DESIGN ENHANCEMENT TOOLS
IN MSC/NASTRAN

D. V. Wallerstein
The MacNeal-Schwendler Corporation
Los Angeles, California

DESIGN SENSITIVITY IN OPTIMIZATION

Design sensitivity (ref. 1) is the calculation of derivatives of constraint functions with respect to design variables. While a knowledge of these derivatives is useful in its own right, the derivatives are required in many efficient optimization methods. Constraint derivatives are also required in some reanalysis methods. Figure 1 shows where the sensitivity coefficients fit into the scheme of a basic organization of an optimization procedure (ref. 2). In the context of this paper the analyzer is to be taken as MSC/NASTRAN. The terminator program monitors the termination criteria and ends the optimization procedure when the criteria are satisfied. This program can reside in several places: in the optimzer itself, in a user written code, or as part of the MSC/EOS (Engineering Operating System) currently under development. Since several excellent optimization codes exist and since they require such very specialized technical knowledge, the optimizer under the new MSC/EOS is considered to be selected and supplied by the user to meet his specific needs and preferences. The one exception to this will be a fully stressed design (FSD) based on simple scaling. The gradients are currently supplied by various design sensitivity options now exisiting in MSC/NASTRAN's design sensitivity analysis (DSA).

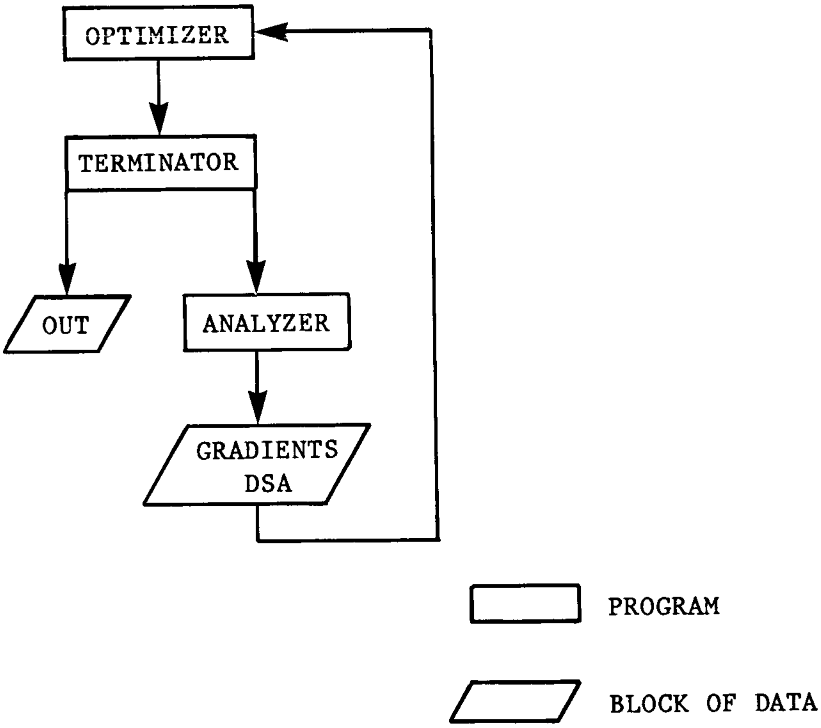


Figure 1

DESIGN CONSIDERATIONS - USER ISSUES

Figure 2 outlines several implementation issues that were considered for the current MSC/DSA and are still valid for future enhancements. From a user standpoint, the computations of the gradients should be a natural extension of the finite element analysis (FEM). At the same time, the user should not be constrained by the FEM to the degree that he or she is forced to make unwanted adaptations of design requirements. To this end, MSC/DSA has been designed to have generality in modeling complex designs and to allow for design variables which are not explicitly defined in terms of standard finite element properties such as area, second moments of area, and thickness. By the same token, MSC/DSA must not and does not impose any size restrictions on the analysis. On the other hand, through its ability to link many finite element property cards and material cards into a single design variable, MSC/DSA allows for computational efficiency by reducing the number of design variables needed for analysis. The output is in two forms. A matrix is generated with both the current value of the constraint and the values of the gradients output to the data base and to a general FORTRAN file. To aid the user in interpreting the results, this same information is output in table form with both numeric and user supplied BCD identifiers.

- EASE OF USE

- GENERALITY TO MODEL COMPLEX DESIGNS

- CAPACITY TO SOLVE LARGE PROBLEMS

- OUTPUT EASY TO COMPREHEND

Figure 2

DESIGN CONSIDERATIONS - ANALYSIS ISSUES

Figure 3 gives the three prime issues considered in selecting the approach used for MSC/DSA. The general philosophy followed was that MSC/DSA be considered a post-processor to a standard MSC/NASTRAN analysis. This requirement was easily met through use of data-basing. Through the data base the LU decomposition of the stiffness matrix is recovered so that only backward operations are used to recover the solution vector needed to compute gradient terms. An important feature of the implementation is the table driven nature of MSC/DSA. Final recovery of all information needed to form gradients is handled by standard MSC/NASTRAN stress, force, displacement, or modal recovery modules. This means that the only new features added to MSC/NASTRAN are those which form the necessary correlation tables between constraints and their derivatives and the needed code to form the right hand side needed for solution vector recovery. The net result is ease and reliability in system maintenance.

- RESTART FROM PRIMARY ANALYSIS

- ONLY BACKWARD OPERATIONS USED TO RECOVER SOLUTION VECTOR

- DATA ORGANIZATION, ASSEMBLY AND RECOVERY

Figure 3

DEFINITION OF TERMS

Figure 4 lists the meaning of some terms used in the following discussion. It is appropriate, however, to mention two things at this point. First, MSC/DSA does not impose any constraint equations. It merely returns the value of the constraint at the current point in design space. Second, the approach is the design space approach and not the adjoint method. The adjoint method is discussed only for comparison purposes. The two methods arise from the technique used to determine the value of the second term in the expression for $\delta\Psi_i$.

$\{b\}$ - Vector of Design Variables

$\{u\}$ - Vector of Displacements

$\{P\}$ - Vector of Loads

$[K]$ - Structural Stiffness

$\{\xi_i\}$ - Vector of Adjoint Variables

$$[H] = \left[\frac{\partial P}{\partial b}\right] - \left[\frac{\partial K}{\partial b}\, u\right]$$

$\Psi_i(b, u) \leq 0$ - Constraint Equation

$$\delta\Psi_i = \left\{\frac{\partial \Psi_i}{\partial b}\right\}^T \{\delta b\} + \left\{\frac{\partial \Psi_i}{\partial u}\right\}^T \{\delta u\} \text{ - Sensitivity Coefficient}$$

$B = b/b_o$ - Normalized Design Variable

Λ_{ij} - Sensitivity Coefficient for ith Constraint and jth Design Variable

Figure 4

SENSITIVITY METHODS

As mentioned in the previous figure, the difference between the adjoint method - sometimes called behavior space or state space and the design space method is how the $\{\delta u\}$ term is replaced by $\{\delta b\}$ in the expression for $\delta\Psi_i$. These two methods are depicted in figure 5. Notice that both require LU decompositions and formation of the $[H]$ matrix. The effect of the right hand sides is discussed in the next figure.

$$\{\delta u\} \longleftarrow \{\delta b\}$$

ADJOINT METHOD

$$[K]\{\xi_i\} = \left\{\frac{\partial\Psi_i}{\partial u}\right\}$$

$$\left\{\frac{\partial\Psi_i}{\partial u}\right\}^T\{\delta u\} = \{\xi_i\}^T[H]\{\delta b\}$$

DESIGN SPACE METHOD

$$[K]\left\{\frac{\partial u}{\partial b}\right\} = [H]$$

$$\{\delta u\} = \left\{\frac{\partial u}{\partial b}\right\}^T\{\delta b\}$$

Figure 5

THEORY COMPARISONS OF TWO APPROACHES

Figure 6 gives a direct comparison of the two methods. The adjoint method uses the $\{\partial\Psi_i/\partial u\}$ vector to generate the right hand side. Only active constraits are used to generate this vector; hence, the number of unknown equations at any given point in design space is equal to the number of active constraints. In the design space method the right hand side is represented by the matrix $[H]$. This matrix is a true pertubation of the load vector for a change in each design variable and represents the perturbed equilibrium state of the structure. There is a column for each load vector perturbed by a design variable.

Approach	Adjoint Method	Design Space
Unknowns	$\{\xi_i\}$	$\left[\frac{\partial u}{\partial b}\right]$
Set of equations to be solved	$[K]\{\xi_i\} = \left\{\frac{\partial \Psi_i}{\partial u}\right\}$	$[K]\left[\frac{\partial u}{\partial b}\right] = [H]$
Number of unknown vectors to be determined	Number of active constraints	Number of design variables times Number of load conditions

Figure 6

IMPLEMENTATION COMPARISON OF TWO METHODS

Figure 7 compares the two methods from a different point of view. Generally, discussion of which method to use stems from the number of right hand side vectors. But such discussion rarely takes into consideration the efficiency of modern equation solvers. When this efficiency is taken into consideration the number of unknowns to solve for takes on less significances than other considerations. Two major advantages of the design space approach have already been alluded to. First, the [H] matrix represents a perturbation of equilibrium and its formulation fits neatly into an existing finite element code (also it must be formed in the adjoint method) and forms a natural load vector. Second, the formulation of the gradients fits in naturally with existing general purpose finite element code data recovery operations. Finally, a major expense in either method is the forming of the necessary correlation tables between constraints and design variables and the forming of the correlation between design variables and individual element properties.

Item	Adjoint Method	Design Space
Few active constraints	Small advantage	--
Many active constraints Few design variables	--	Small advantage
Forming right hand side using existing code structure	--	Major advantage
Forming gradients using existing code structure	--	Major advantage
Forming necessary correlation tables relating element variables to design variables and response variables to constraints	Major expense	Major expense

Figure 7

THE DESIGN SPACE INCREMENTAL APPROACH

Figure 8 shows the necessary incremental equations required for the design space approach. Notice that perturbation to nodal equilibrium occurs from two sources, a change in stiffness arising from design variable perturbation of the element stiffness matrices and a change in the actual load vector arising from the same source. Also note that for accuracy and consistancy a total solution vector not an incremental one is formed.

$$[K^o]\{\Delta u_B\} = \{\Delta P_B\} - [\Delta K_B]\{u^o\}$$

$$\{\Delta P_B\} = \{\Delta P_{temp}\} + \left\{[\Delta M_B]\{\text{Rigid-Body Acceleration}\}\right\}$$

$$\{u\} = \{u^o\} + \{\Delta u_B\}$$

Figure 8

SOME MORE DEFINITIONS

Figure 9 contains some definitions unique to this paper. These definitions are mostly self explanatory.

Q - MSC/NASTRAN OUTPUT SUCH AS:

- Displacement
- Stress
- Force

Q^B - OUTPUT VALUE USING NEW PROPERTY ORIGINAL SOLUTION VECTOR

Q^U - OUTPUT VALUE USING ORIGINAL PROPERTY NEW SOLUTION VECTOR

Q^O - OUTPUT VALUE USING BASE LINE RUN

Figure 9

ACTUAL GRADIENT COMPUTATION

Figure 10 gives the actual gradient computation as used by the design space approach. The top equation is used for "element" type constraints (meaning element stress or force) with self terms. Self terms arise when a specific element is used as a constraint and one or more of its physical properties (such as area, second moment of area, etc.) is included in a design variable and the derivative of the constraint with respect to that specific design variable is required. The bottom equation is used for "element" type constraint without self terms and displacement type constraints. The incremental change in design variable is represented by ΔB_j and Q_{Lim_i} are user supplied limit values used in constraint evaluations. The upper signs are associated with maximum type constraints and the lower signs are associated with minimum type constraints. The various values of Q come directly from stress, force, or displacement output files standard to MSC/NASTRAN.

$$\Lambda_{ij} = \pm \frac{Q_i^{B_j}}{\left\lceil Q_{Lim} \right|_i \Delta B_j} \mp \frac{Q_i^o}{\left\lceil Q_{Lim} \right|_i \Delta B_j} \pm \frac{Q_i^{u_j}}{\left\lceil Q_{Lim} \right|_i \Delta B_j} \mp \frac{Q_i^o}{\left\lceil Q_{Lim} \right|_i \Delta B_j}$$

or

$$\Lambda_{ij} = \pm \frac{Q_i^{u_j}}{\left\lceil Q_{Lim} \right|_i \Delta B_j} \mp \frac{Q_i^o}{\left\lceil Q_{Lim} \right|_i \Delta Bj}$$

Figure 10

CURRENT USER INTERFACE

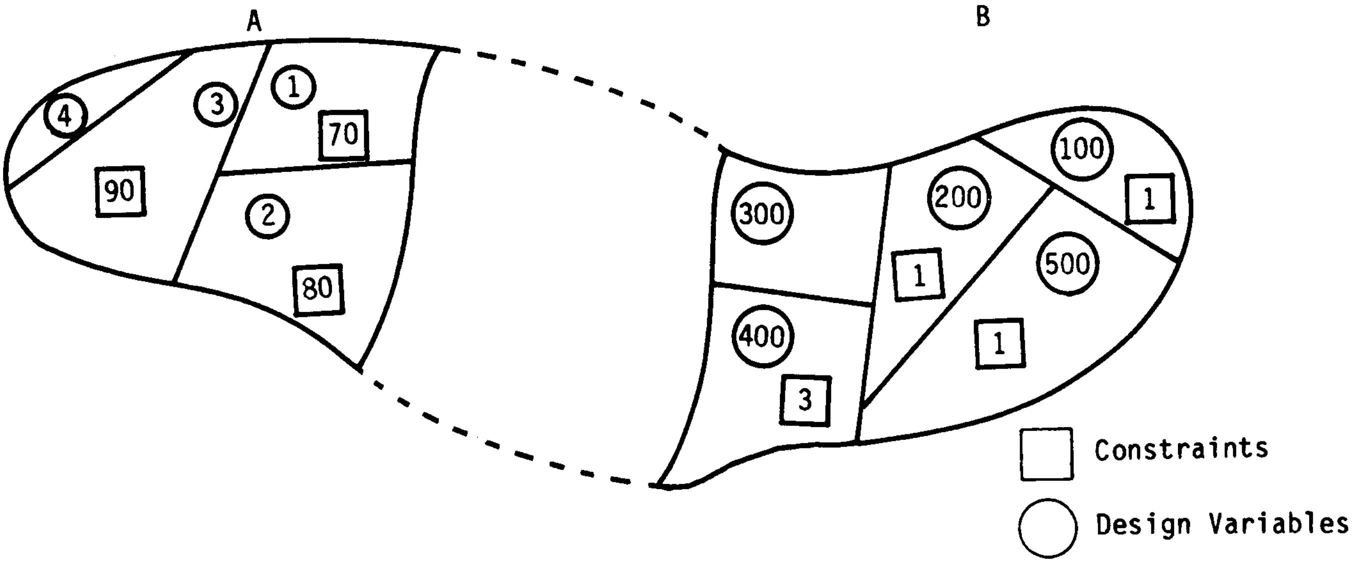
Figure 11 gives the current user interface scheme. MSC/DSA is currently designed to run as a post-processor to the standard MSC/NASTRAN static solution sequence (SOL 61), modal solution sequence (SOL 63), and buckling solution sequence (SOL 65). The MSC/DSA solution sequence numbers correspond to each of the basic solution sequences. To aid in relating specific constraints to specific design variables two new case control cards have been defined. Constraints are defined via the DSCØNS cards and design variables are defined via the DVAR-DVSET cards.

SOL 51 SOL 53 SOL 55	EXECUTIVE
SENSITY SET2	CASE CONTROL
DSCONS DVAR DVSET	BULK DATA

Figure 11

RELATING CONSTRAINTS TO DESIGN VARIABLES

The user may be interested in the derivatives of every constraint with respect to every design variable. If this is so, just set SENSITIVITY = ALL. On the other hand, it may be more reasonable in analysis to relate specific constraints to specific design variables as depicted in figure 12. In this figure, structure located at A is considered far enough removed from structure located at B that derivatives of constraints at A with respect to design variables at B (or vice versa) would be meaningless. The SET, SET2, SENSITIVITY combination allows the user to define specific constraint design variable relationships. This combination shown relates in section A constraints 1 through 4 to design variables 70, 80, and 90. In section B the relationships are constraint 300 related to design variables 1 and 3; and constraints 100, 200, and 500 related to design variables 1, 3, and 4.



SET 1 = 1 THRU 4
SET 2 = 300 Defines sets of DSCONS cards
SET 5 = 100, 200, 500

SET 30 = 70, 80, 90
SET 31 = 4 Defines sets of DVAR cards

Then the following SET2 card may be defined:

SET2 = 18 (1,30), (2,33), ((5), (31,33))

and the following SENSITIVITY card is specified as:

SENSITIVITY = 18

Figure 12

CONSTRAINT DEFINITION

Figure 13 shows how constraints are defined with the DSCØNS bulk data card. Each constraint must have a unique DSCID for internal and external identification. The LABEL is for user convenience in identification of output. TYPE is any one of the following: DISP, FØRCE, STRESS, LAMDA, or FREQ. ID identifies the actual grid element. CØMP identifies the specific displacement component or stress or force component. LIMIT and ØPT define the equations used for the constraint. If LIMIT = 0., then plus or minus the constraint value is returned depending on the value of ØPT.

1	2	3	4	5	6	7	8	9	10
DSCONS	DSCID	LABEL	TYPE	ID	COMP	LIMIT	OPT		
DSCONS	21	DOOR	DISP	4	1	.06	MAX		

$$\text{(Upper Limit)} \quad \Psi_i = \frac{(\text{Constraint Value})^{o}}{|\text{Limit}|} - 1.0{*}\ \text{Sign (Limit)}$$

or

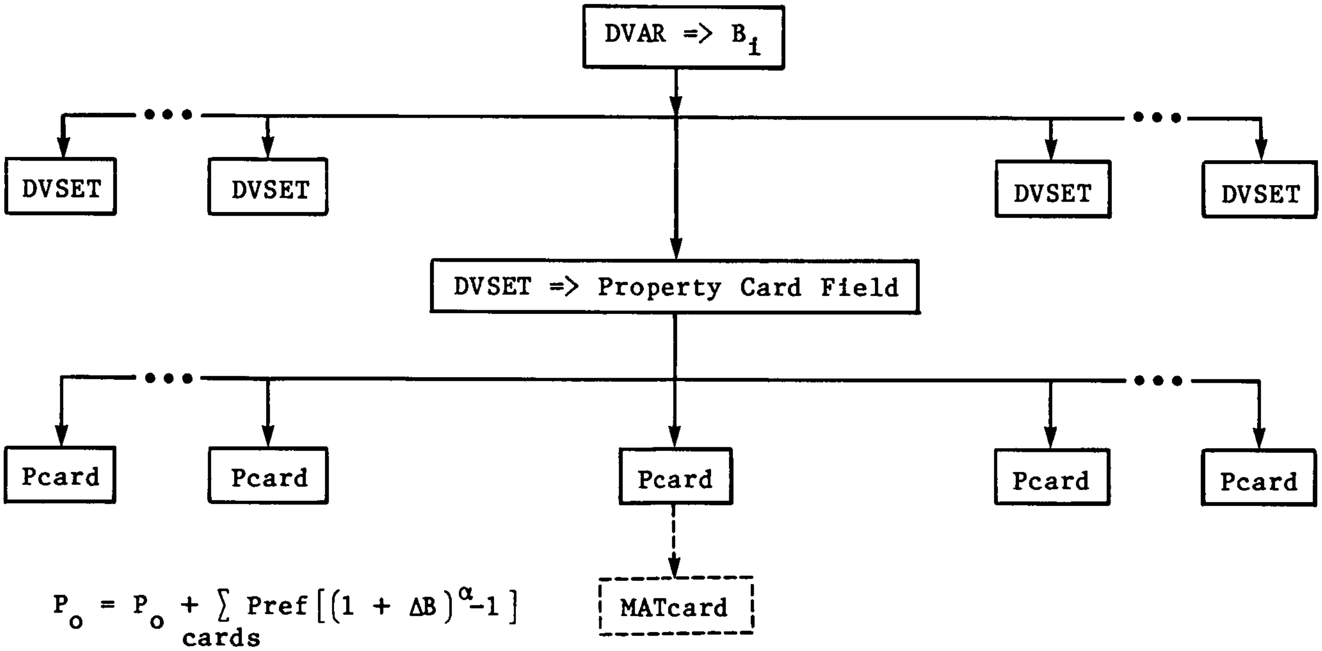
$$\text{(Lower Limit)} \quad \Psi_i = 1.0{*}\ \text{Sign (Limit)} - \frac{(\text{Constraint Value})^{o}}{|\text{Limit}|}$$

Figure 13

DESIGN VARIABLE DEFINITION

Figure 14 shows the DVAR-DVSET cards needed for design variable definition. Each BID must be unique and define a design variable. Again, LABEL is a user convenience. DELTAB defines the incremental change in normalized design variables. VID points to a DVSET card or cards. The VID on the DVSET card need not be unique. TYPE defines the property being modified and FIELD defines the specific field on the property card being modified. PREF and ALPHA along with DELTAB define the actual change in property value. PID1 etc. define the specific property cards modified.

1	2	3	4	5	6	7	8	9	10
DVAR	BID	LABEL	DELTAB	VID	VID	VID	VID	VID	ABC1
DVAR	10	DOOR	.01	2	3	6	60	100	



$$P_o = P_o + \sum_{\text{cards}} \text{Pref}\left[(1 + \Delta B)^{\alpha} - 1\right]$$

Figure 14

OUTPUT

Output in MSC/DSA is of two types. Printed output gives the current constraint values and their derivatives along with all the user identifying labels and matrix output for use in optimization programs. The user will find the printed output so clean and self-explanatory that there is no need to discuss it. Figure 15 shows the form of the matrix output. The matrix is design constrained DC wide and its rows are grouped by design variables. The number of rows in each group is the total number of loads. The very first block gives the current constraint values.

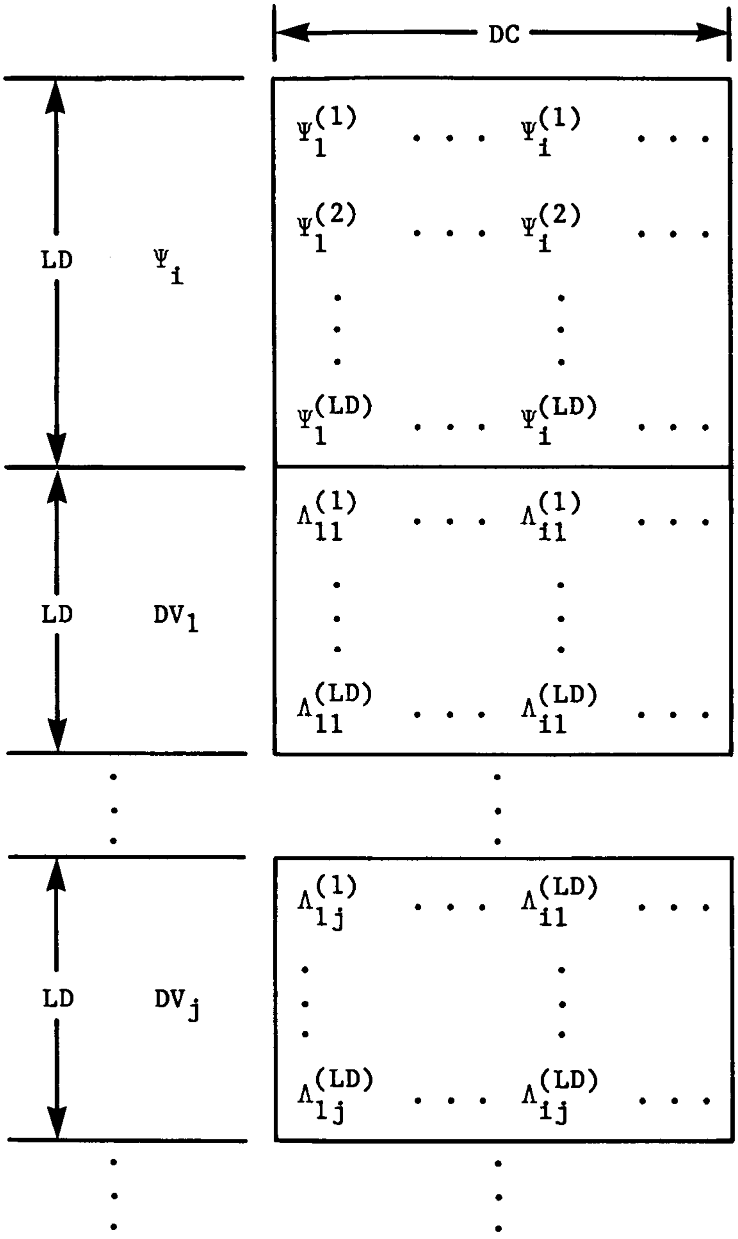

DC
LD
Ψ_i
$\Psi_1^{(1)}$ . . . $\Psi_i^{(1)}$ . . .
$\Psi_1^{(2)}$ . . . $\Psi_i^{(2)}$ . . .
$\Psi_1^{(LD)}$ . . . $\Psi_i^{(LD)}$ . . .
LD
DV_1
$\Lambda_{11}^{(1)}$ . . . $\Lambda_{i1}^{(1)}$ . . .
$\Lambda_{11}^{(LD)}$ . . . $\Lambda_{i1}^{(LD)}$ . . .
LD
DV_j
$\Lambda_{1j}^{(1)}$ . . . $\Lambda_{i1}^{(LD)}$ . . .
$\Lambda_{1j}^{(LD)}$ . . . $\Lambda_{ij}^{(LD)}$ . . .

Figure 15

SENSITIVITY COEFFICIENTS

The primary use of sensitivity coefficients is in optimization. However, it is often useful to look at them in their own right as tools that begin to give the analyst a "feel" of the structure. Figure 16 summarizes their meaning. For example, a positive sensitivity coefficient means that for an increase in design variable the value of the constraint will increase. This will move the design closer to a MAX constraint or further from a MIN constraint. Similar statements hold for a negative sensitivity coefficient.

$$\frac{\Delta\Psi}{\Delta B} > + \text{ B INCREASE } \Psi$$

$$\frac{\Delta\Psi}{\Delta B} < 0 + \text{ B DECREASE } \Psi$$

Figure 16

EXTENSION TO SUPERELEMENT ANALYSIS

Currently MSC/DSA does not allow for substructuring or, as it is called in MSC/NASTRAN, superelement analysis. The inclusion of MSC/DSA into superelement analysis requires the establishment of ground rules as listed in figure 17. To this end, design variables are considered global in nature. This means the design variable must have a unique definition across all superelements. This requirement is necessary because design variables linking can extend across superelement boundaries. DSCØNS cards are local to superelements since they can define local quantities. DVSET cards are local to superelements since they point to property cards which (under current superelement development) are local to superelements. SENSITY cards are by superelement case control. ØBJF (structural weight) is accumulated for all superelements and its derivatives computed for all design variables.

- DVAR CARDS ARE GLOBAL
- DSCONS CARDS ARE LOCAL TO SUPERELEMENT
- DVSET CARDS ARE LOCAL TO SE
- SENSITY CARDS ARE BY SE CASE CONTROL
- OBJF CARD ABOVE SUBCASE LEVEL

Figure 17

USER DEFINED CONSTRAINT RELATIONSHIPS

Currently, constraints are restricted to displacements, or specific stress or force output. It would be advantageous to define response values such as those depicted in figure 18. Here it is assumed that the constraint is related to the panel stress, either through the distortion energy relationship or through the utilization relationship.

$$\text{RESPONSE VALUE} = \text{MAX}\left(\beta_1,\ \beta_2\right)$$

$$\beta_1 = \sqrt{(\sigma_1^2 + \sigma_2^2 - \sigma_1 \sigma_2)/\bar{\sigma}_+} \qquad \text{(TENSION)}$$

$$\beta_2 = \frac{1}{2}\left(\frac{\sigma_L}{\bar{\sigma}_-} + \sqrt{\left(\frac{\sigma_L}{\bar{\sigma}_-}\right)^2 + 4\left(\frac{\tau_{XY}}{\bar{\tau}}\right)^2}\right) \qquad \text{(COMPRESSION)}$$

$$\sigma_L = \text{MIN}(o,\ \sigma_X,\ \sigma_Y)$$

Figure 18

EXTENSION OF CONSTRAINT RELATIONSHIPS

Figure 19 demonstrates the proposed extension of the constraint card to include user defined relationships. Here the constraint points to a user defined equation rather than a specific stress component. Other than the first two fields, the EQN card is free field. The equations are written using standard FORTAN type nomenclature and include standard type functions such as MØD, MAX, MIN, SØRT etc. The Fi expressions are key word type expressions defined via the FUN1 card. The FUN1 card returns a value to Fi based on either the specified stress component directly or a table look-up relating the stress value to some functional relationship such as a current value of an allowable.

1	2	3	4	5	6	7	8	9	10
DSCONS	DSCID	LABEL	TYPE	ID	COMP	LIMIT	OPT	EQUID	
DSCONS	17	SX123	STRESS	417			MAX	34	
EQN	EQID	--	FREE	FIELD	--	--	--	--	
EQN	34	MAX	(SQRT(	F1**	2+F2*	*2-	F1*F	3),	ABC
	☒								
+BC		(F4+	SQRT(	F4**	2+4.*	F5**	2)))		

VALUE = EQN (F1, F2, ... F11)

$$\text{VALUE} = \text{MAX}\left(\sqrt{F1^2 + F2^2 - F1 * F3}\ ,\ \left(F4 + \sqrt{F4^2 + 4\ F\ 5^2}\right)\right)$$

FUN1	NAME	A	COMP	TID	ALPHA	BETA	B		
FUN1	F7		2	32					

$F_i = [A(Y)^{\alpha} + B]^{\beta}$

Y = Function (Value of COMP)

Response Value = EQN (F1, F2, ... FN)

Figure 19

MSC/EØS

Figure 20 is a schematic of the MSC/EØS (Engineering Operating System) currently under development. The keys to this system are the MSC Data Base and the new (currently under code development) MSC/NDDL executive. The NDDL (Nastran Data Definition Language) provides the foundation for the development of a complete engineering data management system to support the various MSC analytical modules shown on the various spokes of figure 20. Since the NDDL provides a means of specifying a logical data structure definition it provides for the unique identification and addressability of the data and provides for the definition of the interdependencies of the data. Data Base managing will include such items as automatic storage and retrieval of data for ease of use; data recovery, integrity, and security procedures to insure data validity; and all current GINØ (general purpose input/output routines) calling sequences. Notice that the axle of the figure represents an extension of GINØ to include direct user interface between a user defined programming system (where for example the actual optimization programs will lie) and the NNDL and hence the Data Base. Through this interface new entry points into functional modules are provided so that the user's optimization routines can for example initiate new MSC/NASTRAN analysis or MSC/DSA analysis, or query the Data Base. Although the user is expected to supply his own optimization, note that one spoke includes a MSC/FSD (Fully Stressed Design) option. As envisioned, the user will have, through the interface module and the NNDL, access to any spoke or the rim or both.

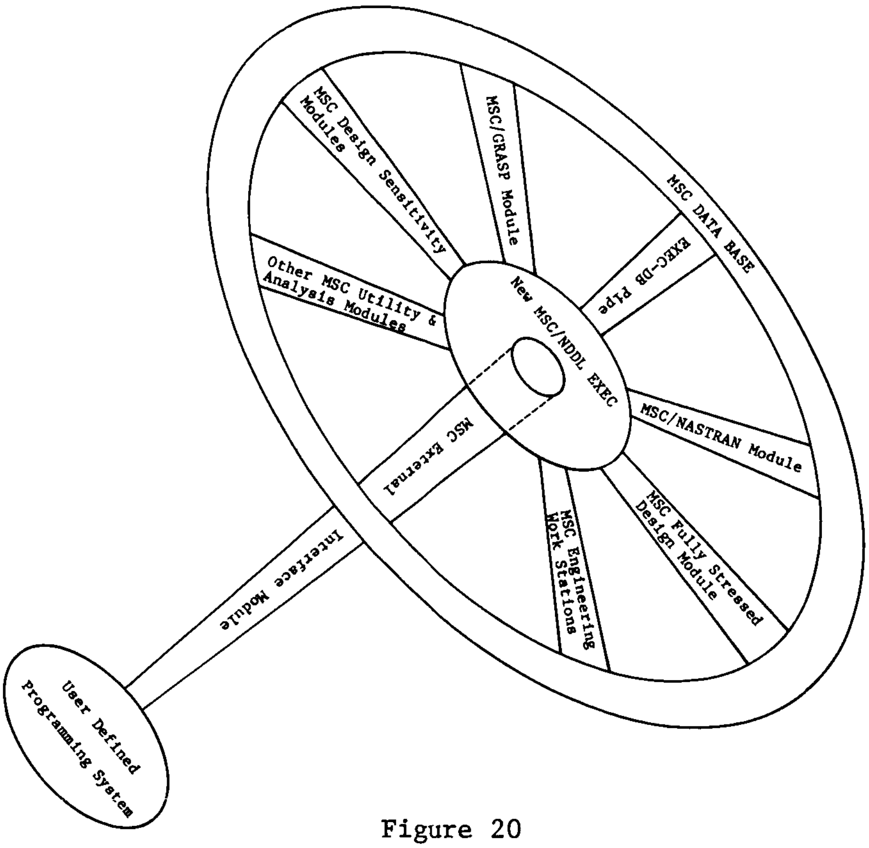

MSC Design Sensitivity Modules
MSC/GRASP Module
MSC DATA BASE
EXEC-DB Pipe
Other MSC Utility & Analysis Modules
New MSC/NDDL EXEC
MSC/NASTRAN Module
MSC External
MSC Engineering Work Stations
MSC Fully Stressed Design Module
Interface Module
User Defined Programming System

Figure 20

REFERENCES

1. Arora, S.J.; and Haug, E.J.: Methods of Design Sensitivity Analysis in Structural Optimization. AIAA J., vol. 17, no. 9, 1979, pp. 970-974.

2. Sobieszczanski-Sobieski, Jaroslaw; and Rogers, James L.: A Programming System for Research and Applications in Structural Optimization. Proceedings International Symposium on Optimum Structural Design, 11th ONR Naval Structural Mechanics Symposium, Oct. 19-22, 1981, University of Arizona, Tuscon Arizona, pp. 11-9 to 11-21.