

# 《数值分析》 第九章

→ 解一阶常微分方程欧拉法

→ Range-Kutta公式

## 常微分方程初值问题

$$(1) \begin{cases} y' = f(x, y) \\ y(x_0) = y_0 \end{cases}$$

其中， $y = y(x)$  是未知函数， $y(x_0) = y_0$  是初值条件，而  $f(x, y)$  是给定的二元函数。

由常微分方程理论知，若  $f(x)$  在  $x \in [a, b]$  连续且  $f$  满足对  $y$  的 **Lipschitz** 条件：

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

（其中  $L$  为 Lipschitz 常数）则初值问题（1）存在唯一的连续解。

## 例1.一阶常微分方程

$$\begin{cases} \frac{dy}{dx} = 2(x - y) \\ y(0) = 0.8 \end{cases}$$

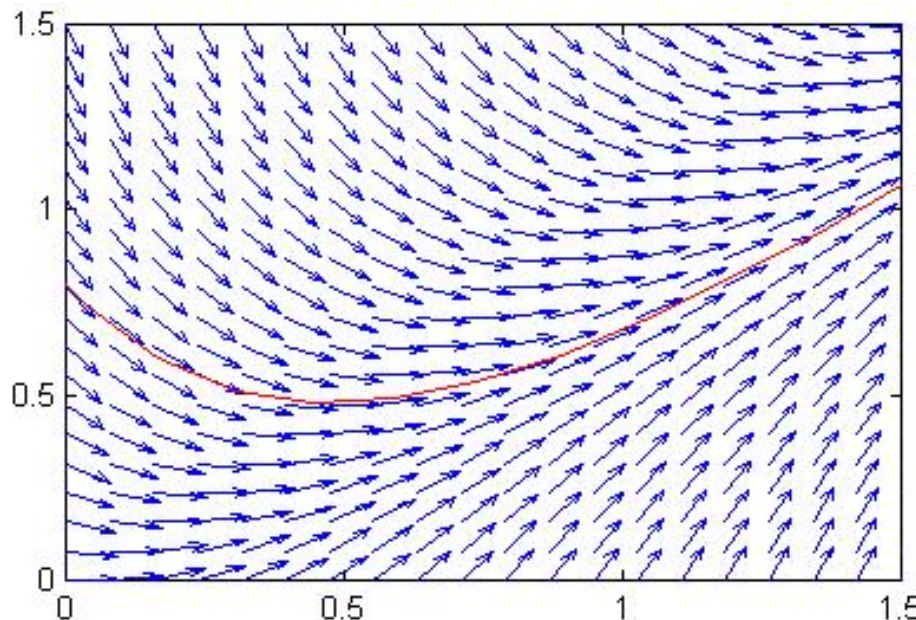
$$f(x, y) = 2(x - y)$$

$$\Rightarrow f(x_i, y_j) = k_{ij} = \tan \alpha_{ij}$$

斜率 $\rightarrow$ 方向余弦

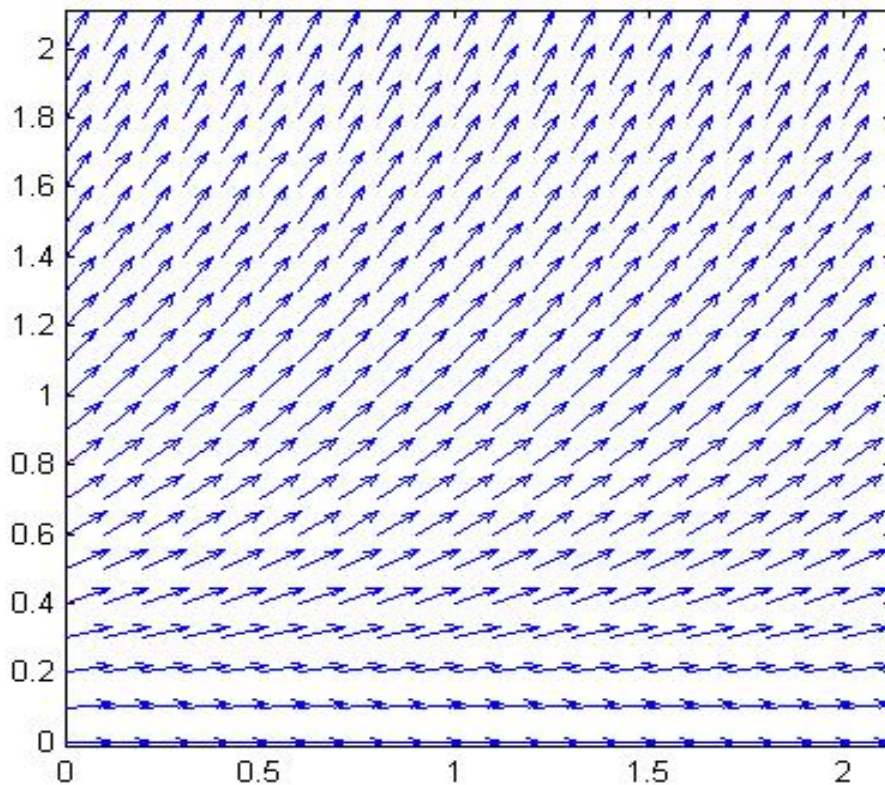
$$[\cos \alpha_{ij}, \sin \alpha_{ij}] = \left[ \frac{1}{\sqrt{1 + k_{ij}^2}}, \frac{k_{ij}}{\sqrt{1 + k_{ij}^2}} \right]$$

求解区域:  $0 \leq x \leq 1.5$

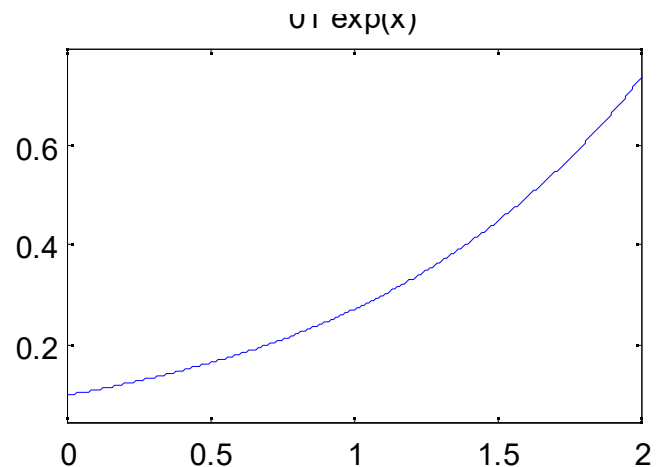


例2. Malthus模型:  $\frac{dp}{dt} = \lambda p, p(t_0) = p_0$

取  $\lambda = 1$  初值条件:  $p(0)=1/10$ 。解析解



$$p = \frac{1}{10} \exp(t)$$



例3. Logistic模型  $\frac{dy}{dx} = ry(1 - \frac{1}{K}y)$

解:  $y(x) = \frac{K}{1 + \exp(-rx - c_0)}$

取  $K = 1, r = 1$

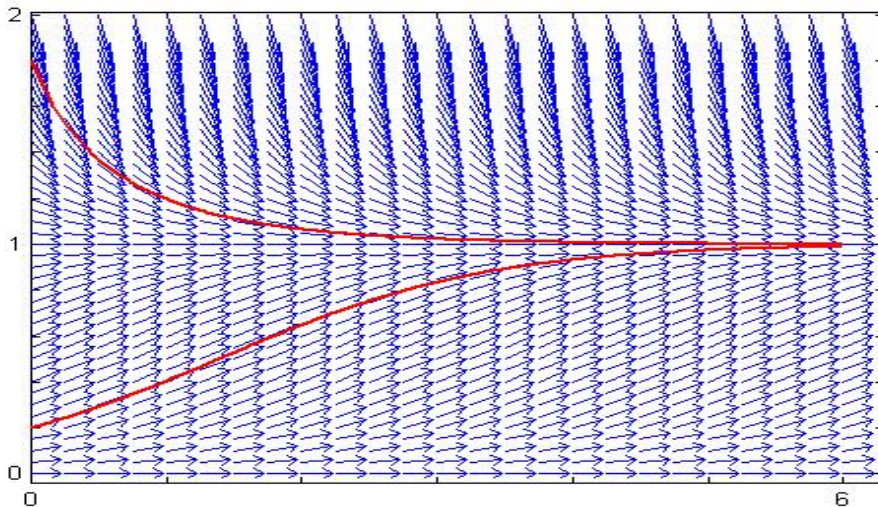
→  $y(x) = \frac{1}{1 + C_1 \exp(-x)}$

初值条件  $y(0) = 1.8$

$$y_1(x) = \frac{1}{1 - 4 \exp(-x) / 9}$$

初值条件  $y(0) = 0.2$

$$y_1(x) = \frac{1}{1 + 4 \exp(-x)}$$



# 解析方法与数值解法.

数值解法就是求 $y(x)$ 在离散节点 $x_1 < \cdots < x_n < \cdots$ 上的近似值 $y_1, \cdots, y_n, \cdots$ .

常用步进式的方法. 单步法, k步法.

研究计算公式的局部截断误差和阶, 数值解 $y_n$ 与准确解 $y(x_n)$ 的误差估计及收敛性, 计算公式的稳定性等.

所谓数值解法，就是寻求解  $y(x)$  在一系列离散节点

$$x_1 < x_2 < \dots < x_n < x_{n+1} < \dots$$

上的近似值  $y_1, y_2, \dots, y_n, y_{n+1}, \dots$  .

相邻两个节点的间距  $h_n = x_{n+1} - x_n$  称为步长.

如不特别说明，总是假定  $h_i = h (i = 1, 2, \dots)$  为定数，

这时节点为  $x_n = x_0 + nh (i = 0, 1, 2, \dots)$

初值问题 (1.1) , (1.2) 的数值解法的基本特点是采取“步进式” .

即求解过程顺序地向前推进.

$$\begin{cases} y' = f(x, y), & (1.1) \\ y(x_0) = y_0. & (1.2) \end{cases}$$



描述这类算法，只要给出用已知信息  $y_n, y_{n-1}, y_{n-2}, \dots$  计算  $y_{n+1}$  的递推公式.

首先对方程  $y' = f(x, y)$  离散化，建立求数值解的递推公式.

一类是计算  $y_{n+1}$  时只用到前一点的值  $y_n$ ，称为单步法.

另一类是用到  $y_{n+1}$  前面  $k$  点的值  $y_n, y_{n-1}, \dots, y_{n-k+1}$ ，称为  $k$  步法.

其次，要研究公式的局部截断误差和阶，数值解  $y_n$  与精确解  $y(x_n)$  的误差估计及收敛性，还有递推公式的计算稳定性等问题.



## 9.2.1 欧拉方法

在 $x = x_0$ 处，用差商代替导数：

$$y'(x_0) \approx \frac{y(x_1) - y(x_0)}{x_1 - x_0} = \frac{y(x_1) - y(x_0)}{h}$$

由  $y'(x_0) = f(x_0, y_0), \quad y(x_0) = y_0$

得  $y(x_1) \approx y_0 + hf(x_0, y_0) = y_1$

同理，在 $\mathbf{x} = \mathbf{x}_n$ 处，用差商代替导数：

$$y'(x_n) \approx \frac{y(x_{n+1}) - y(x_n)}{x_{n+1} - x_n} = \frac{y(x_{n+1}) - y(x_n)}{h}$$

由  $y'(x_n) = f(x_n, y_n)$

得  $y(x_{n+1}) \approx y(x_n) + hf(x_n, y_n)$

若记  $y(x_n) \approx y_n, \quad y(x_{n+1}) \approx y_{n+1}$

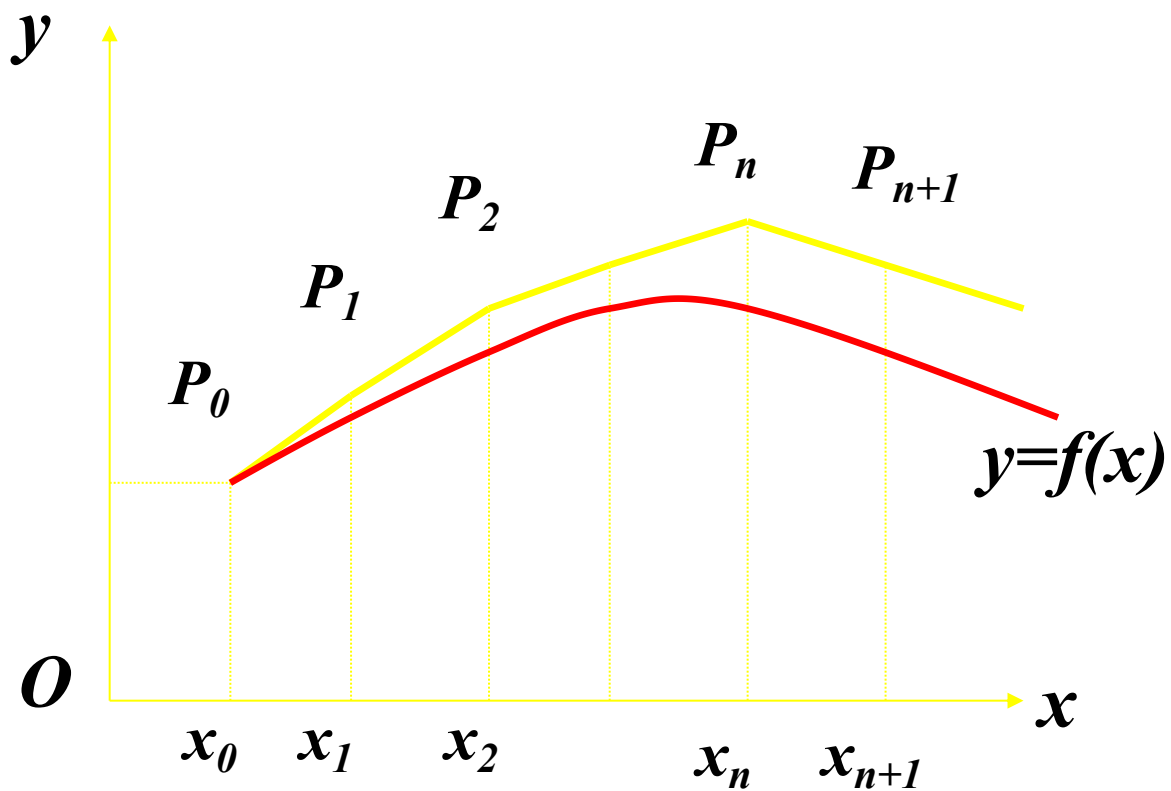
则上式可记为

$$y_{n+1} = y_n + hf(x_n, y_n)$$

此即为求解初值问题的**Euler方法**，又称**显式Euler方法**。

Euler方法的几何意义:

(Euler折线法)



**例：**用Euler方法求解常微分方程初值问题

$$\begin{cases} y' = \frac{y}{x} - 2y^2 & (0 < x < 3) \\ y(0) = 0. \end{cases}$$

并将数值解和该问题的解析解比较。

$$\text{解析解: } y(x) = \frac{x}{1+x^2}$$

**解：** Euler方法的具体格式：

$$y_{n+1} = y_n + h\left(\frac{y_n}{x_n} - 2y_n^2\right)$$

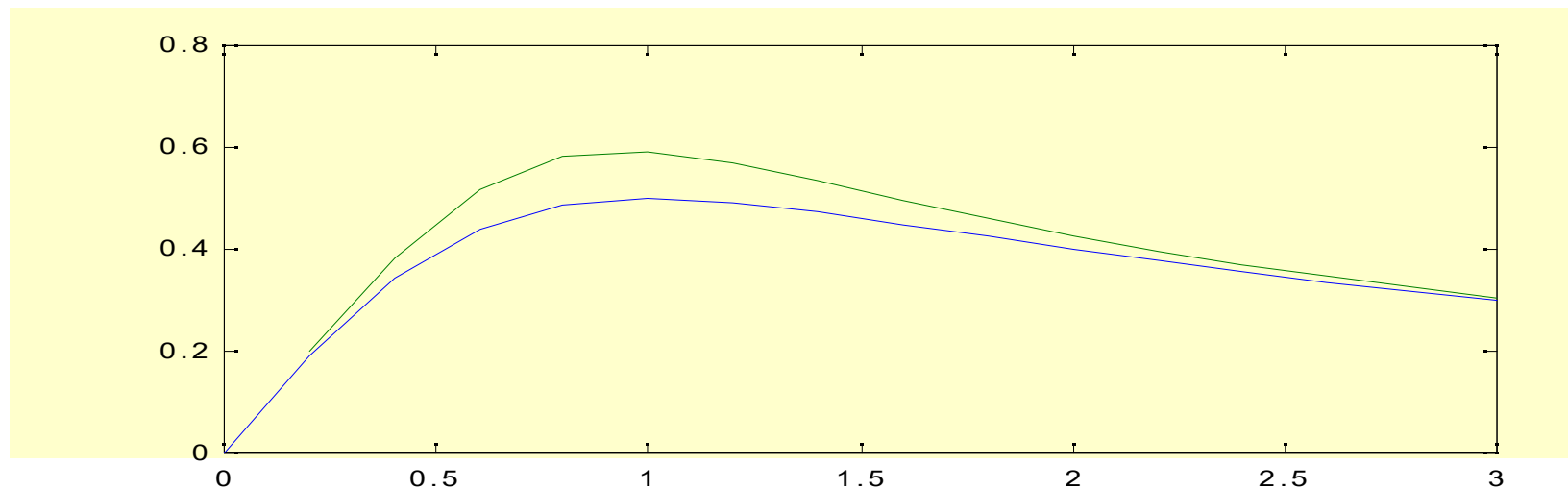
取 $h=0.2$ ,  $x_n=nh, (n=0,1,2,\dots,15)$ ,  $f(x,y)=y/x - 2y^2$  计算中取 $f(0,0)=1$ . 计算结果如下:

$x_n$	$y(x_n)$	$y_n$	$y_n - y(x_n)$
0.0	0	0	0
0.2	0.1923	0.2000	0.0077
0.4	0.3448	0.3840	0.0392
0.6	0.4412	0.5170	0.0758
0.8	0.4878	0.5824	0.0946
1.0	0.5000	0.5924	0.0924
1.2	0.4918	0.5705	0.0787
1.4	0.4730	0.5354	0.0624

$x_n$	$y(x_n)$	$y_n$	$y_n - y(x_n)$
1.6	0.4494	0.4972	0.0478
1.8	0.4245	0.4605	0.0359
2.0	0.4000	0.4268	0.0268
2.2	0.3767	0.3966	0.0199
2.4	0.3550	0.3698	0.0147
2.6	0.3351	0.3459	0.0108
2.8	0.3167	0.3246	0.0079
3.0	0.3000	0.3057	0.0057

由表中数据可以看到，微分方程初值问题的数值解和解析解的误差一般在小数点后第二位或第三位小数上，这说明Euler方法的精度是比较差的。

数值解和解析解的图示比较如下：



***O*** : 数值解; — : 准确解



若直接对 $y'=f(x,y)$ 在 $[x_n, x_{n+1}]$ 积分,

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y(x)) dx$$

利用数值积分中的左矩形公式:

$$\int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx h \cdot f(x_n, y(x_n))$$

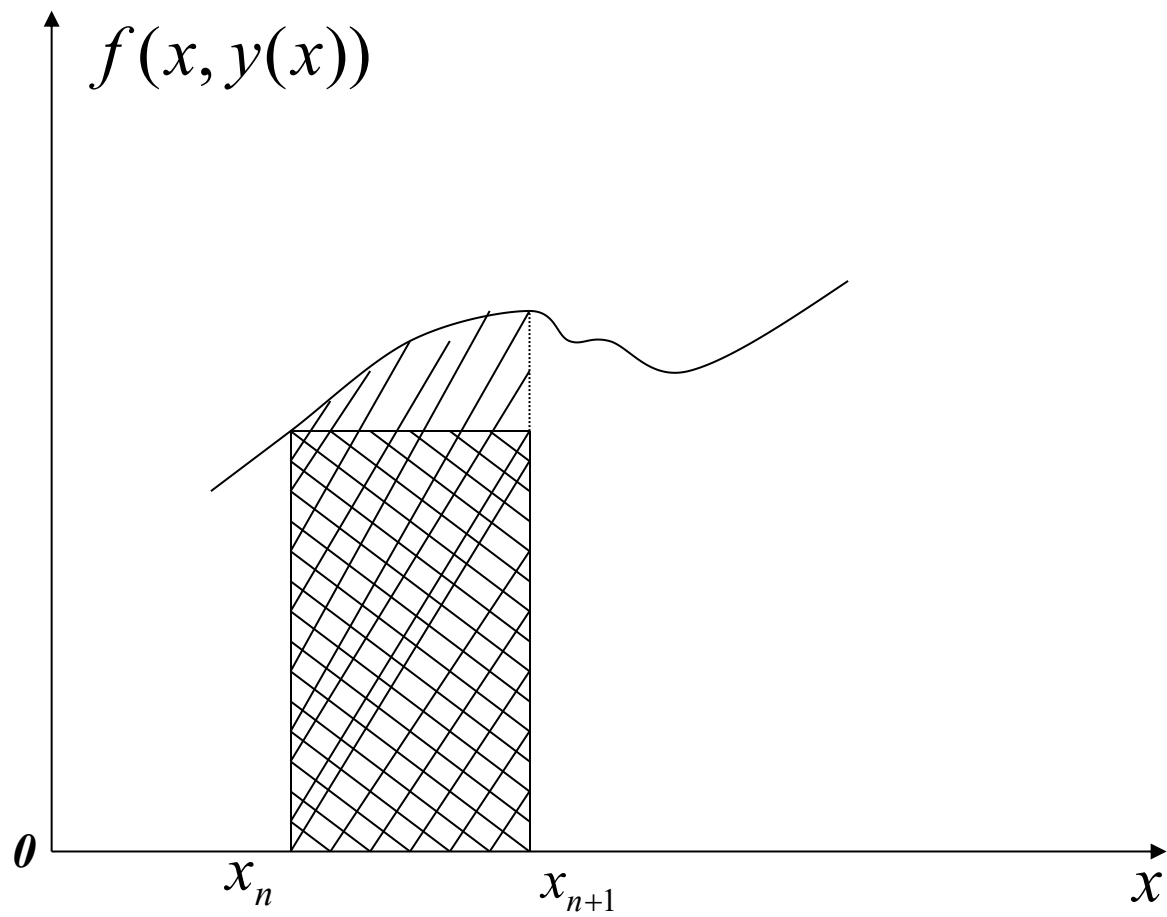
设 $y(x_n) = y_n$ , 则得

$$y_{n+1} = y_n + hf(x_n, y_n)$$

此即为**Euler**公式。

若用右矩形公式:

$$\int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx h \cdot f(x_{n+1}, y(x_{n+1}))$$



欧拉公式推导：左矩形

得

$$y_{n+1} = y_n + hf(x_{n+1}, y_{n+1})$$

上式称**后退的Euler方法**，又称**隐式Euler方法**。

可用迭代法求解：

初值：

$$y_{n+1}^{(0)} = y_n + hf(x_n, y_n)$$

迭代：

$$y_{n+1}^{(k+1)} = y_n + hf(x_{n+1}, y_{n+1}^{(k)}) \quad \mathbf{k=0,1,\dots}$$

因

$$\left| y_{n+1}^{(k+1)} - y_{n+1} \right| = h \left| f(x_{n+1}, y_{n+1}^{(k)}) - f(x_{n+1}, y_{n+1}) \right|$$

$$\leq hL \left| y_{n+1}^{(k)} - y_{n+1} \right|$$

故当 $hL < 1$ 时，迭代法收敛。

## 梯形方法

由

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y(x)) dx$$

利用梯形求积公式：

$$\int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx \frac{h}{2} [f(x_n, y(x_n)) + f(x_{n+1}, y(x_{n+1}))]$$

得

$$y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_{n+1})]$$

上式称**梯形方法**，是一种**隐式方法**。

用迭代法求解:

初值:

$$y_{n+1}^{(0)} = y_n + hf(x_n, y_n)$$

迭代:

$$y_{n+1}^{(k+1)} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{(k)})]$$

**k=0,1,.....**

因

$$\left| y_{n+1}^{(k+1)} - y_{n+1} \right| = \frac{h}{2} \left| f(x_{n+1}, y_{n+1}^{(k)}) - f(x_{n+1}, y_{n+1}) \right|$$

$$\leq \frac{hL}{2} \left| y_{n+1}^{(k)} - y_{n+1} \right|$$

故当 $hL/2 < 1$ 时, 迭代法收敛。

由以上分析可以看出，隐式方法的计算比显式方法复杂，需要用迭代法求解非线性方程才能得出计算结果。

可采用将显式Euler格式与梯形格式结合使用的方法来避免求解非线性方程。

记  $\bar{y}_{n+1} = y_n + hf(x_n, y_n)$

——预测

再用梯形格式计算：

$$y_{n+1} = y_n + \frac{h}{2} [f(x_n, y_n) + f(x_{n+1}, \bar{y}_{n+1})]$$

——校正

上面两式统称预测—校正法，又称改进的Euler方法。  
(公式2.9)

### 9.2.3单步法的局部截断误差和精度

单步法的一般形式为：（ $\varphi$ 与 $f$ 有关）

$$y_{n+1} = y_n + h\varphi(x_n, y_n, y_{n+1}, h)$$

显式单步法形式为：

$$y_{n+1} = y_n + h\varphi(x_n, y_n, h)$$

**整体截断误差：**从 $x_0$ 开始，考虑每一步产生的误差，直到 $x_n$ ，则有误差

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

称为数值方法在节点 $x_n$ 处的整体截断误差。

但 $e_n$ 不易分析和计算，故只考虑从 $x_n$ 到 $x_{n+1}$ 的局部情况。



**定义：** 设 $y(x)$ 是初值问题(1)的精确解，则称

$$T_{n+1} = y(x_{n+1}) - y(x_n) - h\varphi(x_n, y(x_n), h)$$

为显式单步法在节点 $x_{n+1}$ 处的**局部截断误差**。（公式2.12）

若存在最大整数 $p$ 使局部截断误差满足

$$T_{n+1} = y(x_n + h) - y(x_n) - h\varphi(x_n, y_n, h) = O(h^{p+1})$$

则称显式单步法具有**p阶精度**或称**p阶方法**。

**注：** 将 $T_{n+1}$ 表达式各项在 $x_n$ 处作Taylor展开，可得具体表达式。

# Euler方法的局部截断误差

(设 $y_n=y(x_n)$ )

$$T_{n+1} = y(x_{n+1}) - y(x_n) - hf(x_n, y(x_n))$$

$$= y(x_n + h) - y(x_n) - hy'(x_n)$$

$$= y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(\varepsilon_n) - y(x_n) - hy'(x_n)$$

$$= \frac{h^2}{2} y''(\varepsilon_n) = \frac{h^2}{2} y''(x_n) + O(h^3)$$

$$\varepsilon_n \in (x_n, x_{n+1})$$

其中  $(h^2 / 2)y''(x_n)$  称局部截断误差主项。

故 $T_{n+1}=O(h^2)$ ,  $p=1$ ,

即Euler方法具1阶精度。

## 梯形方法的局部截断误差:

(设  $y_n = y(x_n)$ )

$$T_{n+1} = y(x_{n+1}) - y(x_n) - \frac{h}{2}[y'(x_n) + y'(x_{n+1})]$$

$$= hy'(x_n) + \frac{h^2}{2} y''(x_n) + \frac{h^3}{3!} y'''(x_n)$$

$$- \frac{h}{2}[y'(x_n) + y'(x_n) + hy''(x_n) + \frac{h^2}{2} y'''(x_n)] + O(h^4)$$

$$= -\frac{h^3}{12} y'''(x_n) + O(h^4)$$

故  $T_{n+1} = O(h^3)$ , **p=2**,

梯形方法具2阶精度。

局部截断误差主项为:

$$-(h^3 / 12) y'''(x_n)$$

## 9.3 Runge-Kutta法

考虑改进Euler法

$$\bar{y}_{k+1} = y_k + hf(x_k, y_k)$$

$$y_{k+1} = y_k + \frac{h}{2}[f(x_k, y_k) + f(x_{k+1}, \bar{y}_{k+1})]$$

如果将其改成

$$\left\{ \begin{array}{l} y_0 = y(x_0) \\ K_1 = f(x_k, y_k) \\ K_2 = f(x_{k+1}, y_k + hK_1) \\ y_{k+1} = y_k + \frac{h}{2}(K_1 + K_2) \end{array} \right. \quad \text{-----}(1)$$

改进Euler法是由梯形公式和Euler公式复合而成

梯形公式具有2阶精度

同样可以证明,改进Euler法也具有2阶精度

形如(1)式的求解公式称为二阶Runge-Kutta法

# 显式龙格-库塔法的一般形式

若要使得到的公式阶数  $p$  更大,  $\phi$  就必须包含更多的  $f$  值。为此积分形式的公式表示为

$$y(x_{n+1}) - y(x_n) = \int_{x_n}^{x_{n+1}} f(x, y(x)) dx \approx h \sum_{i=1}^r c_i f(x_n + \lambda_i h, y(x_n + \lambda_i h))$$

或表示为 
$$y_{n+1} = y_n + h\phi(x_n, y_n, h)$$

其中 
$$\phi(x_n, y_n, h) = \sum_{i=1}^r c_i K_i$$

$$K_1 = f(x_n, y_n)$$

$$K_i = f(x_n + \lambda_i h, y_n + h \sum_{j=1}^{i-1} \mu_{ij} K_j)$$

这里  $c_i, \lambda_i, \mu_{ij}$  均为常数。称为  $r$  级显式龙格-库塔(Runge-Kutta)法, 简称  $R-K$  方法。

## Runge-Kutta方法的基本思想

由Taylor展式

$$y(x_{n+1}) = y(x_n + h) \\ \approx y_{n+1} = y(x_n) + hy'(x_n) + \frac{h^2}{2} y''(x_n) + \cdots + \frac{h^p}{p!} y^{(p)}(x_n)$$

$T_{n+1} = O(h^{p+1})$ , 若提高 $p$ , 可提高精度。

但因

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

$$y''(x) = f'_x(x, y) + f'_y(x, y) \cdot f(x, y)$$

.....

高阶导数计算复杂, 故可从另外角度考虑。



## 分析Euler公式及改进的Euler公式:

$$\begin{cases} y_{n+1} = y_n + hK_1 \\ K_1 = f(x_n, y_n) \end{cases}$$

$$\begin{cases} y_{n+1} = y_n + h\left(\frac{K_1}{2} + \frac{K_2}{2}\right) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + h, y_n + hK_1) \end{cases}$$

局部截断误差:  $O(h^2)$

局部截断误差:  $O(h^3)$

可用 $f(x,y)$ 在某些点处值的线性组合得 $y_{n+1}$ , 增加计算 $f(x,y)$ 的次数可提高阶数。

## Runge-Kutta方法的基本思想:

设法计算 $f(x,y)$ 在某些点上的函数值，然后对这些函数值作线性组合，构造近似计算公式，再把近似公式和解的泰勒展开式相比较，使前面的若干项吻合，从而获得达到一定精度的数值计算公式。

设

$$\begin{cases} y_{n+1} = y_n + h \sum_{i=1}^r c_i K_i \\ K_1 = f(x_n, y_n) \\ K_i = f(x_n + \lambda_i h, y_n + h \sum_{j=1}^{i-1} \mu_{ij} K_j), (i = 2, 3, \dots, r) \end{cases}$$

$c_i, \lambda_i, \mu_{ij}$  为待定常数。

上面第一个式子的右端在 $(x_n, y_n)$ 作泰勒展开后,  
按 $h$ 的幂次作升序排列 :

$$y_{n+1} = y_n + \gamma_1 h + \frac{1}{2!} \gamma_2 h^2 + \frac{1}{3!} \gamma_3 h^3 + \dots$$

再与初值问题的精确解 $y(x)$ 在点 $x=x_n$ 处的泰勒展开式

$$y(x_{n+1}) = y(x_n + h) = y(x_n) + hy'(x_n)$$

$$+ \frac{h^2}{2!} y''(x_n) + \dots + \frac{h^p}{p!} y^{(p)}(x_n) + O(h^{p+1})$$

相比较, 使其有尽可能多的项重合。

例如，要求

$$\gamma_1 = f_n, \gamma_2 = f'_n, \gamma_3 = f''_n, \dots, \gamma_p = f_n^{(p-1)}$$

就得到p个方程，从而定出参数 $c_i$ ， $\lambda_i$ ， $\mu_{ij}$ ，再代入 $K_1$ ， $K_2$ ，...， $K_r$ 的表达式，就可得到计算微分方程初值问题的数值计算公式：

$$y_{n+1} = y_n + h \sum_{i=1}^r c_i K_i$$

上式称为**r 级Runge-Kutta方法**的计算公式。

若  $T_{n+1} = O(h^{p+1})$ ，则称其为**p 阶r 级R-K方法**。

当 $r=1$ 时，就是**Euler方法**。

要使Runge-Kutta公式具有更高的阶 $p$ ，就要增加 $r$ 的值。下面我们只就 $r=2$ 推导R-K方法。

### 9.3.2 二阶Runge-Kutta方法

$$\begin{cases} y_{n+1} = y_n + h(c_1 K_1 + c_2 K_2) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \lambda_2 h, y_n + \mu_{21} h K_1) \end{cases}$$

其中  $c_1, c_2, \lambda_2, \mu_{21}$  待定。

上式的局部截断误差为：

$$\begin{aligned} T_{n+1} = & y(x_{n+1}) - y_n \\ & - h[c_1 f(x_n, y_n) + c_2 f(x_n + \lambda_2 h, y_n + \mu_{21} h f_n)] \end{aligned}$$

由

$$y(x_{n+1}) = y_n + hy'_n + \frac{h^2}{2!} y''_n + O(h^3)$$

利用二元函数的Taylor展开，得

$$\begin{cases} y'_n = f(x_n, y_n) = f_n \\ y''_n = \frac{d}{dx} f(x_n, y(x_n)) = f'_x(x_n, y_n) + f'_y(x_n, y_n) \cdot f_n \end{cases}$$

$$\begin{aligned} \text{又 } f(x_n + \lambda_2 h, y_n + \mu_{21} h f_n) \\ = f_n + f'_x(x_n, y_n) \lambda_2 h + f'_y(x_n, y_n) \mu_{21} h f_n + O(h^2) \end{aligned}$$

代入 $T_{n+1}$ 的表达式，得

$$T_{n+1} = y(x_{n+1}) - y_n - h[c_1 f(x_n, y_n) + c_2 f(x_n + \lambda_2 h, y_n + \mu_{21} h f_n)]$$

$$= y_n + h y'_n + \frac{h^2}{2!} y''_n + O(h^3) - y_n - h[c_1 f(x_n, y_n) + c_2 f(x_n + \lambda_2 h, y_n + \mu_{21} h f_n)]$$

$$= h f_n + \frac{h^2}{2} [f'_x(x_n, y_n) + f'_y(x_n, y_n) f_n] - h[c_1 f_n + c_2 (f_n + \lambda_2 f'_x(x_n, y_n) h + \mu_{21} f'_y(x_n, y_n) f_n h)] + O(h^3)$$

$$= (1 - c_1 - c_2) f_n h + (\frac{1}{2} - c_2 \lambda_2) f'_x(x_n, y_n) h^2 + (\frac{1}{2} - c_2 \mu_{21}) f'_y(x_n, y_n) f_n h^2 + O(h^3)$$



要使上式 $p=2$ 阶，则需

$$\begin{cases} 1 - c_1 - c_2 = 0 \\ \frac{1}{2} - c_2 \lambda_2 = 0 \\ \frac{1}{2} - c_2 \mu_{21} = 0 \end{cases}$$

即

$$\begin{cases} c_1 + c_2 = 1 \\ c_2 \lambda_2 = \frac{1}{2} \\ c_2 \mu_{21} = \frac{1}{2} \end{cases}$$

方程组解不唯一，可令 $c_2 = a \neq 0$ ，则

$$c_1 = 1 - a, \quad \lambda_2 = \mu_{21} = 1/(2a)$$

满足上述条件的公式都为2阶R-K公式。

如取 $a = \frac{1}{2}$ ，则 $c_1 = c_2 = \frac{1}{2}$ ， $\lambda_2 = \mu_{21} = 1$ ，即为改进Euler公式。

若取 $a = 1$ ，则 $c_1 = 0$ ， $c_2 = 1$ ， $\lambda_2 = \mu_{21} = \frac{1}{2}$ ，得

$$\begin{cases} y_{n+1} = y_n + hK_2 \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1) \end{cases}$$

称中点公式(公式 3.10)，相当于数值积分的中矩形公式：

$$y_{n+1} = y_n + hf(x_n + \frac{h}{2}, y_n + \frac{h}{2}f(x_n, y_n))$$

### 9.3.3 三阶与四阶Runge-Kutta方法

当 $r=3$ 时，R-K公式表示为

$$\begin{cases} y_{n+1} = y_n + h(c_1 K_1 + c_2 K_2 + c_3 K_3) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \lambda_2 h, y_n + \mu_{21} h K_1) \\ K_3 = f(x_n + \lambda_3 h, y_n + \mu_{31} h K_1 + \mu_{32} h K_2) \end{cases}$$

其中  $c_1, c_2, c_3, \lambda_2, \mu_{21}, \lambda_3, \mu_{31}, \mu_{32}$  为8个待定常数。

上式的局部截断误差为

$$T_{n+1} = y_{n+1} - y_n - h[c_1 K_1 + c_2 K_2 + c_3 K_3]$$

类似二阶的推导过程，将 $K_2, K_3$ 按二元函数展开，使 $T_{n+1} = O(h^4)$ ，得

$$\begin{cases} c_1 + c_2 + c_3 = 1 \\ \lambda_2 = \mu_{21} \\ \lambda_3 = \mu_{31} + \mu_{32} \\ c_2 \lambda_2 + c_3 \lambda_3 = \frac{1}{2} \\ c_2 \lambda_2^2 + c_3 \lambda_3^2 = \frac{1}{3} \\ c_3 \lambda_2 \mu_{32} = \frac{1}{6} \end{cases}$$

方程有8个未知数，解不唯一。

满足该条件的公式统称为**三阶R-K公式**。

其中一个常用公式为 **(P289)**：

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(K_1 + 4K_2 + K_3) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1) \\ K_3 = f(x_n + h, y_n - hK_1 + 2hK_2) \end{cases}$$

当 $r=4$ 时，利用相同的推导过程，经过较复杂的计算，可以得出四阶R-K公式的成立条件。

下列经典公式是其中常用的一个（公式3.13）：

$$\left\{ \begin{array}{l} y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1) \\ K_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_2) \\ K_4 = f(x_n + h, y_n + hK_3) \end{array} \right.$$

**例** 
$$\begin{cases} \frac{dy}{dx} = y - xy^2, & 0 < x \leq 2 \\ y(0) = 1 \end{cases} \quad y(x) = \frac{1}{x-1+2e^{-x}}$$

## 数值实验:几种不同求数值解公式的误差比较

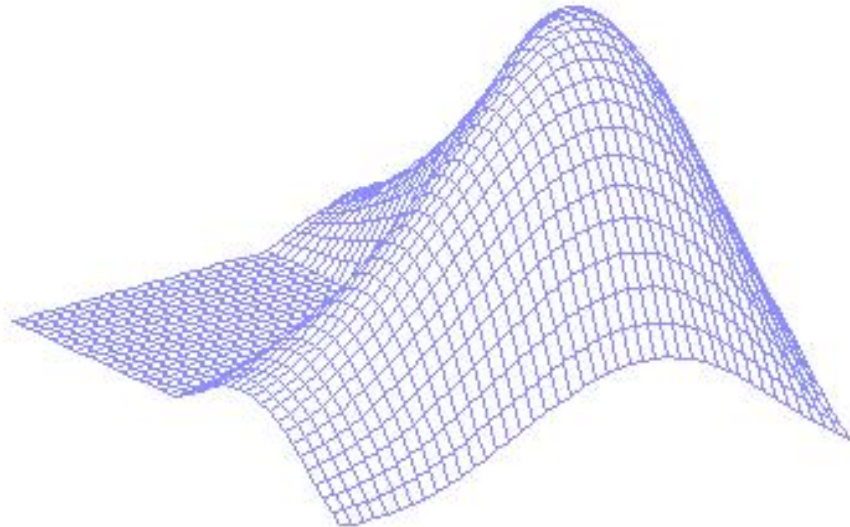
$n$	10	20	30	40
$h$	0.2	0.1	0.0667	0.05
<b>RK4</b>	<b>6.862e-005</b>	<b>3.747e-006</b>	<b>7.071e-007</b>	<b>2.186e-007</b>
<b>RK3</b>	<b>0.0012</b>	<b>1.529e-004</b>	<b>4.517e-005</b>	<b>1.906e-005</b>
<b>RK2</b>	<b>0.0123</b>	<b>0.0026</b>	<b>0.0011</b>	<b>5.9612e-004</b>
<b>Euler</b>	<b>0.1059</b>	<b>0.0521</b>	<b>0.0342</b>	<b>0.0256</b>

**MATLAB求解常微分方程初值问题命令：**

- (1)用临时函数定义一阶微分方程的右端函数；**
- (2)用MATLAB命令ode23()求数值解。**

**使用格式：**  $[T,Y] = \text{ode23}('F',Tspan,y0)$

**其中，  $Tspan = [t_0, t_N]$ 是常微分方程的求解区域，  $y0$ 是解的初值**



## 实验例题1 蛇形曲线的常微分方程初值问题

$$y' = \frac{1}{x^2 + 1} - 2y^2 \quad y(0) = 0$$

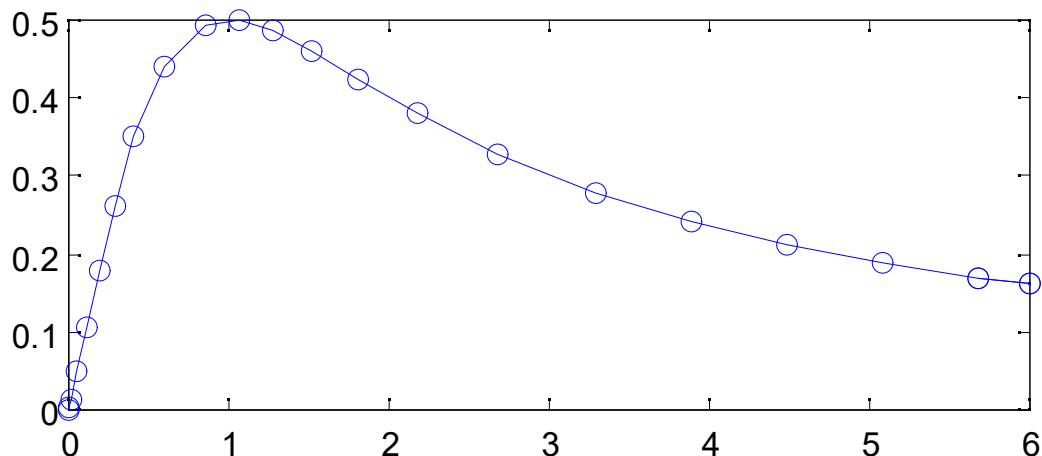
**MATLAB数值求解命令**

```
F=inline('1./(1+x.^2)-2*y.^2');
```

```
ode23(F,[0,6],0)
```

**输出结果为图形**

**[T,y]=ode23(f,[0,6],0)将得到  
自变量和函数的离散数据**





# MATLAB解常微分方程初值问题命令

$$\begin{cases} \frac{dy}{dx} = y - xy^2, & 0 < x \leq 2 \\ y(0) = 1 \end{cases}$$

数值求解命令:

`[x,y] = ode23('f',[a,b],y0)`

`f=inline('y-x.*y.^2');`  
`[x,y]=ode23(f,[0,2],1)`

符号求解命令:

`dsolve('eqn1', ...)`

`syms x y`

`dsolve('Dy=y-x*y^2','y(0)=1','x')`

`ans =`

`1/(x-1+2*exp(-x))`

解析解:

$$y(x) = \frac{1}{x-1+2e^{-x}}$$

# 总结

- 欧拉法，后退欧拉法，改进欧拉法
- Range-Kutta公式

### 9.3.4 变步长的龙格-库塔方法

单从每一步看, 步长越小, 截断误差就越小, 但随着步长的缩小, 在一定求解范围内所要完成的步数就增加了. 步数的增加不但引起计算量的增大, 而且可能导致舍入误差的严重积累. 因此同积分的数值计算一样, 微分方程的数值解法也有个选择步长的问题.

在选择步长时, 需要考虑两个问题:

1. 怎样衡量和检验计算结果的精度?
2. 如何依据所获得的精度处理步长?

我们考察经典的四阶R-K公式(3.13)，从节点 $x_n$ 出发，先以 $h$ 为步长求出一个近似值，记为 $y_{n+1}^{(h)}$ ，由于公式的局部截断误差为 $O(h^5)$ ，故

$$y(x_{n+1}) - y_{n+1}^{(h)} \approx ch^5, \quad (3.14)$$

然后将步长折半，即取为步长 $\frac{h}{2}$ ，从 $x_n$ 跨两步到 $x_{n+1}$ ，再求得一个近似值 $y_{n+1}^{(\frac{h}{2})}$ ，每跨一步的局部截断误差是 $c\left(\frac{h}{2}\right)^5$ ，因此有

$$y(x_{n+1}) - y_{n+1}^{(\frac{h}{2})} \approx 2c\left(\frac{h}{2}\right)^5, \quad (3.15)$$

比较(3.14)式和(3.15)式我们看到，步长折半后，误差大约减少到1/16，即有

$$\frac{y(x_{n+1}) - y_{n+1}^{(\frac{h}{2})}}{y(x_{n+1}) - y_{n+1}^{(h)}} \approx \frac{1}{16}.$$

由此易得下列事后估计式

$$y(x_{n+1}) - y_{n+1}^{(\frac{h}{2})} \approx \frac{1}{15} [y_{n+1}^{(\frac{h}{2})} - y_{n+1}^{(h)}].$$

这样，我们可以通过检查步长，折半前后两次计算结果的偏差

$$\Delta = \left| y_{n+1}^{(\frac{h}{2})} - y_{n+1}^{(h)} \right|$$

来判定所选的步长是否合适，具体地说，将区分以下两种情况处理：

(1) 对于给定的精度 $\varepsilon$ ，如果 $\Delta > \varepsilon$ ，我们反复将步长折半计算，直至 $\Delta < \varepsilon$ 为止，这时取最终得到的 $y_{n+1}^{(\frac{n}{2})}$ 作为结果；

(2) 如果 $\Delta < \varepsilon$ ，我们将反复将步长作加倍计算，直至 $\Delta > \varepsilon$ 为止，这时再将步长折半计算一次，就得到所要的结果。

这种通过加倍或折半处理步长的方法称为变步长方法。表面上看，为了选择步长，每一步的计算量增加了，但总体考虑往往是合算的。

变步长方法还可利用 $p$ 阶与 $p+1$ 阶公式的局部截断误差得到误差控制与变步长的具体方法，参见文献[8]。

## 9.4 单步法的收敛性与稳定性

### 9.4.1 收敛性与相容性

数值解法的基本思想是，通过某种离散化手段将微分方程(1.1)转化为差分方程,如单步法(2.11),即

$$y_{n+1} = y_n + h\varphi(x_n, y_n, h). \quad (4.1)$$

它在点 $x_n$ 处的解为 $y_n$ ，而初值问题(1.1), (1.2) 在点 $x_n$ 处的精确解为 $y(x_n)$ ，记 $e_n = y(x_n) - y_n$ 称为**整体截断误差**. **收敛性**就是讨论当  $x=x_n$  固定且  $h = \frac{x_n - x_0}{n} \rightarrow 0$  时 $e_n \rightarrow 0$ 的问题.

**定义3** 若一种数值方法对于固定的 $x_n=x_0+nh$ , 当 $h\rightarrow 0$ 时有 $y_n\rightarrow y(x_n)$ , 其中 $y(x)$ 是(1.1),(1.2)的准确解, 则称该方法是**收敛**的.

显然数值方法收敛是指 $e_n=y(x_n)-y_n\rightarrow 0$ , 对单步法(4.1)有下述收敛性定理:

**定理3** 假设单步法(4.1)具有 $p$ 阶精度, 且增量函数 $\varphi(x, y, h)$ 关于 $y$ 满足利普希次条件

$$|\varphi(x, y, h) - \varphi(x, \bar{y}, h)| \leq L_\varphi |y - \bar{y}|. \quad (4.2)$$

又设初值 $y_0$ 是准确的, 即 $y_0=f(x_0)$ , 则其**整体截断误差**

$$y(x_n) - y_n = O(h^p). \quad (4.3)$$



**证明** 设以 $\bar{y}_{n+1}$ 表示取 $y_n=y(x_n)$ 用公式(4.1)求得的结果, 即

$$\bar{y}_{n+1} = y(x_n) + h\varphi(x_n, y(x_n), h), \quad (4.4)$$

则 $y(x_n) - \bar{y}_{n+1}$ 为局部截断误差, 由于所给方法具有 $p$ 阶精度, 按定义2, 存在定数 $C$ , 使

$$|y(x_{n+1}) - \bar{y}_{n+1}| \leq Ch^{p+1}.$$

又由式(4.1)与(4.4), 得

$$|\bar{y}_{n+1} - y_{n+1}| \leq |y(x_n) - y_n| + h|\varphi(x_n, y(x_n), h) - \varphi(x_n, y_n, h)|.$$

利用利普希次条件(4.2), 有

$$|\bar{y}_{n+1} - y_{n+1}| \leq (1 + hL_\varphi) |y(x_n) - y_n|.$$

从而有

$$\begin{aligned} |y(x_{n+1}) - y_{n+1}| &\leq |\bar{y}_{n+1} - y_{n+1}| + |y(x_{n+1}) - \bar{y}_{n+1}| \\ &\leq (1 + hL_\varphi) |y(x_n) - y_n| + Ch^{p+1}. \end{aligned}$$

即对整体截断误差 $e_n = y(x_n) - y_n$ 成立下列递推关系式

$$|e_{n+1}| \leq (1 + hL_\varphi) |e_n| + Ch^{p+1}. \quad (4.5)$$

据此不等式反复递推, 可得

$$|e_n| \leq (1 + hL_\varphi)^n |e_0| + \frac{Ch^p}{L_\varphi} [(1 + hL_\varphi)^n - 1]. \quad (4.6)$$

再注意到当 $x=x_0+nh\leq T$ 时

$$(1+hL_\varphi)^n \leq \left(e^{hL_\varphi}\right)^n \leq e^{TL_\varphi},$$

最终得下列估计式

$$|e_n| \leq |e_0|e^{TL_\varphi} + \frac{Ch^p}{L_\varphi}(e^{TL_\varphi} - 1). \quad (4.7)$$

由此可以断定，如果初值是准确的，即 $e_0=0$ ，则(4.3)式成立.

**定理证毕.**

依据这一定理，判断单步法(4.1)的收敛性，归结为验证增量函数 $\varphi$ 能否满足利普希次条件(4.2).

对于欧拉方法，由于其增量函数 $\varphi$ 就是 $f(x, y)$ ，故当 $f(x, y)$ 关于 $y$ 满足利普希次条件时它是收敛的.

再考察改进的欧拉方法，其增量函数 $\varphi$ 已由(3.2)式给出，假定 $f(x, y)$ 关于 $y$ 满足利普希次条件，即

$$|f(x, y) - f(x, \bar{y})| \leq L|y - \bar{y}|.$$

这时有

$$\begin{aligned} |\varphi(x, y, h) - \varphi(x, \bar{y}, h)| &\leq \frac{1}{2} |f(x, y) - f(x, \bar{y})| \\ &\quad + \frac{1}{2} |f(x+h, y+hf(x, y)) - f(x+h, \bar{y}+hf(x, \bar{y}))| \\ &\leq \frac{1}{2} L|y - \bar{y}| + \frac{1}{2} L|y + hf(x, y) - (\bar{y} + hf(x, \bar{y}))| \\ &\leq \frac{1}{2} L|y - \bar{y}| + \frac{1}{2} L|y - \bar{y}| + \frac{1}{2} hL|f(x, y) - f(x, \bar{y})| \\ &\leq L|y - \bar{y}| + \frac{1}{2} hL^2|y - \bar{y}| = L\left(1 + \frac{h}{2} L\right)|y - \bar{y}|. \end{aligned}$$

即

$$\left| \varphi(x, y, h) - \varphi(x, \bar{y}, h) \right| \leq L \left( 1 + \frac{h}{2} L \right) |y - \bar{y}|.$$

设限定步长 $h \leq h_0$  ( $h_0$ 为定数), 上式表明 $\varphi$ 关于 $y$ 的利普希次常数为

$$L_{\varphi} = L \left( 1 + \frac{h}{2} L \right).$$

因此改进的欧拉方法也是收敛的.

类似地, 不难验证其它龙格-库塔方法的收敛性.

定理3表明 $p \geq 1$ 时单步法收敛, 并且当 $y(x)$ 是初值问题(1.1),(1.2)的解, (4.1)具有 $p$ 阶精度时, 则有展开式

$$\begin{aligned} T_{n+1} &= y(x+h) - y(x) - h\varphi(x, y(x), h) \\ &= y'(x)h + \frac{y''(x)}{2}h^2 + \cdots \\ &\quad - h[\varphi(x, y(x), 0) + \varphi'_x(x, y(x), 0)h + \cdots] \\ &= h[y'(x) - \varphi(x, y(x), 0)] + O(h^2). \end{aligned}$$

所以 $p \geq 1$ 的充分必要条件是  $y'(x) - \varphi(x, y(x), 0) = 0$  ,  
而  $y'(x) = f(x, y(x))$  , 于是可给出如下定义:

**定义4** 若单步法(4.1)的增量函数 $\varphi$ 满足

$$\varphi(x, y, 0) = f(x, y),$$

则称单步法(4.1)与初值问题(1.1),(1.2)**相容**.

相容性是指方法逼近微分方程(1.1),即(1.1)离散化得到的数值方法,当 $h \rightarrow 0$ 时可得到 $y'(x) = f(x, y)$ .

于是有下面定理.

**定理4**  $p$ 阶方法(4.1)与初值问题(1.1),(1.2)相容的充分必要条件是 $p \geq 1$ .

以上讨论表明 $p$ 阶方法(4.1)当 $p \geq 1$ 时与(1.1), (1.2)相容, 反之**相容方法至少是1阶的**.

于是由定理3可知方法(4.1)**收敛的充分必要条件**是**此方法是相容的**.

## 9.4.2 绝对稳定性与绝对稳定域

前面关于收敛性的讨论有个前提，必须假定数值方法本身的计算是准确的。实际情形并不是这样，差分方程的求解还会有**计算误差**。譬如由于数字舍入而引起的小扰动。这类小扰动在传播过程中会不会恶性增长，以至于“**淹没**”了差分方程的“**真解**”呢？这就是**差分方程的稳定性问题**。在实际计算时，我们希望某一步产生的扰动值，在后面的计算中**能够被控制**，甚至是**逐步衰减**的。



**定义5** 若一种数值方法在节点值 $y_n$ 上大小为 $\delta$ 扰动, 于以后各节点值 $y_m(m>n)$ 上产生的偏差均不超过 $\delta$ , 则称该方法是**稳定**的.

下面以欧拉法为例考察计算稳定性.

**例4** 用欧拉公式求解初值问题

$$\begin{cases} y' = -100y, \\ y(0) = 1. \end{cases}$$

其准确解  $y(x) = e^{-100x}$  是一个按指数曲线衰减很快的函数.(见书p294的图9-3)

**解** 用欧拉法解方程 $y' = -100y$  得

$$y_{n+1} = (1 - 100h)y_n.$$

若取步长 $h=0.025$ ，则欧拉公式的具体形式为

$$y_{n+1} = -1.5y_n.$$

计算结果见表，明显计算过程不稳定，但取 $h=0.005$ ， $y_{n+1}=0.5y_n$ ，则计算过程稳定。

对后退的欧拉公式，取 $h=0.025$ 时，则计算公式为 $y_{n+1}=(1/3.5)y_n$ 。计算结果见表，这时计算过程是稳定的。

节点 $x_n$	欧拉方法 $y_n$	后退欧拉方法 $y_n$
0.025	-1.5	0.2857
0.050	2.25	0.0816
0.075	-3.375	0.0233
0.100	5.0625	0.0067

例题表明稳定性不但与方法有关，也与步长 $h$ 有关，当然与方程中的 $f(x, y)$ 有关. 为了只考察数值方法本身，通常只检验数值方法用于解模型方程的稳定性，模型方程为

$$y' = \lambda y, \quad (4.8)$$

其中 $\lambda$ 为复数，这个方程分析较简单，对一般方程可以通过局部线性化化为这种形式，例如在 $(\bar{x}, \bar{y})$ 的邻域，可展开为

$$\begin{aligned} y' &= f(x, y) \\ &= f(\bar{x}, \bar{y}) + f'_x(\bar{x}, \bar{y})(x - \bar{x}) + f'_y(\bar{x}, \bar{y})(y - \bar{y}) + \cdots, \end{aligned}$$

略去高阶项，再做变换即可得到 $u'=\lambda u$ 的形式. 对于 $m$ 个方程的常微分方程组, 可线性化为 $y'=Ay$ , 这里矩阵 $A$ 为 $m \times m$ 雅可比矩阵 $(\partial f_i / \partial y_j)$ , 若矩阵 $A$ 有 $m$ 个特征值 $\lambda_1, \lambda_2, \dots, \lambda_m$ , 其中 $\lambda_i$ 可能是复数, 所以, 为了使模型方程结果能推广到常微分方程组, 方程(4.8)式中 $\lambda$ 为复数. 为保证微分方程本身的稳定性, 还应假定 $\text{Re}(\lambda) < 0$ .

下面先研究欧拉方法的稳定性. 模型方程 $y'=\lambda y$ 的欧拉公式为

$$y_{n+1} = (1 + h\lambda)y_n. \quad (4.9)$$

设在节点  $y_n$  上有一扰动值  $\varepsilon_n$ ，它的传播使节点值  $y_{n+1}$  产生大小为  $\varepsilon_{n+1}$  的扰动值，假设用  $y_n^* = y_n + \varepsilon_n$ ，按欧拉公式得出  $y_{n+1}^* = y_{n+1} + \varepsilon_{n+1}$  的计算过程不再有新的误差，则扰动值满足

$$\varepsilon_{n+1} = (1 + h\lambda)\varepsilon_n.$$

可见扰动值满足原来的差分方程(4.9). 这样，如果差分方程的解是不增长的，即有

$$|y_{n+1}| \leq |y_n|.$$

则它就是稳定的. 这一论断对于下面将要研究的其它方法同样适用.

显然，为要保证差分方程(4.9)的解是不增长的，只要选取 $h$ 充分小，使

$$|1 + h\lambda| \leq 1. \quad (4.10)$$

在 $\mu=h\lambda$ 的复平面上，这是以 $(-1,0)$ 为圆心，1为半径的单位圆内部. 称为欧拉法的**绝对稳定域**，一般情形可由下面定义.

**定义6** 单步法(4.1)用于解模型方程 $y'=\lambda y$ ，若得到的解 $y_{n+1}=E(h\lambda)y_n$ ，满足 $|E(h\lambda)|<1$ ，则称方法(4.1)是**绝对稳定**的. 在 $\mu=h\lambda$ 的平面上，使 $|E(h\lambda)|<1$ 的变量围成的区域，称为**绝对稳定区域**，它与实轴的交称为**绝对稳定区间**.

对欧拉法 $E(h\lambda)=1+h\lambda$ ，其绝对稳定域为 $|1+h\lambda|<1$ ，绝对稳定区间为 $-2<h\lambda<0$ ，在例5中 $\lambda=-100$ ， $-2<-100h<0$ ，即 $0<h<2/100=0.02$ 为稳定区间，在例4中取 $h=0.025$ ，故它是不稳定的，当取 $h=0.005$ 时它是稳定的。

对二阶R-K方法，解模型方程(4.1)可得到

$$y_{n+1} = \left[ 1 + h\lambda + \frac{(h\lambda)^2}{2} \right] y_n,$$

故

$$E(h\lambda) = 1 + h\lambda + \frac{(h\lambda)^2}{2}.$$

绝对稳定域由 $|E(h\lambda)|<1$ 得到，于是可得绝对稳定区间为 $-2<h\lambda<0$ ，即 $0<h<2/\lambda$ 。

类似可得三阶及四阶**R-K**方法的 $E(h\lambda)$ 分别为

$$E(h\lambda) = 1 + h\lambda + \frac{(h\lambda)^2}{2!} + \frac{(h\lambda)^3}{3!}.$$

$$E(h\lambda) = 1 + h\lambda + \frac{(h\lambda)^2}{2!} + \frac{(h\lambda)^3}{3!} + \frac{(h\lambda)^4}{4!}.$$

由 $|E(h\lambda)| < 1$ 可得到相应的**绝对稳定域**. 当 $\lambda$ 为实数时, 则得**绝对稳定区间**, 它们分别为(图形复杂p296)

三阶显式**R-K**方法:  $-2.51 < h\lambda < 0$ , 即  $0 < h < -2.51/\lambda$ .

四阶显式**R-K**方法:  $-2.78 < h\lambda < 0$ , 即  $0 < h < -2.78/\lambda$ .

从以上讨论可知显式**R-K**方法的**绝对稳定域**均为有限域, 都对步长 $h$ 有限制. 如果 $h$ 不在所给的**绝对稳定区间**内, 方法就不稳定.



**例5** 分别取 $h=0.1$ 及 $h=0.2$ ，用经典的四阶**R-K**方法(3.13)计算初值问题

$$\begin{cases} y' = -20y, & (0 \leq x \leq 1) \\ y(0) = 1. \end{cases}$$

**解** 本例 $\lambda=-20$ ， $h\lambda$ 分别为 $-2$ 及 $-4$ ，前者在**绝对稳定区间**内，后者则不在，用四阶**R-K**方法计算其误差见下表

$x_n$	0.2	0.4	0.6	0.8	1.0
$h=0.1$	$0.93 \times 10^{-1}$	$0.12 \times 10^{-1}$	$0.14 \times 10^{-2}$	$0.15 \times 10^{-3}$	$0.17 \times 10^{-4}$
$h=0.2$	4.98	25.0	125.0	625.0	3125.0

从以上结果看到，如果步长 $h$ 不满足**绝对稳定条件**，误差增长很快。

对隐式单步法, 可以同样讨论方法的绝对稳定性, 例如对后退欧拉法, 用它解模型方程可得

$$y_{n+1} = \frac{1}{1 - h\lambda} y_n,$$

故

$$E(h\lambda) = \frac{1}{1 - h\lambda}.$$

由 $|E(h\lambda)| < 1$ 可得绝对稳定域为 $|1 - h\lambda| > 1$ , 这是以 $(1, 0)$ 为圆心, 1为半径的单位圆外部. 故方法的绝对稳定区间为 $-\infty < h\lambda < 0$ . 当 $\lambda < 0$ 时, 则 $0 < h < \infty$ , 即对任何步长均为稳定的.

对隐式梯形法，它用于解模型方程(4.8)得

$$y_{n+1} = \frac{1 + \frac{h\lambda}{2}}{1 - \frac{h\lambda}{2}} y_n,$$

故

$$E(h\lambda) = \frac{1 + \frac{h\lambda}{2}}{1 - \frac{h\lambda}{2}}.$$

对 $\text{Re}(\lambda) < 0$ 有 $|E(h\lambda)| < 1$ ，故绝对稳定域为 $\mu = h\lambda$ 的左半平面，绝对稳定区间为 $-\infty < h\lambda < 0$ ，即 $0 < h < \infty$ 时隐式梯形法均是稳定的。

分析得到隐式欧拉法与梯形方法的绝对稳定域均为 $\{h\lambda \mid \operatorname{Re}(h\lambda) < 0\}$ ，在具体计算中步长 $h$ 的选取只需考虑计算精度及迭代收敛性要求而不必考虑稳定性，具有这种特点的方法需特别重视，由此给出下面的定义.

**定义7** 如果数值方法的绝对稳定域包含了集合 $\{h\lambda \mid \operatorname{Re}(h\lambda) < 0\}$ ，那么称此方法是A-稳定的.

由定义知A-稳定方法对步长 $h$ 没有限制.

## 9.5 线性多步法

在逐步推进的求解过程中, 计算 $y_{n+1}$ 之前事实上已经求出了一系列的近似值 $y_0, y_1, \dots, y_n$ , 如果充分利用前面多步的信息来预测 $y_{n+1}$ , 则可以期望会获得较高的精度. 这就是构造所得线性多步法的基本思想.

构造多步法的主要途径基于数值积分方法和基于泰勒展开方法, 前者可直接由方程(1.1)两端积分后利用插值求积公式得到. 本节主要介绍基于泰勒展开的构造方法.

### 9.5.1 线性多步法的一般公式

如果计算 $y_{n+k}$ 时, 除用 $y_{n+k-1}$ 的值, 还要用到 $y_{n+i}$  ( $i=0,1,\cdots,k-2$ )的值, 则称此方法为**线性多步法**. 一般的**线性多步法公式**可表示为

$$y_{n+k} = \sum_{i=0}^{k-1} \alpha_i y_{n+i} + h \sum_{i=0}^k \beta_i f_{n+i}, \quad (5.1)$$

其中 $y_{n+1}$ 为 $y(x_{n+1})$ 的近似,  $f_{n+i}=f(x_{n+i}, y_{n+i})$ , 这里 $x_{n+i}=x_n+ih$ ,  $\alpha_i, \beta_i$ 为常数,  $\alpha_0$ 及 $\beta_0$ 不全为零, 则称(5.1)为**线性 $k$ 步法**, 计算时需先给出前面 $k$ 个近似值 $y_0, y_1, \cdots, y_{k-1}$ , 再由(5.1)逐次求出 $y_k, y_{k+1}, \cdots$ .

如果 $\beta_k=0$ ，则(5.1)称为**显式 $k$ 步法**，这时 $y_{n+k}$ 可直接由(5.1)算出；如果 $\beta_k\neq 0$ ，则(5.1)称为**隐式 $k$ 步法**，求解时与梯形法(2.7)相同，要用迭代法方可算出 $y_{n+k}$ 。(5.1)中系数 $\alpha_i$ 及 $\beta_i$ 可根据方法的局部截断误差及阶确定，其定义为

**定义8** 设 $y(x)$ 是初值问题(1.1), (1.2)的准确解，线性多步法(5.1)在 $x_{n+k}$ 上**局部截断误差**为

$$\begin{aligned} T_{n+k} &= L[y(x_n); h] \\ &= y(x_{n+k}) - \sum_{i=0}^{k-1} \alpha_i y(x_{n+i}) - h \sum_{i=0}^k \beta_i y'(x_{n+i}). \end{aligned} \quad (5.2)$$

若 $T_{n+k}=O(h^{p+1})$ ，则称方法(5.1)是 **$p$ 阶的**，如果 $p\geq 1$ ，则称方法(5.1)与微分方程(1.1)是**相容的**。

由定义8, 对 $T_{n+k}$ 在 $x_n$ 处泰勒展开, 由于

$$y(x_n + ih) = y(x_n) + ihy'(x_n) + \frac{(ih)^2}{2!} y''(x_n) + \frac{(ih)^3}{3!} y'''(x_n) + \cdots,$$

$$y'(x_n + ih) = y'(x_n) + ihy''(x_n) + \frac{(ih)^2}{2!} y'''(x_n) + \cdots.$$

代入(5.2)得

$$\begin{aligned} T_{n+k} = & c_0 y(x_n) + c_1 h y'(x_n) + c_2 h^2 y''(x_n) \\ & + \cdots + c_p h^p y^{(p)}(x_n) + \cdots, \end{aligned} \quad (5.3)$$



其中

$$\left. \begin{aligned} c_0 &= 1 - (\alpha_0 + \alpha_1 + \cdots + \alpha_{k-1}), \\ c_1 &= k - [\alpha_1 + 2\alpha_2 + \cdots + (k-1)\alpha_{k-1}] \\ &\quad - (\beta_0 + \beta_1 + \cdots + \beta_k), \\ c_q &= \frac{1}{q!} [k^q - (\alpha_1 + 2^q \alpha_2 + \cdots + (k-1)^q \alpha_{k-1})] \\ &\quad - \frac{1}{(q-1)!} (\beta_0 + 2^{q-1} \beta_1 + \cdots + k^{q-1} \beta_k) \\ &\quad q = 2, 3, \cdots \end{aligned} \right\} \quad (5.4)$$

若在公式(5.1)中选择系数 $\alpha_i$ 及 $\beta_i$ ，使它满足

$$c_0 = c_1 = \cdots = c_p = 0, \quad c_{p+1} \neq 0.$$

由定义可知此时所构造的多步法是 $p$ 阶的，且

$$T_{n+k} = c_{p+1} h^{p+1} y^{(p+1)}(x_n) + O(h^{p+2}). \quad (5.5)$$

称右端第一项为局部截断误差主项， $c_{p+1}$ 称为误差常数。

根据相容性定义， $p \geq 1$ ，即 $c_0 = c_1 = 0$ ，由(5.4)得

$$\begin{cases} \alpha_0 + \alpha_1 + \cdots + \alpha_{k-1} = 1, \\ \sum_{i=1}^{k-1} i \alpha_i + \sum_{i=0}^k \beta_i = k. \end{cases} \quad (5.6)$$

故线性多步法(5.1)与微分方程(1.1)相容的充分必要条件是(5.6)成立。

显然，当 $k=1$ 时，若 $\beta_1=0$ ，则由(5.6)可求得

$$\alpha_0=1, \beta_0=1.$$

此时公式(5.1)为

$$y_{n+1} = y_n + hf_n,$$

即为欧拉法. 从(5.4)可求得 $c_2=1/2 \neq 0$ ，故方法为1阶精度，且局部截断误差为

$$T_{n+1} = \frac{1}{2}h^2 y''(x_n) + O(h^3).$$

这 and 第2节给出的定义及结果是一致的.

对 $k=1$ ，若 $\beta_1 \neq 0$ ，此时方法为隐式公式，为了确定系数 $\alpha_0, \beta_0, \beta_1$ ，可由 $c_0=c_1=c_2=0$ 解得 $\alpha_0=1, \beta_0=\beta_1=1/2$ 。于是得到公式

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

即为梯形公式。

由(5.4)可求得 $c_2=-1/12$ ，故 $p=2$ ，所以梯形法是二阶方法，其局部截断误差主项是

$$-\frac{h^3}{12}y'''(x_n).$$

这与9.2节中讨论是一致的。

对 $k \geq 2$ 的多步法公式都可利用(5.4)确定系数 $\alpha_i, \beta_i$ ，并由(5.5)式给出局部截断误差，下面只就若干常用的多步法导出具体公式。

## 9.7 一阶方程组与刚性方程组

### 9.7.1 一阶方程组

前面我们研究了单个方程 $y'=f$ 的数值解，只要把 $y$ 和 $f$ 理解为向量，那么，所提供的各种计算公式即可应用到一阶方程组的情形。

考察一阶方程组

$$y'_i = f_i(x, y_1, y_2, \dots, y_N), (i=1, 2, \dots, N)$$

的初值问题，初始条件给为

$$y_i(x_0) = y_i^0, (i=1, 2, \dots, N).$$

若采用向量的记号，记(向量)

$$y = (y_1, y_2, \cdots, y_N)^T,$$

$$y_0 = (y_1^0, y_2^0, \cdots, y_N^0)^T,$$

$$f = (f_1, f_2, \cdots, f_N)^T.$$

则上述方程组的初值问题可表示为

$$\begin{cases} y' = f(x, y), \\ y(x_0) = y_0. \end{cases} \quad (7.1)$$

求解这一初值问题的四阶龙格-库塔公式为(向量)

$$y_{n+1} = y_n + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4),$$

式中(向量)

$$\begin{cases} k_1 = f(x_n, y_n), \\ k_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1), \\ k_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2), \\ k_4 = f(x_n + h, y_n + hk_3). \end{cases}$$

或表示为(分量)

$$y_{i,n+1} = y_{in} + \frac{h}{6}(K_{i1} + 2K_{i2} + 2K_{i3} + K_{i4})$$

$$(i = 1, 2, \dots, N),$$

其中(分量)

$$\begin{cases} K_{i1} = f_i(x_n, y_{1n}, y_{2n}, \dots, y_{Nn}), \\ K_{i2} = f_i(x_n + \frac{h}{2}, y_{1n} + \frac{h}{2}K_{11}, y_{2n} + \frac{h}{2}K_{21}, \dots, y_{Nn} + \frac{h}{2}K_{N1}), \\ K_{i3} = f_i(x_n + \frac{h}{2}, y_{1n} + \frac{h}{2}K_{12}, y_{2n} + \frac{h}{2}K_{22}, \dots, y_{Nn} + \frac{h}{2}K_{N2}), \\ K_{i4} = f_i(x_n + h, y_{1n} + hK_{13}, y_{2n} + hK_{23}, \dots, y_{Nn} + hK_{N3}). \end{cases}$$

这里 $y_{in}$ 是第 $i$ 个因变量 $y_i(x)$ 在节点 $x_n = x_0 + nh$ 的近似值.



为了帮助理解这一公式的计算过程，我们考察两个方程的特殊情形

$$\begin{cases} y' = f(x, y, z), \\ z' = g(x, y, z), \\ y(x_0) = y_0, \\ z(x_0) = z_0. \end{cases}$$

这时四阶龙格-库塔公式具有形式

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4), \\ z_{n+1} = z_n + \frac{h}{6}(L_1 + 2L_2 + 2L_3 + L_4). \end{cases} \quad (7.2)$$

其中

$$\left\{ \begin{array}{l} K_1 = f(x_n, y_n, z_n), \\ K_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1, z_n + \frac{h}{2}L_1), \\ K_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_2, z_n + \frac{h}{2}L_2), \\ K_4 = f(x_n + h, y_n + hK_3, z_n + hL_3), \\ L_1 = g(x_n, y_n, z_n), \\ L_2 = g(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1, z_n + \frac{h}{2}L_1), \\ L_3 = g(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_2, z_n + \frac{h}{2}L_2), \\ L_4 = g(x_n + h, y_n + hK_3, z_n + hL_3). \end{array} \right. \quad (7.3)$$

这是一步法，利用节点 $x_n$ 上的值 $y_n, z_n$ ，由(7.3)式顺序计算 $K_1, L_1, K_2, L_2, K_3, L_3, K_4, L_4$ ，然后代入(7.2)式即可求得节点 $x_{n+1}$ 上的 $y_{n+1}, z_{n+1}$ 。

## 9.7.2 化高阶方程为一阶方程组

关于高阶微分方程(或方程组)的初值问题, 原则上总可以归结为一阶方程组来求解. 例如, 考察下列  $m$  阶微分方程

$$y^{(m)} = f(x, y, y', \cdots, y^{(m-1)}), \quad (7.4)$$

初始条件为

$$y(x_0) = y_0, y'(x_0) = y'_0, \cdots, y^{(m-1)}(x_0) = y_0^{(m-1)}. \quad (7.5)$$

只要引进新的变量

$$y_1 = y, y_2 = y', \cdots, y_m = y^{(m-1)}.$$

即可将  $m$  阶方程(7.4)化为如下的一阶方程组:

$$\begin{cases} y_1' = y_2, \\ y_2' = y_3, \\ \dots\dots\dots \\ y_{m-1}' = y_m \\ y_m' = f(x, y_1, y_2, \dots, y_m). \end{cases} \quad (7.6)$$

初始条件(7.5)则相应地化为

$$y_1(x_0) = y_0, y_2(x_0) = y_0', \dots, y_m(x_0) = y_0^{(m-1)}. \quad (7.7)$$

不难证明初始问题(7.4),(7.5)和初始问题(7.6),(7.7)是彼此等价的.

特别地，对于下列二阶方程的问题

$$\begin{cases} y'' = f(x, y, y'), \\ y(x_0) = y_0, \\ y'(x_0) = y'_0. \end{cases}$$

引进新变量 $z=y'$ ，即可化为下列一阶方程组的初值问题：

$$\begin{cases} y' = z, \\ z' = f(x, y, z), \\ y(x_0) = y_0, \\ z(x_0) = y'_0. \end{cases}$$

针对这个问题应用四阶龙格-库塔公式(7.2), 有

$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4), \\ z_{n+1} = z_n + \frac{h}{6}(L_1 + 2L_2 + 2L_3 + L_4). \end{cases}$$

由(7.3)式可得

$$K_1 = z_n, \quad L_1 = f(x_n, y_n, z_n);$$

$$K_2 = z_n + \frac{h}{2}L_1, \quad L_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1, z_n + \frac{h}{2}L_1);$$

$$K_3 = z_n + \frac{h}{2}L_2, \quad L_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_2, z_n + \frac{h}{2}L_2);$$

$$K_4 = z_n + hL_3, \quad L_4 = f(x_n + h, y_n + hK_3, z_n + hL_3).$$

如果消去 $K_1, K_2, K_3, K_4$ ，则上述格式可表示为

$$\begin{cases} y_{n+1} = y_n + h z_n + \frac{h^2}{6}(L_1 + L_2 + L_3), \\ z_{n+1} = z_n + \frac{h}{6}(L_1 + 2L_2 + 2L_3 + L_4). \end{cases}$$

这里

$$L_1 = f(x_n, y_n, z_n),$$

$$L_2 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}z_n, z_n + \frac{h}{2}L_1\right),$$

$$L_3 = f\left(x_n + \frac{h}{2}, y_n + \frac{h}{2}z_n + \frac{h^2}{4}L_1, z_n + \frac{h}{2}L_2\right),$$

$$L_4 = f\left(x_n + h, y_n + h z_n + \frac{h^2}{2}L_2, z_n + h L_3\right).$$



一阶常微分方程初值问题 
$$\begin{cases} y' = f(x, y) & x \geq x_0 \\ y(x_0) = y_0 \end{cases}$$

欧拉公式:  $y_{n+1} = y_n + h f(x_n, y_n) \quad (n = 0, 1, 2, \dots, N)$

修改的欧拉公式:

$$\begin{cases} y_{n+1} = y_n + \frac{h}{2} [k_1 + k_2] & (n = 0, 1, 2, \dots, N) \\ k_1 = f(x_n, y_n), \\ k_2 = f(x_{n+1}, y_n + h k_1) \end{cases}$$

一阶常微分方程组  
初值问题

$$\begin{cases} \frac{dx}{dt} = f_1(t, x, y) & x(t_0) = x_0 \\ \frac{dy}{dt} = f_2(t, x, y) & y(t_0) = y_0 \end{cases}$$

# 一阶常微分方程组的向量表示

记  $Y = \begin{bmatrix} x \\ y \end{bmatrix}$        $F(t, Y) = \begin{bmatrix} f_1(t, Y) \\ f_2(t, Y) \end{bmatrix}$

$$\begin{cases} \frac{dY}{dt} = F(t, Y) & t > t_0 \\ Y(t_0) = Y_0 \end{cases}$$

欧拉公式:

$$Y_{n+1} = Y_n + hF(t_n, Y_n) \quad (n = 0, 1, \dots, N-1)$$

$$Y_0 = [x_0, y_0]^T$$

修改的欧拉公式:

$$\left\{ \begin{array}{l} Y_{n+1} = Y_n + h[K_1 + K_2] / 2 \quad (n = 0, 1, \dots, N) \\ K_1 = F(t_n, Y_n) \\ K_2 = F(t_n + 0.5h, Y_n + 0.5hK_1) \end{array} \right.$$

经典龙格-库塔公式:

$$\left\{ \begin{array}{l} Y_{n+1} = Y_n + h[K_1 + 2K_2 + 2K_3 + K_4] / 6 \\ K_1 = F(t_n, Y_n) \\ K_2 = f(t_n + 0.5h, Y_n + 0.5hK_1) \\ K_3 = F(t_n + 0.5h, Y_n + 0.5hK_2) \\ K_4 = F(t_n + h, Y_n + hK_3) \end{array} \right.$$

## 捕食者与被捕食者问题

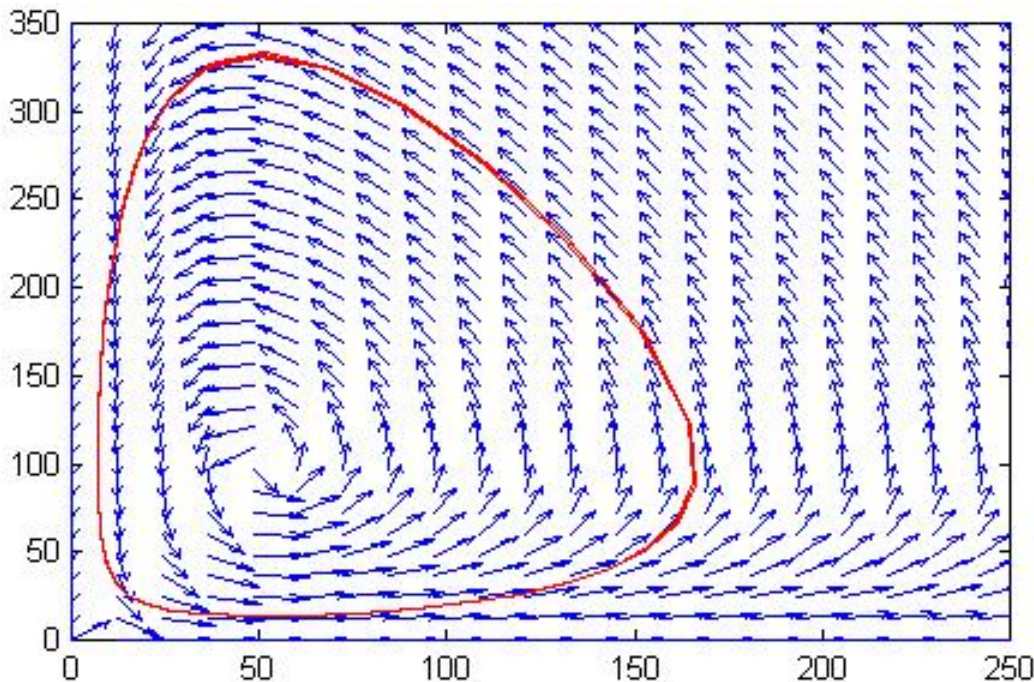
海岛上有狐狸和野兔, 当野兔数量增多时, 狐狸捕食野兔导致狐群数量增长; 大量兔子被捕食使狐群进入饥饿状态其数量下降; 狐群数量下降导致兔子被捕食机会减少, 兔群数量回升。微分方程模型如下

$$\begin{cases} \frac{dx}{dt} = x - 0.01xy & x(0) = 100 \\ \frac{dy}{dt} = -y + 0.02xy & y(0) = 20 \end{cases}$$

计算  $x(t)$ ,  $y(t)$  当  $t \in [0, 20]$  时的数据。绘图并分析捕食者和被捕食者的数量变化规律。

$$\begin{cases} \dot{x} = x - 0.01xy \\ \dot{y} = -y + 0.02xy \end{cases} \quad \begin{bmatrix} f_1(t, x, y) \\ f_2(t, x, y) \end{bmatrix} = \begin{bmatrix} x - 0.01xy \\ -y + 0.02xy \end{bmatrix}$$

平面向量场:  $\begin{bmatrix} p_x \\ p_y \end{bmatrix} = \begin{bmatrix} x - 0.01xy \\ -y + 0.02xy \end{bmatrix}$



——向量场中  
过点:(100, 20)  
的轨线

# 定义方程右端函数

```
function z=fox(t,y)

z(1,:)=y(1)-0.01*y(1).*y(2);
z(2,:)= -y(2)+0.02*y(1).*y(2);
```

**MATLAB命令求解:**

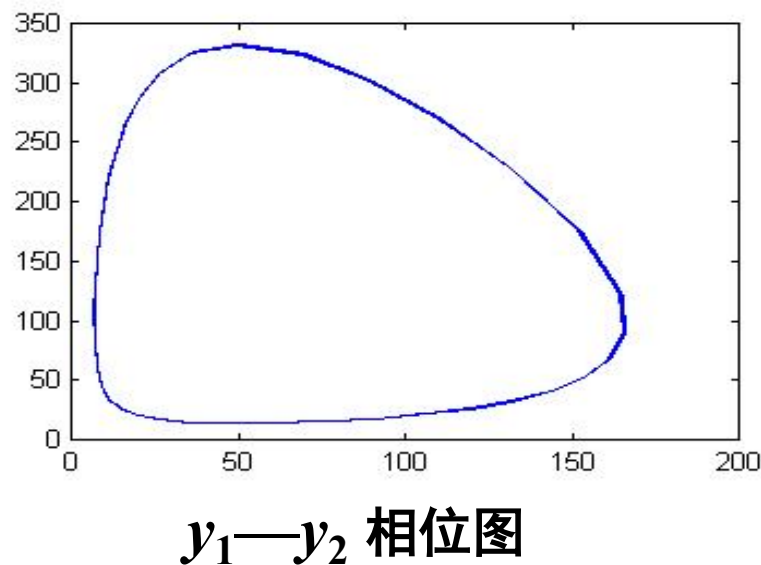
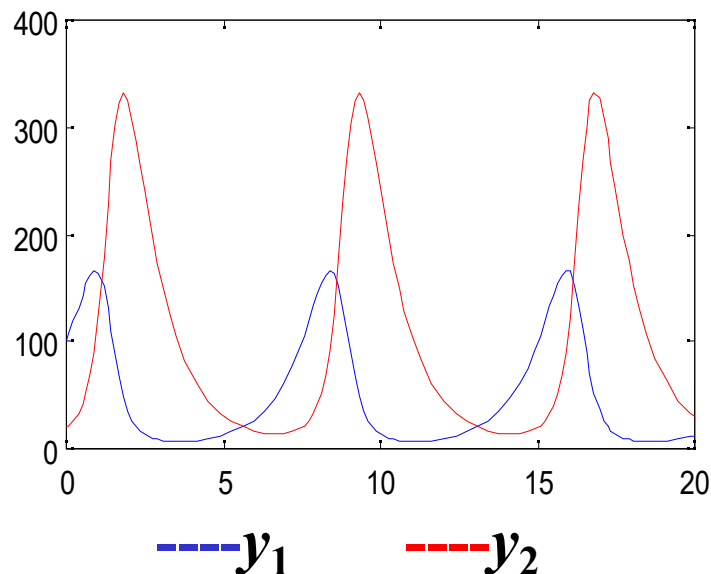
**Y0=[100,20];**

**[t,Y]=ode23('fox',[0,20],Y0);**

**x=Y(:,1);y=Y(:,2);**

**figure(1),plot(t,x,'b',t,y,'r')**

**figure(2),plot(x,y)**



“**蝴蝶效应**” 来源于洛伦兹一次讲演。模型如下

$$\begin{cases} \frac{dx}{dt} = -\beta x + yz \\ \frac{dy}{dt} = -\sigma(y - z) \\ \frac{dz}{dt} = -xy + \rho y - z \end{cases} \quad \begin{aligned} &\text{取 } \beta=8/3, \sigma=10, \rho=28。 \\ &x(0)=0, y(0)=0, z(0)=0.01。 \\ &t \in [0, 80], \end{aligned}$$

求微分方程数值解，绘出解函数曲线

微分方程右端函数：

$$F(t, x, y, z) = \begin{bmatrix} -8x/3 + xy \\ -10y + 10z \\ -xy + 28y - z \end{bmatrix}$$

记向量  $[y_1, y_2, y_3] = [x, y, z]$ , 创建函数文件

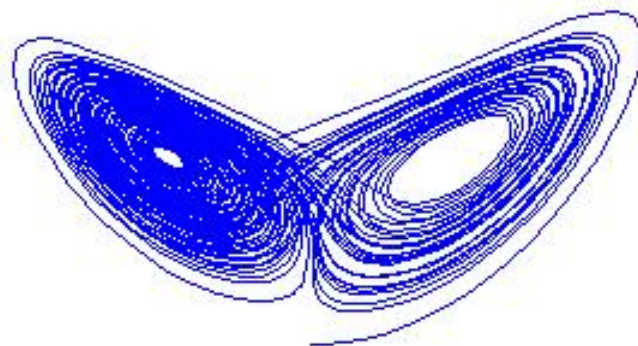
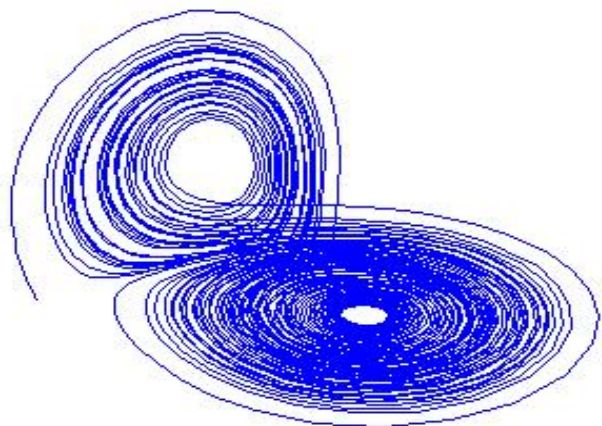
```
function z=flo(t,y)

z(1,:)=-8*y(1)/3+y(2).*y(3);
z(2,:)=-10*(y(2)-y(3));
z(3,:)=-y(1).*y(2)+28*y(2)-y(3);
```

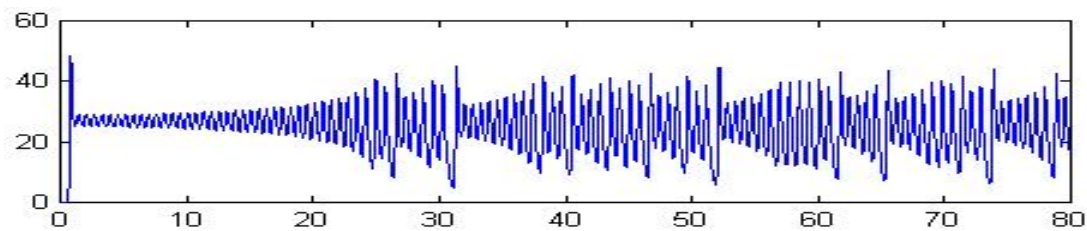
用MATLAB命令求解并绘出Y-X平面的投影图

```
P0=[0;0;0.01];
[T,P]=ode23('flo',[0, 80],P0);
figure(1),plot(P(:,2),P(:,1))
figure(2),comet3(P(:,1),P(:,2),P(:,3))
```

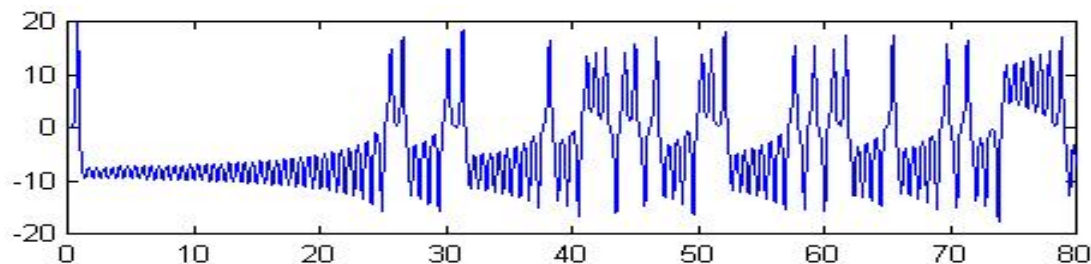




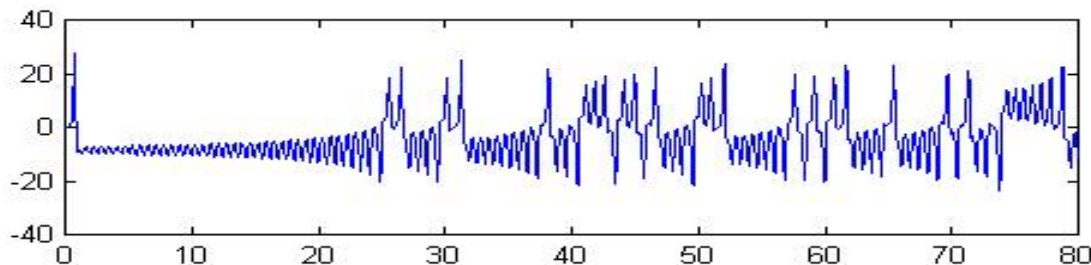
分量  $x$  的误差



分量  $y$  的误差



分量  $z$  的误差



# 总结

单步法的收敛性，稳定性，相容性

多步法

一阶方程组