

%用微分方程解龙头位置

```
figure;  
hold on;  
axis equal; % 保持坐标比例一致
```

% 定义常数

```
b = 0.55 * 16; % 起始半径  
a = 0.55 / (2 * pi); % 每圈增加的半径
```

% 微分方程定义

```
odefun = @(t, y) 1 / sqrt((b - a * y)^2 + a^2);  
tspan = [0 300]; % 时间跨度  
y0 = 0; % 初始条件
```

% 使用ode45求解微分方程

```
options = odeset('RelTol',1e-5,'AbsTol',1e-7);  
[t, y] = ode45(odefun, tspan, y0, options);
```

% 创建每秒一个时间点的向量

```
t_interp = 0:1:300; % 每秒一个点
```

% 插值得到每秒的解

```
y_interp = interp1(t, y, t_interp, 'linear');
```

% 计算等距螺旋线的坐标

```
theta_spiral = linspace(0, -32*pi, 8000);  
r_spiral = b + a * theta_spiral;  
x_spiral = r_spiral .* cos(theta_spiral);  
y_spiral = r_spiral .* sin(theta_spiral);
```

% 将插值解映射到螺旋线的坐标上

```
theta_solution = y_interp; % 使用插值解作为 theta  
r_solution = b + a * theta_solution; % 根据解的角度计算半径  
x_solution = r_solution .* cos(theta_solution); % x坐标  
y_solution = r_solution .* sin(theta_solution); % y坐标
```

% 绘制等距螺旋线

```
plot(x_spiral, y_spiral, 'b')
```

% 绘制插值解的点

```
plot(x_solution, y_solution, 'ro')
```

% 现有螺旋线的参数

```
b = 0.55 * 16; % 起始半径  
a = 0.55 / (2 * pi); % 每圈增加的半径
```

% 现有螺旋线的计算和绘制

```
theta_spiral = linspace(0, -32*pi, 4000);  
r_spiral = b + a * theta_spiral;  
x_spiral = r_spiral .* cos(theta_spiral);  
y_spiral = r_spiral .* sin(theta_spiral);  
plot(x_spiral, y_spiral, 'b-'); % 绘制蓝色实线
```

% 更大螺旋线的计算和绘制

```
b_larger = b + 8.8; % 增大起始半径  
r_spiral_larger = b_larger + a * (theta_spiral - 20 * pi);  
x_spiral_larger = r_spiral_larger .* cos(theta_spiral);
```

```
y_spiral_larger = r_spiral_larger .* sin(theta_spiral);  
plot(x_spiral_larger, y_spiral_larger, 'k--'); % 绘制黑色虚线  
  
% 绘制原始的散点（圆心）  
plot(x_solution, y_solution, 'ro'); % 使用红色圆圈标记  
  
hold off; % 解除保持状态  
xlabel('X');  
ylabel('Y');  
grid on; % 添加网格以便更清楚地查看
```