Online Discovery of Feature Dependencies

Alborz Geramifard, Finale Doshi, Joshua Redding, Nicholas Roy, Jonathan P. How

ICML 2011

Review notes by Xuejun Liao

October 3, 2011

Markov Decision Processes

Definition

A Markov Decision Process (MDP) (Sutton & Barto, 1998) is a tuple defined by $(\mathcal{S}, \mathcal{A}, P^a_{ss'}, R^a_{ss'}, \gamma)$ where $\mathcal{S}$ is a finite set of states, $\mathcal{A}$ is a set of actions, $P^a_{ss'}$ is the probability of getting to state s' by taking action a in state s, $R^a_{ss'}$ is the corresponding reward, and $\gamma \in [0, 1)$ is a discount factor balancing current and future rewards.

Value Function

- Given policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$, the action-value function is

$$Q^{\pi}(s,a) = \mathbb{E}_{\pi}\left[\sum_{t=0}^{\infty} \gamma^{t} r_t \mid a_0 = a, s_0 = s\right]$$

- The value function is governed by Bellman equation

$$Q^{\pi}(s,a) = R(s,a) + \gamma \sum_{s'} P^{a}_{ss'} Q^{\pi}(s', \pi(s'))$$

Temporal Difference (TD) Learning

- Given a trajectory $(s_0, a_0, r_0, s_1, a_1, r_1, \cdots, s_t, a_t, r_t, s_{t+1})$, define TD error

$$\delta_t = r_t + \gamma \max_{a_{t+1}} Q^\pi(s_{t+1}, a_{t+1}) - Q^\pi(s_t, a_t)$$

and a TD learning rule updates

$$Q^\pi(s_t, a_t) \leftarrow Q^\pi(s_t, a_t) + \alpha_t \delta_t$$

where $\alpha_t > 0$ is a learning rate.

Linear Function Approximation

- The value function can approximated as

$$Q^{\pi}(s,a) = \sum_i w_i \phi_i(s,a)$$

  where $\phi_i : \mathcal{S} \times \mathcal{A} \rightarrow \{0,1\}$ maps states and actions to features (binary as considered here).
- Omiting action a for simplicity, the set $\{i : \phi_i(s) = 1\}$ contains the features (indices) that are active for state s.

Basic Idea of Online Feature Discovery

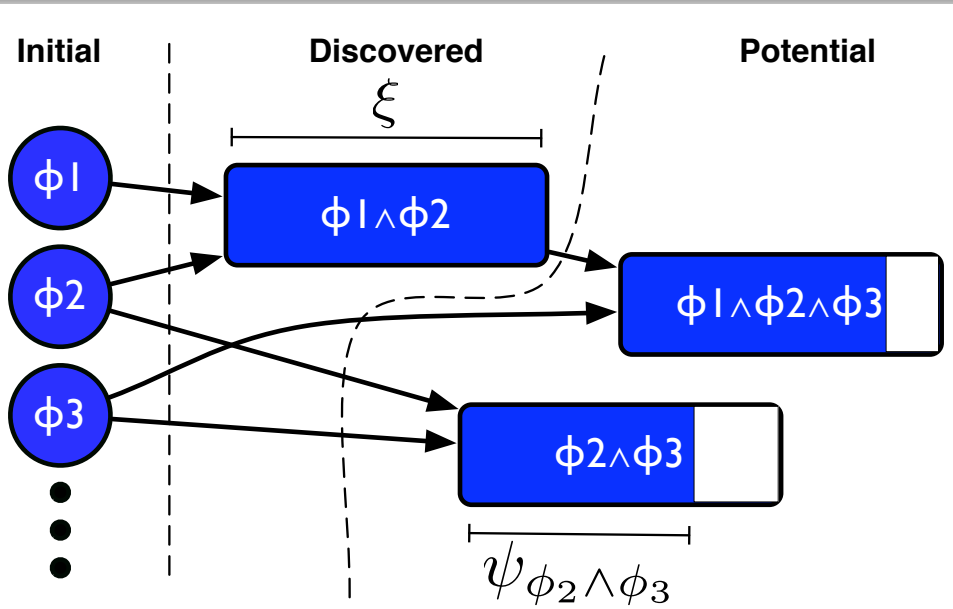


Figure 1. A snapshot of iFDD: Initial features are circles, conjunctive features are rectangles. The *relevance* ψ_f of a potential feature f is the filled part of the rectangle. Potential features are discovered if their relevance ψ reaches the discovery threshold ξ.

- Start from an initial set of binary features.
- Expand features by adding conjunctions of existing features where large errors persist.
- For any state, prune away the active features that are contained in larger conjunction sets.

Discovering New Features

Algorithm 1: Discover

Input: $\phi(s), \delta_t, \xi, \mathbf{F}, \psi$

Output: $\mathbf{F}, \psi$

foreach $(g, h) \in \{(i, j) | \phi_i(s)\phi_j(s) = 1\}$ **do**
  $f \leftarrow g \wedge h$
  if $f \notin \mathbf{F}$ **then**
    $\psi_f \leftarrow \psi_f + |\delta_t|$
    if $\psi_f > \xi$ **then**
      $\mathbf{F} \leftarrow \mathbf{F} \cup f$

Pruning Active Features

Algorithm 2: Generate Feature Vector (ϕ)

Input: $\phi^0(s)$, $\mathbf{F}$
Output: $\phi(s)$

$\phi(s) \leftarrow \bar{0}$
$activeInitialFeatures \leftarrow \{i | \phi_i^0(s) = 1\}$
$Candidates \leftarrow \wp(activeInitialFeatures)$ *sorted
while $activeInitialFeatures \neq \emptyset$ **do**
    $f \leftarrow Candidates$.next()
    if $f \in \mathbf{F}$ **then**
        $activeInitialFeatures \leftarrow activeInitialFeatures - f$
        $\phi_f(s) \leftarrow 1$
return $\phi(s)$

Properties and Performance Guarantees (1/2)

Theorem

Given initial features and a fixed policy that turns the underlying MDP into an ergodic Markov chain, iFDD-TD is guaranteed to discover all possible feature conjunctions or converge to a point where the TD error is identically zero with probability one.

Corollary

If at each step of iFDD-TD the policy changes but still induces an ergodic Markov chain (e.g., via-greedy or Boltzmann exploration), then iFDD-TD will explore all reachable features or converge to a point where the TD error is identically zero with probability one.

Properties and Performance Guarantees (2/2)

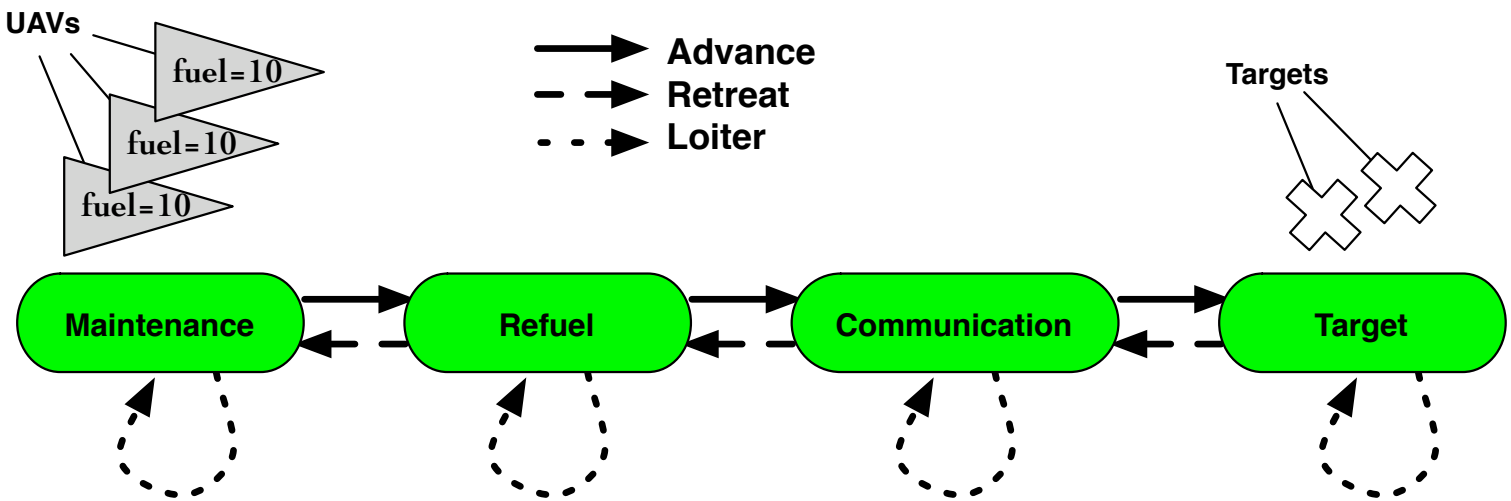
Corollary 3.3 *With probability one, iFDD-TD converges to a weight vector θ and feature matrix $\mathbf{\Phi}$, where the approximated value function error, as originally shown by Tsitsiklis and Van Roy (1997) for a fixed set of linear bases, is bounded by:*

$$||\mathbf{\Phi}\theta - \mathbf{V}^*||_{\mathbf{D}} \leq \frac{1}{1-\gamma}||\mathbf{\Pi}\mathbf{V}^* - \mathbf{V}^*||_{\mathbf{D}},$$

where $\mathbf{D}_{|\mathcal{S}|\times|\mathcal{S}|}$ is a diagonal matrix with the stationary distribution along its diagonal, $\mathbf{\Pi} = \mathbf{\Phi}_\infty(\mathbf{\Phi}_\infty{}^T\mathbf{D}\mathbf{\Phi}_\infty)^{-1}\mathbf{\Phi}_\infty{}^T\mathbf{D}$, and $||.||$ stands for the weighted Euclidean norm.

Unmanned Aerial Vehicle (UAV) Application: Persistent Surveillance

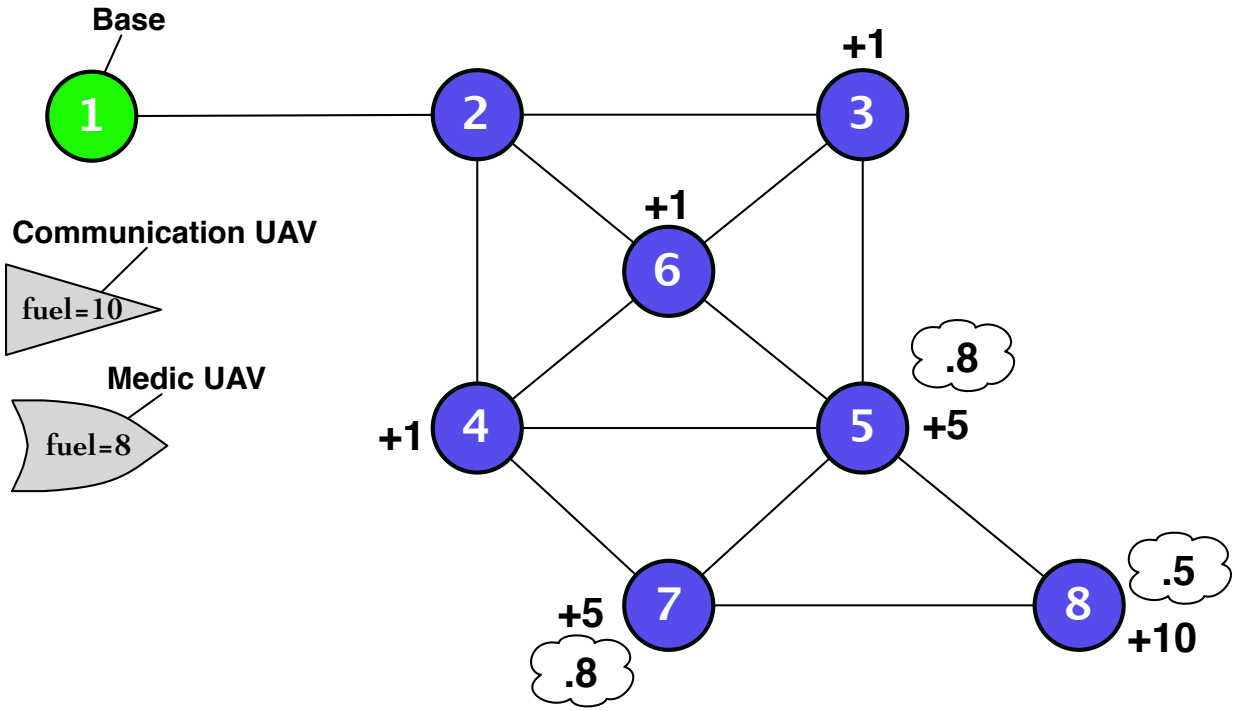
- The state is a 12-dimensional vector of remaining fuel, location, motor sensor status and camera sensor status for each of the three UAVs for a total of approximately **150 million** state-action pairs.



(a) Persistent Surveillance

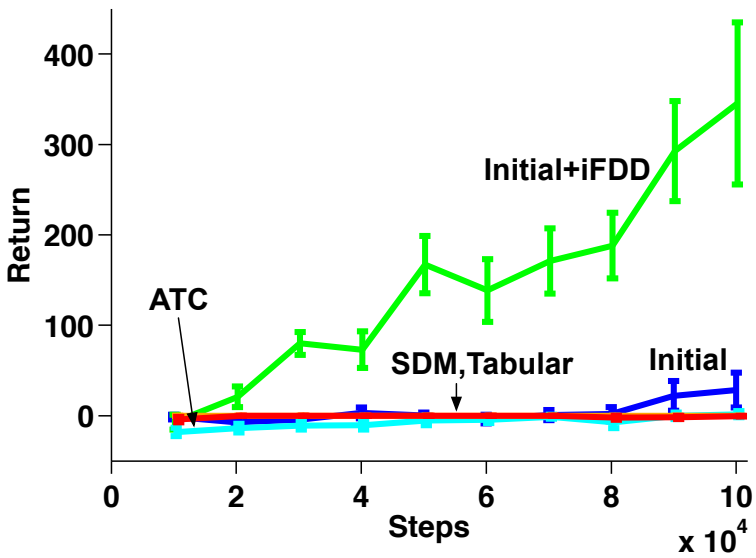
Unmanned Aerial Vehicle (UAV) Application: Rescue Mission

- Initial features were the fuel and position of each UAV, the communication UAV mode (move or perch), and the rescue status at each node. The total state-action pairs exceeded **200 million**.

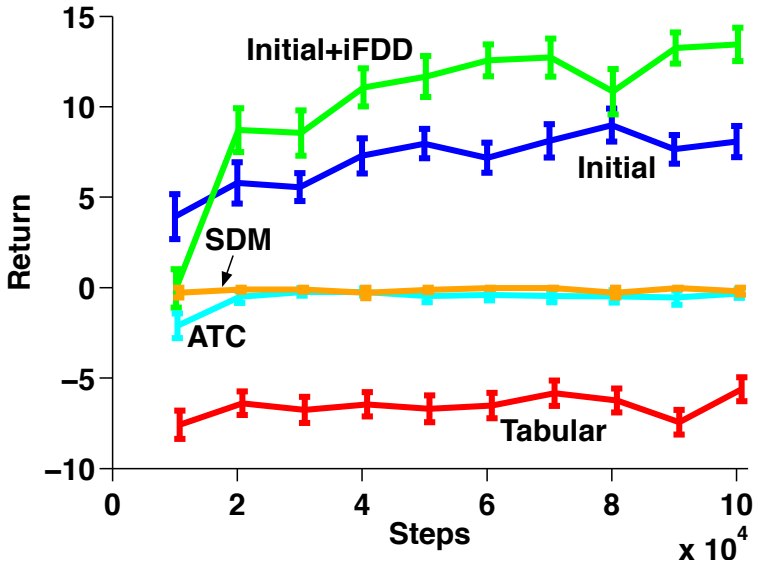


(b) Rescue Mission

Results (1/3): performance



(c) Persistent Surveillance



(d) Rescue Mission

Results (2/3): feature counts

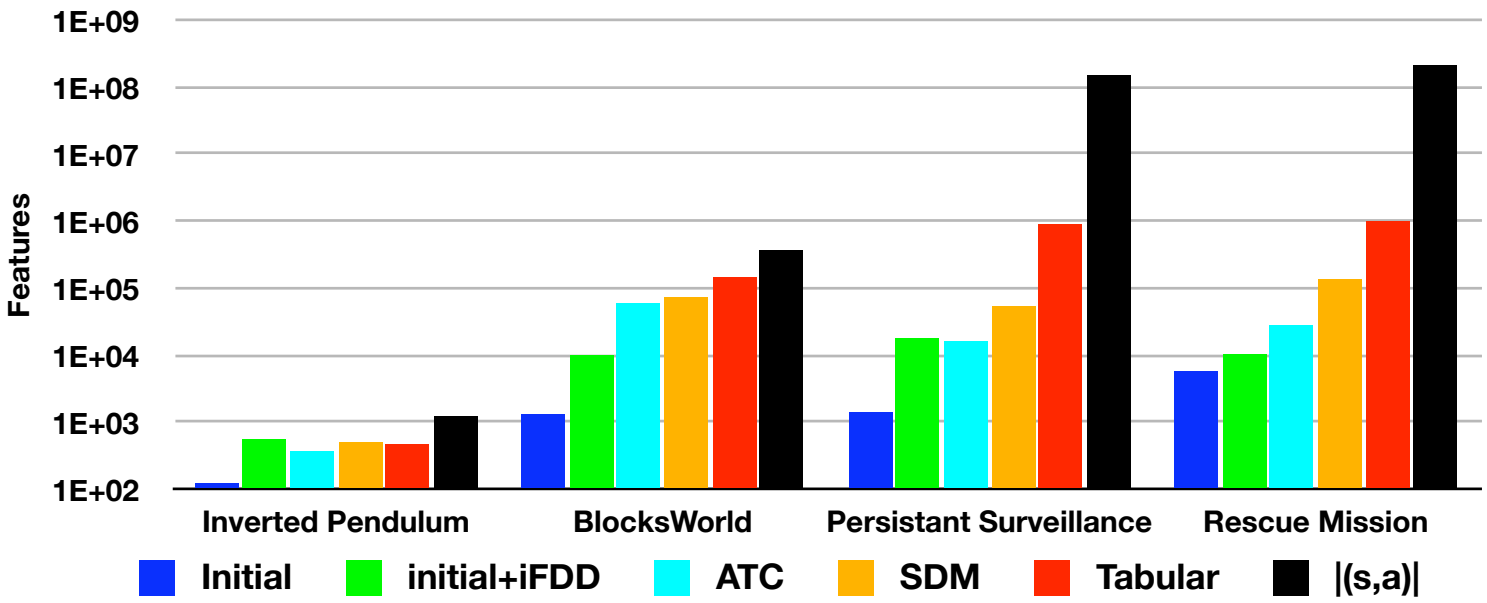


Figure 4. Average final feature counts. ATC and SDM, even using more features, performed poorly on high-dimensional examples. The black bar depicts the total number of state-action pairs.

Results (3/3): comparison to random conjunction features

Table 1. The final performance with 95% confidence intervals of iFDD and random expansion with equal number of features.

	Expansion Scheme	
Domain	Random	iFDD
Inverted Pendulum	2953 ± 30	$\mathbf{3000 \pm 0}$
BlocksWorld	-0.80 ± 0.06	$\mathbf{-0.24 \pm 0.10}$
Persistent Surveillance	174 ± 44	$\mathbf{280 \pm 49}$
Rescue Mission	$10 \pm .74$	$\mathbf{12 \pm .75}$