AA274 (Winter 2017-18): Problem Set 1

Anqi Fu

January 21, 2018

1 Optimal Control

(i) Let $\mathbf{x} = (x, y, \theta)$ denote the robot state and $\mathbf{u} = (V, \omega)$ be the robot control inputs. Our optimal control problem is

$$\begin{array}{ll} \underset{(V,\omega)}{\text{minimize}} & \int_0^{t_f} [\lambda + V(t)^2 + \omega(t)^2] dt \\ \text{subject to} & \dot{x}(t) = V\cos(\theta(t)), \\ & \dot{y}(t) = V\sin(\theta(t)), \\ & \dot{\theta}(t) = \omega(t), \\ & |V(t)| \leq 0.5, \\ & |\omega(t)| \leq 1.0. \end{array} \tag{1}$$

with initial and final conditions

$$\begin{aligned} x(0) = 0, \quad y(0) = 0, \quad \theta(0) = -\pi/2, \\ x(t_f) = 5, \quad y(t_f) = 5, \quad \theta(t_f) = -\pi/2. \end{aligned}$$

Here $\lambda \in \mathbb{R}_+$ is a weighting factor and t_f is free. The Hamiltonian is

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

where $\mathbf{p} = (p_1, p_2, p_3)$ are the Lagrange multipliers associated with $\dot{\mathbf{x}} = (\dot{x}, \dot{y}, \dot{\theta})$. Our optimality conditions are

$$\begin{aligned} \begin{pmatrix} \dot{x}^*(t) \\ \dot{y}^*(t) \\ \dot{\theta}^*(t) \end{pmatrix} &= \begin{pmatrix} V^*(t)\cos(\theta^*(t)) \\ V^*(t)\sin(\theta^*(t)) \\ \omega^*(t) \end{pmatrix} \\ \begin{pmatrix} \dot{p}_1^*(t) \\ \dot{p}_2^*(t) \\ \dot{p}_3^*(t) \end{pmatrix} &= \begin{pmatrix} 0 \\ 0 \\ p_1^*(t)V^*(t)\sin(\theta^*(t)) - p_2^*(t)V^*(t)\cos(\theta^*(t)) \end{pmatrix} \end{aligned} \tag{2}$$

with the additional constraint (for the control inequality constraints)

$$\begin{aligned} &V^*(t)^2 + \omega^*(t)^2 + p_1^*(t)V^*(t)\cos(\theta^*(t)) + p_2^*(t)V^*(t)\sin(\theta^*(t)) + p_3^*(t)\omega^*(t) \\ &\leq V(t)^2 + \omega(t)^2 + p_1^*(t)V(t)\cos(\theta^*(t)) + p_2^*(t)V(t)\sin(\theta^*(t)) + p_3^*(t)\omega(t) \end{aligned} \tag{3}$$

for all $(V(t), \omega(t)) \in \mathbb{R}^2$. Since t_f is free and $\mathbf{x}(t_f)$ is fixed, the boundary conditions amount to

$$\begin{aligned}
&x^*(0) = 0, \quad y^*(0) = 0, \quad \theta^*(0) = -\pi/2, \\
&x^*(t_f) = 5, \quad y^*(t_f) = 5, \quad \theta^*(t_f) = -\pi/2, \\
&\lambda + V^*(t_f)^2 + \omega^*(t_f)^2 + p_1^*(t_f)V^*(t_f)\cos(\theta^*(t_f)) + p_2^*(t_f)V^*(t_f)\sin(\theta^*(t_f)) + p_3^*(t_f)\omega^*(t_f) \\
= &\lambda + V^*(t_f)^2 + \omega^*(t_f)^2 - p_2^*(t_f)V^*(t_f) + p_3^*(t_f)\omega^*(t_f) = 0.
\end{aligned}$$

We make two modifications in order to solve this problem. First, we ignore the control inequality constraints and manually vary λ. This means that (3) is replaced with

$$\begin{aligned}
\begin{pmatrix} 0 \\ 0 \end{pmatrix} &= \begin{pmatrix} 2V^*(t) + p_1^*(t)\cos(\theta^*(t)) + p_2^*(t)\sin(\theta^*(t)) \\ 2\omega^*(t) + p_3^*(t) \end{pmatrix} \\
\Rightarrow \begin{pmatrix} V^*(t) \\ \omega^*(t) \end{pmatrix} &= -\frac{1}{2}\begin{pmatrix} p_1^*(t)\cos(\theta^*(t)) + p_2^*(t)\sin(\theta^*(t)) \\ p_3^*(t) \end{pmatrix}
\end{aligned} \tag{4}$$

Second, we use a change of variables to reformulate the BVP: we rescale the time to $\tau = \frac{t}{t_f} \in [0, 1]$ so derivatives become $\frac{d}{d\tau} := t_f \frac{d}{dt}$, then introduce a dummy state variable r that corresponds to t_f with dynamic $\dot{r} = 0$ and replace all instances of t_f with r. Let $\mathbf{z} = (x, y, \theta, p_1, p_2, p_3, r)$ denote the augmented state vector, then $\frac{d\mathbf{z}}{dt} = \frac{1}{t_f}\frac{d\mathbf{z}}{d\tau} = \frac{1}{r}\frac{d\mathbf{z}}{d\tau}$ and (2) becomes

$$\frac{d\mathbf{z}}{d\tau} = z_7 \begin{pmatrix} V\cos(z_3) \\ V\sin(z_3) \\ \omega \\ 0 \\ 0 \\ V(z_4\sin(z_3) - z_5\cos(z_3)) \\ 0 \end{pmatrix} \tag{5}$$

with boundary conditions

$$\begin{aligned}
&z_1(0) = 0, \quad z_2(0) = 0, \quad z_3(0) = -\pi/2 \\
&z_1(1) = 5, \quad z_2(1) = 5, \quad z_3(1) = -\pi/2 \\
&\lambda + V(1)^2 + \omega(1)^2 - z_5(1)V(1) + z_6(1)\omega(1) = 0
\end{aligned} \tag{6}$$

where we have used (4) to define

$$\begin{pmatrix} V \\ \omega \end{pmatrix} := -\frac{1}{2}\begin{pmatrix} z_4\cos(z_3) + z_5\sin(z_3) \\ z_6 \end{pmatrix} \tag{7}$$

(ii) See submitted code.

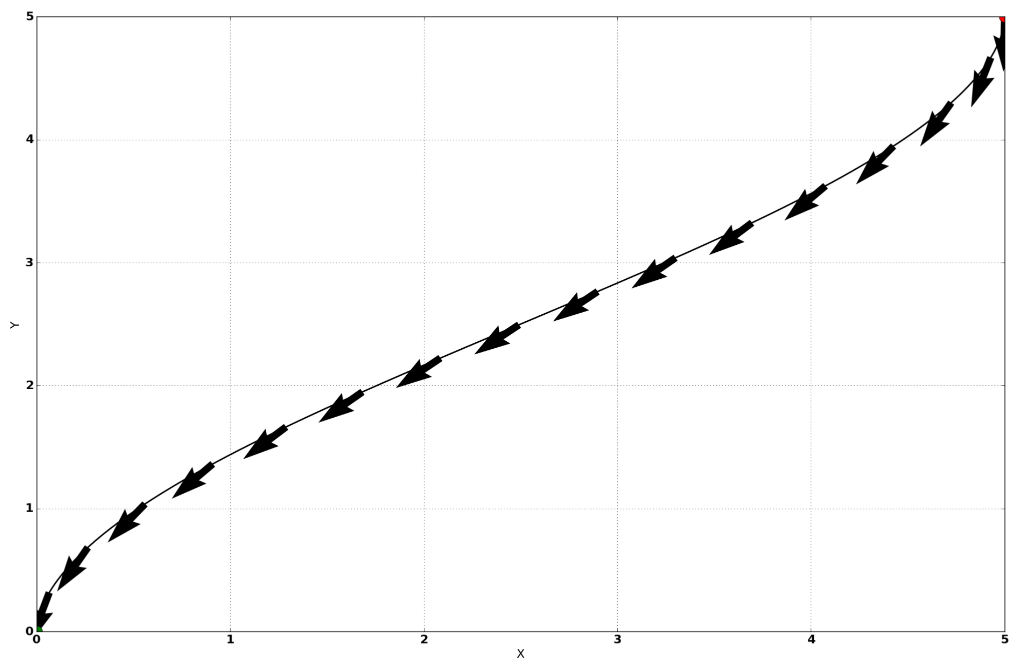
(iii) By choosing the largest feasible λ, we are solving the control problem with the smallest feasible t_f, i.e., our optimal control functions drive the unicycle to its final waypoint in the shortest possible time. This can be seen by separating the objective into

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

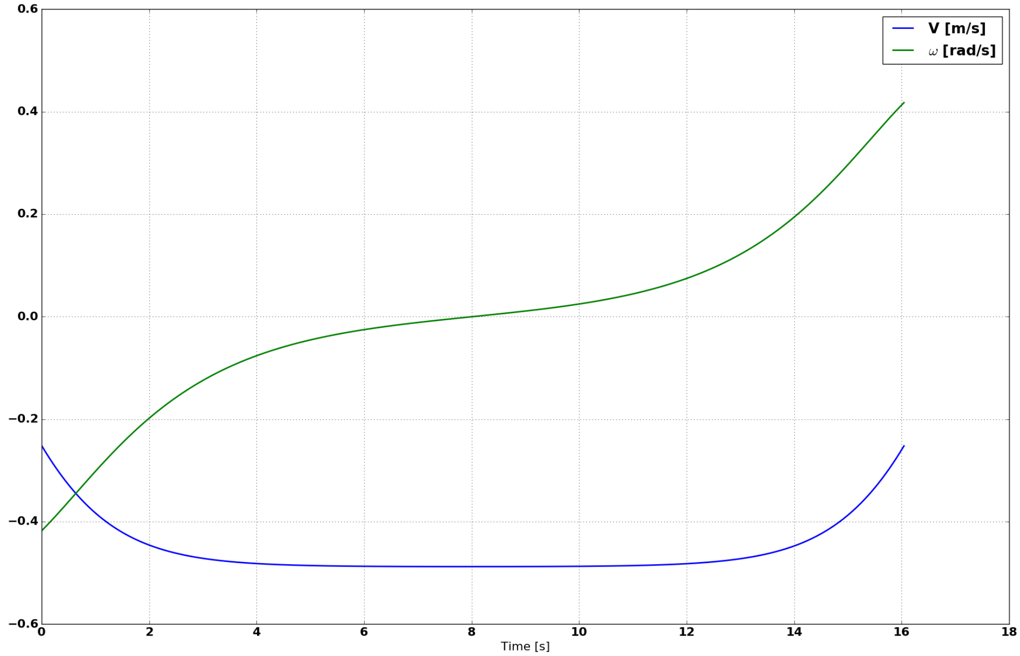
Here $\lambda \geq 0$ acts like a regularization weight: a larger value of λ upweights t_f relative to the rest of the objective, which means (since we are minimizing J) that we place more importance on reducing t_f in the optimization. Selecting the largest feasible λ makes sense because in general, we prefer to travel between the initial and final waypoints as quickly as possible.

(iv) See submitted code. With $\lambda = 0.238$ and solution guess $\mathbf{z}_0 = (0, 0, -\pi/2, 1, 1, 0, 10)$, we get $V < 0$ and the robot backs up from $(0,0)$ to $(5,5)$. Its trajectory and control histories are shown below.

Trajectory $(x(t), y(t)))$

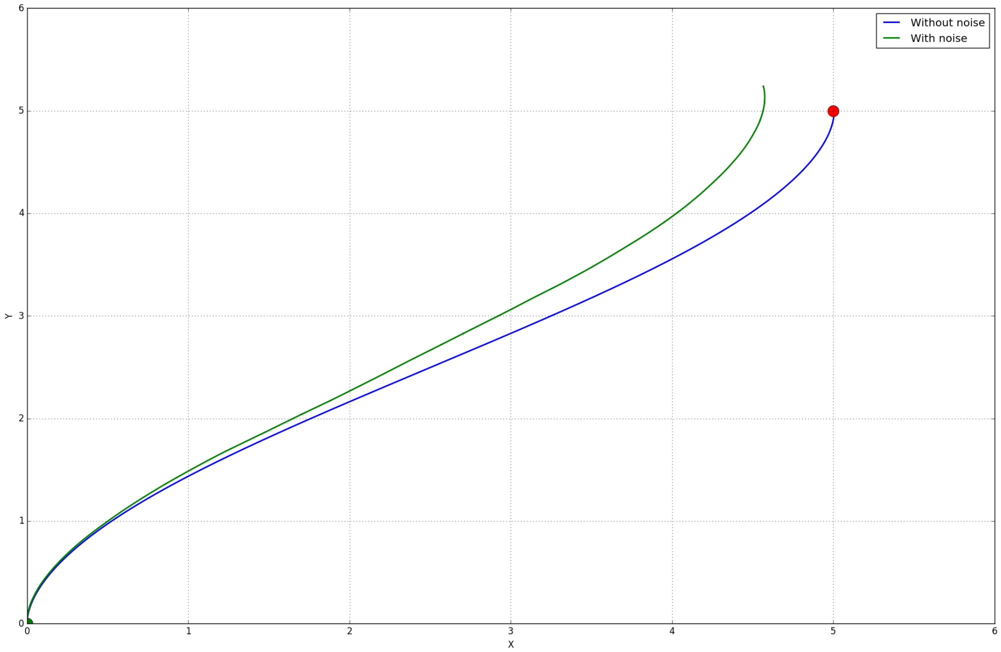


History of V and ω

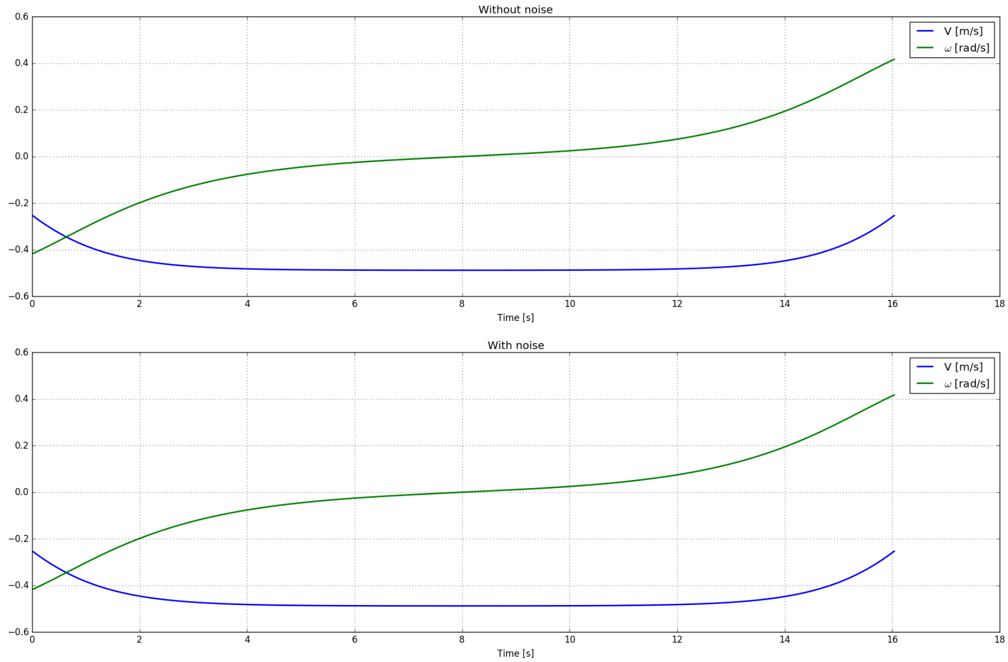


(v) See submitted code.

Trajectory $(x(t), y(t))$



History of V and ω



2 Differential Flatness

(i) Let $\psi_1(t) = 1, \psi_2(t) = t, \psi_3(t) = t^2$, and $\psi_4(t) = t^3$, then $n = 4$ and

$$x(t) = \sum_{i=1}^{n} x_i \psi_i(t) = x_1 + x_2 t + x_3 t^2 + x_4 t^3$$

$$y(t) = \sum_{i=1}^{n} y_i \psi_i(t) = y_1 + y_2 t + y_3 t^2 + y_4 t^3$$

$$\dot{x}(t) = \sum_{i=1}^{n} x_i \dot{\psi}_i(t) = x_2 + 2x_3 t + 3x_4 t^2$$

$$\dot{y}(t) = \sum_{i=1}^{n} y_i \dot{\psi}_i(t) = y_2 + 2y_3 t + 3y_4 t^2$$

so the initial and final conditions can be expressed as

$$
\begin{aligned}
x(0) &= 0 = x_1 \\
y(0) &= 0 = y_1 \\
x(t_f) &= 5 = x_1 + x_2 t_f + x_3 t_f^2 + x_4 t_f^3 \\
y(t_f) &= 5 = y_1 + y_2 t_f + y_3 t_f^2 + y_4 t_f^3 \\
\dot{x}(0) &= V(0)\cos(\theta(0)) = 0.5\cos(-\pi/2) = 0 = x_2 \\
\dot{y}(0) &= V(0)\sin(\theta(0)) = 0.5\sin(-\pi/2) = -0.5 = y_2 \\
\dot{x}(t_f) &= V(t_f)\cos(\theta(t_f)) = 0.5\cos(-\pi/2) = 0 = x_2 + 2x_3 t_f + 3x_4 t_f^2 \\
\dot{y}(t_f) &= V(t_f)\sin(\theta(t_f)) = 0.5\sin(-\pi/2) = -0.5 = y_2 + 2y_3 t_f + 3y_4 t_f^2
\end{aligned}
$$

or more succinctly in matrix form,

$$
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 \end{pmatrix}
\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}
=
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 15 & 225 & 3375 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 30 & 675 \end{pmatrix}
\begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix}
=
\begin{pmatrix} 0 \\ 5 \\ 0 \\ 0 \end{pmatrix}
=
\begin{pmatrix} x(0) \\ x(t_f) \\ \dot{x}(0) \\ \dot{x}(t_f) \end{pmatrix}
$$

$$
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & t_f & t_f^2 & t_f^3 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 2t_f & 3t_f^2 \end{pmatrix}
\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}
=
\begin{pmatrix} 1 & 0 & 0 & 0 \\ 1 & 15 & 225 & 3375 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 30 & 675 \end{pmatrix}
\begin{pmatrix} y_1 \\ y_2 \\ y_3 \\ y_4 \end{pmatrix}
=
\begin{pmatrix} 0 \\ 5 \\ -0.5 \\ -0.5 \end{pmatrix}
=
\begin{pmatrix} y(0) \\ y(t_f) \\ \dot{y}(0) \\ \dot{y}(t_f) \end{pmatrix}
$$

We cannot set $V(t_f) = 0$ because then J would be singular at time t_f, and we could not recover the flat outputs from the virtual control inputs.

(ii) See submitted code. After solving for x_i, y_i for $i = 1, \ldots, n$, we can recover

$$
\begin{pmatrix} x(t) \\ y(t) \\ \dot{x}(t) \\ \dot{y}(t) \\ \ddot{x}(t) \\ \ddot{y}(t) \end{pmatrix}
=
\begin{pmatrix} x_1 & x_2 & x_3 & x_4 \\ y_1 & y_2 & y_3 & y_4 \\ x_2 & 2x_3 & 3x_4 & 0 \\ y_2 & 2y_3 & 3y_4 & 0 \\ 2x_3 & 6x_4 & 0 & 0 \\ 2y_3 & 6y_4 & 0 & 0 \end{pmatrix}
\begin{pmatrix} 1 \\ t \\ t^2 \\ t^3 \end{pmatrix}
$$

along with the alignment angle

$$
\frac{\dot{y}(t)}{\dot{x}(t)} = \frac{V\sin(\theta(t))}{V\cos(\theta(t))} = \tan(\theta(t)) \Rightarrow \theta(t) = \arctan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right).
$$

Since $V > 0$, we take the positive root of

$$
\dot{x}(t)^2 + \dot{y}(t)^2 = V^2\cos^2(\theta(t)) + V^2\sin^2(\theta(t)) = V^2 \Rightarrow V(t) = \sqrt{\dot{x}(t)^2 + \dot{y}(t)^2},
$$

then the matrix J is invertible, so we can calculate

$$J^{-1} = \begin{pmatrix} \cos(\theta) & -V\sin(\theta) \\ \sin(\theta) & V\cos(\theta) \end{pmatrix}^{-1} = \frac{1}{V} \begin{pmatrix} V\cos(\theta) & V\sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix}$$

$$\begin{pmatrix} a \\ \omega \end{pmatrix} = J^{-1} \begin{pmatrix} \ddot{x}(t) \\ \ddot{y}(t) \end{pmatrix} = \begin{pmatrix} \ddot{x}(t)\cos(\theta) + \ddot{y}(t)\sin(\theta) \\ \frac{1}{V}(-\ddot{x}(t)\sin(\theta) + \ddot{y}(t)\cos(\theta)) \end{pmatrix}.$$

Putting this all together, our state-trajectory is

$$\begin{aligned} x(t) &= x_1 + x_2 t + x_3 t^2 + x_4 t^3 \\ y(t) &= y_1 + y_2 t + y_3 t^2 + y_4 t^3 \\ \theta(t) &= \arctan\left(\frac{y_2 + 2y_3 t + 3y_4 t^2}{x_2 + 2x_3 t + 3x_4 t^2}\right) \end{aligned}$$

and our control history is

$$\begin{aligned} V(t) &= \sqrt{(x_2 + 2x_3 t + 3x_4 t^2)^2 + (y_2 + 2y_3 t + 3y_4 t^2)^2} \\ \omega(t) &= \frac{1}{V(t)}\left(-(2x_3 + 6x_4 t)\sin(\theta(t)) + (2y_3 + 6y_4 t)\cos(\theta(t))\right) \end{aligned}$$

(iii) See submitted code. Given $\dot{s}(t) = V(t)$ with $s(0) = 0$, we can integrate to get the path parameter

$$s(t) = \int_0^t V(t')dt' - V(0)$$

We wish to find an alternative velocity control $\tilde{V}(s)$ that satisfies the control saturation constraints. Then, the corresponding angular velocity control and time history are

$$\tilde{\omega}(s) = \omega(s)\frac{\tilde{V}(s)}{V(s)}, \quad \tau(s) = \int_0^s \frac{ds'}{\tilde{V}(s')}.$$

Combining the constraint $|\tilde{V}(s)| \leq 0.5$ m/s with

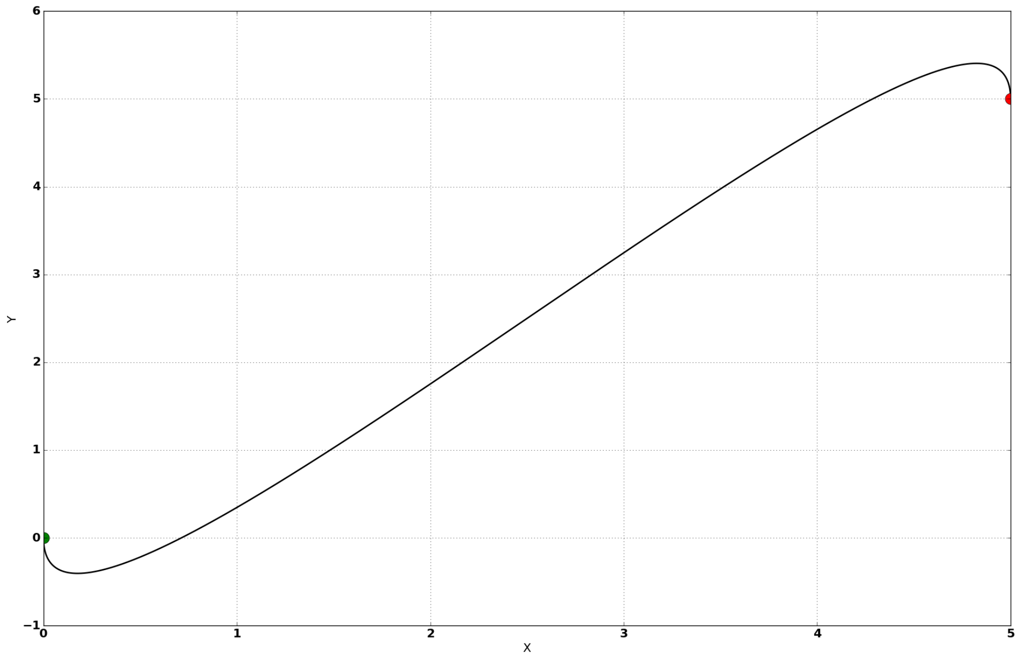
$$|\tilde{\omega}(s)| = \left|\omega(s)\frac{\tilde{V}(s)}{V(s)}\right| \leq 1 \text{ rad/s} \Rightarrow |\tilde{V}(s)| \leq \frac{V(s)}{|\omega(s)|} \text{ for } \omega(s) \neq 0$$

we see that a viable choice is

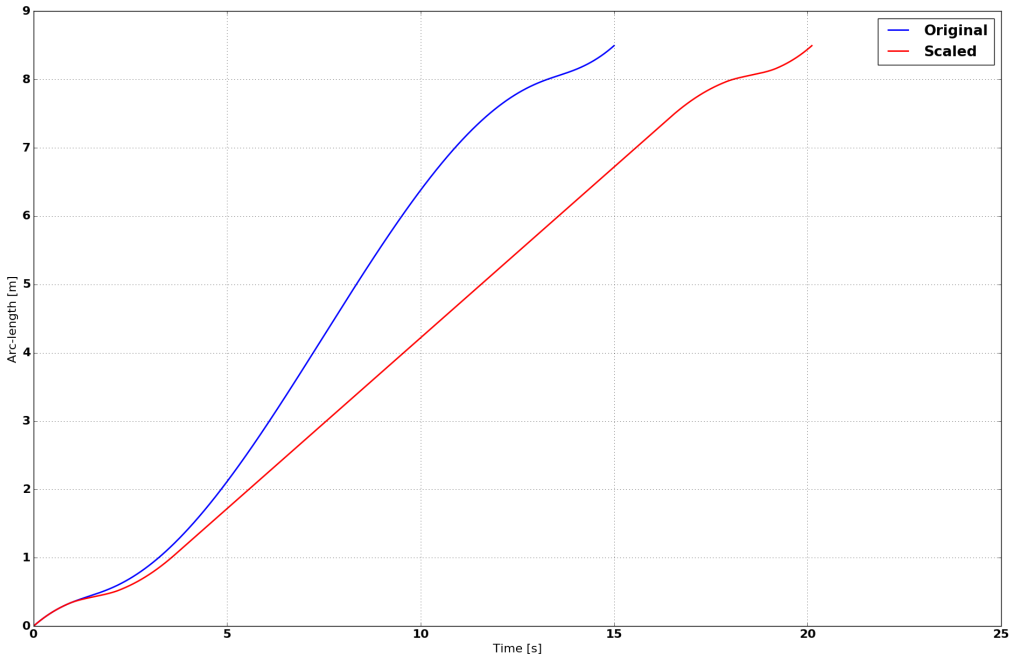
$$\tilde{V}(s) = \begin{cases} \min(V(s), 0.5) & \text{if } \omega(s) = 0 \\ \min\left(V(s), \frac{V(s)}{|\omega(s)|}, 0.5\right) & \text{if } \omega(s) \neq 0 \end{cases}$$

(iv) See submitted code.

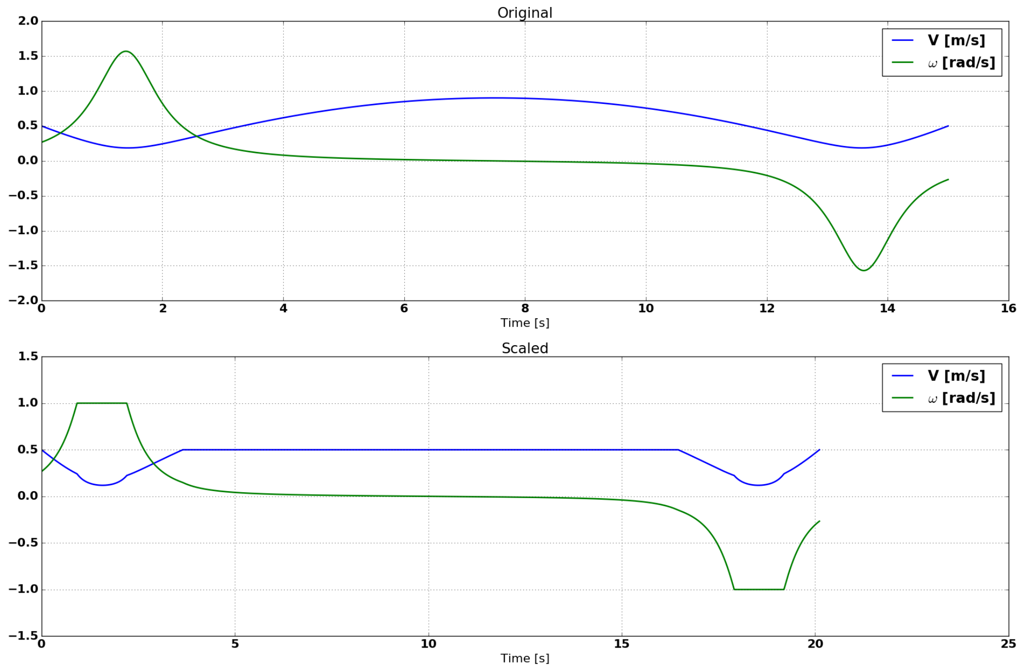
Trajectory $(x(t), y(t))$



Arc-length $s(t)$

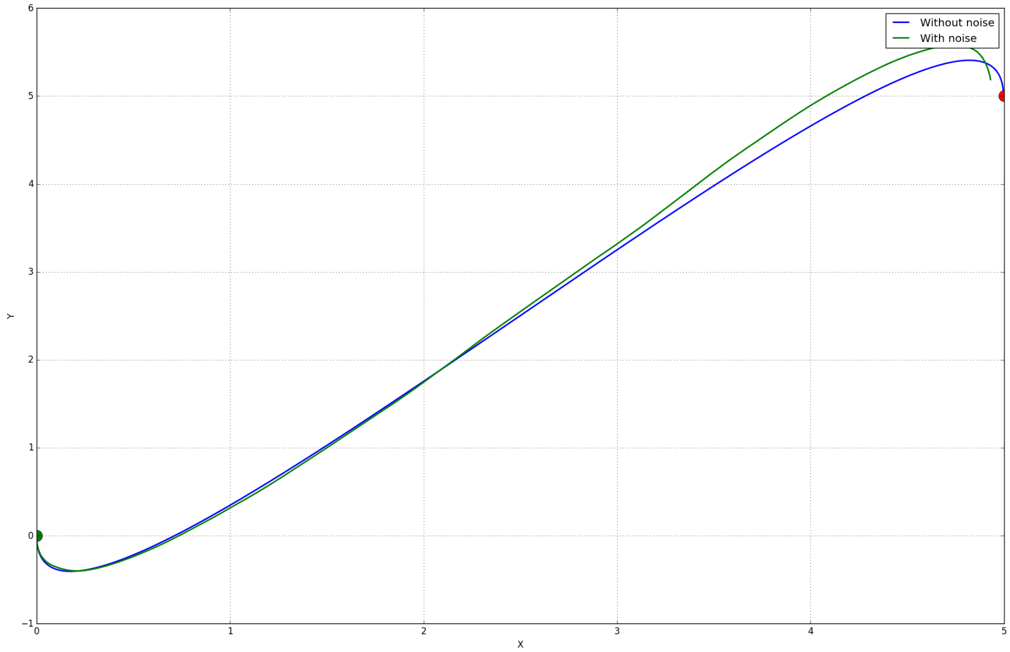


History of V and ω

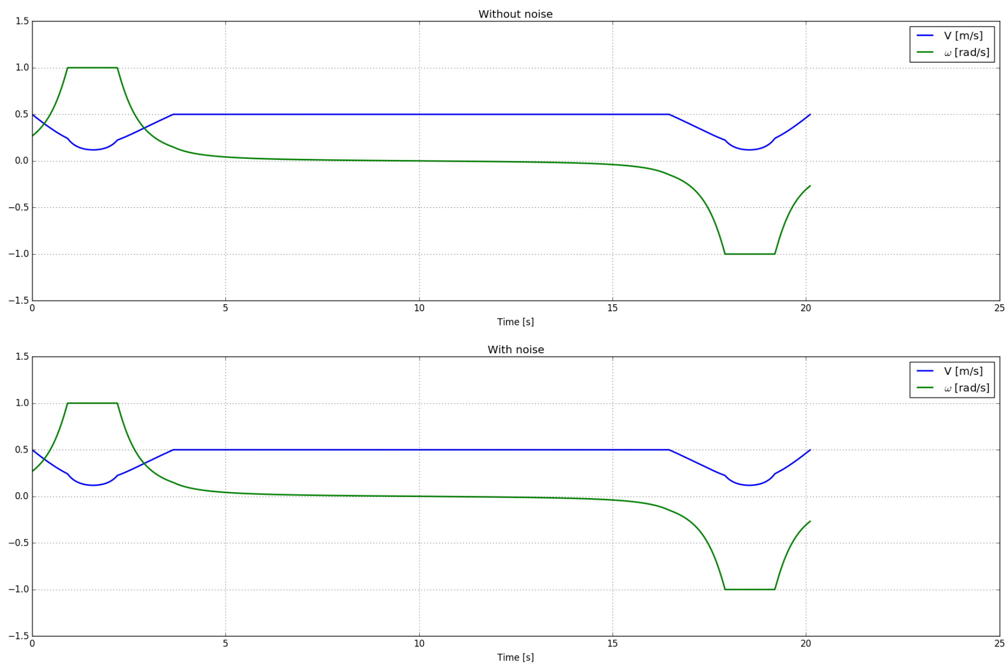


(v) See submitted code.

Trajectory $(x(t), y(t))$



History of V and ω



3 Closed-loop Control I

(i) See submitted code. We are given the control law

$$V = k_1 \rho \cos(\alpha)$$
$$\omega = k_2 \alpha + k_1 \frac{\sin(\alpha)\cos(\alpha)}{\alpha}(\alpha + k_3 \delta)$$

for the ODE system

$$\dot{\rho}(t) = -V(t)\cos(\alpha(t))$$
$$\dot{\alpha}(t) = V(t)\frac{\sin(\alpha(t))}{\rho(t)} - \omega(t)$$
$$\dot{\delta}(t) = V(t)\frac{\sin(\alpha(t))}{\rho(t)}$$

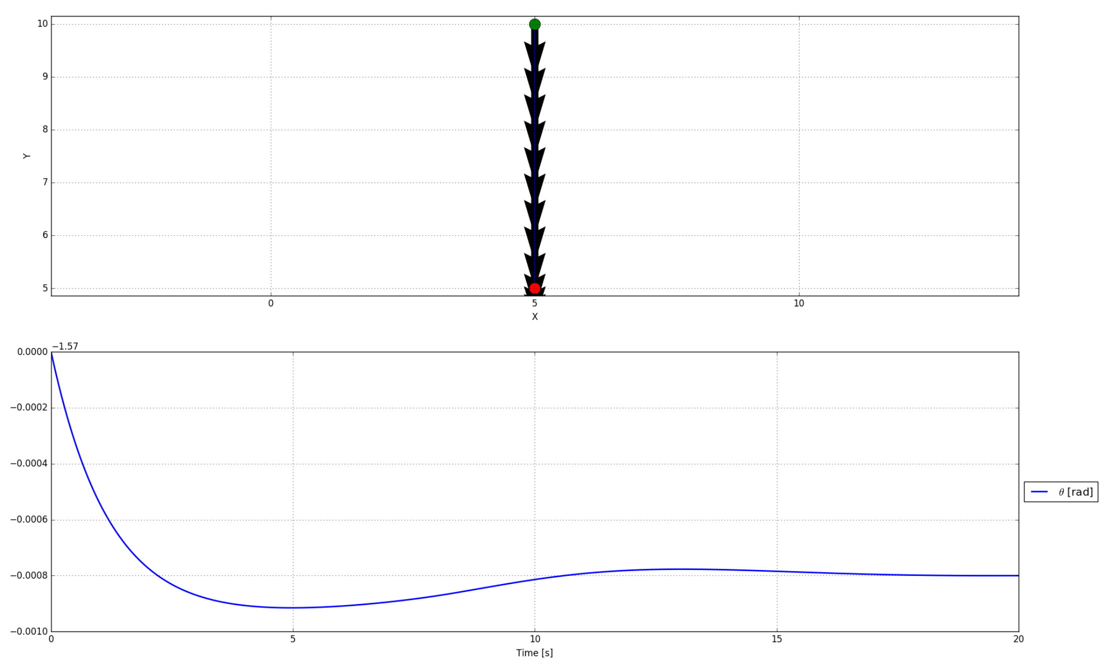
where $k_1, k_2, k_3 > 0$ are constants. From Homework 1, Figure 2, we readily see that for a starting position (x, y, θ) and final position (x_g, y_g, θ_g), the incremental change in polar coordinates is

$$\rho = \sqrt{(x_g - x)^2 + (y_g - y)^2}$$
$$\tan(\alpha + \theta) = \frac{y_g - y}{x_g - x} \Rightarrow \alpha = \arctan\left(\frac{y_g - y}{x_g - x}\right) - \theta$$
$$\alpha + \theta = \delta + \theta_g \Rightarrow \delta = \alpha + \theta - \theta_g$$

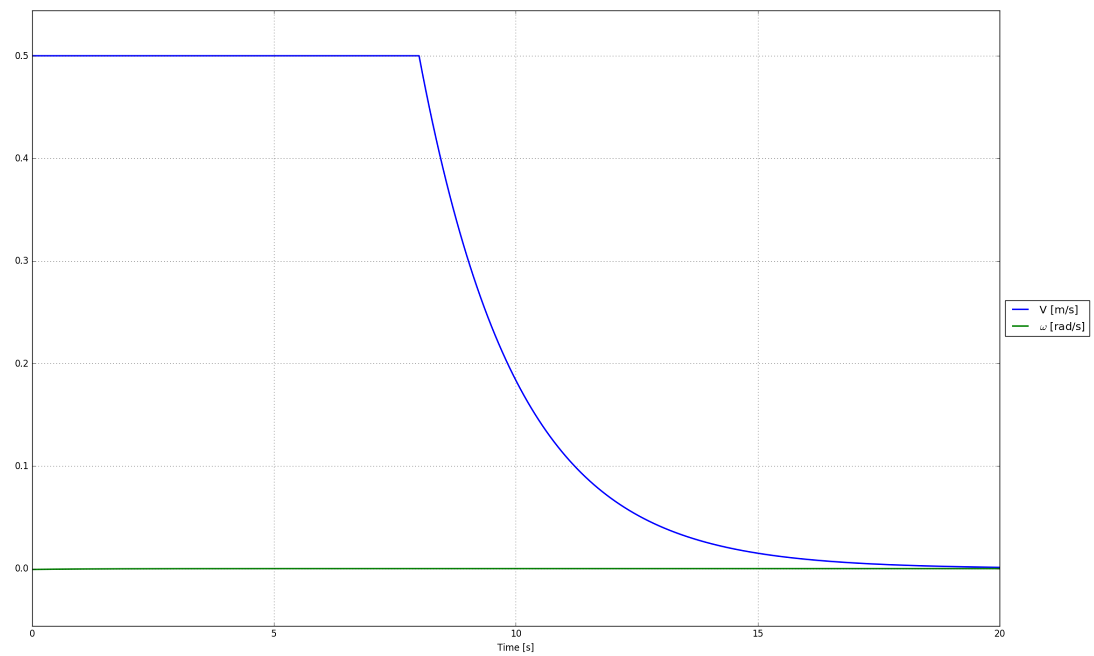
(ii) See submitted code. The parameters used were

$$(x_0, y_0, \theta_0, t_f) = \begin{cases} (5, 10, -\pi/2, 20) & \text{for forward parking} \\ (5, 0, -\pi/2, 12) & \text{for reverse parking} \\ (0, 5, -\pi/2, 18) & \text{for parallel parking} \end{cases}$$
$$(k_1, k_2, k_3) = \begin{cases} (0.5, 0.5, 1.2) & \text{for forward parking} \\ (1.45, 0.01, 1.45) & \text{for reverse parking} \\ (0.5, 0.5, 1.2) & \text{for parallel parking} \end{cases}$$

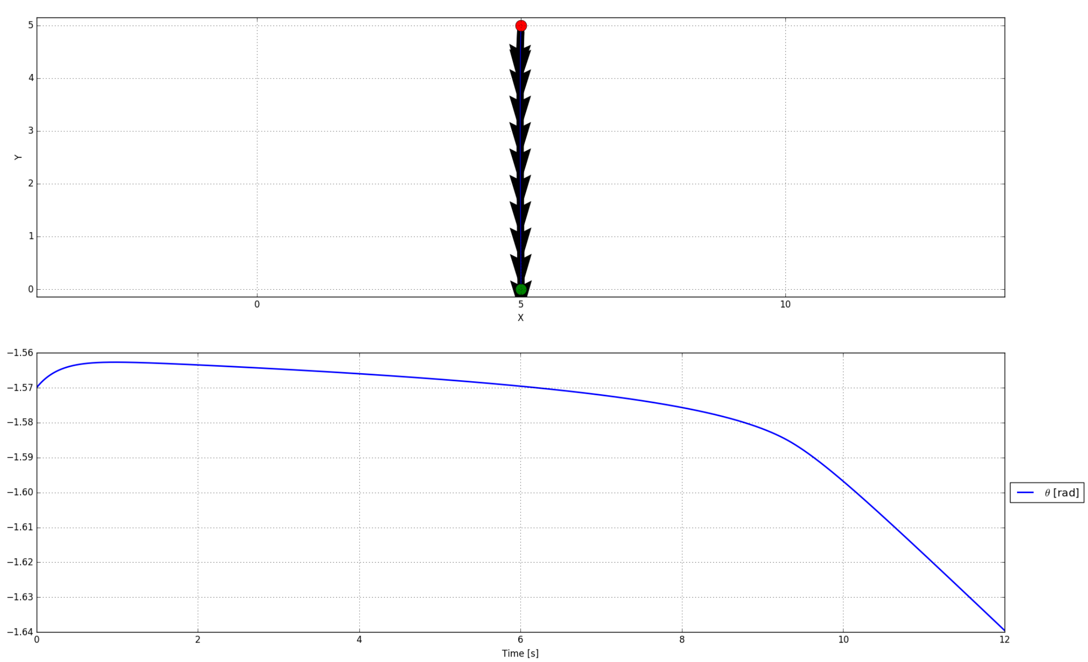
Forward Trajectory



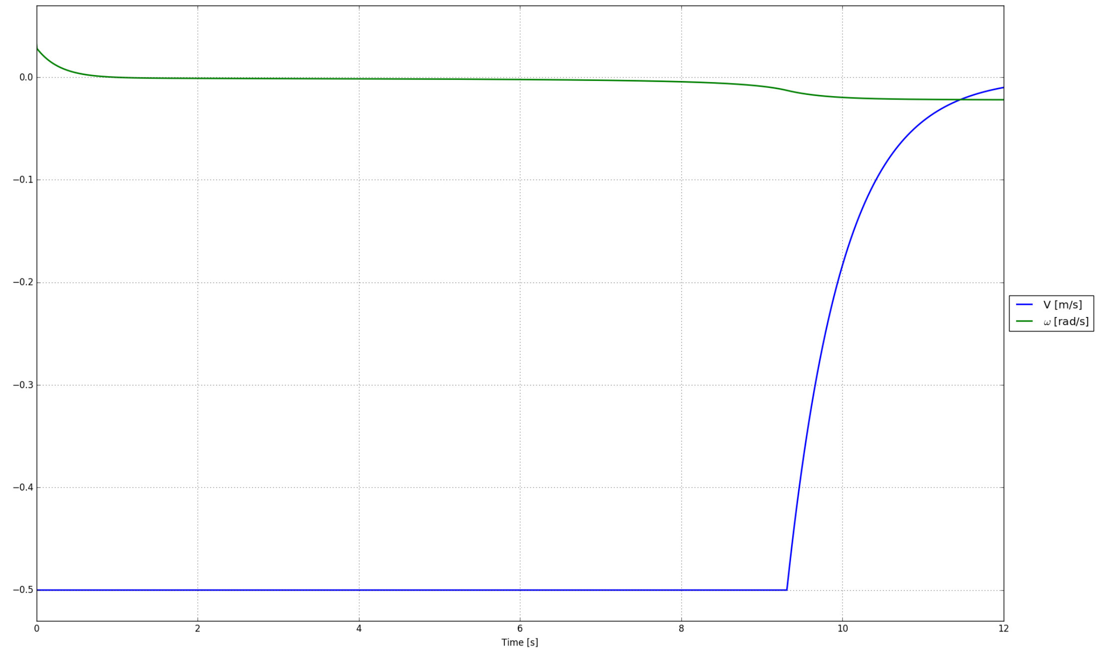
Forward Control



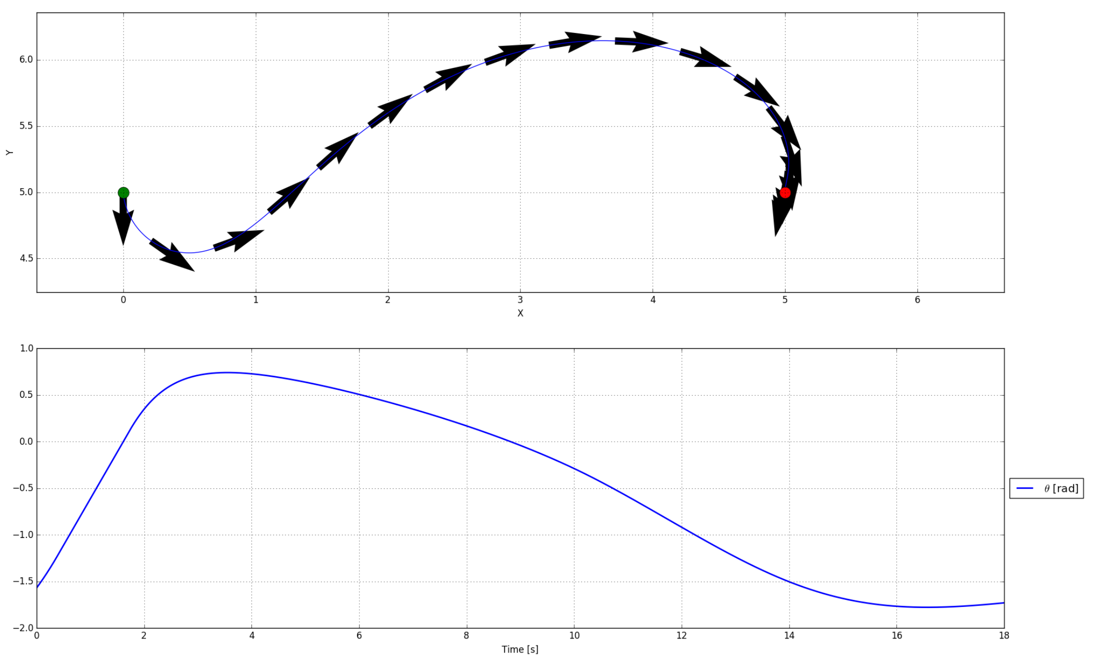
Reverse Trajectory

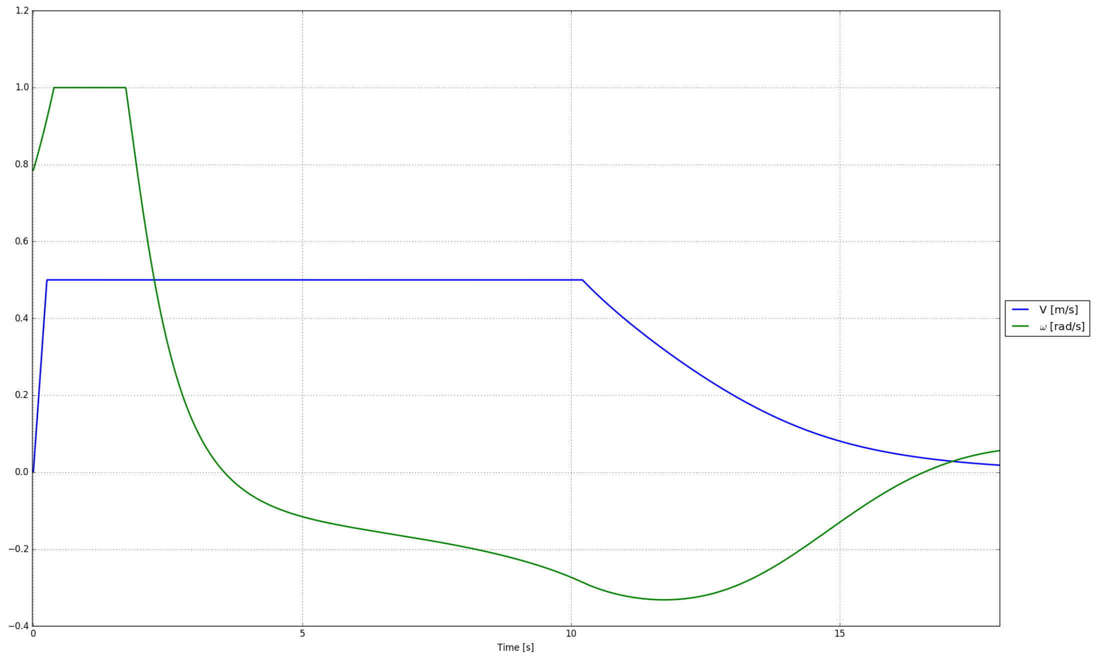


Reverse Control



Parallel Trajectory





4 Closed-loop Control II

(i) We wish to implement the virtual control law

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

with control gains $k_{px}, k_{py}, k_{dx}, k_{dy} > 0$ and desired (differential flatness) trajectory (x_d, y_d). In Problem 2, part (ii), we showed that for a given state $\mathbf{x} = (x, y, \theta)$,

$$\begin{pmatrix} a \\ \omega \end{pmatrix} = \begin{pmatrix} \dot{V} \\ \omega \end{pmatrix} = \begin{pmatrix} u_1 \cos(\theta) + u_2 \sin(\theta) \\ \frac{1}{V}(-u_1 \sin(\theta) + u_2 \cos(\theta)) \end{pmatrix} \tag{8}$$

where the virtual controls are $(u_1, u_2) = (\ddot{x}, \ddot{y})$.

(ii) See submitted code. We assume the virtual control law is implemented at a high enough frequency that the state can be treated as constant. Furthermore, at each step t, we consider the current velocity to be that commanded at $t-1$, i.e.,

$$\dot{x}_t = V_{t-1}\cos(\theta), \quad \dot{y}_t = V_{t-1}\sin(\theta).$$

Using this, we can calculate (u_1, u_2) and plug into (8) to get the current acceleration, $a_t = \dot{V}_t$. We then apply the Euler method to update

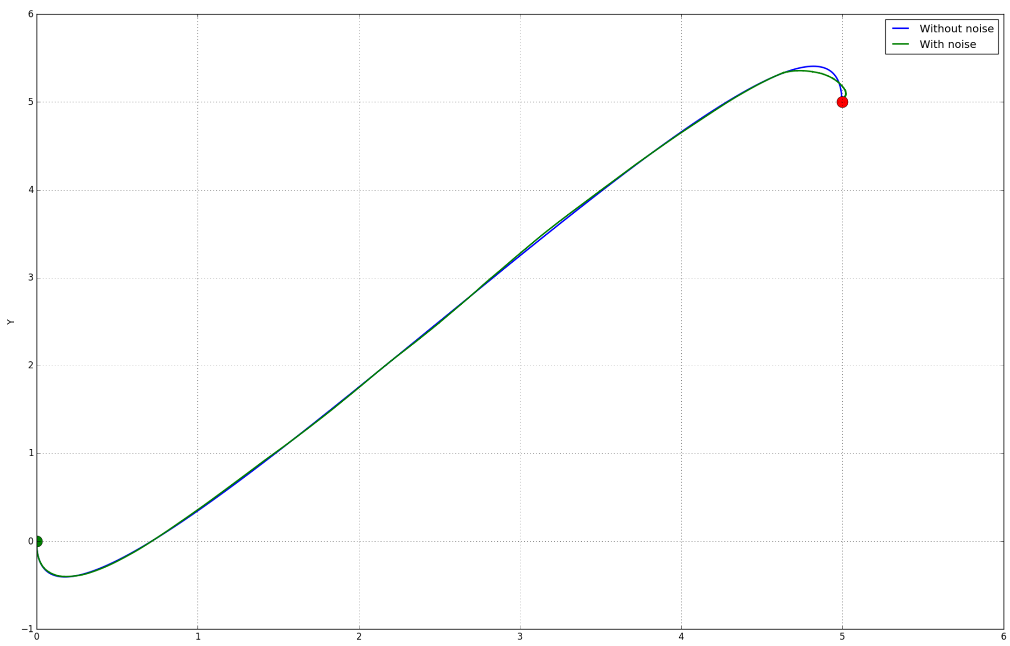
$$V_t = V_{t-1} + (dt)\dot{V}_t \quad \text{where } dt \text{ is the timestep.}$$

If $V_t \approx 0$, we reset it to the nominal (desired) velocity $V_d = \sqrt{\dot{x}_d^2 + \dot{y}_d^2}$ according to the robot kinematic model. Finally, we calculate ω from (8).

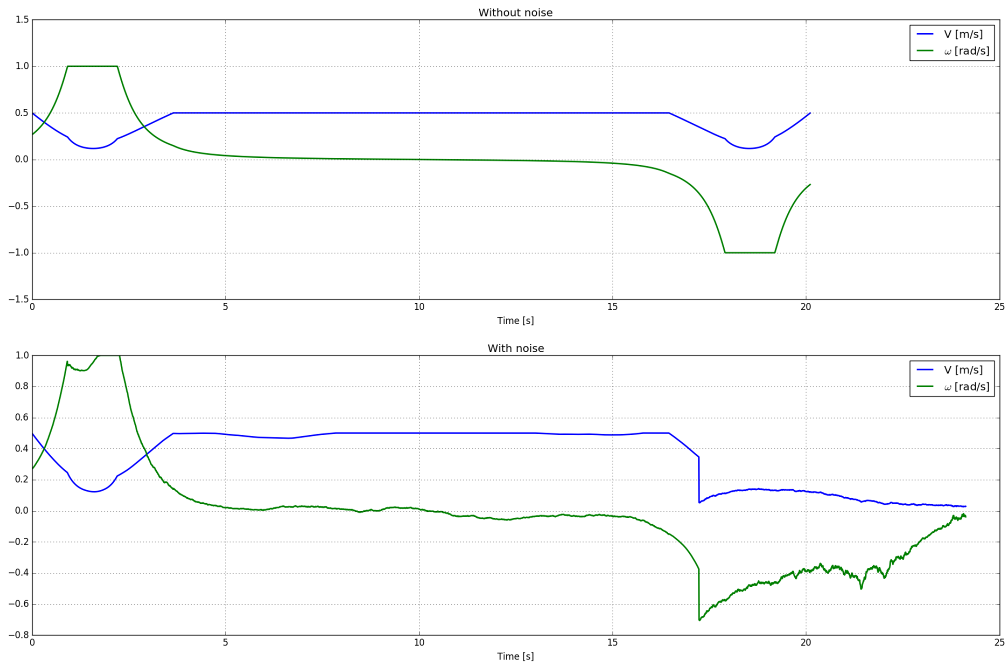
(iii) See submitted code.

(iv) See submitted code. For this simulation, we set $(k_{px}, k_{py}, k_{dx}, k_{dy}) = (1, 1, 0.5, 0.5)$ and reset to the nominal velocity whenever $|V_t| \leq \epsilon = 10^{-5}$. The following plots were made with initial conditions $(x_0, y_0, \theta_0) = (0, 0, -\pi/2)$.

Trajectory $(x(t), y(t))$



History of V and ω



5 Robot Operating System

(i) See submitted code.

(ii) `rosbag play filename.bag`

(iii) See submitted code.