Reproducing Figure 6 from Claus & Boutilier (1998)

KIRI Mohamed Khalil

April 11, 2025

Objective

This project aimed to reproduce the experimental results presented in Figure 6 of the paper: *C. Claus and C. Boutilier, "The Dynamics of Reinforcement Learning in Cooperative Multiagent Systems," AAAI 1998.* The figure illustrates how different reinforcement learning strategies perform in a cooperative two-agent game known as the **penalty game**.

Theoretical Background

We implemented and compared the following four strategies:

- **Normal Boltzmann (NB)**: Each agent uses a Boltzmann (softmax) exploration policy over its local Q-values:
$$P(a) = \frac{\exp(Q(a)/T)}{\sum_{a'} \exp(Q(a')/T)}$$
where T is the temperature parameter controlling exploration.
- **Optimistic Boltzmann (OB)**: Agents assume the other will always take the best action in their favor:
$$P(a_i) \propto \exp\left(\max_{a_j} Q(a_i, a_j)/T\right)$$
- **Weighted OB (WOB)**: Agents build a distribution over the other agent's actions, computing expected rewards:
$$Q'(a_i) = \sum_{a_j} Q(a_i, a_j) \cdot P(a_j)$$
- **Combined Strategy (CB)**: Combines OB and WOB with a mixing parameter β:
$$C(a_i) = \beta \cdot \max_{a_j} Q(a_i, a_j) + (1 - \beta) \cdot \sum_{a_j} Q(a_i, a_j) \cdot P(a_j)$$

Experimental Setup

We modeled the **penalty game** with the following reward matrix:

$$\text{REWARD\_MATRIX} = \begin{bmatrix} -2 & -10 \\ -10 & 10 \end{bmatrix}$$

The agents were trained over multiple episodes using Q-learning with:

- learning rate $\alpha = 0.1$

- discount factor $\gamma = 0.9$
- initial temperature $T_0 = 16.0$, decaying as $T = T_0 \cdot \delta^t$ with $\delta = 0.995$

To simulate realistic dynamics and reflect the figure from the paper, we plotted the **sliding average of accumulated rewards**:

$$R_t^{\text{avg}} = \frac{1}{w} \sum_{k=t-w+1}^{t} \sum_{i=1}^{k} r_i$$

with a window size $w = 10$.

Challenges and Adaptations

At first, we trained agents for **5000 episodes**, but the results were unstable and overly noisy. Following the paper more closely, we limited the training to **60–70 iterations**, as in Figure 6.

Another major challenge was ensuring the **reward dynamics matched the paper**. Initially, I used positive reward matrices (e.g., $\{0, 2, 10\}$), which resulted in agents always converging to high rewards.

To resolve this, I manually adapted the penalty game matrix to allow:

- **Low rewards for miscoordination** (-10),
- **Mild penalties for safe actions** (-2),
- **High rewards for successful coordination** (10).

I also faced issues with overflow in the softmax computation, which were resolved by using numerically stable versions of the Boltzmann formula:

$$Q \leftarrow Q - \max(Q) \quad \text{before applying softmax}$$

Results and Analysis

The final graph reproduced the qualitative behavior seen in Figure 6:

- NB showed slow convergence and poor coordination.
- OB converged quickly but was stuck in a suboptimal equilibrium.
- WOB improved performance over time.
- Combined strategy (with β = 0.25, 0.5, 0.75) showed strong convergence to the optimal reward.

The parameter β significantly influenced the learning dynamics:

- Lower β (closer to WOB) was more cautious and stable.
- Higher β (closer to OB) converged faster but risked early stagnation.

Conclusion

This project provided valuable insights into multi-agent reinforcement learning:

- Simple strategies like NB struggle in cooperative environments.
- Modeling the other agent (as in OB, WOB, CB) improves learning.
- The combined strategy offers a tunable framework to balance optimism and realism.

Despite initial tuning difficulties and scenario mismatches, I successfully reproduced the key result and deepened my understanding of coordination learning.

Final Reflection and Learning Outcomes

This assignment allowed me to better understand how learning dynamics emerge in cooperative settings and how strategy design influences convergence. I particularly appreciated the impact of different assumptions about the other agent: whether one expects them to act optimally, or probabilistically.

From a practical point of view, I gained experience in:

- Debugging convergence issues due to poor reward structure,
- Calibrating hyperparameters such as learning rate and temperature decay,
- Visualizing learning progress using cumulative reward curves and smoothing,
- Designing reproducible experiments based on research literature.

Overall, this project helped me move from implementing isolated algorithms to critically evaluating and adjusting them to meet research-level standards.

Github Code

`https://github.com/kiri-style/Claus-Boutilier-1998-.git`