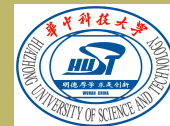


第六章 常微分方程初值问题的数值解法



对于一阶常微分方程初值问题

$$\begin{cases} \frac{dy}{dt} = f(t, y(t)), & t \in [a, b] \\ y(a) = y_0, \end{cases} \quad (1)$$

当右函数 $f(t, y) \in \mathbb{C}([a, b] \times R)$ 满足Lipschitz条件

$$|f(t, y_1) - f(t, y_2)| \leq L|y_1 - y_2|, \quad t \in [a, b], \quad y_1, y_2 \in \mathbb{R} \quad (2)$$

时, 其连续可微解 $y(t)$ 在 $[a, b]$ 上存在且唯一。理论上虽已证明在适当条件下其解的存在性, 但实际求解此类问题时, 仍十分困难, 如: 对于初值问题

$$y'(t) = \exp(-t^2), \quad t \in [0, 1]; \quad y(0) = 1,$$

我们易求得其解析解 $y(t) = 1 + \int_0^t \exp(-x^2) dx$ ($t \in [0, 1]$), 但由于其积分项 $\int_0^t \exp(-x^2) dx$ 的值无法精确得到, 因此解析求解方法存在着极大的局限性。为此本章将介绍求解问题(1)的通用方法—数值解法。

访问主页

标题页

◀ ▶

◀ ▶

第 1 页 共 51 页

返回

全屏显示

关闭

退出



§6.1 基本离散方法

本节将介绍常微分方程初值问题的三种基本离散方法, 即**差商逼近法**, **数值积分法**和**Taylor展开法**。在下列算法构造中, 我们恒记节点 $t_n = a + nh$ ($n = 0, 1, 2, \dots, N$), 步长 $h \leq \frac{b-a}{N}$, $y_n \approx y(t_n)$ 及 $f_n := f(t_n, y_n) \approx y'(t_n)$ 。

§6.1.1 差商逼近法

差商逼近法即是用适当差商逼近导数值而使系统(1)离散化的一种方法. 如: 当 h 充分小时, 我们有

$$y'(t_n) \approx \frac{y(t_{n+1}) - y(t_n)}{h}, \quad y'(t_{n+1}) \approx \frac{y(t_n) - y(t_{n+1})}{-h}, \quad (3)$$

若引入参数 $\theta \in [0, 1]$, 则由上两式得

$$\theta y'(t_n) + (1 - \theta)y'(t_{n+1}) \approx \frac{y(t_{n+1}) - y(t_n)}{h},$$

即

$$y(t_{n+1}) \approx y(t_n) + h[\theta y'(t_n) + (1 - \theta)y'(t_{n+1})]. \quad (4)$$

[访问主页](#)[标题页](#)[<<](#) [>>](#)[<](#) [>](#)

第 2 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



采用先前约定符号即得求解系统(1)的**线性 θ -方法**

$$y_{n+1} = y_n + h[\theta f_n + (1 - \theta)f_{n+1}]. \quad (5)$$

方法(5)在求当前值 y_{n+1} 时, 仅需要先前一步的信息 y_n, f_n , 因此我们称这种方法为**单步法**. 特别, 若 $\theta = 1$ 则得**显式Euler法**

$$y_{n+1} = y_n + hf_n \quad (6)$$

若 $\theta = 0$, 则得**隐式Euler法**

$$y_{n+1} = y_n + hf_{n+1} \quad (7)$$

若 $\theta = \frac{1}{2}$, 则得**梯形法**

$$y_{n+1} = y_n + \frac{h}{2}(f_n + f_{n+1}). \quad (8)$$

方法(6)是一个显式方法, 即其每个计算步只需直接递推就可获得当前数值解 y_{n+1} , 而方法(7)及(8)是隐式方法, 其每个计算步均需解一个关于 y_{n+1} 的隐式方程才可获得当前数值解.

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 3 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



若记

$$Y_{1, n} = y_n, \quad Y_{2, n} = y_n + h[\theta f(t_n, Y_{1, n}) + (1 - \theta)f(t_n + h, Y_{2, n})],$$

则方法(5)也可等价地写成**二级Runge-Kutta方法**

$$\begin{cases} Y_{1, n} = y_n, \\ Y_{2, n} = y_n + h[\theta f(t_n, Y_{1, n}) + (1 - \theta)f(t_n + h, Y_{2, n})], \\ y_{n+1} = y_n + h[\theta f(t_n, Y_{1, n}) + (1 - \theta)f(t_n + h, Y_{2, n})]. \end{cases} \quad (9)$$

§6.1.2 数值积分法

数值积分法的基本思想是先将问题(1)转化为积分方程

$$y(t_m) - y(t_n) = \int_{t_n}^{t_m} f(t, y(t))dt, \quad m > n, \quad (10)$$

然后采用第五章介绍的数值积分公式将上式右端离散化，从而获得原初值问题的差分格式。

访问主页

标题页

◀

▶

◀

▶

第 4 页 共 51 页

返回

全屏显示

关闭

退出



取 $m = n + 1$ ，应用矩形公式于(10)中的积分项，则得单支 θ -方法

$$y_{n+1} = y_n + hf[\theta t_n + (1-\theta)t_{n+1}, \theta y_n + (1-\theta)y_{n+1}], \quad \theta \in [0, 1]. \quad (11)$$

特别，当 $\theta = \frac{1}{2}$ 时，称之为**隐式中点公式**。

若记

$$Y_{1, n} = y_n + h(1 - \theta)f[t_n + (1 - \theta)h, Y_{1, n}],$$

则方法(11)也可以写为**单级Runge-Kutta方法**

$$\begin{cases} Y_{1, n} = y_n + h(1 - \theta)f[t_n + (1 - \theta)h, Y_{1, n}], \\ y_{n+1} = y_n + hf[t_n + (1 - \theta)h, Y_{1, n}]. \end{cases} \quad (12)$$

在公式(10)中取 $m = n + k$ ($k \in \mathbb{N}$)，并将Newton-Cotes公式代入其中，则可得一类**线性 k 步公式**

$$y_{n+k} = y_n + h \sum_{i=0}^k \beta_i f_{n+i}, \quad n = 0, 1, 2, \dots, N - k, \quad (13)$$

其中，
$$\beta_i = \frac{(-1)^{k-i}}{i!(k-i)!} \int_0^k \prod_{j=0, j \neq i}^k (t - j) dt.$$

访问主页

标题页

◀ ▶

◀ ▶

第 5 页 共 51 页

返回

全屏显示

关闭

退出



特别, 当取 $k = 2$ 时, 可得Milne公式

$$y_{n+2} = y_n + \frac{h}{3}(f_{n+2} + 4f_{n+1} + f_n). \quad (14)$$

利用公式(13)计算当前步 y_{n+k} 时, 其需要前 k 步逼近值 $y_n, y_{n+1}, \dots, y_{n+k-1}$ 的有关消息, 故称之为 k 步公式。关于这类公式更广泛的形式将在下段继续讨论。

§6.1.3 Taylor展开法

Taylor展开法的基本思想是首先构造一个关于真解及其有关信息的含参算子, 将算子中诸项在某点处按Taylor展式展开并截去余项, 然后令诸同类项系数为零, 由此确定出原算子中的全部或部分参数, 从而获得一个或一类关于数值解的差分格式。

设 α_i, β_i 为待定系数, 定义算子

$$L[y(t), h] = \sum_{i=0}^k [\alpha_i y(t + ih) - h\beta_i y'(t + ih)]. \quad (15)$$

[访问主页](#)[标题页](#)[<<](#) [>>](#)[<](#) [>](#)

第 6 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



若 $y(t)$ 有 $p+2$ 阶连续导数, 则由Taylor展式有

$$L[y(t), h] = \sum_{j=0}^p C_j h^j y^{(j)}(t) + C_{p+1} h^{p+1} y^{(p+1)}(t) + \mathcal{O}(h^{p+2}), \quad (16)$$

其中

$$\begin{cases} C_0 = \sum_{i=0}^k \alpha_i, & C_1 = \sum_{i=1}^k (i\alpha_i - \beta_i) - \beta_0, \\ C_j = \frac{1}{j!} \sum_{i=1}^k i^{j-1} (i\alpha_i - j\beta_i), & j = 2, 3, \dots, p+1. \end{cases} \quad (17)$$

令

$$C_0 = C_1 = \dots = C_p = 0, \quad C_{p+1} \neq 0 \quad (18)$$

则得诸参数 α_i, β_i 使得算子

$$L[y(t), h] = C_{p+1} h^{p+1} y^{(p+1)}(t) + \mathcal{O}(h^{p+2})$$

的条件。

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 7 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



由上式及(16)得

$$\sum_{i=0}^k [\alpha_i y(t+ih) - h\beta_i y'(t+ih)] = C_{p+1} h^{p+1} y^{(p+1)}(t) + \mathcal{O}(h^{p+2}). \quad (19)$$

在上式中取

$$t = t_n \quad y_{n+i} \approx y(t_{n+i}) \quad f_{n+i} \approx y'(t_{n+i}),$$

去掉余项, 则得线性 k 步方法

$$\sum_{i=0}^k \alpha_i y_{n+i} = h \sum_{i=0}^k \beta_i f_{n+i}, \quad \text{其中 } \alpha_k \neq 0 \quad \alpha_0^2 + \beta_0^2 \neq 0. \quad (20)$$

若(18)成立, 则称方法(20)为 p 阶相容的, 此时称式(19)的右端为该方法的局部截断误差. 特别, 若 $\beta_k \neq 0$, 则称之为隐式的, 否则称之为显式的。

访问主页

标题页

◀ ▶

◀ ▶

第 8 页 共 51 页

返回

全屏显示

关闭

退出



上述分析过程表明:

定理6.1 线性 k 步法(20)为 p 阶相容的充要条件是(18)成立, 且 $C_{p+1} \neq 0$.

在(18)中取 $k = 2, p = 3, \alpha_2 = 1$ 得

$$\begin{cases} \alpha_0 + \alpha_1 + 1 = 0, \\ \alpha_1 + 2 - (\beta_0 + \beta_1 + \beta_2) = 0, \\ \frac{1}{2!}(\alpha_1 + 4) - (\beta_1 + 2\beta_2) = 0, \\ \frac{1}{3!}(\alpha_1 + 8) - \frac{1}{2!}(\beta_1 + 4\beta_2) = 0, \end{cases}$$

解之得

$$\alpha_1 = -1 - \alpha_0, \quad \beta_0 = -\frac{1}{12}(1 + 5\alpha_0), \quad \beta_1 = \frac{2}{3}(1 - \alpha_0), \quad \beta_2 = \frac{1}{12}(5 + \alpha_0).$$

从而得二步方法

$$y_{n+2} - (1 + \alpha_0)y_{n+1} + \alpha_0 y_n = \frac{h}{12}[(5 + \alpha_0)f_{n+2} + 8(1 - \alpha_0)f_{n+1} - (1 + 5\alpha_0)f_n]. \quad (21)$$

访问主页

标题页

◀ ▶

◀ ▶

第 9 页 共 51 页

返回

全屏显示

关闭

退出



对于该方法，我们有

$$C_4 = -\frac{1}{24}(1 + \alpha_0), \quad C_5 = -\frac{1}{360}(17 + 13\alpha_0).$$

因此, 当 $\alpha_0 \neq -1$ 时, $C_4 \neq 0$, 此时方法(21)为三阶的。特别, 若取 $\alpha_0 = -5$, 则得二步显式方法

$$y_{n+2} + 4y_{n+1} - 5y_n = 2h(2f_{n+1} + f_n); \quad (22)$$

若取 $\alpha_0 = 0$, 则得三阶Adams-Monlton方法

$$y_{n+2} - y_{n+1} = \frac{h}{12}(5f_{n+2} + 8f_{n+1} - f_n); \quad (23)$$

若取 $\alpha_0 = -1$, 则 $C_4 = 0$, 但 $C_5 = -\frac{1}{90} \neq 0$, 此时方法(21)即为四阶Milne方法

$$y_{n+2} - y_n = \frac{h}{3}(f_{n+2} + 4f_{n+1} + f_n). \quad (24)$$

[访问主页](#)[标题页](#)[<<](#) [>>](#)[<](#) [>](#)

第 10 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



类似地，若在(18)中取 $k = p = 3$ ， $\alpha_3 = -\alpha_2 = 1$ ， $\alpha_1 = \alpha_0$ ，则得三步Adams方法

$$y_{n+3} = y_{n+2} + \frac{h}{12} \left[(5 - 12\beta_0)f_{n+3} + (8 + 36\beta_0)f_{n+2} - (1 + 36\beta_0)f_{n+1} + 12\beta_0f_n \right], \quad (25)$$

且此时 $C_4 = \beta_0 - \frac{1}{24}$ ， $C_5 = -\frac{4}{45} + \frac{3}{2}\beta_0$ 。因此，当 $\beta_0 \neq \frac{1}{24}$ 时，方法(25)为三阶的；当 $\beta_0 = \frac{1}{24}$ 时， $C_4 = 0$ ，但 $C_5 \neq 0$ ，此时该公式即为四阶Adams方法

$$y_{n+3} = y_{n+2} + \frac{h}{24} [9f_{n+3} + 19f_{n+2} - 5f_{n+1} + f_n]. \quad (26)$$

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)

第 11 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



§6.2 Runge-Kutta 方法

一个 k 步方法当用于计算初值问题(1)时, 除需问题本身的初值 y_0 外, 还需要 $k-1$ 个额外的计算启动值 y_1, y_1, \dots, y_{k-1} . 而一般单步方法则可自起始完成计算. 但是, 如果单步方法的每步计算: $y_n \rightarrow y_{n+1}$ 仅由前一步的逼近值 y_n 直接导出, 则这样的单步方法往往计算精度不高. 如: 线性 θ -方法(5)与单支 θ -方法(11)的局部截断误差均为

$$R_n := (\theta - \frac{1}{2})h^2 y''(t_n) + \mathcal{O}(h^3),$$

则当 $\theta \neq \frac{1}{2}$ 时, 其方法的相容阶仅等于1; 当 $\theta = \frac{1}{2}$ 时, 其方法即分别为梯形法与隐式中点法, 其相容阶达到这类方法的最高阶2. 为提高单步法的精度, Runge与Kutta分别提出了在由 y_n 到 y_{n+1} 的计算过程中增加若干中间逼近值的数值计算方案, 这就形成了我们本节将要介绍的Runge-Kutta 方法. 值得注意的是: 在上节, 我们已指出线性 θ -方法与单支 θ -方法可写成Runge-Kutta 方法形式, 但其属低阶方法, 本节介绍的Runge-Kutta 方法将涵盖高阶方法.

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)[第 12 页 共 51 页](#)[返回](#)[全屏显示](#)[关闭](#)[退出](#)



§6.2.1 方法的一般形式

Runge-Kutta方法的一般形式为

$$\begin{cases} Y_{i,n} = y_n + h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{j,n}), & i = 1, 2, \dots, s, \\ y_{n+1} = y_n + h \sum_{j=1}^s b_j f(t_n + c_j h, Y_{j,n}), & n \geq 0, \end{cases} \quad (27)$$

其中诸系数 a_{ij} , b_j 及横标 c_j 为实数, $h > 0$ 为步长, $t_n = a + nh$, $Y_{i,n} \approx y(t_n + c_i h)$, $y_n \approx y(t_n)$ 。

对于方法(64), 若 $i \leq j$ 时均有 $a_{ij} = 0$, 则称之为显式方法, 否则称之为隐式方法. 为简化Runge-Kutta方法的书写, 我们经常采用所谓Butcher表

$$\begin{array}{c|c} c & A \\ \hline & b^T \end{array} \quad (28)$$

来表征方法(64), 其中

$$A = (a_{ij}) \in \mathbb{R}^{s \times s}, \quad c = (c_1, c_2, \dots, c_s)^T, \quad b = (b_1, b_2, \dots, b_s)^T \in \mathbb{R}^s.$$

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 13 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



一个Runge-Kutta方法(64)称为是 p 阶相容的(或简称为 p 阶的), 若其局部离散误差

$$d_{n+1} := y(t_{n+1}) - \tilde{y}_{n+1} = \mathcal{O}(h^{p+1}), \quad h \rightarrow 0 \quad (29)$$

其中 \tilde{y}_{n+1} 由

$$\begin{cases} y_{i,n} = y(t_n) + h \sum_{j=1}^s a_{ij} f(t_n + c_j h, y_{j,n}), & i = 1, 2, \dots, s, \\ \tilde{y}_{n+1} = y(t_n) + h \sum_{j=1}^s b_j f(t_n + c_j h, y_{j,n}), & n \geq 0 \end{cases} \quad (30)$$

确定. Runge-Kutta方法的相容阶概念是构造具体方法的基本点.

§6.2.2 显式方法

显式Runge-Kutta方法与隐式Runge-Kutta方法相比较, 由于前者在每步无须解 s 个隐式方程, 因此其具有计算简单的特点. 具体显式Runge-Kutta方法的推导可采用Taylor展开法获得. 下面, 我们以三级三阶显式Runge-Kutta法为例来说明其导出过程.

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 14 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



设 $y(t)$ 为系统(1)的充分可微解, 则由Taylor展式有

$$\begin{aligned} y(t_n + h) &= y(t_n) + \sum_{i=1}^3 \frac{1}{i!} h^i y^{(i)}(t_n) + \mathcal{O}(h^4) \\ &= y(t_n) + h \hat{f}_n + \frac{1}{2} h^2 F_n + \frac{1}{6} h^3 (F_n \hat{f}'_n + G_n) + \mathcal{O}(h^4) \end{aligned} \quad (31)$$

其中

$$\begin{aligned} \hat{f}_n &= f(t_n, y(t_n)), \quad \hat{f}'_n = \frac{\partial f(t_n, y(t_n))}{\partial y} F_n = \frac{\partial f(t_n, y(t_n))}{\partial t} + \hat{f}_n \cdot \hat{f}'_n, \\ G_n &= \frac{\partial^2 f(t_n, y(t_n))}{\partial t^2} + 2 \hat{f}_n \frac{\partial^2 f(t_n, y(t_n))}{\partial t \partial y} + \hat{f}_n^2 \frac{\partial^2 f(t_n, y(t_n))}{\partial y^2}. \end{aligned}$$

以下恒设三级显式Runge-Kutta法满足

$$c_1 = 0, \quad c_i = \sum_{j=1}^{i-1} a_{ij}, \quad i = 2, 3. \quad (32)$$

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)

第 15 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 16 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

若方法自精确值 $y(t_n)$ 出发计算一步，则有

$$\begin{cases} y_{1,n} = f(t_n, y(t_n)), \\ y_{2,n} = f(t_n + c_2 h, y(t_n) + h a_{21} y_{1,n}), \\ y_{3,n} = f(t_n + c_3 h, y(t_n) + h a_{31} y_{1,n} + h a_{32} y_{2,n}), \\ \tilde{y}_{n+1} = y(t_n) + h \sum_{j=1}^3 b_j f(t_n + c_j h, y_{j,n}). \end{cases} \quad (33)$$

将 $y_{2,n}, y_{3,n}$ 在 $(t_n, y(t_n))$ 处展开，并注意到条件(32)得

$$\begin{cases} y_{2,n} = \hat{f}_n + h c_2 F_n + \frac{1}{2} h^2 c_2^2 G_n + \mathcal{O}(h^3) \\ y_{3,n} = \hat{f}_n + h c_3 F_n + h^2 (c_2 a_{32} F_n \hat{f}'_n + \frac{1}{2} c_3^2 G_n) + \mathcal{O}(h^3). \end{cases} \quad (34)$$

将上述所获 $y_{i,n}$ ($i = 1, 2, 3$) 的表达式代入(33)的最后一式得

$$\begin{aligned} \tilde{y}_{n+1} = & y(t_n) + h(b_1 + b_2 + b_3) \hat{f}_n + h^2 (b_2 c_2 + b_3 c_3) F_n \\ & + \frac{1}{2} h^3 [2 b_3 c_2 a_{32} F_n \hat{f}'_n + (b_2 c_2^2 + b_3 c_3^2) G_n] + \mathcal{O}(h^4). \end{aligned} \quad (35)$$



由相容阶定义及式(31)与(35)，若要方法有三阶，则必有(32)及下列条件成立：

$$\begin{cases} b_1 + b_2 + b_3 = 1, \\ b_2 c_2 + b_3 c_3 = \frac{1}{2}, \\ b_2 c_2^2 + b_3 c_3^2 = \frac{1}{3}, \\ b_3 c_2 a_{32} = \frac{1}{6}. \end{cases} \quad (36)$$

取{(32), (36)}的一组解

$$b_1 = \frac{1}{4}, \quad b_2 = 0, \quad b_3 = \frac{3}{4}, \quad c_1 = a_{31} = 0, \quad c_2 = a_{21} = \frac{1}{3}, \quad c_3 = a_{32} = \frac{2}{3},$$

可得三阶Heun方法

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & \\ \frac{1}{3} & \frac{1}{3} & 0 & 0 & \\ \frac{2}{3} & 0 & \frac{2}{3} & 0 & \\ \hline & \frac{1}{4} & 0 & \frac{3}{4} & \end{array} \quad (37)$$

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)[第 17 页 共 51 页](#)[返回](#)[全屏显示](#)[关闭](#)[退出](#)



取{(32), (36)}的另一组解

$$b_1 = b_3 = \frac{1}{6}, \quad b_2 = \frac{2}{3}, \quad c_1 = 0, \quad c_2 = a_{21} = \frac{1}{2}, \quad c_3 = -a_{31} = 1, \quad a_{32} = 2,$$

得三阶Kutta方法

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 \\ 1 & -1 & 2 & 0 \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array} \quad (38)$$

仿上, 我们还可获得如下四级四阶方法

$$\begin{array}{c|ccccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ \hline & \frac{1}{6} & \frac{1}{3} & \frac{1}{3} & \frac{1}{6} \end{array} \quad \begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & \frac{\sqrt{2}-1}{2} & \frac{2-\sqrt{2}}{2} & 0 & 0 \\ 1 & 0 & -\frac{\sqrt{2}}{2} & \frac{2+\sqrt{2}}{2} & 0 \\ \hline & \frac{1}{6} & \frac{2-\sqrt{2}}{6} & \frac{2+\sqrt{2}}{6} & \frac{1}{6} \end{array} \quad (39)$$

上二方法分别称为经典Runge-Kutta方法与Gill方法。

访问主页

标题页

◀ ▶

◀ ▶

第 18 页 共 51 页

返回

全屏显示

关闭

退出



当构造显式Runge-Kutta方法时，下述由Butcher给出的最大可达阶定理是值得注意的。

定理6.2 一个 s 级显式Runge-Kutta方法的相容阶不可能超过 s ，且不存在五级五阶显式Runge-Kutta法。

§6.2.3 隐式方法

由上节可见，随着方法的级数与阶数的增加，Runge-Kutta方法利用Taylor展式来获得愈来愈加困难，不言而喻，直接利用Taylor展式来获取隐式方法将更加困难。为克服该困难，Butcher于1964年引入了下列简化条件

$$B(p) : \sum_{i=1}^s b_i c_i^{l-1} = \frac{1}{l}, \quad l = 1, 2, \dots, p$$

$$C(\eta) : \sum_{j=1}^s a_{ij} c_j^{l-1} = \frac{c_i^l}{l}, \quad i = 1, 2, \dots, s; \quad l = 1, 2, \dots, \eta$$

$$D(\xi) : \sum_{i=1}^s a_{ij} b_i c_i^{l-1} = \frac{1}{l} b_j (1 - c_j^l), \quad j = 1, 2, \dots, s; \quad l = 1, 2, \dots, \xi,$$

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 19 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



这里, $B(p)$, $C(\eta)$ 分别意味着系统(1)的真解 $y(t)$ 满足

$$y(t_n + h) = y(t_n) + h \sum_{j=1}^s b_j y'(t_n + c_j h) + \mathcal{O}(h^{p+1}),$$

$$y(t_n + c_i h) = y(t_n) + h \sum_{j=1}^s a_{ij} y'(t_n + c_j h) + \mathcal{O}(h^{\eta+1}).$$

基于上述阶简化条件, Butcher给出了Runge-Kutta方法的相容阶判据。

定理6.3 若Runge-Kutta方法(64)满足阶简化条件 $C(\eta)$, $D(\xi)$, 且 p 为满足 $B(p)$ 及不等式

$$p \leq \min\{\eta + \xi + 1, 2\eta + 2\}$$

的最大正整数, 则该方法为 p 阶相容的.

访问主页

标题页

◀ ▶

◀ ▶

第 20 页 共 51 页

返回

全屏显示

关闭

退出

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 21 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

依据阶简化条件，人们将常用隐式Runge-Kutta方法分为六类.

表6.1 六类隐式Runge-Kutta方法

方 法	阶 条 件	相 容 阶
Gauss	$B(2s), C(s), D(s)$	$2s$
Radau IA	$B(2s-1), C(s-1), D(s)$	$2s-1$
Radau IIA	$B(2s-1), C(s), D(s-1)$	$2s-1$
Lobatto IIIA	$B(2s-2), C(s), D(s-2)$	$2s-2$
Lobatto IIIB	$B(2s-2), C(s-2), D(s)$	$2s-2$
Lobatto IIIC	$B(2s-2), C(s-1), D(s-1)$	$2s-2$

利用阶简化条件及定理6.3, 我们可构造隐式Runge-Kutta方法.
如: 对于二级Runge-Kutta方法, 若有 $B(3), C(1), D(1)$ 成立, 即

$$\begin{cases} \sum_{i=1}^2 b_i c_i^{l-1} = \frac{1}{l}, & l = 1, 2, 3; & \sum_{j=1}^2 a_{ij} = c_i, & i = 1, 2, \\ \sum_{i=1}^2 a_{ij} b_i = b_j(1 - c_j), & j = 1, 2, \end{cases} \quad (40)$$

则由定理6.3可知这类二级方法至少是3阶相容的.



特别, 在(40)中若取 $c_1 = 0$, $a_{11} = \frac{1}{4}$, 则可解得

$$c_2 = \frac{3}{2}, \quad b_1 = \frac{1}{4}, \quad b_2 = \frac{3}{4}, \quad a_{11} = \frac{1}{4}, \quad a_{12} = -\frac{1}{4}, \quad a_{21} = \frac{1}{4}, \quad a_{22} = \frac{5}{12},$$

从而得二级三阶Radau IA方法

$$\begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

在Runge-Kutta方法(64)中取 $s = 3$, $c_1 = 0$, $c_2 = \frac{1}{2}$, $c_3 = 1$, 并使 $B(4)$, $C(2)$, $D(2)$ 成立, 则可得三级四阶Lobatto IIIC方法

$$\begin{array}{c|ccc} 0 & \frac{1}{6} & -\frac{1}{3} & \frac{1}{6} \\ \frac{1}{2} & \frac{1}{6} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

访问主页

标题页

◀

▶

◀

▶

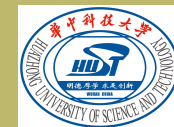
第 22 页 共 51 页

返回

全屏显示

关闭

退出



类似地，依据阶简化条件，我们还可得到其它类型的隐式Runge-Kutta方法。为今后使用方便起见，现将1, 2, 3级各类方法列表如下：

表6.2 2, 4, 6 阶Gauss方法

$\frac{1}{2}$	$\frac{1}{2}$	$\frac{1}{2} - \frac{\sqrt{3}}{6}$	$\frac{1}{4}$	$\frac{1}{4} - \frac{\sqrt{3}}{6}$
		$\frac{1}{2} + \frac{\sqrt{3}}{6}$	$\frac{1}{4} + \frac{\sqrt{3}}{6}$	$\frac{1}{4}$
	1		$\frac{1}{2}$	$\frac{1}{2}$

$\frac{1}{2} - \frac{\sqrt{15}}{10}$	$\frac{5}{36}$	$\frac{2}{9} - \frac{\sqrt{15}}{15}$	$\frac{5}{36} - \frac{\sqrt{15}}{30}$
$\frac{1}{2}$	$\frac{1}{36} + \frac{\sqrt{15}}{24}$	$\frac{2}{9}$	$\frac{5}{36} - \frac{\sqrt{15}}{24}$
$\frac{1}{2} + \frac{\sqrt{15}}{10}$	$\frac{5}{36} + \frac{\sqrt{15}}{30}$	$\frac{2}{9} + \frac{\sqrt{15}}{15}$	$\frac{5}{36}$
	$\frac{5}{18}$	$\frac{4}{9}$	$\frac{5}{18}$

访问主页

标题页



第 23 页 共 51 页

返回

全屏显示

关闭

退出



表6.3 1, 3, 5阶RadauIA方法

$$\begin{array}{c|c} 0 & 1 \\ \hline & 1 \end{array} \qquad \begin{array}{c|cc} 0 & \frac{1}{4} & -\frac{1}{4} \\ \frac{2}{3} & \frac{1}{4} & \frac{5}{12} \\ \hline & \frac{1}{4} & \frac{3}{4} \end{array}$$

$$\begin{array}{c|ccc} 0 & \frac{1}{9} & \frac{-1-\sqrt{6}}{18} & \frac{-1+\sqrt{6}}{18} \\ \frac{6-\sqrt{6}}{10} & \frac{1}{9} & \frac{88+7\sqrt{6}}{360} & \frac{88-43\sqrt{6}}{360} \\ \frac{6+\sqrt{6}}{10} & \frac{1}{9} & \frac{88+43\sqrt{6}}{360} & \frac{88-7\sqrt{6}}{360} \\ \hline & \frac{1}{9} & \frac{16+\sqrt{6}}{36} & \frac{16-\sqrt{6}}{36} \end{array}$$

访问主页

标题页

◀ ▶

◀ ▶

第 24 页 共 51 页

返回

全屏显示

关闭

退出



表6.4 1, 3, 5阶RadauIIA方法

$$\begin{array}{c|c} 1 & 1 \\ \hline & 1 \end{array}$$

$$\begin{array}{c|cc} \frac{1}{3} & \frac{5}{12} & -\frac{1}{12} \\ 1 & \frac{3}{4} & \frac{1}{4} \\ \hline & \frac{3}{4} & \frac{1}{4} \end{array}$$

$\frac{4-\sqrt{6}}{10}$	$\frac{88-7\sqrt{6}}{360}$	$\frac{296-169\sqrt{6}}{1800}$	$\frac{-2+3\sqrt{6}}{225}$
$\frac{4+\sqrt{6}}{10}$	$\frac{296+169\sqrt{6}}{1800}$	$\frac{88+7\sqrt{6}}{360}$	$\frac{-2-3\sqrt{6}}{225}$
1	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$	$\frac{1}{9}$
	$\frac{16-\sqrt{6}}{36}$	$\frac{16+\sqrt{6}}{36}$	$\frac{1}{9}$

访问主页

标题页



第 25 页 共 51 页

返回

全屏显示

关闭

退出



表6.5 2, 4, 6阶LobattoIIIA方法

$$\begin{array}{c|cc} 0 & 0 & 0 \\ 1 & \frac{1}{2} & \frac{1}{2} \\ \hline & \frac{1}{2} & \frac{1}{2} \end{array}$$

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0 \\ \frac{1}{2} & \frac{5}{24} & \frac{1}{3} & -\frac{1}{24} \\ 1 & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \\ \hline & \frac{1}{6} & \frac{2}{3} & \frac{1}{6} \end{array}$$

$$\begin{array}{c|cccc} 0 & 0 & 0 & 0 & 0 \\ \frac{5-\sqrt{5}}{10} & \frac{11+\sqrt{5}}{120} & \frac{25-\sqrt{5}}{120} & \frac{25-13\sqrt{5}}{120} & \frac{-1+\sqrt{5}}{120} \\ \frac{5+\sqrt{5}}{10} & \frac{11-\sqrt{5}}{120} & \frac{25+13\sqrt{5}}{120} & \frac{25+\sqrt{5}}{120} & \frac{-1-\sqrt{5}}{120} \\ 1 & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12} \\ \hline & \frac{1}{12} & \frac{5}{12} & \frac{5}{12} & \frac{1}{12} \end{array}$$

访问主页

标题页



第 26 页 共 51 页

返回

全屏显示

关闭

退出

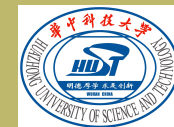


表6.6 2, 4, 6阶LobattoIIIB型方法

0	$\frac{1}{2}$	0
1	$\frac{1}{2}$	0
<hr/>		$\frac{1}{2}$
	$\frac{1}{2}$	$\frac{1}{2}$

0	$\frac{1}{6}$	$-\frac{1}{6}$	0
$\frac{1}{2}$	$\frac{1}{6}$	$\frac{1}{3}$	0
1	$\frac{1}{6}$	$\frac{5}{6}$	0
<hr/>			$\frac{1}{6}$
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

0	$\frac{1}{12}$	$\frac{-1-\sqrt{5}}{24}$	$\frac{-1+\sqrt{5}}{24}$	0
$\frac{5-\sqrt{5}}{10}$	$\frac{1}{12}$	$\frac{25+\sqrt{5}}{120}$	$\frac{25-13\sqrt{5}}{120}$	0
$\frac{5+\sqrt{5}}{10}$	$\frac{1}{12}$	$\frac{25+13\sqrt{5}}{120}$	$\frac{25-\sqrt{5}}{120}$	0
1	$\frac{1}{12}$	$\frac{11-\sqrt{5}}{24}$	$\frac{11+\sqrt{5}}{24}$	0
<hr/>				$\frac{1}{12}$
	$\frac{1}{12}$	$\frac{5}{12}$	$\frac{5}{12}$	$\frac{1}{12}$

访问主页

标题页



第 27 页 共 51 页

返回

全屏显示

关闭

退出

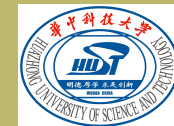


表6.7 2, 4, 6 阶LobattoIIIC方法

0	$\frac{1}{2}$	$\frac{1}{2}$
1	$\frac{1}{2}$	$\frac{1}{2}$
<hr/>		
	$\frac{1}{2}$	$\frac{1}{2}$

0	$\frac{1}{6}$	$-\frac{1}{3}$	$\frac{1}{6}$
$\frac{1}{2}$	$\frac{1}{6}$	$\frac{5}{12}$	$-\frac{1}{12}$
1	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$
<hr/>			
	$\frac{1}{6}$	$\frac{2}{3}$	$\frac{1}{6}$

0	$\frac{1}{12}$	$-\frac{\sqrt{5}}{12}$	$\frac{\sqrt{5}}{12}$	$-\frac{1}{12}$
$\frac{5-\sqrt{5}}{10}$	$\frac{1}{12}$	$\frac{1}{4}$	$\frac{10-7\sqrt{5}}{60}$	$\frac{\sqrt{5}}{60}$
$\frac{5+\sqrt{5}}{10}$	$\frac{1}{12}$	$\frac{10+7\sqrt{5}}{60}$	$\frac{1}{4}$	$-\frac{\sqrt{5}}{60}$
1	$\frac{1}{12}$	$\frac{5}{12}$	$\frac{5}{12}$	$\frac{1}{12}$
<hr/>				
	$\frac{1}{12}$	$\frac{5}{12}$	$\frac{5}{12}$	$\frac{1}{12}$

访问主页

标题页



第 28 页 共 51 页

返回

全屏显示

关闭

退出

§6.3 数值算法理论

§6.1, §6.2中各类算法写成统一的一般多步法形式

$$\sum_{i=0}^k \alpha_i y_{n+i} = h \varphi_f(t_n; y_n, y_{n+1}, \dots, y_{n+k}; h), \quad n = 0, 1, 2, \dots, N - k, \quad (41)$$

其中 $h > 0$ 为积分步长, $k \geq 1$ 是方法的步数, 诸 α_i 为实常数, 且 $\alpha_k \neq 0$, $t_n = a + nh$, $Nh \leq b - a$, $y_n \approx y(t_n)$, $\varphi_f : D_H \rightarrow R$ 是依赖于问题(1)的右函数 f 的映射, 其中

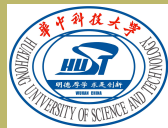
$$D_H = \{(t_n; y_n, y_{n+1}, \dots, y_{n+k}; h) \mid 0 < h \leq H, a \leq t_n \leq b - kh, y_{n+i} \in R\},$$

$H > 0$ 是适当选取的常数。恒设

1. 当 $f \equiv 0$ 时, $\varphi_f \equiv 0$;
2. 当 f 满足(2)时, φ_f 满足Lipschitz条件

$$\begin{aligned} & |\varphi_f(t_n; y_n, y_{n+1}, \dots, y_{n+k}; h) - \varphi_f(t_n; z_n, z_{n+1}, \dots, z_{n+k}; h)| \\ & \leq L_\varphi \sum_{i=0}^k |y_{n+i} - z_{n+i}|, \quad 0 < h \leq h_\varphi, \end{aligned} \quad (42)$$

这里 h_φ 和 L_φ 通常依赖于(2)中的Lipschitz常数 L .

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)[第 29 页 共 51 页](#)[返回](#)[全屏显示](#)[关闭](#)[退出](#)



理论分析表明：只要 $h < \min \left\{ \frac{|\alpha_k|}{L_\varphi}, h_\varphi \right\}$ ，则从任意初值 y_0, y_1, \dots, y_{k-1} 出发，方程(41)可唯一地确定问题(1)的一个数值解序列 $\{y_n\}_{n=0}^N$ 。方法(41)是一类非常广泛的方法，它几乎包含当今各种常用的各种常微分方程数值方法。如取

$$\varphi_f = \sum_{i=0}^k \beta_i f(t_n + ih, y_{n+i}),$$

则得线性多步法(20)；若取

$$\varphi_f = \sum_{j=1}^s b_j f(t_n + c_j h, Y_{j,n}),$$

则得 s 级 Runge-Kutta 方法(64)。此外，(41)也包括 §6.4 中将要介绍的预校算法及迄今为止尚待开发的各种方法。本节将涉及该类方法的相容性、稳定性及收敛性。

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 30 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



§6.3.1 相容性

§6.1和§6.2中局部截断误差及相容阶概念可统一定义如下:

定义6.1 设 f 满足(2), 若问题(1)的真解 $y(t)$ 满足

$$\sum_{i=0}^k \alpha_i y(t_{n+i}) = h\varphi_f(t_n; y(t_n), y(t_{n+1}), \dots, y(t_{n+k}); h) + T_{n+k},$$

$$n = 0, 1, 2, \dots, N - k, \quad (43)$$

则称其余项 T_{n+k} 为方法(41)在点 t_{n+k} 处的局部截断误差。进一步, 若 p 为满足

$$\max_{0 \leq n \leq N-k} |T_{n+k}| = \mathcal{O}(h^{p+1}), \quad h \rightarrow 0 \quad (44)$$

的最大正数, 则称方法(41)是 p 阶相容的。特别若

$$\frac{1}{h} \max_{0 \leq n \leq N-k} |T_{n+k}| \rightarrow 0, \quad h \rightarrow 0, \quad (45)$$

则称方法(41)为相容的。

访问主页

标题页

◀

▶

◀

▶

第 31 页 共 51 页

返回

全屏显示

关闭

退出



定理6.4 方法(41)相容的充要条件是 $\rho(1) = 0$ ，且下列等式关于 n 一致成立

$$\lim_{h \rightarrow 0} \varphi_f(t_n; y(t_n), y(t_{n+1}), \dots, y(t_{n+k}); h) = \rho'(1)y'(t_n), \quad (46)$$

其中 $\rho(\xi) = \sum_{i=0}^k \alpha_i \xi^i$ ($\xi \in C$)。

由定理6.4可直接推得

推论6.1 线性多步法(20)相容的充要条件是

$$\rho(1) = 0, \quad \rho'(1) = \sigma(1),$$

其中 $\sigma(\xi) = \sum_{i=0}^k \beta_i \xi^i$ ($\xi \in C$)。

推论6.2 Runge-Kutta方法(64)相容的充要条件是 $\sum_{j=1}^s b_j = 1$ 。

访问主页

标题页

◀ ▶

◀ ▶

第 32 页 共 51 页

返回

全屏显示

关闭

退出

§6.3.2 稳定性

定义6.2 方法(41)称为是**零稳定的**, 若对任意满足Lipschitz条件的问题(1), 存在正常数 $C, h_0 > 0$ 使当 $0 < h \leq h_0$ 时, 方法(41)的任一解序列 $\{y_n\}$ 与相应的扰动问题 (设诸 ε_n 为任给扰动)

$$\begin{cases} \sum_{i=0}^k \alpha_i z_{n+i} = h[\varphi_f(t_n; z_n, \dots, z_{n+k}; h) + \varepsilon_{n+k}], & n = 0, 1, \dots, N - k, \\ z_j = y_j + \varepsilon_j, & j = 0, 1, \dots, k - 1, \end{cases}$$

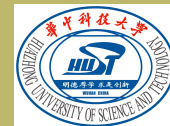
的解序列 $\{z_n\}$ 满足

$$\max_{0 \leq n \leq N} \|z_n - y_n\| \leq C \max_{0 \leq n \leq N} \|\varepsilon_n\|.$$

上述稳定性概念充分刻画了当步长 h 充分小时计算过程中扰动对数值解的影响。显然, 依据定义6.2来判断零稳定是困难的, 下述结论可在一定程度上简化其判定。

定理6.5 方法(41)为零稳定的充要条件是多项式 $\rho(\xi)$ 的每个根的模不超过1, 且模为1的根是单根。

方法(41)满足上述充要条件也称其符合**根条件**。由于对于任何单步方法有 $\rho(\xi) = \xi - 1$, 因此单步方法必满足根条件, 即是零稳定的。进而可知, Runge-Kutta方法均是零稳定。

[访问主页](#)[标题页](#)[<<](#) [>>](#)[<](#) [>](#)

第 33 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



定理6.6 (Dahlquist) 零稳定的线性 k 步法(20)的相容阶 p 满足

$$p \leq \begin{cases} k+2, & \text{若 } k \text{ 是偶数;} \\ k+1, & \text{若 } k \text{ 是奇数;} \\ k, & \text{若 } \frac{\beta_k}{\alpha_k} \leq 0. \end{cases}$$

由定理6.6可知, 零稳定的显式线性 k 步法的相容阶不可能超过 k 阶。零稳定描述的是 $h \rightarrow 0$ 时的方法的稳定性, 然而实际计算中, 步长总是固定的, 为刻画这种情况的稳定性态, 下面引入绝对稳定性概念。

定义6.3 方法(41)称为是**绝对稳定的**, 若该方法应用于线性标量试验方程

$$y' = \lambda y, \quad \lambda \in C \quad (47)$$

时, 其解满足 $\lim_{n \rightarrow \infty} y_n = 0$. 相应地, 称集合

$$S = \{h\lambda \in C \mid \text{方法(41)应用于方程(47)时满足 } \lim_{n \rightarrow \infty} y_n = 0\}$$

为方法(41)的**绝对稳定域**。

访问主页

标题页

◀ ▶

◀ ▶

第 34 页 共 51 页

返回

全屏显示

关闭

退出



作为范例, 考虑线性多步法(20)和Runge-Kutta方法(64), 当其应用于(47)时, 分别产生下列差分方程

$$\sum_{i=0}^k \alpha_i y_{n+i} = \hat{h} \sum_{i=0}^k \beta_i y_{n+i}, \quad (48)$$

$$y_{n+1} = [1 + \hat{h}b^T(I - \hat{h}A)^{-1}e]y_n, \quad (49)$$

其中, $\hat{h} = h\lambda$, I 为 s 级单位阵, $e = (1, 1, \dots, 1)^T \in R^s$. 若设上述差分方程有形如 $y_n = \xi^n$ ($\xi \neq 0$)的解, 将其代入则分别得其特征方程为

$$\rho(\xi) = \hat{h}\sigma(\xi), \quad (50)$$

$$\xi = 1 + \hat{h}b^T(I - \hat{h}A)^{-1}e. \quad (51)$$

由此可知, 线性多步法(20)和Runge-Kutta方法(64)的绝对稳定域分别为

$$\mathbb{S}_{LM} = \{\hat{h} \in C \mid \text{方程 } \rho(\xi) = \hat{h}\sigma(\xi) \text{ 的根模满足 } |\xi| < 1\},$$

$$\mathbb{S}_{RK} = \{\hat{h} \in C \mid |1 + \hat{h}b^T(I - \hat{h}A)^{-1}e| < 1\}.$$

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 35 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



§6.3.3 相容性

当以方法(41)求解满足Lipschitz 条件的问题(1) 时, 若产生唯一的逼近序列 $\{y_n\}$, 则称误差 $\varepsilon_n := y(t_n) - y_n$, ($n = 0, 1, \dots, N$), 为方法(41)的**整体截断误差**。

定义6.5 方法(41)称为 **p 阶收敛的**, 若以该方法求解任意满足Lipschitz 条件的问题(1), 当 f 充分连续可微且

$$\max_{0 \leq j \leq k-1} |y(t_j) - y_j| = \mathcal{O}(h^p), \quad h \rightarrow 0$$

时, 有 $\varepsilon_n = \mathcal{O}(h^p)$, $h \rightarrow 0$ 。特别, 方法称为是**收敛的**, 若 $\forall t \in [a, b]$, 有

$$y_n \rightarrow y(t), \quad \text{当 } h \rightarrow 0, a + nh \rightarrow t, y_i \rightarrow y_0 \ (0 \leq i \leq k-1).$$

定理6.7 若方法(41)相容, 则其零稳定性与收敛性等价。

定理6.8 若方法(41) p 阶相容且零稳定, 则该方法是 **p 阶收敛的**。

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)

第 36 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



由推论6.1, 定理6.1, 6.5, 6.7及定理6.8可分别推得

推论6.3 若线性多步法(20)满足根条件及条件

$$\rho(1) = 0, \quad \rho'(1) = \sigma(1),$$

则该方法是收敛的。

推论6.4 若线性 k 步法(20)满足根条件及(18), 且 $C_{p+1} \neq 0$, 则该方法是 p 阶收敛的。

由于Runge-Kutta方法属单步法, 则必零稳定。据此, 推论6.2及定理6.3, 6.7, 6.8可分别推得

推论6.3 若Runge-Kutta方法(64)满足 $\sum_{j=1}^s b_j = 1$ 则该方法收敛。

推论6.4 若Runge-Kutta方法(64)满足阶简化条件 $C(\eta)$, $D(\xi)$, 且 p 为满足 $B(p)$ 及不等式

$$p \leq \min\{\eta + \xi + 1, 2\eta + 2\}$$

的最大正整数, 则该方法为 p 阶收敛的。

访问主页

标题页

« »

◀ ▶

第 37 页 共 51 页

返回

全屏显示

关闭

退出



§6.4 数值方法的有效实现

一个微分方程数值方法构造出来后，要想真正在计算机上有效实现，求出合乎原问题要求的数值解，还需克服许多困难，如误差如何估计，步长及计算初值如何选取，隐式方法及多步方法如何处理等。在这些问题中，方法的阶与步长的选择是数值求解微分方程的首要问题，它直接影响计算精度和计算量。对于方法的阶，我们一般要求其不超过原问题真解的可微次数。但是，我们也必须注意到：高阶方法的稳定性能通常较差，因此阶数很高的方法一般不适用。对于方法步长的选取，从减少截断误差的角度来看，应采用小步长，但是步长取得过小，计算量将增大，而引起大的舍入误差。为此在步长选择时，我们必须二者兼顾，合理选取。通常可利用误差主项来确定步长，即要求所取步长使方法的误差主项不超过预定精度 ε 且接近 ε ，据此在计算中调整步长。与此同时，我们也要求步长符合稳定性要求。数值方法主要分为显式方法与隐式方法，从计算实现来说，显式方法易于处理。为此，本节主要介绍隐式方法的有效实现，其主要倚重二类技巧，一是迭代技巧，二是预估-校正技巧。以下，我们将应用这二种技巧于隐式线性多步法和隐式Runge-Kutta方法，从而实现隐式方法有效求解常微分方程初值问题。

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)

第 38 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

§6.4.1 迭代技巧

对于隐式线性多步法

$$y_{n+k} = h\beta_k f(t_{n+k}, y_{n+k}) + \omega_n, \quad (52)$$

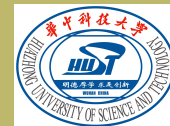
其中 $\beta_k \neq 0, \omega_n = \sum_{i=0}^{k-1} (h\beta_i f_{n+i} - \alpha_i y_{n+i})$, 我们采用Jacobi迭代

$$y_{n+k}^{(r+1)} = h\beta_k f(t_{n+k}, y_{n+k}^{(r)}) + \omega_n, \quad r = 0, 1, 2, \dots, \quad (53)$$

其中迭代初值 $y_{n+k}^{(0)}$ 及计算启动值 y_0, y_1, \dots, y_{k-1} 可通过与该方法同阶的显式单步方法获得。若欲求解的问题(1)满足Lipschitz条件(2), 则由定理2.1可知: 当步长满足 $h < \frac{1}{|\beta_k|L}$ 时, 其迭代格式(53)是收敛的, 且有事后误差估计

$$|y_{n+k}^{(r+1)} - y_{n+k}| \leq \frac{1}{1 - h\beta_k L} |y_{n+k}^{(r+1)} - y_{n+k}^{(r)}|.$$

因此, 当步长足够小时, 我们可用 $|y_{n+k}^{(r+1)} - y_{n+k}^{(r)}| < \varepsilon$ 作为计算终止准则, 其中 ε 为预定精度. 当上准则成立时, 则计算终止, 且取当前步的数值解为 $y_{n+k} \approx y_{n+k}^{(r+1)}$ 。

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 39 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



隐式线性多步法(52)也可采用Newton 迭代法来处理, 应用Newton 迭代法于(52)有

$$y_{n+k}^{(r+1)} = y_{n+k}^{(r)} - \left[1 - h\beta_k \frac{\partial f(t_{n+k}, y_{n+k}^{(r)})}{\partial y} \right]^{-1} \left[y_{n+k}^{(r)} - h\beta_k f(t_{n+k}, y_{n+k}^{(r)}) - \omega_n \right],$$
$$r = 0, 1, 2, \dots, \quad (54)$$

其计算终止准则可取为

$$|y_{n+k}^{(r+1)} - y_{n+k}^{(r)}| < \varepsilon \quad \text{或} \quad \left| y_{n+k}^{(r+1)} - h\beta_k f(t_{n+k}, y_{n+k}^{(r+1)}) - \omega_n \right| < \eta,$$

其中 ε, η 为预定精度. 当上准则之一成立时, 则计算终止, 且取当前步的数值解为 $y_{n+k} \approx y_{n+k}^{(r+1)}$ 。

[访问主页](#)[标题页](#)[<<](#)[>>](#)[<](#)[>](#)

第 40 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 41 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

对于隐式Runge-Kutta方法(64), 若采用Jacobi迭代法, 则得

$$Y_{i,n}^{(r+1)} = y_n + h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{j,n}^{(r)}), \quad i = 1, 2, \dots, s; \quad r = 0, 1, 2, \dots, \quad (55)$$

其每步迭代初值可取为 $Y_{j,n}^{(0)} \approx y_n + c_j h f_n$ ($j = 1, 2, \dots, s$)。又一次由定理2.1及Lipschitz条件可知: 当步长 h 满足 $h < \frac{1}{L \sum_{j=1}^s |a_{ij}|}$ 时, 迭代

格式(55)收敛, 且有事后误差估计

$$|Y_{i,n}^{(r+1)} - Y_{i,n}^{(r)}| \leq \frac{1}{1 - hL \sum_{j=1}^s |a_{ij}|} |Y_{i,n}^{(r+1)} - Y_{i,n}^{(r)}|, \quad i = 1, 2, \dots, s.$$

因此, 当步长足够小时, 我们可用 $|Y_{i,n}^{(r+1)} - Y_{i,n}^{(r)}| < \varepsilon$ 作为计算终止准则, 其中 ε 为预定精度. 当上准则成立时, 则取当前步的级值为 $Y_{i,n} \approx Y_{i,n}^{(r+1)}$ ($i = 1, 2, \dots, s$), 将其代入(64)的第二式即获当前步的数值解 y_{n+1} 。



若应用Newton迭代法到(52), 则得

$$Y_{i,n}^{(r+1)} = Y_{i,n}^{(r)} - \left[1 - h a_{ii} \frac{\partial f(t_n + c_i h, Y_{i,n}^{(r)})}{\partial y} \right]^{-1} \left[Y_{i,n}^{(r)} - y_n - h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{j,n}^{(r)}) \right],$$

$$i = 1, 2, \dots, s; r = 0, 1, 2, \dots, (56)$$

其每步迭代初值取为 $Y_{j,n}^{(0)} \approx y_n + c_j h f_n$ ($j = 1, 2, \dots, s$), 计算终止准则可取为

$$|Y_{i,n}^{(r+1)} - Y_{i,n}^{(r)}| < \varepsilon \quad \text{或} \quad \left| Y_{i,n}^{(r)} - y_n - h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{j,n}^{(r)}) \right| < \eta,$$

其中 ε, η 为预定精度. 当上准则之一成立时, 则取当前步的级值为 $Y_{i,n} \approx Y_{i,n}^{(r+1)}$ ($i = 1, 2, \dots, s$), 将其代入(64)的第二式即获当前步的数值解 y_{n+1} .

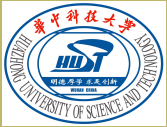
[访问主页](#)
[标题页](#)
[«](#) [»](#)
[◀](#) [▶](#)

第 42 页 共 51 页

[返回](#)
[全屏显示](#)
[关闭](#)
[退出](#)

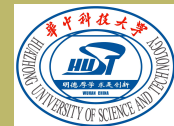
算法 6.1 四阶Gauss方法

```
function z=Gauss(a,b,y0,h)
t0=a; s=2; A=[1/4 1/4-sqrt(3)/6;1/4+sqrt(3)/6 1/4];
B=[1/2 1/2]; c=[1/2-sqrt(3)/6;1/2+sqrt(3)/6];
for n=a+h:h:b
    err1=1; err2=1; t=kron(ones(s,1),t0)+h*c;
    Y0=kron(ones(s,1),y0)+h*diag(c)*kron(ones(s,1),f(t0,y0));
    while err1 >=10^(-10) & err2 >=10^(-10)
        r=Y0-kron(ones(s,1),y0)-h*A*[f(t(1),Y0(1));f(t(2),Y0(2))];
        Y1=Y0-inv(eye(s)-h*diag(diag(A)))*...
            [df(t(1),Y0(1)) 0;0 df(t(2),Y0(2))])*r;
        err1=norm(Y1-Y0); err2=norm(r); Y0=Y1;
    end
    y1=y0+h*B*[f(t(1),Y0(1));f(t(2),Y0(2))];
    hold on;
    plot(t0+h,y1,'b*')
    hold off;
    xlabel('t'); ylabel('y');
    t0=t0+h; y0=y1;
end
err
```

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 43 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



例6.1 试取步长 $h = 0.01$, 应用四阶Gauss方法计算初值问题

$$\begin{cases} y'(t) = y(t) \ln[1+y(t)] - \exp(-t)[1+(1+\exp(t)) \ln(2+\exp(-t))], & t \in [0, 5], \\ y(0) = 2, \end{cases} \quad (57)$$

并求最大整体误差 $\max_{0 \leq n \leq 500} |y(t_n) - y_n|$. (精确解 $y(t) = 1 + \exp(-t)$).

解 取步长 $h = 0.01$, 应用四阶Gauss方法于初值问题(57). 计算过程采用Newton迭代技巧, 每步迭代初值取为 $Y_{j,n}^{(0)} \approx y_n + c_j h f_n$ ($j = 1, 2$), 其中 $c_1 = \frac{1}{2} - \frac{\sqrt{3}}{6}$, $c_2 = \frac{1}{2} + \frac{\sqrt{3}}{6}$. 据此得该初值问题的数值解 (见图6.1), 其最大整体误差 $\max_{0 \leq n \leq 500} |y(t_n) - y_n| = 2.5720e-008$.

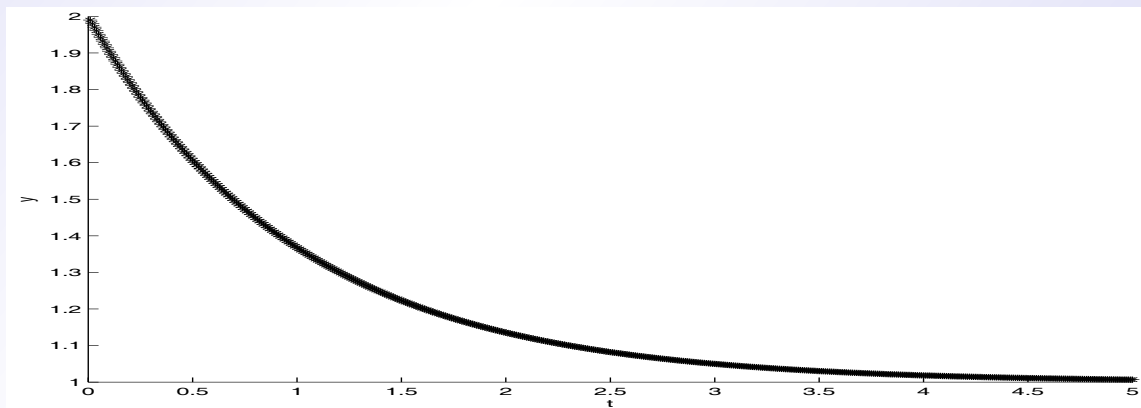


图 6.1 四阶Gauss方法应用于初值问题(57)的数值解.

访问主页

标题页

◀ ▶

◀ ▶

第 44 页 共 51 页

返回

全屏显示

关闭

退出

§6.4.2 预估-校正技巧

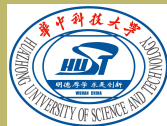
隐式方法的另一个有效处理途径是预估-校正方法。理论分析表明，迭代法中的初始迭代值若以与迭代公式同阶或低一阶的公式计算，则至多只需一次迭代，其迭代值精度即可与迭代公式本身的固有精度同阶。为此，我们可用与迭代公式同阶的显式公式先计算迭代初值（并称此显式公式为预估公式），然后用迭代公式（称之为校正式）修正一次，即可获得其数值解。此方法称为**预估-校正方法**。如我们以二步三阶显式公式(22)作为预估式，而以三阶Adams-Moulton公式(23)作为校正式，则可得预校算法

$$\begin{cases} \text{预估: } \tilde{y}_{n+2} = -4y_{n+1} + 5y_n + 2h[2f_{n+1} + f_n], \\ \text{校正: } y_{n+2} = y_{n+1} + \frac{h}{12}[5f(t_{n+2}, \tilde{y}_{n+2}) + 8f_{n+1} - f_n]. \end{cases} \quad (58)$$

其中启动值 y_0, y_1 分别由原问题初值及3阶Heun方法获得。

为进一步提高预校算法的精度，我们通常以预估式及校正式的误差估计量来分别修正预估值和校正值。如在式(58)中，若取 $y_{n+1} \approx y(t_{n+1}), y_n \approx y(t_n)$ ，则根据(19)有

$$y(t_{n+2}) - \tilde{y}_{n+2} \approx \frac{1}{6}h^4 y^{(4)}(t_n), \quad y(t_{n+2}) - y_{n+2} \approx -\frac{1}{24}h^4 y^{(4)}(t_n).$$

[访问主页](#)[标题页](#)[«](#) [»](#)[◀](#) [▶](#)

第 45 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 46 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)

由上两式可得其事后误差估计式

$$y(t_{n+2}) - \tilde{y}_{n+2} \approx \frac{4}{5}(y_{n+2} - \tilde{y}_{n+2}), \quad y(t_{n+2}) - y_{n+2} \approx -\frac{1}{5}(y_{n+2} - \tilde{y}_{n+2}), \quad (59)$$

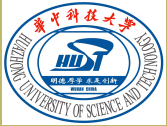
记 $p_{n+2}, m_{n+2}, c_{n+2}$ 分别为第 $n+1$ 个计算步的预估值, 修正值及校正值, 且分别以式(59)的右端作为预估值与校正值的修正量, 则可得如下预估-改进-校正方法

$$\left\{ \begin{array}{l} \text{预估: } p_{n+2} = -4y_{n+1} + 5y_n + 2h(2f_{n+1} + f_n), \\ \text{改进: } m_{n+2} = p_{n+2} + \frac{4}{5}(c_{n+1} - p_{n+1}), \\ \quad F_{n+2} = f(t_{n+2}, m_{n+2}), \\ \text{校正: } c_{n+2} = y_{n+1} + \frac{h}{12}(5F_{n+2} + 8f_{n+1} - f_n), \\ \text{改进: } y_{n+2} = c_{n+2} - \frac{1}{5}(c_{n+2} - p_{n+2}), \\ \quad f_{n+2} = f(t_{n+2}, y_{n+2}). \end{array} \right. \quad (60)$$

上述方案中, 计算预估式的改进值 m_{n+2} 时采用的是前一步的修正量 $\frac{4}{5}(c_{n+1} - p_{n+1})$, 其原因是当前步的校正值 c_{n+2} 此时还未算出。此外, 在启动方案(60)时, 我们可取 $c_1 - p_1 = 0$ 。

算法 6.2 预估-改进-校正方法

```
function z=pmcm(a,b,y0,h)
    t0=a; c1=0; p1=0; A=[0 0 0;1/3 0 0;0 2/3 0];
    B=[1/4 0 3/4]; c=[0 1/3 2/3]'; tc1=t0+c(1)*h;
    tc2=t0+c(2)*h; tc3=t0+c(3)*h; Y1=y0;
    Y2=y0+h*A(2,1)*f(tc1,Y1);
    Y3=y0+h*A(3,1:2)*[f(tc1,Y1);f(tc2,Y2)];
    y1=y0+h*B(1:3)*[f(tc1,Y1);f(tc2,Y2);f(tc3,Y3)];
    for n=a+2*h:h:b
        t1=t0+h; t2=t0+2*h;
        p2=(-4)*y1+5*y0+2*h*(2*f(t1,y1)+f(t0,y0));
        m2=p2+4*(c1-p1)/5; F2=f(t2,m2);
        c2=y1+h*(5*F2+8*f(t1,y1)-f(t0,y0))/12;
        y2=c2-(c2-p2)/5; f2=f(t2,y2);
        hold on;
        plot(t2,y2(1),'b*')
    hold off;
    xlabel('t'); ylabel('x');
    t0=t1; y0=y1; y1=y2; c1=c2; p1=p2;
end
```

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 47 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



例6.2 试取步长 $h = 0.01$, 应用预估-改进-校正方法(60) 计算初值问题(57), 并求最大整体误差 $\varepsilon := \max_{0 \leq n \leq 500} |y(t_n) - y_n|$.

解 取步长 $h = 0.01$, 应用预估-改进-校正方法(60)于初值问题(57), 计算初值分别由问题初值及3阶Heun方法获得。据此得该初值问题的数值解（见图6.2），其最大整体误差 $\max_{0 \leq n \leq 500} |y(t_n) - y_n| = 8.5936e - 004$ 。

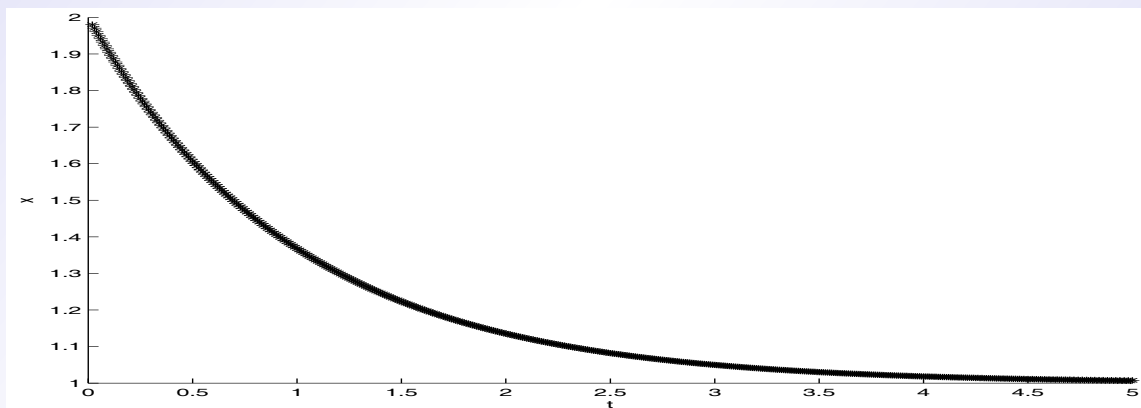


图 6.2 预估-改进-校正方法(60)应用于初值问题(57)的数值解.

访问主页

标题页

◀ ▶

◀ ▶

第 48 页 共 51 页

返回

全屏显示

关闭

退出



§6.5 微分方程组的数值处理

在实际工程技术与科学研究工作中，我们常常会遇到微分方程组初值问题

$$\begin{cases} y'(t) = f(t, y(t)), & t \in [a, b], \\ y(a) = y_0, & y_0 \in R^m, \end{cases} \quad (61)$$

其中

$$y(t) = [y_1(t), y_2(t), \dots, y_m(t)]^T \quad y'(t) = [y'_1(t), y'_2(t), \dots, y'_m(t)]^T$$

$$f(t, y(t)) = [f_1(t, y(t)), f_2(t, y(t)), \dots, f_m(t, y(t))]^T \in \mathbb{R}^m.$$

如: 对于 m 阶非线性标量微分方程初值问题

$$\begin{cases} x^{(m)}(t) = F(t; x^{(m-1)}(t), \dots, x'(t), x(t)) & t \in [a, b], \\ x^{(i-1)}(a) = x_{i-1}, & i = 1, 2, \dots, m, \end{cases} \quad (62)$$

我们通过做变换

$$x^{(i)}(t) = y_{i+1}(t), \quad i = 0, 1, \dots, m-1,$$

[访问主页](#)[标题页](#)[◀](#) [▶](#)[◀](#) [▶](#)

第 49 页 共 51 页

[返回](#)[全屏显示](#)[关闭](#)[退出](#)



可将(62)化为形如(61)的一阶 m 维微分方程组问题，这里

$$f(t, y(t)) = \begin{pmatrix} y_2(t) \\ y_3(t) \\ \vdots \\ y_m(t) \\ F(t; y_m(t), y_{m-1}(t), \dots, y_1(t)) \end{pmatrix}, \quad y_0 = \begin{pmatrix} x_0 \\ x_1 \\ \vdots \\ x_{m-1} \end{pmatrix}$$

先前介绍的各种数值方法可拓展应用于微分方程组(62)。如记:

$$t_{n+i} = a + (n+i)h, \quad y_{n+i,j} \approx y_j(t_{n+i}), \quad f_{n+i,j} \approx y'_j(t_{n+i}), \quad j = 1, 2, \dots, m,$$

$$y_{n+i} = (y_{n+i,1}, y_{n+i,2}, \dots, y_{n+i,m})^T (\approx y(t_{n+i})), \quad i = 0, 1, \dots, k,$$

$$f_{n+i} = (f_{n+i,1}, f_{n+i,2}, \dots, f_{n+i,m})^T (\approx y'(t_{n+i})), \quad i = 0, 1, \dots, k,$$

则先前的线性 k 步方法可拓展为求解微分方程组(62)的 k 步方法

$$\sum_{i=0}^k \alpha_i y_{n+i} = h \sum_{i=0}^k \beta_i f_{n+i}, \quad (63)$$

其中 $h > 0$ 为步长，诸系数 α_i, β_i 为实数，且 $\alpha_k \neq 0, \alpha_0^2 + \beta_0^2 \neq 0$ 。

访问主页

标题页

◀

▶

◀

▶

第 50 页 共 51 页

返回

全屏显示

关闭

退出



若记:

$$t_n = a + nh, \quad Y_{i,j,n} \approx y_j(t_n + c_i h), \quad i = 1, 2, \dots, s; \quad j = 1, 2, \dots, m,$$

$$Y_{i,n} = (Y_{i,1,n}, Y_{i,2,n}, \dots, Y_{i,m,n})^T (\approx y(t_n + c_i h)), \quad y_n \approx y(t_n),$$

则先前的Runge-Kutta方法可拓展为

$$\begin{cases} Y_{i,n} = y_n + h \sum_{j=1}^s a_{ij} f(t_n + c_j h, Y_{j,n}), & i = 1, 2, \dots, s, \\ y_{n+1} = y_n + h \sum_{j=1}^s b_j f(t_n + c_j h, Y_{j,n}), & n \geq 0, \end{cases} \quad (64)$$

其中诸系数 a_{ij}, b_j 及横标 c_j 为实数, $h > 0$ 为步长。

此外, 先前的算法理论、相关概念及实现技巧也完全可拓广到方程组情形。

访问主页

标题页



第 51 页 共 51 页

返回

全屏显示

关闭

退出