Homework 1

Task 1　Discuss the advantages and limits of different wheel drives

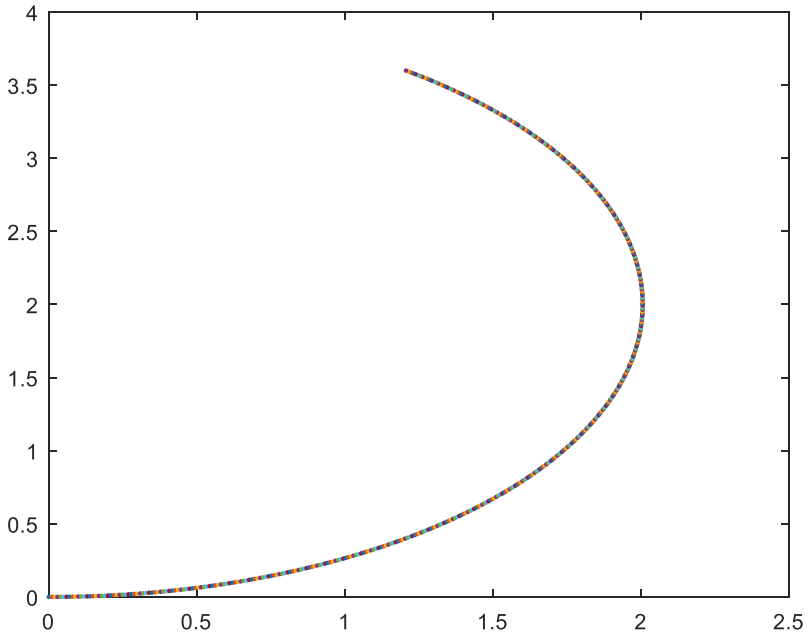
Type	advantages	limits
Differential Drive	1. The concept is simple, Differential wheeled robot can have two independently driven wheels fixed on a common horizontal axis or three wheels where two independently driven wheels and a roller ball or a castor attached to maintain equilibrium. 2. It can be incorporated in almost any kind of robots including legged robots	the robot does not drive as expected. It neither drives along a straight line nor turn exactly at expected angles, especially when we use DC motors.
Ackermann Drive	The advantage in this design is increased control, better stability and maneuverability on road, less slippage and less power consumption.	1. Vehicle cannot turn on the spot, but requires a certain minimum radius 2. Rear driving wheels experience slippage in curves
Synchronous Drive	It can be constructed with three, or four or any number of wheels	The robot has to stop and realign its wheels when going from driving forward to driving sideways. Nor can it drive and rotate at the same time.
XR4000 Drive	It uses four active drive modules resulting in the coordination of eight independent degrees of freedom	Heavy and complex
Mecanum Drive	The rollers are freely rotating and maneuverability	It is difficult to control and less pushing force

Reference:

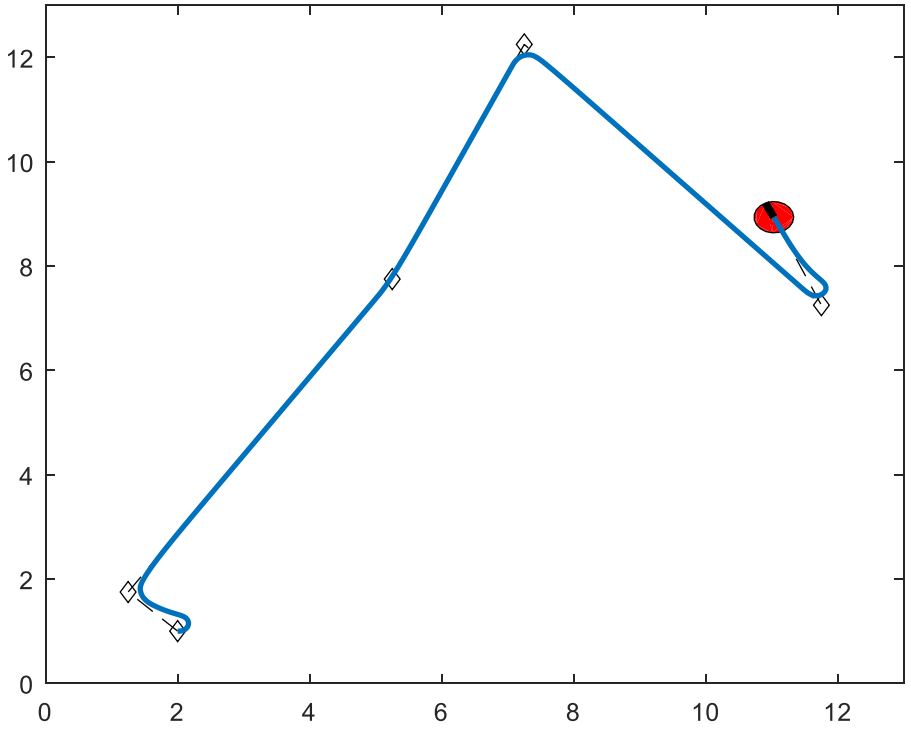
Embedded Robotics: Mobile Robot Design and Applications with Embedded Systems by Thomas Bräunl

http://www.robotplatform.com/knowledge/Classification_of_Robots/wheel_control_theory.html

Task 2 Simulate the kinematics of a robot with a differential drive



Task 3 Simulate the motion control of a robot with a differential driveand show the resulting paths w.r.t. different control laws



Reference:

https://github.com/mlab-upenn/arch-apex/blob/2af0fc3d6b61ad738aca2e100e4966ad394a5218/Code/pure_pursuit.m

源代码：

Task 2:

```
function [x,y,theta] = Differential_drive(xi,v,omega,tmax)
x(1) = xi(1); y(1) = xi(2); theta(1) = xi(3);
deltat = 0.005;
for t=1:tmax
    x(t+1) = x(t) + v*deltat*cos(theta(t));
    y(t+1) = y(t) + v*deltat*sin(theta(t));
    theta(t+1) = theta(t) + omega*deltat;
end

function main
    [x,y,z]= Differential_drive([0,0,0],2,1,500);
    a=[x.' y.'];
    for i=1:size(a,1)
        plot(a(i,1),a(i,2),'.');
        hold on;
        pause(0.1);
    end
end
```

Task 3:

```
path = [2.00 1.00;1.25 1.75;5.25 7.75;7.25 12.25;11.75 7.25;11.00
9.00];

R_current_location=path(1,:);
R_goal_location=path(end,:);
initialOrientation = 0;
R_current_pos=[R_current_location initialOrientation];
R_radius=0.4;

plot(path(:,1), path(:,2),'k--d')
xlim([0 13])
ylim([0 13])

robot = ExampleHelperDifferentialDriveRobot(R_current_pos);

controller=robotics.PurePursuit;
controller.Waypoints = path;
controller.DesiredLinearVelocity = 0.3;
controller.MaxAngularVelocity = 2;
controller.LookaheadDistance = 0.5;
goalRadius = 0.1;

distanceToGoal = norm(R_current_location - robotGoal);

while( distanceToGoal > goalRadius )

    [v, omega] = step(controller, robot.CurrentPose);
    drive(robot, v, omega)
    R_current_location = robot.CurrentPose(1:2);
    distanceToGoal = norm(R_current_location - robotGoal);
end
```