〈硏究論文〉

Journal of the Korean Institute of Industrial Engineers
Vol. 14, No. 2, Dec. 1988

Function Optimization Using Quadratically Convergent Algorithms With One Dimensional Search Schemes

Do Il Kim*

Abstract

In this paper, a unified method to consturct a quadratically convergent algorithm with one dimensional search schemes is described. With this method, a generalized algorithm is derived. As it's particular cases, three quadratically convergent algorithms are performed. They are the rank-one algorithm (Algorithm I), projection algorithm (Algorithm II) and the Fletcher-Reeves algorithm (Algorithm III). As one-dimensional search schemes, the golden-ratio method and dichotomous search are used. Additionally, their computer programming is developed for actual application. The use of this program is provided with the explanation of how to use it, the illustrative examples that are both quadratic and non-quadratic problems and their output. Finally, from the computer output, each algorithm was analyzed from the standpoint of efficiency for performance.

1. INTRODUCTION

This paper is concerned with a method for setting up algorithms for the function optimization of several variables. This algorithm requires the following properties : (a) The algorithms use one dimensional search only ; (b) For quadratic function the algorithms can converge quadratically and the required number of iterations at most must be equal to the number of variables ; (c) The algorithms use only the problem function and it's gradient ; (d) The algorithms employ information of present stage and the just previous stage.

The property (a) avoids a multi-dimensional search. The property (b) is very important because even a non-quadratic function works approximately quadratically in the neighborhood of the optimum point. So, rapid convergence can be assured in the final stage of computation. The property (c) does not need second-order derivatives for computational convenience. The property (d) is required to reduce the computer memory size. This property is also important when we deal with a function with a large number of variables. There are several algorithms that have all above properties. They are the conjugate-gradient algorithm and variable metric algorithm, especially. Many authors have developed these algorithms but they can't give a clear generalized algorithm that can be developed to fulfill all above properties.

* Chung Ang University

A unified method was given so that a general algorithm could have properties (a) – (d) by Huang [4]. As one of it's particular cases, many algorithms could be developed. They are the conjugate-gradient method, variable metric algorithm, quasi-Newton method, projection method and simplified algorithm. In this paper (I) the rank-one algorithm (quasi Newton method), (II) the projection method, and (III) the simplified method (Fletcher-Reeves algorithm) will be considered specially for their particular properties and efficiency in performance. The quasi-Newton method has the property that the H-matrix tends to the inverse of the Hessian of the function while the projection method can be thought for it's property that H-matrix tends to zero after certain number of iterations. The Fletcher-Reeves method is developed for it's simplicity for setting up the H-matrix. When one-dimensional search is performed, these three algorithms work identically along every search. If the function has a unique optimum point along every search direction, the behavior of algorithms is independent of the search method employed. However, if the function has multiple minima along a search direction, different search methods may bring different minimal points ; from these opints, one should investigate (a) their dependence on search methods and (b) their dependence on search stopping conditions.

In getting a step-size, two direct one-dimensional search techniques are employed for their power and simplicity. They are the dichotomous search technique and the golden-ratio search. The dichotomous search technuque is highly efficient from the standpoint of reducing the interval of uncerainty with small number of search points. In the goldenratio search, the location of the first two search points is independent of the total number of search points employed.

Therefore the golden-ratio search technique does not minimize the interval of uncertainty after a prespecified number of search points, so that this method is slightly less efficient. But under the condition that the number of search points required to get a satisfactory answer is not known in advance, the golden-ratio search technique offers decided computational advantage over the dichotomous search technique.

In this paper, these methods are used iteratively until a given precise stopping condition is satisfied. However, the optimal step size could be calculated, conductig one dimensional search. But when the optimal step size is applied to our algorithms, sometimes it can be found that too many number of iterations are needed to reach to optimal point ; the optimal step size from one dimensional search does not guarantee best computational efficiency. So that, under certain condition, the relaxed step size would rather be employed for reaching to the optimal point faster than the optimal step size. The performance of the relaxed step size is also shown in this paper. That relaxed step size method can be either over-relaxation or under-relaxation. Which relaxed step-size is more efficient depends on the characteristic of the problem functions. It is observed that how efficient the under-relaxed step size is in the example problems.

2. ALGORITHM WITH OUADRATIC CONVERGENCE AND IT'S GENERAL FORM

2.1 Nonorthoganality Condition

In order to set up algorithm that converge quadratically to the optimal point, we consider a quadratic function f(X) given by

$$f(X) = a + bX^T + (1/2)X^T cx \tag{1}$$

and the gradient $g(X)$ is given by

$$g(X) = b + cX \tag{2}$$

Consider current point X_i and next point X_{i+1} with step $\alpha_i.P_1$. The recurrence formula will be given by

$$X_{i+1} = X_i + \Delta X_i \tag{3}$$

when

$$\Delta X_i = -\alpha_i P_i \text{ (for minimitation problem)} \tag{4}$$

Where α_i is step size and Pi is step direction.
The function at the next point will be

$$f(X_{i+1}) = f(X_i - \alpha_i P_i) \tag{5}$$

In order to get an algorithm having property (a), we should know P_i, so $f(X_{i+1})$ becomes function of α_i only. At it's minimum point

$$df(X_i - \alpha_i P_i)/d\alpha_i = 0$$

that is so when

$$g_{i+1}^T P_1 = 0 \tag{6}$$

Where g_{i+1} is the gradient $g(X_{i+1})$
from the equation (2)

$$g_{i+1} = g_i + c\,\Delta X_i \tag{7}$$

From the equations (4), (6), (7), we obtain the following expression for step size α_i.

$$\alpha_i = g_i^T P_i / P_i^T C P_i \tag{8}$$

from the equation (1)

$$f(X_{i+1}) = f(X_i) + g_i^T \Delta X_i + (1/2)\ \Delta X_i^T C\ \Delta X_i \tag{9}$$

from the equations (8), (9), (4)

$$f(X_{i+1}) - f(X_i) = (g_i^T P_i)^2 / 2P_i^T CP_i \tag{10}$$

Finally we could get nonorthogonality condition as follows

$$f(X_{i+1}) < f(X_i) \text{ (for minimization)}$$

as iong as

$$g_i^T P_i \neq 0 \tag{11}$$

If any vector satisfying above condition can be used as the search direction, Equations (3), (4), (6) bring a complete algorithm having the descent property for the function f(X). If we choose P_i with g_i, then the algorithm is called the gradient method.

2.2 Conjugacy Condition

At previous section, we could know that gradient g_k at point X_k and g_j at a previous point X_j are related by

$$g_k - g_j = - \sum_{i=j}^{k-1} \alpha_i CP_i \ (k-1 \geq j \geq 0) \tag{12}$$

If we multiply either sides by P_i, Equation (12) will be

$$g_k^T P_j = - \sum_{i=j}^{k-1} P_i^T CP_j \ (k-2 \geq j \geq 0) \tag{13}$$

Here, if following conjugacy condition is satisfied,

$$P_i^T CP_j = 0 \ (k-1 \geq i \geq j \geq 0) \tag{14}$$

the equation (13) will be

$$g_k^T P_j = 0 \ (k-2 > j > 0) \tag{15}$$

and from the equations (10) and (15)

$$g_{\cdot}^T P_j = 0 \ (n-1 \geq j \geq 0) \tag{16}$$

From the linear algebra, we can generate a sequence of n non zero direction $P_0, P_1, \cdots, P_{n-1}$ in this way and these directions are linear independent. So, the only vector g_n satisfying Equation (16) is the null vector. Therefore we can get optimum point in n iterations, at most. That is the quadratic convergent property.

2.3 Construction of Algorithm

There are many ways of choosing such a sequence of search directions, so we consider algorithm satisfying the additional property (c) and (d) among many ways satisfying properties (a) and (b).

2.3.1 Search Direction

If we express P_i in the form

$$P_i = H_i^T g_i \tag{17}$$

from the Equations (14) and (15)

$$H_i C P_j = \delta P_j \ (i-1 \geq j \geq 0) \tag{18}$$

where

δ is an arbitrary constant.

We could know if matrix H_i has the property (18), the conjugacy condition will be satisfied.

2.3.2 The H-matrix

At the previous iteration, Equation (18) will be

$$H_{i-1} CP_j = \delta P_j \text{ for } i-2 \geq j \geq 0 \tag{19}$$

Subtracting Equation (19), From equation (18) we could get

$$(H_i - H_{i-1}) CP_j = 0 \text{ for } i-2 \geq j \geq 0 \tag{20}$$

If the matrix Hi is updated as $H_i = H_{i-1} + \Delta H_{i-1}$ (21)

then, Equation (20) will be

$$\Delta H_{i-1} CP_j = 0 \text{ for } i-2 \geq j \geq 0 \tag{22}$$

Here, if the ΔH_{i-1} has the further property that

$$\Delta H_{i-1} CP_{i-1} = \delta P_{i-1} - H_{i-1} CP_{i-1} \tag{23}$$

then multiplying Equation (22) by α_j and Equation (23) by α_{i-1} and in the light of Equation (4) and Equation (7), we could get

$$\Delta H_{i-1} \Delta g_j = 0 \text{ for } i-2 \geq j \geq 0 \tag{24}$$

and

$$\Delta H_{i-1} \Delta g_{i-1} = \delta \Delta X_{i-1} - H_{i-1} \Delta g_{i-1} \tag{25}$$

where

$$\Delta g_{i-1} = g_i - g_{i-1}$$

Now in order to satisfy Equation (25), we can assume

$$H_{i-1} = \frac{\Delta X_{i-1} y_{i-1}^T}{y_{i-1}^T \Delta g_{i-1}} - \frac{H_{i-1} \Delta g_{i-1} Z_{i-1}^T}{Z_{i-1}^T \Delta g_{i-1}} \tag{26}$$

then if the following conditions are satisfied,

$$y_{i-1}^T \Delta g_j = 0, \ Z_{i-1}^T \Delta g_i = 0 \ (i-2 \geq j \geq 0) \tag{27}$$

ΔH_{i-1} satisfies Equation (26) and Equation (21) will give new matrix H_i. Now, because of the desired property (d), we observe that Equation (14) for the previous iteration is

$$P_{i-1}^T CP_j = 0 \ (i-2 \geq j \geq 0) \tag{28}$$

using Equations (4), (7), then Equation (28) can be

$$\Delta X_{i-1}^T \Delta g_j = 0 \ (i-2 \geq j \geq 0) \tag{29}$$

Next, from Equation (15), at the previous iteration

$$\Delta g_{i-1}^T P_j = 0 \ (i-2 \geq j \geq 0) \tag{30}$$

Because of Equation (19), Equation (30) implies that

$$\Delta g_{i-1} H_{i-1} CP_j = 0 \ (i-2 \geq j \geq 0) \tag{31}$$

using Equations (4), (7), Equation (31) can be

$$\Delta g_{i-1}^T H_{i-1} \Delta g_j = 0 \ (i-2 \geq j \geq 0) \tag{32}$$

Comparing Equation (28) and Equation (31) with Equation (27), we see that y_{i-1} and z_{i-1} can be chosen as ΔX_{i-1} or $H_{i-1}^T \Delta g_{i-1}$ or both of them, so, in general, we can write

$$y_{i-1} = C_1 \Delta X_{i-1} + C_2 H_{i-1}^T \Delta g_{i-1} \tag{33}$$

and

$$Z_{i-1} = K_1 \ \Delta X_{i-1} + K_2 \ H_{i-1}^T \ \Delta g_{i-1} \tag{34}$$

where C_1, C_2, K_1, K_2 are scalar coefficients.

Finally, we can conclude that the conjugacy condition Equations (14) is satisfied, if the matrix H is updated according to Equations (26), (33), (34). By different choices of the constants in Equation (26), different algorithms can be generated.

2.3.3 Initial H-matrix

According to the updation formulas (21), (26), (33), (34), the search direction (17) can be written as

$$P_i = \beta_i \ q_i \tag{35}$$

where β_i is a scalar defined by

$$\beta_i = 1 - K_2(\Delta g_{i-1}^T \ H_{i-1}^T \ g_i)/(z_{i-1}^T \ \Delta g_{i-1}) \tag{36}$$

and q_i is an $(n \times l)$ vector defined by

$$q_i = I - \left[\frac{\Delta X_{i-1} \ \Delta g_{i-1}^T}{\Delta X_{i-1}^T \ \Delta g_{i-1}}\right] \ H_{i-1}^T \ g_i \tag{37}$$

In the light of Equation (35), Equation (10) becomes

$$f(X_{i+1}) - f(X_i) = -(g_i^T q_i)^2/2q_i^T C q_i \tag{38}$$

Therefore, the nonorthogonality condition Equation (11) is replaced by

$$g_i^T q_i \neq \ 0, \ (n-1 \geq i \geq 0) \tag{39}$$

Now, we could know that the nonorthoganility condition Equation (39) can be satisfied if

$$g_i^T H_0^T g_i \neq 0, \ (n-1 \geq i \geq 0) \tag{40}$$

Since the gradients at different points are linearly independent, the initial matrix H_0 which satisfies condition Equation (40) must be such that the matrix A defined by

$$A = 1/2 \ (H_0 + H_0^T) \tag{41}$$

2.3.4 Setting up the General Algorithm

Finally, from the previous analysis of the properties of matrix H and initial matrix H_0, we can generate and algorithm having properties (a)–(d) in the following way :

i) choose the initial matrix H_0 such that A is positive definite or negative definite where $A = (1/2)(H_0 + H_0^T)$. Here if H_0 is symmetric, then $A = H_0$, meaning that H_0 must be positive or negative definite.

ii) update the matrix using the relation

$$H_i = H_{i-1} + \rho \frac{\Delta X_{i-1} (C_1 \Delta X_{i-1} + C_2 H_{i-1}^T \Delta g_{i-1})^T}{(C_1 \Delta X_{i-1} + C_2 H_{i-1}^T \Delta g_{i-1})^T \Delta g_{i-1}}$$

$$- \frac{H_{i-1} \Delta g_{i-1} (K_1 \Delta X_{i-1} + k_2 H_{i-1}^T \Delta g_{i-1})^T}{(K_1 \Delta X_{i-1} + K_2 H_{i-1}^T \Delta g_{i-1})^T \Delta g_{i-1}} \tag{42}$$

where ρ, C_1, C_2, K_1, K_2 are arbitrarily given real numbers and K_1 and K_2 must not vanish simultaneously.

iii) update the point X using the relations

$$P_i = H_i g_i, \ \Delta X_i = \alpha_i P_i, \ X_{i+1} = X_i - \Delta X_i \tag{43}$$

where α_i can be determined by a one-dimensional search along the direction P_i.

3. PARTICULAR OUADRATICALLY CONVERGENT ALGORITHMS

3.1 Algorithm I.

By setting $\delta =\ 1$, $C_1 = +1$, $C_2 = -1$, $K_1 = +1$, $K_2 = -1$ in the generalized form in section 2.3.2, we could get the algorithm

$$H_i = H_{i-1} + \frac{(\Delta X_{i-1} - H_{i-1} \Delta g_{i-1}) (\Delta X_{i-1} - H_{i-1}^T \Delta g_{i-1})^T}{(\Delta X_{i-1} - H_{i-1}^T \Delta g_{i-1})^T \Delta g_{i-1}} \tag{44}$$

where H_0 is defined to be (N x N) identity matrix and

$$\Delta g_{i-1} = g_i - g_{i-1}$$

This algorithm is restarted with $H_i = I$ when the inequality

$$| g_i^T P_i | < E$$

is satisfied where E is given small positive number or this algorithm can be restarted after n iterations with $H_i = H_0 = I$

3.2 Algorithm II

By setting $K_1 = 0$ and $K_2 = 1$ and $\delta = 0$ in the generalized form in section 2.3.2, we get algorithm

$$H_i = H_{i-1} - \frac{H_{i-1} \Delta g_{i-1} \Delta g_{i-1}^T H_{i-1}}{\Delta g_{i-1}^T H_{i-1} \Delta g_{i-1}} \tag{45}$$

This algorithm has $H_0 = I$ and is restarted at every nth iteration with $H_i = H_0 = I$

3.3 Algorithm III.

For the uniqueness of search direction, the sequence of search direction can be defined in the form

$$P_0 = H_0^T g_0 \tag{46}$$

$$P_i = \left[H_0 - \sum_{r=0}^{i-1} \frac{H_0 g_r X_r^T}{X_r^T g_r} \right]^T g_i \tag{47}$$

If the initial matrix H_0 is summetric, the search direction simplifies to

$$P_i = H_0 g_i + \frac{g_i^T H_0 g_i}{P_{i-1}^T g_{i-1}} P_{i-1} \tag{48}$$

because the following condition gold

$$g_j^T H_0^T g_i = 0 \ (i-1 \geq j \geq 0) \tag{49}$$

If the initial matrix H_0 is the identity matrix, that is, if $H_0 = I$, then Equations (46), (48) for the search directions are simplified to

$$P_0 = g_0 \tag{50}$$

and

$$P_i = g_i + \frac{g_i^T g_i}{g_{i-1}^T g_{i-1}} P_{i-1} \tag{51}$$

The algorithm represented by Equations (50), (51) is characterized by the following updating formula for the H–matrix

$$H_i = I + \frac{g_i^T P_{i-1}}{g_{i-1} g_{i-1}} \tag{52}$$

This algorithm is restarted at every $(n+1)$th iteration, counted from the previous starting or restarting point. This algorithm has the property that matrix H_n is the identity matrix.

3.4 Stopping Condition

Since optimum condition

$$g(X) = 0$$

can be achieved at any iteration, so above algorithms are stopped when

i) $g_i^T g_i < E$ when E is small real positive number

ii) The total number of iterations $= n$ when n is given number. In practice the realization of quadratic convergence on computer requires that highly precise arithmetic be used together with the high accuracy in the one dimensional search. In this consideration, the search is stopped when

$$\left| g_{i+1}^T P_i \right| < E \text{ when E is small real positive number}$$

Especially, this precise stopping condition is necessary for non-quadratic function.

4. APPLICATION TO NON-OUADRATIC FUNCTION

To apply Algorithms (I)–(III) to the minimization of a non-quadratic function, the following considerations are taken.

4.1 Starting Condition

For algorithms (I)–(III), any initial matrix H_0 must satisfy the condition that the matrix $(H_0 + H_0^T)/2$ is either positive definite or negative definite. In particular, H_0 can be the identity matrix.

4.2 Restarting Condition.

Generally, at the non-quadratic function, the minimal point can't be reached in n iterations, that means some more further iterations are needed. At this case, the algorithm may be restarted by setting

$$H_i = H_0$$

There are two conditions where the restart can be taken ;

i) when the inequality

$g_i^T P_i \quad < E$ Where E is prescribed small numbr

is satisfied, while the stopping condition is not satisfied

ii) at the nth or $(n+1)$th iteration, counted from the previous starting or restarting point.

4.3 Stopping Condition

For a non-quadratic function, the stopping condition of quadratic function may bring either a minimal point or nonminimal stationary point. But the probability of occurance of nonminimal stationary point can be reduced by selecting very small constant of E in stopping condition of quadratic function.

4.4 Precision Requirement

The precision requirement of quadratic function also hold for non-quadratic functions. This ensures fast convergence in the neighborhood of the minimal point.

5. NUMERICAL EXAMPLE

5.1 Quadratic Function

5.1.1 Starting Condition

$$H_0 = I$$

where I is the identity matrix

5.1.2 Stopping Condition

An algorithm is stopped when

$$g_i^T g_i < E$$

the small number E is given by

$$E = 10^{-4}$$

5.1.3 One-Dimensional Search

The one dimensional search is stopped when uncertainty of interval is less than or equal to E. The small number E is given by

$$E = 10^{-3}$$

5.1.4 Example problem (I)

The following problem of minimizing the quadratic function was considered.

$$f = (X_1 + X_2 + 0.5X_3)^2 + (X + 2X_2 + X_3 + X_4)^2 + (X_2 + X_4 + 1.5X_3)^2 + (0.5_1 + X_2 + 1.5_4 - 0.5)^2$$

This function admits the minimum $f = 0$ at the point defined by

$$X_1 = 0.5,\ X_2 = -0.5,\ X_3 = 0,\ X_4 = 0.5$$

The starting point of this problem is the point defined by

$$X_1 = 4,\ X_2 = 4,\ X_3 = 4,\ X_4 = 4$$

for a given initial matrix, all our three algorithms show same sequence of points, converge to the solution in almost same number of iterations. The result is shown in Table 1 using algorithm I and dichotomous technique.

Table 1.

K	Y	X_1	X_2	X_3	X_4
0	828.2500	4.00000	4.00000	4.00000	4.00000
1	0.5788	1.47216	–1.33851	0.74740	0.37618
2	0.0666	1.32077	–1.39190	0.40092	0.85364
3	0.0649	1.30950	–1.36094	0.36696	0.83314
4	0.0646	1.32535	–1.36572	0.36674	0.83693

Optimum Solution

No. of iterations=5 X value=(o.49996, 0.49999, 0.00002, 0.50001) Y value=0.0000

5.2 Non Quadratic Function

5.2.1 Starting Condition

For any algorithms, the initial matrix H_0 satisfying $(H_0 + H_0^T)/2$ is positive definite or negative definite can be used. The symmetric matrix was better. So that the following H_0 matrix was used

$$H_0 = I$$

5.2.2 Stopping Condition

An algorithm is stopped when

$$g_i^T g_i \leq E$$

The small number E is given by

$$E = 10^{-4}$$

but for a non quadratic function, above condition may lead minimum point or a nonminimal point. The probability of leading nonminimal point can be reduced setting E to very small number.

5.2.3 Restarting Condition

In general, optimal point cannot be reached in n iterations and further iterations may be needed. At this case, an algorithm can be restarted by setting

$$H_i = H_0$$

While the stopping condition is violated, an algorithm needs restarting when the following inequality

$$\left| g_i^T P_i \right| < E$$

is satisfied. The small number E is given by

$$E = 16^{-16}$$

5.2.4 One Dimensional Search

The one-dimensional search is stopped satisfying same condition as the case of quadratic function.

5.2.5 Example Problem (II)

$$f = 100(X_1^2 - X_2)^2 + (X_1 - 1)^2$$

with starting point defined by

$$X_1 = 1.2, \ X_2 = 1.0$$

Table 2.

K	Y	X_1	X_2	Step Size	N
0	19.3999	1.20000	1.00000	0.0	0
1	2.5610	1.09668	1.04297	0.000488	18
2	0.1659	1.05065	1.06344	0.000488	18
3	0.0379	1.04230	1.06739	0.000488	18
4	0.0032	1.03656	1.07011	0.000488	18
5	0.0017	1.03564	1.07054	0.000488	18
6	0.0012	1.03499	1.07083	0.000488	18
7	0.0012	1.03488	1.07087	0.000488	18
8	0.0012	1.03480	1.07088	0.000488	18
9	0.0012	1.03478	1.07088	0.000488	18
10	0.0012	1.03475	1.07086	0.000488	18
11	0.0012	1.03358	1.06775	0.109359	18
12	0.0011	1.00170	1.00287	0.022384	18
13	0.0000	1.00159	1.00292	0.000488	18
14	0.0000	1.00151	1.00296	0.000488	18
15	0.0000	1.00150	1.00297	0.000488	18

Optimum Solution

No. of iterations=16 X value=(1.00149, 1.00297) Y value=0.0000

Minimization of this function was satisfied when

$$X_1 = 1,\ X_2 = 1 \text{ with } f = 0$$

The result of this problem was given by Table 2 using algorithm III and dichotomous technique.

5.2.5 Example Problem (III)

$$f = (X_1 + 10X_2)^2 + 5(X_3 - X_4)^2 + (X_2 - 2X_3)^4 + 10(X_1 - X_4)^4$$

with starting point given by

$$X_1 = 10,\ X_2 = 10,\ X_3 = 10,\ X_4 = 10$$

minimum point can be reached when

$$X_1 = 0,\ X_2 = 0,\ X_3 = 0,\ X_4 = 0$$

having

$f = 0$

The result of this problem using Algorithm II and golden ratio method is given in Table 3.

Table 3.

K	Y	X_1	X_2	X_3	X_4	Step Size
0	22100.0000	10.00000	10.00000	10.00000	10.00000	0.0
1	14501.3828	9.95403	10.37608	8.32853	10.00000	0.000209
2	1621.9902	8.57425	–0.56650	2.73564	9.88178	0.006033
3	1495.3867	9.33373	–0.76857	2.62514	8.36524	0.009117
4	1288.3750	2.59742	–0.21852	2.88539	2.35752	0.193480
5	91.4339	2.59612	0.58644	1.25489	2.36303	0.000946
6	22.0516	2.45322	–0.15869	0.81290	2.30126	0.005996
7	17.3261	2.45682	–0.16751	0.80209	1.78873	0.033668
8	9.6233	0.96251	–0.04383	0.85112	0.76378	0.192287
9	3.1332	0.95172	0.04183	0.50671	0.77319	0.007925
10	1.2616	0.93008	–0.08748	0.38944	0.76006	0.008380
11	0.7821	0.90335	–0.09130	0.33304	0.52578	0.065161
12	0.3625	0.44944	–0.05208	0.35906	0.31052	0.264631
13	0.2499	0.44977	–0.02241	0.32131	0.31592	0.009117
14	0.1125	0.44092	–0.04435	0.23712	0.32469	0.027389
15	0.0591	0.42298	–0.04637	0.13444	0.21122	0.146374
16	0.0309	0.23714	–0.02961	0.13126	0.15545	0.551651
17	0.0228	0.23714	–0.02102	0.17682	0.15712	0.005396
18	0.0125	0.23503	–0.02317	0.14819	0.16733	0.044923
19	0.0049	0.22331	–0.02298	0.10084	0.10044	0.442021
20	0.0019	0.11218	–0.01223	0.08684	0.07621	1.632118
21	0.0015	0.11233	–0.01042	0.08559	0.07707	0.007925
22	0.0009	0.11206	–0.01109	0.07896	0.08148	0.046852
23	0.0002	0.10640	–0.01067	0.04574	0.04543	1.829701
24	0.0000	0.02266	–0.00237	0.03341	0.03123	9.721335
25	0.0000	0.02269	–0.00208	0.03309	0.03152	0.012976

Optimum Solution

No. of iterations=26 X value=(0.02266, 1.00226, 0.03219, 0.03231) Y value=0.0000

6. DISCUSSION OF OUTPUT

For 3 example problems, all three algorithms were performed and for each algorithm, two one dimensional search schemes were carried out. For problem III, the relaxed step size instead of optimal step size was also employed. The results were presented in Table 4

Table 4.

Algorithm	One Dimensional Search	Example: Problem I	Problem II	Problem III	Problem III relaxed
I	Dichotomous	5	8	26	50
	Golden-ratio	5	12	29	45
II	Dichotomous	4	50	26	37
	Golden-ratio	4	38	26	37
III	Dichotomous	5	16	35	77
	Golden-ratio	5	15	31	77

From the above result, the following comments are detected:

a) The use of relaxed step size cause the increase in the number of iterations. The major failure of relaxed step size appears in the final stage. The singular Hessian may be encountered near-by minimal point.

b) For problem II, the initial stage involves a lot of number of iteration. The reason can be thought that the nonminimal stationary area may be encountered. For Problem III, the final stage involves lots of iterations. The Hessian matrix at minimum point is either singular or zero.

c) For problem II, algorithm I is superior to algorithm II while for problem III, algorithm II is better than algorithm I reversively. This result shows that the function f(X) with much higher order than two has a singular Hessian at the minimal point. For that case, algorithm I (quasi-Newton method) converge slowly. But algorithm II (Projection method) seems to be less affected by such situation.

7. CONCLUSION

In this paper, a unified method to construct algorithms for the function minimization with several variables was described. The following properties are required: (a) the algorithms use one dimensional search only; (b) for the quadratic function, the algorithms are capable of converging quadratically to the minimal point in a number of iterations equal at most to the number of variable ; (c) the algorithms employ the function and it's gradient only ; (d) the algorithms employ only information at the present stage and immediately previous stage. It was shown that Algorithm I (rank-on algorithm), Algorithm II (projection algorithm), Algorithm III (fletcher-reeves algorithm) can be obtained as particular cases. The application of these algorithms to non-quadratic function was discussed. Additionally, in order to secure step size, two efficient one-dimensional search schemes ; dichotomous search technique and golden-ration search method, were studied. Particularly, in using step size for getting new search point, sometimes, the relaxed step size can be more efficient than optimal step size taken from one-dimensional search. It's efficiency was examined in the several example problems.

For the easy application of these algorithms to actual function optimization problem, a versatile computer program was developed and performed for several quadratic and non-quadratic function. In order to verify the property of these algorithms, 3 numerical examples ; one quadratic function and two non-quadratic functions, have been studied. For a quadratic function, given initial point X_0 and initial matrix that is symmetric, Algorithms I-III show almost in N iterations at most. From the discussion of the computer results, it was shown that the algorithm I converged rapidly near by the minimal point for non-quadratic function and the restarting scheme took a very important role in determining the number of iterations for convergence.

BIBLIOGRAPHY

1. Fletcher, R., and Reeves, C. M,. "Function Minimization by Conjugate Gradients", *Computer Journal,* Vol. 7. 2, 1964.
2. Huang, H. Y,. "Unified Approach to Quadratically Convergent Algorithms for Function Minimization", *Journal of Optimization Theory and Applications,* Vol. 5, No. 6, 1970.
3. Broyden, C. G., "Quasi-Newton Methods and Their Application to Function Minimization", *Mathematics of Computation,* Vol. 21, No. 99, 1967.
4. Huang, H. Y., and Levy, A. B,. "Numerical Experiments on Quadratically Convergent Algorithms for Function Minimization", *Journal of Optimization Theory and Applications,* Vol. 6, No. 3, 1970.
5. Huang, H. Y., and Chambliss, J. P., "Quadratically Convergent Algorithms and One-Dimensional Search Schemes", *Journal of Optimizaton Theory and Applications,* Vol. 11, No. 2, 1973.
6. Byron S. Gottfried and Joel Weisman, "Introduction to Optimizaton Theory", p. 95–112, Prentice-Hall International Series in Industrial and Systems Engineering, 1973.