Planning and Optimization

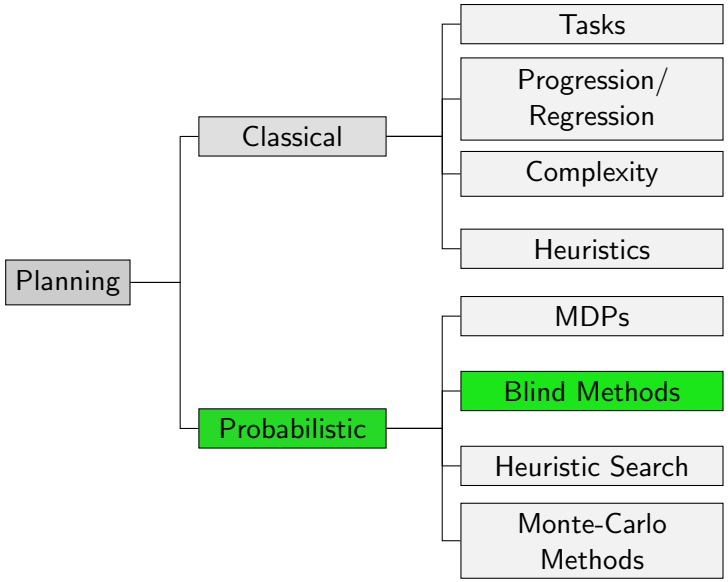
F4. Blind Methods: Value Iteration & Linear Programming

Gabriele Röger and Thomas Keller

Universität Basel

November 26, 2018

Content of this Course



From Policy Iteration to Value Iteration

- Policy Iteration:
  - search over policies
  - by evaluating their state-values
- Value Iteration:
  - search directly over state-values
  - optimal policy induced by final state-values

F4.1 Value Iteration

Value Iteration: Idea

- Value Iteration (VI) was first proposed by Bellman in 1957
- computes estimates $\hat{V}^0, \hat{V}^1, \ldots$ of $V_\star$ in an iterative process
- starts with arbitrary $\hat{V}^0$
- bases estimate $\hat{V}^{i+1}$ on values of estimate $\hat{V}^i$ by applying Bellman optimality equation on all states:

$$\hat{V}^{i+1}(s) := \min_{\ell \in L(s)} c(\ell) + \sum_{s' \in S} T(s, \ell, s') \cdot \hat{V}^i(s')$$

(for SSPs; for FH-MDPs and DR-MDPs accordingly)

- converges to state-values of optimal policy
- terminates when difference of estimates is small

Example: Value Iteration

5	3.85	4.83	1.32	$s_\star$ 4.1
4	2.51	3.43	★ 0.12	0.68
3	1.92	1.13	2.92	0.87
2	3.95	0.47	3.58	3.64
1	s_0 1.15	3.94	0.13	4.06
	1	2	3	4

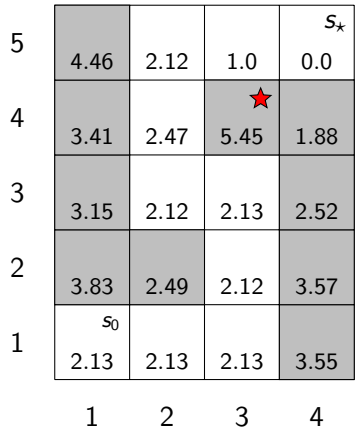
$\hat{V}^0$

Example: Value Iteration

5	4.24	2.32	1.12	$s_\star$ 0.0
4	3.27	1.12	★ 3.34	1.46
3	2.60	1.47	1.12	1.79
2	3.56	1.73	1.13	3.53
1	s_0 2.15	1.13	1.13	3.49
	1	2	3	4

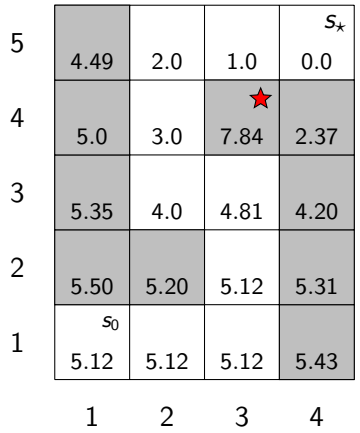
$\hat{V}^1$

Example: Value Iteration



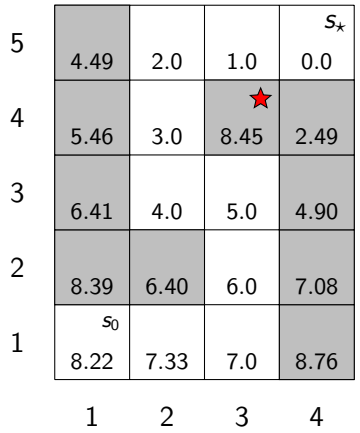
$\hat{V}^2$

Example: Value Iteration



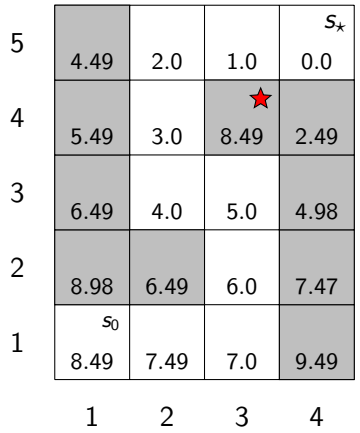
$\hat{V}^5$

Example: Value Iteration



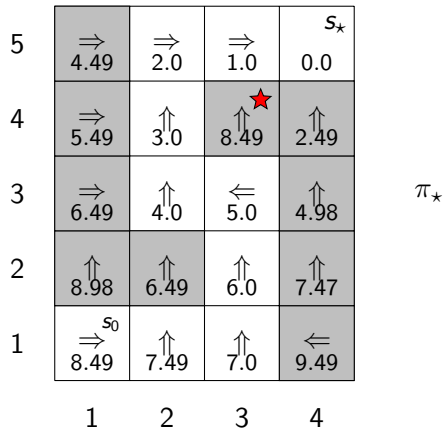
$\hat{V}^{10}$

Example: Value Iteration



$\hat{V}^{19}$

Example: Value Iteration



$\pi_\star$

Value Iteration

Value Iteration for SSP $\mathcal{T}$ and $\epsilon > 0$

initialize $\hat{V}^0$ arbitarily
for $i = 1, 2, \ldots$:
  for all states $s \in S$:
    $\hat{V}^{i+1}(s) := \min_{\ell \in L(s)} c(\ell) + \sum_{s' \in S} T(s, \ell, s') \cdot \hat{V}^i(s')$
  if $\max_{s \in S} |\hat{V}^{i+1}(s) - \hat{V}^i(s)| < \epsilon$:
    return $\pi_{\hat{V}^{i+1}}$

Note: VI for FH-MDPs and DR-MDPs obtained by replacing Bellman optimality equation with corresponding version.

Policy Iteration or Value Iteration?

- PI and VI both have their advantages:
  - often, PI requires only few iterations
  - VI iterations significantly cheaper
- Better versions of both PI and VI exist
  - Modified PI (approximate policy evaluation)
  - Asynchronous VI (update subset of states in each iteration)
- However, both suffer from the problem that the whole state space must eventually be visited
- Impossible in large MDPs / SSPs

F4.2 Linear Programming

Linear Programming for SSPs

- VI iteratively computes solution to the set of Bellman optimality equations
- Linear Programming offers an alternative way to solve optimization problems (see E3)
- Get solution to

$$V_\star(s) := \min_{\ell \in L(s)} c(\ell) + \sum_{s' \in S} T(s, \ell, s') \cdot V_\star(s')$$

  without iterative process
- Problem: equations are not linear due to minimization
- But: can be moved to objective function

Linear Programming

The solution to the following LP provides the state-values $V_\star(s)$ (through the variables X_s) of an optimal policy for an SSP $\mathcal{T} = \langle S, L, c, T, s_0, S_\star \rangle$:

$$\text{maximize} \quad \sum_{s \in S} X_s \quad \text{subject to}$$

$$X_s = 0 \quad \text{for all } s \in S_\star$$

$$X_s \leq c(\ell) + \sum_{s' \in S} T(s, \ell, s') \cdot X_{s'} \quad \text{for all } s \in S \text{ and } \ell \in L(s)$$

$$X_s \geq 0 \quad \text{for all } s \in S$$

Note: Versions for FH-MDPs and DR-MDPs exist.

Linear Programming

- Allows to solve SSPs with existing LP solver
- But: $|S|$ many variables, $|S| \cdot |L|$ many constraints
- Interesting problems usually not solvable with LP solvers (but neither with PI or VI)

$\Rightarrow$ For large SSPs and MDPs, we need different techniques.

F4.3 Summary

Summary

- Value Iteration searches in the space of state-values
- VI applies Bellman optimality equation iteratively
- VI converges to optimal state-values
- Alternative to compute state-values is by compilation to LP