Practice Midterm Exam

COM SCI 188: Intro to Robotics | Winter 2025

Solution

Question	Points
Multiple Choices	18
Fill in the Blank	12
Short Answers	15
DH Parameters	10
Camera Projection	15
FastSLAM	15
Sequential Decision Making	15
Total:	100

Q 1 Multiple Choices

For each question below, select all that applies. Partial credit will be given, but selecting any incorrect option will result in no credit.

(a) (3 points) Which of the following correctly classifies the sensor?

A. LIDAR – passive, exteroceptive
B. IMU – active, exteroceptive
C. Joint encoder – active, exteroceptive
D. LIDAR – active, exteroceptive

(b) (3 points) A robot is said to be redundant if:

A. It has more than 6 DoF
B. It lacks sufficient DoF for the task
C. It has more DoF than required to complete the task
D. It has exactly the number of DoF needed for 3D pose control

(c) (3 points) You apply the following convolutional kernel to the image:

$$\begin{bmatrix} -1 & -2 & -3 & -2 & -1 \\ -2 & -4 & -6 & -4 & -2 \\ 0 & 0 & 0 & 0 & 0 \\ 2 & 4 & 6 & 4 & 2 \\ 1 & 2 & 3 & 2 & 1 \end{bmatrix}$$

What is the most likely resulting image after applying the kernel? **[D]**

(A) (B) (C) (D)

(d) (3 points) What is the likely result of increasing the derivative gain too much in a PID controller?

A. System becomes more responsive
B. System becomes slower to converge
C. System becomes overly damped and sensitive to noise
D. Control error increases steadily

(e) (3 points) In which of the scenarios is PRM more suitable than RRT?

A. A robot planning in a known, static environment

B. A robot solving many queries with different start and goal pairs

C. A robot operating in a dynamic environment with moving obstacles

D. A single, fast path is needed from a start to a goal in an unfamiliar environment

(f) (3 points) Which of the following are true about camera calibration?

A. The Kabsch algorithm is used to align 2D pixel coordinates across cameras

B. The Kabsch algorithm uses SVD to align two sets of 3D points

C. Extrinsic parameters describe camera distortion and perspective

D. Hand-eye calibration between an RGB-D camera and a robot's end-effector requires at least 3 non-collinear 3D point correspondences

Q 2 Fill in the Blank

(a) (3 points) While **[A]** learns a direct mapping from states to actions using supervised learning, **[C]** models trajectories as attractor-based systems that can generalize to new goals, and **[B]** aims to recover the underlying reward function that explains expert behavior.

(b) (3 points) Monte Carlo methods estimate solutions to complex problems by using **[E]** and performing **[F]** on the results.

(c) (3 points) In solving MDPs, **[I]** supports early stopping because it can monitor convergence of the algorithm, whereas **[J]** requires full policy evaluation steps before assessing convergence.

(d) (3 points) In EKF-SLAM, the **[L]** step uses the motion model to compute the predicted mean and covariance, while the **[K]** step incorporates sensor data using the observation model and updates the belief using **[N]** rule.

Options:

A. behavior cloning

B. inverse reinforcement learning

C. dynamic movement primitives

D. reinforcement learning

E. random sampling

F. statistical analysis

G. exhaustive search

H. symbolic reasoning

I. value iteration

J. policy iteration

K. update

L. prediction

M. Kalman

N. Bayes

O. resampling

Q 3 Short Answers

Provide a short answer to each questions.

(a) (4 points) If a small gear with 10 teeth drives a larger gear with 40 teeth, what is the gear ratio? How does it affect the output speed and torque?

> **Solution:** The gear ratio is:
>
> $$\text{Gear Ratio} = \frac{\text{Number of teeth on driven gear}}{\text{Number of teeth on driving gear}} = \frac{40}{10} = 4 : 1$$
>
> The output speed is reduced by a factor of 4, and the output torque is increased by a factor of 4, assuming no losses (ideal case).

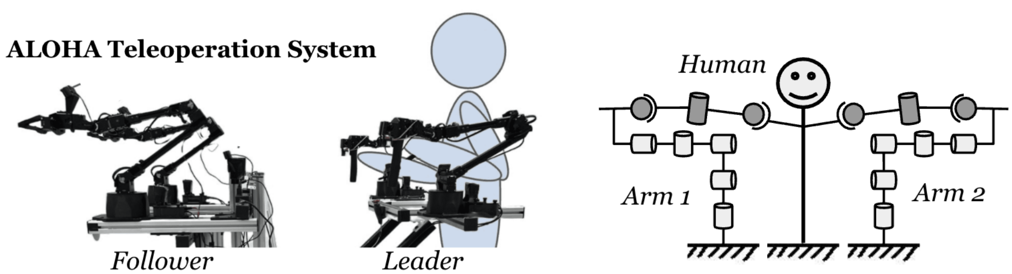
(b) (3 points) What is the Markov property?

> **Solution:** Given current state, future state is independent of the history.

(c) (4 points) What are the two major challenges in imitation learning?

> **Solution:** (1) compounding errors/distributional shift, and (2) multimodal actions

(d) (4 points) The ALOHA teleoperation system is a "puppeteering" setup where a human demonstrator kinesthetically move a set of leader arms that control the follower arms :

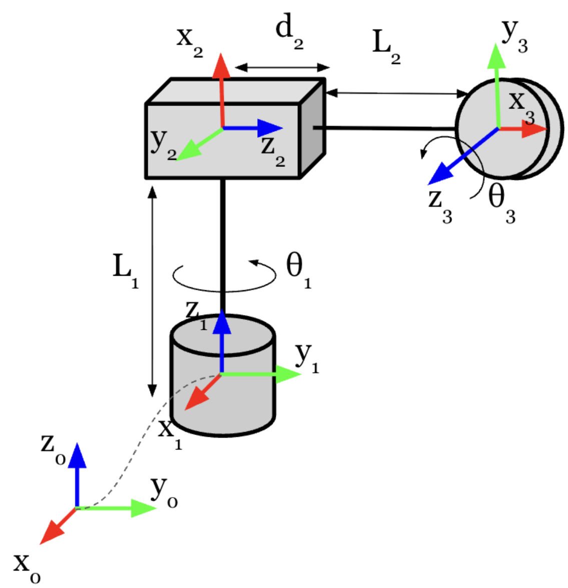


What is the DoF of the system formed by the human and the leader arms? (assume the demonstrator can only move their arms)

> **Solution:**
> \# of links = 4 (human arms) + 8 (robot arms) + 2 (hands) + 1 (ground) = 15
> \# of revolute joints = 12
> \# of spherical joints = 4
> dof = 6x(15-1) - (12x5 + 4x3) = 12

Q 4 DH Parameters

Consider the following RPR manipulator:



(a) (3 points) Write down the DH Paramters:

Solution:

i	a_{i-1}	α_{i-1}	d_i	ϕ_i
1	0	0	0	θ_1
2	0	$-\pi/2$	d_2	$-\pi/2$
3	0	$-\pi/2$	0	$\theta_3 - \pi/2$

(b) (4 points) Derive the forward kinematics for this manipulator (find $^0T_3$)

Solution: $^0T_3 = ^0 T_1^1 T_2^2 T_3$
(spell out each transformation using DH parameters with formula in equation sheet)

(c) (3 points) Briefly describe what inverse kinematics is. Explain why solving inverse kinematics can be challenging for robotic manipulators.
(List at least two reasons.)

Solution: Inverse kinematics find the configuration space joint angle values given workspace end-effector coordinates. Due to joint limits and arm redundancy, IK can be very hard.

Q 5 Camera Projection

You are given a calibrated RGB camera and an RGB-D camera. A 2D pixel is predicted in the RGB image, and you want to estimate its corresponding 3D location using the RGB-D depth map. You will:

1. Convert pixels in the depth map into 3D points in the RGB-D camera frame,
2. Transform these 3D points into the RGB camera frame,
3. Project them into the RGB image, and
4. Find the 3D point whose projection is closest to the target pixel.

You are given:

- Depth map $D(u, v)$ from the RGB-D camera,
- Intrinsic matrices $K_{\text{RGB-D}}$ and K_{RGB},
- Extrinsic transformation $T =^{\text{RGB}} T_{\text{RGB-D}} \in SE(3)$.
- Target pixel $(u^*, v^*)_{RGB}$ in RGB image

(a) (4 points) Find the 3D point $\mathbf{P}_{\text{RGB-D}}$ corresponding to pixel (u, v) of the RGB-D image.

Solution:

$$\mathbf{P}_{\text{RGB-D}} = D(u, v) \cdot K_{\text{RGB-D}}^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix}$$

(b) (4 points) Transform ransform $\mathbf{P}_{\text{RGB-D}}$ into the RGB camera frame $\mathbf{P}_{\text{RGB}}$.

Solution: $\tilde{\mathbf{P}}_{\text{RGB-D}} = \begin{bmatrix} \mathbf{P}_{\text{RGB-D}} \\ 1 \end{bmatrix}$ $\quad \tilde{\mathbf{P}}_{\text{RGB}} = T \cdot \tilde{\mathbf{P}}_{\text{RGB-D}}$

$T = \begin{bmatrix} R & \mathbf{t} \\ 0 & 1 \end{bmatrix}$ $\quad \mathbf{P}_{\text{RGB}} = R \cdot \mathbf{P}_{\text{RGB-D}} + \mathbf{t}$

(c) (4 points) Provide the equation to project the transformed 3D point into RGB image coordinates (u', v').

Solution:

$$\begin{bmatrix} u' \\ v' \\ 1 \end{bmatrix} \propto K_{\text{RGB}} \cdot \mathbf{P}_{\text{RGB}}$$

Normalized pixel coordinates:

$$u' = \frac{x'}{z'}, \quad v' = \frac{y'}{z'}$$

where $\mathbf{P}_{\text{RGB}} = [x', y', z']^\top$.

(d) (3 points) Find the corresponding 3D point P^*_{RGB-D} closest to (u^*, v^*) in the RGB image.

Solution:

Search over all projected points (u', v') and select the 3D point $\mathbf{P}_{\text{RGB-D}}$ that minimizes:

$$\|(u', v') - (u^*, v^*)\|$$

Thus, the closest point is:

$$P^*_{\text{RGB-D}} = \arg\min_{\mathbf{P}_{\text{RGB-D}}} \|(u', v') - (u^*, v^*)\|$$

Q 6 FastSLAM

FastSLAM is an algorithm that addresses the SLAM problem by combining:

- A particle filter to estimate the robot's trajectory,
- A set of local EKFs to estimate the locations of landmarks.

FastSLAM factorizes the joint posterior by representing the robot's trajectory uncertainty with particles, while maintaining an independent EKF for each landmark within every particle.

Suppose you are using FastSLAM with $M = 50$ particles. Each particle tracks its own pose and maintains a local map of $n = 20$ landmarks using EKFs.

(a) (3 points) What is an Extended Kalman Filter (EKF)? Briefly describe how it is used in state estimation.

Solution: An EKF is a recursive estimator for nonlinear systems. It linearizes the models to apply Kalman filter updates. In SLAM, EKFs estimate landmark positions by fusing sensor data with prior estimates, using covariance to model uncertainty.

(b) (2 points) How many EKFs are maintained across the entire set of particles?

Solution:

$$\text{Total EKFs} = M \times N = 50 \times 20 = 1000$$

(c) (5 points) What is the computational complexity of FastSLAM **per time step** in terms of the number of landmarks n and number of particles M?

Solution:

$$\mathcal{O}(M \times n)$$

(d) (5 points) In practice, when is FastSLAM computationally more efficient than EKF-SLAM? (Give a condition involving n and M)

Solution: EKF-SLAM has per time step complexity $\mathcal{O}(n^2)$, while FastSLAM has $\mathcal{O}(M \times n)$. Therefore, FastSLAM is more efficient when:

$$M \times n \ll n^2 \quad \Longrightarrow \quad M \ll n$$

Q 7 Sequential Decision Making

MDP, value iteration and policy iteration.

(a) (5 points) Match each term on the left with its corresponding euqation.

Term	Equation
Bellman Equation	$\pi_{\text{new}}(s) = \arg\max_a \sum_{s'} P(s'\|s,a)[R(s,a,s') + \gamma V^{\pi_{\text{old}}}(s')]$
Value Iteration	$\pi(s) = \arg\max_a \sum_{s'} P(s'\|s,a)[R(s,a,s') + \gamma V^{\pi}(s')]$
Policy Extraction	$V_k(s) = \max_a \sum_{s'} P(s'\|s,a)[R(s,a,s') + \gamma V_k(s')]$
Policy Evaluation	$V(s) = \max_a \sum_{s'} P(s'\|s,a)[R(s,a,s') + \gamma V(s')]$
Policy Improvement	$V_{k+1}^{\pi}(s) = \sum_{s'} P(s'\|s,\pi(s))[R(s,\pi(s),s') + \gamma V_k^{\pi}(s')]$

(b) (2 points) How does the value of the discount factor γ of an MDP influences the optimal policy behavior?

> **Solution:** The smaller γ is, the more myopic the optimal policy behaves. With larger γ, the agent values future rewards more than immediate ones.

(c) (4 points) In state s, two actions are available:

- Action A: expected reward 3, next state value $V(s') = 10$.
- Action B: expected reward 2, next state value $V(s') = 20$.

Discount factor: $\gamma = 0.8$.
Compute the Q-values and select the best action.

> **Solution:**
>
> $$Q(s, A) = 3 + 0.8 \times 10 = 11$$
>
> $$Q(s, B) = 2 + 0.8 \times 20 = 18$$
>
> Best action: **Action B**

(d) (4 points) A policy π always chooses Action A in state s:

- $R(s, A) = 4$,
- Next state is always s itself,
- Discount factor $\gamma = 0.95$.

Compute $V_\pi(s)$ under this policy.

> **Solution:** $V(s) = R(s, A) + \gamma V(s) \quad \Longrightarrow \quad V(s) = 4 + 0.95V(s)$
>
> $$V(s) - 0.95V(s) = 4 \quad \Longrightarrow \quad 0.05V(s) = 4$$
>
> $$V(s) = \frac{4}{0.05} = 80$$