《数值分析》21

主要内容：

- 一阶常微分方程欧拉法
- 欧拉法和修正的欧拉法(单步法)
- 局部截断误差和p阶精度
- Range-Kutta公式
- 线性单步法的收敛性和稳定性
- 线性多步法简介

一阶常微分方程欧拉法

例1. 一阶常微分方程初值问题

$$\begin{cases} \dfrac{dy}{dx} = 2(x-y) \\ y(0) = 0.8 \end{cases} \qquad f(x,y) = 2(x-y)$$

例2. Logistic模型 $\dfrac{dy}{dx} = y(1-y)$

初值条件$y(0)$=0.2

如何求y(x)？

解析解？或者数值近似解？

一阶常微分方程欧拉法

一阶常微分方程初值问题:
$$\begin{cases} \dfrac{dy}{dx} = f(x,y),\ x > x_0 \\ y(x_0) = y_0 \end{cases}$$

其中, $y = y(x)$ 是未知函数,

右端函数 $f(x, y)$ 是已知函数, 初值 y_0 是已知数据。

数值方法——取定离散点: $x_0 < x_1 < x_2 < \cdots < x_N$

要求：未知函数 $y(x)$在这些离散点上的近似值

$$y_1,\ y_2,\ y_3,\ \cdots\cdots,\ y_N$$

$$\frac{dy}{dx} = f(x,y) \quad \Rightarrow \quad \frac{y_{n+1} - y_n}{h} = f(x_n, y_n)$$

近似解求法（Euler法）：

❖ **问题：** $\begin{cases} \dfrac{dy}{dx} = f(x,y), \ x > x_0 \\ y(x_0) = y_0 \end{cases}$

1) **区域离散：** 取定步长 h, 记

$$x_n = x_0 + nh, \quad (n = 1,2,\cdots,N)$$

2) **格式：** Euler公式（法），

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

问：$f(x_n, y_n)$ 其实可以看作什么？

近似解求法（修正-Euler法）：

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

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

2.1) 左矩形公式： $\int_{x_n}^{x_{n+1}} f(x, y(x))dx \approx hf(x_n, y_n)$

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

2.2) 梯形公式:

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

3) 由梯形公式推出的预-校方法:

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

$$\Rightarrow \begin{cases} \tilde{y}_{n+1} = y_n + hf(x_n, y_n) \\ y_{n+1} = y_n + \frac{h}{2}[f(x_n, y_n) + f(x_{n+1}, \tilde{y}_{n+1})] \end{cases}$$

预–校方法又称为**修正的Euler法**,算法如下

$$k_1 = f(x_n, y_n),$$

$$k_2 = f(x_{n+1}, y_n + h k_1),$$

$$y_{n+1} = y_n + \frac{h}{2}[k_1 + k_2]$$

局部截断误差（欧拉公式）：

设 $y_n=y(x_n)$, 称 $R_{n+1}=y(x_{n+1})-y_{n+1}$ 为局部截断误差.

由泰勒公式

$$y(x_{n+1})=y(x_n)+(x_{n+1}-x_n)y'(x_n)+\frac{(x_{n+1}-x_n)^2}{2}y''(\xi)$$

即 $$y(x_{n+1})=y(x_n)+hf(x_n,y_n)+\frac{h^2}{2}y''(\xi)$$

Euler公式: $y_{n+1}=y_n+hf(x_n,y_n)$ 的局部截断误差

$$y(x_{n+1})-y_{n+1}=y(x_n)-y_n+O(h^2)=O(h^2)$$

Euler公式的局部截断误差记为：$O(h^2)$

称Euler公式具有1阶精度。

若局部截断误差为 $O(h^{p+1})$，则称显式单步法（如欧拉法）具有 p 阶精度。(计算格式的精度概念)

例 3. 证明修正的Euler法具有2阶精度

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

将预测公式 $\tilde{y}_{n+1} = y_n + hf(x_n, y_n)$ 代入

得 $y_{n+1} = y_n + 0.5h[f(x_n,y_n)+f(x_{n+1}, y_n+hf(x_n, y_n))]$

$$f(x_{n+1}, y_n+hf(x_n, y_n)) = f(x_n+h, y_n+hf(x_n, y_n))$$

$$= f(x_n, y_n)+h[f_x']_n + hf(x_n,y_n)[f_y']_n + O(h^2)$$

$$y'(x_n) = f(x_n, y_n) \rightarrow y''(x_n) = \frac{d}{dx} f(x_n, y_n)$$

$$0.5h[f(x_n,y_n)+f(x_{n+1}, y_n+hf(x_n, y_n))]$$

$$=hy'(x_n)+0.5h^2y''(x_n)+O(h^3)$$

$$y_{n+1}=y_n+hy'(x_n)+0.5h^2y''(x_n)+O(h^3)$$

利用泰勒展开：

$$y(x_{n+1})=y(x_n)+hy'(x_n)+0.5h^2y''(x_n)+O(h^3)$$

局部截断误差：$y(x_{n+1})-y_{n+1}=y(x_n)-y_n+O(h^3)$

$$=O(h^3)$$

故修正的Euler法具有2阶精度。

Range-Kutta公式

由欧拉方法的截断误差可知：

1)用**一阶泰勒多项式**近似函数可以得其**局部截断误差**为一阶泰勒余项$O(h^2)$

2)类似地，若用p**阶泰勒多项式**近似函数，则**局部截断误差**应该为p阶泰勒余项$O(h^{p+1})$

3)因此，理论上只要$y(x)$**充分光滑**，利用函数的泰勒展开可以构造任意精度的数值方法

但是：

具体实施时是很困难的！因为：若直接对$y(x)$用高次泰勒多项式近似，**会出现$f(x,y)$的各阶偏导数**，使得计算变得十分复杂并且工作量巨大。

因此一般**不直接使用**泰勒展开方法, 但**可以间接使用**。

所以：此处介绍的**Runge-Kutta方法**就属于**间接使用泰勒展开**的方法

建立高精度的单步递推格式。

单步递推法的基本思想是从 (x_i, y_i) 点出发，以某一斜率沿直线达到 (x_{i+1}, y_{i+1}) 点。

注意：欧拉法及其各种变形所能达到的最高精度为2阶。

考察改进的欧拉法，可以将其改写为：

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

斜率
一定取K_1 K_2
的平均值吗？

步长一定是一个h 吗？

Range-Kutta公式

将改进欧拉法推广为：

$$\begin{cases} y_{i+1} = y_i + h[\lambda_1 K_1 + \lambda_2 K_2] \\ K_1 = f(x_i, y_i) \\ K_2 = f(x_i + p_1 h, y_i + p_2 \end{cases}$$

$$y''(x) = \frac{d}{dx} f(x,y) = f_x(x,y) + f_y(x,y)\frac{dy}{dx} = f_x(x,y) + f_y(x,y) f(x,y)$$

首先希望能确定系数 λ_1、λ_2、

有2阶精度，即在 $y_i = y(x_i)$

$$R_i = y(x_{i+1}) - y_{i+1}$$

Step 1: 将 K_2 在 (x_i, y_i)点作 Taylor 展 …… 元函数泰勒展开)

$$K_2 = f(x_i + p_1 h, y_i + p_2 h K_1)$$

$$= f(x_i, y_i) + p_1 h f_x(x_i, y_i) + p_2 h K_1 f_y(x_i, y_i) + O(h^2)$$

Step 2: 将 K_2 代入第1式，得到

$$y_{i+1} = y_i + (\lambda_1 + \lambda_2) h y'(x_i) + \lambda_2 h^2 [p_1 f_x(x_i, y_i) + p_2 f(x_i, y_i) f_y(x_i, y_i)] + O(h^3)$$

Step 3: 将 y_{i+1} 与 $y(x_{i+1})$ 在 x_i 点的泰勒展开作比较

$$y_{i+1}=y_i+(\lambda_1+\lambda_2)hy'(x_i)+\lambda_2h^2[p_1f_x(x_i,y_i)+p_2f(x_i,y_i)f_y(x_i,y_i)]+O(h^3)$$

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

要求 $R_i=y(x_{i+1})-y_{i+1}=O(h^3)$，则必须有：

$$\lambda_1+\lambda_2=1;$$
$$\lambda_2p_1=\frac{1}{2};\quad \lambda_2p_2=\frac{1}{2};$$

这里有 4 个未知数， 3 个方程。

满足上述方程的一族公式称为二阶龙格 - 库塔格式 。

注意到，当 $p_1=p_2$, $\lambda_1=\lambda_2=\frac{1}{2};p_1=p_2=1$
就是改进的欧拉法。

问题: 为获得更高的精度，应该如何进一步推广?

二阶龙格 - 库塔格式推广得到更高精度格式

$$
\begin{cases}
y_{i+1} = y_i + h[\lambda_1 K_1 + \lambda_2 K_2 + \ldots + \lambda_m K_m] \\
K_1 = f(x_i, y_i) \\
K_2 = f(x_i + \alpha_2 h, y_i + \beta_{21} hK_1) \\
K_3 = f(x_i + \alpha_3 h, y_i + \beta_{31} hK_1 + \beta_{32} hK_2) \\
\ldots\ldots \\
K_m = f(x_i + \alpha_m h, y + \beta_{m1} hK_1 + \beta_{m2} hK_2 + \ldots + \beta_{m\,m-1} hK_{m-1})
\end{cases}
$$

其中λ_i ($i = 1, \ldots, m$)，α_i ($i = 2, \ldots, m$) 和 β_{ij} ($i = 2, \ldots, m; j = 1, \ldots, i-1$) 均为待定系数，确定这些系数的步骤与前面相似。

- 最常用为四阶经典龙格-库塔法 /* Classical Runge-Kutta Method */：

$$
\begin{cases}
y_{i+1} = y_i + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\
K_1 = f(x_i, y_i) \\
K_2 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_1) \\
K_3 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}K_2) \\
K_4 = f(x_i + h, y_i + hK_3)
\end{cases}
$$

注:

☞ 龙格-库塔法的运算在于计算 K_i 的值，即计算 f 的值。Butcher 于1965年给出了计算量与可达到的最高精度阶数的关系 （本书不作要求）：

每步须算K_i的个数	2	3	4	5	6	7	$n\geq 8$
可达到的最高精度	$O(h^2)$	$O(h^3)$	$O(h^4)$	$O(h^4)$	$O(h^5)$	$O(h^6)$	$O(h^{n-2})$

☞ 由于龙格-库塔法的导出基于泰勒展开，故精度主要受解函数的光滑性影响。对于光滑性不太好的解，最好采用低阶算法而将步长h 取小（低阶算法不需要高次可导）。

类似二阶推导，可得三阶、四阶RK公式：

三阶Range-Kutta公式一般形式

$$y_{n+1}=y_n+h[k_1+4k_2+k_3]/6$$
$$k_1=f(x_n,y_n),$$
$$k_2=f(x_n+0.5h,\ y_n+0.5hk_1)$$
$$k_3=f(x_n+h,\ y_n-hk_1+2hk_2)$$

四阶Range-Kutta公式一般形式

$$y_{n+1}=y_n+h[k_1+2k_2+2k_3+k_4]/6$$
$$k_1=f(x_n,y_n),$$
$$k_2=f(x_n+0.5h,\ y_n+0.5hk_1)$$
$$k_3=f(x_n+0.5h,\ y_n+0.5hk_2),$$
$$k_4=f(x_n+h,\ y_n+hk_3)$$

Range-Kutta公式

例4 $\begin{cases} \dfrac{dy}{dx} = y - xy^2, & 0 < x \le 2 \\ y(0) = 1 \end{cases}$

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

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
RK4	6.862e-005	3.747e-006	7.071e-007	2.186e-007
RK3	0.0012	1.529e-004	4.517e-005	1.906e-005
RK2	0.0123	0.0026	0.0011	5.9612e-004
Euler	0.1059	0.0521	0.0342	0.0256

一、收敛性 /*Convergence*/

Def 1 对于初值问题 $\begin{cases} \dfrac{dy}{dx} = f(x,y) \\ y(x_0) = y_0; x \ge x_0 \end{cases}$ (*) 的一种

显式单步法 $y_{n+1} = y_n + h\phi(x_n, y_n, h)$ 产生的近似解，

如果对于任一固定的 $x_n = x_0 + nh$，均有 $\lim\limits_{h\to 0} y_n = y(x_n)$ ，

则称该**单步法是收敛的**。

类似地可以定义隐式单步法的收敛性

定理8.2： 设初值问题（*）对应的下列单步法是p阶精度，

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

且函数ϕ 满足对y的**Lipschitz**条件，即存在常数 $L > 0$

$$|\phi(x, y_1, h) - \phi(x, y_2, h)| \le L|y_1 - y_2| \quad \forall y_1, y_2$$

并且初值y_0是准确的，即$y_0 = y(x)$, 则该单步法是**收敛的**，
且 $y(x_n) - y_n = O(h^p)$

证明： 记 $e_n = y(x_n) - y_n$ 由截断误差的定义

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

$$e_{n+1} = e_n + h[\phi(x_n, y(x_n), h) - \phi(x_n, y_n, h)] + T_{n+1}$$

因为单步法是p 阶的：$\exists h_0, 0 < h \le h_0$ 满足$|T_{n+1}| \le Ch^{p+1}$

$$|e_{n+1}| \le |e_n| + hL|e_n| + Ch^{p+1} = \alpha|e_n| + \beta$$

其中 $\alpha = 1 + hL, \beta = Ch^{p+1}$

$$\boxed{e_n = O(h^p)}$$

$$|e_n| \le \alpha|e_{n-1}| + \beta \le \alpha^2|e_{n-2}| + \alpha\beta + \beta$$

$$\le \alpha^3|e_{n-3}| + \beta(1 + \alpha + \alpha^2) \le \cdots \le$$

$$\boxed{|e_n| \le \alpha^n|e_0| + \beta(1 + \alpha + \alpha^2 + \ldots + \alpha^{n-1})}$$

$$|e_n| \le \exp[L(x_n - x_0)]|e_0| + Ch^p L^{-1}\{\exp[L(x_n - x_0)] - 1\}$$

$$= Ch^p L^{-1}\{\exp[L(x_n - x_0)] - 1\} \to 0 (h \to 0)$$

二、绝对稳定性 /*Absolute Stibility*/

计算过程中产生的舍入误差对计算结果的影响

首先以Euler公式为例，来讨论一下舍入误差的传播:

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

设实际计算得到的点 x_n 的近似函数值为 $\bar{y}_n=y_n+\rho_n$ ，

其中 y_n 为精确值，ρ_n 为误差

$$\bar{y}_{n+1}=\bar{y}_n+hf(x_n,\bar{y}_n) \qquad \rho_{n+1}=\bar{y}_{n+1}-y_{n+1}$$

$$\rho_{n+1}=\rho_n+h[f(x_n,\bar{y}_n)-f(x_n,y_n)]=[1+hf_y(x_n,\eta)]\rho_n$$

如果 $|1+hf_y|\le 1$，则误差是不增的，故可认为是稳定的

线性单步法的收敛性和稳定性

例如：对于初值问题 $\begin{cases} y' = y \\ y(x_0) = a \end{cases}$ 精确解为 $y = ae^{x-x_0}$

而实际求解的初值问题为 $\begin{cases} y' = y \\ y(x_0) = a + \Delta a \end{cases}$

精确解为 $y = (a + \Delta a)e^{x-x_0}$ 在 x_n 处的误差为 $\Delta a e^{x_n - x_0}$

可见误差随着 x_n 的增加呈指数函数增长

如果初值问题为 $\begin{cases} y' = -y \\ y(x_0) = a \end{cases}$ 精确解为 $y = ae^{x_0 - x}$

线性单步法的收敛性和稳定性

实际求解的初值问题为 $\begin{cases} y' = -y \\ y(x_0) = a + \Delta a \end{cases}$

精确解为 $y = (a+\Delta a)e^{x_0 - x}$ 在 x_n 处的误差为 $\Delta a e^{x_0 - x_n}$

可见误差随着 x_n 的增加呈指数函数递减

当 $f_y > 0$ 时，微分方程是不稳定的；
而 $f_y < 0$ 时，微分方程是稳定的。

上面讨论的稳定性，与数值方法和方程中 f 有关

实验方程：$y' = \lambda y \quad \lambda \in C, \mathrm{Re}(\lambda) < 0$

Def 3 对单步法 $y_{n+1} = y_n + h\phi(x_n, y_n, h)$ 应用实验方程，如果 $y_{n+1} = E(\lambda h) y_n$，当 $|E(\lambda h)| < 1$ 时，则称该单步法是绝对稳定的，在复平面上复变量 λh 满足 $|E(\lambda h)| < 1$ 的区域，称为该单步法的绝对稳定域，它与实轴的交集称为绝对稳定区间。

例4：分别求Euler法和经典的R-K法的绝对稳定区间。

解： ❶ Euler公式： $y_{n+1} = y_n + hf(x_n, y_n)$

将其应用于实验方程 $y_{n+1} = y_n + h(\lambda y_n) = (1 + h\lambda)y_n$

$$E(\lambda h) = 1 + \lambda h$$

绝对稳定域： $|1 + \lambda h| < 1$

当 $\lambda \in R$ 时， $|1 + \lambda h| < 1 \Rightarrow -2 < \lambda h < 0$

绝对稳定区间：$(-2, 0)$

❷经典的R-K公式：$y_{n+1} = y_n + \dfrac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$

$$k_1 = f(x_n, y_n) = \lambda y_n$$

$$k_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_1) = \lambda(y_n + \frac{h}{2}k_1) = (\lambda + \frac{\lambda^2 h}{2})y_n$$

$$k_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}k_2) = \lambda(y_n + \frac{h}{2}k_2) = (\lambda + \frac{\lambda^2 h}{2} + \frac{\lambda^3 h^2}{4})y_n$$

$$k_4 = f(x_n + h, y_n + hk_3) = \lambda(y_n + hk_3) = \lambda y_n + \lambda h k_3$$

$$y_{n+1} = (1 + \lambda h + \frac{\lambda^2 h^2}{2!} + \frac{\lambda^3 h^3}{3!} + \frac{\lambda^4 h^4}{4!})y_n$$

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

当$\lambda \in R$ 时，

$$|E(\lambda h)| = \left|1+\lambda h+\frac{(\lambda h)^2}{2!}+\frac{(\lambda h)^3}{3!}+\frac{(\lambda h)^4}{4!}\right| < 1$$

可以证明：$g(t)=1+t+\frac{t^2}{2!}+\frac{t^3}{3!}+\frac{t^4}{4!}$ 存在唯一极小值点

$$t^* = -1.596;\ g(t^*) = 0.27 > 0$$

由 $g(t)=1+t+\frac{t^2}{2!}+\frac{t^3}{3!}+\frac{t^4}{4!}<1$ 得

绝对稳定区间：$(-2.785, 0)$

线性多步法（理解思想即可）

单步法在计算y_{n+1}时，只用到前一步的信息y_n。为提高精度，需重新计算多个点处的函数值，如RK方法，计算量较大。如何通过较多地利用前面的已知信息，如y_n，$y_{n-1}, \cdots, y_{n\text{-}r}$，来构造高精度的算法计算$y_{n+1}$，这就是多步法的基本思想。

线性多步(k步)法格式

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

其中, $x_{n+i}=x_0+(n+i)h$, $f_{n+i}=f(x_{n+i},y_{n+i})$

局部截断误差

$$T_{n+k}=y(x_{n+k})-\sum_{i=0}^{k-1}\alpha_i y(x_{n+i})-h\sum_{i=0}^{k}\beta_i y'(x_{n+i})$$

Adamas显格式:

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

$$y' = f(x, y) \quad \Rightarrow \quad \int_{x_{n+1}}^{x_{n+2}} y'(x)dx = \int_{x_{n+1}}^{x_{n+2}} f(x, y(x))dx$$

$$y(x_{n+2}) - y(x_{n+1}) = \int_{x_{n+1}}^{x_{n+2}} f(x, y(x))dx$$

在区间$[x_n, x_{n+1}]$上插值，插值函数如下：

$$f(x) \approx [(x_{n+1} - x)f_n + (x - x_n)f_{n+1}] / h$$

$$\frac{1}{h}\int_{x_{n+1}}^{x_{n+2}} [(x_{n+1} - x)f_n + (x - x_n)f_{n+1}]dx$$

$$= \frac{1}{h}[-\frac{h^2}{2}f_n + \frac{3h^2}{2}f_{n+1}] = h[3f_{n+1} - f_n]/2$$

二阶Adamas显格式： $y_{n+2} = y_{n+1} + h(3f_{n+1} - f_n)/2$

有兴趣请看补充ppt: 线性多步法的推导过程（p1-p11）

补充两点：

❖ 一阶常微分方程组初值问题

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

一阶常微分方程组的向量表示

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

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

欧拉公式:

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

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

❖ 高阶常微分方程组初值问题

$$\begin{cases} y'' = f(x, y, y') \\ y(x_0) = y_0,\ \ y'(x_0) = m_0 \end{cases}$$

令 $$\begin{cases} y_1(x) = y(x) \\ y_2(x) = y'(x) \end{cases}$$

一阶常微分方程组:

$$\begin{cases} y_1' = y_2 \\ y_2' = f(x, y_1, y_2) \end{cases}$$

初值条件:

$$y_1(x_0) = y_0,$$
$$y_2(x_0) = m_0$$

$$Y(x) = \begin{bmatrix} y_1(x) \\ y_2(x) \end{bmatrix} \quad F(x, Y) = \begin{bmatrix} y_2(x) \\ f(x, y_1, y_2) \end{bmatrix} \quad Y_0 = \begin{bmatrix} y_0 \\ m_0 \end{bmatrix}$$

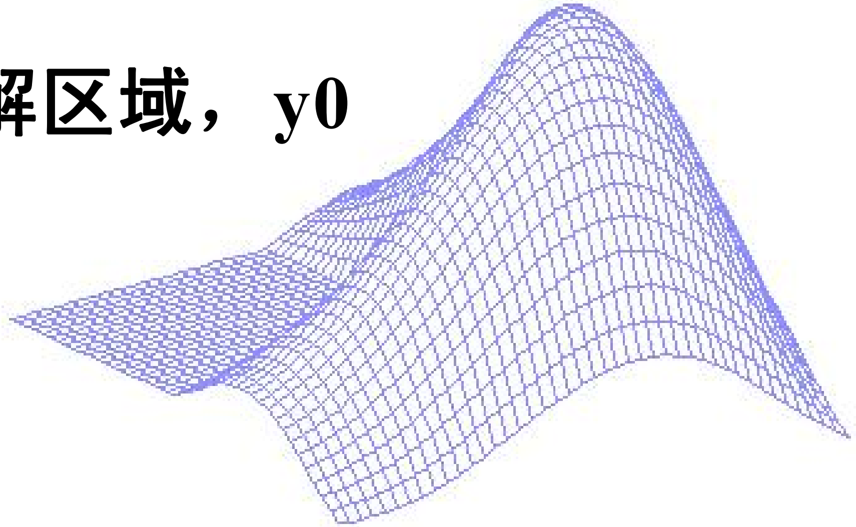
常微分方程组 $$\begin{cases} \dfrac{dY}{dx} = F(x, Y) \\ Y(x_0) = Y_0 \end{cases}$$

MATLAB求解常微分方程初值问题命令（从此页自看）：

(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] = 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

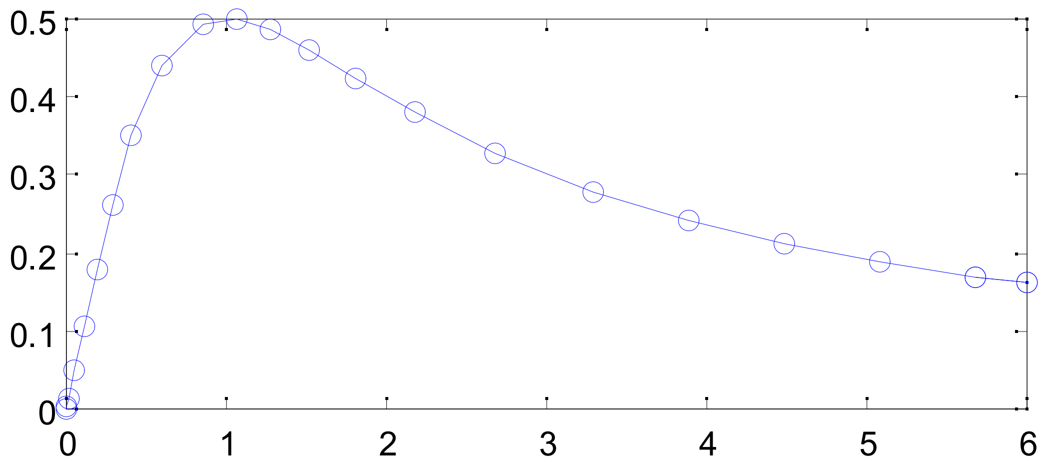
$$y' = \frac{1}{x^2+1} - 2y^2 \qquad y(0) = 0$$

MATLAB数值求解命令

F=inline('1./(1+x.^2)-2*y.^2');
ode23(F,[0,6],0)

[T,y]=ode23(f,[0,6],0)将得到自变量和函数的离散数据

输出结果为图形



MATLAB解常微分方程初值问题命令

$$\begin{cases} \dfrac{dy}{dx} = y - xy^2, \quad 0 < x \le 2 \\ y(0) = 1 \end{cases}$$

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:

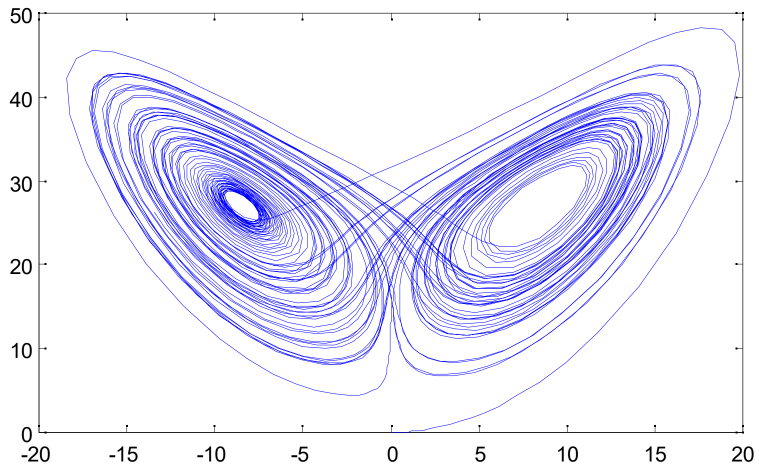
$$y(x) = \frac{1}{x - 1 + 2e^{-x}}$$

1. 创建罗伦茨模型右端函数的M文件

```
function z=flo(t,P)
A=[-8./3 0 P(2);0 -10 10;-P(2) 28 -1];
z=A*P;
```

2. 用MATLAB命令求解并绘出Y-X平面的投影图

```
P0=[0;0;0.01];
[x,y]=ode23('flo',[0, 80],P0);
plot(P(:,2),P(:,1))
```



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

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

修改的欧拉公式：

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

一阶常微分方程组初值问题

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

一阶常微分方程组的向量表示

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

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

欧拉公式:

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

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

修改的欧拉公式:

$$\begin{cases} Y_{n+1} = Y_n + h[K_1 + K_2]/2 \qquad (n = 0, 1, \cdots\cdots, N) \\ K_1 = F(t_n, Y_n) \\ K_2 = F(t_n + 0.5h, Y_n + 0.5hK_1) \end{cases}$$

经典龙格-库塔公式:

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

捕食者与被捕食者问题

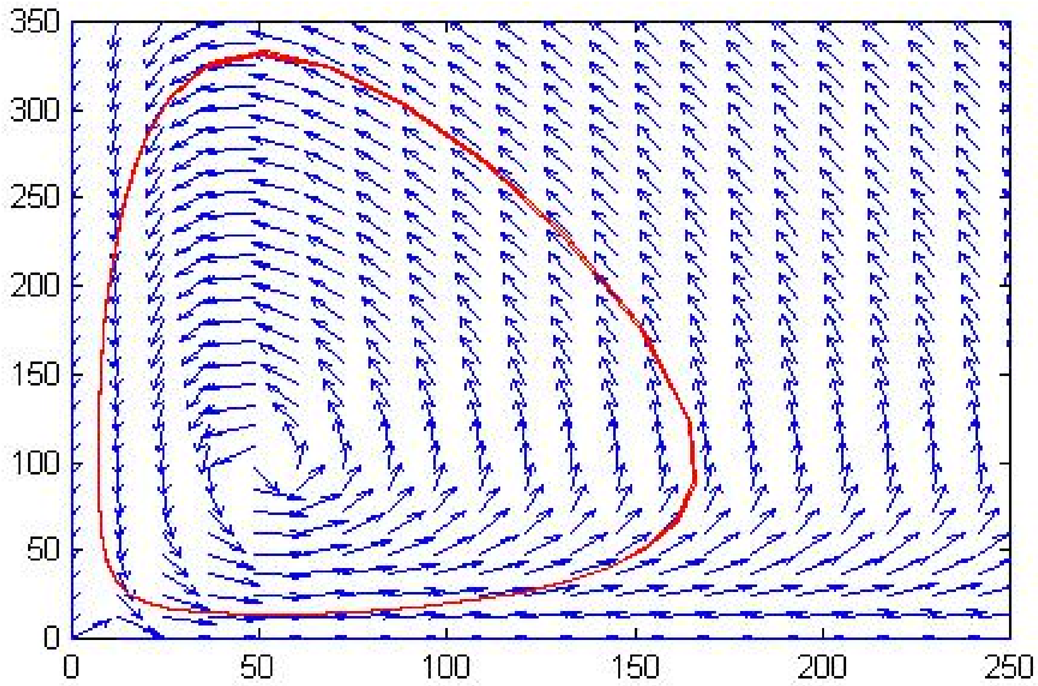
海岛上有狐狸和野兔，当野兔数量增多时，狐狸捕食野兔导致狐群数量增长；大量兔子被捕食使狐群进入饥饿状态其数量下降；狐群数量下降导致兔子被捕食机会减少，兔群数量回升。微分方程模型如下

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

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。

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

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



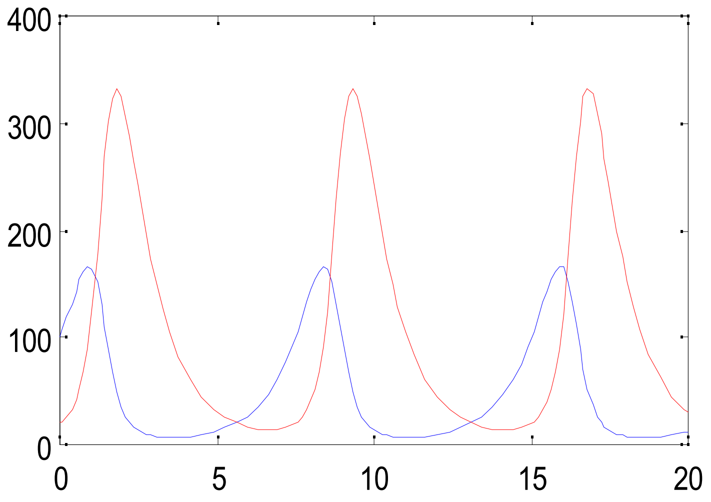
——向量场中过点:(100, 20)的轨线

定义方程右端函数

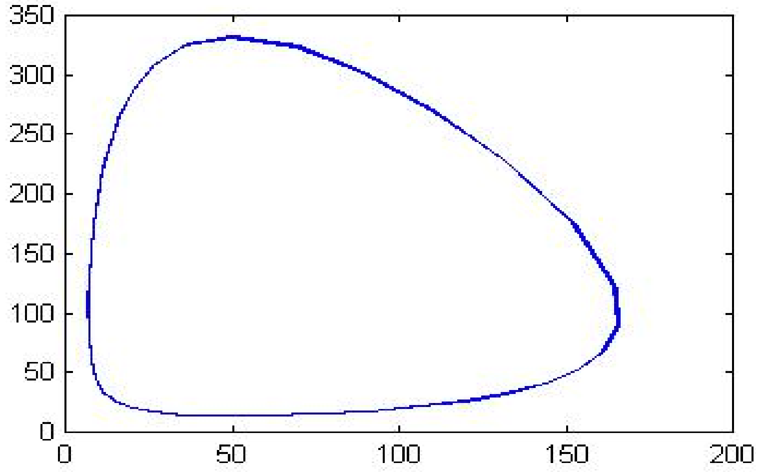
```
function z=fox(t,y)

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)
```



----y_1 ----y_2



y_1—y_2 相位图

“**蝴蝶效应**”来源于洛伦兹一次讲演。模型如下

$$\begin{cases} \dfrac{dx}{dt} = -\beta\, x + yz \\ \dfrac{dy}{dt} = -\sigma(y - z) \\ \dfrac{dz}{dt} = -xy + \rho\, y - z \end{cases}$$

取 β=8/3， σ=10， ρ=28。

x(0)=0， y(0)=0， z(0)=0.01。

$t\in$[0， 80]，

求微分方程数值解，绘出解函数曲线

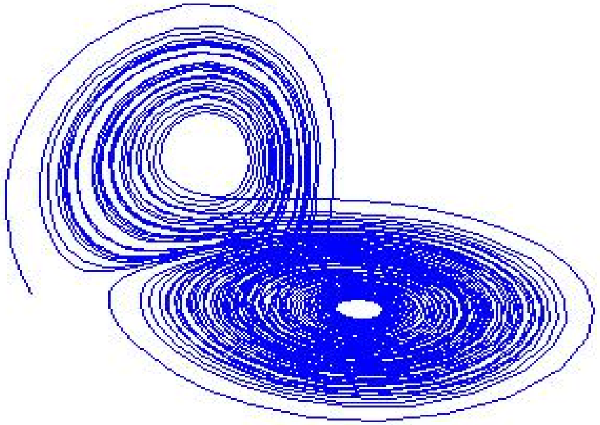
微分方程右端函数：
$$F(t,x,y,z) = \begin{bmatrix} -8x/3 + xy \\ -10y + 10z \\ -xy + 28y - z \end{bmatrix}$$

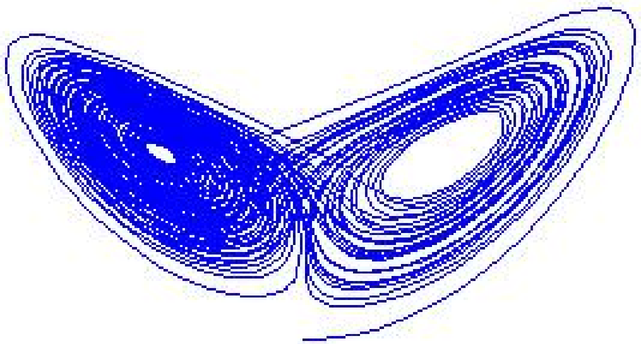
记向量 $[y_1, y_2, y_3] = [x, y, z]$，创建函数文件

```
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);
```

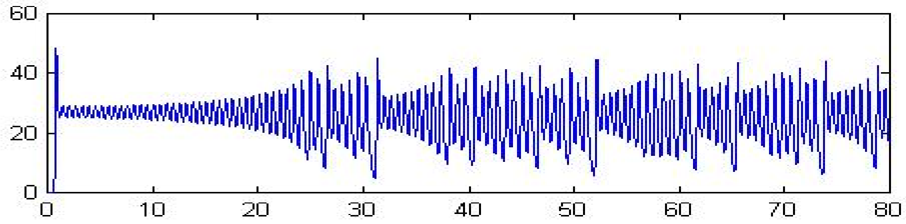
用MATLAB命令求解并绘出Y-X平面的投影图

```
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))
```

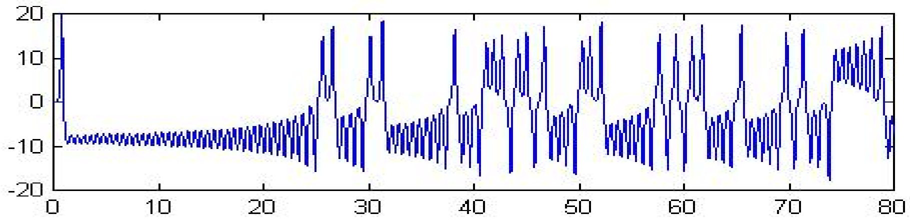




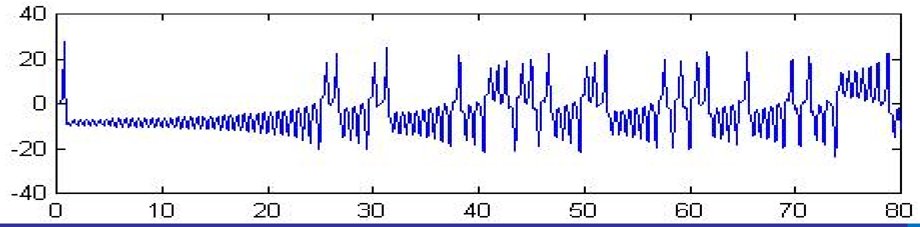
分量 x 的误差



分量 y 的误差



分量 z 的误差



二阶常微分方程问题

$$\ddot{x}+k^2x=0 \qquad \text{（简谐振动）}$$

$$\ddot{x}+p\dot{x}+k^2x=0 \qquad \text{（衰减振动）}$$

$$\ddot{x}+p\dot{x}+k^2x=f(t) \qquad \text{（受迫振动）}$$

n 阶贝塞尔方程

$$x^2y''+xy'+(x^2-n^2)y=0$$

n 阶勒让德方程

$$(1-x^2)y''-2xy'+n(n+1)y=0$$

$$\begin{cases} y'' = f(x, y, y') \\ y(x_0) = y_0, \quad y'(x_0) = m_0 \end{cases}$$

令 $\begin{cases} y_1(x) = y(x) \\ y_2(x) = y'(x) \end{cases}$

一阶常微分方程组: 初值条件:

$$\begin{cases} y_1' = y_2 \\ y_2' = f(x, y_1, y_2) \end{cases}$$

$$y_1(x_0) = y_0,$$
$$y_2(x_0) = m_0$$

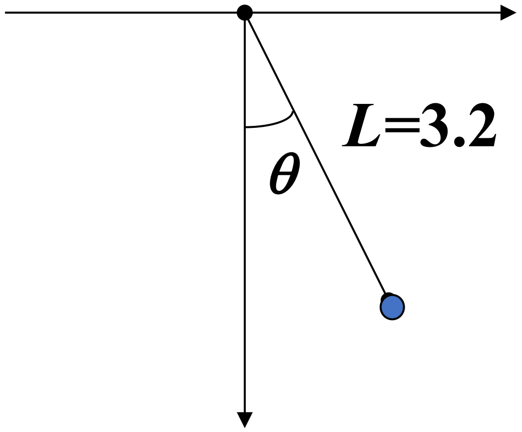
$$Y(x) = \begin{bmatrix} y_1(x) \\ y_2(x) \end{bmatrix} \qquad F(x, Y) = \begin{bmatrix} y_2(x) \\ f(x, y_1, y_2) \end{bmatrix} \qquad Y_0 = \begin{bmatrix} y_0 \\ m_0 \end{bmatrix}$$

常微分方程组
$$\begin{cases} \dfrac{dY}{dx} = F(x, Y) \\ Y(x_0) = Y_0 \end{cases}$$

例3. 单摆的数学模型

$$\theta'' = -a\sin\theta \quad 其中, a = g/L$$

初值条件: $\theta(0)=0.4$， $\theta'(0)=0$



第一步: 转化为一阶方程组

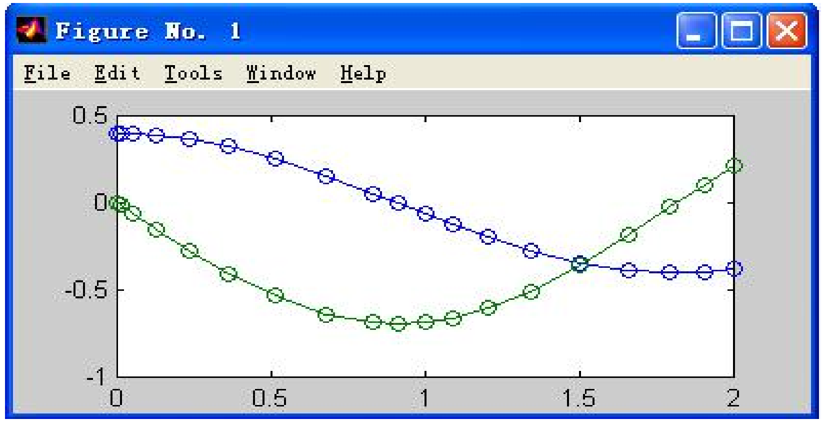
令: $y_1=\theta, y_2=\theta'$

初值条件: $y_1(0)=0.4, y_2(0)=0$

$$\begin{cases} y_1' = y_2 \\ y_2' = -a\sin y_1 \end{cases}$$

第二步: 求解方程组

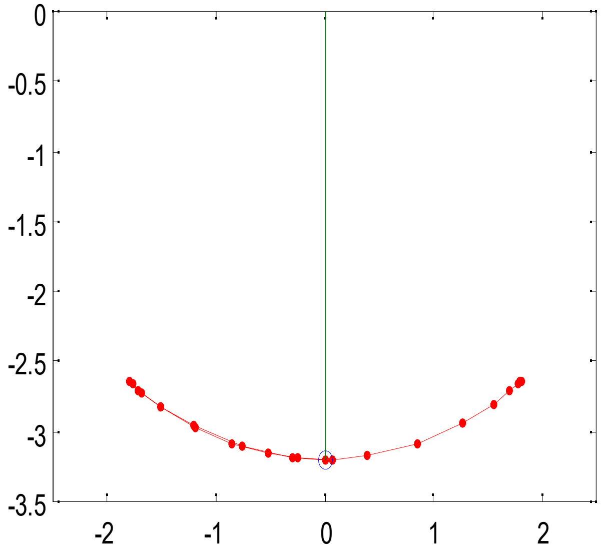
```
function f=dan(x,y)
f(1,:)=y(2);
f(2,:)=-9.8*sin(y(1))/3.2;
```




```
ode23('dan',[0,2],[0.4,0]);
```

单摆的动态模拟程序

```
[t,thata]=ode23('dan',[0,2.755],[0.6,0]);
R=3.2;n=length(t);
alpha=thata(:,1);
x=R*sin(alpha);
y=R*cos(alpha);
X=[0,0];Y=[0,-3.5];
for k=1:n
   xk=x(1:k);yk=y(1:k);
   Xk=x(k);Yk=y(k);
   plot(xk,-yk,'.-r',Xk,-Yk,'o',[0,Xk],[0,-Yk]),
   axis([-2.5,2.5,-3.5,0])
   pause(.5)
end
```



人造卫星的轨道模型

万有引力定律 $|F| = G\dfrac{Mm}{x^2+y^2}$

$$F = G\frac{Mm}{x^2+y^2}\left(-\frac{x}{\sqrt{x^2+y^2}}, -\frac{y}{\sqrt{x^2+y^2}}\right)$$

常微分方程
$$\begin{cases} \dfrac{d^2x}{dt^2} = -\dfrac{GMx}{(x^2+y^2)^{3/2}} \\ \dfrac{d^2y}{dt^2} = -\dfrac{GMy}{(x^2+y^2)^{3/2}} \end{cases}$$

地球引力参数：$GM = 3.986005 \times 10^5$（$km^3/s^2$）

例4 求解边值问题的数值方法算例

$$\begin{cases} y'' + y + x = 0, 0 < x < 1 \\ y(0) = 0, y(1) = 0 \end{cases}$$

解：取正整数n, 令h=1/(n+1), $x_j = jh$, (j =0, 1, ⋯, n+1).

将常微分方程离散化

$$\frac{y_{j+1} - 2y_j + y_{j-1}}{h^2} + y_j + x_j = 0$$

整理, 得:

$$-y_{j-1} + (2 - h^2)y_j - y_{j+1} = x_j h^2 \qquad (j = 1,2,\cdots,n)$$

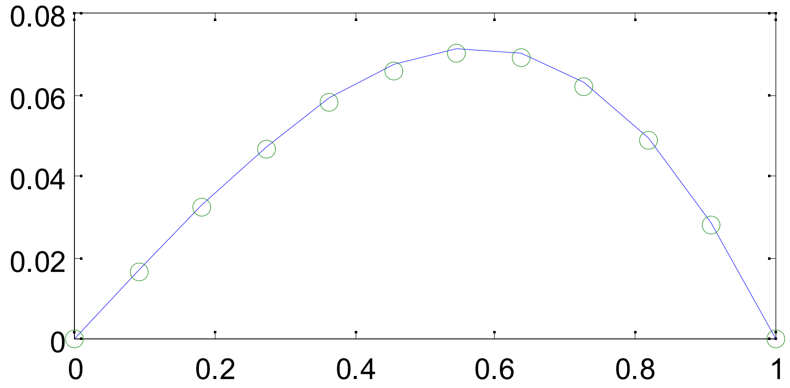
$$y_0 = 0, y_n = 0$$

1. 打靶法; 2. 高斯消元法

$$-y_{j-1}+(2-h^2)y_j-y_{j+1}=x_j h^2 \qquad (j=1,2,\cdots,n)$$

三对角方程组 $AY=F$

$$A=\begin{bmatrix} 2-h^2 & -1 & & & \\ -1 & 2-h^2 & -1 & & \\ & \ddots & \ddots & \ddots & \\ & & -1 & 2-h^2 & -1 \\ & & & -1 & 2-h^2 \end{bmatrix}_{n\times n}$$



— $y(x_n)$;

o y_n

学到了什么?

- 一阶常微分方程欧拉法
- 欧拉法和修正的欧拉法(单步法)
- 局部截断误差和p阶精度
- Range-Kutta公式
- 线性单步法的收敛性和稳定性
- 线性多步法简介