Computer-Aided Engineering

Adapted from

C.N.Nightingle and J.K.Fidler ., Computer-Aided Circuit and System Design.,U.K +IEEE papers

Lecture 4

Prof .Dr.Eng.ARN

Basic Search Strategy

In the optimization methods, a basic step is to find the minimum of n+1-dimensional error space starting at initial point x_0. However, most of the optimization methods are logically divided into two parts:

(a) The first part has the task of selecting a direction $\underline{d}$, where $\underline{d}$ is the direction of search; while

(b) the second part searches along the direction d to find the minimum of the function along $\underline{d}$.

Basic Search Strategy

Consider Figure 3.4 in which x_0 represents the vector corresponding to the initial guess of x_0 and x_1 and the vector $\underline{d}$ represents the direction between x_0 and the new point $\underline{x}_1$. From the figure, it is seen that the vector $\underline{x}_1$ is a vector sum of the other two components, i.e.

$$\underline{x}_1 = \underline{x}_0 + \underline{d} \qquad (3.17)$$

Basic Search Strategy

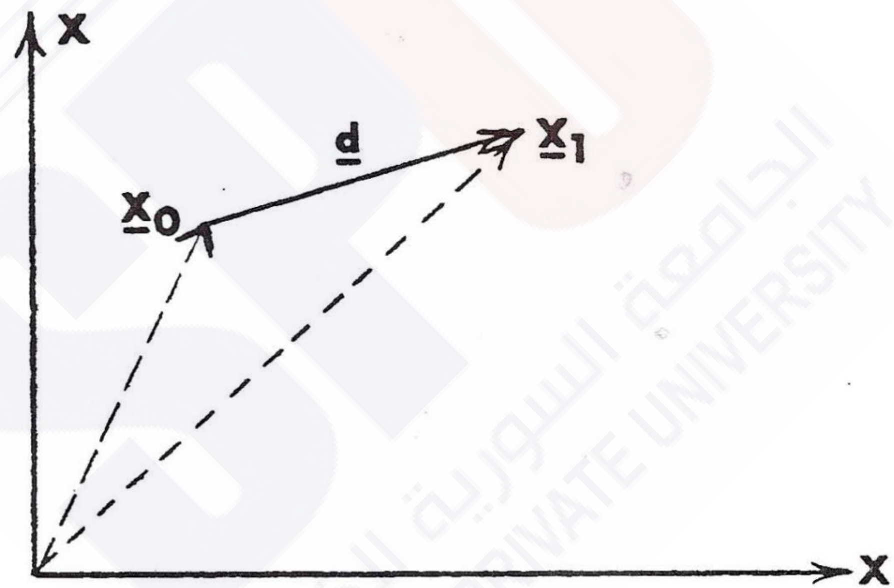

Figure 3.4 Linear search in a space of 2-independent variables

Basic Search Strategy

It is possible to vary the distance moved in the direction $\underline{d}$ by adding a scalar α, so that

$$\underline{x}_1 = \underline{x}_0 + \alpha\underline{d} \qquad (3.18)$$

A fundamental iterative expression for generalized search procedure is:-

$$\underline{x}_{i+1} = \underline{x}_i + \alpha_i\underline{d}_i \qquad (3.19)$$

Basic Search Strategy

where x_i is the starting or current point, and $\underline{d}_i$ is an n-dimensional search direction vector, and α_i is a positive scalar defining the distance moved from x_i to x_{i+1} in the direction of $\underline{d}_i$.

Multi-variable unconstrained non-linear function minimization algorithms based on the iterative nature of equation (3.19) fall natura into two classes [60-62]:

a) Direct search methods; and

b) Gradient search methods.

Basic Search Strategy

The direct search methods are those which do not require explicit evaluation of any partial derivatives of the function. They attempt to reduce the value of the error function evaluation by use of tests near an estimate of the solution. The test will find a direction of search in which the minimum is expected to lie. Gradient optimization methods

Basic Search Strategy

are those which use values of the partial derivatives of the error function f with respect to the system variables, as well as values of f itself. In some cases, higher derivatives of the objective function are required to establish a suitable direction for a minimum. Both of the above mentioned methods are dependent on information gained from earlier iterations. Some optimization methods belonging to the above categories will be discussed next.

3.3 Direct Search Optimization Algorithms

The strategy of the direct search methods is based on the comparison of values of the objective function only; such methods make no use of any of the derivatives of the function. The progress towards the minimum is made by the examination of trial solutions involving comparison of each trial solution with the "best" obtained up to that time. This is repeated until no further improvement can be obtained. Some direct search methods have been proved extremely effective in practice, particularly in applications in which the objective function is non-differentiable or has discontinuous first derivatives. Many direct search methods have been reported in the literature. The main three classes can be grouped as follows:-

Basic Search Strategy

- Tabulation methods;
- Sequential methods; and
- Linear methods

Basic Search Strategy

In tabulation methods the minimum is assumed to lie within the region given by equation (3.4), i.e. the error function to be evaluated at some set of points, and the smallest function value found is taken as a minimum. Note that random search methods and generalized Fibonacci search methods can be regarded as tabulation methods [62].

Basic Search Strategy

In the sequential optimization methods, the error function is investigated by performing function evaluations at the nodes of some geometric configuration in the space of the independent variables. This type of method has developed from the technique of "evolutionary operation" originally devised by Box (1957). Examples of this type of method are "factorial designs", "the simplex" [ibid], and the modification of the simplex methods such as those of Nelder and Mead [63].

Linear methods make use of a set of direction vectors throughout the search in the error space. Explorations are made along these directions, and a new direction or a sequence of directions, of search is determined according to the result obtained.

Probably the simplest of the linear methods is the "alternating variable", "univariate", or "one-at-a-time" search method in which each independent variable is considered in turn and altered until a minimum of the function is located, the remaining (n-1) variables remaining fixed. Hence the search proceeds parallel to each co-ordinate direction in turn, changing direction when a minimum in the current direction of search is reached. As an example of linear direct search methods, it is appropriate to study the widely used method which was devised in 1954 by Hooke and Jeeves [64]. This method attempts to align a search direction with the principal axis of the error function using

Basic Search Strategy

two strategies:

(a) Exploratory move; and

(b) Pattern moves.

Exploratory move examines the local behaviour of the objective function from the base (or current) point in the error space, and consists of a series of n univariate searches leading to a new point with lower error than the previous one. With this direction of the minimization path established by the exploratory move, a pattern move utilizes the information yielded by the explorations by progressing in the direction of reducing error function values.

A pattern move consists of a single step from the present base point, that step having both magnitude and direction of the line joining the previous base point to the current one. Thus, if the pattern move is now made in the direction established by the exploratory move, to the point $\bar{x}$ where

$$\bar{x} = x^{k} + (x^{k} - x^{k-1}) = 2x^{k} - x^{k-1} \qquad (3.20)$$

It can be seen from the above equation that pattern move requires twice the distance between the previous and the new points, and establishes the starting point from which the next exploratory move should be made. The above procedure continues in the alternating manner until a set of exploratory moves about a current point all fail. This indicates that either the minimum has been located or the search has descended into a steep valley which cannot be negotiated using the current step sizes. Convergence is assumed when the step-length has been reduced below some specified limit.

Direct search methods are particularly expedient for cases when the first partial derivatives of the function are discontinuous or if the function is non-differentiable or when the function is subject to random error, all of which are problems which can cause difficulties using the more theoretically based gradient methods. Note that direct search methods are simple to program and they can have important considerations in some practical optimization problems, but they are inefficient in practice because of very slow convergence [59, 61].

Gradient methods, as the name implies, are those in which values of partial derivatives of the error function are used to select the direction of the greatest change of the function in the region of $\underline{x}$.

Gradient optimization algorithms can, for convenience, be divided into two classes, with regard to the order of the partial derivatives shown in equation (3.8).

(a) "First-order" methods, which neglect the second order partial derivatives of the Taylor series expansion, using only the Jacobian gradient vector g to find the minimum of the objective function $f(\underline{x})$, as well as the function value itself.

(b) "Second-order" methods, which require the evaluation of the Hessian matrix H, in addition to the requirements of the "first-order" methods.

Many gradient optimization methods have been reported

(a) Steepest descent methods

(b) Newton's method

(c) Quasi-Newton methods

(d) Least squares methods.

Optimization Methods:-

Linear Programming

Quadratic Programming

Mixed-Integer

Global Optimization

Genetic Algorithms

Simulation Methods:-

Risk Analysis

Simulation

Monte Cairo Methods

Simulation Optimization

Stochastic Programming

Using the above mentioned classification, it has to be said that steepest descent optimization methods are first-order methods, whereas Newton's methods are second-order methods : the quasi-Newton and least squares methods are categorized as first-order since second-order derivatives are not explicitly calculated, but their characteristics are essentially those of the second-order methods.

The essential difference between gradient optimization algorithms lies in the manner in which a vector $\underline{d}_i$ is determined and in the strategy adopted for the selection of α_i (see equation 3.19). Some of these methods, such as steepest descent methods, require that the direction of search should be kept fixed. While the others change the direction of search to improve the search efficiency.

Newton's Method

Newton's method is based on an approximation of a quadratic surface to the actual minimum. If the objective function is quadratic, then the approximation is exact. Otherwise, a new surface has been produced at this point, and so on [62].

Newton's method makes use of the Hessian matrix H of second order partial derivatives to determine the correction vector $\underline{\Delta x}$ approximately. Suppose $\underline{g}$ and H are fixed, partially differentiating (3.8) with respect to Δx_j and setting the result to zero, then:-

$$\frac{\partial f(\underline{x})}{\partial x_j} + \sum_{i=1}^{n} \sum_{j=1}^{n} \Delta \underline{x}_i \frac{\partial^2 f(\underline{x})}{\partial \underline{x}_i \partial \underline{x}_j} = 0 \qquad (3.24)$$

neglecting higher-order terms of the Taylor expansion series.

neglecting higher-order terms of the Taylor expansion series.

Equation (3.24) can be written in matrix form as:

$$\underline{g} = -H(\underline{x}) \cdot \Delta\underline{x} \quad (3.25)$$

or

$$\Delta\underline{x} = -H^{-1}(\underline{x}) \cdot \underline{g}(\underline{x}) \quad (3.26)$$

The minimum $\underline{x}^*$ then, can be estimated from equations (3.7) and (3.26) as:

$$\underline{x}^* = \underline{x} - H^{-1}(\underline{x}) \cdot \underline{g}(\underline{x}) \tag{3.27}$$

or in general, the i^{th} iteration is given by

$$\underline{x}_{i+1} = \underline{x}_i - H^{-1}(\underline{x})\underline{g}(\underline{x}) \tag{3.28}$$

This equation forms the basis of Newton's methods, where the gradient $\underline{g}$ is evaluated at $\underline{x}_i$. Note that the direction $-H^{-1}\underline{g}$ is sometimes referred to as the Newton-Raphson direction [67]. The step-length given by (3.26) does not give the minimum along the direction $-H^{-1}\underline{g}$ for a general non-quadratic function. Consequently, there is some modification on the iterative procedure to be as:

$$\underline{x}_{i+1} = \underline{x}_i - \alpha_i [H^{-1}(\underline{x}_i) \cdot g(\underline{x}_i)] \qquad (3.29)$$

The bracketted term in the above equation is regarded as a search direction. α_i will be positive if H is positive definite. If α_i has to be negative to reduce the error, then the direction $-H^{-1}\underline{g}$ is unhelpful.

Newton's method is very efficient in the neighbourhood of the minimum even on non-quadratic surfaces, but when the current point $\underline{x}$ is not close to the minimum $\underline{x}^*$ and the function is not quadratic the algorithm may diverge [61]. Moreover, the computational effort involved in determining the matrix of second derivatives H and its inverse H^{-1} is often considerable.