Journal of Computational Mathematics, Vol.23, No.2, 2005, 153–164.

SINGLY DIAGONALLY IMPLICIT RUNGE-KUTTA METHODS COMBINING LINE SEARCH TECHNIQUES FOR UNCONSTRAINED OPTIMIZATION *1)

Xin-long Luo
(*School of Information Engineering, Beijing University of Posts and Telecommunications, P.O. Box 101,Beijing 100876, China*)

Abstract

There exists a strong connection between numerical methods for the integration of ordinary differential equations and optimization problems. In this paper, we try to discover further their links. And we transform unconstrained problems to the equivalent ordinary differential equations and construct the LRKOPT method to solve them by combining the second order singly diagonally implicit Runge-Kutta formulas and line search techniques. Moreover we analyze the global convergence and the local convergence of the LRKOPT method. Promising numerical results are also reported.

Mathematics subject classification: 65K05, 90C30, 65L06.
Key words: Global convergence, Superlinear convergence, Runge-Kutta method, Unconstrained optimization.

1. Introduction

In this paper, we mainly consider numerical methods for the following unconstrained optimization problem

$$\min_{x \in R^n} f(x), \tag{1.1}$$

where f is a continuously differentiable function. The main idea of solving the unconstrained optimization problem (1.1) is that we search for the next iteration point

$$x_{k+1} = x_k + \alpha_k d_k$$

via choosing the descent direction d_k and the step length α_k based on the current iteration point x_k such that $f(x_{k+1})$ satisfies some descent criteria, such as the Armijo line search criterion [13, 27, 35].

It has been extensively studied for choosing the descent direction d_k based on the Newton direction (see [1, 2, 4, 6, 9]), the conjugate gradient direction (see [10, 14, 15]) and the negative gradient direction (see [3, 5, 6, 16, 18, 30]) last decades, where $\nabla f(x_k)$ and $\nabla^2 f(x_k)$ are the gradient and the Hessian matrix of the function f at the current point x_k, respectively. But there are few researches for other descent directions. In the next section, we will consider search directions other than the negative gradient direction or the Newton direction. And we construct the LRKOPT method that has the superlinear convergence and global convergence by discretizing the following initial value problem of ordinary differential equations

$$\frac{dx}{dt} = -\nabla f(x), \tag{1.2}$$

$$x(0) = x_0, \tag{1.3}$$

* Received June 10, 2003, final revised June 23, 2004.
1) This work is partially supported by Chinese National Science Foundation (No.19731010).

where x_0 is an any given initial value. It is well-known that the solution $x(t)$ of differential equations (1.2)-(1.3) converges to the stable point $x^\star$ of the function f as t tends to infinity, namely $\lim_{t\to\infty} \|\nabla f(x(t))\| = 0$ (see [18, 22]).

There are some discussions on numerical methods for solving (1.2)-(1.3) in [30], which point out the importance of studying this class of numerical methods. Schropp [32] applied linear multistep methods to the gradient system (1.2)-(1.3) and studied the qualitative properties of discrete solutions of (1.2)-(1.3). In [23], we give a reasonable explanation that Backward Difference Formulas (BDFs) which are popular methods for solving stiff ordinary differential equations are low efficient for the gradient system (1.2)-(1.3) on the view of unconstrained optimization and we will restate the explanation in Section 2. Thus we mainly consider Runge-Kutta methods for solving the gradient system (1.2)-(1.3).

This paper is organized as follows. In the next section we consider the second order Singly Diagonally Implicit Runge-Kutta methods (SDIRK) for solving the gradient system (1.2)-(1.3) and construct the LRKOPT method with the superlinear convergence for the unconstrained optimization problem (1.1). In Section 3 we analyze the global convergence and the local convergence of the LRKOPT method. Finally, we report some numerical results of the LRKOPT method and the IMPBOT method which is given by Brown and Bartholomew-Biggs (see [6]) in Section 4. Throughout the paper $\|\cdot\|$ denotes the Euclidean vector norm or its induced norm.

2. The LRKOPT Method

We know that the class of methods for solving the gradient system (1.2)-(1.3) need satisfy the L stability via studying the linear test ordinary differential equation if those methods have the good local behavior (see [23]). Because linear multistep methods except for the backward Euler method do not satisfy the L stability, we focus on Runge-Kutta methods for solving the gradient system (1.2)-(1.3).

Runge-Kutta methods for solving the gradient system (1.2)-(1.3) have the following general form

$$K_i = h \cdot g(x_k + \sum_{j=1}^{s} a_{ij} K_j), \ i = 1, 2, \cdots, s, \tag{2.1}$$

$$x_{k+1} = x_k + \sum_{i=1}^{s} b_i K_i, \tag{2.2}$$

where $g(x) = -\nabla f(x)$, $h > 0$ is the time step, a_{ij} and b_i are constants. It is favorable for stiff ordinary differential equations if the numerical method has the A stability. Because the highly nonlinear problem (1.1) can introduce stiff ordinary differential equations (1.2)-(1.3). Thus we consider implicit Runge-Kutta methods for solving the gradient system (1.2)-(1.3).

Before introducing the particular scheme for solving the gradient system (1.2)-(1.3) we give some short descriptions of A-stable, L-stable and B-stable. A numerical method is called A-stable if, for the linear test equation $dx/dt = \mu x$ with $Re(\mu) \le 0$ and for all time steps $h \ge 0$, the stability function $R(z) = 1 + zb^T(I - zA)^{-1}e$ satisfies $|R(z)| \le 1$, where $z = \mu h$, the elements of the matrix A are $a_{ij}(i, j = 1, 2, \cdots, s)$, the vector b equals to $[b_1, b_2, \cdots, b_s]^T$ and all elements of e are one (see [17, 34]). The step length h does not have the stable restriction if the numerical method is A-stable. Furthermore the numerical method is A-stable and satisfies $\lim_{z\to-\infty} R(z) = 0$ then it is called L-stable (see [17, 34]).

Let two sequences $\{x_k\}$ and $\{z_k\}$ of approximation computed by a Runge-Kutta method for the same following autonomous differential equations

$$\frac{dx}{dt} = g(x), \quad g : R^n \to R^n. \tag{2.3}$$