RL – CIA 2

Comparison of Reinforcement Learning Methods: Policy Iteration vs. Q-Learning

Policy Iteration

Policy Iteration is a model-based reinforcement learning algorithm that iteratively improves a policy until convergence. The process involves two main steps:

1. **Policy Evaluation**: For a given policy, calculate the value function for all states, which indicates the expected return starting from that state and following the policy thereafter.
2. **Policy Improvement**: Update the policy by choosing the action that maximizes the value function for each state.

The process continues until the policy stabilizes, meaning no further changes occur during the improvement step.

Findings

- **Execution Time**: 14.4119 seconds
- **Effectiveness**: Policy Iteration often converges quickly to an optimal policy since it evaluates and improves the policy in a systematic way.

Q-Learning

Q-Learning is an off-policy, model-free reinforcement learning algorithm that learns the value of action-state pairs. Unlike Policy Iteration, it does not require a model of the environment, making it applicable in scenarios where the transition dynamics are unknown.

1. **Action Selection**: At each step, the agent chooses an action based on an exploration-exploitation strategy (e.g., ε-greedy).
2. **Q-Value Update**: After taking an action and observing the resulting reward and next state, the Q-value for the state-action pair is updated using the Q-learning formula.

Findings

- **Execution Time**: 71.2443 seconds

- **Effectiveness**: Despite taking longer to execute, Q-Learning was able to yield a better result in terms of policy quality, demonstrating its robustness, especially in complex environments.

Comparison with Dynamic Programming

Dynamic Programming (DP) techniques, such as Value Iteration and Policy Iteration, are foundational algorithms that leverage a complete model of the environment's dynamics. DP methods typically provide optimal policies for known environments but are often computationally expensive and may not scale well to large state spaces due to the "curse of dimensionality."

General Performance of DP Approaches

- **Pros**:
    - Guarantees finding the optimal policy.
    - Efficient when the environment model is known and manageable.
- **Cons**:
    - Requires exhaustive computations, making it impractical for large state spaces.
    - Not suited for online learning scenarios where the model is not known beforehand.

In practice, while DP approaches can yield optimal solutions, they often struggle in environments with large state spaces. In contrast, Q-Learning's ability to learn from experience and adapt makes it more suitable for complex, real-world applications.

Conclusion

Both Policy Iteration and Q-Learning have their strengths and weaknesses. Policy Iteration is efficient and converges quickly, while Q-Learning provides flexibility and robustness at the cost of increased computational time. Although the DP approach remains a reliable method for smaller environments with known dynamics, it may not be the best choice in larger, more complex scenarios.