Noname manuscript No.
(will be inserted by the editor)

Adaptive Aggregation for Reinforcement Learning in Average Reward Markov Decision Processes

Ronald Ortner

the date of receipt and acceptance should be inserted later

Abstract We present an algorithm which aggregates online when learning to behave optimally in an average reward Markov decision process. The algorithm is based on the reinforcement learning algorithm UCRL and uses confidence intervals for aggregating the state space. We derive bounds on the regret our algorithm suffers with respect to an optimal policy. These bounds are only slightly worse than the original bounds for UCRL.

1 Introduction

One big problem which makes reinforcement learning algorithms infeasible for most practical applications is that typical algorithms are not efficient in environments with large state spaces. While many real world problems could in principle be handled by representing them as Markov decision processes (MDPs), such representations usually have a large state space, so that most reinforcement learning algorithms are too costly, as their complexity and regret (the lost total reward with respect to an optimal strategy) grows linearly or even polynomially with the number of states and actions. Unlike humans, reinforcement learning algorithms are not able to make use of the environment's structure, which prevents exploitation of symmetries and similarities in a learning problem.

In this paper we pursue the idea of simplifying the complexity of a reinforcement learning problem by employing adaptive aggregation while learning optimal behavior in an MDP. We present a respective algorithm – an adaptation of an ordinary online reinforcement learning algorithm – and show that it is competitive to the original version of the algorithm which does not use aggregation.

While there is a lot of literature concerning aggregation and similarity in MDPs, most of this work considers the approximate dynamic programming view, when the underlying MDP is known. Specific topics of this work include e.g. approximate value or policy iteration [26]. For an overview see e.g. [4,22,20].

INRIA Lille-Nord Europe, équipe SequeL, 40 avenue Halley, 59650 Villeneuve d'Ascq, France
E-mail: ronald.ortner@unileoben.ac.at

There is also work on learning in MDPs with exploitation of an underlying (similarity or other) structure. Thus, [17] considers a (discrete) state space which is partitioned into sets of states of the same type, where states of the same type are assumed to have the same transition dynamics. That way, less experience is needed until good performance is achieved when compared to standard algorithms. Also, reinforcement learning in factored MDPs has been considered in [28,19,9], including sample complexity bounds for a variant of the R-Max algorithm [18].

The setting when the underlying symmetries of the MDP are not known beforehand as considered here, has been discussed in the literature only to a limited extent. Thus, [21] considers an online clustering algorithm for discounted reinforcement learning with options. A heuristic algorithm for adaptive *soft* state aggregation, where states may belong to several clusters (with some probability) is suggested in [27]. A different kind of adaptive aggregation is considered in [3], where states with similar progress in policy iteration are aggregated.

2 Setting

Definition 1 A *Markov decision process (MDP) M* is given by a (finite) state space $\mathcal{S}$, a finite action space $\mathcal{A}$, an initial state $s_1 \in \mathcal{S}$, reward distributions with support in $[0,1]$ and mean $r(s,a)$, and transition probabilities $p\big(s'|s,a\big)$, which determine the probability of a transition from state s to state s' when action a is chosen in state s.

A (stationary) *policy* $\pi : \mathcal{S} \to \mathcal{A}$ assigns an action to each state, and we are interested in an optimal policy that maximizes the mean average reward

$$\lim_{T\to\infty} \tfrac{1}{T} \sum_{t=1}^{T} \mathbb{E}\left[r\big(s_t, \pi(s_t)\big)\right],$$

where s_t denotes the state visited at step t. It is well known [24] that in *communicating* MDPs (i.e., MDPs with finite *diameter*, cf. below) the average reward is maximized by a stationary policy. We measure the performance of an algorithm learning to behave optimally by the regret it suffers with respect to the average reward ρ^* achievable by an optimal policy. More precisely, the *regret after T steps* is given by

$$T\rho^* - \sum_{t=1}^{T} r_t,$$

where r_t is the reward collected by the algorithm at step t.

We will make use of the *diameter* as a parameter of the transition structure of an MDP as introduced in [14]. The diameter D is the time it takes at most to move from any state s to any other state s', using an appropriate policy.

Definition 2 Given an MDP M, let $T(s'|M,\pi,s)$ be the random variable for the first time step in which state s' is visited when starting in state s and choosing actions in each state according to policy π. The *diameter* of M then is defined as

$$D(M) := \max_{s\neq s'\in\mathcal{S}} \min_{\pi:\mathcal{S}\to\mathcal{A}} \mathbb{E}\left[T(s'|M,\pi,s)\right].$$

3 Aggregation

If there are certain symmetries present in an MDP M, it may be possible to aggregate M, thus obtaining an MDP with smaller state space. First, let us consider the following kind of structure. A partition $\{S_1,\ldots,S_n\}$ of the state space may serve as state space of such an aggregated MDP, if for all s,s' in the same set S_i and for each a in $\mathcal{A}$ it holds that $r(s,a)=r(s',a)$, and for each S_j it holds that $\sum_{s''\in S_j} p(s''|s,a)=\sum_{s''\in S_j} p(s''|s',a')$. These values also are the mean rewards and transition probabilities in the aggregated MDP, respectively. That is, if $s\in S_i$, then one can set $r(S_i,a):=r(s,a)$ and $p(S_j|S_i,a):=\sum_{s''\in S_j} p(s''|s,a)$. In this case aggregation suits perfectly, and there is no loss when moving from the original MDP to the aggregated MDP (cf. [23]).

Rather than such *perfect* aggregations we want to consider aggregations that approximate the original MDP. Thus an aggregation with state space $\{S_1,\ldots,S_n\}$ (again a partition of the original state space) is called an *ε-aggregation*, if for all s,s' in the same set S_i and for each a in $\mathcal{A}$ it holds that $|r(s,a)-r(s',a)|<\varepsilon$, and for each S_j it holds that $\big|\sum_{s''\in S_j} p(s''|s,a)-\sum_{s''\in S_j} p(s''|s',a)\big|<\varepsilon$.

In the aggregated MDP, one has to specify the rewards and transition probabilities. This can be e.g. done by taking the average over all states contained in an aggregated state, so that $r(S_i,a):=\frac{1}{|S_i|}\sum_{s\in S_i} r(s,a)$ and $p(S_j|S_i,a):=\frac{1}{|S_i|}\sum_{s\in S_i}\sum_{s''\in S_j} p(s''|s,a)$. Alternatively, one may pick an arbitrary reference state s in each S_i and set (as for perfect aggregations) $r(S_i,a):=r(s,a)$ as well as $p(S_j|S_i,a):=\sum_{s''\in S_j} p(s''|s,a)$. Perfect as well as ε-aggregations can be found in the literature under various names, see e.g. [12,10,11,23,25].

3.1 Aggregation of Bounded Parameter MDPs

Here we will consider an extension of Markov decision processes where for rewards and transition probabilities not a single value but a range of values is specified.

Definition 3 A *bounded parameter MDP (BPMDP)* is defined by a state space $\mathcal{S}$, an action space $\mathcal{A}$, an initial state $s_1\in\mathcal{S}$, and ranges $\bar{r}(s,a)$ and $\bar{p}(\cdot|s,a)$ for rewards and transition probabilities, respectively. Thus the range $\bar{r}(s,a)$ of values for the rewards for a given state-action pair s,a is defined by

$$\bar{r}(s,a):=\big\{r\in[0,1]:|\hat{r}(s,a)-r|\le d_r(s,a)\big\},\tag{1}$$

while the range of transition probabilities $\bar{p}(\cdot|s,a)$ is defined to be the set of all transition probability distributions $p(\cdot)$ over $\mathcal{S}$ such that

$$\big\|\hat{p}(\cdot|s,a)-p(\cdot)\big\|_1\ \le\ d_p(s,a).\tag{2}$$

Here, $\hat{r}(s,a)$ and $\hat{p}\,(\cdot|s,a)$ are given reference values and $d_r(s,a)$, $d_p(s,a)$ distance values for rewards and transition probabilities, respectively.

Bounded parameter MDPs have been considered in similar form before, see e.g. [13,32]. The reference values of a BPMDP can be interpreted as estimates, where the range indicates an uncertainty about the estimates of the rewards and transition probabilities. We will be interested in BPMDPs where the ranges are

given by confidence intervals.
On the other hand, a BPMDP can be considered to be an MDP with continuous action space when one is allowed to pick an arbitrary value from the ranges. In [14], *extended value iteration* has been introduced in order to find an optimal policy on a BPMDP. This has been employed by the UCRL algorithm also given in [14], which uses extended value iteration in order to find an optimal policy on an optimistically chosen MDP. The latter is obtained by choosing optimistic values (with respect to the achievable average reward) in the ranges of a BPMDP given by confidence intervals.

Now we are interested in aggregating BPMDPs. Given a BPMDP, we aggregate states for which the ranges given by (1) and (2) intersect. Generally, one may want to aggregate states whose ranges have nonempty intersection. However, finding such aggregations is computationally hard (see Section 6 below and [10]), so that we will rather aggregate states for which the union of ranges is connected.

Definition 4 Given a BPMDP M with reward ranges $\bar{r}(s,a)$ and transition probability ranges $\bar{p}(\cdot|s,a)$, a partition of the state space $\{S_1,\dots,S_n\}$ constitutes an *aggregation of* M if for all S_j it holds that $\bigcup_{s\in S_j}\bar{r}(s,a)$ as well as $\bigcup_{s\in S_j}\bar{p}(\cdot|s,a)$ are connected sets for all actions a.

If the intersection of ranges is nonempty, that is, $\bigcap_{s\in S_j}\bar{r}(s,a)\neq\varnothing$ and $\bigcap_{s\in S_j}\bar{p}(\cdot|s,a)\neq\varnothing$ for all S_j and all a, the partition obviously constitutes an aggregation, which we call *neat*.

In the following we will refer to the elements of the partition of the state space as *aggregated* states or *meta-states*. In most cases we will use variables S_j, S_j' to denote aggregated states from some particular partition of the state space which constitutes an aggregation.

As for ε-aggregations for ordinary MDPs, we have to set the rewards and transition probabilities in the aggregated BPMDP. That is, we have to define respective ranges. While the union of the ranges of rewards again gives an interval, this does not hold for a union of intersecting transition probability simplices as given by (2). Thus, in order to define the range for the transition probabilities in the aggregated BPMDP we will just take some simplex which contains the ranges of the respective states aggregated. For this, we will need the following concept.

Definition 5 Let $\mathcal{B}=\{B_i\}_{i=1}^n$ be a family of at least two sets $B_i\subseteq X$, and assume that $\bigcup_{i=1}^n B_i$ is connected. Two sets B_j, B_k are said to be *m-connectible* if there are $(m-2)$ pairwise disjoint sets $B_{\ell_1},\dots,B_{\ell_{m-2}}$ in $\mathcal{B}\setminus\{B_j,B_k\}$ such that

$$\begin{aligned}
&B_j\cap B_{\ell_1}\neq\varnothing,\\
&B_{\ell_i}\cap B_{\ell_{i+1}}\neq\varnothing \text{ for } i=1,\dots,m-3,\\
&B_{\ell_{m-2}}\cap B_k\neq\varnothing.
\end{aligned}$$

If $|\mathcal{B}|=1$, we set $\mathcal{B}$ to be 1-connectible. Further, we say that $\mathcal{B}$ is *m-connected* if m is the smallest number such that any two sets in $\mathcal{B}$ are m-connectible.

Then, given a BPMDP M with reward ranges $\bar{r}(s,a)$, transition probability ranges $\bar{p}(\cdot|s,a)$, distance values $d_r(s,a)$, $d_p(s,a)$ and an aggregation $\mathcal{S}^{\text{agg}}=\{S_1,\dots,S_n\}$ of M, we set the reward range $\bar{r}(S_j,a)$ for each aggregated state-action pair S_j,a to be

$$\bar{r}(S_j,a):=\bigcup_{s\in S_j}\bar{r}(s,a), \tag{3}$$

while the range for transition probabilities $\bar{p}(\cdot|S_j,a)$ is defined to be the set of all transition probability distributions $p(\cdot)$ over the set $\mathcal{S}^{\text{agg}}$ for which

$$\sum_{S'_j} \left|p(S'_j|s_0,a) - p(S'_j)\right| \leq c \cdot \max_{s\in S_j} d_p(s,a), \tag{4}$$

where s_0 is an arbitrary fixed state in S_j, $p(S'_j|s_0,a) := \sum_{s\in S'_j} p(s|s_0,a)$, and $\{\bar{p}(\cdot|s,a) : s \in S_j\}$ is c-connected. If one considers $p(\cdot|s,a)$ as a transition probability distribution over $\mathcal{S}^{\text{agg}}$ (by summing over the states contained in a meta-state as in the formula for $p(S'_j|s_0,a)$), then one can rewrite (4) as

$$\left\|p(\cdot|s_0,a) - p(\cdot)\right\|_1 \leq c \cdot \max_{s\in S_j} d_p(s,a),$$

that is, the range of the transition probabilities is defined according to (2). In the same fashion, it is straightforward to see that each transition probability distribution (considered again over $\mathcal{S}^{\text{agg}}$) contained in a range $\bar{p}(\cdot|s,a)$ with $s \in S_j$ is also contained in the range $\bar{p}(\cdot|S_j,a)$, that is, $\bigcup_{s\in S_j} \bar{p}(\cdot|s,a) \subseteq \bar{p}(\cdot|S_j,a)$.

Definition 6 An aggregation $\{S_1,\ldots,S_n\}$ of a BPMDP has *reward connectivity* c_r, if c_r is the smallest number such that $\{\bar{r}(s,a) : s \in S_j\}$ is at most c_r-connected for each S_j. Similarly, the *transition connectivity* is the smallest number c_p such that $\{\bar{p}(\cdot|s,a) : s \in S_j\}$ is at most c_p-connected for each S_j.

Note that the reward and transition connectivity of neat aggregations is $c_r = c_p = 2$.

4 Online Aggregation

4.1 The Algorithm

Our algorithm UCAgg (shown as Algorithm 1) employs confidence intervals for rewards and transition probabilities just like the algorithm UCRL introduced in [14]. Similarly as the original UCRL algorithm our algorithm proceeds in episodes, in which the same policy is employed. Also for UCAgg the idea is to implement the *optimism in the face of uncertainty* maxim by assuming in each episode the most promising values within the confidence intervals so that the average reward is maximized. However, unlike UCRL, UCAgg tries to aggregate the BPMDP given by the confidence intervals in order to reduce the size of the MDP. Only on the aggregated BPMDP the original UCRL algorithm is executed.

Condition (7) in the algorithm guarantees that the lengths of confidence intervals of states aggregated are comparable. Indeed, it is easy to give examples where the lack of such a condition may lead to bad behavior of the algorithm. (The following example has been suggested by Peter Auer.) Thus, it may happen that two states s^-, s^+ are aggregated, where the confidence intervals of s^- are small and the confidence intervals of s^+ are large. If s^- gives low reward while s^+ gives high reward, the aggregated state containing s^-, s^+ will look promising (given the high reward and the large confidence intervals of s^+). However, an algorithm visiting

Algorithm 1 The UCAgg algorithm

```
Input: A confidence parameter $\delta \in (0,1)$, $\mathcal{S}$ and $\mathcal{A}$.
Initialization: Set $t := 1$, and observe the initial state $s_1$.
for episodes $k = 1, 2, \ldots$ do
  Initialize episode $k$:
  1. Set the start time of episode $k$, $t_k := t$.
  2. For $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$ let $N_k(s,a)$ be the state-action counts prior to episode $k$, and
     $v_k(s,a)$ the state-action counts in episode $k$. Further, set $S := |\mathcal{S}|$, $A := \mathcal{A}$, and let
     $R_k(s,a)$ be the observed accumulated rewards before episode $k$, and $P_k(s,a,s')$ the
     number of times a transition from $s$ to $s'$ has been observed after choosing action $a$
     in $s$ before episode $k$. Compute estimates
        $\hat{r}_k(s,a) := \frac{R_k(s,a)}{\max\{1, N_k(s,a)\}}, \quad \hat{p}_k(s'|s,a) := \frac{P_k(s,a,s')}{\max\{1, N_k(s,a)\}}.$
     Further, for each estimate define confidence intervals as follows: Let
        $\mathrm{conf}_r(s,a) := \left\{ r^c \in [0,1] : \left| r^c - \hat{r}_k(s,a) \right| \leq \sqrt{\frac{7\log(2SAt_k/\delta)}{2\max\{1,N_k(s,a)\}}} \right\}$   (5)
     and set $\mathrm{conf}_p(s,a)$ to be the set of all transition probability distributions $p^c(\cdot)$ with
        $\left\| p^c(\cdot) - \hat{p}_k(\cdot|s,a) \right\|_1 \leq \sqrt{\frac{14S\log(2At_k/\delta)}{\max\{1,N_k(s,a)\}}}.$   (6)
  Aggregate and compute policy $\tilde{\pi}_k^{\mathrm{agg}}$:
  3. Calculate an aggregation of the BPMDP with ranges as given by (5) and (6) such
     that for two states $s, s'$ in the same aggregated state:
        $\frac{1}{2} \leq |N_k(s,a)|/|N_k(s',a)| \leq 2,$   (7)
  4. Use extended value iteration as given in [14] on the aggregated BPMDP to find a
     $1/\sqrt{t_k}$-optimal policy $\tilde{\pi}_k^{\mathrm{agg}}$ and an optimistic MDP $\tilde{M}_k^{\mathrm{agg}}$.
  Execute policy $\tilde{\pi}_k^{\mathrm{agg}}$ on aggregated MDP:
  5. while $v_k(s_t, \tilde{\pi}_k^{\mathrm{agg}}(S_t)) < \max\{1, N_k(s_t, \tilde{\pi}_k^{\mathrm{agg}}(S_t))\}$ do
     (a) Choose action $a_t = \tilde{\pi}_k^{\mathrm{agg}}(S_t)$, where $S_t$ is the aggregated state which contains $s_t$,
         obtain reward $r_t$, and observe next state $s_{t+1}$.
     (b) Set $t := t + 1$.
     end while
end for
```

the respective aggregated state may only visit s^-. This will not reduce the confidence intervals of s^+, and the algorithm will keep playing the suboptimal policy visiting s^- instead of s^+.

Condition (7) also levels the amount of exploration in similar states: If two sets of states have similar rewards and transition probabilities, yet states in one set have been sampled significantly more often than states in the other set, the latter set will be more interesting due to the larger confidence intervals. Consequently, states in this set are likely to be sampled more often, so that the amount of exploration will be leveled over the two sets.

The algorithm UCAgg leaves open how to aggregate. For this, we propose a simple algorithm in Section 6 below.

4.2 A Regret Bound for Online Aggregation

Theorem 1 *Given an MDP with S states, A actions and diameter D, with probability of at least $1-\delta$ it holds that for any initial state $s_1 \in \mathcal{S}$ and any $T>1$, the regret of* UCAgg *using an aggregation algorithm which guarantees reward and transition connectivity of at most C is bounded by*

$$49 \cdot CDS\sqrt{AT\log\left(\tfrac{T}{\delta}\right)}.$$

Compared to the regret bound for the UCRL algorithm, the cost of aggregation is basically the additional factor C, that is, the additional regret depends on the quality of the used aggregation algorithm. In particular, if the aggregation algorithm always produces a neat aggregation, the bounds for the regret are essentially the same as for the original UCRL algorithm.

Note that the theorem does not contain any assumptions about the underlying MDP. In particular, it is not necessary that the MDP has some structure that favors aggregation. Indeed, the algorithm may be conducted also on other MDPs with some benefit in computation time, as extended value iteration is conducted on an MDP which will have a smaller state space especially in the first episodes when the confidence intervals are large.

5 Proof of Theorem 1

5.1 Preliminaries

Let M be the true underlying MDP with state space $\mathcal{S}$, action space $\mathcal{A}$, rewards $r(s,a)$ and transition probability distributions $p(\cdot|s,a)$. Further, for a given episode k consider the BPMDP determined by the ranges specified in (5) and (6). Alternatively, we may consider this BPMDP as the set $\mathcal{M}_k$ of all plausible MDPs with state space $\mathcal{S}$ and action space $\mathcal{A}$, whose transition probabilities and rewards are contained in the confidence intervals given by (5) and (6). Although the confidence intervals eventually used by the aggregation are larger than the ones given in (5) and (6), we still make use of the latter. This is convenient, as it makes the analysis not only simpler, we can also use large parts of the proofs given in [14], where the same confidence intervals are used. Thus, in particular we also leave the general structure of the proof of Theorem 2 in [14] intact.

5.2 Splitting into Episodes

Recall that r_t is the random reward our algorithm receives at step t when starting in the initial state s_1. For given state-action counts $N(s,a)$ after T steps, the r_t are independent random variables, so that Hoeffding's inequality implies that the regret of UCAgg is bounded as

$$T\rho^* - \sum_{t=1}^{T} r_t \;<\; T\rho^* - \sum_{s,a} N(s,a)\cdot r(s,a) + \sqrt{\tfrac{5}{8}T\log\left(\tfrac{8T}{\delta}\right)}$$

with probability at least $1-\frac{\delta}{12T^{5/4}}$. Denoting the number of episodes started up to step T by m, we have $\sum_{k=1}^m v_k(s,a)=N(s,a)$ and $\sum_{s,a}N(s,a)=T$. Therefore, writing $\Delta_k:=\sum_{s,a}v_k(s,a)\big(\rho^*-r(s,a)\big)$, it follows that

$$T\rho^*-\sum_{t=1}^T r_t \;\le\; \sum_{k=1}^m \Delta_k+\sqrt{\tfrac{5}{8}T\log\left(\tfrac{8T}{\delta}\right)} \tag{8}$$

with probability at least $1-\frac{\delta}{12T^{5/4}}$.

5.3 Dealing with Failing Confidence Regions

Considering the regret of episodes in which the set of plausible MDPs $\mathcal{M}_k$ does not contain the true MDP M, $\sum_{k=1}^m \Delta_k \mathbb{1}_{M\notin\mathcal{M}_k}$, it was shown in [14], eq.(9) on p.1574 that with probability at least $1-\frac{\delta}{12T^{5/4}}$,

$$\sum_{k=1}^m \Delta_k \mathbb{1}_{M\notin\mathcal{M}_k} \;\le\; \sqrt{T}. \tag{9}$$

5.4 Episodes with $M\in\mathcal{M}_k$

Now let us assume that $M\in\mathcal{M}_k$ and consider the regret in a single episode k. Recall that $\tilde{\pi}_k^{\mathrm{agg}}$ is the optimal policy on the optimistically chosen aggregated MDP $\tilde{M}_k^{\mathrm{agg}}$ (cf. step 4 of the algorithm). Further, let $\tilde{\rho}_k^{\mathrm{agg}}$ be the average reward of $\tilde{\pi}_k^{\mathrm{agg}}$ on $\tilde{M}_k^{\mathrm{agg}}$, and denote the rewards and transition probabilities in $\tilde{M}_k^{\mathrm{agg}}$ by $\tilde{r}_k^{\mathrm{agg}}(S_j,a)$ and $\tilde{p}_k^{\mathrm{agg}}(\cdot|S_j,a)$, respectively.

By the assumption of $M\in\mathcal{M}_k$, and Theorem 7 of [14] about extended value iteration we get that $\tilde{\rho}_k^{\mathrm{agg}}\ge\rho^*-1/\sqrt{t_k}$. Thus for the regret Δ_k accumulated in episode k we obtain by definition of the optimistic rewards in the aggregated states, now writing $S_j(s)$ for the aggregated state which contains s

$$\begin{aligned}
\Delta_k &= \sum_{s,a} v_k(s,a)\big(\rho^*-r(s,a)\big)\\
&\le \sum_{s,a} v_k(s,a)\big(\tilde{\rho}_k^{\mathrm{agg}}-r(s,a)\big)+\sum_{s,a}\frac{v_k(s,a)}{\sqrt{t_k}}\\
&= \sum_{s,a} v_k(s,a)\Big(\tilde{\rho}_k^{\mathrm{agg}}-\tilde{r}_k^{\mathrm{agg}}(S_j(s),a)\Big)\\
&\quad+\sum_{s,a} v_k(s,a)\Big(\tilde{r}_k^{\mathrm{agg}}(S_j(s),a)-r(s,a)\Big)+\sum_{s,a}\frac{v_k(s,a)}{\sqrt{t_k}}\\
&\le \sum_{S_j,a} v_k^{\mathrm{agg}}(S_j,a)\big(\tilde{\rho}_k^{\mathrm{agg}}-\tilde{r}_k^{\mathrm{agg}}(S_j,a)\big)\\
&\quad+\sum_{s,a} v_k(s,a)\Big(\tilde{r}_k^{\mathrm{agg}}(S_j(s),a)-r(s,a)\Big)+\sum_{s,a}\frac{v_k(s,a)}{\sqrt{t_k}}\;,
\end{aligned} \tag{10}$$

where $v_k^{\mathrm{agg}}(S_j,a):=\sum_{s\in S_j}v_k(s,a)$ is the number of visits in states contained in aggregated state S_j in episode k.

5.4.1 Extended Value Iteration on the Aggregated Optimistic MDP

One of the results about extended value iteration [14] which easily can be adapted to the aggregated case is that

$$\max_{S_j} u_i(S_j) - \min_{S_j} u_i(S_j) \leq D. \tag{11}$$

Indeed, as shown in [14], eq.(11) on p.1575, the BPMDP given by (5) and (6) has diameter at most D. This also holds in the aggregated BPMDP, as the confidence intervals used are even larger. Further, when extended value iteration terminates at iteration i, then by eq.(9) of [14]

$$|u_{i+1}(S_j) - u_i(S_j) - \tilde{\rho}_k^{\mathrm{agg}}| \leq \frac{1}{\sqrt{t_k}} \tag{12}$$

for all aggregated states S_j. By definition of extended value iteration, we have

$$u_{i+1}(S_j) \;=\; \tilde{r}_k^{\mathrm{agg}}(S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j)) + \sum_{S'_j} \tilde{p}_k^{\mathrm{agg}}\left(S'_j | S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j)\right) \cdot u_i(S'_j),$$

and hence by (12)

$$\left| \Big(\tilde{\rho}_k^{\mathrm{agg}} - \tilde{r}_k^{\mathrm{agg}}(S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j)) \Big) - \Big(\sum_{S'_j} \tilde{p}_k^{\mathrm{agg}}\left(S'_j | S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j)\right) \cdot u_i(S'_j) - u_i(S_j) \Big) \right| \leq \frac{1}{\sqrt{t_k}}. \tag{13}$$

Let $\boldsymbol{r}_k := \big(\tilde{r}_k^{\mathrm{agg}}(S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j))\big)_{S_j}$ be the (column) vector of rewards for policy $\tilde{\pi}_k^{\mathrm{agg}}$, $\tilde{\boldsymbol{P}}_k^{\mathrm{agg}} := \big(\tilde{p}_k^{\mathrm{agg}}\left(S'_j | S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j)\right)\big)_{S_j, S'_j}$ the transition matrix of $\tilde{\pi}_k^{\mathrm{agg}}$ on $\tilde{M}_k^{\mathrm{agg}}$, $\boldsymbol{v}_k := \big(v_k(s, \tilde{\pi}_k^{\mathrm{agg}}(s))\big)_s$ the (row) vector of visit counts for each state and the corresponding action chosen by $\tilde{\pi}_k^{\mathrm{agg}}$ in the respective meta-state $S_j(s)$, and $\boldsymbol{v}_k^{\mathrm{agg}} := \big(v_k(S_j, \tilde{\pi}_k^{\mathrm{agg}}(S_j))\big)_{S_j}$ the respective aggregated vector of state visits. Then — recalling that $v_k(S_j, a) = 0$ for $a \neq \tilde{\pi}_k^{\mathrm{agg}}(S_j)$ — we can use (13) to obtain from (10) that

$$\Delta_k \leq \boldsymbol{v}_k^{\mathrm{agg}}\big(\tilde{\boldsymbol{P}}_k^{\mathrm{agg}} - \boldsymbol{I}\big)\boldsymbol{u}_i + \sum_{s,a} v_k(s,a)\Big(\tilde{r}_k^{\mathrm{agg}}\big(S_j(s), a\big) - r(s,a)\Big) + 2\sum_{s,a} \frac{v_k(s,a)}{\sqrt{t_k}}.$$

Since the rows of $\tilde{\boldsymbol{P}}_k^{\mathrm{agg}}$ sum to 1, we can replace $\boldsymbol{u}_i$ with — note that i actually depends on k — $\boldsymbol{w}_k^{\mathrm{agg}}$ defined by

$$w_k^{\mathrm{agg}}(S_j) := u_i(S_j) - \frac{\min_{S_j} u_i(S_j) + \max_{S_j} u_i(S_j)}{2},$$

such that it follows from (11) that $\|\boldsymbol{w}_k\|_\infty \leq D/2$. Further, since we assume $M \in \mathcal{M}_k$, we have by (5) and condition (7)

$$\tilde{r}_k^{\mathrm{agg}}\big(S_j(s), a\big) - r(s,a) \;\leq\; 2c_{r,k}\sqrt{\tfrac{7\log(2SAt_k/\delta)}{\max\{1, N_k(s,a)\}}},$$

where $c_{r,k}$ is the reward connectivity of the aggregation in episode k. Consequently,

$$\Delta_k \leq \boldsymbol{v}_k^{\mathrm{agg}}\big(\tilde{\boldsymbol{P}}_k^{\mathrm{agg}} - \boldsymbol{I}\big)\boldsymbol{w}_k^{\mathrm{agg}} + c_{r,k}\sum_{s,a} v_k(s,a)\sqrt{\tfrac{28\log(2SAt_k/\delta)}{\max\{1, N_k(s,a)\}}} + 2\sum_{s,a} \frac{v_k(s,a)}{\sqrt{t_k}}. \tag{14}$$

As $\max\{1, N_k(s,a)\} \leq t_k \leq T$, it follows from (14) that

$$\varDelta_k \leq \boldsymbol{v}_k^{\text{agg}}(\tilde{\boldsymbol{P}}_k^{\text{agg}} - \boldsymbol{I})\boldsymbol{w}_k^{\text{agg}} + c_{r,k}\left(\sqrt{28\log\left(\tfrac{2SAT}{\delta}\right)} + 2\right)\sum_{s,a}\tfrac{v_k(s,a)}{\sqrt{\max\{1,N_k(s,a)\}}}. \qquad (15)$$

5.4.2 The True Transition Matrix

We continue analyzing the term $\boldsymbol{v}_k^{\text{agg}}(\tilde{\boldsymbol{P}}_k^{\text{agg}} - \boldsymbol{I})\boldsymbol{w}_k$. Recall that S_t is the aggregated state at step t, and let $\boldsymbol{e}_\ell$ be the unit vector with ℓ-th coordinate 1 and all other coordinates 0. Now observe that

$$\begin{aligned}\boldsymbol{v}_k^{\text{agg}}(\tilde{\boldsymbol{P}}_k^{\text{agg}} - \boldsymbol{I})\boldsymbol{w}_k^{\text{agg}} &= \sum_{t=t_k}^{t_{k+1}-1}\Big(\tilde{p}_k^{\text{agg}}\left(\cdot|S_t,a_t\right) - \boldsymbol{e}_{S_t}\Big)\boldsymbol{w}_k^{\text{agg}} \qquad (16)\\ &= \sum_{t=t_k}^{t_{k+1}-1}\tilde{p}_k^{\text{agg}}\left(\cdot|S_t,a_t\right)\boldsymbol{w}_k^{\text{agg}} - \sum_{t=t_k}^{t_{k+1}-1} w_k^{\text{agg}}(S_{t+1}) + w_k^{\text{agg}}(S_{t_{k+1}}) - w_k^{\text{agg}}(S_{t_k}).\end{aligned}$$

Now recalling that $p\left(S_j|s,a\right) := \sum_{s'\in S_j} p\left(s'|s,a\right)$ for all s, a, S_j, we have

$$\begin{aligned}\sum_{t=t_k}^{t_{k+1}-1}\tilde{p}_k^{\text{agg}}\left(\cdot|S_t,a_t\right)\boldsymbol{w}_k^{\text{agg}} &= \sum_{t=t_k}^{t_{k+1}-1}\sum_{S_j}\tilde{p}_k^{\text{agg}}\left(S_j|S_t,a_t\right)w_k^{\text{agg}}(S_j) \qquad (17)\\ &= \sum_{t=t_k}^{t_{k+1}-1}\sum_{S_j}\Big(\tilde{p}_k^{\text{agg}}\left(S_j|S_t,a_t\right) - p\left(S_j|s_t,a_t\right)\Big)w_k^{\text{agg}}(S_j) + \sum_{t=t_k}^{t_{k+1}-1}\sum_s p\left(s|s_t,a_t\right)w_k(s),\end{aligned}$$

where we set $w_k(s) := w_k(S_j)$ for $s \in S_j$. By (6), (7), and the assumption that $M \in \mathcal{M}_k$ we have

$$\sum_{S_j}\Big|\tilde{p}_k^{\text{agg}}\left(S_j|S_t,a_t\right) - p\left(S_j|s_t,a_t\right)\Big| \leq 2c_{p,k}\sqrt{\tfrac{14S\log(2At_k/\delta)}{\frac{1}{2}\max\{1,N_k(s_t,a_t)\}}},$$

where $c_{p,k}$ is the transition connectivity of the aggregation used in episode k. Hence, recalling that $\|w_k\|_\infty \leq D/2$,

$$\begin{aligned}&\sum_{t=t_k}^{t_{k+1}-1}\sum_{S_j}\Big(\tilde{p}_k^{\text{agg}}\left(S_j|S_t,a_t\right) - p\left(S_j|s_t,a_t\right)\Big)w_k^{\text{agg}}(S_j)\\ &\quad\leq \sum_{t=t_k}^{t_{k+1}-1} D\cdot c_{p,k}\sqrt{\tfrac{14S\log(2At_k/\delta)}{\frac{1}{2}\max\{1,N_k(s_t,a_t)\}}} \leq c_{p,k}D\sum_{s,a}v_k(s,a)\sqrt{\tfrac{28S\log(2At_k/\delta)}{\max\{1,N_k(s,a)\}}}.\end{aligned}$$

Together with (16) and (17) this yields

$$\begin{aligned}\boldsymbol{v}_k^{\text{agg}}(\tilde{\boldsymbol{P}}_k^{\text{agg}} - \boldsymbol{I})\boldsymbol{w}_k^{\text{agg}} \;&\leq\; c_{p,k}D\sqrt{28S\log\left(\tfrac{2AT}{\delta}\right)}\sum_{s,a}\tfrac{v_k(s,a)}{\sqrt{\max\{1,N_k(s,a)\}}} \qquad (18)\\ &\quad+ \sum_{t=t_k}^{t_{k+1}-1}\sum_s p\left(s|s_t,a_t\right)w_k(s) - \sum_{t=t_k}^{t_{k+1}-1} w_k^{\text{agg}}(S_{t+1}) + w_k^{\text{agg}}(S_{t_{k+1}}) - w_k^{\text{agg}}(S_{t_k}).\end{aligned}$$

5.4.3 Summing over Episodes with $M \in \mathcal{M}_k$

Now we are ready to sum up the regret over the individual episodes. First, a martingale argument shows (see eq.(18) on p.1577 of [14]) that

$$\sum_{k=1}^{m} \left(\sum_{t=t_k}^{t_{k+1}-1} \sum_{s} p\left(s|s_t, a_t\right) w_k(s) - \sum_{t=t_k}^{t_{k+1}-1} w_k^{\mathrm{agg}}(S_{t+1}) + w_k^{\mathrm{agg}}(S_{t_{k+1}}) - w_k^{\mathrm{agg}}(S_{t_k}) \right)$$
$$\leq D\sqrt{\tfrac{5}{2} T \log\left(\tfrac{8T}{\delta}\right)} + DSA \log_2\left(\tfrac{8T}{SA}\right) \qquad (19)$$

with probability at least $1 - \frac{\delta}{12T^{5/4}}$. The second term in the bound of (19) stems from an upper bound on the number of episodes. As the criterion for episode termination is the same as for the UCRL algorithm, the same bound applies. Now setting $C := \max\{\max_k c_{r,k}, \max_k c_{p,k}\}$ and summing (15) over all episodes with $\mathcal{M} \in \mathcal{M}_k$, using (18) and (19), yields that with probability at least $1 - \frac{\delta}{12T^{5/4}}$

$$\begin{aligned}
\sum_{k=1}^{m} \Delta_k \mathbb{1}_{\mathcal{M}\in\mathcal{M}_k} &\leq \sum_{k=1}^{m} \boldsymbol{v}_k^{\mathrm{agg}} (\tilde{\boldsymbol{P}}_k^{\mathrm{agg}} - \boldsymbol{I}) \boldsymbol{w}_k^{\mathrm{agg}} \mathbb{1}_{\mathcal{M}\in\mathcal{M}_k} \\
&\quad + \sum_{k=1}^{m} \left(c_{r,k} \sqrt{28 \log\left(\tfrac{2SAT}{\delta}\right)} + 2 \right) \sum_{s,a} \tfrac{v_k(s,a)}{\sqrt{\max\{1, N_k(s,a)\}}} \\
&\leq CD \sqrt{28 S \log\left(\tfrac{2AT}{\delta}\right)} \cdot \sum_{k=1}^{m} \sum_{s,a} \tfrac{v_k(s,a)}{\sqrt{\max\{1, N_k(s,a)\}}} \\
&\quad + D\sqrt{\tfrac{5}{2} T \log\left(\tfrac{8T}{\delta}\right)} + DSA \log_2\left(\tfrac{8T}{SA}\right) \\
&\quad + C \left(\sqrt{28 \log\left(\tfrac{2SAT}{\delta}\right)} + 2 \right) \sum_{k=1}^{m} \sum_{s,a} \tfrac{v_k(s,a)}{\sqrt{\max\{1, N_k(s,a)\}}} \,. \qquad (20)
\end{aligned}$$

Now, as shown in [14], eq.(20) on p.1578, it holds that

$$\sum_{s,a} \sum_{k} \tfrac{v_k(s,a)}{\sqrt{\max\{1, N_k(s,a)\}}} \leq \left(\sqrt{2} + 1\right) \sqrt{SAT}, \qquad (21)$$

and we get from (20) after some minor simplifications that with probability at least $1 - \frac{\delta}{12T^{5/4}}$

$$\begin{aligned}
\sum_{k=1}^{m} \Delta_k \mathbb{1}_{\mathcal{M}\in\mathcal{M}_k} &\leq D\sqrt{\tfrac{5}{2} T \log\left(\tfrac{8T}{\delta}\right)} + DSA \log_2\left(\tfrac{8T}{SA}\right) \\
&\quad + \left(2CD \sqrt{28 S \log\left(\tfrac{2AT}{\delta}\right)} + 2 \right) \left(\sqrt{2} + 1\right) \sqrt{SAT} \,. \qquad (22)
\end{aligned}$$

Now, according to (8) we obtain by summing $\varDelta_k$ over all episodes, using (9) and (22), that the regret is bounded by

$$\begin{aligned}
T\rho^* - \sum_{t=1}^{T} r_t &\le \sum_{k=1}^{m} \varDelta_k \mathbb{1}_{\mathcal{M}\notin\mathcal{M}_k} + \sum_{k=1}^{m} \varDelta_k \mathbb{1}_{\mathcal{M}\in\mathcal{M}_k} + \sqrt{\tfrac{5}{8} T \log\left(\tfrac{8T}{\delta}\right)} \\
&\le \sqrt{T} + D\sqrt{\tfrac{5}{2} T \log\left(\tfrac{8T}{\delta}\right)} + DSA \log_2\left(\tfrac{8T}{SA}\right) \\
&\quad + \left(2CD\sqrt{28S\log\left(\tfrac{2AT}{\delta}\right)} + 2\right)\left(\sqrt{2}+1\right)\sqrt{SAT} + \sqrt{\tfrac{5}{8} T \log\left(\tfrac{8T}{\delta}\right)}
\end{aligned}$$

with probability at least $1 - \frac{\delta}{12T^{5/4}} - \frac{\delta}{12T^{5/4}} - \frac{\delta}{12T^{5/4}}$. Further simplifications (analogous to [14], Appendix C.4) and a union bound over all possible values of T, using that $\sum_{T=2}^{\infty} \frac{\delta}{4T^{5/4}} < \delta$, yield the bound of the theorem.

6 An Algorithm for Aggregation

So far, we have left it open how to aggregate. In this section we present a simple algorithm for aggregating BPMDPs. Algorithm 2 is a straightforward adaptation of the algorithm for approximate ε-aggregation given in [10]: Where the algorithm of [10] demands ε-closeness of rewards and transition probabilities, we have the condition of non-empty intersection of the respective ranges. Thus, the algorithm starts by building a graph with vertices in $\mathcal{S}$, where two states are connected if their reward ranges intersect for all actions. Then the algorithm checks whether the transition probabilities of two states behave similarly (i.e., ranges intersect) with respect to the current partition induced by the graph's connected components.

It is easy to see that if one is interested in neat partitions, one has to identify all the cliques in the graph. Thus, it is not surprising that finding a neat aggregation with minimal number of aggregated states is NP-hard, cf. [10].

Note that it is easy to adapt Algorithm 2 so that the output aggregation also satisfies assumption (7) of UCAgg. It is sufficient to add (7) as additional condition when building the similarity graph at the beginning of the algorithm. Thus, two vertices s, s' in the graph are only connected if their state-action counts $N_k(s,a)$, $N_k(s',a)$ satisfy (7) for all actions a.

The following results for the aggregation algorithm can be straightforwardly adapted from [10].

Lemma 1 *If there is a neat aggregation of a BPMDP such that states s, s' are contained in the same meta-state, then s, s' will not be separated by the algorithm.*

Lemma 2 *Let $\{S_1, \dots, S_n\}$ be the partition output by the algorithm. If $\max_i \mathit{diameter}(S_i) \ge \sqrt{|\mathcal{S}|}$ then there are at least $\sqrt{\frac{|\mathcal{S}|}{2}}$ states in the minimal neat aggregation.*

Theorem 2 *Algorithm 2 gives an aggregation which is by a factor of at most $2\sqrt{|\mathcal{S}|}$ larger than the minimal neat aggregation. Further, the reward and transition connectivity of the arising aggregation is at most $\sqrt{|\mathcal{S}|}$.*

Algorithm 2 Compute aggregation from BPMDP

```
Input: A BPMDP with state space $\mathcal{S}$, action space $\mathcal{A}$, reward ranges $\bar{r}(s,a)$ and transition probability ranges $\bar{p}(\cdot|s,a)$.
Let $G = (\mathcal{S}, E)$ be a graph with $(s,s') \in E$ if $\bar{r}(s,a) \cap \bar{r}(s',a) \neq \varnothing$ for all $a \in \mathcal{A}$.
repeat
  deleted := false
  for each $(s,s') \in E$ do
    for each $a \in \mathcal{A}$ do
      if $\bar{p}(\cdot|s,a) \cap \bar{p}(\cdot|s',a) = \varnothing$ then
        delete edge $(s,s')$ from $G$
        deleted := true
      end if
    end for
  end for
until deleted=false
compute the connected components $S_1,\ldots,S_n$ of $G$
if $\max_i diameter(S_i) \geq \sqrt{|\mathcal{S}|}$ then
  return $\mathcal{S}$
else
  return $\{S_1,\ldots,S_n\}$
end if
```

The following corollary follows immediately from Theorems 1 and 2.

Corollary 1 *Given an MDP with S states, A actions and diameter D, with probability of at least $1-\delta$ it holds that for any initial state $s_1 \in \mathcal{S}$ and any $T > 1$, the regret of* UCAGG *using Algorithm 2 for aggregation is bounded by*

$$49 \cdot DS^{3/2}\sqrt{AT\log\left(\tfrac{T}{\delta}\right)}.$$

Thus, compared to the original bound of UCRL, we have an additional factor of $\sqrt{S}$ in the regret bound.

7 Discussion

7.1 Related Regret Bounds in the Literature

So far, we have compared the regret bounds for our algorithm only to the original UCRL algorithm [14]. Here we would like to complement this with a brief summary of similar bounds in the literature. For a more detailed comparison we refer to Section 1.1 of [14].

In the wake of logarithmic regret bounds for the simpler multi-armed bandit problem [16,15,5,1], such bounds have been derived for the more general MDP setting as well, starting with the work of Burnetas and Katehakis [6]. Similarly to UCRL or UCAGG, the so-called *index policies* of [6] (cf. also the more recent [31]) choose actions optimistically by employing confidence bounds (however unlike in our case only for the estimates in the current state). The respective regret bound for *ergodic* MDPs is asymptotically logarithmic in the horizon T. Unlike the bound for UCAGG, this bound depends on the gap between the performance of the best and the second best action. A corresponding gap-dependent logarithmic bound holds for UCRL, too (cf. Theorem 4 in [14]). As our bound, this bound holds uniformly over time and under the weaker assumption that the underlying MDP is *communicating*. We believe that a corresponding logarithmic bound for UCAgg can be derived – adapting the proof of Theorem 4 in [14] – as well.

More recently, the results for UCRL have been extended by Bartlett and Tewari [2]. In the regret bound for their REGAL algorithm, the diameter is replaced with a smaller transition parameter D_1. Further, this bound also holds when the MDP has some *transient* states. However, as an additional assumption an upper bound on the parameter D_1 is needed. When no upper bound on D_1 is known, the bound's dependence on the number of states S deteriorates from S to $S^{3/2}$, similarly to our bound for UCAgg.

Another line of research pursues similar ideas in the setting with discounted rewards. Thus, Chang et al. [7,8] employ upper confidence bounds for a sampling-based approach, while the MBIE algorithm of Strehl and Littman [29,30] uses confidence bounds – similarly to UCRL – to compute an optimistic policy. In either case regret bounds which are logarithmic in the horizon are shown. While the two considered notions of regret differ from each other, they both consider the regret on the path chosen by the respective algorithm. Unlike that, in the undiscounted setting considered here the regret is measured with respect to the (expected) path chosen by the optimal policy, which is more difficult. For a detailed comparison of the notions of regret in discounted and undiscounted setting see Section 1.1 of [14].

7.2 Computational Considerations

The most costly step in the original UCRL algorithm is the calculation of the near-optimal policy in an optimistic BPMDP (corresponding to step 4 of UCAgg) by extended value iteration. Extended value iteration takes $O(AS^2)$ computation steps per iteration [14]. Thus, the speed up for the extended value iteration step for UCAgg is considerable when aggregation decreases the size of the state space. This has been confirmed by some preliminary experiments comparing the computational performance of UCRL with UCAgg. In most cases, UCAgg performed significantly faster than UCRL also when taking into account the time for computing the aggregation (the complexity of which is of the same order as that of a single step of extended value iteration). However, in some cases – in particular when extended value iteration converged in few iteration steps – the computational overhead caused by the proposed aggregation algorithm outweighed the gain for extended value iteration. On the other hand, concerning the regret UCRL and UCAgg seem to perform equally well, in spite of the slightly worse theoretical bound for UCAgg.

One way to speed up aggregation is to aggregate hierarchically. Thus, instead of recalculating the aggregation in each episode from the scratch (as in the given algorithm), one refines the aggregation from the previous episode. It is clear that this saves a lot of computation time: Instead of comparing each two states to each other, it is sufficient to compare states already contained in an aggregated state. While experiments indeed show improved computational performance, one has to be careful about the theoretical guarantees for hierarchical aggregation. It could happen that aggregations in earlier episodes are based on wrong confidence intervals, from which the algorithm may not recover subsequently. Thus, hierarchical aggregation has to be complemented by the use of appropriate confidence intervals in order to maintain the introduced regret bounds.

7.3 Outlook

Beside the question of improved aggregation techniques, another topic which has not been touched in this paper concerns the actions. We have only been checking for aggregation of states with respect to the same action and have neglected the possibility that two states s, s' may behave similarly when mapping the actions of s to the actions of s' in a suitable way. More generally, aggregation of actions may also make sense.

Acknowledgments

The author would like to thank Peter Auer for discussion of some preliminary ideas, Shiau Hong Lim for sharing his power plug adapter at Cumberland Lodge, and the anonymous reviewers for their comments, which helped to improve the paper. The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreements n° 216886 (PASCAL2 Network of Excellence), and n° 231495 (project CompLACS). The final version of this paper has been prepared when the author was supported by the Austrian Science Fund (FWF): J 3259-N13.

References

1. P. Auer, N. Cesa-Bianchi, and P. Fischer. Finite-time analysis of the multi-armed bandit problem. *Mach. Learn.*, 47:235–256, 2002.
2. P. L. Bartlett and A. Tewari. REGAL: A regularization based algorithm for reinforcement learning in weakly communicating MDPs. In *UAI 2009, Proc. 25th Annual Conference on Uncertainty in Artificial Intelligence*, pages 35–42, 2009.
3. D. P. Bertsekas and D. A. Castañon. Adaptive aggregation methods for infinite horizon dynamic programming. *IEEE Trans. Autom. Control*, 34(6):589–598, 1989.
4. L. Buşoniu, B. D. Schutter, and R. Babuška. Approximate dynamic programming and reinforcement learning. In *Interactive Collaborative Information Systems*, pages 3–44. 2010.
5. A. N. Burnetas and M. N. Katehakis. Optimal adaptive policies for sequential allocation problems. *Adv. in Appl. Math.*, 17(2):122–142, 1996.
6. A. N. Burnetas and M. N. Katehakis. Optimal adaptive policies for Markov decision processes. *Math. Oper. Res.*, 22(1):222–255, 1997.
7. H. S. Chang, M. C. Fu, J. Hu, and S. I. Marcus. An adaptive sampling algorithm for solving Markov decision processes. *Oper. Res.*, 53(1):126–139, 2005.
8. H. S. Chang, M. C. Fu, J. Hu, and S. I. Marcus. *Simulation-based Algorithms for Markov Decision Processes.* Springer, London, 2007.
9. C. Diuk, L. Li, and B. R. Leffler. The adaptive k-meteorologists problem and its application to structure learning and feature selection in reinforcement learning. In *Proc. 26th Annual International Conference on Machine Learning, ICML 2009*, page 32, 2009.
10. E. Even-Dar and Y. Mansour. Approximate equivalence of Markov decision processes. In *Computational Learning Theory and Kernel Machines, 16th Annual Conference on Computational Learning Theory and 7th Kernel Workshop*, pages 581–594, 2003.
11. N. Ferns, P. Panangaden, and D. Precup. Metrics for finite Markov decision processes. In *UAI '04, Proc. 20th Conference in Uncertainty in Artificial Intelligence*, pages 162–169, 2004.

12. R. Givan, T. Dean, and M. Greig. Equivalence notions and model minimization in Markov decision processes. *Artif. Intell.*, 147(1-2):163–223, 2003.
13. R. Givan, S. M. Leach, and T. Dean. Bounded-parameter Markov decision processes. *Artif. Intell.*, 122(1-2):71–109, 2000.
14. T. Jaksch, R. Ortner, and P. Auer. Near-optimal regret bounds for reinforcement learning. *J. Mach. Learn. Res.*, 11:1563–1600, 2010.
15. M. N. Katehakis and H. Robbins. Sequential choice from several populations. *Proceedings of the National Academy of Sciences of the United States of America*, 92(19):8584–8585, 1995.
16. T. L. Lai and H. Robbins. Asymptotically efficient adaptive allocation rules. *Adv. in Appl. Math.*, 6:4–22, 1985.
17. B. R. Leffler, M. L. Littman, and T. Edmunds. Efficient reinforcement learning with relocatable action models. In *Proc. 22nd AAAI Conference on Artificial Intelligence*, pages 572–577, 2007.
18. L. Li. *A Unifying Framework for Computational Reinforcement Learning Theory.* PhD thesis, Rutgers University, 2009.
19. L. Li, M. L. Littman, T. J. Walsh, and A. L. Strehl. Knows what it knows: a framework for self-aware learning. *Mach. Learn.*, 82(3):399–443, 2011.
20. L. Li, T. J. Walsh, and M. L. Littman. Towards a unified theory of state abstraction for MDPs. In *Proc. 9th International Symposium on Artificial Intelligence and Mathematics*, pages 531–539, 2006.
21. S. Mannor, I. Menache, A. Hoze, and U. Klein. Dynamic abstraction in reinforcement learning via clustering. In *Machine Learning, Proc. 21st International Conference, ICML 2004*, 2004.
22. R. Munos. Approximate dynamic programming. In O. Sigaud and O. Buffet, editors, *Markov Decision Processes in Artificial Intelligence*, chapter 3, pages 67–98. 2010.
23. R. Ortner. Pseudometrics for state aggregation in average reward Markov decision processes. In *Algorithmic Learning Theory, 18th International Conference, ALT 2007*, pages 373–387, 2007.
24. M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming.* John Wiley & Sons, Inc., New York, NY, USA, 1994.
25. B. Ravindran and A. G. Barto. SMDP homomorphisms: An algebraic approach to abstraction in semi-Markov decision processes. In *IJCAI-03, Proc. 18th International Joint Conference on Artificial Intelligence*, pages 1011–1018, 2003.
26. B. V. Roy. Performance loss bounds for approximate value iteration with state aggregation. *Math. Oper. Res.*, 31(2):234–244, 2006.
27. S. P. Singh, T. Jaakkola, and M. I. Jordan. Learning without state-estimation in partially observable Markovian decision processes. In *Machine Learning, Proc. 11th International Conference, ICML 1994*, pages 284–292, 1994.
28. A. L. Strehl, C. Diuk, and M. L. Littman. Efficient structure learning in factored-state MDPs. In *Proc. 22nd AAAI Conference on Artificial Intelligence*, pages 645–650, 2007.
29. A. L. Strehl and M. L. Littman. A theoretical analysis of model-based interval estimation. In *Machine Learning, Proc. 22nd International Conference, ICML 2005*, pages 857–864, 2005.
30. A. L. Strehl and M. L. Littman. An analysis of model-based interval estimation for Markov decision processes. *J. Comput. System Sci.*, 74(8):1309–1331, 2008.
31. A. Tewari and P. Bartlett. Optimistic linear programming gives logarithmic regret for irreducible MDPs. In *Adv. Neural Inf. Process. Syst. 20*, pages 1505–1512. 2008.
32. A. Tewari and P. L. Bartlett. Bounded parameter Markov decision processes with average reward criterion. In *Learning Theory, 20th Annual Conference on Learning Theory, COLT 2007*, pages 263–277, 2007.