# ON ASYMPTOTICALLY OPTIMAL REINFORCEMENT LEARNING

**by**

**DANIEL PIRUTINSKY**

**A dissertation submitted to the**

**Graduate School - Newark**

**Rutgers, The State University of New Jersey**

**In partial fulfillment of the requirements**

**for the degree of**

**Doctor of Philosophy**

**Graduate Program in Management**

**written under the direction of**

**Professors Michael N. Katehakis and C. Wesley Cowan**

**and approved by**

_____

_____

_____

_____

_____

_____

_____

**Newark, New Jersey**

**October, 2020**

## ABSTRACT OF THE DISSERTATION

# On Asymptotically Optimal Reinforcement Learning

**By Daniel Pirutinsky**

**Dissertation Director:**

**Professors Michael N. Katehakis and C. Wesley Cowan**

We consider the problem of minimizing the long term average expected regret of an agent in an online reinforcement learning environment. In particular, we model this as a Markov Decision Process (MDP) where the underlying transition laws are unknown. There have been many recent successful applications in this area as well as many recent advances in theoretical techniques. However, there still is a significant gap between rigorous theoretical techniques and those that are in actual use. This work represents a step towards shrinking that gap.

In the first part we develop a set of properties sufficient to guarantee that any policy satisfying them will achieve asymptotically minimal regret (up to constant factor of the logarithmic term). The goal in this is to, rather than simply add one more learning policy to the mix, build a flexible framework that may be adapted to a variety of estimative and adaptive policies that are already in use and grant confidence in the performance. To that aim, this work lays the groundwork for what we believe is a useful technique for proving asymptotically minimal rate of regret growth. The conditions are presented here along with hints for how a verifier may prove that their particular algorithm satisfies these conditions. The ideas in this work build strongly on those of [1].

In the second part of this work, we derive an efficient method for computing the indices associated with an asymptotically optimal upper confidence bound algorithm (MDP-UCB) of [1]

that only requires solving a system of two non-linear equations with two unknowns, irrespective of the cardinality of the state space of the MDP. In addition, we develop the MDP-Deterministic Minimum Empirical Divergence (MDP-DMED) algorithm extending the ideas of [2] for the Multi-Armed Bandit (MAB-DMED) and we derive a similar acceleration for computing these indices that involves solving a single equation of one variable. We provide experimental results demonstrating the computational time savings and regret performance of these algorithms. In these comparison we also consider the Optimistic Linear Programming (OLP) algorithm [3] and a method based on Posterior (Thompson) sampling (MDP-PS).

# Acknowledgements

This dissertation and my entire educational journey has only been possible because of the assitance of many. While impossible to name everyone who has contributed, I will do my best. I acknowledge and thank the following.

Dr. Michael Katehakis, for taking a chance on me despite my non-traditional background and encouraging me to pursue a doctorate. Your decision to support me has changed my life for the better in ways that are hard to overstate. Thank you for believing in me.

Dr. Wesley Cowan, without whom I would never have completed this work. Thank you for the countless hours spent discussing the topics herein, arguing over technical minutiae, and helping form the foundation that this thesis is built upon. Your friendship is one that I will always cherish.

Luz Kosar and Monnique DeSilva your assistance in getting things done and navigating campus issues as they came up has been invaluable. James Poinsett for introducing me to the wider world of 'secular' higher education, helping me forge connections, and giving me confidence in my own abilities.

Footsteps for being the resource I continue to lean on when things get rough. I cannot think of another group of humans who have collectively done more to help me grow into who I am today. Jewish Queer Youth, Eshel, and Young Advocates for Fair Education, both for what you have done for me personally, and for what you do to assist others like me. Your work is honorable, valuable, and sadly, desperately needed.

My co-parent Esti, who, battered by the storm of my changing personal life, met the challenge with kindness, love, and understanding. You have always put the needs of our children first and kept them healthy, happy, stable, and loved. I know it has not been easy for you. Thank you for making our kids life, and by extension mine, so rewarding.

Lastly, my children Aliza and Nechama, who consistently inspire me to do better.

# Dedication

To those who bend the arc of the moral universe towards greater equality.

# Table of Contents

# Chapter 1

# Introduction, Background, and Formulation

## 1.1 Introduction

Reinforcement learning is a rapidly growing area of research with new techniques, problem specifications, and applications being constantly being developed. There have been many successful applications in this area, including Atari games [4], Go [5], currency trading [6], optimizing trade execution [7], clinical trials [8] and many more [9]. Although these seem to enjoy significant empirical success, lack a rigorous theoretical foundation that can give performance guarantees remains an issue. The increasing reliance in industry, government, healthcare, safety, robotics on these successful techniques come with an increasing concern about edge cases, safety, and performance guarantees. As noted in [10], among others, there still is a significant gap between rigorous theoretical results and techniques and those that are in actual use. This work represents an attempt to start closing that gap.

In Chapter 2 we approach this by developing a set of properties sufficient to guarantee that any policy satisfying them will achieve asymptotically minimal regret (up to constant factor of the logarithmic term). The goal in this is to, rather than simply add one more learning policy to the mix, build a flexible framework that may be adapted to a variety of estimative and adaptive policies and grant confidence in the performance. To that aim, work lays the groundwork for what we believe is a very useful technique for proving asymptotically minimal rate of regret growth. The conditions are presented here along with hints for how a verifier may prove that their particular algorithm satisfies these conditions. The ideas in this work build strongly on those of [1].

The practical use of the asymptotically optimal UCB algorithm (MDP-UCB) of [1] has been hindered [3, 11] by the computational burden of the upper confidence bound indices c.f. Eq. (3.1). In Chapter 3 we derive an efficient method for computing these indices that only requires

solving a system of two non-linear equations with two unknowns, irrespective of the cardinality of the state space of the MDP. In addition, we develop a similar acceleration for computing the indices for the MDP-Deterministic Minimum Empirical Divergence (MDP-DMED) algorithm developed in [12], based on ideas from [2], that involves solving a single equation of one variable. We provide experimental results demonstrating the computational time savings and regret performance of these algorithms. In these comparison we also consider the Optimistic Linear Programming (OLP) algorithm [3] and a method based on Posterior sampling (MDP-PS).

The particular problem under consideration in this work is the online maximization of the expected average long run reward for an agent in an MDP with unknown transition probabilities. First introduced in [13] and later studied by [1] among others. The particulars of the problem are an important part of the results and the problem is of interest in itself. Long term average reward is ideally suited to online applications and long term training. We also believe that the techniques used here can be extended to other related forms of RL problems, for example, those with unknown rewards. Other criteria for evaluating the effectiveness of a particular RL algorithm include Probably Approximately Correct [14], Minimax regret [15], among others.

## 1.2 Reinforcement Learning Related Works

As this is a fast growing area of research, there is a lot of recent work. A good resource for reinforcement learning problems and their potential solution methods is [16]. For a more bandit focused approach, [17] has a nice overview of the current state of the art. Most directly relevant to this paper are Chapters 8, 10, and 38 therein. [18] discuss online learning while minimizing regret for predicting individual sequences of various forms, with Chapter 6 (bandit related problems) therein being most relevant here.

## 1.3 Formulation

Reinforcement learning problems are commonly expressed in terms of a *controllable*, *probabilistic*, *dynamic system*, where the dynamics must be learned over time. The classical model

for this is that of a discrete time, finite state and action Markov decision process (MDP). See for example, [19] and [11]. In particular, learning is necessary when the underlying dynamics (the transition laws) are unknown, and must be learned by observing the effects of actions and transitions of the system over time.

A finite MDP is specified by a quadruple $(S,A,R,P)$, where $S$ is a finite state space, $A = [A(x)]_{x\in S}$ is the action space, with $A(x)$ being the finite set of admissible actions in state $x$, $R = [r_{x,a}]_{x\in S,a\in A(x)}$, is the expected reward structure and $P = [p^a_{x,y}]_{x,y\in S,a\in A(x)}$ is the transition law. Here $r_{x,a}$ and $p^a_{x,y}$ are respectively the one step expected reward and transition probability from state $x$ to state $y$ under action $a$. Stated more explicitly, when in state $x$, taking action $a \in A(x)$ yields a fixed reward of $r_{x,a}$ and a transition to state $y$ with probability $p^a_{x,y}$.

It is convenient to denote subsets of the full set of action $A$, by $A' \subset A$ which is taken to mean $A'(x) \subset A(x)$ (and $A'(x)$ is non-empty) for all $x \in S$.

The central problem of interest is then to determine, for each state, which action the agent should take in that state. In this work, we consider the problem of identifying the optimal actions to maximize the *long term expected average returns*. Taking $X_t$ to be the state at time $t$ and $H_t$ the total history up to time $t$, and $\pi(x,h) \in A(x)$ to be the action taken in state $x$ given history $h$, the problem can be stated as follows: Find a policy $\pi$ to realize the maximum

$$\phi^* = \max_{\pi} \lim_{n\to\infty} \frac{1}{n}\mathbb{E}\left[\sum_{t=0}^{n-1} r_{X_t,\pi(X_t,H_t)}\right]. \tag{1.1}$$

When all elements of $(S,A,R,P)$ are known the model is said to be an MDP with *complete information* (CI-MDP). In this case, optimal polices can be obtained via the appropriate version of Bellman's equations, given the prevailing optimization criterion, state, action, time conditions and regularity assumptions; c.f. [20], [21]. When some of the elements of $(S,A,R,P)$ are unknown the model is said to be an MDP with incomplete or *partial information* (PI-MDP). This is the primary model of interest for reinforcement learning, when some aspect of the dynamics must be learned through interaction with the system.

For the body of this work we consider the following partial information model: the transition probability vector $\underline{p}^a_x = [p^a_{x,y}]_{y\in S}$ is taken to be an element of the parameter space

$$\Theta = \left\{ \underline{p} \in \mathbb{R}^{|S|} : \sum_{y\in S} p_y = 1, \forall y \in S, p_y > 0 \right\},$$

that is, the space of all $|S|$-dimensional probability vectors.

The assertion of this parameter space deserves some unpacking. It is at first simply a theoretical convenience—it ensures that for any control policy, the resulting Markov chain is irreducible. It also represents a complete lack of prior knowledge about the transition dynamics of the MDP. Knowing that certain state-state transitions are impossible requires prior model specific knowledge (such knowing the rules of chess). Learning based purely on finite observed data could never conclude that a given transition probability is zero. Thus, we assert a uniform Bayesian prior on the transition probabilities and therefore the likelihood associated with $p = 0$ is 0. In this way, asserting this parameter space starting out represents a fairly agnostic initial view of the underlying learning problem. A possible future direction of study is to examine how to efficiently incorporate prior knowledge, for instance modifying the specified parameter space, into the learning process without compromising on the learning rate. [22] and [23] discuss hidden parameterized transition models, for example, which leverage additional prior knowledge about the transition probability space.

We will take this unknown transition law to be the only source of incomplete information about the underlying MDP. The reward structure $R = [r_{x,a}]_{x\in S, a\in A(x)}$ is taken to be known (at least in expectation), and constant. Much of the discussed algorithms will generalize to the situation where the distribution of rewards must also be learned, but we reserve this for future work.

As final notation, it is convenient to define the specific data available at any point in time, under a given (understood) policy $\pi$: let $T_x(t), T^a(t), T^a_{x,y}(t)$ be, respectively, the number of visits to state $x$, the number of times action $a$ was taken in state $x$, and the number of transitions from $x$ to $y$ under action $a$, that are observed in the first $t$ rounds.

In the next subsection, we consider the case of the controller having complete information (the best possible case) and use this to motivate notation and machinery for the remainder of the work.

### 1.3.1 MDPs under Complete Information

A classical result of [19] is that in order to maximize the long-term expected average value, as in Eq. (1.1), it suffices for the agent to choose actions based only on the current state. Hence,

we can restrict ourselves to policies that depend *only* on the current state $X_t$. Indeed, in the irreducible case, the optimal policy can be derived from the solution $(\phi^*, \underline{v}^*)$ to the following system of Bellman's equations:

$$\Pi(A,P) = (\phi, \underline{v}) \quad \text{such that} \quad \left\{ \forall x \in S : \phi + v_x = \max_{a \in A(x)} \left[ r_{x,a} + \sum_{y \in S} p^a_{x,y} v_y \right] \text{ and } \sum_{y \in S} v_y = 0 \right\}. \tag{1.2}$$

Given such a solution $(\phi^*, \underline{v}^*)$ to the above, an optimal policy (i.e., one that realizes the maximal long term average expected value) may be realized by the agent taking the maximizing action in every state, i.e.,

$$\pi^*(x) = \arg\max_{a \in A(x)} \left[ r_{x,a} + \sum_{y \in S} p^a_{x,y} v^*_y \right]. \tag{1.3}$$

It is worth noting that in the case there are multiple such actions, any arbitrary selection of them will suffice. Indeed, any optimal deterministic policy will realize equality in this system [19].

The constant $\phi^*$ above represents the maximal long term average expected reward of an optimal policy. The vector $\underline{v}^*$, or more precisely, $v^*_x$ for any $x \in S$, represents in some sense the immediate value of being in state *x relative to the long term average expected reward*. The value $v_x$ essentially encapsulates the future opportunities for value available due to being in state $x$ [19].

### 1.3.2 MDPs under Partial Information and Regret

Determining the policy $\pi^*$ as in Eq. (1.3) depends on having complete knowledge of the transition law $P$. If it is unknown, or only approximately known, the underlying optimality values $(\phi^*, \underline{v}^*)$ will be unknown, and the optimal actions cannot be determined with certainty. This may frequently be the case, especially when the transition law must be learned via experimentation or approximated based on available data. In this case, the agent need not only attempt to determine the optimal actions, but also determine the actions worth experimenting with so as to collect more data - the classical exploration vs exploitation dilemma.

To quantify the performance of a policy, we utilize the concept of regret or regretful actions: the number of times an agent takes an action that is sub-optimal. For any policy $\pi$, since

$T^a(n) \leq n$ we immediately get that for *all* policies $\pi$, and any transition law $P$, for any sub-optimal action, $a$, we have:

$$\mathbb{E}[T^a(n)] = O(n). \tag{1.4}$$

This is trivial. A good policy should therefore achieve a tighter bound (e.g. $o(n)$) for all possible $P$. In [1], they consider policies that are "uniformly fast" for all $P$, that is, for all sub-optimal actions $a$, $\mathbb{E}[T^a(n)]$ grows slower than any polynomial function of $n$. Formally,

$$\mathbb{E}[T^a(n)] = o(n^\alpha) \; \forall \alpha > 0. \tag{1.5}$$

The central result there is that for any uniformly fast policy, logarithmic regret is the best that can be achieved universally, i.e., for any uniformly fast policy $\pi$,

$$\liminf_{n\to\infty} \frac{\mathbb{E}[T^a(n)]}{\ln n} \geq C^*(S,A,R,P), \tag{1.6}$$

for some positive constant $C^*(S,A,R,P)$. One interpretation of this lower bound is that learning *requires* mistakes. Some amount of experimentation with sub-optimal actions is necessary to be able to correctly identify the true optimum, and this will incur mistakes at least at this rate. This is in fact tight, as that paper goes on to show, demonstrating both rate constants on this lower bound, and a specific MDP policy that achieves this lower bound.

# Chapter 2

# Sufficient Conditions for Asymptotically Optimal Reinforcement Learning

## 2.1 Introduction

This chapter can be seen as an extension or generalization of the ideas in [1]. Rather than focusing on specific index policy, we use the insights and structure of the analysis in [1] to guide the development of sufficient conditions that will guarantee asymptotic logarithmic growth of regret for a wide range of algorithmic techniques. We present a set of sufficient conditions such that if a verifier can show that their algorithm satisfies them, they can enjoy guaranteed asymptotic minimal regret. Armed with these results the verifier can be confident that their technique is actually optimal. Where appropriate we include hints for how the verifier may show these conditions, and why we think these can be more widely applied to existing techniques in practice.

### 2.1.1 Related Work

Early work in [21], [24], and [25], focused on establishing conditions under which consistent algorithms could be derived. For conditions on the algorithms, the main result of [26] is that under some general conditions, value iteration methods (e.g. Q-learning [27]) will converge to true state-action values. The problem they study is a finite MDP with the goal of minimizing total expected discounted cost. Their goal of simplifying the proof of the convergence of value iteration RL methods dovetails nicely with the results presented here. Their results can be seen as a step towards proving condition 2. Namely, if the technique that an algorithm uses for value estimation is a value iteration method of the type discussed and can then be shown to converge fast enough, it will satisfy condition 2. An important caveat is that their results show

that these methods will converge with probability 1 but fail to address the rate at which these methods converge, which is of central importance here. [26] addresses the estimation of the value function. This is an important part of the reinforcement learning problem, but doesn't address the exploration vs exploitation aspect. In other words, at what rate are we converging? Can we go faster? How are we choosing the specific actions to take? In fact, we conjecture that our technique here, coupled with strong results about the convergence rate of value function estimators, can be used to provide further rigorous guarantees for the growth rate of regret for a wide range of algorithms.

### 2.1.2 Chapter Structure

This chapter is organized as follows. Section 2.2 introduces the concept of over-sampling and minimally sampling, which aside from being important to the results of this work, are of interest in themselves as they give some nice intuition about the underlying learning problem. In Section 2.3 we leverage the previous discussion about sampling rates to formalize the effective estimation of the unknown quantities, namely the transition probabilities and the state values, and introduce the concept of the 3 different versions of the MDP, $\Pi^*$, $\Pi_t$, and $\hat{\Pi}_t$. This section is the basis for conditions 1 and 2. In Section 2.4 we establish some general notation, some regularity events, and convenient results. The basis for condition 3 is established here as well. The fundamental problem of exploration and exploitation is discussed in Section 2.5 which lead to conditions 4 and 5. Section 2.6 is where the conditions are explicitly stated along with hints for a verifier on how they might proceed to establish them. In Section 2.7 we state the main theorem of this paper and give the proof. In Section 2.8 we conclude by discussing possible future directions that this research can take. Finally, Section 2.9 establishes and proves some useful lemmas about various events for this reinforcement learning problem.

## 2.2 Sampling Rates

As was hinted at in Section 1.3.2, the logarithmic lower bound of regret from [1], suggests that for each action there is a logarithmic sampling rate, dependent on the unknown transition probabilities, at which the action must be taken in order to distinguish it from the optimal

actions. For any action $a$, let $g^a(n) = \beta^a \ln(n)$, where $\beta^a$ is some positive constant that may depend on the unknown underlying MDP instance, be this minimal sampling rate. We say an action has been minimally sampled at time $n$ if it has been taken at least $g^a(n)$ times. Formally,

**Definition 1.** ***Minimally Sampled Events***

*Let the event minimally sampled, $MS_{n,a}$ denote the event that action a has been sampled at least $g^a(n) = \beta^a \ln(n)$ times by time n. More precisely,*

$$MS_{n,a} = \{T^a(n) \geq g^a(n)\} \tag{2.1}$$

Next we introduce a useful lemma relating this minimal sampling rate to regret.

**Lemma 1.** ***Minimally Sampled Counts***

*The expected number of times an action a has been taken when it has not been sampled $g^a(t)$ times by time n is less than $g^a(n) = O(\ln(n))$. Formally,*

$$\mathbb{E}\left[\sum_{t=0}^{n} \mathbb{1}\left\{\pi_t = a, MS_{t,a}^c\right\}\right] < g^a(n) = O(\ln(n)) \tag{2.2}$$

Proof of this lemma is provided in Section 2.9.1.

This minimal sampling rate also implies that if we are to achieve logarithmic regret, the **only** actions that should be taken super-logarithmically are the truly optimal actions. With this as motivation we define the over-sampled rate. Let $b(n)$ be a sub-linear, super-logarithmic function, for example $b(n) = \ln^2(n)$. We say an action is over-sampled at time $n$ if it has been sampled at least $b(n)$ times. Formally,

**Definition 2.** ***Over-Sampled Action***

*For a sub-linear, super-logarithmic function $b(n)$, we say an action a is over-sampled at time n, if it has been sampled at least $b(n)$ times. That is,*

$$T^a(n) \geq b(n) \tag{2.3}$$

Figure 2.1 is an illustrative plot of these rates to give a better intuitive sense of these rates. Consider some action $a$ and plot the number of times it has been been sampled by time $n$, $T^a(n)$. We can divide this plot into three distinct areas.

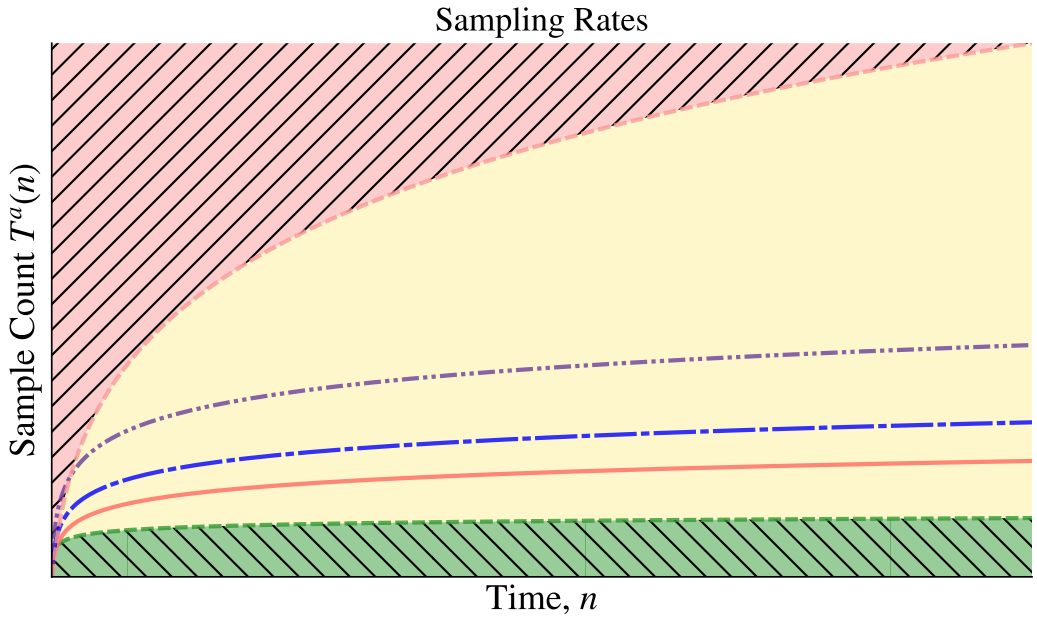


Figure 2.1:

The bottom green area represents a sub-logarithmic sampling rate. If the number of samples, $T^a(n)$, is in that area, we know that we do not have enough information to have ruled it out as being optimal. That is, we have under sampled the action and regardless of our current estimates we should continue to sample it in order to collect more information. The top red area represents a super-logarithmic sampling rate that is greater than or equal to $b(n)$. If the number of samples, $T^a(n)$, is in that area, then we know have *more* than enough information to determine if it is optimal or not. Indeed, only optimal actions should be sampled more than $b(n)$ times, so if we estimate that action $a$ is sub-optimal, we should stop taking it. Both of these rates are in some sense "obvious". That is, they hold, independent of the unknown transition probabilities, and thus the controller can make decisions about these directly. This insight will motivate Definition 7 and lead us to condition 3, explicitly defined in Section 2.6.

The key difficulty is the yellow area in the middle. For a controller to achieve minimal regret they must ensure that all actions are minimally sampled, $g^a(n)$, but not over-sample sub-optimal actions (top red area). We plot three example possible forms of $g^a(n)$ in that area. Since the minimal sampling rate $g^a(n)$, depends on the unknown transition probabilities, it is

within this band that the controller must learn the transition probabilities, attempt to identify the necessary minimal sample rate, and balance exploration and exploitation. This balancing act will be discussed in more detail in Section 2.5. First, however, we discuss ensuring that we use the data we collect to effectively and efficiently estimate the properties of the MDP.

## 2.3 Estimating the Unknown Probabilities

As indicated previously, knowledge of $P$ and $\underline{v}$ is sufficient to define and utilize the true optimal policy. However, if that information is not available, it must be estimated from the data available at a given time, i.e., observed transitions for actions taken up to a given time. This makes separating the action policy from the estimators difficult - the data available to construct the estimators at any time depends explicitly on the policy through the actions taken. Further, while the transition probabilities for a given state and action represent local information, a control policy might construct estimators for those transition probabilities based on globally sampled data (i.e., transitions observed from other potentially related states and actions). In both these cases, the properties of estimators for $P$ and $\underline{v}$ at a specific time may depend heavily on the properties of the policy that brought the controller to that point. However, we can at least require that our estimators use the data that they have collected to efficiently estimate what they can. Let $\hat{P}_t$ be an estimate of $P$, constructed from whatever information is available at time $t$ (the implicit dependence on $\pi$ is suppressed). The estimate $\hat{P}_t$ is a **fast local $\varepsilon$-effective estimator** if for all sufficiently small $\varepsilon > 0$, it quickly provides accurate estimates for the local transition vectors as more data is collected. Formally,

**Comment [CWC1]:** only serious isssue is all the $1/e^k$ order bounds are wrong needs to be something like $O(1/\eta^k)$ for some $\eta > 1$ ↑

**Definition 3.** ***Fast local $\varepsilon$-effective estimators***

*$\hat{P}_t$ is a fast local $\varepsilon$-effective estimator if for all sufficiently small $\varepsilon > 0$, for sufficiently large local time $k = T^a(t)$, if the estimator depends solely on local time $k$,*

$$\mathbb{P}\left(||\hat{\underline{P}}^a_x(t) - \underline{P}^a_x|| > \varepsilon \text{ and } T^a(t) = k\right) \leq O\left(\frac{1}{e^k}\right). \tag{2.4}$$

*If the estimator depends on global time, we require,*

$$\mathbb{P}\left(\exists t' \geq t : |\hat{\underline{P}}^a_x(t') - \underline{P}^a_x| > \varepsilon \text{ and } T^a(t') = k\right) \leq O\left(\frac{1}{e^k}\right) \tag{2.5}$$

This is indeed condition 1 discussed in Section 2.6.

Either of the above are natural properties for any worthwhile estimator to satisfy, essentially guaranteeing that the decision to take a given action will improve knowledge *at least about that action*. This guarantees that actions are always locally well-informed, but this does not guarantee global performance. Constructing a worthwhile policy then depends on successfully integrating this local information to infer global knowledge.

We can begin this process with the following observation, if every action is better known the more often it is taken, there must be some subset of actions over which the estimator is accurate. Indeed, recalling our discussion on sampling rates in Section 2.2, if an action is minimally sampled it will be well estimated. However, determining the minimal sampling rate $g^a(t)$ requires knowledge about the unknown transition probabilities, which requires us to sample at the minimal sampling rate! To address this dilemma, recall that if an action is over-sampled it will be minimally sampled regardless of its unknown transition probabilities. Thus, if we restrict our attention to just over-sampled actions we are guaranteed to be using well estimated actions. This motivates the following definition.

**Definition 4.** *The set of over-sampled actions $A_t$, is the set of actions such that, for each $x \in S$:*

$$A_t(x) = \{a \in A(x) : T^a(t) \geq b(T_x(t))\}. \tag{2.6}$$

Where $b(t)$ is the over-sample rate function defined in Section 2.2. [1]

Hence, the locally informative estimator $\hat{P}_t$ can be said to give *globally* accurate information, at least about the restricted problem on actions in $A_t$. This definition is independent of the underlying policy - it simply states that whatever actions are used sufficiently often ($b$-often), the transition estimates for those actions are good. It is straightforward to construct and verify such estimators in this environment but it is worth taking a more general probabilistic view for the following reason. In applications, an agent may take any number of approaches toward approximating the transition law (naive estimates, neural networks, etc) and in fact prior information as to the structure of the transition law (probabilities peaking around related or similar

[1] If a given $x$ has been visited sufficiently rarely that $T^a(t) < b(T_x(t))$ for all such $a$, it is convenient to take $A_t(x) = A(x)$. Recalling that $b$ increases sub-linearly and with high probability we have visited every state linearly often (see Lemma 8) this will occur rarely.

states, dropping for distant or dissimilar states) may inform and tighten the resulting estimates. Whatever the case, all results to follow depend simply on the agent's estimators converging in the indicated way, however the agent arrives at that construction.

In particular, if we have an effective estimator of the underlying transition law, then with high probability, we have good estimates of at least *some* of the available actions in each state. While there may be uncertainty about the *entire* MDP, we can use these estimates to derive good information about a *restricted* MDP.

This can be summarized in the following way. There are essentially three MDP problems we are interested in at any point in time, and therefore three sets of Bellman's Equations (1.2) we are interested in solving:

- $\Pi^*$: the full 'unrestricted' problem $\Pi(A,P)$ over action set $A$ and transition law $P$,
- $\Pi_t$: the 'restricted' problem at time $t$, $\Pi(A_t, P[A_t])$, over action set $A_t$ and transition law $P[A_t]$,
- $\hat{\Pi}_t$: the 'estimated' restricted problem at time $t$, $\Pi(A_t, \hat{P}_t[A_t])$, over action set $A_t$ and transition law $\hat{P}_t[A_t]$.

Note that at any time, $\Pi^*$ the true unknown MDP problem. The problem $\hat{\Pi}_t$ is a known MDP and solvable, as it depends only on the known estimates available at any time. The problem $\Pi_t$ is an interesting midpoint between these two, as it is an unknown MDP which is restricted to actions that are over-sampled, and therefore have good estimates for their transition laws. We will show that solving $\hat{\Pi}_t$ serves as a good estimate for $\Pi_t$, and how to leverage this information about $\Pi_t$ into discovering the solution for $\Pi^*$.

For $\Pi_t$ and $\Pi^*$, we may take the solutions to the Bellman equations to be $(\phi_t, \underline{v}_t)$ and $(\phi^*, \underline{v}^*)$ respectively. For $\hat{\Pi}_t$, it is convenient to potentially consider approximate solutions $(\hat{\phi}_t, \underline{\hat{v}}_t)$, as solving the full system of Bellman equations may be expensive. Based on the solutions $(\phi^*, \underline{v}^*), (\phi_t, \underline{v}_t)$, and $(\hat{\phi}_t, \underline{\hat{v}}_t)$, it is convenient to denote the optimal action sets corresponding with each as $O^* \subset A, O_t^* \subset A_t$, and $\hat{O}_t^* \subset A_t$, respectively. As the solution $(\hat{\phi}_t, \underline{\hat{v}}_t)$ may only be an estimate, like the estimators $\hat{P}_t$ above, we introduce the following definition. The estimate $(\hat{\phi}_t, \underline{\hat{v}}_t)$ is a **fast global $\delta$-effective estimator** if for all sufficiently small $\delta > 0$, it quickly provides accurate estimates for the global state values as more data is collected. Formally,

**Definition 5.** ***Fast global $\delta$-effective estimators***

*For all sufficiently small $\delta > 0$, for sufficiently large global time $t$,*

$$\mathbb{P}\left(||\hat{\Pi}_t - \Pi_t|| > \delta\right) \leq O\left(\frac{1}{e^t}\right). \tag{2.7}$$

This is indeed condition 2 discussed in Section 2.6.

As in the case of the estimators $\hat{P}_t$ above, the existence and an explicit construction of such an approximate solution can be easily shown, but these results will hold regardless of how such a solution is obtained.

The following result summarizes what knowledge effective estimators grant about the underlying restricted problem.

**Definition 6.** ***Good Estimation Event***

*The Good Estimation event, $GE_t$ is defined to be*

$$GE_t = \left\{ \begin{array}{l} |\hat{\underline{P}}^a_{x,t'} - \underline{P}^a_x| \leq \varepsilon || \textit{ for all } a \textit{ such that } MS_{t',a}, \\ ||(\hat{\phi}_{t'}, \hat{\underline{h}}_{t'}) - (\phi_{t'}, \underline{h}_{t'})|| \leq \delta, \\ \hat{O}^*_{t'} \subset O^*_{t'} \textit{ for all } t' \geq t \end{array} \right\} \tag{2.8}$$

**Lemma 2.** ***Good Estimation is likely***

*Under fast local $\varepsilon$-effective estimators and fast global $\delta$-effective estimators, then for sufficiently large $t$,*

$$\mathbb{P}(GE^c_t) \leq O\left(\frac{1}{t}\right). \tag{2.9}$$

Proof of this lemma is given in Section 2.9.2

This result warrants some remarks both on its strength and its shortcomings: as to its strengths, this result holds under *any* policy. If the estimators are good, there will always be *some* subset of actions over which not only are the transitions and optimality values well-estimated, but as a result, the optimal actions for this restricted problem are known. If we could additionally guarantee that the restricted problem $\Pi_t$ converges to the true problem $\Pi^*$, this would guarantee (probabilistic) discovery of the truly optimal actions.

It is worth noting as well - the growing confidence in the restricted optimal actions does not necessarily translate to discovery of the unrestricted optimal actions. Consider the hypothetical

policy that always takes the same (sub-optimal) action in every state. Under this policy, the single actions taken will be extremely over-sampled, the transitions for these actions well-estimated, and the optimality values well-estimated as well. Indeed, we can easily determine the specific optimal actions for this restricted problem (as there would be only one available action for each state under the restricted problem). However, this policy would fail to discover the truly optimal actions, as it performs no experimentation at all. But, under a guarantee of sufficient exploration and exploitation, we can guarantee that $\hat{\underline{v}}_t \to \underline{v}^*$, and ultimately that for any sub-optimal action $a$, we have

$$\mathbb{E}[T^a(n)] = O(\ln n). \tag{2.10}$$

This will be discussed in Section 2.5, but we first need to establish some regularity results about the estimation process.

## 2.4 Preliminaries

Before we discuss the main portion of this chapter, namely the optimal balancing of exploration and exploitation, it is useful to establish some general notation and results related to the regularity of the estimation of an MDP.

### 2.4.1 Regularity

Recall from the discussion in Section 2.2 that only optimal actions should be sampled more than $b(t)$ times, and thus stay in the over-sampled set $A_t$. It is convenient to define the following event.

**Definition 7.** ***Retainment and Release Events, $RE_t$***

*Let the retainment and release event, $RE_t$ denote that for all current and future time, believed optimal actions are retained in the over-sampled set and believed sub-optimal actions are not taken. Formally,*

$$RE_t = \left\{ \hat{O}^*_{t'} \subset A_{t'+1}, \pi_{t'} \notin A_{t'} \setminus \hat{O}^*_{t'} \text{ for all } t' \geq t \right\} \tag{2.11}$$

A good algorithm should guarantee that this event occurs with high probability. Indeed, this will be condition 3 explicitly defined in Section 2.6.

In order to be able to discuss the local visits to a state over some interval in global time, we define a mesh that represents the global times over the interval at which the state has been visited. Formally,

**Definition 8.** ***Mesh over an Interval***

*The mesh of state x over the time interval $I = [I^-, I^+]$ is the ordered set,*

$$M_x(I) = \{m \mid X_m = x, I^- \leq m \leq I^+\} \tag{2.12}$$

In this way $M_x(I)$ represents the ordered global times over the interval $I$ that we have visited state $x$. Having thus established a notional way to refer to the visits to a particular state, we now address how often a state is visited over an interval.

**Definition 9.** ***Frequent Visits Events, $FV_t$***

*Let the event frequent visits, $FV_t$ denote the event that over any interval I with lower bound $I^- = O(t + t/b(t) \geq t$ and width $(O(t/b(t))$, for any state x, we have visited that state linearly often. Formally,*

$$FV_t = \left\{ T_x(I^+) - T_x(I^-) \geq \rho \cdot O\left(\frac{t}{b(t)}\right) \right\} \tag{2.13}$$

Results of [1] that this event is occurs with high probability. This claim is made more precise in Section 2.9.3

Next, recall the definition of the Good Estimation event $\mathrm{GE}_t$, Definition 6 and the fact that this event also occurs with high probability, Lemma 2.

For simplicity, we combine all these events into one regularity event, $\mathrm{R}_t$.

**Definition 10.** ***Regularity Events, $R_t$***

*Let the event regularity, $R_t$ denote the event that all the events $GE_t, RE_t$, and $FV_t$ hold. That is,*

$$R_t = \{GE_t, RE_t, FV_t\} \tag{2.14}$$

So under regularity $\mathrm{R}_t$, for all future time after $t$, we have good estimates for any minimally sampled action, retainment and release is in effect, and we have frequent visits to every state. The conditions that we will place on the algorithms will guarantee that with high probability, the regularity event will occur. This claim is made precise and proven in Section 2.9.3.

### 2.4.2 MDP Ordering

For the following discussion, it is convenient to introduce an ordering relation for MDPs. For action sets $A', A'' \subset A$, systems of Bellman's equations $\Pi' = \Pi(A', P)$, $\Pi'' = \Pi(A'', P)$, with respective solutions $(\phi^{*\prime}, \underline{v}^{*\prime})$, $(\phi^{*\prime\prime}, \underline{v}^{*\prime\prime})$, and respective optimal action sets $O^{*\prime}$ , $O^{*\prime\prime}$

- if $\phi^{*\prime} < \phi^{*\prime\prime}$, then take $\Pi' < \Pi''$
- if $\phi^{*\prime} = \phi^{*\prime\prime}$ and $O^{*\prime} \subset O^{*\prime\prime}$, then take $\Pi' = \Pi''$

Thus, two MDPs are taken to be equal if and only if they yield equivalent optimal $\phi$s, and share at least one optimal policy. Note that we have, by the uniqueness of the solution to Bellman's equations that

$$\Pi' = \Pi'' \implies \underline{v}^{*\prime} = \underline{v}^{*\prime\prime} \tag{2.15}$$

This provides a convenient language for discussing the progression $\{\Pi_t, \Pi_{t+1}, \ldots\}$. In fact this progression can be achieved using the notion of improving actions.

**Definition 11.** ***Improving Action***

*For an MDP $\Pi'$ restricted to some subset of actions $A' \subset A$, with solution $(\phi^{*\prime}, \underline{v}^{*\prime})$, an improving action $a^+ \in A \setminus A'$ is such that,*

$$r_{x,a^+} + \sum_{y \in S} p^{a^+}_{x,y} v^{*\prime}{}_y > \max_{a \in A'} \left[ r_{x,a} + \sum_{y \in S} p^{a}_{x,y} v^{*\prime} \right]. \tag{2.16}$$

That is, adding the action $a^+$ to the restricted action set strictly improves the optimality values $(\phi^*, \underline{v}^*)$. Using our language of ordering MDPs, we can write $\Pi' = \Pi(A', P) < \Pi'' = \Pi(A' \cap a^+, P)$. Incidentally, this notion is the basis of policy iteration schemes for solving known MDPs.

## 2.5 Optimal Balancing of Exploration and Exploitation

As was mentioned in the end of Section 2.2, the key component of a good algorithm is managing the exploration and exploitation within the middle portion of Figure 2.1. In this section we present two conditions that if an algorithm can satisfy under relatively simple circumstances, it will optimally balance these two competing interests.

### 2.5.1 Sufficient Exploration

The goal of *exploration* is to discover actions that perform better than those currently available. In particular, since over-sampled actions and the restricted MDP are highly likely to be well-understood, as in Lemma 2, better actions must be found *outside* the over-sampled set. If the transition law was known, then these actions could be computed as the basis of a policy-iteration based scheme for computing the unrestricted optimal policy. However, the transition law is *not* known, and thus at any time $t$ the improving actions may be unknown. Sufficient exploration should guarantee that improving actions will be discovered over time. This will lead to overall improvement of the optimality values for the restricted problem and convergence to and discovery of the true, unrestricted, optimal actions. In an effort to make this as simple as possible for a verifier to satisfy this condition we attempt to quantify this in its purest form.

Consider a growing interval of time, under which things behave regularly, our MDP problem is constant and sub-optimal, where there is a single action outside the over-sampled set, and that action is an improving action. We want to guarantee that the probability of *never* taking such an action (i.e. never exploring when we should) decreases over time. Thus, we may quantify sufficient exploration in the following way:

**Sufficient Exploration**

For sufficiently large $t$, for any interval $I = [I^-, I^+]$ with a lower bound $I^- = O(t + t/b(t))$ and width $O(t/b(t))$, under the regularity conditions $\mathrm{R}_{I^-}$, $\Pi_{t'} = \Pi < \Pi^*$ for all $t' \in I$, for at least one improving action $a^+$ in state $x_{a^+}$, and where for every action $a$ except $a^+$, $T^a_{x^+_a}(t') \geq b(I^+)$, we have,

$$\mathbb{P}\left(\pi_m \neq a^+ \text{ for all } m \in M_{x_{a^+}}(I)\right) \leq O\left(\frac{1}{I^+ b(I^+)}\right). \tag{2.17}$$

This property guarantees (with high probability) that if there is an improving action to be found, over a growing time interval we are performing at least some exploration to find it. Indeed, this will be condition 4, discussed more in Section 2.6.

### 2.5.2 Sufficient Exploitation

The goal of *exploitation* is to leverage our knowledge to gather the maximum rewards. Put another way, if we know enough about all action to determine that it is sub-optimal we should stop taking it.

Similarly to the exploration condition above, consider a growing interval of time, under which things behave regularly, but now we have a constant but *optimal* MDP. We have thus correctly identified the optimal actions. To ensure that we exploit our correct identification of the optimal actions, we want to guarantee that the probability of taking a sub-optimal action decreases over time. Thus, we may quantify sufficient exploitation in the following way:

**Sufficient Exploitation**

For sufficiently large $t$, for any interval $I = [I^-, I^+]$ with a lower bound $I^- = O(t + t/b(t))$ and width $O(t/b(t))$, under the regularity conditions $\mathrm{R}_{I^-}$, $\Pi_{t'} = \Pi^*$ for all $t' \in I$, for any sub-optimal action $a \notin O^*(X_{I+})$ with, $\mathrm{MS}_{I^+,a}$, we have,

$$\mathbb{P}(\pi_{I+} = a) \leq O\left(\frac{1}{I+}\right). \tag{2.18}$$

This property guarantees that if we have sampled an action enough to determine it is sub-optimal (minimally sampled), the probability that we continue to take it decreases over time. Indeed, this will be condition 5, discussed more in Section 2.6.

## 2.6 Sufficient Conditions

In this section we provide an explicit statement of each of the conditions presented in this paper, give some intuition to aid the reader, and explain how a verifier might use the specifics of these conditions to bound their own algorithm's performance.

The first two conditions are related to the efficiency and accuracy of our estimators as discussed in Section 2.3. Firstly, we must have fast local (transition) $\varepsilon$-effective estimators. Let $\hat{P}_t$ be an estimate of $P$, constructed from whatever information is available at time $t$, then the condition is,

**Condition 1.** $\hat{P}_t$ ***is a fast local*** $\varepsilon$***-effective estimators***

*For all sufficiently small* $\varepsilon > 0$*, for sufficiently large local time* $k = T^a(t)$*, if the estimator depends solely on local time k,*

$$\mathbb{P}\left(||\underline{\hat{P}}_x^a(t) - \underline{P}_x^a|| > \varepsilon \text{ and } T^a(t) = k\right) \leq O\left(\frac{1}{e^k}\right). \tag{2.19}$$

*If the estimator depends on global time, we require,*

$$\mathbb{P}\left(\exists t' \geq t : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') = k\right) \leq O\left(\frac{1}{e^k}\right) \tag{2.20}$$

Intuitively this condition guarantees that after gathering sufficient data, we are able to quickly and accurately estimate the transition probabilities. This can be achieved by a direct tabular form of estimation. Note however, that this condition allows for other potential estimation techniques which may, for example, take into account a priori knowledge of the MDP structure, or use non-tabular methods for example as in a neural network approach.

Secondly, we must have fast global (restricted MDP) $\delta$-effective estimators. Let $(\hat{\phi}_t, \underline{\hat{v}}_t)$ be an estimated solution to the restricted MDP, $\Pi_t$, then the condition is,

**Condition 2.** $(\hat{\phi}_t, \underline{\hat{v}}_t)$ ***is a fast global*** $\delta$***-effective estimators****,*

*For all sufficiently small* $\delta > 0$*, for sufficiently large global time t,*

$$\mathbb{P}\left(||\hat{\Pi}_t - \Pi_t|| > \delta\right) \leq O\left(\frac{1}{e^t}\right). \tag{2.21}$$

In a similar way to the condition above, this guarantees that if we have collected enough data, we can quickly and accurately estimate the state values of the restricted MDP $\Pi_t$. That is, considering only the actions which have been over-sampled, can we at least accurately estimate the state values of that version of the MDP? This can be accomplished directly, namely by explicitly solve the restricted MDP using our estimates of the transition probabilities. Note however, that this condition allows for other potential estimation techniques which may not require explicitly solving the MDP, may allow the incorporation of a priori knowledge, etc..

The third condition, which is the first condition on how the algorithm actually takes actions, is that the algorithm must retain what it is almost sure is optimal and must release what it is

sure is sub-optimal. Recall the discussion motivating this particular form in Section 2.2, the related Definition 7, and the definition of the good estimation event, $\mathrm{GE}_t$, Definition 6.

**Condition 3.** ***Retainment and Release****,*

*For sufficiently large global time $t$, under the good estimation event $GE_t$, our best guess at the optimal is taken at least $b(t)$ times (Retainment) and our best guess at the sub-optimals are taken at most $b(t)$ times (Release). Formally, Retainment,*

$$\mathbb{P}\left(\hat{O}_t^* \not\subset A_{t+1}\right) \leq \frac{1}{e^t} \tag{2.22}$$

*and Release,*

$$\mathbb{P}\left(\pi_t \in A_t \setminus \hat{O}_t^*\right) \leq \frac{1}{e^t} \tag{2.23}$$

Intuitively, this condition can be thought of as follows Ensure the optimal action stays over-sampled (retainment) and don't over sample clearly sub-optimal actions (release). We know that if we want to obtain minimal regret the **only** actions that should be taken more than logarithmically often are the optimal actions and we only need to take sub-optimal actions logarithmically often (up to a constant factor) to determine that they are sub-optimal. Thus, if we have taken an action more than logarithmically often (and thus we as the controller are guaranteed to have a good estimate of it) if it seems sub-optimal, don't take it anymore (release) and if it seems optimal, keep our estimate of it accurate (retainment).

How can a verifier ensure that their algorithm satisfies this? Firstly, this can easily be satisfied by fiat in any policy. Take your arbitrary MDP policy layer these two rules on top. Note that this is not in general necessary. Under the good estimates event $\mathrm{GE}_t$, everything we think we know about $\Pi_t$ is accurate, thus any good algorithm should, with high probability not take clearly sub-optimal actions and at least super logarithmically often it should take a clearly optimal action.

The last two conditions pertain to balancing exploration and exploitation within the "logarithmically often" band, introduced in Section 2.2 and made more explicit in Section 2.5. The fourth condition, is that the algorithm must be explore sufficiently often to ensure it will eventually discover the true optimal action.

**Condition 4.** ***Sufficient Exploration***

*For sufficiently large $t$, for any interval $I = [I^-, I^+]$ with a lower bound $I^- = O(t + t/b(t))$ and width $O(t/b(t))$, under the regularity conditions $R_{I^-}$, $\Pi_{t'} = \Pi < \Pi^*$ for all $t' \in I$, for at least one improving action $a^+$ in state $x_{a^+}$, and where for every action $a$ except $a^+$, $T^a_{x_a^+}(t') \geq b(I^+)$, we have,*

$$\mathbb{P}\left(\pi_m \neq a^+ \text{ for all } m \in M_{x_{a^+}}(I)\right) \leq O\left(\frac{1}{I^+ b(I^+)}\right). \tag{2.24}$$

Intuitively, this condition states that if all actions except one are over-sampled, and thus we know they are well estimated, and that one action is in truth an improving action, there shouldn't be long stretches of time where we don't take it. In other words, the probability of not exploring the one action that will yield improvement at all, over longer and longer stretches of time should decrease.

How can a verifier ensure that their algorithm satisfies this and why is it relatively straightforward? Notice that the improving actions set is constant, the estimated values of states, $v_x$ are constant within some small $\delta$, there are linear visits to a state with improving actions, every action except the improving action is over-sampled and all actions taken at least $g^a(I^-)$ are well-estimated. Thus the only reason we should be not taking the improving action is because it has not been sampled enough to obtain good estimates. A verifier must now show that the probability of going for long stretches of time without ever taking the under sampled improving action decreases at the appropriate rate.

The fifth, and last condition is that the algorithm must exploit often enough to guarantee that it will maximize its expected average reward or equivalently, incur minimal regret.

**Condition 5.** ***Sufficient Exploitation***

*For sufficiently large $t$, for any interval $I = [I^-, I^+]$ with a lower bound $I^- = O(t + t/b(t))$ and width $O(t/b(t))$, under the regularity conditions $R_{I^-}$, $\Pi_{t'} = \Pi^*$ for all $t' \in I$, for any sub-optimal action $a \notin O^*(X_{I+}$ with, $MS_{I^+,a}$, we have,*

$$\mathbb{P}(\pi_{I+} = a) \leq O\left(\frac{1}{I+}\right). \tag{2.25}$$

Intuitively this condition states that if we have indeed discovered the optimal actions, and have minimally-sampled a sub-optimal action and thus have an accurate estimation for its value, the probability that we take this sub-optimal action decreases over time.

How can a verifier ensure that their algorithm satisfies this and why is it relatively straightforward? Since $\Pi_{t'} = \Pi^*$ we can assume that all optimal actions have been taken at least $b(t)$ times. Recall that any action we take was taken at least $g^a(I^+)$ times. In particular for the sub-optimal action $a$ we have $\mathrm{MS}_{I^+,a}$ and thus transition probability estimates are good, so why did we take it?

## 2.7 Main Theorem and Proof

These conditions are enough to guarantee that an algorithm has at most asymptotic logarithmic regret.

**Theorem 1.** *Given **Fast Local $\varepsilon$-effective Estimators**, **Fast Global $\delta$-effective Estimators**, and a policy that satisfies **Retainment and Release**, **Sufficient Exploration**, and **Sufficient Exploitation**, we have at most asymptotic logarithmic regret, i.e. For any sub-optimal action a, in state x,*

$$\mathbb{E}[T^a(n)] = \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\{\pi_t = a\}\right] = O(\ln n) \tag{2.26}$$

*Proof.* We first split this expectation in to two events, where action $a$ has been minimally sampled $(\mathrm{MS}_{t,a})$, and where it has not been minimally sampled $\left(\mathrm{MS}^c_{t,a}\right)$.

$$\begin{aligned}
\mathbb{E}[T^a(n)] &= \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\{\pi_t = a\}\right] \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\{\pi_t = a, \mathrm{MS}_{t,a}\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}^c_{t,a}\right\}\right] \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\{\pi_t = a, \mathrm{MS}_{t,a}\}\right] + O(\ln(n))
\end{aligned} \tag{2.27}$$

Where the last line follow directly from Lemma 1.

We continue splitting the first event in to two events, regular$\left(\mathrm{R}_{t/2}\right)$ , and non-regular

$\left(\mathrm{R}^c_{t/2}\right)$.[2]

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}\right\}\right] \\
=&\mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}^c_{t/2}\right\}\right] \\
\leq&\mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\mathrm{R}^c_{t/2}\right\}\right] \\
\leq&\mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] + O(\ln(n))
\end{aligned} \tag{2.28}$$

Where this time the last line follows directly from Lemma 6. It remains to show that the first term, sub-optimal activations of a minimally sampled action, under regular circumstances, happens at most logarithmically often. Focusing on that term,

$$\mathbb{E}\left[\sum_{t=0}^{n-1}\mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] = \sum_{t=0}^{n-1}\mathbb{P}\left(\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right) \tag{2.29}$$

We will proceed by bounding the probability $\mathbb{P}\left(\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right)$ for sufficiently large $t$. Consider the interval $[t/2, t]$. Recalling the definition of $\mathrm{R}_{t/2}$ (Definition 10), we have good estimation over the entire interval, i.e. $\mathrm{GE}_{t/2}$. In particular, actions that are estimated to be optimal for the estimated restricted MDP, $\hat{\Pi}_{t'}$ are indeed optimal for the true restricted MDP $\Pi_{t'}$. More precisely,

$$\hat{O}^*_{t'} \subset O^*_{t'} \text{ for all } t' \geq t/2 \tag{2.30}$$

Under $\mathrm{R}_{t/2}$ we also have retainment, $\mathrm{RE}_{t/2}$. That is, for all $t' \geq t/2$, actions that are optimal to the restricted problem are kept from leaving the over-sampled set. Thus, if $\phi_{t'}$ is the restricted optimality value at time $t'$, there is a set of actions in $\hat{O}^*_{t'}$ capable of achieving this optimality value. Retainment ensures that $\hat{O}^*_{t'} \subset A_{t'+1}$, hence there are actions in the over-sampled set at time $t'+1$ capable of achieving $\phi_{t'}$ as well. Since $\phi_{t'}$ is at least achievable with over-sampled actions at time $t'$, we must have that $\phi_{t'} \leq \phi_{t'+1}$.

[2] The $t/2$ is a somewhat arbitrary value. As will become apparent later in the proof, we will be looking backwards from $t$ over some interval with lower bound, $I^- = O(t)$. We also technically are considering $\lfloor t/2 \rfloor$ instead of $t/2$ but we suppress this detail so as not to obscure the important portions of the proof.

We can summarize this with the following:

$$\Pi_{t/2} \leq \Pi_{t/2+1} \leq \ldots \leq \Pi_t \tag{2.31}$$

Hence we have monotonically non-decreasing (in terms of the optimality values) sequence of restricted problems. The exploration condition will guarantee sufficient exploration to push improving actions into the over-sampled set, resulting in a steady increase in the optimality values of the restricted problem, until the unrestricted optimum is discovered. Then the exploitation condition will guarantee that we are taking this truly optimal action sufficiently often so as to incur at most logarithmic regret.

Let $D$ be the number of possible unequal $\Pi_{t'}$, looking at every possible restricted action set of $A$. Since $D$ must be finite, we can split the interval $[t/2,t]$ into $D$ intervals of equal width, $I_1, I_2, \ldots I_D$, where the lower bound of the $i$th interval is $I_i^- = t/2 + (i-1)\frac{t}{2}\frac{1}{D}$, upper bound $I_i^+ = t/2 + i\frac{t}{2}\frac{1}{D}$, and width $|I_i| = \frac{t}{2}\frac{1}{D}$. We also note that $I_1^- = t/2$ and $I_D^+ = t$.

Now, for sufficiently large $t$ such that $\frac{t}{2} \geq D$, there must be at least one interval $I_i$ over which the restricted problem is constant. That is, $\Pi_{t'} = \Pi_{I_i^-}$ for $t' \in I_i$. To see this, recall that there are at most $D$ possible values of $\Pi_{t'}$, we have a monotonic non-decreasing sequence of $\Pi_{t'}$, and we have $D$ different intervals. For an interval to not be constant the restricted MDP must increase at some point in that interval, that is $\Pi_{I_i^-} < \Pi_{I_i^+}$. So we have at most $D-1$ increases (remember that we start with some $\Pi_{t/2}$) but we have $D$ intervals. Thus, there must be at least one interval with no increase and therefore constant $\Pi_{t'}$.

We can further split the event of interest, $\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}$ into two events[3]. Let the bad exploitation event $\mathrm{F}_t$ denote the event that the last interval was the optimal MDP, but we still took the sub-optimal action $a$ at the last point. That is,

$$\begin{aligned} \mathrm{F}_t &= \left\{\pi_t = a, \mathrm{MS}_{t,a}, \pi_{D^+} \notin O^*(X_{D^+}), \mathrm{R}_{t/2}, \Pi_{t'} = \Pi_{I_D^-} = \Pi^* \text{ for } t' \in I_D\right\} \\ &\subseteq \left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}, \Pi_{t'} = \Pi_{I_D^-} = \Pi^* \text{ for } t' \in I_D\right\} \end{aligned} \tag{2.32}$$

Where the second line follows, by noting that $a$ was assumed to be a sub-optimal action and $D^+ = t$.

[3]These events are not mutually exclusive but we will still be able to use them to get an upper bound on the probability.

Let the bad exploration event $\mathrm{S}_{t,i}$ denote the event that there exists some interval $I_i$ with a constant sub-optimal restricted MDP. That is,

$$\mathrm{S}_{t,i} = \left\{ \pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}, \Pi_{t'} = \Pi_{I_i^-} < \Pi^* \text{ for } t' \in I_i \right\} \tag{2.33}$$

In this manner we have,

$$\mathbb{P}\left(\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right) \leq \mathbb{P}\left(\mathrm{F}_t\right) + \sum_{i=1}^{D} \mathbb{P}\left(\mathrm{S}_{t,i}\right) \tag{2.34}$$

First we turn to bounding the probability of the first event, $\mathbb{P}\left(\mathrm{F}_t\right)$. Noting that we have regularity, $\mathrm{RE}_{t/2}$, $\Pi_{t'} = \Pi^*$ (constant optimal MDP) over the interval $I_D$ with lower bound $O(t)$ and width $O(t)$, and $a$ was minimally sampled $\mathrm{MS}_{t,a}$. This is exactly what is assumed in the exploitation condition 5. Thus we can bound this probability by,

$$\mathbb{P}\left(\mathrm{F}_t\right) \leq O\left(\frac{1}{I_D^+}\right) = O\left(\frac{1}{t}\right) \tag{2.35}$$

Next we look at the bad exploration event $\mathrm{S}_{t,i}$, where we have an interval $I_i$ for which we had a constant sub-optimal restricted MDP. As noted previously in Eq. (2.15), this implies that over this interval the $\underline{h}_t$ are constant and thus from Definition 11, the set of improving actions for each state is non-empty and constant for at least one state, since by assumption the optimality values are sub-optimal. Let $a^+$ be one such improving action and $x_{a^+}$ be the state with that improving action.

Divide the interval $I_i$ into $|A|b(I_i^+)$ sub-intervals of equal width, $I_{i_1}, I_{i_2}, \ldots I_{i_{|A|b(I_i^+)}}$. The lower bound for the $k$th interval is,

$$\begin{aligned} I_{i_k}^- &= I_i^- + (k-1) I_i^+ \frac{1}{|A|b(I_i^+)} \\ &= \left(\frac{t}{2} + (i-1)\frac{t}{2D}\right) + (k-1)\left(\frac{t}{2} + i\frac{t}{2D}\right) \frac{1}{|A|b\left(\frac{t}{2} + i\frac{t}{2D}\right)} \\ &= O\left(t + \frac{t}{b(t)}\right) \end{aligned} \tag{2.36}$$

the upper bound is,

$$\begin{aligned} I_{i_k}^+ &= I_i^- + k I_i^+ \frac{1}{|A|b(I_i^+)} \\ &= \left(\frac{t}{2} + (i-1)\frac{t}{2D}\right) + k\left(\frac{t}{2} + i\frac{t}{2D}\right) \frac{1}{|A|b\left(\frac{t}{2} + i\frac{t}{2D}\right)} \\ &= O\left(t + \frac{t}{b(t)}\right) \end{aligned} \tag{2.37}$$

and the width is,

$$\begin{aligned} |I_{i_k}| &= I_i^+ \frac{1}{|A| b(I_i^+)} \\ &= \left(\frac{t}{2} + i\frac{t}{2D}\right) \frac{1}{|A| b\left(\frac{t}{2} + i\frac{t}{2D}\right)} \\ &= O\left(\frac{t}{b(t)}\right). \end{aligned} \tag{2.38}$$

In order to be able to discuss the visits to this state over each sub-interval recall the definition of the mesh for a state $x$ over an interval $I$, $M_x(I)$ (Definition 8). By the frequent visits event $\mathrm{FV}_{t/2}$ of the regularity condition $\mathrm{R}_{t/2}$, $|M_{x_{a^+}}(I_{i_k})| \geq \rho |I_{i_k}|$. That is, we have visited the state $x_{a^+}$ and taken an action in $A(x_{a^+})$, at least $\rho O\left(\frac{t}{b(t)}\right) \geq 1$ times in each sub-interval.

Now, we must have at least one such sub-interval for which the improving action is never taken, all other actions are over-sampled, and only the believed to be optimal actions $a^* \in \hat{O}^*_{t'}$ are taken.

To see this we argue as follows. Firstly, any sub-optimal action that is in the over-sampled set $A_{t'} \setminus \hat{O}^*_{I_i^-}$ will not be taken by the "Release" condition of the Retainment and Release event. Any sub-optimal and thus non-improving action that would be outside the over-sampled set $T^a(I_i^-) < b(I_i^+)$ by the end of the interval if not taken, was taken at most $b(I_i^+)$ times. This is because, if at any point in $I_i$ it was taken $b(I_i^+)$ times, it would enter the over-sampled set and stay there, not taken, for the rest of the interval. Lastly any improving action $a^+$ must have been taken less than $b(I_i^+)$ times, because otherwise it would enter the over-sampled set and improve the restricted MDP $\Pi_{I_i^-}$ (by the good estimation event, $\mathrm{GE}_{t/2}$ and the retainment event, $\mathrm{RE}_{t/2}$ of the regularity condition $\mathrm{R}_{t/2}$) which we have already *assumed to be constant*. Recalling that we have $|A| b(I_i^+)$ sub-intervals, with at least 1 visit (and thus 1 action taken) to state $x_{a^+}$ in each, there must be at least one sub-interval $k$, over which all actions except improving actions are in the over-sampled set, and the only action taken on the mesh of that interval, $M_{x_{a^+}}(I_{i_k})$ is believed optimal actions in $\hat{O}^*_{I_i^-}$.

Next we argue that the probability of never taking the improving action over a sub-interval is small. The probability of never taking the improving action over some sub-interval $I_{i_k}$ is given by,

$$\mathbb{P}\left(\pi_m \neq a^+ \text{ for all } m \in M_{x_{a^+}}(I_{i_k})\right). \tag{2.39}$$

Recall that we are assuming the regularity condition, $\mathrm{R}_{t/2}$, a constant sub-optimal restricted MDP, $\Pi_{t'} = \Pi_{I_i^-} < \Pi^*$, all other actions are over-sampled, only the believed to be optimal actions in $\hat{O}^*_{t'}$ are taken, and we are looking over an interval of lower bound $O\left(t + \frac{t}{b(t)}\right)$ and width $O\left(\frac{t}{b(t)}\right)$. This is exactly what we assumed in the exploration condition 4. Thus we can bound this probability like so,

$$\mathbb{P}\left(\pi_m \neq a^+ \text{ for all } m \in M_{x_{a^+}}(I_{i_k})\right) \leq O\left(\frac{1}{I^+_{i_k} b(I^+_{i_k})}\right) \tag{2.40}$$

Summing over all $|A|b(I_i^+)$ sub-intervals,

$$\begin{aligned}
\mathbb{P}\left(\mathrm{S}_{t,i}\right) &\leq \sum_{k=1}^{|A|b(I_i^+)} \mathbb{P}\left(\pi_m \neq a^+ \text{ for all } m \in M_{x_{a^+}}(I_{i_k})\right) \\
&\sum_{k=1}^{|A|b(I_i^+)} O\left(\frac{1}{I^+_{i_k} b(I^+_{i_k})}\right) \\
&\leq |A|b(I_i^+) O\left(\frac{1}{I_i^+ b(I_i^+)}\right) \\
&= O\left(\frac{1}{I_i^+}\right)
\end{aligned} \tag{2.41}$$

and so,

$$\begin{aligned}
\mathbb{P}\left(\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right) &\leq \sum_{i=1}^{D} \mathbb{P}\left(\mathrm{S}_{t,i}\right) + \mathbb{P}\left(\mathrm{F}_t\right) \\
&\leq \sum_{i=1}^{D} O\left(\frac{1}{I_i^+}\right) + \mathbb{P}\left(\mathrm{F}_t\right) \\
&\leq D \cdot O\left(\frac{1}{I_1^+}\right) + \mathbb{P}\left(\mathrm{F}_t\right) \\
&= O\left(\frac{1}{t}\right) + \mathbb{P}\left(\mathrm{F}_t\right)
\end{aligned} \tag{2.42}$$

Thus,

$$\begin{aligned}
\mathbb{P}\left(\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right) &\leq O\left(1/t\right) + \mathbb{P}\left(\mathrm{F}_t\right) \\
&\leq O\left(1/t\right) + O\left(1/t\right) \\
&= O\left(1/t\right)
\end{aligned} \tag{2.43}$$

Putting it all together,

$$
\begin{aligned}
\mathbb{E}\left[T^a(n)\right] &= \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a\right\}\right] \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}^c\right\}\right] \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}\right\}\right] + O(\ln(n)) \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}^c\right\}\right] + O(\ln(n)) \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\mathrm{R}_{t/2}^c\right\}\right] + O(\ln(n)) \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right\}\right] + O(\ln(n)) \\
&\leq \sum_{t=0}^{n-1} \mathbb{P}\left(\pi_t = a, \mathrm{MS}_{t,a}, \mathrm{R}_{t/2}\right) + O(\ln(n) \\
&\leq \sum_{t=0}^{n-1} \sum_{i=0}^{D} \mathbb{P}\left(\mathrm{S}_{t,i}\right) + \sum_{t=0}^{n-1} \mathbb{P}\left(\mathrm{F}_t\right) + O(\ln(n)) \\
&\leq \sum_{t=0}^{n-1} O\left(\frac{1}{t}\right) + \sum_{t=0}^{n-1} O\left(\frac{1}{t}\right) + O(\ln(n)) \\
&\leq O(\ln(n))
\end{aligned}
\tag{2.44}
$$

and we have finally the main result,

$$
\mathbb{E}\left[T^a(n)\right] \leq O\left(\ln n\right) \tag{2.45}
$$

□

## 2.8 Future Work

The most immediate extension of this work would be to apply these conditions to algorithms that are in actual use to demonstrate their effectiveness. It is also of import to extend these results to the case with unknown rewards or additional apriori knowledge about the structure of states, rewards, and transitions.

## 2.9 Event Lemmas and Proofs

The purpose of this section is to hold various lemmas and proofs for events that are ultimately used in the proof of Theorem 1.

### 2.9.1 Minimally Sampled Counts

We restate Lemma 1

**Lemma 1.** ***Minimally Sampled Counts***

*The expected number of times an action a has been taken when it has not been sampled $g^a(t)$ times by time n is less than $g^a(n) = O(\ln(n))$. Formally,*

$$\mathbb{E}\left[\sum_{t=0}^{n} \mathbb{1}\left\{\pi_t = a, MS_{t,a}^c\right\}\right] < g^a(n) = O(\ln(n)) \tag{2.2}$$

*Proof.*

$$\begin{aligned}
&\sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, \mathrm{MS}_{t,a}^c\right\} = \sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, T^a(t) < g_a(t)\right\} \\
&\leq \sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a, T^a(t) < g_a(n)\right\} \text{ (because } g^a(t) < g^a(n)) \\
&= \sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a\right\} \mathbb{1}\left\{T^a(t) < g_a(n)\right\} \\
&\leq \sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a\right\} \sum_{k=0}^{g_a(n)} \mathbb{1}\left\{T^a(t) = k\right\} \\
&= \sum_{t=0}^{n-1} \sum_{k=0}^{g_a(n)} \mathbb{1}\left\{\pi_t = a\right\} \mathbb{1}\left\{T^a(t) = k\right\} \text{ (because } a(b+c) = ab+ac) \\
&= \sum_{k=0}^{g_a(n)} \sum_{t=0}^{n-1} \mathbb{1}\left\{\pi_t = a\right\} \mathbb{1}\left\{T^a(t) = k\right\} \text{ ( because } ab+ac = ba+ca)
\end{aligned} \tag{2.46}$$

We proceed by noting that for a fixed $k$, both events can only simultaneously occur for at most one value of $t$. To see this suppose that $\pi_t = a$ and $T^a(t) = k$, then $T^a(t+1) = T^a(t)+1 = k+1$ and $T^a(t+i)$ for any positive integer $i$ is $\leq k+1$. Thus,

$$
\begin{aligned}
&= \sum_{k=0}^{g_a(n)} \sum_{t=0}^{n-1} \mathbb{1}\{\pi_t = a\} \mathbb{1}\{T^a(t) = k\} \\
&\leq \sum_{k=0}^{g_a(n)} (1) \\
&= g_a(n)
\end{aligned}
\tag{2.47}
$$

Recalling that $g_a(n) = \beta_a \ln(n) = O(\ln(n))$ the proof is complete. □

### 2.9.2 Good Estimation Lemmas and Proofs

Instead of proving Lemma 2 directly. It is convenient to first break it into individual events and then bound the total probability by the sum of the probabilities of each individual event.

**Definition 12.** ***Good Local Estimates***

*Let the event good local estimates, $GLE_t$ denote that for any future time $t' \geq t$ and minimally sampled action a with $\left(MS_{t',a}\right)$, transition estimators are accurate. More precisely,*

$$
GLE_t = \left\{ |\hat{\underline{P}}^a_{x,t'} - \underline{P}^a_x| \leq \varepsilon || \text{ for all } t' \geq t, \text{ for all } a \text{ such that } MS_{t',a} \right\} \tag{2.48}
$$

**Definition 13.** ***Good Global Estimates***

*Let the event good global estimates, $GGE_t$ denote that for any future time $t' \geq t$ our estimates of the restricted MDP $\Pi_t$ are accurate. More precisely,*

$$
GGE_t = \left\{ ||(\hat{\phi}_{t'}, \hat{\underline{h}}_{t'}) - (\phi_{t'}, \underline{h}_{t'})|| \leq \delta \text{ for all } t' \geq t \right\} \tag{2.49}
$$

**Definition 14.** ***Good Optimal Beliefs***

*Let the event good optimal beliefs, $GOB_t$ denote that for any future time $t' \geq t$ actions that we estimate to be optimal for the restricted MDP $\Pi_t$ are indeed optimal. More precisely,*

$$
GOB_t = \left\{ \hat{O}^*_{t'} \subset O^*_{t'} \text{ for all } t' \geq t \right\} \tag{2.50}
$$

Armed with these definitions, the good estimation event $\mathrm{GE}_t = \{\mathrm{GLE}_t, \mathrm{GGE}_t, \mathrm{GOB}_t\}$. Next we state and prove lemmas that bound the probability of each individual event.

**Lemma 3.** *Under fast local $\varepsilon$-effective estimators the following is true.*

$$
\mathbb{P}(GLE^c_t) \leq O\left(\frac{1}{t}\right) \tag{2.51}
$$

*Proof.* Let $G_t$ be the set of actions $a$ such that $\text{MS}_{t,a}$ i.e., $T^a(t) \geq g^a(t)$.

$$\begin{aligned}
\mathbb{P}\left(\text{GLE}_t^c\right) &= \mathbb{P}\left(\exists t' \geq t : ||\hat{P}_{t'}[G_{t'}] - P[G_{t'}]|| > \varepsilon\right) \\
&\leq \mathbb{P}\left(\exists t' \geq t, x, a : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') \geq g^a(t')\right) \\
&\leq \mathbb{P}\left(\exists t' \geq k \geq g^a(t'), x, a : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') = k\right) \\
&\leq \sum_{x \in S} \sum_{a \in A(x)} \sum_{k=\lceil g^a(t) \rceil}^{\infty} \mathbb{P}\left(\exists t' \geq k : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') = k\right)
\end{aligned} \tag{2.52}$$

For sufficiently large $t$, we will have sufficiently large minimum local time $T^a(t) \geq g^a(t)$. If the local estimators depend solely on the local time $k$ then the estimator $\underline{\hat{P}}_x^a(t')$ remains constant for a fixed local time $k$. Let $t'' = \min\{t' \geq t \mid T^a(t') = k\}$, that is the first global time that we have a local time $k$, then by the definition of fast local estimators we have,

$$\mathbb{P}\left(||\underline{\hat{P}}_x^a(t'') - \underline{P}_x^a|| > \varepsilon \text{ and } T^a(t'') = k\right) \leq \frac{1}{e^k}. \tag{2.53}$$

and thus we can bound the existence probability directly.

$$\begin{aligned}
&\mathbb{P}\left(\exists t' \geq t : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') = k\right) \\
&\leq \mathbb{P}\left(||\underline{\hat{P}}_x^a(t'') - \underline{P}_x^a|| > \varepsilon \text{ and } T^a(t'') = k\right) \\
&\leq \frac{1}{e^k}
\end{aligned} \tag{2.54}$$

If however, the local estimators depend on global information, and thus global time, we appeal to the second version of the fast local estimators condition

$$\mathbb{P}\left(\exists t' \geq t : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') = k\right) \leq \frac{1}{e^k} \tag{2.55}$$

In either case we have,

$$\begin{aligned}
&\sum_{x \in S} \sum_{a \in A(x)} \sum_{k=\lceil g^a(t) \rceil}^{\infty} \mathbb{P}\left(\exists t' \geq k : |\underline{\hat{P}}_x^a(t') - \underline{P}_x^a| > \varepsilon \text{ and } T^a(t') = k\right) \\
&\leq \sum_{x \in S} \sum_{a \in A(x)} \sum_{k=\lceil g^a(t) \rceil}^{\infty} \frac{1}{e^k} \\
&\leq |A||S| \sum_{k=\lceil g^a(t) \rceil}^{\infty} \frac{1}{e^k} \\
&= O\left(\frac{1}{t}\right).
\end{aligned} \tag{2.56}$$

Where the last line follows by recalling that $g^a(t) = O(ln(t))$. □

Bounding the probability of the next event,

**Lemma 4.** *Under fast global $\delta$-effective estimators the following is true.*

$$\mathbb{P}\left(GGE_t^c\right) \leq O\left(\frac{1}{t}\right) \tag{2.57}$$

*Proof.*

$$\begin{aligned}
\mathbb{P}\left(\mathrm{GGE}_t^c\right) &= \mathbb{P}\left(\left\{||(\hat{\phi}_{t'}, \hat{\underline{h}}_{t'}) - (\phi_{t'}, \underline{h}_{t'})|| > \delta \text{ for some } t' \geq t\right\}\right) \\
&= \mathbb{P}\left\{||\hat{\Pi}_t - \Pi_t|| > \delta \text{ for some } t' \geq t\right\} \\
&\leq \sum_{t'=t}^{\infty} \mathbb{P}\left\{||\hat{\Pi}_{t'} - \Pi_{t'}|| > \delta\right\} \\
&\leq \sum_{t'=t}^{\infty} \frac{1}{e^{t'}} \\
&\leq O\left(\frac{1}{t}\right)
\end{aligned} \tag{2.58}$$

Where the penultimate line follows directly from the definition of fast global estimators.

□

**Lemma 5.** *Under fast local and fast global estimators the following is true.*

$$\mathbb{P}\left(GOB_t^c\right) \leq O\left(\frac{1}{t}\right) \tag{2.59}$$

*Proof.* To show the result, it suffices to argue that in the event that we have good local and global estimates ($\mathrm{GLE}_t$ and $\mathrm{GGE}_t$), then the actions that realize the maximum for the estimated case will also realize the maximum for the true (restricted) case. In particular, if $\hat{P}_t$ is a fast local estimator, and $(\hat{\phi}_t, \hat{\underline{h}}_t)$ is a fast global estimator, and we have good local and global estimates

(GLE$_t$ and GGE$_t$), we have for any state $x$ and action $a$,

$$\begin{aligned}
&\left|\left(r_{x,a}+\sum_{y\in S}P^a_{x,y}h_{y,t}\right)-\left(r_{x,a}+\sum_{y\in S}\hat{P}^a_{x,y}(t)\hat{h}_{y,t}\right)\right| \\
&=\left|\left(\sum_{y\in S}P^a_{x,y}h_{y,t}-\sum_{y\in S}P^a_{x,y}\hat{h}_{y,t}\right)-\left(\sum_{y\in S}\hat{P}^a_{x,y}(t)\hat{h}_{y,t}-\sum_{y\in S}P^a_{x,y}\hat{h}_{y,t}\right)\right| \\
&=\left|\left(\sum_{y\in S}P^a_{x,y}\left[h_{y,t}-\hat{h}_{y,t}\right]\right)-\left(\sum_{y\in S}\left[\hat{P}^a_{x,y}(t)-P^a_{x,y}\right]\hat{h}_{y,t}\right)\right| \\
&\leq\sum_{y\in S}P^a_{x,y}|h_{y,t}-\hat{h}_{y,t}|+\sum_{y\in S}|\hat{P}^a_{x,y}(t)-P^a_{x,y}||\hat{h}_{y,t}| \\
&\leq\sum_{y\in S}P^a_{x,y}\delta+\sum_{y\in S}\varepsilon\left(|h_{y,t}|+\delta\right) \\
&\leq\delta+\varepsilon\sum_{y\in S}|h_{y,t}|+|S|\varepsilon\delta
\end{aligned} \tag{2.60}$$

Let $D=\delta+\varepsilon\sum_{y\in S}|h_{y,t}|+|S|\varepsilon\delta$, be this gap. From the above, we get that if $\varepsilon$ and $\delta$ are sufficiently small (i.e., the estimates are sufficiently good), then the gap, $D$, between the estimated action value and the true action value can be made arbitrarily small.

For any state $x$, let $V_a$ and $V_s$ be the true values of the maximal action and the second largest action, respectively, i.e.

$$V_a=\max_{c\in A_t(x)}\left[r_{x,c}+\sum_{y\in S}P^c_{x,y}h_{y,t}\right]$$

and

$$V_s=\max_{c\in\{A_t(x)\setminus O^*_t(x)\}}\left[r_{x,c}+\sum_{y\in S}P^c_{x,y}h_{y,t}\right]$$

Let $\hat{V}_a$ be the minimum estimated value of any *truly* optimal action and $\hat{V}_b$ be the largest estimated action value of any *truly* sub-optimal action, i.e.

$$\hat{V}_a=\min_{c\in O^*_t(x)}\left[r_{x,c}+\sum_{y\in S}\hat{P}^c_{x,y}(t)\hat{h}_{y,t}\right]$$

and

$$\hat{V}_b=\max_{c\in\{A_t(x)\setminus O^*_t(x)\}}\left[r_{x,c}+\sum_{y\in S}\hat{P}^c_{x,y}(t)\hat{h}_{y,t}\right]$$

If we can show that $\hat{V}_b<\hat{V}_a$, we will have shown that given Good Estimators, actions estimated to be optimal are indeed optimal ( i.e. if $a\in\hat{O}^*_t$, then we must have $a\in O^*_t$)

Taking $\varepsilon$ and $\delta$ sufficiently small such that, $D<\frac{1}{2}(V_a-V_s)$, and noting that $0<V_a-V_s\leq V_a-V_c$ for any $c\in\{A_t(x)\setminus O^*_t(x)\}$, we have,

$$|\hat{V}_b - V_b| < \frac{1}{2}(V_a - V_s) \leq \frac{1}{2}(V_a - V_b) \implies \hat{V}_b < V_b + \frac{1}{2}(V_a - V_b)$$

and

$$|\hat{V}_a - V_a| < \frac{1}{2}(V_a - V_s) \leq \frac{1}{2}(V_a - V_b) \implies -\hat{V}_a < \frac{1}{2}(V_a - V_b) - V_a \implies V_a - \frac{1}{2}(V_a - V_b) < \hat{V}_a$$

and together we have,

$$\hat{V}_b < V_b + \frac{1}{2}(V_a - V_b) = V_a - \frac{1}{2}(V_a - V_b) < \hat{V}_a.$$

This, combined with the probability bounds on good local and global estimates, $\text{GLE}_t$ and $\text{GGE}_t$ from Lemmas 3 and 4 respectively, yields the result. □

Finally we restate Lemma 2

**Lemma 2.** ***Good Estimation is likely***

*Under fast local $\varepsilon$-effective estimators and fast global $\delta$-effective estimators, then for sufficiently large t,*

$$\mathbb{P}(GE_t^c) \leq O\left(\frac{1}{t}\right). \tag{2.9}$$

*Proof.* Recalling that $\text{GE}_t = \{\text{GLE}_t, \text{GGE}_t, \text{GOB}_t\}$ the result is immediate from Lemmas 3, 4 and 5, by bounding the total probability by the sum of the probabilities of each individual event. □

### 2.9.3 Regularity Lemmas and Proofs

**Lemma 6.** *Under local $\varepsilon$-effective estimators, fast global $\delta$-effective estimators, and the Retainment and Release condition 3 , then non-regularity contributes at most $O(\ln(n))$ expected regret. Formally,*

$$\mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\{R_t^c\}\right] \leq O(\ln(n)) \tag{2.61}$$

Similarly to 2.9.2, instead of proving Lemma 6 directly, it is convenient to bound the probability of individual events and then bound the total probability by their sum.

**Lemma 7.** *If an adaptive policy satisfies the Retainment and Release condition 3 then the Retainment and Release event 7 is likely. Formally,*

$$\mathbb{P}\left(RE_t^c\right) \leq O\left(\frac{1}{t}\right) \tag{2.62}$$

*Proof.*

$$\begin{aligned}
\mathbb{P}\left(\mathrm{RE}_t^c\right) &\leq \mathbb{P}\left(\hat{O}_{t'}^* \subset A_{t'+1} \text{ for some } t' \geq t\right) + \mathbb{P}\left(\pi_{t'} \notin A_{t'} \setminus \hat{O}_{t'}^* \text{ for some } t' \geq t\right) \\
&\leq \sum_{t'=t}^{\infty} \mathbb{P}\left(\hat{O}_{t'}^* \subset A_{t'+1}\right) + \mathbb{P}\left(\pi_{t'} \notin A_{t'} \setminus \hat{O}_{t'}^*\right) \\
&\leq \sum_{t'=t}^{\infty} O\left(\frac{1}{e^{t'}}\right) \\
&\leq O\left(\frac{1}{t}\right)
\end{aligned} \tag{2.63}$$

Where the penultimate line follows directly from the definition the retainment and release condition 3. □

**Lemma 8.** *Frequent visits are likely. Formally,*

$$\mathbb{P}\left(FV_t^c\right) \leq O\left(\frac{1}{t}\right) \tag{2.64}$$

*Proof.* The proof relies on the following proposition on MDPs from [1], given there as Prop. 2 (i):

**Proposition 1.** *There exist $A > 0, \beta > 0$ such that for all $x \in S$, $t \geq |S|$, $\rho > 0$, and all policies $\pi$,*

$$\mathbb{P}\left(T_x(t) \leq \rho t\right) \leq Ae^{-\beta t}. \tag{2.65}$$

This implies that, with high probability, every state is visited roughly linearly often. Consider "restarting" the MDP at time $I^-$. For sufficiently large $t$, $|I| = O(t/b(t))$ will be greater than $|S|$. By Proposition 1 we know that regardless of the policy $\pi$ with high probability we visit any state $a$, $\rho|I|$ times. In particular,

$$\begin{aligned}
\mathbb{P}\left(T_x(I^+) - T_x(I^-) < \rho|I|\right) &< Ae^{-\beta|I|} \\
&= Ae^{-\beta\frac{t}{b(t)}} \\
&\leq O(1/t)
\end{aligned} \tag{2.66}$$

□

Restating Lemma 6

**Lemma 6.** *Under local $\varepsilon$-effective estimators, fast global $\delta$-effective estimators, and the Retainment and Release condition 3 , then non-regularity contributes at most $O(\ln(n))$ expected regret. Formally,*

$$\mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{R_t^c\right\}\right] \leq O(\ln(n)) \tag{2.61}$$

*Proof.* The proof directly follows from the previous lemmas 2, 7, 8.

$$\begin{aligned}
&\mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\mathrm{R}_t^c\right\}\right] \\
&\leq \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\mathrm{GE}_t^c\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\mathrm{RE}_t^c\right\}\right] + \mathbb{E}\left[\sum_{t=0}^{n-1} \mathbb{1}\left\{\mathrm{FV}_t^c\right\}\right] \\
&\leq \sum_{t=0}^{n-1} \mathbb{P}\left(\mathrm{GE}_t^c\right) + \sum_{t=0}^{n-1} \mathbb{P}\left(\mathrm{RE}_t^c\right) + \sum_{t=0}^{n-1} \mathbb{P}\left(\mathrm{FV}_t^c\right) \\
&\leq \sum_{t=0}^{n-1} \left(\mathbb{P}\left(\mathrm{GE}_t^c\right) + \mathbb{P}\left(\mathrm{RE}_t^c\right) + \mathbb{P}\left(\mathrm{FV}_t^c\right)\right) \\
&\leq \sum_{t=0}^{n-1} \left(O\left(\frac{1}{t}\right) + O\left(\frac{1}{t}\right) + O\left(\frac{1}{t}\right)\right) \\
&\leq O(\ln(n))
\end{aligned} \tag{2.67}$$

□

# Chapter 3

# Accelerating the Computation of UCB and Related Indices for Reinforcement Learning

## 3.1 Introduction

The practical use of the asymptotically optimal UCB algorithm (MDP-UCB) of [1] has been hindered [3, 11] by the computational burden of the upper confidence bound indices c.f. Eq. (3.1), that involves the solution of a non-linear constrained optimization problem of dimension equal to the cardinality of the state space of the MDP under consideration. In this chapter we derive an efficient computational method that only requires solving a system of two non-linear equations with two unknowns, irrespective of the cardinality of the state space of the MDP. In addition, we develop a similar acceleration for computing the indices for the MDP-Deterministic Minimum Empirical Divergence (MDP-DMED) developed in [12], that involves solving a single equation of one variable. In Section 3.3 we present these computationally efficient formulations and provide experimental results demonstrating the computational time savings.

The body of the paper is devoted to presenting and discussing four computationally simple algorithm that are either provably asymptotically optimal, or at least appear to be. While no proofs of optimality are presented, the results of numerical experiments are presented demonstrating the efficacy of these algorithm. Proof of optimality for these algorithm will be discussed in future works.

### 3.1.1 Related Work

In addition to the papers upon which the algorithms here are explicitly based, there are many other approaches for adaptively learning MDPs while minimizing expected regret. [28] propose

an algorithm, UCRL2, a variant of the UCRL algorithm of [11], that achieves logarithmic regret asymptotically, as well as uniformly over time. UCRL2, defines a set of plausible MDPs and chooses a near-optimal policy for an optimistic version of the MDP through so called "extended value iteration". This approach, while similarly optimistic in flavor, is sufficiently different than the algorithms presented here that we will not be comparing them directly. The algorithms in this chapter act upon the estimated transition probabilities of actions for only our current state, for a fixed estimated MDP. Specifically, MDP-UCB and OLP inflate the right hand side of the optimality equations by perturbing the estimated transition probabilities for actions in the current state. MDP-DMED estimates the rates at which actions should be taken by exploring nearby plausible transition probabilities for actions in the current state. Finally, MDP-PS obtains posterior sampled estimates, again, only for, the transition probabilities for actions in the current state.

Recently, [29] show that model-based algorithms (which all the algorithms discussed here are), that use 1-step planning can achieve the same regret performance as algorithms that perform full-planning. This allows for a significant decrease in the computational complexity of the algorithms. In particular they propose UCRL2-GP, which uses a greedy policy instead of solving the MDP as in UCRL2, at the beginning of each episode. They find that this policy matches UCRL2 in terms of regret (up to constant and logarithmic factors), while benefiting from decreased computational complexity. The setting under consideration however, is a finite horizon MDP and the regret bounds are in PAC terms [14] and optimal minimax [30]. Further analysis is required to transfer these results to the setting of this paper. Namely, an infinite horizon MDP with bounds on the asymptotic growth rate of the expected regret. A fruitful direction of study would be to examine the relationship between UCRL2-GP, UCRL2, and the algorithms presented here, more closely, paying particular attention to the varying dependencies on the dimensionality of the state space.

[31] analyze and compare the expected regret and computational complexity of PS-type algorithms (PSRL therein) versus UCB-type (OFU therein) algorithms, in the setting of finite horizon MDPs. The PSRL algorithm presented there is similar to MDP-PS here. However, their optimistic inflation or stochastic optimism is done across the MDP as a whole, either over plausible MDPs in the case of OFU, or for a fixed MDP in the PSRL case. By contrast, in this

paper we present non-episodic versions where the inflations are done only for the actions of our current state for a fixed estimated MDP. They also argue therein that any OFU approach which matches PSRL in regret performance will likely result in a computationally intractable optimization problem. Through that lens, the main result of this paper, proving a computationally tractable version of the optimization problem shows that actually a provably asymptotically optimal UCB approach **can** compete with a PS approach both in terms of regret performance as well as computational complexity. A more thorough analysis is required in order to determine what parts of our analysis here, with an undiscounted infinite horizon MDP, can carry over to the finite horizon MDP setting of [31] and [30].

### 3.1.2 Chapter Structure

This chapter is organized as follows. In Section 3.2 we present four simple algorithms for adaptively optimizing the average reward in an unknown irreducible MDP. The first is the asymptotically optimal UCB algorithm (MDP-UCB) of [1] that uses estimates for the MDP and choose actions by maximizing an inflation of the estimated right hand side of the average reward optimality equations. The second (MDP-DMED) is inspired by the DMED method for the multi-armed bandit problem developed in [32, 2] and estimates the optimal rates at which actions should be taken and attempts to take actions at that rate. The third is the Optimistic Linear Programming (OLP) algorithm [3] which is based on MDP-UCB but instead of using the KL divergence to inflate the optimality equations, uses the $L_1$ norm. The fourth (MDP-PS) is based on ideas of greedy posterior sampling that go back to [33] and similar to PSRL in [31]. The main contribution of this chapter is in Section 3.3, where we present the efficient formulations and demonstrate the computational time savings. Various computational challenges and simplifications are discussed, with the goal of making these algorithms practical for broader use. In Section 3.4 we compare the regret performance of these algorithms in numerical examples and discuss the relative advantages of each. While no proofs of optimality are presented, the results of numerical experiments are presented demonstrating the efficacy of these algorithms. Proof of optimality for these algorithms will be discussed in future works, especially in light of the conditions in Chapter 2.

### 3.1.3 Notation

It will be convenient in what is to follow to define the following notation:

$$L(x,a,\underline{p},\underline{v}) = r_x(a) + \sum_{y \in S} p_y v_y.$$

The function $L$ represents the value of a given action in a given state, for a given transition vector—both the immediate reward, and the expected future value of whatever state the MDP transitions into. The value of an asymptotically optimal action for any state $x$ is thus given by $L^*(x,A,P) = L(x,a^*(x,P),\underline{p}_x^{a^*(x,P)},\underline{v}(A,P))$.

In general, for the unknown transition laws case, we have the following bound due to [1], for any uniformly fast policy $\pi$, any sub-optimal action must be sampled at least at a minimum rate. In particular, for a suboptimal action $a$,

$$\liminf_T \frac{\mathbb{E}\left[T_x^a(T)\right]}{\ln T} \geq \frac{1}{\mathbf{K}_{x,a}(P)}.$$

where $\mathbf{K}_{x,a}(P)$ represents the minimal Kullback-Leibler divergence between $\underline{p}_x^a$ and any $\underline{q} \in \Theta$ such that substituting $\underline{q}$ for $\underline{p}_x^a x$ in $P$ renders $a$ the unique optimal action for $x$. Recall, the Kullback-Leibler divergence is given by $\mathbf{I}(\underline{p},\underline{q}) = \sum_{x \in S} p_x \ln(p_x/q_x)$.

This can be interpreted in the following way: for a sub-optimal action, the "closer" the transition law is to an alternative transition law that would make it the best action, the more data we need to distinguish between the truth and this plausible alternative hypothesis, and therefore the more times we need to sample the action to distinguish the truth. Anything less than this "base rate", we risk convincing ourselves of a plausible, sub-optimal hypothesis and therefore incurring high regret when we act on that belief.

Policies that achieve this lower bound, for all $P$, are referred to as asymptotically optimal. Achieving this bound, or at least the desired logarithmic growth requires careful exploration of actions. In the next section, we present four algorithms to accomplish this.

## 3.2 Algorithms for Optimal Exploration

Common reinforcement learning algorithms solve the exploration/exploitation dilemma in the following way: most of the time, select an action (based on the current data) that seems best,

otherwise select some other action. This alternative action selection is commonly done uniformly at random. As long as this is done infrequently, but not too infrequently, the optimal actions and policy will be discovered, potentially at the cost of high regret. Minimizing regret requires careful consideration of which alternative actions are worth taking at any given point in time. The following algorithms are methods for performing this selection; essentially, instead of blindly selecting from the available actions to explore, each algorithm evaluates the currently available data to determine which action is most worth exploring. Each accomplishes this through an exploration of the space of plausible transition hypotheses.

The benefit of this is that through careful exploration, optimal (minimal) regret can be achieved. The cost however, is additional computation. The set of alternative transition laws is large and high dimensional, and can be difficult to work with. In Section 3.3 we show several simplifications, however, that make this exploration practical.

### 3.2.1 A UCB-Type Algorithm for MDPs Under Uncertain Transitions

Classical upper confidence bound (UCB) decision algorithms (for instance as in multi-armed bandit problems, c.f. [34], [35], [36]), approach the problem of exploration in the following way: in each round, given the current estimated transition law, we consider "inflated" estimates of the values of each actions, by finding the best (value-maximizing) plausible hypothesis within some confidence interval of the current estimated transition law. The more data that is available for an action, the more confidence there is in the current estimate, and the tighter the confidence interval becomes; the tighter the confidence interval becomes, the less exploration is necessary for that action. The algorithm we present here is a version of the MDP-UCB algorithm presented in [1].

At any time $t \geq 1$, let $x_t$ be the current (given) state of the MDP. We construct the following estimators:

- Transition Probability Estimators: for each state $y$ and action $a \in A(x_t)$, construct $\hat{P}_t$ based on

$$\hat{p}^a_{x_t,y} = \frac{T^a_{x_t,y}(t) + 1}{T^a_{x_t}(t) + |S|}.$$

  Note the biasing terms (the 1 in the numerator, $|S|$ in the denominator). Including these,

biases the estimated transition probabilities away from 0, so that our estimates $\underline{p}^a_{x_t}$ will be in $\Theta$. Additionally, these guarantee that the above is in fact the maximum likelihood estimate for the transition probability, given the observed data and uniform priors.

- "Good" Action Sets: construct the following subset of the available actions $A(x_t)$,

$$\hat{A}_t = \left\{ a \in A(x_t) : T^a_{x_t}(t) \geq (\ln T_{x_t}(t))^2 \right\}.$$

The set $\hat{A}_t$ represents the actions available from state $x_t$ that have been sampled frequently enough that the estimates of the associated transition probabilities should be "good". In the limit, we expect that sub-optimal actions will be taken only logarithmically, and hence for sufficiently large $t$, $\hat{A}_t$ will contain only actions that are truly optimal. If no actions have been taken sufficiently many times, we take $\hat{A}_t = A(x_t)$ to prevent it from being empty.

- Value Estimates: having constructed these estimators, we compute $\hat{\phi}_t = \phi(\hat{A}_t, \hat{P}_t)$ and $\hat{\underline{v}}_t = \underline{v}(\hat{A}_t, \hat{P}_t)$ as the solution to the optimality equations in Eq. (1.2), essentially treating the estimated probabilities as correct and computing the optimal values and policy for the resulting estimated MDP.

At this point, we implement the following decision rule: for each action $a \in A(x_t)$, we compute the following *index* over the set of possible transition laws:

$$u_a(t) = \sup_{\underline{q} \in \Theta} \left\{ L(x_t, a, \underline{q}, \hat{\underline{v}}_t) : \mathbf{I}(\hat{\underline{p}}^a_{x_t}, \underline{q}) \leq \frac{\ln t}{T_{x_t,a}(t)} \right\}, \tag{3.1}$$

where $\mathbf{I}(\underline{p}, \underline{q}) = \sum_y p_y \ln(p_y/q_y)$ is the Kullback-Leibler divergence, and take action

$$\pi(t) = \arg\max_{a \in A(x_t)} u_a(t).$$

This is a natural extension of several classical KL-divergence based UCB algorithms for the multi-armed bandit problem c.f. [13], [35], [36] taking the view of the $L$ function as the 'value' of taking a given action in a given state, estimated with the current data. In [35], a modified version of the above algorithm is in fact shown to be asymptotically optimal. The modification is largely for analytical benefit however, the pure index algorithm as above shows excellent performance c.f. Figure 3.3. Further discussion of the performance of this algorithm is given in Section 3.4.

An important and legitimate concern to the practical usage of the MDP-UCB algorithm that has been noted in [3] among others, is actually calculating the index in Eq. (3.1). This and other issues are discussed in more depth in Section 3.3, where a computationally efficient formulation is presented. Additionally, in Section 3.4, we highlight beneficial behavior of this algorithm that makes it worth pursuing.

### 3.2.2 A Deterministic Minimum Empirical Divergence Type Algorithm for MDPs Under Uncertain Transitions

In the classical DMED algorithm for multi-armed bandit problems [32], rather than considering (inflated) values for each action to determine which should be taken, DMED attempts to estimate how often each action ought to be taken. Recall the interpretation of [35] given previously, that for any uniformly fast policy $\pi$, for any sub-optimal action $a$ we have

$$\liminf_T \frac{\mathbb{E}[T_x^a(T)]}{\ln T} \geq \frac{1}{\mathbf{K}_{x,a}(P)},$$

where $\mathbf{K}_{x,a}(P)$ measures (via the Kullback-Leibler divergence) how much the transition law for action $a$ would need to be changed to make action $a$ optimal.

DMED proceeds by the following reasoning. If we estimate that the sub-optimal action $a$ is close to being optimal (low $K_{x,a}$), make sure we take it often enough to differentiate between them (ensure $T_x^a$ is high). If, on the other hand, we estimate that the sub-optimal action $a$ is far from being optimal (high $K_{x,a}$), we don't need to take is as often (ensure $T_x^a$ is low). As with the MDP-UCB and OLP algorithms, this requires an exploration of the possible transition laws "near" the current estimated transition law.

In general, computing the function $\mathbf{K}_{x,a}(P)$ is not easy. We consider the following substitute, then:

$$\tilde{\mathbf{K}}_{x,a}(P,\underline{v},a^*) = \inf_{\underline{q}\in\Theta}\left\{\mathbf{I}(\underline{p}_x^a,\underline{q}) : L(x,a,\underline{q},\underline{v}) \geq L(x,a^*,\underline{p}_x^{a^*},\underline{v})\right\}.$$

This is akin to exploratory *policy iteration*. That is, determining, based on the current value estimates, how much modification would produce an improving action.

The function $\mathbf{K}$ measures how far the transition vector associated with $x$ and $a$ must be

perturbed (under the KL-divergence) to make $a$ the optimal action for $x$. The function $\tilde{\mathbf{K}}$ measures how far the transition vector associated with $x$ and $a$ must be perturbed (under the KL-divergence) to make the value of $a$, as measured by the $L$-function, no less than the value of an optimal action $a^*$. As will be shown in Section 3.3, $\tilde{\mathbf{K}}$ may be computed fairly simply, in terms of the root of a single non-linear equation.

In this way, we have the following approximate MDP-DMED algorithm (see [32] and [2] for the multi-armed bandit version of this algorithm).

At any time $t \geq 1$, let $x_t$ be the current state, and construct the estimators as in the MDP-UCB algorithm in Section 3.2.1, $\hat{P}_t$, $\hat{A}_t$, and utilize these to compute the estimated optimal values, $\hat{\phi}_t = \phi(\hat{A}_t, \hat{P}_t)$ and $\underline{\hat{v}}_t = \underline{v}(\hat{A}_t, \hat{P}_t)$.

Let $\hat{a}_t^* = \arg\max_{a \in A(x_t)} L(x_t, a, \underline{\hat{p}}_{x_t}^a, \underline{\hat{v}}_t)$ be the estimated "best" action to take at time $t$. For each $a \neq \hat{a}_t^*$, compute the discrepancies

$$D_t(a) = \ln t / \tilde{\mathbf{K}}_{x_t,a}(\hat{P}_t, \underline{\hat{v}}_t, \hat{a}_t^*) - T_{x_t,a}(t).$$

If $\max_{a \neq \hat{a}_t^*} D_t(a) \leq 0$, take $\pi(t) = \hat{a}_t^*$, otherwise, take $\pi(t) = \arg\max_{a \neq \hat{a}_t^*} D_t(a)$.

Following this algorithm, we perpetually reduce the discrepancy between the estimated sub-optimal actions, and the estimated rate at which those actions should be taken. The exchange from $\mathbf{K}$ to $\tilde{\mathbf{K}}$ sacrifices some performance in the pursuit of computational simplicity, however it also seems clear from computational experiments that MDP-DMED as above is not only computationally tractable, but also produces reasonable performance in terms of achieving small regret c.f. Figure 3.3. Further discussion of the performance of this algorithm is given in Section 3.4.

### 3.2.3 Optimistic Linear Programming, Another UCB-Type Algorithm for MDPs Under Uncertain Transitions

As we have previously noted, [3] raises some legitimate computational concerns. They propose an alternative, algorithm which they term "optimistic linear programming" (OLP), which is closely related to the MDP-UCB algorithm presented here. The main difference between OLP and MDP-UCB is that OLP does not use the KL divergence to determine the confidence interval. Instead, OLP uses $L_1$ distance, which allows the resulting index to be computed via

solving linear programs. This reduces the computational complexity at the cost of performance. As we will show in Section 3.3, the MDP-UCB optimization problem can be simplified drastically, to render the use of OLP, at least with respect to the computational issues, unnecessary. The algorithm we present here is a version of OLP algorithm presented in [3].

At any time $t \geq 1$, let $x_t$ be the current state, and construct the estimators as in the MDP-UCB algorithm in Section 3.2.1, $\hat{P}_t$, $\hat{A}_t$, and utilize these to compute the estimated optimal values, $\hat{\phi}_t = \phi(\hat{A}_t, \hat{P}_t)$ and $\underline{\hat{v}}_t = \underline{v}(\hat{A}_t, \hat{P}_t)$.

At this point, we implement the following decision rule: for each action $a \in A(x_t)$, we compute the following *index*, again maximizing value within some distance of the current estimates:

$$u_a(t) = \sup_{\underline{q} \in \Theta} \left\{ L(x_t, a, \underline{q}, \underline{\hat{v}}_t) : ||\underline{\hat{p}}^a_{x_t} - \underline{q}||_1 \leq \sqrt{\frac{2 \ln t}{T^a_{x_t}(t)}} \right\},$$

and take action

$$\pi(t) = \arg\max_{a \in A(x_t)} u_a(t).$$

### 3.2.4 A Thompson-Type Algorithm for MDPs Under Uncertain Transitions

In MDP-UCB, MDP-DMED, and OLP, above, we realized the notion of "exploration" in terms of considering alternative hypotheses that were "close" to the current estimates within $\Theta$, interpreting closeness in terms of "plausibility". In this section, we consider an alternative form of exploration through random sampling over $\Theta$, based on the current available data. Given a uniform prior over $\Theta$, the posterior for $\underline{p}^a_x$ is given by a Dirichlet distribution with the observed occurrences. Posterior Sampling (MDP-PS) proceeds in the following way:

At any time $t \geq 1$, let $x_t$ be the current state, and construct the estimators as in the MDP-UCB algorithm in Section 3.2.1, $\hat{P}_t$, $\hat{A}_t$, and utilize these to compute the estimated optimal values, $\hat{\phi}_t = \phi(\hat{A}_t, \hat{P}_t)$ and $\underline{\hat{v}}_t = \underline{v}(\hat{A}_t, \hat{P}_t)$. In addition, generate the following random vectors:

For each action $a \in A(x_t)$, let $\underline{T}^a_{x_t}(t) = [T^a_{x_t,y}(t)]_{y \in S}$ be the vector of observed transition counts from state $x_t$ to $y$ under action $a$. Generate the random vector $\underline{Q}$ according to

$$\underline{Q}^a(t) \sim \mathrm{Dir}(\underline{T}^a_{x_t}(t)).$$

The $\underline{Q}^a(t)$ are distributed according to the joint posterior distribution of $\underline{p}^a_{x_t}$ with a uniform prior.

At this point, define the following values as posterior sampled estimates of the potential value $L$ of each action:

$$W_a(t) = r_{x_t,a} + \sum_y Q_y^a(t)\hat{v}_y,$$

and take action $\pi(t) = \arg\max_{a \in A(x_t)} W_a(t)$.

In this way, we probabilistically explore likely hypotheses within $\Theta$, and act according to the action with best hypothesized value.

## 3.3 Accelerating Computation

All of the above algorithms require computing the estimated optimality values $\hat{\phi}_t, \underline{\hat{v}}_t$ each round. This is an issue, but efficient linear programming formulations exist to solve the optimality equations in Eq. (1.2) see for example [19]. It may also be possible to adapt the method of [37] for approximately solving MDPs, among others, to our undiscounted and potentially changing MDP setting.

However, each of these algorithms additionally has unique computational challenges, through computations over the high dimensional parameter space $\Theta$ due to the typically high cardinality of the state space.

### 3.3.1 MDP-UCB

We will first examine the MDP-UCB algorithm from Section 3.2.1. Recalling the notation that $\mathbf{I}(\underline{p}, \underline{q}) = \sum_x p_x \ln(p_x/q_x)$, MDP-UCB has to repeatedly solve the following optimization problem:

$$C(\underline{p}, \underline{v}, \delta) = \sup_{\underline{q} \in \Theta} \left\{ \sum_x q_x v_x : \mathbf{I}(\underline{p}, \underline{q}) \leq \delta \right\}.$$

The index of the MDP-UCB algorithm may be efficiently expressed in terms of the $C$ function above, $u_a(t) = r_{x_t}(a) + C\left(\underline{p}^a_{x_t}, \underline{\hat{v}}_t, \frac{\ln t}{T_{x_t,a}(t)}\right)$. We will refer to this formulation as the $\underline{q}$-Formulation.

This represents an $|S|$-dimensional non-linear constrained optimization problem which is not, in general, easy to solve.

For mathematical completeness, as well as for practical implementation, we first analyze some trivial cases. Let $\mu_p = \sum_x p_x v_x$ and $V = \max_x v_x$, then

**Theorem 2.** *The value of $C(\underline{p}, \underline{v}, \delta)$ can be easily found in the following cases:*

- *If $\delta < 0$ then the optimization problem, $C(\underline{p}, \underline{v}, \delta)$ is infeasible and we say $C(\underline{p}, \underline{v}, \delta) = -\infty$.*
- *If $\delta = 0$, then $C(\underline{p}, \underline{v}, \delta) = \mu_p$.*
- *If $\delta > 0$ and $v_{x_1} = v_{x_2}$ for all $x_1, x_2 \in S$, then $C(\underline{p}, \underline{v}, \delta) = \mu_p$.*

Proof of this theorem is provided in Section 3.6.1.

For other cases, we can reduce this to solving a 2 dimensional system of non-linear equations, with unknowns $\mu_q^*$ and $\lambda$ as follows.

**Theorem 3.** *For any $\delta > 0$ and $\underline{v}$ such that $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$,*

$$C(\underline{p}, \underline{v}, \delta) = \mu_q^*,$$

*where*

$$\sum_{x \in S} p_x \ln\left(1 + \frac{v_x - \mu_q^*}{\lambda}\right) = \delta,$$

$$\sum_x p_x \frac{\lambda}{\lambda + v_x - \mu_q^*} = 1,$$

$$\mu_p < \mu_q^* < V \text{ and } \lambda < \mu_q^* - V.$$

Proof of this theorem is provided in Section 3.6.2.

Solving these systems, which we will refer to as the $(\mu_q^*, \lambda)$-Formulation, provides dramatic speed increases for the implementation of the algorithm (Figure 3.1). We also note that the $(\mu_q^*, \lambda)$-Formulation scales manageably with the dimension of the state space, as opposed to the $q$-Formulation. Additionally, the structure of the equations admits several nice solution methods since, for a given $\mu_q$, the second equation has a unique solution for $\lambda$ in the indicated range, and given that solution, the summation in the first equation is increasing to infinity as a function of $\mu_q$.

### 3.3.2 MDP-DMED

Next we examine the MDP-DMED algorithm from Section 3.2.2. Again, recalling the notation that $\mathbf{I}(\underline{p},\underline{q}) = \sum_x p_x \ln(p_x/q_x)$, MDP-DMED has to repeatedly solve the following optimization problems:

$$D(\underline{p},\underline{v},\rho) = \inf_{\underline{q}\in\Theta}\left\{\mathbf{I}(\underline{p},\underline{q}) : \sum_x q_x v_x \geq \rho\right\}.$$

The rate function $\tilde{\mathbf{K}}$ of the MDP-DMED algorithm may be efficiently expressed in terms of the $D$ function above, $\tilde{\mathbf{K}}_{x_t,a}(\hat{P}_t,\hat{\underline{v}}_t,\hat{a}_t^*) = D(\underline{p}^a_{x_t},\hat{\underline{v}}_t,L(x_t,a^*,\underline{p}^{a^*}_x,\hat{\underline{v}}_t) - r_{x_t}(a))$. We will refer to as the $\underline{q}$-Formulation. This represents an $|S|$-dimensional non-linear constrained optimization problems, which is not, in general, easy to solve.

As before, we consider some trivial cases first. Let $\mu_p = \sum_x p_x v_x$ and $V = \max_x v_x$, then

**Theorem 4.** *The value of $D(\underline{p},\underline{v},\rho)$ and by extension $D_t(a)$ can be easily found in the following cases:*

- *If $\rho > V$ then the optimization problem, $D(\underline{p},\underline{v},\rho)$ is infeasible and we say $D(\underline{p},\underline{v},\rho) = \infty$ and $D_t(a) = -T_{x_t,a}(t)$.*
- *If $\rho \leq \mu_p$ then $D(\underline{p},\underline{v},\rho) = 0$ and we say $D_t(a) = \infty$.*
- *If $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$ and $\rho = V$, then optimization problem $D(\underline{p},\underline{v},\rho)$ diverges to infinity and we say $D(\underline{p},\underline{v},\rho) = \infty$ and $D_t(a) = -T_{x_t,a}(t)$.*

Proof of this theorem is provided in Section 3.6.3.

For other cases, this optimization problem reduces to solving a 1-dimensional system of non-linear equations with one unknown, $\lambda$, as follows:

**Theorem 5.** *For any $\underline{v}$ such that $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$ and $\mu_p < \rho < V$,*

$$D(\underline{p},\underline{v},\rho) = \sum_x p_x \ln(1 + (\rho - v_x)\lambda),$$

*where*

$$\sum_x p_x \frac{\rho - v_x}{1 + (\rho - v_x)\lambda} = 0,$$
$$0 < \lambda < \frac{1}{V - \rho}.$$

Proof of this theorem is provided in Section 3.6.4.

As with the MDP-UCB case, solving this system, which we will refer to as the $\lambda$-Formulation, provides dramatic speed increases for the implementation of the algorithm (Figure 3.1). We also note that the $\lambda$-Formulation scales manageably with the dimension of the state space, as opposed to the $q$-Formulation. Additionally, the $\lambda$-Formulation structurally lends itself well to solutions. Over the indicated range, the summation is positive and constant in the limit as $\lambda \to 0$, and monotonically decreasing, diverging to negative infinity as $\lambda \to 1/(V-\rho)$. Hence the solution is unique, and can easily be found via bisection.

### 3.3.3 OLP

Next we examine the OLP algorithm from Section 3.2.3. OLP has to repeatedly solve the following optimization problem:

$$B(\underline{p},\underline{v},\delta) = \sup_{\underline{q}\in\Theta}\left\{\sum_x q_x v_x : ||\underline{\hat{p}}^{x_t} - \underline{q}||_1 \leq \delta\right\}.$$

The index of the OLP algorithm may be efficiently expressed in terms of the $B$ function above, $u_a(t) = r_{x_t}(a) + B\left(\underline{\hat{p}}^a_{x_t}, \underline{\hat{v}}_t, \sqrt{\frac{2\ln t}{T^a_{x_t}(t)}}\right)$. $B(\underline{p},\underline{v},\delta)$ is equivalent to the following linear program:

$$\max_{\underline{q}^+,\underline{q}^-} \quad \sum_{x\in S} v_x(q^-{}_x - q^+{}_x + p_x),$$

s.t.

$$\begin{aligned}
&\sum_{x\in S} q^+{}_x + q^-{}_x \leq \delta, \\
&\sum_{x\in S} q^-{}_x - q^+{}_x = 0, \\
&q^+{}_x - q^-{}_x \leq p_x \qquad \forall x \in S, \\
&q^+{}_x, q^-{}_x \geq 0 \qquad \forall x \in S.
\end{aligned}$$

This represents an $|S|$-dimensional linear program, which can generally be computed quite efficiently. However, as the dimension of the state space increases we incur a greater computational burden (Figure 3.1).

### 3.3.4 MDP-PS

The most attractive advantage of MDP-PS is the reduced computational cost, relative to the other three proposed algorithms (Figure 3.2). Notice there is no extra optimization problem that needs to be solved. In the MDP-UCB algorithm, at every time $t$, we had to iteratively solve $|A(x_t)|$ instances of $C(\underline{p}, \underline{v}, \delta)$, for OLP $|A(x_t)|$ instances of $B(\underline{p}, \underline{v}, \delta)$, and for MDP-DMED, $|A(x_t)|$ instances of $D(\underline{p}, \underline{v}, \rho)$. Under MDP-PS, the computational burden stems from sampling from the Dirichlet distribution for each action (again, $|A(x_t)|$ steps), but this is a well studied problem with many efficiently implemented solutions (see for example [38]). Specific properties of the MDP-PS algorithm may still make these other algorithms worth pursuing, however, as seen in Section 3.4.

### 3.3.5 Computation Time Comparison

To demonstrate the computational time savings achieved by these simplifications we randomly generated the parameters for 15 different action indices and timed how long each algorithm took to solve. We repeated this for 4 different values of $|S|$, the dimension of the state space, 10, 100, 1,000, and 10,000. In Figure 3.1, we plot the mean computation time as $|S|$ increases, for each algorithm, [1] MDP-PS, [2] MDP-DMED $\lambda$-Formulation, [3] MDP-UCB $(\mu_q^*, \lambda)$-Formulation , [4] MDP-DMED $\underline{q}$-Formulation , [5] MDP-UCB $\underline{q}$-Formulation, and [6] OLP, along with a 95% confidence interval.

In order to keep the comparisons as equitable as possible, the optimization problem for all the algorithms (with the exception of MDP-PS) were solved to within 4 digits of accuracy using TensorFlow for Python [39]. MDP-PS used SciPy's random Dirichlet generator. They were all run on a MacBook Pro with a 3.1 Ghz i7 processor with 16GB DDR3 RAM.

The top three fastest algorithms were [1] MDP-PS, [2] MDP-DMED $\lambda$-Formulation, and [3] MDP-UCB $(\mu_q^*, \lambda)$-Formulation. Figure 3.2 shows these three in more detail.

From Figure 3.1 we can see the dramatic savings achieved by [2] MDP-DMED using the $\lambda$-Formulation, and [3] MDP-UCB using the $(\mu_q^*, \lambda)$-Formulation as compared to [4,5] the $\underline{q}$-Formulations. [6] OLP also suffers from increasing computation time as the dimension of the state space increases. OLP performs the worst in terms of computational time which is likely

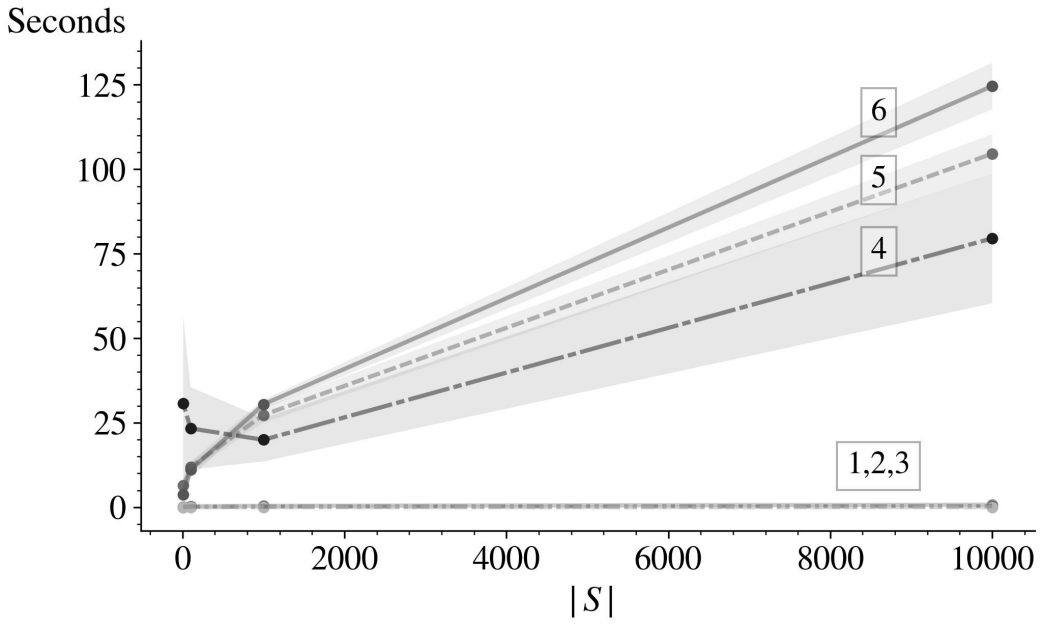


Figure 3.1: Computation time as $|S|$ increases

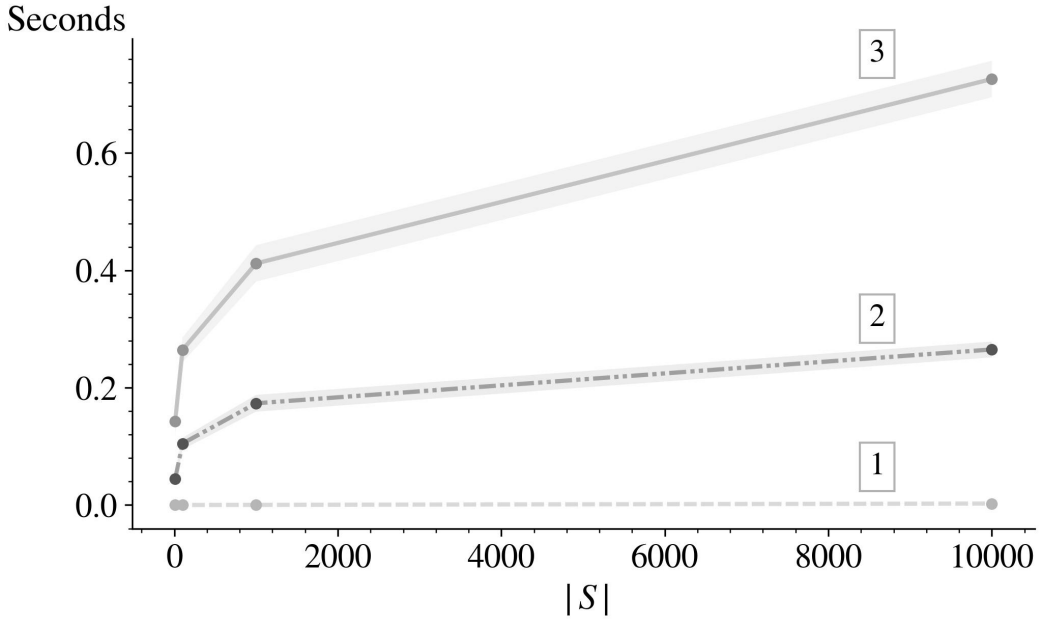


Figure 3.2: Computation time as $|S|$ increases for the top three performers

due to the fact that we are not using a specialized fast LP solver but rather TensorFlow.

In Figure 3.2 we can see the relative performances of the top three algorithms. [1] MDP-PS, unsurprisingly with the fastest, followed by [2] MDP-DMED using the $\lambda$-Formulation with its single unknown, and then [3] MDP-UCB using the $(\mu_q^*, \lambda)$-Formulation with its two unknowns.

The absolute time is not as important as the relative time. There are numerous ways to achieve significantly faster absolute time but our focus here is to demonstrate the relative speed increase gained by using our simplifications. In addition, one can get raw computational time savings by developing a devoted optimizer for problems of this type but if we restrict to using a generic black box optimizer, the method we employed seems a reasonable reflection of what one would do.

## 3.4 Comparison of Performance

In this section we discuss the results of our simulation test of these algorithms on a small example problem. There is nothing particularly special about the values for this example, and we observe similar results under other values. Our example had 3 states ($x_1, x_2$, and $x_3$) with 2 available actions ($a_1$ and $a_2$) in each state. Below we show the transition probabilities, as well as the reward, returned under each action.

$P[a_1] =$

| | $x_1$ | $x_2$ | $x_3$ |
|---|---|---|---|
| $x_1$ | 0.04 | 0.69 | 0.27 |
| $x_2$ | 0.88 | 0.01 | 0.11 |
| $x_3$ | 0.02 | 0.46 | 0.52 |

,

$P[a_2] =$

| | $x_1$ | $x_2$ | $x_3$ |
|---|---|---|---|
| $x_1$ | 0.28 | 0.68 | 0.04 |
| $x_2$ | 0.26 | 0.33 | 0.41 |
| $x_3$ | 0.43 | 0.35 | 0.22 |

,

$R =$

| | $x_1$ | $x_2$ | $x_3$ |
|---|---|---|---|
| $a_1$ | 0.13 | 0.47 | 0.89 |
| $a_2$ | 0.18 | 0.71 | 0.63 |

.

If these transition probabilities were known, the optimal policy for this MDP would be $\pi^*(x_1) = a_1, \pi^*(x_2) = a_2$, and $\pi^*(x_3) = a_1$.

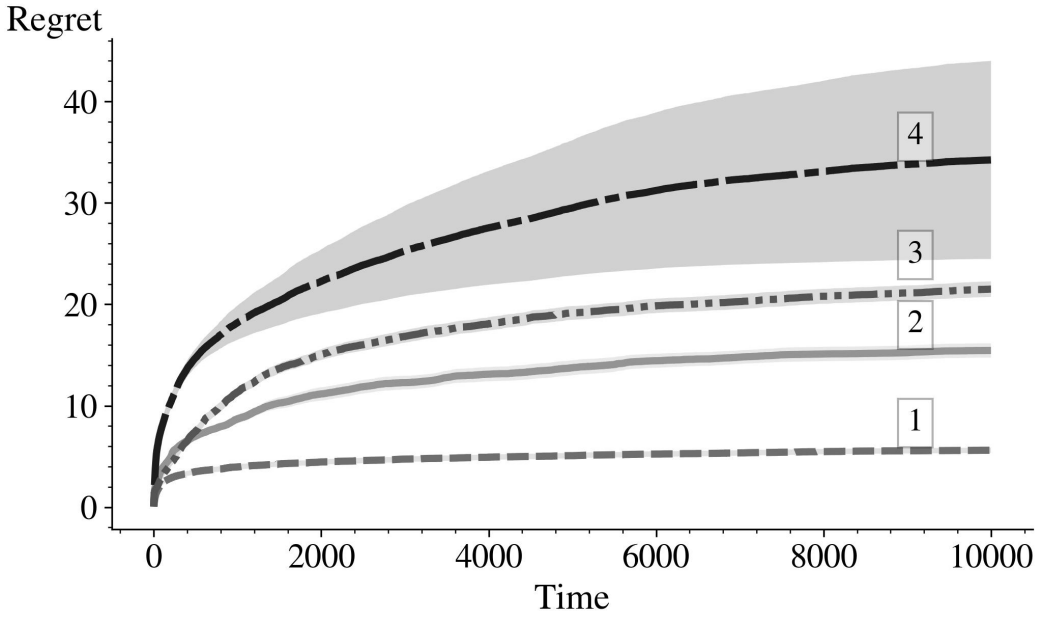


Figure 3.3: Average cumulative regret over time for each algorithm

We simulated each algorithm 100 times over a time horizon of 10,000 and for each time step we computed the mean regret as well as the variance. In Figure 3.3, we plot the mean regret over time for each algorithm, [1] MDP-PS, [2] MDP-UCB, [3] OLP, and [4] MDP-DMED, along with a 95% confidence interval for all sample paths.

We can see that all algorithms seem to have logarithmic growth of regret. There are a few interesting differences that the plot highlights, at least for these specific parameter values:

MDP-DMED has not only the highest finite time regret, but also large variance that seems to increase over time. This seems primarily due to the "epoch" based nature of the algorithm, which results in exponentially long periods when the algorithm may get trapped taking sub-optimal actions, incurring large regret until the true optimal actions are discovered. The benefit of this epoch structure is that once the optimal actions are discovered, they are taken for exponentially long periods, to the exclusion of sub-optimal actions.

As expected, see [3], OLP has a higher finite time regret when compared to MDP-UCB, but still achieves logarithmic growth.

MDP-PS seems to perform best, exhibiting lowest finite time regret as well as the tightest

variance. This seems largely in agreement with the performance of PS-type algorithms in other bandit problems as well, in which they are frequently asymptotically optimal c.f. [36] and references therein.

### 3.4.1 Algorithm Robustness—Inaccurate Priors

How do these algorithms respond to potentially "unlucky" or non-representative streaks of data? How does bad initial estimates effect their performance? Can these algorithms be fooled, and what are the resulting costs before they recover? This is a practically important question, in terms of data security and risk assessment, but also an important element of evaluating a learning algorithm. How does the learning agent respond to non-ideal conditions?

To test these algorithms, we "rigged" or biased the first 60 actions and transitions, such that under the estimated transition probabilities the optimal policy would be to activate the sub-optimal action in each state. In more detail, let $T^a_{x,y}$ be the number of times we transitioned from state $x$ to state $y$ under action $a$. Then we rigged $T^a$ so that it started like so,

$$T[a_1] = \begin{array}{|c|c|c|c|} \hline & x_1 & x_2 & x_3 \\ \hline x_1 & 8 & 1 & 1 \\ \hline x_2 & 1 & 1 & 8 \\ \hline x_3 & 8 & 1 & 1 \\ \hline \end{array},$$

$$T[a_2] = \begin{array}{|c|c|c|c|} \hline & x_1 & x_2 & x_3 \\ \hline x_1 & 1 & 1 & 8 \\ \hline x_2 & 8 & 1 & 1 \\ \hline x_3 & 1 & 1 & 8 \\ \hline \end{array}$$

Under the resulting (bad) estimated transition probabilities, we have that the (estimated) optimal policy is $\hat{\pi}^*(x_1) = a_2, \hat{\pi}^*(x_2) = a_1$, and $\hat{\pi}^*(x_3) = a_1$, which in fact chooses the sub-optimal action in each state.

The subsequent performances of the MDP algorithms are plotted in Figure 3.4. All algorithms still appear to have logarithmic growth in regret, suggesting they can all 'recover' from the initial bad estimates. It is striking though, the extent to which the average regrets for MDP-DMED and MDP-PS are affected, increasing dramatically as a result, MDP-PS demonstrating

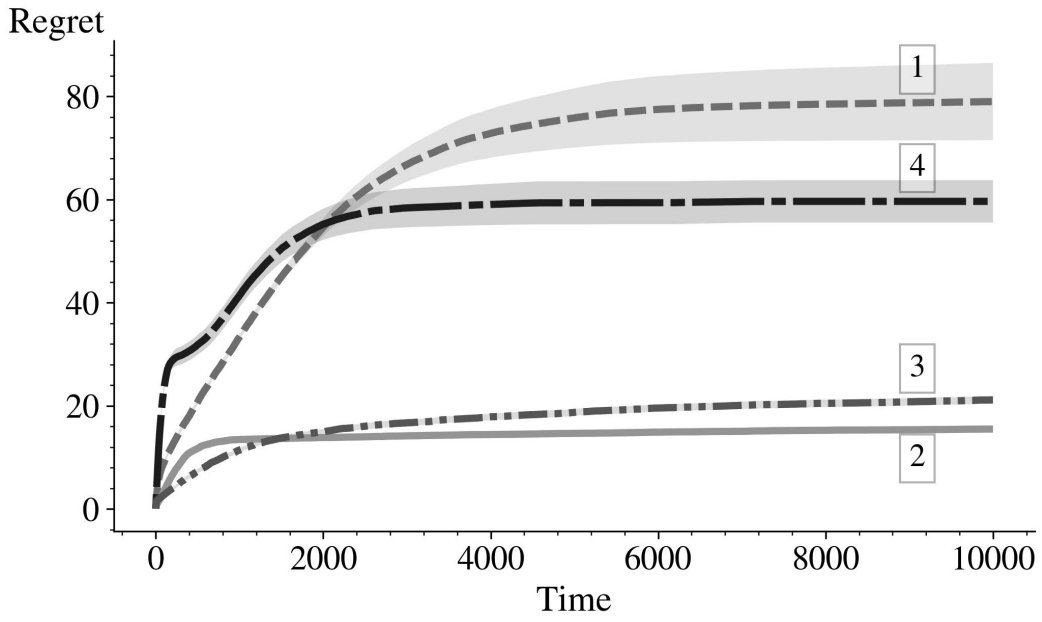


Figure 3.4: Robustness test. MDP-UCB seems to be largely unaffected by the inaccurate priors.

an increase in variance as well. However, the MDP-UCB algorithm seems relatively stable: its average regret has barely increased, and maintains a small variance. Empirically, this phenomenon appears common for the MDP-UCB algorithm under other extreme conditions. The underlying cause and a rigorous examination of these intuitions, will be explored in a future work.

## 3.5 Conclusion and Future Work

We have presented four algorithms adapted from classical multi-armed bandit algorithms that either are provably asymptotically optimal or at least give that appearance in practice. The simplifications for MDP-UCB and MDP-DMED presented here have been shown to dramatically reduce the computational burden for these algorithms, rendering them more useful in practice. As a result, the provably worse performing OLP, no longer has any advantage over them. MDP-DMED under the $\lambda$-Formulation is fast and possibly optimal, but has a high variance for regret that increases over time. While MDP-PS is very fast and appears to be optimal, it is highly sensitive to incorrect priors or extreme sampling errors. MDP-UCB is provably optimal has

stable performance under various extreme conditions, and can be computed quickly using the $(\mu_q^*, \lambda)$-Formulation.

The most immediately obvious extension of this work is to show how the algorithms here satisfy the sufficient conditions developed in Chapter 2. This will not only provide guarantees about the algorithms themselves, but also potentially allow these algorithms to be modified to use other state value estimators (for example, Q-learning [40]) while maintaining their theoretical guarantees.

There are various interesting directions to continue this work, we mention a few potential avenues here. The idea of "exploring the hypothesis space" is something that extends immediately to the case of unknown rewards. Each of the algorithms presented here can generalize immediately to such situations, though the computational simplifications would need to be modified significantly.

From a practical computational point of view we could consider systems where we can't easily iterate over all possible states, and how these algorithms can be modified to address this.

## 3.6 Proof of Theorems of Section 3.3

### 3.6.1 Proof of Theorem 2

First we restate Theorem 2:

The value of $C(\underline{p}, \underline{v}, \delta)$ can be easily found in the following cases:

- If $\delta < 0$ then the optimization problem, $C(\underline{p}, \underline{v}, \delta)$ is infeasible and we say $C(\underline{p}, \underline{v}, \delta) = -\infty$.
- If $\delta = 0$, then $C(\underline{p}, \underline{v}, \delta) = \mu_p$.
- If $\delta > 0$ and $v_{x_1} = v_{x_2}$ for all $x_1, x_2 \in S$, then $C(\underline{p}, \underline{v}, \delta) = \mu_p$.

*Proof.* Recall that $\mathbf{I}(\underline{p}, \underline{q})$ is the KL Divergence from $\underline{p}$ to $\underline{q}$. We then have by Gibb's inequality that $\mathbf{I}(\underline{p}, \underline{q}) \geq 0$, with equality if and only if $\underline{p} = \underline{q}$. Thus, if $\delta < 0$ then the optimization problem is infeasible. If $\delta = 0$ then it has the trivial solution $\underline{q}^* = \underline{p}$. We therefore take $\delta > 0$. Now, if $v_{x_1} = v_{x_2}$ for all $x_1, x_2 \in S$ then any feasible probability vector $\underline{q}$ is also optimal with $C(\underline{p}, \underline{v}, \delta) = v_x = \mu_p$. □

### 3.6.2 Proof of Theorem 3

In this section we will prove Theorem 3, which we restate here.

Let $\mu_p = \sum_x p_x v_x$ and $V = \max_x v_x$. Then for any $\underline{v}$ such that $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$ and $\delta > 0$,

$$C(\underline{p}, \underline{v}, \delta) = \mu_q^*,$$

where

$$\sum_{x \in S} p_x \ln\left(1 + \frac{v_x - \mu_q^*}{\lambda}\right) = \delta,$$

$$\sum_x p_x \frac{\lambda}{\lambda + v_x - \mu_q^*} = 1,$$

$$\mu_p < \mu_q^* < V \text{ and } \lambda < \mu_q^* - V.$$

Before giving the formal proof, it may be helpful to understand the overall conception of the proof. The main idea is the use of Lagrange multiplier techniques, which greatly reduces the dimensionality of the problem to be solved. We are able to exchange from trying to find the optimal probability vector $\underline{q}^*$, to a problem where we need only find two moments of the optimal $\underline{q}^*$, a dramatic dimension reduction. In the MDP-UCB case, it suffices to find the unknown optimal mean of the optimal distribution, $\underline{q}^*$, $\mu_q^*$, and a value $\lambda = \sigma_{q^*}^2/(\mu_p - \mu_q^*)$ which depends on the optimal, unknown variance.

*Proof.* Recall that,

$$C(\underline{p}, \underline{v}, \delta) = \sup_{\underline{q} \in \Theta} \left\{ \sum_x q_x v_x : \mathbf{I}(\underline{p}, \underline{q}) \leq \delta \right\} \tag{3.2}$$

Since $\{\underline{q} : \underline{q} \in \Theta, \mathbf{I}(\underline{p}, \underline{q}) \leq \delta\}$ is a closed compact set, the supremum will be realized by a

maximum, and we may express the problem of computing $C(\underline{p}, \underline{v}, \delta)$ in the following form:

$$\max_{\underline{q}} \quad \mu_q = \sum_{x \in S} q_x v_x, \tag{3.3}$$

s.t.

$$\sum_{x \in S} p_x \ln\left(\frac{p_x}{q_x}\right) \leq \delta, \tag{3.4}$$

$$\sum_{x \in S} q_x = 1,$$

$$q_x > 0 \qquad\qquad x \in S.$$

Let $\mu_q^* = \sum_{x \in S} q_x^* v_x$ be the optimal value of the objective function, $\mu_p = \sum_{x \in S} p_x v_x$, and $V = \max_x v_x$. First we will argue that,

$$\mu_p \leq \mu_q^* < V.$$

To see the first inequality, observe that $\underline{q} = \underline{p}$ satisfies the constraints and is therefore feasible, hence the objective function at $\underline{q} = \underline{p}$ is less than or equal to the optimum: $\mu_p \leq \mu_q^*$. To see the second, note that $\mu_q^*$ will be an expected value over the $\{v_x\}$, and hence less than or equal to the maximum, $V$. Because the probabilities in $\underline{q}^*$ are strictly positive, the expected value $\mu_q^*$ must actually be strictly less than the maximum: $\mu_q^* < V$.

Utilizing Lemma 9 in Appendix 3.7, for any feasible $\underline{q}$ such that the KL Divergence constraint is not achieved with equality, a different feasible $\underline{q'}$ exists with an improved value of the objective function. Hence we can rewrite the optimization problem as,

$$\max_{\underline{q}} \quad \mu_q = \sum_{x \in S} q_x v_x,$$

s.t.

$$\sum_{x \in S} p_x \ln\left(\frac{p_x}{q_x}\right) = \delta, \tag{3.5}$$

$$\sum_{x \in S} q_x = 1, \tag{3.6}$$

$$q_x > 0 \qquad\qquad x \in S. \tag{3.7}$$

We now turn to the main task, reducing the dimension of the optimization problem. Using Lagrange multipliers we have the following auxiliary function,

$$L(\underline{q},\lambda,\mu) = \sum_{x\in S} q_x v_x + \lambda\left(\sum_{x\in S} p_x \ln\left(\frac{p_x}{q_x}\right) - \delta\right) + \mu\left(\sum_{x\in S} q_x - 1\right).$$

Note that when using the Lagrange multipliers, we can safely ignore the positivity inequality constraints in Eq. (3.7) because they are strict inequalities, thus inactive, and removing them will not change the local optimum.

Taking partial derivatives, we get,

$$\begin{aligned}
L'_{q_x}(\underline{q},\lambda,\mu) &= v_x - \frac{\lambda p_x}{q_x} + \mu \quad , \ \forall x \in S,\\
L'_{\lambda}(\underline{q},\lambda,\mu) &= \sum_{x\in S} p_x \ln\left(\frac{p_x}{q_x}\right) - \delta,\\
L'_{\mu}(\underline{q},\lambda,\mu) &= \sum_{x\in S} q_x - 1.
\end{aligned}$$

Setting them to zero, results in the following system of equations for the optimal solution, $\underline{q}^*$,

$$\begin{aligned}
v_x + \mu &= \frac{\lambda p_x}{q_x^*} \quad , \ \forall x \in S, \qquad (3.8)\\
\sum_{x\in S} p_x \ln\left(\frac{p_x}{q_x^*}\right) &= \delta,\\
\sum_{x\in S} q_x^* &= 1.
\end{aligned}$$

We are looking for a solution $\underline{q}^*$ to this system, and any such solution will be a global maximum. To see this, observe that our optimization problem is a convex optimization problem. This can be seen more easily when put in its original form, as in Eq. (3.2). We are maximizing a linear (and thus concave) function, the inequality constraint is convex, and the equality constraints are affine. Thus, any stationary point will be a local maximum and any local maximum will be a global maximum. [41]

Multiplying Eq. (3.8) through by $q_x^*$, we have,

$$\lambda p_x = q_x^*(v_x + \mu) \quad , \ \forall x \in S. \qquad (3.9)$$

Summing Eq. (3.9) over $x$, we have

$$\lambda = \mu_q^* + \mu. \tag{3.10}$$

We now introduce a quantity, $\sigma_{q^*}^2$, the variance under transition law $\underline{q}^*$, explicitly defined as follows

$$\sigma_{q^*}^2 = \sum_{x \in S} q_x^* v_x^2 - \mu_{q^*}^2. \tag{3.11}$$

Looking at Eq. (3.9) again, but this time, multiplying through by $v_x$ we get,

$$\lambda p_x v_x = q_x^* v_x^2 + q_x^* v_x \mu \quad , \ \forall x \in S.$$

Summing this over $x$ yields,

$$\mu_p \lambda = \sigma_{q^*}^2 + \mu_{q^*}^2 + \mu \mu_{q^*}. \tag{3.12}$$

Equations (3.10) and (3.12) form a system of equations with two unknowns $\mu$ and $\lambda$. Solving this system yields,

$$\mu = \frac{\sigma_{q^*}^2 + \mu_{q^*}^2 - \mu_p \mu_{q^*}}{\mu_p - \mu_{q^*}},$$
$$\lambda = \frac{\sigma_{q^*}^2}{\mu_p - \mu_{q^*}}.$$

Substituting them into the first equation in the original system Eq. (3.8), and recalling the relationship between $\lambda$ and $\mu$ from Eq. (3.10), we get that for each x:

$$\begin{aligned} \frac{p_x}{q_x^*} &= \frac{v_x}{\lambda} + \frac{\mu}{\lambda} \\ &= \frac{v_x}{\lambda} + \frac{\mu}{\mu_{q^*} + \mu} \\ &= \frac{v_x}{\lambda} + \frac{\mu_{q^*} + \mu - \mu_{q^*}}{\mu_{q^*} + \mu} \\ &= 1 + \frac{v_x - \mu_{q^*}}{\lambda}. \end{aligned} \tag{3.13}$$

We can now rewrite the optimization problem in Eq. (3.2) in terms of our new variables using Eq. (3.13).

The positivity constraint in Eq. (3.7) and recalling that $p_x > 0$ for all $x \in S$, yields,

$$\frac{p_x}{q_x^*} = 1 + \frac{v_x - \mu_{q^*}}{\lambda} > 0,$$

the normalization constraint in Eq. (3.6) yields,

$$\sum_x \frac{p_x}{1+\frac{v_x-\mu_{q^*}}{\lambda}} = 1,$$

and the KL divergence constraint in Eq. (3.5) yields,

$$\sum_{x\in S} p_x \ln\left(1+\frac{v_x-\mu_q^*}{\lambda}\right) = \delta.$$

Observe that $\mu_p$ must be strictly less than $\mu_q^*$. To see this, take $\underline{q} = \underline{p}$, then $\underline{q}$ is feasible and the left hand side of Eq. (3.4) is 0 which is less than $\delta$. Lemma 9 implies there exists some feasible $\underline{q'}$ with a strictly greater objective function, i.e. $\mu_p = \mu_q < \mu_q' \leq \mu_q^*$. We also know that $\lambda < 0$ because $\sigma_{q^*}^2 > 0$ by definition in Eq. (3.11).

Thus we can rewrite the optimization problem in Eq. (3.2) as, follows:

$$\max_{\mu_q,\lambda} \quad \mu_q,$$

s.t.

$$\begin{aligned}
&\sum_{x\in S} p_x \ln\left(1+\frac{v_x-\mu_q}{\lambda}\right) = \delta,\\
&\sum_x p_x \frac{\lambda}{\lambda+v_x-\mu_q} = 1,\\
&1+\frac{v_x-\mu_q}{\lambda} > 0 \qquad\qquad \forall x\in S,\\
&\mu_p < \mu_q < V \text{ and } \lambda < 0.
\end{aligned} \tag{3.14}$$

Having established that $\lambda$ is strictly less than zero we can simplify the last constraint, Eq. (3.14), as follows. Let $V = \max_x v_x$

$$\begin{aligned}
1+\frac{v_x-\mu_q}{\lambda} &> 0,\ \forall x\in S\\
\frac{v_x-\mu_q}{\lambda} &> -1,\ \forall x\in S\\
v_x-\mu_q &< -\lambda,\ \forall x\in S\\
\mu_q - v_x &> \lambda,\ \forall x\in S\\
\implies \mu_q - V &> \lambda.
\end{aligned}$$

Thus we have,

$$\max_{\mu_q,\lambda} \quad \mu_q,$$

$$\text{s.t.}$$
$$\sum_{x\in S} p_x \ln\left(1+\frac{v_x-\mu_q}{\lambda}\right) = \delta,$$
$$\sum_{x} p_x \frac{\lambda}{\lambda+v_x-\mu_q} = 1,$$
$$\mu_p < \mu_q < V \text{ and } \lambda < \mu_q - V.$$

Which is just two equations with two unknowns. Recalling that any feasible solution will be a global maximum by our discussion of the convexity of the optimization problem, we have the desired result,

$$C(\underline{p},\underline{v},\delta) = \mu_q^*,$$

Where the only unknowns are $\mu_q^*$ and $\lambda$, and they satisfy these constraints:

$$\sum_{x\in S} p_x \ln\left(1+\frac{v_x-\mu_q^*}{\lambda}\right) = \delta,$$
$$\sum_{x} p_x \frac{\lambda}{\lambda+v_x-\mu_q^*} = 1,$$
$$\mu_p < \mu_q^* < V \text{ and } \lambda < \mu_q^* - V.$$

□

### 3.6.3 Proof of Theorem 4

First we restate Theorem 4:

The value of $D(\underline{p},\underline{v},\rho)$ and by extension $D_t(a)$ can be easily found in the following cases:

- If $\rho > V$ then the optimization problem, $D(\underline{p},\underline{v},\rho)$ is infeasible and we say $D(\underline{p},\underline{v},\rho) = \infty$ and $D_t(a) = -T_{x_t,a}(t)$.
- If $\rho \leq \mu_p$ then $D(\underline{p},\underline{v},\rho) = 0$ and we say $D_t(a) = \infty$.

- If $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$ and $\rho = V$, then optimization problem $D(\underline{p}, \underline{v}, \rho)$ diverges to infinity and we say $D(\underline{p}, \underline{v}, \rho) = \infty$ and $D_t(a) = -T_{x_t,a}(t)$.

*Proof.* For $\rho > V = \max_x v_x$, the optimization problem is infeasible because there is no feasible $\underline{q}$ that will have an average more than $V$ (i.e. $\sum_x q_x v_x \leq V$). In that case we take $D(\underline{p}, \underline{v}, \rho) = \infty$ and the corresponding DMED discrepancy index $D_t(a) = -T_{x_t,a}(t)$.

For any $\rho \leq \mu_p$, i.e. less than or equal to the expected value under the current estimates, $D(\underline{p}, \underline{v}, \rho) = 0$ by simply taking $\underline{q}^* = \underline{p}$ and we take the corresponding DMED discrepancy index $D_t(a) = \infty$.

If $v_{x_1} = v_{x_2}$ for all $x_1, x_2 \in S$ then $\mu_p = v_x = V$ and depending on the value of $\rho$ one of the previous two situations apply.

If $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$ and $\rho = V$ we have the following. Any feasible $\underline{q}$ such that $\sum_x q_x v_x = V$ must have $q_x = 0$ for some $x \in S$ such that $v_x < V$, in which case $\underline{q}$ falls outside of $\Theta$ - and it is in fact not feasible. We therefore take $D(\underline{p}, \underline{v}, \rho) = \infty$ and the corresponding DMED discrepancy index $D_t(a) = -T_{x_t,a}(t)$. □

### 3.6.4 Proof of Theorem 5

In this section we will prove Theorem 5, which we restate here. Let $V = \max_x v_x$. Then, for any $\underline{v}$ such that $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$ and for $\sum_{x \in S} p_x v_x < \rho < V$,

$$D(\underline{p}, \underline{v}, \rho) = \sum_x p_x \ln(1 + (\rho - v_x)\lambda),$$

where

$$\sum_x p_x \frac{\rho - v_x}{1 + (\rho - v_x)\lambda} = 0,$$
$$0 < \lambda < \frac{1}{V - \rho}.$$

Before giving the formal proof, it may be helpful to understand the overall conception of the proof. The main idea is the use of Lagrange multiplier techniques, which greatly reduces the dimensionality of the problem to be solved. We are able to exchange from trying to find the optimal probability vector $\underline{q}^*$, to a problem where we need only find two moments of the optimal $\underline{q}^*$, a dramatic dimension reduction. In the MDP-DMED case we are able to simplify

even further, because the optimal unknown mean $\mu_q^*$ is given as $\rho$, and it suffices to find $\lambda = (\mu_q^* - \mu_p)/\sigma_{q^*}^2$ which is a function of the unknown optimal variance.

The proof follows along similar lines as the one for MDP-UCB in Appendix 3.6.2.

*Proof.* Recall that,

$$D(\underline{p}, \underline{v}, \rho) = \inf_{\underline{q} \in \Theta} \left\{ \mathbf{I}(\underline{p}, \underline{q}) : \sum_x q_x v_x \geq \rho \right\}. \tag{3.15}$$

We want to show that the infimum in EQ. (3.15) is realized by a minimum.

Let $0 < \varepsilon < 1$ and $x^* = \arg\max v_x$. Consider the probability vector $\underline{q}'$ defined as $q'_{x^*} = 1 - \varepsilon$ and $q'_x = \varepsilon/|S|$ for $x \neq x^*$. For the appropriate choice of $\varepsilon$, we will have $\sum_x q'_x v_x = \rho < V$ with finite valued $\mathbf{I}(\underline{p}, \underline{q}')$. Thus, $D(\underline{p}, \underline{v}, \rho) \leq \mathbf{I}(\underline{p}, \underline{q}')$ and we can restrict to only considering $\underline{q} \in \Theta$ such that $\mathbf{I}(\underline{p}, \underline{q}) \leq \mathbf{I}(\underline{p}, \underline{q}')$. This feasible set is closed and compact, and hence the infimum is realized by a minimum over this set. Since $\mathbf{I}(\underline{p}, \underline{q}')$ is diverging to infinity as $\varepsilon \to 0$, this minimum must occur in the interior of the constrained feasible region. Hence the infimum *without* the additional constraint on feasibility will also be realized by a minimum within the interior of the set $\{\underline{q} \in \Theta, \sum_x q_x v_x \geq \rho\}$.

Thus, we can rewrite the problem of computing $D(\underline{p}, \underline{v}, \rho)$ in the following form:

$$\min_{\underline{q}} \quad \sum_{x \in S} p_x \ln \frac{p_x}{q_x},$$

s.t.

$$\begin{aligned} &\sum_{x \in S} q_x v_x \geq \rho, \\ &\sum_{x \in S} q_x = 1, \\ &q_x > 0 \qquad x \in S. \end{aligned} \tag{3.16}$$

Here we can use Lemma 10 in Appendix 3.7 to observe that for any feasible $\underline{q}$ where the constraint in Eq. (3.16) is strict, we can construct a feasible $\underline{q}'$ with a strictly smaller objective function (KL divergence w.r.t. $\underline{p}$). As such, the optimum must occur when this constraint is satisfied with equality, and the optimization problem can be re-written as so:

$$\min_{\underline{q}} \quad \sum_{x\in S} p_x \ln \frac{p_x}{q_x},$$

s.t.

$$\sum_{x\in S} q_x v_x = \rho, \tag{3.17}$$

$$\sum_{x\in S} q_x = 1, \tag{3.18}$$

$$q_x > 0 \qquad x \in S. \tag{3.19}$$

We now turn to the main task, reducing the dimension of the optimization problem. Using Lagrange multipliers we have the following auxiliary equation,

$$L(\underline{q},\lambda,\mu) = -\sum_{x\in S} p_x \ln \frac{p_x}{q_x} + \lambda\left(\sum_{x\in S} q_x v_x - \rho\right) + \mu\left(\sum_{x\in S} q_x - 1\right).$$

Note when using the Lagrange multipliers, we can safely ignore the positivity constraints in Eq. (3.19) because they are strict inequalities, thus inactive, and thus have a Lagrange multiplier of zero.

Taking partial derivatives, we get,

$$\begin{aligned}
L'_{q_x}(\underline{q},\lambda,\mu) &= \frac{p_x}{q_x} + \lambda v_x + \mu \quad , \ \forall x \in S,\\
L'_{\lambda}(\underline{q},\lambda,\mu) &= \sum_{x\in S} q_x v_x - \rho,\\
L'_{\mu}(\underline{q},\lambda,\mu) &= \sum_{x\in S} q_x - 1.
\end{aligned}$$

Setting them to zero, results in the following system of equations for the optimal solution, $\underline{q}^*$,

$$-\frac{p_x}{q_x^*} = \lambda v_x + \mu \quad , \ \forall x \in S, \tag{3.20}$$

$$\sum_{x\in S} q_x^* v_x = \rho,$$

$$\sum_{x\in S} q_x^* = 1.$$

We are looking for a solution $\underline{q}^*$ to this system, and any such solution will be a global minimum. To see this, observe that our optimization problem is a convex optimization problem. We are minimizing a convex function, with affine equality constraints. Thus, any stationary point will be a local minimum, and any local minimum will be a global minimum. [41]

Consider the first equation: multiply through by $q_x^*$ to get $-p_x = \lambda v_x q_x^* + \mu q_x^*$. Summing this over $x$ and simplifying accordingly, we get $-1 = \lambda\rho + \mu$.

If we take $-p_x = \lambda v_x q_x^* + \mu q_x^*$ and multiply through by $v_x$, we get $-v_x p_x = \lambda v_x^2 q_x^* + \mu v_x q_x^*$. We now introduce two new quantities, $\rho_p$, the mean under transition law $\underline{p}$, and $\sigma_{q^*}^2$, the variance under transition law $\underline{q}^*$, explicitly defined as follows

$$\rho_p = \sum_x p_x v_x,$$

$$\sigma_{q^*}^2 = \sum_x v_x^2 q_x^* - \rho^2. \tag{3.21}$$

Summing $-v_x p_x = \lambda v_x^2 q_x^* + \mu v_x q_x^*$ over $x$ and simplifying accordingly, we get $-\rho_p = \lambda(\sigma_{q^*}^2 + \rho^2) + \mu\rho$. So we have two equations and two unknowns,

$$\begin{aligned} -1 &= \lambda\rho + \mu, \\ -\rho_p &= \lambda(\sigma_{q^*}^2 + \rho^2) + \mu\rho. \end{aligned}$$

Solving these for $\lambda$ and $\mu$ we have,

$$\begin{aligned} \lambda &= \frac{\rho - \rho_p}{\sigma_{q^*}^2}, \\ \mu &= -1 - \frac{\rho - \rho_p}{\sigma_{q^*}^2}\rho. \end{aligned} \tag{3.22}$$

Substituting them into the first equation in the original system Eq. (3.20), and noting that Eq. (3.22) implies $\mu = -1 - \lambda\rho$, we get that for each x:

$$\begin{aligned} \frac{p_x}{q_x^*} &= -\lambda v_x - \mu \\ &= -\lambda v_x + 1 + \lambda\rho \\ &= 1 + (\rho - v_x)\lambda. \end{aligned} \tag{3.23}$$

In order to reduce the original problem to a 1-dimensional problem, we now express each of the constraints in terms of our new variables using Eq. (3.23). The positivity constraint in Eq. (3.19) and recalling that $p_x > 0$ for all $x \in S$, yields,

$$\frac{p_x}{q_x^*} = 1 + (\rho - v_x)\lambda > 0,$$

the normalization constraint in Eq. (3.18) yields,

$$\sum_x \frac{p_x}{1 + (\rho - v_x)\lambda} = 1,$$

and the mean constraint in Eq. (3.17) yields,

$$\sum_{x \in S} \frac{p_x}{1 + (\rho - v_x)\lambda} v_x = \rho.$$

Therefore, we can express the problem in Eq. (3.15), noting Eq. (3.23) above for the $p_x/q_x^*$ term, as follows:

$$\min_\lambda \quad \sum_x p_x \ln\left(1 + \lambda(\rho - v_x)\right),$$

s.t.

$$\begin{aligned} &\sum_x \frac{p_x}{1 + (\rho - v_x)\lambda} = 1, \\ &\sum_{x \in S} \frac{p_x}{1 + (\rho - v_x)\lambda} v_x = \rho, \\ &1 + \lambda(\rho - v_x) > 0 \qquad \forall x \in S. \end{aligned} \tag{3.24}$$

We next establish feasible bounds for $\lambda$. Observe that the variance, $\sigma_{q^*}^2$ is strictly greater than 0 by definition in Eq. (3.21) and by recalling that there exists some $x_1, x_2 \in S$ such that $v_{x_1} \neq v_{x_2}$. We also know that $\rho > \rho_p = \sum_x p_x v_x$ by assumption. Thus, $\lambda > 0$.

Having established that $\lambda$ is strictly greater than zero we can simplify the last constraint, Eq. (3.24), as follows. Let $V = \max_x v_x$,

$$
\begin{aligned}
& 1+\lambda(\rho-v_x)>0,\ \forall x\in S \\
\implies & 1+\lambda(\rho-V)>0 \\
& 1+\lambda\rho-\lambda V>0 \\
& 1+\lambda\rho>\lambda V \\
& 1>\lambda(V-\rho) \\
& \frac{1}{(V-\rho)}>\lambda.
\end{aligned}
$$

Where the last step is justified by recalling that by assumption $V$ is strictly greater than $\rho$.

So, $0<\lambda<\frac{1}{(V-\rho)}$ and our optimization problem becomes,

$$\min_\lambda \quad \sum_x p_x \ln\left(1+\lambda(\rho-v_x)\right),$$

s.t.

$$
\begin{aligned}
& \sum_x \frac{p_x}{1+(\rho-v_x)\lambda}=1, \\
& \sum_{x\in S} \frac{p_x}{1+(\rho-v_x)\lambda} v_x=\rho, \\
& 0<\lambda<\frac{1}{(V-\rho)}.
\end{aligned}
\tag{3.25}
$$

Taking a closer look at the normalization constraint, Eq. (3.25),

$$
\begin{aligned}
0 &= \sum_x \frac{p_x}{1+\lambda(\rho-v_x)}-1 \\
&= \sum_x p_x\left(\frac{1}{1+\lambda(\rho-v_x)}-1\right) \\
&= \sum_x p_x\left(\frac{1}{1+\lambda(\rho-v_x)}-\frac{1+\lambda(\rho-v_x)}{1+\lambda(\rho-v_x)}\right) \\
&= \sum_x p_x\left(\frac{1-1-\lambda(\rho-v_x)}{1+\lambda(\rho-v_x)}\right) \\
&= -\lambda\sum_x p_x\left(\frac{(\rho-v_x)}{1+\lambda(\rho-v_x)}\right).
\end{aligned}
$$

However, recalling that $\lambda$ is strictly positive, it must be that $\sum_x p_x\left(\frac{(\rho-v_x))}{1+\lambda(\rho-v_x)}\right)=0$. Hence

we have:

$$\min_\lambda \quad \sum_x p_x \ln\left(1+\lambda(\rho-v_x)\right),$$

s.t.

$$\sum_x p_x \left(\frac{(\rho-v_x))}{1+\lambda(\rho-v_x)}\right) = 0, \tag{3.26}$$

$$\sum_{x\in S} \frac{p_x}{1+(\rho-v_x)\lambda} v_x = \rho, \tag{3.27}$$

$$0 < \lambda < \frac{1}{(V-\rho)}.$$

Next we show that any $\lambda$ that satisfies Eq. (3.26) will also satisfy Eq. (3.27) and thus we can remove that constraint,

$$\begin{aligned}
0 &= \sum_x p_x \left(\frac{(\rho-v_x))}{1+\lambda(\rho-v_x)}\right) \\
&= \sum_x \frac{-p_x v_x}{1+\lambda(\rho-v_x)} + \sum_x \frac{p_x \rho}{1+\lambda(\rho-v_x)} \\
&= \sum_x \frac{-p_x v_x}{1+\lambda(\rho-v_x)} + \rho \sum_x \frac{p_x}{1+\lambda(\rho-v_x)} \\
&= \sum_x \frac{-p_x v_x}{1+\lambda(\rho-v_x)} + \rho \cdot 1.
\end{aligned}$$

Where the last line is justified by recalling Eq. (3.25). Thus we have established that,

$$\sum_x \frac{-p_x v_x}{1+\lambda(\rho-v_x)} = -\rho \implies \sum_x \frac{p_x v_x}{1+\lambda(\rho-v_x)} = \rho,$$

which is Eq. (3.27).

Thus we can write the optimization problem as,

$$\min_\lambda \quad \sum_x p_x \ln\left(1+\lambda(\rho-v_x)\right),$$

s.t.

$$\sum_x p_x \left(\frac{(\rho-v_x))}{1+\lambda(\rho-v_x)}\right) = 0, \tag{3.28}$$

$$0 < \lambda < \frac{1}{V-\rho}.$$

Recall that any feasible solution will be a global minimum, by our discussion of the convexity of the optimization problem. To find a feasible solution, notice that the derivative of the objective function with respect to $\lambda$ is simply the first constraint, Eq. (3.28). Therefore any stationary point of the objective function will satisfy the constraint, be feasible, and thus be a global minimum. Hence, we may replace the original optimization problem with the problem of solving,

$$\sum_x p_x \left( \frac{(\rho - v_x))}{1 + \lambda(\rho - v_x)} \right) = 0,$$

subject to $0 < \lambda < \frac{1}{V-\rho}$.

Thus we have the desired result,

$$D(\underline{p}, \underline{v}, \rho) = \sum_x p_x \ln(1 + (\rho - v_x)\lambda),$$

Where the only unknown is $\lambda$, and it satisfies these constraints:

$$\sum_x p_x \frac{\rho - v_x}{1 + (\rho - v_x)\lambda} = 0,$$
$$0 < \lambda < \frac{1}{V - \rho}.$$

□

## 3.7 KL Divergence Optimization Lemmas

The purpose of this section is to state and prove a number of lemmas associated with convex optimization problems involving KL-Divergence terms. They are relevant, but tangential to most of the content of the paper.

In this section, we take $\underline{p} \in \Theta$ to be a distribution over $S$, with $\underline{v}$ to be the vector of intermediate state values. It is convenient to define $\mu_p = \sum_x p_x v_x$ and $V = \max_x v_x$. The vector $\underline{q}$ is taken to be another distribution over $S$, with possibly zero-valued elements. The KL Divergence between $\underline{p}$ and $\underline{q}$ is given by

$$\mathbf{I}(\underline{p}, \underline{q}) = \sum_x p_x \ln \frac{p_x}{q_x}.$$

**Lemma 9.** *Let $\underline{q} \in \Theta$ be such that $\boldsymbol{I}(\underline{p}, \underline{q}) < \delta < \infty$, and suppose $v_{x_1} > v_{x_2}$ for some $x_1, x_2 \in S$. Then there is a valid probability distribution $\underline{q}'$ such that $\boldsymbol{I}(\underline{p}, \underline{q}') \leq \delta$, and*

$$\sum_{x \in S} q_x v_x < \sum_{x \in S} q'_x v_x.$$

*Proof.* Consider constructing an alternative $\underline{q}' \in \Theta$ in the following way. Define $q'_{x_1} = q_{x_1} + \Delta$, $q'_{x_2} = q_{x_2} - \Delta$, and $q'_x = q_x$ for $x \neq x_1, x_2$. Note that for $0 \leq \Delta < \min(q_{x_1}, q_{x_2})$, $\underline{q}'$ will be a valid probability distribution vector over $S$.

We have that for $\Delta > 0$,

$$\begin{aligned}\sum_x q'_x v_x - \sum_x q_x v_x &= (q_{x_1} + \Delta)v_{x_1} + (q_{x_2} - \Delta)v_{x_2} - q_{x_1}v_{x_1} - q_{x_2}v_{x_2} \\ &= \Delta(v_{x_1} - v_{x_2}) \\ &> 0.\end{aligned}$$

It remains to show that the KL Divergence $\mathbf{I}(\underline{p}, \underline{q}')$ does not exceed $\delta$. Note the following relations,

$$\begin{aligned}\mathbf{I}(\underline{p}, \underline{q}') &= \sum_x p_x \ln \frac{p_x}{q'_x} \\ &= \sum_{x \neq x_1, x_2} p_x \ln \frac{p_x}{q_x} + p_{x_1} \ln \frac{p_{x_1}}{q_{x_1} + \Delta} + p_{x_2} \ln \frac{p_{x_2}}{q_{x_2} - \Delta} \\ &= \sum_x p_x \ln \frac{p_x}{q_x} + p_{x_1} \ln \frac{p_{x_1}}{q_{x_1} + \Delta} - p_{x_1} \ln \frac{p_{x_1}}{q_{x_1}} + p_{x_2} \ln \frac{p_{x_2}}{q_{x_2} - \Delta} - p_{x_2} \ln \frac{p_{x_2}}{q_{x_2} - \Delta} \\ &= \mathbf{I}(\underline{p}, \underline{q}) + p_{x_1} \ln \frac{q_{x_1}}{q_{x_1} + \Delta} + p_{x_2} \ln \frac{q_{x_2}}{q_{x_2} - \Delta}.\end{aligned}$$

So, if $\Delta = 0$ then $\mathbf{I}(\underline{p}, \underline{q}') = \mathbf{I}(\underline{p}, \underline{q}) < \delta$. Noting that additional terms in the last equation above are smooth functions of $\Delta$, $\mathbf{I}(\underline{p}, \underline{q}')$ will not exceed $\delta$ in a neighborhood of $\Delta = 0$. Thus for sufficiently small $\Delta > 0$, the Lemma holds. □

**Lemma 10.** *For any $\underline{q}$ such that*

$$\sum_{x \in S} q_x v_x > \rho \geq \sum_{x \in S} p_x v_x, \tag{3.29}$$

*if $v_{x_1} \neq v_{x_2}$ for some $x_1, x_2 \in S$, there exist distributions $\underline{q}'$ such that $\boldsymbol{I}(\underline{p}, \underline{q}') \leq \boldsymbol{I}(\underline{p}, \underline{q})$ and*

$$\sum_{x \in S} q_x v_x > \sum_{x \in S} q'_x v_x \geq \rho.$$

*Proof.* As a consequence of our assumption that $\sum_x q_x v_x > \sum_x p_x v_x$, there must be some $v_{x_1} \neq v_{x_2}$ such that $\underline{q}$ puts more weight on the larger and $\underline{p}$ puts more weight on the smaller. Let $v_{x_1} > v_{x_2}$, with $q_{x_1} > p_{x_1}$ and $q_{x_2} < p_{x_2}$.

Consider constructing an alternative distribution $\underline{q}' \in \Theta$ in the following way. For $0 \leq \Delta < q_{x_1}$, define $\underline{q}'$ by $q'_{x_1} = q_{x_1} - \Delta$, $q'_{x_2} = q_{x_2} + \Delta$, and $q'_x = q_x$ for $x \neq x_1, x_2$. As before, for $\Delta$ in this range, $\underline{q}' \in \Theta$ represents a valid probability distribution on $S$.

As in the proof of Lemma 9, we have that for $\Delta > 0$,

$$\begin{aligned}\sum_x q'_x v_x - \sum_x q_x v_x &= (q_{x_1} + \Delta)v_{x_1} + (q_{x_2} - \Delta)v_{x_2} - q_{x_1}v_{x_1} - q_{x_2}v_{x_2} \\ &= \Delta(v_{x_1} - v_{x_2}) \\ &> 0.\end{aligned}$$

Taking $\Delta$ sufficiently small (so that the mean does not drop below $\rho$), we have that

$$\sum_{x \in S} q_x v_x > \sum_{x \in S} q'_x v_x \geq \rho.$$

It remains to show that $\mathbf{I}(\underline{p}, \underline{q}') \leq \mathbf{I}(\underline{p}, \underline{q})$. Similar to the proof of Lemma 9, we have that

$$\mathbf{I}(\underline{p}, \underline{q}') = \mathbf{I}(\underline{p}, \underline{q}) + p_{x_1} \ln \frac{q_{x_1}}{q_{x_1} - \Delta} + p_{x_2} \ln \frac{q_{x_2}}{q_{x_2} + \Delta}.$$

Hence we see that $\mathbf{I}(\underline{p}, \underline{q}') = \mathbf{I}(\underline{p}, \underline{q})$ when $\Delta = 0$. Looking at the derivative of $\mathbf{I}(\underline{p}, \underline{q}')$ with respect to $\Delta$ at $\Delta = 0$, we see

$$\frac{d}{d\Delta}\mathbf{I}(\underline{p}, \underline{q}')|_{\Delta=0} = \frac{p_{x_1}}{q_{x_1}} - \frac{p_{x_2}}{q_{x_2}} < 0,$$

where the last step follows since $p_{x_1}/q_{x_1} < 1$ and $p_{x_2}/q_{x_2} > 1$, as discussed initially. Hence while the KL divergences are equal for $\Delta = 0$, $\mathbf{I}(\underline{p}, \underline{q}')$ is decreasing within some small neighborhood, and the KL divergence between $\underline{p}$ and $\underline{q}'$ is reduced. □

# Bibliography

[1] A. N. Burnetas and Michael N. Katehakis. “Optimal Adaptive Policies for Markov Decision Processes”. In: *Mathematics of Operations Research* 22.1 (1997), pp. 222–255. ISSN: 0364-765X.

[2] Junya Honda and Akimichi Takemura. “An Asymptotically Optimal Policy For Finite Support Models In The Multiarmed Bandit Problem”. In: *Machine Learning* 85.3 (Dec. 2011), pp. 361–391. ISSN: 1573-0565. URL: https://doi.org/10.1007/s10994-011-5257-4.

[3] Ambuj Tewari and Peter L. Bartlett. “Optimistic Linear Programming gives Logarithmic Regret for Irreducible MDPs”. In: *Advances in Neural Information Processing Systems 20*. Ed. by J. C. Platt et al. Vol. 25. Curran Associates, Inc., 2008, pp. 1505–1512. URL: http://papers.nips.cc/paper/3329-optimistic-linear-programming-gives-logarithmic-regret-for-irreducible-mdps.pdf.

[4] Volodymyr Mnih et al. “Human-level control through deep reinforcement learning”. In: *Nature* 518.7540 (2015), pp. 529–533.

[5] David Silver et al. “Mastering the game of go without human knowledge”. In: *nature* 550.7676 (2017), pp. 354–359.

[6] Ralph Neuneier. “Enhancing Q-learning for optimal asset allocation”. In: *Advances in neural information processing systems*. 1998, pp. 936–942.

[7] Zhiyong Tan, Chai Quek, and Philip Y. K. Cheng. “Stock trading with cycles: A financial application of ANFIS and reinforcement learning”. In: *Expert Systems with Applications* 38.5 (2011), pp. 4741–4755. ISSN: 0957-4174. URL: http://www.sciencedirect.com/science/article/pii/S095741741000905X.

[8] Sofia S. Villar, Jack Bowden, and James Wason. “Multi-armed bandit models for the optimal design of clinical trials: benefits and challenges”. In: *Statistical science: a review journal of the Institute of Mathematical Statistics* 30.2 (2015), p. 199.

[9] Djallel Bouneffouf and Irina Rish. “A survey on practical applications of multi-armed and contextual bandits”. In: *arXiv preprint arXiv:1904.10040* (2019).

[10] Ian Osband et al. *Behaviour Suite for Reinforcement Learning*. 2019. arXiv: 1908.03568 [cs.LG].

[11] Peter Auer and Ronald Ortner. “Logarithmic Online Regret Bounds for Undiscounted Reinforcement Learning”. In: *Advances in Neural Information Processing Systems 19*. Ed. by B. Schölkopf, J. C. Platt, and T. Hoffman. MIT Press, 2007, pp. 49–56. URL: http://papers.nips.cc/paper/3052-logarithmic-online-regret-bounds-for-undiscounted-reinforcement-learning.pdf.

[12] Wesley Cowan, Michael N. Katehakis, and Daniel Pirutinsky. “Reinforcement Learning: a Comparison of UCB Versus Alternative Adaptive Policies”. In: *Proceedings of First Congress of Greek Mathematicians*. De Gruyter Proceedings in Mathematics. 2019. ISBN: 978-3110663075. eprint: arXiv:1909.06019. URL: https://www.degruyter.com/view/product/533848 (visited on 11/23/2019).

[13] Tze Leung Lai and Herbert Robbins. “Asymptotically efficient adaptive allocation rules”. In: *Advances in applied mathematics* 6.1 (1985), pp. 4–22.

[14] Christoph Dann, Tor Lattimore, and Emma Brunskill. “Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning”. In: *Advances in Neural Information Processing Systems*. 2017, pp. 5713–5723.

[15] Mohammad Gheshlaghi Azar, Ian Osband, and Rémi Munos. “Minimax regret bounds for reinforcement learning”. In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 263–272.

[16] Dimitri Bertsekas. *Reinforcement Learning and Optimal Control*. Athena Scientific, 2019. ISBN: 1886529396. URL: https://www.amazon.com/Reinforcement-Learning-Optimal-Control-Bertsekas/dp/1886529396?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&

tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=1886529396.

[17] Tor Lattimore and Csaba Szepesvári. *Bandit Algorithms*. 2018. URL: https://tor-lattimore.com/downloads/book/book.pdf (visited on 07/10/2019).

[18] Nicolo Cesa-Bianchi and Gabor Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, 2006. ISBN: 0521841089. URL: https://www.amazon.com/Prediction-Learning-Games-Nicolo-Cesa-Bianchi/dp/0521841089?SubscriptionId=AKIAIOBINVZYXZQZ2U3A&tag=chimbori05-20&linkCode=xm2&camp=2025&creative=165953&creativeASIN=0521841089.

[19] Cyrus Derman. *Finite State Markovian Decision Processes*. Vol. 19. Orlando, FL, USA: Academic Press, Inc., 1970, pp. 389–390. ISBN: 0122092503.

[20] Eugene A. Feinberg, Pavlo O. Kasyanov, and Michael Z. Zgurovsky. "Partially observable total-cost Markov decision processes with weakly continuous transition probabilities". In: *Mathematics of Operations Research* 41.2 (2016), pp. 656–681.

[21] H. Robbins. "Some Aspects of the Sequential Design of Experiments". In: *Bull. Amer. Math. Monthly* 58 (1952), pp. 527–536.

[22] Taylor W. Killian et al. "Robust and efficient transfer learning with hidden parameter Markov decision processes". In: *Advances in Neural Information Processing Systems*. 2017, pp. 6250–6261.

[23] Finale Doshi-Velez and George Konidaris. "Hidden parameter Markov decision processes: A semiparametric regression approach for discovering latent task parametrizations". In: *IJCAI: proceedings of the conference*. Vol. 2016. NIH Public Access. 2016, p. 1432.

[24] Bennett L. Fox, John E. Rolph, et al. "Adaptive policies for Markov renewal programs". In: *The Annals of Statistics* 1.2 (1973), pp. 334–341.

[25] R. Agrawal. "Adaptive Control of Markov Chains under the Weak Accessibility Condition". In: *Proc. 29th Conf. on Decision and Control*. 1990, pp. 1426–1431.

[26] Csaba Szepesvári and Michael L. Littman. “A Unified Analysis of Value-Function-Based Reinforcement-Learning Algorithms”. In: *Neural Computation* 11.8 (Nov. 1999), pp. 2017–2060. URL: https://doi.org/10.1162/089976699300016070.

[27] Christopher J. C. H. Watkins and Peter Dayan. “Q-learning”. In: *Machine Learning* 8.3 (May 1992), pp. 279–292. ISSN: 1573-0565. URL: https://doi.org/10.1007/BF00992698.

[28] Thomas Jaksch, Ronald Ortner, and Peter Auer. “Near-optimal regret bounds for reinforcement learning”. In: *Journal of Machine Learning Research* 11.Apr (2010), pp. 1563–1600.

[29] Yonathan Efroni et al. “Tight Regret Bounds for Model-Based Reinforcement Learning with Greedy Policies”. In: *CoRR* abs/1905.11527 (2019). arXiv: 1905.11527. URL: http://arxiv.org/abs/1905.11527.

[30] Ian Osband and Benjamin Van Roy. “On Lower Bounds for Regret in Reinforcement Learning”. In: *arXiv preprint arXiv:1608.02732* (2016).

[31] Ian Osband and Benjamin Van Roy. “Why is posterior sampling better than optimism for reinforcement learning?” In: *Proceedings of the 34th International Conference on Machine Learning-Volume 70*. JMLR. org. 2017, pp. 2701–2710.

[32] Junya Honda and Akimichi Takemura. “An Asymptotically Optimal Bandit Algorithm for Bounded Support Models.” In: vol. 85. Jan. 2010, pp. 67–79.

[33] William R Thompson. “On the likelihood that one unknown probability exceeds another in view of the evidence of two samples”. In: *Biometrika* 25.3/4 (1933), pp. 285–294.

[34] Peter Auer and Ronald Ortner. “UCB revisited: Improved regret bounds for the stochastic multi-armed bandit problem”. In: *Periodica Mathematica Hungarica* 61.1-2 (2010), pp. 55–65. ISSN: 0031-5303.

[35] A. N. Burnetas and Michael N. Katehakis. “Optimal Adaptive Policies for Sequential Allocation Problems”. In: *Advances in Applied Mathematics* 17 (1996), pp. 122–142. ISSN: 0196-8858.

[36] Wesley Cowan, Junya Honda, and Michael N. Katehakis. “Normal bandits of unknown means and variances”. In: *The Journal of Machine Learning Research* 18.1 (2017), pp. 5638–5665.

[37] Chandrashekar Lakshminarayanan, Shalabh Bhatnagar, and Csaba Szepesvári. “A Linearly Relaxed Approximate Linear Program for Markov Decision Processes”. In: *IEEE Transactions on Automatic Control* 63.4 (2017), pp. 1185–1191.

[38] David McKay. *Information Theory, Inference and Learning Algorithms*. 2003. Chap. 23. URL: `http://www.inference.org.uk/mackay/itila/book.html` (visited on 07/09/2019).

[39] Martin Abadi et al. “TensorFlow: A system for large-scale machine learning”. In: *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*. 2016, pp. 265–283. URL: `https://www.usenix.org/system/files/conference/osdi16/osdi16-abadi.pdf`.

[40] Christopher John Cornish Hellaby Watkins. “Learning from Delayed Rewards”. PhD thesis. Cambridge, UK: King’s College, May 1989. URL: `http://www.cs.rhul.ac.uk/~chrisw/new_thesis.pdf`.

[41] Stephen Boyd and Lieven Vandenberghe. *Convex Optimization*. Cambridge university press, 2004.