



浙江大学  
ZheJiang University

# 8 Solution of Differential Equations

## 第8章 微分方程求解



# 上节课知识回顾

## 1. 龙贝格 (Romberg) 积分的设计思路是什么？

用若干个精度较低的积分近似值来推算更精确近似值的方法

通过缩小步长，构建龙贝格积分表格

## 2. 数值微分的工程应用和基本思路？

差商近似，通过2点求斜率，包括：向前差商、向后差商、中心差商

通过泰勒公式展开，利用 $n$ 个数值点，消除余项，计算函数的 $n$ 阶导数

## 3. 插值多项式数值微分是怎么提出的？

利用多项式函数的导数容易求得，对 $n$ 个数值点进行

# 问题来源-微分方程

根据牛顿第二运动定律

$$mg \sin \theta = ma = -ml \frac{d^2 \theta}{dt^2}$$

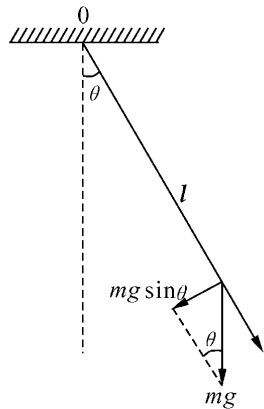
既  $ml \frac{d^2 \theta}{dt^2} = -mg \sin \theta$

即  $\frac{d^2 \theta}{dt^2} + \frac{g}{l} \sin \theta = 0$  齐次微分方程

当  $\theta$  很小时,  $\sin \theta \approx \theta$ , 令  $\omega^2 = \frac{g}{l}$

则  $\frac{d^2 \theta}{dt^2} + \omega^2 \theta = 0$   

二阶常系数线性齐次微分方程



方程的解为

$$\theta = A \sin(\omega t + \varphi)$$

简谐运动

# 问题来源-微分方程

$$\frac{dy}{dt} = 1 - e^{-t}$$



$$y(t) = t + e^{-t} + C$$

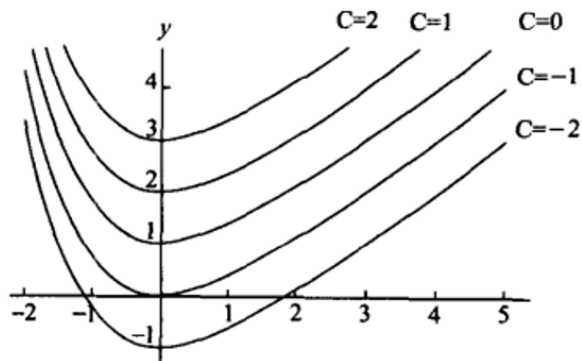


图 9.2 解曲线  $y(t) = t + e^{-t} + C$

对于一个常微分方程：

$$y' = \frac{dy}{dx} = f(x, y)$$

通常会有无穷个解

# 问题来源-微分方程

对于一个常微分方程  $y' = \frac{dy}{dx} = f(x, y)$  通常会有无穷个解。

$$\frac{dy}{dx} = \cos(x) \Rightarrow y = \sin(x) + a, \forall a \in R$$

因此，我们要加入一个限定条件。通常会在端点给出，如下面的初值问题：

$$\begin{cases} \frac{dy}{dx} = f(x, y) & , x \in [a, b] \\ y(a) = y_0 \end{cases}$$

为了使解存在唯一，一般要加限制条件在 $f$ 上，要求 $f$ 对 $y$ 满足**利普希茨Lipschitz条件**：

$$|f(x, y_1) - f(x, y_2)| \leq L|y_1 - y_2|$$

# 问题来源-微分方程

**定义 9.2** 假定矩形  $R = \{(t, y) : a \leq t \leq b, c \leq y \leq d\}$ , 设  $f(t, y)$  在  $R$  上连续。称  $f$  在  $R$  上关于变量  $y$  满足李普希茨条件, 若存在一常数  $L > 0$  对任意  $(t, y_1), (t, y_2) \in R$ , 具有性质:

$$|f(t, y_1) - f(t, y_2)| \leq L |y_1 - y_2| \quad (8)$$

常数  $L$  称为  $f$  的李普希茨常数。

**定理 9.1** 设  $f(t, y)$  定义在区域  $R$  上, 若存在一常数  $L > 0$ , 使得:

$$|f_y(t, y)| \leq L, (t, y) \in R \quad (9)$$

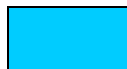
则  $f$  在区域  $R$  上关于变量  $y$  满足李普希茨条件, 李普希茨常数为  $L$ 。

**定理 9.2(存在性和惟一性)** 设  $f(t, y)$  在一区域  $R = \{(t, y) : t_0 \leq t \leq b, c \leq y \leq d\}$  上连续, 若  $f$  在区域  $R$  上关于变量  $y$  满足李普希茨条件, 且  $(t_0, y_0) \in R$ , 则初值问题式(6),  $y' = f(t, y)$ ,  $y(t_0) = y_0$ , 在某子区间  $t_0 \leq t \leq t_0 + \delta$  上有惟一解  $y = y(t)$ 。

# 问题来源-微分方程

常微分方程作为微分方程的基本类型之一，在自然界与工程界有很广泛的应用。很多问题的数学表述都可以归结为常微分方程的定解问题。很多偏微分方程问题，也可以化为常微分方程问题来近似求解。但只有极少数特殊的方程有解析解，而对于绝大部分的微分方程是没有解析解

$$\begin{cases} \frac{dy}{dx} = f(x, y), x \in [a, b] \\ y(a) = y_0 \end{cases}$$



$$y = f(x)$$

解析解难以求解

$$|f(x, y_1) - f(x, y_2)| < L |y_1 - y_2|$$

利普希茨 *Lipschitz* 条件



数值求解

$$a = x_0 < x_1 < x_2 < \cdots < x_n = b$$

$y(x_i)$  的近似值  $y_i (i = 0, 1, 2, \cdots, n)$

# 欧拉 (Euler) 方法

$$\begin{cases} \frac{dy}{dx} = f(x, y), x \in [a, b] \\ y(a) = y_0 \end{cases}$$

根据数值微分原理,  
用差商代替导数

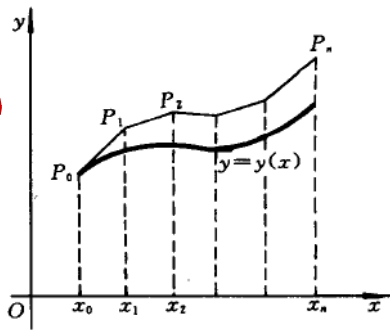
① 求出a处斜率

$$x_i = a + ih$$

$$(h = \frac{b-a}{n}; i = 0, 1, \dots, n)$$

$$y'(x_i) = f(x_i, y(x_i))$$

$$\frac{y(x_{i+1}) - y(x_i)}{h} \approx f(x_i, y(x_i))$$



解析法所得值为离散点

$$y(x_{i+1}) \approx y(x_i) + hf(x_i, y(x_i))$$

欧拉(前推)公式

②

$$y_{i+1} = y_i + hf(x_i, y_i)$$



# 欧拉 (Euler) 方法

例 1 用欧拉方法求初值问题

$$y_{i+1} = y_i + hf(x_i, y_i)$$

$$\begin{cases} y' = x - y + 1, x \in [0, 0.5] & \text{微分方程} \\ y(0) = 1 & \text{初始条件} \end{cases}$$

的数值解(取  $h = 0.1$ ), 并将计算结果与准确解  $y(x) = x + e^{-x}$  进行比较.

解 本题  $f(x, y) = x - y + 1, h = 0.1$ . 故由欧拉公式(6.4)可得求数值解的计算公式  
 $y_{i+1} = y_i + 0.1 \times (x_i - y_i + 1)$ , 即

$$y_{i+1} = 0.1x_i + 0.9y_i + 0.1$$

计算结果见表 6-1.

表 6-1

$x_i$	$y_i$	$y(x_i)$	$ y(x_i) - y_i $
0.0	1.000000	1.000000	0.000000
0.1	1.000000	1.004837	0.004837
0.2	1.010000	1.018731	0.008731
0.3	1.029000	1.040818	0.011818
0.4	1.056100	1.070320	0.014220
0.5	1.090490	1.106531	0.016041


# 欧拉 (Euler) 方法

## 欧拉公式的截断误差与精度分析


定义 1 若  $y_{i+1}$  是在  $y_j = y(x_j) (j \leq i)$  的假定下, 由某数值方法得到的  $y(x_{i+1})$  的近似值, 则称  $R_{i+1} = y(x_{i+1}) - y_{i+1}$  为该数值方法在  $x_{i+1}$  处的局部截断误差.

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \frac{h^2}{2} y''(\xi_i) \quad \text{泰勒展开}$$

$$y'(x_i) = f(x_i, y(x_i)) \quad \text{且} \quad y_i = y(x_i) \quad \text{微分方程}$$


$$y(x_{i+1}) = y_i + hf(x_i, y_i) + \frac{h^2}{2} y''(\xi_i)$$

$$y_{i+1} = y_i + hf(x_i, y_i) \quad \text{欧拉公式}$$


$$R_{i+1} = y(x_{i+1}) - y_{i+1} = \frac{h^2}{2} y''(\xi_i)$$

定义 2 如果一个数值方法的局部截断误差为  $O(h^{p+1})$ , 则称这个方法的阶数为  $p$ , 或称它是一个  $p$  阶方法.

欧拉公式是1阶方法

例 9.4 用欧拉方法求解区间  $[0, 3]$  内的初值问题并, 其中,  $h = 1, \frac{1}{2}, \frac{1}{4}$  和  $\frac{1}{8}$  并比较它们的解。

$$y' = \frac{t-y}{2}, y(0) = 1$$

解:

图 9.6 显示 4 个欧拉解和精确解曲线  $y(t) = 3e^{-t/2} - 2 + t$ , 表 9.2 个解值。对步长  $h = 0.25$ , 计算为:

$$y_1 = 1.0 + 0.25 \left( \frac{0.0 - 1.0}{2} \right) = 0.875$$

$$y_2 = 0.875 + 0.25 \left( \frac{0.25 - 0.875}{2} \right) = 0.796875$$

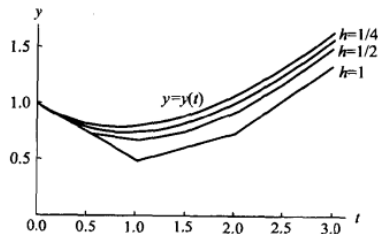


表 9.2 区间  $[0, 3]$  内  $y' = (t - y)/2, y(0) = 1$  不同步长欧拉方法的比较

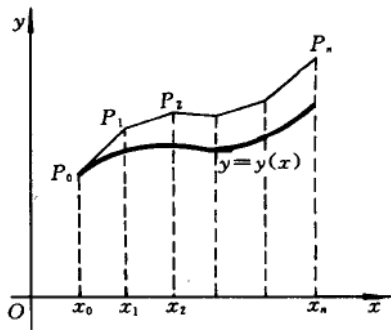
$t_k$	$y_k$				精确解 $y(t_k)$
	$h = 1$	$h = \frac{1}{2}$	$h = \frac{1}{4}$	$h = \frac{1}{8}$	
0	1.0	1.0	1.0	1.0	1.0
0.125				0.9375	0.943239
0.25				0.886719	0.897491
0.375				0.846924	0.862087
0.50				0.817429	0.836402
0.75			0.759766	0.786802	0.811868
1.00	0.5	0.6875	0.758545	0.790158	0.819592
1.50		0.765625	0.846386	0.882855	0.917100
2.00	0.75	0.949219	1.030827	1.068222	1.103638
2.50		1.211914	1.289227	1.325176	1.359514
3.00	1.375	1.533936	1.604252	1.637429	1.669390

$$R_{i+1} = y(x_{i+1}) - y_{i+1} = \frac{h^2}{2} y''(\xi_i)$$

# 欧拉 (Euler) 方法

## 后退 (隐式) 欧拉法

$$\frac{y(x_i) - y(x_{i-1}))}{h} \approx f(x_i, y(x_i))$$



$$y(x_i) \approx y(x_{i-1}) + hf(x_i, y(x_i)) \quad (i = 1, 2, \dots, n)$$

$$y_{i+1} = y_i + hf(x_{i+1}, y_{i+1}) \quad (i = 0, 1, \dots, n-1)$$

未知量

$$y_{i+1} = y_i + hf(x_i, y_i) \quad R_{i+1} = -\frac{h^2}{2}y''(x_i) + O(h^3)$$

欧拉前进公式

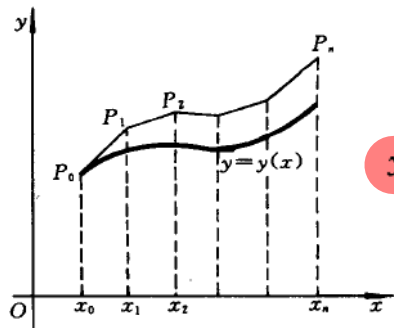
$$y_{i+1}^{(k+1)} = y_i + hf(x_{i+1}, y_{i+1}^{(k)}) \quad (k = 0, 1, 2, \dots)$$

# 欧拉 (Euler) 方法

## 改进欧拉法 (Heun休恩法)

欧拉前进公式  $R_{i+1} = y(x_{i+1}) - y_{i+1} = \frac{h^2}{2} y''(\xi_i)$

欧拉后退公式  $R_{i+1} = -\frac{h^2}{2} y''(x_i) + O(h^3)$



$$y(x_{i+1}) - \frac{1}{2}(\tilde{y}_{i+1} + \tilde{y}_{i+1}) = O(h^3)$$

$$y_{i+1} = y_i + \frac{h}{2} [f(x_i, y_i) + f(x_{i+1}, y_{i+1})]$$

$$(i = 0, 1, \dots, n-1)$$

# 欧拉 (Euler) 方法

$$\begin{cases} y_{i+1}^{(0)} = y_i + hf(x_i, y_i) & \text{欧拉前进公式} \\ y_{i+1}^{(k+1)} = y_i + \frac{h}{2} [f(x_i, y_i) + f(x_{i+1}, y_{i+1}^{(k)})] & \text{改进欧拉公式} \end{cases}$$

$$\begin{cases} \tilde{y}_{i+1} = y_i + hf(x_i, y_i) \\ y_{i+1} = y_i + \frac{h}{2} [f(x_i, y_i) + f(x_{i+1}, \tilde{y}_{i+1})] \quad (i=0, 1, \dots, n-1) \end{cases}$$

$$y_{i+1} = y_i + \frac{h}{2} [f(x_i, y_i) + f(x_{i+1}, y_i + hf(x_i, y_i))] \quad (i=0, 1, \dots, n-1)$$

**例 1** 用欧拉方法求初值问题

$$\begin{cases} y' = x - y + 1, x \in [0, 0.5] \\ y(0) = 1 \end{cases}$$

的数值解(取  $h=0.1$ ), 并将计算结果与准确解  $y(x) = x + e^{-x}$  进行比较.

# 欧拉 (Euler) 方法

**定理 9.4 (休恩方法的精度)** 设  $y(t)$  为初值问题式(1)的解, 若  $y(t) \in C^3[t_0, b]$ , 且  $\{(t_k, y_k)\}_{k=0}^M$  为休恩方法产生的一个逼近序列, 则:

$$\begin{aligned} |e_k| &= |y(t_k) - y_k| = O(h^2) \\ |\epsilon_{k+1}| &= |y(t_{k+1}) - y_k - h\Phi(t_k, y_k)| = O(h^3) \end{aligned} \quad (9)$$

其中,  $\Phi(t_k, y_k) = y_k + (h/2)(f(t_k, y_k) + f(t_{k+1}, y_k + hf(t_k, y_k)))$ 。

特别是, 区间终点处的最终全局误差满足:

$$E(y(b), h) = |y(b) - y_M| = O(h^2) \quad (10)$$

例 9.6 例和 9.7 说明了定理 9.4。若以步长  $h$  和  $h/2$  计算逼近, 则对较大的步长, 有:

$$E(y(b), h) \approx Ch^2 \quad (11)$$

且:

$$E\left(y(b), \frac{h}{2}\right) \approx C \frac{h^2}{4} = \frac{1}{4} Ch^2 \approx \frac{1}{4} E(y(b), h) \quad (12)$$

因此定理 9.4 的含义是若休恩方法中的步长减半, 则期望最终全局衰减系数大约减少到原来的  $\frac{1}{4}$ 。

# 欧拉 (Euler) 方法

精确解  $y(t) = 3e^{-t/2} - 2 + t$

例 9.6 利用休恩方法求解区间  $[0, 3]$  内初值问题:

0.8975

$$y' = \frac{t-y}{2}, y(0) = 1$$

比较  $h = 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$  的解。

$$\begin{cases} \tilde{y}_{i+1} = y_i + hf(x_i, y_i) \\ y_{i+1} = y_i + \frac{h}{2} [f(x_i, y_i) + f(x_{i+1}, \tilde{y}_{i+1})] \quad (i=0, 1, \dots, n-1) \end{cases}$$

对步长  $h = 0.25$ , 一个实际的计算例子是:

①  $f(\tilde{t}_0, y_0) = \frac{0-1}{2} = -0.5$

$y_{i+1} = 1.0 + 0.25(-0.5) = 0.875$

②  $f(t_1, p_1) = \frac{0.25 - 0.875}{2} = -0.3125$

$y_{i+1} = 1.0 + 0.125(-0.5 - 0.3125) = 0.8984375$



例 9.7 比较用休恩方法, 求解  $[0, 3]$  内  $y' = \frac{t-y}{2}$ ,  $y(0) = 1$ , 步长为  $1, \frac{1}{2}, \dots, \frac{1}{64}$  时的最终全局误差。

解:

表 9.5 给出了最终全局误差, 并显示出当步长减半时, 近似  $y(3)$  的误差衰减因子约为  $\frac{1}{4}$ :

$$E(y(3), h) = y(3) - y_M = O(h^2) \approx Ch^2, \text{ 其中 } C = -0.0432$$

表 9.5 区间  $[0, 3]$  内  $y' = (t - y)/2$ ,  $y(0) = 1$  步长与休恩方法最终全局误差的关系


步长 $h$	步数 $M$	$y(3)$ 逼近, $y_M$	$t = 3$ 时 F. G. E. 的误差 $y(3) - y_M$	$O(h^2) \approx Ch^2$ 其中 $C = -0.0432$
1	3	1.732422	-0.063032	-0.043200
$\frac{1}{2}$	6	1.682121	-0.012731	-0.010800
$\frac{1}{4}$	12	1.672269	-0.002879	-0.002700
$\frac{1}{8}$	24	1.670076	-0.000686	-0.000675
$\frac{1}{16}$	48	1.669558	-0.000168	-0.000169
$\frac{1}{32}$	96	1.669432	-0.000042	-0.000042
$\frac{1}{64}$	192	1.669401	-0.000011	-0.000011

# 龙格-库塔 (Runge-Kutta) 方法

$$y(x_{i+1}) - y(x_i) = y'(\zeta)(x_{i+1} - x_i), \zeta \in [x_i, x_{i+1}]$$

$$\frac{dy}{dx} = f(x, y) \quad \text{微分方程}$$

微分中值定理


$$y(x_{i+1}) = y(x_i) + hf(\zeta, y(\zeta))$$

$$f(\zeta, y(\zeta)) \stackrel{(\text{记})}{=} K^* \quad \text{平均斜率 (最优斜率)}$$

龙格-库塔 (Runge-Kutta) 方法的构造原理

$$\begin{cases} y_{i+1} = y_i + hK_1 \\ K_1 = f(x_i, y_i) \end{cases} \quad f(x_i, y(x_i)) \approx f(x_i, y_i) \stackrel{(\text{记})}{=} K_1$$

欧拉公式

# 龙格-库塔 (Runge-Kutta) 方法

又如, 若用点  $x_i$  处的斜率近似值  $K_1$  与右端点  $x_{i+1}$  处的斜率

$$f(x_{i+1}, y(x_{i+1})) \approx f(x_i + h, y_i + hf(x_i, y_i)) \stackrel{\text{(记)}}{=} K_2$$

的算术平均值作为平均斜率  $K^*$  的近似值, 则得

$$\begin{cases} y_{i+1} = y_i + \frac{h}{2}(K_1 + K_2) \\ K_1 = f(x_i, y_i) \\ K_2 = f(x_i + h, y_i + hK_1) \end{cases}$$

改进欧拉公式

# 龙格-库塔 (Runge-Kutta) 方法

## 基本思路

在  $[x_i, x_{i+1}]$  内多预估几个点 (通常将左端点  $x_i$  考虑在内) 上的斜率值

$K_1, K_2, \dots, K_m$ , 然后用它们的加权平均值作为平均斜率  $K^*$  的近似值,

$$\begin{cases} y_{i+1} = y_i + h(\alpha_1 K_1 + \alpha_2 K_2 + \dots + \alpha_m K_m) \\ K_1 = f(x_i, y_i) \\ K_2 = f(x_i + \lambda_2 h, y_i + \mu_2 h K_1) \\ \vdots \\ K_m = f(x_i + \lambda_m h, y_i + \mu_m h K_{m-1}) \end{cases}$$

$\alpha, \lambda, \mu$  为待定参数, 应使公式的截断误差的阶数尽量高

# 龙格-库塔 (Runge-Kutta) 方法

## 二阶龙格-库塔公式

$$\begin{cases} y_{i+1} = y_i + h(a_1 K_1 + a_2 K_2) \\ K_1 = f(x_i, y_i) \\ K_2 = f(x_i + \lambda_2 h, y_i + \lambda_2 h K_1) \end{cases}$$

将  $K_2$  在  $(x_i, y_i)$  处泰勒展开, 并注意到  $y_i = y(x_i)$ , 就有

$$y_i = y(x_i) + (a_1 + a_2)hy'(x_i) + a_2\lambda_2 h^2 y''(x_i) + O(h^3)$$

$$y(x_{i+1}) = y(x_i) + y'(x_i)h + \frac{1}{2}y''(x_i)h^2 + O(h^3)$$

$$\begin{cases} a_1 + a_2 = 1 \\ a_2\lambda_2 = \frac{1}{2} \end{cases}$$

$$a_1 = a_2 = \frac{1}{2}, \lambda_2 = 1$$

$$y_{i+1} = y_i + \frac{h}{2}[f(x_i, y_i) + f(x_{i+1}, y_i + hf(x_i, y_i))]$$

改进欧拉公式

$$a_1 = 0, a_2 = 1, \lambda_2 = \frac{1}{2}$$

$$y_{i+1} = y_i + hf[x_i + \frac{1}{2}h, y_i + \frac{h}{2}f(x_i, y_i)]$$

中点公式

# 龙格-库塔 (Runge-Kutta) 方法

## 4阶经典龙格-库塔法

$$y_{k+1} = y_k + \frac{h(f_1 + 2f_2 + 2f_3 + f_4)}{6}$$

①  $y_k \sim y_{k+1}$   
内两点

$$f_1 = f(t_k, y_k)$$

② 用于求  $y_{k+1}$

$$f_2 = f\left(t_k + \frac{h}{2}, y_k + \frac{h}{2}f_1\right)$$

欧拉前进公式

$$f_3 = f\left(t_k + \frac{h}{2}, y_k + \frac{h}{2}f_2\right)$$

欧拉后退公式

$$f_4 = f(t_k + h, y_k + hf_3)$$

# 龙格-库塔 (Runge-Kutta) 方法

## 4阶经典龙格-库塔法

$$\frac{dy}{dt} = f(t, y) \quad \square \quad y(t_1) - y(t_0) = \int_{t_0}^{t_1} f(t, y(t)) dt \quad (8)$$

若应用辛普生公式和步长  $h/2$ , (8)式的积分近似为:

$$\int_{t_0}^{t_1} f(t, y(t)) dt \approx \frac{h}{6} (f(t_0, y(t_0)) + 4f(t_{1/2}, y(t_{1/2})) + f(t_1, y(t_1))) \quad (9)$$

其中  $t_{1/2}$  为区间中点。需要 3 个函数值, 因此很显然, 我们选择  $f(t_0, y(t_0)) = f_1$  和  $f(t_1, y(t_1)) \approx f_4$ 。对于中点的值, 我们选择为  $f_2$  和  $f_3$  的平均值:

$$f(t_{1/2}, y(t_{1/2})) \approx \frac{f_2 + f_3}{2}$$

将这些值代入式(9), 并使用得到的结果在式(8)中求  $y_1$ :

$$y_1 = y_0 + \frac{h}{6} \left( f_1 + \frac{4(f_2 + f_3)}{2} + f_4 \right) \quad (10)$$

# 龙格-库塔 (Runge-Kutta) 方法

例 9.10 用 RK4 方法求解区间  $[0, 3]$  内初值问题  $y' = (t - y)/2, y(0) = 1$ 。比较  $h = 1, \frac{1}{2}, \frac{1}{4},$

$\frac{1}{8}$  时的解。

①  $(0.1)$  处代方程  $\rightarrow f_1$

②  $y(x_{i+1}) = y(x_i) + f_i \cdot h \rightarrow$  下一个点的  $y_2$

③  $(x_2, y_2)$  处代方程  $\rightarrow f_2 \dots\dots$

解:

表 9.8 给出了选定横坐标上的值, 对步长  $h = 0.25$ , 一个计算的实例为:

$$f_1 = \frac{0.0 - 1.0}{2} = -0.5$$

$$f_2 = \frac{0.125 - (1 + 0.25(0.5)(-0.5))}{2} = -0.40625$$

$$f_3 = \frac{0.125 - (1 + 0.25(0.5)(-0.40625))}{2} = -0.4121094$$

$$f_4 = \frac{0.25 - (1 + 0.25(-0.4121094))}{2} = -0.3234863$$

$y(0.25)$  !

$$y_1 = 1.0 + 0.25 \left( \frac{-0.5 + 2(-0.40625) + 2(-0.4121094) - 0.3234863}{6} \right)$$

$$= 0.8974915$$



# 龙格-库塔 (Runge-Kutta) 方法

表 9.8 比较区间 $[0,3]$ 内  $y' = (t - y)/2, y(0) = 1$  的 RK4 不同步长时的解

$t_k$	$y_k$				$y(t_k)$ 精确值
	$h = 1$	$h = \frac{1}{2}$	$h = \frac{1}{4}$	$h = \frac{1}{8}$	
0	1.0	1.0	1.0	1.0	1.0
0.125				0.9432392	0.9432392
0.25			0.8974915	0.8974908	0.8974917
0.375				0.8620874	0.8620874
0.50		0.8364258	0.8364037	0.8364024	0.8364023
0.75			0.8118696	0.8118679	0.8118678
1.00	0.8203125	0.8196285	0.8195940	0.8195921	0.8195920
1.50		0.9171423	0.9171021	0.9170998	0.9170997
2.00	1.1045125	1.1036826	1.1036408	1.1036385	1.1036383
2.50		1.3595575	1.3595168	1.3595145	1.3595144
3.00	1.6701860	1.6694308	1.6693928	1.6693906	1.6693905

# 龙格-库塔 (Runge-Kutta) 方法

例 9.11 比较用 RK4 计算  $y' = (t-y)/2, y(0) = 1$  时的最终全局误差, 步长采用  $1, \frac{1}{2}, \frac{1}{4}$  和  $\frac{1}{8}$ 。

表 9.9 列出了不同步长的最终全局误差, 说明当步长减半时,  $y(3)$  的近似值的误差衰减因子约为  $\frac{1}{16}$ 。

$$E(y(3), h) = y(3) - y_M = O(h^4) \approx Ch^4, \text{ 其中 } C = -0.000614$$

表 9.9 区间  $[0, 3]$  内  $y' = (t-y)/2, y(0) = 1$  的 RK4 解的步长与最终全局误差的关系

步长 $h$	步数 $M$	$y(3)$ 逼近, $y_M$	$t = 3$ 时 F. G. E. 的误差 $y(3) - y_M$	$O(h^4) \approx Ch^4$ 其中 $C = -0.000614$
1	3	1.6701860	-0.0007955	-0.0006140
$\frac{1}{2}$	6	1.6694308	-0.0000403	-0.0000384
$\frac{1}{4}$	12	1.6693928	-0.0000023	-0.0000024
$\frac{1}{8}$	24	1.6693906	-0.0000001	-0.0000001

# 泰勒展开方法

定理 9.5 (泰勒定理) 设  $y(t) \in C^{N+1}[t_0, b]$ , 且  $y(t)$  有在固定值  $t = t_k \in [t_0, b]$  处展开的  $N$  次泰勒级数:

$$y(t_k + h) = y(t_k) + hT_N(t_k, y(t_k)) + O(h^{N+1}) \quad (1)$$

其中:

$$T_N(t_k, y(t_k)) = \sum_{j=1}^N \frac{y^{(j)}(t_k)}{j!} h^{j-1} \quad (2)$$

$y^{(j)}(t) = f^{(j-1)}(t, y(t))$  表示函数  $f$  关于  $t$  的  $(j-1)$  次全导数。求导公式可以递归计算:

$$\begin{aligned} y'(t) &= f \\ y''(t) &= f_t + f_y y' = f_t + f_y f \\ y^{(3)}(t) &= f_{tt} + 2f_{ty} y' + f_{yy} y'' + f_{yy} (y')^2 \\ &= f_{tt} + 2f_{ty} f + f_{yy} f^2 + f_y (f_t + f_y f) \\ y^{(4)}(t) &= f_{ttt} + 3f_{t_{yy}} y' + 3f_{tyy} (y')^2 + 3f_{yy} y'' \\ &\quad + f_y y''' + 3f_{yy} y' y'' + f_{yyy} (y')^3 \\ &= (f_{ttt} + 3f_{t_{yy}} f + 3f_{tyy} f^2 + f_{yyy} f^3) + f_y (f_{tt} + 2f_{ty} f + f_{yy} f^2) \\ &\quad + 3(f_t + f_y f)(f_{ty} + f_{yy} f) + f_y^2 (f_t + f_y f) \end{aligned} \quad (3)$$

# 泰勒展开方法

$N$  次泰勒方法的一般步骤为:

$$y_{k+1} = y_k + d_1 h + \frac{d_2 h^2}{2!} + \frac{d_3 h^3}{3!} + \cdots + \frac{d_N h^N}{N!} \quad (5)$$

其中在各步  $k=0, 1, \cdots, M-1$ ,  $d_j = y^{(j)}(t_k)$ , 对  $j=1, 2, \cdots, N$ .

**定理 9.6 (N 次泰勒方法的精度)** 设  $y(t)$  为初值问题的解, 若  $y(t) \in C^{N+1}[t_0, b]$ , 且  $\{(t_k, y_k)\}_{k=0}^M$  为  $N$  次泰勒方法产生的逼近序列, 则:

$$\begin{aligned} |e_k| &= |y(t_k) - y_k| = O(h^{N+1}) \\ |\epsilon_{k+1}| &= |y(t_{k+1}) - y_k - hT_N(t_k, y_k)| = O(h^N) \end{aligned} \quad (6)$$

特别是, 区间终点处的最终全局误差满足:

$$E(y(b), h) = |y(b) - y_M| = O(h^N) \quad (7)$$

$$E(y(b), h) \approx Ch^4$$

$$E\left(y(b), \frac{h}{2}\right) \approx C \frac{h^4}{16} = \frac{1}{16} ch^4 \approx \frac{1}{16} E(y(b), h)$$

# 泰勒展开方法

例 9.8 用 4 次泰勒方法求解区间  $[0, 3]$  内的  $y' = (t - y)/2$ ,  $y(0) = 1$ , 比较  $h = 1, \frac{1}{2}, \frac{1}{4}, \frac{1}{8}$  时的结果。

首先必须求出  $y(t)$  的导数, 回忆  $y(t)$  的解是  $t$  的函数, 对  $y'(t) = f(t, y(t))$  关于  $t$  求导, 得到  $y^{(2)}(t)$ 。然后重复该过程得到高阶导数:

$$y'(t) = \frac{t - y}{2}$$

$$y^{(2)}(t) = \frac{d}{dt} \left( \frac{t - y}{2} \right) = \frac{1 - y'}{2} = \frac{1 - (t - y)/2}{2} = \frac{2 - t + y}{4}$$

$$y^{(3)}(t) = \frac{d}{dt} \left( \frac{2 - t + y}{4} \right) = \frac{0 - 1 + y'}{4} = \frac{-1 + (t - y)/2}{4} = \frac{-2 + t - y}{8}$$

$$y^{(4)}(t) = \frac{d}{dt} \left( \frac{-2 + t - y}{8} \right) = \frac{-0 + 1 - y'}{8} = \frac{1 - (t - y)/2}{8} = \frac{2 - t + y}{16}$$

要求出  $y_1$ , 必须在点  $(t_0, y_0) = (0, 1)$  处求上述导数, 计算得:

$$d_1 = y'(0) = \frac{0.0 - 1.0}{2} = -0.5$$

$$d_2 = y^{(2)}(0) = \frac{2.0 - 0.0 + 1.0}{4} = 0.75$$

$$d_3 = y^{(3)}(0) = \frac{-2.0 + 0.0 - 1.0}{8} = -0.375$$

$$d_4 = y^{(4)}(0) = \frac{2.0 - 0.0 + 1.0}{16} = 0.1875$$

然后将导数  $\{d_j\}$  和  $h = 0.25$  代入式(5), 用嵌套乘法来计算值  $y_1$ :

$$y_1 = 1.0 + 0.25 \left( -0.5 + 0.25 \left( \frac{0.75}{2} + 0.25 \left( \frac{-0.375}{6} + 0.25 \left( \frac{0.1875}{24} \right) \right) \right) \right) = 0.8974915$$

# 泰勒展开方法

要计算  $y_2$ , 必须在点  $(t_1, y_1) = (0.25, 0.8974915)$  处计算导数  $\{d_j\}$ 。计算量非常大, 手算十分枯燥, 计算的结果为:

$$d_1 = y'(0.25) = \frac{0.25 - 0.8974915}{2} = -0.3237458$$

$$d_2 = y^{(2)}(0.25) = \frac{2.0 - 0.25 + 0.8974915}{4} = 0.6618729$$

$$d_3 = y^{(3)}(0.25) = \frac{-2.0 + 0.25 - 0.8974915}{8} = -0.3309364$$

$$d_4 = y^{(4)}(0.25) = \frac{2.0 - 0.25 + 0.8974915}{16} = 0.1654682$$

然后将这些导数  $\{d_j\}$  和  $h = 0.25$  代入式(5), 用嵌套乘法计算  $y_2$ :

$$\begin{aligned} y_2 &= 0.8974915 + 0.25(-0.3237458 \\ &\quad + 0.25\left(\frac{0.6618729}{2} + 0.25\left(\frac{-0.3309364}{6} + 0.25\left(\frac{0.1654682}{24}\right)\right)\right)) \\ &= 0.8364037 \end{aligned}$$

# 泰勒展开方法

解点为 $(t_2, y_2) = (0.50, 0.8364037)$ 。表 9.6 给出各种不同步长在选择横坐标处的解值。

表 9.6 区间 $[0, 3]$ 内  $y' = (t - y)/2$ ,  $y(0) = 1$  的  $N=4$  泰勒解比较

$t_k$	$y_k$				$y(t_k)$ 精确值
	$h = 1$	$h = \frac{1}{2}$	$h = \frac{1}{4}$	$h = \frac{1}{8}$	
0	1.0	1.0	1.0	1.0	1.0
0.125				0.9432392	0.9432392
0.25			0.8974915	0.8974908	0.8974917
0.375				0.8620874	0.8620874
0.50		0.8364258	0.8364037	0.8364024	0.8364023
0.75			0.8118696	0.8118679	0.8118678
1.00	0.8203125	0.8196285	0.8195940	0.8195921	0.8195920
1.50		0.9171423	0.9171021	0.9170998	0.9170997
2.00	1.1045125	1.1036826	1.1036408	1.1036385	1.1036383
2.50		1.3595575	1.3595168	1.3595145	1.3595144
3.00	1.6701860	1.6694308	1.6693928	1.6693906	1.6693905

# 泰勒展开方法

例 9.9 比较例 9.8 中区间  $[0,3]$  内  $y' = (t - y)/2, y(0) = 1$  的泰勒解的最终全局误差。

表 9.7 列出了这些步长下的最终全局误差,表明  $y(3)$  的误差当步长减半时衰减因子约为  $1/16$ :

$$E(y(3), h) = y(3) - y_M = O(h^4) \approx Ch^4, C = -0.000614$$

表 9.7 区间  $[0,3]$  内  $y' = (t - y)/2$  泰勒解的步长与最终全局误差的关系

步长 $h$	步数 $M$	$y(3)$ 逼近, $y_M$	$t = 3$ 时 F.G.E. 的误差 $y(3) - y_M$	$O(h^2) \approx Ch^4$ 其中 $C = -0.000614$
1	3	1.6701860	-0.0007955	-0.0006140
$\frac{1}{2}$	6	1.6694308	-0.0000403	-0.0000384
$\frac{1}{4}$	12	1.6693928	-0.0000023	-0.0000024
$\frac{1}{8}$	24	1.6693906	-0.0000001	-0.0000001



# Matlab解常微分方程

Runge-Kutta 法:

ODE解函数: ode23、ode45、ode113、  
ode15s、ode23s

参数选择函数: odeset、odeget

输出函数: odeplot、odephas2、  
odephas3、odeprint

# Matlab解常微分方程

例  $y' = -2y + 2x^2 + x$  其中  $(0 < X < 0.5), y(0) = 1$

fun.m

```
function f=fun(x,y)
```

```
f=-2*y+2*x^2+2*x;
```

```
[x,y]=ode23('fun', [0,0.5],1);
```

```
[x,y]=ode23('fun', [0:0.1:1.0],1)
```

# Matlab解常微分方程

## 符号解法

```
dsolve('Df=f+sin(t)','f(pi/2)=0')
```

```
ans  
=-1/2*cos(t)-1/2*sin(t)+1/2*exp(t)/(cosh(1/2*pi)+sin  
h(1/2*pi))
```

```
dsolve('D2y=-a^2*y','y(0)=1,Dy(pi/a)=0')
```

```
ans =cos(a*t)
```



浙江大学  
ZheJiang University

Thanks for your  
attention!



