SOME RESULTS IN NONSMOOTH OPTIMIZATION*1)

YUAN YA-XIANG (袁亚湘)

(Computing Centre, Academia Sinica, Beijing, China; Department of Applied Mathematics and Theoretical Physics, University of Cambridge, Cambridge, England)

Abstract

Some main results in nondifferentiable optimization are reviewed. In Section 2, we discuss subgradient methods. Section 3 is about the cutting plane method and the bundle methods are studied in Section 4. Trust region methods for composite nonsmooth optimization are discussed in Section 5.

§ 1. Introduction

A general nonsmooth optimization is to seek a point that attains the smallest value of a nonsmooth function $f(x)$, where $f(x)$ defined on $\mathbb{R}^n$ is continuous, but not necessarily differentiable. In other words, we need to solve the problem

$$\min f(x), \quad x\in\mathbb{R}^n. \tag{1.1}$$

A necessary condition for x^* to be a solution of (1.1) is that the null vector is in the subdifferential of $f(x)$ at x^*. The definition of the subdifferential can be found in Clark (1975), and is expressed as (2.6) in the next section.

Methods for solving (1.1), to be discussed in the next four sections, are all iterative. That means, to start calculation, an initial guess for the solution has to be made. Then at every iteration, a method would give a search direction or a trial step. In the first case, some kind of line search techniques are needed to choose a step-size $\alpha_k>0$ in order to move the approximate point from the original location (x_k, say) to a new one ($x_{k+1}=x_k+\alpha_k d_k$), where d_k is the search direction. In the second case, some kind of tests are needed to judge whether we should accept the trial step ($x_{k+1}=x_k+d_k$) or take a null step ($x_{k+1}=x_k$).

In the next section, we present some results of subgradient methods for (1.1). Most of the research in this area has been done by the Soviet scientists. In Section 3, we give a brief introduction to the cutting plane method. Section 4 is about the bundle methods, with an introduction of the conjugate subgradient method. In Section 5, we consider a class of trust region methods for the so-called composite optimization problem.

There are also many other methods that will not be discussed here. The readers are referred to Blinski and Wolfe (1975), Lemarechal and Mifflin (1978) and Nurminskii (1982).

* Received January 13, 1986.

1) Presented at the 50th Anniversary Conference of the Chinese Mathematical Society, December 1985, Shanghai.

§ 2. Subgradient Methods

For the moment, assume that the function $f(x)$ is continuously differentiable. The steepest descent method for solving (1.1) sets

$$x_{k+1}=x_k-\alpha_k\nabla f(x_k), \tag{2.1}$$

where $\alpha_k>0$ is a step-length. There are different techniques for choosing α_k, among which are the "exact line search" and the "Armijo type search". The former requires that

$$f(x_k-\alpha_k\nabla f(x_k))=\min_{t>0} f(x_k-t\nabla f(x_k)), \tag{2.2}$$

and the latter one chooses such α_k that satisfy the inequality

$$f(x_k-\alpha_k\nabla f(x_k))-f(x_k)\leqslant -c_1\alpha_k\|\nabla f(x_k)\|^2, \tag{2.3}$$

where c_1 is a parameter in $(0, 1)$. For either search technique (2.2) or (2.3), it can be proved that any accumulation point of $\{x_k\}$ is a stationary point of (1.1), that is, the gradient of $f(x)$ is the null vector.

The subgradient method is a generalization of the steepest descent method (2.1). At every iteration, it lets

$$x_{k+1}=x_k-\alpha_k g_k, \tag{2.4}$$

where g_k is any subgradient of $f(x)$ at the point x_k. That is, we have

$$g_k\in\partial f(x_k), \tag{2.5}$$

where $\partial f(x_k)$ is subdifferential of $f(x)$ at x_k. The subdifferential $\partial f(x)$ of a function is defined by

$$\partial f(x)=\operatorname{conv}\{g\in\mathbb{R}^n \mid g=\lim_{i\to\infty}\nabla f(x_i),\ x_i\to x,\quad \nabla f(x_i) \text{ exist}, \nabla f(x_i) \text{ converge}\}. \tag{2.6}$$

For more details, see Clark (1975).

Hence a class of subgradient methods for solving (1.1) can be described below:

Algorithm **2.1** (The Subgradient Method).

Step 0: Given initial vector x_1.

Step 1: Calculate $f(x_k)$, and obtain a vector $g_k\in\partial f(x_k)$.

Step 2: Choose a step-size $\alpha_k>0$.

Step 3: Set

$$x_{k+1}=x_k-\alpha_k g_k. \tag{2.7}$$

Set $k=k+1$ and go to Step 1.

The difficulty in choosing α_k in Algorithm 2.1 is that we can not use the exact line search or the Armijo type line search.

For the exact line search, take the problem of minimizing the 1-norm of the variable in $\mathbb{R}^2$ for example, that is, to solve (1.1) with

$$f(x)=\|x\|_1,\quad x\in\mathbb{R}^2. \tag{2.8}$$

Suppose we let the initial vector be $x_1=(t_1\quad 0)^T$, where t_1 is a positive constant. For any positive constant t_2 in $(0, 1)$, we can choose $g_1=(1\quad -t_2)^T$. Thus we have

$$x_2^T=[0\quad t_1t_2]. \tag{2.9}$$