A NEW ARC ALGORITHM FOR UNCONSTRAINED OPTIMIZATION

ISRAEL ZANG

Faculty of Management, Tel Aviv University, Tel Aviv, Israel

Received 6 June 1977
Revised manuscript received 17 November 1977

The gradient path of a real valued differentiable function is given by the solution of a system of differential equations. For a quadratic function the above equations are linear, resulting in a closed form solution. A quasi-Newton type algorithm for minimizing an n-dimensional differentiable function is presented. Each stage of the algorithm consists of a search along an arc corresponding to some local quadratic approximation of the function being minimized. The algorithm uses a matrix approximating the Hessian in order to represent the arc. This matrix is updated each stage and is stored in its Cholesky product form. This simplifies the representation of the arc and the updating process. Quadratic termination properties of the algorithm are discussed as well as its global convergence for a general continuously differentiable function. Numerical experiments indicating the efficiency of the algorithm are presented.

Key words: Optimization, Non-linear Programming, Unconstrained Optimization, Gradient-path Algorithms, Quasi-Newton Methods, Arc Algorithms.

1. Introduction

Several new approaches to unconstrained optimization have been lately suggested. Generally, these approaches aim to obtain better information regarding the curvature of the function being optimized, or to use existing information in a better way. See for example [1], [2], [9], [17], [19], [28] and [29] where some more references are cited.

Recently, a new approach has been independently suggested by Vial and Zang [29] and by Botsaris and Jacobson [4] and Botsaris [3]. The basic idea underlying this approach is that a good algorithm for minimizing a differentiable function $f : \mathbb{R}^n \to \mathbb{R}$ should follow (starting at a point $x^0 \in \mathbb{R}^n$) the arc defined by the solution to the differential equation

$$\dot{x}(t) = -\nabla f(x(t)), \quad x(0) = x^0. \quad (1.1)$$

This arc is known as the gradient path of $f$ emerging at $x^0$. If $f$ is quadratic, an explicit expression of this arc is obtainable. The algorithms suggested by [29], [4] and [3], approximate $f$ at a given point by some quadratic function and follow the gradient path of this approximation in order to obtain a new point where $f$ is
re-approximated. Vial and Zang [29] have suggested a quasi-Newton type algorithm which maintains and updates approximate second order information. A different algorithm based upon approximate second order information was suggested by Botsaris [3]. However, the processes of updating this information are quite involved.

In Section 2 of this paper we present a new descent algorithm which uses a different arc obtained by some transformation in the space of variables and for which the updating process becomes a simple task. This updating is carried out according to any quasi-Newton updating scheme. In Section 3, we prove convergence of the new algorithm using an inexact search scheme along arcs. The efficiency of the algorithm is demonstrated in Section 4, where we present the results of some numerical experimentations.

2. The algorithm

2.1. Motivation and description

Let \( \phi : \mathbb{R}^n \to \mathbb{R} \) be the quadratic function

\[
\phi(x) = \frac{1}{2}x^TQx + x^Tb, \tag{2.1}
\]

where \( Q \in \mathbb{R}^{n \times n} \) is a symmetric positive definite matrix and \( b \in \mathbb{R}^n \). For this function, the gradient path emerging at \( x^0 \) (solution to (1.1)) is given by [29]:

\[
x(t) = x^0 + [e^{-Qt} - I]Q^{-1}\nabla \phi(x^0). \tag{2.2}
\]

Relying on the above expression, a quasi-Newton type algorithm for minimizing a general differentiable function \( f : \mathbb{R}^n \to \mathbb{R} \) was suggested by Vial and Zang [29]. At the beginning of the \( k \)-th iteration, this algorithm requires a point \( x^k \) which is the current approximation to the minimum sought, and a symmetric positive definite\(^1 \) matrix \( B_k \), approximating the Hessian of \( f \) at \( x^k \). In order to move to the next point \( x^{k+1} \), a search has to be carried out along the arc

\[
x^k(t) = x^k + [e^{-B_kt} - I]B_k^{-1}\nabla f(x^k), \tag{2.3}
\]

which is the gradient path of the current local quadratic approximation to \( f \). A value of \( t \) denoted \( t^k \) is determined to satisfy some stopping criteria, and \( x^{k+1} = x^k(t^k) \) is fixed. The \( k \)-th iteration ends once \( B_k \) is updated according to one of the well-known quasi-Newton updating formulae, as suggested in [29].

The initial direction (i.e., \( \frac{\partial}{\partial x} \phi(x^0) \)) of the arc (2.3) is always that of the steepest descent; therefore, the algorithm is a descent one. Also note that by letting \( t = \infty \)

\(^1\)This property is stronger than required in [29], but will be needed throughout this paper.