Problem Set 1

Connor Anderson
AA274 Principles of Robotic Autonomy

January 25, 2017

1 Problem 1

i) Derive the Hamiltonian and conditions for optimality and formulate the problem as a 2P - BVP.

We start with basic dynamics equations:

$$\dot{x}(t) = V cos(\theta(t))$$
$$\dot{y}(t) = V sin(\theta(t))$$
$$\dot{\theta}(t) = \omega(t)$$

We can think of these as a function $a(x(t), u(t), t)$ and say that the derivative of the state vector $\dot{x}(t) = a(x(t), u(t), t)$.

Our goal is to minimize:

$$J = \int_0^{t_f} [\lambda + V(t)^2 + w(t)^2]dt = \lambda t_f + \int_0^{t_f} [V(t)^2 + w(t)^2]dt$$

Now we can express this equation in terms of the desired functions, g, and h.

$$h(t_f) = \lambda t_f$$

$$g(x(t), u(t), t) = V(t)^2 + w(t)^2$$

Next, we now form the hamiltonian, in terms of the above functions and our co-state vector $p = (p_1; p_2; p_3)$.

$$H := g(x(t), u(t), t) + p^T(t)[a(x(t), u(t), t)]$$

$$H = V(t)^2 + \omega(t)^2 + p_1 V(t) cos(\theta(t)) + p_2 V(t) sin(\theta(t)) + p_3 \omega(t)$$

Next, we take the partial derivatives of H in order to find the Hamiltonian equations.

$$\dot{\tilde{X}} = \frac{\partial H}{\partial p} = \begin{bmatrix} V(t)cos(\theta(t)) \\ V(t)sin(\theta(t)) \\ w(t) \end{bmatrix}$$

$$\dot{\tilde{p}} = -\frac{\partial H}{\partial X} = \begin{bmatrix} \frac{\partial H}{\partial x} \\ \frac{\partial H}{\partial y} \\ \frac{\partial H}{\partial \theta} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ p_1 V(t)sin(\theta(t)) - p_2 V(t)cos(\theta(t)) \end{bmatrix}$$

$$0 = \frac{\partial H}{\partial u} = \begin{bmatrix} \frac{\partial H}{\partial V} \\ \frac{\partial H}{\partial w} \end{bmatrix} = \begin{bmatrix} p_1 cos(\theta(t)) + p_2 * sin(\theta(t)) + 2V(t) \\ 2w(t) + p_3 \end{bmatrix}$$

Using these equations, we can express our new dynamics equations as:

$$\dot{\tilde{x}} = \tilde{V}(t)cos(\tilde{\theta}(t))$$
$$\dot{\tilde{y}} = \tilde{V}(t)sin(\tilde{\theta}(t))$$
$$\dot{\tilde{\theta}} = \tilde{w}(t)$$
$$\dot{\tilde{p_1}} = 0$$
$$\dot{\tilde{p_2}} = 0$$
$$\dot{\tilde{p_3}} = \tilde{p_1}\tilde{V}(t)sin(\tilde{\theta}(t)) - \tilde{p_2}\tilde{V}(t)cos(\tilde{\theta}(t))$$

We can also express $\tilde{V}$ and $\tilde{w}$ as:

$$\tilde{V} = \frac{-\tilde{p_1}cos(\tilde{\theta}(t)) - \tilde{p_2}sin(\tilde{\theta}(t))}{2}$$

$$\tilde{w} = \frac{-\tilde{p_3}}{2}$$

From problem description, we have 6 boundary conditions. We require one last equation (from slide 9 of lecture notes) to finish problem formulation. This takes form:

$$\frac{\partial h}{\partial t}(t_f) + H(\tilde{x}(t_f), \tilde{u}(t_f), \tilde{p}(t_f)) = 0$$

$$\lambda + \tilde{V}(t_f)^2 + \tilde{w}(t_f)^2 + \tilde{p_1}\tilde{V}(t)cos(\tilde{\theta}(t)) + \tilde{p_2}\tilde{V}(t)sin(\tilde{\theta}(t)) + \tilde{p_3}\tilde{w}(t_f) = 0$$

Lastly, we rescale the time using a new time variable τ and add a dummy state r to represent t_f.

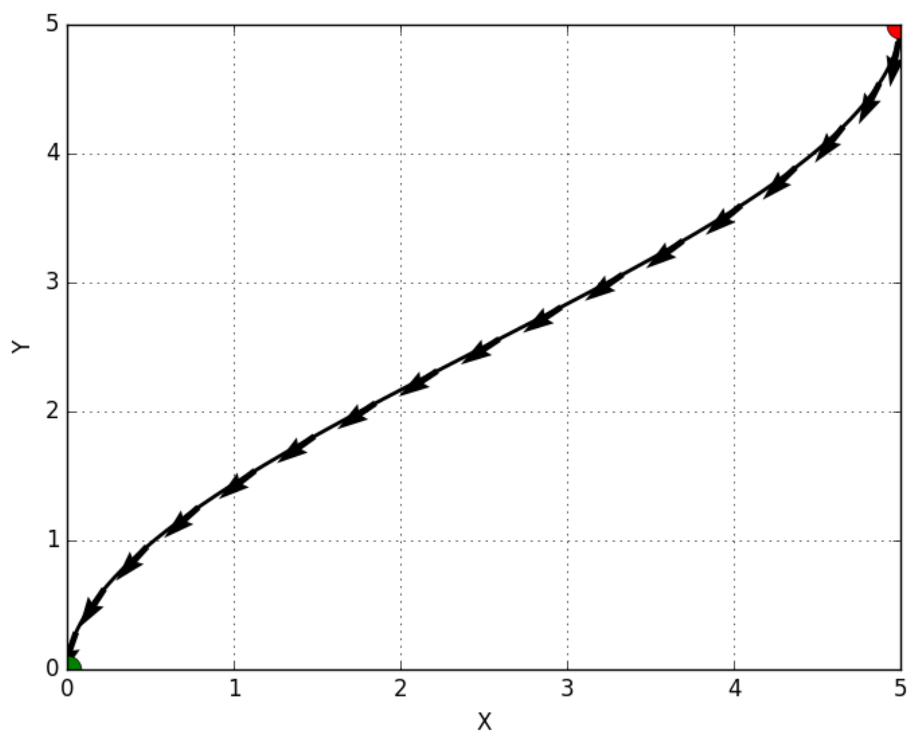
Now our dynamics equations are

$$\tilde{x}\prime = r\tilde{V}(t)cos(\tilde{\theta}(t))$$
$$\tilde{y}\prime = r\tilde{V}(t)sin(\tilde{\theta}(t))$$
$$\tilde{\theta}\prime = r\tilde{w}(t)$$
$$\tilde{p_1}\prime = 0$$
$$\tilde{p_2}\prime = 0$$
$$\tilde{p_3}\prime = r[\tilde{p_1}\tilde{V}(t)sin(\tilde{\theta}(t)) - \tilde{p_2}\tilde{V}(t)cos(\tilde{\theta}(t))]$$
$$\tilde{r}\prime = 0$$

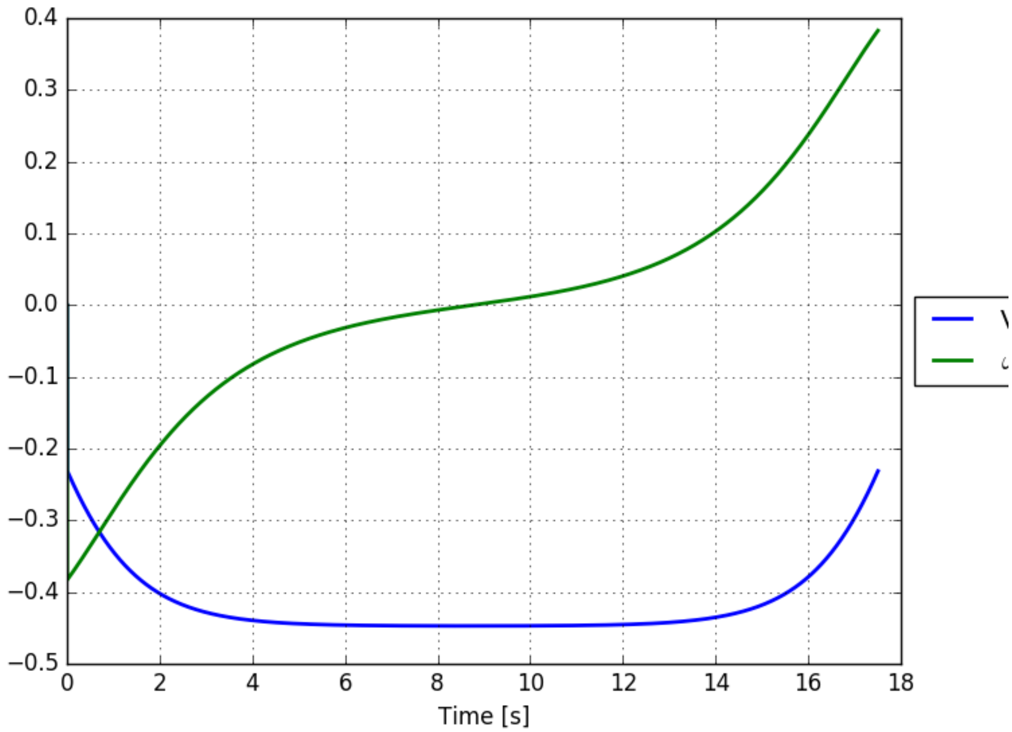
With boundary conditions:

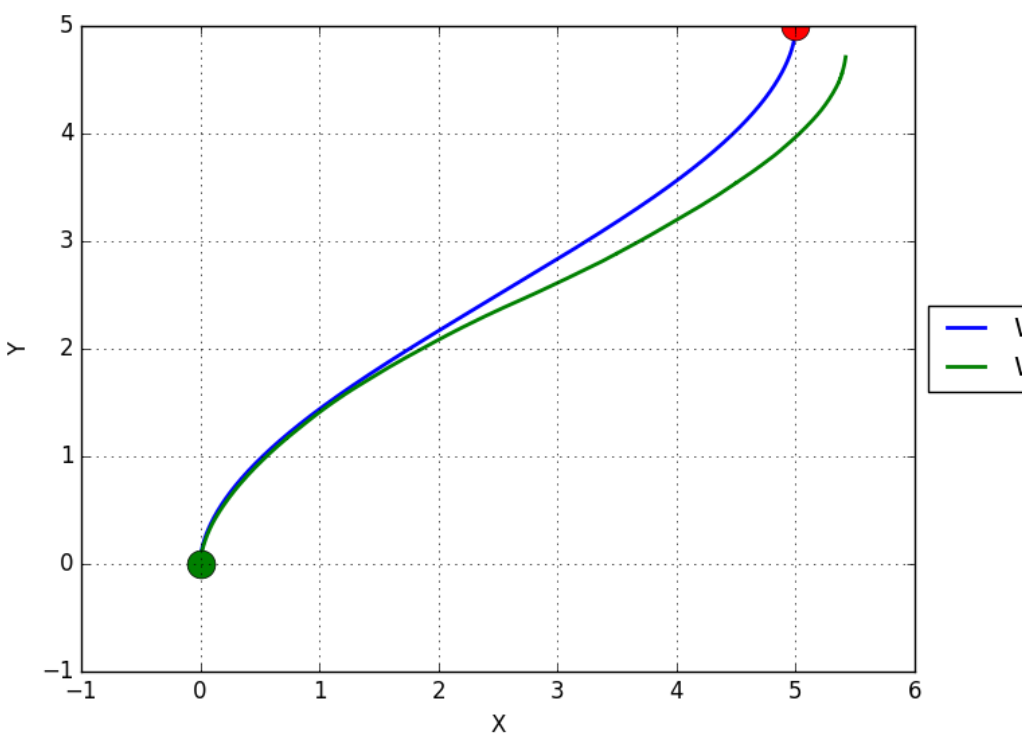
$$\tilde{x}(0) = 0$$
$$\tilde{y}(0) = 0$$
$$\tilde{\theta}(0) = -\pi/2$$
$$\tilde{x}(1) = 5$$
$$\tilde{y}(1) = 5$$
$$\tilde{\theta}(1) = -\pi/2$$
$$\lambda + \tilde{V}(1)^2 + \tilde{w}(1)^2 + \tilde{p_1}\tilde{V}(1)cos(\tilde{\theta}(1)) + \tilde{p_2}\tilde{V}(1)sin(\tilde{\theta}(1)) + \tilde{p_3}\tilde{w}(1) = 0$$

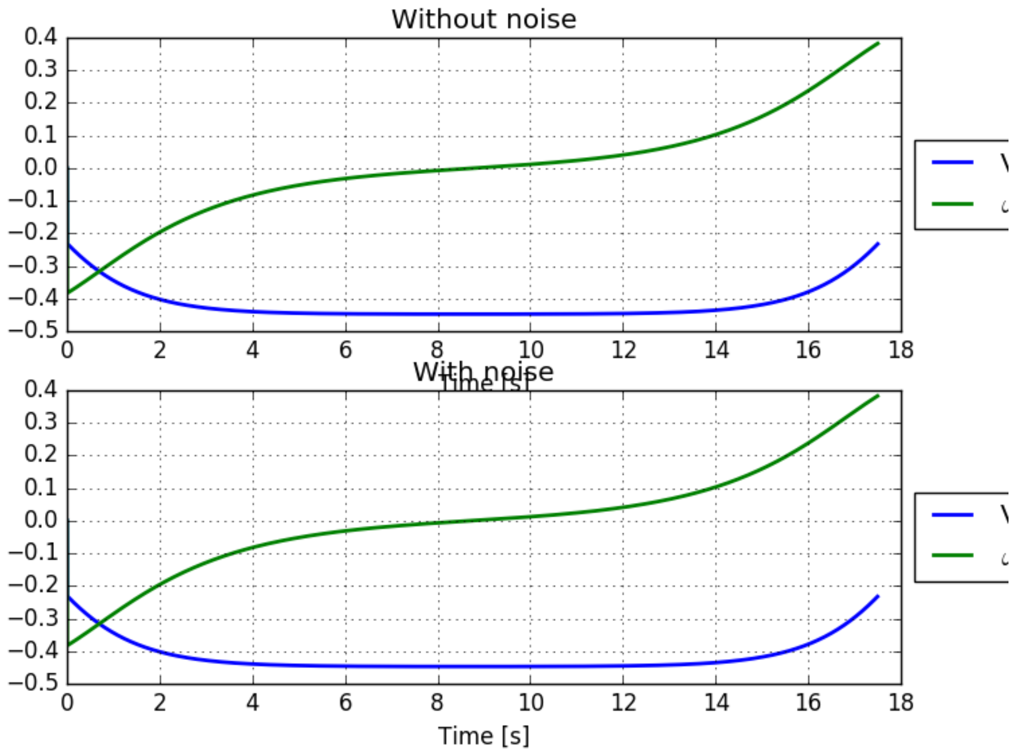
ii) Implemented in code



iii)


Time [s]


Y
X



2 Problem 2

i) We start with the given expression for x and y:

$$x = x_1 * \psi_1 + x_2 * \psi_2 + x_3 * \psi_3 + x_4 * \psi_4 = x_1 + x_2 t + x_3 t^2 + x_4 t^3$$
$$y = y_1 + y_2 t + y_3 t^2 + y_4 t^3$$
$$\dot{x} = x_2 + 2x_3 t + 3x_4 t^2$$
$$\dot{y} = y_2 + 2y_3 t + 3y_4 t^2$$

Now we can use these expressions to form a matrix equation of constraints. Particularly, we write:

$$x_1 = 0$$
$$y_1 = 0$$
$$y_1 + y_2 t_f + y_3 t_f^2 + y_4 t_f^3 = 5$$
$$x_1 + x_2 t_f + x_3 t_f^2 + x_4 t_f^3 = 5$$
$$x_2 = 0$$
$$x_2 + 2x_3 t_f + 3x_4 t_f^2 = 0$$

$$y_2 = -0.5$$

$$y_2 + 2y_3 t_f + 3y_4 t_f^2 = -0.5$$

Finally, as a matrix equation, these coefficients can be solved for by writing:

$$Ax = b = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 2t_f & 3t_f^2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ y_1 \\ y_2 \\ y_3 \\ y_4 \end{bmatrix} = \begin{bmatrix} 0 \\ 5 \\ 0 \\ 5 \\ 0 \\ 0 \\ -0.5 \\ -0.5 \end{bmatrix}$$

Solution is :

$$x = A^{-1}b$$

ii) Once we have solved for these parameters, we can scale our velocity to ensure that we've stayed below the max allowable velocity.

We first define $s = \int_0^{t_f} V(t)dt$ and $\tilde{V} = min(0.5, V(s), \frac{1}{|w_s|})$, where $w_s = w/V$.

Upon having this rescaled V, we need to now find a rescaled s(t) parameter. To do this, we simply integrate $\tilde{V}(s)$ wrt s in order to find the final time. Once we have the final time, we can integrate $\tilde{V}$ wrt t to find s. To find a complete t vector, we use the "cumtrapz" function in numpy.

$$t_f = \int_0^{s_f} \frac{1}{\tilde{V}(s)} ds$$

$$\tilde{t} = cumtrapz(\frac{1}{\tilde{V}}, s)$$

$$\tilde{s} = \int_0^{t_f} \tilde{V}(t)dt = cumtrapz(\tilde{V}(\tilde{t}), \tilde{t})$$

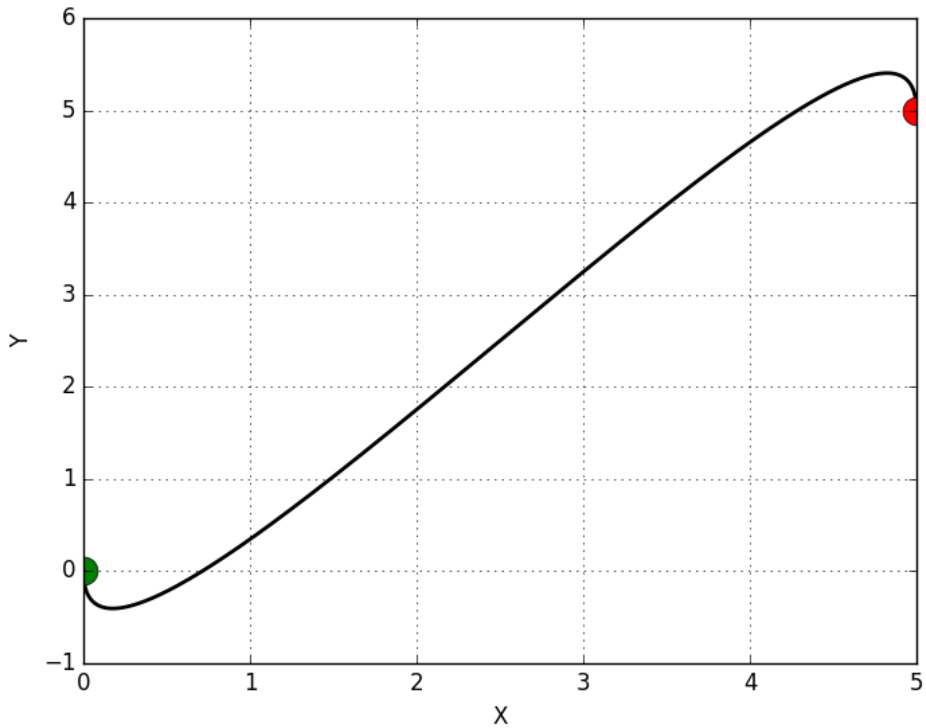
Finally, we get both t and s into 5ms increments by using an interpolation of $\tilde{s}$ and $\tilde{t}$ over the longer time horizon.

To obtain x,y, and θ, we note that x is invariant to path. Therefore $x(s) = x(\tilde{s})$. We simply perform an interpolation of x(s) using $\tilde{s}$ and we have the final solution. This process is used for y and θ as well. We can also solve for the derivatives of x and y using:

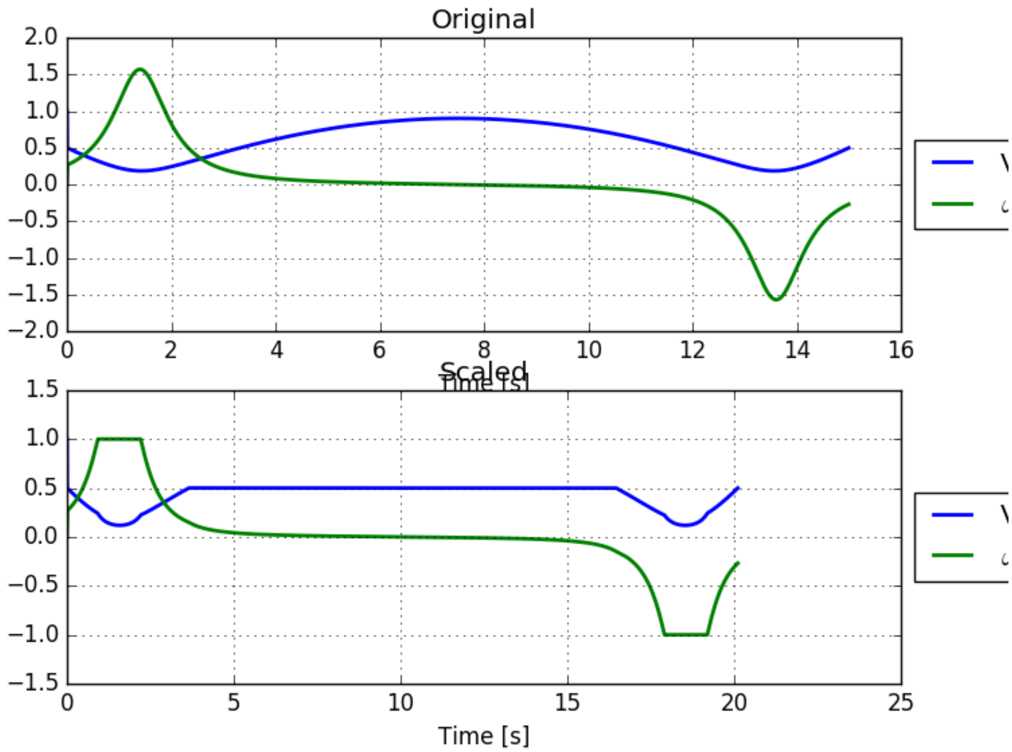
$$\dot{x} = V cos(\theta)$$

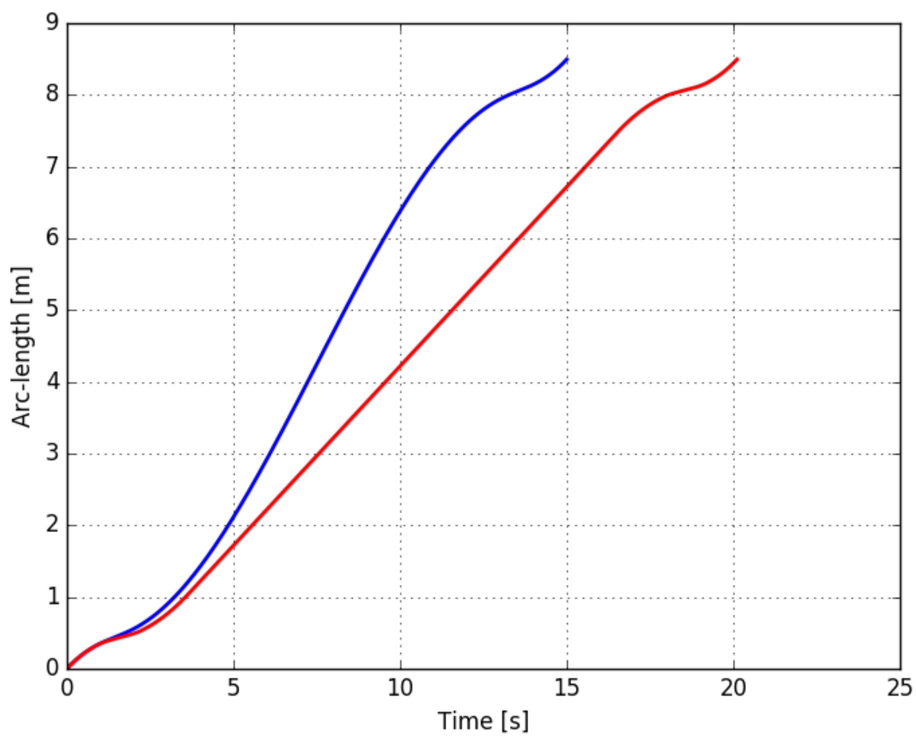
$$\dot{y} = V sin(\theta)$$

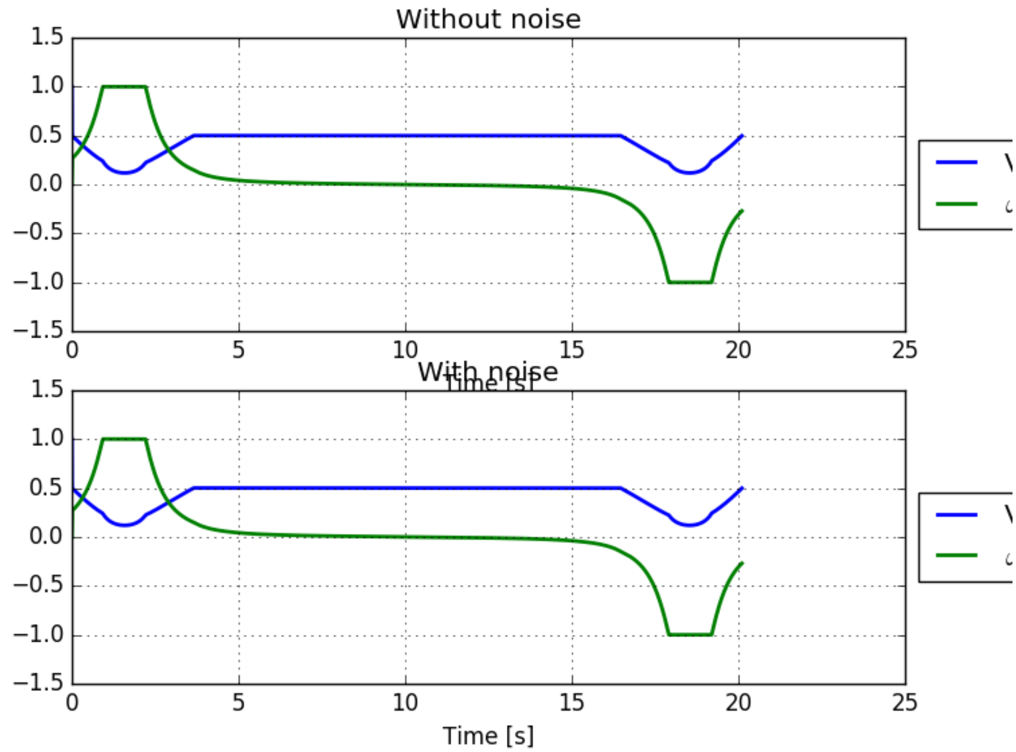
The double derivatives of x and y are solved for by using the "diff" function in python's numpy library, making sure to divide by dt.

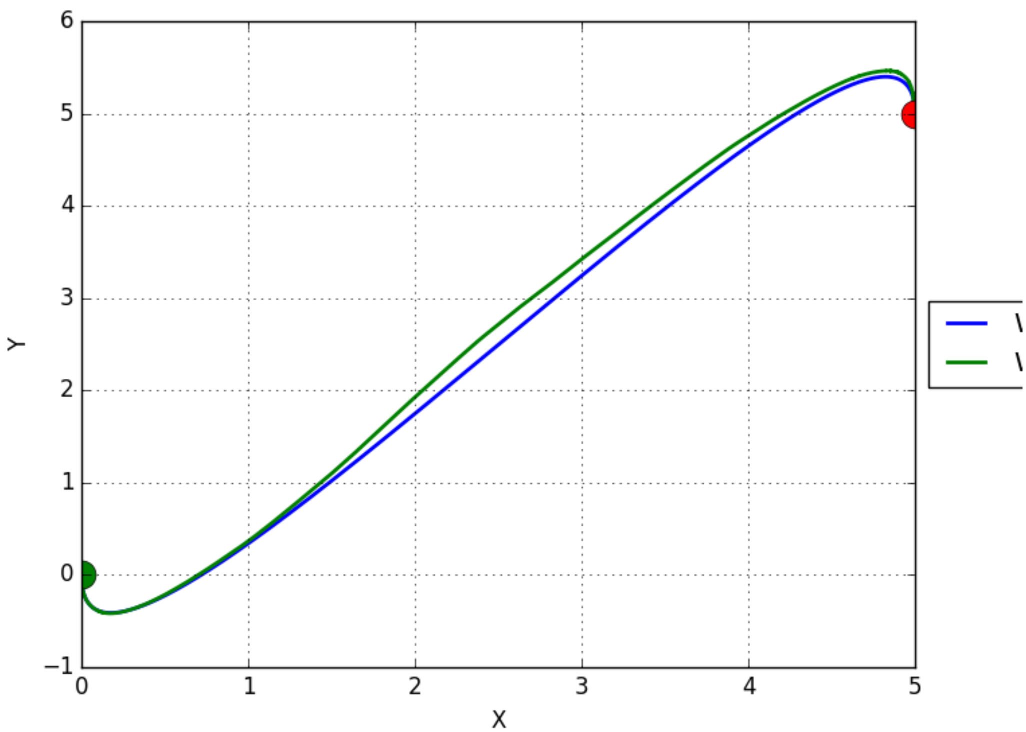


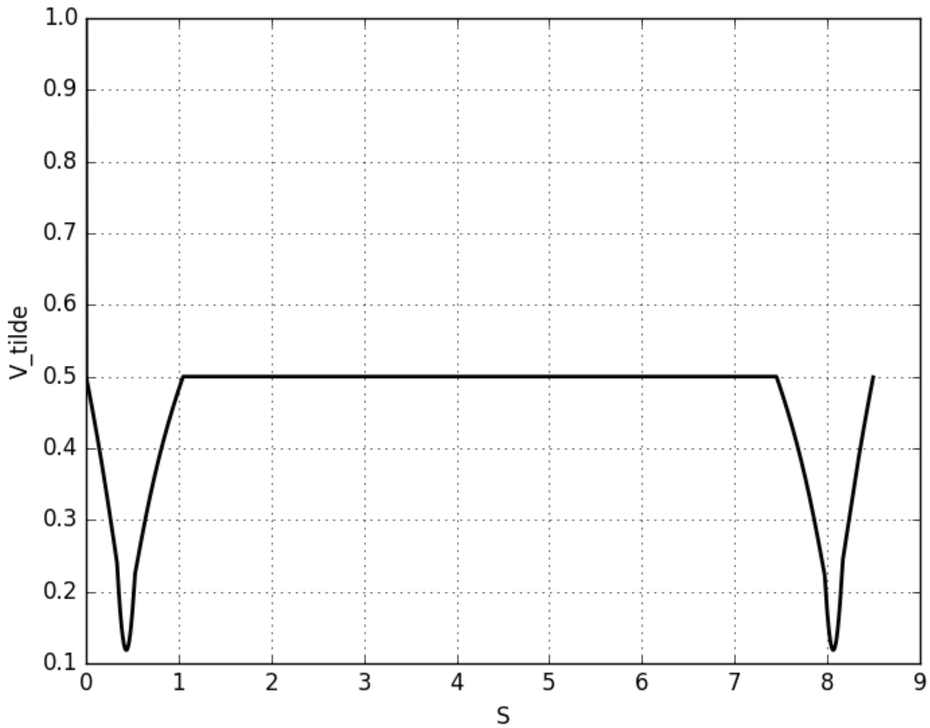
iii)


Original
Time [s]
Scaled
Time [s]


Arc-length [m]
Time [s]

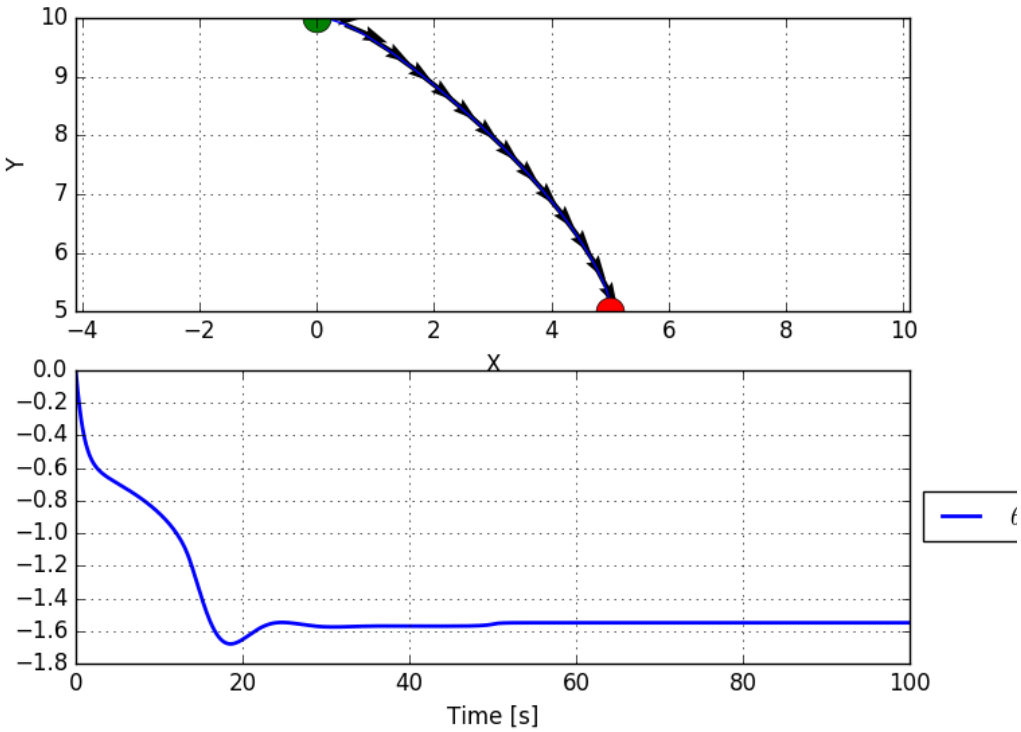

Without noise
1.5
1.0
0.5
0.0
−0.5
−1.0
−1.5
0
5
10
15
20
25
Time [s]
With noise
Time [s]

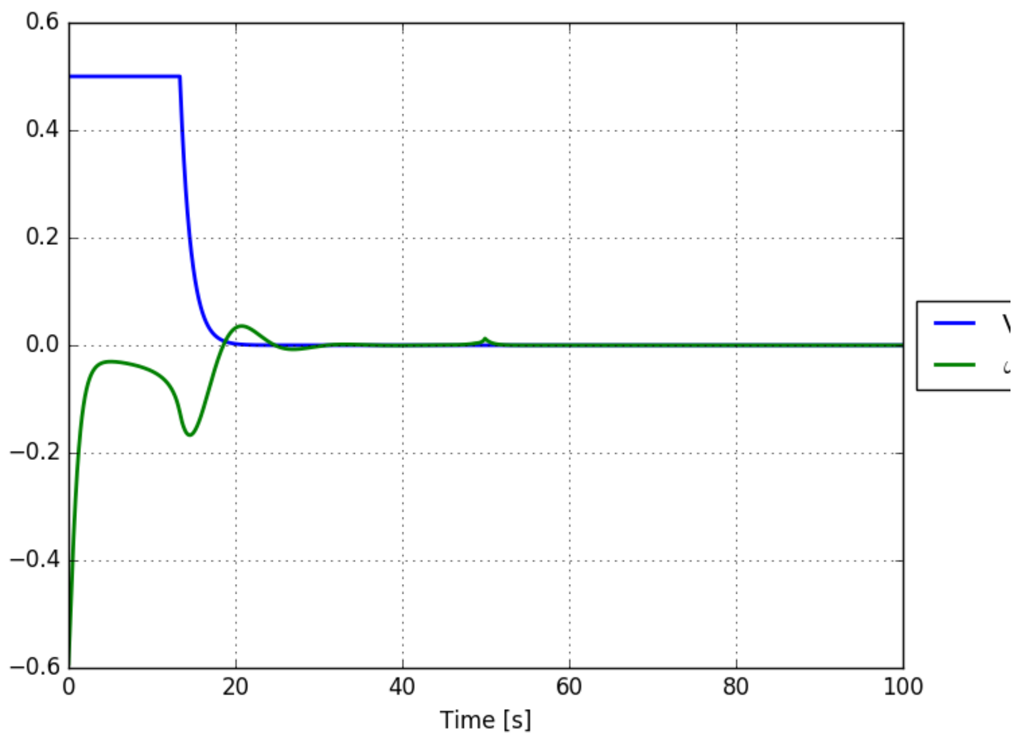

Y
X
6
5
4
3
2
1
0
−1



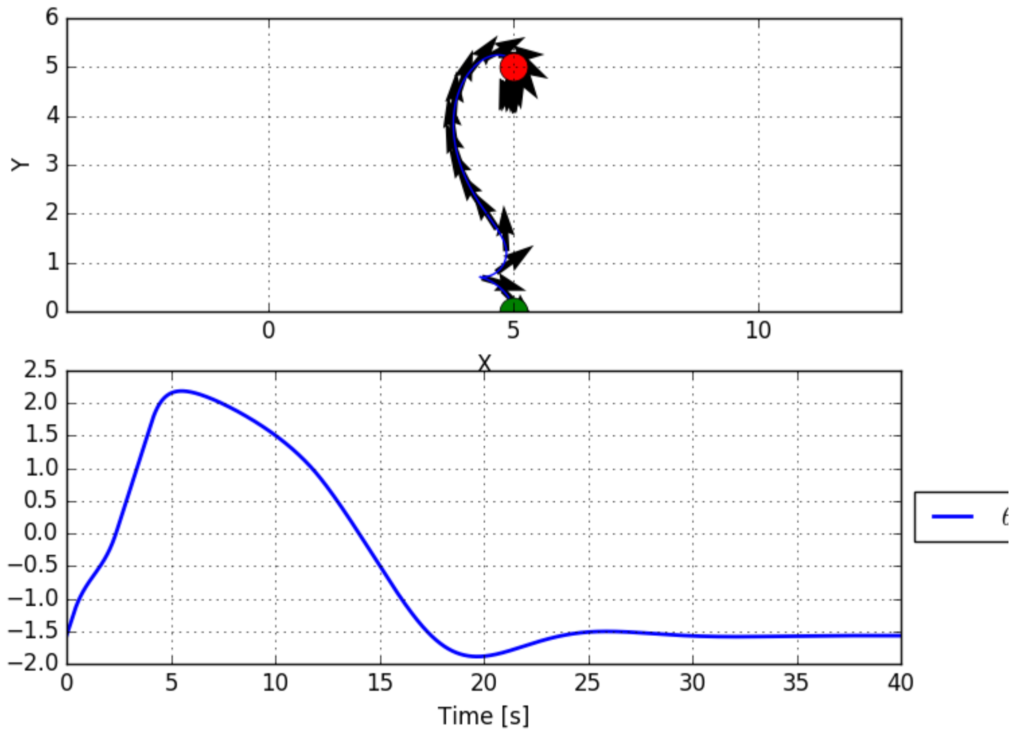
3 Problem 3

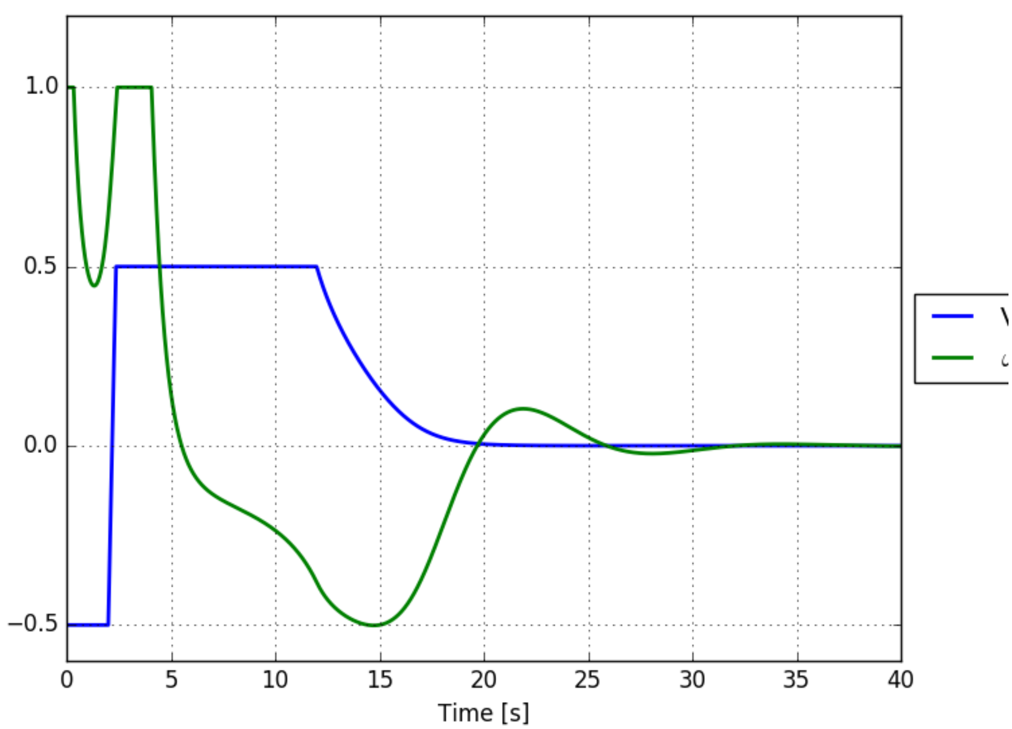
Forward Park



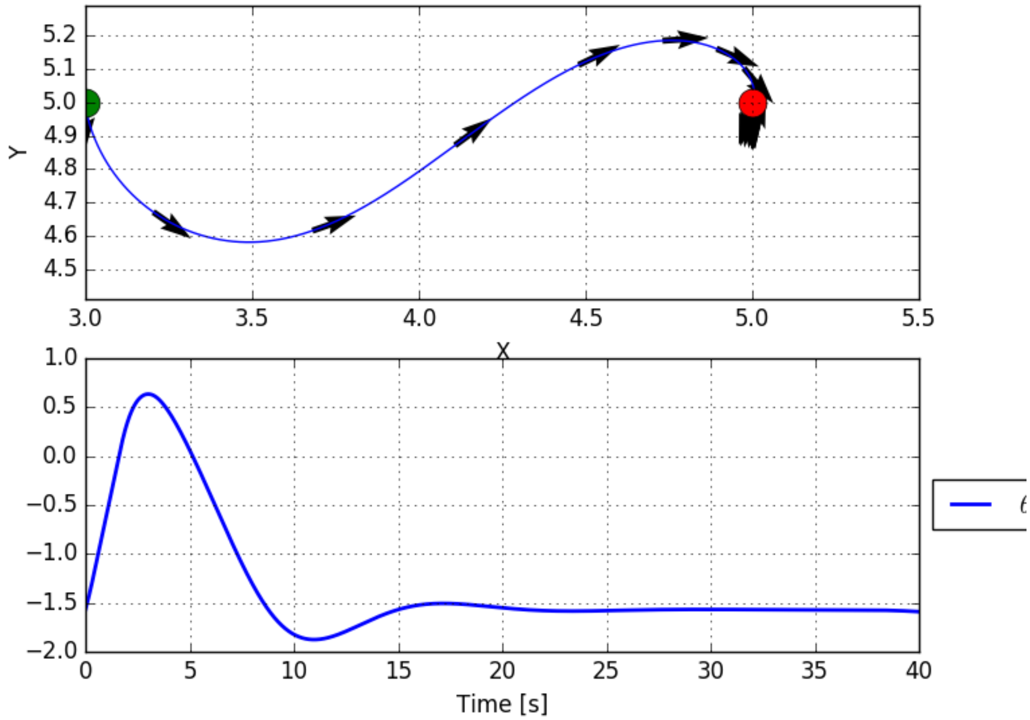


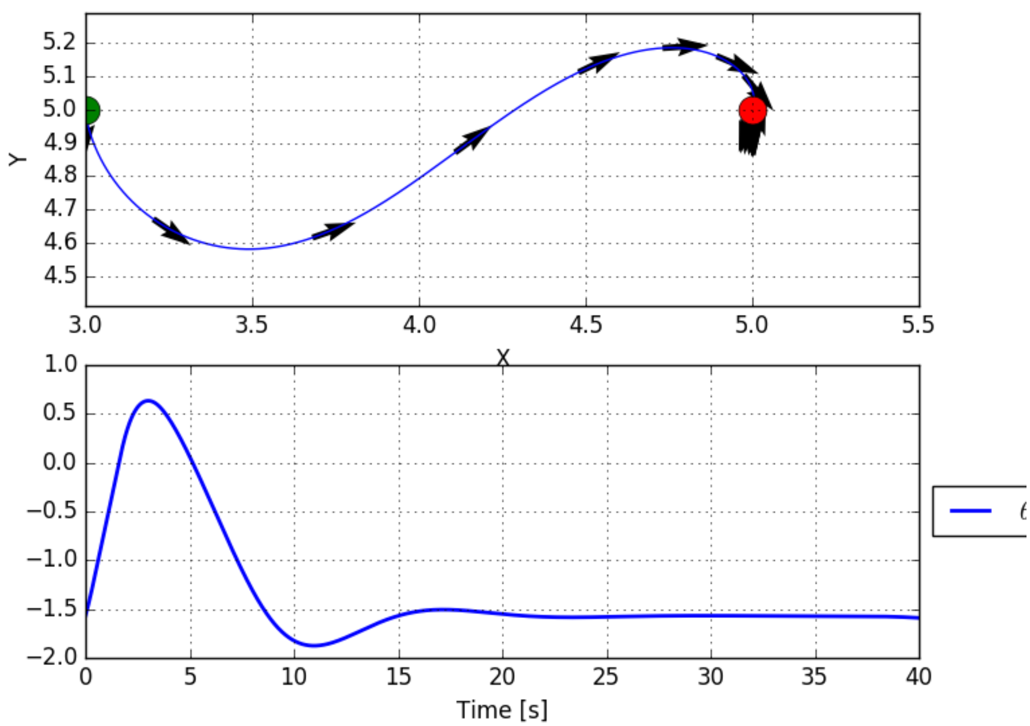
Reverse Park





Parallel Park


Y
5.2
5.1
5.0
4.9
4.8
4.7
4.6
4.5
3.0
3.5
4.0
4.5
5.0
5.5
X
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
5
10
15
20
25
30
35
40
Time [s]


Y
5.2
5.1
5.0
4.9
4.8
4.7
4.6
4.5
3.0
3.5
4.0
4.5
5.0
5.5
X
1.0
0.5
0.0
−0.5
−1.0
−1.5
−2.0
0
5
10
15
20
25
30
35
40
Time [s]

4 Problem 4

i) Write down the form of the dynamic compensator as a set of differential-algebraic equations with input (u1; u2) and output (V; w). Clearly define any internal states that you may need.

Given in the problem statement, we have:

$$u_1 = \ddot{x} + k_{px}(x_d - x) + k_{dx}(\dot{x_d} - x)$$
$$u_2 = \ddot{y} + k_{py}(y_d - y) + k_{dy}(\dot{y_d} - y)$$

We now introduce the matrix equation from problem 2:

$$\begin{bmatrix} a \\ w \end{bmatrix} = \begin{bmatrix} \dot{V} \\ w \end{bmatrix} = \begin{bmatrix} cos(\theta) & -V sin(\theta) \\ sin(\theta) & V cos(\theta) \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

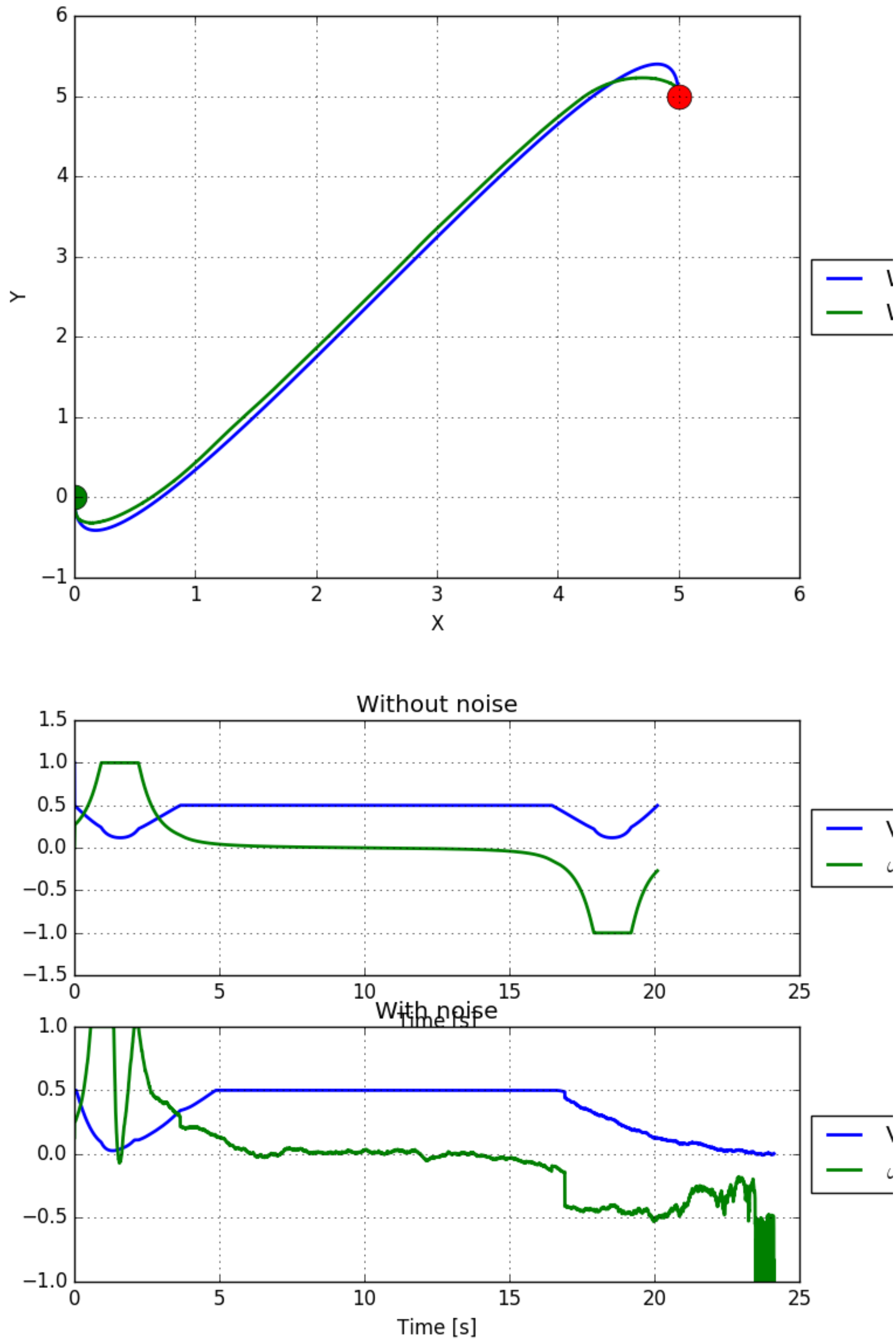
As demonstrated in a reference paper,[1] we create a new variable ζ to represent an intermediate state between control input a and V. We write:

$$V = \zeta$$

Finally we have:

$$\boxed{\begin{bmatrix} a \\ w \end{bmatrix} = \begin{bmatrix} \dot{\zeta} \\ w \end{bmatrix} = \begin{bmatrix} cos(\theta) & -\zeta sin(\theta) \\ sin(\theta) & \zeta cos(\theta) \end{bmatrix}^{-1} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}}$$

[1] http://ieeexplore.ieee.org/stamp/stamp.jsp?arnumber=1058053



Y
X
Without noise
With noise
Time [s]

5 Problem 5

Attached in rosbag files