

```

%                                     问题四:最小掉头路径                                     %

% 计算 4.5/(1.7/(2*pi)) 的结果
theta=4.5/(1.7/(2*pi));
ro=4.5;
dy=(1.7/(2*pi))*sin(theta)+(1.7/(2*pi))*theta*cos(theta);
dx=(1.7/(2*pi))*cos(theta)-(1.7/(2*pi))*theta*sin(theta);
kk=dy/dx;
k=-1/kk;
y=ro*sin(theta);
x=ro*cos(theta);

% 计算直线方程的参数
bb= y - kk*x;
b = y - k * x; % 直线方程为 y = kx + b, b为截距

% 计算直线到原点的距离
dd = abs(bb) / sqrt(1 + kk^2); %切线
d = abs(b) / sqrt(1 + k^2);

% 显示结果
disp(['垂线距离为: ', num2str(d)]);
disp(['切线距离为: ', num2str(dd)]);

% 定义目标函数和约束条件
objective = @(a) 2*d*a./sin(pi-a);
constraint = @(a) (2*d*a./sin(a)).*(1 + cos(pi-a))./a - 2*dd;

% 定义范围
a_min = pi/2;
a_max = pi;

% 设置优化选项
options = optimoptions('fmincon','Display','off');

% 初始猜测
a0 = (a_min + a_max) / 2;

% 使用 fmincon 进行优化
[a_opt, s_opt] = fmincon(@(a) 2*d*a./sin(pi-a), a0, [], [], [], [], ...
    a_min, a_max, @(a) deal([], constraint(a)), options);

% 输出结果
disp(['s最小值为: ', num2str(s_opt)]);
disp(['对应的phi值为: ', num2str(a_opt)]);

%求题目给出比例的圆弧长
syms l theta0
assume(theta0 > pi/2 & theta0 < pi);
eq1 = l*(1+(cos(pi-theta0))) == 2*dd;
eq2 = l*sin(pi-theta0) == 2*d;
sol = solve([eq1, eq2], [l, theta0]);
x_sol = sol.l;
y_sol = sol.theta0;
fprintf('题设条件下 m = %.4f\n', x_sol);
fprintf('题设条件下 phi = %.4f\n', y_sol);
fprintf('题设条件下 s = %.4f\n', x_sol*y_sol);

```

