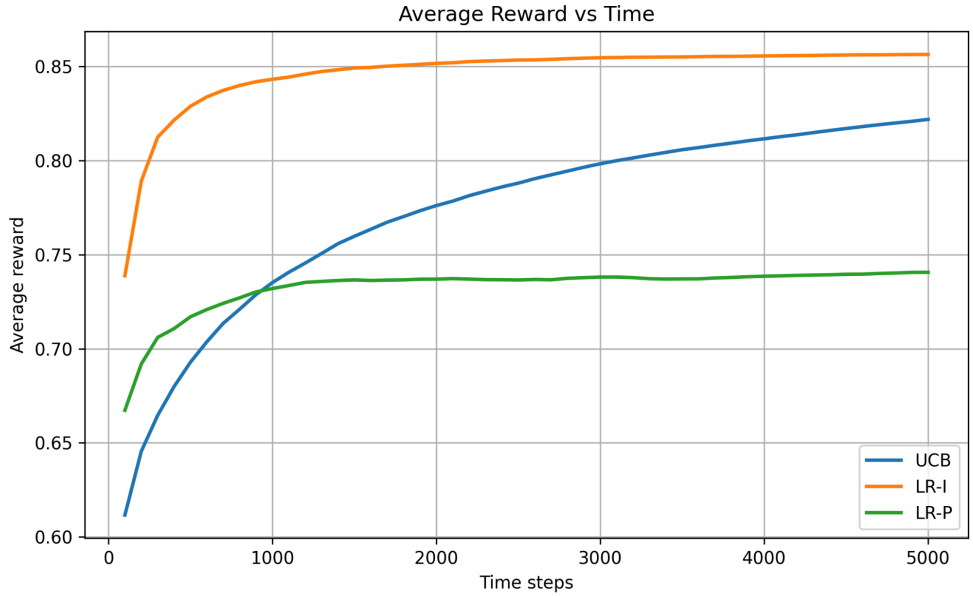
Arshia Zakeri Rad
Tigran Tumanyan

Report 2: Comparison of UCB, LR-I, and LR-P

Observations:

Average Reward vs Time



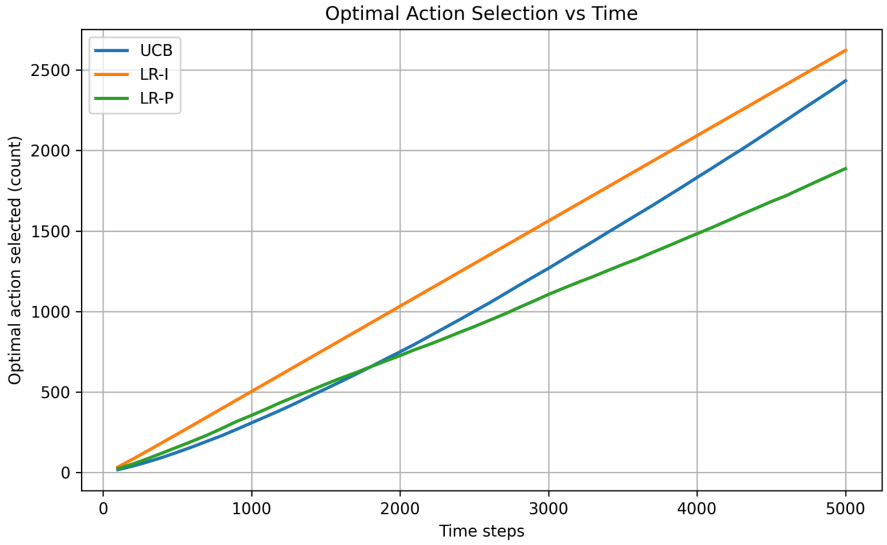
We made a python script to plot from a CSV file that our cpp code produces (both plots)

From the average reward plot, the LR-I algorithm achieves the highest average reward throughout the experiment. It improves very quickly in the early stages and converges to a stable reward of approximately 0.85. This indicates fast learning when only reward-based updates are used.

The UCB algorithm starts with the lowest average reward due to systematic exploration but increases steadily over time. Although its convergence is slower than LR-I, it eventually reaches a relatively high average reward (around 0.82), showing reliable long-term performance.

The LR-P algorithm improves early but plateaus at a lower average reward (around 0.74). Penalizing actions after failures appears to slow learning and limits its final performance.

Optimal Action Selection vs Time



The optimal action selection plot shows that LR-I selects the optimal action most frequently over time. Its curve grows faster and remains above the other algorithms, indicating quicker identification and exploitation of the optimal action.

The UCB algorithm initially selects the optimal action less often due to enforced exploration, but its performance steadily improves and approaches LR-I as time increases.

The LR-P algorithm selects the optimal action the least frequently, confirming that its penalty-based updates reduce learning efficiency.