Replacing Reward Normalization with Kalman Filtering in Reinforcement Learning

Zixuan Xia Quanxi Li

University of Bern
Master in Computer Science

May 20, 2025

Outline

Why Normalize Rewards? Why Replace It?

- Rewards in RL are noisy and non-stationary
- Normalization (mean/std) is:
  - Ad-hoc and sensitive to outliers
  - Unstable in sparse reward environments
- Goal: Replace reward normalization with a principled method

Why Kalman Filter Instead of Mean?

Theoretical Rationale:

- Kalman Filter is an optimal recursive estimator under Gaussian noise
- Estimates running mean with uncertainty tracking
- Adapts to non-stationary reward distributions

Compared to Mean Normalization:

- Batch mean is sensitive to outliers and ignores history
- Kalman filter combines prior belief with new evidence adaptively
- Effective when reward is noisy or sparse

Intuition Behind Kalman Baseline

- Think of Kalman filter as a weighted moving average:
  - More weight on stable past estimates when noise is high
  - More reactive when recent signal is reliable
- Like a learnable scheduler for the baseline
- Smoothes the reward estimate, leading to:
  - Lower variance in gradient
  - More stable learning

Sample Mean vs. Kalman Estimator (MSE)

- Let true reward: r_{true}, observed: $\hat{r}_t = r_{\text{true}} + v_t$
- **Sample Mean:**

$$\bar{r}_t = \frac{1}{t}\sum_{i=1}^{t} \hat{r}_i, \quad \text{MSE} = \frac{\sigma^2}{t}$$

- **Kalman Filter:**

$$r_t = r_{t-1} + w_t, \quad \hat{r}_t = r_t + v_t$$

- MSE converges to steady-state:

$$P_\infty = \frac{\sqrt{Q^2 + 4QR} - Q}{2}$$

Comparison of MSE Behavior

- **Sample Mean:**

$$\text{MSE}(\bar{r}_t) = \frac{\sigma^2}{t} \to 0 \text{ (slow)}$$

- **Kalman:**

$$\text{MSE}(r_t) \to P_\infty < R \text{ (fast)}$$

- When is Kalman better?

$$\frac{\sigma^2}{t} > P_\infty \quad \Rightarrow \quad t < \frac{\sigma^2}{P_\infty}$$

- **Small t regime:** Kalman filter has lower error and faster convergence

Why Kalman is Suited for RL?

- RL often begins with:
  - Few samples (early training)
  - Sparse rewards (e.g., only a few non-zero rewards)
- In this regime:
  - Sample mean is unstable
  - Kalman filter quickly stabilizes to reliable estimate
- Especially helpful in:
  - On-policy learning
  - Exploration-heavy tasks (e.g. Atari, sparse navigation)

Using Kalman Filtering for Reward Estimation

- Model reward as a hidden state:

$$r_{k|k-1} = r_{k-1}, \qquad P_{k|k-1} = P_{k-1} + Q$$

$$K_k = \frac{P_{k|k-1}}{P_{k|k-1} + R}, \qquad r_k = r_{k|k-1} + K_k(\hat{r}_k - r_{k|k-1})$$

$$P_k = (1 - K_k)P_{k|k-1}$$

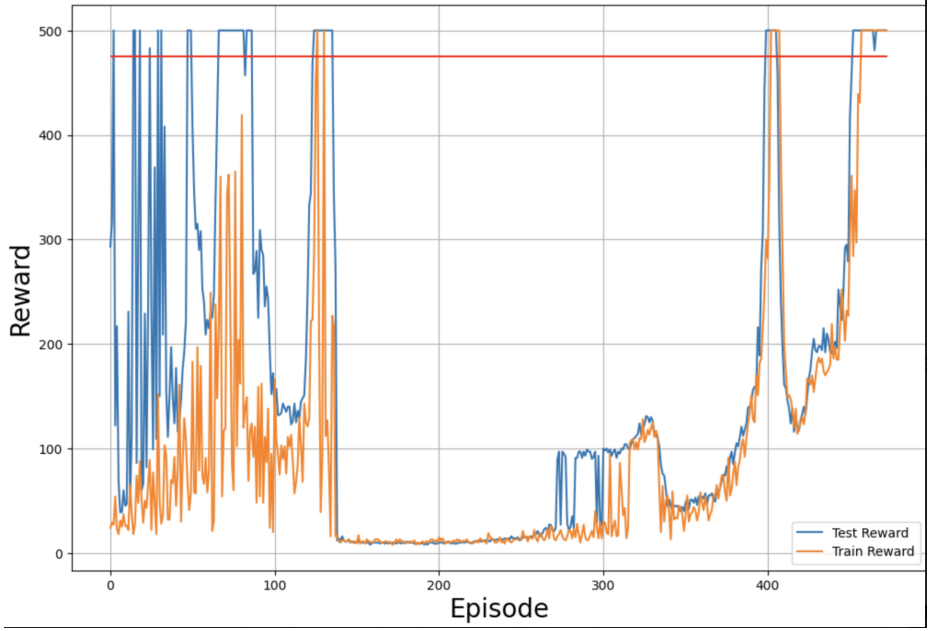
- Recursive and adaptive
- Replaces normalization with smoothed reward signal

System Integration

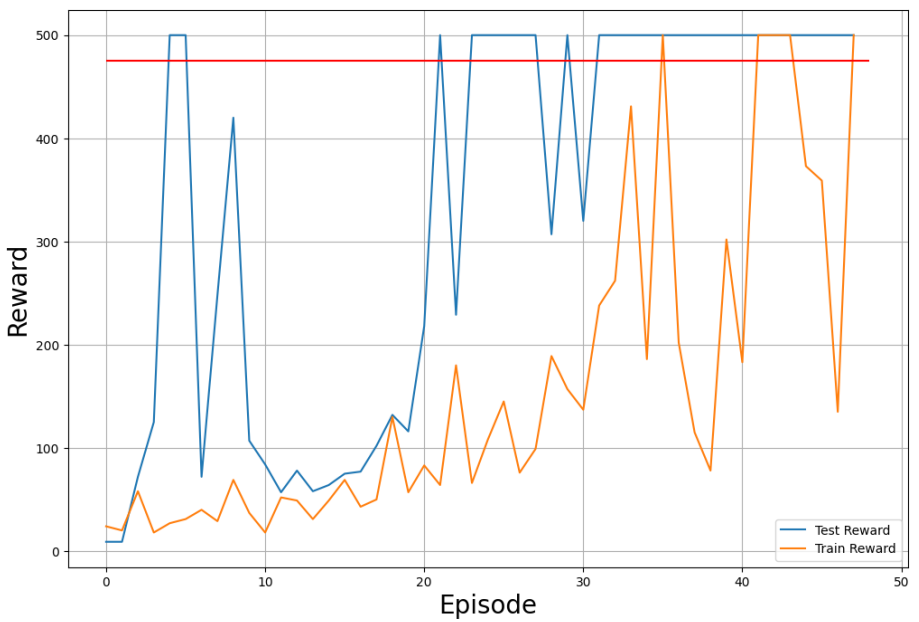
- Kalman filter update called every reward step
- Minimal change to training loop
- Parameters Q and R can be grid-searched or fixed

Experimental Setup

- Environments: LunarLander, CartPole
- Methods compared:
  - Normalized reward
  - Kalman-filtered reward (ours)

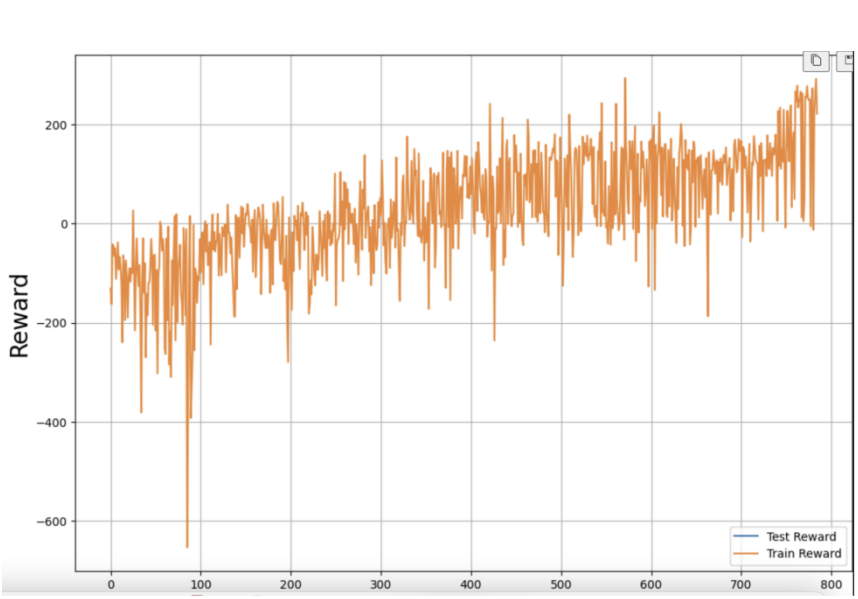


(a) Original Normalization

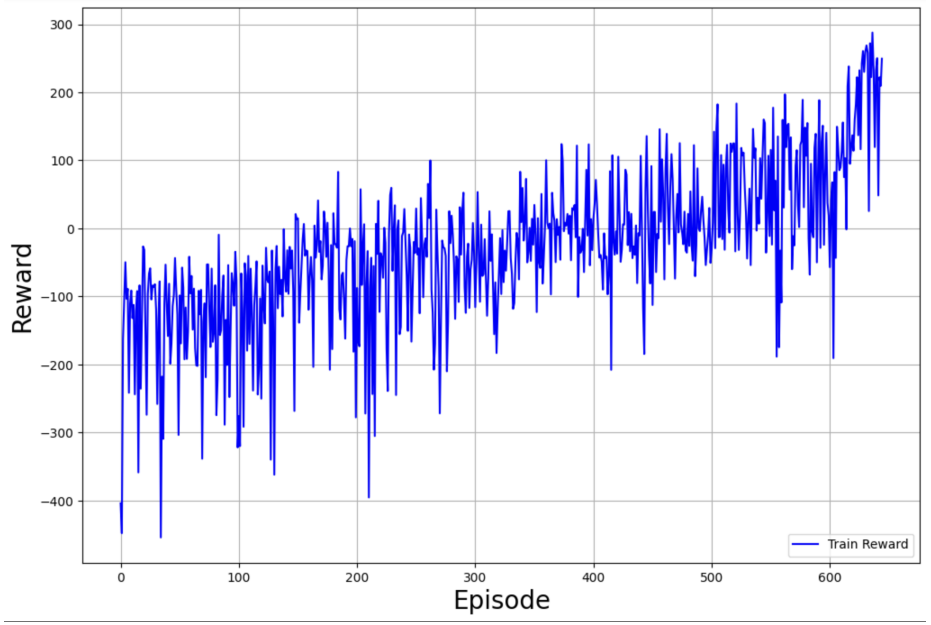


(b) Kalman Filter Normalization

Figure: Training and test reward curves across episodes(Cart-Pole).



(a) Original Normalization



(b) Kalman Filter Normalization

Figure: Training and test reward curves across episodes(Lunar-lander).

Advantages of Kalman Filtering

- Smooth reward signal
- Robust to outliers and variance
- Easy to integrate into existing RL pipelines

Conclusion and Future Work

- Kalman filtering is a principled alternative to normalization
- Shows empirical improvements in stability and convergence
- Future work:
  - Integrate into PPO
  - Learn noise parameters Q, R adaptively
  - Extend the normalization to some other tasks(E.g.: Deep Learning)

Thank you!

zixuan.xia@students.unibe.ch