

## PROBLEM

- Learning is difficult in highly stochastic environments
- Uncertainty in action-value function estimates propagates
- Some algorithms face this problem focusing on the bias of the estimate

## CONTRIBUTIONS

1. Split the estimate in two components:
  - The expected reward  $\tilde{R}(x, u)$
  - The expected next state value function  $\tilde{Q}(x, u)$
2. Use different learning rates for the two components
3. We provide empirical results showing the effectiveness of our approach

## RQ-LEARNING ALGORITHM

### IDEA

To improve estimates, exploit:

- The structure of the Bellman update
- The uncertainty of the estimation

### APPROACH

Split the action-value function in two components:

- $\tilde{R}(x, u) = \mathbb{E} [r(x, u, x')]$   
 $x' \sim \mathcal{P}(x' | x, u)$
- $\tilde{Q}(x, u) = \mathbb{E} \left[ \max_{x'} Q^*(x', u') \right]$   
 $x' \sim \mathcal{P}(x' | x, u)$
- $Q^*(x, u) = \tilde{R}(x, u) + \gamma \tilde{Q}(x, u)$

Compute the update as follows:

- $\tilde{R}_{t+1}(x, u) \leftarrow \tilde{R}_t(x, u) + \alpha_t (R(x, u, x') - \tilde{R}_t(x, u))$
- $\tilde{Q}_{t+1}(x, u) \leftarrow \tilde{Q}_t(x, u) + \beta_t (\max_{u'} Q_t(x', u') - \tilde{Q}_t(x, u))$

Exploit the variance of estimation to set the learning rate:

1. Estimate the variance of the estimator  $\tilde{Q}$ , using the sample variance of the target:

$$\text{Var} [\tilde{Q}] = S_t^2 \omega_t$$

$$\omega_{t+1} = (1 - \beta_t)^2 \omega_t + \beta_t^2$$

2. Compute the learning rate

- Select a  $\beta$  that decreases when the estimate precision increases:

$$\beta_t = \frac{\sigma_e^2(t)}{\sigma_e^2(t) + \eta}$$

- Or, select a  $\delta$  that increases when the estimate precision increases:

$$\delta_t = e^{\frac{\sigma_e^2}{\eta} \log \frac{1}{2}}$$

### ALGORITHMS

Different effects on the choice of  $\alpha$  and  $\beta$ :

**Q-Learning**  $\alpha_t = \beta_t$

**RQ $_{\delta}$ -Learning**  $\beta_t = \alpha_t \delta_t$

**RQ-Learning**  $\beta_t \neq \alpha_t$

## EMPIRICAL RESULTS

