

## 1 Abstract

We present an algorithm, UCRL2, which we show to be nearly optimal by a new analysis of the “**Optimism in the face of uncertainty**” paradigm. We consider undiscounted rewards and bound the **regret**, which is the sum of missed rewards (also during learning!) compared to an optimal policy. In order to describe the transition structure of an MDP we propose a new parameter: An MDP has **diameter**  $D$  if for any pair of states  $s, s'$  there is a policy which moves from  $s$  to  $s'$  in at most  $D$  steps (on average). We provide the best known bounds for undiscounted reinforcement learning. The total regret of UCRL2 is  $O(DS\sqrt{AT\log\frac{T}{\epsilon}})$  after  $T$  steps for any unknown MDP with  $S$  states,  $A$  actions per state, and diameter  $D$ . This bound holds with high probability and corresponds to a PAC-like bound of  $\Omega\left(\frac{D^2S^2A}{\epsilon^2}\log\frac{DS^2A}{\epsilon\delta}\right)$  steps until the average per step regret is at most  $\epsilon$ . We also present a lower bound of  $\Omega(\sqrt{DSAT})$  on the total regret of any learning algorithm. These new bounds demonstrate the utility of the diameter as structural parameter of an MDP.

## 2 Markov Decision Processes

We consider **Markov decision processes** (MDPs)  $M$  with

- $\mathcal{S}$  ... state space,
- $\mathcal{A}$  ... a set of actions available in each state,
- $\bar{r}(s, a)$  ... mean rewards for action  $a$  in state  $s$  with support in  $[0, 1]$ ,
- $p(s'|s, a)$  ... probability for transition to state  $s'$  from state  $s$  with action  $a$ .

Reinforcement learning of MDPs is considered as a standard model for learning with delayed feedback. In the undiscounted setting, the **accumulated reward** of an algorithm  $\mathfrak{A}$  after  $T$  steps in an MDP  $M$  is defined as

$$R(M, \mathfrak{A}, s, T) := \sum_{t=1}^T r_t,$$

where  $r_t$  are the rewards received during execution of  $\mathfrak{A}$  with initial state  $s$ . A stationary **policy** on an MDP  $M$  is a mapping  $\pi : \mathcal{S} \rightarrow \mathcal{A}$ . The **average reward**

$$\rho(M, \mathfrak{A}, s) := \lim_{T \rightarrow \infty} \frac{1}{T} \mathbb{E}[R(M, \mathfrak{A}, s, T)]$$

can be maximized by an appropriate stationary policy. We focus on MDPs where for each pair of states  $s, s'$  there is a policy which eventually reaches  $s'$  when starting in  $s$ . Then the **optimal average reward**  $\rho^*$  does not depend on the initial state, and we set

$$\rho^*(M) := \rho^*(M, s) := \max_{\pi} \rho(M, \pi, s).$$

## 3 Objective

We do not consider a separate learning phase and are interested in the total accumulated reward, also during learning. The optimal average reward is the natural benchmark for a learning algorithm  $\mathfrak{A}$ , and we define the **total regret** of  $\mathfrak{A}$  after  $T$  steps as

$$\Delta(M, \mathfrak{A}, s, T) := T\rho^*(M) - R(M, \mathfrak{A}, s, T).$$

We assume an unknown MDP  $M$  to be learned, with  $S := |\mathcal{S}|$  states, and  $A := |\mathcal{A}|$  actions. Only  $\mathcal{S}$  and  $\mathcal{A}$  are known to the learner. Our objective is to minimize the regret.

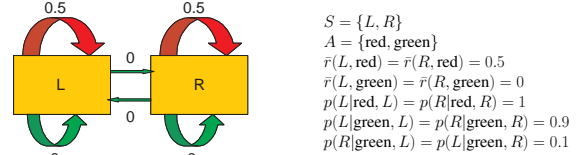
## 4 The UCRL2 Algorithm

### The Basic Idea: Optimistic Estimates

Our algorithm is a variation of the UCRL algorithm of Auer and Ortner [1]. As its predecessor it implements the paradigm of “optimism in the face of uncertainty”. Based on the observations made by the algorithm, we construct a

set  $\mathcal{M}$  of **plausible** MDPs which are statistically consistent with these observations. From this set  $\mathcal{M}$ , UCRL2 optimistically chooses an MDP  $\tilde{M}$  with (nearly) largest optimal average reward among all MDPs in  $\mathcal{M}$ , and executes a policy  $\tilde{\pi}$  which is (nearly) optimal on  $\tilde{M}$ .

### Example MDP.



**Figure 1.** A simple MDP with two states ( $L$  and  $R$ ) and two actions (red and green) per state. Numbers are expected rewards for actions, arrows indicate possible transitions where larger arrows mean higher transition probability.

### Proceeding in Episodes

Figure 1 shows an MDP where frequent policy switching yields high regret. To avoid such effects UCRL2 proceeds in episodes and computes a new policy only when the number of observations of some state-action pair has doubled.

### Estimates and Confidence Intervals

UCRL2 (see Figure 2) calculates estimates of mean rewards  $\hat{r}(s, a)$  and transition probabilities  $\hat{p}(s'|s, a)$  only at the beginning of a new episode. Let

- $N_k(s, a)$  ... the number of steps in which action  $a$  was chosen in state  $s$ ,
- $R_k(s, a)$  ... the accumulated reward when action  $a$  was chosen in state  $s$ ,
- $P_k(s, a, s')$  ... the number of transitions to state  $s'$  when  $a$  was chosen in  $s$ ,

before episode  $k$  starts, which gives estimates

$$\hat{r}_k(s, a) := \frac{R_k(s, a)}{\max\{1, N_k(s, a)\}}, \quad \text{and} \quad \hat{p}_k(s'|s, a) := \frac{P_k(s, a, s')}{\max\{1, N_k(s, a)\}}. \quad (1)$$

Since these estimates will deviate from the true values, we use **optimistic estimates** to determine a policy. Optimistic estimates are calculated from **confidence intervals** for  $\hat{r}_k(s, a)$  and  $\hat{p}_k(s'|s, a)$ .

**Input:** A confidence parameter  $\delta \in (0, 1]$ .

**Initialization:** Set  $t := 1$ , and observe the initial state  $s_1$ .

**For** episodes  $k = 1, 2, \dots$  **do**

**Initialize episode**  $k$ :

1. Set the start time of episode  $k$ ,  $t_k := t$ .
2. Initialize the visit counts for episode  $k$ ,  $v_k(s, a) = 0$ .
3. Compute estimates for  $\hat{r}_k(s, a)$  and  $\hat{p}_k(s'|s, a)$  according to (1).

**Compute policy**  $\tilde{\pi}_k$ :

4. Let  $\mathcal{M}_k$  be the set of all MDPs with states and actions as in  $M$ , and with transition probabilities  $\tilde{p}(\cdot|s, a)$  close to  $\hat{p}_k(\cdot|s, a)$ , and rewards  $\tilde{r}(s, a) \in [0, 1]$  close to  $\hat{r}_k(s, a)$ , that is,

$$\left| \tilde{r}(s, a) - \hat{r}_k(s, a) \right| \leq \sqrt{\frac{3 \log(2SA_t/\delta)}{\max\{1, N_k(s, a)\}}} \approx O\left(\sqrt{\frac{\log(t_k)}{N_k(s, a)}}\right), \quad (4)$$

$$\left\| \tilde{p}(\cdot|s, a) - \hat{p}_k(\cdot|s, a) \right\|_1 \leq \sqrt{\frac{12S \log(2At_k/\delta)}{\max\{1, N_k(s, a)\}}} \approx O\left(\sqrt{\frac{\log(t_k)}{N_k(s, a)}}\right). \quad (5)$$

5. Set  $\epsilon_k := 1/\sqrt{t_k}$  and use extended value iteration (see below) to find a policy  $\tilde{\pi}_k$  and an optimistic MDP  $\tilde{M}_k \in \mathcal{M}_k$  such that for all  $s \in \mathcal{S}$ :

$$\rho(\tilde{M}_k, \tilde{\pi}_k, s) \geq \max_{M' \in \mathcal{M}_k} \rho^*(M') - \epsilon_k.$$

**Execute policy**  $\tilde{\pi}_k$ :

6. **While**  $v_k(s_t, \tilde{\pi}_k(s_t)) < \max\{1, N_k(s_t, \tilde{\pi}_k(s_t))\}$  **do**
  - (a) Choose action  $a_t := \tilde{\pi}_k(s_t)$ , obtain reward  $r_t$ , and observe next state  $s_{t+1}$ .
  - (b) Update  $v_k(s_t, a_t) := v_k(s_t, a_t) + 1$ .
  - (c) Set  $t := t + 1$ .

**Figure 2.** The UCRL2 algorithm.

## Extended Value Iteration

The rewards  $\tilde{r}_k(s, a)$  in  $\tilde{M}_k$  can simply be set to the maximal admissible value as given in (4). Thus, to identify  $\tilde{M}_k$ , one has to maximize over the admissible probability distributions (according to (5)) for each action in addition to maximizing over the actions, as in standard value iteration. It is sufficient to consider the vertices of the convex polytope specified by the intersection of the respective confidence bounds and the probability simplex, since the function maximized is linear. Denote this set of probability distributions by  $\mathcal{P}(s, a)$  for state-action pair  $(s, a)$ . With  $u_0(s) := 0$ , the iterated state values are

$$u_{i+1}(s) := \max_{a \in \mathcal{A}} \left\{ \tilde{r}_k(s, a) + \max_{p \in \mathcal{P}(s, a)} \left\{ \sum_{s' \in \mathcal{S}} p(s') u_i(s') \right\} \right\} \quad \forall s \in \mathcal{S}. \quad (6)$$

The iteration is done until  $\max_s \{u_{i+1}(s) - u_i(s)\} - \min_s \{u_{i+1}(s) - u_i(s)\} \leq \varepsilon_k$ . Then the maximizing choice of actions and probability distributions in (6) can be shown to identify an  $\varepsilon_k$ -optimal MDP-policy pair  $(\tilde{M}_k, \tilde{\pi}_k)$ . Further, policy  $\tilde{\pi}_k$  has state independent average reward  $\hat{\rho}_k$ .

## 5 The Diameter of an MDP

The difficulty of learning an MDP does not only depend on its size, but also on its internal structure. In order to measure this internal structure we propose a new parameter, the **diameter**  $D$  of the MDP.

**Definition 1.** The diameter  $D$  is the minimal average time it takes to move from any state  $s_1$  to any other state  $s_2$ , using an appropriate policy.

More formally, let  $T(s_2|M, \pi, s_1)$  be the first (random) time step in which state  $s_2$  is reached when policy  $\pi$  is executed on MDP  $M$  with initial state  $s_1$ . Then the diameter of  $M$  is given by

$$D(M) := \max_{s_1, s_2 \in \mathcal{S}} \min_{\pi: \mathcal{S} \rightarrow \mathcal{A}} \mathbb{E}[T(s_2|M, \pi, s_1)].$$

A finite diameter seems necessary for learning an optimal policy, since otherwise some parts of the MDP are not reachable. Moreover, recovering from being taken to a “bad part” of the MDP while exploring some actions, may take  $D$  steps. Thus it is necessary that the diameter (or a related quantity) appears in the regret bound.

## 6 Results

### High probability bound:

**Theorem 2.** For any initial state  $s \in \mathcal{S}$  and any  $T \geq 1$ , with probability  $1 - \delta/T$  the regret of UCRL2 is bounded by

$$\Delta(M, \text{UCRL2}, s, T) \leq 49 \cdot DS \sqrt{TA \log \frac{T}{\delta}}.$$

### PAC-like bound:

**Corollary 3.** With probability  $1 - \delta$ , after

$$T \geq 4 \cdot \frac{49^2 D^2 S^2 A}{\varepsilon^2} \log \left( \frac{49 D S A}{\delta \varepsilon} \right)$$

steps, the average per-step regret is at most  $\varepsilon$ .

### Logarithmic bound on the expected regret:

**Theorem 4.** For any initial state  $s \in \mathcal{S}$ , the expected regret of UCRL2 (with parameter  $\delta := 1/T^2$ ) is

$$\mathbb{E}[\Delta(M, \text{UCRL2}, s, T)] = O \left( \frac{D^2 S^2 A \log(T)}{g} \right),$$

where  $g$  is the gap between the optimal average reward and the second largest average reward achievable in  $M$ ,  $g := \rho^*(M) - \max_{\pi, s} \{\rho(M, \pi, s) : \rho(M, \pi, s) < \rho^*(M)\}$ . Using the doubling trick to set the parameter  $\delta$ , the same bound can be achieved without knowledge of the horizon  $T$ .

### Accompanying lower bound:

**Theorem 5.** For any algorithm  $\mathfrak{A}$  and any natural numbers  $T, S, A > 1$ , and  $D \geq \log_A S$  there is an MDP  $M$  with  $S$  states,  $A$  actions, and diameter  $D$ , such that for any initial state  $s \in \mathcal{S}$  the expected regret of  $\mathfrak{A}$  after  $T$  steps is

$$\mathbb{E}[\Delta(M, \mathfrak{A}, s, T)] = \Omega \left( \sqrt{D S A T} \right).$$

### Changing MDPs:

**Theorem 6.** Assume that the MDP (i.e. its transition probabilities and reward distributions) is allowed to change  $k$  times up to step  $T$ , such that the diameter is always at most  $D$ . Restarting UCRL2 with parameter  $\delta/k^2$  at steps  $\lceil i^3/k^2 \rceil$  for  $i = 1, 2, 3, \dots$ , the regret is upper bounded with probability  $1 - 2\delta$  by

$$77 \frac{1}{k^3} T^{\frac{2}{3}} D S \sqrt{A \log \frac{kT}{\delta}}.$$

## 7 Discussion

Typical reinforcement learning algorithms have a distinct learning phase with some guarantee to converge to an optimal policy. Some of these algorithms could be converted to online reinforcement learning algorithms, but the regret bounds would be worse than ours. We discuss some of the differences between our algorithm and other reinforcement learning algorithms, and the results obtained.

### Comparison to $E^3$ and R-Max [2, 3, 4]

**PAC- and online regret bounds:**  $E^3$  and R-Max take as input a confidence parameter  $\delta$  and an accuracy parameter  $\varepsilon$ . These algorithms achieve  $\varepsilon$ -optimal average reward with probability  $1 - \delta$  after time polynomial in  $\frac{1}{\varepsilon}, \frac{1}{\delta}, S, A$ , and the mixing time  $T_\varepsilon^{\text{mix}}$  (see below). As this polynomial dependence on  $\varepsilon$  is of order  $1/\varepsilon^3$ , the PAC bounds translate into  $T^{2/3}$  regret bounds at the best. Also, the exponents of the parameters  $S$  and  $A$  in the PAC bounds of [2, 3] are substantially larger than in our bound.

**Use of mixing time parameters:** Both algorithms need the  $\varepsilon$ -return mixing time  $T_\varepsilon^{\text{mix}}$  of an optimal policy  $\pi^*$  as input parameter. This parameter  $T_\varepsilon^{\text{mix}}$  is the number of steps until the average reward of  $\pi^*$  over these  $T_\varepsilon^{\text{mix}}$  steps is  $\varepsilon$ -close to the optimal average reward  $\rho^*$ . It is easy to construct MDPs of diameter  $D$  with  $T_\varepsilon^{\text{mix}} \approx D/\varepsilon$ . This additional dependency on  $\varepsilon$  further increases the above mentioned regret bounds for  $E^3$  and R-max. Further, the exponents  $T_\varepsilon^{\text{mix}}$  in the PAC bounds of [2, 3, 4] are larger than the exponent of  $D$  in our bound.

### Other Precursors in the Literature

**The UCRL algorithm of Auer and Ortner [1]** is the basis our new work builds on. There are only minor adaptations to the algorithm, but our new analysis improves the regret bounds in several respects:

- Instead of assuming an **ergodic** MDP, where **any** policy will visit every state, we only make the much weaker and more natural assumption of a finite diameter.
- The mixing time is replaced by the newly introduced diameter.
- The exponents of  $S$  and  $g$  have been decreased.
- A high probability and a PAC-like bound are provided.

**The MBIE algorithm of Strehl and Littman [5, 6]** has guaranteed polylogarithmic **average loss**. **Average loss** is defined as the difference between the rewards of an optimal policy and the rewards of the learning algorithm **along the trajectory taken by the learning algorithm**. In contrast, we are interested in the regret of the learning algorithm in respect to the rewards of the optimal policy **along the trajectory of the optimal policy**.

The **OLP algorithm of Tewari and Bartlett** [7] is a generalization of the **index policies** of Burnetas and Katehakis [8]. These index policies choose actions optimistically by using confidence bounds only for the estimates in the current state. The regret bounds for the **index policies** of [8] and the OLP algorithm of [7] are **asymptotically** logarithmic in  $T$ . However, unlike our bounds those bounds also depend on the gap between the “quality” of the best and the second best action, and hide an additive term exponential in the number of states. The corresponding bound in Theorem 4 for UCRL2 holds uniformly over time and under weaker assumptions: While [7] and [8] consider ergodic MDPs exclusively, we only assume a finite diameter.

**Changing MDPs:** MDPs with changing rewards have been considered in [9], where (best possible) upper bounds of  $O(\sqrt{T})$  are derived. Unlike in the setting considered in Theorem 6, in [9] the learner is assumed to know the transition probabilities which do not change, while the rewards are allowed to change at every step.

## 8 Proof Sketch for Theorem 2

The main source of regret is the amount by which the average reward of the optimistically chosen policy is overestimated. To see this, we first consider a single episode, assuming that all confidence intervals hold. The contribution to the regret from episodes where  $M \notin \mathcal{M}_k$  is small. (We sloppily do not mention that statements only hold with high probability.) Summing over the episodes gives the bound claimed.

### 8.1 The Regret of a Single Episode

Consider a single episode  $k$  with its optimistic policy  $\tilde{\pi}_k$  which has (almost) optimal average reward  $\hat{\rho}_k$  on the optimistically assumed MDP  $\tilde{M}_k$ , that is

$$\hat{\rho}_k \geq \rho^* - \varepsilon_k.$$

Thus the regret  $\Delta_k$  of our algorithm during episode  $k$  is bounded as

$$\Delta_k := \sum_{t=t_k}^{t_{k+1}-1} (\rho^* - r_t) \leq \sum_{t=t_k}^{t_{k+1}-1} (\hat{\rho}_k - r_t) + \sum_{t=t_k}^{t_{k+1}-1} \varepsilon_k,$$

where  $r_t$  is the reward obtained by UCRL2 in step  $t$ . The sum of  $\varepsilon_k$ 's is small and will not be considered any further in this proof sketch. Thus we need to bound the difference between the optimistic average reward  $\hat{\rho}_k$  and the actual rewards  $r_t$ .

#### Obtained rewards:

Ignoring the random fluctuation of the rewards (which is also relatively small and can be bounded using Chernoff bounds), the sum of rewards obtained can be written using visit counts  $v_k$  and mean rewards  $\bar{r}$ ,

$$\sum_{t=t_k}^{t_{k+1}-1} r_t \approx \sum_{(s,a)} v_k(s,a) \bar{r}(s,a).$$

Replacing the true mean rewards  $\bar{r}(s,a)$  by their optimistic estimates  $\tilde{r}_k(s,a)$ , we get

$$\Delta_k \approx \sum_{(s,a)} v_k(s,a) (\hat{\rho}_k - \tilde{r}_k(s,a)) + \sum_{(s,a)} v_k(s,a) (\tilde{r}_k(s,a) - \bar{r}(s,a)). \quad (7)$$

The second term in (7) is bounded by the widths of the confidence intervals given in (4), thus its contribution to the regret is small. Bounding the first term is the crucial part of the proof.

### 8.2 The Poisson Equation

Let  $\tilde{P}_k := (\tilde{p}_k(s'|s, a_s))_{s,s'}$  be the transition matrix of  $\tilde{\pi}_k$  on  $\tilde{M}_k$ , and  $\tilde{r}_k := (\tilde{r}_k(s, \tilde{\pi}_k(s)))_s$  be the (column) vector of the optimistic mean rewards. Then the Poisson equation for the optimistic MDP  $\tilde{M}_k$  and the policy  $\tilde{\pi}_k$

$$\tilde{\lambda}_k = \tilde{r}_k - \tilde{\rho}_k \mathbf{1} + \tilde{P}_k \tilde{\lambda}_k \quad (8)$$

defines the **bias**  $\tilde{\lambda}_k(s)$  of state  $s$ : While in the long run the average per-step rewards are equal for starting in any state, the **advantage** (difference in bias) measures the difference of the total sums of rewards.

#### Bounded advantage:

For the chosen policy  $\tilde{\pi}_k$  on  $\tilde{M}_k$ , **any advantage is bounded by the diameter**:

$$\tilde{\lambda}_k(s) - \tilde{\lambda}_k(s') \leq D.$$

Otherwise, an improved policy that moves from  $s'$  to  $s$  in at most  $D$  steps can be shown to earn more than  $\hat{\rho}_k$  on average, which contradicts the optimality of  $\tilde{\pi}_k$ . Since the Poisson equation (8) holds also for a translated bias vector  $z_k = \tilde{\lambda}_k + c\mathbf{1}$  we can choose  $z_k$  with  $\|z_k\|_\infty \leq D/2$ .

### 8.3 The Main Regret Term

Let  $v_k := (v_k(s, \tilde{\pi}_k(s)))_s$  be the (row) vector of visit counts for each state and the corresponding action chosen by  $\tilde{\pi}_k$ . Plugging the Poisson equation into the first term on the right hand side of (7) we get for a suitable vector  $z_k$  with  $\|z_k\|_\infty \leq D/2$

$$\sum_{(s,a)} v_k(s,a) (\hat{\rho}_k - \tilde{r}_k(s,a)) = v_k (\tilde{P}_k - \mathbf{I}) z_k. \quad (9)$$

Now we expand

$$v_k (\tilde{P}_k - \mathbf{I}) z_k = v_k (\tilde{P}_k - P_k) z_k + v_k (P_k - \mathbf{I}) z_k,$$

where  $P_k$  is the transition matrix of the policy  $\tilde{\pi}_k$  in the **true** MDP. The first term can be bounded using the confidence intervals in (5), yielding the dominant regret term:

$$\begin{aligned} v_k (\tilde{P}_k - P_k) z_k &\leq \|v_k (\tilde{P}_k - P_k)\|_1 \|z_k\|_\infty \\ &\leq \sum_s v_k(s, \tilde{\pi}_k(s)) \cdot 2 \cdot \sqrt{\frac{12S \log(2AT/\delta)}{\max\{1, N_k(s,a)\}}} \cdot \frac{D}{2}. \end{aligned} \quad (10)$$

For the second term, consider the distribution over the states corresponding to  $v_k$ . Then  $v_k P_k$  corresponds to the distribution after making one additional step according to  $P_k$ . As  $v_k$  is the vector of state visits under transition matrix  $P_k$ , this one step can be shown not to make too much difference, i.e.  $v_k P_k - v_k$  is “small”. To turn these intuitions into a concrete bound, one has to sum up over all episodes to find that

$$\sum_k v_k (P_k - \mathbf{I}) z_k \leq D \sqrt{2T \log \frac{T}{\delta}} + D \cdot \#(\text{episodes}).$$

As the number of episodes is logarithmic in  $T$  this term is negligible compared to the main term.

#### Putting things together:

Summing up (7), (9), and (10) we find that

$$\Delta_k \approx \text{const} \cdot D \sqrt{S \log(T/\delta)} \sum_{(s,a)} v_k(s,a) \sqrt{\frac{1}{N_k(s,a)}}.$$

Summing over all episodes gives, since  $v_k(s,a) = N_{k+1}(s,a) - N_k(s,a)$  and setting  $N(s,a) := \sum_k v_k(s,a)$

$$\begin{aligned} \sum_k \Delta_k &\leq \text{const} \cdot D \sqrt{S \log(T/\delta)} \cdot \sum_k \sum_{(s,a)} \frac{v_k(s,a)}{\sqrt{N_k(s,a)}} \\ &\leq \text{const} \cdot D \sqrt{S \log(T/\delta)} \cdot \sum_{(s,a)} \sqrt{N(s,a)} \\ &\leq \text{const} \cdot D \sqrt{S \log(T/\delta)} \cdot \sqrt{AST}. \end{aligned}$$

#### References

- [1] Peter Auer and Ronald Ortner. Logarithmic online regret bounds for reinforcement learning. In *Proc. 19th NIPS*, pages 49–56. MIT Press, 2006.
- [2] Michael J. Kearns and Satinder P. Singh. Near-optimal reinforcement learning in polynomial time. *Mach. Learn.*, 49:209–232, 2002.
- [3] Ronen I. Brafman and Moshe Tennenholtz. R-max – a general polynomial time algorithm for near-optimal reinforcement learning. *J. Mach. Learn. Res.*, 3:213–231, 2002.
- [4] Sham M. Kakade. *On the Sample Complexity of Reinforcement Learning*. PhD thesis, University College London, 2003.
- [5] Alexander L. Strehl and Michael L. Littman. An empirical evaluation of interval estimation for Markov decision processes. In *Proc. 16th ICTAI*, pages 128–135. IEEE Computer Society, 2004.
- [6] Alexander L. Strehl and Michael L. Littman. A theoretical analysis of model-based interval estimation. In *Proc. 22nd ICML 2005*, pages 857–864, 2005.
- [7] Ambuj Tewari and Peter Bartlett. Optimistic linear programming gives logarithmic regret for irreducible mdp. In J.C. Platt, D. Koller, Y. Singer, and S. Roweis, editors, *Advances in Neural Information Processing Systems 20*, pages 1505–1512. MIT Press, Cambridge, MA, 2008.
- [8] Apostolos N. Burnetas and Michael N. Katehakis. Optimal adaptive policies for Markov decision processes. *Math. Oper. Res.*, 22(1):222–255, 1997.
- [9] Eyal Even-Dar, Sham M. Kakade, and Yishay Mansour. Experts in a Markov decision process. In *Proc. 17th NIPS*, pages 401–408. MIT Press, 2004.