第7章 常微分方程初值问题的数值解法

7.1 引言

7.2 简单的数值方法与基本概念

7.3 龙格-库塔方法

7.4 稳定性

7.5* 线性多步法

7.1 引言

一般地，微分方程形式为

$$f(y^{(n)}, y^{(n-1)}, \cdots, y', y, x) = 0$$

如果 $y = \varphi(x)$ 满足方程成立，则称 $y = \varphi(x)$ 是方程的解.

微分方程的解析解方法

- 格式：

  y=dsolve(f_1, f_2, …, f_m)

- 格式：指明自变量

  y=dsolve(f_1, f_2, …, f_m ,'x')

f_i即可以描述微分方程，又可描述初始条件或边界条件。如：

描述微分方程时

描述条件时 $y^{(4)}(t)=7 \Rightarrow D4y=7$

$\ddot{y}(2)=3 \Rightarrow D2y(2)=3$

例：输入信号为 $u(t) = e^{-5t}\cos(2t+1)+5$，

求该微分方程的通解

$$y^{(4)}(t)+10y'''(t)+35y''(t)+50y'(t)+24y(t) = 5u''(t)+4u'(t)+2u(t)$$

*step*1:

```
>> syms t; u=exp(-5*t)*cos(2*t+1)+5;
>> uu=5*diff(u,t,2)+4*diff(u,t)+2*u
uu =
87*exp(-5*t)*cos(2*t+1)+92*exp(-5*t)*sin(2*t+1)+10
```

*step*2:

```
>> syms t y;
>> y=dsolve(['D4y+10*D3y+35*D2y+50*Dy+24*y=',...
    '87*exp(-5*t)*cos(2*t+1)+92*exp(-5*t)*sin(2*t+1)+10'])
```

$$y(t)=\frac{5}{12}-\frac{343}{520}\mathrm{e}^{-5t}\cos(2t+1)-\frac{547}{520}\mathrm{e}^{-5t}\sin(2t+1)+$$
$$C_1\mathrm{e}^{-4t}+C_2\mathrm{e}^{-3t}+C_3\mathrm{e}^{-2t}+C_4\mathrm{e}^{-t}$$

假设已知 $y(0)=3, y'(0)=2, y''(0)=y'''(0)=0$

```
>> y=dsolve(['D4y+10*D3y+35*D2y+50*Dy+24*y=',...
   '87*exp(-5*t)*cos(2*t+1)+92*exp(-5*t)*sin(2*t+1) ...
  +10'], 'y(0)=3',  'Dy(0)=2', 'D2y(0)=0', 'D3y(0)=0')
```

$$y(t)=\frac{5}{12}-\frac{343}{520}\mathrm{e}^{-5t}\cos(2t+1)-\frac{547}{520}\mathrm{e}^{-5t}\sin(2t+1)+$$
$$\left(-\frac{445}{26}\cos 1-\frac{51}{13}\sin 1-\frac{69}{2}\right)\mathrm{e}^{-2t}+\left(-\frac{271}{30}\cos 1+\frac{41}{15}\sin 1-\frac{25}{4}\right)\mathrm{e}^{-4t}+$$
$$\left(\frac{179}{8}\cos 1+\frac{5}{8}\sin 1+\frac{73}{3}\right)\mathrm{e}^{-3t}+\left(\frac{133}{30}\cos 1+\frac{97}{60}\sin 1+19\right)\mathrm{e}^{-t}$$

- 例：求 $x'(t) = x(t)(1 - x^2(t))$ 的解析解

```
>> syms t x
>> x=dsolve('Dx=x*(1-x^2)')
x =
[  1/(1+exp(-2*t)*C1)^(1/2)]
[ -1/(1+exp(-2*t)*C1)^(1/2)]
```

改变原微分方程： $x'(t) = x(t)(1 - x^2(t)) + 1$

```
>> syms t x;  x=dsolve('Dx=x*(1-x^2)+1')
Warning: Explicit solution could not be found; implicit solution
    returned.
> In D:\MATLAB6p5\toolbox\symbolic\dsolve.m at line 292
x =
t-Int(1/(a-a^3+1),a=``..x)+C1=0
```

故只有部分非线性微分方程有解析解。

虽然求解微分方程有许多解析方法，但解析方法只能够求解一些特殊类型的方程，从实际意义上来讲，我们更关心的是某些特定的自变量在一系列离散点上的近似值。

一组近似解称为微分方程在该范围内的数值解，寻找数值解的过程称为微分方程的数值解法。

本章着重讨论一阶常微分方程初值问题

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

在区间$[a,b]$上的数值解法。

此类问题多数不能求出解析解或近似解，只能求助于数值解法，给出离散时间点上的近似解。在数值求解之前，先考虑解的存在唯一性.

如果存在实数 L>0，使得

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

则称 $f(x,y)$关于 y 满足 Lipschitz 条件，L 称 Lipschitz 系数.

只要$\left|\dfrac{\partial f(x,y)}{\partial y}\right|\le L$，显然 f 关于 y 满足 Lipschitz 条件.

如果 $f(x,y)$连续，且关于 y 满足 Lipschitz 条件，则初值问题(1)存在唯一的连续可微解.

常微分方程数值解法的思路：

- 对求解区间进行剖分
- 把微分方程离散成节点处的近似方程
- 结合定解条件，求出未知函数在节点处的近似值

数值解法得到的近似解是一个离散形式的函数表。

数值解法：就是寻求解$y(x)$在一系列离散节点

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

上的近似值 $y_1, y_2, \cdots, y_n, y_{n+1}, \cdots$。相邻两节点间的间距 $h_n = x_{n+1} - x_n$称为步长。假定$h_n = h$为定数，这时节点为$x_n = x_0 + nh, n = 0, 1, 2, \cdots$

初值问题数值解法的基本特点：它们都采用“步进式”，即求解过程顺着节点排列的次序一步一步地向前推进。描述这类算法，只要给出已知信息$y_n, y_{n-1}, y_{n-2}, \cdots$计算$y_{n+1}$的递推公式。

本章主要考查一阶常微分方程的数值解法.

一阶常微分方程初值问题的常用解法

- 单步法：

  Euler法、后退Euler法

  梯形法、改进Euler法

  Runge-Kutta法

- 多步法*

  Adams法

单步法： 计算 y_{n+1} 时只用到 x_{n+1}, x_n, y_n.

多步法： 计算 y_{n+1} 时除了用到 x_{n+1}, x_n, y_n，还需要

$x_{n-p}, y_{n-p}\ (p = 1, 2, \ldots, k; k > 0)$.

7.2 简单的数值方法与基本概念

1、Euler 法

由数值微分的向前差分公式可以得到(1)中 y' 的数值计算问题:

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

因此 $y(x_{n+1}) \approx y(x_n)+hy'(x_n)$，再由(1)有 $y'(x_n)=f(x_n,y(x_n))$，再由 $y_n \approx y(x_n), y_{n+1} \approx y(x_{n+1})$, 得到

$$y_{n+1} = y_n + hf(x_n, y_n) \quad n=0,1,\ldots$$

Euler 公式

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

计算过程：

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

若初值y_0已知，则依公式可逐步算出

$$y_1 = y_0 + hf(x_0, y_0)$$

$$y_2 = y_1 + hf(x_1, y_1)$$

…

例： 利用欧拉法求解初值问题

$$\begin{cases} y' = y - \dfrac{2x}{y} & (0 \le x \le 1) \\ y(0) = 1 \end{cases}$$

步长 $h = 0.1$

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

解：求解本题的欧拉公式为

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

取步长$h = 0.1$, 计算结果见下表 $(x_0, y_0) = (0,1)$

x_n	y_n	$y(x_n)$		x_n	y_n	$y(x_n)$
0.1	1.1000	1.0954		0.6	1.5090	1.4832
0.2	1.1918	1.1832		0.7	1.5803	1.5492
0.3	1.2774	1.2649		0.8	1.6498	1.6125
0.4	1.3582	1.3416		0.9	1.7178	1.6733
0.5	1.4351	1.4142		1.0	1.7848	1.7321

2、后退Euler法

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

如果对方程(1)从x_n到x_{n+1}积分，得

$$y(x_{n+1}) = y(x_n) + \int_{x_n}^{x_{n+1}} f(t, y(t))dt \qquad (2)$$

右端积分用左矩形公式 $hf(x_n, y(x_n))$近似，再以y代替$y(x_n)$ y_{n+1}代替$y(x_{n+1})$也得到欧拉公式。

如果上式右端积分用右矩形公式$hf(x_{n+1}, y(x_{n+1}))$近似，则得另一个公式

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

称为后退的欧拉法。

欧拉法是显式的，后退欧拉法是隐式的，隐式比显式有更好的数值稳定性. 用隐式欧拉法解上述例题。

再解上例

$$\begin{cases} y' = y - \dfrac{2x}{y} & (0 \le x \le 1) \\ y(0) = 1 \end{cases}$$

步长 $h = 0.1$

后退欧拉公式为

$$y_{n+1} = y_n + h\left(y_{n+1} - \frac{2x_{n+1}}{y_{n+1}}\right), \quad h = 0.1, (x_0, y_0) = (0,1)$$

3、梯形方法

等式(2)右端积分中若用梯形公式近似，则得到梯形方法。

用 y_n 代替 $y(x_n)$，用 y_{n+1} 代替 $y(x_{n+1})$

$$y_{n+1} = y_n + \frac{h}{2}\left[f(x_n, y_n) + f(x_{n+1}, y_{n+1}) \right] \qquad \text{(梯形方法)}$$

梯形方法是隐式方法，可以利用迭代法显式化求解。

例 用梯形方法求解初值问题

$$\begin{cases} y' = x + y, & 0 \le x \le 0.5 \\ y(0) = 1 \end{cases}$$

取步长 $h = 0.1$, 并与准确解 $y(x) = -x - 1 + 2e^x$ 比

解: 梯形方法计算公式为

$$y_{n+1} = y_n + \frac{1}{2} h \left[x_n + y_n + x_{n+1} + y_{n+1} \right]$$

解得

$$y_{n+1} = \frac{1}{1 - h/2} \left[(1 + \frac{h}{2}) y_n + \frac{h(x_n + x_{n+1})}{2} \right], \qquad n = 0, 1, 2, 3, 4$$

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

x_n	y_n	$\lvert y(x_n)-y_n\rvert$
0.1	1.110526	0.000184479
0.2	1.243213	0.000407779
0.3	1.400393	0.000676027
0.4	1.584645	0.000996210
0.5	1.798818	0.001376285

4、单步法的局部截断误差与阶

定义：对于求解式（1）的某差分格式，h 为步长，假设 $y_1,\cdots,y_n$ 是准确的，称

$$\varepsilon_{n+1}=y(x_{n+1})-y_{n+1}$$

为该差分格式的局部截断误差。当 $\varepsilon_{n+1}=O(h^{p+1})$ 时，称该差分格式具有 p 阶精度。

局部截断误差可以理解为用方法计算一步的误差 根据定义可以得到欧拉方法的局部截断误差。

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

欧拉法的代数精度：

由泰勒展开，有

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

由欧拉公式，有

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

$$\varepsilon_{n+1} = y(x_{n+1}) - y_{n+1} = \frac{h^2}{2} y''(x_n) + O(h^3) = O(h^2)$$

因此欧拉公式具有一阶精度.

后退欧拉法的代数精度：

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

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

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

$$= y'(x_n) + y''(x_n)h + O(h^2),$$

$$y_{n+1} = y_n + hf(x_{n+1}, y_{n+1}) = y(x_n) + hy'(x_{n+1})$$

$$= y(x_n) + h[y'(x_n) + y''(x_n)h + O(h^2)],$$

$$\varepsilon_{n+1} = y(x_{n+1}) - y_{n+1} = -\frac{h^2}{2} y''(x_n) + O(h^3) = O(h^2)$$

因此后退欧拉法也具有一阶精度.

类似得到梯形法的局部截断误差为

$$\varepsilon_{n+1} = y(x_{n+1}) - y_{n+1} = -\frac{h^3}{12} y'''(x_n) + O(h^4) = O(h^3)$$

因此梯形法具有2阶精度。

5、改进欧拉法 (预-校法)

先用欧拉公式求得一个初步的近似值$\bar{y}_{n+1}$称之为预测值。再用梯形公式校正一次，迭代一次得y_{n+1}称校正值，这样建立的预测-校正系统通常称为改进的欧拉公式：

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

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

或表示为平均化形式：

$$\begin{cases} y_p = y_n + hf(x_n, y_n) \\ y_c = y_n + hf(x_{n+1}, y_p) \\ y_{n+1} = \dfrac{1}{2}(y_p + y_c) \end{cases}$$

例 用改进的欧拉算法求解初值问题

$$\begin{cases} y' = y - \dfrac{2x}{y}, & 0 \le x \le 0.5 \\ y(0) = 1 \end{cases}$$

取步长 $h = 0.1$.

解：改进的欧拉公式为

$$\begin{cases} y_p = y_n + h\left(y_n - \dfrac{2x_n}{y_n}\right) \\ y_c = y_n + h\left(y_p - \dfrac{2x_{n+1}}{y_p}\right) \\ y_{n+1} = \dfrac{1}{2}(y_p + y_c) \end{cases}$$

$$\begin{cases} y_p = y_n + hf(x_n, y_n) \\ y_c = y_n + hf(x_{n+1}, y_p) \\ y_{n+1} = \dfrac{1}{2}(y_p + y_c) \end{cases}$$

x_n	y_n	$y(x_n)$		x_n	y_n	$y(x_n)$
0.1	1.0959	**1.0954**		0.6	1.4860	**1.4832**
0.2	1.1841	**1.1832**		0.7	1.5525	**1.5492**
0.3	1.2662	**1.2649**		0.8	1.6153	**1.6125**
0.4	1.3434	**1.3416**		0.9	1.6782	**1.6733**
0.5	1.4164	**1.4142**		1.0	1.7379	**1.7321**

可以类似证明改进欧拉法具有2阶精度。

改进欧拉法算法实现

（1）计算步骤

① 输入 x_0, x_1, h, N

② 使用以下改进欧拉法公式进行计算

$$\begin{cases} y_p = y_i + hf(x_i, y_i) \\ y_c = y_i + hf(x_{i+1}, y_p) \\ y_{i+1} = \dfrac{1}{2}(y_p + y_c) \end{cases}$$

③ 输出 x_1, y_1，并使 $x_1 \Rightarrow x_0, y_1 \Rightarrow y_0$

转到 ② 直至 $n > N$ 结束。

6、单步法的收敛性

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

定义. 若一种数值方法,对于固定的$x_n = x_0 + nh$, 当$h \to 0$时,有$y_n \to y(x_n)$,其中$y(x)$是(1)的准确解，则称该方法是收敛的。

定理：假设单步法具有 p 阶精度，即局部截断误差为$O(h^{p+1})$，且增量$\varphi(x, y, h)$关于 y 满足利普希茨条件：

$$|\varphi(x, y, h) - \varphi(x, \bar{y}, h)| \le L_\varphi |y - \bar{y}|$$

又设初值y_0是准确的，则整体截断误差为

$$y(x_n) - y_n = O(h^p)$$

设$f(x, y)$关于y满足利普希茨条件，则欧拉法，改进欧拉法都是收敛的。

MATLAB 中主要用 dsolve 求符号解析解，ode45,ode23,ode15s 求数值解。

s=dsolve('方程 1', '方程 2',…,'初始条件 1', '初始条件 2' …,'自变量')

用字符串方程表示，自变量缺省值为 t。导数用 D 表示，2 阶导数用 D2 表示，以此类推。S 返回解析解。在方程组情形，s 为一个符号结构。

[tout,yout]=ode45('yprime',[t0,tf],y0) 采用变步长四阶 Runge-Kutta 法和五阶 Runge-Kutta-Felhberg 法求数值解，yprime 是用以表示 f(t,y)的 M 文件名，t0 表示自变量的初始值，tf 表示自变量的终值，y0 表示初始向量值。输出向量 tout 表示节点 $(t_0,t_1, \cdots,t_n)^T$,输出矩阵 yout 表示数值解，每一列对应 y 的一个分量。若无输出参数，则自动作出图形。

ode45 是最常用的求解微分方程数值解的命令，对于刚性方程组不宜采用。ode23 与 ode45 类似，只是精度低一些。ode12s 用来求解刚性方程组，是用格式同 ode45。可以用 help dsolve, help ode45查阅有关这些命令的详细信息.

[T,Y]=solver(odefun,tspan,y0)

该函数表示在区间tspan=[t0,tf]上，用初始条件y0 求解显式常微分方程 $y'=f(t,y)$。

solver 为命令 ode45，ode23，ode113，ode15s，ode23s，ode23t，ode23tb 之一，这些命令各有特点。我们列表说明如下：

求解器	特点	说明
ode45	一步算法，4,5 阶 Runge-Kutta 方法累积截断误差 $(\Delta x)^5$	大部分场合的首选算法
ode23	一步算法，2,3 阶 Runge-Kutta 方法累积截断误差 $(\Delta x)^5$	使用于精度较低的情形
ode113	多步法，Adams算法，高低精度均可达到 $10^{-3}\sim10^{-6}$	计算时间比 ode45 短
ode23t	采用梯形算法	适度刚性情形
ode15s	多步法，Gear's 反向数值积分，精度中等	若 ode45 失效时，可尝试使用
ode23s	一步法，2 阶 Rosebrock 算法，低精度。	当精度较低时，计算时间比 ode15s 短

odefun为显式常微分方程 $y'=f(t,y)$ 中的 $f(t,y)$

tspan为求解区间，要获得问题在其他指定点 $t_0,t_1,t_2,\cdots$ 上的解，则令 $tspan=[t_0,t_1,t_2,\cdots,t_f]$（要求 t_i 单调递增或递减），y0 初始条件。

下面将以ode45为例具体介绍函数的使用方法。

1)函数格式

[T,Y] = ode45(‘odefun’ ,tspan,y0)

[T,Y] = ode45(‘odefun’ ,tspan,y0,options)

[T,Y,TE,YE,IE] = ode45(‘odefun’ ,tspan,y0,options)

sol = ode45(‘odefun’ ,[t0 tf],y0...)

其中: odefun是函数句柄,可以是函数文件名,匿名函数句柄或内联函数名;

tspan 是求解区间 [t0 tf]，或者一系列散点[t0,t1,...,tf];

y0 是初始值向量

T 返回列向量的时间点

Y 返回对应T的求解列向量

options 是求解参数设置,可以用odeset在计算前设定误差,输出参数,事件等

TE 事件发生时间

YE 事件发生时之答案

IE 事件函数消失时之指针i

在 t=[3.9,4]区间内求解常微分方程 $y''=-ty+e^t y'+3\sin 2t, y_{|t=3}=8, y'_{|t=3}=2$。

在matlab中新建脚本文件，编写函数如下：

odefun.m*

```
function dx=odefun(t,x)
dx=zeros(2,1);                          %初始化dx
dx(1)=x(2);
dx(2)=-t*x(1)+exp(t)*x(2)+3*sin(2*t);
end
```

main.m | odefun.m*

```
%% ode45求解
tspan=[3.9 4];                          %求解区间
y0=[8 2];                               %初值
[t,x]=ode45('odefun',tspan,y0);
%% 画图
plot(t,x(:,1),'-o',t,x(:,2),'-*')
legend('y','y''')
title('y'''' =-t*y+exp(t)*y''+3*sin(2*t)')
xlabel('t')
ylabel('y')
```

https://blog.csdn.net/lynn15600693998

练 习

7-1. 用梯形方法求解初值问题

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

取步长h=0.1,并与精确解$y=e^{-x}$相比较.

7-2. 确定两步方法

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

的阶.

7-3 用欧拉法求解初值问题

$$\begin{cases} y' = ax + b, \\ y(0) = 0, \end{cases}$$

(1) 导出近似解 y_n 的近似表达式;

(2) 证明正体截断误差为 $y(x_n) - y_n = \frac{1}{2}anh^2$.

7-4 用改进欧拉法求初值问题

$$\begin{cases} y' + y + y^2 \sin x = 0 \\ y(1) = 1 \end{cases},$$

要求取步长 $h=0.2$, 试计算 $y(1.2)$及 $y(1.4)$的近似值.（小数点后至少保留 5 位）