

# 常微分方程

浙江大学控制学院

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法

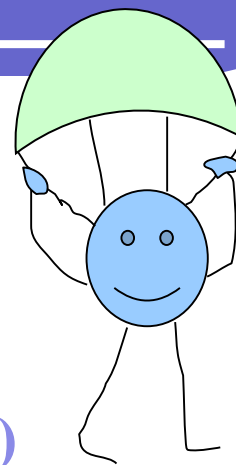
# 一阶线性常微分方程

- 降落伞问题

$$\frac{dv}{dt} = g - \frac{cv}{m} \quad \text{初值 } t=0, v=0 \quad \longrightarrow$$

$$v(t) = \frac{gm}{c} (1 - e^{-(c/m)t})$$

解析解



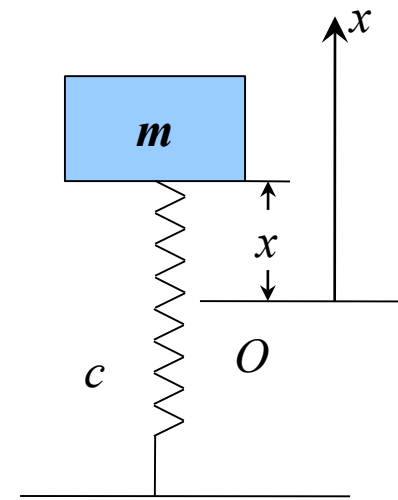
- 含有未知函数及其导数的方程——**微分方程(Differential Equations)**
- 被微分的量( $v$ )称为**因变量(Dependent Variable)**，与 $v$ 的微分有关的量 $t$ 称为**自变量(Independent Variable)**。如果函数只含有一个自变量，方程称为**常微分方程(ODE, Ordinary Differential Equation)**。如果函数含有两个或更多自变量，称为**偏微分方程(PDE, Partial Differential Equation)**。
- 定解条件：给出积分曲线在初始点的状态(**初始条件**)，称为**初值问题**；给出积分曲线首尾两端的状态(**边界条件**)，称为**边值问题**。

# 二阶微分方程

- 微分方程中的最高阶导数是一阶导数——一阶方程(First-order Equation)
- 微分方程中的最高阶导数是二阶导数——二阶方程(Second-order Equation)
- 例：弹簧-质量系统的振动问题

- 定义 
$$\frac{d^2x}{dt^2} + \frac{c}{m}x = 0$$
$$y = \frac{dx}{dt} \Rightarrow \frac{dy}{dt} = \frac{d^2x}{dt^2}$$
$$\left. \begin{array}{l} \frac{d^2x}{dt^2} + \frac{c}{m}x = 0 \\ \frac{dy}{dt} = \frac{d^2x}{dt^2} \end{array} \right\} \frac{dy}{dt} = -\frac{c}{m}x$$
- 定解条件：初值  $x(t_0) = x_0$   
 $x'(t_0) = x'_0$

高阶常微分方程可简化为一阶常微分方程进行求解。



# 常微分方程

- 一阶常微分方程的初值问题

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

- 若函数 $f(x, y)$ 在区域 $G = \{(x, y) | a \leq x \leq b, -\infty < y < \infty\}$ 中连续, 且关于 $y$ 满足李普希兹(Lipschitz)条件, 即存在常数 $L$ , 使得

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

对所有 $a \leq x \leq b$ 以及任何 $y_1, y_2$ 都成立, 则上述初值问题存在唯一的连续可微解 $y=y(x)$ 。

# 常微分方程的数值解法

## ● 数值解法

- 求问题的解 $y(x)$ 在一系列点 $a=x_0 < x_1 < x_2 < \dots < x_n = b$ 上值 $y(x_i)$ 的近似值 $y_i$  ( $i=0,1,2,\dots,n$ )。相邻的两个节点之间的距离 $h_i = x_{i+1} - x_i$ 称为由 $x_i$ 到 $x_{i+1}$ 的**步长**，通常取成常量 $h$ ，称为等步长，此时 $x_i = x_0 + ih$  ( $i=1,2,\dots,n$ )或 $x_{i+1} = x_i + h_i$  ( $i=0,1,2,\dots,n-1$ )

$$\left. \frac{dy}{dx} \right|_{x=x_i} = f(x, y) \Big|_{x=x_i} \xrightarrow{\text{数值微分}} \frac{y_{i+1} - y_i}{h} \approx f(x_i, y_i)$$
$$y_{i+1} = y_i + hf(x_i, y_i)$$

——**数值离散方法**：求未知函数 $y$ 在一系列离散点列上的值的近似

- (1) 变量离散：对区间作分割
- (2) 递推：由已知的 $y_0$ 逐步计算出解在一系列点上的值  
单步和多步

# 离散化方法

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

## ● 差商法

- $y'(x_i) \approx [y(x_{i+1}) - y(x_i)]/h$
- $[y(x_{i+1}) - y(x_i)]/h \approx f(x_i, y(x_i))$
- $y_{i+1} \approx y_i + h f(x_i, y_i)$

## ● 泰勒级数法

$$\begin{aligned} y(x_{i+1}) &= y(x_i + h) \\ &= y(x_i) + hy'(x_i) + \frac{h^2}{2!} y''(x_i) + \cdots + \frac{h^P}{P!} y^{(P)}(x_i) + \frac{h^{P+1}}{(P+1)!} y^{(P+1)}(\xi_i) \end{aligned}$$

## ● 数值积分法

$$y(x_{i+1}) = y(x_i) + \int_{x_i}^{x_{i+1}} f(x, y(x)) dx \quad \Longrightarrow \quad y(x_{i+1}) \approx y(x_i) + hf(x_i, y(x_i))$$

$$\int_{x_i}^{x_{i+1}} f(x, y(x)) dx \approx f(x_i, y(x_i))(x_{i+1} - x_i) = hf(x_i, y(x_i))$$

# 常微分方程的数值解法

- 为了考察数值方法提供的数值解是否有实用价值，需要知道如下几个结论：
  - 收敛性问题：步长充分小时，所得到的数值解能否逼近问题的真解
  - 稳定性问题：产生的舍入误差，在以后各步计算中，是否会无限制扩大
  - 误差估计

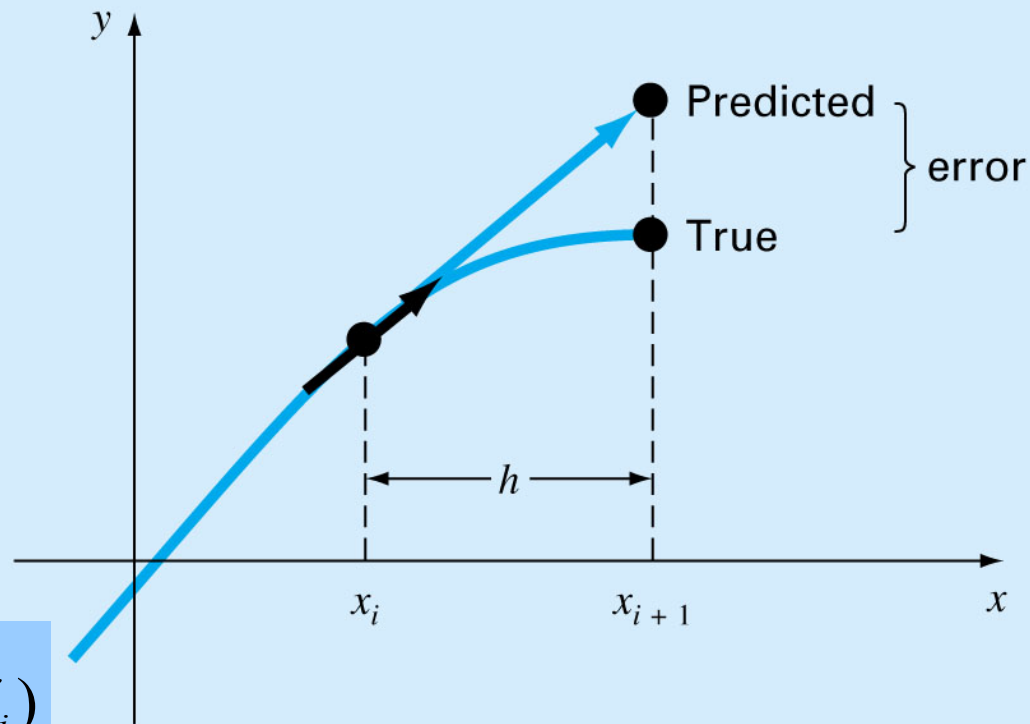


# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉 (Euler) 法
  - 龙格-库塔 (Runge-Kutta) 法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法

# 欧拉法

- 新值=旧值+斜率×步长
- 欧拉-柯西法
- 折线法



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

$$\frac{y(x_{i+1}) - y(x_i)}{h} = f(x_i, y(x_i)) + \frac{h}{2} y''(\xi_i)$$

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

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

$$E_a = \frac{h^2}{2} f'(x_i, y_i)$$

局部截断误差（第  $i$  步计算是精确的前提下）。这个误差在逐步计算过程中会传播、积累。因此还要估计这种积累。

# 局部截断误差的估计方法

- 对同一种求解公式采用两种不同的步长 $h_1$ 和 $h_2$ ， $y(x_{i+1})$ 的近似值分别是 $y_{i+1}$ 和 $y_{i+1}^*$

$$y(x_{i+1}) - y_{i+1} = \lambda h_1^{P+1} y^{P+1}(\xi_1)$$

$$y(x_{i+1}) - y_{i+1}^* = \lambda h_2^{P+1} y^{P+1}(\xi_2)$$

设 $h_2 > h_1$ ， $y^{P+1}(\xi_1) \approx y^{P+1}(\xi_2)$

$$y_{i+1}^* - y_{i+1} \approx \left(1 - \frac{h_2^{P+1}}{h_1^{P+1}}\right) \lambda h_1^{P+1} y^{P+1}(\xi_1) = \left(1 - \frac{h_2^{P+1}}{h_1^{P+1}}\right) [y(x_{i+1}) - y_{i+1}]$$

$$y(x_{i+1}) - y_{i+1} \approx (y_{i+1}^* - y_{i+1}) / [1 - (h_2/h_1)^{P+1}]$$

与预先给定的误差限 $\varepsilon$ 相比较，如果上式右端的值（估计值）较大，说明实际计算应缩小步长；反之如果估计值太小，应放大步长，以节省计算时间，并减少舍入误差的积累。由此可见，应用上式可以控制步长的大小。

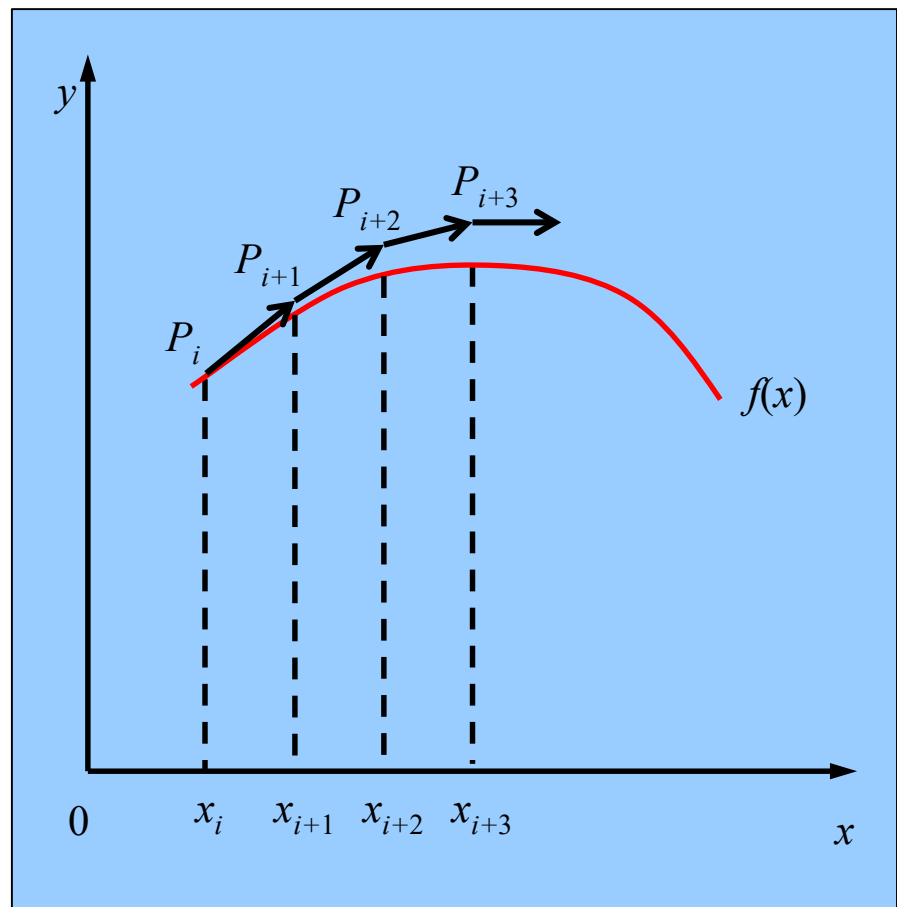
# 欧拉法的截断误差——一阶方法

## ● 局部截断误差

- 在假设  $y_i = y(x_i)$ ，即第  $i$  步计算是精确的前提下，考虑的截断误差  $R_i = y(x_{i+1}) - y_{i+1}$  称为局部截断误差 (local truncation error)。
- 若某算法的局部截断误差为  $O(h^{p+1})$ ，则称该算法有  $p$  阶精度。
- 欧拉格式的精度为一阶，局部截断误差为  $O(h^2)$ 。

## ● 全局截断误差

- 传播截断误差与局部截断误差之和
- 收敛性
- 欧拉格式的全局截断误差为  $O(h)$

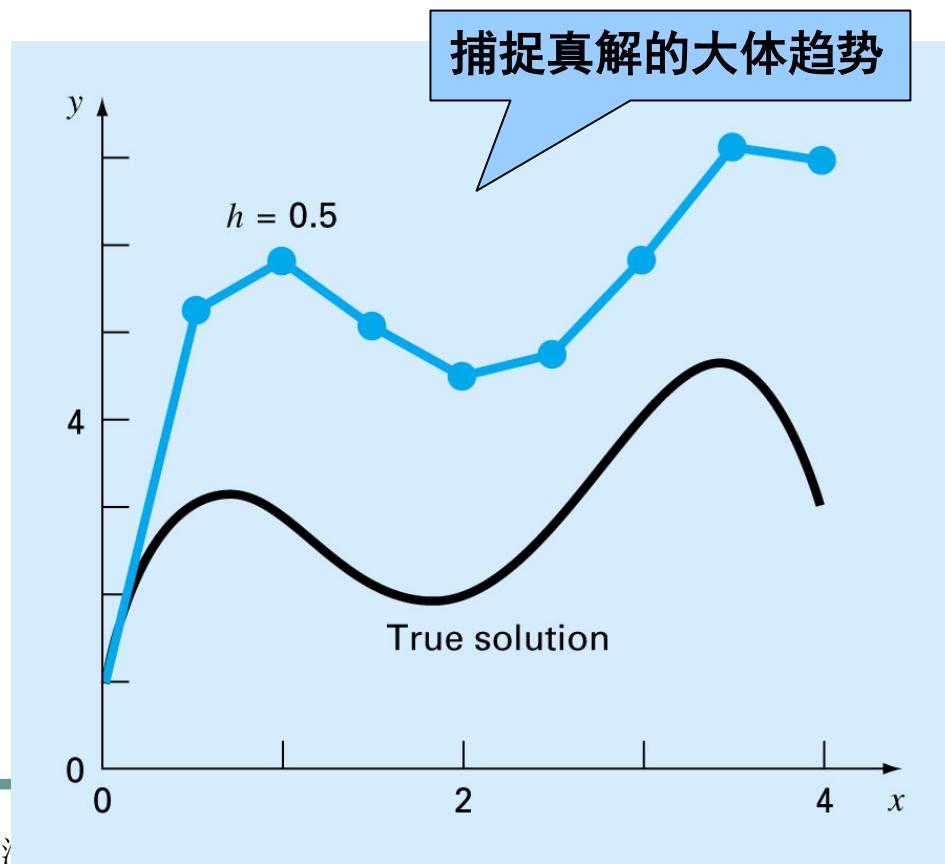


# 欧拉法——例

- 利用欧拉法求 $y$ ,  $x \in [0, 4]$ ,  $x=0$ 时 $y=1$ ,  $h=0.5$
- 解:  $y(0.5)=y(0)+f(0,1)h=5.25$

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$

$x$	$y_{\text{true}}$	$y_{\text{Euler}}$	百分比相对误差	
			全局	局部
0.0	1.00000	1.00000		
0.5	3.21875	5.25000	-63.1	-63.1
1.0	3.00000	5.87500	-95.8	-28.0
1.5	2.21875	5.12500	-131.0	-1.41
2.0	2.00000	4.50000	-125.0	20.5
2.5	2.71875	4.75000	-74.7	17.3
3.0	4.00000	5.87500	-46.9	4.0
3.5	4.71875	7.12500	-51.0	-11.3
4.0	3.00000	7.00000	-133.3	-53.0

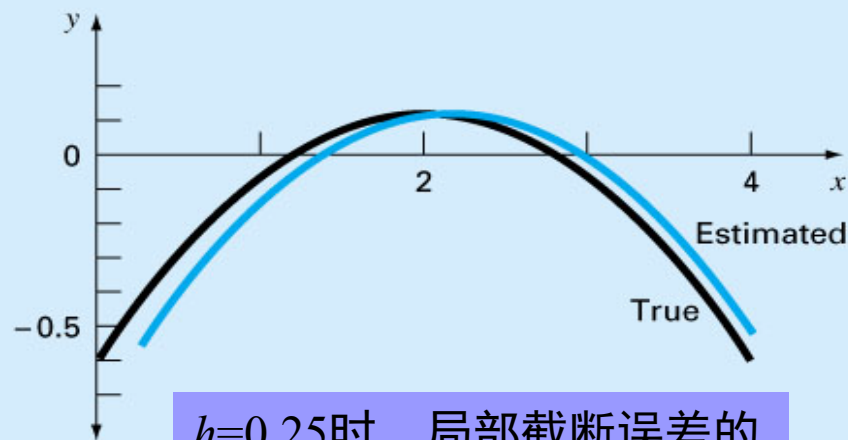
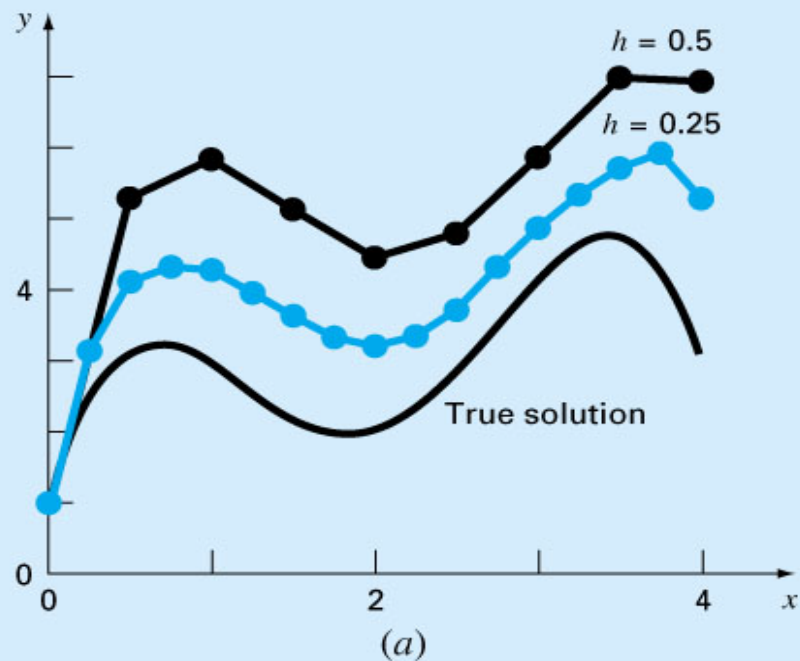


# 减小步长对欧拉法的影响

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$

●  $h=0.25$

- 全局截断误差减小了一半，局部截断误差变为原来的四分之一。
- 局部截断误差保持同号一般会使数值解越来越偏离真实解。
- 进一步缩小步长会使误差减小，但需要很大的计算量才能达到较高的精度。
- 尽管效率不高，但简单是主要的优势。



$h=0.25$ 时，局部截断误差的真实值和估计值

# 欧拉法——向后差商公式

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

$$\frac{y(x_{i+1}) - y(x_i)}{h} = f(x_{i+1}, y(x_{i+1})) - \frac{h}{2} y''(\xi_i)$$

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

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

- 隐式格式，要迭代求解

$$y_{i+1}^{(k+1)} = y_i + hf(x_{i+1}, y_{i+1}^{(k)})$$

$y_{i+1}^{(0)}$ 可以由向前差商公式求出

$$E = -\frac{h^2}{2} y''(\xi_i)$$

# 欧拉法——中心差商公式

$$y'(x_i) = \frac{y(x_{i+1}) - y(x_{i-1}))}{2h} - \frac{h^2}{6} y^{(3)}(\xi_i)$$

$$y_{i+1} = y_{i-1} + 2hf(x_i, y_i)$$

$$E = \frac{h^3}{3} y^{(3)}(\xi_i)$$

- 需要用到*i*-1和*i*的信息，多步，2阶格式
- 又称中点欧拉格式或两步欧拉格式

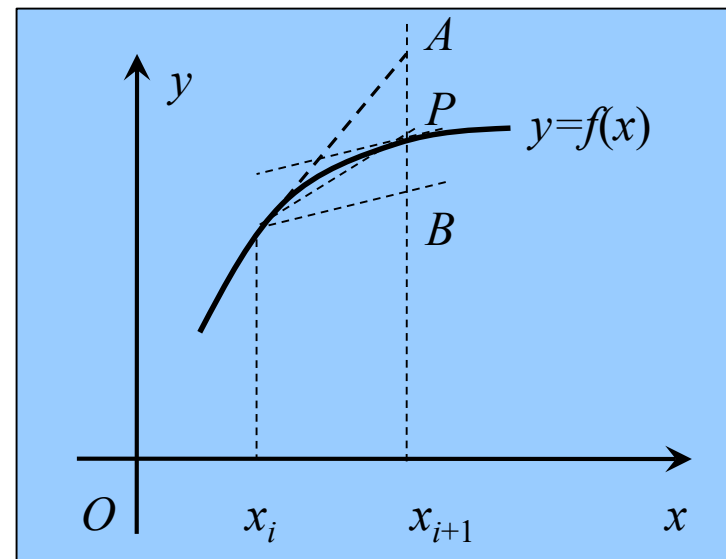


# 欧拉法——梯形格式

- 梯形积分公式 
$$\int_{x_i}^{x_{i+1}} f(x, y(x)) dx \approx \frac{h}{2} [f(x_i, y(x_i)) + f(x_{i+1}, y(x_{i+1}))]$$

$$y(x_{i+1}) = y(x_i) + \frac{h}{2} [f(x_i, y(x_i)) + f(x_{i+1}, y(x_{i+1}))] + \left( -\frac{h^3}{12} f''(\xi_i) \right)$$

- 梯形公式需要解方程求出 $y_{i+1}$ ，隐式公式
- 2阶精度
- 显式公式计算工作量小，但精度低，梯形公式提高了精度，但需要迭代求解



# 改进的欧拉方法

一种单步的预估-校正方法

- 修恩 (Heun) 法

- 标准欧拉格式

$$y_{i+1}^0 = y_i + hf(x_i, y_i) \quad \text{预估}$$

- 利用这个估计值计算在右端的斜率

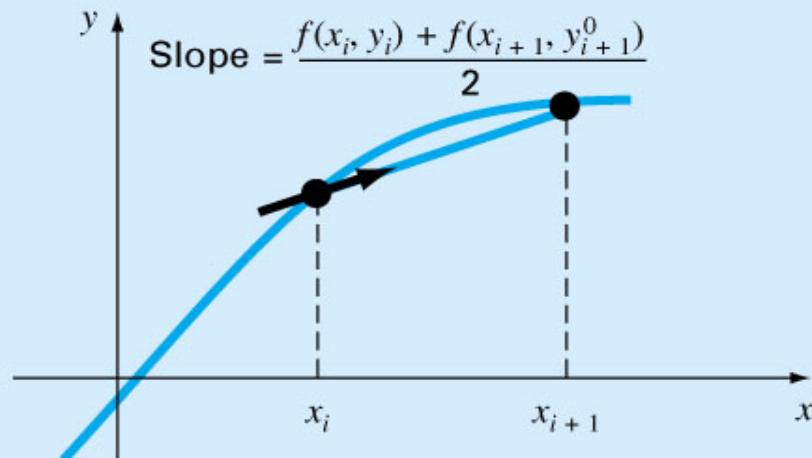
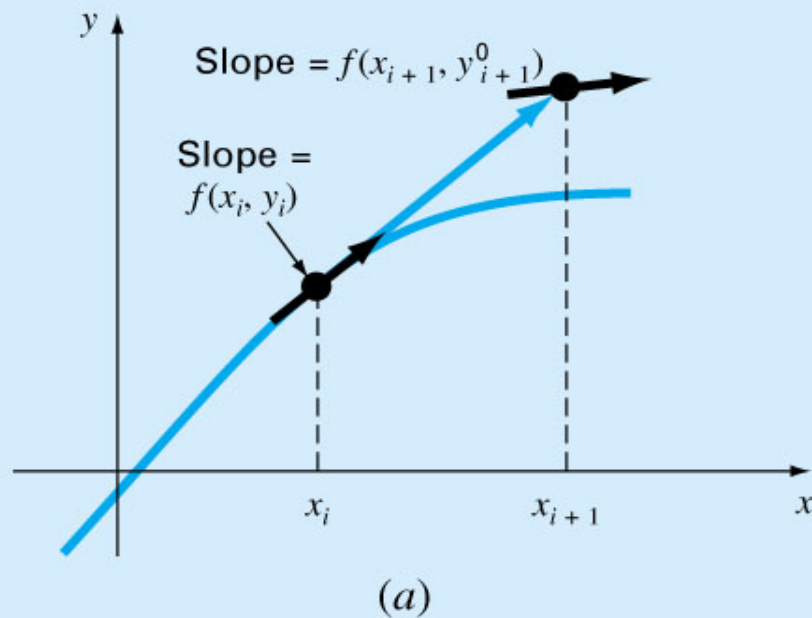
$$y'_{i+1} = f(x_{i+1}, y_{i+1}^0)$$

- 得出区间内的平均斜率

$$\bar{y}' = \frac{y'_i + y'_{i+1}}{2} = \frac{f(x_i, y_i) + f(x_{i+1}, y_{i+1}^0)}{2}$$

$$y_{i+1} = y_i + \frac{f(x_i, y_i) + f(x_{i+1}, y_{i+1}^0)}{2} h \quad \text{校正}$$

改进的欧拉方法具有2阶精度



校正可以迭代进行，不一定收敛到真实解，但收敛到一个截断误差有限的估计值

# 改进的欧拉方法——例

● 计算  $y' = 4e^{0.8x} - 0.5y$  由  $x=0$  到  $x=4$  的积分，取  $h=1$ ，初始条件  $x=0$  时  $y=2$

● 解：真实解为

$$y = \frac{4}{13}(e^{0.8x} - e^{-0.5x}) + 2e^{-0.5x}$$

$$y'_0 = 4e^0 - 0.5(2) = 3$$

$$y_1^0 = 2 + 3(1) = 5$$

$$y'_1 = f(1, 5) = 6.402164$$

$$\bar{y}' = \frac{6.402164 + 3}{2} = 4.701082$$

$$y_1 = 2 + 4.701082(1) = 6.701082$$

迭代过程中有时误差会增大，但步长充分小时，可收敛到一个值。校正过程收敛终止条件：

$$|\varepsilon_a| = \left| \frac{y_{i+1}^j - y_{i+1}^{j-1}}{y_{i+1}^j} \right| 100\%$$

迭代：

$$y_1 = 2 + \frac{[3 + 4e^{0.8(1)} - 0.5(6.701082)]}{2}(1) = 6.275811$$
$$y_1 = 2 + \frac{[3 + 4e^{0.8(1)} - 0.5(6.275811)]}{2}(1) = 6.382129$$

x	y <sub>true</sub>	校正的迭代次数			
		1		15	
		y <sub>Heun</sub>	ε <sub>t</sub>   (%)	y <sub>Heun</sub>	ε <sub>t</sub>   (%)
0	2.0000000	2.0000000	0.00	2.0000000	0.00
1	6.1946314	6.7010819	8.18	6.3608655	2.68
2	14.8439219	16.3197819	9.94	15.3022367	3.09
3	33.6771718	37.1992489	10.46	34.7432761	3.17
4	75.3389626	83.3377674	10.62	77.7350962	3.18

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法

# 龙格-库塔 (Runge-Kutta) 法

记为 $hT_{i+1}$

- Taylor展开

$$y(x_{i+1}) = y(x_i) + hy'(x_i) + \cdots + \frac{h^k}{k!} y^{(k)}(x_i) + \frac{h^{k+1}}{(k+1)!} y^{(k+1)}(\xi_i)$$

$$y'(x) = f(x, y) \quad y''(x) = f_x(x, y) + f_y(x, y) \cdot y' \quad y'''(x) = \cdots$$

$$y_{i+1} = y_i + hf(x_i, y_i) + \frac{h^2}{2!} (f_x(x_i, y_i) + f_y(x_i, y_i) \cdot f(x_i, y_i)) + \cdots$$

- 使用到了各阶偏导数，使用不便

$$y(x_{i+1}) = y(x_i) + hF(h, x_i, y(x_i), f) + hT_{i+1}$$

- 取 $(x_i, y_i)$ 及其附近的点做线性组合，表示 $F$ ——增量函数。

- 要求此时的展开精度相同。这种方法称为***Runge-Kutta***法

# 龙格-库塔 (Runge-Kutta) 法

- $a, p, q$  均为常数
- $k$  之间存在递归关系, 提高了龙格-库塔法的计算效率
- 项数  $n$  对应不同类型的龙格-库塔法,  $n=1$  时, 实际上就是欧拉法
- 令各项与 Taylor 展开式中各对应项相等, 就可以求出各个  $a, p, q$  的值

$$F(h, x, y, f) = a_1 k_1 + a_2 k_2 + \cdots + a_n k_n$$

$$k_1 = f(x_i, y_i)$$

$$k_2 = f(x_i + p_1 h, y_i + q_{11} k_1 h)$$

$$\vdots$$

$$k_n = f(x_i + p_{n-1} h, y_i + h \sum_{j=1}^{n-1} q_{(n-1),j} k_j)$$

# 二阶龙格-库塔法

$$y_{i+1} = y_i + (a_1 k_1 + a_2 k_2)h \quad k_1 = f(x_i, y_i) \quad k_2 = f(x_i + p_1 h, y_i + q_{11} k_1 h)$$

$$y(x_{i+1}) = y(x_i) + h \left[ f(x_i, y_i) + \frac{h}{2!} \left( f_x(x_i, y_i) + f_y(x_i, y_i) \cdot f(x_i, y_i) \right) \right] + hT_{i+1}$$

$$k_2 = f(x_i, y_i) + p_1 h f_x(x_i, y_i) + q_{11} k_1 h f_y(x_i, y_i) + O(h^2)$$

$$\begin{cases} a_1 + a_2 = 1 \\ a_2 p_1 = 1/2 \\ a_2 q_{11} = 1/2 \end{cases} \quad \begin{cases} a_2 = 1 - a_1 \\ p_1 = 1/(2a_2) \\ q_{11} = 1/(2a_2) \end{cases}$$

- 四个未知数，三个方程
- 一族格式，统称为二阶龙格-库塔格式

# 二阶龙格-库塔法

$$\begin{cases} a_2 = 1 - a_1 \\ p_1 = q_{11} = 1 / (2a_2) \end{cases}$$

局部截断误差  $O(h^3)$   
全局截断误差  $O(h^2)$

- 单步校正的修恩法

$$\begin{cases} a_1 = a_2 = 1/2 \\ p_1 = q_{11} = 1 \end{cases} \rightarrow$$

$$\begin{cases} y_{i+1} = y_i + h[k_1 + k_2] / 2 \\ k_1 = f(x_i, y_i) \\ k_2 = f(x_i + h, y_i + hk_1) \end{cases}$$

- 中点公式

$$\begin{cases} a_1 = 0, a_2 = 1 \\ p_1 = q_{11} = 1/2 \end{cases} \rightarrow$$

$$\begin{cases} y_{i+1} = y_i + hk_2 \\ k_1 = f(x_i, y_i) \\ k_2 = f(x_i + h/2, y_i + hk_1/2) \end{cases}$$

欧拉法预测出的  
区间中点

对应的截断  
误差界最小

- 罗森 (Ralston) 方法

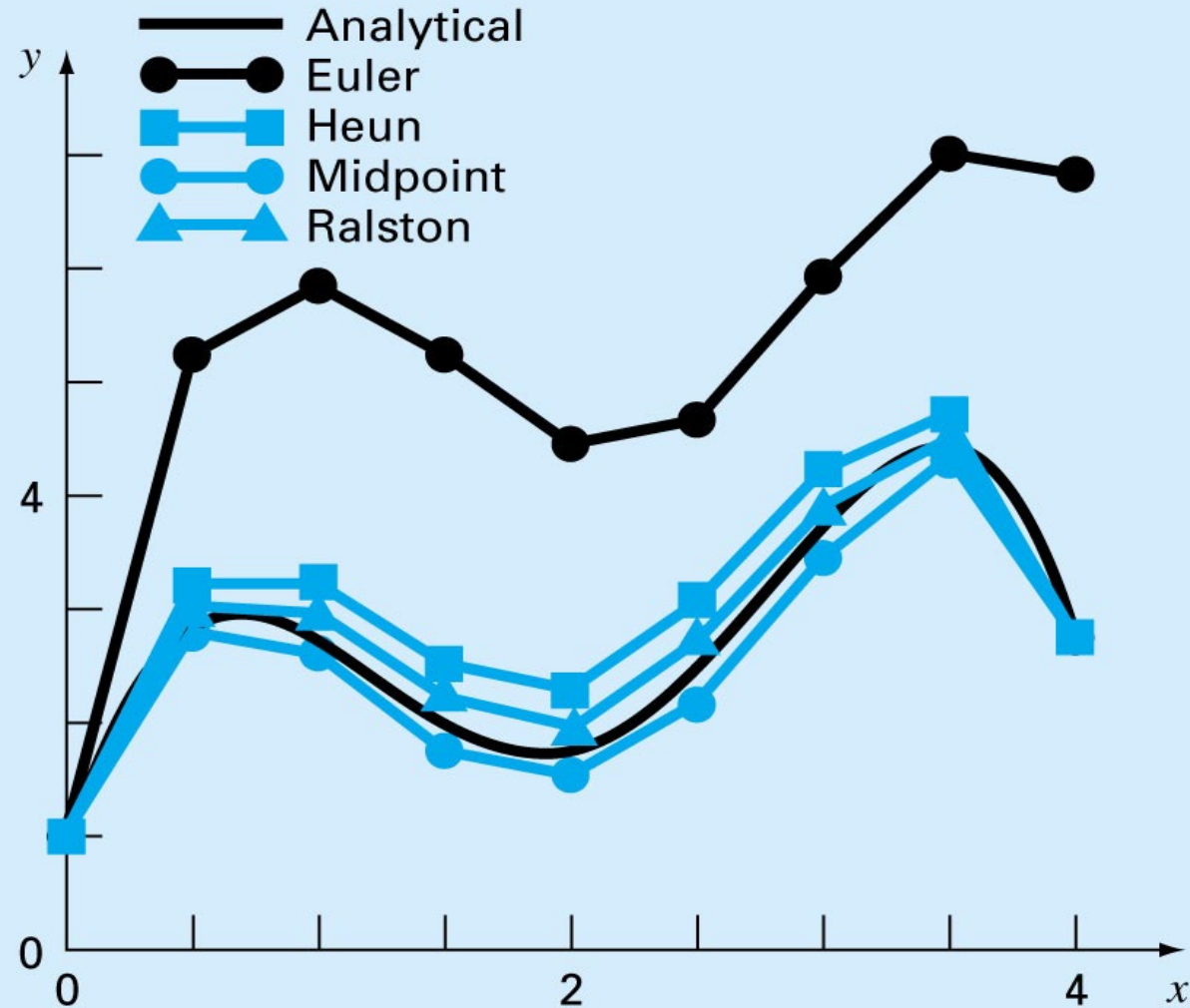
$$\begin{cases} a_1 = 1/3, a_2 = 2/3 \\ p_1 = q_{11} = 3/4 \end{cases} \rightarrow$$

$$\begin{cases} y_{i+1} = y_i + h(k_1/3 + 2k_2/3) \\ k_1 = f(x_i, y_i) \\ k_2 = f(x_i + 3h/4, y_i + 3hk_1/4) \end{cases}$$



# 二阶龙格-库塔法——例

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$



# 二阶龙格-库塔法——例

$$\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$$

$x$	$y_{\text{true}}$	修恩法		中点方法		二阶罗森方法	
		$y$	$ \varepsilon_t (\%)$	$y$	$ \varepsilon_t (\%)$	$y$	$ \varepsilon_t (\%)$
0.0	1.00000	1.00000	0	1.00000	0	1.00000	0
0.5	3.21875	3.43750	6.8	3.109375	3.4	3.277344	1.8
1.0	3.00000	3.37500	12.5	2.81250	6.3	3.101563	3.4
1.5	2.21875	2.68750	21.1	1.984375	10.6	2.347656	5.8
2.0	2.00000	2.50000	25.0	1.75	12.5	2.140625	7.0
2.5	2.71875	3.18750	17.2	2.484375	8.6	2.855469	5.0
3.0	4.00000	4.37500	9.4	3.81250	4.7	4.117188	2.9
3.5	4.71875	4.93750	4.6	4.609375	2.3	4.800781	1.7
4.0	3.00000	3.00000	0	3	0	3.031250	1.0

# 三阶龙格-库塔法

- 含有8个未知数的6个方程，需要指定两个未知数的值
- 一类常用的公式

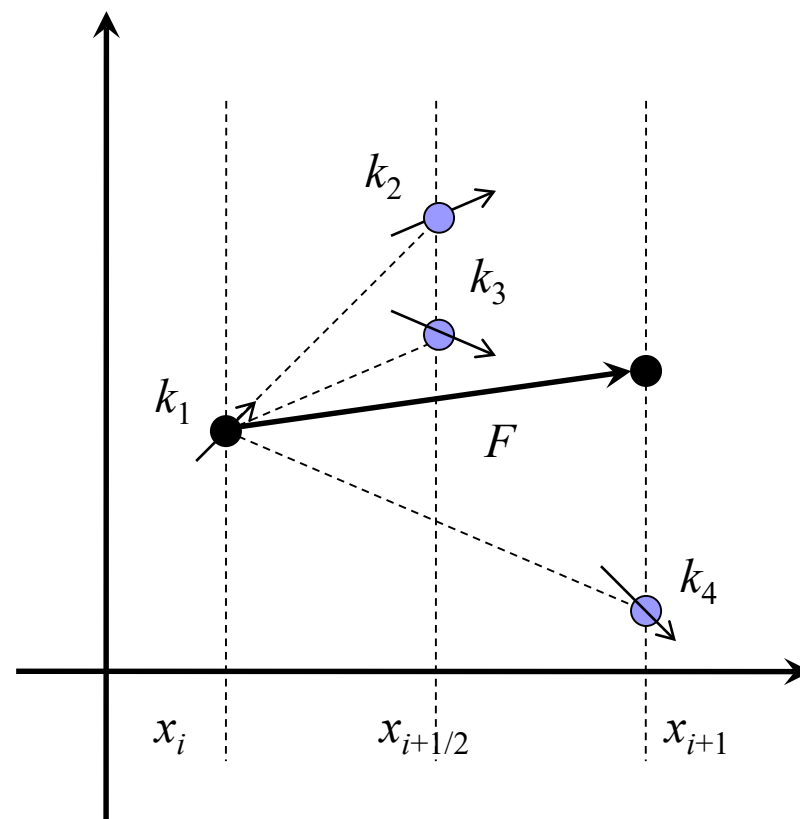
$$\begin{cases} y_{i+1} = y_i + \frac{h}{6}[k_1 + 4k_2 + k_3] \\ k_1 = f(x_i, y_i) \\ k_2 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_1) \\ k_3 = f(x_i + h, y_i - hk_1 + 2hk_2) \end{cases}$$

- 如果导数是 $x$ 的函数，则方法退化为Simpson1/3法则
- 局部截断误差 $O(h^4)$ ，全局截断误差 $O(h^3)$

# 四阶龙格-库塔法

- 经典的四阶龙格-库塔法

$$\begin{cases} y_{i+1} = y_i + \frac{h}{6}[k_1 + 2k_2 + 2k_3 + k_4] \\ k_1 = f(x_i, y_i) \\ k_2 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_1) \\ k_3 = f(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_2) \\ k_4 = f(x_i + h, y_i + hk_3) \end{cases}$$



# 四阶龙格-库塔法——例

- $\frac{dy}{dx} = -2x^3 + 12x^2 - 20x + 8.5$

$$k_1 = 8.5, k_2 = 4.21875, k_3 = 4.21875, k_4 = 1.25$$

$$y(0.5) = 3.21875$$

结果精确，因为真解是四次多项式

- $y' = 4e^{0.8x} - 0.5y$

$$k_1 = 3, k_2 = 3.510611, k_3 = 3.446785, k_4 = 4.105603$$

$$y(0.5) = 3.751699$$

$$y_{true}(0.5) = 3.751521$$

# 高阶龙格-库塔法

## ● 5阶Butcher公式

$$\left\{ \begin{array}{l} y_{i+1} = y_i + \frac{h}{90} [7k_1 + 32k_3 + 12k_4 + 32k_5 + 7k_6] \\ k_1 = f(x_i, y_i) \\ k_2 = f(x_i + \frac{h}{4}, y_i + \frac{h}{4}k_1) \\ k_3 = f(x_i + \frac{h}{4}, y_i + \frac{h}{8}k_1 + \frac{h}{8}k_2) \\ k_4 = f(x_i + \frac{h}{2}, y_i - \frac{h}{2}k_2 + hk_3) \\ k_5 = f(x_i + \frac{3h}{4}, y_i + \frac{3h}{16}k_1 + \frac{9h}{16}k_4) \\ k_6 = f(x_i + h, y_i - \frac{3h}{7}k_1 + \frac{2h}{7}k_2 + \frac{12h}{7}k_3 - \frac{12h}{7}k_4 + \frac{8h}{7}k_5) \end{array} \right.$$

四阶以上的公式，精度的提高不足以弥补计算量和复杂性的增加

函数值计算次数	2	3	4	6	7	$p+2$
阶 $p$	2	3	4	5	6	$\geq 7$

# 步长的自动选择

- $y(x_{i+1}) - y_{i+1} = \lambda h_1^{p+1} y^{p+1}(\xi_1)$
- 取步长分别为  $h$  和  $h/2$

$$\begin{aligned}y(x_{i+1}) - y_{i+1}^{(h)} &= ch^{p+1} \\ y(x_{i+1}) - y_{i+1}^{(h/2)} &= 2c(h/2)^{p+1}\end{aligned}$$

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

$$y(x_{i+1}) = \frac{2^p y_{i+1}^{(h/2)} - y_{i+1}^{(h)}}{2^p - 1}$$

计算不同步长的  
龙格-库塔法的  
预测值，相减作  
为误差估计

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

- 记

$$\Delta = |y_{i+1}^{(h/2)} - y_{i+1}^{(h)}|$$

- 如果  $\Delta < \varepsilon$ ，则反复加倍步长进行计算，直到  $\Delta > \varepsilon$  并输出上一次步长的计算结果（或校正后的结果）
- 如果  $\Delta > \varepsilon$ ，则反复减半步长进行计算，直到  $\Delta < \varepsilon$  并输出最后一次步长的计算结果（或校正后的结果）

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法



# 线性多步法

- 用前面若干节点处的函数值和导数值的线性组合来计算 $y(x_{i+1})$ 的近似值 $y_{i+1}$ ，通称为线性多步法

$$y_{i+1} = \sum_{j=1}^k \alpha_j y_{i+1-j} + h \sum_{j=0}^k \beta_j f_{i+1-j}$$

$\alpha_j$ 和 $\beta_j$ 为常数

$$f_{i+1-j} = f(x_{i+1-j}, y_{i+1-j})$$

- $k=1$ 时为单步法， $k>1$ 时为多步法（ $k$ 步法）
- 当  $\beta_0 \neq 0$  时，为隐式公式；  $\beta_0 = 0$  则为显式公式
- 无法自动启动，需要其它方法计算出前几步的值
- 系数可用泰勒展开后的待定系数法或数值积分法确定

# 基于数值积分的构造法

## ——阿达姆斯（Adams）方法

- 将  $y' = f(x, y)$  从  $x_i$  到  $x_{i+1}$  积分

$$y(x_{i+1}) = y(x_i) + \int_{x_i}^{x_{i+1}} f(x, y(x)) dx$$

- 只要近似地算出右边的积分  $I_k \approx \int_{x_i}^{x_{i+1}} f(x, y(x)) dx$ ，则可通过  $y_{i+1} = y_i + I_k$  近似  $y(x_{i+1})$ 。而选用不同近似式  $I_k$ ，可得到不同的计算公式
- 以  $f_i, f_{i-1}, \dots, f_{i-k}$  共  $k+1$  个数据构造一个  $k$  阶牛顿向后插值多项式，可以得到 **Adams显式方法**
- 以  $f_{i+1}, f_i, \dots, f_{i-k+1}$  共  $k+1$  个数据构造一个  $k$  阶牛顿内插多项式，可以得到 **Adams隐式方法**

# 显式Adams格式

$$y_{i+1} = y_i + h \sum_{j=0}^k \beta_j f_{i-j}$$

● 局部截断误差

$$R_{i+1} = B_k h^{k+2} y^{(k+2)}(\xi_{i+1})$$

$k$	$\beta_0$	$\beta_1$	$\beta_2$	$\beta_3$	$\beta_4$	$\beta_5$	$B$
0	1						1/2
1	3/2	-1/2					5/12
2	23/12	-16/12	5/12				3/8
3	55/24	-59/24	37/24	-9/24			251/720
4	1901/720	-2774/720	2626/720	-1274/720	251/720		95/288
5	4277/1440	-7923/1440	9982/1440	-7298/1440	2877/1440	-476/1440	10987/60480

$k=0$ 为显式欧拉格式

$k=1$ 时:

$$y_{i+1} = y_i + \frac{h}{2}(3f_i - f_{i-1})$$

Adams四步显式格式,  
一种最常用的多步算法,  
四阶精度(Adams-  
Bashforth method)

$k=3$ 时:

$$y_{i+1} = y_i + \frac{h}{24}(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3})$$

数值计算方法

# 隐式Adams格式

$$y_{i+1} = y_i + h \sum_{j=0}^k \beta_j f_{i-j+1}$$

● 局部截断误差  $R_{i+1} = B_k h^{k+2} y^{(k+2)}(\xi_{i+1})$

$k$	$\beta_0$	$\beta_1$	$\beta_2$	$\beta_3$	$\beta_4$	$\beta_5$	$B$
0	1						-1/2
1	1/2	1/2					-1/12
2	5/12	8/12	-1/12				-1/24
3	9/24	19/24	-5/24	1/24			-19/720
4	251/720	646/720	-261/720	106/720	-19/720		-3/160
5	475/1440	1427/1440	-798/1440	482/1440	-173/1440	27/1440	-863/60480

$k=0$ 为隐式欧拉格式

$k=1$ 为梯形格式

$k=3$ 时:

$$y_{i+1} = y_i + \frac{h}{24} (9f_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})$$

三步四阶Adams隐式格式,  
四阶精度(Adams–  
Moulton method)

# Adams显式与隐式方法的比较

- 同一阶数下，隐式的局部截断误差的系数绝对值小于显式的局部截断误差的系数绝对值
- 显式的计算工作量比隐式的小
- 隐式的稳定范围比显式的大

# Adams预测-校正方法

- 把显式和隐式Adams格式结合起来，构成预测-校正方法
- 先用显式方法算出近似值，作为隐式方法的预测值

预测：

$$\bar{y}_{i+1} = y_i + \frac{h}{24}(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3})$$

$$\bar{f}_{i+1} = f(x_{i+1}, \bar{y}_{i+1})$$

校正：

$$y_{i+1} = y_i + \frac{h}{24}(9\bar{f}_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})$$

$$f_{i+1} = f(x_{i+1}, y_{i+1})$$

# Adams预测-校正方法的改进

- 与Richardson外推法相结合，可以提高计算精度而不会增加过大的计算工作量

$p_i$ 和 $C_i$ 分别为第 $i$ 步的预测值和校正值

预测:

$$p_{i+1} = y_i + \frac{h}{24}(55f_i - 59f_{i-1} + 37f_{i-2} - 9f_{i-3})$$

改进:

$$\bar{y}_{i+1} = p_{i+1} + \frac{251}{270}(C_i - p_i)$$

$$\bar{f}_{i+1} = f(x_{i+1}, \bar{y}_{i+1})$$

校正:

$$C_{i+1} = y_i + \frac{h}{24}(9\bar{f}_{i+1} + 19f_i - 5f_{i-1} + f_{i-2})$$

改进:

$$y_{i+1} = C_{i+1} - \frac{19}{270}(C_{i+1} - p_{i+1})$$

$$f_{i+1} = f(x_{i+1}, y_{i+1})$$

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法



# 算法的稳定性和收敛性

- 稳定性
  - 反映某一步计算中出现的舍入误差对计算结果的影响，讨论这种误差的积累能否得到控制的问题
- 收敛性
  - 讨论在 $h \rightarrow 0$ 时数值解 $y_i$ 能否收敛到方程准确解 $y(x_i)$ 的问题
- 收敛性与稳定性从两个不同角度描述了微分方程数值解的实用价值，只有既收敛又稳定的方法，才可能提供比较可靠的计算结果。

# 算法的稳定性——例

$$\begin{cases} y'(x) = -30y(x) \\ y(0) = 1 \end{cases} \quad \text{精确解: } y = e^{-30x}$$

$h=0.1$ , 计算 $y(1.5)$

三种方法所得结果相差悬殊, 都与精确解相差甚巨, 算法稳定性差!

欧拉方法	四阶R-K方法	Adams预测-校正方法	精确解
$-3.27675 \times 10^4$	$1.18719 \times 10^2$	$2.41152 \times 10^6$	$2.86252 \times 10^{-20}$

定义: 绝对稳定性

对于给定的微分方程和给定的步长 $h$ , 如果在 $y_i$ 上有大小为 $\delta$ 的误差, 即计算得 $y_i^* = y_i + \delta$ , 而引起其后值 $y_j (j > i)$ 的变化小于 $\delta$ , 即 $|y_j^* - y_j| < |\delta|$ , 则称方法是绝对稳定的。

# 算法的稳定性

$$\begin{cases} \frac{dy}{dx} = \lambda y & (\operatorname{Re} \lambda < 0) \\ y(0) = y_0 \end{cases}$$

$$\sum_{j=0}^k a_j y_{n+j} = h \sum_{j=0}^k b_j f(x_{n+j}, y_{n+j})$$

$$\sum_{j=0}^k a_j y_{n+j} = \lambda h \sum_{j=0}^k b_j y_{n+j} \quad \operatorname{Re}(\lambda) < 0$$

对于给定的初始误差  $e_0, e_1, \dots, e_{k-1}$ ，误差方程具有同样的形式

$$\sum_{j=0}^k a_j e_{n+j} = \lambda h \sum_{j=0}^k b_j e_{n+j} \quad \operatorname{Re}(\lambda) < 0$$

# 算法的稳定性

对于一般的微分方程，可近似地取 $\lambda = -|\partial f / \partial y|$ 进行稳定判别，以确定步长应取的范围，使 $\lambda h$ 属于绝对稳定区间。

定义：差分方程称为绝对稳定的，若差分方程作用到微分方程

$$\frac{dy}{dx} = \lambda y \quad (\operatorname{Re} \lambda < 0)$$

时，对任意的初值，总存在左半复平面上的一个区域，当 $\lambda h$ 在这个区域时，差分方程的解趋于0。这个区域称为稳定区域。

考察隐式欧拉法

$$y_{i+1} = y_i + h\lambda y_{i+1}$$

$$y_{i+1} = \left( \frac{1}{1 - \lambda h} \right) y_i \Rightarrow e_{i+1} = \left( \frac{1}{1 - \lambda h} \right)^{i+1} e_0$$

可见绝对稳定区域为： $|1 - \lambda h| > 1$

显式欧拉法

$$y_{i+1} = y_i + \lambda h y_i$$

$$e_{i+1} = e_i + \lambda h e_i$$

$$|1 + \lambda h| < 1$$

如果要保证算法的稳定性，则步长受到限制， $\lambda$ 的绝对值越大，限制的 $h$ 值越小。隐式欧拉法的绝对稳定性比同阶的显式法的好。

# 收敛性

$$y_{i+1} = y_i + h\Phi(x_i, y_i, h)$$

增量函数

定义：如果一个数值方法对任意固定点 $x_i = x_0 + ih$ ，当 $h = (x_i - x_0)/i \rightarrow 0$  (即 $i \rightarrow \infty$ ) 时有 $y_i \rightarrow y(x_i)$ ，则称该法是收敛的。

- 数值方法的收敛性并不涉及计算过程中的舍入误差，而只与方法的整体截断误差有关。

定理：若增量函数关于 $y$ 满足Lipschitz 条件，且单步法的局部截断误差满足 $E(x, h) \leq Mh^{p+1}$ ，则其整体截断误差 $\varepsilon_i = y(x_i) - y_i$ 满足

$L$ 为增量函数关于 $y$ 的Lipschitz常数

$$|\varepsilon_i| = |y(x_i) - y_i| \leq e^{L(b-a)} |y(x_0) - y_0| + \frac{Mh^p}{L} (e^{L(b-a)} - 1)$$

如果初值不存在误差， $\varepsilon_0 = y(x_0) - y_0 = 0$ ， $p$ 阶单步法的整体截断误差为 $O(h^p)$ 。

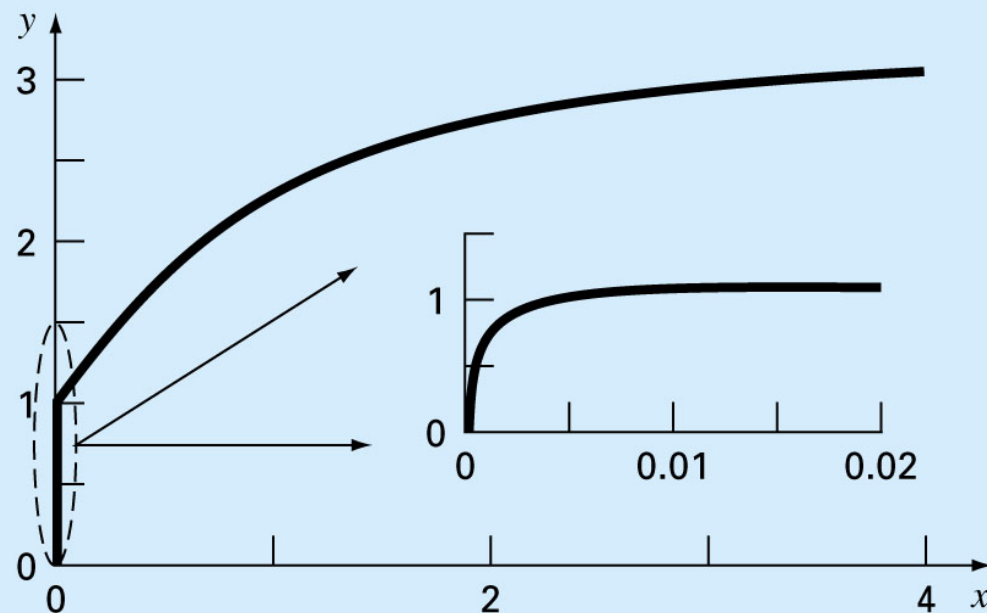
# 刚性（stiff）常微分方程

- 刚性系统指的是同时存在快变成分和慢变成分的系统。多数情况下，快变成分发生在短暂的瞬间，然后迅速衰减，此后，解由慢变成分控制。
- 尽管瞬变现象仅存在于积分区间的一个很小的部分，但是它们决定了整个求解过程中的步长

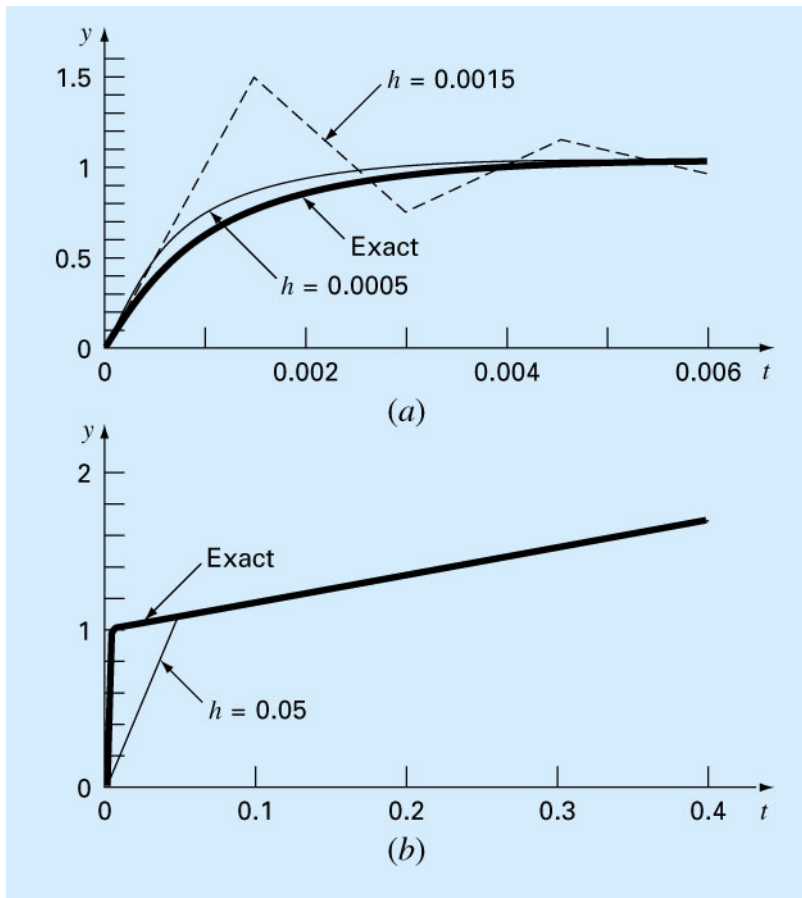
$$\frac{dy}{dt} = -1000y + 3000 - 2000e^{-t}$$

$$y(0) = 0 \quad \downarrow$$

$$y = 3 - 0.998e^{-1000t} + 2.002e^{-t}$$



# 显式和隐式欧拉方法



$$\frac{dy}{dt} = -1000y + 3000 - 2000e^{-t}$$

$$y_{i+1} = y_i + (-1000y_i + 3000 - 2000e^{-t_i})h$$

$$y_{i+1} = y_i + (-1000y_{i+1} + 3000 - 2000e^{-t_i})h$$
$$\Rightarrow y_{i+1} = \frac{y_i + (3000 - 2000e^{-t_i})h}{1 + 1000h}$$

隐式欧拉法只有一阶精度，其他求解刚性微分方程的方法请参考相关文献

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法



# 一阶常微分方程组

$$\begin{cases} \frac{dy_1}{dx} = f_1(x, y_1, y_2, \dots, y_n) \\ \vdots \\ \frac{dy_n}{dx} = f_n(x, y_1, y_2, \dots, y_n) \\ y_1(a) = y_{10} \\ \vdots \\ y_n(a) = y_{n0} \end{cases}$$

- 在当前步中，对每个方程求解

# 一阶常微分方程组

$$\begin{cases} \frac{dy}{dx} = f(x, y, z) \\ \frac{dz}{dx} = g(x, y, z) \\ y(a) = y_0 \\ z(a) = z_0 \end{cases}$$

- 欧拉公式 
$$\begin{cases} y_{i+1} = y_i + hf(x_i, y_i, z_i) \\ z_{i+1} = z_i + hg(x_i, y_i, z_i) \end{cases}$$
- 改进的欧拉法

$$\begin{aligned} P \quad & y_{i+1}^0 = y_i + hf(x_i, y_i, z_i) \\ & z_{i+1}^0 = z_i + hg(x_i, y_i, z_i) \\ C \quad & y_{i+1} = y_i + \frac{h}{2}[f(x_i, y_i, z_i) + f(x_{i+1}, y_{i+1}^0, z_{i+1}^0)] \\ & z_{i+1} = z_i + \frac{h}{2}[g(x_i, y_i, z_i) + g(x_{i+1}, y_{i+1}^0, z_{i+1}^0)] \end{aligned}$$

# 一阶常微分方程组

- 四阶龙格-库塔法

$$\begin{pmatrix} y_{i+1} \\ z_{i+1} \end{pmatrix} = \begin{pmatrix} y_i \\ z_i \end{pmatrix} + \frac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4)$$

$$k_1 = \begin{pmatrix} f(x_i, y_i, z_i) \\ g(x_i, y_i, z_i) \end{pmatrix}$$

$$k_2 = \begin{pmatrix} f(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_1^{(1)}, z_i + \frac{h}{2}k_1^{(2)}) \\ g(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_1^{(1)}, z_i + \frac{h}{2}k_1^{(2)}) \end{pmatrix}$$

$$k_3 = \begin{pmatrix} f(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_2^{(1)}, z_i + \frac{h}{2}k_2^{(2)}) \\ g(x_i + \frac{h}{2}, y_i + \frac{h}{2}k_2^{(1)}, z_i + \frac{h}{2}k_2^{(2)}) \end{pmatrix}$$
$$k_4 = \begin{pmatrix} f(x_i + h, y_i + hk_3^{(1)}, z_i + hk_3^{(2)}) \\ g(x_i + h, y_i + hk_3^{(1)}, z_i + hk_3^{(2)}) \end{pmatrix}$$

# 一阶常微分方程组——例

- 取步长 $h=0.5$ ，积分到2
- 欧拉法——第一步
  - $y_1(0.5)=4+[-0.5(4)]0.5=3$
  - $y_2(0.5)=6+[4-0.3(6)-0.1(4)]0.5=6.9$

$x$	欧拉法		四阶龙格-库塔法	
	$y_1$	$y_2$	$y_1$	$y_2$
0	4	6	4	6
0.5	3	6.9	3.115234	6.857670
1.0	2.25	7.715	2.426171	7.632106
1.5	1.6875	8.44525	1.889523	8.326886
2.0	1.265625	9.094087	1.471577	8.946865

$$\begin{cases} \frac{dy_1}{dx} = -0.5y_1 \\ \frac{dy_2}{dx} = 4 - 0.3y_2 - 0.1y_1 \\ y_1(0) = 4 \\ y_2(0) = 6 \end{cases}$$

# 高阶微分方程

$$\begin{cases} \frac{d^n y}{dx^n} = f_1(x, y, y', \dots, y^{(n-1)}) \\ y(a) = y_0 \\ y'(a) = y'_0 \\ \vdots \\ y^{(n-1)}(a) = y_0^{(n-1)} \end{cases}$$

$$\begin{cases} y = y_1 \\ \frac{dy_1}{dx} = y_2 \\ \vdots \\ \frac{dy_{n-1}}{dx} = y_n \end{cases}$$



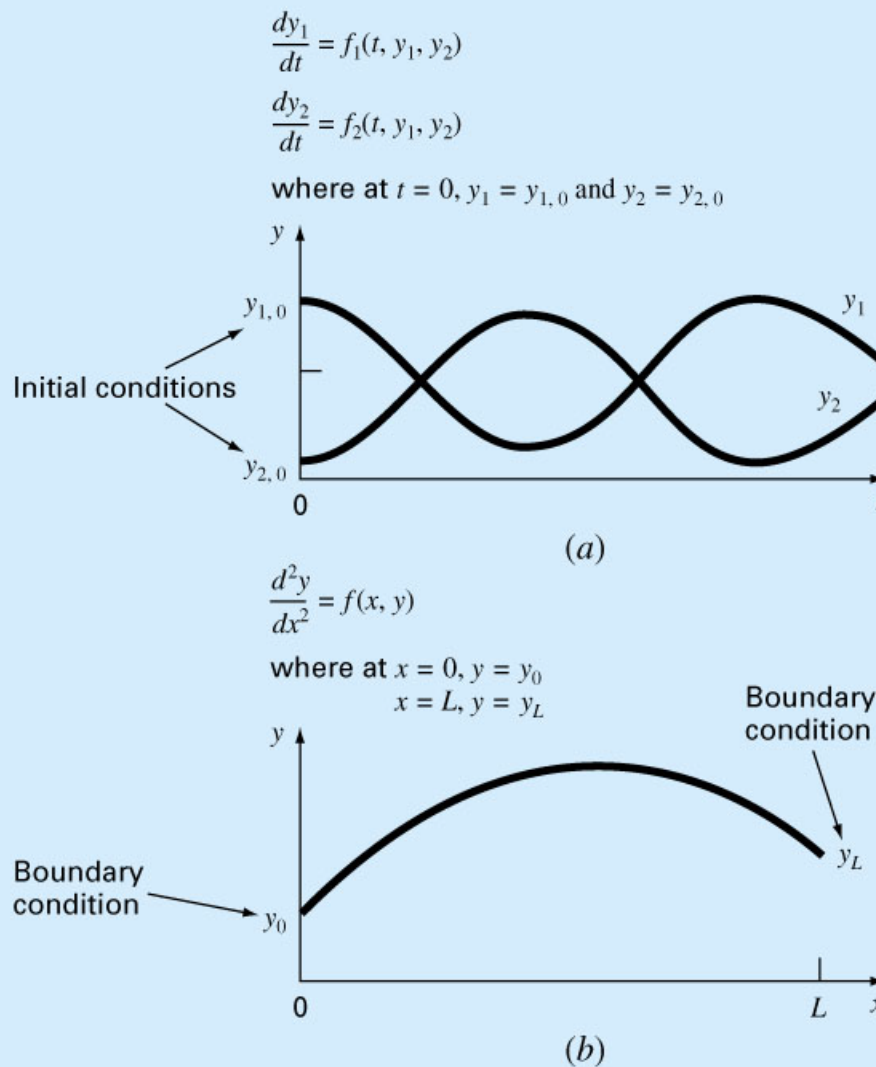
$$\begin{cases} \frac{dy_1}{dx} = y_2 \\ \vdots \\ \frac{dy_n}{dx} = f(x, y_1, \dots, y_n) \\ y_1(a) = y_0 \\ \vdots \\ y_n(a) = y_0^{(n-1)} \end{cases}$$

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法

# 初值问题与边值问题

- 初值问题(Initial-Value):
  - 所有的条件都在自变量的同一取值处指定
- 边值问题(Boundary-Value):
  - 已知条件在自变量的不同点处给出
  - 一般为系统的极值点或边界点



# 二阶常微分方程的边值问题

- $y''=f(x,y,y')$   $x\in[a,b]$

第一边值条件  $y(a)=\alpha, y(b)=\beta$

第二边值条件  $y'(a)=\alpha, y'(b)=\beta$

第三边值条件  $y'(a)-\alpha_0 y(a)=\alpha_1, y'(b)+\beta_0 y(b)=\beta_1$   $\alpha_0, \beta_0 \geq 0, \alpha_0 + \beta_0 > 0$

- 边界条件的不同, 可使边值问题的解不存在或有无穷个解。

$$y''+y=0$$

通解为  $y=C_1 \sin x + C_2 \cos x$

边界条件为:

$$y(0)=0; y(2\pi)=0$$

有无穷个解

边界条件为:

$$y(0)=0; y(2\pi)=\alpha \neq 0$$

无解



# 边值问题的通用方法

$$\frac{d^2 T}{dx^2} + h'(T_a - T) = 0$$

$$T_a = 20$$

$$L = 10\text{ m}$$

$$h' = 0.01\text{ m}^{-2}$$

热传导系数

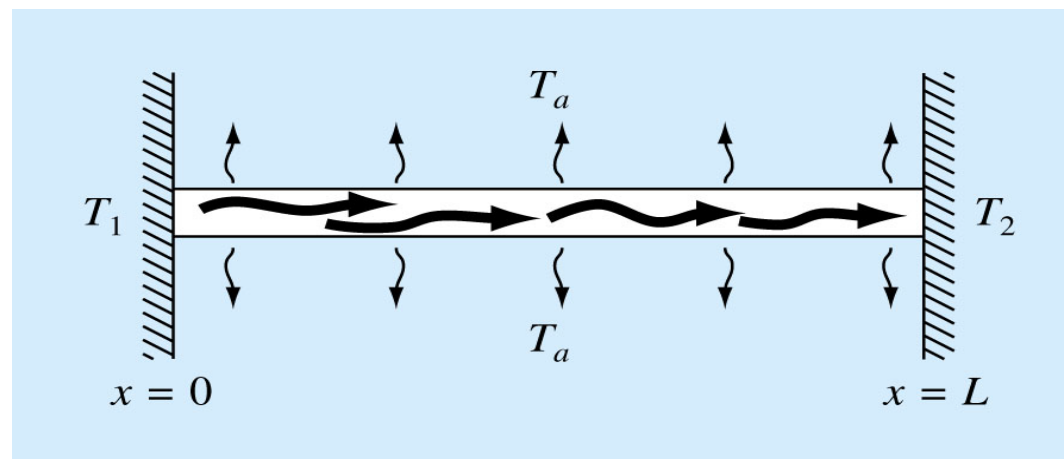
$$T(0) = T_1 = 40$$

$$T(L) = T_2 = 200$$

边界条件

解析解

$$T = 73.4523e^{0.1x} - 53.4523e^{-0.1x} + 20$$



位于两物体之间的非绝热杆

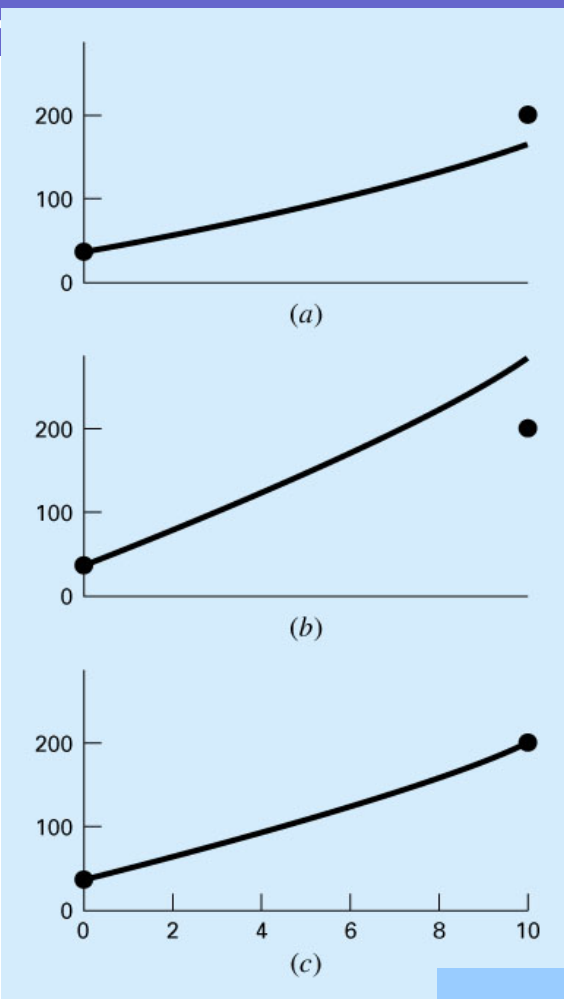
$$T_1 > T_a \quad T_2 > T_a$$

系统处于稳定状态，需要确定杆上 $[0, L]$ 的温度

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法

# 打靶法 (Shooting Method)



## 基本思想

- 把边值问题转化为初值问题

$$\frac{d^2 T}{dx^2} + h'(T_a - T) = 0$$

线性

$$\begin{aligned} \frac{dT}{dx} &= z \\ \frac{dz}{dx} &= h'(T - T_a) \end{aligned}$$

- 为了求解一阶常微分方程组，需要已知 $z(0)$
- $z(0)=10$ ，可根据步长为2的四阶R-K方法得到 $T(10)=168.3797$
- $z(0)=20$ ，可根据步长为2的四阶R-K方法得到 $T(10)=285.8980$

满足另一边  
界条件

$$z(0) = 10 + \frac{20 - 10}{285.8980 - 168.3797} (200 - 168.3797) = 12.6907$$

数值计算方法

# 打靶法

- 对一般的非线性情况，确定初值条件可以通过求解非线性方程得到。
- 求解高阶方程时，需要假设两个或更多条件，造成求解困难。

# 本章内容

- 概述
- 初值问题的数值解法
  - 欧拉（Euler）法
  - 龙格-库塔（Runge-Kutta）法
  - 多步法
  - 稳定性、收敛性和刚性问题
  - 微分方程组和高阶微分方程
- 边值问题
  - 打靶法
  - 差分法

# 差分法

- 基本思想

- 用有限差商代替导数，转化为联立的代数方程组

$$\frac{d^2T}{dx^2} = \frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2}$$

二阶导数的有限差商逼近

$$\frac{T_{i+1} - 2T_i + T_{i-1}}{\Delta x^2} - h'(T_i - T_a) = 0$$

将区间分成 $n$ 等分，取等距点，每个内点适用此方程

$$-T_{i-1} + (2 + h'\Delta x^2)T_i - T_{i+1} = h'\Delta x^2 T_a$$

三对角方程组

$$\begin{bmatrix} 2.04 & -1 & 0 & 0 \\ -1 & 2.04 & -1 & 0 \\ & -1 & 2.04 & -1 \\ & & -1 & 2.04 \end{bmatrix} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \\ T_4 \end{bmatrix} = \begin{bmatrix} 40.8 \\ 0.8 \\ 0.8 \\ 200.8 \end{bmatrix}$$

取 $\Delta x=2\text{m}$

# 边值问题——例

$x$	真解	打靶法	差分法
0	40	40	40
2	65.9518	65.9520	65.9698
4	93.7478	93.7481	93.7785
6	124.5036	124.5039	124.5382
8	159.4534	159.4538	159.4795
10	200	200	200

缩短步长可以减小误差。

有限差分法易于推广到更复杂的情形，一般优先使用差分法。

# MATLAB中的函数

函数	描述
ode23	二阶、三阶R-K方法，求解非刚性微分方程的低阶方法，较低精度 $O(h^3)$ 场合
ode45	四阶、五阶R-K方法，求解非刚性微分方程的中阶方法，大多数场合的首选算法，中等精度 $O(h^6)$
ode113	非刚性微分方程，可变阶
ode23s	解刚性微分方程的低阶方法
ode15s	刚性微分方程，可变阶
dsolve	符号解



# 第六章 总结——常微分方程数值解法

方法	初始值	是否需要迭代	全局误差	步长调整的难易	编程难度	备注
欧拉方法	1	否	$O(h)$	容易	容易	非常适合快速估计
修恩法	1	是	$O(h^2)$	容易	适中	
中点公式	1	否	$O(h^2)$	容易	适中	
二阶罗森方法	1	否	$O(h^2)$	容易	适中	截断误差最小的二阶R-K方法
四阶龙格-库塔法	1	否	$O(h^4)$	容易	适中	应用广泛
四阶Adams方法	4	是	$O(h^5)$	困难	中到难	