

Slide 1

# Implicit Imitation in Multiagent Reinforcement Learning

Bob Price and Craig Boutilier  
ICML-99

Slides: Dana Dahlstrom  
CSE 254, UCSD  
2002.04.23

Slide 2

## Overview

- Learning by imitation entails watching a mentor perform a task.
- The approach here combines direct experience with an environment model extracted from observations of a mentor.
- This approach shows improved performance and convergence compared to a non-imitative reinforcement learning agent.

Slide 3

## Background

- Other multi-agent learning schemes include:
  - explicit teaching (demonstration)
  - sharing of privileged information
  - elaborate psychological imitation theory
- All these require explicit communication, and usually voluntary cooperation by the mentor.
- A common thread: the observer explores, guided by the mentor.

Slide 4

## Implicit Imitation

In *implicit* imitation, the learner observes the mentor's state transitions but not its actions.

- No demands are made of the mentor beyond ordinary behavior.
  - no voluntary cooperation
  - no explicit communication
- The learner can take advantage of multiple mentors.
- The learner is not forced to follow in the mentor's footsteps.
  - can learn from negative examples without paying a penalty

Slide 5

## Markov Decision Processes

A preliminary assumption: the learner and mentor(s) act concurrently in a single environment, but their actions are noninteracting.

Therefore the underlying *multi-agent Markov decision process (MMDP)* can be factored into separate single-agent MDPs  $\langle S, A, \text{Pr}, R \rangle$ .

- $S$  is the set of states.
- $A$  is the set of actions.
- $\text{Pr}(t|s, a)$  is the probability of transitioning to state  $t$  when performing action  $a$  in state  $s$ .
- $R(s, a, t)$  is the reward received when action  $a$  is performed in state  $s$  and there is a transition to state  $t$ .

Slide 6

## Further Assumptions

- The learner and mentor have identical state spaces:  $S = S_m$
- All the mentor's actions are available to the learner:  $A \supseteq A_m$
- The mentor's transition probabilities apply to the learner: for all states  $s$  and  $t$ , if  $a \in A_m$  then  $\text{Pr}(t|s, a) = \text{Pr}_m(t|s, a)$ .
- The learner knows its own reward function  $R(s, a, t) = R(s)$ .
- The learner can observe the mentor's state transitions  $\langle s, t \rangle$ .
- The horizon is infinite with discount factor  $\gamma$ .

Slide 7

## The Reinforcement Learning Task

The task is to find a policy  $\pi : S \rightarrow A$  that maximizes the total discounted reward. Under such an optimal policy  $\pi^*$ , the total discounted reward  $V^*(s)$  at state  $s$  is given by the Bellman equation:

$$V^*(s) = R(s) + \gamma \max_{a \in A} \left\{ \sum_{t \in S} \Pr(t|s, a) V^*(t) \right\} \quad (1)$$

- Given samples  $\langle s, a, t \rangle$  the agent could
  - estimate an action-value function directly via Q-learning, or
  - estimate  $\Pr$  and solve for  $V^*$  in Equation 1.
- Prioritized sweeping converges on a solution to the Bellman equation as its estimate of  $\Pr$  improves.

Slide 8

## Estimating the Transition Probabilities

The transition probabilities can be estimated by observed frequencies

$$\widehat{\Pr}(t|s, a) = \frac{\text{count}(\langle s, a, t \rangle)}{\sum_{t' \in S} \text{count}(\langle s, a, t' \rangle)}$$

For all states  $t$ , as the number of times the learner has performed action  $a$  in state  $s$  approaches infinity, the estimate  $\widehat{\Pr}(t|s, a)$  converges to the actual probability  $\Pr_m(t|s, a)$ .

Slide 9

## Estimating the Mentor's Transition Probabilities

Assuming the mentor uses a stationary, deterministic policy  $\pi_m$ ,

$$\Pr_m(t|s) = \Pr_m(t|s, \pi_m(s))$$

In this case the mentor's transition probabilities too can be estimated by observed frequencies

$$\widehat{\Pr}_m(t|s) = \frac{\text{count}_m(\langle s, t \rangle)}{\sum_{t' \in S} \text{count}_m(\langle s, t' \rangle)}$$

For all states  $t$ , as the mentor's visits to state  $s$  approach infinity, the estimate  $\widehat{\Pr}_m(t|s)$  converges to the actual probability  $\Pr_m(t|s)$ .

Slide 10

## Augmenting the Bellman Equation

**Lemma:** The imitation learner's state-value function is specified by the *augmented* Bellman equation

$$V^*(s) = R(s) + \gamma \max \left\{ \sum_{t \in S} \Pr_m(t|s) V^*(t), \max_{a \in A} \left\{ \sum_{t \in S} \Pr(t|s, a) V^*(t) \right\} \right\} \quad (2)$$

**Proof idea:** Since  $\Pr_m(t|s) = \Pr(t|s, \pi_m(s))$ , the **first summation** is equal to the second when  $a = \pi_m(s)$ . We know  $\pi_m(s) \in A$  because  $\pi_m(s) \in A_m$  and  $A_m \subseteq A$ ; therefore the **first summation** is redundant and Equation 2 simplifies to Equation 1.

Extension to multiple mentors is straightforward.

Slide 11

## Augmented Bellman Backups

Bellman backups update state-value estimations. The augmented Bellman equation suggests the update rule

$$\widehat{V}(s) \leftarrow (1 - \alpha)\widehat{V}(s) + \alpha R(s) + \alpha\gamma \max \left\{ \sum_{t \in S} \widehat{\Pr}_m(t|s)\widehat{V}(t), \max_{a \in A} \left\{ \sum_{t \in S} \widehat{\Pr}(t|s, a)\widehat{V}(t) \right\} \right\}$$

where  $\alpha$  is the learning rate.

Slide 12

## Confidence Estimation

The learner must rely on estimates  $\widehat{\Pr}(t|s, a)$  and  $\widehat{\Pr}_m(t|s)$ . It is best to account for the unreliability of these estimates.

- $\Pr(t|s, a)$  and  $\Pr_m(t|s)$  are multinomial distributions; assume Dirichlet priors over them.
- Compute the learner's value function  $V(s)$  and the mentor's value function  $V_m(s)$  within suitable confidence intervals; let  $v^-$  and  $v_m^-$  be the lower bounds of these intervals.
- If  $v_m^- < v^-$ , then ignore mentor observations; either the mentor's policy is suboptimal or confidence in  $\widehat{\Pr}_m$  is too low.

Slide 13

## Accommodating Action Costs

When the reward function  $R(s, a)$  depends on the action, how can it be applied to mentor observations without knowing the mentor's action?

Let  $\kappa(s)$  denote an action whose transition distribution at state  $s$  has minimum Kullback-Leibler (KL) distance from  $\Pr_m(t|s)$ :

$$\kappa(s) = \operatorname{argmin}_{a \in A} \left\{ - \sum_{t \in S} \Pr(t|s, a) \log \Pr_m(t|s) \right\} \quad (3)$$

Using the guessed mentor action  $\kappa(s)$ , the augmented Bellman equation can be rewritten as

$$V^*(s) = \max \left\{ \begin{array}{l} R(s, \kappa(s)) + \gamma \sum_{t \in S} \Pr_m(t|s) V^*(t), \\ R(s, a) + \gamma \max_{a \in A} \left\{ \sum_{t \in S} \Pr(t|s, a) V^*(t) \right\} \end{array} \right\}$$

Slide 14

## Prioritized Sweeping

In prioritized sweeping (Moore & Atkeson, 1993)  $N$  backups are performed per transition.

- Maintain a queue of states whose value would change upon backup, prioritized by the magnitude of change.
- At each transition  $\langle s, t \rangle$ :
  1. If a backup would change its value more than a threshold amount  $\theta$ , insert  $s$  into the queue.
  2. Do backups for the top  $N$  states in the queue, inserting their graphwise predecessors (or updating their priorities) if backups would change their values more than  $\theta$ .

Slide 15

## Implicit Imitation in Prioritized Sweeping

To incorporate implicit imitation into prioritized sweeping:

- do backups for mentor transitions as well as learner transitions
- use augmented Bellman instead of standard Bellman backups
- ignore the mentor-derived model when confidence in it is too low

Slide 16

## Implicit Imitation in Q-Learning

Model extraction can be incorporated into algorithms other than prioritized sweeping, such as Q-learning.

- Augment the action space with a placeholder action  $a_m \in A$ .
- For each transition  $\langle s, t \rangle$  use the update rule:

$$Q(s, a) \leftarrow (1 - \alpha)Q(s, a) + \alpha \left( R(t) + \gamma \max_{a' \in A} Q(t, a') \right)$$

where  $a = a_m$  for observed mentor transitions, and  $a$  is the action performed by the learner otherwise.

Slide 17

## Action Selection

An  $\varepsilon$ -greedy action selection policy ensures exploration:

- with probability  $\varepsilon$ , pick an action uniformly at random
- with probability  $1 - \varepsilon$ , pick the greedy action

The “greedy action” is here defined as the  $a$  whose estimated distribution  $\widehat{\Pr}(t|s, a)$  has minimum KL distance from  $\widehat{\Pr}_m(t|s)$ .

Slide 18

## Experimental Setup

To evaluate their technique, the authors simulated three different agents:

- an expert mentor following an  $\varepsilon$ -greedy policy with  $\varepsilon \in \Theta(0.01) \odot$
- an imitative prioritized sweeping learner observing the mentor
- a non-imitative prioritized sweeping learner

They compare the imitation learner’s performance to that of the non-imitation learner, as a control.

- The learners use the same parameters, including a fixed number of backups per sample.
- The learners’  $\varepsilon$  decays over time.

Slide 19

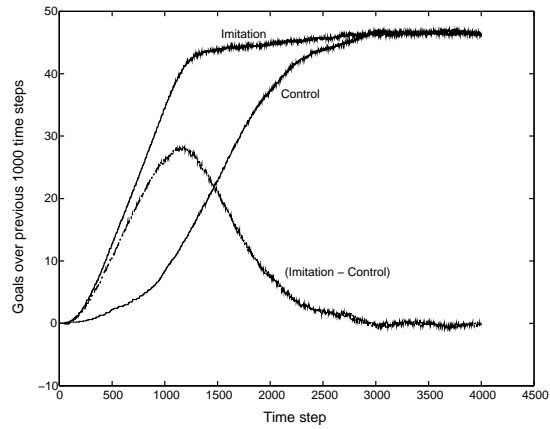


Figure 1: Performance in a  $10 \times 10$  grid world with 10% noisy actions.

Slide 20

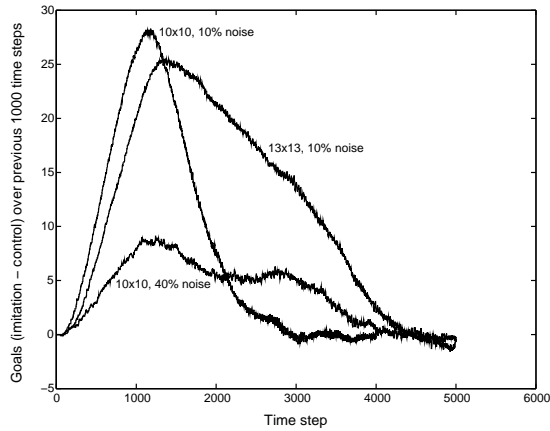


Figure 2: Imitation vs. control for different grid-world parameters.

Slide 21

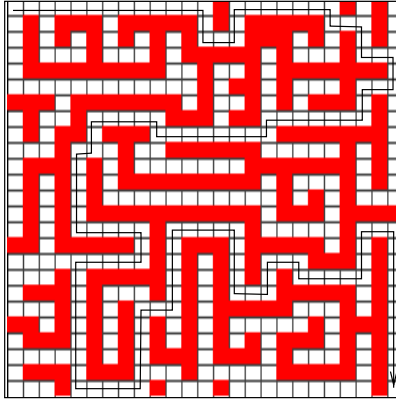


Figure 5: A “complex maze” grid world.

Slide 22

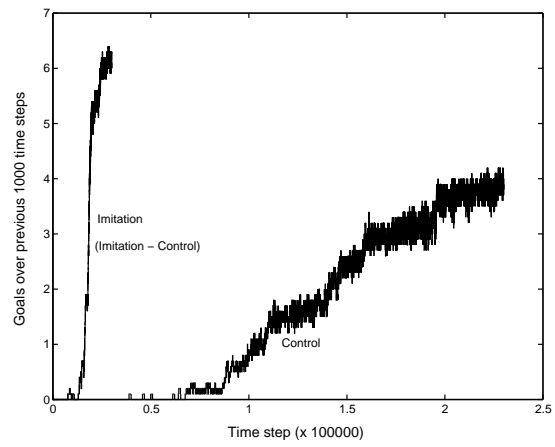


Figure 6: Performance in the grid world of Figure 5.

Slide 23

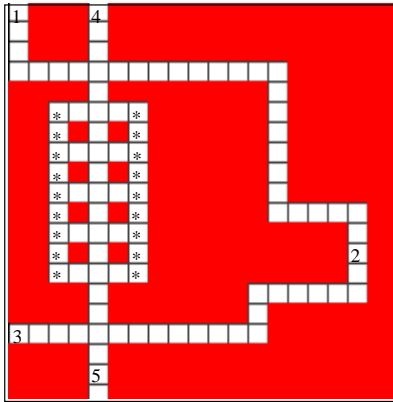


Figure 7: A “perilous shortcut” grid world.

Slide 24

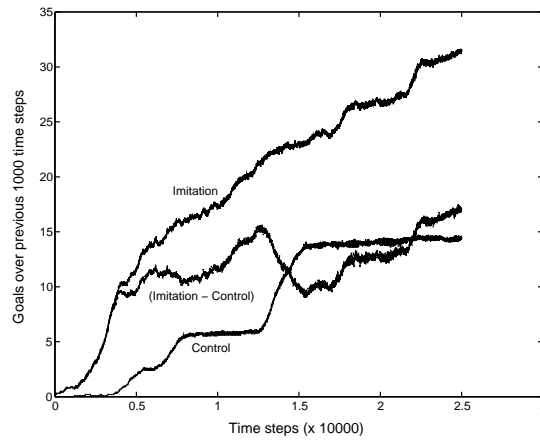


Figure 8: Performance in the grid world of Figure 7.

Slide 25

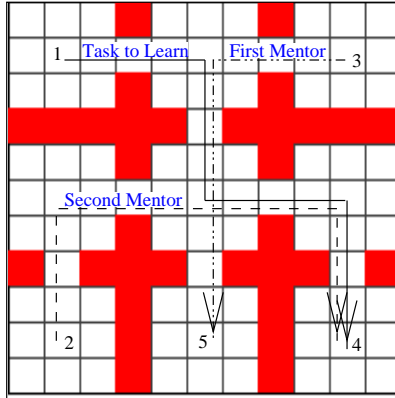


Figure 9: A grid world with multiple mentors whose trajectories are different from, but overlapping with, the learner's target trajectory.

Slide 26

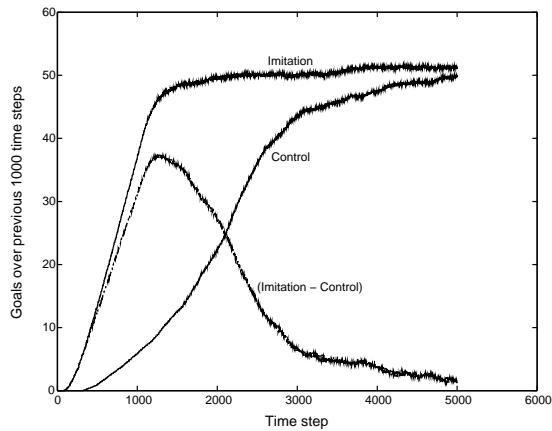


Figure 10: Performance in the grid world of Figure 9.

**Slide 27**

### Summary: Assumptions

- Multiple agents' actions are noninteracting.
- The learner and mentor have "similar" capabilities:
  - Their state spaces are identical.
  - All actions the mentor can take are available to the learner.
  - All the mentor's transition probabilities apply to the learner.
- The learner knows its own reward function.
- The learner can observe the mentor's state transitions.
  - For convergence, the observation period is indefinite.
  - The mentor's policy is stationary.

**Slide 28**

### Summary: Results

Implicit imitation shows:

- improvement over standard learning (given an expert mentor)
- tolerance to noise (Figures 1 and 2)
- the ability to integrate subskills from multiple mentors (Figure 10)
- benefits that increase with problem difficulty (Figures 5 and 6)