

Planning and Optimization

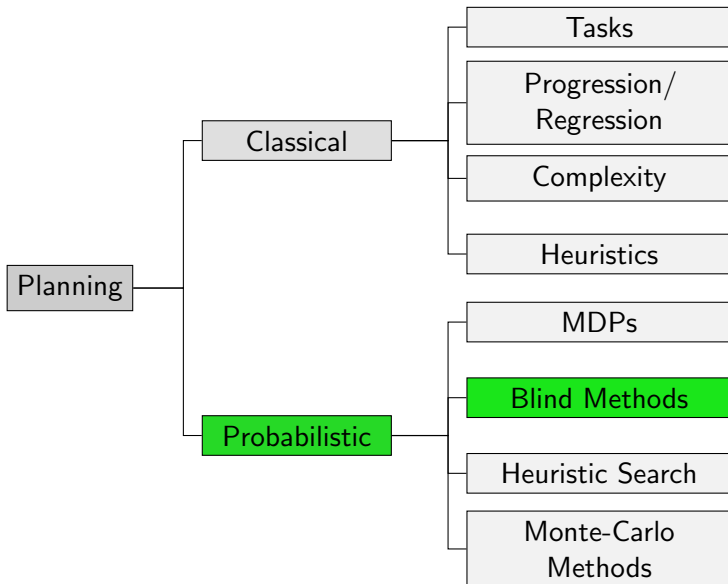
F4. Blind Methods: Value Iteration & Linear Programming

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

Value Iteration

Value Iteration: Idea

- Value Iteration (VI) was first proposed by Bellman in 1957
- computes estimates $\hat{V}^0, \hat{V}^1, \dots$ of V_* 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_*
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	3.94	0.13	4.06
	1	2	3	4

 \hat{V}^0

Example: Value Iteration

5	4.24	2.32	1.12	s_* 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

5	4.46	2.12	1.0	s_* 0.0
4	3.41	2.47	5.45	1.88
3	3.15	2.12	2.13	2.52
2	3.83	2.49	2.12	3.57
1	s_0 2.13	2.13	2.13	3.55
	1	2	3	4


 \hat{V}^2

Example: Value Iteration

5	4.49	2.0	1.0	s_* 0.0
4	5.0	3.0	7.84	2.37
3	5.35	4.0	4.81	4.20
2	5.50	5.20	5.12	5.31
1	s_0 5.12	5.12	5.12	5.43
	1	2	3	4


 \hat{V}^5

Example: Value Iteration

5	4.49	2.0	1.0	s_* 0.0
4	5.46	3.0	8.45 	2.49
3	6.41	4.0	5.0	4.90
2	8.39	6.40	6.0	7.08
1	s_0 8.22	7.33	7.0	8.76
	1	2	3	4


\hat{V}^{10}

Example: Value Iteration

5	4.49	2.0	1.0	s_* 0.0
4	5.49	3.0	8.49 	2.49
3	6.49	4.0	5.0	4.98
2	8.98	6.49	6.0	7.47
1	s_0 8.49	7.49	7.0	9.49
	1	2	3	4

\hat{V}^{19}

Example: Value Iteration

5	\Rightarrow 4.49	\Rightarrow 2.0	\Rightarrow 1.0	s_* 0.0
4	\Rightarrow 5.49	\Uparrow 3.0	\Uparrow 8.49 	\Uparrow 2.49
3	\Rightarrow 6.49	\Uparrow 4.0	\Leftarrow 5.0	\Uparrow 4.98
2	\Uparrow 8.98	\Uparrow 6.49	\Uparrow 6.0	\Uparrow 7.47
1	\Rightarrow 8.49 s_0	\Uparrow 7.49	\Uparrow 7.0	\Leftarrow 9.49
	1	2	3	4

π_*

Value Iteration

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

initialize \hat{V}^0 arbitrarily

for $i = 1, 2, \dots$:

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

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 $V_*(s)$ (through the variables X_s) of an **optimal policy** for an SSP $\mathcal{T} = \langle S, L, c, T, s_0, S_* \rangle$:

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

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

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

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

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)

⇒ For large SSPs and MDPs, we need different techniques.

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**