Optimism in Reinforcement Learning Based on Kullback-Leibler Divergence

Sarah Filippi, Olivier Cappé, Aurélien Garivier

Optimism in Reinforcement Learning Based on Kullback-Leibler Divergence

Sarah Filippi, Olivier Cappé, and Aurélien Garivier *

LTCI, TELECOM ParisTech and CNRS
46 rue Barrault, 75013 Paris, France
(filippi,cappe,garivier)@telecom-paristech.fr,

Abstract. We consider model-based reinforcement learning in finite Markov Decision Processes (MDPs), focussing on so-called optimistic strategies. Optimism is usually implemented by carrying out extended value iterations, under a constraint of consistency with the estimated model transition probabilities. In this paper, we strongly argue in favor of using the Kullback-Leibler (KL) divergence for this purpose. By studying the linear maximization problem under KL constraints, we provide an efficient algorithm for solving KL-optimistic extended value iteration. When implemented within the structure of UCRL2, the near-optimal method introduced by [2], this algorithm also achieves bounded regrets in the undiscounted case. We however provide some geometric arguments as well as a concrete illustration on a simulated example to explain the observed improved practical behavior, particularly when the MDP has reduced connectivity. To analyze this new algorithm, termed KL-UCRL, we also rely on recent deviation bounds for the KL divergence which compare favorably with the L_1 deviation bounds used in previous works.

Key words: Reinforcement learning; Markov decision processes; Model-based approaches; Optimistim; Kullback-Leibler divergence; Regret bounds

1 Introduction

In reinforcement learning, an agent interacts with an unknown environment aiming to maximize its long-term payoff [15]. This interaction is commonly modelled by a Markov Decision Process (MDP) and it is assumed that the agent does not know the parameters of this process but has to learn how to act directly from experience. The agent thus faces a fundamental trade-off between gathering experimental data about the consequences of the actions (exploration) and acting consistently with past experience to maximize rewards (exploitation).

We consider in this article a MDP with finite state and action spaces for which we propose a *model-based* reinforcement learning algorithm, i.e., an algorithm that is based on a running estimate of the model parameters (transitions probabilities and expected rewards)[6,8,11,16]. A well-known approach to balance exploration and exploitation in model-based algorithms is the so-called

* This publication is partially supported by Orange Labs under contract n°289365.

optimism in the face of uncertainty principle, first proposed in the multi-armed bandit context by [12]: instead of acting optimally according to the estimated model, the agent follows the optimal policy for a model, named *optimistic model*, which is close enough to the latter to make the observations sufficiently likely, but which leads to a higher long-term reward. The performance of such an algorithm can be analysed in term of *regret* which consists in comparing the rewards collected by the algorithm with the rewards obtained always following an optimal policy. The study of the asymptotic regret due to [12] in the multi-armed context has been extended to MDPs by [7], proving that an optimistic algorithm can obtain logarithmic regret. The subsequent works of [3,2,4] introduced algorithms that guarantee non-asymptotic logarithmic regret, in a large class of MDPs –see further discussion of relevant assumptions below. In these works, the optimistic model is computed using the L1 (or total variation) norm as a measure of proximity between the estimated and optimistic transition probabilities.

In addition to the efficiency guarantees, the authors of [2] underline that the optimistic modifications of the estimates used in their algorithm (called UCRL2) has a simple interpretation. Indeed, the computing of the optimistic MDP from the estimated transition probabilities consists, for each state, in adding a bonus to the more promising transition (i.e. the transition that leads to a state with highest value), and in removing this probability to the less promising ones. These operations can be done very efficiently and the computational complexity of the resulting procedure is remarkably low. Besides, such interpretability properties are important in applications, as the explorational aspect of the policy is explicit and as the agent's decisions are founded in interpretable hopes.

Such behaviour has however some undesirable side-effects. First, due to the non-smoothness of the L1-neighboorhoods, the optimistic model is not continuous with respect to the estimated parameters – small changes in the estimates may result in very different optimistic models. More importantly, the optimistic model may give a zero probability to a transition that has actually been observed, which makes it hardly compatible with the optimism principle. Besides, the optimistic model always includes non-zero transitions from all states to the most promizing one, even if much evidence has been accumulated against the existence of such a transition, and even if the most promizing state is expected to be hardly better than others. This appears to be unsatisfactory in practice, in particular when the ground-truth MDP has reduced connectivity, (i.e. when each individual state may only lead to a limited set of successor states).

Based on these observations, we describe a novel algorithm, termed *KL-UCRL*, that overcomes with those shortcomings while capitalizing on the structure of UCRL2 to ensure a logarithmic regret bound. To do so, we propose to re-introduce the use of the Kullback-Leibler (KL) pseudo-distance rather than L1 metric, as in [7]. We will show that the resulting algorithm has the following properties:

– the KL-optimistic model, which is continuous with respect to the expected reward function, always gives strictly positive probability mass to observed transitions;

- for every unobserved transition from a state x to a state y, a trade-off between the relative attractivity of y and the statistical evidence accumulated in x is computed to decided whether it should have positive probability or not in the optimistic model;
- it is based on novel concentration inequalities for the KL-divergence which compares favorably with the L1-norm bounds used in the aforementioned works, as the KL-divergence is the pseudo-metric on the simplex induced by the theory of large deviation;
- the linear maximization problem under KL constraints can be done very efficiently, using an algorithm based on one-dimensional line searches described below;
- the analysis of [2,4] can be easily adapted: similar non-asymptotic regret bounds under weak hypotheses can be derived, knowledge about the MDP's underlying state structure is not required;
- simulations show a significative improvement in practice.

The paper is organized as follows. The model and a brief survey of needed results on MDPs are presented in Section 2. Section 3 is devoted to the description of the KL-UCRL algorithm together with an explicit method to compute the maximization over a KL-ball. Section 4 contains the regret bounds of the KL-UCRL algorithm, with corresponding proofs in the Appendix. Section 5 provides some results from simulations on a classic benchmark example (the *river swim* environment of [14]). In Section 6, the advantages of using a KL- rather that L1-confidence balls are illustrated and discussed.

2 Markov Decision Process

Consider a Markov decision process (MDP) $\mathbf{M} = (\mathcal{X}, \mathcal{A}, P, r)$ with finite state space $\mathcal{X}$, and action space $\mathcal{A}$. Let $X_t \in \mathcal{X}$ and $A_t \in \mathcal{A}$ denote respectively the state of the system and the action chosen by the agent at time t. Once the action is executed, the sytem transits from state X_t to state X_{t+1} with probability $P(X_{t+1}; X_t, A_t)$. At the same time, the agent receives a random reward $R_t \in [0, 1]$ with mean $r(X_t, A_t)$. The aim of the agent is to choose the sequence of actions so as to maximize the cumulated reward. To select an action, the agent follows a *stationary policy* $\pi : \mathcal{X} \to \mathcal{A}$.

In this paper, we consider *weakly communicating* MDPs, i.e., MDPs satisfying the *weak accesssibility* conditions [5]: the set of states can be partitioned into two subsets $\mathcal{X}_t$ and $\mathcal{X}_c$ such that all states in $\mathcal{X}_t$ are transient under every stationary policy and, for any states $x, x' \in \mathcal{X}_c$, there exists a policy $\pi_{x,x'}$ that takes one from x to x'. For those MDPs, it is known that the *average reward* following a stationary policy π, denoted by $\rho^\pi(\mathbf{M})$ and defined as

$$\rho^\pi(\mathbf{M}) = \lim_{n\to\infty} \frac{1}{n} \mathrm{E}\left[\sum_{t=0}^{n} R_t \,\middle|\, \pi, \mathbf{M} \right] ,$$

is state independent [13]. Let $\pi^*(\mathbf{M}) : \mathcal{X} \to \mathcal{A}$ and $\rho^*(\mathbf{M})$ denote respectively the optimal policy and the optimal average reward:

$$\rho^*(\mathbf{M}) = \sup_{\pi} \rho^{\pi}(\mathbf{M}) = \rho^{\pi^*(\mathbf{M})}(\mathbf{M}) \; .$$

The notation $\rho^*(\mathbf{M})$ and $\pi^*(\mathbf{M})$ are meant to highlight the fact that both the optimal average reward and the optimal policy depends on the model $\mathbf{M}$. The optimal average reward satisfies the so-called *optimality equation*

$$\forall x \in \mathcal{X} \; , \qquad h^*(\mathbf{M}, x) + \rho^*(\mathbf{M}) = \max_{a \in \mathcal{A}} \left(r(x,a) + \sum_{x' \in \mathcal{X}} P(x'; x, a) h^*(\mathbf{M}, x') \right) \; .$$

where the $|\mathcal{X}|$-dimensional vector $h^*(\mathbf{M})$ is called a bias vector. Note that an infinity of bias vector satisfies this equation: if h^* is a bias vector then, for any constant c, $h^* + c\ \mathbf{e}$ is also a bias vector where $\mathbf{e}$ is the all 1's vector. For a fixed MDP $\mathbf{M}$, the optimal policy $\pi^*(\mathbf{M})$ can be specified solving the optimality equation and defining, for all $x \in \mathcal{X}$,

$$\pi^*(\mathbf{M}, x) \in \operatorname*{argmax}_{a \in \mathcal{A}} \left(r(x,a) + \sum_{x' \in \mathcal{X}} P(x'; x, a) h^*(\mathbf{M}, x) \right) \; .$$

In practice, the optimal average reward and the optimal policy may be computed using the value iteration algorithm [13].

3 The KL-UCRL algorithm

In this paper, we focus on the reinforcement learning problem in which the agent does not know the model $\mathbf{M}$ beforehand, i.e. the transition probabilities and the distribution of the rewards are unknown. More specifically, we consider a model-based reinforcement learning method which consists in learning the model throughout the experiment and acting according to it. Denote by $\hat{P}_t(x'; x, a)$ the estimate at time t of the transition probability from state x to state x' conditionnally to the action a and by $\hat{r}_t(x, a)$ the mean reward received in state x when action a has been choosen. We have :

$$\hat{P}_t(x'; x, a) = \frac{N_t(x, a, x')}{\max(N_t(x, a), 1)} \qquad \text{and} \qquad \hat{r}_t(x, a) = \frac{\sum_{k=1}^{t-1} R_k \mathbb{1}_{\{X_k = x, A_k = a\}}}{\max(N_t(x, a), 1)} \; , \tag{1}$$

where $N_t(x, a, x') = \sum_{k=0}^{t-1} \mathbb{1}_{\{X_k = x, A_k = a, X_{k+1} = x'\}}$ is the number of visits, up to time t, to the state x followed by a visit to x' if the action a has been choosed, and similarly, $N_t(x, a) = \sum_{k=0}^{t-1} \mathbb{1}_{\{X_k = x, A_k = a\}}$. The optimal policy associated with the estimated model $\hat{\mathbf{M}}_t = (\mathcal{X}, \mathcal{A}, \hat{P}_t, \hat{r}_t)$ may be misleading due to estimation errors: pure exploitation policies are commonly known to fail with positive probability. To avoid this problem, *optimistic model-based approaches* consist in computing

a set $\mathcal{M}_t$ of potential MDPs including $\hat{\mathbf{M}}_t$ and choosing the MDP in this set leading to the largest average reward. The set $\mathcal{M}_t$ is defined as follows:

$$\mathcal{M}_t = \{\mathbf{M} = (\mathcal{X}, \mathcal{A}, P, r) \ , \ \forall x \in \mathcal{X}, \forall a \in \mathcal{A}, \ \ |r_t(x,a) - r(x,a)| \leq C_R \\ \text{and } d(\hat{P}_t(.;x,a), P(.;x,a)) \leq C_P\} \ ,$$

where C_P and C_R are fixed constant and d measures the difference between the transition probabilities.

In contrast to UCRL2, which uses the L1-distance, we propose to rely on the Kullback-Leibler divergence as the seminal article [7]. Contrary to the approach of [7], no prior knowledge on the state structure of the MDP is needed. Recall that the Kullback-Leibler divergence is defined for all n-dimensional probability vectors p and q by $KL(p,q) = \sum_{i=1}^{n} p_i \log \frac{p_i}{q_i}$ (with the convention that $0 \log 0 = 0$). In Section 6, the advantages of using a KL-divergence instead of the L1-norm are illustrated and argumented.

3.1 The KL-UCRL algorithm

The KL-UCRL, described below, is a variant of the efficient model-based algorithm UCRL2, introduced by [2] and extended to more general MDPs by [4]. The structure of KL-UCRL, which is common to UCRL2, reflects the "optimism under uncertainty" principle and is given below for self-containment. The key step of the algorithm is the seek of the optimistic model on line 12. It is detailed below and relies on Procedure 2. The KL-UCRL algorithm proceeds in episodes.

Algorithm 1 KL-UCRL

1: Initialization: let $j = 0$ and $\forall a \in \mathcal{A}, \forall x \in \mathcal{X}, n_j(x,a) = 0$, a random policy π_0.
2: **for** all $t \geq 1$ **do**
3: Observe X_t.
4: **if** $n_j(X_t, \pi_j(X_t)) < \max(N_t(X_t, \pi_j(X_t)), 1)$ **then**
5: *Continue the same policy:*
6: Choose action $A_t = \pi_j(X_t)$ and receive reward R_t.
7: Update the count: $n_j(X_t, A_t) = n_j(X_t, A_t) + 1$.
8: **else**
9: *Begin a new episode:* $j = j + 1$
10: Reinitialize: $\forall a \in \mathcal{A}, \forall x \in \mathcal{X} \ , \ n_j(x,a) = 0$
11: Estimate the model $\hat{\mathbf{M}}_t = (\mathcal{X}, \mathcal{A}, \hat{P}_t, \hat{r}_t)$ according to (1).
12: Find the optimistic model $\mathbf{M}_j \in \mathcal{M}_t$ and the related policy π_j.
13: Choose action $A_t = \pi_j(X_t)$ and receive reward R_t.
14: Update the count: $n_j(X_t, A_t) = n_j(X_t, A_t) + 1$.
15: **end if**
16: **end for**

Let t_j be the starting time of episode j; the length of the j-th episode depends on the number of visits $N_{t_j}(x,a)$ to the state-action pair (x,a) before t_j compared

to the number of visits $n_j(x,a)$ to the same pair during the j-th episode. More precisely, an episode lasts as soon as $n_j(x,a) < N_{t_j}(x,a)$ for a state-action pair (x,a). The policy π_j, followed during the j-th episode, is a near optimal policy related to the optimistic MDP $\mathbf{M}_j = (\mathcal{X}, \mathcal{A}, P_j, r_j) \in \mathcal{M}_{t_j}$:

$$\forall x \in \mathcal{X}\ ,\ \pi_j(x) \in \operatorname*{argmax}_{a\in\mathcal{A}} \left(r_j(x,a) + \sum_{x'\in\mathcal{X}} P_j(x';x,a) h^*(\mathbf{M}_j, x') \right)\ ,$$

where h^* is a bias vector satisfying the *extended optimality equations*:

$$\begin{aligned} \forall x \in \mathcal{X}\ ,\ h^*(x) + \rho^* = \max_{P,r} \max_{a\in\mathcal{A}} \left(r(x,a) + \sum_{x'\in\mathcal{X}} P(x';x,a) h^*(x') \right) \qquad (2) \\ \text{such that } \forall x \in \mathcal{X}, \quad \forall a \in \mathcal{A}, KL(\hat{P}_{t_j}(.;x,a), P(.;x,a)) \le C_P(x,a,t_j) \\ \forall x \in \mathcal{X}, \forall a \in \mathcal{A}, \quad |r_{t_j}(x,a), r(x,a))| \le C_R(x,a,t_j)\ . \end{aligned}$$

Denote by P_j and r_j respectively the transition probability and the mean reward that maximizes those equations. Remark that the diameter C_P (resp. C_R) of the neighborhood around the estimated transition probability $P_{t_j}(.;x,a)$ (resp. the mean reward $\hat{r}_{t_j}(x,a)$) depends on the state action pair (x,a) and on t_j. The *extended value iteration* algorithm may be used to approximately solve the fixed point equation (2) [13,2].

3.2 Maximization of a linear function on a KL-ball

At each step of the extended value iteration algorithm, the maximization problem (2) has to be solved. Remark that, for every action a, the maximization in $r(x,a)$ is obviously solved taking $r(x,a) = \hat{r}_{t_j}(x,a) + C_R(x,a,t_j)$, so that the main difficulty lies in maximizing the dot product between the probability vector $q = P(.;x,a)$ and the bias vector also called *value vector* $V = h^*$ over a KL-ball around the fixed probability vector $p = \hat{P}_{t_j}(.;x,a)$:

$$\max_{q\in\mathbb{S}^{|\mathcal{X}|}} V'q \quad \text{s.t.} \quad KL(p,q) \le \epsilon\ , \qquad (3)$$

where the constant $0 < \epsilon < 1$ controls the size of the confidence ball[1] and $\mathbb{S}^n$ denotes the set of n-dimensional probability vectors. This maximization of a linear function under convex constraints is solved explicitely in Appendix A; the resulting algorithm is shown in 2. It relies on the function f (depending on the parameter V) defined for all $\nu \ge \max_{i\in\bar{Z}} V_i$, where $\bar{Z} = \{i, p_i > 0\}$, by

$$f(\nu) = \sum_{i\in\bar{Z}} p_i \log(\nu - V_i) + \log\left(\sum_{i\in\bar{Z}} \frac{p_i}{\nu - V_i} \right)\ . \qquad (4)$$

[1] The prime is used to denote transposition.

If the estimated transition to a state with a potentially highest value V_{i^*} is equal to 0, there is a dilemma between two possibilities: first, one may improve the dot product by giving a little probability to that transition; second, by conceding that this transition is unlikely one may rather choose to add probability to other transitions. The dilemma's solution involves both V_{i^*}, the other components of V and the exploration bonus: namely, $f(V_{i^*})$ is compared to the diameter of the neighborhood ϵ, and as a result a decision is taken to abandon this transition or not.

Algorithm 2 Function MaxKL

Require: A value function V , a probability vector p, a constant ϵ
Ensure: A probability vector q that maximizes (3)
1: Let $Z = \{i, p_i = 0\}$ and $\bar{Z} = \{i, p_i > 0\}$. Let $I^* = Z \cap \operatorname{argmax}_i V_i$
2: **if** $f(V_i) < \epsilon$ for $i \in I^*$ **then**
3: Let $\nu = V_j$ and $r = 1 - \exp(f(\nu) - \epsilon)$.
4: For all $i \in I^*$, assign values of q_i such that $\sum_{i \in I^*} q_i = r$.
5: For all $i \in Z/I^*$, let $q_i = 0$.
6: **else**
7: For all $i \in Z$, let $q_i = 0$. Let $r = 0$.
8: Find ν solution of the equation $f(\nu) = \epsilon$ using Newton's method.
9: **end if**
10: For all $i \in \bar{Z}$, let $q_i = \frac{\tilde{q}_i}{r + \sum_{i \in \bar{Z}} \tilde{q}_i}$ where $\tilde{q}_i = \frac{p_i}{\nu - V_i}$.

In practice, f being a convex positive decreasing function (see Appendix B), Newton's method can be applied to find ν such that $f(\nu) = \epsilon$ (step 10 of the algorithm), so that numerically solving (3) can be done very efficiently. Indeed, only a few steps of the Newton's algorithm are needed to solve the maximization. Appendix B contains a discussion on the initialization of Newton's algorithm using asymptotic arguments.

4 Regret bounds

To analyse the performance of the KL-UCRL algorithm, we compare the rewards collected following the algorithm with the rewards obtained always following an optimal policy. The so called *regret* of an algorithm after T steps is defined as :

$$\text{Regret}_T = \sum_{t=1}^{T} \rho^*(\mathbf{M}) - R_t \ .$$

We propose to adapt the regret bound analysis for the UCRL2 algorithm to the use of KL-neighborhoods and obtain similar theorems. Let $D(\mathbf{M}) = \max_{x,x'} \min_\pi \mathbb{E}(\tau^{\mathbf{M},\pi}(x,x'))$, where $\tau^{\mathbf{M},\pi}(x,x')$ is the first random time step in which state x' is reached when policy π is followed on MDP $\mathbf{M}$ with initial state x. This constant will appear in the regret bounds. For all communicating

MDPs $\mathbf{M}$, $D(\mathbf{M})$ is finite. Theorem 1 establishes an upper bound on the regret following the KL-UCRL algorithm with

$$C_P(x,a,t,\delta,T) = \frac{|\mathcal{X}|\left(B + \log(B+1/\log(T)) + \frac{\log(B+1/\log(T))}{B} + 1/\log(T)\right)}{\max(N_{t_k}(x,a),1)}$$

where $B = \log\left(\frac{2e|\mathcal{X}|^2|\mathcal{A}|\log(T)}{\delta}\right)$ and

$$C_R(x,a,t,\delta,T) = \sqrt{\frac{\log\left(\frac{4|\mathcal{X}||\mathcal{A}|\log(T)}{\delta}\right)}{1.99\max(N_t(x,a),1)}}\ .$$

Theorem 1. *With probability $1-\delta$, it holds that for a large enough $T>1$, the regret of KL-UCRL is bounded by*

$$\mathit{Regret}_T \leq CD|\mathcal{X}|\sqrt{|\mathcal{A}|T\log(\log(T)/\delta)}\ .$$

for a contant C independent of the model.

It is also possible to prove a logarithmic upper bound of the expected regret. This bound, presented in Theorem 2, depends on the model through another constant $\Delta(\mathbf{M})$ defined as follows

$$\Delta(\mathbf{M}) = \rho^*(\mathbf{M}) - \max_{\pi,\rho^\pi(\mathbf{M})<\rho^*(\mathbf{M})} \rho^\pi(\mathbf{M})\ .$$

Theorem 2. *For a large enough horizon $T>1$, the expected regret of KL-UCRL is bounded by*

$$\mathbb{E}(\mathit{Regret}_T) \leq CD^2\frac{|\mathcal{X}|^2|\mathcal{A}|\log(T)}{\Delta(\mathbf{M})}\ .$$

for a contant C independent of the model.

The proof of Theorem 1 is analogous to the one in [2] or in [4]. Due to the lack of space, we do not describe it in details but focus on the steps of the proof that differ from Theorem 2 in [1]. First, the following proposition enables to ensure that, with high probability, the true model belongs to the set of models $\mathcal{M}_t$ at each time step.

Proposition 1. *For T large enough and $\delta>0$, $\mathbb{P}\left(\forall t\leq T\ ,\ \mathbf{M}\in\mathcal{M}_t\right)\geq 1-2\delta$.*

Proof (of Proposition 1). The proof relies on the two following concentration inequalities due to Garivier and Leonardi [10] and Garivier and Moulines [9]: for all $x\in\mathcal{X}$, $a\in\mathcal{A}$, and any $\epsilon_P>0$, and $\epsilon_R>0$, we have

$$\mathbb{P}\left(\forall t\leq T,\ KL(\hat{P}_t(.;x,a),P(.;x,a)) > \frac{\epsilon_P}{N_t(x,a)}\right) \leq 2e(\epsilon_P\log(T)+|\mathcal{X}|)e^{-\frac{\epsilon_P}{|\mathcal{X}|}} \tag{5}$$

$$\mathbb{P}\left(\forall t \leq T, \quad |\hat{r}_t(x,a) - r(x,a)| \leq \frac{\epsilon_R}{\sqrt{N_t(x,a)}}\right) \leq 4\log(T)e^{-1.99\epsilon_R} .$$

Then, taking $\epsilon_P = N_t(x,a)C_P(x,a,t,\delta,T)$ and $\epsilon_R = \sqrt{N_t(x,a)}C_R(x,a,t,\delta,T)$ and summing over all state-action pairs, we have the result of the Proposition.

To upper-bound the regret, the geometry of the neigborhood around the estimated transition probabilities only plays a role to bound the term

$$\sum_{k=1}^{m(T)} \sum_{x,x' \in \mathcal{X}} n_k(x,\pi_k(x))(P_k(x';x,\pi_k(x)) - P_k(x';x,\pi_k(x)))h_k(x')$$

where P_k and π_k denote respectively the transition probability of the optimistic model and the optimal policy in the k-th episode and $m(T)$ is the number of episodes until time T. Using the Cauchy-Schwartz and Pinsker's inequalities, it is easy to show that it is upper-bounded by

$$\begin{aligned}
&D\sum_{k=1}^{m(T)} \sum_{x,x' \in \mathcal{X}} n_k(x,\pi_k(x))\|P_k(x';x,\pi_k(x)) - P_k(x';x,\pi_k(x))\|_1 \\
&\leq 2D\sum_{k=1}^{m(T)} \sum_{x} n_k(x,\pi_k(x))\sqrt{2\ KL(\hat{P}_{t_k}(.;x,\pi_k(x)); P(.;x,\pi_k(x)))} \\
&\leq 2D\sqrt{2}\sum_{k=1}^{m(T)} \sum_{x} n_k(x,\pi_k(x))\sqrt{C_P(x,\pi_k(x),t_k,\delta,T)} .
\end{aligned}$$

For T large enough, this term dominating the remaining terms in the upper-bound of the regret, there exists a constant C such that

$$\text{Regret}_T \leq CD\log(|\mathcal{X}|)\log(\log(T)/\delta)\sum_{k=1}^{m(T)} \sum_{x} \frac{n_k(x,\pi_k(x))}{\sqrt{N_{t_k}(x,\pi_k(x))}} .$$

And Theorem 1 follows using the fact that $\sum_{k=1}^{m(T)} \sum_x \frac{n_k(x,\pi_k(x))}{\sqrt{N_{t_k}(x,\pi_k(x))}} \leq \sqrt{|\mathcal{X}||\mathcal{A}|T}$ (see Appendix B.1 of [2]). The proof of the Theorem 2 follows from Theorem 1 using the same arguments as in the proof of Theorem 4 in [2].

5 Simulations

To illustrate the behaviour of the algorithm compared to the UCRL2 algorithm of [2], we consider the benchmark *RiverSwim* environment proposed by [14]. It consists of six states. The agent, starting from one of the states near the left side of the row, can either swim left or right. Swimming to the right, against the current of the river, will either leaves the agent in the same state (with a high probability equal to 0.6), transitions the agent to the right (with probability

0.35) or transitions it to the left (see Figure 1). Swimming to the left, with the current, always succeeds. The agent receives a small reward of five units when it reaches the leftmost state and a much larger reward, of ten thousand units, for swimming upstream and reaching the rightmost state. This MDP requires an efficient exploration since starting near the left side, the agent has to reach the right side of the row to learn that it is the states related to the highest reward.

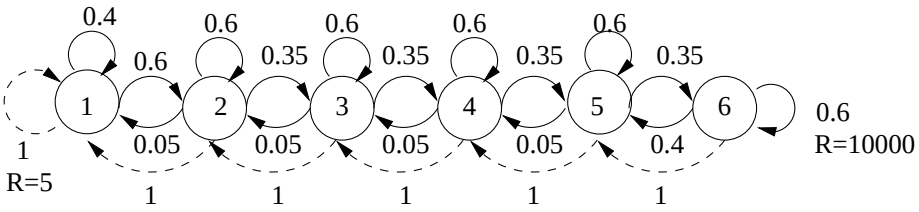

Fig. 1. *RiverSwim* Transition Model: the continuous line (resp. dotted) arrows represent the transitions if action 1 (resp. 2) has been chosen.

We compare the performance of the KL-UCRL algorithm to the UCRL2 algorithm applying them on 20 Monte-Carlo replications. For both algorithms, the constants C_P and C_R are settled to ensure that the upper bounds of the regret of Theorem 1 and Theorem 2 in [1] hold with probability 0.95. We observe in Figure 2 that the KL-UCRL algorithm accomplishes a smaller average regret than the UCRL algorithm. Indeed, in this environment, it is crucial for the agent to quickly learn that there is no possible transition between one of the first four states and the state 6 with the highest reward.

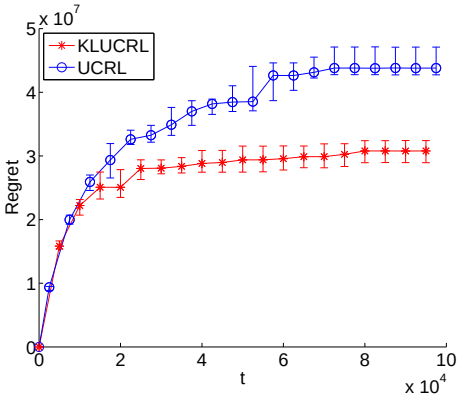

Fig. 2. Comparison of the regret of the UCRL2 and KL-UCRL algorithms on the RiverSwim Environment.

6 Discussion

In this section, we expose the advantages of using a confidence ball based on the Kullback-Leibler divergence around the estimated transition probabilities rather than an L1-ball as proposed in [2,16] in the computation of the optimistic policy (see equation (2)). In this paper, we propose to maximize the linear function $V'q$ over a KL-ball (see (3)) whereas [2,16] use a L1-ball:

$$\max_{q \in \mathbb{S}^{|\mathcal{X}|}} V'q \quad \text{s.t.} \quad \|p - q\|_1 \leq \epsilon' \ . \tag{6}$$

The function $V'q$ being linear in $q \in \mathbb{S}^{|X|}$, the probability that maximizes equations (3) and (6) lies respectively in the border of the smooth convex shape $\{q \in \mathbb{S}^{|X|}, KL(p,q) \leq \epsilon\}$ and in one of the verteces of the polytope $\{q \in \mathbb{S}^{|X|}, \|p-q\|_1 \leq \epsilon'\}$. A first noteworthy difference between those neighborhoods is that, due to the smoothness of the KL-neighborhood, the maximizer over the KL-ball is continuous with respect to the vector V which is not true for the maximization over a L1-ball. The L1 and KL-balls around 3-dimensional probability vectors are displayed in Figure 3. The set of 3-dimensional probability vectors is represented by a triangle whose verteces are the vectors $(1,0,0)'$, $(0,1,0)'$ and $(0,0,1)'$, the probability vector p by a white star and the q vector that maximizes respectively equation (6) and (3) by a white point; the arrow describes the gradient of the value vector V. We observe that the points maximizing equation (6) can vary significantly for small changes of the value function which does not happen when maximizing over a KL-ball. This difference is specially significant when, as the case in the optimistic algorithms, the vector V is not a priori known but computed from estimated transition probabilities.

Consider an estimated transition probability vector p and denote by q the probability vector which maximizes (6). Let $i_m = \text{argmin}_j V_j$ and $i_M = \text{argmax}_j V_j$. As underlined by [2], $q_{i_m} = \max(p_{i_m} - \epsilon'/2, 0)$ and $q_{i_M} = \min(p_{i_M} + \epsilon'/2, 1)$. This has two consequences:

1. if p is such that $0 < p_{i_m} < \epsilon'/2$, then the vector $q_{i_m} = 0$; so the optimistic model may give a zero probability to a transition that has actually been observed, which makes it hardly compatible with the optimism principle: indeed, we could hope that an optimistic MDP does not prevent transitions that really exists even if they lead to states with small values;
2. if p such that $p_{i_M} = 0$, then $q(i_M)$ is never equal to 0; therefore, an optimistic algorithm that uses a maximization over a L1-ball to compute the optimistic policy may overestimate the value of some states x assuming that it may transit with a positive probability to the state x' with the largest value even if this transition is impossible under the true MDP.

In contrast, the KL-optimistic solution always puts strictly positive probability masses on observed transitions and eventually puts a zero probability mass on an unobserved transition, even if the corresponding target state has a potentially large reward. This can be observed in the Procedure 2 to compute the solution of the maximization described in (3):

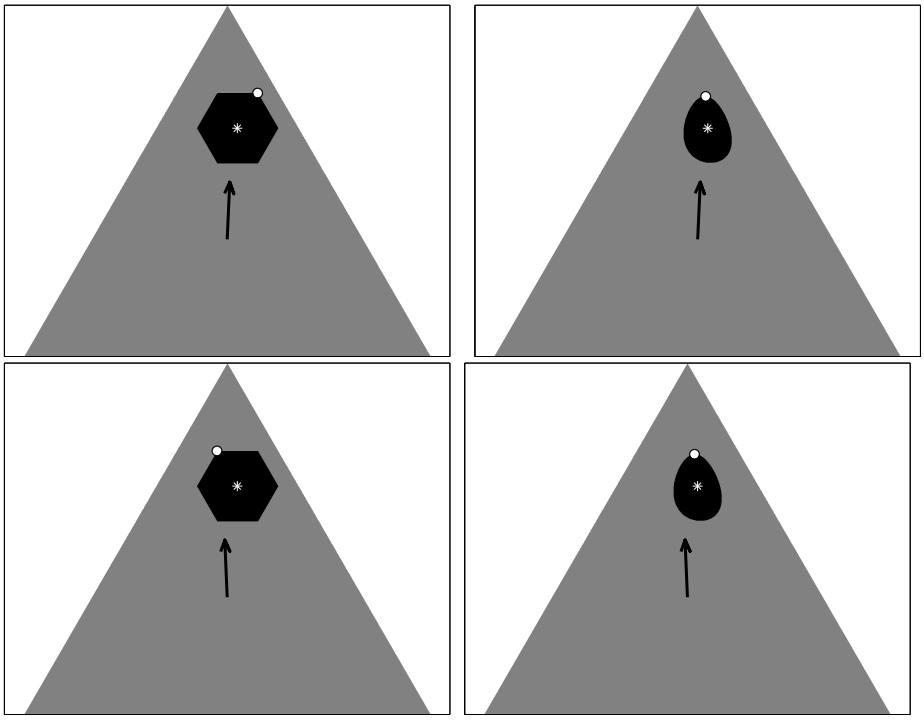

Fig. 3. The L1-neighborhood $\{q \in \mathbb{S}^3, \|p-q\|_1 \leq 0.2\}$ (left) and KL-neighborhood $\{q \in \mathbb{S}^3, KL(p,q) \leq 0.02\}$ (right) around the probability vector $p = (0.15, 0.2, 0.65)'$ (white star). The white point is the maximizer of equations (3) and (6) with $V = (0, 0.05, 1)'$ (up) and $V = (0, -0.05, 1)'$ (down).

1. for all i such that $p(i) \neq 0$, $q(i) \neq 0$.
2. for all i such that $p(i) = 0$, $q(i) = 0$ except if $p(i_M) = 0$ and $f(V_{i_M}) < \epsilon$ in which case $q(i_M) = 1 - \exp(f(V_{i_M}) - \epsilon)$. Remark that, as soon as ϵ is small enough, this exception can not be anymore satisfied.

We illustrate those two important differences in Figure 4 representing the L1 and KL neighborhoods, together with the maximizer of (6) and (3) if respectively $p(i_m)$ is very small but not equal to 0 or $p(i_M)$ is equal to 0. Figure 5 also

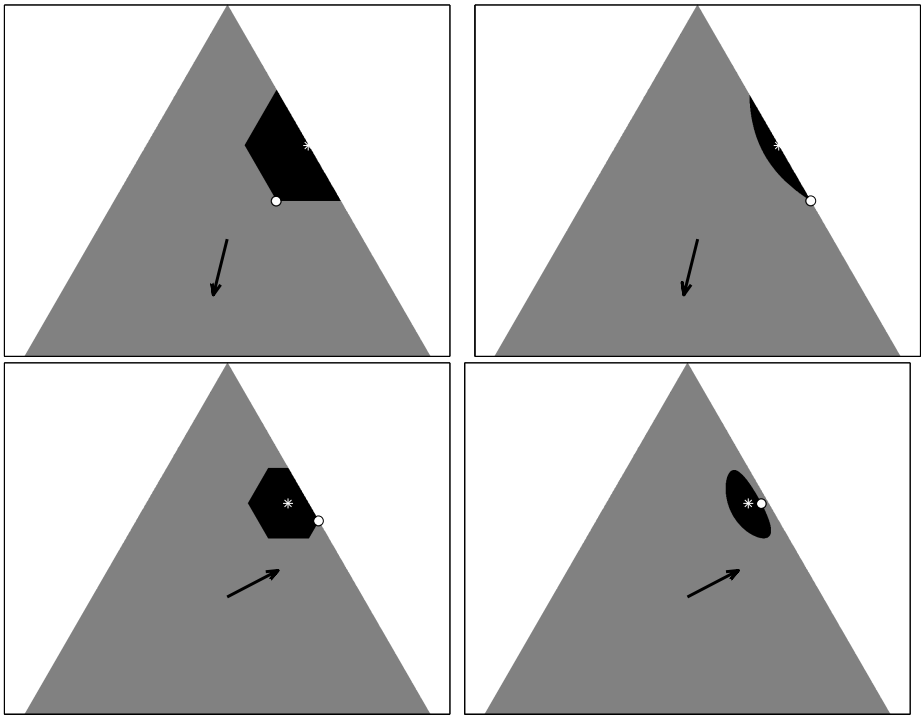

Fig. 4. The L1 (left) and KL-neighborhoods (right) around the probability vector $p = (0, 0.4, 0.6)'$ (up) and $p = (0.05, 0.35, 0.6)'$ (down). The white point is the maximizer of the equations (3) and (6) with $V = (-1, -2, -5)'$ (up) and $V = (-1, 0.05, 0)'$ (down). We took, $\epsilon = 0.05$ (up), $\epsilon = 0.02$(down) and $\epsilon' = \sqrt{2\epsilon}$.

illustrates this behaviour representing the evolution of the probability vector q that maximizes both (6) and (3) for an example with $p = (0.3, 0.7, 0)'$, $V = (1, 2, 3)'$ and ϵ decreasing from $1/2$ to $1/200$.

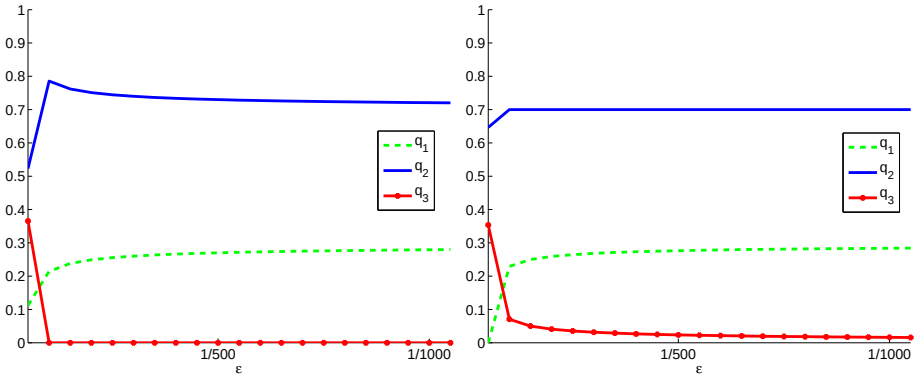

Fig. 5. Evolution of the probability vector q that maximizes both (3) (left) and (6) (right) with $p = (0.3, 0.7, 0)'$, $V = (1, 2, 3)'$ and ϵ' decreasing from $1/2$ to $1/200$

A Maximization of a linear function on a KL-ball

In this section, we expose how to solve the maximization defined in (3). The Lagrangian function for this maximization is

$$L(q, \lambda, \nu, \mu_1, \ldots, \mu_N) = \sum_{i=1}^{N} q_i V_i - \lambda \left(\sum_{i=1}^{N} p_i \log \frac{p_i}{q_i} - \epsilon \right) - \nu \left(\sum_{i=1}^{N} q_i - 1 \right) + \sum_{i=1}^{N} \mu_i q_i \ .$$

Therefore, if q is a maximum, there exists $\lambda \in \mathbb{R}$, $\nu, \mu_i \in [0, \infty[$, $i = 1 \ldots N$, such that all the following conditions are satisfied:

$$\begin{cases} V_i + \lambda \dfrac{p_i}{q_i} - \nu + \mu_i = 0 \qquad \forall i : p_i > 0 & (7) \\ V_i - \nu + \mu_i = 0 \qquad \forall i : p_i = 0 & (8) \\ \lambda \left(\displaystyle\sum_{i=1}^{N} p_i \log \frac{p_i}{q_i} - \epsilon \right) = 0 & (9) \\ \nu \left(\displaystyle\sum_{i=1}^{N} q_i - 1 \right) = 0 & (10) \\ \mu_i q_i = 0 & (11) \end{cases}$$

Let $Z = \{i, p_i = 0\}$. We can easily show that $\lambda \neq 0$ and $\nu \neq 0$; otherwise some of the conditions (7) to (11) are not satisfied. For $i \in \bar{Z}$, equation (7) implies that $q_i = \lambda \frac{p_i}{\nu - \mu_i - V_i}$. Since $\lambda \neq 0$, $q_i > 0$ and then, according to (11), $\mu_i = 0$. Therefore,

$$\forall i \in \bar{Z} \ , \quad q_i = \lambda \frac{p_i}{\nu - V_i} \ . \tag{12}$$

Let $r = \sum_{i \in Z} q_i$. Summing on $i \in \bar{Z}$ and using equations (12) and (10), we have

$$\lambda \sum_{i \in \bar{Z}} \frac{p_i}{\nu - V_i} = \sum_{i \in \bar{Z}} q_i = 1 - r \ . \tag{13}$$

Using (12) and (13), we can write $\sum_{i \in \bar{Z}} p_i \log \frac{p_i}{q_i} = f(\nu) - \log(1-r)$ where f is defined in (4). Then, q satisfies confition (9) if and only if

$$f(\nu) = \epsilon + \log(1-r) \ . \tag{14}$$

Study now the case where $i \in Z$. Let $I^* = Z \cap \operatorname{argmax}_i V_i$. Note that, for all $i \in Z/\{I^*\}$, $q_i = 0$. Indeed, otherwise, μ_i should be nul and then $\nu = V_i$ according to (8) which involves a possible negative denominator in (12). According to (11), for all $i \in I^*$, either $q_i = 0$ or $\mu_i = 0$. The second case implies that $\nu = V_i$ and $r > 0$ which requires that $f(\nu) < \epsilon$ so that (14) can be checked with $r > 0$. Therefore,

- if $f(V_i) < \epsilon$ for $i \in I^*$, then $\nu = V_i$ and the constant r can be computed solving equation $f(\nu) = \epsilon - \log(1-r)$; the values of q_i for $i \in I^*$ may be chosen in any way such that $\sum_{i \in I^*} q_i = r$;
- if for all $i \in I^*$ $f(V_i) \geq \epsilon$, then $r = 0$, $q_i = 0$ for all $i \in Z$ and ν is the solution of the equation $f(\nu) = \epsilon$.

Once ν and r have been determined, the other components of q can be computed, according to (12): we have that for $i \in \bar{Z}$, $q_i = \frac{\tilde{q}_i}{r + \sum_{i \in \bar{Z}} \tilde{q}_i}$ where $\tilde{q}_i = \frac{p_i}{\nu - V_i}$.

B Behaviour of the f function

In this section, we study the function f defined in 4 which plays an important role in the procedure to maximize a linear function over a KL-ball (see Section 3.2)..

Proposition 2. *f is a convex positive decreasing function from* $]\max_{i \in \bar{Z}} V_i; \infty[$ *to* $]0; \infty[$.

Proof (of Theorem 2). Applying Jensen's inequality to the function $x \mapsto \log(x)$, we have that the function f is positive. The first derivative of f with respect to ν is equal to

$$f'(\nu) = \frac{\left(\sum_i \frac{p_i}{\nu - V_i}\right)^2 - \sum_i \frac{p_i}{(\nu - V_i)^2}}{\sum_i \frac{p_i}{\nu - V_i}} \ .$$

Applying the Jensen's equality, we easily show that f is strictly decreasing. In addition, the second derivative of f with respect to ν satisfies

$$f''(\nu) = -\sum_i \frac{p_i}{(\nu - V_i)^2} + \frac{2 \sum_i \frac{p_i}{(\nu - V_i)^3} \sum_i \frac{p_i}{\nu - V_i} - \left(\sum_i \frac{p_i}{(\nu - V_i)^2}\right)^2}{\left(\sum_i \frac{p_i}{\nu - V_i}\right)^2} \ .$$

Let Z be the positive random value such that $\mathbb{P}\left(Z = \frac{1}{\nu - V_i}\right) = p_i$. The second derivative of f can then be written as follows:

$$f''(\nu) = \frac{2\mathbb{E}(Z^3)\mathbb{E}(Z) - \mathbb{E}(Z^2)\mathbb{E}(Z)^2 - \mathbb{E}(Z^2)^2}{\mathbb{E}(Z)^2} \ .$$

Using Cauchy-Schwartz inequality, we have $\mathbb{E}(Z^2)^2 = \mathbb{E}(Z^{3/2}Z^{1/2})^2 \leq \mathbb{E}(Z^3)\mathbb{E}(Z)$. In addition $\mathbb{E}(Z^2)^2 = \mathbb{E}(Z^2)\mathbb{E}(Z^2) \geq \mathbb{E}(Z^2)\mathbb{E}(Z)^2$. These two inequalities show that $f''(\nu) \geq 0$. We conclude that f is a convex function.

As mentioned in Section 3.2, teh Newton's method can be applied to solve the equation $f(\nu) = \epsilon$ for a fixed value of ϵ. When ϵ is close to 0, the solution of this eqaution is quite large and an appropriate initialization of Newton's algorithm enables to accelerate his convergence to the solution. According to the following proposition, for ν large enough, $f(\sqrt{\frac{\sigma_{p,V}}{2\epsilon}}) \sim \epsilon$ where $\sigma_{p,V} = \sum_i p_i V_i^2 - (\sum_i p_i V_i)^2$. So, we propose to intialize the algorithm taking $\nu_0 = \sqrt{\frac{\sigma_{p,V}}{2\epsilon}}$.

Proposition 3. *For ν near ∞, we have $f(\nu) \sim \frac{\sigma_{p,V}}{2\nu^2}$ where $\sigma_{p,V} = \sum_i p_i V_i^2 - (\sum_i p_i V_i)^2$.*

Proof (Proof of Theorem 3). Remark that the function f defined in (4) can be written as follows:

$$f(\nu) = \sum_i p_i \log\left((\nu - V_i)\sum_j \frac{p_j}{\nu - V_j}\right) \ . \tag{15}$$

For ν near ∞, using a second-order Taylor's-series approximation, we have, for all j, $\frac{p_j}{\nu - V_j} = \frac{p_j}{\nu}\left(1 + \frac{V_j}{\nu} + \frac{V_j^2}{\nu^2} + o(\frac{1}{\nu^2})\right)$. Then, including this result in equation (15), for ν near ∞, we have

$$f(\nu) = \sum_i p_i \log\left(1 - \frac{V_i - pV}{\nu} + \frac{pV^2 - V_i pV}{\nu^2} + o(\frac{1}{\nu^2})\right) \ ,$$

where we used the notation $pV \stackrel{\text{def}}{=} \sum_i p_i V_i$ and $pV^2 \stackrel{\text{def}}{=} \sum_i p_i V_i^2$. Using the Taylor serie of the logarithm function, we have, for ν near ∞,

$$f(\nu) = \frac{pV^2 - (pV)^2}{\nu^2} - \frac{1}{2\nu^2}(pV^2 - (pV)^2) + o(\frac{1}{\nu^2}) = \frac{1}{2\nu^2}\mathrm{Var}_p(V) + o(\frac{1}{\nu^2}) \ .$$

References

1. Auer, P., Jaksch, T., Ortner, R.: Near-optimal regret bounds for rein- forcement learning (full version). Tech. rep., URL : http://institute.unileoben.ac.at/infotech/publications/ ucrl2.pdf. (2009)

2. Auer, P., Jaksch, T., Ortner, R.: Near-optimal regret bounds for reinforcement learning. In: Advances in Neural Information Processing Systems. vol. 21 (2009)
3. Auer, P., Ortner, R.: Logarithmic online regret bounds for undiscounted reinforcement learning. Advances in Neural Information Processing Systems: Proceedings of the 2006 Conference p. 49 (2007)
4. Bartlett, P., Tewari, A.: REGAL: A Regularization based Algorithm for Reinforcement Learning in Weakly Communicating MDPs (2009)
5. Bertsekas, D.: Dynamic Programming and Optimal Control, Two Volume Set. Athena Scientific (1995)
6. Brafman, R., Tennenholtz, M.: R-max-a general polynomial time algorithm for near-optimal reinforcement learning. The Journal of Machine Learning Research 3, 213–231 (2003)
7. Burnetas, A., Katehakis, M.: Optimal adaptive policies for Markov decision processes. Mathematics of Operations Research pp. 222–255 (1997)
8. Even-Dar, E., Mannor, S., Mansour, Y.: Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. The Journal of Machine Learning Research 7, 1079–1105 (2006)
9. Garivier, A., Moulines, E.: On upper-confidence bound policies for non-stationary bandit problems. Arxiv preprint arXiv:0805.3415 (2008)
10. Garvier, A., Leonardi, F.: Context tree selection: A unifying view (20109)
11. Kearns, M., Singh, S.: Near-optimal reinforcement learning in polynomial time. Mach. Learn. 49(2-3), 209–232 (2002)
12. Lai, T., Robbins, H.: Asymptotically efficient adaptive allocation rules. Advances in Applied Mathematics 6(1), 4–22 (1985)
13. Puterman, M.: Markov Decision Processes: Discrete Stochastic Dynamic Programming. John Wiley & Sons, Inc. New York, NY, USA (1994)
14. Strehl, A., Littman, M.: An analysis of model-based interval estimation for Markov decision processes. Journal of Computer and System Sciences 74(8), 1309–1331 (2008)
15. Sutton, R.: Reinforcement Learning. Springer (1992)
16. Tewari, A., Bartlett, P.: Optimistic linear programming gives logarithmic regret for irreducible MDPs. Advances in Neural Information Processing Systems 20, 1505–1512 (2008)