1 [30pt] Parameter Estimation

Consider a velocity-controlled 2D vehicle with an unknown goal point x_G and control gains a and b:

$$x_{k+1} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_k + \begin{bmatrix} T \\ T \end{bmatrix} (u_k + w_k) \tag{1}$$

$$u_k = \begin{bmatrix} a & b \end{bmatrix} (x_G - x_k) \tag{2}$$

where $T = 0.1$ is the sampling time and $w_k \sim \mathcal{N}(0, 0.0001)$ is the process noise that is assumed to be Gaussian white i.i.d. The observer can observe the state sequence $\{x_k\}_{0:N}$. The goal is to estimate the goal position $x_G \in \mathbb{R}^2$ as well as the control gains $a \in \mathbb{R}$ and $b \in \mathbb{R}$ from the observation. Assume the ground truth values are: $x_0 = [0; 0]$, $x_G = [10; 10]$, $a = b = 0.1$. Let $N = 100$.

1.1 [10pt] Assume a and b are known. We need to estimate x_G. Apply one parameter estimation method (e.g., KF, EKF, UKF, RLS, SGD, etc.) to estimate x_G. Write down all equations and plot the trajectories for x_k and $\hat{x}_G$.

1.2 [10pt] Assume x_G is known. We need to estimate a and b. Apply one parameter estimation method (e.g., KF, EKF, UKF, RLS, SGD, etc.) to estimate a and b. Write down all equations and plot the trajectories for x_k, $\hat{a}$, and $\hat{b}$.

1.3 [10pt] Now let us estimate a, b and x_G simultaneously. Apply one parameter estimation method (e.g., KF, EKF, UKF, RLS, SGD, etc.) to estimate x_G. Write down all equations and plot the trajectories for x_k, $\hat{a}$, $\hat{b}$ and $\hat{x}_G$.

2 [40pt] Value Approximation

Consider the unicycle in HW1 with state $x = [p_x, p_y, v, \theta]^T \in \mathbb{R}^4$ and control $u = [\dot{v}, \dot{\theta}]^T \in \mathbb{R}^2$. The discrete time dynamics are

$$x_{k+1} = x_k + \dot{x}_k T + w_k \tag{3}$$

$$\dot{x} = \begin{bmatrix} v\cos\theta \\ v\sin\theta \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} u \tag{4}$$

$$\tag{5}$$

where $T = 0.1s$ is the sampling time and $w_k \sim \mathcal{N}(0, 0.0001 I_4)$ is the process noise that is assumed to be Gaussian white i.i.d.

We need to solve for a control policy $u = \pi(x)$ that moves the vehicle from an initial state $x_0 = [p_{x,0}, p_{y,0}, 0, 0]^T$ to a target state $x_G = [p_{x,G}, p_{y,G}, 0, \theta_G]^T$. The run-time cost is $l(x_t, u_t) = \frac{1}{2}(x_t - x_G)^T Q (x_t - x_G) + \frac{1}{2} u_t^T R u_t$ where $Q \in \mathbb{R}^{4\times4}$ and $R \in \mathbb{R}^{2\times2}$ are both positive definite. Let the discount be $\delta = 0.9$. The problem terminates when the goal state is reached, i.e., $\|x_k - x_G\| \leq 0.01$. Let $x_0 = [0, 0, 0, 0]^T$, $x_G = [10, 10, 0, \pi/2]^T$, $Q = I$, $R = 0.1I$.

2.1 [5pt] Pick a parameterized value function for the problem and justify your parameterization. Write down the gradient of the value function with respect to the parameters. Write down the associated policy.

2.2 [10pt] Learn the parameterized value function and the associated policy $u = \pi(x)$ using the gradient Monte Carlo algorithm. Plot the trajectories in different episodes. Run the learning algorithm multiple rounds and plot the mean and variance of the reward in learning.

2.3 [10pt] Learn the parameterized value function and the associated policy $u = \pi(x)$ using episodic Semi-Gradient Sarsa. Plot the trajectories in different episodes. Run the learning algorithm multiple rounds and plot the mean and variance of the reward in learning.

2.4 [10pt] Learn the parameterized value function and the associated policy $u = \pi(x)$ using episodic Semi-Gradient Q-Learning. Plot the trajectories in different episodes. Run the learning algorithm multiple rounds and plot the mean and variance of the reward in learning.

2.5 [5pt] Compare the results from 2.2 to 2.4, what conclusion can you draw?

3 [30pt] Policy Gradient

Consider the problem in Question 2. Let us now use policy gradient to solve the problem.

3.1 [5pt] Pick a parameterized policy function for the problem and justify your parameterization. Write down the gradient of the policy (π) as well as the gradient of the ln policy ($\ln \pi$) with respect to the parameters.

3.2 [10pt] Learn the parameterized policy using REINFORCE. Plot the trajectories in different episodes. Run the learning algorithm multiple rounds and plot the mean and variance of the reward in learning.

3.3 [10pt] Learn the parameterized policy using actor critic. Plot the trajectories in different episodes. Run the learning algorithm multiple rounds and plot the mean and variance of the reward in learning.

3.4 [5pt] Compare the results from 3.2 and 3.3. Compare policy gradient methods with value approximation methods. what conclusion can you draw?