed model of a Caterpillar Incorporated bulldozer blade was used to simulate the production environment. A series of optimization studies was performed using objectives and constraints developed by Caterpillar during the actual design process.

The results of this project show that the optimization capability is accurate and relatively easy to use, but can be extremely CPU intensive. It contains a number of features that are very helpful to the engineer, but also contains some errors and limitations that affected the work of the authors.

I. INTRODUCTION

Automated Analysis Corporation (AAC) is an consulting engineering company headquartered in Ann Arbor, Michigan. AAC consults in all areas of computer aided engineering, including structural and dynamic analysis, computational fluid dynamics, thermal analysis, and kinematics. Its primary industrial applications include all transportation related activities, ie. the automotive, aerospace, heavy truck, heavy equipment, and marine industries. Since its founding in 1983, AAC has had close ties to MSC and MSC/NASTRAN. AAC has both provided MSC with scientists on contract basis and placed former MSC engineers on its staff. AAC has also been very active in the field of structural optimization. The Chairman of the Board, Dr. William J. Anderson, is a professor of aerospace engineering at the University of Michigan and is a recognized expert in the field of structural optimization. Dr. Curtis Hoff, one of the co-authors, received his Phd in the field of nonlinear structural optimization from the University of Michigan in 1985. Mr. Scott Bird received his Masters in engineering in 1986, and received his initial finite element training while working in the client support department of MSC. AAC has also made extensive use of the Design Sensitivities Analysis (DSA) capability within MSC/NASTRAN in its consulting activities for many years.

As stated in the abstract, Version 66 of MSC/NASTRAN contains a structural optimization capability in Solution 200 of its library of solutions. This capability is designed to enable the engineer to vary several design variables simultaneously in order maximize or minimize an objective function while satisfying any number of design constraints. Currently, the capability is limited to parametric optimization, in which the design variables are found in fields on bulk data property fields. The current capability also enables the user to write equations that relate user defined design variables to property fields. Shape optimization is not found in MSC/NASTRAN, nor is it practically feasible for large scale problems given today's state of the art in finite element technology.

The purpose of this paper is to evaluate this capability as it relates to large, production oriented problems. The code's accuracy will first be checked through a comparison with a small, well documented problem. A large production problem will be optimized to examine the utility of the capability in a production environment. Throughout the paper, the features and weaknesses of the code will be discussed. Special attention will be given to the hardware implementation of this capability toward the end of the paper. All of the work described in this paper was performed on the IBM 3090 Vector Facility computers at Kingston, NY.

Two problems are presented in this paper. The first, used to evaluate the accuracy of the code, is a simply supported square plate under a constant pressure load. The second is a model of a Caterpillar Incorporated bulldozer blade under two design loads. The model was developed for Caterpillar by AAC for a recently completed design analysis. The loads and optimization parameters were developed in conjunction with Caterpillar in order to closely approximate the goals of a production environment. A third design study, a dynamic analysis of heavy truck cab, was originally planned for this paper, but was omitted due to errors in dynamic optimization capabilities. It is the understanding of the authors that these errors are to be fixed in Versions 66a and 67 of MSC/NASTRAN. The authors intend to perform further evaluations using this model when the capabilities become available.

The remainder of this paper will describe in detail the two projects completed, and present a discussion concerning the overall evaluation of the features found in Version 66, particularly Solution 200.

II. SIMPLY SUPPORTED PLATE

A 72 inch square, simply supported plate was chosen to evaluate the optimization capability in terms of accuracy. The results for this problem have been published [1]. A uniform pressure of 1.39 psi was applied over the plate. The initial plate parameters are:

Thickness	t	0.05 inches
Young's Modulus	E	3.0×10^{7} psi
Poisson's Ratio	ν	0.3
Density	ρ	7.43×10^{-4} lbs/in^3

Due to symmetry about both axes, only one quarter of the plate was modelled. An 11 node by 11 node mesh was used resulting in 121 grids and 100 QUAD4 elements.

There were four optimization problems performed on this plate model. The first considered the plate to be of constant thickness, and the last three allowed each element to have its own thickness. Allowing each element to have its own thickness provides a significantly more difficult problem to solve and also allows thickness contours to be presented. A summary of the optimization inputs is provided in Table 1.

Table 1. Plate Optimization Inputs

Run	Design Variables[1]	Stress Constraint[2]	Displacement Constraint
1	1	36,000 psi	n/a
2	100	36,000 psi	n/a
3	100	36,000 psi	4 in.
4	100	36,000 psi	3 in.

[1] Number of plate thicknesses allowed to vary.

[2] The maximum von Mises stress allowable on the top or bottom surface of the plate.

The first run was rather trivial and was used to debug the capability and check against hand calculated solutions. Even so, this was the point at which the authors encountered the first major error in the capability. The stress item codes which are used to identify design responses and constraints are the same as the plotting codes described in section 4.3 of the *MSC/NASTRAN Users' Manual* [2]. The plotting codes offer the user the option of either Maximum Shear stress or von Mises stress, based on the case control request made by the user. However, this request is not recognized by the optimization modules in Version 66, and the Maximum Shear stress is always calculated. This provided the authors with the opportunity to use the equation writing feature in Version 66, which worked as advertised with fairly little difficulty. The optimization converged to the theoretical solution in six iterations. Both the weight and the displacement of the center of the plate are plotted as a function of iteration number in Figure 1.

Runs two, three, and four were all run without incident or error, and worked quite well, as shown in Table 2. In all cases, the results were the same or better than the published optimal. Tables of weight and center displacement versus iteration number are presented in Figures 2 through 4. Contours of plate thickness are also presented for each case in Figures 5 through 7, where the reference contours from the documented solution are also plotted [1].

Table 2. Plate Results

Run	Iters	Weight (lbs)	Ref Weight (lbs)	% Diff
1	6	.980	.980	0.0
2	13	.752	.752	0.0
3	12	.743	.768	-3.2
4	16	.742	.840	-11.7

Although the thickness contours differ from the published results, the displacements and stresses generated by those thicknesses in a statics run all fall within the design constraints. The reader will also note the significant improvement in weight of 11.7% from the published optimum.

As a side study, the authors also ran the plate problem with 2, 4, 5, 10, 20, 25, and 50 design variables in order to study the effect of design variables on required iterations and CPU time. The results are shown in Figure 8. This study showed that the CPU time is linearly related to the number of design variables. This relation does not appear until there are a larger number of variables due to the relatively small size of the problem.

The data point for 4 iterations is missing on the chart due to the fact that this problem would not converge to a feasible solution given the initial conditions supplied to the problem. Many different initial conditions were supplied in an attempt to make this particular configuration converge. The only condition that allowed convergence was when the initial conditions were placed very close to the optimum (within 4%). MSC was contacted concerning this error, who responded by admitting that the optimizer could "get stuck," and offering some advise on how to solve that type of problem. The advice given included moving the initial condition of the design variables to the center of the allowable range, or relaxing the convergence criteria until a feasible solution is found, and then restarting with tighter tolerances. Moving the initial conditions to the center of the range is the technique that allowed the authors' problem to converge.

III. BULLDOZER BLADE

The bulldozer blade designed by Caterpillar Incorporated is a complex steel structure. It has a front blade surface and a rear surface comprised of three channels welded together to form a torsionally rigid structure. A number of castings and assemblies are welded onto the rear surface for the attachment of various control arms and cylinders. The entire structure is closed with endcaps. Figure 9 shows the front surface of the blade. Figure 10 shows the rear surface and attachments. Both of these also show the end plates. Figure 11 shows the internal channel structure of the blade. The finite element entities used to model the blade are shown in Table 3.

Table 3. Bulldozer Blade Entities

Entity	Quantity
DOF	28,324
GRID	5528
QUAD	4705
TRIA	339
HEX	472
PENTA	16
BEAM	12
ROD	3

For design analysis purposes, two loads are applied to each side of the blade in order to evaluate the structure. These two loads are described as "penetrate" and "pryout." The penetrate load condition simulates the act of pushing the blade into the ground. The pryout load simulates the simultaneous pushing and lifting of the blade. These loads simulate the cyclical usage and loading on the blade. In Caterpillar's production environment, these loads are applied with both the right and left blade corners restrained. For the purposes of this project, given the fact that the structure is symmetric except for the tag link ball and bracket, only the right corner will be constrained. The ends of the push arms and tag link arms will also be constrained. All loads are applied to the right corner of the blade, the two lift arm brackets and the point of attachment to the lift arm hydraulic cylinders. The constraints and relative magnitudes of the loads are shown in Figure 12.

In preparation for the optimization study, all physical parts of the model were given their own property identification numbers (PID's). This was done for the original analysis. Also, in order to satisfy a limitation in the optimization capabilities, all TRIA3 elements were given their own PID within each part. This was not done in the original modelling, and caused significant additional modelling time to be spent prior to the analysis. Proper PID numbering allowed the thickness of each part to be varied individually during the optimization. The thickness of each individual part was called out as a design variable through the use of the DESVAR card. The property cards (PSHELL's) were related to the design variables through the DVPREL1 bulk data input. The DVPREL1 cards allowed a single design variable to control the thicknesses of both the QUAD4's and TRIA3's. The original blade had thicknesses ranging from 8 mm to 40 mm; the thickness limits for the optimization were placed at 6 and 50 mm. In certain areas of the blade, overlapped welded plates were modelled with three different PID's, one for each of the plates, and a third for the overlapped region, whose thickness was the sum of the individual plates. This linking relationship can be easily handled by the DVPREL1 card, as it describes the thickness as a linear combination of design variables.

For this study, only the steel plates in the blade itself were considered as design variables. The parts of the blade modelled with solid elements (the push arm castings) were not included, nor were the beams representing the push arms, tilt braces, and tag link arm. The stress constraints were placed on the Maximum and Minimum Principal stresses on both the top and bottom surface of each plate, in accordance with Caterpillar practice. At the original design, there were ten elements located in five parts that violated the stress constraints.

The first run included every part of the blade. The highest stress was 144% of the allowable and was located in the tilt brace assembly at the connection to the tilt brace arm. This is an artificially high stress brought on by the modelling and loading. This high stress caused the optimizer to double the thickness of the tilt brace assembly, as well as triple the thickness of the channel underneath. This was clearly not the direction the authors had intended for the optimizer, and the run was stopped after four iterations. The maximum stress in the tilt brace assembly and the overall weight are plotted as a function of iteration number in Figure 13. It should be made clear that the optimizer was working correctly, lowering the stress in the part, but the modelling of the stress was not accurate compared to the in-service condition.

There were two lessons learned from this run. The first was to very carefully screen the responses and design constraints before submitting the run in order to avoid wasting expensive computer time on a misdirected run. The second was to submit the run for a small number of iterations at a time in order to watch the trends and convergence parameters. The authors, working on an IBM 3090 Vector Facility, were able to monitor the run in real time, and stop the run as early as possible. The ability to view the output file (.F06) during the run time is made possible through the MVS operating system, and this proved invaluable throughout the course of the project.

Prior to the second run, the design constraints were removed from the tilt brace assembly. This also required the removal of the design variable for that part. If the thickness of a part is a design variable and there are no stress constraints on that part, the optimizer would immediately reduce the thickness to zero without any consideration given to stresses. This run ran for seven iterations before an unrelated system crash ended it. In this run, all stress constraints were satisfied after one iteration, and the weight had been reduced 12%. The run had not converged yet, but the weight had not changed more than 1% from the fifth iteration. The authors considered this to be a successful run. Figure 14 shows the weight of the blade as a function of iteration for this run.

The authors were not yet satisfied without a run that fully converged. There were minor modifications made to the second run, and it was submitted again for 20 iterations. The two changes included a change in a plate thickness that was previously mistyped and the removal of the right corner of the blade as a design variable. This was removed for the same reason that the tilt brace assembly was removed prior to the second run. This run was left unattended and did not converge after 20 iterations.

There are two convergence checks made on the objective function in MSC/NASTRAN: a relative check, and an absolute check. The relative check uses the formula [3]:

$$DOBJR = \left| \frac{OBJ(p)-OBJ(p-1)}{OBJ(p-1)} \right| < CONV1$$

while the absolute check uses the formula [3]:

$$DOBJA = \left| OBJ(p)-OBJ(p-1) \right| < CONV2$$

The defaults for CONV1 and CONV2 are 0.001 and 0.01 respectively. Given a dozer blade that weighs over 3000 kg, where a 1 mm change in thickness in a given plate could change the weight of the blade 30 kilograms, it is easy to understand why the absolute convergence criteria of .01 kg change between iterations was never satisfied. The authors question why a dimensional quantity was placed the code. A new lesson was learned in taming this great capability. The CONV1 and CONV2 parameters can be changed on the DOPTPRM (Design OPTimization PaRaMeter) card.

The error described above, was corrected by setting:

$CONV2 = CONV1 * OBJ(0)$

and converged in six iterations with a reduction in weight of 9.8%. Figure 15 shows weight as a function of iteration number. The previous run of twenty iterations shows the identical results carried out to twenty iterations. Table 4 summarizes the four dozer blade runs.

Table 4. Bulldozer Blade Results

Run	Iters	ΔWeight	Comments
1	4	+23.3%	Aborted due to user errors
2	7	-12.1%	Machine error stopped run
3	20	-10.8%	Never converged
4	6	-9.8%	Successful run

The answers for this run seem reasonable. There were 22 design variables, five of which were unchanged, five of which increased in thickness, and twelve of which were reduced in thickness. The channel plate underneath the tilt brace assembly doubled in thickness, and the next largest increase was 40%, to 14 mm from 10 mm. The largest reduction in thickness was 60%.

In summary, there were many lessons learned in the dozer blade analysis. These lessons fall under three general categories:

* Model the component in anticipation of the optimization
* Understand the modelling and analysis results
* Monitor the optimizer

In the modelling phase, the PID numbering for different parts, differing element types, and different areas within parts are all critical to the success of the optimization process. Proper numbering of the above items will allow the full flexibility of the optimizer to be utilized. Proper understanding of the state of stresses and how the modelling affects those stresses will prevent the expense of wasted runs in both CPU and calendar time. Lastly, proper attention to convergence parameters will prevent long, wasted runs. A series of short runs with fewer iterations will accomplish the same objective. This is made possible by the excellent restart capability found in Version 66.

IV. HEAVY TRUCK CAB

The third model contemplated for this study was a heavy truck cab. The detailed model was comprised of 4851 grids, and was used for both static and dynamic analyses. The original goal was to optimize this structure for both the static and dynamic conditions. The dynamic capabilities were not available in MSC/NASTRAN at the time this project was performed. Further studies will be performed that will exercise this capability, and future papers will report on the findings of these studies.

V. EVALUATION OF VERSION 66

The evaluation of Version 66 is broken into two primary sections: An evaluation of the changes from Version 65 to Version 66 that were utilized during this project, and a more detailed review of Solution 200 as it was utilized during this project.

V.A. Changes From Version 65

One of the authors worked for MacNeal Schwendler at the time when Version 66 was initially being announced and clients were extremely anxious concerning the changes being made to MSC/NASTRAN. The concerns of the users included those such as increased CPU time, increased storage requirements, changes to input decks, and changes to DMAP sequences and programming. Throughout this project, the authors kept a close eye on changes to their work brought on by Version 66. In terms of CPU time, a static run on Version 66 takes no more CPU than Version 65, sometimes less. Version 66 requires 10% more database size, but no more scratch space than Version 65. For a basic superelement statics run, the only change to the input deck was to the solution sequence number. The results are identical to Version 65.

There are two new features that were useful in Version 66. The first is that the maximum use of scratch space is printed in the execution summary (day file, .F04). This feature is extremely helpful when predicting storage requirements for future work. The second relates to post-processing with OUTPUT2 files and using pre-compiled solution sequences. The use of OUTPUT2 for generating files for various post-processors has been incorporated into Version 66 through a parameter system. For post-processing in IBM/CAEDS, all the authors would have had to do was set a parameter in the bulk data:

PARAM,POST,-2

Furthermore, the particular input to the OUTPUT2 file can be controlled via parameters. For instance, the bulk data parameter NOGEOM controls whether or not geometry information will be written to the OUTPUT2 file, and the parameters OTAPE1 and OTAPE2 control the FORTRAN unit number for output. Prior to this version, the user was forced to understand a few DMAP statements in order to control these items. Unfortunately, this feature is not fully implemented yet, and the authors were forced to write a DMAP alter to correct a few minor bugs in the system. MSC has been informed of these errors, and the authors assume they will be corrected in the near future.

In Version 66, most user written alters will be constant across the structured solution sequences. This is due to major organizational changes in the DMAP, where unique analysis phases are grouped in sub-DMAP's. For example, all geometry processing is performed in a sub-DMAP called SUPER1, and all data recovery is performed in SEDRCVR (SuperElement Data ReCoVeRy). The SEDRCVR sub-DMAP is the same across all solution sequences, and is included in all sequences. Therefore, all user written post-processing alters would be the same for all solution sequences. Due to the ability to store pre-compiled DMAP sequences, any regularly used alters can be stored and retrieved transparently to the end user. For instance, a sub-DMAP which contains a commonly used alter can be stored on the user database of compiled DMAP's and recalled for use by the engineer without the engineer ever knowing what that particular DMAP or alter is.

V.B. Solution 200 Evaluation

The evaluation of the optimization capability contained in Solution 200 is divided into the following sections:

Successes
Limitations and Errors
Philosophical differences
Proposed Enhancements
Production environment concerns
Hardware issues

V.B.1. Successes

The optimization performed on the plate was accurate, and sometimes provided better results than the published optimal design. The answers achieved on the dozer blade were believable. The results and interpretation have been provided in previous sections.

Overall, the capability was understandable and easy to use. One of the biggest advantages of this capability is that it does not look like or act like the nonlinear capabilities found in Version 65. This was a fear of many users, that because optimization is an iterative process that it would behave similar to Solution 66.

The documentation was clear and concise. In fact, the *Handbook for Structural Optimization* was the only reference used during the project for optimization related information [3]. The "Introduction to Version 66" seminar notes were invaluable for Version 66 issues such as restarts, DMAP changes, DMAP alters, database terminology, solution sequence changes, and file management [4].

The restart capability built into Solution 200 worked well. It allowed for accurate restarts with minimal effort. The equation writing feature also works, and proved to be invaluable for the plate problem. This feature also allows element properties to be linked to design variables, a capability that was not exercised in this project, but is well documented in the *Handbook for Structural Optimization* [3]. There are two other types of linking that were used during the project. The first is the linking of a PSHELL thickness to multiple design variables, as in the case of the overlapped plates. The second is the linking of design variables to each other. This capability was used in some of the early plate studies to relate all the design variables to the first. The flexibility built into the program by all of these features should allow the engineer to construct his optimization problem in any combination of variables that he chooses, and is a valuable addition to the capability.

V.B.2. Limitations and Errors

There were a number of errors and limitations associated with the optimization capability. These will be discussed individually, with attention given to how these errors impacted the authors' work, as well as when MSC expects these errors to be corrected.

The first error that will be addressed is the OUTPUT2 errors. As previously stated, the parameter driven data recovery does not work correctly in Version 66. In Solution 200, this error is exaggerated. The OUTPUT2 alters exist in the SUPER1 (geometry processing) and SUPER3 (data recovery) sub-DMAPs in the structured solution sequences. Solution 200 does not use the SUPER1 sub-DMAP, which required an extensive DMAP alter to be written. Furthermore, the iteration scheme in Solution 200 will loop through data recovery on every iteration, if requested with the NASPRT bulk data parameter. This will lead to multiple copies of data blocks with the same name being written to the OUTPUT2 file. Current post-processing conversion programs will not accept this. This error is documented internally at MSC, and will be fixed in Version 66a.

Dynamics optimization does not work in Version 66. According to MSC, this will be resolved in Version 66a. Also, setting the printing parameters, P1 and P2, to anything besides the default will cause the run to fail. These parameters are found on the DOPTPRM bulk data card. MSC is also aware of this error, which will be fixed in Version 66a. According to MSC, the error concerning von Mises stress, described in the section discussing the plate problem, will be fixed in Version 67.

There is a current requirement for the user to input unique PID's for both QUAD4's and TRIA3's that belong to the same part and would ordinarily have the same PID. When asked about this, one MSC employee stated that this was due to potential changes to the element plotting codes. Since then, MSC has reported that this has been recognized as an error and will be fixed in version 66a.

The last error to be discussed refers to the printing of certain summary data on restarts. One of the nice features of the optimization capability is the printing of both the objective function and the design variable histories at the end of the run. The value of the objective function at each iteration is printed in a summary table at the end of the run, as well as the values of all of the design variables at each iteration. These tables provide concise detailed information concerning the behavior of the optimization. However, despite the excellent restart capabilities, these histories are not printed on restart runs. This run is also reported to be fixed in Version 67. An example of these summaries is shown in Figure 16.

V.B.3. Philosophical Differences

On a larger scale, there are three topics where the authors disagree with the actions or decisions made by MSC. The purpose behind the absolute convergence criteria escapes the authors as well. In a world where almost all equations have been non-dimensionalized, why intentionally place this anomaly in the code?

The last two differences are of a larger order and not restricted to the optimization capability. The first is the magnitude of spelling errors in the code output. This may be the first major release of MSC/NASTRAN in some time, but it is also the first release where errors of this kind have been apparent to the user. It sends the user a message of sloppiness and lack of attention to detail. This is hardly the attitude the authors believe that MSC wants to convey to the users.

The last is policy decision on the part of MSC's management. Throughout the course of the project, as various errors were discovered, it was also discovered that there were major changes to the DMAP code from machine type to machine. VAX Version 66 is not the same as IBM Version 66, nor is it the same as VAX Version 66a. It is expected that IBM Version 66a will be different than any of the other versions. When trying to resolve errors, the client support staff did not even have the same DMAP sequence to review as the authors. This made errors and DMAP programming extremely difficult. At one point in time, it was the philosophy of MSC to restrict changes with a release to minor user-transparent modifications, while leaving the major changes for distinct versions of MSC/NASTRAN. What happened to this policy?

V.B.4. Proposed Enhancements

Use of the optimization capability would benefit most from a pre-processor. The input can be rather tedious. At the minimum, Solution 200 requires the input of six additional bulk data input for each property card in the deck. Each PSHELL requires one DESVAR and one DVPREL1 bulk data card, as well as one DRESP1 (design response) and one DCONSTR (design constraint) bulk data card for each stress quantity of interest. This huge ratio gets even worse when property cards with more than one entry per card (PBEAMs) or second level design responses are considered. For the plate problem with 100 design variables, this eventually resulted in an additional 1000 bulk data cards added to the input deck. For the bulldozer problem, this resulted in 150 additional cards. An automated processor that would at least provide bulk data input with defaulted values to provide a framework to develop a meaningful optimization problem would make the optimization process significantly easier.

On a smaller note, there are some user conveniences that could be included in the output. The optimization capability prints a convergence test summary at the end of each iteration. These convergence summaries can be difficult to read, given their format. By the same token, it is possible to print out all of the retained responses (those deemed critical by the optimizer) at each iteration. These lists can grow very large and difficult to read for large problems. It would be a relatively easy task to flag the convergence parameters that indicate convergence, as well as to flag those retained responses that exceed the design constraint value. These additions would make the output clearer to read, and make the capability even more useful to the engineer. An example of the convergence summary is shown in Figure 17. Figure 18 shows a portion of the retained response output.

Occasionally, there is a single design response that is of particular importance to the user. There should be a relatively simple way to insure that this response is included in the retained responses. Currently, there are ways to insure this through the use of separate PID's, separate response regions, and use of the DSCREEN card, but this is somewhat tedious, and these authors did not take the time to unlock the secret to this procedure.

The authors had the opportunity to work on the IBM 3090 Vector Facility computers. This facility, with all of its other advantages, also provides the capability to view the output file during run time. This allowed the authors to review runs and cancel them as soon as an error was detected. This proved to be invaluable during the bulldozer runs. Is it possible to provide an interactive interrupt that would cancel the run and exit cleanly, thereby preserving the database? If so, this capability could eliminate large amounts of wasted CPU and clock time in this capability as well as in the nonlinear capability. Could this happen on this or any other operating system?

V.5. Optimization in a Production Environment

There are three topics to discuss when reviewing this capability for use in a production environment:

Training
Hardware requirements
General utility to the structure being analyzed

Concerning training, an engineer with five years of experience in finite element modelling, structural analysis, and finite element analysis, preferably using MSC/NASTRAN should be able to use this capability within two weeks of picking up the *Handbook for Structural Analysis* [3]. The primary skill that the engineer must have is good engineering judgement with respect to the modelling and preparation of the optimization input. The anticipated course of action should be reading all of the available material, exercising all of the available features on a well documented problem, and performing an analysis on a common structure for that engineer's department. There will naturally be situations where someone who is knowledgeable in structural optimization will be required. This person should be available to the department. MSC provides telephone "hot line" support for clients, which should be used but not relied upon for immediate answers.

There is one main feature that makes this capability much more accessible to the user than the nonlinear capabilities found in Version 65. The optimization algorithms are much more stable than the nonlinear algorithms, and the user is not expected to fully understand these in order to run the program. Much of the strength and weakness of the old nonlinear capability is based on user knowledge of the iteration techniques and the element formulations. This requirement allowed the expert user to obtain accurate solutions but excluded the average user from ever using the capability. The optimization capability does not have this limitation or requirement.

It appears that the optimization capability will be well suited to structures that are modelled with plate shell elements. The bulldozer blade was optimized to a 12% weight reduction within one week of preparing the input. The authors were able to prepare the input in two days and launch the first run on the second night. The second, third, and fourth runs were made on consecutive nights, and the analysis and post-processing were performed during the day.

During the process of setting up the optimization run, Caterpillar engineers noted that the most important parameter was the stress range between the two load cases. The two load cases represent a cyclical type of load used in fatigue calculations. The ability to write equations between subcases and create synthetic responses across subcases is not included in this version of optimization. This may limit the use of this capability in an environment where fatigue considerations are critical to the success of the design and the analysis.

The original purpose behind the cab optimization was to optimize the mode shapes for acoustic purposes. This is currently not possible within MSC/NASTRAN, and the project was reduced to a eigenvalue optimization, which will be feasible with the next version of MSC/NASTRAN. Again, this is a severe limitation that will limit optimization until it is removed. Full utilization of optimization will not be achieved until these capabilities are included in the code.

The computer runs associated with the optimization required a significant amount of CPU time. Table 5 shows the CPU time required for the baseline statics runs, as well each of the optimization runs. It is anticipated that a typical large scale optimization run will require somewhere in the neighborhood of ten times the CPU of the statics run. These runs were made on an IBM 3090 Vector Facility, which is roughly 60 times faster than a VAX 11/780. Memory resident scratch files were used during these runs in order to reduce I/O times. This enabled the authors to reduced the elapsed time of the runs by one half. These runs did not take advantage of the parallel processing available to the 3090, which would reduce the matrix decomposition elapsed times in proportion to the number of CPU's utilized. These runs were also made on a production system, and at no time was the machine dedicated to the optimization runs. The efficiency of the machine is primarily due to the excellent memory management algorithms implemented on the MVS/XA operating system. The data base size required is the same as that of the static run for large problems, and the scratch space required is ten times that of the static run.

Table 5. Hardware Requirements

	Plate		Dozer	
	Static	Optimized	Static	Optimized
CPU[1]	9	1958	757	7606
DB Space[2]	217	301	29,772	30,337
Scratch[2]	35	158	391	2,957

[1] Seconds on an IBM 3090

[2] MSC/NASTRAN blocks

VI. CONCLUSIONS

Solution 200 is accurate, relatively easy to use, but can be extremely CPU intensive. It is the authors opinion that the optimization capability is a worthwhile addition to MSC/NASTRAN, and a worthwhile addition to the complement of tools that every analysis engineer should have at his disposal, provided the company has the proper computers to support the engineer. The IBM 3090 Vector Facility was ideal for supporting the authors in this project.

VII. ACKNOWLEDGEMENTS

AAC wishes to express its thanks to Caterpillar for their time, attention, assistance, and patience throughout the course of this project. AAC would also like to thank the IBM Corporation for the use of their facilities, and thank MSC's client support, development, and regional marketing staffs for their technical assistance during this project.

REFERENCES

[1] Haug, E.J., and Arora J.S., *Applied Optimal Design*, 1979, John Wiley and Sons.

[2] Gockel, M., *MSC/NASTRAN Users' Manual*, 1985, The MacNeal-Schwendler Corporation.

[3] Miura, H., *MSC/NASTRAN Handbook for Structural Optimization*, 1988, The MacNeal-Schwendler Corporation.

[4] Swan, J., *Introduction to MSC/ NASTRAN Version 66*, 1989, The MacNeal-Schwendler Corporation.

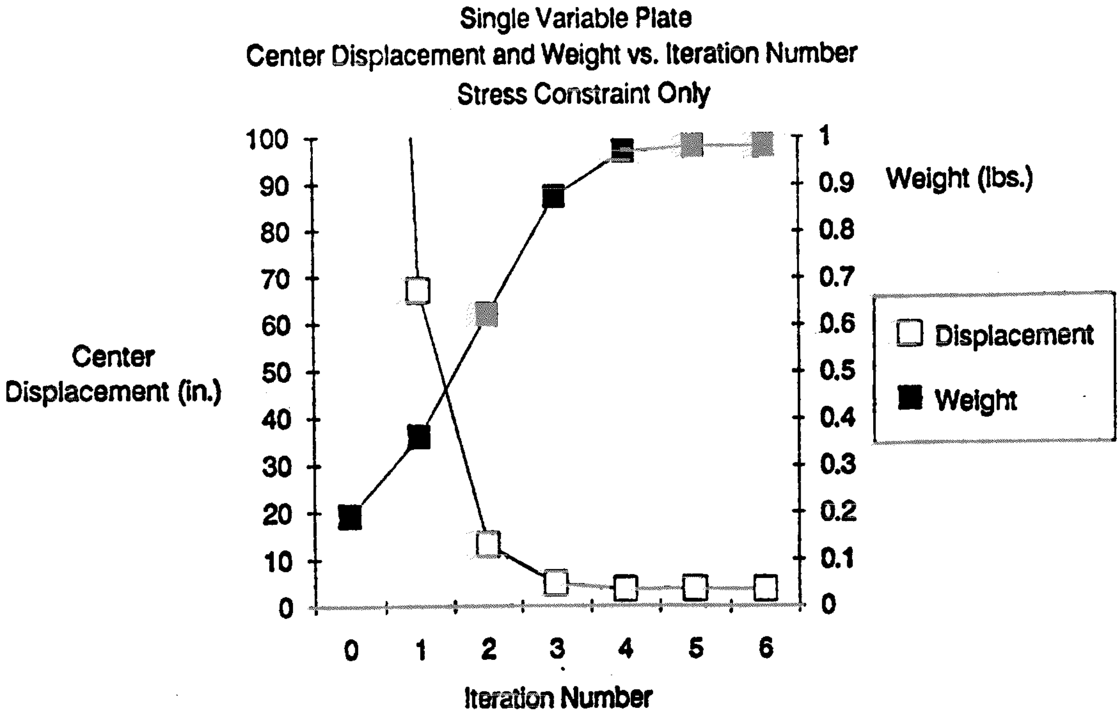


Figure 1.

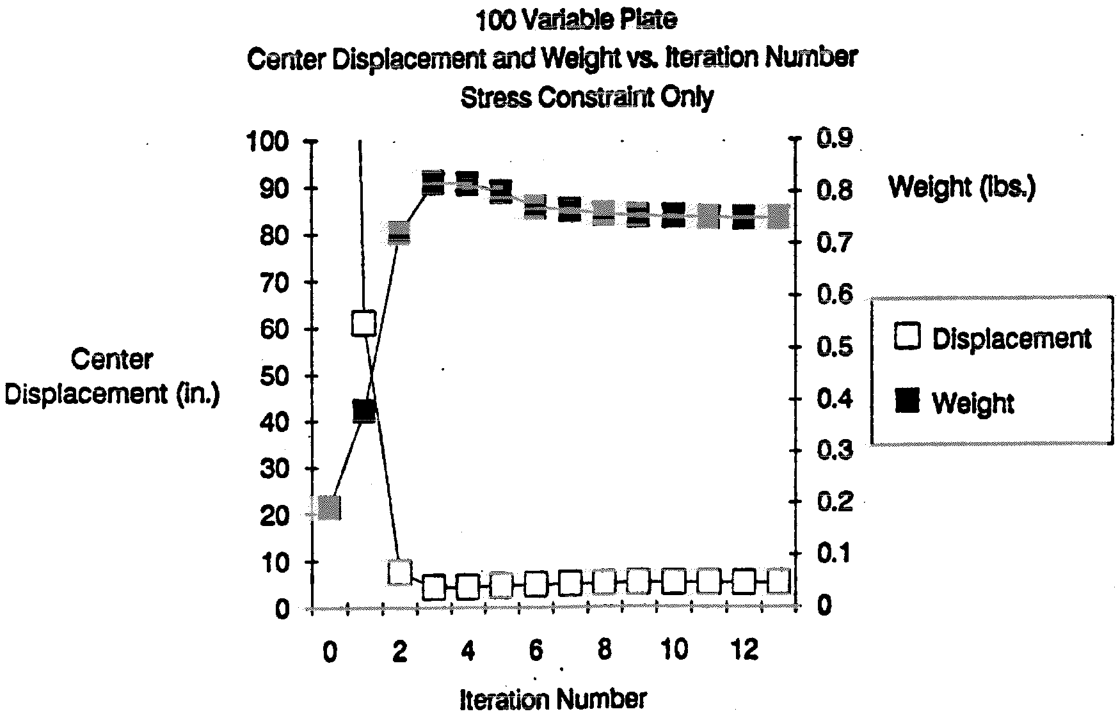


Figure 2.

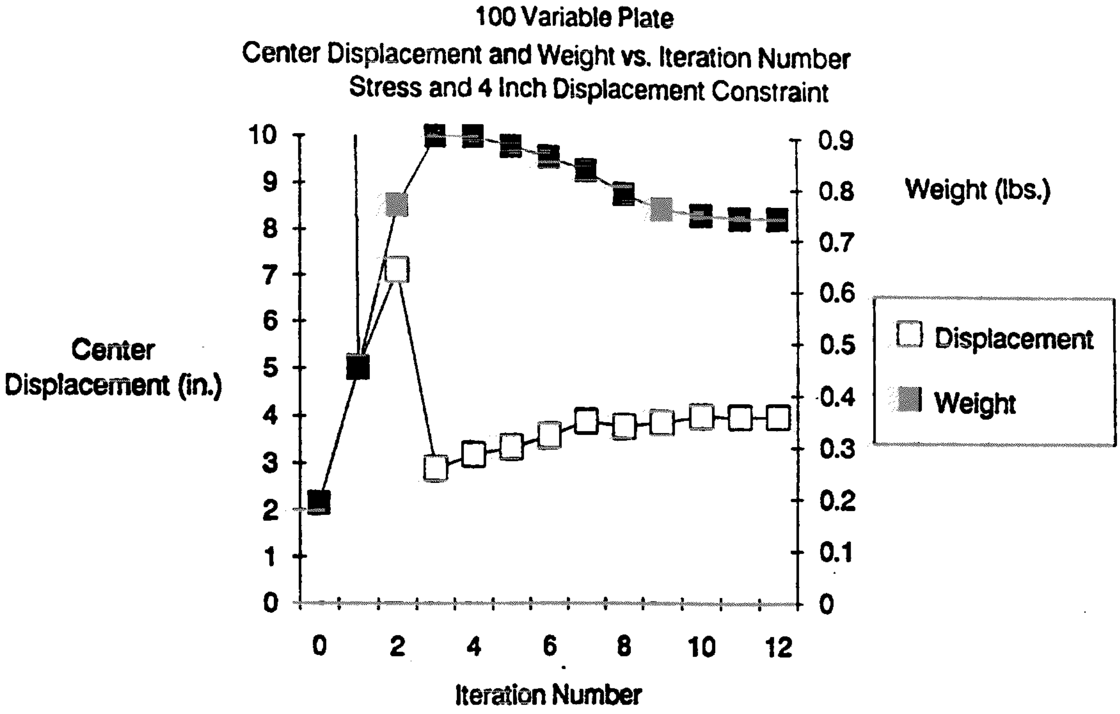


Figure 3.

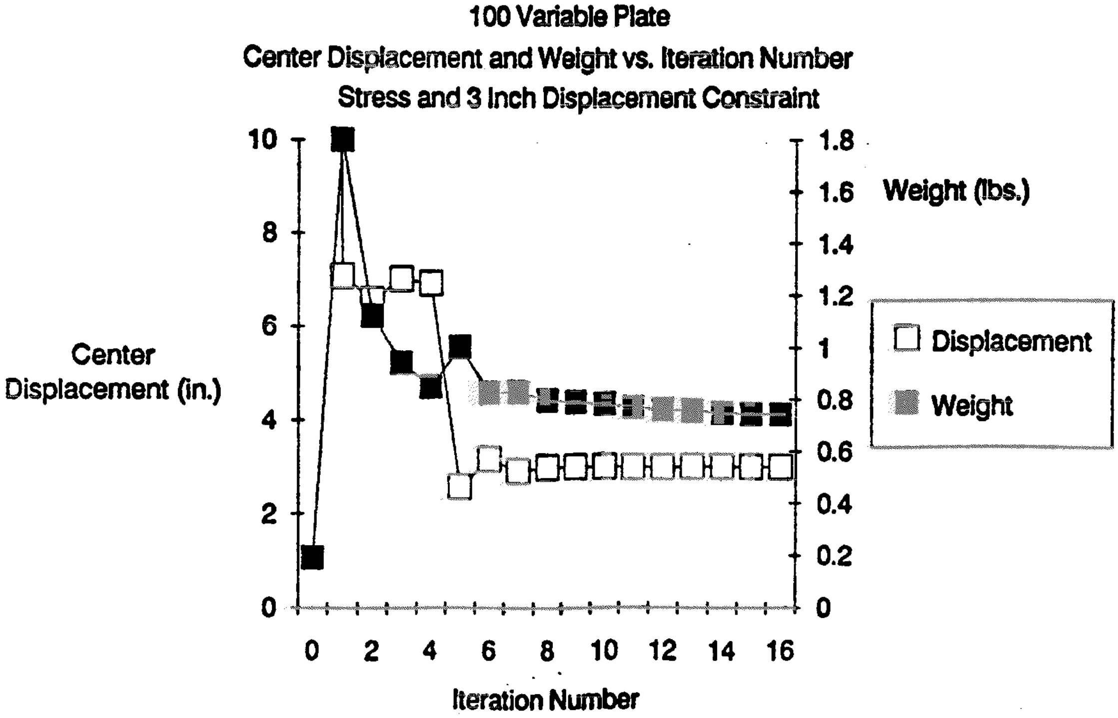
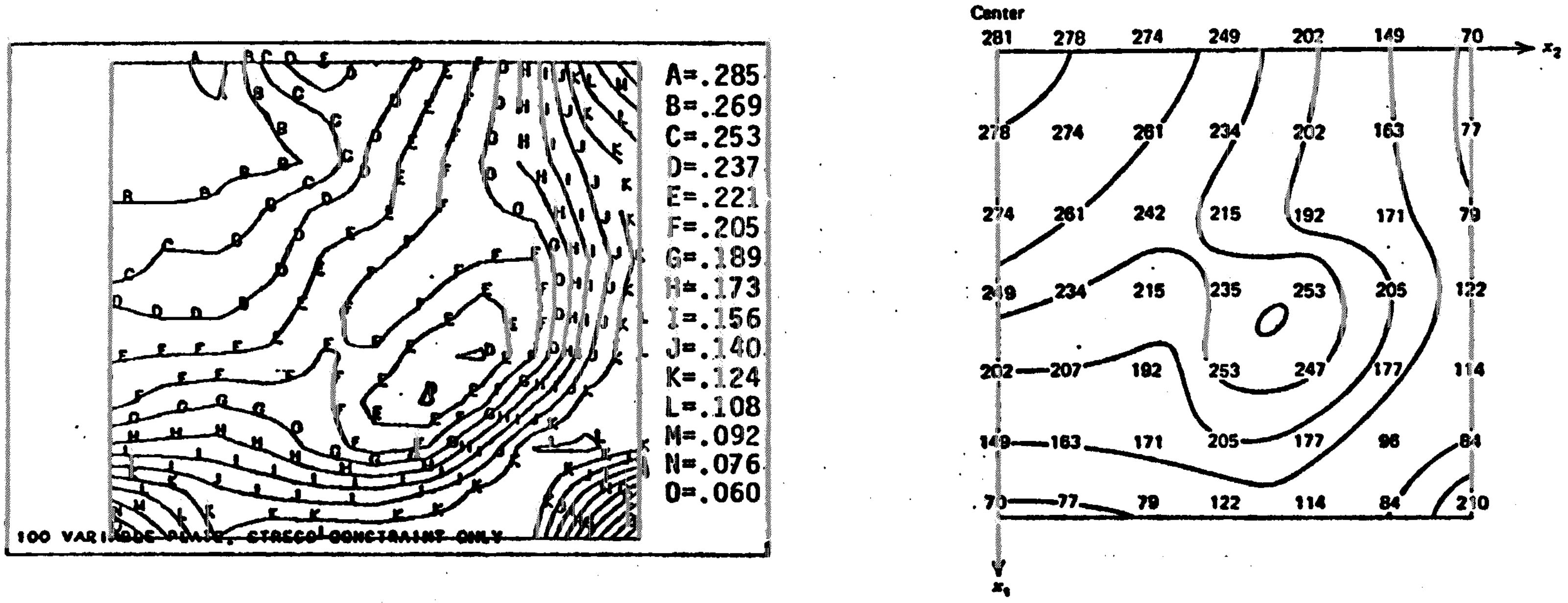


Figure 4.



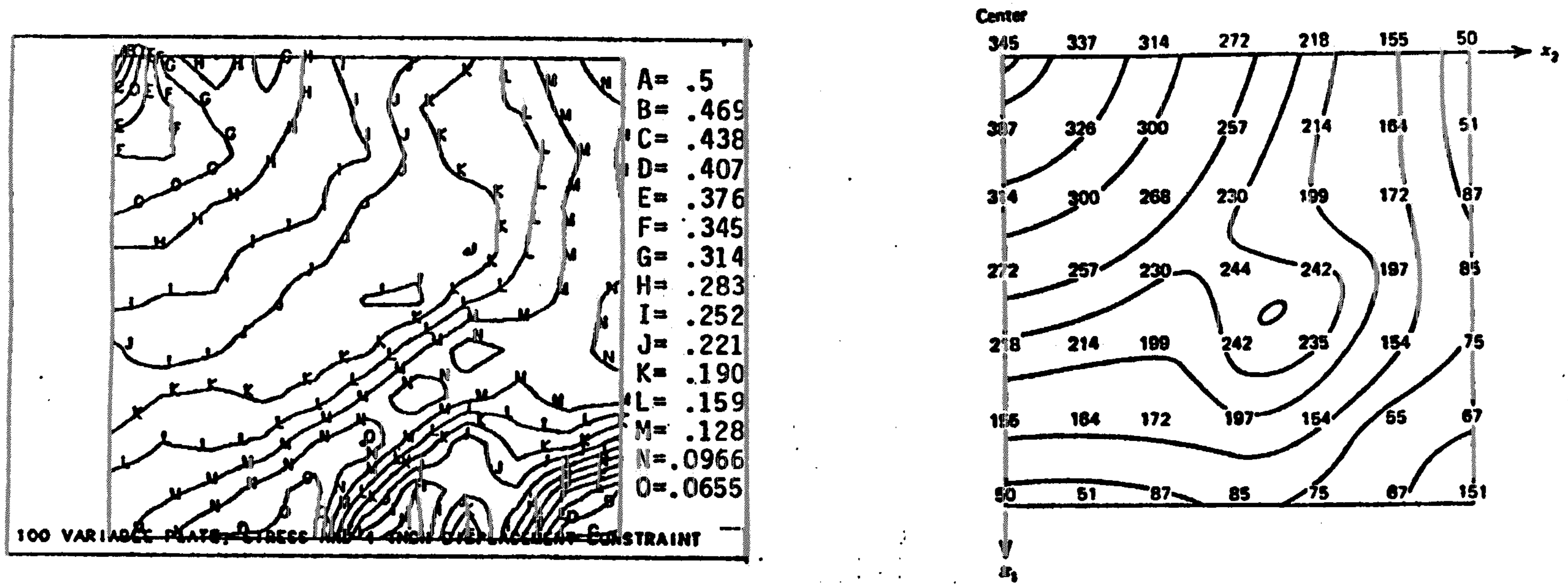
a. Calculated

b. Published

Thickness Contours

100 Variable Plate, Stress Constraint Only

Figure 5.



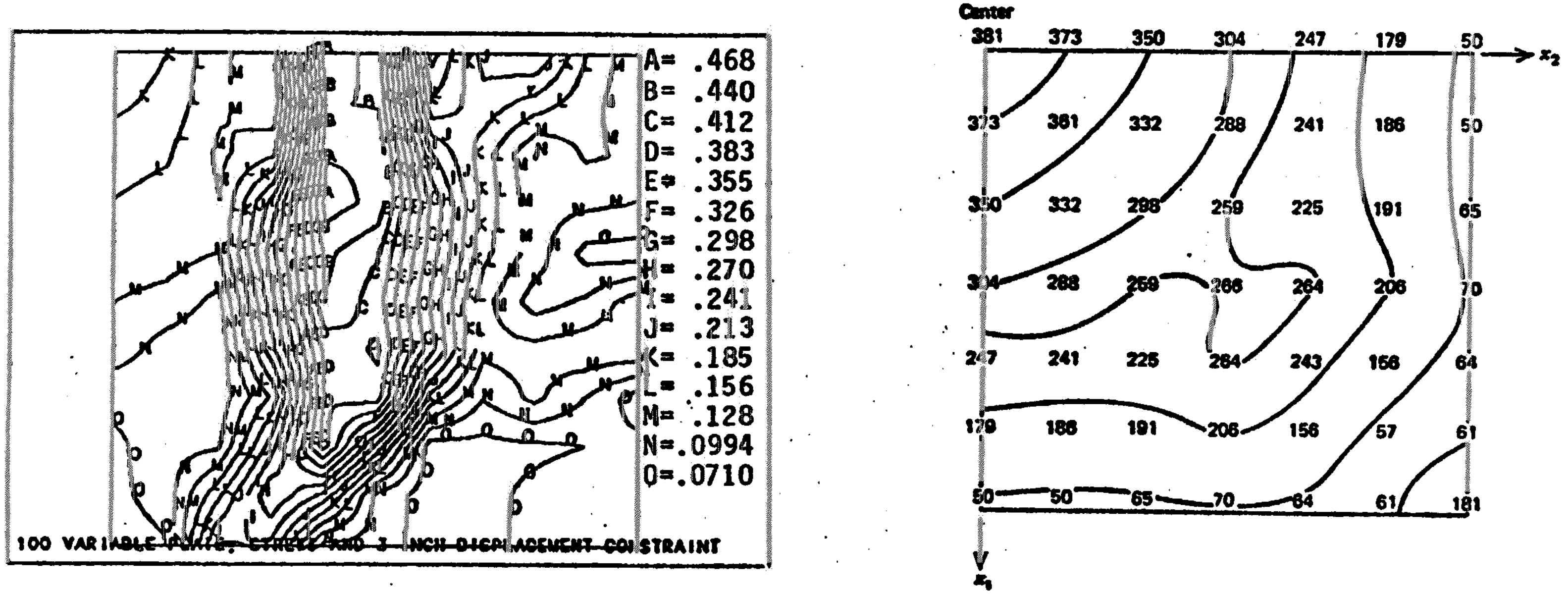
a. Calculated

b. Published

Thickness Contours

100 Variable Plate, Stress and 4 Inch Displacement Constraint

Figure 6.



a. Calculated

b. Published

Thickness Contours

100 Variable Plate, Stress and 3 Inch Displacement Constraint

Figure 7.

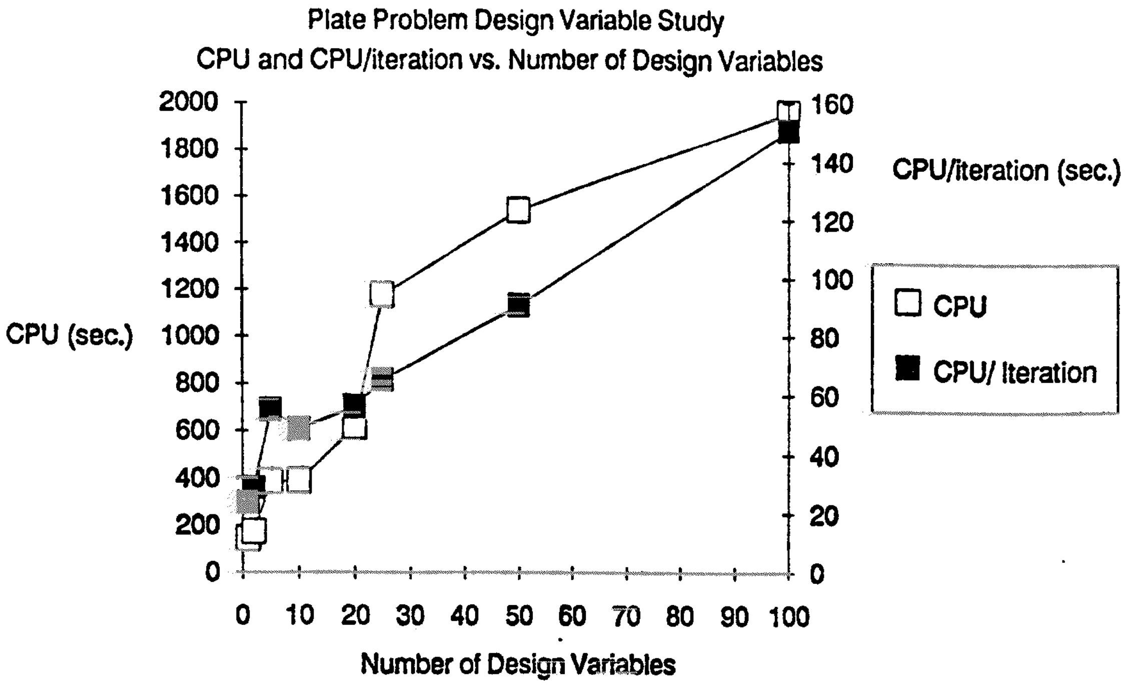
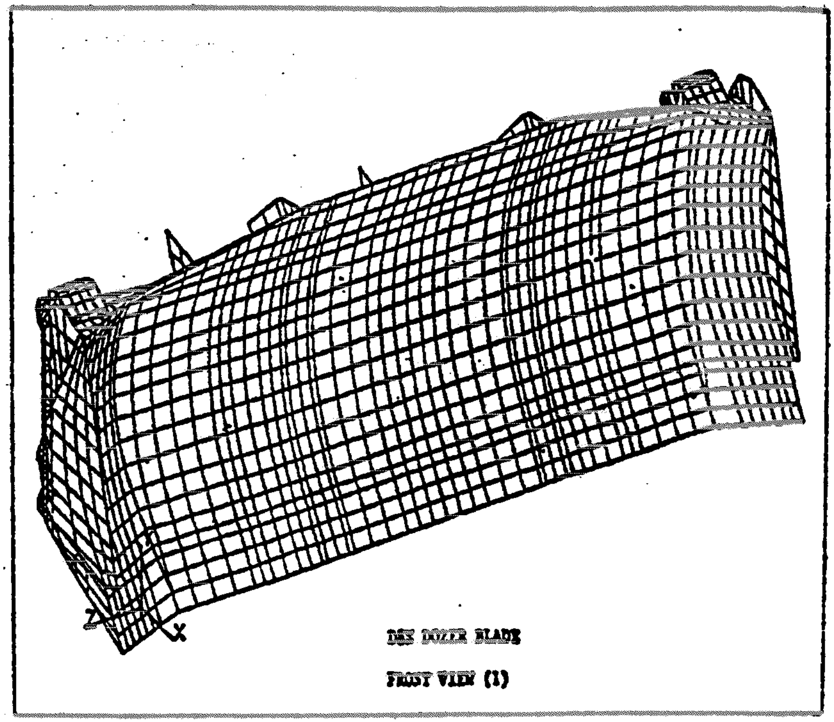
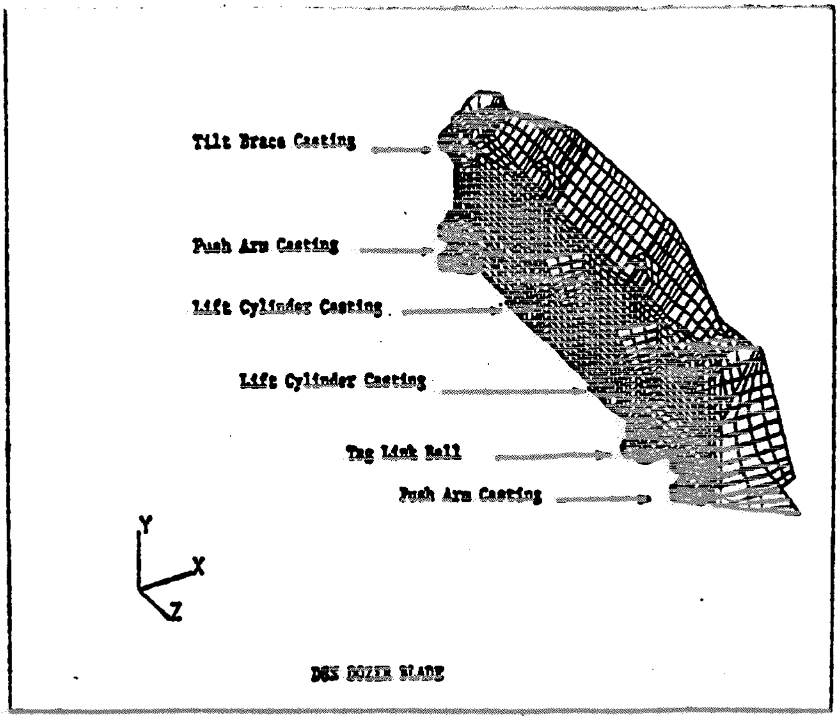


Figure 8.



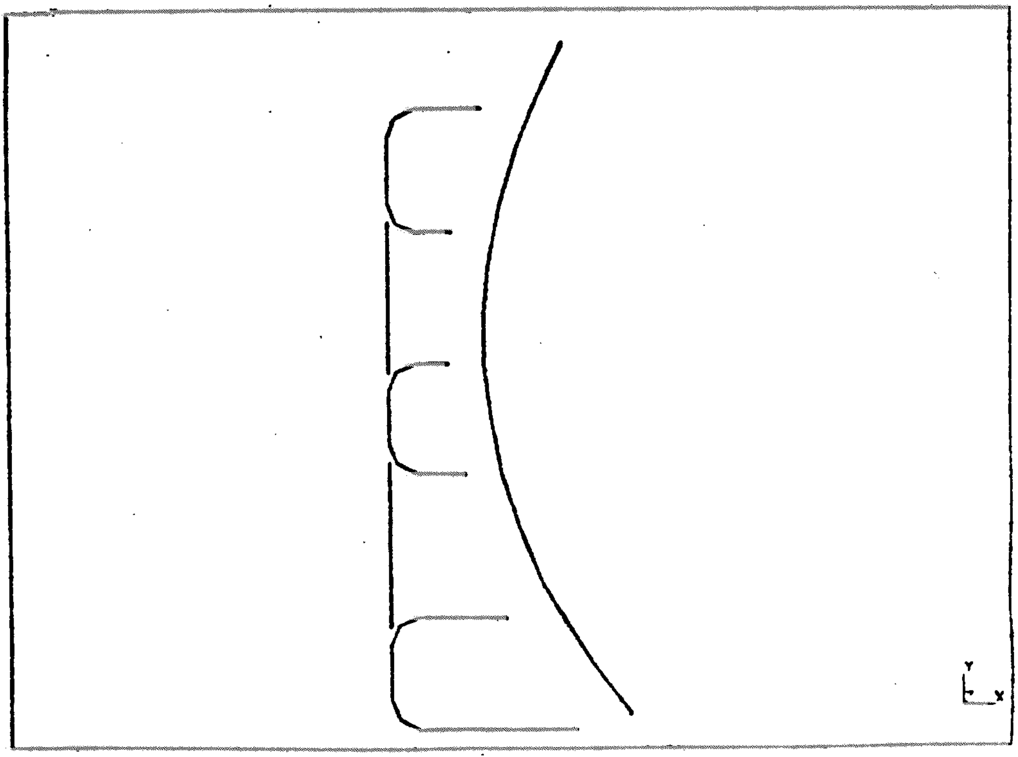
Bulldozer Blade, Front View

Figure 9.



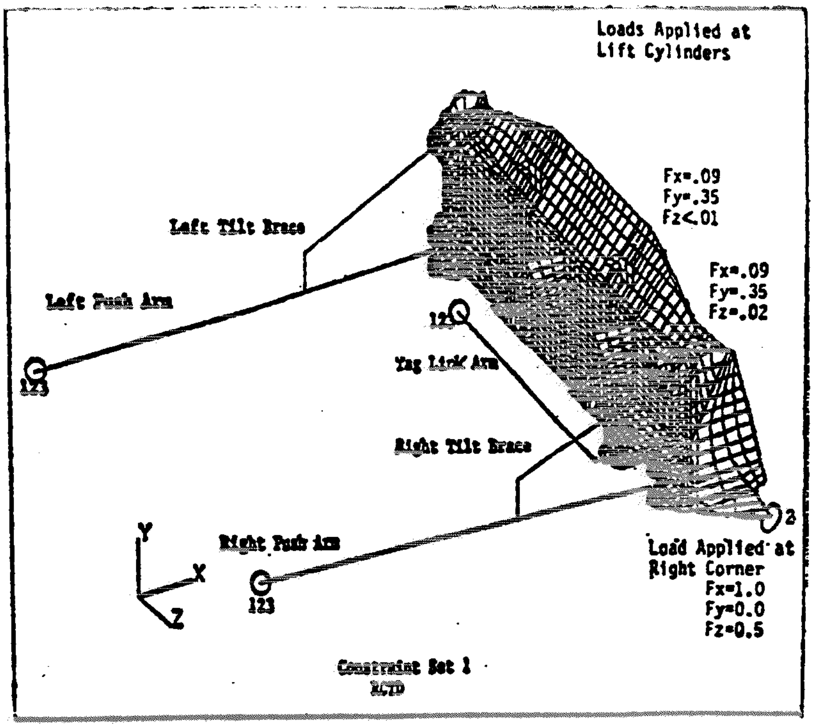
Bulldozer Blade, Rear View

Figure 10.



Bulldozer Blade, End View Exploded

Figure 11.



Bulldozer Blade, Constraints and Loads

Figure 12.

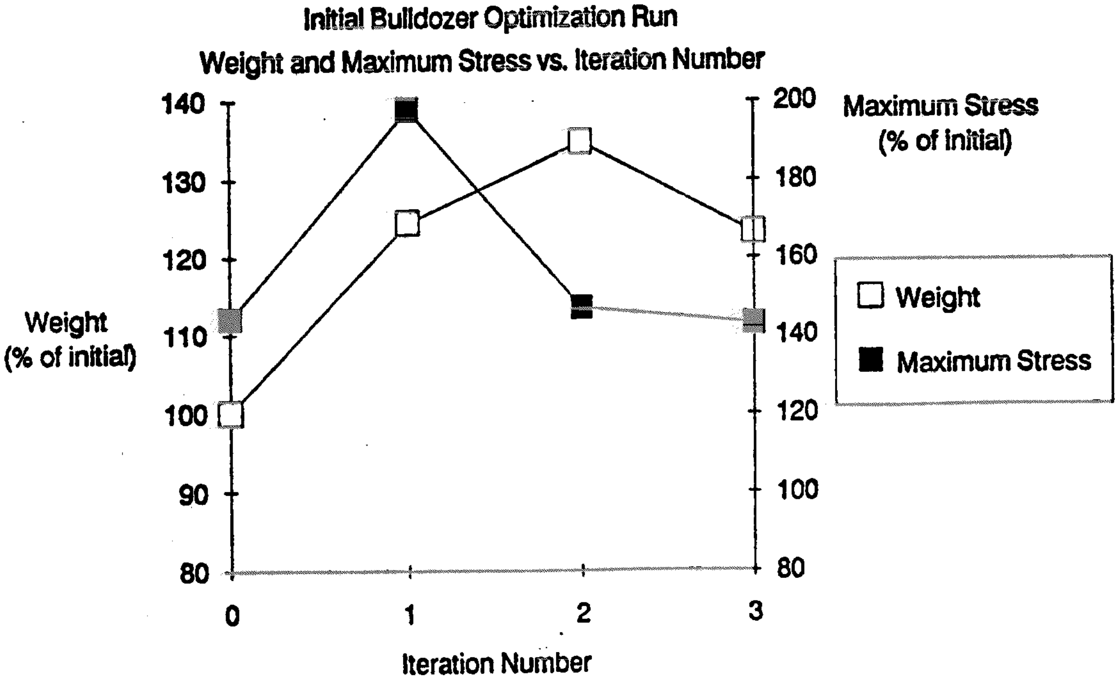

Initial Bulldozer Optimization Run
Weight and Maximum Stress vs. Iteration Number

Figure 13.

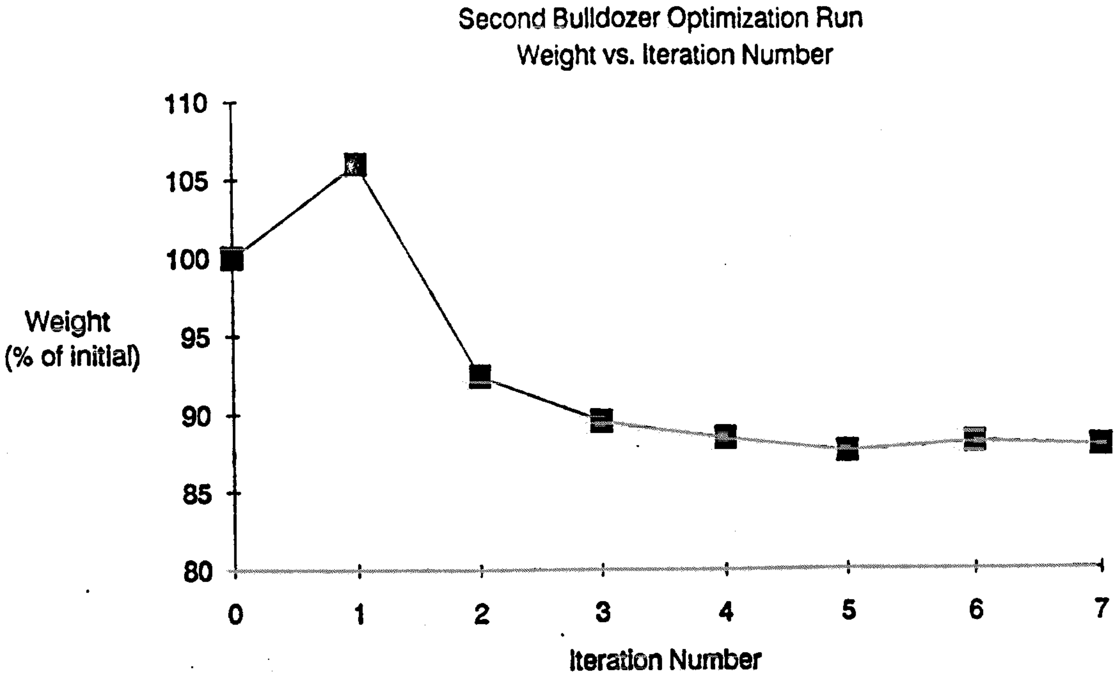


Figure 14.

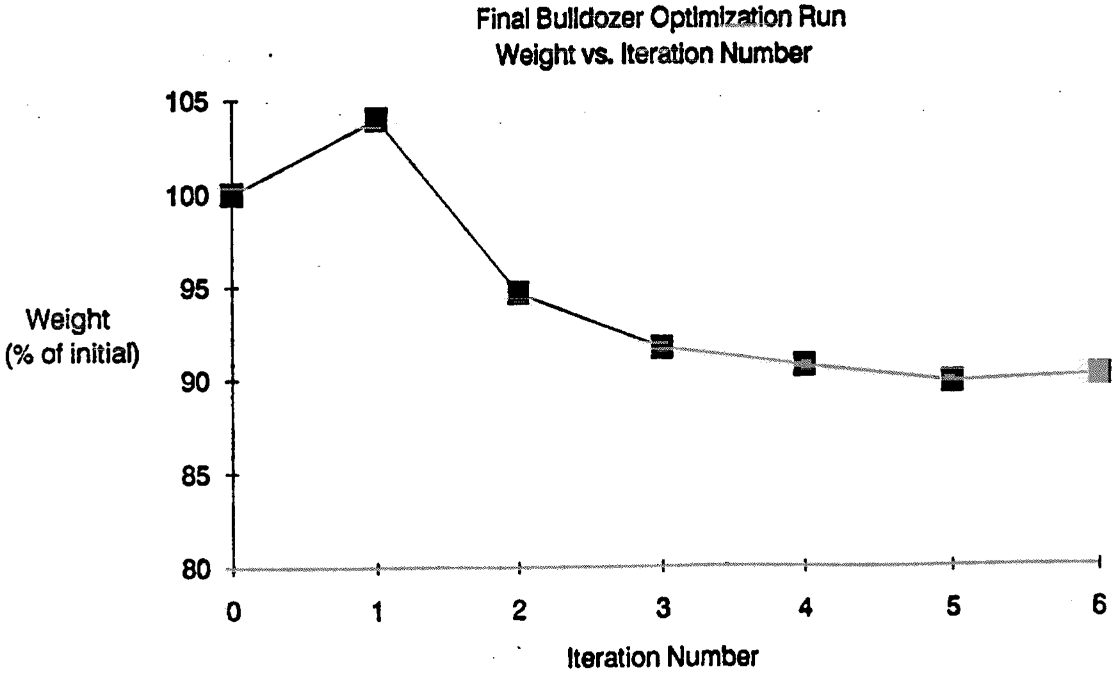


Figure 15.

```
                              ********************************************************
                              S U M M A R Y   O F   I T E R A T I O N   H I S T O R Y
                              ********************************************************

                                           (HARD CONVERGENCE ACHIEVED)

                                           (SOFT CONVERGENCE ACHIEVED)


                               NUMBER OF FINITE ELEMENT ANALYSES COMPLETED          7
                               NUMBER OF OPTIMIZATIONS W.R.T. APPROXIMATE MODELS    6


                                          OBJECTIVE FUNCTION HISTORY
             ------------------------------------------------------------------------------------------
             ITERATION   OPTIMAL WITH RESPECT    EXACT EVALUATION     FRACTIONAL ERROR     MAXIMUM VALUE OF
              NUMBER      TO APPROXIMATION     BY COMPLETE ANALYSIS   OF APPROXIMATION       CONSTRAINTS
             ------------------------------------------------------------------------------------------

              INITIAL                             0.481462E-01                               0.250906E+02

                1           0.902638E-01          0.902624E-01          0.158484E-04         0.641777E+01

                2           0.155574E+00          0.155576E+00         -0.141755E-04         0.149207E+01

                3           0.219133E+00          0.219137E+00         -0.165918E-04         0.252609E+00

                4           0.242770E+00          0.242770E+00          0.736557E-06         0.193229E-01

                5           0.245009E+00          0.245006E+00          0.973112E-05         0.657443E-03

                6           0.245006E+00          0.245006E+00          0.243278E-06         0.657769E-03
             ------------------------------------------------------------------------------------------


                                          DESIGN VARIABLE HISTORY
------------------------------------------------------------------------------------------------------------------
: DV. ID. :   INITIAL   :      1      :      2      :      3      :      4      :      5      :      6      :      7
------------------------------------------------------------------------------------------------------------------
    1     :  0.5000E-01 :  0.9374E-01 :  0.1616E+00 :  0.2276E+00 :  0.2521E+00 :  0.2544E+00 :  0.2544E+00 :
------------------------------------------------------------------------------------------------------------------
```

Objective and Design Variable Histories

Figure 16.

```
   CONVERGENCE NOT ACHIEVED YET
..............................................................................

 NUMBER OF ITERATION LOOP(S) COMPLETED  :    3
 MAXIMUM NUMBER OF ITERATIONS REQUESTED :   20
 MAXIMUM VIOLATION OF CONSTRAINTS       :  0.2526E+00  MUST BE LESS THAN   0.5000E-02
 RELATIVE CHANGE IN OBJECTIVE           :  0.2901E+00  MUST BE LESS THAN   0.1000E-02
 ABSOLUTE CHANGE IN OBJECTIVE           :  0.6356E-01  MUST BE LESS THAN   0.1000E-01
 MAXIMUM OF RELATIVE DV CHANGES         :  0.4085E+00  MUST BE LESS THAN   0.1000E-02
 MAXIMUM OF RELATIVE PROP CHANGES       :  0.4085E+00  MUST BE LESS THAN   0.1000E-02
..............................................................................
```

Convergence Summary

Figure 17.

```
                       -----   RETAINED 2ND LEVEL CONSTRAINTS AT THE   FINAL DESIGN  -----

..........................................................................................................
   RESPONSE     RESPONSE     EQUATION       LOWER            APPROX.          UPPER           INTERNAL
      ID          LABEL         ID          BOUND             VALUE           BOUND          REGION ID
..........................................................................................................
      401       TOPVON         502      -3.600000E+04     2.975693E+04     3.600000E+04          1
      401       TOPVON         502      -3.600000E+04     3.075847E+04     3.600000E+04          1
      401       TOPVON         502      -3.600000E+04     1.542598E+04     3.600000E+04          1
      401       TOPVON         502      -3.600000E+04     1.979329E+04     3.600000E+04          1
      401       TOPVON         502      -3.600000E+04     3.016812E+04     3.600000E+04          1
```

Retained Responses

Figure 18.