Chapter-1

A Study on Conjugate Gradient Methods and their modifications.

Introduction:

Because of the advances in Science, Engineering, Economics etc studies on global and local optimization for unconstrained problems have become a topic of great concern. In recent years there has been the great deal of interest in the development of optimization algorithms that deal with the problems of finding a global or local minimum of a given problem.. Unconstrained optimization problem arise in virtually in areas in Science and Engineering, and in many areas of the Social Sciences. A significant percentage of real world optimization problems are data fitting problem. The size of real world unconstrained optimization problem is widely distributed, varying from small problems to large problems. In many cases, the objective function $f(x)$ is a complete routine that is expensive to evaluate so that even small problems are expensive and difficult to solve. The user of an unconstrained optimization problem is expected to provide the function $f(x)$ and a starting guess to the solution x_o .The routine is expected to return and estimate of local minimiser x^* (say) of f(x).But in most cases they are not provided and instead is approximated in various ways by the algorithm. Approximating these derivatives is one of the main challenges of creating unconstrained optimization method. The other main challenges to create methods that will converge to a local minimiser even if x_o far from any minimum points. These referred to as the global phase of the method. The part of the method that converges to x_o, once it is closed to it is referred as the local phase of the method. For problems with large number of variables, the number of arithmetic operations required by the method and storage requirement of the method become increasing important.

There are different methods to solve the unconstrained optimization problems. Some of the popular methods are as follows:

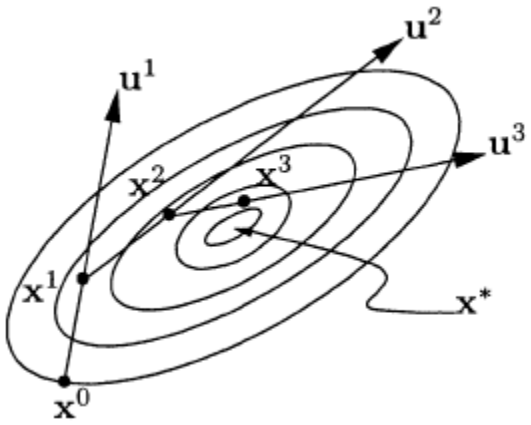
Direct Search method

Over the last few decades many powerful ***direct search algorithms*** have been developed for the unconstrained minimization of general functions. These algorithms require an initial estimate to the optimum point, denoted by x^0. With this estimate as starting point, the algorithm generates a sequence of estimates x^0,

x^1, x^2, . . . , by successively searching *directly* from each point in a direction of *descent* to determine the next point. The process is terminated if either no further progress is made, or if a point x^k is reached at which the first necessary condition $\nabla f(x) = 0$

Descent method with line search:

An important sub-class of direct search methods, specifically suitable for smooth functions, are the so-called *line search* descent methods. Basic to these methods is the selection of a descent direction u^{i+1} at each iterate x* that ensures descent at x* in the direction u^{i+1}, i.e. it is required that the directional derivative in the direction u^{i+1} be negative i,e., $\nabla^T f(x^i)u^{i+1} < 0$



Sequence of line search descent directions and steps

Descent method with trust region :

Soft line search method :

Many researchers in optimization have proved their inventiveness by producing new line search methods or modifications to known methods. In the early days of optimization exact line search was dominant. Now, soft line search is used more and more, and we rarely see new methods presented which require exact line search.

An advantage of soft line search over exact line search is that it is the faster of the two. If the first guess on the step length is a rough approximation to the

minimizer in the given direction, the line search will terminate immediately if some mild criteria are satisfied. The result of exact line search is normally a good approximation to the result, and this can make descent methods with exact line search find the local minimizer in fewer iterations than what is used by a descent method with soft line search. However, the extra function evaluations spent in each line search often makes the descent method with exact line search a loser.

The purpose of the algorithm is to find α_s, and acceptable argument for the function $\psi(\alpha) = f(x+\alpha h)$.

The acceptability is decided by the criteria

$\psi(\alpha_s) \le \lambda(\alpha_s)$ where $\lambda(\alpha) = \psi(0) + \rho\psi'(0)$ with $0 < \rho < 0.5$

and $\psi'(\alpha_s) \ge \beta\psi'(0)$ with $\rho < \beta < 1$

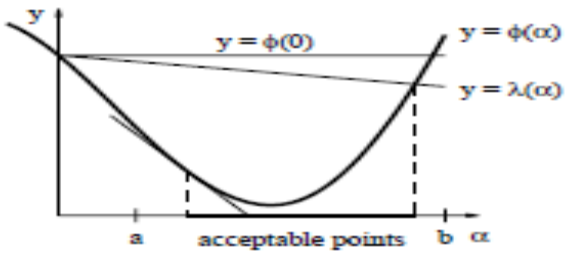
This criteria express the demands that α_s must be sufficiently small to give a useful decrease in the objective function and sufficiently large to ensure that the starting tangent of the curve $y = \psi(\alpha)$ for $\alpha \ge 0$

The algorithm has two parts:

Part 1: To find an interval [a,b] that contains acceptable points

Part 2: Successive reduction of the interval.

The graphical representation is as follows

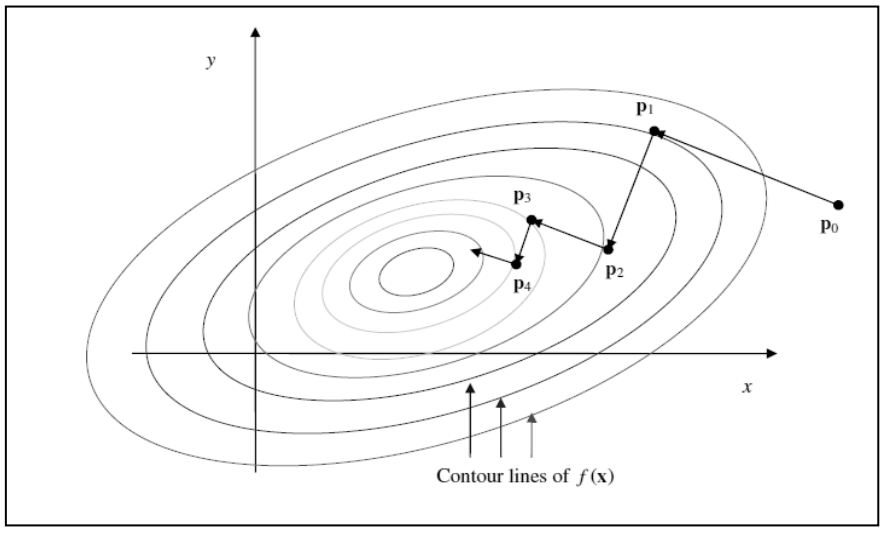


Steepest descent method:

In this method, for each iteration of line minimization the direction is chosen to be the local downhill gradient $-\nabla f(p)$. However, though along the downhill gradient to begin with at p, the vector n becomes perpendicular to the local gradient of f(x) where the current line minimum occurs.

Consequently, the vector n has to make a 90° turn for every iteration.

This results in a zigzag path along a "long valley" to the final minimum of f(x)



Newton's method:

It is widely used for solving systems of nonlinear equations and until recent it was also widely used for solving unconstrained optimization problem.

In order to derive *Newton's method* in the version used in optimization, truncated Taylor expansion is use that the current iterate x

$f(x+h) \approx q(h)$ where *q(**h**)* is the quadratic model of *f* in the vicinity of **x**,

$$q(h) = f(x) + h^T f'(x) + \frac{1}{2} h^T f''(x) h$$

The idea now is to minimize the model *q* at the current iterate. If $f''(x)$ is positive definite, then *q* has a unique minimiser at a point where the gradient of *q* equals zero, ie where $f'(x) + f''(x)h = 0$

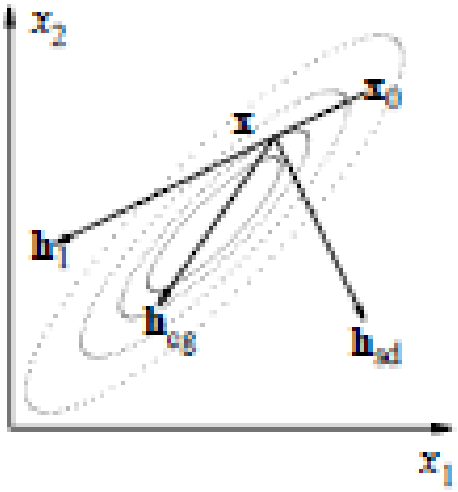
Hence in Newton's method the new iteration step is obtained as a solution to the system $f'(x) + f''(x)h = 0$

Newton's method is well defined as long as $f''(x)$ remains non-singular. Also, if the Hessian is positive definite, then h is downhill. Further, if $f''(x)$ stays positive definite in all the steps and if the starting point is sufficiently closed to a minimiser, the method usually converges rapidly towards such a solution.

Conjugate gradient method

The Conjugate gradient method represents major contribution to panoply of methods for solving large scale optimization problems. They are characterised by

- Low memory requirements .
- Strong local and global convergence properties.



The popularity of these method is remarkable partially due to their simplicity both in the algebraic expression and their implementation in computer codes and partially due their efficiency in solving large scale unconstrained optimization problem.

The development of conjugate gradient method begins with research of Cornelius Lanczos and Magnus Hestens and others(Forsythe, Motzkin, Rosser, Stein) at the institute at Numerical Analysis (National Applied Mathematics Laboratories of United States National Bureau of Standards in Los Angeles),and with independent research of Eduard Stiefel.

The first paper on conjugate gradient method was presented in 1952 by Magnus Hestenes and Eduard Stiefel[45] .In that paper an algorithm for solving symmetric, positive definite linear algebraic system has been presented**.** After a relatively sort period of stagnation the paper by Reid got the conjugate gradient method as a main active area of research in unconstrained optimization, In 1964 the method has been extended by Fletcher and Reeves[44] , which is usually considered as the first nonlinear Conjugate Gradient algorithm. Since then a large number of variants of Conjugate Gradient algorithms have been suggested. Even if the Conjugate Gradient methods are now 50 years old, they continue to be a considerable interest particularly due to their convergence properties, a very easy implementation effort in computer programme and due to their efficiency in solving large scale problems.

In this survey, we focus on conjugate gradient methods applied to the linear unconstrained optimization problem

$$\min\{ f(x) : x \in R^n \} \tag{1.1}$$

Where $f : R^n \to R$ is a continuously differentiable function especially if the dimension n is large.

They are of the form $x_{k+1} = x_k + \alpha_k d_k$ (1.2)

Where α_k is a step size obtained by a line search and d_k is the search direction botanised by

$$d_k = \begin{cases} -g_k, k = 1 \\ -g_k + \beta_k d_{k-1}, k \geq 2 \end{cases} \tag{1.3}$$

Where β_k is a parameter and g_k denotes $\nabla f(x_k)$ where the gradient $\nabla f(x_k)$ of f at x_k is a row vector and g_k is a column vector .Different C.G methods correspond to different choices for the scalar β_k .

It is known from (1.2) and (1.3) that only the step size α_k and the parameter β_k remain to be determined in the definition of Conjugate Gradient method. In this case that if f is a convex quadratic, the choice of β_k should be such that the method (1.2)-(1.3) reduces to the linear Conjugate Gradient method if the line search is exact namely

$$\alpha_k = \arg\min\{ f(x_k + \alpha d_k); \alpha > 0\} \tag{1.4}$$

For non linear functions, different formulae for the parameter β_k result in different Conjugate Gradient methods and their properties can be significantly different. To differentiate the linear Conjugate Gradient method, sometimes we call the Conjugate Gradient method for unconstrained optimization by nonlinear Conjugate Gradient method. Meanwhile the parameter β_k is called Conjugate Gradient parameter. The equivalence of the linear system to the minimization problem of $\frac{1}{2} x^T Ax - b^T x$ Motivated Fletcher and Reeves to extend the linear Conjugate Gradient method for nonlinear optimization. This work of Fletcher and Reeves in 1964 not only opened the door of nonlinear C.G Field but greatly stimulated the study of nonlinear optimization. In general the nonlinear Conjugate Gradient method without restarts is only linearly convergent(See Crowder and Wolfe[54]) while n-step quadratic convergence rate can be established if the method is restarted along the negative gradient every n-step.(See Cohen [55] and McCormick and Ritter[56]).

In 1964 the method has been extended to nonlinear problems by Fletcher and Reeves [44], which is usually considered as the first nonlinear Conjugate Gradient algorithm. Since then a large number of variations of Conjugate Gradient algorithms have been suggested. A survey on their definition including 40 nonlinear Conjugate Gradient algorithms for unconstrained optimization is given by Andrei[57].Since the exact line search is usually expensive and impractical, the strong Wolfe line search is often consider the implementation of the nonlinear Conjugate Gradient methods .It aims to find a step size satisfying the strong Wolfe conditions.

$$f(x_k + \alpha_k d_k) - f(x_k) \le \rho \alpha_k g_k^T d_k \tag{1.5}$$

$$\left| g(x_k + \alpha_k d_k)^T \right| \le -\sigma g_k^T d_k \tag{1.6}$$

$$where \;\; 0 < \rho < \sigma < 1$$

The strong Wolfe line search is often regarded as a suitable extension of the exact line search since it reduces to the latter. If σ is equal to zero, in practical

computation a typical choice for σ that controls the inexactness of the line search is σ=0.1.On the other hand general non linear function ,one may be satisfy with a step size satisfying the standard wolf conditions , namely (1.5) and

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k \qquad (1.7)$$
$$where\ 0 < \rho < \sigma < 1.$$

As is well known the standard Wolf line search is normally used in the implementation of Quasi-Newton methods, another important class of methods for unconstrained optimization. The work of Dai and Yuan indicates that the use of standard Wolfe line search is possible in the nonlinear Conjugate Gradient field. A requirement for an optimization method to use the above line searches is that, the search direction d_k must have descent property namely

$$g_k^T d_k < 0 \qquad (1.8)$$

For Conjugate Gradient method, by multiplying (1.3) with g_k^T , we have

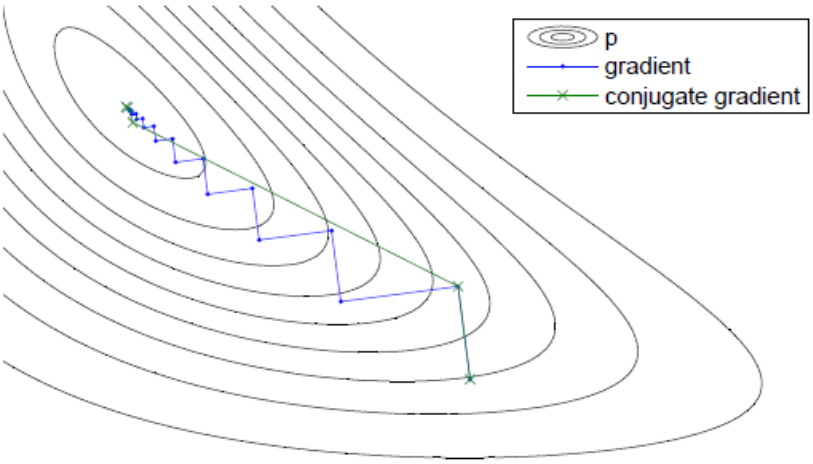
$$g_k^T d_k = -\| g_k \|^2 + \beta_k g_k^T d_{k-1}$$

Thus if the line search is exact, we have $g_k^T d_k = -\| g_k \|^2$ since $g_k^T d_{k-1} = 0$ .Consequently d_k is descent provided $g_k \neq 0$.In this paper we say that a Conjugate Gradient method is descent if (1.8) holds for all k and is sufficient descent if the sufficient descent condition

$$g_k^T d_k \leq -c \| g_k \|^2$$

Holds for all k and some constant c > 0.However we have to point out that the borderlines between these Conjugate Gradient methods are not strict.

If $s_k = x_{k+1} - x_k$ and in the following $y_k = g_{k+1} - g_k$ .Different Conjugate Gradient algorithms corresponds to different choices for the parameter β_k .



Sequence of points obtained by Conjugate Gradient Method

The Conjugate Gradient method (CG) has always played a special role in nonlinear optimization. It is related to quasi-Newton methods in many interesting ways that are being investigated to this day with the purpose of designing faster algorithms for large optimization. The Conjugate Gradient method can also be modified to produce a class of algorithms called nonlinear Conjugate Gradient methods that possess unique properties among optimization methods. In addition the Conjugate Gradient method can be used as an iterative linear solver in implementations Newton and quasi-Newton methods, and by fully exploiting the subspace minimization properties CG, these implementations give rise to robust, economical and rapidly convergent optimization methods.

Let us begin with a brief historical account of the Conjugate Gradient method in nonlinear optimization. The story begins with Davidon's invention of Quasi-Newton (or variable metric)methods in the late 1950s.Unknowing the existence of the Conjugate Gradient method, Davidon proposed an algorithm for nonlinear optimization that possess a fast rate of convergence and finite termination on quadratic objective functions. A few years later Fletcher and Powell showed that the algorithm is equivalent to the Conjugate Gradient method when applied, with exact line searches , to convex quadratic functions; the algorithm thus came to be known as the Davidon-Fletcher-Powell(DFP) method. During the next ten years, several refinements and variations of the DFP method gave rise to the very effective quasi-Newton algorithms used today with great, success in a great

variety of areas of application. The very popular BFGS method is a direct descendent of the DFP method.

Almost immediately after the publication of the DFP method, Fletcher and Reeves [44] proposed another algorithm for nonlinear optimization that appeared to be even more closely related to the Conjugate Gradient method. Unlike quasi-Newton methods, the algorithm of

Fletcher and Reeves does not require matrix storage and is very similar in form to the Conjugate Gradient method. It was the first non linear Conjugate Gradient method and subsequent research showed that a simple variation due to Polak and Ribiere gives good practical performance. Nonlinear Conjugate Gradient methods are designed so as to be equivalent to the (linear) Conjugate Gradient method .

The DFP and Fetcher Rletcher-Reeves methods marked the beginning of a new era in nonlinear optimization and much of the research performed during the last 30 years is directly related to these two seminal algorithms. It has been shown [68] ,[69],[70],[71],[72], that quasi Newton and nonlinear Conjugate Gradient methods can be related in various ways, most notably by introducing adaptive preconditioning techniques in the nonlinear Conjugate Gradient methods. Research Performed during the 1980s showed that there is a class of algorithms that fills the gap between quasi-Newton and Conjugate Gradient methods and considerable effort was devoted to finding an algorithm with the right balance between these two approaches. It turned that the two most successful algorithms for large scale optimization that emerged in the 1980s-quasi-Newton methods for partially separable optimization and limited memory methods-lie completely in the domain of quasi-Newton methods, Thus the pendulum has swung towards the quasi-Newton approach, and at present, nonlinear Conjugate Gradient methods do not play a dominant role in numerical optimization.

Nevertheless the linear Conjugate Gradient method continues to gain importance in the implementation of Newton-type methods for both constrained and unconstrained optimization. In addition, the interplay between the linear Conjugate Gradient method and nonlinear optimization algorithms is currently

being explored with the goal of designing more robust and cost-effective algorithms for large scale optimization.

An iterative method for minimizing a real function f on R^n can be described by sequence of moves from an initial point x^0 to a new point x^1 and so on, where the successive points are given by the relation

$$x^k = x^{k-1} + \alpha_k z^k \tag{1..9}$$

Where x^{k-1} is the current point , z^k is the direction vector along which we move and α_k is the step length. Suppose that the direction z^k is given and α_k is chosen so that the function f is minimized along z^k .Let

$$F(\alpha_k) = f(x^{k-1} + \alpha_k z^k) \tag{1.10}$$

and at α_k^* ,the minimum of F ,we have

$$\frac{dF(\alpha_k^*)}{d\alpha_k} = (z^k)^T \nabla f(x^{k-1} + \alpha_k^* z^k) = (z^k)^T \nabla f(x^k) = 0 \tag{1.11}$$

Assume that f is quadratic function, given as before by

$$f(x) = a + b^T x + \frac{1}{2} x^T Q x \tag{1.12}$$

Where Q Is an $n \times n$ symmetric positive definite matrix. In this case the gradients of f at any two points are related by

$$\nabla f(x^k) = \nabla f(x^{k-1}) + Q(x^k - x^{k-1}) \tag{1.13}$$

If $x^k = x^{k-1} + \alpha_k^* z^k$,then from (1.9),(1.11)and (1.13),we obtain an explicit formula for α_k^*:

$$\alpha_k^* = -\frac{(z^k)^T \nabla f(x^{k-1})}{(z^k)^T Q z^k} \tag{1.14}$$

The relation between the function values at two points is given by

$$f(x^k) = f(x^{k-1}) + (x^k - x^{k-1})^T \nabla f(x^{k-1}) + \frac{1}{2}(x^k - x^{k-1})^T Q (x^k - x^{k-1}) \tag{1.15}$$

and by (1.9) to (1.15),

$$f(x^{k-1}) - f(x^k) = \frac{[(z^k)^T \nabla f(x_{k-1})]^2}{2(z^k)^T Q z^k} \tag{1.16}$$

Since Q is assumed to be positive definite, the right hand side of this equation is nonnegative for $z^k \neq 0$ and is positive if z^k is not orthogonal to $\nabla f(x^{k-1})$.In the latter case, the algorithm is called a descent method, since $f(x^{k-1}) > f(x^k)$ .We only require $(z^k)^T \nabla f(x^{k-1}) \neq 0$ so it follows from (1.14) that if $(z^k)^T \nabla f(x^{k-1}) > 0$,then $\alpha_k^* < 0$ (In our discussion of the steepest descent method we knew that the preceding scalar product is negative ,which implies $\alpha_k^* > 0$).

In addition to having a descent minimization method, we would also like to have an algorithm that converges rapidly or, even better, that terminates in a finite number of steps when applied to minimizing a quadratic function .Since general nonlinear function can be reasonably well approximated by a quadratic function in the neighbourhood of a minimum ,the quadratic termination property seems desirable for fast convergence in the case of general functions .It follows that if the search direction z^k are mutually conjugate with respect to Q for $k = 1, 2, ..., n$,then the point x^n attained will be the exact minimum of the quadratic function f .The choice of the conjugate directions can be done in the following way.

Suppose that we start at a point $x^0 \in R^n$ and choose

$$z^1 = -\nabla f(x^0)$$

The next point is

$$x^1 = x^0 + \alpha_1^* z^1$$

Where α_1^* is given by (1.14).Evaluate $\nabla f(x^1)$ and set

$$z^2 = -\nabla f(x^1) + \beta_{11} z_1$$

Where β_{11} is a number chosen so that z^1 and z^2 will be conjugate with respect to Q. Hence

$$(z^1)^T Q z^2 = (z^1)^T Q(-\nabla f(x^1) + \beta_{11} z^1) = 0$$

And when move from x^1 along the direction z^2 to a new point x^2 then compute $\nabla f(x^2)$.The new direction z^3 (provided that $n \geq 3$)should be conjugate both z^1 and z^2.

Considering $z^3 = -\nabla f(x^2) + \beta_{21} z^1 + \beta_{22} z^2$

Where β_{21} *and* β_{22} are chosen so that $(z^1)^T Q z^3 = (z^2)^T Q z^3 = 0$.

In general it is obtained as

$$z^{k+1} = -\nabla f(x^k) + \sum \beta_{kj} z^j, \quad k = 0,1,....., n-1 \qquad (1.17)$$

The difficulty with this formula is that the coefficients β_{kj} are functions of Q ,and in trying to use (1.17) for a non-quadratic function, it is required to compute Hessian matrices, an undesirable operation. These directions can be generated without the explicit use of Q.