POMDP Planning and Execution in an Augmented Space

Marek Grześ and Pascal Poupart

UNIVERSITY OF WATERLOO

International Conference on Autonomous Agents and Multiagent Systems

Paris, May 5–9, 2014

POMDP

Partially Observable Markov Decision Process

- a discrete time, dynamical system with controls (actions)
- a policy of action optimises a utility function
- the state of the system is partially observable through noisy sensors

Motivation for Investigating Upper Bounds

- Most point-based value iteration as well as branch-and-abound algorithms (including online planning) guide their optimisation by upper bounds
- There is growing interest in performance guarantees to estimate how far from optimal a policy can be; helps to check if a model fits a particular application
- An upper bound policy can be good and methods of fast execution are desirable
- Upper bounds are hard to improve; better understanding and methods are required

POMDP—Formally

- $\langle S, A, O, T, Z, R, b_0, \gamma \rangle$
- $T_a = \begin{array}{c c} & \begin{array}{ccc} s'_1 & \dots & s'_n \end{array} \\ \begin{array}{c} s_1 \\ \dots \\ s_n \end{array} & \begin{pmatrix} P(s'_1|a, s_1) & \dots & P(s'_n|a, s_1) \\ \dots & \dots & \dots \\ P(s'_1|a, s_n) & \dots & P(s'_n|a, s_n) \end{pmatrix} \end{array}$
- $Z_a = \begin{array}{c c} & \begin{array}{ccc} o_1 & \dots & o_k \end{array} \\ \begin{array}{c} s'_1 \\ \dots \\ s'_n \end{array} & \begin{pmatrix} P(o_1|s'_1, a) & \dots & P(o_k|s'_1, a) \\ \dots & \dots & \dots \\ P(o_1|s'_n, a) & \dots & P(o_k|s'_n, a) \end{pmatrix} \end{array}$

$T_{a,o}$ Matrices

- $T_a = \begin{array}{c} \\ s_1 \\ \dots \\ s_n \end{array} \begin{array}{c} \begin{array}{ccc} s'_1 & \dots & s'_n \end{array} \\ \begin{pmatrix} P(s'_1|a, s_1) & \dots & P(s'_n|a, s_1) \\ \dots & \dots & \dots \\ P(s'_1|a, s_n) & \dots & P(s'_n|a, s_n) \end{pmatrix} \end{array}$

- $Z_a = \begin{array}{c} \\ s'_1 \\ \dots \\ s'_n \end{array} \begin{array}{c} \begin{array}{ccc} o_1 & \dots & o_k \end{array} \\ \begin{pmatrix} P(o_1|s'_1, a) & \dots & P(o_k|s'_1, a) \\ \dots & \dots & \dots \\ P(o_1|s'_n, a) & \dots & P(o_k|s'_n, a) \end{pmatrix} \end{array}$

- $T_{a,o} = T_a diag(Z_a(o)) =$

$$\begin{array}{c} \\ s_1 \\ \dots \\ s_n \end{array} \begin{array}{c} \begin{array}{ccc} s'_1 & \dots & s'_n \end{array} \\ \begin{pmatrix} P(s'_1, o|a, s_1) & \dots & P(s'_n, o|a, s_1) \\ \dots & \dots & \dots \\ P(s'_1, o|a, s_n) & \dots & P(s'_n, o|a, s_n) \end{pmatrix} \end{array}$$

Upper Bounds for POMDPs

- MDP:
  $Q(s,a) = R_a(s) + \gamma \sum_{s'} T_a(s,s') \max_{a'} Q(s',a') \; \forall s,a$

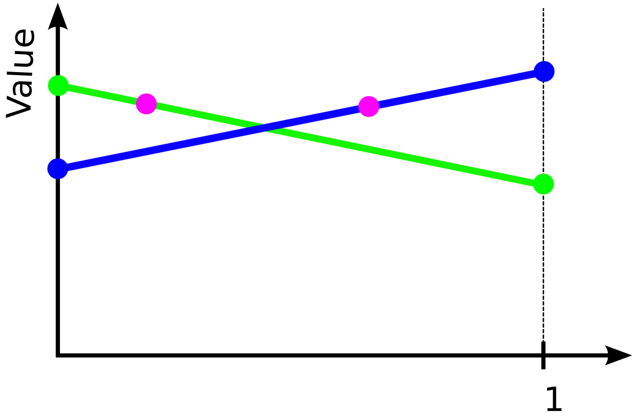
Upper Bounds for POMDPs

- MDP:
  $Q(s,a) = R_a(s) + \gamma \sum_{s'} T_a(s,s') \max_{a'} Q(s',a') \; \forall s,a$
- QMDP:
  $Q(s,a) = R_a(s) + \gamma \sum_{o} \sum_{s'} T_{a,o}(s,s') \max_{a'} Q(s',a') \; \forall s,a$

Upper Bounds for POMDPs

- MDP:
  $Q(s,a) = R_a(s) + \gamma \sum_{s'} T_a(s,s') \max_{a'} Q(s',a') \; \forall s,a$
- QMDP:
  $Q(s,a) = R_a(s) + \gamma \sum_{o} \sum_{s'} T_{a,o}(s,s') \; \max_{a'} \; Q(s',a') \; \forall s,a$
- FIB:
  $Q(s,a) = R_a(s) + \gamma \sum_{o} \max_{a'} \sum_{s'} T_{a,o}(s,s') Q(s',a') \; \forall s,a$

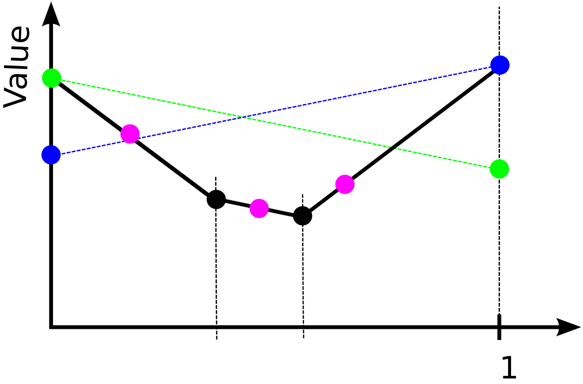
Upper Bounds for Arbitrary Beliefs

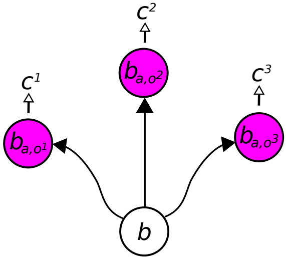


$$\bar{V}(b) = \max_a \sum_s b(s)Q(s,a)$$

$$\pi(b) = \arg\max_a \sum_s b(s)Q(s,a)$$

Upper Bounds with Interior Beliefs





$$\pi^G(b) = \arg\max_a \left\{ \sum_s b(s)R(s,a) + \gamma \sum_{o \in O} P(o|b,a)\bar{V}^G(b_{a,o}) \right\}$$

Augmented POMDPs

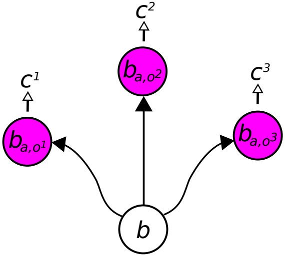
- Add m interior beliefs to the set of n states of the original POMDP
- An initial belief $Pr_0(b) = c(b)$ corresponds to interpolation of b_0 by the convex combination c of anchor beliefs
- $T_{a,o}(b, b') = c(b')O_a(o|b)$

$$T_{a,o} =$$

$$\begin{array}{c} \\ s_1 \\ \dots \\ s_n \\ b_{n+1} \\ \dots \\ b_{n+m} \end{array}
\begin{array}{c}
\begin{array}{cccccc} s'_1 & \dots & s'_n & b'_{n+1} & \dots & b'_{n+m} \end{array} \\
\begin{pmatrix}
c_{1,1} & \dots & c_{1,n} & c_{1,n+1} & \dots & c_{1,n+m} \\
\dots & \dots & \dots & \dots & \dots & \dots \\
c_{n,1} & \dots & c_{n,n} & c_{n,n+1} & \dots & c_{n,n+m} \\
c_{n+1,1} & \dots & c_{n+1,n} & c_{n+1,n+1} & \dots & c_{n+1,n+m} \\
\dots & \dots & \dots & \dots & \dots & \dots \\
c_{n+m,1} & \dots & c_{n+m,n} & c_{n+m,n+1} & \dots & c_{n+m,n+m}
\end{pmatrix}
\end{array}$$

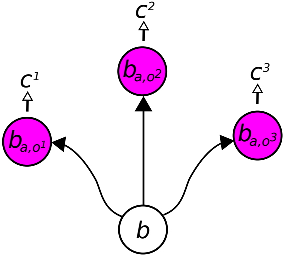
Avoiding Lookahead

- Lookahead

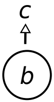


Avoiding Lookahead

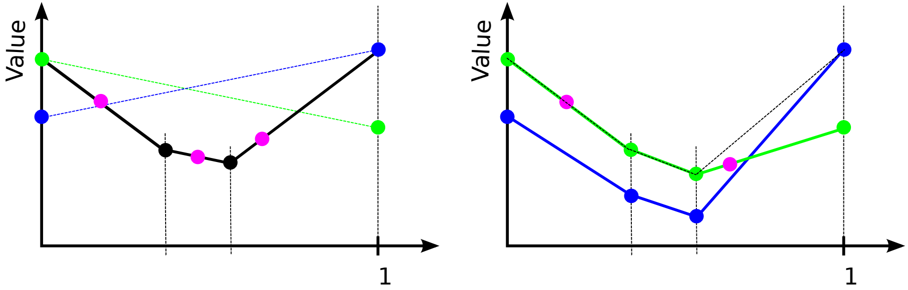
- Lookahead



- Observe that the convex combination of b can be seen as its embedding in the augmented space **(c becomes a belief in the augmented space)**, and the policy can be queried directly



Avoiding Lookahead—ctd



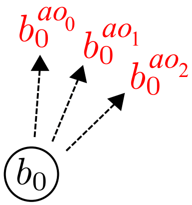
Execution in an Augmented Space

- Recall that c **is a belief in the augmented space**—applies to the initial belief as well
- In the augmented space, $T_{a,o}$ is available, hence **a POMDP can be executed in the augmented space**—executed for the purpose of updating beliefs and querying its policy
- **No need to do interpolations or approximations**
- This process can be efficient even though the number of states grows because $T_{a,o}$ **becomes sparser when more states are added** (in what follows, we refer to deterministic POMDPs to explain why)

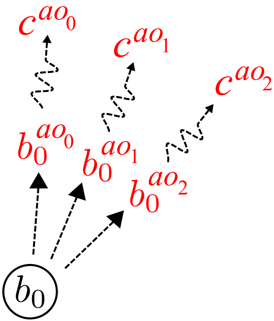
Execution of Upper Bound Policies

b_0

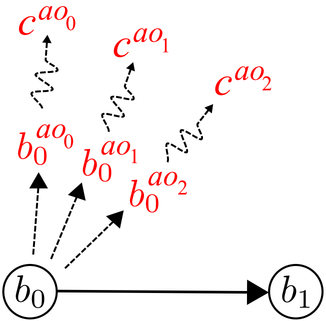
Execution of Upper Bound Policies



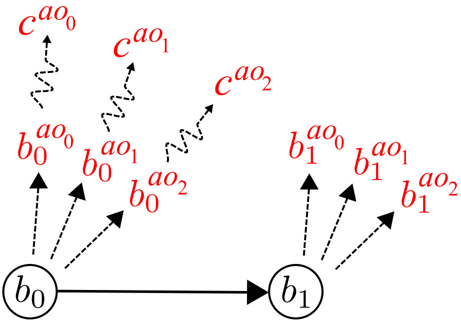
Execution of Upper Bound Policies



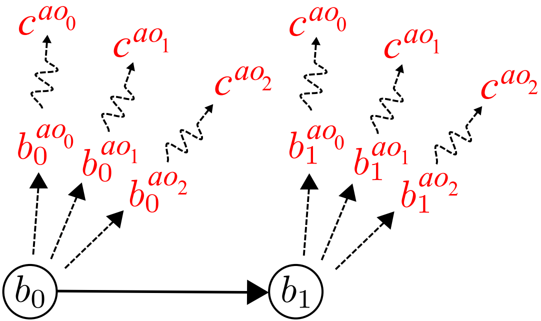
Execution of Upper Bound Policies



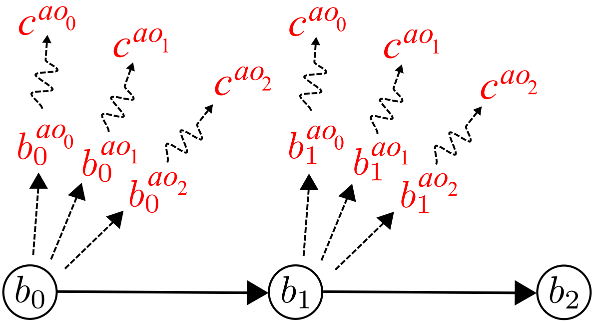
Execution of Upper Bound Policies



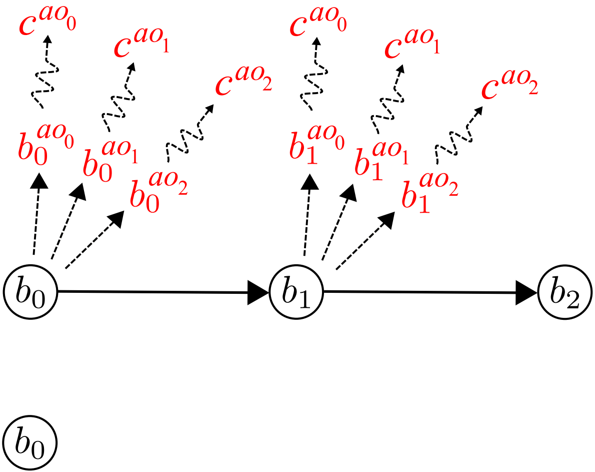
Execution of Upper Bound Policies



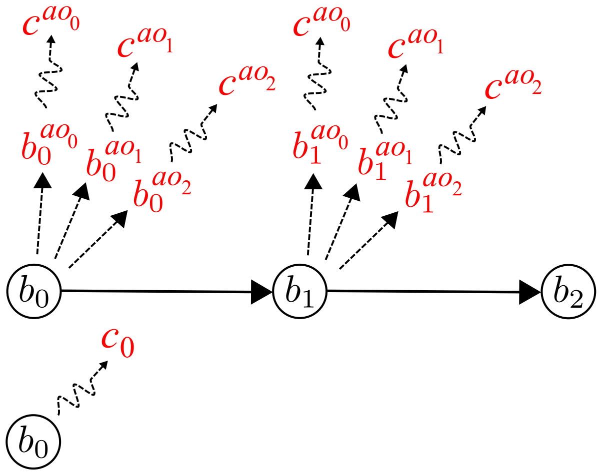
Execution of Upper Bound Policies



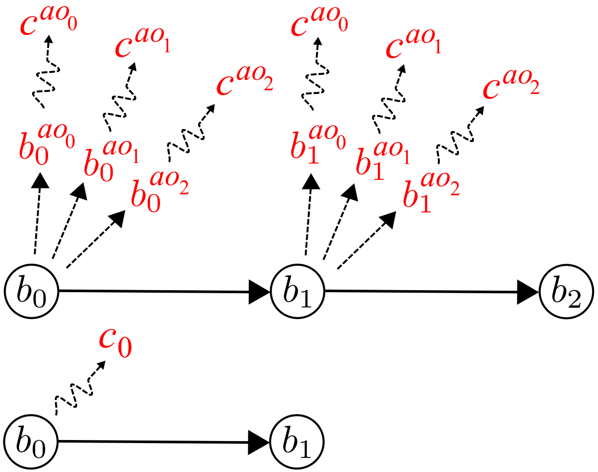
Execution of Upper Bound Policies



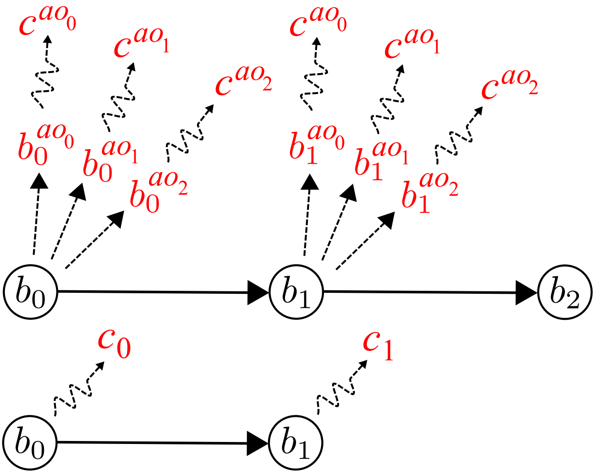
Execution of Upper Bound Policies



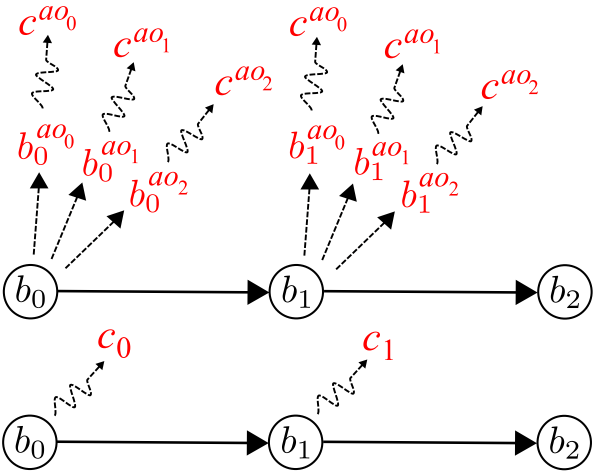
Execution of Upper Bound Policies



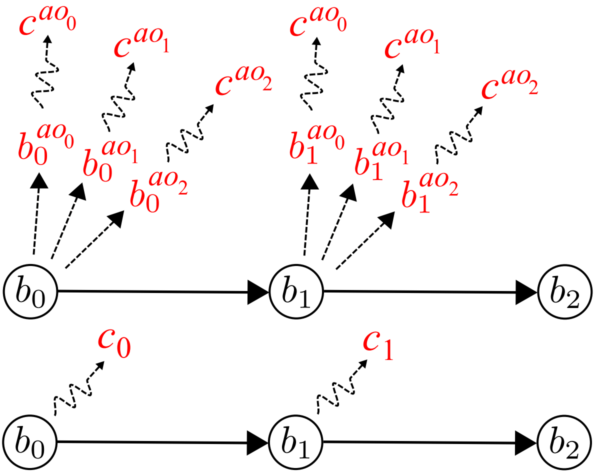
Execution of Upper Bound Policies



Execution of Upper Bound Policies

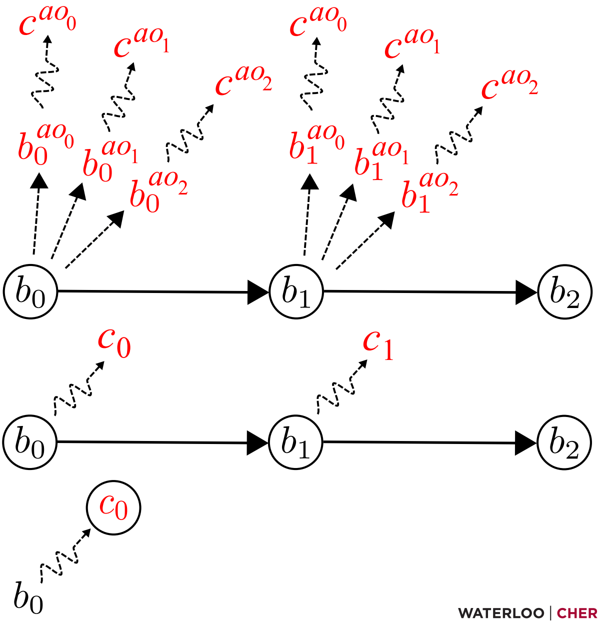


Execution of Upper Bound Policies

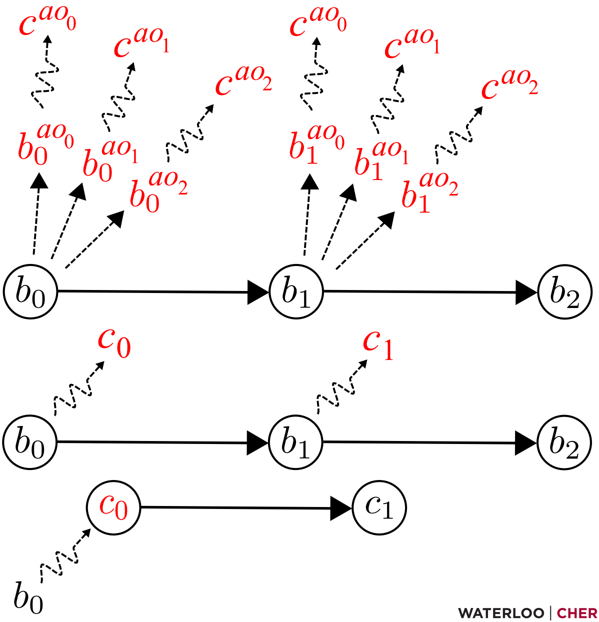


b_0

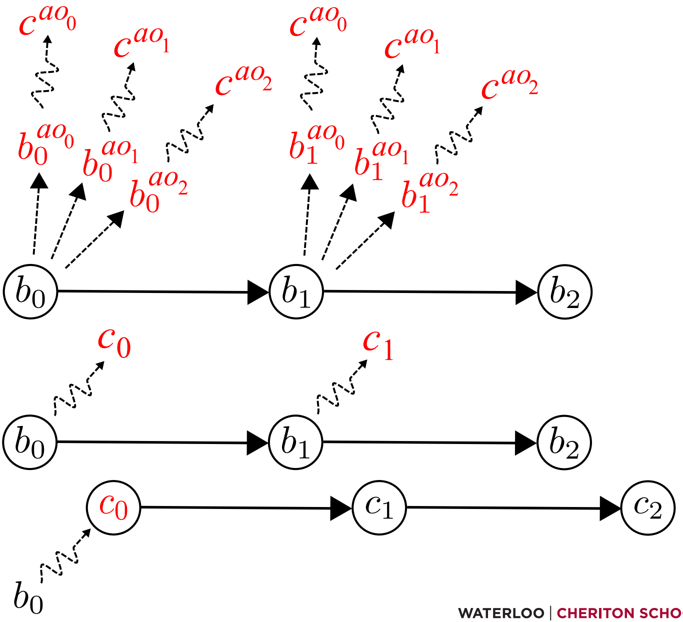
Execution of Upper Bound Policies



Execution of Upper Bound Policies



Execution of Upper Bound Policies



AO-deterministic POMDPs

- Deterministic POMDPs in Littman's thesis have deterministic T and O (all probabilities are either zero or one)
- Quasi-deterministic POMDPs have deterministic T (Besse and Chaib-draa 2009)
- **We introduce AO-deterministic POMDPs when all $T_{a,o}$ matrices have at most one non-zero entry in every row**—actions can be stochastic!
- All deterministic and quasi-deterministic POMDPs are AO-deterministic, but there exist POMDPs that are AO-deterministic but are neither deterministic nor quasi-deterministic (e.g. **baseball**)
- A few other benchmarks form ICAPS-IPPC are AO-deterministic, e.g., **rockSample-7_8** and **underwterNav**

Why the AO-deterministic definition matters?

- $T_{a,o} =$

$$
\begin{array}{c} \\ s_1 \\ \dots \\ s_n \\ b_{n+1} \\ \dots \\ b_{n+m} \end{array}
\begin{array}{c}
\begin{array}{cccccc} s'_1 & \dots & s'_n & b'_{n+1} & \dots & b'_{n+m} \end{array} \\
\begin{pmatrix}
c_{1,1} & \dots & c_{1,n} & c_{1,n+1} & \dots & c_{1,n+m} \\
\dots & \dots & \dots & \dots & \dots & \dots \\
c_{n,1} & \dots & c_{n,n} & c_{n,n+1} & \dots & c_{n,n+m} \\
c_{n+1,1} & \dots & c_{n+1,n} & c_{n+1,n+1} & \dots & c_{n+1,n+m} \\
\dots & \dots & \dots & \dots & \dots & \dots \\
c_{n+m,1} & \dots & c_{n+m,n} & c_{n+m,n+1} & \dots & c_{n+m,n+m}
\end{pmatrix}
\end{array}
$$

- If $b_{a,o}$ is a state of the augmented POMDP, then the row for (b, a, o) has at most one non-zero entry—$T_{a,o}$ is becoming "more deterministic" when upper bounds are improved
- Theorem: **Policies that are optimal for the underlying MDP of an AO-deterministic POMDP are also optimal at the corner beliefs of this POMDP.**

Our Algorithm

The key conclusion: **search for new beliefs going forward from corners** as well (not only from b_0 as it is the case in GapMin, HSVI, or SARSOP)

Algorithm 1: New Anchor Beliefs ($N = 50$ in all experiments)

Data: S, G, $\bar{V}^G$, OCF, N, Q- in augmented space

$G_{new} \leftarrow \emptyset$
if POMDP is AO-deterministic then
    for *i=1 to N* **do**
        if $b_0 \in G$ **then**
            return G_{new}; /* nothing to improve */
        else
            $b \leftarrow$ **ForwardSearch** or **LAO***
            add b into G_{new}
else
    $H \leftarrow$ **SampleCorners**(G, OCF, N) ; /* sample among corners with non-deterministic transitions only */
    for *all corner beliefs $b \in H$* **do**
        repeat
            $c \leftarrow$ embed b into augmented space
            $a^* \leftarrow$ action for c using augmented Q-values
            sample observation o according to $P(o|b, a^*)$
            $b \leftarrow b_{a,o}$
        until $b \notin G \cup G_{new}$
        add b into G_{new}
return G_{new}

Results—Execution Time and Quality of Upper Bound Policies

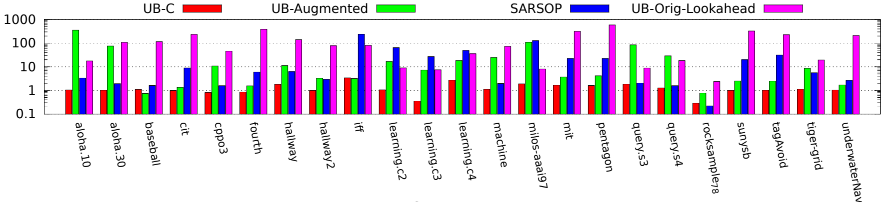


Figure: Ratio of the execution time to QMDP execution time

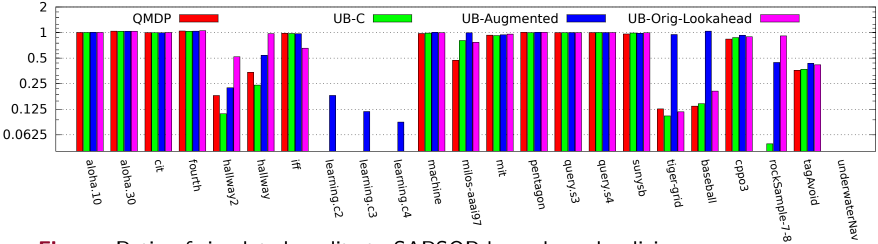


Figure: Ratio of simulated quality to SARSOP lower bound policies

Results—AO-deterministic POMDPs

Table: The quality of upper bounds (UB) after 1000 seconds of planning (AO-deterministic POMDPs).

problem	algorithm	gap	LB	UB	$\|\Gamma\|$	$\|\bar{V}\|$	time	UB	$\|\bar{V}\|$	time
baseball	hsvi2	1e-3	0.6412	**0.6412**	991	n.a.	999			
$\|\mathcal{S}\| = 7681$	sarsop	7e-4	0.6412	0.6419	1453	1694	400	0.6412	3878	2346
$\|\mathcal{A}\|=6$ $\|\mathcal{O}\|=9$	GapMin	5.01	0.6346	5.6500	1	1	281	0.6434	52	15219
$\gamma = 0.999$	Aug-OCF			0.6413		3051	970			
rockSample-7_8	hsvi2	3.56	20.91	**24.46**	4752	n.a.	998			
$\|\mathcal{S}\| = 12545$	sarsop	4.12	20.91	25.02	3119	2473	999	24.46	8520	9806
$\|\mathcal{A}\|=13$ $\|\mathcal{O}\|=2$	GapMin	25.07	7.35	32.42	1	1	6.18	26.84	30	13855
$\gamma = 0.950$	Aug-OCF			24.81		3351	978			
underwaterNav	hsvi2	23.4	729.9	**753.3**	3545	n.a.	1000			
$\|\mathcal{S}\| = 2653$	sarsop	23.4	731.0	754.4	7918	2820	999	754.0	7947	10014
$\|\mathcal{A}\|=6$ $\|\mathcal{O}\|=103$	GapMin	80.2	675.06	755.3	1	1	742	754.8	115	10113
$\gamma = 0.950$	Aug-OCF			754.6		1830	471.0			

Results—non-AO-deterministic POMDPs

Table: The quality of upper bounds (UB) after 1000 seconds of planning (non AO-deterministic POMDPs).

problem	algorithm	gap	LB	UB	$\|\Gamma\|$	$\|\bar{V}\|$	time
aloha.10	hsvi2	9.0	535.4	544.4	4729	n.a.	997
$\|\mathcal{S}\| = 30$	sarsop	9.5	535.2	544.7	48	2151	1000
$\|\mathcal{A}\| = 9$, $\|\mathcal{O}\| = 3$	GapMin	10.7	533.5	544.2	81	223	972
$\gamma = 0.999$	Aug-H			539.6 $\pm$ 0.01		1999.1 $\pm$ 21.7	981.9 $\pm$ 3.6
	Aug-OCF			**539.0** $\pm$ 0.01		3345 $\pm$ 22.8	984.5 $\pm$ 2.8
hallway2	hsvi2	0.5250	0.3612	0.8862	2393	n.a.	997
$\|\mathcal{S}\| = 92$	sarsop	0.5247	0.3737	0.8984	262	1519	992
$\|\mathcal{A}\| = 5$, $\|\mathcal{O}\| = 17$	GapMin	0.4495	0.3497	**0.7992**	122	218	835.5
$\gamma = 0.950$	Aug-H			0.897 $\pm$ 0.0		1349.6 $\pm$ 11.5	896.2 $\pm$ 17.6
	Aug-OCF			0.805 $\pm$ 0.0		861.0 $\pm$ 6.3	944.1 $\pm$ 12.1
hallway	hsvi2	0.250	0.945	1.195	1367	n.a.	996
$\|\mathcal{S}\| = 60$	sarsop	0.210	0.995	1.206	456	1713	998
$\|\mathcal{A}\| = 5$, $\|\mathcal{O}\| = 21$	GapMin	0.132	0.989	1.122	94	176	974
$\gamma = 0.950$	Aug-H			1.186 $\pm$ 0.0		1189.7 $\pm$ 13.0	947.1 $\pm$ 13.3
	Aug-OCF			**1.095** $\pm$ 0.0		951.0 $\pm$ 7.0	946.1 $\pm$ 11.5
machine	hsvi2	3.49	63.18	66.66	662	n.a.	982
$\|\mathcal{S}\| = 256$	sarsop	3.57	63.18	66.75	150	2742	998
$\|\mathcal{A}\| = 4$, $\|\mathcal{O}\| = 16$	GapMin	3.48	62.38	65.87	58	208	898
$\gamma = 0.990$	Aug-H			64.68 $\pm$ 0.0		972.0 $\pm$ 0.0	809.0 $\pm$ 4.1
	Aug-OCF			**63.84** $\pm$ 0.01		965.0 $\pm$ 12.3	918.7 $\pm$ 17.1
tagAvoid	hsvi2	3.207	-6.150	-2.943	2896	n.a.	1000
$\|\mathcal{S}\| = 870$	sarsop	3.455	-6.142	-2.686	9324	8049	989
$\|\mathcal{A}\| = 5$, $\|\mathcal{O}\| = 30$	GapMin	12.70	-14.0	-1.291	77	310	773
$\gamma = 0.950$	Aug-H			-0.672 $\pm$ 0.0		5840.3 $\pm$ 55.8	949.0 $\pm$ 5.5
	Aug-OCF			**-3.660** $\pm$ 0.0		6861.0 $\pm$ 50.8	990.5 $\pm$ 1.4

Conclusion

1. Efficient execution of upper bound policies (e.g. in an augmented space) was shown—useful for deploying upper bound policies or using them to guide branch-and-bound
2. AO-deterministic POMDPs generalise existing definitions of deterministic and quasi-deterministic POMDPs, yet are specific enough to explain the process of refining upper bounds and to show where the augmented POMDP is converging to
3. AO-deterministic POMDPs lead to a straightforward approach that can compute the tightest upper bounds without any use of lower bounds