AI Enabled Control Engineering

Laboratory Report 3: MPC, DQN, PPO and TD3 Control of a Rotary Inverted Pendulum

Group: Group XX

Students: Name 1; Name 2; Name 3; Name 4

Student IDs: ID 1; ID 2; ID 3; ID 4

Submission date: YYYY-MM-DD

Submit one group ZIP through one designated representative. The report must contain MPC, DQN, PPO and TD3 results from simulation training/tuning, ten repeated simulation tests and ten repeated physical-system tests.

Teaching Assistants: Zhixiang Ju and Haozhe Wang

Submission Requirements – Remove Before Final Submission

Before submission, change `\instructionstrue` to `\instructionsfalse`. Remove all instructional placeholders, empty rows that are not needed, and sample wording. Retain only the group's own settings, results, figures, calculations and analysis.

Scope of Report 3

The report covers four controllers:

1. constrained model predictive control (MPC);
2. Deep Q-Network control (DQN);
3. Proximal Policy Optimization (PPO);
4. Twin Delayed Deep Deterministic Policy Gradient (TD3).

For every controller, the group must complete ten fixed-setting simulation trials and ten fixed-setting physical-system trials. All failed trials must be retained. Do not replace failed trials with additional favourable trials.

Required ZIP structure

```
GroupXX_LabReport3.zip
|-- GroupXX_LabReport3.pdf
|-- configs/
|   |-- dqn_config_snapshot.py
|   |-- ppo_config_snapshot.py
|   `-- td3_config_snapshot.py
|-- models/
|   |-- dqn_selected_model.*
|   |-- ppo_selected_checkpoint_or_deploy_model.*
|   `-- td3_best_model.zip
|-- training/
|   |-- dqn/   # logs csv , metrics and original training figures
|   |-- ppo/
|   `-- td3/
|-- data/
|   |-- mpc/sim/ and mpc/hardware/
|   |-- dqn/sim/ and dqn/hardware/
|   |-- ppo/sim/ and ppo/hardware/
|   `-- td3/sim/ and td3/hardware/
`-- bonus/     # only when a bonus is claimed
```

Every file referenced in the report must be included and open correctly. Use clear run identifiers such as `TD3_SIM_R01.csv` and `TD3_HW_R01.png`.

MPC requirement

Select one final set of Q, R, prediction horizon N and projected-gradient iteration count n_{iter}. Report the tuning process and then hold the final values fixed for ten simulation trials and ten physical-system trials. Changing any final MPC parameter requires restarting the ten-run set.

RL requirement
For each of DQN, PPO and TD3, provide one complete selected hyperparameter set, including the executable reward function, learning rate(s), network structure, training length, exploration/noise settings, replay or rollout settings, evaluation settings and domain-randomization schedule. Include original training curves and explain why the selected checkpoint is preferred over the final checkpoint.
The selected training result must improve on the corresponding course benchmark in at least one meaningful simulation-training dimension, for example:

- a smoother reward curve with lower moving variance;
- no late-stage reward collapse;
- higher deterministic evaluation reward;
- higher capture or stable-success rate;
- better survival under parameter randomization;
- similar success with lower control effort.

A claim without matched settings, numerical evidence and original curves does not count as an improvement.

Model selection and deployment
Use the selected best checkpoint, not automatically the final checkpoint. For TD3, click **Choose Model File** in the final deployment panel and select the intended file named `best_model.zip`. Do not select a stale `*.td3_deploy.npz` cache for the benchmark comparison. Record the exact checkpoint path and training step in the report.
For PPO, preserve the matching normalization statistics and follow the Class-12 deployment workflow. State whether the physical controller uses the standard hybrid swing-up plus PPO stabilizer or the Bonus-2 single-policy solution.

Controlled comparisons and common test protocol
Within each ten-run set, keep all model, controller, calibration, safety, initial-condition and duration settings fixed. For comparisons between methods, use the same control rate, initial-condition rule, PWM safety limit, test duration, success definition and random seeds wherever the software permits. If an unavoidable mismatch exists, state it explicitly and explain its effect.

Minimum experimental outcome
The simulation-test pipeline and the physical-system pipeline must both produce successful control for MPC, DQN, PPO and TD3. A ten-run set may contain failed trials, but all failures must be reported and explained. A method that never captures and controls the pendulum in either simulation or hardware cannot receive full marks for that section.

Report grading and bonuses

The three lab reports together contribute 50% of the final course grade, while the final examination contributes the remaining 50%. The relative weights among Report-1, Report-2, and Report-3 will not be announced in advance. However, Report-3 covers the most comprehensive experimental content, including MPC, DQN, PPO, TD3, simulation evaluation, and real-hardware validation. Therefore, Report-3 is expected to have a larger contribution within the overall report component. The final weighting will be determined after grading to ensure a fair overall evaluation and to avoid disadvantages caused by fixing the ratio before the quality of all reports is known.

Report-3 is graded independently using a 100-point scale:

Component	Marks
MPC design, parameter selection, repeated tests and analysis	20
DQN configuration, training, repeated tests and analysis	20
PPO configuration, training, repeated tests and analysis	15
TD3 configuration, training, repeated tests and analysis	15
Analysis and conclusions	10
Course Experience and Suggestions	20
Bonus 1: clear and substantial benchmark improvement(DQN or PPO or TD3)	+10
Bonus 2: PPO single policy performs swing-up and stabilization	+20
Maximum Report-3 score	100

Bonus 1: Benchmark improvement

Students may receive an additional 10 points if their proposed method shows clear and reproducible improvement over the provided benchmark. The improvement should be supported by quantitative evidence, such as: higher reward, smoother training curves, reduced performance degradation, better robustness under parameter randomization, higher success rate, or improved sim-to-real transfer performance.

Bonus 2: PPO single-policy swing-up and stabilization

Students may receive an additional 10 points if PPO achieves complete single-policy control, including swing-up from the natural hanging-down position, upright capture, and stabilization, without using external controllers such as energy shaping, PID, LQR, or MPC switching.

Bonus 3: ROS framework integration

Bonus 3 is independent from the Report-3 score cap. A complete ROS-based rotary inverted pendulum framework integration may receive an additional 5 points in the overall course grade.

The qualifying framework should integrate the course deployment workflow, control panel, Python interface, and embedded firmware into a unified system, including reproducible simulation testing and real-hardware deployment.

Contents

Group Member Contributions

The percentages must sum to 100%. Group size does not change the grading criteria or total group score; the declared contribution percentages may affect the relative individual scores within the group.

Table 1: Contribution of each group member

Member	Name	Main contribution	Contribution (%)
Member 1			
Member 2			
Member 3			
Member 4			
		Total	**100%**

Remove unused rows for groups with fewer than four members.

1 Part I: Common Experimental Protocol and Evaluation Metrics

1.1 Common settings

Record the fixed protocol used to make the four controllers comparable. State all exceptions explicitly.

Table 2: Common simulation and physical-test protocol

Setting	Value	Setting	Value
PWM safety limit		Test duration	
Capture angle threshold		Stable angle threshold	
Stable velocity threshold		Required stable duration	
Simulation seed list		Randomization levels tested in sim test	

1.2 Success, final capture and stable-stage definitions

Use one report-wide definition unless the software imposes a stricter method- specific definition. Let k_s be the final accepted entry into the selected stable region. It is accepted only when the pendulum remains within the region for the required duration and does not subsequently lose control before the end of the trial.

For successful trials, report

$$\text{Mean } |\alpha| = \frac{1}{N_s} \sum_{k \in \mathcal{S}} |\alpha_k|, \tag{1}$$

$$\text{Std } |\alpha| = \sqrt{\frac{1}{N_s} \sum_{k \in \mathcal{S}} (|\alpha_k| - \text{Mean } |\alpha|)^2}, \tag{2}$$

$$\text{Mean } |PWM| = \frac{1}{N_s} \sum_{k \in \mathcal{S}} |PWM_k|, \tag{3}$$

where $\mathcal{S}$ is the final stable-stage sample set. The final-capture time is t_{k_s}. Report "Fail" for unsuccessful trials and "–" for unavailable stable-stage metrics.

1.3 Across-run statistics

For each metric z, calculate the mean and sample standard deviation across the successful trials only:

$$\bar{z} = \frac{1}{n} \sum_{i=1}^{n} z_i, \qquad s_z = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (z_i - \bar{z})^2}. \tag{4}$$

Always state n, the number of successful trials, together with the success rate. Do not hide failed trials by excluding them from the ten-row table.

1.4 Training-curve stability for RL methods

For each RL method, define the moving-window length used in the report and calculate at least:

- moving mean reward per step;
- moving standard deviation of reward per step;
- best and final deterministic evaluation reward;
- post-peak drawdown or an explicit collapse indicator;
- nominal and randomized capture/stable-success rates.

State whether episode reward, reward per step or normalized reward is plotted. Do not compare unlike quantities.

2 Part II: Model Predictive Control

2.1 MPC formulation and selected final parameters

State the state order, local prediction model, objective function, constraints, estimator and swing-up-to-MPC switching logic. Write the selected cost in the form

$$J = \sum_{i=0}^{N-1} \left(x_i^T Q x_i + u_i^T R u_i \right) + x_N^T P x_N . \tag{5}$$

Table 3: Final MPC parameter set used for all reported trials

Parameter	Selected value	Parameter	Selected value
q_θ		$q_{\dot\theta}$	
q_α		$q_{\dot\alpha}$	
Input weight R		Terminal weight P	
Prediction horizon N		PGD iterations n_{iter}	
Estimator mode		PWM limit	
MPC enter angle		MPC exit angle	
Swing-up setting			

2.2 MPC simulation: ten fixed-setting trials

Table 4: Ten nonlinear-simulation trials using the final MPC parameter set.

Trial	Success	Final capture / s	Mean $\lvert\alpha\rvert$ / rad	Std. $\lvert\alpha\rvert$ / rad	Mean $\lvert PWM\rvert$	Max. $\lvert\theta\rvert$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

2.3 MPC physical system: ten fixed-setting trials

Table 5: Ten physical-system trials using the final MPC parameter set.

Trial	Success	Final capture / s	Mean $\|\alpha\|$ / rad	Std. $\|\alpha\|$ / rad	Mean $\|PWM\|$	Max. $\|\theta\|$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

Explain any difference from simulation using evidence from the logs, including friction, backlash, dead zone, calibration, sensor noise, velocity estimation, solver convergence and computation time.

2.4 MPC sim-to-real summary

The relative difference between simulation and hardware results is defined as

$$\text{Relative difference}(\%) = \frac{|M_{\text{hardware}} - M_{\text{simulation}}|}{M_{\text{simulation}}} \times 100\%.$$

For metrics where a larger value indicates better performance (e.g., success rate), the same definition is used to quantify the absolute performance deviation. For metrics where lower values indicate better performance (e.g., capture time, angle error, and control effort), the relative difference represents the degradation from simulation to hardware.

Table 6: MPC simulation-to-real comparison, reported as mean ± standard deviation

Metric	Simulation	Hardware	Relative difference / %
Success rate			
Final capture time / s			
Mean $\|\alpha\|$ / rad			
Std. $\|\alpha\|$ / rad			
Mean $\|PWM\|$			
Max. $\|\theta\|$ / rad			

3 Part III: Deep Q-Network Control

3.1 DQN observation, action and reward

State the discrete action set and network output ordering. Reproduce the executable reward function used in training, including all coefficients, gates, bonuses and termination penalties.

$$r_k = \boxed{\text{insert the exact DQN reward implemented in code}} \tag{6}$$

Table 7: Selected DQN observation, action and reward design

Item	Student design
Discrete action/PWM set	
Network input/output dimensions	
Reward function	
Termination conditions	
Episode length and control rate	

3.2 Selected DQN hyperparameters

The course benchmark uses a $6 \rightarrow 64 \rightarrow 64 \rightarrow 10$ ReLU network, Adam learning rate 5×10^{-4}, replay capacity 30,000, batch size 512, $\gamma = 0.95$ and a five-million-step nominal-to-randomized curriculum. Record one complete student-selected configuration below.

Table 8: Complete selected DQN hyperparameter set.

Parameter	Selected value	Reason / expected effect
Total environment steps		
Learning rate and optimizer		
Discount factor γ		
Replay capacity		
Learning-start / warm-up steps		
Batch size		
Training frequency		
Gradient steps per event		
Target-network synchronization		
Behavior-network synchronization		
Initial/minimum ϵ		
Exploration decay schedule		
Nominal-stage duration		
Domain-randomization stages		
Randomized parameters/ranges		
Evaluation frequency/episodes		

3.3 DQN training result and benchmark improvement

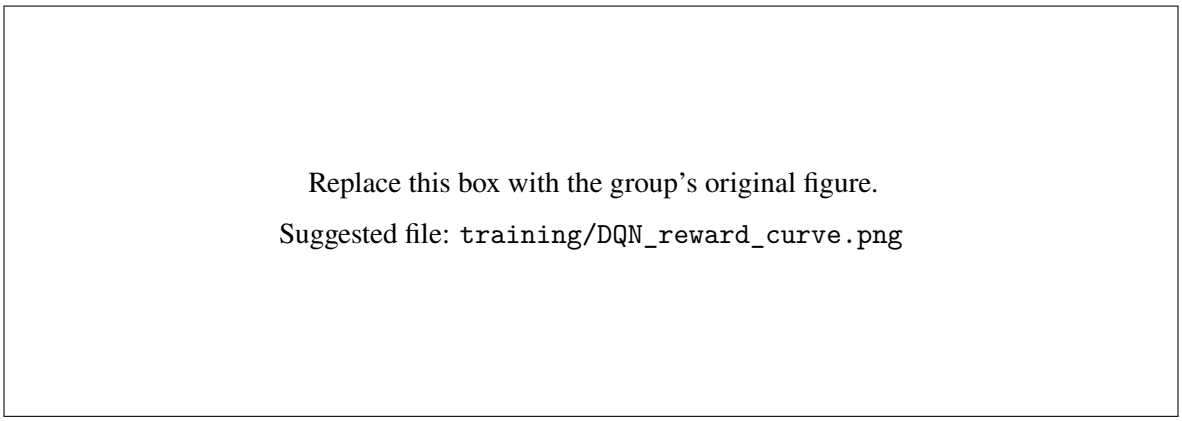

Figure 1: DQN training reward per step with moving mean and moving standard deviation. Mark all randomization-stage boundaries.

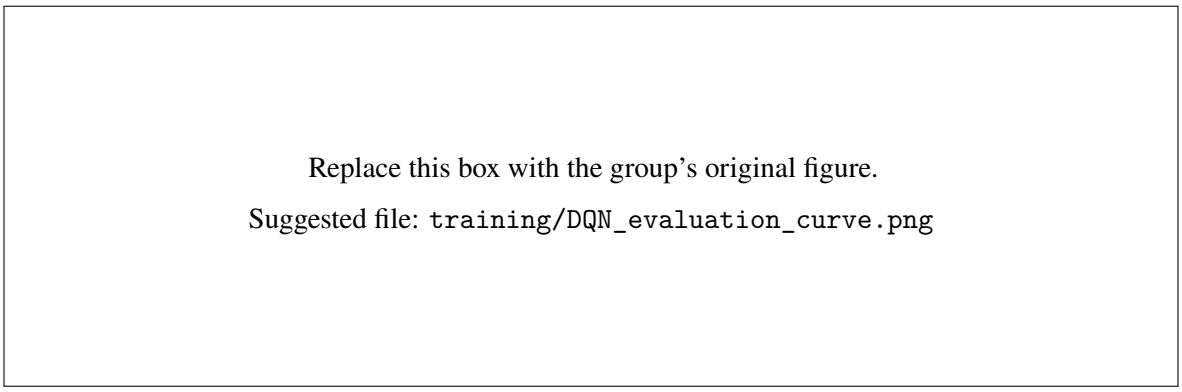

Figure 2: DQN deterministic evaluation, capture rate and stable-success rate across training.

Table 9: DQN benchmark comparison for the selected student model.

Metric	Course benchmark	Student model	Improvement / evidence
Best moving-mean reward per step	–	–	–
Moving standard deviation in selected window	–	–	–
Final moving-mean reward per step	–	–	–
Best deterministic evaluation reward per step	–	–	–
Nominal capture or completion rate	–	–	–
Nominal stable-success rate	–	–	–
Highest tested randomization level	–	–	–
Stable-success rate at that level	–	–	–
Post-peak collapse observed?	–	–	–
Selected checkpoint step	–	–	–

Explain which parameter or code changes improved the benchmark and why the selected checkpoint is more suitable than the final checkpoint. If collapse ,discuss why.

3.4 DQN simulation: ten fixed-model trials

Table 10: Ten simulation trials using the selected DQN checkpoint and fixed test settings.

Trial	Success	Final capture / s	Mean $\|\alpha\|$ / rad	Std. $\|\alpha\|$ / rad	Mean $\|PWM\|$	Max. $\|\theta\|$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

Replace this box with the group's original figure.

Suggested file: `figures/DQN_SIM_representative.png`

Figure 3: Representative DQN simulation response from natural hanging-down swing-up through stabilization.

3.5 DQN physical system: ten fixed-model trials

Table 11: Ten physical-system trials using the selected DQN deployment model.

Trial	Success	Final capture / s	Mean $\lvert\alpha\rvert$ / rad	Std. $\lvert\alpha\rvert$ / rad	Mean $\lvert PWM\rvert$	Max. $\lvert\theta\rvert$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

Replace this box with the group's original figure.

Suggested file: `figures/DQN_HW_representative.png`

Figure 4: Representative DQN physical-system response.

Discuss simulation performance, physical performance, model-selection details, failed runs and the DQN sim-to-real gap.

4 Part IV: Proximal Policy Optimization

4.1 PPO task, observation, action and reward

State whether the base experiment uses the standard hybrid controller (hand-coded swing-up plus PPO stabilization) or another approved architecture. Describe the PPO observation, action scaling, Actor/Critic networks, normalization and exact executable reward.

$$r_t = \boxed{\text{insert the exact PPO reward implemented in code}} \tag{7}$$

Table 12: Selected PPO task and model definition

Item	Student design
Task: balance-only or complete swing-up	
Swing-up controller used in base experiment	
Observation/history vector and scaling	
Continuous action and PWM mapping	
Actor/Critic architecture and activation	
Observation/reward normalization	
Reward terms and coefficients	
Termination and success definition	

4.2 Selected PPO hyperparameters

The course benchmark uses 16 environments, rollout length 1024 per environment, batch size 1024, 10 epochs, learning rate 3×10^{-4}, $\gamma = 0.9975$, GAE $\lambda = 0.95$ and clip range 0.2. Record the complete selected configuration.

Table 13: Complete selected PPO hyperparameter set.

Parameter	Selected value	Reason / expected effect
Actor/Critic hidden sizes		
Activation function		
Total timesteps		
Number of parallel environments		
Rollout length n_{steps}		
Batch size		
Optimization epochs		
Learning rate / schedule		
Discount factor γ		
GAE λ		
Clip range		
Target KL		
Entropy coefficient		
Value coefficient		
Maximum gradient norm		
Initial log standard deviation		
Observation normalization		
Reward normalization		
Domain-randomization curriculum		

Parameter	Selected value	Reason / expected effect
Randomized parameters/ranges		
Evaluation frequency/episodes		
Random seeds		
Selected checkpoint and normalization file		
Physical deployment model/workflow		

4.3 PPO training result and benchmark improvement

Replace this box with the group's original figure.

Suggested file: `training/PPO_reward_curve.png`

Figure 5: PPO raw training reward per step with moving mean and moving standard deviation. Show curriculum/randomization progress.

Replace this box with the group's original figure.

Suggested file: `training/PPO_evaluation_curve.png`

Figure 6: PPO deterministic nominal and randomized evaluation curves.

Table 14: PPO benchmark comparison for the selected student model.

Metric	Course benchmark	Student model	Improvement / evidence
Best moving-mean reward per step	–	–	–
Moving standard deviation in selected window	–	–	–
Final moving-mean reward per step	–	–	–
Best deterministic evaluation reward per step	–	–	–
Nominal capture or completion rate	–	–	–
Nominal stable-success rate	–	–	–
Highest tested randomization level	–	–	–
Stable-success rate at that level	–	–	–
Post-peak collapse observed?	–	–	–
Selected checkpoint step	–	–	–

Explain the selected checkpoint, matching normalization statistics, late-stage stability and any difference between normalized training reward and raw evaluation reward.

4.4 PPO simulation: ten fixed-model trials

Table 15: Ten simulation trials using the selected PPO controller and fixed test settings.

Trial	Success	Final capture / s	Mean $\lvert\alpha\rvert$ / rad	Std. $\lvert\alpha\rvert$ / rad	Mean $\lvert PWM\rvert$	Max. $\lvert\theta\rvert$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

Replace this box with the group's original figure.

Suggested file: `figures/PPO_SIM_representative.png`

Figure 7: Representative PPO simulation response, including the swing-up/stabilization mode when a hybrid controller is used.

4.5 PPO physical system: ten fixed-model trials

Table 16: Ten physical-system trials using the selected PPO deployment controller.

Trial	Success	Final capture / s	Mean $\|\alpha\|$ / rad	Std. $\|\alpha\|$ / rad	Mean $\|PWM\|$	Max. $\|\theta\|$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

Replace this box with the group's original figure.

Suggested file: `figures/PPO_HW_representative.png`

Figure 8: Representative PPO physical-system response.

Discuss capture, stable balance, mode switching, normalization mismatch, physical repeatability and failed runs.

5 Part V: Twin Delayed Deep Deterministic Policy Gradient

5.1 TD3 observation, action and reward

The course deployment Actor is already a compact $7 \rightarrow 64 \rightarrow 64 \rightarrow 1$ network and does not require distillation. State the exact seven-dimensional observation order, continuous action scaling, previous-action convention and reward.

$$r_k = \boxed{\text{insert the exact TD3 reward implemented in code}} \tag{8}$$

Table 17: Selected TD3 task and model definition

Item	Student design
Observation vector and order	
Previous-action/history convention	
Actor architecture and activation	
Twin-Critic architecture	
Continuous action and PWM mapping	
Reward terms and coefficients	
Termination and success definition	
Checkpoint selected with Choose Model File	

5.2 Selected TD3 hyperparameters

The course benchmark uses replay capacity 1,000,000, learning starts 10,000, batch size 128, Actor learning rate 10^{-4}, Critic learning rate 10^{-3}, $\gamma = 0.99$, $\tau = 0.005$, policy delay 2, exploration sigma 0.1, target noise 0.2 and target-noise clip 0.5.

Table 18: Complete selected TD3 hyperparameter set.

Parameter	Selected value	Reason / expected effect
Actor hidden sizes/activation		
Critic hidden sizes/activation		
Actor learning rate		
Critic learning rate		
Discount factor γ		
Polyak coefficient τ		
Replay capacity		
Learning-start steps		
Batch size		
Train frequency		
Gradient steps		

Parameter	Selected value	Reason / expected effect
Policy delay		
Exploration-noise sigma		
Target-policy noise		
Target-noise clip		
Gradient clipping		
Total timesteps		
Nominal-stage duration		
Domain-randomization stages		
Randomized parameters/ranges		
Evaluation frequency/episodes		
Random seeds		
Selected `best_model.zip` path and step		

5.3 TD3 training result and benchmark improvement

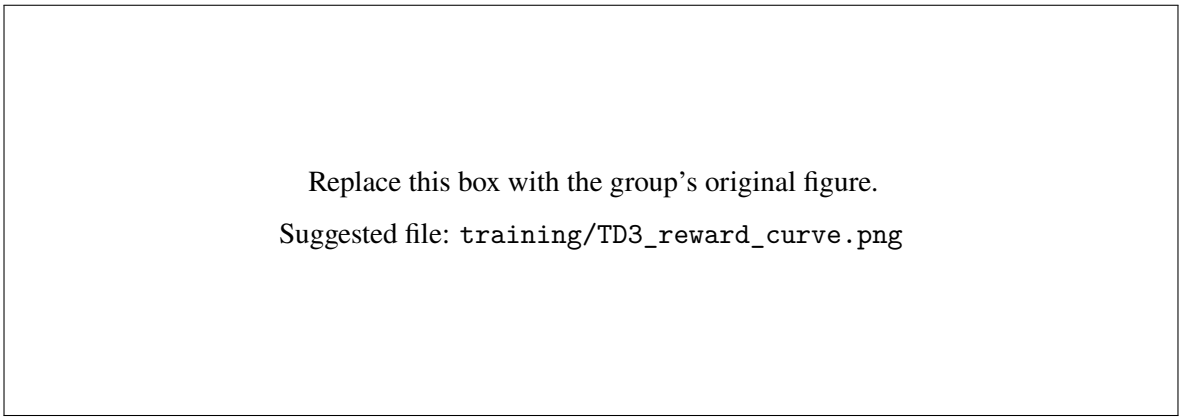

Figure 9: TD3 episode reward per step with moving mean and moving standard deviation. Mark all stage boundaries.

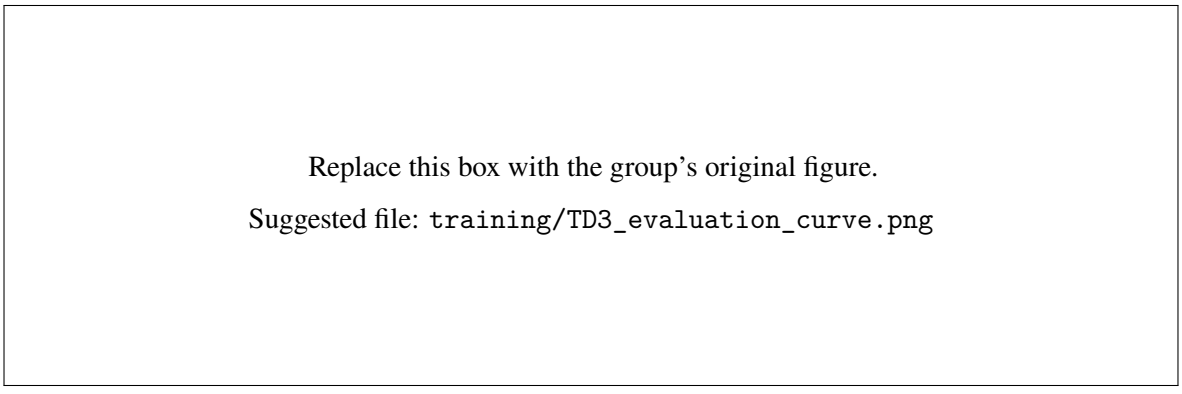

Figure 10: TD3 deterministic evaluation reward, capture rate and stable-success rate.

Table 19: TD3 benchmark comparison for the selected student model.

Metric	Course benchmark	Student model	Improvement / evidence
Best moving-mean reward per step	–	–	–
Moving standard deviation in selected window	–	–	–
Final moving-mean reward per step	–	–	–
Best deterministic evaluation reward per step	–	–	–
Nominal capture or completion rate	–	–	–
Nominal stable-success rate	–	–	–
Highest tested randomization level	–	–	–
Stable-success rate at that level	–	–	–
Post-peak collapse observed?	–	–	–
Selected checkpoint step	–	–	–

Explain how delayed Actor updates, action noise, target noise, Critic learning rate, batch size and randomization affected curve stability. State why the selected `best_model.zip` is preferred over the final model.

5.4 TD3 simulation: ten fixed-model trials

Table 20: Ten simulation trials using the selected TD3 `best_model.zip`.

Trial	Success	Final capture / s	Mean $\lvert\alpha\rvert$ / rad	Std. $\lvert\alpha\rvert$ / rad	Mean $\lvert PWM\rvert$	Max. $\lvert\theta\rvert$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

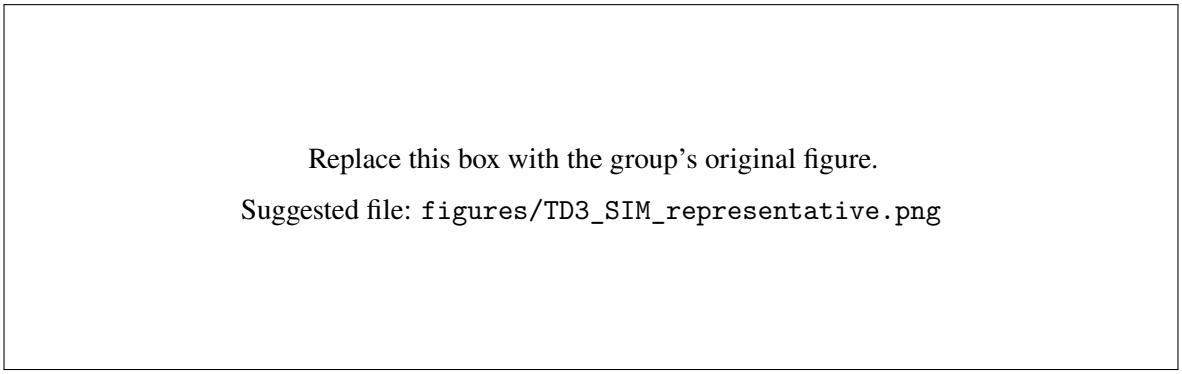

Figure 11: Representative TD3 full-task simulation response from natural hanging-down swing-up to upright stabilization.

5.5 TD3 physical system: ten fixed-model trials

Table 21: Ten physical-system trials using the selected TD3 `best_model.zip`.

Trial	Success	Final capture / s	Mean $\|\alpha\|$ / rad	Std. $\|\alpha\|$ / rad	Mean $\|PWM\|$	Max. $\|\theta\|$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

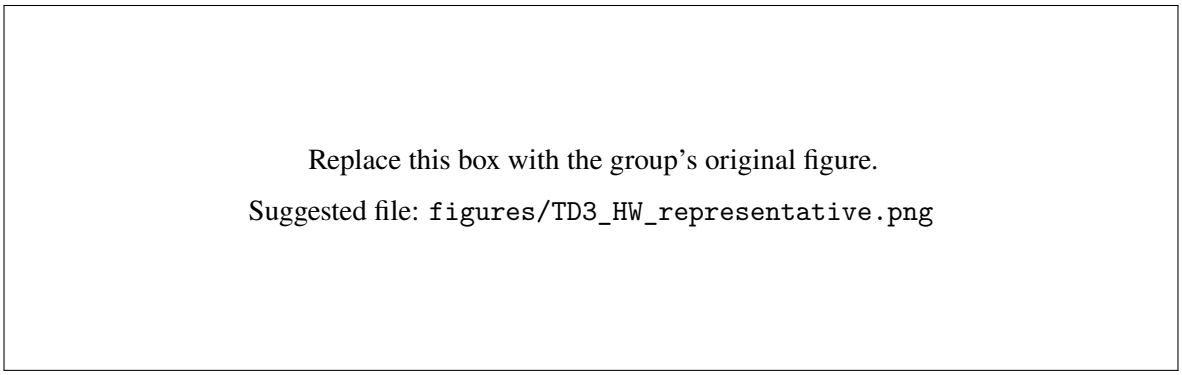

Figure 12: Representative TD3 physical-system response.

Discuss full-policy swing-up, first-action/PWM behaviour, previous-action input, model upload verification, physical repeatability and failed trials.

6 Part VI: Cross-Controller Statistical Comparison

6.1 Simulation comparison

Use the ten-run summary statistics, not one favourable trial. Do not compare the internal training rewards of different algorithms as though they were the same objective. Compare common physical-control metrics.

Table 22: Ten-run simulation comparison. Report mean ± standard deviation for successful runs.

Method	Success / %	Capture / s	Mean $\|\alpha\|$ / rad	Std. $\|\alpha\|$ / rad	Mean $\|PWM\|$	Max. $\|\theta\|$ / rad
MPC						
DQN						
PPO						
TD3						

6.2 Physical-system comparison

Table 23: Ten-run physical-system comparison. Report mean ± standard deviation for successful runs.

Method	Success / %	Capture / s	Mean $\|\alpha\|$ / rad	Std. $\|\alpha\|$ / rad	Mean $\|PWM\|$	Max. $\|\theta\|$ / rad
MPC						
DQN						
PPO						
TD3						

6.3 Sim-to-real degradation by method

Table 24: Method-by-method sim-to-real degradation

Method	Success-rate change	Capture-time change	$\|\alpha\|$ change	Mean-$\|PWM\|$ change
MPC				
DQN				
PPO				
TD3				

6.4 Engineering interpretation

Discuss the following using numerical evidence:

- which method is most repeatable in simulation and in hardware;
- which method captures fastest and which has the smallest stable angle;
- which method uses the least control effort and causes the least arm excursion;
- sensitivity to parameter randomization and model mismatch;

- offline training/tuning cost versus online computation cost;
- interpretability, safety, debugging and ease of deployment;
- why a method with the best simulation reward may not be the best physical controller.

7 Part VII: Benchmark-Improvement and Bonus Evidence

7.1 Required RL benchmark-improvement summary

For each RL method, state one aspect in which the selected model exceeds the course benchmark. Use "not demonstrated" when evidence is insufficient.

Table 25: Summary of claimed benchmark improvements

Method	Claimed improvement	Numerical magnitude	Evidence location
DQN			
PPO			
TD3			

7.2 Bonus 1 claim: clear and substantial improvement

Check one:

- ☐ Bonus 1 is not claimed.
- ☐ Bonus 1 is claimed for DQN / PPO / TD3:
  Identify parameter change, numerical improvement and why the change is scientifically meaningful.

7.3 Bonus 2 claim: one PPO policy for swing-up and stabilization

Check one:

- ☐ Bonus 2 is not claimed.
- ☐ Bonus 2 is claimed and the evidence below is complete.

Provide the observation, reward, reset distribution, network and checkpoint details. Use the final policy for ten simulation and ten physical trials.

Table 26: Bonus-2 PPO single-policy simulation trials.

Trial	Success	Final capture / s	Mean $\lvert\alpha\rvert$ / rad	Std. $\lvert\alpha\rvert$ / rad	Mean $\lvert PWM\rvert$	Max. $\lvert\theta\rvert$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

Table 27: Bonus-2 PPO single-policy physical-system trials.

Trial	Success	Final capture / s	Mean $\lvert\alpha\rvert$ / rad	Std. $\lvert\alpha\rvert$ / rad	Mean $\lvert PWM\rvert$	Max. $\lvert\theta\rvert$ / rad
1	–	–	–	–	–	–
2	–	–	–	–	–	–
3	–	–	–	–	–	–
4	–	–	–	–	–	–
5	–	–	–	–	–	–
6	–	–	–	–	–	–
7	–	–	–	–	–	–
8	–	–	–	–	–	–
9	–	–	–	–	–	–
10	–	–	–	–	–	–
Across-run mean	–	–	–	–	–	–
Across-run std.	–	–	–	–	–	–
Success rate	–			Successful runs / 10 =		

7.4 Bonus 3 claim: unified ROS deployment framework

Check one:

- ☐ Bonus 3 is not claimed.
- ☐ Bonus 3 is claimed; the complete Python and firmware package is included.

Table 28: Bonus-3 implementation checklist

Required capability	Completed?	File / evidence
One unified Python entry point		
MPC/DQN/PPO/TD3 controller selection		
Simulation and physical modes		
Port selection and manual connection		
Calibration workflow		
Model/config selection and upload		
SAVE, GO and STOP safety workflow		
CSV/figure logging		
Matching STM32/Arduino `.ino` firmware		
README and reproducible launch instructions		

Describe the software architecture and identify the improvements over the original WeChat ROS framework.

8 Part VIII: Conclusions

Summarise the main findings without repeating the entire report. State the best-performing method in simulation, the best-performing method on hardware, the most robust method, the most efficient method and the most important lesson from the sim-to-real comparison. Distinguish evidence from opinion.

A Part IV: Course Experience and Suggestions

Instructions and privacy statement
Each group member must complete one copy of this survey independently. Do not discuss or coordinate the answers with other group members before completing the survey.
To protect students' privacy, do not write your name or student ID. Identify yourself only using the group member number assigned in this report, for example, Group Member 1, Group Member 2, or Group Member 3.
The information collected in this section will be treated confidentially and used solely to improve the content, organisation, documentation, software tools, and laboratory experience of future versions of this course. Individual responses will not affect the technical mark of Report-3 or any other course grade.

Individual Survey: Group Member No. 1

Group number	_____
Group member number	**1**
Date completed	_____

A. Prior Background

Complete this table based on your experience **before taking this course**.

Table 29: Prior background of Group Member No. 1

Item	Response
Previous coursework in classical control	None / Basic / Extensive
Previous experience with PID control	None / Basic / Extensive
Previous experience with LQR or state-space control	None / Basic / Extensive
Previous experience with MPC	None / Basic / Extensive
Previous experience with Python	None / Basic / Extensive
Previous experience with C/C++ or embedded systems	None / Basic / Extensive
Previous experience with reinforcement learning	None / Basic / Extensive
Previous experience deploying algorithms to real hardware	None / Basic / Extensive

B. Retrospective Learning Gains

For each competency, rate your ability both **before taking this course** and **at the end of this course**.
Use the following scale:

1 = unable, 2 = limited, 3 = basic, 4 = confident, 5 = able to work independently.

Table 30: Retrospective competency comparison for Group Member No. 1

Competency	Before course	After course
Tune a dual-loop PID controller using simulation and experimental evidence		
Explain the effects of the LQR weighting matrices Q and R		
Tune an LQR controller using quantitative experimental evidence		
Explain the effects of MPC parameters Q, R, prediction horizon, and solver iterations		
Tune an MPC controller using quantitative experimental evidence		
Explain the main differences among DQN, PPO, and TD3		
Design or modify a reinforcement-learning reward function		
Interpret reward, evaluation, capture-rate, and success-rate curves		
Diagnose unstable or collapsed reinforcement-learning training		
Select useful reinforcement-learning hyperparameters based on experimental results		
Explain the purpose of domain randomization		
Evaluate controller robustness using repeated simulation experiments		
Transfer a controller from simulation to real hardware		
Calibrate sensors and verify motor direction and PWM limits		
Upload a trained neural-network controller to the STM32 platform		
Diagnose serial communication, model-upload, and firmware problems		
Design a reproducible control experiment and report statistical results		

C. Time Investment, Difficulty, and Independence

For “Difficulty”, use:

1 = very easy, 5 = very difficult.

For “Independent completion”, use:

1 = required continuous assistance, 5 = completed independently.

Table 31: Estimated workload of Group Member No. 1

Course component	Time spent / h	Difficulty (1–5)	Independent completion (1–5)
Dual-loop PID modelling, tuning, and simulation			
Dual-loop PID real-hardware experiments			
LQR modelling, design, and simulation			
LQR real-hardware experiments			
MPC modelling, tuning, and simulation			
MPC real-hardware experiments			
DQN training and simulation testing			
DQN real-hardware deployment			
PPO training and simulation testing			
PPO real-hardware deployment			
TD3 training and simulation testing			
TD3 real-hardware deployment			
Debugging software, serial communication, and firmware			
Preparing Report-1, Report-2, and Report-3			

D. Sources of Difficulty

Select the **five** factors that consumed the most time or caused the most difficulty. Place a check mark in the second column.

Table 32: Main sources of difficulty reported by Group Member No. 1

Potential source of difficulty	Selected?
Understanding rotary inverted pendulum dynamics	
Understanding dual-loop PID control	
Tuning the inner and outer PID loops	
Understanding state-space modelling and LQR theory	
Selecting suitable LQR weighting matrices Q and R	
Designing the MPC cost function	
Selecting the MPC prediction horizon and solver iterations	
Understanding reinforcement-learning theory	
Designing the reinforcement-learning reward function	
Selecting reinforcement-learning hyperparameters	
Long training time or limited computing resources	
Interpreting training and evaluation curves	
Understanding domain randomization	
Python dependencies or software installation	
Using the graphical training and control panels	
Serial communication and model upload	
STM32 firmware compilation or flashing	
Sensor calibration and encoder direction	
Motor direction, PWM limits, or safety settings	
Differences between simulation and hardware	
Insufficient access to the physical platform	
Coordinating work among group members	
Report workload or documentation requirements	
Other: _____	

E. Evaluation of Course Design

Rate each statement from 1 to 5, where

1 = strongly disagree, 5 = strongly agree.

Table 33: Course-design evaluation by Group Member No. 1

Statement	Rating (1–5)
The progression from dual-loop PID to LQR, MPC, DQN, PPO, and TD3 was logically organised.	
Learning dual-loop PID helped me understand hierarchical feedback control.	
Learning LQR helped me connect mathematical models with feedback-control design.	
Learning MPC helped me understand prediction, constraints, and online optimisation.	
Comparing classical control and reinforcement learning improved my understanding of both.	
Comparing DQN, PPO, and TD3 improved my understanding of different reinforcement-learning methods.	
Repeated simulation tests helped me evaluate controller robustness.	
Real-hardware experiments improved my understanding of simulation-to-real gaps.	
The requirement to report means and standard deviations improved the quality of my experimental analysis.	
The provided benchmark controllers and models were useful reference points.	
Attempting to improve the provided benchmark encouraged meaningful experimentation.	
The graphical training and deployment panels reduced unnecessary setup effort.	
The control panels still allowed me to understand the underlying control and deployment process.	
The technical manuals contained enough information to complete the tasks.	
The balance among theory, simulation, and hardware experiments was appropriate.	
The amount of access to the physical platform was sufficient.	
The overall workload was appropriate for the intended learning outcomes.	
The group-work structure allowed each member to make a meaningful contribution.	
After this course, I feel able to begin an independent control, reinforcement-learning, or embedded-systems project.	

F. Most Valuable Learning Activities

Rank the following activities from 1 to 10, where

1 = the activity that contributed most to your learning.

Use each rank only once.

Table 34: Ranking of learning activities by Group Member No. 1

Activity	Rank
Dual-loop PID modelling and parameter tuning	
LQR modelling and weighting-matrix design	
MPC modelling, cost-function design, and parameter tuning	
DQN training and deployment	
PPO training and deployment	
TD3 training and deployment	
Comparing all controllers using common quantitative metrics	
Improving performance beyond the provided benchmark	
Simulation-to-real transfer and real-hardware debugging	
Writing technical reports and analysing repeated experiments	

Privacy reminder
This survey is identified only by Group Member No. 1. Do not include your name, student ID, email address, or other personally identifiable information.
Your responses will be used solely to improve future versions of the course. They will not affect your Report-3 mark or overall course grade.