Chapter 3 Linear Search Methods

- **One-dimensional (1-D) search methods, a. k. a. linear search methods, are the fundamental algorithms in the optimization theory. Basically, all the nonlinear algorithms can be attributed to the linear searches along a sequence of generated descent directions.**
- **For $f(X)$ with multiple variables, starting from X_k, the 1-D search along the direction S_k can be expressed as:**

$$\min f(X_k + \alpha S_k) = f(X_k + \alpha_k S_k)$$

$$X_{k+1} = X_k + \alpha_k S_k$$

which represents the minimization of $f(X_k + \alpha S_k)$ w.r.t. single variable α. The optimal step length α_k and then the minimum point X_{k+1} along S_k can be obtained.

One-dimensional Search Methods

- **1-D search is actually an iteration algorithm for minimization of functions with single variable. It can be simply written as:**

$$\min f(\alpha)$$

or in a more general form as:

$$\min f(x)$$

- **The algorithm for 1-D search consists of two steps:**

1) an initial internal including the minimum point needs to be determined.

2) then the length of the interval is reduced after every iteration until the minimum point is found.

3.1 Determination of Initial Interval

- **Unimodal Function**:

A function f of single variable x on an interval [a, b] is said to be unimodal if it has only one local minimum point in this interval.

On the left side of the minimum point, the function values decrease monotonically;

On the right side of the minimum point, the function values increase monotonically.

See the illustration graphs:

3.1 Determining Initial Interval (II)

- **Assuming $f(x)$ is a unimodal function on the search interval, if three adjacent points $x_1 < x_2 < x_3$ and their function values $f(x_1)$, $f(x_2)$, $f(x_3)$ are known, then the location of the minimum point can be determined by comparing these three function values.**
- **The graphs are illustrated as follows:**

1) if $f(x_1) > f(x_2) > f(x_3)$, the minimum point is located on the right hand side of x_2.

2) if $f(x_1) < f(x_2) < f(x_3)$, the minimum point is located on the left hand side of x_2.

3) if $f(x_1) > f(x_2) < f(x_3)$, the minimum point is located between x_1 and x_3. $[x_1, x_3]$ is the initial search interval.

3.1 Determining Initial Interval (III)

- With the algorithm, a sequence of searching points can be generated along some direction.
- By comparing the function values at these searching points, three points can be found to form a convex function by connecting these three points. The interval comprised of these three points must include the minimum point. This interval is called the Initial Search Interval, denoted as [a, b].
- The search interval is also referred to as Interval of Uncertainty (IU).

3.1 Determining Initial Interval (IV)

Basic steps for determining the initial interval:

1. Given an initial estimate of the minimum point, x_0, an initial step length h; Set $x_1=x_0$, denote $f_1=f(x_1)$;

2. Let $x_2=x_0+h$, denote $f_2 = f(x_2)$;

3. Compare the values of f_1 and f_2 to decide the search direction (forward or backward):

i) If $f_1 > f_2$, then double the step length by setting $h=2h$, go to step 4 and search forward;

ii) If $f_1 < f_2$, then change the search direction by setting $h = -h$, switch x_1 and x_2 as well as the function values f_1 and f_2; go to step 4 and search backward;

3.1 Determining Initial Interval (V)

4. Let $x_3 = x_0 + h$, denote $f_3 = f(x_3)$;

5. Comparing the values of f_2 and f_3:

i) If $f_2 < f_3$, then the initial interval is found as $[x_1, x_3]$;

ii) If $f_2 > f_3$, then increase the step length by setting $h = 2h$, $x_1 = x_2$, $x_2 = x_3$, go to step 4 and keep searching.

- **During the process of determining the initial interval, the initial step length should be chosen appropriately. In general, the initial step length is set as $h = 1.0$.**

3.2 Shortening the Search Interval

- **1-D search is an algorithm to obtain the minimum point by iteratively shortening the search interval (also called Interval of Uncertainty) along a given direction.**
- **The basic strategy of shortening the interval of uncertainty can be described as follows:**

Interpolate two points inside the current known interval; By comparing the function values at these two points, a smaller interval of uncertainty with the minimum point inside can be identified.

3.2 Shortening the Interval (II)

e.g., on a current interval of [a, b], arbitrarily choose two interpolation points x_1 and x_2:

if $f(x_1) < f(x_2)$, then the new search interval is [a, x_2];

if $f(x_1) > f(x_2)$, the new search interval becomes [x_1, b];

if $f(x_1) = f(x_2)$, then the new search interval is [x_1, x_2].

- **Repeat the above process, the interval of uncertainty can be shortened gradually.**
- **When $|b_n - a_n| \le \varepsilon$, or $|x_{n+1} - x_n| \le \varepsilon$, the given convergence accuracy, some point inside this interval can be approximately considered as the minimum point along this search direction.**

3.2 Shortening the Interval (III)

- **Any two interpolation points can shorten the search interval once. Different ways of generating these interpolation points construct different 1-D search algorithms, and consequently different ratio of reduction of the interval length.**
- **Ratio of Interval Reduction is defined as the ratio between the length of current interval and previous search interval:**

$$\tau_k = \frac{b_k - a_k}{b_{k-1} - a_{k-1}} \qquad (0 < \tau_k < 1) \qquad (k = 1,\ 2, \cdots)$$

3.3 Golden Section Method (GSM)

- **Golden Section Method, a.k.a. the 0.618 Method, is a 1-D search method choosing the interpolation points via the Symmetry Principle.**
- **The ratio of interval reduction is a constant, which can be deducted to be the Golden Section Number, i.e., 0.618. This is the way this search method is named.**

$$\tau_k = \frac{b_1 - a_1}{b - a} = \frac{b_2 - a_2}{b_1 - a_1} = \cdots = \frac{b_k - a_k}{b_{k-1} - a_{k-1}} = \tau = 0.618$$

- **The deduction of the ratio of interval reduction is presented here:**

Facts of Gold Section Number

Most Perfect Body:

Distance from navel to the bottom of the foot : Height = 0.618

Most Beautiful Face:

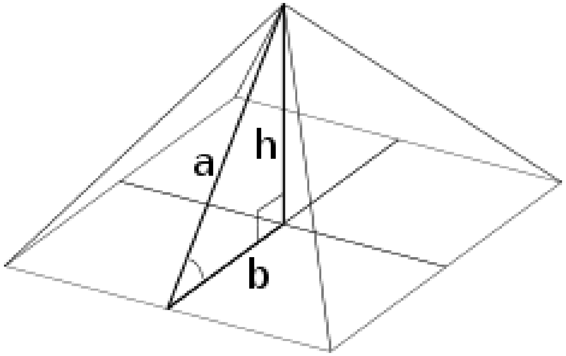
Distance from eyebrow to neck : distance from top of head to neck = 0.618

Most Comfortable Temperature:

0.618 * Body Temperature (37°C) = 23°C

Egyptian Pyramid:

b:a = 0.618



3.3 GSM (II)

- The iteration formula of the Golden Section Method (GSM) can be expressed as:

$$x_1 = \mathrm{a} + 0.382(\mathrm{b}\text{-}\mathrm{a})$$

$$x_2 = \mathrm{a} + 0.618(\mathrm{b}\text{-}\mathrm{a})$$

where, [a, b] represents the current search interval.

- Convergence Judgement: when $b_n - a_n \le \varepsilon$, the convergence accuracy, the algorithm is said to be converged. Then the middle point of that interval is considered as the approximation of the minimum point, i.e., the minimum point is chosen as:

$$x^* = (b_n + a_n)/2$$

3.3 GSM (III)

- **Given the convergence accuracy ε and the initial search interval [*a*, *b*], then the iteration times *n* to achieve the convergence can be determined as:**

$$0.618^n(b-a) \leq \varepsilon$$

$$\Rightarrow \quad n \geq \ln[\varepsilon/(b-a)] \,/\, \ln(0.618)$$

- **Therefore, the smaller the convergence accuracy ε is, or the larger the length of the interval *b*-*a* is, more iterations *n* are needed to get the algorithm converged.**

3.3 GSM (IV) - Algorithm

The algorithm of the GSM:

1. **Initial search interval [a_0, b_0] and the convergence accuracy ε are given;**

2. **let $a=a_0$, $b=b_0$;**

3. **The interpolation points are generated by GSM, the function values are also computed:**

$$x_1 = a + 0.382(b-a) \qquad f_1 = f(x_1)$$

$$x_2 = a + 0.618(b-a) \qquad f_2 = f(x_2)$$

4. **Compare f_1 and f_2 to determine the new interval:**

a) if $f_1 < f_2$, then the new search interval is [a, x_2]; let $b = x_2$, $x_2 = x_1$, $f_2 = f_1$ and denote $N_1 = 0$;

3.3 GSM - Algorithm (Cont'd)

b) if $f_1 > f_2$, then the new search interval is $[x_1, b]$; let $a = x_1$, $x_1 = x_2$, $f_1 = f_2$ and denote $N_1 = 1$;

5. Convergence judgement: when $b\text{-}a \leq \varepsilon$ is satisfied, the algorithm is terminated and the minimum point is $x^* = (b+a)/2$; otherwise go to step 6;

6. Generate new interpolation points:

if $N_0 = 0$, set $x_1 = a + 0.382(b\text{-}a)$, $f_1 = f(x_1)$;

if $N_0 = 1$, set $x_2 = a + 0.618(b\text{-}a)$, $f_2 = f(x_2)$;

go to step 4 to continue the search.

3.3 Example for GSM

Example 3-1:

Determine the minimum point of $f(x)=3x^3-4x+2$ using the Golden Section Method, given $x_0=0$, h=1, $\varepsilon=0.2$.

Sol: (1) (0, 2), (1, 1), (2, 18) $\Rightarrow$ IU is (0, 2)

(2) 1st: (0.764, 0.282), (1.236, 2.72) $\Rightarrow$ (0, 1.236)

2nd: (0.472, 0.317) $\Rightarrow$ (0.472, 1.236)

3rd: (0.944, 0.747) $\Rightarrow$ (0.472, 0.944)

4th: (0.652, 0.223) $\Rightarrow$ (0.472, 0.764)

5th: (0.584, 0.262) $\Rightarrow$ (0.584, 0.764) $\Rightarrow$ $b_n - a_n = 0.18 < \varepsilon = 0.2$

$\Rightarrow$ $x^* = (b_n + a_n)/2 = 0.674$ $\Rightarrow$ $f(x^*) = f(0.674) = 0.222$

3.4 Fibonacci Search Method (FSM)

- Similar to GSM, the Fibonacci Method (FSM) is also a 1-D search method shortening the search interval by choosing the interpolation points via the Symmetry Principle.
- FSM is derived based on the Fibonacci numbers. The ratio of interval reduction τ can be expressed as:

$$\tau_k = F_{n-k} / F_{n-k+1} \quad (n \text{ is a pre-determined integer})$$

where the Fibonacci sequence $\{F_k\}$ is generated as:

$$F_0 = F_1 = 1$$

$$F_k = F_{k-1} + F_{k-2} \qquad k = 2, 3, \ldots$$

3.4 FSM (II)

- **Procedures of the Fibonacci algorithm:**

Step 1 (k=1) : $\tau_1 = F_{n-1} / F_n = L_2 / L_1$

Step k : $\tau_k = F_{n-k} / F_{n-k+1} = L_{k+1} / L_k$

$$\Rightarrow \frac{L_k}{L_1} = \frac{L_k}{L_{k-1}} \cdot \frac{L_{k-1}}{L_{k-2}} \cdot \ \cdots \ \cdot \frac{L_2}{L_1} = \frac{F_{n-k+1}}{F_{n-k+2}} \cdot \frac{F_{n-k+2}}{F_{n-k+3}} \cdot \ \cdots \ \cdot \frac{F_{n-1}}{F_n} = \frac{F_{n-k+1}}{F_n}$$

$$\Rightarrow L_k = \frac{F_{n-k+1}}{F_n} L_1 \quad \text{when} \quad k = n \quad \Rightarrow \quad L_n = \frac{F_1}{F_n} L_1$$

- **How to determine n?**

Given the initial search interval [a, b] and the convergence accuracy $\varepsilon > 0$, then:

$$L_n = (F_1/F_n)L_1 = (b-a)/F_n \le \varepsilon \quad \Rightarrow \quad F_n \ge (b-a)/\varepsilon \quad \Rightarrow \quad n$$

3.4 FSM (III) - Further Discussion

- **Assuming that the ratio of two consecutive Fibonacci numbers, F_{k-1}/F_k, converges to a finite limit α, then:**

$$\alpha = \lim_{k\to\infty}\frac{F_{k-1}}{F_k} \;\Rightarrow\; \frac{1}{\alpha} = \lim_{k\to\infty}\frac{F_k}{F_{k-1}} = \lim_{k\to\infty}\frac{F_{k-1}+F_{k-2}}{F_{k-1}} = 1+\lim_{n\to\infty}\frac{F_{k-2}}{F_{k-1}} = 1+\alpha$$

$$\Rightarrow\; \alpha^2+\alpha-1=0 \;\Rightarrow\; \alpha = \frac{\sqrt{5}-1}{2} = 0.618$$

- **Interval Reduction Ratio:**

$$\frac{\tau_{GSM}}{\tau_{FSM}} = (0.618)^n F_n \approx 1.17 \quad \text{when} \quad n>5$$

⇒ **FSM is approximately 17% better than GSM in interval reduction. Disadvantage is that the total number of iterations *n* needs to be determined before starting the search.**

3.4 Example for FSM

Example 3-2:

Determine the minimum point of $f(x) = 3x^3 - 4x + 2$ using FSM, given $x_0 = 0$, h = 1, $\varepsilon = 0.2$.

Sol: (1) (0, 2), (1, 1), (2, 18) $\Rightarrow$ IU is (0, 2)

(2) $F_n \geq (b\text{-}a)/\varepsilon = 10 \Rightarrow n = 6 \Rightarrow \tau = F_5/F_6 = 8/13 \Rightarrow L_2 = 16/13$

1st: $f_1 = f(10/13) = 0.288 < f_2 = f(16/13) = 2.672 \Rightarrow (0,\ 16/13)$

2nd: $f_1 = f(6/13) = 0.448 > f_2 = f(10/13) = 0.288 \Rightarrow (6/13,\ 16/13)$

3rd: $f_1 = f(10/13) = 0.288 < f_2 = f(12/13) = 0.671 \Rightarrow (6/13,\ 12/13)$

4th: $f_1 = f(8/13) = 0.237 < f_2 = f(10/13) = 0.288 \Rightarrow (6/13,\ 10/13)$

5th: $x_1 = x_2 = 8/13 \Rightarrow x^* = 8/13 \Rightarrow f^* = f(8/13) = 0.237$

3.5 Powell's Quadratic Interpolation Method

- **Powell's Quadratic Interpolation Method**, is also called the **parabolic method**, is another 1-D search method. It choose the minimum point of the quadratic interpolation function as the new interpolation point to shorten the search interval.
- The interpolation curve is formed by connecting several known points, the function represented by this curve is called the interpolation function.
- The interpolation methods include the polynomial interpolation, the sample interpolation, etc.

3.5 Powell's QIM (II)

- The polynomial interpolations mainly consist of linear, quadratic and cubic interpolations, where the Quadratic Interpolation is mostly used.
- As described before, the initial search interval is a closed interval comprised by three points which comprise a convex function. Assume these points are a<c<b, with the function values $f_a > f_c < f_b$.
- Denote x_1=a, x_2=c, x_3=b and f_1=f_a, f_2=f_c, f_3=f_b, where $f_1 > f_2 < f_3$. Thus, three points (x_1, f_1), (x_2, f_2), (x_3, f_3) can be acquired in the coordinate plane.

3.5 Powell's QIM (III)

- A quadratic (parabolic) curve can be obtained by connecting these three points. The corresponding interpolation function can be expressed:

$$p(x) = \alpha_2 x^2 + \alpha_1 x + \alpha_0$$

- The minimum point of this function, x_p, can be determined by taking the derivative of $p(x)$ with respect to x. It can be written as:

$$x_p = -\alpha_1 / 2\alpha_2$$

In order to make sure x_p is the minimum point, should have $d^2p/dx^2 = 2\alpha_2 > 0$.

3.5 Powell's QIM (IV)

Plug in x_1, x_2, x_3 and f_1, f_2, f_3 to get:

$$f_1 = \alpha_2 x_1^2 + \alpha_1 x_1 + \alpha_0$$

$$f_2 = \alpha_2 x_2^2 + \alpha_1 x_2 + \alpha_0$$

$$f_3 = \alpha_2 x_3^2 + \alpha_1 x_3 + \alpha_0$$

Solve these three equations for $\alpha_0, \alpha_1, \alpha_2$:

$$\begin{bmatrix} \alpha_0 \\ \alpha_1 \\ \alpha_2 \end{bmatrix} = \frac{-1}{(x_1 - x_2)(x_2 - x_3)(x_3 - x_1)} \begin{bmatrix} f_1(x_2 - x_3)x_2 x_3 + f_2(x_3 - x_1)x_3 x_1 + f_3(x_1 - x_2)x_1 x_2 \\ -f_1(x_2^2 - x_3^2) - f_2(x_3^2 - x_1^2) - f_3(x_1^2 - x_2^2) \\ f_1(x_2 - x_3) + f_2(x_3 - x_1) + f_3(x_1 - x_2) \end{bmatrix}$$

3.5 Powell's QIM (IV)

Then x_p can be expressed as:

$$x_p = -\frac{\alpha_1}{2\alpha_2} = \frac{1}{2}\frac{(x_2^2 - x_3^2)f_1 + (x_3^2 - x_1^2)f_2 + (x_1^2 - x_2^2)f_3}{(x_2 - x_3)f_1 + (x_3 - x_1)f_2 + (x_1 - x_2)f_3}$$

To ensure x_p is the minimum point, it should have:

$$\mathbf{d^2}p/\mathbf{d}x^2 = 2\alpha_2 > 0 \quad \Rightarrow \quad \alpha_2 > 0$$

$$\alpha_2 = -\frac{(x_2 - x_3)f_1 + (x_3 - x_1)f_2 + (x_1 - x_2)f_3}{(x_2 - x_3)(x_3 - x_1)(x_1 - x_2)} > 0$$

3.5 Powell's QIM (V)

- It can be seen that x_p is the minimum point of the interpolation function, but also an approximate minimum point of the original objective function. Use this point as the interpolation point for next search interval will definitely accerlate the shortening of the search interval.
- Convergence Criterion of QIM:

when $|x_2 - x_p| \le \varepsilon$, the algorithm can be terminated, and the one with the smaller function value between x_2 and x_p is considered as the minimum point.

3.5 Algorithm for Powell's QIM

- **Algorithm procedures for QIM:**

1. **Given the initial estimate x_0, initial step length h and convergence accuracy $\varepsilon > 0$;**
2. **Determine initial search interval $[a, b]$ and another point c inside this interval;**
3. **Let $x_1 = a < x_2 = c < x_3 = b$ and $f_1 = f(x_1) > f_2 = f(x_2) < f_3 = f(x_3)$;**
4. **Calculate the interpolation point x_p and $f_p = f(x_p)$;**
5. **Convergence Judgement: if $|x_2 - x_p| \le \varepsilon$, then the search can be terminated and the minimum point is found. Otherwise go to step 6;**

3.5 Algorithm for QIM (Cont'd)

6. Shortening the search interval with consideration of the following cases:

i) $x_p \le x_2, f_p \le f_2$, new IU is $[x_1, x_2]$ having x_p;

ii) $x_p \le x_2, f_p > f_2$, new IU is $[x_p, x_3]$ including x_2;

ii) $x_p > x_2, f_p \le f_2$, new IU is $[x_2, x_3]$ holding x_p;

iv) $x_p > x_2, f_p > f_2$, new IU is $[x_1, x_p]$ containing x_2.

7. Go to step 4 and search for new interpolation point.

3.5 Examples for Powell's QIM

Ex 3-3: Determine the minimum point of $f(x)=3x^3-4x+2$ through QIM, given $x_0 = 0$, h = 1, $\varepsilon = 0.2$.

Sol: (1) (0, 2), (1, 1), (2, 18) $\Rightarrow$ IU is (0, 2) including 1.

(2) 1st: $x_p = 5/9 < x_2 = 1$, $f_p = 0.292 < f_2 = 1$

$\Rightarrow$ new interval is (0, 1) including 5/9.

2nd: $x_p = 0.607 > x_2 = 5/9$, $f_p = 0.243 < f_2 = 0.292$

$\Rightarrow$ new interval is (5/9, 1) including 0.607.

$$x_p - x_2 = 0.607 - 0.556 = 0.051 < \varepsilon = 0.2$$

$$\Rightarrow f^* = \min\{f_p, f_2\} = 0.243 \Rightarrow x^* = x_p = 0.607$$

	GSM	QIM	FSM	Analytical Solution
Iteration	5	2	5	0
Result	0.675	0.607	0.617	0.667
Error	+1.0%	-9.0%	-7.5%	0%

3.6 Davidon's Cubic Interpolation Method

- **Davidon's Cubic Interpolation Method, is generally better than Powell's QIM, if derivatives of the objective function $f(x)$ are easy to evaluate.**

Consider the problem: min$f(x)$ along $x=x_0+\lambda$, where x_0 is the current point and λ is an unknown parameter. Let $f_0=f(x_0)$ and $f_\alpha=f(x_0+\alpha)$ where α is a given value of λ. Suppose we know:

$$G_0 = \frac{df}{d\lambda}\Big|_{\lambda=0} = f_0' \quad \text{and} \quad G_\alpha = \frac{df}{d\lambda}\Big|_{\lambda=\alpha} = f_\alpha'$$

- **Here $G_0 < 0$ is assumed. To cover the case $G_0 > 0$, i.e., minimum to the left, use $f_\alpha = f(x_0-\alpha)$.**

3.6 Davidon's CIM (II)

- **Minimization occurs in 3 steps:**

(a) Order of magnitude of λ_m - the minimized value of λ - is established;

(b) Upper and lower bounds are found for λ_m;

(c) Cubic interpolation is used for more precise bounds.

Step (a): Initial estimation of λ_m:

$$\alpha = \min\left\{ k,\ \frac{-2(f_0 - f_e)}{G_0} \right\}$$

where: k = some representative magnitude, usually $k = 2$

f_e = preliminary estimate of $f(x_0 + \lambda_m)$

3.6 Davidon's CIM (III)

Note: for quadratic functions, there stands:

$$\lambda_{\mathrm{m}} = \frac{-2(f_0 - f_e)}{G_0}$$

where f_{e} is an exact estimate of minimum of $f(x_0 + \lambda)$.

Proof: $f(x_0 + \lambda) = a(x_0 + \lambda)^2 + b(x_0 + \lambda) + c$

$\cdots\cdots$

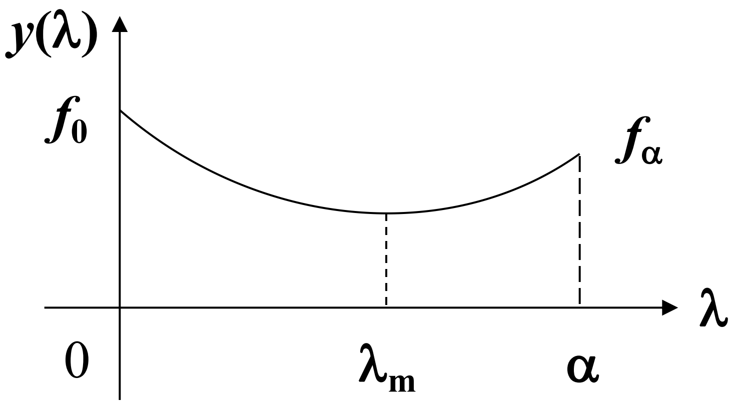
Step (b): **Assume cubic y(λ) approximates $f(x+\lambda)$, then:**

$y(0) = f_0 \quad \cdots (1) \qquad y(\alpha) = f_\alpha \quad \cdots (2)$

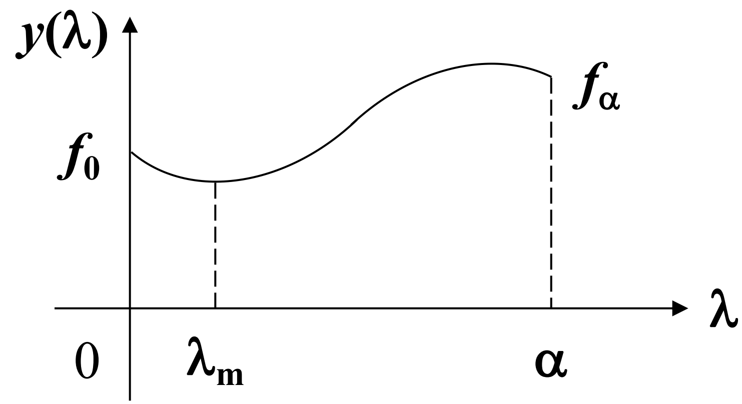
$y'(0) = G_0 \quad \cdots (3) \qquad y'(\alpha) = G_\alpha \quad \cdots (4)$

3.6 Davidon's CIM (IV)

Since $G_0 < 0$, λ_m will be located between [0, α] if either
$G_\alpha > 0$ (A) or $f_\alpha > f_0$ (B)



(A)



(B)

If neither, then replace $x+\alpha$ by $x+2\alpha$. Repeating if necessary, till the minimum of $y(\lambda)$ is bracketed; then start the interpolation.

3.6 Davidon's CIM (V)

Step (c): Cubic interpolation formula:

Assume $y(\lambda) = f_0 + G_0\,\lambda + y_2\lambda^2 + y_3\lambda^3 \qquad \cdots (5)$

. . . . . .

$\Rightarrow\ y_2 = -(G_0+Z)/\alpha$

$y_3 = (G_0+G_\alpha+2Z)/3\alpha^2$

where $Z = G_0+G_\alpha+3(f_0-f_\alpha)/\alpha$

. . . . . .

$\Rightarrow\ \lambda_m/\alpha = (G_0+Z+w)/(G_0+G_\alpha+2Z) = (Z+w-G_0)/(G_\alpha+2w-G_0)$

where $w = (Z^2-G_0G_\alpha)^{1/2}$

3.6 Algorithm for Davidon's CIM

1. Calculate $f_0 = f(x_0)$ and $G_0 = f(x_0)$; check that $G_0 < 0$; choose k and f_e and determine α (normally $k = 2$).

2. Evaluate $f_\alpha = f(x_0+\alpha)$ and $G_\alpha = f'(x_0+\alpha)$.

3. If $G_\alpha > 0$ or $f_\alpha > f_0$, go to Step 5; otherwise go to Step 4.

4. Replace α by 2α, evaluate new f_α and G_α, back to Step 3.

5. Interpolate on the interval $[0, \alpha]$ for λ_m using the cubic interpolation formula;

6. Return to Step 5 to repeat the interpolation on smaller interval $[0, \lambda_m]$ if $f'(x_0+\lambda_m) \geq 0$ or $f(x_0+\lambda_m) > f(x_0)$ OR $[\lambda_m, \alpha]$ if $f'(x_0+\lambda_m) < 0$.

7. Stop if λ_m is within ε distance to the endpoints or the length of the interval is less than ε.

3.6 Example for Davidon's CIM

Ex 3-5: Use Davidon's CIM to find the minimum point of $f(x) = x^4 - 4x + 1$, given $x_0 = 0$, $\varepsilon = 0.05$.

Sol: (1) $x_0 = 0$, $f_0 = 1$, $G_0 = -4$, $\alpha = \min\{k, -2(f_0 - f_e)/G_0\} = 1$ or 2

(2) choose $\alpha = 2 \Rightarrow f_\alpha = 9$, $G_\alpha = 28$

(3) $\lambda_m/\alpha = (\omega - G_0 + z)/(G_\alpha - G_0 + 2\omega) \Rightarrow$

$z = 3(f_0 - f_\alpha)/\alpha + G_0 + G_\alpha = 12$, $\omega = (z^2 - G_0 G_\alpha)^{1/2} = 16$

$\Rightarrow \lambda_m = 1$

(4) $f'(x_0 + \lambda_m) = f'(0+1) = 0 \Rightarrow$ new interval is $[0, 1]$

$\Rightarrow \alpha = 1 \Rightarrow f_\alpha = -2$, $G_\alpha = 0 \Rightarrow z = 5$, $\omega = 5$

$\Rightarrow \lambda_m = 1$ as before! $\Rightarrow x^* = x_0 + \lambda_m = 1$, $f(x^*) = -2$

Chapter 4 Unconstrained Optimization Methods

- **The kernel of unconstrained optimization methods is the descent algorithm. Descent algorithms have uniform iteration formulae. The key problems are how to choose the search direction and develop the 1-D search along these directions.**
- **This chapter mainly covers the methods that construct the search directions by utilizing the information of the first-order and second-order derivatives of the objective function, including Steepest Descent Method, Newton Method, Quasi-Newton Method, and Conjugate Gradient Method.**

4.1 Steepest Descent Method (SDM)

- **In this method, the negative gradient direction at the iteration point is used as the searching direction, since the negative gradient direction is the direction that the function value decreases most rapidly.**
- **Iteration Formula for SDM:**

$$S_k = -\nabla f(X_k) = -g_k,\ X_{k+1} = X_k + \alpha_k S_k$$

or $$X_{k+1} = X_k - \alpha_k \nabla f(X_k) = X_k - \alpha_k g_k$$

where α_k is determined by 1-D search:

$$f(X_k + \alpha_k S_k) = \min f(X_k + \alpha S_k)$$

To obtain the optimal value of the step length, take the derivative with respect to α ($df / d\alpha = 0$):

4.1 Steepest Descent Method (II)

$$[\nabla f(X_k + \alpha_k S_k)]^T S_k = [\nabla f(X_{k+1})]^T S_k = -g_{k+1}{}^T g_k = 0$$

$$\Rightarrow \quad g_{k+1}{}^T g_k = 0 \quad \text{or} \quad s_{k+1}{}^T s_k = 0$$

where, $g_k = \nabla f(X_k)$ is the gradient at point X_k.

- This represents that the gradients of two consecutive iteration points are orthogonal, i.e., these two searching directions are perpendicular to each other.
- When using SDM to generate iteration points, the path of approaching the minimum point is a step type curve. The closer to the minimum point, the step is smaller, and the approaching speed gets slower, as shown below:

4.1.2 Algorithm for SDM

1. Initial estimate X_0 and convergence accuracy $\varepsilon > 0$ are given; Set $k = 0$.

2. Calculate the gradient at this point and construct the search direction $S_k = -\nabla f(X_k)$;

3. Use the 1-D search to find the new iteration point:

$$f(X_k + \alpha_k S_k) = \min f(X_k + \alpha S_k)$$

$$X_{k+1} = X_k + \alpha_k S_k$$

4. Convergence judgment: if $\|\nabla f(X_{k+1})\| \le \varepsilon$, then the search can be terminated and the optimal solution is $X^* = X_{k+1}$; otherwise let $k=k+1$, go to Step 2 continue;

4.1.3 Further Discussion on SDM

- Utilize SDM to solve problems, quite a lot iterations are needed to get an approximation of the optimal solution, which is attributed to the characteristics of the gradient.
- The gradient is the description of the local variation of the function around certain point. Along the negative gradient direction, the function value decrease most rapidly. However, outside the neighborhood of this point, the function values may not decrease.
- Use the negative gradient direction as the searching direction, every iteration is able to make the function values decrease rapidly, from the local point of view; But from the global point of view, the path to the minimum point is not the best way. It can be proven that SDM only has a linear convergence rate.

4.1.4 Example for SDM

Ex.4-1: Solve the following problem using SDM.

$$\min f(X) = x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1$$

Given $X_0 = [1, 1]^T$, $\varepsilon = 0.1$.

Sol: $\nabla f(X) = [2x_1 - 2x_2 - 4, -2x_1 + 2x_2]^T$

(1) $\nabla f(X_0) = [-4, 2]^T \Rightarrow s_0 = [4, -2]^T \Rightarrow X_1 = [1+4\alpha_0, 1-2\alpha_0]^T$

$df(X_1)/d\alpha_0 = 0 \Rightarrow \alpha_0 = 1/4 \Rightarrow X_1 = [2, 0.5]^T \Rightarrow \nabla f(X_1) = [-1, -2]^T$

(2) $s_1 = [1, 2]^T \Rightarrow X_2 = [2+\alpha_1, 0.5+2\alpha_1]^T$

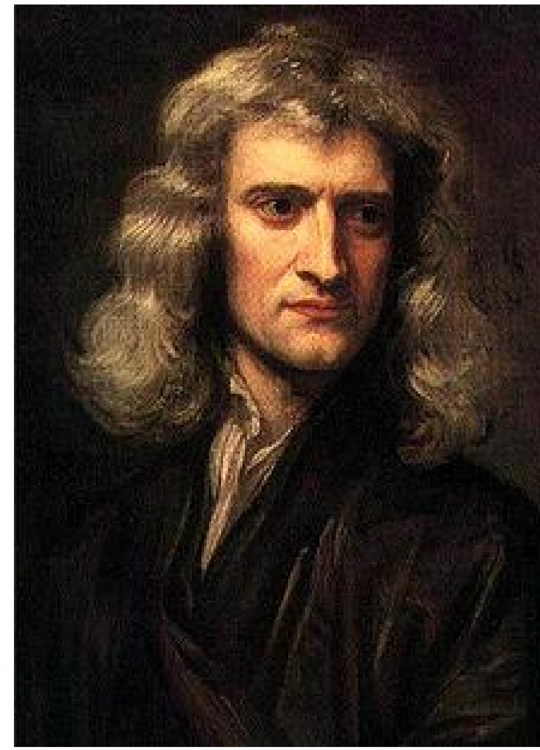
$df(X_2)/d\alpha_1 = 0 \Rightarrow \alpha_1 = 1/2 \Rightarrow X_2 = [2.5, 1.5]^T \Rightarrow \nabla f(X_2) = [-2, 1]^T$

……

See graphic illustration.

About Sir Isaac Newton

Isaac Newton (1642--1727) was an English physicist, mathematician, astronomer, natural philosopher, alchemist, and theologian, has been "considered by many to be the greatest and most influential scientist who ever lived."



Godfrey Kneller's 1689 Portrait of Isaac Newton (age 46)

Newton’s major achievements:

- His monograph *Philosophiae Naturalis Principia Mathematica*, published in 1687 (written in Latin), lays the foundations for most of classical mechanics. In this work, Newton described universal gravitation and the three laws of motion, which dominated the scientific view of the physical universe for the next three centuries.
- Newton built the first practical reflecting telescope and developed a theory of color based on the observation that a prism decomposes white light into the many colors that form the visible spectrum.
- In mathematics, Newton shares the credit with Gottfried Leibniz for the development of differential and integral calculus. He also demonstrated the generalized binomial theorem, developed Newton's method for approximating the roots of a function, and contributed to the study of power series.

4.2 Newton's Method

- **Newton's Method** is a classic optimization method. Its search direction is constructed by utilizing the information of the negative gradient and the Hessian matrix of the function. This direction is called the **Newton Direction**.
- Newton's method is derived upon the base of the Taylor's expansion for functions with multiple variables. Expand $f(X)$ at the current iteration point X_k to the second order approximation:

$$f(X) \approx f(X_k) + \left[\nabla f(X_k)\right]^T (X - X_k) + \frac{1}{2}(X - X_k)^T \nabla^2 f(X_k)(X - X_k)$$

4.2.1 Newton's Method

$$\left.\frac{\partial f(X)}{\partial X}\right|_{X=X_{k+1}} = 0 \;\Rightarrow\; \nabla f(X_k) + \nabla^2 f(X_k)(X_{k+1} - X_k) = 0$$

$$\Rightarrow X_{k+1} = X_k - [\nabla^2 f(X_k)]^{-1} \nabla f(X_k)$$

Let $\quad S_k = -[\nabla^2 f(X_k)]^{-1} \nabla f(X_k) \;\Rightarrow\; X^{k+1} = X_k + S_k$

- **It's the iteration formula for Newton's method, where**

$$\boldsymbol{S_k = -[\nabla^2 f(X_k)]^{-1} \nabla f(X_k)}$$

is called Newton's Direction.

- **Comparing with the generalized iteration formula for descent algorithms ($X_{k+1} = X_k + \alpha_k S_k$), this formula has no step length α_k, or say $\alpha_k = 1$, which means there is no need for 1-D search in basic Newton's method.**

4.2.1 Newton's Method (II)

- For the positive definite quadratic function (PDQF), the second-order Taylor expansion is itself, so the X_{k+1} obtained via Newton's method is the exact minimum point of the objective function. Hence, using the basic Newton's method for the PDQF, any initial starting point can approach to the minimum point quickly, as long as it follows the Newton's direction.
- For nonlinear functions with positive definite Hessians, the second-order Taylor expansion is an approximation to the objective function. Thus the X_{k+1} acquired by Newton's method is an approximate minimum point. Use this point as the starting point for the next iteration can accelerate the approaching to the minimum point.
- For general nonlinear functions, if their Hessians are not positive definite, then the Newton's Method may not be applicable for these functions.

4.2.3 Algorithm for Newton's Method

1. Initial estimate X_0 and convergence accuracy $\varepsilon > 0$ are given, set $k = 0$;

2. Calculate the gradient, the Hessian and its inverse at point X_k;

3. Construct Newton's direction: $S_k = -[\nabla^2 f(X_k)]^{-1} \nabla f(X_k)$;

4. Develop the 1-D search along the direction S_k to obtain the step length α_k and the next iteration point $X_{k+1} = X_k + \alpha_k S_k$;

5. Convergence Judgement: if $\|\nabla f(X_{k+1})\| \leq \varepsilon$, the search can be terminated and the optimal solution is $X^* = X_{k+1}$, otherwise, let $k = k+1$ and return to Step 2 and continue.

4.2.4 Examples for Newton's Method

Ex 4-2: Use Newton's Method to solve

$$\min f(X) = x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1$$

Given $X_0 = [1, 1]^T$, $\varepsilon = 0.1$.

Sol: $\nabla f(X) = [2x_1 - 2x_2 - 4, -2x_1 + 2x_2]^T$, $\nabla^2 f(X)$ is P. D.

(1) $\nabla f(X_0) = [-4, 2]^T$, $[\nabla^2 f(X)]^{-1} = \ldots\ldots$

$\Rightarrow s_0 = -[\nabla^2 f(X)]^{-1} \nabla f(X_0) = [3, 1]^T$

$\Rightarrow X_1 = X_0 + s_0 = [4, 2]^T$

$\Rightarrow \nabla f(X_1) = [0, 0]^T$

$\Rightarrow X^* = X_1 = [4, 2]^T \Rightarrow f(X^*) = -8$

4.3 Quasi-Newton's Method

- **Quasi-Newton Method** is the modification of Newton's method. The search directions of this method are generated in a recursive way and approach to Newton's direction eventually, without calculating the Hessian and its inverse of the function. It can be proven that **Quasi-Newton Method has the superlinear convergence rate**.
- Use the Taylor's expansion, we can get:

$$y_k = G_k s_k \quad \text{or} \quad s_k = G_k^{-1} y_k$$

where
$$y_k = \nabla f(X_{k+1}) - \nabla f(X_k) = g_{k+1} - g_k$$

$$s_k = X_{k+1} - X_k \qquad G_k = \nabla^2 f(X_k)$$

4.3 Quasi-Newton's Method (II)

- **Previous discussion has shown that the Hessian G_k can't be ensured to be positive definite at every iteration. Thus we may use a positive definite and symmetric matrix H_k to approximate G_k^{-1}, and update H_k with H_{k+1} after each iteration, to make it closer to G_k^{-1}.**

  Hence, one can have $s_k = H_{k+1} y_k$

  This is called "Quasi-Newton Condition".

  Q: How to update H_k?

- **Updating Formula:**

$$H_{k+1} = H_k + \Delta H_k$$

where, ΔH_k is called the updating matrix.

4.3.1 Updating Formulae

- **There are two ways to obtain the updating matrix ΔH_k, known as "Rank One Correction" and "Rank Two Correction".**
- **Rank One Correction:**

$$\boldsymbol{H_{k+1} = H_k + auu^{\mathrm{T}}}$$

$$\Rightarrow \quad H_{k+1} = H_k + \frac{(s_k - H_k y_k)(s_k - H_k y_k)^{\mathrm{T}}}{(s_k - H_k y_k)^{\mathrm{T}} y_k}$$

- **Rank Two Correction:**

$$\boldsymbol{H_{k+1} = H_k + auu^{\mathrm{T}} + bvv^{\mathrm{T}}}$$

$$\Rightarrow \quad H_{k+1} = H_k + \frac{s_k s_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k} - \frac{H_k y_k (H_k y_k)^{\mathrm{T}}}{y_k^{\mathrm{T}} H_k y_k}$$

4.3.2 Algorithm for Quasi-Newton's Method

1. X_0, H_0 (usually H_0=I) and convergence accuracy $\varepsilon>0$ are given.

2. Set $k=0$ and compute $g_k=\nabla f(X_k)$.

3. Choose $p_k=-H_k g_k$ as the search direction.

4. Develop 1-D search and determine the optimal step length α_k, identify the new iteration point $X_{k+1}=X_k+\alpha_k p_k$.

5. Convergence Judgement: if $\|\nabla f(X_{k+1})\| \le \varepsilon$, search can be terminated and the optimal solution is $X^*=X_{k+1}$; otherwise continue searching.

6. Update H_k to H_{k+1}: $H_{k+1}=H_k+\Delta H_k$.

7. let $k=k+1$, return to Step 2.

4.3.3 DFP Algorithm

- **The algorithm using above "Rank Two Correction" updating formula is called DFP algorithm (Davidon-Fletcher-Powell), i.e.:**

$$\Rightarrow \quad H_{k+1} = H_k + auu^{\mathrm{T}} + bvv^{\mathrm{T}} = H_k + \frac{s_k s_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k} - \frac{H_k y_k (H_k y_k)^{\mathrm{T}}}{y_k^{\mathrm{T}} H_k y_k}$$

Ex 4-3： Solve the problem via DFP algorithm.

$$\min f(X) = x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1$$

Given: $X_0 = [1, 1]^{\mathrm{T}}$, $\varepsilon = 0.1$.

4.3.4 Solution to Example 4-3

Sol: $\nabla f(X) = [2x_1 - 2x_2 - 4, -2x_1 + 2x_2]^T$

(1) $g_0 = \nabla f(X_0) = [-4, 2]^T$, $H_0 = \mathrm{I}$ $\Rightarrow$ $p_0 = -H_0 g_0 = [4, -2]^T$

$\Rightarrow X_1 = X_0 + \alpha_0 p_0 = [1+4\alpha_0, 1-2\alpha_0]^T$

$df(X_1)/d\alpha_0 = 0 \Rightarrow \alpha_0 = 1/4 \Rightarrow X_1 = [2, 0.5]^T \Rightarrow \nabla f(X_1) = [-1, -2]^T$

(2) $H_1 = H_0 + s_0 s_0^{\mathrm{T}} / s_0^{\mathrm{T}} y_0 - (H_0 y_0)(H_0 y_0)^{\mathrm{T}} / y_0^{\mathrm{T}} H_0 y_0$

$H_0 = \mathrm{I}$, $s_0 = X_1 - X_0 = [1, -0.5]^T$, $y_0 = g_1 - g_0 = [3, -4]^T$

$\Rightarrow$ $s_0^{\mathrm{T}} y_0 = 5$, $y_0^{\mathrm{T}} y_0 = 25$ $\Rightarrow$ $H_1 = [0.84, 0.38, \quad 0.38, 0.41]$

$\Rightarrow$ $p_1 = -H_1 g_1 = [1.6, 1.2]^T$

$\Rightarrow X_2 = X_1 + \alpha_1 p_1 = [2+1.6\alpha_1, 0.5+1.2\alpha_1]^T$

$df(X_2)/d\alpha_1 = 0 \Rightarrow \alpha_1 = 5/4 \Rightarrow X_2 = [4, 2]^T \Rightarrow \nabla f(X_1) = [0, 0]^T$

$\Rightarrow X^* = X_2 = [4, 2]^T \Rightarrow f(X^*) = -8$

4.3.5 BFGS Algorithm

- **Another well-known Rank Two Correction updating formula is BFGS formula (Broyden-Fletcher-Goldfarb-Shanna), the corresponding algorithm is called BFGS algorithm. In general, BFGS algorithm is better than DFP algorithm.**
- **The idea of BFGS algorithm is to use a positive definite symmetric matrix B_k to approximate G_k, NOT G_k^{-1}.**

Hence, the Quasi-Newton Condition becomes:

$$y_k = B_{k+1} s_k \quad \text{where} \quad B_k = H_k^{-1}$$

Rank Two Correction becomes: $B_{k+1} = B_k + a u u^{\mathrm{T}} + b v v^{\mathrm{T}}$

$$\Rightarrow \quad H_{k+1} = H_k + \left(1 + \frac{y_k^{\mathrm{T}} H_k y_k}{s_k^{\mathrm{T}} y_k}\right) \frac{s_k s_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k} - \frac{s_k y_k^{\mathrm{T}} H_k + H_k y_k s_k^{\mathrm{T}}}{s_k^{\mathrm{T}} y_k}$$

4.4 Conjugate Gradient Method

- **Conjugate Gradient Method is an unconstrained optimization method using the conjugate directions as the search directions.**
- **Conjugate Direction:**

  H is a symmetric positive definite matrix. If there are a set of nonzero vectors $s_1, s_2, \ldots, s_n$ satisfying:

  $$s_i^{\mathrm{T}} H s_j = 0 \qquad (i \neq j)$$

  Then these set of vectors are said to be conjugate with respect to H, and they are called as a set of conjugate vectors / directions of H.
- **When H is the unit matrix I, there has: $s_i^{\mathrm{T}} s_j = 0$ $(i \neq j)$. Vectors s_i $(i = 1, \ldots, n)$ are called orthogonal to each other.**
- **Conjugation is the generalization of orthogonality and orthogonality is a special case of conjugation.**

4.4.1 Properties of Conjugate Directions

- **Properties of Conjugate Directions:**

(1) If s_i (i = 1, 2,…, n) are the conjugate directions of positive definite symmetric H, then s_i (i = 1, 2,…, n) are linear independent.

(2) Positive definite symmetric matrix $H = (a_{ij})_{n \times n}$ has no more than n different conjugate directions.

(3) If s_i (i = 1, 2,…, n) are n vectors conjugate with respect to H, then for PDQFs, starting from any initial point and do linear search along these n directions, at most n iterations are needed to reach the minimum point of the object functions.

Prove (1)

4.4.2 Generation of Conjugate Directions

- **Generation procedures of conjugate directions:**

1. Given X_0, let $p_0 = -g_0 = -\nabla f(X_0)$, calculate $G_0 = \nabla^2 f(X_0)$;

2. $\min f(X_0 + \alpha p_0) \Rightarrow \alpha_0 \Rightarrow X_1 = X_0 + \alpha_0 p_0 \Rightarrow g_1 = -\nabla f(X_1)$, $G_1 = \nabla^2 f(X_1)$;

3. At X_1, set search direction $p_1 = -g_1 + \beta_0 p_0$, β_0 can be determined by properties of the conjugate directions:

at point X_1, vectors p_0, p_1 are conjugate with respect to symmetric positive definite matrix $G_1 \Rightarrow$

$$p_1^T G_1 p_0 = 0 \;\Rightarrow\; (-g_1 + \beta_0 p_0)^T G_1 p_0 = 0 \;\Rightarrow\; \beta_0 = \frac{g_1^T G_1 p_0}{p_0^T G_1 p_0}$$

4.4.2 Generation of CDs (Cont'd)

Similarly, at point X_k, the search direction is expressed as $p_k = -g_k + \beta_{k-1} p_{k-1}$, where β_{k-1} can be determined by the conjugate relationship among the search directions:

$$\beta_{k-1} = \frac{g_k^{\mathrm{T}} G_k p_{k-1}}{p_{k-1}^{\mathrm{T}} G_k p_{k-1}}$$

$\Rightarrow$ $X_{k+1} = X_k + \alpha_k p_k$, where min $f(X_k + \alpha_k p_k)$ $\Rightarrow$ α_k

…

until $k = n$-1.

In this way, n conjugate directions $p_0, p_1, \ldots, p_{n-1}$ can be identified.

4.4.3 Algorithm for CGM

1. Given X_0 and convergence accuracy $\varepsilon > 0$, calculate $p_0 = -g_0 = -\nabla f(X_0)$, $G_0 = \nabla^2 f(X_0)$; Set $k = 0$;

2. Compute α_k: $f(X_k + \alpha_k p_k) = \min f(X_k + \alpha\, p_k) \Rightarrow \alpha_k$

3. $X_{k+1} = X_k + \alpha_k p_k \ \Rightarrow\ g_{k+1} = \nabla f(X_{k+1})$, $G_{k+1} = \nabla^2 f(X_{k+1})$;

4. Convergence Judgment: if $\|\nabla f(X_{k+1})\| \leq \varepsilon$, terminate the algorithm, and the optimal solution is $X^* = X_{k+1}$; otherwise continue searching;

5. $p_{k+1} = -g_{k+1} + \beta_k p_k$; $\beta_k = \dfrac{g_{k+1}^{\mathrm{T}} G_{k+1} p_k}{p_k^{\mathrm{T}} G_{k+1} p_k}$

6. let $k = k+1$,return to Step 2.

4.4.3 PRP and FR algorithms

- **In the algorithm of the conjugate gradient method:**

$$\beta_k = \frac{g_{k+1}^{\mathrm{T}} G_{k+1} p_k}{p_k^{\mathrm{T}} G_{k+1} p_k}$$

- **PRP (Polyok-Ribere-Polak) algorithm:**

$$\beta_k = \frac{g_{k+1}^{\mathrm{T}} (g_{k+1} - g_k)}{\left\| g_k \right\|^2}$$

- **FR (Fletcher-Reeves) algorithm:**

$$\beta_k = \frac{\left\| g_{k+1} \right\|^2}{\left\| g_k \right\|^2}$$

4.4.4 Further discussion on CGM

- **Conjugate Gradient Method utilizes the gradients of the function to construct the conjugate directions. It has the superlinear convergence rate.**
- **For PDQFs with *n* variables, conduct 1-D search along the conjugate gradient directions, at most *n* iterations are needed to reach the minimum point.**

Ex 4-4: Solve the problem using FR algorithm.

$$\min f(X) = x_1^2 + 2x_2^2 - 2x_1x_2 - 4x_1$$

Given $X_0 = [1, 1]^T$, $\varepsilon = 0.1$.

4.4.5 Solution to Example 4-4

Sol: $\nabla f(X) = [2x_1 - 2x_2 - 4, -2x_1 + 2x_2]^T$

(1) $g_0 = \nabla f(X_0) = [-4, 2]^T \Rightarrow p_0 = -g_0 = [4, -2]^T$

$\Rightarrow X_1 = X_0 + \alpha_0 p_0 = [1+4\alpha_0, 1-2\alpha_0]^T$

$df(X_1)/d\alpha_0 = 0 \Rightarrow \alpha_0 = 1/4 \Rightarrow X_1 = [2, 0.5]^T \Rightarrow \nabla f(X_1) = [-1, -2]^T$

(2) $\beta_0 = g_1^T g_1 / g_0^T g_0 = 5/20 = 1/4$

$\Rightarrow p_1 = -g_1 + \beta_0 p_0 = [2, 1.5]^T$

$\Rightarrow X_2 = X_1 + \alpha_1 p_1 = [2+2\alpha_1, 0.5+1.5\alpha_1]^T$

$df(X_2)/d\alpha_1 = 0 \Rightarrow \alpha_1 = 1 \Rightarrow X_2 = [4, 2]^T \Rightarrow \nabla f(X_1) = [0, 0]^T$

$\Rightarrow X^* = X_2 = [4, 2]^T \Rightarrow f(X^*) = -8$