

Farid Tavakkolmoghaddam HW4

```
%Steering control
clear; close all; clc
T=50;
x0 =[0;1;pi/4]; % Feel free to change the initial state and sampling horizon.

% design steering control to follow a straight lane while maintaining a given velocity
% the lateral position yr= 2;

%TODO: param is the additional parameter to pass to the ode function.
[T,X] = ode45(@(t,x) ode_dubins(t,x), (0:T), x0);
tt=0:1:50;
XX=10*tt; % desired trajectory x
YY=2*ones(1,51); % desired trajectory y
theta=2*zeros(1,51); % desired trajectory y

% plot your state trajectories for both 1 and 2, using the following code or else.
figure
title(' X position vs. time')
plot(T,X(:,1),'k',tt,XX,'b*','LineWidth',2);
xlabel('t (s)');
ylabel('x (m)','FontSize',12,'FontWeight','bold','Color','k');
legend('Actual','desired')
grid

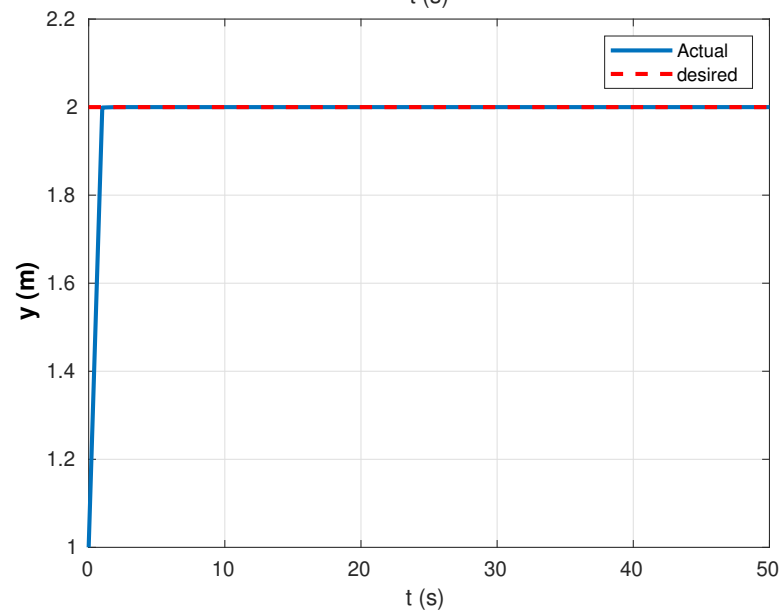
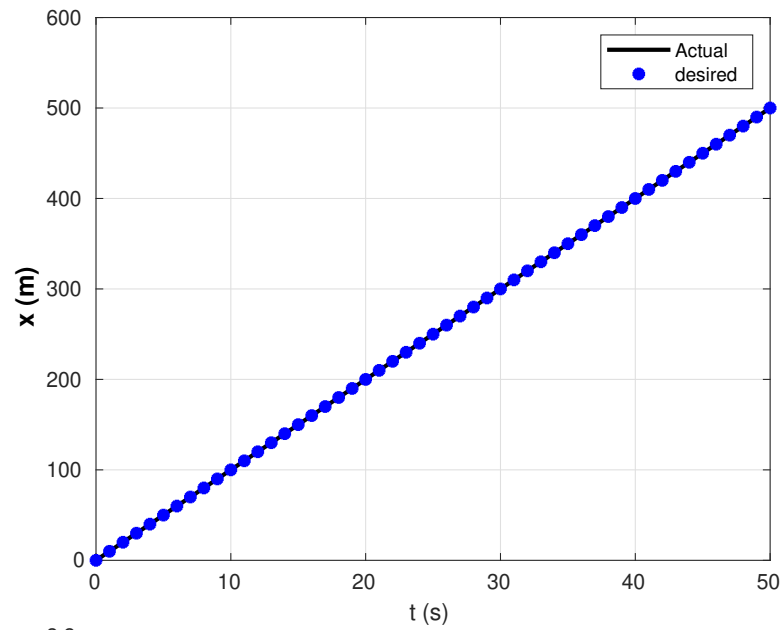
figure
title(' Y position vs. time')
plot(T,X(:,2),tt,YY,'r--','LineWidth',2);
xlabel('t (s)');
ylabel('y (m)','FontSize',12,'FontWeight','bold','Color','k');
legend('Actual','desired')
grid

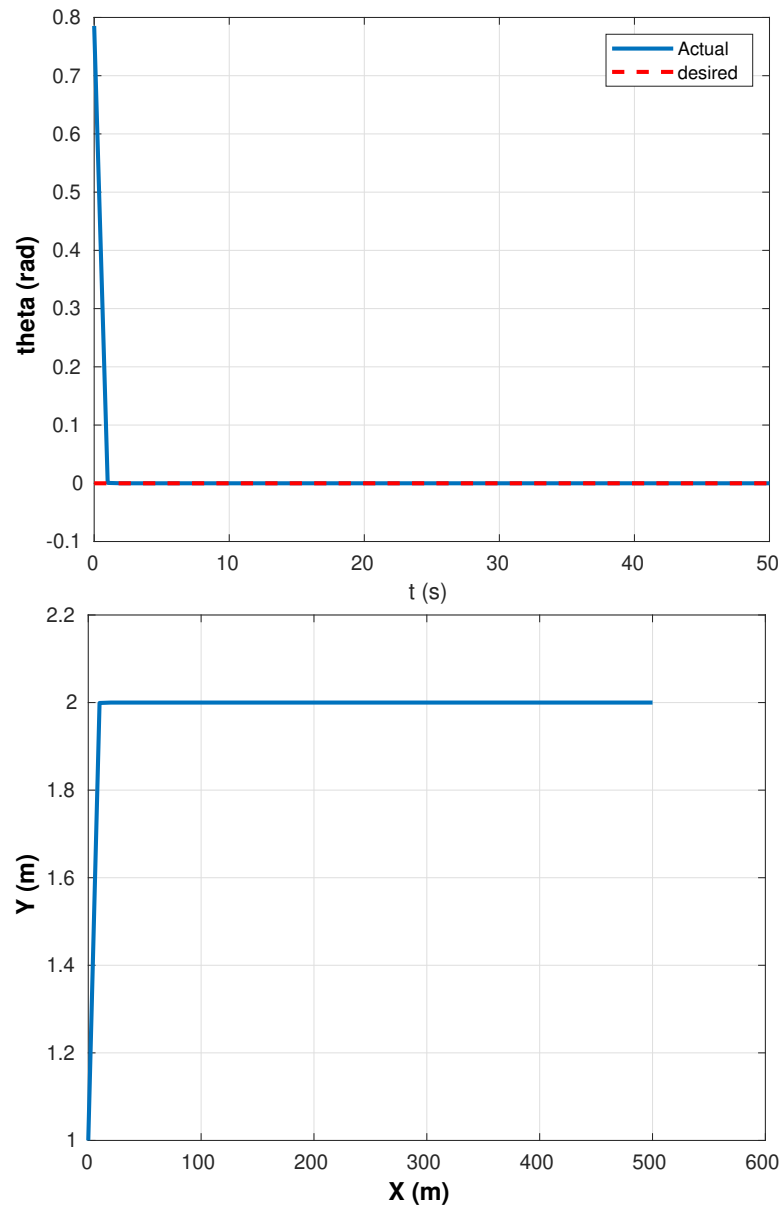
figure
title(' Orientation vs. time')
plot(T,X(:,3),tt,theta,'r--','LineWidth',2);
xlabel('t (s)');
ylabel('theta (rad)','FontSize',12,'FontWeight','bold','Color','k');
legend('Actual','desired')
grid
```

```

figure('name','X v.s y')
title(' x vs. y')
plot(X(:,1), X(:,2),'LineWidth',2)
xlabel('X (m)','FontSize',12,'FontWeight','bold','Color','k');
ylabel('Y (m)','FontSize',12,'FontWeight','bold','Color','k');
grid

```





1 ODE Dubins

article graphicx color

Contents

- TODO: Here is the code for control input.

- the controller needs to provide control input: linear velocity and
- steering angle: v , δ

```
function dz = ode_dubins(t,z)
```

```
% use z for [x,y,theta]
dz =zeros(3,1);
```

TODO: Here is the code for control input.

the controller needs to provide control input: linear velocity and

steering angle: v , δ

```
% step 1: calculate the value of deviation variables given the
% desired state and input and the actual state, exacted from z.
v_d = 10;
w_d = 0;
u_d = [v_d ; w_d];

y_d = 2;
theta_d = 0;
v_r = 10;
x_d = [v_r*t; y_d ;theta_d];
e_x = z - x_d;

% de_x = dx - dx_d;

% step 2: based on the feedback controller, calculate the e_u=
% [e_v, e_w].
k=[5000 200 300;
   1000 200 300];
e_u = -k*e_x;

% step 3: based on the relation between u and e_u, derive the
% desired input v, and w.
% e_u= u - u_d; --> u = e_u + u_d
u= -k*e_x + u_d;
v= u(1);
w= u(2);
theta= z(3);

dz(1) = v*cos(theta);
dz(2) = v*sin(theta);
dz(3) = w;
```

Not enough input arguments.

Error in ode_dubins (line 18)
x_d = [v_r*t; y_d ;theta_d];

end