

N87-11744

THE ADS GENERAL-PURPOSE OPTIMIZATION PROGRAM

G. N. Vanderplaats  
Naval Postgraduate School  
Monterey, California 93943

## THE GENERAL OPTIMIZATION TASK

The mathematical statement of the general nonlinear optimization problem is given in Figure 1. The vector of design variables,  $X$ , includes all those variables which may be changed by the ADS program in order to arrive at the optimum design. The objective function  $F(X)$  to be minimized may be weight, cost or some other performance measure. If the objective is to be maximized, this is accomplished by minimizing  $-F(X)$ . The inequality constraints  $g_j(X)$  include limits on stress, deformation, aeroelastic response or controllability, as examples, and may be nonlinear implicit functions of the design variables,  $X$ . The equality constraints  $h_k(X)$  represent conditions that must be satisfied precisely for the design to be acceptable. Equality constraints are not fully operational in version 1.0 of the ADS program (ref. 1) although they are available in the Augmented Lagrange Multiplier method. The side constraints given by the last equation are used to directly limit the region of search for the optimum. The ADS program will never consider a design which is not within these limits.

The ADS program was developed under NASA Research Grant 57910.

FIND THE VECTOR OF DESIGN VARIABLES,  $X$ , THAT WILL

MINIMIZE  $F(X)$

SUBJECT TO

$$G_J(X) \leq 0 \quad J=1,M$$

$$H_K(X) = 0 \quad K=1,L$$

$$X_I^L \leq X_I \leq X_I^U \quad I=1,N$$

Figure 1

## ADS PROGRAM STRUCTURE

Figure 2 shows the overall design program structure with the particular structure of the ADS program on the right. The main program is provided by the user, as well as the routines to evaluate the objective and constraint functions (analysis) and their gradients (if available). If gradient information cannot be supplied by the user, an option is included in ADS to calculate this information by finite difference. The ADS program is itself a subroutine which controls the flow of the optimization process. When function or gradient evaluations are required, control is returned to the calling program. After the information is evaluated, ADS is called again and the optimization proceeds. This program organization provides the flexibility for the user to terminate the program any time control is returned to the main program and then re-start from this point at a later time. This also provides a convenient means of performing multi-level and multi-discipline optimization where several modules in the main program call ADS independently. Also, if during analysis a sub-optimization task is performed, this may call ADS, even though the results become input to a higher level in the overall optimization process. Within the ADS program a three level structure exists, with the control routine directing the flow of information. The Strategy level is used if the problem is to be solved by conversion to a sequence of unconstrained minimizations, sequential linear programming, or other technique whereby the optimization task is converted to a sequence of problems. The Optimizer performs the actual optimization task either directly, as in the method of feasible directions, or as a sub-problem within a Strategy. Finally, the One-Dimensional Search portion performs a line search to minimize the objective in a direction specified by the Optimizer. The particular techniques used at each of these three levels are described in the figures to follow.

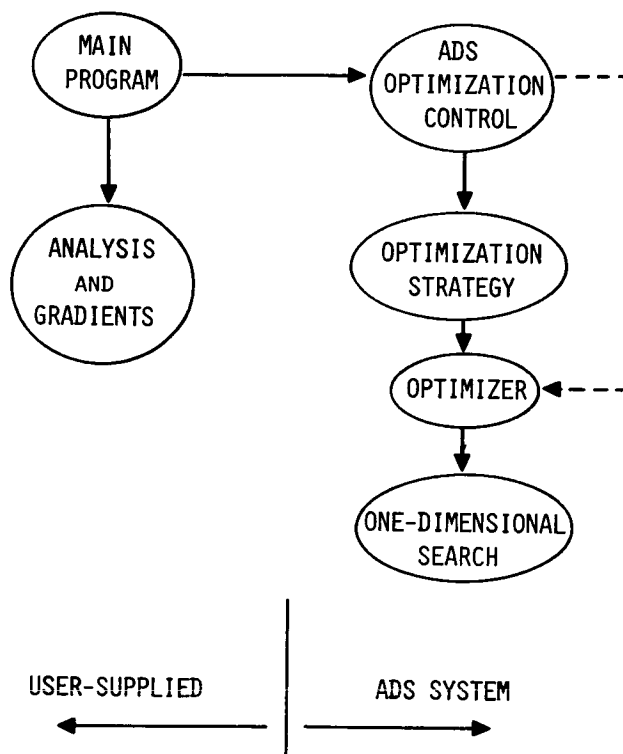


Figure 2

## OPTIMIZATION STRATEGIES

Figure 3 lists the Strategies available in version 1.0 of the ADS program. The parameter ISTRAT is used to designate the strategy to be used. ISTRAT=0 would be used if optimization is to be performed by a direct method such as the method of feasible directions or if the problem is unconstrained. Options 1-5 are various forms of Sequential Unconstrained Minimization Techniques. Option 6 is classical Sequential Linear Programming with move limits to insure stability. Option 7 is also a form of sequential linear programming, but instead of producing a sequence of infeasible designs as the optimum is approached, this method produces a sequence of improving feasible designs. Option 8 is a relatively new algorithm whereby the Lagrangian function is approximated as a quadratic and the constraints are linearized. This approach retains the essential nonlinearity of the problem, even for linear objective functions. Theoretically, a Quadratic Programming sub-problem is solved in this method. However, by using a direct method for optimization as opposed to a special purpose QP optimizer, some of the theoretical difficulties with the method are overcome. Early experience with the ADS program has shown that Sequential Linear Programming is more effective than is usually thought and that Sequential Quadratic Programming as coded here is a particularly powerful strategy.

<u>ISTRAT</u>	<u>METHOD</u>
0	NONE. GO DIRECTLY TO OPTIMIZER
1	EXTERIOR PENALTY FUNCTION METHOD
2	LINEAR EXTENDED INTERIOR PENALTY FUNCTION METHOD
3	QUADRATIC EXTENDED INTERIOR PENALTY FUNCTION METHOD
4	CUBIC EXTENDED INTERIOR PENALTY FUNCTION METHOD
5	AUGMENTED LAGRANGE MULTIPLIER METHOD
6	SEQUENTIAL LINEAR PROGRAMMING
7	METHOD OF INSCRIBED HYPERSPHERES (METHOD OF CENTERS)
8	SEQUENTIAL QUADRATIC PROGRAMMING

Figure 3

## OPTIMIZERS

The optimization algorithms available in the ADS program are listed in Figure 4. The parameter IOPT is used to indicate the optimizer to be used. Options 2, 4 and 5 are used for solving unconstrained problems or for the unconstrained minimization sub-problem in a Sequential Unconstrained Minimization strategy. When an unconstrained optimization method is used, the design is still limited by the side constraints. This insures that, for example, minimum gage constraints are never violated, even when using a Sequential Unconstrained Minimization Technique at the Strategy level. Two feasible directions algorithms are available for constrained optimization. These are used for direct optimization of constrained problems as well as for solving the linear or quadratic programming sub-problem of strategies 6-8. The Method of Feasible Directions algorithm is essentially the same as that contained in the earlier program, CONMIN (ref. 2). The Modified Method of Feasible Directions (ref. 3) is similar to the Generalized Reduced Gradient Method, but is more storage efficient. Also, in the one-dimensional search, this method uses a least squares technique rather than Newton's method for maintaining feasibility.

### UNCONSTRAINED

<u>IOPT</u>	<u>METHOD</u>
2	FLETCHER-REEVES CONJUGATE DIRECTION METHOD
4	DAVIDON-FLETCHER-POWELL VARIABLE METRIC METHOD
5	BROYDON-FLETCHER-GOLDFARB-SHANNO VARIABLE METRIC METHOD

### CONSTRAINED

<u>IOPT</u>	<u>METHOD</u>
1	METHOD OF FEASIBLE DIRECTIONS
3	MODIFIED METHOD OF FEASIBLE DIRECTIONS

Figure 4

## ONE-DIMENSIONAL SEARCH

Figure 5 lists the One-Dimensional Search routines available in ADS. Five algorithms are available for both constrained and unconstrained problems. The parameter IONED is used to identify the algorithm to be used. Options 1 and 6 are usually not useful except for special purpose applications. The remaining methods are different combinations of the Golden Section method and Polynomial Interpolation. The Golden Section method is normally only useful if function evaluations are very cheap and if high precision of the one-dimensional search is desired. While this method is usually applied only to unconstrained problems, it has been modified to find the constrained minimum for use in ADS. Normally options 4 and 9 are the most efficient and reliable, where the solution is first bounded and then refined by polynomial interpolation. One unique feature of the ADS program is that, if no feasible solution can be found, a design is sought which minimizes the constraint violations. Thus, on termination, the constraints which must be relaxed to produce a realistic design are easily identified.

### UNCONSTRAINED

<u>IONED</u>	<u>METHOD</u>
1	BOUNDS ONLY
2	GOLDEN SECTION METHOD
3	GOLDEN SECTION + POLYNOMIAL INTERPOLATION
4	BOUNDS + POLYNOMIAL INTERPOLATION
5	POLYNOMIAL INTERPOLATION/EXTRAPOLATION WITHOUT FIRST GETTING BOUNDS

### CONSTRAINED

<u>IONED</u>	<u>METHOD</u>
6	BOUNDS ONLY
7	GOLDEN SECTION METHOD
8	GOLDEN SECTION + POLYNOMIAL INTERPOLATION
9	BOUNDS + POLYNOMIAL INTERPOLATION
10	POLYNOMIAL INTERPOLATION/EXTRAPOLATION WITHOUT FIRST GETTING BOUNDS

Figure 5

## ADS PROGRAM OPTIONS

Not all combinations of Strategy, Optimizer and One-Dimensional search are appropriate. For example, an unconstrained minimization method would not be a valid optimization technique for use with Sequential Linear Programming. Figure 6 shows the acceptable combinations of modules, where the numbers correspond to the control parameters ISTRAT, IOPT and IONED. In this table, an X is used to denote a valid combination of methods. First, a Strategy is chosen appropriate to the problem at hand. Next, moving across the row, the Optimizer is chosen from among the valid options, and finally, moving down the column, the One-Dimensional search routine to be used is chosen. An example is shown by the solid line beginning at Strategy number 5 (the Augmented Lagrange Multiplier Method). The Broydon-Fletcher-Goldfarb-Shanno optimizer (IOPT=5) is used to solve the unconstrained minimization sub-problem and the One-Dimensional search is to be performed using polynomial interpolation after first bounding the solution (IONED=4). It is clear from this table that a large number of independent combinations of methods are available. It may be expected that, as experience is gained with the program, many of the options will prove not to be useful for practical design and that a few will survive as preferred options. One of the unique features of the program organization is that one-to-one comparisons may be made between methods by changing only four input parameters to ADS. No other coding modifications are needed.

STRATEGY	OPTIMIZER				
	1	2	3	4	5
0	X	X	X	X	X
1	0	X	0	X	X
2	0	X	0	X	X
3	0	X	0	X	X
4	0	X	0	X	X
⑤	0	X	0	X	ⓧ
6	X	0	X	0	0
7	X	0	X	0	0
8	X	0	X	0	0
ONE-DIM. SEARCH					
1	0	0	0	0	0
2	0	X	0	X	X
3	0	X	0	X	X
4	0	X	0	X	ⓧ
5	0	X	0	X	X
6	0	0	0	0	0
7	X	0	X	0	0
8	X	0	X	0	0
9	X	0	X	0	0
10	X	0	X	0	0

Figure 6

## BLOCK DIAGRAM FOR USING ADS

Figure 7 shows the program organization for using ADS. The user must begin by allocating array storage for the vector of design variables, constraints, and design variable bounds, as well as work arrays for use in ADS. Also at this point the basic control parameters such as the number of design variables, the options ISTRAT, IOPT and IONED and similar parameters are defined. The basic program flow control parameter INFO is initialized here to -2. Then the ADS subroutine is called to initialize all internal parameters to their default values. These include convergence criteria, finite difference steps and the like. Control is then returned to the calling program and the user is free to over-ride the default values of the internal parameters if desired. ADS is then called again and the optimization process proceeds from here. Whenever function or gradient information is needed, control is returned to the calling program with INFO=1 or 2 respectively. The user evaluates the needed information and calls ADS again. This iterative process continues until the optimization task is complete, at which point ADS returns a value of INFO=0. All information is transferred to and from ADS via a single parameter list. No additional common blocks or data transfer mechanisms are required. Any time that control is returned to the calling program, the user may store the contents of the parameter list on mass storage and terminate the program. The program may be restarted from here by reading the information back and continuing the program flow. In the event that the user wishes to use all default options in ADS, the first call to ADS may be omitted. In this case INFO is initialized to zero and the optimization proceeds without the first initialization step.

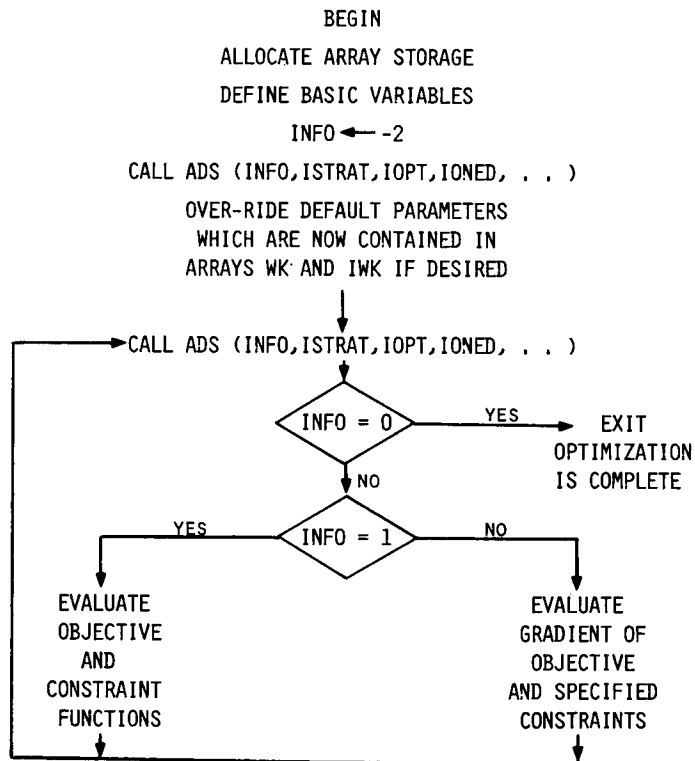


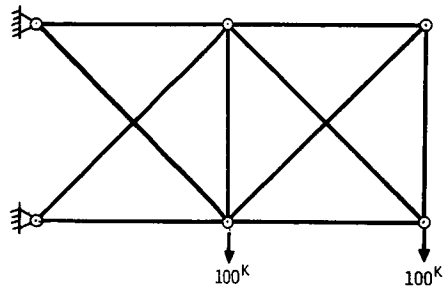
Figure 7



## DESIGN EXAMPLE

Figure 8 gives an example of the ADS program for solution of the 10-bar truss commonly found in the literature. Here no special effort has been made to formulate the problem for efficient optimization. The design variables are the member cross-sectional areas (as opposed to reciprocal variables which would be much more efficient). The structure is stress constrained, subject to the single loading case shown. The table gives the optimization results for various combinations of ISTRAT, IOPT and IONED. The number of function and gradient evaluations is also given. In this example, all default parameters in ADS are used and no attempt was made to "fine tune" the program to this problem. Also, it should be noted that as experience is gained with the program, these defaults will be modified to improve efficiency of the general design task. As may be expected, Sequential Unconstrained Minimization methods required a relatively high number of function and gradient evaluations. Also, the use of the Golden Section algorithm in the One-Dimensional search dramatically increases the number of function evaluations without noticeable improvement in the result. Direct methods appear to be relatively efficient and the Sequential Quadratic Programming method is seen to be a particularly powerful Strategy. Assuming early experience with this method is indicative of its efficiency for general design problems, this appears to offer an impressive capability for engineering design. As experience is gained with the program, further refinements can be expected. The over-all motivation in the development of the ADS program has been to provide a user-friendly, general, and efficient tool for a wide variety of engineering design problems of practical interest.

DESIGN EXAMPLE: THE 10-BAR TRUSS



ISTRAT	IOPT	IONED	OPTIMUM	ANALYSES	GRADIENTS
0	1	7	1516.8	305	39
0	1	9	1519.7	120	30
0	3	8	1497.8	489	8
0	3	9	1497.3	114	6
1	2	4	1648.4	114	27
1	4	2	1534.2	384	37
1	4	5	1549.4	109	33
2	4	5	1522.7	133	41
2	5	2	1505.2	528	51
3	4	4	1511.3	211	51
4	5	4	1500.8	209	52
5	4	4	1504.0	210	47
5	5	4	1496.3	235	54
6	3	9	1510.3	20	20
7	3	10	1509.5	47	47
8	1	9	1498.0	28	6
		CONMIN	1500.9	104	31

Figure 8

#### REFERENCES

1. Vanderplaats, G. N., Sugimoto, H. and Sprague, C. M., "ADS-1: A New General-Purpose Optimization Program," Proc. AIAA/ASME/ASCE/AHS 24th Structures, Structural Dynamics and Materials Conference, Lake Tahoe, NV, May 1983, AIAA Paper No. 83-0831-CP.
2. Vanderplaats, G. N., "CONMIN - A FORTRAN Program for Constrained Function Minimization, User's Manual," NASA TM X-62282, Aug. 1973.
3. Vanderplaats, G. N., "A Robust Feasible Directions Algorithm for Design Synthesis," Proc. AIAA/ASME/ASCE/AHS 24th Structures, Structural Dynamics and Materials Conference, Lake Tahoe, NV, May 1983.