perplexity

Report: Analysis of Algorithm Effectiveness, Efficiency, and Stability in Wireless Channel Selection

Introduction

This study evaluates the performance of two reinforcement learning algorithms—Q-Learning and Deep Q-Networks (DQN)—in the context of wireless channel selection. The task involves selecting one of five channels with varying probabilities of success, aiming to maximize long-term rewards. The algorithms were compared based on their ability to approximate true channel probabilities, convergence behavior, and stability over 10,000 episodes.

Results and Observations

1. **Effectiveness**:
    - Both Q-Learning and DQN successfully learned to maximize rewards over time, as evidenced by their average rewards approaching the highest probability channel (0.9).
    - Q-Learning approximated the true channel probabilities more closely due to its tabular nature, which directly updates values for each action. This is particularly effective in environments with discrete states and actions.
    - DQN also performed well but exhibited slightly more variance in its learned Q-values due to the inherent stochasticity of neural network training.
2. **Efficiency**:
    - Q-Learning demonstrated faster convergence, stabilizing around 2,000 episodes. This is attributed to its simplicity and direct updates without relying on function approximation.
    - DQN required more episodes (~3,000–4,000) to converge due to factors such as replay memory sampling, neural network optimization dynamics, and target network synchronization delays.
3. **Stability**:
    - Q-Learning showed high stability after convergence with minimal oscillations in its learning curve.
    - DQN exhibited greater variability throughout training, especially in early episodes. Oscillations were caused by neural network training dynamics (e.g., gradient updates) and replay memory sampling. However, these oscillations diminished over time as the policy stabilized.

Insights

- **Algorithm Suitability**:
  - Q-Learning is better suited for this specific environment because it directly models discrete states and actions without requiring function approximation.
  - DQN's strength lies in generalizing across large or continuous state spaces; however, this environment does not require such generalization.
- **Exploration vs. Exploitation**:
  - The epsilon-greedy strategy worked effectively for both algorithms, gradually favoring exploitation as epsilon decayed. This allowed both algorithms to identify the optimal channel (with a success probability of 0.9).
- **Impact of Function Approximation**:
  - While DQN's neural network introduces flexibility for complex tasks, it also adds noise and instability in simpler tasks like this one. This highlights the importance of algorithm selection based on problem complexity.

Recommendations

1. For environments with discrete states and actions (like wireless channel selection), simpler methods like Q-Learning are preferable due to their efficiency and stability.
2. For DQN:
   - Use a larger replay memory or batch size to reduce variance during training.
   - Experiment with more frequent target network updates to stabilize learning.
3. Future work could explore hybrid approaches that combine tabular methods with neural networks for environments that mix discrete and continuous states.

Conclusion

Both Q-Learning and DQN effectively solved the wireless channel selection problem by approximating true channel probabilities and maximizing long-term rewards. However, Q-Learning outperformed DQN in terms of efficiency and stability due to its simplicity and direct updates. These findings emphasize the importance of selecting an algorithm that aligns with the complexity of the environment.

⁂