연구논문집 제 7 권 제 1 호 p. 9~17. 1976

Fast Computing Algorithm of Constrained Minimization

Park, W. S. Cin, Q Young
Dept. of Electrical Engineering

〈**Abstract**〉

In finding out the minimun value of an objective function with constraints, the pattern move is strongly applicable to ill-behavior functions.

In this paper a fast computing algorithm for minimization of an objective function is proposed using pattern move. The basic idea comes from reducing penalty term weighting factor according to one cycle of pattern move.

제약 조건하의 최소화 문제에 있어서의 빠른 계산 알고리즘에 대하여

박 원 식 · 신 규 영
전기공학과

〈요 약〉

복직 함수의 1차도함수를 구하는 과정을 피하기 위하여 패턴무브(Pattern move)를 기본기술로 한 계산방법을 제시하였다. 기본적 구조는 SUMT와 같으나 Pattern move의 시행과정중에 Penalty function의 부게치(weighting value)를 삽입하여 계산과정이 격어지는 이점을 갖는 알고리즘을 개발하였다.

I. Introduction.

For many years since 1960's, the control Engineers and applied mathematicians have endeavored to find an efficient method to minimize/or maximize a given objective function under several constraints.

But these sort of problems were begun earlier in 18th century as in post-office parcel problems, in which it is required to find the largest volume given some constraints concerning to length, width and depth. Eearliest workers in this field were:[2] George Danzig(1947 Simplex method), T. Koopmans[3], H. W. Kuhn[4] and A. W. Tucker(1951. Nonlinear programing).

In 1954. A. Charnes and C. Lemke[5] published an approximation method of treating problems which have an objective, the minimization of a separable function, subject to linear constraints.

In mid-fifties, Quadratic programming was studied by E. Barankin and R. Dorfman, (1955) E. M. L Beale(1955), M. Frank and P. Wolfe(1956), H. Markowitz(1956), C. Hildreth(1957), H. Houthakker(1957), and P. Wolfe(1959)

In 1960 H. H. Rosenbrock[6] contrived an algorithm to find the greatest or least value of a function with the aid of first derivatives.

In 1964 A. V. Fiacco and G. McCormick[7][8] Proposed an algorithm well known as SUMT and extended their idea when the constraints are a mixture of inequalities and equalities.

Afterwards Powell, Fletcher and Zanwill were the main contributors in this field.

M. J. D Powell, while, has extended the idea of pattern move in 1964 which leads to a method of conjugate directions.

As for the penalty functions, Zangwill[9] has rendered a complete theoretical basis which shall be reviewed briefly in this paper.

The basic techniques used in finding the maximum or minimum of a function are univariate method, pattern move, steepest descent method etc. But when the first derivative of an objective function is not available, it seems the pattern move is one of the strongest tools for our purpose.

So in this paper, we shall treat the pattern move as a basic tool and review Zangwill's theory briefly and present an Algorithm which will shorten the time comsumed in computing.

II. The Pattern Move.[10]

The pattern move is a useful technique for improving the convergence. As we see in the

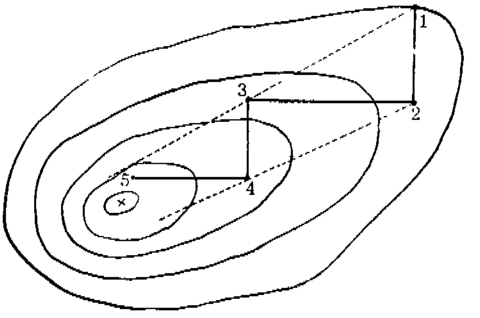


Fig. 1

fig 1, the alternate points in the iteration(1, 3; 2, 4 ; 3, 5 ; etc) define lines which lie in the general direction of the minimum. This Property is so strong in 2 dimensions that if the function being minimized is quadratic, all such lines pass through the minimum.

In others words, they pass through the common center of the family of ellipses that are the contours of the quadratic. Unfortunately, this property does not carry through directly to higher dimensions, but the idea can still

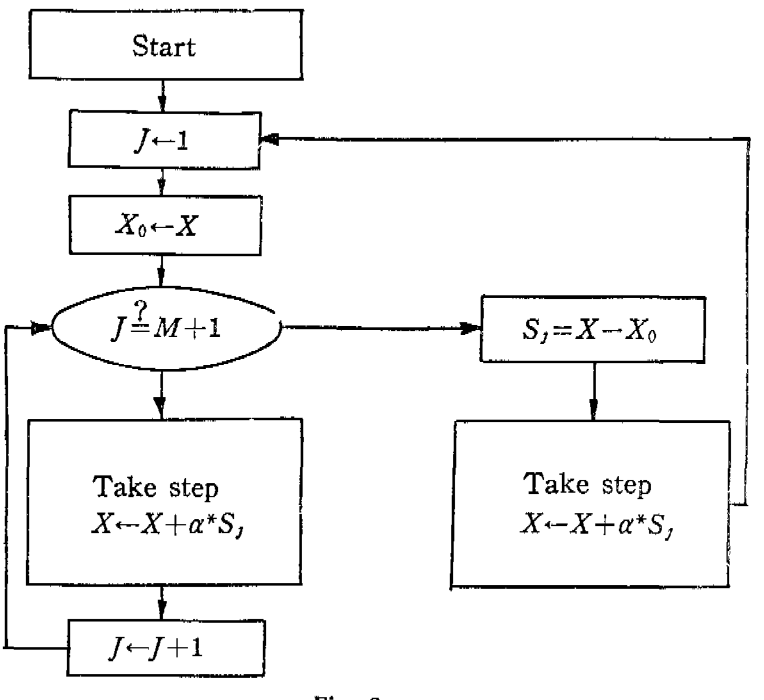


Fig. 2

greatly speed convergence in these problems.

A general method involving pattern moves is to take m univariate steps (often $m=n$, if there are n Variables in the problem) and take a move in the direction defined by

$$S_q = X_q - X_{q-m} \qquad (2.1)$$

The algorithm for $m=n$ is defined by the bloc diagram in Fig2. Blocks A and B of Fig2 imply the whole sequence of steps required to determine α^*. sequence shall be treated later.

Ⅲ. Minimizing Steps : Quadratic interpolation.

The procedure determining the step length α^* is quite simple if we choose quadratic interpolation.

Although the cubic interpolation is sometimes more accurate, we will consider only quadratic interpolation Consider any vector S_q and the move prescription

$$X = X_q + \alpha S_q \qquad (3.1)$$

where, if α is considered a variable, the locus of X for a range of values of α is a straight line. Substituting this formally into $F(X)$, we obtain

$$F(X) = F(X_q+\alpha S_q)=F(\alpha) \qquad (3.2)$$

Since the objective function F can be considered a function of α alone (X_q and S_q are considered fixed). We seek here the value of α which minimizes $F(\alpha)$. Note that this value, denoted by α^*, does not produce the global minimum of F unless the line $X=X_q+\alpha S_q$ contains the global minimum point.

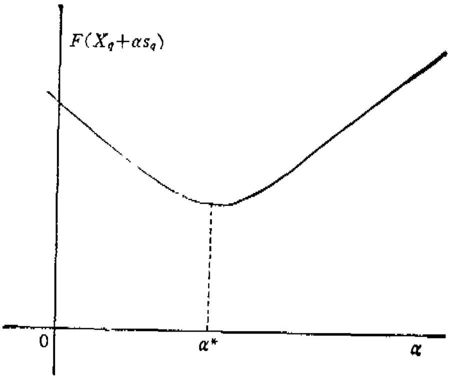


Fig. 3

With this concept, the problem of minimizing $F(X)$ can be reduced to a succession of 1-dimensional minimization problem regardless of the dimensionality of X. If F is a simple explicit function of X, eq. (3.1) can be written directly in terms of the variable α, the quantity α^* can be computed exactly. However, in practice we rarely have the good fortune to carry out the operation and must usually resort to numerical means for finding α^*

Consider approximating the function $F(\alpha)$ by a function $H(\alpha)$ which has an easily determined minimum point.

The simplest one-variable function possessing a minimum is the quadratic

$$H(\alpha)=a+b\alpha+c\alpha^2 \qquad (3.3)$$

the minimum of which occurs where

$$\frac{dH}{d\alpha}=b+2c\alpha=0 \qquad (3.4)$$

$$\text{or } \alpha^*=-\frac{b}{2c} \qquad (3.5)$$

The constants b and c for the approximating quadratic (a is not needed) can be determined by sampling the function at three different α values α_1, α_2 and α_3 and solving the equations.

$$\begin{aligned} f_1&=a+b\alpha_1+c\alpha_1^2\\ f_2&=a+b\alpha_2+c\alpha_2^2\\ f_3&=a+b\alpha_3+c\alpha_3^2 \end{aligned} \qquad (3.6)$$

Where f_1 denotes the value of $F(\alpha_1)$, etc. If we use 0, t and $2t$ for α_1, α_2, and α_3, where t is a preselected trial step, eq. s(3.6) are particularly easy to solve and we can save one function evaluation. With this choice, eq. s(3.6) become

$$\begin{aligned} f_1&=a\\ f_2&=a+bt+ct^2\\ f_3&=a+2bt+4ct^2 \end{aligned} \qquad (3.7)$$

From these we obtain

$$a=f_1$$

$$b=\frac{4f_2-3f_1-f_3}{2t} \qquad (3.8)$$

$$c=\frac{f_3+f_1-2f_2}{2t^2}$$

and $$\alpha^*\simeq\frac{4f_2-3f_1-f_3}{4f_2-2f_3-2f_1}t \qquad (3.9)$$

For α^* to correspond to a minimum, it must satisfy

$$\frac{\alpha^2 H}{d\alpha^2}\bigg|_{\alpha=\alpha^*}>0 \qquad (3.10)$$

The case in which H is quadratic requires $C>0$, or

$$f_3+f_1>2f_2 \qquad (3.10)$$

This means that the value of f_2 must be below the line connecting f_1 and f_3. (see fig 4)

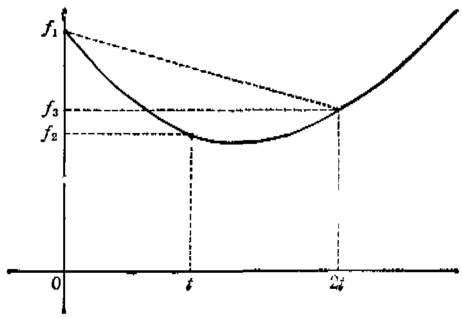
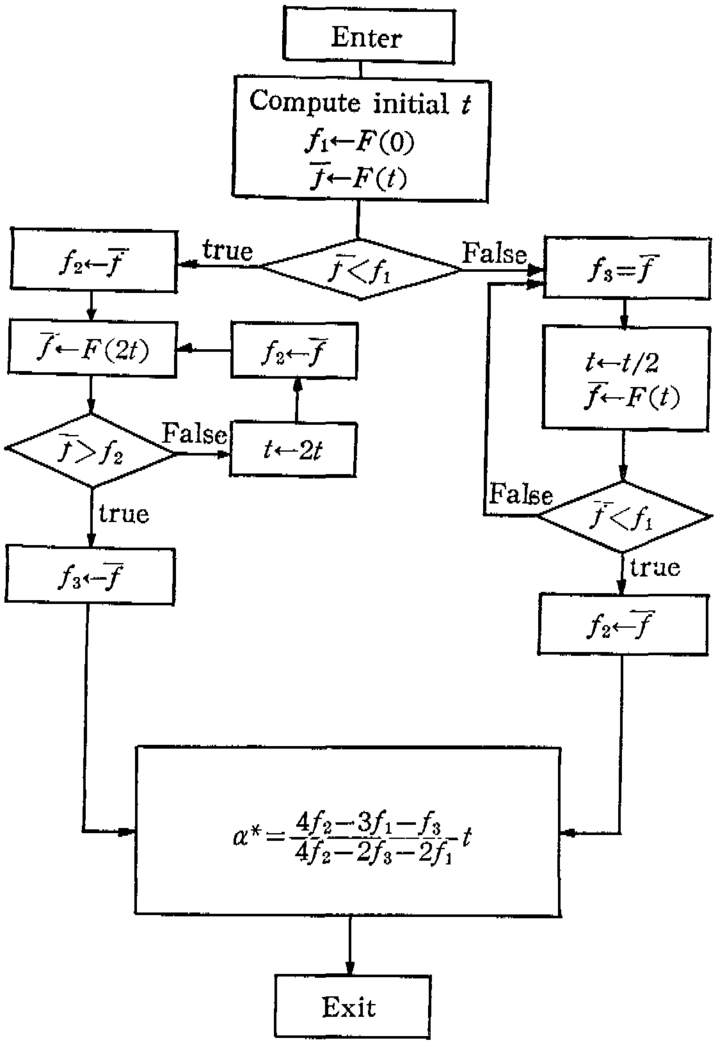


Fig. 4



A scheme for ensuring that eq. (3.11) is satisfied and that the minimum lies in the interval $0 < \alpha < 2t$ is as follows:

1. Choose an initial value for t based on previous iterations or other information regarding a reasonable value for the step length. Ideally t would be on the order of α^*

2. Compute $F(t)$

3. If $F(t) > F(0) = f_1$, then set $f_3 = F(t)$, Cut t in half, and repeat step 2 : Otherwise set $f_2 = F(t)$, double t, and repeat step 2.

4. When a value t has been obtained such that $f_2 < f_1$ and $f_3 > f_2$, compute α^* according to e_q (3.9)

The logic for the quadratic interpolation algorithm described above is given in the flow diagram shown in Fig 5

Ⅳ. Brief Review of Zangwill's theory.

So far we have discussed on pattern move and quadratic interpolation, which shall be basic tools for our nonlinear programming, and now we shall examine the fundamental theory for nonlinear programming. The problem is to

maximize $f(X)$ (4.1)

subject to the constraints

$g_i(X) = 0 \quad i = 1, \cdots, m'$ (4.2)

$g_i(X) \geq 0 \quad i = m'+1, \cdots, m$ (4.3)

where $F(\cdot)$ and $g_i(\cdot)$ are real valued functions defined on E^n, Euclidean n space, and $X = (X_1, \cdots, X_n)$ is a point in E^n. A point x^* which solve (4.1), (4.2) and (4.3) is called an optimal point. Let

$S = \{X \mid g_i(X) = 0,\ i = 1, \cdots, m',\ g_i(X) \geqq 0,$
$i = m'+1, \cdots m\}.$

S is called the feasible set or feasible region. Define the real valued function

$P(X, r) = f(x) + rL[g(x)]$ (4.4)

where $r \geq 0$ $L(0)$ is a loss function which will be defined below.

Put $[y_i]^+ = |y_i| \quad i = 1, \ m'$ (4.5)

$[y_i]^+ = -\min[y_i, 0] \quad i = m'+1, \ m$

$L(y) = L^+([y]^+) \begin{cases} = 0 \text{ if } [y^+] = 0 \\ < 0 \text{ if } [y]^+ \neq 0 \end{cases}$ (4.6)

where y is a m component vector.

Note that for $r > 0$, $P(X, r) = f(x)$ if $X \in S$, and $P(X, r) < f(x)$ for $X \notin S$ (4.7)

The basic steps in the algorithm can easily be stated.

Consider an increasing **non-negative sequence of scalars $\{r^k\}_{k=1}^{\infty}$ where $\lim_{k\to\infty} r^k \longrightarrow +\infty$. For each k let

$P(X^k, r^k) = \max_x P(X, r^k)$ (4.8)

where X^k is the X which gives the unconstrained maximum of $P(X, r^k)$. Under reasonable assumptions, the maximizing X^k will exist and the sequence $\{X^k\}_{k=1}^{\infty}$ will have a convergent subsequence.

Let Z be the limit of such a convergent sequence. The pair Z, $f(x)$ will be an optimal selection to the nonlinear programing and furthermore,

$\lim_{k\to\infty} f(X^k) = \lim_{k\to\infty} P(X^k, r^k) = f(x^*)$ (4.9)

(See references 7 and a for rigorous Proofs)

Ⅴ. The Algorithm

The penalty function can be classified into two categories; exterior penalty function and interior penalty function. But the latter has a number of computational and egineering advantages. So we will consider the interior penalty function mainly.

The most commonly used function of this sort is

$$\phi(x, r) = F(x) + r \sum_{j=1}^{m} \frac{1}{g_j(x)} \qquad (5.1)$$

Where F is to be minimized over all X satisfying $g_j(X) \geq 0$

$j = 1, \cdots m.$

* His original paper is to maximize a function, but this can be easily modified to a problem to minimize a function. The details will be omitted here.

** In minimizing problems r^k sequence is a decreasing non negative one where $\lim_{k\to\infty} r^k \longrightarrow 0$.

r will take decreasing non-negative sequence of real number the penalty term $r\sum_{j=1}^{m} \frac{1}{g_j(X)}$ is added to the original objective function $F(X)$ to prohibit the sequence of X_i's going outside the feasible region. The effect of penalty function is shown in fig 6, for one dimensional case.

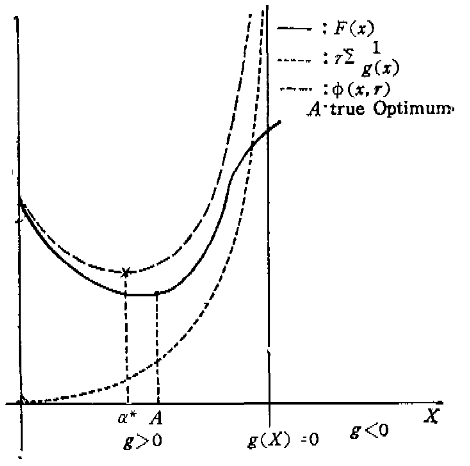


Fig. 6

The Algorithm for this case is

1. Given a starting point X_0 satisfying all $g_i(X)>0$ and an initial value for r, minimize ϕ to obtain X_M.
2. Check for convergence of X_M to the optimum.
3. If the convergence criterion is not satisfied, reduce r by $r\leftarrow rc$, where $c<1$,

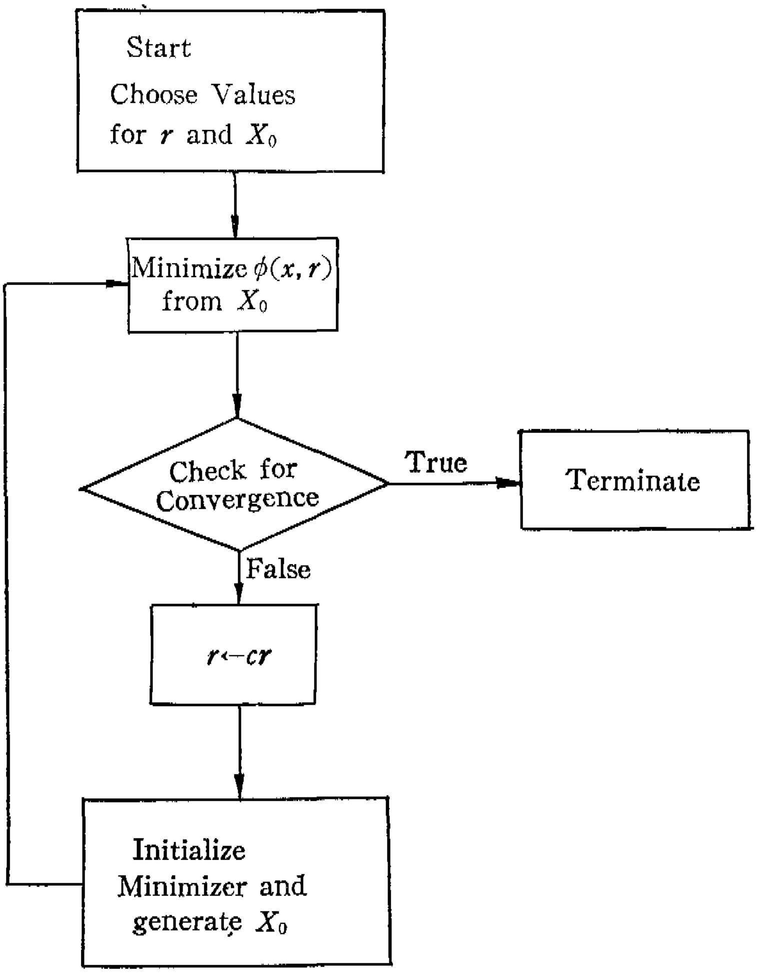


Fig. 7

4. Compute a new starting point usually the last optimized point of former sequence for the minimization, initialize the minimization algorithm, and repeat from step 1.

The logic diagram for this algorithm is shown in fig7

But, if we follow the logic depicted in fig7 for n-variable function, for each r say r_1, it is reasonable that we must take n step, that is, n times of computing, and for r_2, n step etc. So for a s Sequence of r, we compute ns times, and this would be a large number if the convergence criterion is not satisfied.

Hence there arises a motive for modifying this method. In pattern move approach, each n step forms a cycle and this cycle continues until a reasonable convergence criterion is satisfied.

If we reduce the value of r at each cycle, the computing step will be greatly reduced.

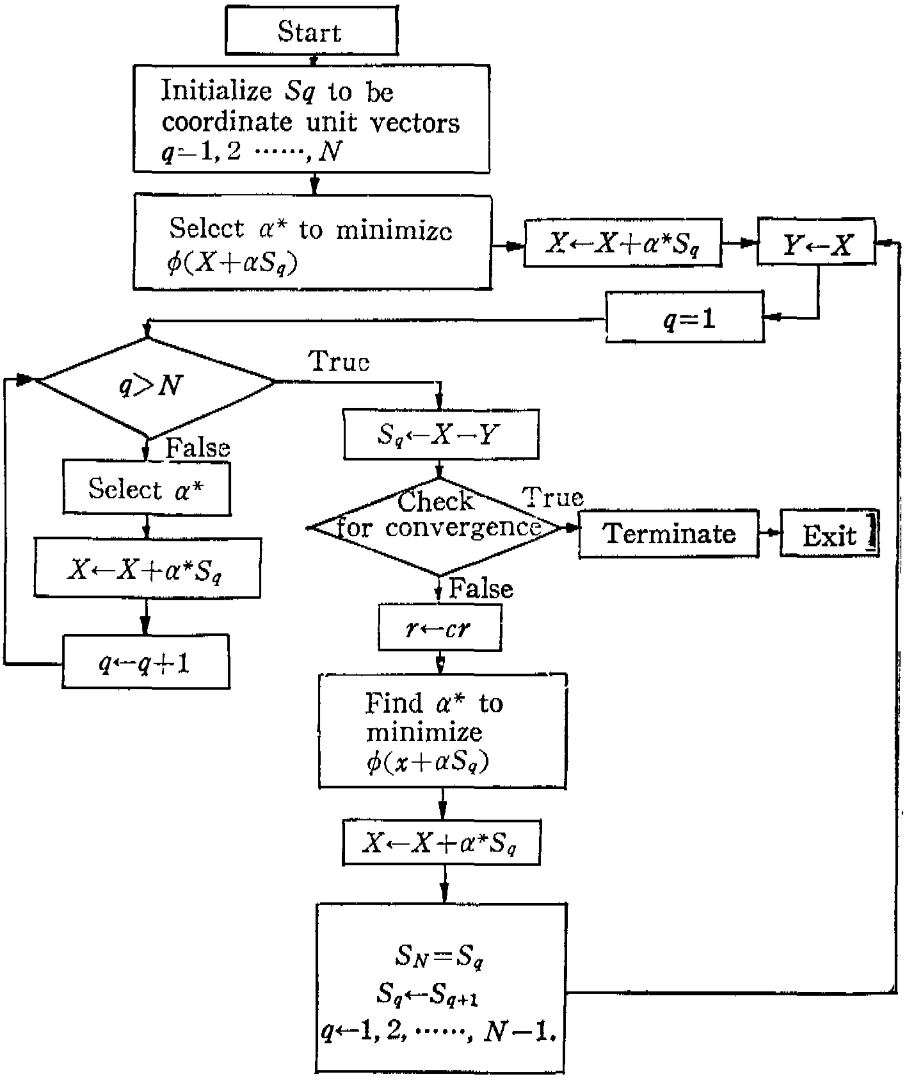


Fig. 8

The algorithm based upon this idea is

1. Given a starting point X_0 in feasible region, and an initial value of r, Minimize $\phi(X, r)$ according to pattern move just one cycle.

2. reduce r and begin second cycle

3. check for convergence and if satisfactory terminate

4. if step 3 is not satisfactory go to step 2. The logic diagram for this algorithm is shown in fig 8

Ⅵ. Other Problems

1. Convergence criteria

One of the perennial problems in minimization is the termination of the process.

A simple criterion is to compute the relative difference

$$\delta \equiv \frac{|F_{min}(r_{i-1}) - F_{min}(r_i)|}{|F_{min}(r_i)|}$$

and stop when this value drops below a prescribed value.

An equally appealing criterion for convergence is to compute

$$\Delta = X_M(r_{i-1}) - X_M(r_i)$$

where $X_M(r_i)$ means minimal point of $\phi(X, r_i)$ under fixed r_i, and then compare the norm of Δ with a small number ε.

or $$|\Delta| = \left(\sum_{j=1}^{n} \Delta_j^2\right)^{\frac{1}{2}} < \varepsilon$$

Where Δ_j is the j-th component of Δ.

Another method is to compute the penalty term at the minimum point

$$r_i \sum_{j=1}^{N} \frac{1}{g_j(X)}\bigg|_{X=X_M(r_i)} < \varepsilon$$

But this method must be carefully programmed lest r_i should fall down too rapidly.

2. Starting Point

In engineering problems we can usually find the feasible point, but when the problem considered is complicated, it is not easy to find a feasible point X_0

Suppose we selected a point X_0 which satisfies $g_j(X_0) \geq 0$, for $j = 1, 2, \cdots p$, and $g_j(X_0) < 0$ for $j = p+1, \cdots, m$.

Take $k (k \in \{p+1, \cdots, m\})$ for which the $g_k(X_0)$ is severely violated, and we temporarily set the $g_k(X)$ as the objective function to be maximized. And so the problem is that

Find $g_k(X) \longrightarrow \max.$

$$g_j(X) \geq 0 \quad j = 1, 2, \cdots, p$$

$$g_j(X) - g_j(X_0) \geq 0 \quad j = p+1, \cdots, m.$$

When this process is finished, at least one constraint is satisfied besides the formerly established ones. And this process is continued until all the constraints are satisfied.

3. Initial value of r and reduction ratio.

Although the matter of selecting an initial value r has been discussed in the literature[10], the task is still mainly an art, and the reduction ratio c can assume a value between 0 and 1.

Another approach for reducing r is to take exponentially decreasing r_i, that is

$$r_i = r_0 e^{-i\gamma}$$

where γ can be either $\gamma > 1$, or $\gamma < 1$, according to the nature of the problem. If γ is large, r_i will converge to zero rapidly which will nullify the effect of penalty term but speeds up the process inside the feasible region. If γ is small, the penalty term will remain as an unignorable part and the process will take a much longer time.

Ⅶ. Conclusions

So far we have discussed the pattern move, theoretical basis for SUMT and finally our algorithm. If, for a fixed r, the number of cycles of the pattern move is s and we have to reduce γ k-times, the steps of X_i will take ks times. But in our algorithm, by choosing γ appropriately, the steps of X_i can be reduced to s steps only.

This characteristic will be more powerfully applied for the functions with ill-behavior and many variables as in computation of E. L. D.

VIII. Aknowlegement

This research was carried out under the financial support of ministry of Education, R.O.K. The authors also thank for Prof. Y. M. Park, Dept. of Electrical Engineering, S.N.U. for his valuable advices.

And finally, we owe Miss H. Jou for her handwriting of the manuscripts.

References

1. FOX, Richard L., *Optimization Methods for Engineering Design.* ADDISON-WESLEY, 19
2. HADLEY, G., *Nonlinear and Dynamic Programming,* 2nd ed. ADDISON-WESLEY, 1972
3. KOOPMANS, T. (ed.), *Activity Analysis of Production and Allocation.* New York, WILEY, 1951.
4. KUHN, H.W., and A.W. Tucker, *Proceedings Second Berkeley Symposium On Methmatical Studies and Probability,* pp. 481—492. (1951)
5. CHARNES, R., and C. Lemke *Naval Research Logistics Quaterly,* pp. 301—312. (1954)
6. Rosenbrock, H.H., *Computer J.*, 3(3), 1960.
7. FIACCO, A. V., and Garth P. McCormick, *Management Science,* Vol. 10, No 2, Jan., 1964.
8. FIACCO, Anthomy V., and Garth P. Mc Cormick, *Management Science,* Vol. 12, No. 11, July, 1966.
9. ZANGWILL, Willard I., *Management Science,* Vol. 13, No. 5, Jan., 1967.
10. FIACCO, A.V. and Garth P. McCormick, *RESEARCH ANALYSIS CORPORATION, Bethesda, Md. Tech. Paper,* RAC-TP-96, 1963.