系统仿真算法

—— 数值积分法

燕山大学

电气工程学院

电工电能新技术研究室

2018/05

目 录

Contents Page

YANSHAN UNIVERSITY
电工电能新技术研究室
Electr. Eng. & Power Convers. LAB

研究室首页 研究室概况 导师简介 学生信息 辅助教学 研究室动态 研究室资料 多彩生活

辅助教学

当前位置： 辅助教学 > 教学课件

教学课件

教学课件

教学解答

共 5 条信息 当前页次：1/1页

序号	标 题	作者	人气	日期
5.	电力电子电路仿真——第二次作业 new	张迪	0	2018-06-08
4.	电力电子电路仿真——PDF课件	张迪	287	2018-05-14
3.	电力电子电路仿真——matlab例程	张迪	606	2018-05-07
2.	MATLAB 2015b下载链接+详细安装步骤	张迪	365	2018-04-28

http://iee.ysu.edu.cn/~eepc/board.php?sub=3

目 录

Contents Page

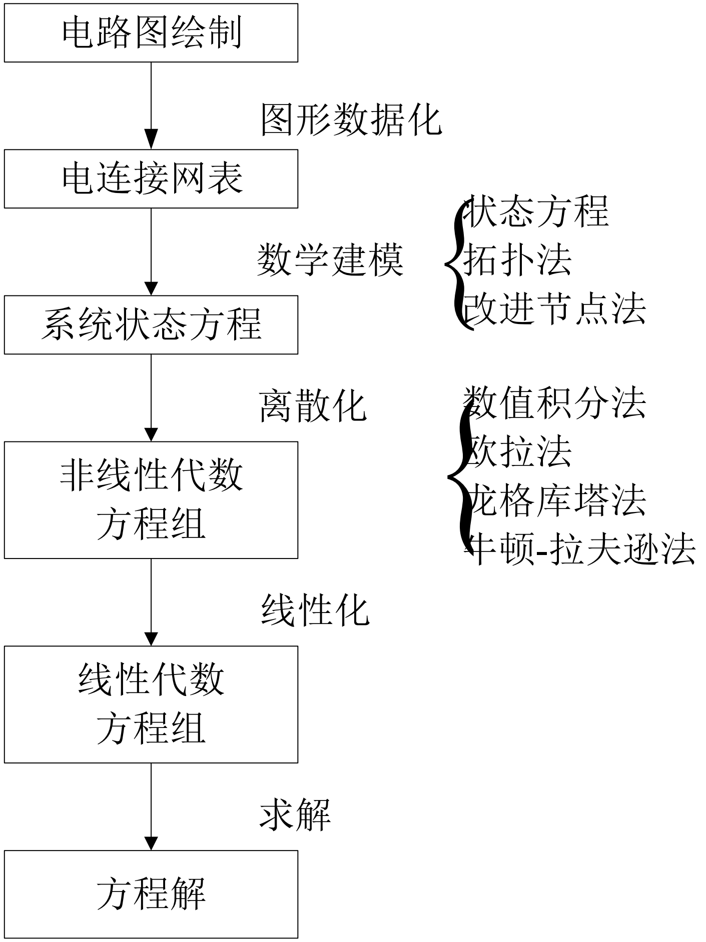
目 录

Contents Page

一、 数值积分法的基本原理

MATLAB

- 在数学仿真中，从一个实际系统抽象出数学模型只是第一步，这一步将实际系统变成了数学模型，可以称之为系统建模或系统辨识，这是第一次模型化过程。这次模型化所得到的只是数学方程式，必须使用一定的仿真工具才能求解。将已获得的数学模型变成能在一定仿真工具中运算求解的仿真模型，这是第二次模型化过程。

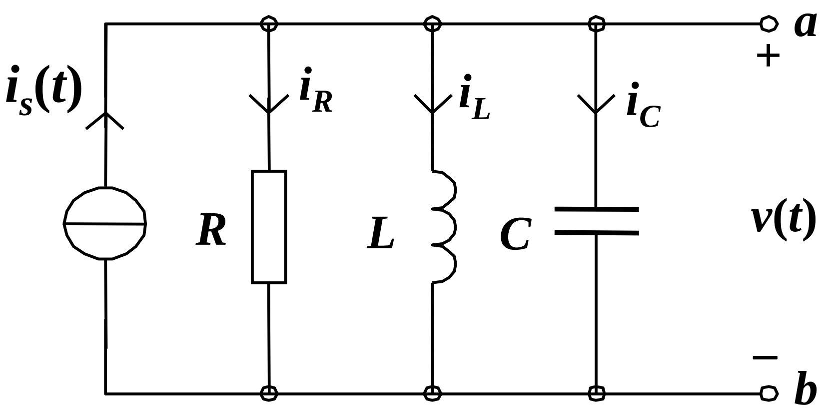


一、 数值积分法的基本原理

MATLAB

■ 系统的动态特性通常由高阶微分方程或状态方程来描述。

◆ 求并联电路的端电压$v(t)$与激励$i_s(t)$间的关系



➢ 电阻：$i_R(t)=\dfrac{1}{R}v(t)$

➢ 电感：$i_L(t)=\dfrac{1}{L}\int_{-\infty}^{t}v(\tau)\mathrm{d}\tau$

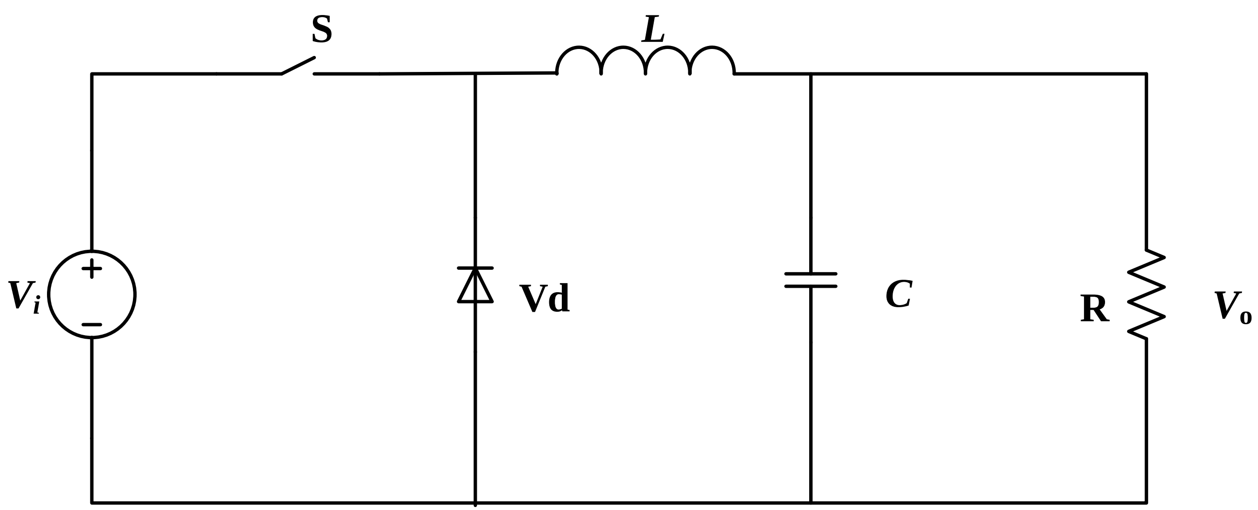
➢ 电容：$i_C(t)=C\dfrac{\mathrm{d}v(t)}{\mathrm{d}t}$

➢ 根据KCL：$i_R(t)+i_L(t)+i_C(t)=i_s(t)$

➢ 最后可得：$C\dfrac{\mathrm{d}^2v(t)}{\mathrm{d}t^2}+\dfrac{1}{R}\dfrac{\mathrm{d}v(t)}{\mathrm{d}t}+\dfrac{1}{L}v(t)=\dfrac{\mathrm{d}i_s(t)}{\mathrm{d}t}$ (二阶微分方程)

一、 数值积分法的基本原理

MATLAB常用交互界面



■ Buck电路小信号模型

$$LC\frac{\mathrm{d}^2\hat{v}_0(t)}{\mathrm{d}t^2}+\frac{L}{R}\frac{\mathrm{d}\hat{v}_0(t)}{\mathrm{d}t}+\hat{v}_0(t)=V_i\bullet\hat{\mathrm{d}}+D\bullet\hat{v}_i(t)$$

一、 数值积分法的基本原理

数值积分法

- 对于复杂的系统数学模型，求解其解析解很繁琐、很困难。大多数情况求不出其解析解，或者根本不存在解析解。
- 必须借助数值解法对连续系统进行仿真研究。
- 数值积分法
- 针对连续系统的微分方程形式。利用数值积分方法对常微分方程（组）建立离散化形式的数学模型——差分方程，并求其数值解 。

一、 数值积分法的基本原理

数值积分法的基本原理

■ 把被仿真系统表示成一阶微分方程组或状态方程的形式:

$$\begin{cases} \dot{Y} = F(t,\ Y) \\ Y(t_0) = Y_0 \end{cases}$$

- $t = t_0$， $Y_0 = Y(t_0)$
- $t = t_1$， $Y_1 = Y(t_1) = Y(t_0) + \int_{t_0}^{t_1} F(t,\ Y)\mathrm{d}t \Rightarrow Y(t_0) = Y(t_1) - \int_{t_0}^{t_1} F(t,\ Y)\mathrm{d}t$
- $t = t_2$， $Y_2 = Y(t_2) = Y(t_0) + \int_{t_0}^{t_2} F(t,\ Y)\mathrm{d}t = Y(t_1) + \int_{t_1}^{t_2} F(t,\ Y)\mathrm{d}t$
- $t = t_\mathrm{n}$ $Y_n = Y(t_n) = Y(t_0) + \int_{t_0}^{t_n} F(t,\ Y)\mathrm{d}t \Rightarrow Y(t_0) = Y(t_n) - \int_{t_0}^{t_n} F(t,\ Y)\mathrm{d}t$
- $t = t_{\mathrm{n+1}}$ $Y_{n+1} = Y(t_{n+1}) = Y(t_0) + \int_{t_0}^{t_{n+1}} F(t,\ Y)\mathrm{d}t = Y(t_n) + \int_{t_n}^{t_{n+1}} F(t,\ Y)\mathrm{d}t$

一、 数值积分法的基本原理

数值积分法的基本原理

■ 把被仿真系统表示成一阶微分方程组或状态方程的形式：

$$\begin{cases} \dot{Y} = F(t, \ Y) \\ Y(t_0) = Y_0 \end{cases}$$

■ 则上式在$t = t_0 , t_1 ,\dots , t_n , t_{n+1}$处的形式上的连续解为

初值

$$Y_{n+1} = Y(t_{n+1}) = Y(t_0) + \int_{t_0}^{t_{n+1}} F(t, \ Y)\mathrm{d}t = Y(t_n) + \boxed{\int_{t_n}^{t_{n+1}} F(t, \ Y)\mathrm{d}t} = \boxed{Y_n} + \boxed{Q_n}$$

■ 通过上式的演化，连续系统的数值解就转化为相邻两个时间点上的数值积分问题。

数值积分增量

一、 数值积分法的基本原理

数值积分法的基本原理

■ 所谓数值解法，就是寻求初值问题式（3-1）的真解在一系列离散点$t_1 < t_2 <\ldots< t_n <\ldots$上的近似数值解 $Y_1, Y_2 ,\ldots , Y_n , \ldots$

$$Y_{n+1} = Y(t_{n+1}) = Y(t_0) + \int_{t_0}^{t_{n+1}} F(t, Y)\mathrm{d}t = Y(t_n) + \int_{t_n}^{t_{n+1}} F(t, Y)\mathrm{d}t = Y_n + Q_n$$

■ 相邻两个时间离散点的间隔$h_n = t_{n+1} - t_n$，称为计算步距或步长，通常取$h_n = h$为定值。

■ **数值积分法主要问题归结为对函数$F(t, Y)$的数值积分问题。**

一、 数值积分法的基本原理

数值积分法的基本原理

- 数值解的一般求法
  - 把连续变量问题用数值积分法转化为离散的差分方程初值问题。
  - 根据已知初始条件y_0，逐步的递推计算后续时刻的数值解y_i。
- 采用步进式逐步进行递推计算，递推算法不同，数值积分方法就不同。
- 不同的数值积分公式，对系统的求解精度、速度和稳定性均有不同的影响。

目 录

Contents Page

二、 Euler 法

Euler法基本原理

- 对于初值问题: $\begin{cases} \dot{y} = f(t, y) \\ y(0) = y_0 \end{cases}$
- 假定$y(t)$为其解析解，将展成泰勒级数:

$$f(x) = \frac{f(x_0)}{0!} + \frac{f'(x_0)}{1!}(x - x_0) + \frac{f''(x_0)}{2!}(x - x_0)^2 + \ldots + \frac{f^{(n)}(x_0)}{n!}(x - x_0)^n + R_n(x)$$

$$\begin{cases} y(t_{n+1}) = y(t_n) + \dot{y}(t_n)(t_{n+1} - t_n) + \ddot{y}(t_n)\dfrac{(t_{n+1} - t_n)^2}{2} + \ldots \\ t_{n+1} - t_n = h \end{cases}$$

- 根据上式可得: $y(t+h) = y(t) + \dot{y}(t)h + \ddot{y}(t_n)\dfrac{h^2}{2} + \ldots$
- 当h充分小时，则有: $\boxed{y(t+h) \approx y(t) + hf(t, y)}$

二、 Euler 法

Euler法应用

- 写成差分形式:

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

- 只要给定初始条件y_0及步长h，就可根据f (t_0, y_0)算出y_1，再由y_1，算出y_2，如此递推算出y_3，y_4，y_5…，y_n。
- 例3-1：设系统方程为：$\dot{y} + y^2 = 0$，$y(0)=1$，取步长 $h = 0.1$，$0 \leqslant t \leqslant 1$，试用Euler法求其数值解。

目 录

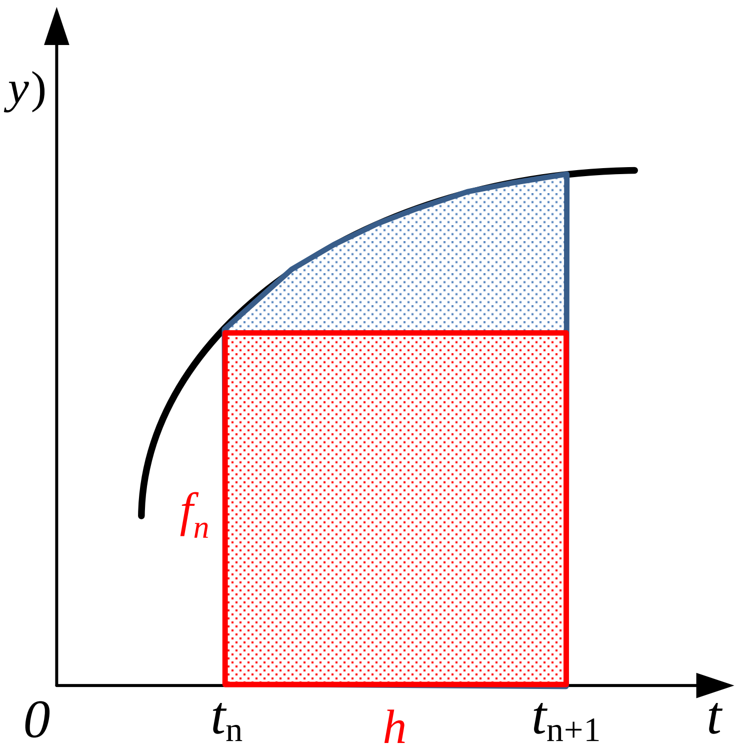
Contents Page

三、改进 Euler 法

Euler存在的问题

$$Y_{n+1}=Y(t_{n+1})=Y(t_0)+\int_{t_0}^{t_{n+1}} F(t,\ Y)\mathrm{d}t=Y(t_n)+\boxed{\int_{t_n}^{t_{n+1}} F(t,\ Y)\mathrm{d}t}$$

- 数值积分的原理是用上一时刻的值y_n加上步长时间h内导数函数f $\dot{y}=f(t,y)$ $(t,\ y)$的定积分的近似解，来得到本时刻的值y_{n+1}。
- 这个导数函数定积分的近似方法，就会影响到计算的精度。
- 根据定积分的几何定义，积分值应该为右图中曲边梯形的面积。
- 而欧拉法是用矩形面积来代替曲边梯形的面积，误差较大。



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

三、改进 Euler 法

改进Euler法

$$Y_{n+1}=Y(t_{n+1})=Y(t_0)+\int_{t_0}^{t_{n+1}}F(t,\ Y)\mathrm{d}t=Y(t_n)+\boxed{\int_{t_n}^{t_{n+1}}F(t,\ Y)\mathrm{d}t}$$

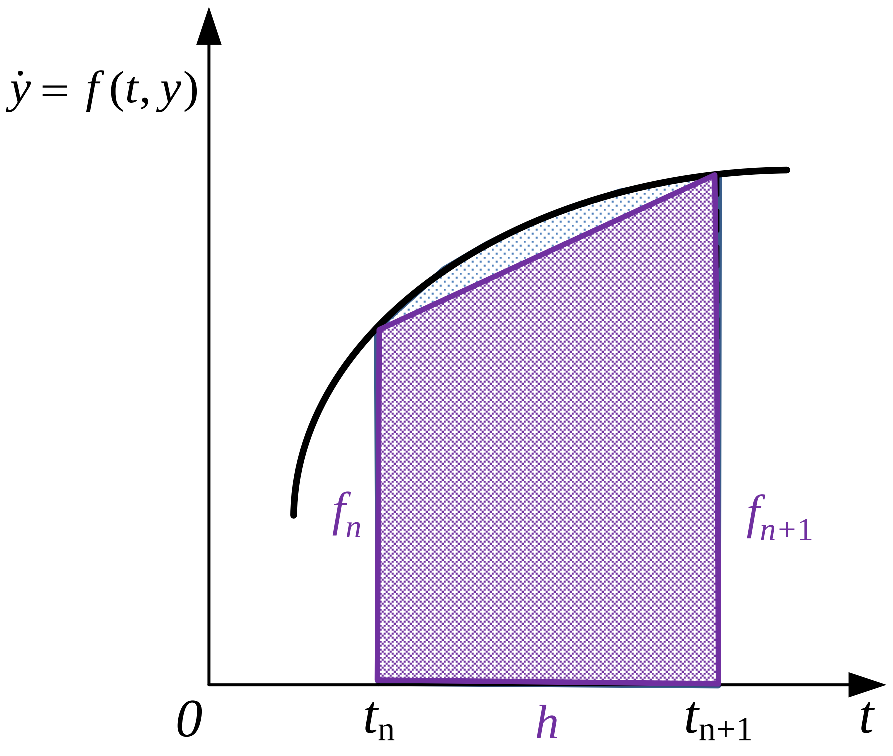
- 改用直边梯形面积代替每一个小区间的曲边梯形的面积，则可提高精度。

- 曲边梯形的面积为

$$S_1=\int_{t_n}^{t_{n+1}}f(t,y)dt=y(t_{n+1})-y(t_n)$$

- 直边梯形的面积为

$$S_2=\frac{1}{2}h[f(t_n,y_n)+f(t_{n+1},y_{n+1})]$$



三、改进 Euler 法

改进Euler法

$$Y_{n+1}=Y(t_{n+1})=Y(t_0)+\int_{t_0}^{t_{n+1}}F(t,\ Y)\mathrm{d}t=Y(t_n)+\boxed{\int_{t_n}^{t_{n+1}}F(t,\ Y)\mathrm{d}t}$$

- 当h比较小时，以直边梯形面积取代曲边梯形的面积，可得 $\dot{y}=f(t,y)$

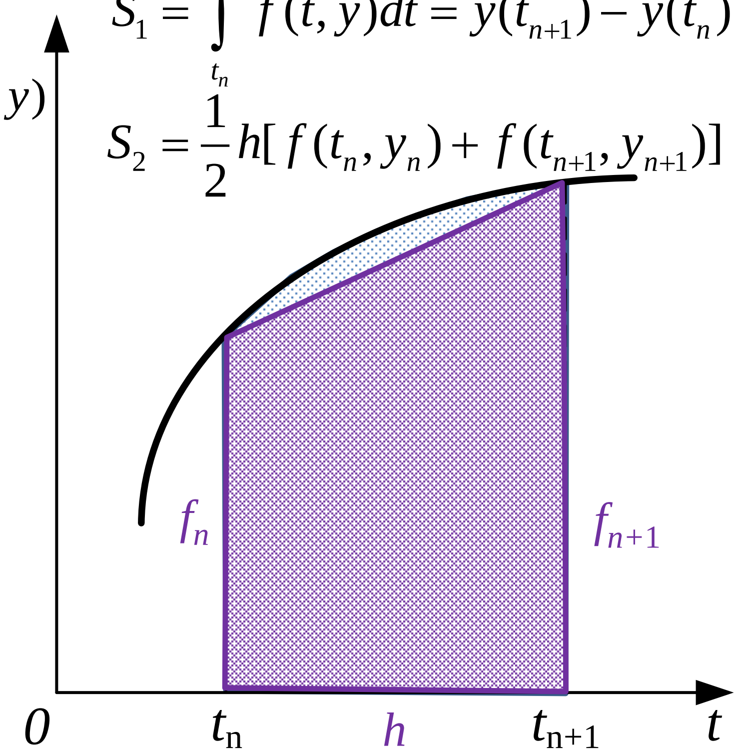
$$y(t_{n+1})-y(t_n)=\frac{1}{2}h[f(t_n,y_n)+f(t_{n+1},y_{n+1})]$$

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

$$=y_n+\frac{1}{2}h[f_n+f_{n+1}]$$

$$S_1=\int_{t_n}^{t_{n+1}}f(t,y)dt=y(t_{n+1})-y(t_n)$$

$$S_2=\frac{1}{2}h[f(t_n,y_n)+f(t_{n+1},y_{n+1})]$$



- 也称为梯形公式或二阶隐式Adams公式，

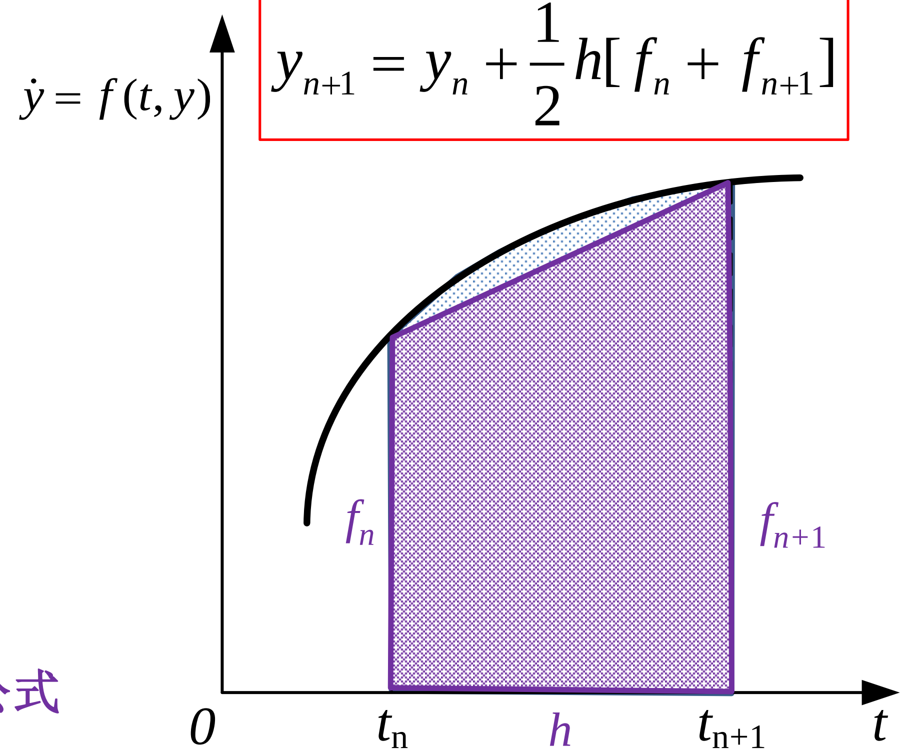
三、改进 Euler 法

改进Euler法

$$Y_{n+1}=Y(t_{n+1})=Y(t_0)+\int_{t_0}^{t_{n+1}}F(t,\ Y)\mathrm{d}t=Y(t_n)+\boxed{\int_{t_n}^{t_{n+1}}F(t,\ Y)\mathrm{d}t}$$

- 公式右端隐含有待求量，故梯形法不能自行起步运算，而需有其他算法的帮助。
- 为了提高计算精度常常需要迭代运算。为减小计算量，常常迭代一次就求得近似解。这样就可以得到改进的Euler公式。

$$\boxed{y_{n+1}=y_n+\frac{1}{2}h[f_n+f_{n+1}]}$$



$$\begin{cases} y_{n+1}^{p}=y_n+hf(t_n,y_n) & \text{预估公式} \\ y_{n+1}^{c}=y_n+\dfrac{1}{2}h[f(t_n,y_n)+f(t_{n+1},y_{n+1}^{p})] & \text{校正公式} \end{cases}$$

三、改进 Euler 法

改进Euler法

$$\begin{cases} y_{n+1}^{p} = y_n + hf(t_n, y_n) & \text{预估公式} \\ y_{n+1}^{c} = y_n + \dfrac{1}{2}h[f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{p})] & \text{校正公式} \end{cases}$$

- 修正过程：
- 先用欧拉法，估算y_{n+1}

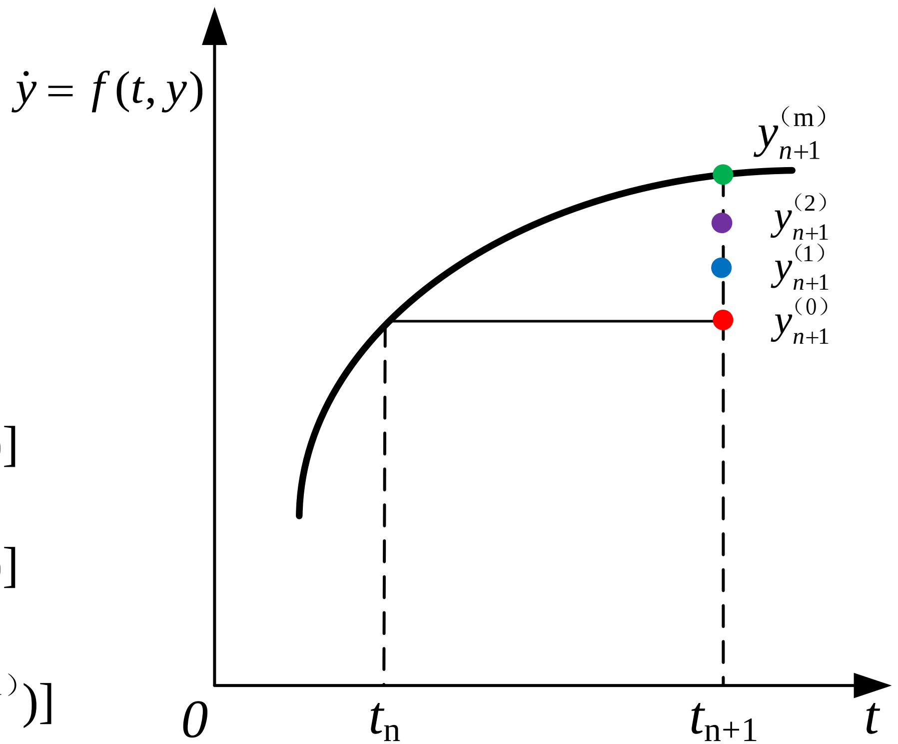
$$y_{n+1}^{(0)} = y_n + hf(t_n, y_n)$$

- 校正过程：

$$y_{n+1}^{(1)} = y_n + \frac{1}{2}h[f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{(0)})]$$

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

$$y_{n+1}^{(m)} = y_n + \frac{1}{2}h[f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{(m-1)})]$$



- 当 $y_{n+1}^{(m)}$与 $y_{n+1}^{(m-1)}$近似相等时，迭代结束，此时我们认为 $y_{n+1}=y_{n+1}^{(m)}$

三、改进 Euler 法

改进Euler法

$$\begin{cases} y_{n+1}^{p} = y_n + hf(t_n, y_n) \\ y_{n+1}^{c} = y_n + \dfrac{1}{2}h[f(t_n, y_n) + f(t_{n+1}, y_{n+1}^{p})] \end{cases}$$

- 改进Euler法实际上是采用了$[t_n, y(t_n)]$ 和 $[t_{n+1}, y(t_{n+1})]$ 两点斜率平均值的结果。
- 将这种思想引申如果在每个积分步中多取几个点（如取r个点），分别求出其斜率 k_1，k_2，…，k_r，然后取不同的权值，则公式为:

$$k = \omega_1 k_1 + \omega_2 k_2 + \ldots + \omega_r k_r$$

即可得出一系列龙格 — 库塔（Runge-Kutta）积分法公式。

目 录

Contents Page

四、数值积分法的基本概念

基本概念

- 算法自启动
- 单步法与多步法
- 显式与隐式
- 截断误差
- 舍入误差
- 初始误差

四、数值积分法的基本概念

基本概念

- 算法自启动
  - 无需其他算法辅助，只要知道方程的初值$y(t_0)$，便可利用算法公式求得后续一系列点的值。
  - Euler法与改进Euler法都能自启动吗?
- 单步法与多步法
  - 单步法可根据当前时刻的值来求得下一时刻的值，无需其他时刻的任何信息，可以自启动。
  - 多步法不仅要知道当前时刻t_n的数据，还要知道过去时刻t_{n-1}，t_{n-2}，……处的数据，无法自启动。

四、数值积分法的基本概念

基本概念

- 显式与隐式
  - 在递推公式中只用到当前时刻t_n及其之前的若干时刻的值，而不用t_{n+1}以及其后时刻的数值，称为显式算法。
  - 如果计算y_{n+1}的递推公式含有未知量y_{n+1}，则称为隐式算法。
  - 隐式算法需要借助显式公式估计初值，然后再用隐式公式进行迭代运算，预估——校正公式，如改进型Euler法。
  - 隐式算法不能自启动。

四、数值积分法的基本概念

基本概念

■ 截断误差

$$y(t_n+h)=y(t_n)+h\dot{y}(t_n)+\ldots+\frac{h^r}{r!}y^{(r)}(t_n)+O(h^{r+1})$$

- Euler法只保留了展开式前两项。
- 由这种方法单独进行一步运算所引起的附加误差，称为局部截断误差。
- 是由该方法给出的近似解与微分方程精确解之间的差。
- 一般若截断误差为$O(h^{r+1})$，则称该方法为r阶。
- 方法的阶次可作为衡量方法精度的一个重要标志
- Euler法为一阶精度，改进Euler法为二阶精度。

四、数值积分法的基本概念

基本概念

- 舍入误差
  - 由计算机字长引起的误差 。
  - 舍入误差与步长h成反比，h越小，计算次数越多，舍入误差越大。
- 初始误差
  - 给定初始值与真实值之间的差异。

目 录

Contents Page

五、龙格—库塔法

龙格库塔法的基本思想

- 将Taylor级数展开式多取几项后截断，能提高截断误差的阶次，从而提高精度。

$$y(t_n+h)=y(t_n)+h\dot{y}(t_n)+\ldots+\frac{h^r}{r!}y^{(r)}(t_n)+O(h^{r+1})$$

- 直接采用Taylor级数展开方法需要计算函数在某一点的高阶导数，使用起来不方便。提出间接利用Taylor级数展开式的方法。

- 是用几个点上的的一阶导函数值的线性组合来近似代替在某一点的各阶导数，然后用Taylor级数展开式确定线性组合中各加权系数。这就是Runge-Kutta（简称RK）法的基本思想。

五、龙格—库塔法

一阶和二阶RK公式

- 一阶RK公式为： $y(t+h)=y(t)+hf(t, y)$
  - 就是Euler公式，也就是说，Euler公式是RK公式的特例
- 二阶RK公式为每步取两个斜率加权。
  - 第一斜率： $k_1=f(t_n,y_n)=f_n$
  - 第二斜率： $k_2=f(t_n+\alpha h,y_n+\beta hk_1)$
  - 加权后递推公式： $y_{n+1}=y_n+hk_\omega=y_n+h(\omega_1 k_1+\omega_2 k_2)$
  - 经过与泰勒级数对比可确定四个参数$\alpha\ \beta\ \omega_1\ \omega_2$,可得二阶RK法的计算公式

五、龙格—库塔法

二阶RK公式

- 二阶RK法的计算公式

- 当$\alpha=\beta=0.5$，

  $\omega_1=0$， $\omega_2=1$时

$$
\begin{cases}
k_1 = f(t_n, y_n) \\
k_2 = f(t_n + \frac{h}{2}, y_n + \frac{1}{2}hk_1) \\
y_{n+1} = y_n + hf(t_n + \frac{h}{2}, y_n + \frac{1}{2}hk_1)
\end{cases}
$$

- 当$\alpha=\beta=1$，

  $\omega_1=\omega_2=0.5$时

$$
\begin{cases}
y_{n+1} = y_n + \frac{h}{2}(k_1 + k_2) \\
k_1 = f(t_n,\ y_n) \\
k_2 = f(t_n + h,\ y_n + hk_1)
\end{cases}
$$

五、龙格—库塔法

三阶和四阶RK公式

■ 三阶RK公式

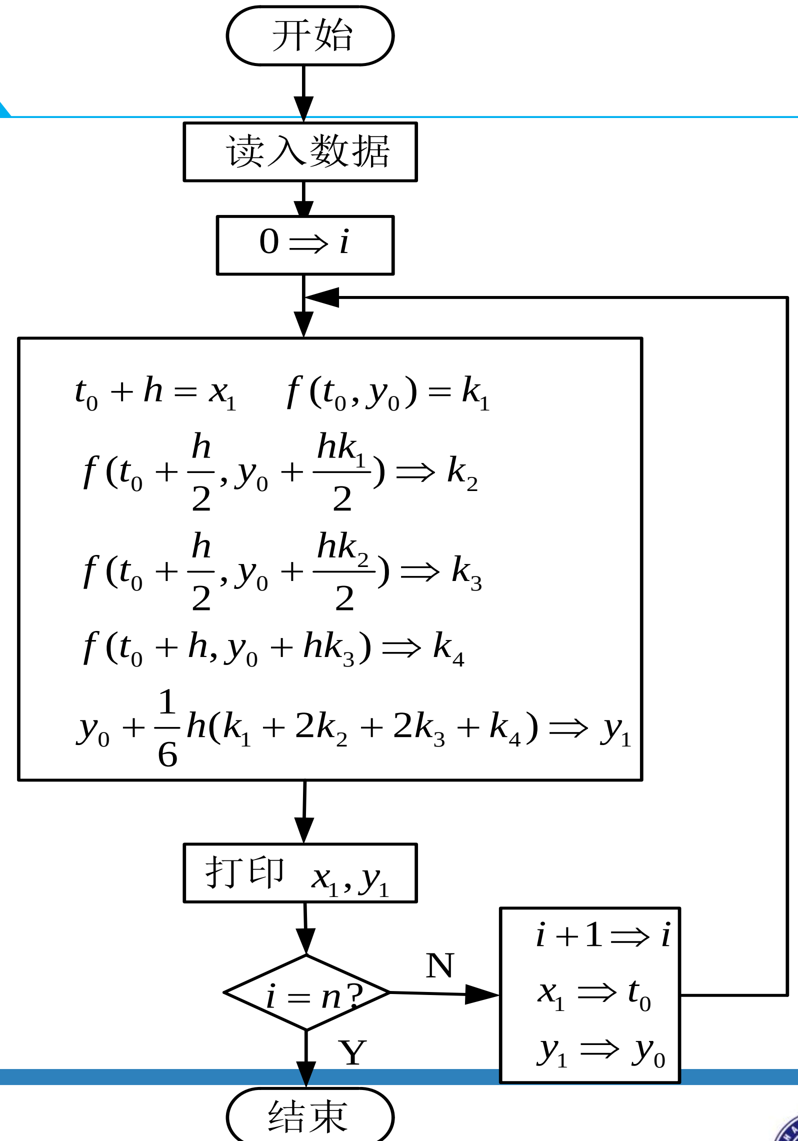
$$
\begin{cases}
y_{n+1} = y_n + \dfrac{h}{4}(k_1 + 3k_3) \\
k_1 = f(t_n,\ y_n) \\
k_2 = f(t_n + \dfrac{h}{3},\ y_n + \dfrac{h}{3}k_1) \\
k_3 = f(t_n + \dfrac{2h}{3},\ y_n + \dfrac{2h}{3}k_2)
\end{cases}
$$

■ 四阶RK公式

$$
\begin{cases}
y_{n+1} = y_n + \dfrac{h}{6}(k_1 + 2k_2 + 2k_3 + k_4) \\
k_1 = f(t_n,\ y_n) \\
k_2 = f(t_n + \dfrac{h}{2},\ y_n + \dfrac{h}{2}k_1) \\
k_3 = f(t_n + \dfrac{h}{2},\ y_n + \dfrac{h}{2}k_2) \\
k_4 = f(t_n + h,\ y_n + hk_3)
\end{cases}
$$

五、龙格—库塔法

四阶RK程序框图



五、龙格—库塔法

例3-2

- 已知系统方程$\ddot{y}+0.5\dot{y}-2y=0$，$\dot{y}(0)=0$，$y(0)=1$，取步长$h=0.1$，试用RK4公式计算$t=0.1$，$t=0.2$时y的值。

$$\begin{cases} y_{n+1}=y_n+\frac{h}{6}(k_1+2k_2+2k_3+k_4) \\ k_1=f(t_n,\ y_n) \\ k_2=f(t_n+\frac{h}{2},\ y_n+\frac{h}{2}k_1) \\ k_3=f(t_n+\frac{h}{2},\ y_n+\frac{h}{2}k_2) \\ k_4=f(t_n+h,\ y_n+hk_3) \end{cases}$$

➢ 转换为下列方程组:

$$\begin{cases} \dot{y}_1=y_2, y_1(0)=1 \\ \dot{y}_2=2y_1-0.5y_2, y_2(0)=0 \end{cases}$$

y_1为原函数
y_2为原函数的导数

- $t=0.1$,n=0时

k_{11}为y_1的第1个斜率
k_{21}为y_2的第1个斜率

$$\begin{cases} k_{11}=f_1(t_0,y_{10},y_{20})=y_{20}=0 \\ k_{21}=f_2(t_0,y_{10},y_{20})=2y_{10}-0.5y_{20}=2 \end{cases}$$

即y_1的在t_0时刻的导数值

即y_2的在t_0时刻的导数值

k_{12}为y_1的第2个斜率
k_{22}为y_2的第2个斜率

$$\begin{cases} k_{12}=f_1(t_0+\frac{h}{2},y_{10}+\frac{h}{2}k_{11},y_{20}+\frac{h}{2}k_{21})=y_{20}+\frac{h}{2}k_{21}=0.1 \\ k_{22}=f_2(t_0+\frac{h}{2},y_{10}+\frac{h}{2}k_{11},y_{20}+\frac{h}{2}k_{21})=2(y_{10}+\frac{h}{2}k_{11})-0.5(y_{20}+\frac{h}{2}k_{21})=1.95 \end{cases}$$

即y_1的在$t_0+h/2$时刻的导数值

即y_2的在$t_0+h/2$时刻的导数值

五、龙格—库塔法

数值积分法的选择

■ 精度、速度和可靠性

➢ 一般应该考虑以下因素：方法本身的复杂程度、计算量和误差的大小、步长和易调整性以及系统本身的刚性程度，要特别注意稳定性的要求。

■ 精度要求

■ 影响精度的因素包括截断误差、舍入误差、初始误差等。

■ 一般来说，当步长取定时，算法阶次越高，截断误差越小；同一算法，步距越小，截断误差越小；

五、龙格—库塔法

数值积分法的选择

■ 精度要求

- ■ 算法阶次取定后，多步法精度比单步法高，隐式公式的精度比显式公式高。
- ■ 当要求高精度仿真时，可采用高阶的隐式多步法，并取较小的步长。但步长不能太小，因为步长太小会增加迭代次数，增加计算量，同时也会加大舍入误差和累积误差。

五、龙格—库塔法

数值积分法的选择

■ 计算速度

■ 计算速度主要取决于每步积分运算所花费的时间以及积分的总次数。每步积分的运算量同具体的积分方法有关，它主要取决于导函数的复杂程度和每步积分应计算导函数的次数。

五、龙格—库塔法

数值积分法的选择

■ 计算速度

■ 计算速度主要取决于每步积分运算所花费的时间以及积分的总次数。每步积分的运算量同具体的积分方法有关，它主要取决于导函数的复杂程度和每步积分应计算导函数的次数。

■ 数值解的稳定性

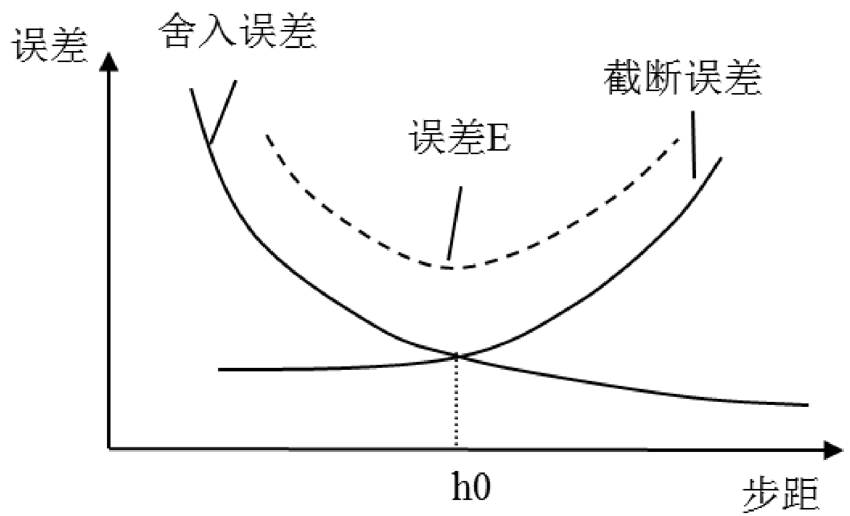
■ 保证数值解的稳定性是进行数字仿真的先决条件。否则计算将失去意义，导致仿真失败。

五、龙格—库塔法

数值积分法的选择

■ 积分步长的确定

- ■ 步长太大，会导致较大的截断误差，甚至出现数值解不稳定现象。
- ■ 步长太小，势必会增加计算次数，造成舍入误差的积累，使总误差加大。



五、龙格—库塔法

数值积分法的选择

■ 数值积分法稳定性分析

■ 对于给定的步长，如果计算结果对初始误差或计算误差不敏感，就说该算法是稳定的；否则就说该算法是不稳定的。

Thank you！

Question and Answer？