LOCAL MONOTONICITY ANALYSIS

by
Shapour Azarm
Graduate Student
and
Panos Papalambros
Assistant Professor

Presented at the Symposium on
"Optimization and Its Engineering Uses"
TIMS/ORSA Joint National Meeting
Chicago, April 25-27, 1983
ABSTRACT

The method of monotonicity analysis has been used to solve analytically nonlinear programming formulations of engineering design problems. The analytical approach guarantees a global solution. This paper explores the possibility of implementing the method numerically for identification of only local minima. Actual examples are included.
INTRODUCTION

The importance of identifying the critical requirements for a design during the decision-making process is well recognized by engineering designers. In the mathematical programming context this corresponds to determining the active constraints at the optimal solution. The method of monotonicity analysis originally proposed by Wilde (1975) and developed by Papalambros (1979) aimed at establishing rules for activity identification in the presence of coordinatewise monotone functions. A recent summary of the developments in this area is given by Papalambros and Li (1982) together with additional references.

The original methodology is based on global monotonicity properties and the optimum identified is a proven global one, usually with many active constraints. For design problems, partly due to modeling practices, the optimum is often constraint-bound (i.e. a vertex solution) or nearly so. Thus global monotonicity analysis is very appealing. However, there are several limitations. The need for analytical results requires algebraic manipulations that can become intractable for larger problems. Thus one is led naturally to examining the possibility of coupling the ideas of monotonicity analysis with the more traditional numerical optimization schemes. This is particularly appealing because of the evident similarity between monotonicity analysis and the simplex method of linear programming.

A numerical implementation of monotonicity analysis will, in general, utilize local information, therefore it is referred to as Local Monotonicity Analysis (LMA). By necessity, monotonicity
properties will be only local estimates and will be essentially a basis for an active set strategy. A first attempt for a local procedure in an interactive mode was reported by Zhou and Mayne (1982). An automated local procedure was recently developed by Azarm and Papalambros (1983b). The present paper describes some results obtained using a particular algorithm based on this later procedure. The algorithm is referred to as ACCME (Automated Constraint Criticality by Monotonicity Evaluations).

The entire approach was put in the context of active set strategies by Azarm and Papalambros (1983a). The particular argument was made that monotonicity analysis rules can form the basis for an active set strategy that blends local and global information. In fact such information may be obtained by non-analytical means, such as design experience or intuition, yet used to guide the optimization process. This knowledge-based strategy can be useful when complete optimization models are unavailable, either because of high modeling cost or because of lack of good analytical understanding. A simple example of blending local and global rules is included for illustration at the end of this paper.

A short review of the concepts of monotonicity and dominance is described in the next section, followed by an explanation of the program ACCME; this is essentially a summary of the work contained in Azarm and Papalambros (1983a,b). Subsequent sections present new results of applying ACCME to test examples selected from Hock and Schittkowski (1981) and Himmelblau (1972). A design example involving the vehicle suspension optimization for riding comfort is included. The final example reexamines the
design of an explosively actuated cylinder which had been previously treated using global monotonicity analysis by Papalambros and Wilde (1980).
GLOBAL, REGIONAL AND LOCAL MONOTONICITY

The general case of non-differentiable monotone functions with possible discontinuities has been examined elsewhere. In that case, the proofs of various properties are based upon the concept of boundedness, see Papalambros (1979). Here the discussion is restricted to smooth functions using the Karush-Kuhn-Tucker (KKT) optimality conditions. The regularity assumption of the points under consideration is that the gradient vectors of the active constraints are linearly independent at the regular points.

We consider the problem

minimize \( f(\bar{z}) \)

subject to \( g(\bar{z}) \leq 0 \) \hspace{1cm} (1)

where \( \bar{z} \in \mathbb{R}^n \) and \( g = (g_1, \ldots, g_m)^T \), the functions being all real-valued. We assume that any equality constraints have been eliminated already, either through direct elimination or through an appropriate projection of the original problem to the subspace of the equality constraints. Now, let \( \bar{z} \) be a solution (local minimum) to (1). We may partition \( \bar{z} \) in the following way:

\[
\bar{z} \triangleq (\bar{x}, \bar{u})^T, \quad \bar{x} = (\bar{x}_1, \ldots, \bar{x}_{n_1}), \quad \bar{u} = (\bar{u}_1, \ldots, \bar{u}_{n_2})
\]

\[
\bar{x}_i \in X = \{ \bar{x}_i \mid \partial f/\partial \bar{x}_i \neq 0, \text{ for } \bar{x}_i = \bar{z}_i, i \in I_1 \} \hspace{1cm} (2)
\]

\[
\bar{u}_i \in U = \{ \bar{u}_i \mid \partial f/\partial \bar{u}_i = 0, \text{ for } \bar{u}_i = \bar{z}_i, i \in I_2 \}
\]

The sets of indices \( I_1 = 1, \ldots, n_1 \), \( I_2 = 1, \ldots, n_2 \), with \( n_1 + n_2 = n \), are obtained after rearrangement. This partitioning simply segregates the variables which, if perturbed, may or may
not produce a change in the objective function, at least to first order.

The KKT conditions at $\bar{z}$ may now be expresses as follows (all evaluations performed at $\bar{z} = (\bar{x}, \bar{u})$):

$$\frac{\partial f}{\partial x_i} + \sum_{j=1}^{m} \mu_j \left(\frac{\partial g_j}{\partial x_i}\right) = 0 , \quad i \in I_1$$

$$\mu_j \geq 0 , \quad \mu_j g_j = 0 , \quad j \in J_1$$

$$\sum_{j=1}^{m} \mu_j \left(\frac{\partial g_j}{\partial u_i}\right) = 0 , \quad i \in I_2$$

$$\mu_j \geq 0 , \quad \mu_j g_j = 0 , \quad j \in J_2$$

where $J_1, J_2 \subseteq J , J = \{1, \ldots, m\}$. Clearly, if the set $X$ is empty, the point $\bar{z}$ is an interior (unconstrained) solution and the $\bar{u}$ values may be representing a range, as long as feasibility is not violated. If $X$ is non-empty, then the solution is on the boundary and some constraints must be active. In fact, the following cases may occur:

1) For each $i \in I_1$ with $\frac{\partial f}{\partial x_i} > 0$, there exists a $j \in J_1$, such that $\mu_j > 0$ and $\frac{\partial g_j}{\partial x_j} < 0$.

2) For each $i \in I_1$, with $\frac{\partial f}{\partial x_i} < 0$, there exists a $j \in J_1$, such that $\mu_j > 0$ and $\frac{\partial g_j}{\partial x_j} > 0$.

3) If $i \in I_2$ and $\frac{\partial g_j}{\partial u_i} = 0$ for all $j \in J_2$, then the point is not a regular one.

4) If $i \in I_2$ and $\frac{\partial g_j}{\partial u_i} > 0$ for at least one $j \in J_2$, then there are two possibilities:
(a) If \( \mu_j > 0 \), then there exists at least one \( j' \in J_2 \) such that \( \mu_{j'} > 0 \) and \( \partial g_j / \partial u_i < 0 \).

(b) If \( \mu_j = 0 \), then (excluding degeneracy with active constraint and zero multiplier) either \( \mu_{j'} = 0 \) for all \( j' \in J_2 \), \( j' \neq j \); or, there exists a \( j' \neq j \) with \( \partial g_j / \partial u_i \neq 0 \) and case 4(a) or 5(a) applies.

5) If \( i \in I_2 \) and \( \partial g_j / \partial u_i < 0 \), then similar arguments as under 4) can be made.

The above discussion has assumed all evaluations performed at \((\bar{x}, \bar{y})\), so the information is local. If we agree to say that, a constraint \( g_j \leq 0 \) is monotonic with respect to \((w.r.t.)\) a component \( x_i \) or \( u_i \), if the function \( g_j \) is monotonic \( w.r.t \) the same variable, then locally we have from above cases the obvious rules:

1. If the objective is monotonic \( w.r.t. \) a particular variable in the neighborhood of a local minimum, then there exists at least one active constraint with opposite monotonicity \( w.r.t. \) that variable in that neighborhood.

2. If the objective is stationary \( w.r.t. \) a particular variable in the neighborhood of a local minimum, then either all constraints containing that variable are inactive, or there exist at least two active constraints having opposite monotonicities \( w.r.t. \) that variable in that neighborhood.

We may refer to the above properties as **local monotonicity** rules. However, if we imagine the "neighborhood" expanding to include a range of values beyond the linear approximation of the functions, then we may refer to a **regional monotonicity**. In the extreme case, where the entire feasible space is included, we
have \textit{global} \textbf{monotonicity}.

The closely associated concept of constraint dominance can be put in a similar framework. Suppose, for simplicity, that two constraints \( g_1(z) \leq 0 \) and \( g_2(z) \leq 0 \) can be resolved at a point \( \bar{z} \) to take the form \( \bar{z}_i = B_1(\bar{z}_i^C) \) and \( \bar{z}_i = B_2(\bar{z}_i^C) \) respectively. Here \( \bar{z}_i \) is some coordinate of \( \bar{z} \) and \( \bar{z}_i^C = (z_1, \ldots, \bar{z}_{i-1}, \bar{z}_{i+1}, \ldots, z_n) \). If we can show that

\[ B_1(\bar{z}_i^C) \leq B_2(\bar{z}_i^C) \quad (4) \]

then we say that \( g_1 \) is \textit{locally} dominant over \( g_2 \) (w.r.t. \( z_i \)) at \( \bar{z} \). This means that if a lower bound is required for \( z_i \), \( g_1 \) will provide the greatest lower bound. Similarly for an upper bound.

If now we consider the above condition (4) holding true over a range of values for (all components of) \( \bar{z}_i^C \) then we have \textit{regional} dominance, and if that range is the entire feasible domain we have \textit{global} dominance. This latter case generally means that the dominated constraint may be redundant.
COMPUTATIONAL IMPLEMENTATION

A flow chart of the main steps in program ACCME is shown in Figure 1. The starting point labelled "current point" can be either the first one supplied by the user, or the result of a previous iteration. As will be pointed out later, it need not be feasible. At the current point, the partial derivatives of the objective and constraint functions are evaluated w.r.t. each component of \( x \). If some inequality or equality constraints are active at this point, then the partial constrained derivatives are evaluated. This is a well-known operation, see, e.g., Wilde and Beightler (1967).

With the derivative information known, the point can be tested for the KKT conditions which are taken here as the optimality criterion. If the point is not the optimum, then the first rule is applied w.r.t. to all current decision variables. If several candidate variables exist, the new active variable is taken to be the one corresponding to the largest (absolute) partial derivative of the objective. This decision is similar to determining the entering basic variable in the Simplex Method of linear programming and aims at producing the locally largest amount of objective function descent. Given a new active variable, there may or may not exist candidate active constraints. If none exists, then the program switches to a usual descent procedure which generates a new point. The descent-type steps will continue until either the optimum or a new active constraint has been identified.

If several candidate active constraints have been identified, a simple dominance decision is made to select the active one. The
distance of the intercept of the tangent to the constraint, at the current point, with the particular coordinate axis, is used as the measure of dominance. The decision is simply made as follows: Let \( x_i \) be the current candidate solution variable and \( g_j, 1 \leq j \leq q \), be the candidate active constraints; then

a) if \( \partial f / \partial x_i > 0 \), find the \( j, 1 < j < q \), such that \( \max_j \{ x_i - g_j / (\partial g_j / \partial x_i) \} \);

b) if \( \partial f / \partial x_i > 0 \), find the \( j, 1 < j < q \), such that \( \min_j \{ x_i - g_j / (\partial g_j / \partial x_i) \} \).

This decision is interpreted graphically in Figure 2. In Figure 2(a), where \( j = 3 \), the intercepts of the tangents to \( g_1, g_2, g_3 \) at \( x_i(k) \) are the points 1, 2 and 3 respectively on the \( x_i \) axis. Point 2 is furthest from the origin, therefore constraint \( g_2 \) is considered locally dominant. Similarly, in Figure 2(b), constraint \( g_1 \) is dominant. Again, this is similar to determining the leaving basic variable in linear programming.

The next step is to apply the second rule in an attempt to identify more active constraints. If several candidates exist, a dominance decision is taken as before. Thus we arrive at the point where a new active set has been identified. They represent a system of equations which must be solved in order to move from the current point to the new one. Since the current and new point may be substantially far from each other, some difficulty may be anticipated in the solution of the nonlinear system.

After the new point is evaluated, our previous local prediction of monotonicities must be checked. First the objective function is examined; if a monotonicity has not been preserved and a change of sign in the partial derivative is found, we deactivate
the corresponding constraint(s) and revert to a one-dimensional minimization between the current and the new point. Then the constraint functions are examined; if a change of sign is observed, then a descent move is initiated from the current point. If all monotonicities are preserved, then the new point is retained as is, to serve as the current point of the next iteration. Here there is an argument that there may be more than one sign changes between current and new point. Although no specific measures are taken against this possibility, there is no evidence yet that this can cause the algorithm to break down. Note that in both cases above, where a stepwise search is initiated, this is done after the offending constraints have been deactivated.
TWO-DIMENSIONAL EXAMPLES

The examples in this section have been selected for illustrating the iteration steps followed by ACCME on two-dimensional contour maps.

Example 1. Reported by Hock and Schittkowski (1981) as problem No. 18, it is stated as follows (Figure 3):

\[
\text{minimize } f(x) = 0.01 x_1^2 + x_2^2 \\
\text{subject to} \\
g_1 : 25 - x_1 x_2 \leq 0 \\
g_2 : 25 - x_1^2 - x_2^2 \leq 0 \\
\text{and} \\
2 \leq x_1 \leq 50 , \quad 0 \leq x_2 \leq 50
\]

The program starts at (2,2), an infeasible point, and proceeds to (2,12.5) where \(g_1\) is active together with the lower bound on \(x_1\). Then it attempts to get a better bound on \(x_1\) by making \(g_2\) active. However, since \(g_1\) and \(g_2\) have no common intersection, it ends at point (4.865,2.795) which is infeasible. Keeping \(g_1\) active, two attempts to reduce \(x_2\) to its lower bound of zero are made followed by 1D searches giving the subsequent points (11.453,2.1829) and (15.811,1.5811). The latter satisfies the KKT conditions and the program ends.

Example 2. Reported by Hock and Schittkowski (1981) as problem No. 20, it is stated as follows (Figure 4):

\[
\text{minimize } f(x) = 100 (x_2 - x_1^2)^2 + (1-x_1)^2 \\
\text{subject to} \\
\]
\begin{align*}
g_1 & : \quad -x_1 - x_2^2 \leq 0 \\
g_2 & : \quad -x_2 - x_1^2 \leq 0 \\
g_3 & : \quad 1 - x_1^2 - x_2^2 \leq 0 \\
\end{align*}

and
\[-0.5 \leq x_1 \leq 0.5\]

This problem is slightly nonconvex with a vertex solution. The program starts at the infeasible point \((-2,1)\) and proceeds to \((-0.0002,1)\) where \(g_3\) is taken active. The next decision is to keep \(g_3\) active and also bring \(x_1\) to its upper bound 0.5. The vertex \((0.5,0.86603)\) satisfies the KKT conditions and the program terminates.

Example 3. Reported by Hock and Schittkowski (1981) as problem No. 23, it is stated as follows (Figure 5):

\begin{align*}
\text{minimize} & \quad f(x) = x_1^2 + x_2^2 \\
\text{subject to} & \\
\quad g_1 & : \quad 1 - x_1 - x_2 \leq 0 \\
\quad g_2 & : \quad 1 - x_1^2 - x_2^2 \leq 0 \\
\quad g_3 & : \quad 9 - 9x_1^2 - x_2^2 \leq 0 \\
\quad g_4 & : \quad x_2 - x_1^2 \leq 0 \\
\quad g_5 & : \quad x_1 - x_2^2 \leq 0 \\
\end{align*}

and
\[-50 \leq x_i \leq 50 \quad , \quad i = 1,2\]

The problem is again slightly nonconvex with a corner solution at a cusp. The program starts at the infeasible point \((3,1)\) and moves to \((1,1)\) in order to make \(g_5\) active. At this point, \(g_4\) is also
requested to become active and since this requirement is already satisfied, the point is checked against the termination criteria and the program ends after one iteration.

**Example 4.** Constructed by Himmelblau (1972), this problem has a complicated objective function exhibiting several extrema and a saddle point with a variety of constraints. The problem statement is given in Appendix A and a graphical representation is shown in Figure 6. This particular version has an extremely narrow feasible domain and several redundant constraints. The program ACCME starts at the infeasible point \((95.0, 10.0)\) and proceeds to \((70.0, 10.0)\) making \(g_5\) active. Next, constraint \(g_{12}\) is brought in the active set and the intersection \((21.9726, 31.8579)\) is located. At the next iteration \(g_5\) is dropped from the active set and \(g_9\) is brought in. The new point located is \((54.87847, 52.42405)\) where both \(g_9\) and \(g_{12}\) are supposed to be active. The program terminates having verified the KKT conditions. A closer observation shows that at this point \(g_9\) is actually loose, rather than tight, but numerical imprecision in derivative calculations appears to inhibit the algorithm from moving further. It is interesting to note that all methods that performed well terminated at about the same point. The actual optimal point constructed as the intersection of four constraints is singular and very difficult to detect.
EXAMPLES OF HIGHER DIMENSION

In this section two larger problems will be discussed. They are both drawn from Hock and Schittkowski (1981) (abbr. H & S (1981)) but they have been reported in several other sources. The reader is referred to H & S (1981) for further details.

Example 5. Reported by H & S (1981) as problem No. 83, this problem has five variables, six inequality constraints, and ten variable bounds. It is mildly nonlinear. From the present viewpoint, two of the variables do not appear in the objective function and this fact is exploited by the second monotonicity rule to achieve fast convergence.

The problem statement is:

\[
\text{minimize } f(x) = b_1 x_3^2 + b_2 x_1 x_5 + b_3 x_1 + b_4
\]

subject to:

\[
g_4 : \quad 0 \leq a_1 + a_2 x_2 x_5 + a_3 x_1 x_4 - a_4 x_3 x_5 \leq 92 \quad : g_1
\]

\[
g_5 : \quad 0 \leq a_5 + a_6 x_2 x_5 + a_7 x_1 x_2 + a_8 x_3^2 - 90 \leq 20 \quad : g_2
\]

\[
g_6 : \quad 0 \leq a_9 + a_{10} x_3 x_5 + a_{11} x_1 x_3 + a_{12} x_3 x_4 - 20 \leq 5 \quad : g_3
\]

\[
g_8 : \quad 78 \leq x_1 \leq 102 \quad : g_7
\]

\[
g_{10} : \quad 33 \leq x_2 \leq 45 \quad : g_9
\]

\[
g_{12} : \quad 27 \leq x_3 \leq 45 \quad : g_{11}
\]

\[
g_{14} : \quad 27 \leq x_4 \leq 45 \quad : g_{13}
\]

\[
g_{16} : \quad 27 \leq x_5 \leq 45 \quad : g_{15}
\]

\[a_1, b_1 \text{ are given in } H & S (1981).\]

The program starts at \((78, 33, 27, 27, 27)\), an infeasible point, and proceeds to \((78, 33, 34.433, 45.27)\) where \(g_6\) and \(g_{13}\) are found to be
active by first and second monotonicity rules. Next it is found that by both rules $g_1$ and $g_{10}$ are active so that the program moves to $(78,33,29.995,45,36.776)$, where it is found that $g_8$ is active, therefore the point $(78,33,29.995,45,36.776)$ is reached, which satisfies the KKT conditions and the program ends. The same problem is solved from the feasible point $(78.62,33.44,31.07,44.18,35.22)$ suggested by Himmelblau (1972) and converged to the same point.

**Example 6.** Reported by H & S (1981) as problem No. 86. There are five variables, a polynomial objective function, ten linear inequality constraints, and five variable bounds. The problem statement is:

\[
\text{minimize } f(x) = \sum_{j=1}^{5} e_j x_j + \sum_{i=1}^{5} \sum_{j=1}^{5} c_{ij} x_i x_j + \sum_{j=1}^{5} d_j x_j^3
\]

subject to

\[
\sum_{j=1}^{5} a_{ij} x_j - b_i \geq 0 \quad , \quad i = 1,10
\]

\[
x_i \geq 0 \quad , \quad i = 1,5
\]

$a_{ij}$, $b_i$, $c_{ij}$, $d_j$, $e_j$ are given in H & S (1981).

The program starts at $(0,0,0,0,1)$, feasible point, where it is found that with respect to $x_4$ constraint $g_4$ may be active. Taking $g_4$ active resulted to the monotonicity change in the objective function, thus $g_4$ is deactivated and a one dimensional search on the objective function with respect to $x_4$ is done to find the point $(0,0,0,0.647,1.)$. In the following iterations it is found that $g_6$, $g_3$, $g_5$, and $g_9$ are active which results to the point $(9.3,0.33347,0.4,0.42831,0.22396)$. This point satisfies KKT conditions and the program ends.
A DESIGN APPLICATION

In this application, the optimal design of a passive suspension is stated in a nonlinear programming format. The problem has been modelled and solved by global monotonicity analysis in Lu, Li and Papalambros (1983). The dynamic model shown in Figure 7 is used and vehicle vibrations and dynamic loads are calculated based on statistical analysis of random input of road roughness. The details can be found in the above reference. Here the final model, after rearrangements, is presented. The necessary nomenclature is given in Appendix B.

The problem is to minimize the mean square value of the vertical vibration acceleration of the vehicle body, which represents a measure of the riding comfort of the vehicle. The constraints used are as follows:

$g_1$: The road holding ability of the vehicle should not be below some acceptable limit; this is represented by limiting the tire-road relative dynamic load below a specified probability value $b_0$ that the vehicle tires will leave the ground.

$g_2$: The rolling angle of the vehicle must be limited from above.

$g_3$: The suspension dynamic displacement must be limited so that bumper hitting is avoided; this is often referred as the rattle-space constraint.

$g_4$: A minimum allowable value on tire stiffness must be imposed because the tire life is an increasing function of the tire stiffness.

The complete model is:

\[
\text{minimize } Z^2 = (\pi AV/m^2)(C_K + (M+m)C)^2K^{-1}
\]
subject to:

\[ g_1 : \quad \frac{\pi AV_m}{b_0 g_{\infty}^2} \left( \frac{C_k}{M+M} \right)^{C_k^2} + \frac{C^2}{mM} + \frac{C_k^2}{mM^2} \leq 1 \]

\[ g_2 : \quad 7.6394 \left( 4000 (Mg)^{-1.5} C_1 \right)^{-1} \leq 1 \quad (10) \]

\[ g_3 : \quad 0.5 \left( Mg \right)^{1/2} \left( k^2 C_k C^{-1} (M+m)^{-1} + C \right)^{-1/2} \leq 1 \]

\[ g_4 : \quad \left( (M+m)g \right)^{0.877} C_k^{-1} \leq 1 \]

where \( C, C_k \) and \( K \) are the design variables and \( A, V, M, m, b_o \) are design parameters.

In the original paper referenced above, a large number of cases were examined during a parametric study in order to produce general design rules. Here we will only demonstrate with one numerical example. The parameter values selected are:

\[ A = 1 \text{cm}^2/\text{cycle/m} \quad , \quad V = 10 \text{m/sec} \quad , \quad M = 3.2633 \text{ Kg sec}^2/\text{cm} \]

\[ m = 0.8158 \text{ Kg sec}^2/\text{cm} \quad , \quad b_o = 0.27. \]

The program ACCME starts at \((100, 200, 50)\), an infeasible point, where constraint \( g_3 \) is detected to be active. Iteration \#2 starts with \((100, 200, 37.8)\), still infeasible, where constraint \( g_2 \) is active. Program continues to iteration \#3 with infeasible point \((391.2, 200, 57.1)\) where constraint \( g_4 \) is found active. Having \( g_2, g_3, \) and \( g_4 \) active the program goes to the point \((391.21, 1442.6, 21.27)\) where it satisfies the KKT conditions and program ends.
IMPLEMENTATION OF GLOCAL RULES

The blending of global and local information in an active set strategy will be now demonstrated in the context of monotonicity through a well-studied example. This problem involves the design of an explosively actuated cylinder and was first studied in Siddall (1972) and later it was solved by Johnson's method, Ellis (1976), and monotonicity analysis, Papalambros and Wilde (1980). Zhou and Mayne (1982) used it as an example of their algorithm for monotonicity analysis. The model has the following form:

\[ \text{minimize} \quad f = x_1 + x_2 \]

subject to

\[ g_1 = -x_1 \leq 0 \quad g_2 = -x_2 \leq 0 \]
\[ g_3 = -x_3 \leq 0 \quad g_4 = -x_4 \leq 0 \]
\[ g_5 = -x_5 \leq 0 \quad g_6 = x_3 - 1 \leq 0 \]
\[ g_7 = 600 + 5000 x_4 V_1^{1.2} (V_2^{-0.2} - V_1^{-0.2}) \leq 0 \]
\[ V_1 = 0.084 + (\pi/4)x_1 x_5^2 \]
\[ V_2 = V_1 + (\pi/4)x_2 x_5^2 \]
\[ g_8 = 1000(\pi/4)x_4 x_5^2 - 700 \leq 0 \]
\[ g_9 = [x_4 (x_3^2 + x_5^2)] / [x_3^2 - x_5^2] + x_4 - 41.6667 \leq 0 \]
\[ g_{10} = x_5 - x_3 \leq 0 \]
\[ g_{11} = x_1 + x_2 - 2 \leq 0 \]

This problem was first solved directly with local analysis using ACCME and a starting point of \((0.2, 1.5, 0.8, 10.0, 0.032)\). The fol-
lowing iterations occurred: 1) Constraint $g_1$ is made active with $x_1$ the active variable; the current point is infeasible; the new point is $(0,1.5,0.018,10.0,0.32)$ and also infeasible; 2) At the new point, $g_7$ is made active with $x_2$ being the active variable from the first rule; $g_8$ is made active with $x_4$ being the active variable from the second rule; the new point generated is $(0,1.515,0.80,8.70,0.32)$; 3) At this new point, $g_9$ is now made active with $x_5$ the active variable from the first rule; $g_6$ is made active with $x_3$ the active variable from the second rule; previously active constraints remain in the set; the new point is $(0.0,1.0777,1.0000,19.900,0.2116)$; 4) At the new point, KKT conditions are satisfied; the optimum is reached and it is constraint-bound.

Now let us examine the global information. It was shown in Papalambros and Wilde (1979), with little analytical effort and a more judicious study of the model, that the monotonicity rules can be utilized **globally** to yield the following design rules:

1) Constraint $g_7$ is always active.
2) Constraints $g_{10}$ and $g_{11}$ are always inactive.
3) Either $g_8$, or $g_9$, or both are active.
4) Constraints $g_9$ and $g_6$ can be active only simultaneously.

These rules were incorporated in ACCME in a very simple way: whenever a decision on activity was called for, choices based on local information were overridden by applicable global rules, as for example in the dominance decisions.

The iterations executed by ACCME in this new version are shown in Figures 8 and 9. These figures represent typical output from ACCME.
Some observations can be made: The number of iterations in the purely local procedure and in the enhanced with global rules one, is the same. The points however are not the same, i.e. a slightly different path is followed. Looking at the values of the objective function, the local procedure generates the sequence \{1.700, 1.500, 1.5153, 1.0777\}, while the global-local one gives the sequence \{1.700, 1.5164, 1.500, 1.0777\}. Thus, in the second case the objective is monotonically decreasing at each iteration, and feasibility is maintained after the first iteration. Our current experience and algorithmic analysis of ACCME does not allow us to claim a generalization of this result. However, it represents encouraging evidence of the usefulness of the global rules.
CONCLUSION

The local utilization of monotonicity rules appears promising for an efficient active set strategy. A complete algorithm in a standard nonlinear programming format must utilize also numerical search methods for the unconstrained optimization part. The gradient search implemented in ACCME is not necessarily the most suitable one. Global convergence of ACCME has not yet been proven. In general, the proof will depend on the particular choice of strategy for using the monotonicity rules.

An additional difficulty that needs further investigation is the solution of the nonlinear system of active constraints. Convergence may be inhibited when large changes in the values of the design variables are required. Also, for highly nonlinear problems a potential multitude of constraint intersections far outside the feasible domain may make the current procedure inefficient by requiring additional steps for regaining feasibility.

Acknowledgement

This research was partially supported by NSF Grant CME-80-06687. This support is gratefully acknowledged.
REFERENCES


APPENDIX A: Example 4 Problem Statement

minimize \( f(x) = B_1 + B_2 x_1 + B_3 x_1^2 + B_4 x_1^3 + B_5 x_1^4 + \ldots \)

\[ + B_6 x_2 + B_7 x_1 x_2 + B_8 x_1^2 x_2 + \ldots \]

\[ + B_{10} x_1^4 x_2 + B_{11} x_2^2 + B_{12} x_2^3 + B_{13} x_2^4 \]

\[ + B_{14}/(x_2+1) + B_{15} x_1^2 x_2 + B_{16} x_1^3 x_2^2 \]

\[ + B_{17} x_1^3 x_2 + B_{18} x_1 x_2^2 + B_{19} x_1 x_2^3 \]

\[ + B_{20} \exp[0.0005 x_1 x_2] \}

subject to:

\[ g_5 : \quad x_1 x_2 - 700 \geq 0 \]

\[ g_6 : \quad 75 - x_1 \geq 0 \]

\[ g_7 : \quad 65 - x_2 \geq 0 \]

\[ g_8 : \quad x_2 - 5 (x_1/25)^2 \geq 0 \]

\[ g_9 : \quad (x_1 - 50)^2 - 5(x_1 - 55) \geq 0 \]

\[ g_{10} : \quad 1.5(x_2 - 45) - (x_1 - 45) \geq 0 \]

\[ g_{12} : \quad x_1 - 35 - \frac{40}{25} (x_2 - 40) \geq 0 \]

\( B_i \) are given in Himmelblau (1972).
APPENDIX B:  Nomenclature for Vehicle Suspension Design

A = road irregularity coefficient (cm$^2$ cycle/m)

b_o = dynamic load coefficient

C = suspension stiffness (kg/cm)

C_K = tire stiffness (kg/cm)

g = gravity acceleration (cm/sec$^2$)

K = damping force coefficient (kg/cm/sec)

M = spring mass (kg/cm/sec$^2$)

m = unsprung mass (kg/cm/sec$^2$)

V = vehicle velocity (m/sec)

z = vertical displacement of sprung mass (vehicle body) (cm)

$\ddot{z}$ = the mean square value of vertical vibration acceleration of vehicle body (cm/sec$^2$)
Figure 3
Figure 6
Automated Constraint Criticality by Monotonicity Evaluations

***************ITERATION NO. (1)***************

Current point:

Values of design variables:
- $x(1) = 0.20000$
- $x(2) = 1.50000$
- $x(3) = 0.80000$
- $x(4) = 10.000$
- $x(5) = 0.32000$

Value of objective function:
- $f = 1.7000$

Values of constraints:
- $g(1) = -0.20000$
- $g(2) = -1.50000$
- $g(3) = -0.80000$
- $g(4) = -10.000$
- $g(5) = -0.32000$
- $g(6) = -0.20000$
- $g(7) = -26.426$
- $g(8) = 104.25$
- $g(9) = -17.857$
- $g(10) = -0.48000$
- $g(11) = -0.30000$

THIS POINT IS INFEASIBLE!!

Monotonicity and dominance analysis:

For this point, based on first monotonicity rule, one of the following constraints wrt $x(1)$ may be active:
- $g(1)$
- $g(7)$

NOTE: At any point, based on first monotonicity rule, wrt $x(1)$ constraint $g(7)$ is GLOBALLY active.

For this point, based on second monotonicity rule, one of the following constraints wrt $x(4)$ may be active:
- $g(8)$
- $g(9)$

For this point, wrt $x(4)$ constraint $g(8)$ may be dominant.

Decisions taken for this iteration:

>>> AT THIS ITERATION, BASED ON FIRST MONOTONICITY RULE,
>>> WRT $x(1)$ CONSTRAINT $g(7)$ IS ACTIVE.
>>> AT THIS ITERATION, BASED ON SECOND MONOTONICITY RULE,
>>> WRT $x(4)$ CONSTRAINT $g(8)$ IS ACTIVE.

***************ITERATION NO. (2)***************

Current point:

Values of design variables:
- $x(1) = 0.16409E-01$
- $x(2) = 1.50000$
- $x(3) = 0.80000$
- $x(4) = 8.7038$
- $x(5) = 0.32000$

Value of objective function:
- $f = 1.5164$

Values of constraints:
- $g(1) = -0.16409E-01$
- $g(2) = -1.50000$
- $g(3) = -0.80000$
- $g(4) = -9.7038$
- $g(5) = -0.32000$
- $g(6) = -0.20000$
- $g(7) = 0.27780E-07$
- $g(8) = 0.56843E-13$
- $g(9) = -20.943$
- $g(10) = -0.48000$
- $g(11) = -0.48359$

Monotonicity and dominance analysis:

For this point, based on first monotonicity rule, one of the following constraints wrt $x(5)$ may be active:
- $g(1)$
- $g(5)$
- $g(9)$

For this point, wrt $x(5)$ constraint $g(1)$ may be dominant.
Monotonicity of active constraints are preserved.

Decisions taken for this iteration:

>>> AT THIS ITERATION, BASED ON FIRST MONOTONICITY RULE,
>>> CONSTRAINT $g(1)$ IS ACTIVE.
ITERATION NO. (3)

Current point:

Values of design variables:
\[ x(1) = -0.47003E-23 \]
\[ x(2) = 1.5000 \]
\[ x(3) = 0.80000 \]
\[ x(4) = 8.8405 \]
\[ x(5) = 0.31752 \]

Value of objective function:
\[ f = 1.5000 \]

Values of constraints:
\[ g(1) = 0.47003E-23 \]
\[ g(2) = -1.5000 \]
\[ g(3) = 0.80000 \]
\[ g(4) = -8.8405 \]
\[ g(5) = 0.31752 \]
\[ g(6) = 0.20000 \]
\[ g(7) = 0.61819E-11 \]
\[ g(8) = 0.19423E-09 \]
\[ g(9) = -20.680 \]
\[ g(10) = 0.46248 \]
\[ g(11) = 0.50000 \]

Monotonicity and dominance analysis:

For this point, based on first monotonicity rule,
one of the following constraints wrt \( x(2) \) may be active:
\[ g(2), g(5), g(9) \]
For this point, wrt \( x(2) \) constraint \( g(9) \) may be dominant.

For this point, based on second monotonicity rule,
one of the following constraints wrt \( x(3) \) may be active:
\[ g(6) \]

Note: At any point, based on second monotonicity rule,
constraints \( g(6) \) and \( g(9) \) must be simultaneously active.

Decisions taken for this iteration:

>>> AT THIS ITERATION, BASED ON FIRST MONOTONICITY RULE,
>>> WRT x(2) CONSTRAINT g(9) IS ACTIVE.
>>> AT THIS ITERATION, BASED ON SECOND MONOTONICITY RULE,
>>> WRT x(3) CONSTRAINT g(6) IS ACTIVE.
>>> FROM ITERATION(1): WRT x(1) CONSTRAINT g(7) IS ACTIVE.
>>> FROM ITERATION(1): WRT x(4) CONSTRAINT g(8) IS ACTIVE.
>>> FROM ITERATION(2): WRT x(5) CONSTRAINT g(1) IS ACTIVE.

ITERATION NO. (4)

Current point:

Values of design variables:
\[ x(1) = 0.23799E-23 \]
\[ x(2) = 1.0777 \]
\[ x(3) = 1.0000 \]
\[ x(4) = 19.900 \]
\[ x(5) = 0.21163 \]

Value of objective function:
\[ f = 1.0777 \]

Values of constraints:
\[ g(1) = -0.23799E-23 \]
\[ g(2) = -1.0777 \]
\[ g(3) = -1.0000 \]
\[ g(4) = -19.900 \]
\[ g(5) = -0.21163 \]
\[ g(6) = 0.0 \]
\[ g(7) = 0.39790E-12 \]
\[ g(8) = 0.11369E-12 \]
\[ g(9) = -0.35527E-14 \]
\[ g(10) = -0.78837 \]
\[ g(11) = -0.92225 \]

Decisions taken for this iteration:

>>> FROM ITERATION(1): WRT x(1) CONSTRAINT g(7) IS ACTIVE.
>>> FROM ITERATION(1): WRT x(4) CONSTRAINT g(8) IS ACTIVE.
>>> FROM ITERATION(2): WRT x(5) CONSTRAINT g(1) IS ACTIVE.
>>> FROM ITERATION(3): WRT x(2) CONSTRAINT g(9) IS ACTIVE.
>>> FROM ITERATION(3): WRT x(3) CONSTRAINT g(6) IS ACTIVE.

Figure 9

**TERMINATION CRITERION WRT KARUSH-KUHN-TUCKER CONDITIONS SATISFIED**
Automated Constraint Criticality by Monotonicity Evaluations

***************ITERATION NO. (1)***************

Current point:

Values of design variables:
\[ x(\, 1) = 0.20000 \]
\[ x(\, 2) = 1.5000 \]
\[ x(\, 3) = 0.80000 \]
\[ x(\, 4) = 10.000 \]
\[ x(\, 5) = 0.32000 \]

Value of objective function:
\[ f = 1.7000 \]

Values of constraints:
\[ g(\, 1) = -0.20000 \]
\[ g(\, 2) = -1.5000 \]
\[ g(\, 3) = -0.80000 \]
\[ g(\, 4) = -10.000 \]
\[ g(\, 5) = -0.32000 \]
\[ g(\, 6) = -0.20000 \]
\[ g(\, 7) = -26.426 \]
\[ g(\, 8) = 104.25 \]
\[ g(\, 9) = -17.857 \]
\[ g(\, 10) = -0.48000 \]
\[ g(\, 11) = -0.30000 \]

THIS POINT IS INFEASIBLE!!

Monotonicity and dominance analysis:

For this point, based on first monotonicity rule,
one of the following constraints wrt \( x(\, 1) \) may be active:
\[ g(\, 1) \]
\[ g(\, 7) \]

NOTE: At any point, based on first monotonicity rule,
\( x(\, 1) \) constraint \( g(\, 7) \) is GLOBALLY active.

For this point, based on second monotonicity rule,
one of the following constraints wrt \( x(\, 4) \) may be active:
\[ g(\, 6) \]
\[ g(\, 9) \]

For this point, \( x(\, 4) \) constraint \( g(\, 8) \) may be dominant.

Decisions taken for this iteration:

>>>>>AT THIS ITERATION, BASED ON FIRST MONOTONICITY RULE,
>>>>>WRT \( x(\, 1) \) CONSTRAINT \( g(\, 7) \) IS ACTIVE.
>>>>>AT THIS ITERATION, BASED ON SECOND MONOTONICITY RULE,
>>>>>WRT \( x(\, 4) \) CONSTRAINT \( g(\, 8) \) IS ACTIVE.
Current point:

Values of design variables:
\[
\begin{align*}
x(1) &= 0.16409E-01 \\
x(2) &= 1.5000 \\
x(3) &= 0.80000 \\
x(4) &= 8.7038 \\
x(5) &= 0.32000 \\
\end{align*}
\]

Value of objective function:
\[ f = 1.5164 \]

Values of constraints:
\[
\begin{align*}
g(1) &= -0.16409E-01 \\
g(2) &= -1.5000 \\
g(3) &= -0.80000 \\
g(4) &= -8.7038 \\
g(5) &= -0.32000 \\
g(6) &= -0.20000 \\
g(7) &= 0.17860E-07 \\
g(8) &= 0.56843E-13 \\
g(9) &= -20.943 \\
g(10) &= -0.48000 \\
g(11) &= -0.48359 \\
\end{align*}
\]

Monotonicity and dominance analysis:

For this point, based on first monotonicity rule, one of the following constraints wrt \( x(5) \) may be active:
\[
\begin{align*}
g(1) \\
g(5) \\
g(9) \\
\end{align*}
\]

For this point, wrt \( x(5) \) constraint \( g(1) \) may be dominant. Monotonicity of active constraints are preserved.

Decisions taken for this iteration:

>>> AT THIS ITERATION, BASED ON FIRST MONOTONICITY RULE, 
>>> WRT \( x(5) \) CONSTRAINT \( g(1) \) IS ACTIVE. 
>>> FROM ITERATION (1): WRT \( x(1) \) CONSTRAINT \( g(7) \) IS ACTIVE. 
>>> FROM ITERATION (1): WRT \( x(4) \) CONSTRAINT \( g(8) \) IS ACTIVE.
Current point:

Values of design variables:
\[ x(1) = -0.47003 \times 10^{-23} \]
\[ x(2) = 1.5000 \]
\[ x(3) = 0.80000 \]
\[ x(4) = 8.8405 \]
\[ x(5) = 0.31752 \]

Value of objective function:
\[ f = 1.5000 \]

Values of constraints:
\[ g(1) = 0.47003 \times 10^{-23} \]
\[ g(2) = -1.5000 \]
\[ g(3) = -0.80000 \]
\[ g(4) = -8.8405 \]
\[ g(5) = -0.31752 \]
\[ g(6) = -0.20000 \]
\[ g(7) = 0.81819 \times 10^{-11} \]
\[ g(8) = 0.19423 \times 10^{-9} \]
\[ g(9) = -20.680 \]
\[ g(10) = -0.48248 \]
\[ g(11) = -0.50000 \]

Monotonicity and dominance analysis:

For this point, based on first monotonicity rule, one of the following constraints with respect to \( x(2) \) may be active:
\[ g(2) \]
\[ g(5) \]
\[ g(9) \]

For this point, with respect to \( x(2) \) constraint \( g(9) \) may be dominant.

For this point, based on second monotonicity rule, one of the following constraints with respect to \( x(3) \) may be active:
\[ g(6) \]

Note: At any point, based on second monotonicity rule, constraints \( g(6) \) and \( g(9) \) MUST be simultaneously active.

Decisions taken for this iteration:

>>> AT THIS ITERATION, BASED ON FIRST MONOTONICITY RULE,
>>> WRT \( x(2) \) CONSTRAINT \( g(9) \) IS ACTIVE.
>>> AT THIS ITERATION, BASED ON SECOND MONOTONICITY RULE,
>>> WRT \( x(3) \) CONSTRAINT \( g(6) \) IS ACTIVE.
>>> FROM ITERATION(1): WRT \( x(1) \) CONSTRAINT \( g(7) \) IS ACTIVE.
>>> FROM ITERATION(1): WRT \( x(4) \) CONSTRAINT \( g(8) \) IS ACTIVE.
>>> FROM ITERATION(2): WRT \( x(5) \) CONSTRAINT \( g(1) \) IS ACTIVE.
***************ITERATION NO. (4)***************

Current point:

Values of design variables:
\[
\begin{align*}
x(1) &= 0.23799E-23 \\
x(2) &= 1.0777 \\
x(3) &= 1.0000 \\
x(4) &= 19.900 \\
x(5) &= 0.21163
\end{align*}
\]

Value of objective function:
\[
f = 1.0777
\]

Values of constraints:
\[
\begin{align*}
g(1) &= -0.23799E-23 \\
g(2) &= -1.0777 \\
g(3) &= -1.0000 \\
g(4) &= -19.900 \\
g(5) &= -0.21163 \\
g(6) &= 0.0 \\
g(7) &= -0.39790E-12 \\
g(8) &= -0.11369E-12 \\
g(9) &= -0.35527E-14 \\
g(10) &= -0.78837 \\
g(11) &= -0.92225
\end{align*}
\]

Decisions taken for this iteration:

\[
\begin{align*}
\text{>>> FROM ITERATION( 1):WRT x( 1) CONSTRAINT g( 7) IS ACTIVE.} \\
\text{>>> FROM ITERATION( 1):WRT x( 4) CONSTRAINT g( 8) IS ACTIVE.} \\
\text{>>> FROM ITERATION( 2):WRT x( 5) CONSTRAINT g( 1) IS ACTIVE.} \\
\text{>>> FROM ITERATION( 3):WRT x( 2) CONSTRAINT g( 9) IS ACTIVE.} \\
\text{>>> FROM ITERATION( 3):WRT x( 3) CONSTRAINT g( 6) IS ACTIVE.}
\end{align*}
\]

*TERMINATION CRITERION MET, KARUSH-KUHN-TUCKER CONDITIONS SATISFIED*