Journal of Computational Mathematics, Vol.19, No.6, 2001, 639–650.

PARALLEL COMPOUND METHODS FOR SOLVING PARTITIONED STIFF SYSTEMS*

Li-rong Chen   De-gui Liu
(*Beijing Institute of Computer Application and Simulation Technology, Beijing 100854, China*)

Abstract

This paper deals with the solution of partitioned systems of nonlinear stiff differential equations. Given a differential system, the user may specify some equations to be stiff and others to be nonstiff. For the numerical solution of such a system Parallel Compound Methods(PCMs) are studied. Nonstiff equations are integrated by a parallel explicit RK method while a parallel Rosenbrock method is used for the stiff part of the system.

Their order conditions, their convergence and their numerical stability are discussed, and the numerical tests are conducted on a personal computer and a parallel computer.

Key words: Parallel compound methods, Stiff systems, Order conditions, Convergence, Stability.

1. Introduction

Many stiff systems occurring in practice have a special structure, and they can be split into two coupled subsystems

$$\begin{aligned} y'_S(x) &= f_S(x, y_S(x), y_N(x)), \quad y_S(x_0) = y_{S0}, \quad y_S \in R^{n_S} \\ y'_N(x) &= f_N(x, y_S(x), y_N(x)), \quad y_N(x_0) = y_{N0}, \quad y_S \in R^{n_N} \end{aligned} \tag{1}$$

where y_S denotes the vector of stiff components and y_N denotes the vector of nonstiff components. For such partitioned systems a partitioned discretization method is used, i.e. the stiff subsystem is solved by a "stiff" method and the nonstiff by a classical method (see [5–8]). This paper deals with a class of parallel compound methods. The compound method consists of a parallel explicit Runge-Kutta method [2] for the solution of the nonstiff subsystem and a parallel Rosenbrock method [3] for the solution of the stiff subsystem. The internal stages of RK and Rosenbrock methods can be computed in parallel.

The paper discusses order conditions, convergence and numerical stability as well as the implementation and usage of such compound methods. Test results for three partitioned stiff initial value problems are given with respect to speedup and efficiency.

* Received August 16, 1998.

2. Parallel Compound Methods

For simplicity, consider the autonomous partitioned stiff systems

$$\begin{aligned} y_S'(x) &= f_S(y_S(x), y_N(x)), \quad y_S(x_0) = y_{S0}, \quad y_S \in R^{n_S} \\ y_N'(x) &= f_N(y_S(x), y_N(x)), \quad y_N(x_0) = y_{N0}, \quad y_S \in R^{n_N} \end{aligned} \tag{2}$$

An s-stage Parallel Compound Method(PCM) is defined by:

$$\begin{aligned} y_{Sn+1} &= y_{Sn} + \sum_{i=1}^{s} c_i l_{in} \\ y_{Nn+1} &= y_{Nn} + \sum_{i=1}^{s} c_i k_{in} \\ k_{in} &= h f_N(y_{Sn} + \sum_{j=1}^{i-1} \alpha_{ij} l_{jn-1},\ y_{Nn} + \sum_{j=1}^{i-1} \alpha_{ij} k_{jn-1}) \\ (I - h\gamma J) l_{in} &= h f_S(y_{Sn} + \sum_{j=1}^{i-1} \alpha_{ij} l_{jn-1},\ y_{Nn} + \sum_{j=1}^{i-1} \alpha_{ij} k_{jn-1}) + hJ \sum_{j=1}^{i-1} \gamma_{ij} l_{jn-1} \\ & i = 1, 2, \ldots, s \end{aligned} \tag{3}$$

where $\gamma,\ \alpha_{ij},\ \gamma_{ij},\ c_i$ are the real coefficients, I denotes $n_S \times n_S$ identity matrix, $J = \dfrac{\partial f_S}{\partial y_S}(y_{Sn}, y_{Nn})$. The method (3) can be briefly characterized:

- The nonstiff components y_N are computed explicitly, the stiff components y_S semi-implicitly. At each integration step a system of linear equations of $n_S \leq n_S + n_N$ must be solved,
- Through a frontal approach the internal stages of RK and Rosenbrock methods k_{in} and $l_{in} (i = 1, 2, \ldots, s)$ can be computed in parallel on $2s$ processors.

The following abbreviations are used:

$$\begin{aligned} & \alpha_{ij} = 0 \qquad j \geq i, \qquad \gamma_{ij} = 0 \qquad j > i \\ & \beta'_{ij} = \alpha_{ij} + \gamma_{ij}, \qquad \gamma_{ii} = \gamma \\ & \beta_{ij} = \begin{cases} \beta'_{ij}, & i > j \\ 0, & i \leq j \end{cases} \\ & \alpha_i = \sum_{j=1}^{s} \alpha_{ij}, \quad \beta'_i = \sum_{j=1}^{s} \beta'_{ij}, \quad \beta_i = \sum_{j=1}^{s} \beta_{ij} \end{aligned}$$

As the quantities $y_n,\ k_{in-1},\ l_{in-1}\ (i = 1, 2, \ldots, s-1)$ are known, $k_{in},\ l_{in}\ (i = 1, 2, \ldots, s)$ can be evaluated on $2s$ processors in parallel, and more y_{n+1} obtained. The information flow that describes the parallel execution of two-stage formula on four processors is showed in Fig. 1.