† *Department of Statistics and Curriculum in Operations Research and Systems Analysis, University of North Carolina at Chapel Hill. This work was supported by a grant from the Office of Naval Research, contract No. N-00014-67-0321-0003 (NR047-095).*

CONTINUOUSLY DIFFERENTIABLE EXACT PENALTY FUNCTIONS FOR NONLINEAR PROGRAMS WITH TOLERANCES

by

F. J. Gould†

Department of Statistics
University of North Carolina at Chapel Hill

Institute of Statistics Mimeo Series No. 744

April, 1971

CONTINUOUSLY DIFFERENTIABLE EXACT PENALTY FUNCTIONS
FOR
NONLINEAR PROGRAMS WITH TOLERANCES

by

F. J. Gould[†]

I. INTRODUCTION

The purpose of this paper is, first, to suggest an approach different from optimization theory for real-world problem solving and, second, to present a specific example of an implementation of the suggested approach. The implementation is in terms of the nonlinear programming problem

$$P_o: \quad \max f(x), \text{ subject to}$$
$$g_j(x) \le 0,\ j = 1,\cdots, m,$$

where it shall be assumed that all functions are real valued on $\mathbb{R}^n$ and all are elements of $C^k(\mathbb{R}^n)$. Relative to this problem the following question is of theoretic interest and of possible computational importance. Is it possible to explicitly exhibit a function $\phi(x)$, $\phi \epsilon C^k(\mathbb{R}^n)$, such that ϕ attains an *unconstrained* maximum on $\mathbb{R}^n$ and any global (local) unconstrained maximum of ϕ is a global (local) solution to P_o? To date it is

---

[†] *Department of Statistics and Curriculum in Operations Research and Systems Analysis, University of North Carolina at Chapel Hill.* *This work was supported by a grant from the Office of Naval Research, contract No. N-00014-67-0321-0003 (NR047-095).*

not known whether or not this question can be answered in the affirmative.[1/] (The major difficulty seems to be the stipulation $\phi \epsilon C^k(\mathbb{R}^n)$.) If it can be shown that the answer is yes, then it is plausible that such knowledge could make it easier to solve P_o. (This question, however, would need to be further studied, for the function ϕ could be sufficiently complicated that it would in fact be more difficult to find an unconstrained maximum of ϕ than to find a constrained maximum of f by currently known techniques.) The new approach to be exposited in this paper allows a certain amount of theoretic progress toward affirmatively answering the above posed question.

II. AN APPROXIMATION SEEKING APPROACH

In describing the rationale of this method, it is first noted that in many practical problems optimization is unwarranted. That is, the problem P_o is often a model builder's respresentation of some real-world process. For this reason alone (the human filter) most models are an inexact representation of reality. Furthermore, even given an exact model the data is often "fuzzy"[2/] and exact input data may not be obtainable. For example, the model builder may well be willing to replace the right hand side of P_o with any scalars b_j such that $-\varepsilon \leq b_j \leq \varepsilon$, where ε is some acceptable tolerance. (In solving problems with computer arithmetic one implicitly is always making such an assumption, where ε may be on the order of 10^{-70}.

[1/] *Such a function,* $\phi(x)$, *is sometimes termed an exact penalty function for* P_o.

[2/] *The ideas herein discussed do not appear to be fundamentally related to the "fuzzy set" theory as, for example, exposited by Bellman and Zahdeh in* [1], *though it may be of interest to further explore possible relations.*

We are here suggesting that in many situations a "larger" value of ε may also be acceptable to the practitioner.) Thus, in practice, we see an exact optimization theory being applied to inexact models, and all too frequently the theory is computationally hopeless for large scale systems. What seems to be a possibility here is that a more powerful (computationally more useful) theory can be obtained by explicitly and deliberately building a certain about of inexactness into the optimization.

In essence, then, we wish to consider those situations where there is no compelling reason to optimize P_o. In particular, we consider situations where the following ground rules are in effect: it is acceptable to replace the "conventional problem" P_o with a continuum of surrogate problems, in the sense that one is willing to settle for an optimal solution to *any one* of the infinitely many problems, without caring which one, and perhaps even with indifference as to knowing which one has been solved. (The contemplated replacement could be made in numerous ways; tolerances on right hand sides are only one such way. For example, in linear programming there might also be tolerances on objective function coefficients, or perhaps only on objective function coefficients.) The point of making such a replacement would be the hope that it might be easier to find an optimal solution to some unspecified member of the surrogate collection than to the one specific problem P_o.

A difficulty with this approach is that it is not particularly compatible with existing optimization theory. Most current algorithms are exact in the sense of being "optimality seeking" techniques as opposed to being inexact or "approximation seeking." For example, the simplex method is a very sharp optimality seeking method. If the procedure is stopped somewhere along the way, short of optimality, one does not even have an estimate of how good the current solution is. Some algorithms, though primarily optimality seeking, do have some approximation value. For example, in nonlinear programming, SUMT can be stopped along the line and a window is at hand within which must lie the difference between the current objective value and the true optimal value to P_o. Also, along the way, SUMT solves many subproblems, but in general the relation of these subproblems to the original problem will not be apocalyptic.[1/]

T. C. Hu in [7], defines a *nice algorithm* to be one that is tailor made for a problem. It capitalizes on special structure. For example, a nice algorithm for a network flow problem in integers might be one that works well for the integral case but not at all for noninteger data. Analogously, what one would seem to want in our context is an "inexact algorithm" for approximation seeking which is nice in the following sense. It guarantees and optimal solution to *some* one of the surrogate problems, and works exceptionally well for this purpose, but theoretically does not work at all as and optimality seeking procedure for P_o (or for any *single specified* optimization problem). One might then have reason to feel that he is making some interesting use of his special (conceptual) structure.

III. A SPECIFIC ILLUSTRATION

We use the problem P_o as a target for illustrating the above ideas. The illustration is only theoretic, for the implementation is incomplete.

[1/] *At each iteration, SUMT (as is true with other nonexact penalty function procedures) will solve a subproblem with a right hand side which generally does not lie within a tight tolerance of zero (some constraints may have much slack). Such a problem may be equivalent to a problem "closer" to* P_o *(i.e with tighter tolerances) but a knowledge of whether or not this is true cannot generally be attained.*

While, undoubtedly, improved solution techniques are to a large extent due to those who obtain and present computational comparisons, there is in many quarters an increasingly rigid and even indiscriminate resistance to the publication of any form of suggested solution procedure without numerical results (see, for example, a panel discussion at Keele as reported in [3]). It would seem likely that such a policy may have the undesirable effect of interdicting the timely transmission of possibly worthwhile theoretic results. In the present case, there is no claim as to the computational usefulness of the specific results presented below, although computational experiments on several large scale problems will be undertaken and reported in a future paper. The emphasis in this paper is on the replacement concept leading to the "inexact algorithm" and the possibility of there being interesting analogues in numerous other areas of operations research such as linear programming, integer programming, and network theory. The objective would be to discover new solution techniques with structures specifically designed to the mode of approximation discussed above in section II.

For the illustration of an approximation seeking algorithm for P_o, define the *multiplier function* [6] $\lambda: R \to R \cup \{+\infty\}$ as follows

$$\lambda(\xi) = \begin{cases} 0 & \xi \leq -\varepsilon \\ (\xi + \varepsilon)^3 & -\varepsilon < \xi \leq 0 \\ \dfrac{-3\varepsilon^4}{\xi - \varepsilon} - 2\varepsilon^3 & 0 < \xi < \varepsilon \\ \infty & \xi \geq \varepsilon \end{cases}$$

where ε is the user specified parameter discussed in section II. Now define the penalty function $\phi: R^n \to R \cup \{-\infty\}$ as $\phi(x) = f(x) - \sum_{j=1}^{m} \lambda(g_j(x))$.

Finally, define

$$S_\varepsilon = \{x: g_j(x) \le \varepsilon, j = 1,\cdots, m\}$$

$$\overset{o}{S}_\varepsilon = \{x: g_j(x) < \varepsilon, j = 1,\cdots, m\}.$$

Theorem: Assume S_ε is compact. Then

1. ϕ attains a global maximum and any global maximizer is in the set $\overset{o}{S}_\varepsilon$.

2. If the functions in P_o are all in $C^1(\mathbb{R}^n)$ $\left[\text{or } C^2(\mathbb{R}^n)\right]$ then ϕ is continuously (twice) differentiable on $\overset{o}{S}_\varepsilon$.

3. If $\hat{x}$ is any element of $\overset{o}{S}_\varepsilon$ which is a global (local) maximizer of $\phi(x)$ then $\hat{x}$ is a global (local) solution to

$$\begin{aligned}&\max f(x), \text{ subject to}\\ &g_i(x) \le g_i(\hat{x}),\ i\epsilon\{i: -\varepsilon < g_i(\hat{x}) < \varepsilon\} = J_1\\ &g_i(x) \le -\varepsilon,\ i\epsilon\{i: g_i(\hat{x}) \le -\varepsilon\} = J_2.\end{aligned}$$

Proof: The first conclusion is intuitively clear and is a standard result on barrier functions (penalty functions which are infinite on the constraint set boundary). The proof is a straightforward analogue of one appearing both in [4] and in [5]. The second conclusion is easily verified from the definition of $\lambda(\cdot)$ and $\phi(\cdot)$. To prove the third conclusion, suppose $\hat{x}$ is a global maximizer of $\phi(x)$. Then, noting that $J_1 \cup J_2 = \{i: 1 \le i \le m\}$,

$$\phi(\hat{x}) = f(\hat{x}) - \sum_{J_1} \lambda\ (g_i(\hat{x})) - \sum_{J_2} \lambda\ (g_i(\hat{x})) \ge f(x) - \sum_{J_1} \lambda\ (g_i(x)) - \sum_{J_2} \lambda\ (g_i(x))$$

for all x in $\mathbb{R}^n$. Hence, since $\lambda(g_i(\hat{x})) = \lambda(-\varepsilon) = 0$ for $i\epsilon J_2$, $f(\hat{x}) - \sum_{J_1} \lambda\ (g_i(\hat{x})) - \sum_{J_2} \lambda\ (-\varepsilon) \ge f(x) - \sum_{J_1} \lambda\ (g_i(x)) - \sum_{J_2} \lambda\ (g_i(x))$ for all x in $\mathbb{R}^n$. This implies

$$f(\hat{x}) \ge f(x) + \sum_{J_1} \left[\lambda\ (g_i(\hat{x})) - \lambda\ (g_i(x))\right] + \sum_{J_2} \left[\lambda\ (-\varepsilon) - \lambda\ (g_i(x))\right] \quad \text{for all } x$$

in $\mathbb{R}^n$. Finally, this implies $f(\hat{x}) \leq f(x)$ whenever $g_i(x) \leq g_i(\hat{x})$, $i \in J_1$, $g_i(x) \leq -\varepsilon$, $i \in J_2$. The proof for the local result is analogous.∎

It can also be shown that the above penalty function $\phi(x)$ can be replaced with one which is k-times continuously differentiable, provided the problem functions in P_o are k-times continuously differentiable, and results 1 and 3 still hold. Furthermore, simple modifications can be made to allow only left hand tolerances about zero (i.e. to force the approximate solution $\hat{x}$ to be feasible in P_o) and upper bounds for the optimal objective value in P_o can be obtained. It is to be noted that the penalty function $\phi(x)$ is essentially parameter free, the ε being a prespecified number. Parameters can be added by adding them in the definition of the multiplier function $\lambda(\cdot)$ in various ways. The motivation for the form of the penalty function discussed in this section is derived from the study of supports to the f_{sup} function, as discussed in [2] and in [6].

The hope here of course is that the function $\phi(x)$ is "well enough conditioned" and sufficiently simple to evaluate that the single unconstrained maximization of ϕ, for large scale problems, will be easier than other known procedures for solving P_o. This, however, remains unverified.

REFERENCES

[1] R. E. Bellman and L. A. Zahdeh, Decision making in a fuzzy environment, *Man. Sci.*, 17 (1970), pp B-141—B-164.

[2] J. P. Evans, F. J. Gould, and J. W. Tolle, "Exact penalty functions and duality in nonlinear programming," submitted to this journal.

[3] *Optimization Theory*, R. Fletcher, ed., pp 345-347, "To what extent should an algorithm be tested before publication," John Wiley, New York, 1969.

[4] A. V. Fiacco and G. P. McCormick, The sequential unconstrained minimization technique for nonlinear programming, a primal-dual method, *Man. Sci.*, 10 (1964), pp 360-366.

[5] F. J. Gould, A class of inside-out algorithms for general programs, *Man. Sci.*, 16 (1970), pp 35C-356.

[6] __________, Extensions of Lagrange multipliers in nonlinear programming, *SIAM* J. Appl. Math. 17 (1969) pp 1280-1297.

[7] T. C. Hu, The development of network flow and related areas in programming, *MRC technical summary report number 1096*, August 1970, Mathematics Research Center, University of Wisconsin, Madison, Wisconsin 53706.