Journal of Computational Mathematics, Vol.18, No.2, 2000, 157–164.

PARALLEL MULTI-STAGE & MULTI-STEP METHOD IN ODES*

Xiao-qiu Song
(*Institute of Computer Application and Simulation Technology P.O.Box 3929, Beijing 100854, China*)

Abstract

In this paper, the theory of parallel multi-stage & multi-step method is discussed, which is a form of combining Runge-Kutta method with linear multi-step method that can be used for parallel computation.

Key words: Ordinary differential equations, Parallel simulation.

1. Introduction

Early in 1966, W.L.Miranker and W.Liniger [3] gave a kind of parallel Runge-Kutta method of order 2 and order 3. But unfortunately, when the method given in [3] was applied to test equation $y' = \lambda y$, $\mathrm{Re}\lambda < 0$, it's numerically unstable. In that time, W.L.Miranker and W.Liniger then expressed their expectation for a parallel Runge-Kutta method with numerical stability. At the beginning of 1990's, some kinds of parallel Runge-Kutta method were considered in [1] and [2] with some modification of those formulae given in [3], and their absolute stability regions were discussed. Recently, the theory of combination method is set up, and some formulae of this method are given in [4], which combines Runge-Kutta method with linear multi-step method. This combination method can be parallel computed just like the parallel Runge-Kutta method. In this paper, the theory of parallel multi-stage & multi-step method is discussed, which is another form of combining Runge-Kutta method with linear multi-step method.

Some algorithms based on this method are part of PASL (Parallel Algorithm Software Library) on S10 parallel computer (MIMD) which was made by Aero-Space Department of China in 1991.

2. Basic Theory

Consider the system of differential equations

$$y' = f(y), \quad y(x_0) = y_0, \quad f : \mathbf{R}^m \to \mathbf{R}^m \tag{2.1}$$

* Received August 13, 1996.

parallel r-stage & r-step method is constructed with the following form

$$y_{n+1} = BY_n + hCU_nD \tag{2.2}$$

where

$$B = (b_1, b_2, ..., b_r), \qquad C = (1, c_2, ..., c_r)$$

$$D = (d_1, d_2, ..., d_r)^T, \qquad Y_n = (y_n, y_{n-1}, ..., y_{n-r+1})^T$$

$$U_n = \begin{pmatrix} K_{1,n} & K_{1,n-1} & \cdots & K_{1,n-r+1} \\ 0 & K_{2,n-1} & \cdots & K_{2,n-r+1} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & K_{r,n-r+1} \end{pmatrix}$$

$$K_{1,n} = f(y_n), K_{1,i} = f(y_i), K_{j,i} = f(\sum_{l=1}^{j} w_{j,l} y_{i+l-1} + h(\sum_{l=1}^{j-1} \beta_{j,l} K_{l,i})), \sum_{l=1}^{j} w_{j,l} = 1$$

$$j = 2, 3, ..., n-i+1, \quad i = n-1, n-2, ..., n-r+1.$$

We can find out the method (2.2) can be executed in parallel by r-processes, that $K_{1,n}$, $K_{2,n-1}, ..., K_{r,n-r+1}$ can be computed synchronouslly.

Theorem 2.1. *If the following conditions are satisfied*

(i) *the module of roots of*

$$\lambda^r - B(\lambda^{r-1}, \lambda^{r-2}, ..., \lambda^0)^T = 0, \quad \lambda \epsilon \mathbf{C}$$

are no more than 1 and the roots with module 1 are single

(ii) *for* $y_i = y(x_i), \quad i = n, n-1, ..., n-r+1$, *have*

$$y(x_{n+1}) - BY_n - hCU_nD = O(h^{m+1})$$

then (2.2) is convergent with order m.

Proof. Let

$$\bar{B} = \begin{pmatrix} b_1 & b_2 & \cdots & b_r \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 \end{pmatrix}, \bar{C} = \begin{pmatrix} 1 & c_2 & \cdots & c_r \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}$$

so we have

$$Y_{n+1} = \bar{B}Y_n + h\bar{C}U_nD$$

$$Y(x_{n+1}) = \bar{B}Y(x_n) + h\bar{C}U(x_n)D + O(h^{m+1})$$

Let

$$e_n = y(x_n) - y_n$$

$$E_n = (e_n, e_{n-1}, \cdots, e_{n-r+1})^T$$

so we have

$$E_{n+1} = \bar{B}E_n + h\bar{C}(U(x_n) - U_n)D + O(h^{m+1})$$

Because the eigenvalues of $\bar{B}$ are equivalent to the roots of

$$\lambda^r - B(\lambda^{r-1}, \lambda^{r-2}, \cdots, \lambda^0)^T = 0$$

so the module of eigenvalues of $\bar{B}$ are no more than 1 and the eigenvalues with module 1 are single , therefore we can find a normal $\| \bullet \|$ with $\|\bar{B}\| = 1$[6]. For this normal, with $nh = x_T - x_0$, we have

$$\begin{aligned}
\|E_{n+1}\| &\leq \|\bar{B}\|\|E_n\| + h\|\bar{C}\|\|U(x_n) - U_n\|\|D\| + O(h^{m+1}) \\
&\leq \|E_n\| + h\|\bar{C}\|\mathrm{L}\|E_n\|\|D\| + O(h^{m+1}) \\
&= (1 + h\mathrm{L}\|\bar{C}\|\|D\|)\|E_n\| + O(h^{m+1}) \\
&\cdots \\
&\leq (1 + h\mathrm{L}\|\bar{C}\|\|D\|)^{n+1}\|E_0\| + (\sum_{i=0}^{n}(1 + h\mathrm{L}\|\bar{C}\|\|D\|)^i)O(h^{m+1}) \\
&\leq K_1\|E_0\| + K_2 nO(h^{m+1}) \\
&= K_1\|E_0\| + K_2 O(h^m)
\end{aligned}$$

So the method (2.2) is order m.

Theorem 2.2. *The region of absolute stability of (2.2) is*

$$\Omega = \{\bar{h}\epsilon\mathbf{C} \mid \rho(\mathbf{G}(\bar{\mathbf{h}})) < \mathbf{1}\}$$

where

$$G = \bar{B} + \bar{h}\bar{C}\Phi$$

$$\bar{B} = \begin{pmatrix} b_1 & b_2 & \cdots & b_r \\ 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & \cdots & 1 & 0 \end{pmatrix}, \bar{C} = \begin{pmatrix} 1 & c_2 & \cdots & c_r \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 0 \end{pmatrix}$$

$$K_{j,i} = (a_{j,i,1}, a_{j,i,2}, ..., a_{j,i,r})Y_n \quad \textit{with} \quad f(x,y) = \lambda y$$

$$\phi_{ij} = \sum_{k=i}^{r} a_{i,n-k+1,j}d_k, \quad \Phi = (\phi_{ij})_{r\times r}.$$

Proof. When the test equation $y' = \lambda y$ is applied to the method (2.2), we have

$$Y_{n+1} = \bar{B}Y_n + \bar{h}\bar{C}\begin{pmatrix} \sum_{l=1}^{r} a_{1,n,l}y_{n-l+1} & \sum_{l=1}^{r} a_{1,n-1,l}y_{n-l+1} & \cdots & \sum_{l=1}^{r} a_{1,n-r+1,l}y_{n-l+1} \\ 0 & \sum_{l=1}^{r} a_{2,n-1,l}y_{n-l+1} & \cdots & \sum_{l=1}^{r} a_{2,n-r+1,l}y_{n-l+1} \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & \sum_{l=1}^{r} a_{r,n-r+1,l}y_{n-l+1} \end{pmatrix} D$$

$$= \bar{B}Y_n + \bar{h}\bar{C}\begin{pmatrix} \sum_{k=1}^{r}\sum_{l=1}^{r} a_{1,n-k+1,l}y_{n-l+1}d_k \\ \sum_{k=2}^{r}\sum_{l=1}^{r} a_{2,n-k+1,l}y_{n-l+1}d_k \\ \cdots \\ \sum_{k=r}^{r}\sum_{l=1}^{r} a_{r,n-k+1,l}y_{n-l+1}d_k \end{pmatrix}$$

$$= \bar{B}Y_n + \bar{h}\bar{C}\begin{pmatrix} \sum_{k=1}^{r} a_{1,n-k+1,1}d_k & \sum_{k=1}^{r} a_{1,n-k+1,2}d_k & \cdots & \sum_{k=1}^{r} a_{1,n-k+1,r}d_k \\ \sum_{k=2}^{r} a_{1,n-k+1,2}d_k & \sum_{k=2}^{r} a_{1,n-k+1,2}d_k & \cdots & \sum_{k=2}^{r} a_{1,n-k+1,r}d_k \\ \cdots & \cdots & \cdots & \cdots \\ \sum_{k=r}^{r} a_{1,n-k+1,2}d_k & \sum_{k=r}^{r} a_{1,n-k+1,2}d_k & \cdots & \sum_{k=r}^{r} a_{1,n-k+1,r}d_k \end{pmatrix} Y_n$$

So we have

$$Y_{n+1} = (\bar{B} + \bar{h}\bar{C}\Phi)Y_n$$

therefore the region of absolute stability is $\bar{h}$ for $\rho(\bar{B} + \bar{h}\bar{C}\Phi) < 1.\Box$

3. Some Examples

As an example, let r=2

$$y_{n+1} = (b_1, b_2)\begin{pmatrix} y_n \\ y_{n-1} \end{pmatrix} + h(1, c_2)\begin{pmatrix} K_{1,n} & K_{1,n-1} \\ 0 & K_{2,n-1} \end{pmatrix}\begin{pmatrix} d_1 \\ d_2 \end{pmatrix}$$

$$= b_1y_n + b_2y_{n-1} + h(d_1K_{1,n} + d_2K_{1,n-1} + c_2d_2K_{2,n-1})$$

$$K_{1,n} = f(y_n)$$
$$K_{1,n-1} = f(y_{n-1})$$
$$K_{2,n-1} = f(w_{2,1}y_{n-1} + w_{2,2}y_n + h\beta_{2,1}K_{1,n-1})$$

where $K_{1,n}$ and $K_{2,n-1}$ can be computed synchronouslly.

By theorem 2.1, we get the conditions for order 3

$$\begin{array}{l} 0 \leq b_1 < 2 \\ b_1 + d_1 + d_2 + c_2 d_2 = 2 \\ b_1 + 2d_1 + 2c_2 d_2 (w_{2,2} + \beta_{2,1}) = 4 \\ w_{2,1} + w_{2,2} = 1 \\ w_{2,2} = (w_{2,2} + \beta_{2,1})^2 \\ b_1 + 3d_1 + 3w_{2,2} c_2 d_2 = 8 \end{array} \tag{3.1}$$

By theorem 2.2, we get the stability region is

$$\Omega = \{\bar{h} \epsilon \mathbf{C} \mid \rho(\mathbf{G}(\bar{\mathbf{h}})) < \mathbf{1}\}$$

where

$$G = \begin{pmatrix} b_1 & b_2 \\ 1 & 0 \end{pmatrix} + \bar{h} \begin{pmatrix} 1 & c_2 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} d_1 & d_2 \\ w_{2,2} d_2 & w_{2,1} d_2 + \bar{h} \beta_{2,1} d_2 \end{pmatrix}$$

$$= \begin{pmatrix} b_1 + \bar{h}(d_1 + w_{2,2} c_2 d_2) & b_2 + \bar{h}(d_2 + w_{2,1} c_2 d_2 + \bar{h} \beta_{2,1} c_2 d_2) \\ 1 & 0 \end{pmatrix}$$

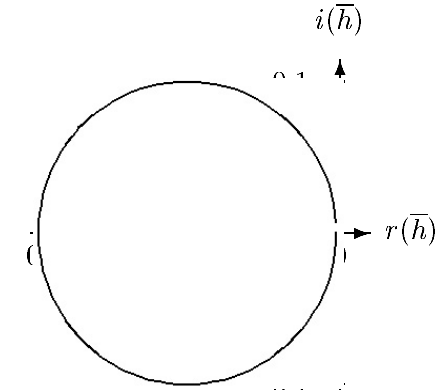
To solve the equations (3.1), we can choose $0 \leq b_1 < 2$ and $w_{22} > 0$, let $\beta_{21} = \sqrt{w_{22}} - w_{22}$, $w_{21} = 1 - w_{22}$, $b_2 = 1 - b_1$, and solve out d_1, d_2 and c_2 from

$$\begin{pmatrix} d_1 \\ d_2 \\ c_2 d_2 \end{pmatrix} = \begin{pmatrix} 1 & 1 & 1 \\ 2 & 0 & 2(w_{22} + \beta_{21}) \\ 3 & 0 & 3w_{22} \end{pmatrix}^{-1} \begin{pmatrix} 2 - b_1 \\ 4 - b_1 \\ 8 - b_1 \end{pmatrix}$$

For example, if to choose $b_1 = 1$ and $w_{22} = 4$, then $\beta_{21} = -2$, $d_1 = \dfrac{2}{3}$, $d_2 = -\dfrac{1}{12}$, $c_2 = -5$, $b_2 = 0$, $w_{21} = -3$, and

$$G = \begin{pmatrix} 1 + \frac{7}{3}\bar{h} & \bar{h}(-\frac{4}{3} - \frac{5}{6}\bar{h}) \\ 1 & 0 \end{pmatrix} \tag{3.2}$$

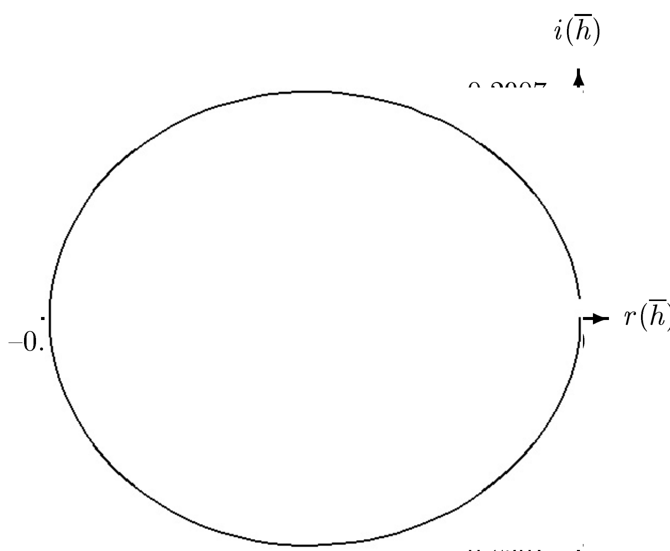
the region of absolute stability is



For another example, if to choose $b_1 = \frac{3}{2}$ and $w_{22} = 4$, then $\beta_{21} = -2$, $d_1 = \frac{1}{3}$, $d_2 = -\frac{7}{24}$, $c_2 = -\frac{11}{7}$, $b_2 = -\frac{1}{2}$, $w_{21} = -3$, and

$$G = \begin{pmatrix} \frac{3}{2} + \frac{19}{6}\bar{h} & -\frac{1}{2} - \bar{h}(\frac{5}{3} + \frac{11}{12}\bar{h}) \\ 1 & 0 \end{pmatrix} \tag{3.3}$$

the region of absolute stability is



By comparing, we can easily find out that the absolute stable region of (3.3) is much larger than that of (3.2).

In paper [5], the one-parameter family of methods

$$\begin{aligned} f_n &= f(y_n), \quad a_1 = \frac{1}{2} + \frac{5}{6c} \pm \frac{1}{2}\sqrt{1+\frac{10}{3c}} \\ g_n &= f(a_1 y_n + (1-a_1)y_{n-1} - \frac{5}{6c}hf_{n-1}), \quad c \neq 0 \\ y_{n+1} &= y_n + h[(\frac{7}{3} - ca_1)f_n + (-\frac{4}{3} - c + ca_1)f_{n-1} + cg_n] \end{aligned} \tag{3.4}$$

was discussed. It was proved that this method is third-order and possesses the real stability interval $[-\frac{1}{5}(11-\sqrt{61}),0]$.

By analyzing, the method (3.4) is only the specifical family of the parallel 2-stage & 2-step method with $b_1 = 1$, $b_2 = 0$, $d_1 = \frac{7}{3} - ca_1$, $d_2 = -\frac{4}{3} - c + ca_1$, $c_2 d_2 = c$, $w_{2,1} = 1 - a_1$, $w_{2,2} = a_1$, $\beta_{2,1} = -\frac{5}{6c}$, $a_1 = \frac{1}{2} + \frac{5}{6c} \pm \frac{1}{2}\sqrt{1+\frac{10}{3c}}$, $c \neq 0$. But the method (3.3) possesses the real stability interval $[-0.719, 0]$ which is larger than the real stability interval $[-0.638, 0]$ of the method (3.4).

The speed-up for some parallel algorithms based on Parallel 2-stage & 2-step Method was implemented on S10 parallel computer (MIMD, it's very similar to MASSCOMP) which was made by Aero-Space Department of China in 1991. For parallel computation by using the parallel 2-stage & 2-step method, $K_{1,j}$, $j = 1, 2, \cdots$, are computed by process $\mathbf{P_1}$, and $K_{2,j}$, $j = 1, 2, \cdots$, are computed by process $\mathbf{P_2}$.

For the problem

$$Y' = AY, \quad Y(0) = (-2, 2, -2, 2, \cdots, -2, 2)^T$$

where $Y = (y_1, y_2, \cdots, y_n)^T, n = 10000$, and

$$A = \begin{pmatrix} -1 & -2 & & & & \\ 1 & 1 & & & & \\ & & -1 & -2 & & \\ & & 1 & 1 & & \\ & & \cdots & \cdots & & \\ & & \cdots & \cdots & & \\ & & & & -1 & -2 \\ & & & & 1 & 1 \end{pmatrix}_{n \times n}$$

Take $h = 0.01$, the computation values at $x_T = 9.0$ are given in Table 1.

Table 1 $h = 0.01$, $x_T = 9.0$, $s = 0, 1, \cdots, 4999$

Entry	Exact Values	Computation Values	Error
y_{2s+1}	0.99802350997925	0.9980149386787841	-8.5713004659175e-006
y_{2s+2}	-1.82226049900050	-1.822257264818129	2.9341823714928e-006

Serial method used about 10.65 seconds, while parallel method used about 5.43 seconds, so the speed-up is 1.96, efficiency is 0.98.

4. Discussion and Conclusions

Parallel Multi-stage & Multi-step Method given in this paper is a general form that can be used for parallel computation in ODEs. It's a form of combining Runge-Kutta method with linear multi-step method, so the conditions of convergence and stability are more complex than that of Runge-Kutta method or linear multi-step method respectively. But by the theory of this paper, algebraic conditions that ensure convergence and stability are easy to be set up. The advantage of this method is that it's equivalent to calculating one right hand function in one step although multi-stage involved.

References

[1] Song Xiaoqiu, etc., Some kinds of parallel Runge-Kutta methods, *Journal of Computational Mathematics,* (1992), 79-85.

[2] Liu Zhiliang, A kind of parallel Runge-Kutta methods, *Computer Engineering and Design,* **5** (1990), 59-64.

[3] W.L. Miranker, W. Liniger, Parallel methods for the numerical integration of ordinary differential equations, *Math. Comp.,* **21** (1967), 303-320.

[4] Song Xiaoqiu, etc., Combination Method for Parallel Computation in ODEs, The Proceeding of 3nd Beijing International Conference on System Simulation and Scientific Computing, 1995, 93-95.

[5] P.J. Van Der Houwen, B.P. Sommeijer, P.A. Van Mourik, Note on explicit parallel multistep Runge-Kutta methods, *Journal of Computational and Applied Mathematics,* **27** (1989), 411-420.

[6] Song Xiaoqiu, Parallel Methods in ODEs, Doctroal Thesis, Beijing Institute of Computer Application and Simulation Technology, 1991.