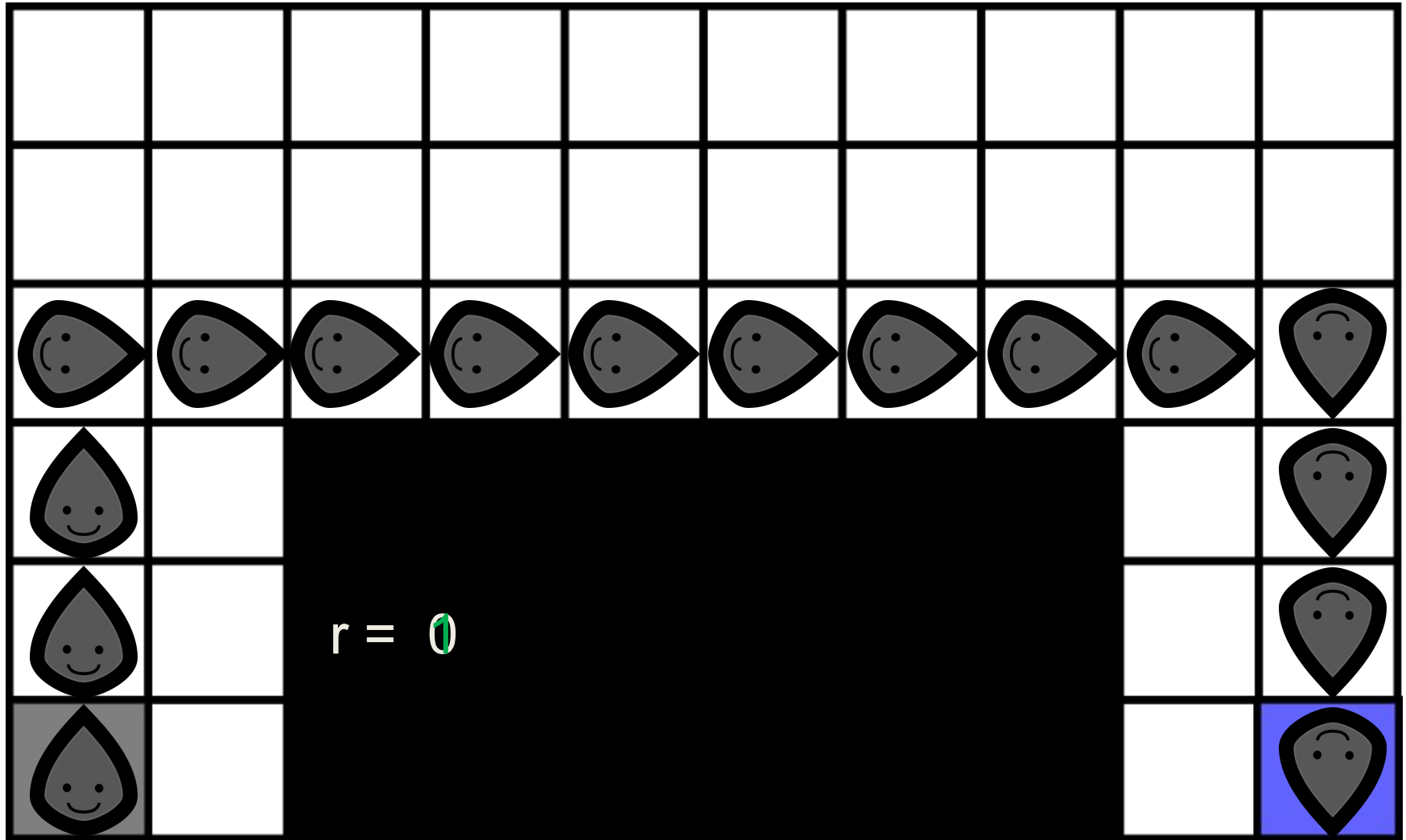
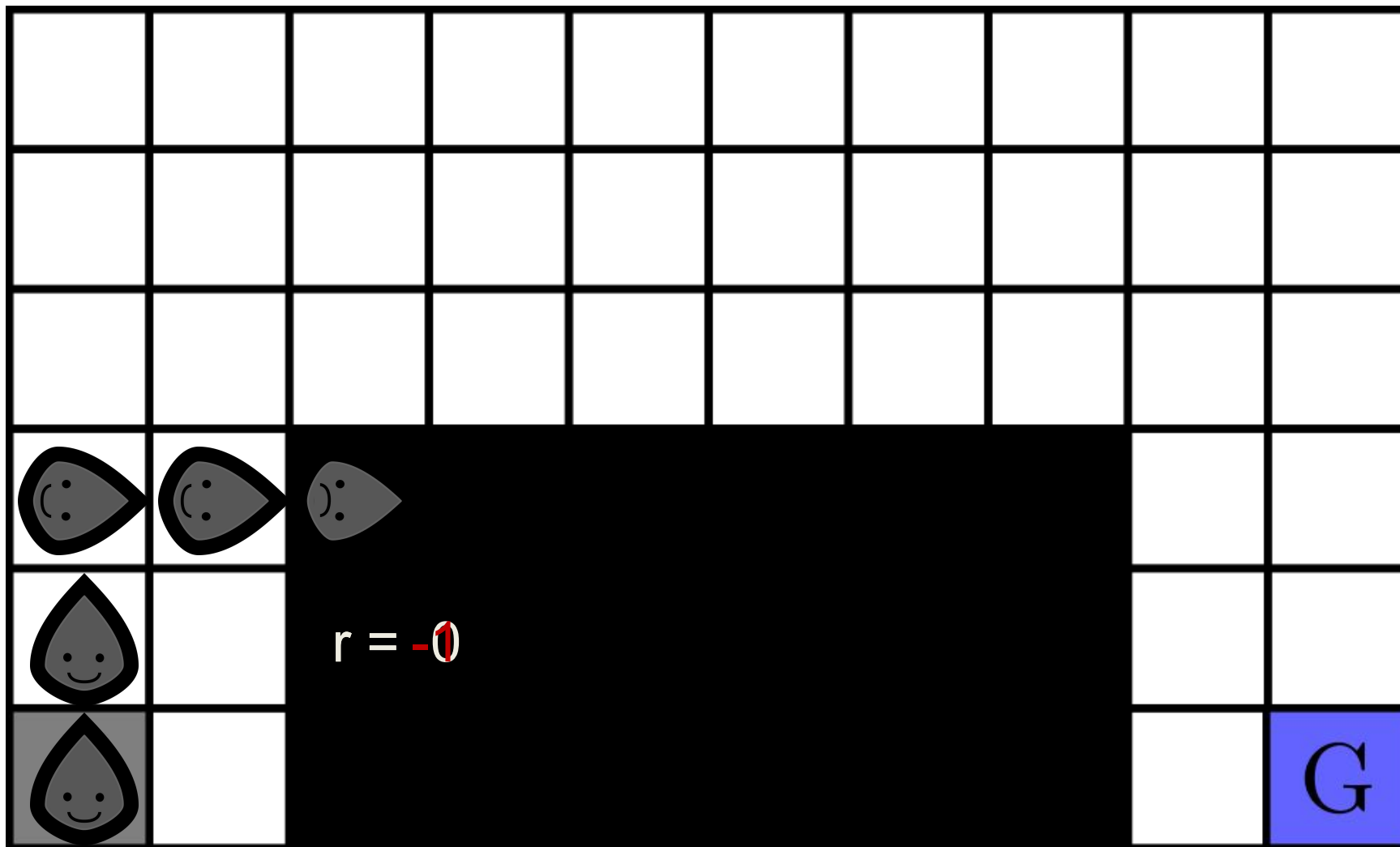


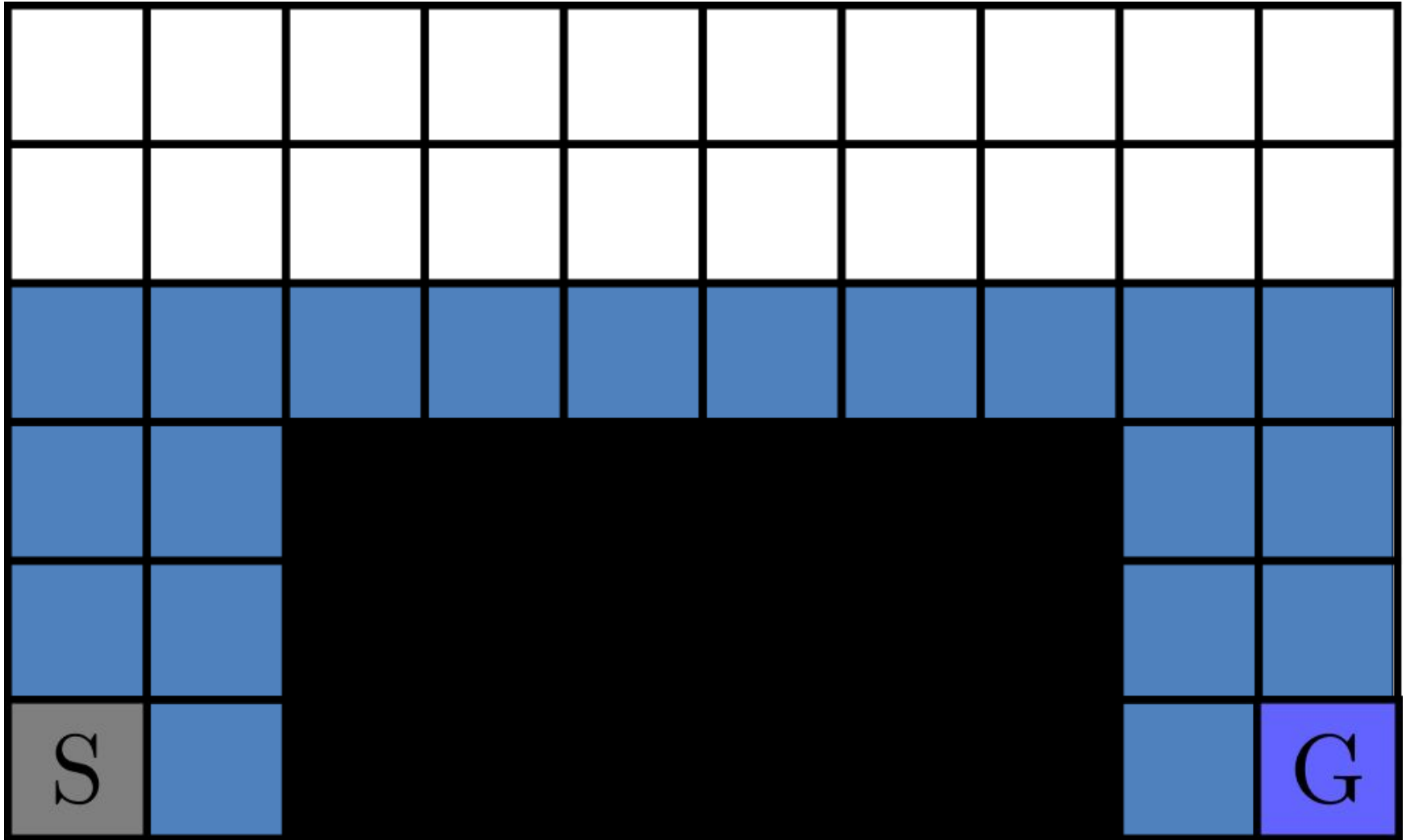
TempoRL: Learning When to Act

André Biedenkapp, Raghu Rajan,
Frank Hutter & Marius Lindauer

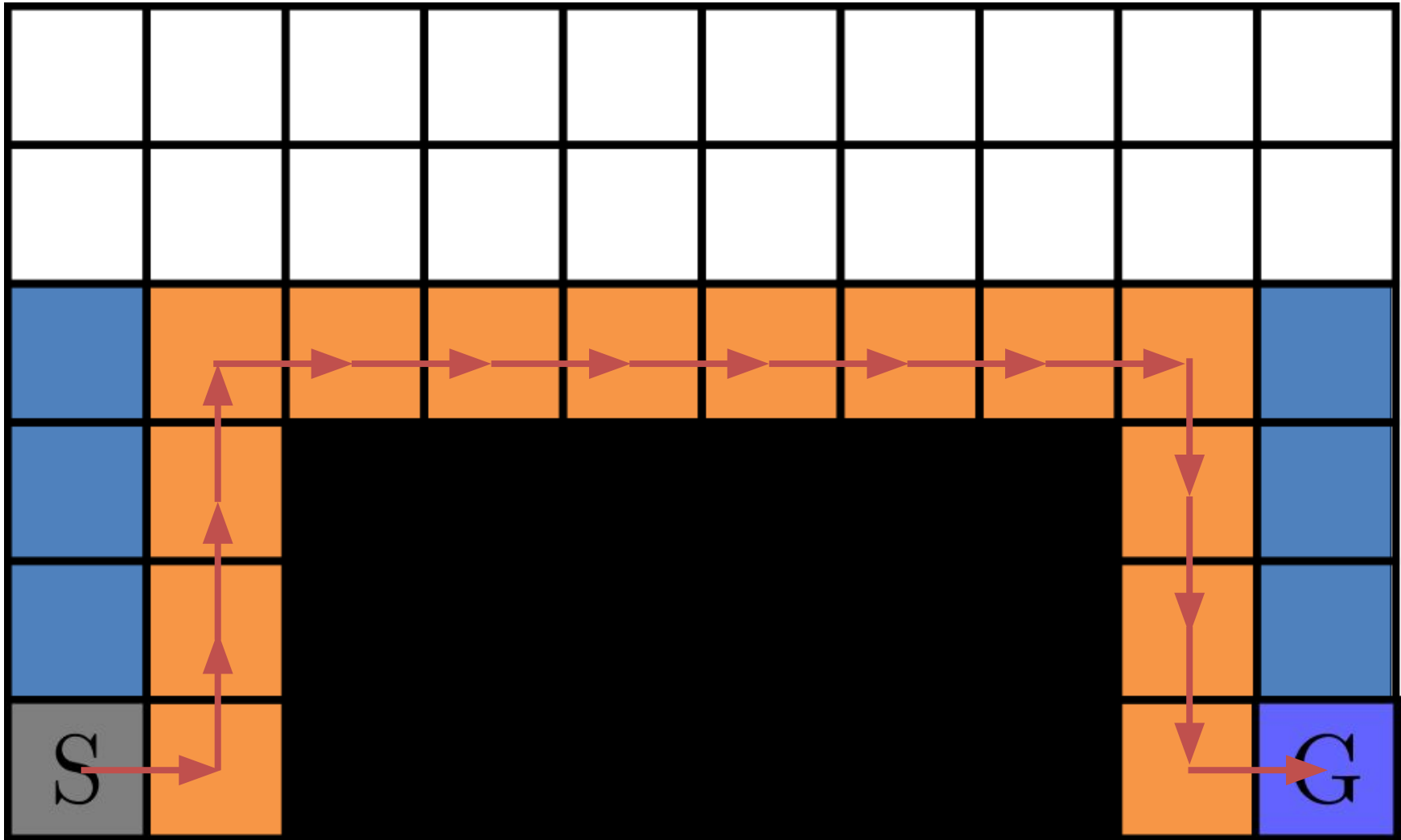
1. We propose a proactive way of doing RL
2. We introduce skip-connections into MDPs
 - use of action repetition
 - faster propagation of rewards
3. We propose a novel algorithm using skip-connections
 - learn *what* action to take & *when* to make a new decision
 - condition *when* on *what*
4. We evaluate our approach with in a variety of settings
 - tabular Q-learning on Gridworlds
 - DQN on featurized environments
 - DDPG on featurized environments
 - DQN with image states on Atari environments



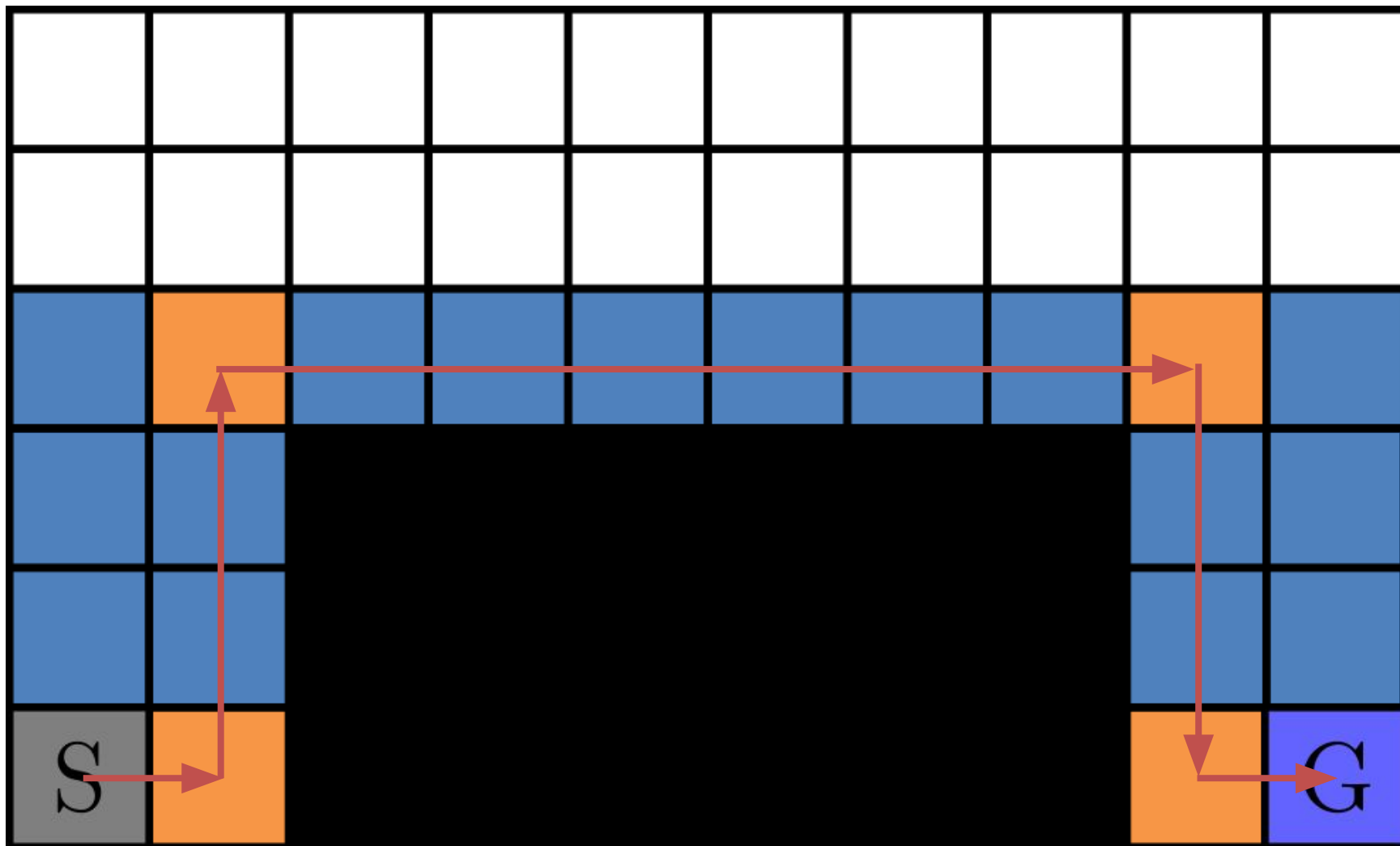




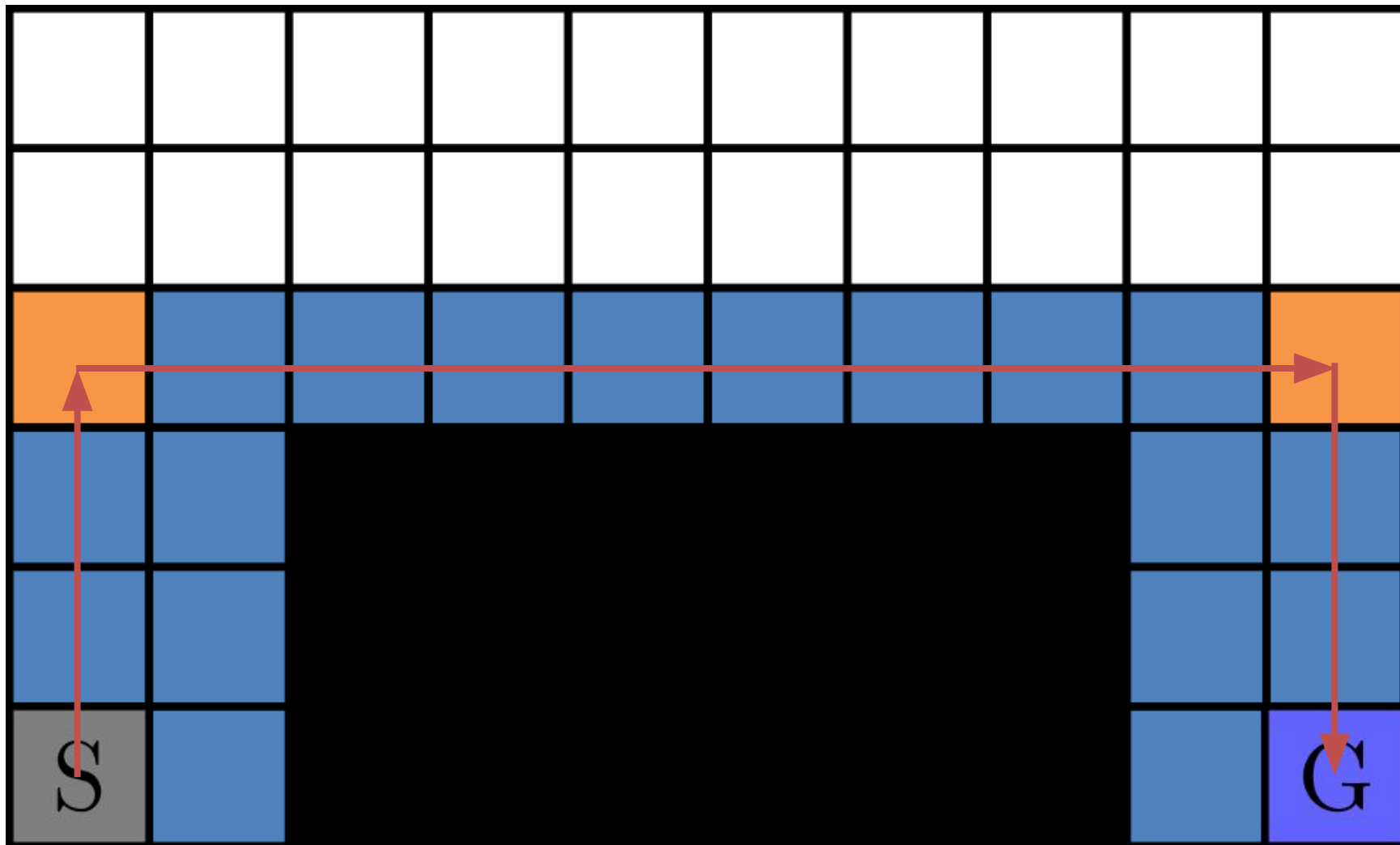
- Optimal policies will only cross the blue shaded area.



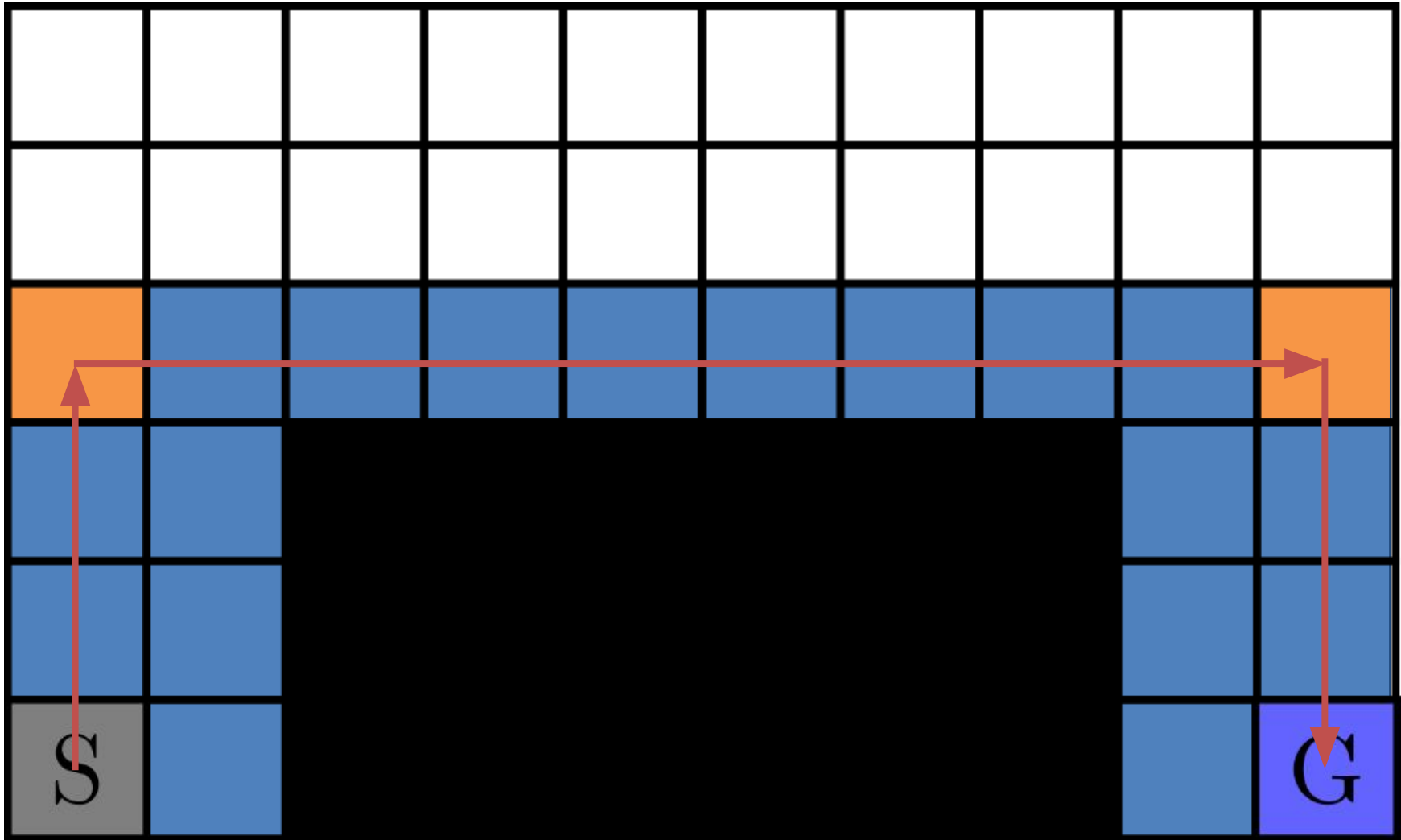
- Example trajectory of an optimal policy requiring **# Steps: 16**
Decisions: 16



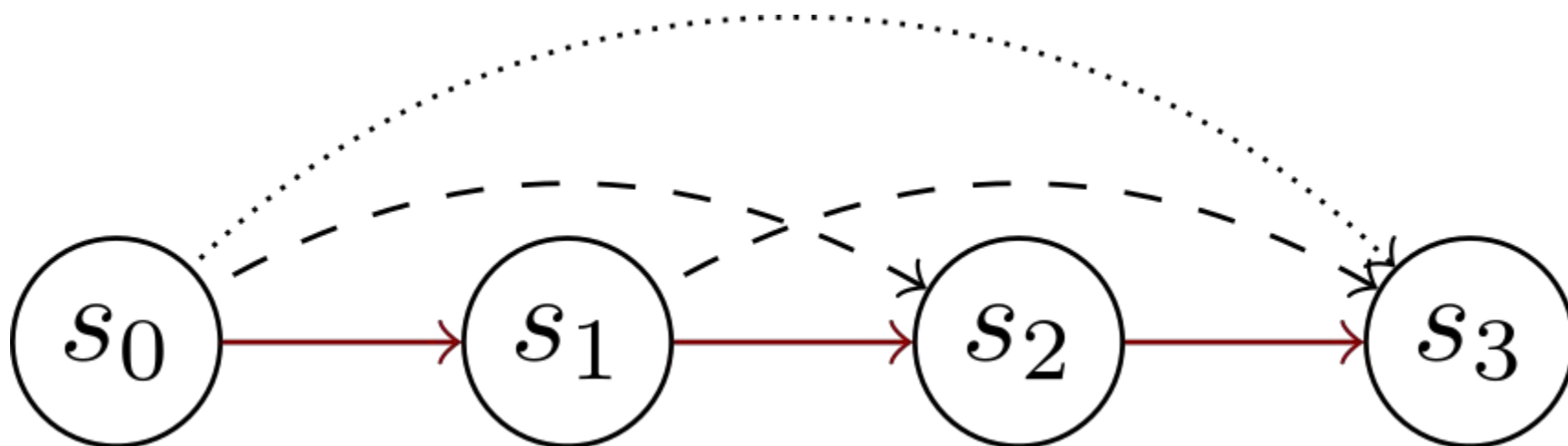
- Simplified trajectory of an optimal policy requiring $\# \text{ Steps: } 16$
 $\# \text{ Decisions: } 5$



- Simplified trajectory of an optimal policy requiring **# Steps: 16**
Decisions: 3



- Proactive decision making requires **~80% fewer decisions**
- Much simpler policies



- Action repetition induces skips
- Information can be propagated faster along skips
- With large skips, multiple smaller skips can be observed

1. Use standard agent (e.g. Q-learning) to determine the behaviour given the state

$$Q^{\pi}(s_t, a) \longrightarrow a$$

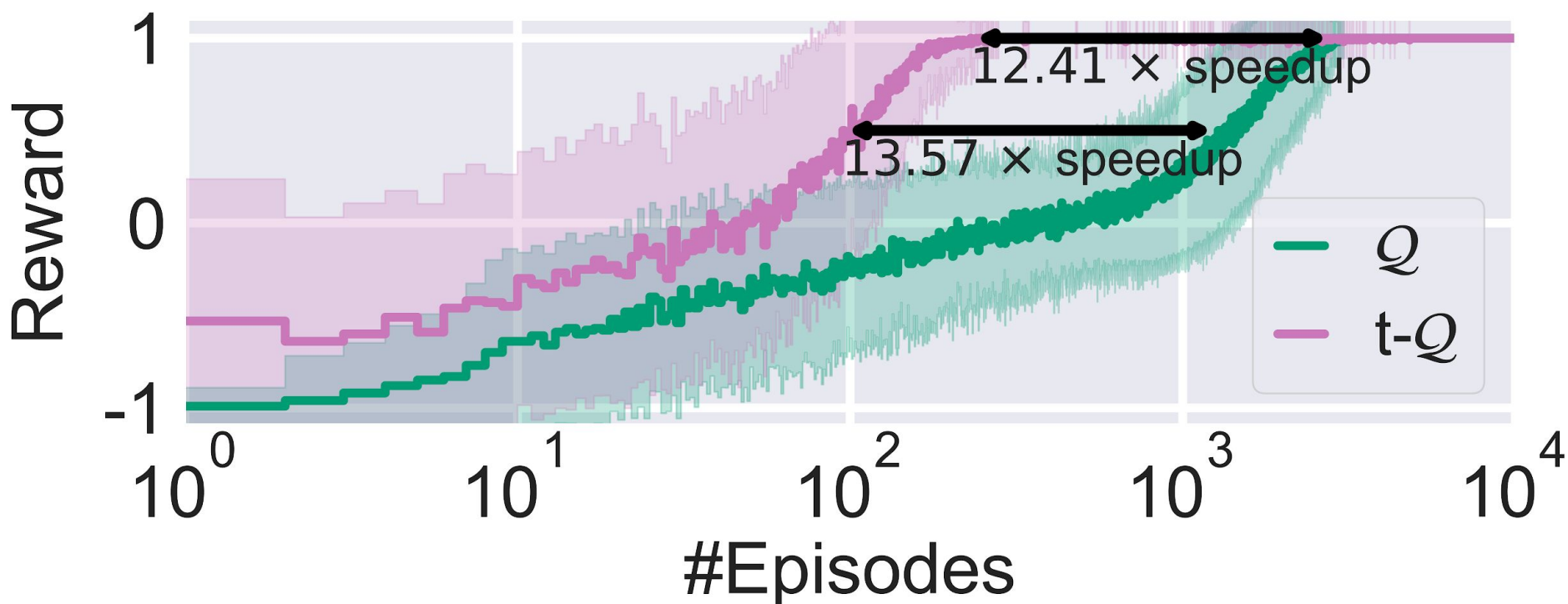
2. Condition skips on the chosen action

$$Q^{\pi_j}(s_t, j|a) \longrightarrow j$$

3. Play action a for the next j steps

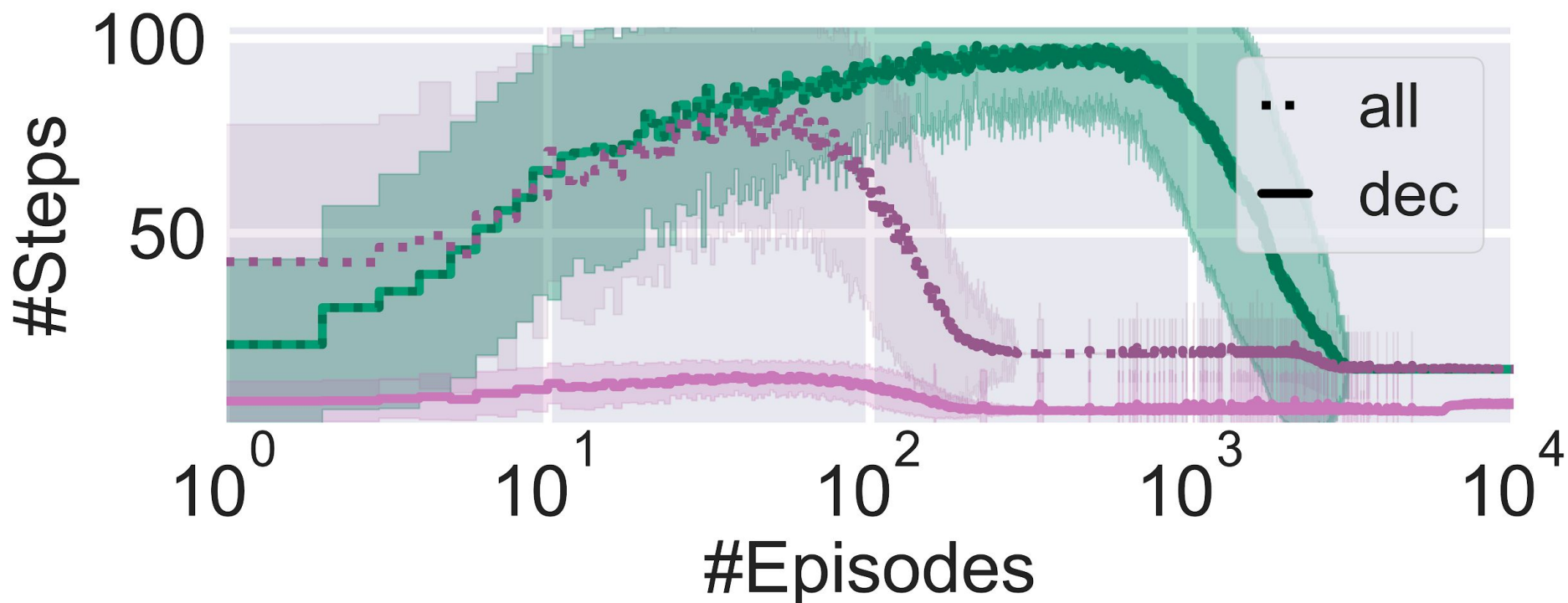
- Behaviour policy can be learned with vanilla agents
- The skip Q-function can be learned using n-step updates

- Comparison of vanilla and TempoRL Q-learning on the example gridworld



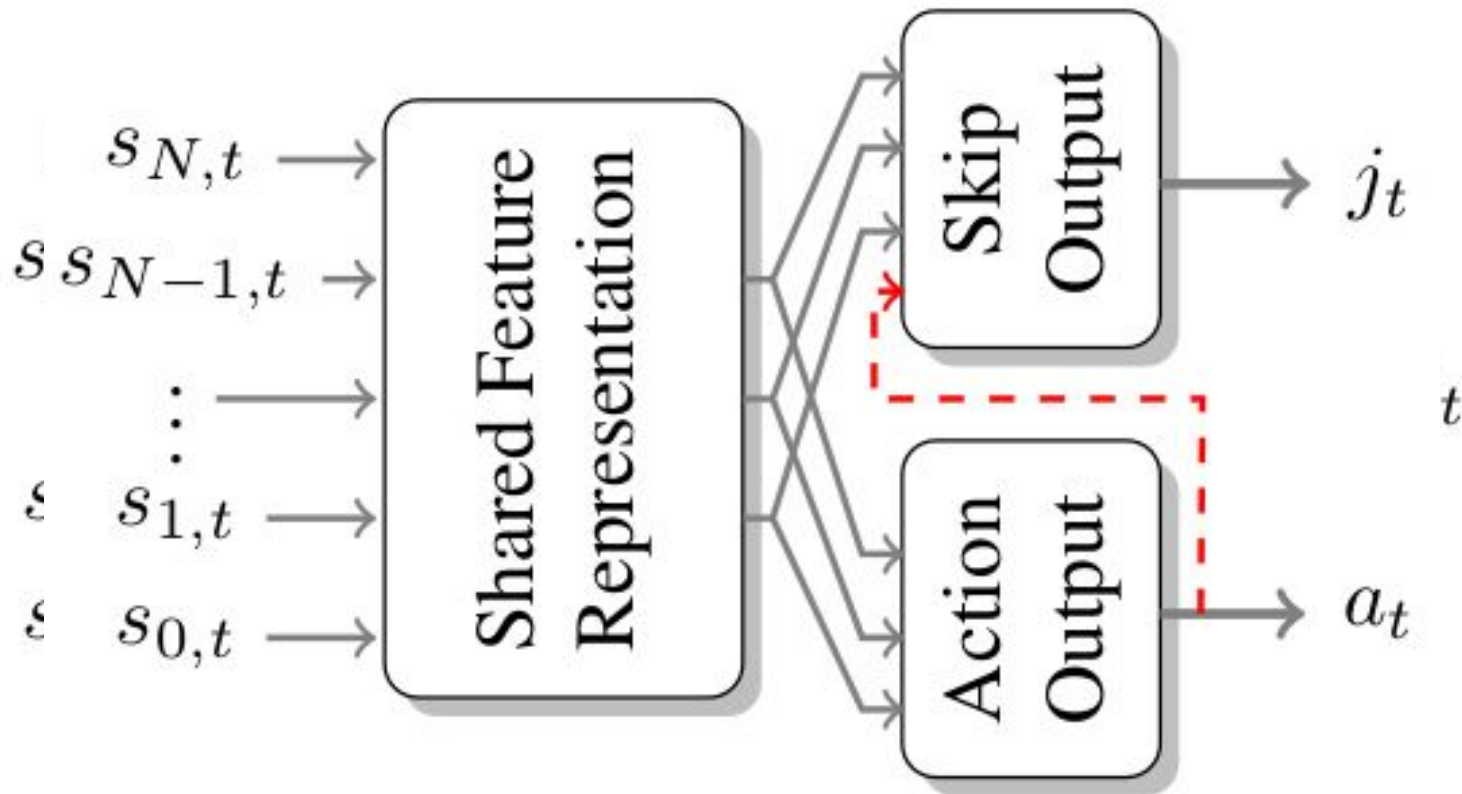
- TempoRL learns well performing policies faster

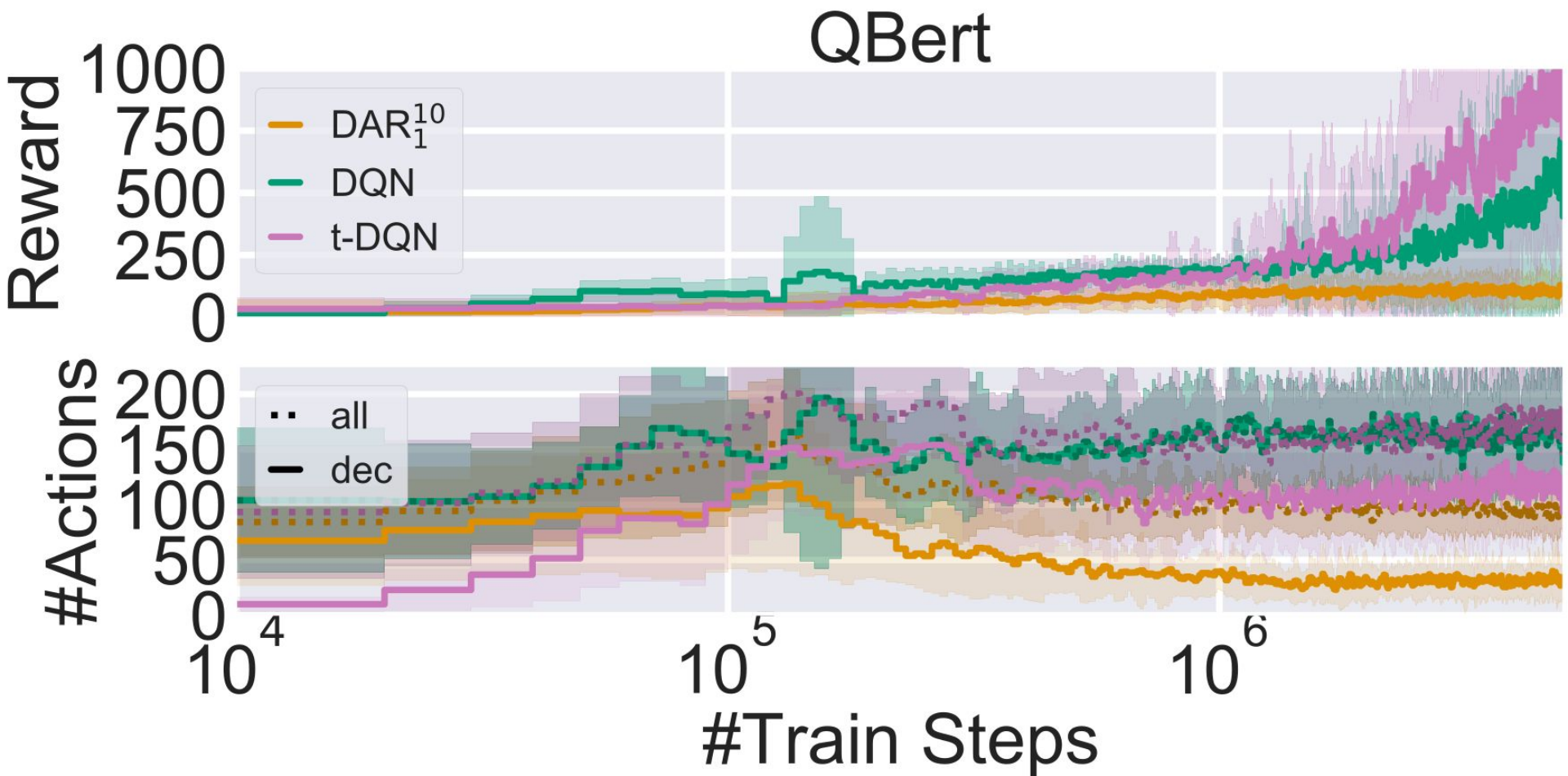
- Comparison of vanilla and TempoRL Q-learning on the example gridworld



- TempoRL learns well performing policies faster requiring far fewer decisions by learning *when* to switch actions

Depending on the state modality we consider different architectures





- TempoRL allows for
 - better exploration
 - faster learning
 - better explainability

Code, learned policies, videos of rollouts and learning curves are available at



- Further results in the paper
 - TempoRL DDPG
 - Influence of TempoRL hyperparameters
 - Improved exploration through TempoRL
- Future Work
 - distributional TempoRL
 - changing TempoRL exploration

Looking forward to meeting you at the poster!