GENERALIZED MATHEMATICAL MODELS IN DESIGN OPTIMIZATION

Panos Y. Papalambros
and

J. R. Jagannatha Rao[*]
Department of Mechanical Engineering
and Applied Mechanics
The University of Michigan
Ann Arbor, MI

[*]Graduate Student

Introduction

We make the usual distinction between the vector of parameters and the vector of variables in an optimization problem. A parametric optimal design problem is considered as a system whose input is the vector of parameters. Corresponding to each input parameter vector, the output is, collectively, the feasible domain, the optimal objective value and the optimal variable vector. Hence, this input to output relation is characterized by a point-to-set map. When the stability arguments of such a map are augmented with the need to identify the optimal *input* for a given system, a very useful framework emerges for design problems. Recent results seem to indicate that such an approach would address the issue of studying the *modeling of the design itself*, in conjunction with the numerical procedures that are used to solve the optimization problem. While the concept of parametric optimization is not new, its interpretation in the form of an input to output mapping, and the associated solution strategy that could make explicit the inner stability of a problem, are considered very useful generalizations of traditional design optimization models (see fig. 1).
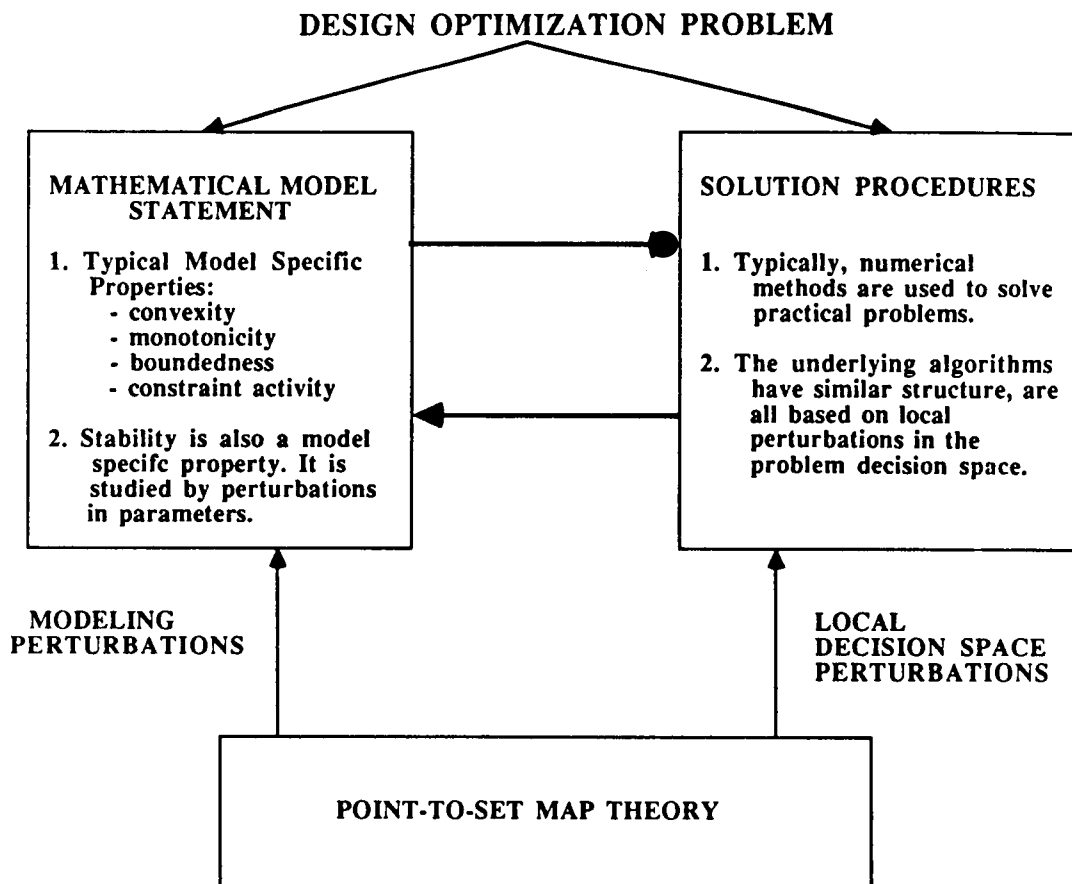
DESIGN OPTIMIZATION PROBLEM

MATHEMATICAL MODEL STATEMENT

1. Typical Model Specific Properties:
   - convexity
   - monotonicity
   - boundedness
   - constraint activity

2. Stability is also a model specifc property. It is studied by perturbations in parameters.

SOLUTION PROCEDURES

1. Typically, numerical methods are used to solve practical problems.

2. The underlying algorithms have similar structure, are all based on local perturbations in the problem decision space.

MODELING PERTURBATIONS

LOCAL DECISION SPACE PERTURBATIONS

POINT-TO-SET MAP THEORY

Figure 1

Problem Statement

The terminology used here closely follows that used in ref. 1. The general design optimization model is assumed to be stated as:

$$minimize \quad f(x,p)$$

$$P(x,p) \quad subject \ to$$

$$g_k(x,p) \le 0 \quad k \in L = \{1,\dots,l\}$$

$$h_i(x,p) = 0 \quad i \in M = \{1,\dots,m\} \tag{1}$$

$$p = (p_i) \quad \in P \subset \Re^p$$

$$x = (x_i) \quad \in X \subset \Re^n$$

where f, g_k and h_i are the scalar objective, inequality, and equality constraint functions respectively. The variable vector $x = (x_i)$ usually describes the conventional design variables such as member areas in a truss structure. The vector p is a parameter that could describe, for example, the allowable yield stress for a truss member.

For any particular $p = p^* \in P$, the model $P(x,p)$ is the usual mathematical programming statement:

$$minimize \quad f(\mathbf{x}), \quad \mathbf{x} \in \Re^n$$
$$subject \ to$$
$$\mathbf{h(x)} = \mathbf{0} \tag{2}$$
$$\mathbf{g(x)} \le \mathbf{0}$$

A Generalized Model as an Input-Output System

The basic idea in *input-output* (IO) formulations is as follows (fig. 2). We consider the model $P(x, p)$ to be a system whose input is a particular $p \in P$. Corresponding to each such admissible input, the output is defined as collectively $\{F(p), F^*(p), f^*(p)\}$ where

$F(p) = \{x \in \Re^n : g_k(x, p) \leq 0 \ k \in L, h_i(x, p) = 0, \ i \in M\}$ is the feasible set

$F^*(p) = \{x^*(p)\}$, is the set of optimal solutions x^*

$f^*(p) = f(x^*(p), p)$, the optimal value function

STRUCTURE OF AN IO FORMULATION
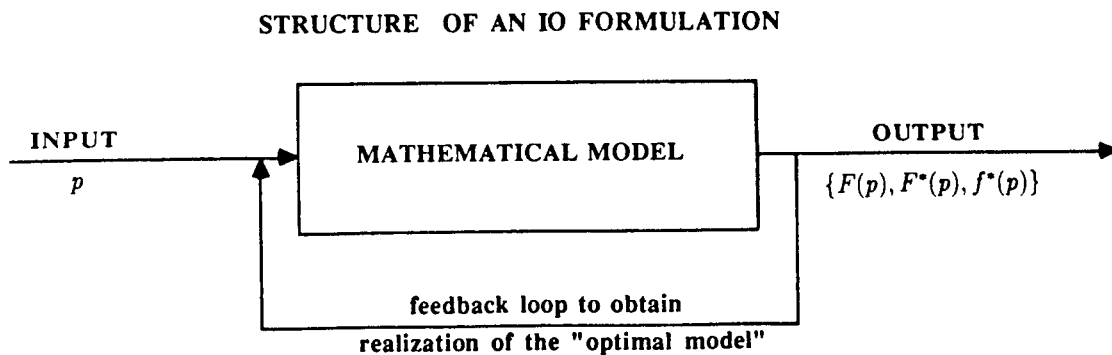
DESIRED OUTCOME FROM AN IO FORMULATION

- OPTIMAL REALIZATION OF THE MATHEMATICAL MODEL

- THE BEST STABLE (AND FEASIBLE) PATH FROM THE INITIAL INPUT TO THE OPTIMAL INPUT

Figure 2

IO and Conventional NLP*

One must at the outset try to answer under what circumstances would such an input-output formulation be more useful than the following single problem formulation:

$$
\begin{aligned}
&minimize \quad f(\mathbf{z}), \quad where \ \mathbf{z} = (\mathbf{x}, \mathbf{p}) \in \Re^{n+p} \\
&subject \ to \\
&\qquad \mathbf{h}(\mathbf{z}) = \mathbf{0} \\
&\qquad \mathbf{g}(\mathbf{z}) \leq \mathbf{0}
\end{aligned}
\tag{3}
$$

While the formulation above appeals to simplicity in its treatment since conventional optimality conditions and numerical methods apply to it, there are however some issues of note (fig. 3):

- **THE SOLUTIONS TO THE PROBLEMS (1) and (3) MAY NOT COINCIDE.** In some cases, a new set of necessary conditions can be derived for the IO solution (ref. 1).

- **A SIMPLER (such as convex or monotonic) SUBPROBLEM** can sometimes be derived by splitting the original variable vector z into a new variable x, of smaller dimension, and a *parameter* θ. Relates to decomposition techniques used in large-scale programming.

- Most importantly, an IO formulation, if successfully implemented, would **CHARACTERIZE THE INNER STABILITY OF THE PROGRAM.** The behavior of critical points on an IO path is identified as explicit model specific properties and enables us to view *modeling* and *solution techniques* in a unified way.

Figure 3

*Nonlinear Programming (NLP).

Stability Properties in Generalized Modeling

Following the usual notion of well-posedness, we say that a mathematical problem is well-posed if its solution is continuous w.r.t.* to the data. As we have seen, $P(x,p)$ is an imbedded or parametrized family of mathematical programs. For these programs, the notion of stability is defined directly in terms of the continuity of the solution w.r.t. the perturbation vector p. In any study of such a perturbed family of related problems, the question of stability becomes central for two important reasons: Unstable points or regions should be identified to see if there is an attendant physical interpretation to the loss of stability; and if the problem is stable everywhere then standard optimality conditions would apply in the problem $P(z)$ where $z = (x, p)$. The continuity of parameter-dependent feasible sets, solution sets and extremal value functions is useful to answer many questions such as:

- For which optimal design models, the intuitive argument that the accuracy of the solution obtained increases with the degree of approximation of the initial data, is indeed justified.

- If an exact solution can be viewed as an approximation to a solution of the problems that correspond to small perturbations of the initial parameters. This is particularly important if the calculation of solution requires substantial time and expense.

- Finally, what are the quantitative bounds on the solution as the initial data is substantially varied.

The stability information that we refer to here is directly related to (upper and lower) continuity properties of certain point-set-maps. As an example of such a map, we have:

$$\Omega : \Re^p \to \Re^n \times \Re^n \times \Re, \quad \Omega(\theta) = \{F(\theta), F^*(\theta), f^*(\theta)\}$$

Several studies have been reported on the theoretical properties of such maps (e.g. ref. 2 and ref. 3).

Many of the early results in stability of optimal value function in nonlinear programming were obtained for right–hand–side perturbations of constraints (ref. 4). While studying general perturbations in the problem $P(x,p)$:

$$P(x,p) : min\{f(x,p) : g(x,p) \leq 0, h(x,p) = 0, (x,p) \in X \times P\},$$

an interesting observation from ref. 5 is that at least when the dependence of f, g, and h on p is locally Lipschitz, a problem $P(x,y,p)$ with equal $f^*(p)$ can be formulated having only right–hand–side perturbations (Note that now the variable vector is (x,y)) :

$$P(x,y,p) : min\{f(x,y) : g(x,y) \leq 0, h(x,y) = 0, -y + p = 0, (x,y,p) \in X \times P \times P\}$$

---

*with respect to (w.r.t.)

Regularity of Constraints and Stability

The study of constraint qualifications and nontrivial abnormality in Lagrange multiplier values is closely related to the study of stability of the optimal value function (ref. 5 and ref. 6). For example, if the second order sufficiency conditions hold at a local minimizer at which the constraints are *regular*, then that local minimizer persists under small perturbations (ref. 7). And the well known Mangasarian-Fromovitz constraint qualification is a necessary and sufficient condition for the set of Lagrange multiplier vectors associated with a stationary solution to form a compact polyhedron (ref. 8). While stability of optimal value function has been treated extensively in literature, the stability properties of the optimal solution set have also been considered (ref. 3 and ref. 9). From a computational standpoint, numerical continuation and bifurcation techniques can be applied to the critical points characterized by some suitable first-order necessary conditions (ref. 10 and also ref. 11). One can then classify the singularities based on:

1. Loss of the strict complementarity condition (see fig. 4)

2. Linear dependence of the gradients of the active constraints

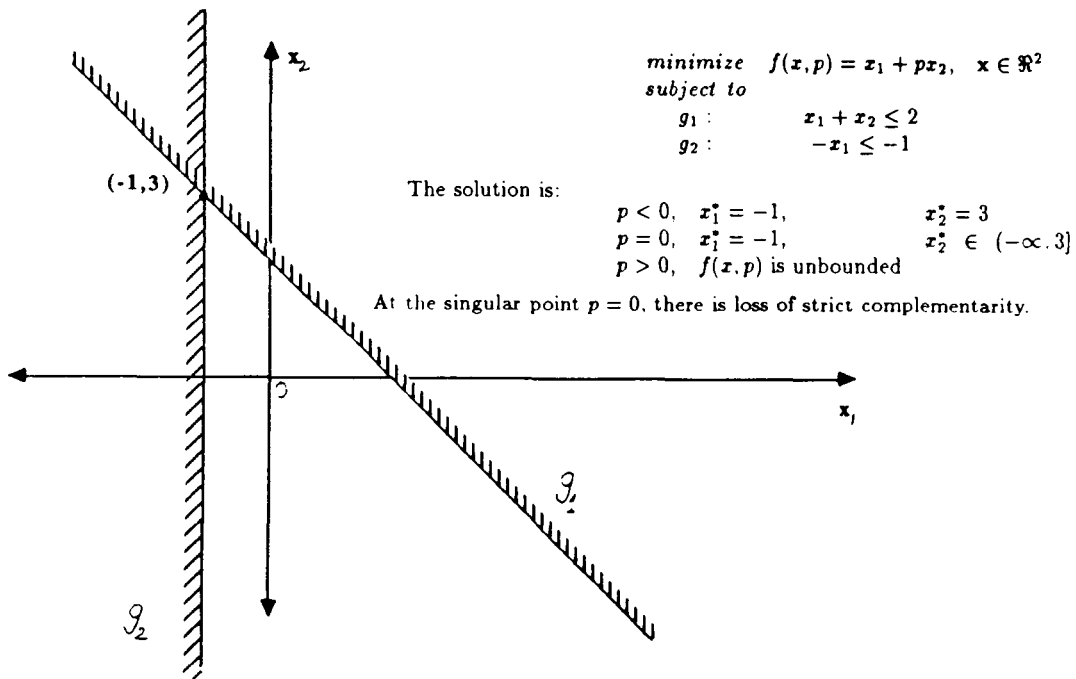3. Singularity of the Hessian of the Lagrangian on the tangent space

$$minimize \quad f(x,p) = x_1 + px_2, \quad x \in \Re^2$$
$$subject\ to$$
$$g_1: \quad x_1 + x_2 \le 2$$
$$g_2: \quad -x_1 \le -1$$

The solution is:

$$p < 0, \quad x_1^* = -1, \qquad x_2^* = 3$$
$$p = 0, \quad x_1^* = -1, \qquad x_2^* \in (-\infty, 3]$$
$$p > 0, \quad f(x,p)\ \text{is unbounded}$$

At the singular point $p = 0$, there is loss of strict complementarity.

Figure 4

A Design Plus Processing Model

A common characteristic of traditional design optimization models is that they operate on a single hierarchical level, with the possible exception of some decomposition strategies. However, there still does not exist a way to properly study the *modeling interactions* that occur in such practical systems where a model structure appears not just because the overall system is made up of individual components or subsystems, but rather because the same system can be viewed from two or more viewpoints. For example, a typical structural design model may be developed from its functional viewpoint, and another one from a fabrication or processing viewpoint. Usually, to study such an *integrated* system, models are blended in a single formulation by treating many additional constants (parameters) as variables and/or by introducing a rather arbitrary multiobjective formulation. In these cases, it is very difficult to understand and represent explicitly the relation of different quantities in the solution, or interpret their physical meaning. It is hoped that an input-output formulation will make more explicit the properties of model interactions. Figure 5 below depicts a typical system level abstraction in a design plus processing model.

Definition: Drawing is the process of reducing the cross-sectional area and/or the shape of a rod, bar, tube or wire (cold or hot) by pulling through a die.
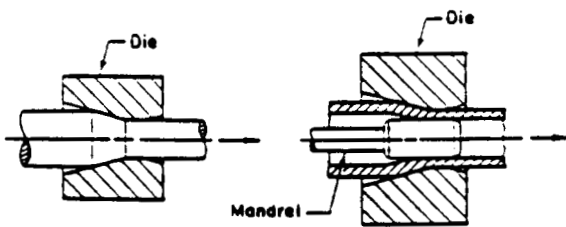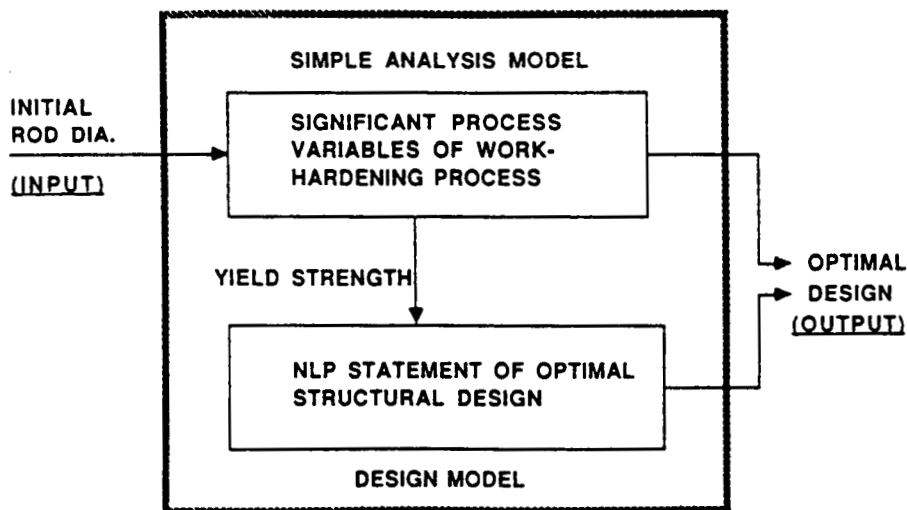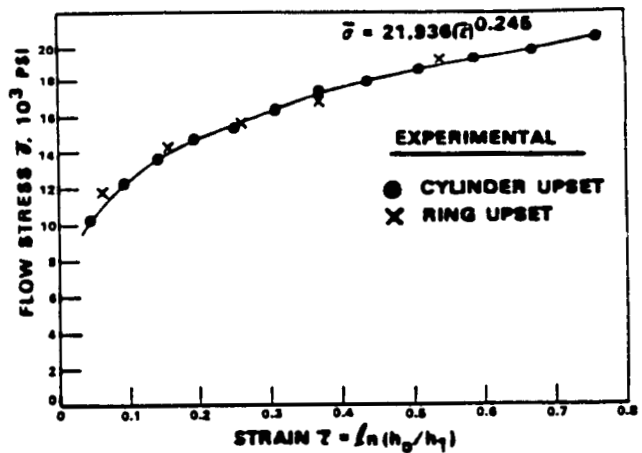

Figure: Drawing of (left) rod or wire and (right) tube

MODEL INTERACTIONS IN IO FORMULATION

Figure 5

An Example of IO Formulation

As mentioned previously, it is of considerable importance to be able to study the behavior of optimum *design models* in conjunction with some model representation (referred to here as the processing model) of a class of suitable fabrication processes. Detailed processing models are in general much harder to obtain as compared to the *design* models, so one is often left with highly simplified processing models having a very limited range of validity. A natural approach seems then to take the more tractable *design* model and treat the *processing variables* as input. One then hopes to study the extremal behavior of *optimal design models* when the input (or in generic terms the *process*)is perturbed in a well-defined way. Using a stress-strain power law as being the simplest representation of those processes where the material work hardens, an IO formulation for a simple cantilever beam is described below. The design variables are the moments of inertia of the two segments of the stepped cantilever beam and the design objective is to minimize the weight of the beam subject to a deflection constraint at the tip load as well as a constraint on the stress at the fixed end (fig. 6).

$$
\begin{aligned}
&min \quad f = \gamma(\sqrt{x_1} + \sqrt{x_2}) && (weight)\\
&Sub.\,to\\
&\mathbf{h}: \quad \mathbf{K(X)\,U = F} && (equilibrium)\\
&g_1: \quad u_3 + l u_4 - c \le 0 && (deflection)\\
&g_2: \quad (64.0/\pi)^{0.25} p\, l / x_1^{0.75} - (K/s)\left(2\ln\frac{d_i}{(64x_1/\pi)^{0.25}}\right)^n \le 0 && (stress)\\
&g_3: \quad 2\ln\frac{d_i}{(64x_1/\pi)^{0.25}} \le 0.7 && (processing)\\
&g_4: \quad -2\ln\frac{d_i}{(64x_1/\pi)^{0.25}} \le -0.1 && (processing)
\end{aligned}
$$

$$
p = (d_0) \in P = [d_{imin}, d_{imax}] \subset \Re
$$

where K and n are material constants for the flow stress-strain equation. and

$$
\mathbf{K} = (E/l^3)\begin{bmatrix}
12(x_1+x_2) & -6(x_1-x_2)l & -12x_2 & 6x_2l\\
 & 4l^2(x_1+x_2) & -6x_2l & 2x_2l^2\\
 & & 12x_2 & -6x_2l\\
symm & & & 4x_2l^2
\end{bmatrix}
$$

and

$$
\mathbf{F} = (0,0,p,0)^T
$$

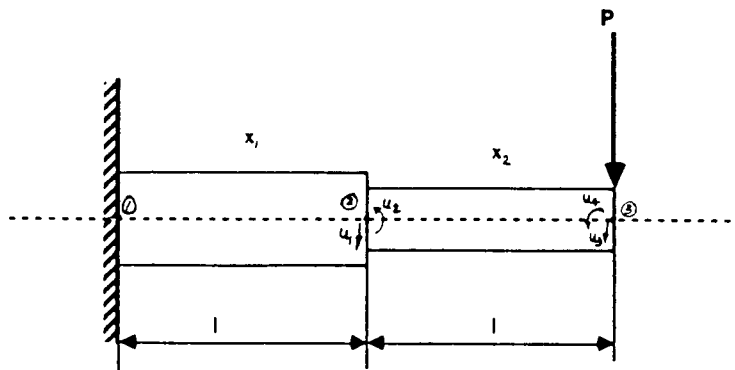**Variables:** $(x_1, x_2, u_1, u_2, u_3, u_4)^T$

Figure 6

Proposed Numerical Procedure

Note that in this IO formulation, the standard design problem has been imbedded in a family of programs, parametrized by the scalar input parameter d_i, the initial stock diameter. For a fixed input (leading to a standard NLP), the problem can be solved by a generalized reduced gradient algorithm to get:

Design Variables $x_1 = 30.89$, $x_2 = 11.17$
for Input $d_i = 3.5$ leading to $\epsilon^* = 0.434$

Solving the problem again for different values of input, a trend is obtained as shown in the figure below. This should give an indication of the IO solution strategy where we need to track the solution continuously as the input varies and to see if any critical or unstable points lead to bifurcation of the Kuhn-Tucker curve. Given an initial input p_0 for a feasible program, a rough outline of a numerical procedure to accomplish this can be presented as follows:

Step 1 From the solution of the IO problem at p_0, obtain a descent direction δp in the p space for the problem.

Step 2 Using this direction, obtain a step size $\bar{t}$ in the p space.

Step 3 Using $p = p_0 + t\delta p$, $t \in (0, \bar{t}\,]$, formulate the equations of appropriate first order necessary conditions (such as Fritz-John) in the reduced, scalar parameter space of t. Apply a continuation method for this locally parametrized process to identify the nature of critical points along this curve in the reduced t space.

Step 4 At a bifurcation point, identify the type of singularity, and continue if a minimum persists along a branch.

Note that in effect, we have here a combination of a descent and a continuation method (ref. 1 and ref. 12). It remains to be seen if the conjectured prevalence of singular critical points in practical IO problems justifies what appears to be a costly computational procedure. (**Fig. 7.**)

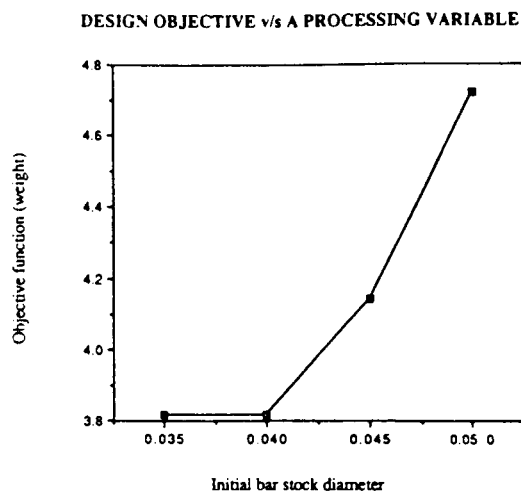DESIGN OBJECTIVE v/s A PROCESSING VARIABLE

Figure 7

Conclusions

The theory of optimality conditions of extremal problems can be extended to problems continuously deformed by an input vector. The connection between the sensitivity, well-posedness, stability and approximation of optimization problems is steadily emerging. We believe that the important realization here is that the underlying basis of all such work is still the study of point-to-set maps and of small perturbations, yet what has been identified previously as being just related to *solution procedures* is now being extended to study *modeling* itself in its own right.

Many important studies related to the theoretical issues of parametric programming and large deformation in nonlinear programming have been reported in the last few years, and the challenge now seems to be in devising effective computational tools for solving these generalized design optimization models.

Acknowledgements

References

[1] Zlobec S., 1987, "Survey of Input Optimization," *Optimization*, Vol 18, No. 3, 309-348.

[2] Huard P., ed., 1979, **Point-to-Set Maps and Mathematical Programming**, Mathematical Programming Study - 10, North Holland.

[3] Robinson S. M., and Day, R. H., 1974, "A Sufficient Condition for Continuity of Optimal Sets in Mathematical Programming," *Jl. of Mathematical Analysis and Applications*, 45, 506-511.

[4] Evans J. P., and Gould F. J., 1970, "Stability in Nonlinear Programming," *Operations Research*, 18(1), 107-118.

[5] Clarke Frank H., 1983, **Optimization and Nonsmooth Analysis**, John Wiley, New York.

[6] Fiacco A. V., 1983, **Introduction to Sensitivity and Stability Analysis in Nonlinear Programming**, Academic Press, New York.

[7] Robinson S, M., 1974, "Perturbed Kuhn-Tucker Points and Rates of Convergence for a Class of Nonlinear-Programming Algorithms," *Math. Programming 7*, 1-16.

[8] Gauvin J., 1977, "A Necessary and Sufficient Regularity Condition to have Bounded Multipliers in Nonconvex Programming," *Math. Programming 12*, 136-138.

[9] Malanowski K., 1987, **Stability of Solutions to Convex Problems of Optimization**, Springer-Verlag, Berlin.

[10] Poore A B., and Tiahart C. A., 1987, "Bifurcation Problems in Nonlinear Parametric Programming," *Math. Programming 39*, 189-205.

[11] Kojima M., and Hirabayashi R., 1984, "Continuous Deformation of Nonlinear Programs," *Math. Programming Study 21*, 150-198. North-Holland.

[12] Rheinboldt W, C., 1986, **Numerical Analysis of Parametrized Nonlinear Equations**, John-Wiley, New York.