§ 7.2 改进的欧拉法

§ 7.1.1 梯形公式

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

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

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

$$= -\frac{h^3}{12} y'''(\xi) \qquad (x_n < \xi < x_{n+1}) \quad R(T) = -\frac{h^3}{12} f''(\eta)$$

梯形公式为二阶方法，属隐式格式，一般用迭代法求解。

$$\begin{cases} y_{n+1}^{(0)} = y_n + hf(x_n, y_n) \\ y_{n+1}^{(k+1)} = y_n + \dfrac{h}{2}[f(x_n, y_n) + f(x_{n+1}, y_{n+1}^{(k)})] \quad (k = 0,1,\ldots) \end{cases}$$

根据Lipschitz条件：

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

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

收敛条件：

$$0 < \frac{hL}{2} < 1$$

§ 7.2.2 改进Euler法

在梯形公式中，隐式公式的求解只迭代一次：

$$
\begin{cases}
\bar{y}_{n+1} = y_n + hf(x_n, y_n) & \text{预测} \\
y_{n+1} = y_n + \dfrac{h}{2}[f(x_n, y_n) + f(x_{n+1}, \bar{y}_{n+1})] & \text{校正}
\end{cases}
$$

为编程方便，改写为：

$$
\begin{cases}
y_p = y_n + hf(x_n, y_n) \\
y_q = y_n + hf(x_n + h, y_p) \\
y_{n+1} = (y_p + y_q)/2
\end{cases}
$$

算法 7.1

(1)输入$a, b, f(x, y), N, y_0$

(2)$h = \dfrac{b-a}{N}, n = 0, x = a, y = y_0$,输出$(x, y)$

(3)$\begin{cases} y_p = y_n + hf(x_n, y_n) \\ y_q = y_n + hf(x_n + h, y_p) \end{cases}$,

$(x_p + x_q)/2 \Rightarrow y, x + h \Rightarrow x$,输出$(x, y)$

(4)若$n < N-1, n+1 \Rightarrow n$,转(3);否则退出。

【例7.2】用改进Euler法求解(h=0.1, N=10)：

$$\begin{cases} y' = x - y \quad (0 \le x \le 1) \\ y(0) = 0 \end{cases}$$

迭代形式为：

$$\begin{cases} y_p = y_n + 0.1(x_n - y_n) \\ y_q = y_n + 0.1(x_n + 0.1 - y_p) \\ y_{n+1} = (y_p + y_q)/2 \end{cases}$$

表 7-2 数值计算结果

Euler

x_n	y_n	$y(x_n)$	$y(x_n)-y_n$	$y(x_n)-y_n$
0	0.000000	0.000000	0.000000	0.000000
0.1	0.005000	0.004837	−0.000163	0.004837
0.2	0.019025	0.018731	−0.000294	0.008731
0.3	0.041218	0.040818	−0.000400	0.011818
0.4	0.070802	0.070320	−0.000482	0.014220
0.5	0.107076	0.106531	−0.000545	0.016041
0.6	0.149404	0.148812	−0.000592	0.017371
0.7	0.197211	0.196585	−0.000626	0.018288
0.8	0.249976	0.249329	−0.000647	0.018862
0.9	0.307228	0.306570	−0.000658	0.019150
1.0	0.368541	0.367879	−0.000662	0.019201

§7.3 龙格-库塔(Runge-Kutta)法

§7.3.1龙格-库塔法基本思想

(1)Euler公式：

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

(2)改进的Euler公式：

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

利用函数f(x,y)在某些点上的线性组合来计算$y(x_{n+1})$的近似值y_{n+1}。

可利用近似Taylor展开式的更多项来增加精度。

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

§7.3.2 龙格-库塔法的构造

算法 7.1

$$\boxed{\begin{cases} y_{n+1} = y_n + h\sum_{i=1}^{p} c_i K_i \\ K_1 = f(x_n, y_n) \\ K_i = f(x_n + a_i h, y_n + h\sum_{j=1}^{i-1} b_{ij} K_j) \end{cases}}$$

参数确定原则：其Taylor展开式与y(x)在x_n处尽可能多项重合。

在p=2时：

$$\begin{cases} y_{n+1} = y_n + h(c_1 K_1 + c_2 K_2) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + a_2 h, y_n + h b_{21} K_1) \end{cases}$$

迭代公式的Taylor展开式：

$$
\begin{aligned}
y_{n+1} &= y_n + h[c_1 f(x_n, y_n) + c_2 f(x_n + a_2 h, y_n + hb_{21} f(x_n, y_n))] \\
&= y_n + h\{c_1 f(x_n, y_n) + c_2[f(x_n, y_n) + a_2 hf'_x(x_n, y_n) \\
&\quad + hb_{21} f'_y(x_n, y_n) f(x_n, y_n)]\} + O(h^3) \\
&= y_n + (c_1 + c_2) f(x_n, y_n) h + c_2[a_2 f'_x(x_n, y_n) \\
&\quad + b_{21} f'_y(x_n, y_n) f(x_n, y_n)] h^2 + O(h^3)
\end{aligned}
$$

$y(x_{n+1})$在x_n处的Taylor展开式：

$$
\begin{aligned}
y(x_{n+1}) &= y(x_n) + hy'(x_n) + h^2/2y''(x_n) + O(h^3) \\
&= y_n + f(x_n, y_n) h \\
&\quad + h^2/2[f'_x(x_n, y_n) + f'_y(x_n, y_n) f(x_n, y_n)] O(h^3)
\end{aligned}
$$

要求局部截断误差为O(h^3)，则前两式的**前三项**相同，得：

$$\begin{cases} c_1 + c_2 = 1 \\ c_2 a_2 = 1/2 \\ c_2 b_{21} = 1/2 \end{cases}$$

上式有无穷多解，如取$c_1=c_2=1/2$, $a_2=b_{21}=1$，则：

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

取$c_1=0, c_2=1, a_2=b_{21}=1/2$，则：

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

常用的三阶方法：

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

常用的经典四阶方法：

$$\begin{cases} y_{n+1} = y_n + h/6 \cdot (K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + h/2, y_n + hK_1/2) \\ K_3 = f(x_n + h/2, y_n + hK_2/2) \\ K_4 = f(x_n + h, y_n + hK_3) \end{cases}$$

【例7.3】用四阶RK方法求解(h=0.1, N=5)：

$$\begin{cases} y' = x - y \quad (0 \le x \le 1) \\ y(0) = 0 \end{cases}$$

$$\begin{cases} y_{n+1} = y_n + \dfrac{0.2}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = x_n - y_n \\ K_2 = x_n + \dfrac{h}{2} - \left(y_n + \dfrac{h}{2}K_1\right) = 0.9(x_n - y_n) + 0.1 \\ K_3 = x_n + \dfrac{h}{2} - \left(y_n + \dfrac{h}{2}K_2\right) = 0.91(x_n - y_n) + 0.09 \\ K_4 = x_n + h - (y_n + hK_3) = 0.818(x_n - y_n) + 0.182 \end{cases}$$

四阶RK方法

x_n	y_n	$y(x_n)$	$y(x_n)-y_n$	Euler	改进Euler
0	0.000000	0.000000	0.000000	0.000000	0.000000
0.2	0.018733	0.018731	−0.000002	0.008731	-0.000294
0.4	0.070324	0.070320	−0.000004	0.014220	-0.000482
0.6	0.148817	0.148812	−0.000005	0.017371	-0.000592
0.8	0.249335	0.249329	−0.000006	0.018862	-0.000647
1.0	0.367886	0.367879	−0.000007	0.019201	-0.000662

§7.3.3 变步长的龙格-库塔法

y(x)变化可能不均匀，等步长求解可能有些地方精度过高，有些地方精度过低。

如何根据精度自动调节步长？

以p阶公式、步长h计算： $y(x_{n+1}) - y_{n+1}^{(h)} \approx ch^{p+1}$

以步长h/2计算两次： $y(x_{n+1}) - y_{n+1}^{(h/2)} \approx 2c\left(\frac{h}{2}\right)^{p+1}$

$$(2^p - 1)y(x_{n+1}) - 2^p y_{n+1}^{(h/2)} + y_{n+1}^{(h)} \approx 0$$

有： $\boxed{\left|y(x_{n+1}) - y_{n+1}^{(h)}\right| \approx \frac{2^p}{(2^p - 1)}\left|y_{n+1}^{(h)} - y_{n+1}^{(h/2)}\right| \approx \Delta}$

缩小或放大**h**直到达到要求计算精度。

§7.4 线性多步法

前面的方法在计算y_{n+1}时仅利用了y_n，是否还可以利用y_{n-1} , y_{n-2}, …？

一般形式：

$$\boxed{y_{n+1}=\sum_{i=0}^{r}a_i y_{n-i}+\sum_{i=-1}^{r}\beta_i f_{n-i}}$$

若$\beta_{-1}=0$，显式；若 $\beta_{-1}\neq 0$，隐式。

§7.4.1 线性多步公式的导出

- 将线性多步公式和$y(x_{n+1})$分别在x_n处进行Taylor展开；
- 根据阶数要求指定项的系数相等，解方程组得出多步公式的各项系数。
- 由于线性多步公式中有2r+3个待定系数，最多可以达到2r+2阶精度。

$$\sum_{i=0}^{r} a_i = 1$$

$$\sum_{i=0}^{r} (-i)^k a_i + k\sum_{i=-1}^{r} (-i)^{k-1} \beta_i = 1$$

§7.4.2 常用的线性多步公式

1、阿达姆斯(Admas)公式：

r=3, $a_1=a_2=a_3=\beta_{-1}=0$，由方程组：

$$\begin{cases} \sum_{i=0}^{3} a_i = 1 \\ \sum_{i=1}^{3} (-i)^k a_i + k \sum_{i=-1}^{3} (-i)^{k-1} \beta_i = 1 \end{cases}$$

得到四阶**显**性公式：

$$y_{n+1} = y_n + \frac{h}{24}(55 f_n - 59 f_{n-1} + 37 f_{n-2} - 9 f_{n-3})$$

四阶**隐**性公式：$y_{n+1} = y_n + \dfrac{h}{24}(9 f_{n+1} + 19 f_n - 5 f_{n-1} + f_{n-2})$

2、米恩尔(Milne)公式：

$r=3, a_0=a_1=a_2=\beta_{-1}=0$，得到四阶四步显式公式：

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

3、海明(Hamming)公式：

$r=2, a_1=\beta_2=0$，得到四阶三步隐式公式：

$$y_{n+1}=\frac{1}{8}(9y_n-y_{n-2})+\frac{3}{8}h(f_{n+1}+2f_n-f_{n-1})$$

【例 7.4】 分别用四阶 Adams 显式和隐式公式求初值问题 $\begin{cases} y' = x - y \\ y(0) = 0 \end{cases}$ $0 \leqslant x \leqslant 1$ 的数值解，取 $h = 0.1$。

表 7-4 计算结果

x_n	Admas 显式法		Admas 隐式法	
	y_n	$\lvert y(x_n) - y_n \rvert$	y_n	$\lvert y(x_n) - y_n \rvert$
0.0	0		0	
0.1	0.00483742		0.00483742	
0.2	0.01873075		0.01873075	
0.3	0.04081822		0.04081801	2.1×10^{-7}
0.4	0.07032292	2.87×10^{-6}	0.07031966	3.8×10^{-7}
0.5	0.10653548	4.82×10^{-6}	0.10653014	5.2×10^{-7}
0.6	0.14881841	6.77×10^{-6}	0.14881101	6.3×10^{-7}
0.7	0.19659339	8.09×10^{-6}	0.19658459	7.1×10^{-7}
0.8	0.24933816	9.19×10^{-6}	0.24932819	7.7×10^{-7}
0.9	0.30657961	9.95×10^{-6}	0.30656885	8.1×10^{-7}
1.0	0.36788996	1.052×10^{-5}	0.36787860	8.4×10^{-7}

隐式公式精度高，稳定性好，但计算量大。

§7.4.3 预测-校正系统

用显式公式计算预测值，用隐式公式进行校正。

Admas预测-校正公式：

$$\begin{cases} \bar{y}_{n+1} = y_n + \dfrac{h}{24}(55 f_n - 59 f_{n-1} + 37 f_{n-2} - 9 f_{n-3}) & \text{预测} \\ y_{n+1} = y_n + \dfrac{h}{24}[9 f(x_{n+1}, \bar{y}_{n+1}) + 19 f_n - 5 f_{n-1} + f_{n-2}] & \text{校正} \end{cases}$$

Milne-Hamming预测-校正公式：

$$\begin{cases} \bar{y}_{n+1} = y_{n-3} + \dfrac{4}{3}h(2 f_n - f_{n-1} + f_{n-2}) & \text{预测} \\ y_{n+1} = \dfrac{1}{8}(9 y_n - y_{n-2}) + \dfrac{3}{8}h[9 f(x_{n+1}, \bar{y}_{n+1}) + 2 f_n - f_{n-1}] & \text{校正} \end{cases}$$

本章小结

- 欧拉(Euler)法

$$\begin{cases} y_{n+1} = y_n + hf(x_n, y_n) \quad (n = 0,1,\ldots) \\ y_0 = y(a) \end{cases}$$

- 改进的欧拉法

$$\begin{cases} \bar{y}_{n+1} = y_n + hf(x_n, y_n) & \text{预测} \\ y_{n+1} = y_n + \dfrac{h}{2}[f(x_n, y_n) + f(x_{n+1}, \bar{y}_{n+1})] & \text{校正} \end{cases}$$

龙格-库塔(Runge-Kutta)法

根据**Taylor**展开确定系数：

$$\boxed{\begin{cases} y_{n+1} = y_n + \sum_{i=1}^{p} c_i K_i \\ K_1 = f(x_n, y_n) \\ K_i = f(x_n + a_i h, y_n + h\sum_{j=1}^{i-1} b_{ij} K_j) \end{cases}}$$

根据精度自动调节步长：

$$\left|y(x_{n+1}) - y_{n+1}^{(h)}\right| \approx \frac{2^p}{(2^p - 1)}\left|y_{n+1}^{(h)} - y_{n+1}^{(h/2)}\right| \approx \Delta$$

- 线性多步法

$$y_{n+1}=\sum_{i=0}^{r}a_i y_{n-i}+\sum_{i=-1}^{r}\beta_i f_{n-i}$$

根据**Taylor**展开确定系数；

结合显式、隐式形成预测-校正系统。

课 后 作 业

第七章习题的1、2、5、6。