

一类求解刚性常微分方程的多步插值法 及其平行实现^{*1)}

廖文远 李庆扬

(清华大学应用数学系)

A CLASS OF MULTISTEP INTERPOLATION METHODS FOR SOLVING STIFF ORDINARY DIFFERENTIAL EQUATIONS & PARALLEL IMPLEMENTATION

Liao Wen-yuan Li Qing-yang

(Dept. of Applied Mathematics of Tsinghua University)

Abstract

In this paper, we combine implicit RK method with implicit linear multistep method to propose a class of multistep interpolation methods which make use of the derivatives at the point of t_{n-k+j} ($j = 1, 2, \dots, k-1$) and base on general multistep RK method proposed in [1]. By choosing free parameters in these methods we can make them stable, parallable. The other advantages are their high orders and less computation. Numerical experiments show that these methods are more efficient in solving stiff ODEs than implicit RK method, implicit linear multistep method and multistep RK method in [1].

§ 1. 引 言

考虑 m 维刚性常微分方程的初值问题

$$\begin{cases} y' = f(t, y), \\ y(t_0) = y_0, \end{cases} \quad (1)$$

其中 f 是充分光滑的非线性函数 $f: R \times R^m \rightarrow R^m$.

当问题 (1) 为非刚性时, 有许多可用的方法. 但当问题为刚性时, 仅有隐式方法才可能是刚性稳定的. 其中隐式线性多步法及隐式 Runge-kutta 法是广泛使用的方

* 1997年6月24日收到.

1) 国家自然科学基金资助项目.

法. 但隐式线性多步法精度较低 (A 稳定的隐式线性多步法不超过 2 阶). 当 s 级隐式 Runge-kutta 方法在矩阵 D 不为对角阵或下三角阵时, 每步计算需求解一个 sm 阶非线性方程组, 计算量很大. 基于此, 并为了利用并行机这一硬件环境, 本文构造了将隐式线性多步法与隐式 Runge-kutta 方法结合的多步插值算法. 在求解公式系数时设置自由参数, 通过对自由参数的选取, 调节, 保证了方法的稳定性 (至少为 $A(\alpha)$ 稳定, 部分公式可达到 A 稳定). 另外, 令矩阵 D 为对角阵, 使公式具有完全的并行性. 从而得出了一类集良好的稳定性, 较高的收敛阶, 以及完全的并行性于一体的计算公式. 本文分如下几部分: 在第二节中简要叙述了方法的构造以及计算公式的推导, 并给出了一些结果; 第三节提出了系数的自由参数的概念, 并在此基础上对系数的稳定性进行了分析; 第四节给出了一个具体的计算公式作为代表并用此公式对二个例题进行了计算, 给出了结果分析. 第五节对本文进行了小结.

§ 2. 方法简述及公式推导

求解常微分方程实际上就是计算积分, 在隐式 Runge-kutta 方法中, 由于没有尽可能多的利用前面的信息, 因而所得方法的阶并没有达到最高. 而隐式线性多步法虽然利用了前面的信息, 却不能在保证方法的稳定性前提下获得较高的阶, 即使是文献 [1] 所提出的方法, 也因为公式中未含前面节点处的导数值而没有充分利用已知的信息. 我们知道在整数节点上的值由于方法的阶较内部节点处的阶高而使其精度比内部节点处的高. 本文所提算法即是针对这个不足之处并考虑到了并行性.

令 k, s 为两个正整数, k 步 s 级的多步插值法按如下方式定义:

对于已知的 $y_{n-k+1}, y_{n-k+2}, \dots, y_n, (y_i = y(t_i), t_i = t_0 + ih)$, 我们构造 $2k - s - 1$ 次多项式 $u(t)$ 满足如下条件:

$$\begin{cases} u(t_j) = y_j, & j = n - k + 1, \dots, n, \\ u'(t_j) = f(t_j, u(t_j)), & j = n - k + 1, \dots, n, \\ u'(t_n + c_i h) = f(t_n + c_i h, u(t_n + c_i h)), & i = 1, 2, \dots, s. \end{cases} \quad (2)$$

正整数 k, s 分别为方法的外插步数和内插步数. $c_1, c_2, \dots, c_s, (0 < c_i \leq 1)$ 为内插节点. 令方程在点 t_{n+1} 处的数值解 $y_{n+1} = u(t_{n+1})$, 则等价的计算公式为

$$\begin{cases} g_i = \sum_{j=1}^k a_{ij} y_{n-k+j} + h \sum_{j=1}^k b_{ij} f(t_{n-k+j}, y_{n-k+j}) + h d_{ii} f(t_n + c_i h, g_i), & i = 1, 2, \dots, s. \\ y_{n+1} = \sum_{j=1}^k a_j y_{n-k+j} + h \sum_{j=1}^k b_j f(t_{n-k+j}, y_{n-k+j}) + h \sum_{j=1}^s e_j f(t_n + c_j h, g_j). \end{cases} \quad (3)$$

当 $k > 1$ 时, 为使方法具有刚性稳定性, 固定 $c_s = 1$, 适当选取 c_1, c_2, \dots, c_{s-1} . 方法的最高阶可达到 $2k + 2s - 2$. 为兼顾方法的并行性, 令 $d_{ij} = 0$, 当 $i \neq j$ 时, 即 D 为对角阵. 这样, 内插阶仍可达到 $2k$ 阶, 而方法却具有完全的并行性. 有如下结果:

定理 2.1. a) 对多步插值法 (2) 以下三条等价:

i) 方法的阶为 p ;

ii) $\sum_{j=1}^k a_j = 1$, 且 $\sum_{j=1}^k a_j \frac{(j-k)^{l+1}}{l+1} + \sum_{j=1}^s kb_j(j-k)^l + \sum_{j=1}^s e_j c_j^l = \frac{1}{l+1}$, $l = 0, 1, 2, \dots, p-1$;

1:

iii) (3) 中第二式对所有次数不大于 $p-1$ 次的多项式精确成立.

b) 若 $\sum_{j=1}^k a_{ij} = 1$, 且 $\sum_{j=1}^k a_{ij} \frac{(j-k)^{l+1}}{l+1} + \sum_{j=1}^k b_{ij}(j-k)^l + d_{ii}c_i^l = \frac{c_i^{l+1}}{l+1}$, $i = 1, 2, \dots, s$.

对所有整数 $l = 0, 1, 2, \dots, q-1$ 都精确成立, 则方法的内插阶为 q .

通过适当选取节点 c_1, c_2, \dots, c_{s-1} , (其中 $c_s = 1$ 固定) 及系数 $a_{ij}, b_{ij}, d_{ii}, a_i, b_i, e_i$, 可以使公式达到最高阶 $2k + 2s - 2$. 相应地, 内插阶为 $2k$. 但这样的公式稳定性较差, 因此, 还需作相应的修改. 在第三节将结合稳定性补充相应的条件, 可得到稳定性更好的公式. 定理 2.1 的证明比较简单, 只需令 (3) 对多项式序列 $1, t, t^2, \dots, t^{p-1}$ 精确成立即可. 在定理 2.1 中直接求解 c_i 及各系数将比较困难, 因为这是一个关于 c_1, c_2, \dots, c_{s-1} 的高次方程组, 可以根据方程的特点进行消元. 本文采用另一种较为简便的方法, 通过先求得节点 $c_i (i = 1, 2, \dots, s-1)$, 然后代入定理 2.1 中, 得到关于系数的线性方程组. 一般有如下结果:

定理 2.2. (3) 中的最佳节点 c_1, c_2, \dots, c_{s-1} 为以下非线性方程组的解:

$$\begin{cases} \sum_{j=1}^k \frac{1}{c_i - r_j} + \sum_{j=1, j \neq i}^s \frac{1}{c_i - c_j} = 0, \\ i = 1, 2, \dots, s-1, \end{cases} \quad (4)$$

其中 $r_j = -k + j$, $c_s = 1$. 确定 c_i 以后, (3) 中的 l 的取值只需至 $2k + s - 2$ 即可. 利用文献 [3] 的方法, 构造各插值点上的基函数, 利用基函数在节点处的值及其导数值, 可以比较容易地证明这个结论, 此处从略. 具体的证明过程参见文献 [3].

§ 3. 稳定性分析

将方法 (3) 应用于试验方程 $y' = \lambda y$, $\lambda \in C$ 得

$$\begin{cases} G = A_1 Y_n + z B_1 Y_n + z D G, \\ Y_{n+1} = A_2 Y_n + z B_2 Y_n + z E G, \end{cases} \quad (5)$$

其中 $Y_n = (y_{n-k+1}, y_{n-k+2}, \dots, y_n)^T$, $G = (g_1, g_2, \dots, g_s)^T$, $z = h\lambda$.

$$A_1 = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \cdots & \cdots & \cdots & \cdots \\ a_{s1} & a_{s2} & \cdots & a_{sk} \end{pmatrix}, \quad B_1 = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1k} \\ b_{21} & b_{22} & \cdots & \cdots \\ \cdots & \cdots & \cdots & \cdots \\ b_{s1} & b_{s2} & \cdots & b_{sk} \end{pmatrix}.$$

$$A_2 = \begin{pmatrix} 0 & 1 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \\ a_1 & a_2 & \cdots & a_k \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ b_1 & b_2 & \cdots & b_k \end{pmatrix},$$

$$D = \begin{pmatrix} d_1 & 0 & \cdots & 0 \\ 0 & d_2 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & d_s \end{pmatrix}, \quad E = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ e_1 & e_2 & \cdots & e_s \end{pmatrix}.$$

此处 d_i 即为 (3) 中的 d_{ii} .

由 (5) 得 $Y_{n+1} = C(z)Y_n$, 其中 $C(z) = A_2 + zB_2 + zE(I - zD)^{-1}(A_1 + zB_1)$. $C(z)$ 有如下形式:

$$C(z) = \begin{pmatrix} 0 & 1 & \cdots & 0 \\ 0 & 0 & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & 1 \\ u_0(z) & u_2(z) & \cdots & u_{k-1}(z) \end{pmatrix},$$

$u_i(z) = \frac{p_i(z)}{\prod_{i=1}^s (1 - d_i z)}$. 而 $p_i(z)$ 为 $s+1$ 次多项式. $C(z)$ 的特征值 λ 满足

$$u_0(z) + u_1(z)\lambda + \cdots + u_{k-1}(z)\lambda^{k-1} - \lambda^k = 0. \quad (6)$$

按绝对稳定性定义, 绝对稳定域 $S = \{z \in C \mid (6) \text{ 的单根按模 } \leq 1, \text{ 且 } (6) \text{ 的重根按模 } < 1\}$. 利用边界个轨迹法, 令 $\lambda = e^{i\theta}$, 有 $u_0(z) + u_1(z)e^{i\theta} + \cdots + u_{k-1}(z)e^{i(k-1)\theta} - e^{ik\theta} = 0$. 取 $\theta \in [0, 2\pi]$. 画出 z 的轨迹, 轨迹曲线将整个复平面分为几个区域, 其中某些为稳定区域而另一部分区域为不稳定区域. 从其中任一区域中取一点即可验证该点所属区域是否为稳定区域, 从而可得方法的稳定域. 为保证方法的稳定性, 取 d_1, d_2, \cdots, d_s 为待定参数, 令 $p_i(z)$ 的 $s+1$ 次系数为 0 (共需要 k 个参数, 因此一般要求 $k \leq s$). 由此确定出自由参数, 从而得出具体的计算公式. 有时为了使方法的稳定性更好, 还可以多设置自由参数以改变 $C(z)$ 从而改良方法的稳定性, 这样相应地方法的阶就会有所下降.

$k=1$ 是一种比较简单的情形, 取 $d_{ij} = 0, i \neq j, d_1 = d_2 = \cdots = d_s$, 令 $u_0(z)$ 的最高次项系数为 0, 可解得 d_i . 代入 $C(z)$ 的表达式, 可得 $u_0(z)$ 的表达式, 由此可得到稳定域为整个左半平面, 因而方法是 A 稳定的. 但此时方法的级阶很低, 只有一阶精度, 而公式的方法阶为 $p = 2s$. 以 $s = 3$ 为例, 具体的计算公式为

$$c_1 = 0.276393, \quad c_2 = 0.723607, \quad c_3 = 1,$$

$$A_1 = \begin{pmatrix} 1 \\ 1 \\ 1 \end{pmatrix}, \quad A_2 = (1),$$

$$B_1 = \begin{pmatrix} \frac{-\sqrt{15}}{10} \\ \frac{\sqrt{15}}{10} \\ \frac{1}{2} \end{pmatrix}, \quad B_2 = \left(\frac{1}{12}\right),$$

$$D = \begin{pmatrix} 0.5 & 0 & 0 \\ 0 & 0.5 & 0 \\ 0 & 0 & 0.5 \end{pmatrix}, \quad E = \begin{pmatrix} 5 & 5 & 1 \\ 12 & 12 & 12 \end{pmatrix}.$$

本文取 $k=2, s=3$ 进行了具体分析. 由定理 2.1, 取 d_1, d_2, d_3 为自由参数, 则所得的计算公式不能满足稳定性要求, 因此可令 $p_0(z), p_1(z)$ 的四次项及三次项系数均为 0. 这样就需要 4 个自由参数, 故再令 b_1 为自由参数, 相应地方方法的阶将有所下降. 确定出自由参数后, 代入系数所满足的方程组, 解出方法的系数如下:

$$c_1 = 0.328356, \quad c_2 = 0.761369, \quad c_3 = 1$$

$$A_1 = \begin{pmatrix} -0.128807 & 1.128807 \\ -0.353413 & 1.353413 \\ -6.08996 & 7.08996 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 1 \\ 0.408763 & 0.591237 \end{pmatrix},$$

$$B_1 = \begin{pmatrix} -0.0526846 & 0.0523634 \\ -0.185023 & 0.223235 \\ -2.62081 & -3.3933 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & 0 \\ 1.04623 & -0.593548 \end{pmatrix},$$

$$D = \begin{pmatrix} -0.199869 & 0 & 0 \\ 0 & 0.369755 & 0 \\ 0 & 0 & 0.9224163 \end{pmatrix}, \quad E = \begin{pmatrix} 0 & 0 & 0 \\ -0.593548 & 1.00527 & -0.162 \end{pmatrix}$$

由于令 b_1 为自由参数, 使得 c_1, c_2, \dots, c_{s-1} 的选取不能提高方法的阶. 由此得出的方法的阶及级阶分别为 $p=5, q=3$,

$$u_0(z) = \frac{0.408763 + 0.210005z + 0.0850959z^2}{1 - 1.49379z + 0.60032z^2 - 0.068298z^3},$$

$$u_1(z) = \frac{0.591237 - 0.295029z - 0.47292z^2}{1 - 1.49379z + 0.60032z^2 - 0.068298z^3}.$$

据此画出方法的稳定域, 见图 1(其中两条封闭曲线的公共外部区域为方法的稳定域, 即图中的区域 1. 所以方法是 A 稳定的).

$K=2, s=5$:

取 d_1, d_2, \dots, d_5 为自由参数, 令 $p_0(z)$ 和 $p_1(z)$ 的 6 次项及 5 次项系数为 0, 得出一个关于 d_1, d_2, \dots, d_5 的包含 4 个方程的方程组. 调节其中的一个变量, 使方法的稳定性达到最佳, 同时确定出各系数. 解出方法的系数如下: (由于系数 a_1 非常接近 0, 因此为方便计算, 令其为 0, 这将使方法的阶降低 1 阶).

$$c_1 = 0.135506, \quad c_2 = 0.390743 \quad c_3 = 0.671021 \quad c_4 = 0.894024, c_5 = 1$$

$$A_1 = \begin{pmatrix} -0.0413862 & 1.0413862 \\ -9.73648 & 10.73648 \\ 1.204404 & -0.204404 \\ 0.562613 & 0.437387 \\ 6.5 & -5.5 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 1 \\ 0 & 1 \end{pmatrix},$$

$$B_1 = \begin{pmatrix} -0.0148519 & -0.0106089 \\ -3.69211 & -9.01639 \\ 0.452577 & 1.3079 \\ 0.169818 & 0.962342 \\ 2.625 & 5 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & 0 \\ 0.0000247459 & 0.3346073 \end{pmatrix},$$

$$D = \begin{pmatrix} 0.109236 & 0 & 0 & 0 & 0 \\ 0 & 3.3444 & 0 & 0 & 0 \\ 0 & 0 & 0.0998502 & 0 & 0 \\ 0 & 0 & 0 & 0.31848 & 0 \\ -0.125 & 0 & 0 & 0 & 0 \end{pmatrix},$$

$$E = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ -0.377478 & 0.691739 & 0.0415053 & 0.314893 & -0.0169788 \end{pmatrix}.$$

方法的阶可以直接从定理 2.2 中得出: $p = 11, q = 3$,

$$u_0(z) = \frac{-6.60224z - 0.229508z^2 + 0.759542z^3 - 0.0724503z^4}{1 - 3.74697z + 1.35789z^2 - 0.032486z^3 - 0.0212143z^4 + 0.0014522z^5},$$

$$u_1(z) = \frac{1 + 3.85548z - 8.2604z^2 + 1.93393z^3 - 0.122696z^4}{1 - 3.74697z + 1.35789z^2 - 0.032486z^3 - 0.0212143z^4 + 0.0014522z^5}.$$

据此画出方法的稳定域, 见图 2. 由于 $u_0(z)$ 和 $u_1(z)$ 的分子和分母中分别出现了 z^4, z^5 项, 所以在用边界轨迹法时所画出的曲线将是两条封闭的曲线. 这两条曲线将整个复平面分成四个区域, 我们对其标为 1, 2, 3, 4. 显然, 区域 1 是稳定域, 而区域 3 是不稳定域. 对区域 2 和区域 4 分别在其中选取一点验证其是否属于方法的稳定点, 从而可以判断该点所属的区域是否为方法的稳定域. 经过验证知方法的稳定域为区域 1 与区域 2 的并集, 由此可知方法是 $A(\alpha)$ 稳定的.

一般说来, 在保证方法的稳定性的前提下, 当 $2k \leq s$ 时, 方法的阶可达到 $2k+2s-2$, 级阶可达到 $2k-1$. 例如当 $k=2, s=5$ 时, 方法的阶达到了 11, 如果不事先设置 $a_1=0$.

则方法的阶可达到 12 阶, 而且对方法的稳定性没有影响. 事实上, 当 $k = 2, s = 4$ 时, 我们可以得到 $p = 2k + 2s - 2 = 10, q = 2k - 1 = 3$ 的刚性稳定的公式

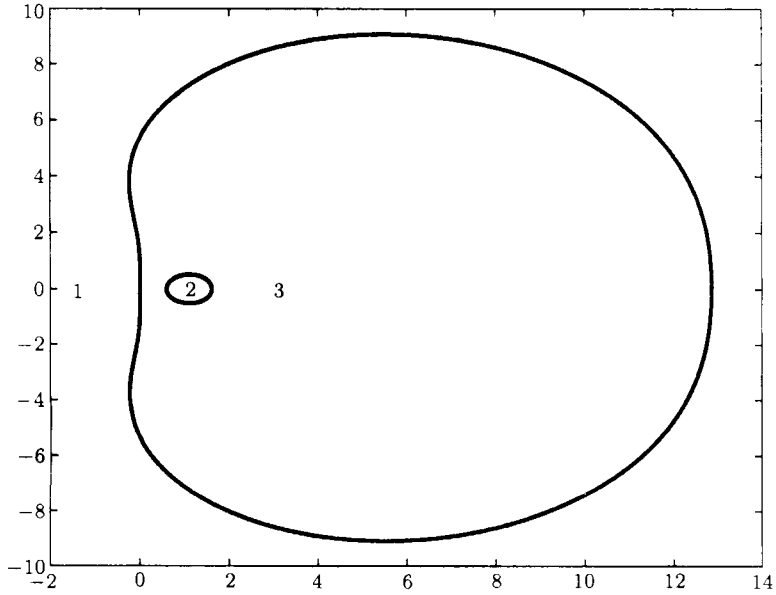


图 1 $k = 2, s = 3$

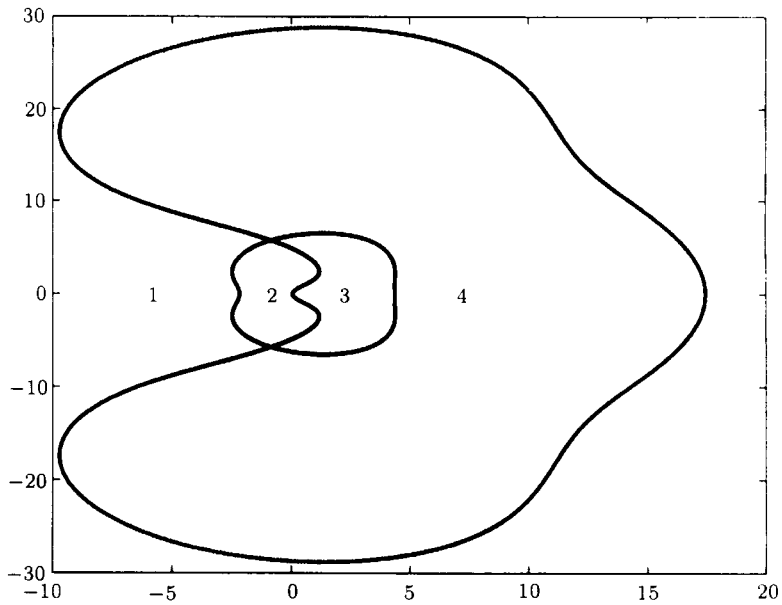


图 2 $k = 2, s = 5$

$$c_1 = 0.202863 \quad c_2 = 0.541359 \quad c_3 = 0.84718 \quad c_4 = 1$$

$$A_1 = \begin{pmatrix} -0.303644 & 1.303644 \\ -1.40219 & 2.40219 \\ 2.24536 & -1.24536 \\ 3.92712 & -2.92712 \end{pmatrix}, \quad A_2 = \begin{pmatrix} 0 & 1 \\ 0.00035949 & 0.999641 \end{pmatrix},$$

$$B_1 = \begin{pmatrix} -0.111005 & -0.289544 \\ -0.567565 & -0.805782 \\ 0.86454 & 2.11082 \\ 1.52316 & 3.28474 \end{pmatrix}, \quad B_2 = \begin{pmatrix} 0 & 0 \\ 0.0000694706 & 0.0660034 \end{pmatrix},$$

$$D = \begin{pmatrix} 0.301702 & 0 & 0 & 0 \\ 0 & 0.517593 & 0 & 0 \\ 0 & 0 & 0.129044 & 0 \\ 0 & 0 & 0 & 0.119209 \end{pmatrix},$$

$$E = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0.299843 & 0.357339 & 0.240439 & 0.0366662 \end{pmatrix}.$$

但有时在对阶的要求不是特别高的时候, 可以事先给定某些系数以简化公式的推导以及降低计算的复杂度. 在某些特殊情况下, 为保证方法较好的稳定性, 可能还将多设置一些自由参数, 为此, 方法的阶可能有所下降.

§ 4. 方法的并行实现及数值试验结果分析

由于 D 为对角阵, 因而求解 g_1, g_2, \dots, g_s 实际上是相互独立的过程, 采用 SIMD 模式 (即单指令流多数据流) 在并行机上很容易实现并行计算, 并且程序结构简单, 并行效果良好, 程序简洁明了. 通用性较高, 对不同的问题, 只需作很小的修改即可使用. 在计算过程中所需的处理器个数为 s , 即方法的内插点个数, 利用第三节中得到的 $k=2, s=3$ (此时公式的阶 $p=5$) 的计算公式在 $sp2$ 机 (分布式内存, 各工作站间通过高速开关网和内部以太网连接) 上对以下几个例题进行了计算, 并与用此公式串行计算 (用一个同样的工作站) 的结果以及用 $s=3$ 的 Radau IIA 方法 (此时公式的阶 $p=5$) 计算的结果 (也用同样的工作站) 进行了比较. 例题及计算结果以及比较结果列于表 1.

表 1 计算结果分析

	T_1	T_2	mp	T_3	sp	e
例题 1	0.53	0.13	4.08	0.06	2.17	0.723
例题 2	327.31	107.99	3.31	39.09	2.963	0.921

注. 在进行非线性方程组的牛顿迭代时所用的线性方程组的求解方法为全主元高斯消去法

例题 1. 求解化学动力学中关于反应速度的刚性问题

$$\begin{cases} y_1'(t) = -0.013y_1 - 1000y_1y_3, \\ y_2'(t) = 2500y_2y_3, \\ y_3'(t) = -0.013y_1 - 1000y_1y_3 - 2500y_2y_3. \end{cases} \quad (7)$$

当 $t = 0$ 时, $y_1 = 1.0, y_2 = 1.0, y_3 = 0$.

计算结果: 当 $t = 1.0$ 时, $y_1 = 0.000000, y_2 = 1.000009, y_3 = 0.000000$, 这里 y_1, y_2, y_3 表示反应物的浓度.

例题 2. 求解一个由偏微分方程 (Burgers Equation) 转化而得到的问题 $u_t + uu_x = \mu u_{xx}$. 其中, $\mu = 0.0003, 0 \leq x \leq 1.0, u(0, t) = 0, u(x, 0) = (\sin(3\pi x))^2(1-x)^{3/2}$. 利用差商将上述问题转化为常微分方程组. 取 $\Delta x = \frac{1}{101}$, 即得到一个 100 阶常微分方程组 $u_i = \frac{u_i(u_{i+1} - u_{i-1})}{2\Delta x} + \frac{\mu(u_{i+1} - 2u_i + u_{i-1})}{(\Delta x)^2}$ (由于此方程组规模较大, 此处略去数值结果).

表 1 中 T_1 为按隐式 Runge-kutta 方法串行计算所用的时间, T_2 为按多步插值法串行计算所用的时间, T_3 为按多步插值法并行计算所用的时间, sp 为加速比. $sp = \frac{T_2}{T_3}, e$ 为并行效率, $e = \frac{sp}{S}$, 其中 S 为所用 CPU 个数.

结果表明本文得出的计算方法比同阶的隐式 Runge-kutta 方法节省了时间 3—4 倍 (在串行计算的情况下), 并行效率也较高, 对维数大的问题, 并行效果更好.

§5. 结 束 语

由于本方法综合了隐式线性多步法与隐式 Runge-kutta 法的特点, 并利用了并行机的优点, 因此, 无论是从计算量, 计算速度, 稳定性, 还是从方法的阶来看, 都明显优于前两种方法. 与文献 [1] 中的方法相比, 除公式具有很好的并行性外, 在绝大多数情况下具有更高的内插阶和方法阶. 仅在 $s > k$ 时, 本方法的内插阶低于文献 [1] 中所提出的方法. 总之, 本方法主要的优点在于构造了适合于并行计算的公式 (也可用于串行计算) 而同时仍然保持较高的级阶和方法阶, 且至少为刚性稳定的.

参 考 文 献

- [1] Stefan Schneider, Numerical Experiments with a Multistep Radau method, *BIT*, **33** (1993), 332—350.
- [2] 李庆扬, 常微分方程数值解法 (刚性问题和边值问题), 高等教育出版社, 1991.
- [3] E. Hairer, G. Wanner, Solving Ordinary Differential Equations II. Stiff and Differential-Algebra Problems. Springer-Verlag, Berlin (1991).
- [4] B. Orel, Parallel Runge-kutta methods with real eigenvalues, *Appl. Numer. Math.*, **11** (1993), 241—250.
- [5] Z. Jackiewicz, Stability analysis of time-point of parallel relaxation Runge-kutta methods with respect to tridiagonally systems of differential equations, *Appl. Numer. Math.*, **11** (1993), 189—210.