



信阳师范学院  
数学与统计学院  
SCHOOL OF MATHEMATICS AND STATISTICS

## 第4章 函数与数值积分



讲授人：牛言涛



日期：2020年2月25日

# 目录

## CONTENTS



函数的表示



数学函数图像的绘制



函数极值



函数求解



数值积分



# 1. 单变量非线性函数求解

根据初始值，猜测 $y=0$ 的解，可以看到迭代次数

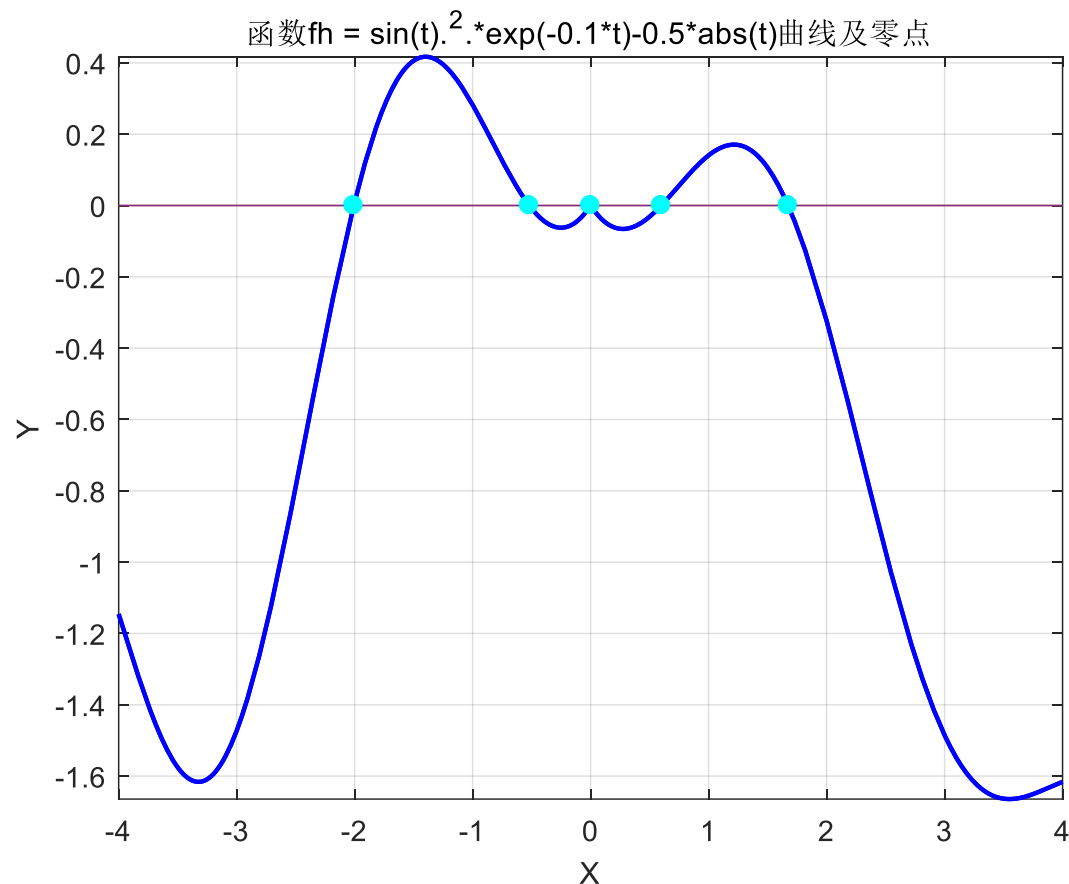
函数 `fzero()` 用来求一元函数的零点，调用格式如下：

- `x = fzero(fun,x0)`, `x = fzero(fun,[x1,x2])`: 寻找 `x0` 附近或者区间 `[x1,x2]` 内 `fun` 的零点，返回该点的 `x` 坐标；
- `[x,fval] = fzero(fun,x0,options)`: 给定参数 `options`，返回零点的同时返回该点的函数值；
- `[x, fval, exitflag, output] = fzero(fun,x0,options)`: 返回零点、该点的函数值、程序退出的标志及选定的输出结果。
- `x = fzero(problem)`: 对 `problem` 指定的求根问题求解。

# 1. 单变量非线性函数求解

- 例：求函数  $f(t) = (\sin^2 t)e^{-0.1t} - 0.5|t|$  的零点。

```
>> fh = @(t)sin(t).^2.*exp(-0.1*t)-0.5*abs(t);  
>> fplot(fh, [-4, 4], 'b-', 'LineWidth', 1.5)  
>> grid on; hold on  
>> fplot(@(t)0*t, [-4, 4])  
>> %分别求各点附近的根  
>> [t1, y1] = fzero(fh, -2)  
t1 =  
    -2.0074  
y1 =  
    2.2204e-16  
>> [t2, y2] = fzero(fh, -0.5)  
t2 =  
    -0.5198  
y2 =  
     0  
>> [t3, y3] = fzero(fh, 0.1);  
>> [t4, y4] = fzero(fh, 0);  
>> [t5, y5] = fzero(fh, 2);  
>> plot([t1, t2, t3, t4, t5], [y1, y2, y3, y4, y5], 'co', 'MarkerFaceColor', 'c')
```



# 1. 单变量非线性函数求解

```
>> fh = @(t)sin(t).^2.*exp(-0.1*t)-0.5*abs(t);
>> options = optimset('PlotFcns',{@optimplotx,@optimplotfval});
>> [t,y] = fzero(fh,0.1,options)
t =
    0.5993
y =
    1.1102e-16
```

```
>> problem.objective = @(t)sin(t).^2.*exp(-0.1*t)-0.5*abs(t);
>> problem.x0 = 0.1;
>> problem.solver = 'fzero'; % a required part of the structure
>> problem.options = optimset('Display','iter','PlotFcns',{@optimplotx,@optimplotfval});
>> [t,y] = fzero(problem)
```

围绕 0.1 搜索包含符号变换的区间:

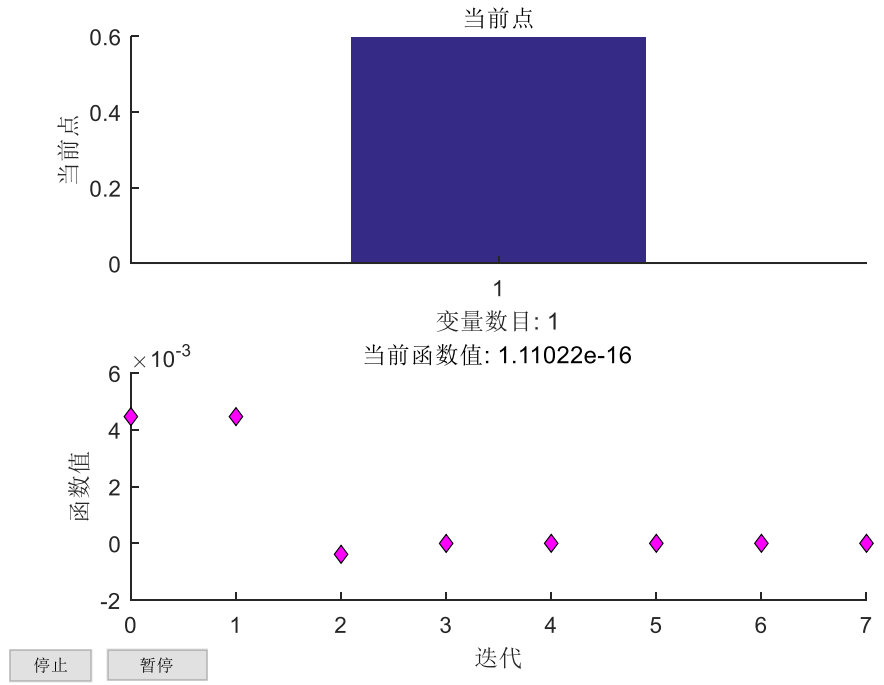
Func-count	a	f(a)	b	f(b)	Procedure
1	0.1	-0.0401325	0.1	-0.0401325	initial interval
3	0.0971716	-0.0392642	0.102828	-0.0409855	search

在区间 [-0.412, 0.612] 中搜索零:

Func-count	x	f(x)	Procedure
33	0.612	0.00446291	initial
34	0.612	0.00446291	interpolation
35	0.598152	-0.000387467	interpolation
36	0.599258	-3.51262e-06	interpolation
37	0.599268	1.39217e-10	interpolation
38	0.599268	-9.4369e-16	interpolation
39	0.599268	-1.11022e-16	interpolation
40	0.599268	1.11022e-16	interpolation

在区间 [-0.412, 0.612] 中发现零

```
t =
    0.5993
y =
    1.1102e-16
```



## 2. 多元非线性函数求解

•  $[x, fval, exitflag, output, jacobian] = fsolve(fun, x0, options)$

• 例：求解非线性方程组

$$\begin{cases} \sin x + y + z^2 e^x - 4 = 0 \\ x + yz = 0 \\ xyz = 0 \end{cases}$$

① 定义函数文件

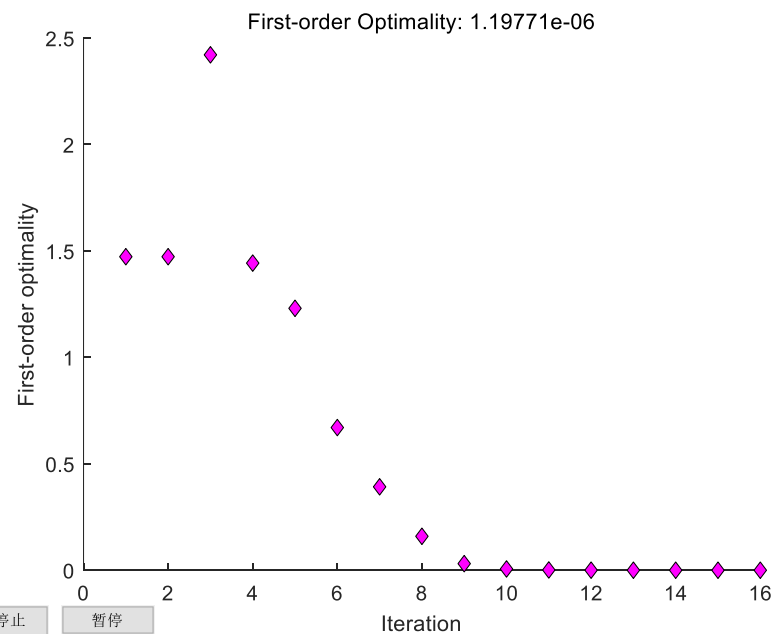
② 命令行窗口获取函数句柄

③ 设置options参数

④ 调用函数求解

```
>> fh = @nlineqsfun;
>> options = optimoptions('fsolve','Display','none','PlotFcn',@optimplotfirstorderopt);
>> [x, fval] = fsolve(fh, [1, 1, 1], options)
x =
    0.0005   -0.0002    1.9995
fval =
    1.0e-06 *
   -0.2636    0.1200   -0.2131
```

```
nlineqsfun.m  x  +
1  %建立M文件:
2  function f = nlineqsfun(v)
3      x = v(1);
4      y = v(2);
5      z = v(3);
6      f(1) = sin(x)+y+z^2.*exp(x)-4;
7      f(2) = x+y*z;
8      f(3) = x*y*z;
9  end
```



## 2. 多元非线性函数求解

- 例：在给定的初值(1, 1, 1)下，求方程组的数值解。

$$\begin{cases} \sin x + y^2 + \ln z - 7 = 0 \\ 3x + 2^y - z^3 + 1 = 0 \\ x + y + z - 5 = 0 \end{cases}$$

```
x =  
    0.5991    2.3959    2.0050  
fval =  
    1.0e-10 *  
    0.2213    0.3803   -0.0004  
exitflag =  
    1  
output =  
包含以下字段的 struct:  
  
    iterations: 7  
    funcCount: 29  
    algorithm: 'trust-region-dogleg'  
    firstorderopt: 4.4767e-10  
    message: 'Equation solved. ...'
```

```
>> fh = @(x)[sin(x(1))+x(2)^2+log(x(3))-7, 3*x(1)+2^x(2)-x(3)^3+1, x(1)+x(2)+x(3)-5];  
>> options = optimoptions('fsolve','Display','iter','PlotFcn',@optimplotfirstorderopt);  
>> [x,fval,exitflag,output] = fsolve(fh,[1,1,1],options)
```

Iteration	Func-count	f(x)	Norm of step	First-order optimality	Trust-region radius
0	4	55.6104		22.2	1
1	8	23.0855	1	14.4	1
2	9	23.0855	2.13102	14.4	2.5
3	13	9.93091	0.532754	9.5	0.533
4	17	0.582066	0.875905	4.9	1.33
5	21	0.00159514	0.219095	0.403	2.19
6	25	5.02061e-10	0.00610298	0.000124	2.19
7	29	1.93608e-21	7.63728e-06	4.48e-10	2.19

[Equation solved.](#)

fsolve completed because the vector of function values is near zero as measured by the default value of the [function tolerance](#), and the [problem appears regular](#) as measured by the gradient.



---

# 感谢聆听

---