

DOT

Design Optimization Tools

DESCRIPTION

DOT is a general purpose numerical optimization software package which can be used to solve a wide variety of nonlinear optimization problems. The user provides a main program for calling DOT, and an analysis program to evaluate the necessary functions. DOT is linked with the user's codes to create the design optimization program. DOT will change the input parameters to the analysis in order to minimize or maximize the user defined objective, subject to constraints (limits) on other user defined responses. To achieve this, DOT calls the analysis program repeatedly while searching for the optimum.

The user supplied analysis program has a set of inputs we put in a vector, X , called the vector of design variables.

The user supplied analysis program has a set of outputs called responses. One of these responses can be chosen to be the objective function which is to be minimized or maximized. Lower and upper bounds can be placed on other responses and these are called constraints.

The responses calculated in the analysis program can be linear or nonlinear functions of the design variables. They may be calculated as very simple analytical functions or they may be highly complicated implicit functions of the design variables. Very little formal knowledge of optimization techniques is needed to make efficient use of DOT.

MATHEMATICAL PROBLEM SOLVED

DOT numerically solves the following problem:

Find the values of the N design variables contained in X that will

Minimize or maximize $OBJ = F(X)$

where OBJ is the objective function

Subject to (such that);

$G_j(X) \leq 0$ for $j = 1, NCON$

where $NCON$ is number of constraints.

$X_i^L \leq X_i \leq X_i^U$ for $i = 1, NDV$

where NDV is the number of design variables.

Array G contains the constraints that must be satisfied. For example, if some response, R ,

must be greater than -5 and less than 100 , this would lead to the following two (normalized) constraints;

$$G(1) = \frac{(-5-R)}{5} \leq 0$$

$$G(2) = \frac{(R-100)}{100} \leq 0$$

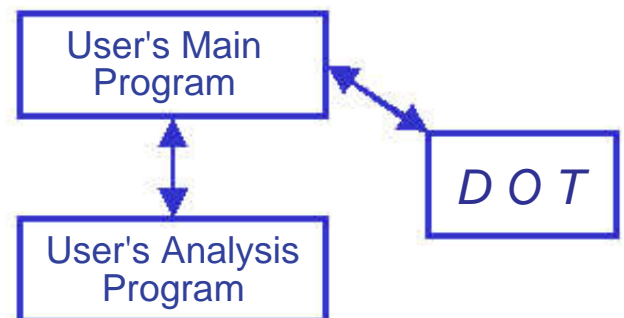
Note that $G(1)$ and $G(2)$ must both be less than or equal to zero for the design to be acceptable. There are a total of $NCON$ constraints, and $NCON$ can be zero (unconstrained) or can be very large.

The X^L and X^U are referred to as "side constraints" and they simply limit the region of search for the optimum.

Equality constraints are imposed by providing two equal and opposite inequality constraints. Equality constraints can be linear or nonlinear functions of the design variables.

HOW DOES DOT WORK?

The figure below defines the structure of a program which will call DOT for optimization.



The user must provide a main program which defines various arrays and parameters. The user also provides an analysis subroutine or function which evaluates the objective and constraint functions, and calculates gradient information if that option is used. The main program calls DOT to proceed with optimization. DOT will modify the design variables in search of the optimum. When DOT requires the values of the objective function and constraints corresponding to a proposed design, it returns control to the main program and the analysis is called to evaluate them. DOT is then called again, and this process is repeated until DOT returns a parameter to indicate that optimization is complete.



ALGORITHMS

For Constrained Optimization:

- Modified Method of Feasible Directions (MMFD).
- Sequential Linear Programming (SLP) with adjustable move limits.
- Sequential Quadratic Programming (SQP).

For Unconstrained Optimization:

- Broydon-Fletcher-Goldfarb-Shanno (BFGS) Algorithm.
- Fletcher-Reeves (FR) Algorithm.

The algorithm to be used for optimization is controlled with a single input parameter to DOT.

These algorithms have been extensively developed and tested and have been demonstrated to be both efficient and reliable for a wide range of engineering applications.

OPTIONS

Default values are provided for all control parameters.

- These can be easily modified by the user.
- All information is transferred via the DOT subroutine/function parameter list.
- The user can stop optimization at any time and restart optimization later.
- DOT calculates gradient information by finite difference. The user can provide analytic gradients if this is desired.
- The user can direct design history information to a separate output file for later graphic processing.

CONFIGURATIONS:

RAM Memory:

DOT does not have any fixed memory requirements. You will programmatically set them.

Disk Space:

The DOT libraries and documentation require approximately 3 MB of free disk space.

Program Development Environments:

DOT is delivered as an object library that you can link with your programs. DOT has been tested with Microsoft PowerStation Fortran 4, Digital Visual Fortran 5/6, Microsoft Visual C++ 5/6 and all major UNIX workstation compilers.

RELATED PRODUCTS FROM VR&D

VisualDOC

- Graphics-based, general-purpose design optimization software system.
- Design variables may be continuous, discrete, integer or mixed.
- Supports linear and nonlinear design variable linking.
- Supports multi-objective optimization.
- Supports Response Surface Methodology.
- Supports Design of Experiments (DOE).
- Interface easily to your own code or third-party analysis programs such as, ABAQUS and LS-DYNA.

GENESIS

- GENESIS is a fully integrated finite element analysis and design optimization software package.
- Analysis is based on the finite element method (FEM) for static, normal modes, direct and modal frequency analysis, heat transfer and system buckling calculations.
- Shape, sizing and topology optimization are the design options available to the user.
- Optimization is performed by the DOT optimizer.
- Typically the optimization requires less than ten detailed finite element analyses, even for large and complex design tasks.

WHY BUY VR&D PRODUCTS?

- VR&D provides robust, state-of-the-art software and services to speed up your product development process.
- To exploit the full potential of our software, VR&D provides hands-on training, both at your site and our corporate office at Colorado Springs.
- VR&D provides world class technology, customer driven products and cost effective solutions to your product development.

Vanderplaats Research & Development, Inc.



1767 South 8th Street, Suite 100
 Colorado Springs, CO 80906
 Ph. 719-473-4611 FAX 719-473-4638
 Email: sales@vrand.com
 Web: www.vrand.com