Q-Learning Performance on the CartPole Environment Under Observation Noise and Reward Variants

Authors: Fan Wu, Steven Ren, Taieba Tasnim, Berkeley Wu, Mohammad Rahman

International Conference of Modern Systems Engineering Solutions
Modern Systems 2025

Presenter: Dr. Wu Fan
Head of Computer Sciences department at Tuskegee University, USA
fwu@tuskegee.edu

Dr. Fan Wu

Dr. Fan Wu is a professor and head of Computer Sciences department at Tuskegee University. Dr. Wu joined the faculty of Computer Sciences department at Tuskegee University as an assistant professor in 2009. He received his Ph.D. degree in Computer Science from Worcester Polytechnic Institute (WPI) in 2008.

Dr. Fan Wu's research interests are in broad areas of Mobile Security, Information Assurance, Data Science, Mobile Graphics, Mobile Computing, Computer Graphics, Bioinformatics, Biostatistics, High Performance Computing with GPGPU Technology, and Robotics.

Research Motivation and Contributions

- Reinforcement Learning (RL) enables agents to learn control strategies from interaction with their environment
- The CartPole task is a classical RL benchmark used to evaluate control algorithms
- In real world cyber-physical systems, factors like sensor noise and reward design can significantly affect learning stability and performance
- Evaluate Q-learning's behavior and policy stability under varying observation noise and different reward function
- Contributions:
  - Application of a Q-learning algorithm to CartPole under noisy observation inputs
  - Comparison of a standard reward function with a cosine-based reward function shaped by the pole angle
  - Evaluation of convergence episodes, pole angle statistics, and performance variance across noise levels
- Goal of improving understanding of how noise and reward design affect reinforcement learning for robust cyber-physical control

Related Work

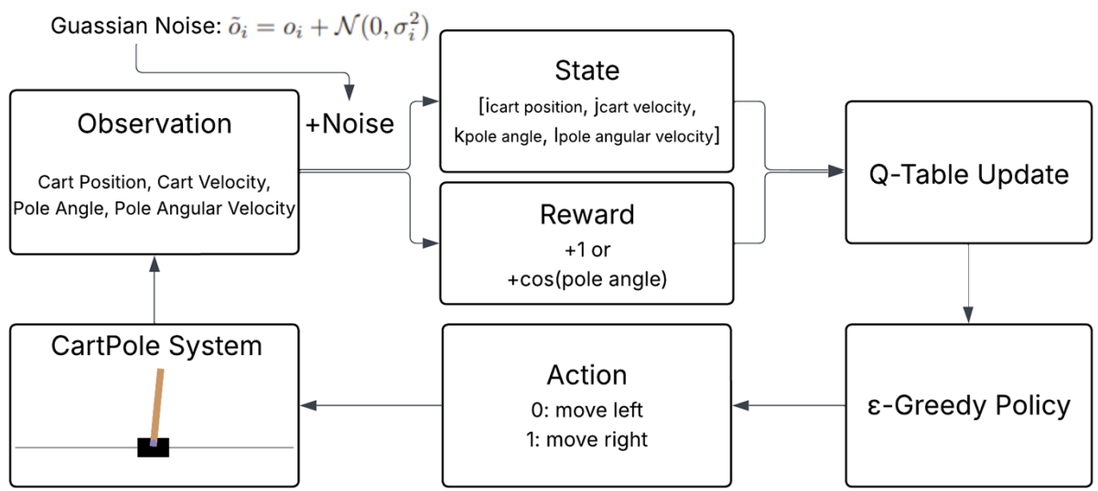
- Early Q-learning applications proved effective in robotics and control problems
- The ε-greey policy helps balance exploration and exploitation during learning
- Variants such as Efficient Q-Learning and Deep Q-Networks (DQN) improved scalability and performance
- Recent research emphasizes robustness under noise and uncertainty, especially for safety-critical systems.
  - Krish et al. analyzed observation noise effects in neural network controllers for systems like CartPole and LunarLander
  - Nazrul applied RL to optimize sampling frequency in cloud-based control systems for better efficiency and performance.
- This study also introduces SARSA as a baseline comparison—unlike Q-learning (off-policy), SARSA learns the value of the current policy (on-policy).

Q-Learning Overview

- Q-Leaning: Model-free RL algorithm that estimates the optimal action-value function Q(s, a)
- Update rule:
  - $$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha \left[r_{t+1} + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t) \right]$$
- Key parameters:
  - α: learning rate, controls update speed
  - γ: discount factor, weights future vs immediate rewards
  - ε: exploration probability in ε-greedy policy
- Learns through iterative exploration and exploitation, improving policy based on accumulated experience
- Particularly suited for discrete state and action spaces such as CartPole

CartPole Environment Description

- The CartPole problem models an inverted pendulum mounted on a moveable cart
- Goal: Keep the pole balanced upright by moving cart left or right
- Observation vector:
  - Cart position x
  - Cart velocity v
  - Pole angle θ
  - Pole angular velocity ω
- Actions:
  - 0 = Push cart left
  - 1 = Push cart right



- Episode ends when poles falls beyond a defined threshold or cart moves out of bounds
- Maximum episode reward of 500 indicates full balance or convergence

Observation Noise Modeling

- Real-world sensors produce uncertain or inaccurate readings, this is simulated by additive Gaussian noise
- Formula:
  - $\tilde{o}_i = o_i + \mathcal{N}(0, \sigma_i^2)$
- Tested noise levels: 0.0 (none), 0.01, 0.05, 0.1
- Each observation receives noise proportional to its range
- add here?
- Noise is applied before state discretization, potentially causing incorrect state classification or unstable learning transitions

State Discretization and Representation

- Q-learning requires a finite state space; thus, continuous observations are binned:
  - Cart position: 8 bins with range of [-4.8, 4.8]
  - Cart velocity: 8 bins with range of [-5.0, 5.0]
  - Pole angle: 20 bins with range of [-0.418, 0.418] radians
  - Pole angular velocity: 20 bins with range of [-10.0, 10.0]
- Each unique combination defines a state index in the Q-table
- Tradeoff:
  - More bins: increases state precision but at the cost of increased computation
  - Fewer bins: reduced computation but less detailed state representation
- Observation noise can cause transitions between neighboring bins, introducing non-determinism into state transitions

Reward Functions

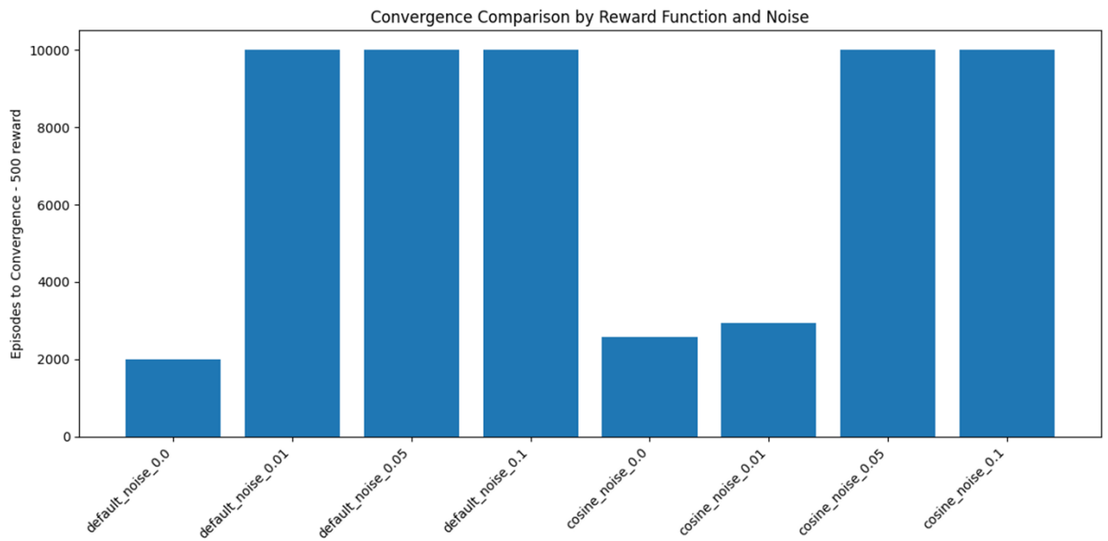
- Default reward: +1 per step while balanced
  - Simple heuristic of long survival is better
  - Encourages maximizing episode duration
  - Lacks explicit feedback for pole angle deviation
- Cosine-based reward: $r = \cos(\theta)$
  - Rewards upright pole (max = 1 at $\theta = 0$)
  - Penalized deviations
  - Encourages smoother, stable control behavior rather than just lasting longer

Experimental Setup

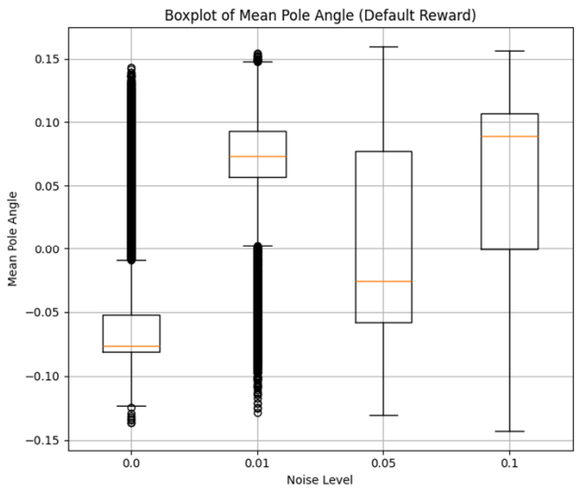
- Training setup:
  - 10,000 episodes, each capped at 500
- Q-learning Hyperparameters:
  - $\alpha = 0.1$ (learning rate)
  - $\gamma = 0.95$ (discount factor)
  - ε decays from 1.0 to 0.001
- Metrics recorded per episode
  - Total reward
  - Mean and variance of pole angle of each episode
- Baseline: SARSA algorithm for on-policy comparison trained under same settings

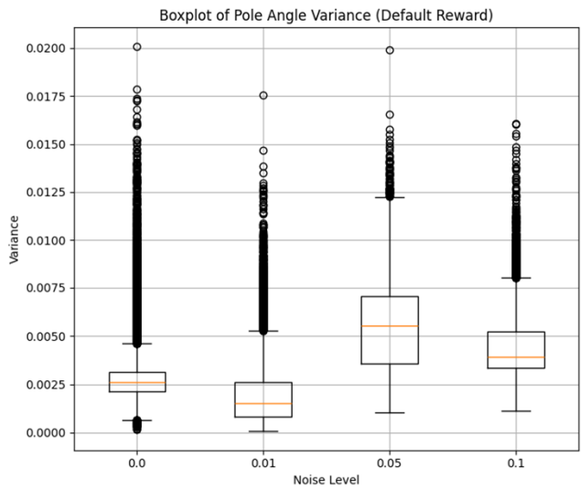
Q-Learning Results and Analysis

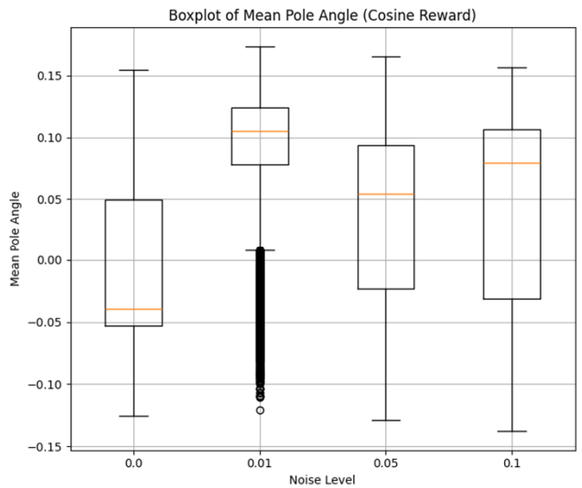
- Bar plot:
  - Convergence = reaching a total reward of 500
  - Shows when the training converges, does not converge if 10,000 episodes reached
- Box plot:
  - Default reward effected by noise significantly
  - Cosine reward better variance and fewer outliers
- Observation noise disrupts Q-learning performance under the default reward
- Cosine-based reward promotes robust, consistent control by penalizing large pole angles

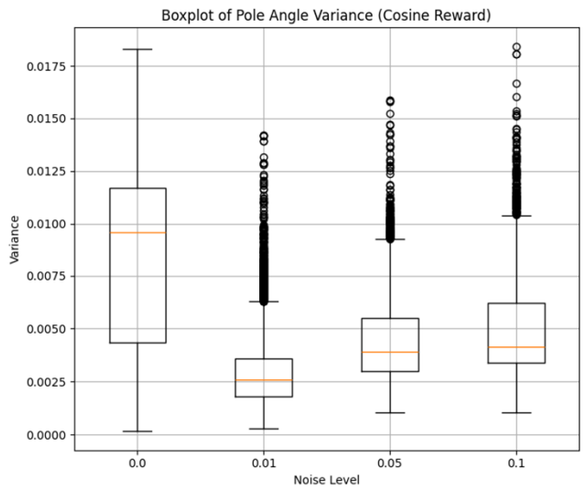


Q-Leaning Pole Angle Mean and Variance



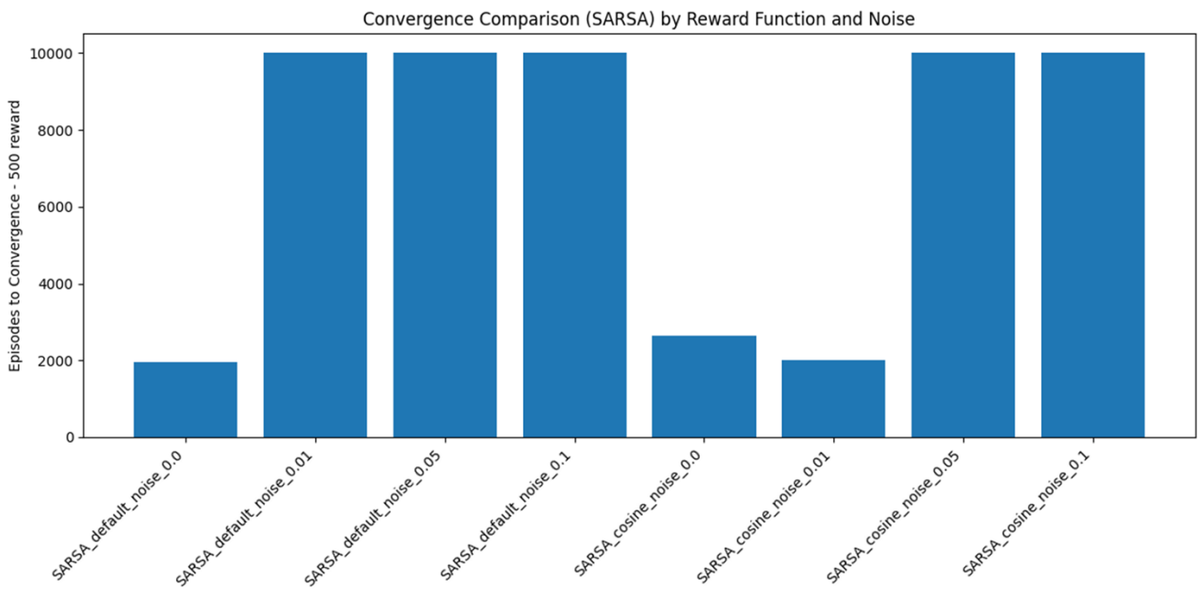




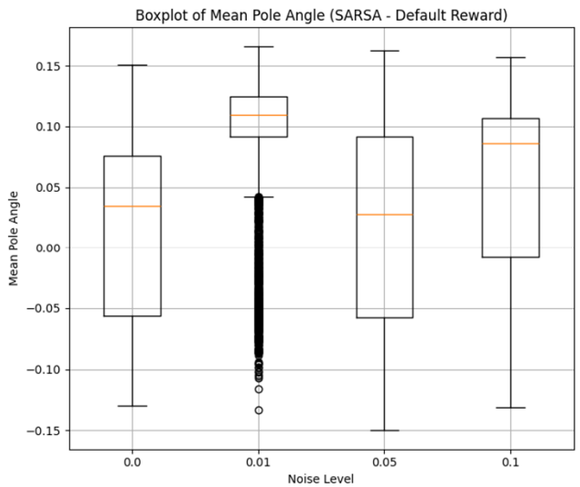


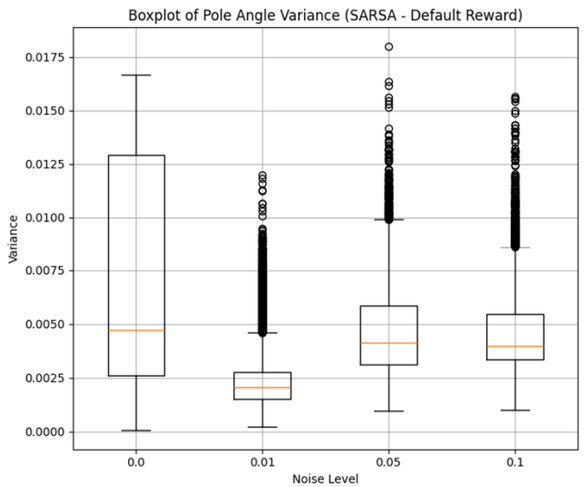
SARSA Comparison

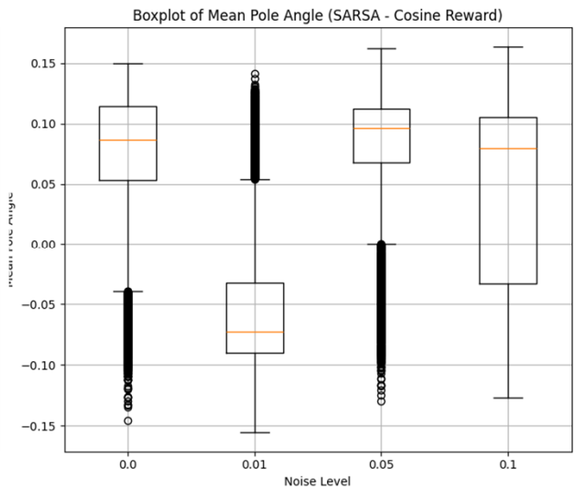
- SARSA shows similar overall patterns to Q-learning
- Convergence improved under cosine reward; stability worsened with higher noise
- Shows that a good reward choice positively affects stability under noise

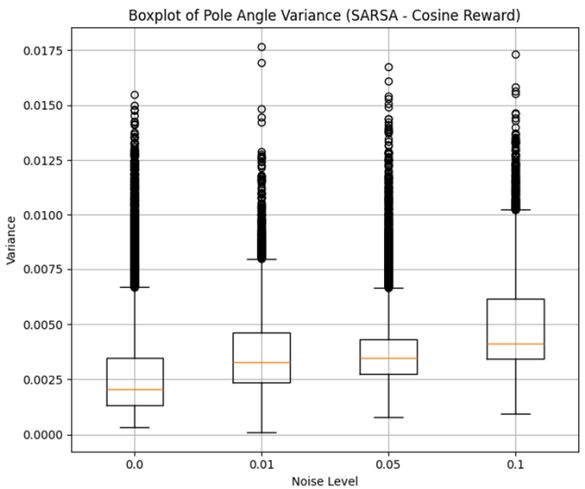


SARSA Pole Angle Mean and Variance









Conclusion and Future Work

- Observation noise significantly hinders learning under standard rewards
- Cosine-based reward improves robustness, convergence, and pole stability
- SARSA results show similar reward patterns
- Showing reward design is crucial for deploying RL in noisy, real-world systems
- Future directions:
  - Apply framework to physical hardware with real sensors for testing
  - Combine with noise filtering or adaptive learning strategies
  - Extend to Deep Q-Learning for continuous and scalable tasks
- This work contributes to robust RL for cyber-physical systems.