
Oracle-Efficient Regret Minimization in Factored MDPs with Unknown Structure

Aviv Rosenberg¹ Yishay Mansour^{1 2}

Abstract

We study regret minimization in non-episodic factored Markov decision processes (FMDPs), where all existing algorithms make the strong assumption that the factored structure of the FMDP is known to the learner in advance. In this paper, we provide the first algorithm that learns the structure of the FMDP while minimizing the regret. Our algorithm is based on the optimism in face of uncertainty principle, combined with a simple statistical method for structure learning, and can be implemented efficiently given oracle-access to an FMDP planner. Moreover, we give a variant of our algorithm that remains efficient even when the oracle is limited to non-factored actions, which is the case with almost all existing approximate planners. Finally, we leverage our techniques to prove a novel lower bound for the known structure case, closing the gap to the regret bound of Chen et al. [2021].

1. Introduction

Reinforcement learning (RL) considers an agent interacting with an unknown stochastic environment with the aim of maximizing its expected cumulative reward. This is usually modeled by a Markov decision process (MDP) with a finite number of states. The vast majority of provably-efficient RL has focused on the tabular case, where the state space is assumed to be small. Starting with the UCRL algorithm (Jaksch et al., 2010), near-optimal regret bounds were proved (Azar et al., 2017; Fruit et al., 2018; Jin et al., 2018; Zanette & Brunskill, 2019; Efroni et al., 2019). Unfortunately, many real-world RL applications involve problems with a huge state space, yielding the tabular MDP model impractical as it requires the regret to unavoidably scale polynomially with the number of states.

In many practical scenarios, prior knowledge about the envi-

ronment can be leveraged in order to develop more efficient algorithms. A popular way to model additional knowledge about the structure of the environment is by *factored MDPs* (FMDPs; Boutilier et al. (1995; 1999)). The state of an FMDP is composed of d components, called *factors*, and each component is determined by only m other factors, called its *scope*. FMDPs arise naturally in many applications like games, robotics, image-based applications and production lines (where only neighbouring machines affect one another). The common property of all these examples is the huge state space exponential in d , but the very small scope size m (e.g., in images each pixel is a factor and it depends only on neighboring pixels).

The key benefit of FMDPs is the combinatorial state space that allows compact representation. That is, although the number of states is exponential in d , the FMDP representation is only exponential in m (which is much smaller) and polynomial in d . Early works (Kearns & Koller, 1999; Guestrin et al., 2002; Strehl, 2007; Szita & Lőrincz, 2009) show that FMDPs also reduce the sample complexity exponentially, thus avoiding polynomial dependence on the number of states. Recently, this was further extended to algorithms with near-optimal regret bounds (Osband & Van Roy, 2014; Xu & Tewari, 2020; Tian et al., 2020; Chen et al., 2021; Talebi et al., 2021). However, all these works make the strong assumption that the underlying FMDP structure is fully known to the learner in advance.

In this paper we provide the first regret minimization algorithm for FMDPs with unknown structure, thus solving an open problem from Osband & Van Roy (2014). Our algorithm is built on a novel concept of *consistent scopes* and guarantees **near-optimal** \sqrt{T} **regret** that scales polynomially with the FMDP encoding and is therefore exponentially smaller than the number of states (and the regret in tabular MDPs). Moreover, our algorithm features an innovative construction that can incorporate elimination of inconsistent scopes into the optimistic regret minimization framework, while maintaining computational efficiency given oracle-access to an FMDP planner. Keeping computational oracle-efficiency is a difficult challenge in factored MDPs and especially hard when structure is unknown, since the number of possible structure configurations is highly exponential. Furthermore, our algorithm easily accommodates

¹Tel-Aviv University ²Google Research, Tel Aviv. Correspondence to: Aviv Rosenberg <avivros007@gmail.com>.

any level of structure knowledge, and is therefore extremely useful when additional prior domain knowledge is available. We note that while structure learning in FMDPs was previously studied by [Strehl et al. \(2007\)](#); [Diuk et al. \(2009\)](#); [Chakraborty & Stone \(2011\)](#); [Hallak et al. \(2015\)](#); [Guo & Brunskill \(2017\)](#), none of them provide regret guarantees.

To make our algorithms compatible with existing approximate FMDP planners, we also study FMDPs with non-factored actions. To the best of our knowledge, existing planners require small non-factored action space which is not compatible with the FMDP regret minimization literature. To mitigate this gap, we show that even when the oracle is limited to non-factored actions, a variant of our algorithm can still be implemented efficiently and achieve similar near-optimal regret bounds.

Finally, we leverage the techniques presented in this paper to prove a novel lower bound for regret minimization in FMDPs with known structure. This is the first lower bound to show that the regret must scale exponentially with the scope size m , and the first to utilize connections between different factors in a non-trivial way (i.e., with scope size larger than 1). Furthermore, it improves previous lower bounds by a factor of \sqrt{d} and closes the gap to the state-of-the-art regret bound of [Chen et al. \(2021\)](#), thus establishing the minimax optimal regret in this setting.

Our algorithms make oracle use of FMDP planners. However, even where an FMDP can be represented concisely, solving for the optimal policy may take exponentially long in the most general case ([Goldsmith et al., 1997](#); [Littman, 1997](#)). Our focus in this paper is upon the statistical aspect of the learning problem, and we therefore assume oracle-access to an FMDP planner. We emphasize that the oracle assumption appears in all previous regret minimization algorithms. Furthermore, except for the DORL algorithm of [Xu & Tewari \(2020\)](#), all previous algorithms run in time exponential in d even with access to a planning oracle (and known structure). We stress that in many cases of interest, effective approximate planners do exist ([Boutilier et al., 2000](#); [Koller & Parr, 2000](#); [Schuermans & Patrascu, 2001](#); [Guestrin et al., 2001; 2003](#); [Sanner & Boutilier, 2005](#); [Delgado et al., 2011](#)).

2. Preliminaries

An infinite-horizon average-reward MDP is described by a tuple $M = (S, A, P, R)$, where S and A are finite state and action spaces, respectively, $P : S \times A \rightarrow \Delta_S$ is the transition function¹, and $R : S \times A \rightarrow \Delta_{[0,1]}$ is the reward function with expectation $r(s, a) = \mathbb{E}[R(s, a)]$.

The interaction between the MDP and the learner proceeds

as follows. The learner starts in an arbitrary initial state $s^1 \in S$. For $t = 1, 2, \dots$, the learner observes the current state $s^t \in S$, picks an action $a^t \in A$ and earns a reward r^t sampled from $R(s^t, a^t)$. Then, the environment draws the next state $s^{t+1} \sim P(\cdot | s^t, a^t)$ and the process continues.

A policy $\pi : S \rightarrow A$ is a mapping from states to actions, and its *gain* is defined by the average-reward criterion:

$$\lambda(M, \pi, s) \stackrel{\text{def}}{=} \lim_{T \rightarrow \infty} \frac{1}{T} \mathbb{E} \left[\sum_{t=1}^T r(s^t, \pi(s^t)) \mid s^1 = s \right],$$

where $s^{t+1} \sim P(\cdot | s^t, \pi(s^t))$. In order to derive non-trivial regret bounds, one must constrain the connectivity of the MDP ([Bartlett & Tewari, 2009](#)). We focus on *communicating* MDPs, i.e., MDPs with finite *diameter* $D < \infty$.

Definition 1. Let $T(s' | M, \pi, s)$ be the random variable for the first time step in which state s' is reached when playing a stationary policy π in an MDP M with initial state s . The diameter of M is defined as

$$D(M) \stackrel{\text{def}}{=} \max_{s \neq s' \in S} \min_{\pi: S \rightarrow A} \mathbb{E}[T(s' | M, \pi, s)].$$

For communicating MDPs, neither the optimal policy nor its gain depend on the initial state s^1 . We denote them by $\pi^*(M) = \arg \max_{\pi: S \rightarrow A} \lambda(M, \pi, s^1)$ and $\lambda^*(M) = \lambda(M, \pi^*, s^1)$, respectively. We measure the performance of the learner by the *regret*. That is, the difference between the expected gain of the optimal policy in T steps and the cumulative reward obtained by the learner up to time T , i.e.,

$$\text{Reg}_T(M) \stackrel{\text{def}}{=} \sum_{t=1}^T (\lambda^*(M) - r^t),$$

where $r^t \sim R(s^t, a^t)$ and a^t is chosen by the learner.

2.1. Factored MDPs

Factored MDPs inherit the above definitions, but also possess some conditional independence structure that allows compact representation. We follow the factored MDP definition of [Osband & Van Roy \(2014\)](#), which generalizes the original definition of [Boutilier et al. \(2000\)](#); [Kearns & Koller \(1999\)](#) to allow a factored action space as well. We start with a definition of a factored set and scope operation.

Definition 2. A set X is called *factored* if it can be written as a product of n sets X_1, \dots, X_n , i.e., $X = X_1 \times \dots \times X_n$. For any subset of indices $Z = \{i_1, \dots, i_{|Z|}\} \subseteq \{1, \dots, n\}$, define the scope set $X[Z] = X_{i_1} \times \dots \times X_{i_{|Z|}}$. Further, for any $x \in X$ define the scope variable $x[Z] \in X[Z]$ to be the value of the variables $x_i \in X_i$ with indices $i \in Z$. For singleton sets we write $x[i]$ for $x[\{i\}]$.

Next, we define the factored reward and transition functions. We use the notations $X = S \times A$ for the state-action space,

¹ Δ_X denotes the set of distributions over a set X .

d for the number of state factors and n for the number of state-action factors.

Definition 3. A reward function R is called factored over $X = X_1 \times \dots \times X_n$ with scopes Z_1^r, \dots, Z_ℓ^r if there exist functions $\{R_j : X[Z_j^r] \rightarrow \Delta_{[0,1]}\}_{j=1}^\ell$ with expectations $r_j(x[Z_j^r]) = \mathbb{E}[R_j(x[Z_j^r])]$ such that for all $x \in X$:

$$R(x) = \frac{1}{\ell} \sum_{j=1}^\ell R_j(x[Z_j^r]).$$

Note that when a reward $r = \frac{1}{\ell} \sum_{j=1}^\ell r_j$ is sampled from $R(x)$, the learner observes every r_j individually.

Definition 4. A transition function P is called factored over $X = X_1 \times \dots \times X_n$ and $S = S_1 \times \dots \times S_d$ with scopes Z_1^P, \dots, Z_d^P if there exist functions $\{P_i : X[Z_i^P] \rightarrow \Delta_{S_i}\}_{i=1}^d$ such that for all $x \in X$ and $s' \in S$:

$$P(s' | x) = \prod_{i=1}^d P_i(s'[i] | x[Z_i^P]).$$

That is, given a state-action pair x , factor i of s' is independent of its other factors, and is determined only by $x[Z_i^P]$.

Thus, a factored MDP (FMDP) is defined by an MDP whose reward and transition functions are both factored, and is fully characterized by the tuple

$$M = \left(\{X_i\}_{i=1}^n, \{S_i, Z_i^P, P_i\}_{i=1}^d, \{Z_j^r, R_j\}_{j=1}^\ell \right).$$

As opposed to previous works (Osband & Van Roy, 2014; Xu & Tewari, 2020; Tian et al., 2020; Chen et al., 2021) that assume known factorization, in this paper the learner does not have any prior knowledge of the scopes Z_1^P, \dots, Z_d^P or Z_1^r, \dots, Z_ℓ^r , and they need to be learned from experience. However, the learner has a bound m on the size of the scopes, i.e., $|Z_i^P| \leq m$ and $|Z_j^r| \leq m \forall i, j$. See remarks on unknown scope size and variable scope sizes in Appendix B.

Remark (FMDP encoding size). Let the action factorization $A = A_{d+1} \times \dots \times A_n$, factor size $W = \max\{\max_{1 \leq i \leq d} |S_i|, \max_{d+1 \leq i \leq n} |A_i|\}$ and $L = \max_{Z: |Z|=m} |X[Z]|$. The encoding size is $O(dWL + \ell L + (d + \ell)m \log n)$. Importantly, the encoding is only polynomial in d while the number of states W^d is exponential. It is however exponential in the (much smaller) scope size as $L \approx W^m$.

3. Structure Learning in FMDPs

In order to keep sample efficiency even when the structure of the FMDP is unknown, the learner must be able to detect the actual scopes Z_1^P, \dots, Z_d^P and Z_1^r, \dots, Z_ℓ^r . Let's focus on learning the scopes for the transition function first, as

the technique for the reward function is similar. Our structure learning approach is based on a simple yet powerful observation by Strehl et al. (2007). Since the i -th factor of the next state depends only on the scope Z_i^P , an empirical estimate of P_i should remain relatively similar whether it is computed using Z_i^P or $Z_i^P \cup Z$ for any other scope $Z \subseteq \{1, \dots, n\}$.

Formally, define the empirical transition function for factor i based on scope Z at time step t as

$$\bar{P}_{i,Z}^t(w | v) = \frac{N_{i,Z}^t(v, w)}{\max\{N_Z^t(v), 1\}}$$

for every $(v, w) \in X[Z] \times S_i$, where $N_Z^t(v)$ is the number of times we have visited a state-action pair x such that $x[Z] = v$ up to time step t , and $N_{i,Z}^t(v, w)$ is the number of times this visit was followed by a transition to a state s' such that $s'[i] = w$. Regardless of the additional scope Z , the expected value of $\bar{P}_{i,Z_i^P \cup Z}^t(s'[i] | x[Z_i^P \cup Z])$ remains $P_i(s'[i] | x[Z_i^P])$.

We leverage this observation to define *consistent* scopes. A scope Z of size m is consistent for factor i if for every other scope Z' of size m , $v \in X[Z \cup Z']$ and $w \in S_i$,

$$|\bar{P}_{i,Z \cup Z'}^t(w|v) - \bar{P}_{i,Z}^t(w|v[Z])| \leq 2 \cdot \epsilon_{i,Z \cup Z'}^t(w|v), \quad (1)$$

where $\epsilon_{i,Z}^t(w | v) \stackrel{\text{def}}{=} \sqrt{\frac{18 \bar{P}_{i,Z}^t(w|v) \tau^t}{\max\{N_Z^t(v), 1\}}} + \frac{18 \tau^t}{\max\{N_Z^t(v), 1\}}$ is the radius of the confidence set, $\tau^t = \log(6dWLt/\delta)$ is a logarithmic factor and δ is the confidence parameter.

There are two important properties that hold by a simple application of Hoeffding inequality. First, the actual scope Z_i^P will always be consistent with high probability. Second, if a different scope Z is consistent, then the empirical estimates $\bar{P}_{i,Z_i^P}^t$ and $\bar{P}_{i,Z}^t$ must be close, since both are close to $\bar{P}_{i,Z_i^P \cup Z}^t = \bar{P}_{i,Z \cup Z_i^P}^t$. Therefore, they are close to the true transition function P_i with high probability.

Thus, our approach for structure learning is to eliminate inconsistent scopes. In the next section we show how this idea can be combined with the method of *optimism in face of uncertainty* for regret minimization in FMDPs. This approach works similarly for learning the scopes of the reward function. Formally, define the empirical reward function for reward factor j based on scope Z at time t as $\bar{r}_{j,Z}^t(v) = \frac{1}{\max\{N_Z^t(v), 1\}} \sum_{h=1}^{t-1} r_j^h \cdot \mathbb{I}\{(s^h, a^h)[Z] = v\}$ for every $v \in X[Z]$, where $\mathbb{I}\{\cdot\}$ is the indicator. Similarly to the transitions, a scope Z of size m is *reward consistent* for reward factor j if for every other scope Z' of size m and $v \in X[Z \cup Z']$,

$$|\bar{r}_{j,Z \cup Z'}^t(v) - \bar{r}_{j,Z}^t(v[Z])| \leq 2 \cdot \epsilon_{j,Z \cup Z'}^t(v) \stackrel{\text{def}}{=} 2 \cdot \sqrt{18 \tau^t / \max\{N_{Z \cup Z'}^t(v), 1\}}.$$

4. The SLF-UCRL Algorithm

Our algorithm Structure Learn Factored UCRL (SLF-UCRL) follows the known framework of optimism in face of uncertainty while learning the structure of the FMDP. A sketch is given in [Algorithm 1](#) and the full algorithm can be found in [Appendix A](#). Similarly to the UCRL algorithm ([Jaksch et al., 2010](#)), we split the time into episodes. In the beginning of every episode we compute an optimistic policy and play it for the entire episode. The episode ends once the number of visits to some $v \in X[Z \cup Z']$ is doubled, where $Z \neq Z'$ are two scopes of size m . That is, the number of times we visited a state-action pair x with $x[Z \cup Z'] = v$ is doubled. Note that the standard doubling technique of [Jaksch et al. \(2010\)](#), i.e., when the number of visits to some state-action pair is doubled, will result in regret that depends polynomially on the size of the state-action space, which is exponentially larger than the size of its factors. Moreover, our doubling scheme is different than [Xu & Tewari \(2020\)](#), where the episode size grows arithmetically. This allows us to obtain tighter regret bound that depends on the different sizes of all the factors, and not just the biggest one $L \approx W^m$.

While optimism is a standard framework for regret minimization, our algorithm features two novel techniques to handle unknown structure. First, we show how structure learning can be combined with optimism through the concept of consistent scopes. This already gives an algorithm with bounded regret, but requires exponential running time and space complexity. Second, in [Sections 4.1 and 4.2](#) we present a novel construction that allows to compute the optimistic policy in an oracle-efficient and space efficient manner, although the number of consistent factored structures is clearly exponential.

For every factor i we maintain a set \tilde{Z}_i^k of its consistent scopes up to episode k (we keep a similar set $\tilde{\mathcal{R}}_j^k$ for every reward factor j). In the beginning of the episode we construct an optimistic MDP \tilde{M}^k out of all possible configurations of consistent scopes. We then compute the optimal policy $\tilde{\pi}^k$ of \tilde{M}^k , extract the optimistic policy π^k and play it throughout the episode. In what follows, we denote by t_k the first time step of episode k , and slightly abuse notation by using $\bar{P}^k, \epsilon^k, N^k$ for $\bar{P}^{t_k}, \epsilon^{t_k}, N^{t_k}$.

Remark (Computational complexity). The computational complexity of our algorithm scales exponentially with the scope size m , but polynomially with the number of factors n, d, ℓ . This dependence is unavoidable ([Abbeel et al., 2006](#); [Strehl et al., 2007](#)) since the number of possible scopes is $\binom{n}{m}$ and the size of the FMDP encoding is also exponential in m . In fact, the complexity of all previous regret minimization algorithms (except [Xu & Tewari \(2020\)](#)) is exponential even in the number of factors d and not just in the scope size m . Since FMDPs with large scope size are not practical

(their representation is huge), one should think of m as very small compared to n, d, ℓ .

Algorithm 1 SLF-UCRL Sketch

Input: $\delta, m, S = \{S_i\}_{i=1}^d, S \times A = X = \{X_i\}_{i=1}^n$.
 Initialize visit counters and sets of consistent scopes.
for $k = 1, 2, \dots$ **do**
 Start new episode k , and compute empirical transition function \bar{P}^k and confidence bounds ϵ^k .
 Eliminate inconsistent scopes ([Algorithm 2](#)), and construct optimistic MDP \tilde{M}^k .
 Compute optimal policy $\tilde{\pi}^k$ of \tilde{M}^k using oracle, and extract optimistic policy π^k .
 Execute policy π^k until there are scopes $Z \neq Z'$ of size m and $v \in X[Z \cup Z']$ such that the number of visits to some state-action pairs x with $x[Z \cup Z'] = v$, is doubled.
end for

Algorithm 2 Eliminate Inconsistent Scopes Sketch

for $i = 1, \dots, d$ and $Z \in \tilde{Z}_i^{k-1}$ **do**
 for $Z' \subseteq \{1, \dots, n\}$ of size m and $v \in X[Z \cup Z']$ and $w \in S_i$ **do**
 if $|\bar{P}_{i,Z \cup Z'}^k(w|v) - \bar{P}_{i,Z}^k(w|v[Z])| > 2 \cdot \epsilon_{i,Z \cup Z'}^k(w|v)$ **then**
 Eliminate inconsistent scope: $\tilde{Z}_i^k \leftarrow \tilde{Z}_i^{k-1} \setminus \{Z\}$, and BREAK.
 end if
 end for
end for
 # Inconsistent reward scopes are eliminated from $\tilde{\mathcal{R}}_j^k$ similarly, for every $j = 1, \dots, \ell$.

Remark (Partial structure knowledge). The SLF-UCRL algorithm easily accommodates any level of knowledge regarding the structure. That is, the consistent scopes sets can be adjusted if some scopes are known or have a known compact representation (e.g., decision trees). The algorithm's complexity and regret scale naturally with the level of structure knowledge, making it extremely useful when specific domain knowledge is available (e.g., dynamics of some physical systems in robotics).

4.1. Constructing the Optimistic MDP \tilde{M}^k

As our construction generalizes the one of [Xu & Tewari \(2020\)](#) to the case of unknown structure, we start with a brief overview of their method. With known structure, their optimistic MDP keeps the same state space S but has an extended action space $A \times S$, where playing action (a, s') in state s corresponds to playing action a and using a transition function that puts all the uncertainty in the direction of state

s' , such that for each factor i the L_1 distance between the empirical and optimistic transition functions is bounded by

$$\sum_{w \in S_i} \epsilon_{i,Z_i^P}^k(w \mid (s,a)[Z_i^P]) = \tilde{O}(\sqrt{|S_i|/N_{Z_i^P}^k((s,a)[Z_i^P])}).$$

Formally, let $\mathcal{W}_{i,Z}^k(w|v) = \min\{\epsilon_{i,Z}^k(w|v), \bar{P}_{i,Z}^k(w|v)\}$ and then the probability that in the optimistic MDP the i -th factor of the next state is w after playing (a, s') in state s is

$$\begin{aligned} & \bar{P}_{i,Z_i^P}^k(w \mid (s,a)[Z_i^P]) - \mathcal{W}_{i,Z_i^P}^k(w \mid (s,a)[Z_i^P]) \\ & + \mathbb{I}\{w = s'[i]\} \cdot \sum_{w' \in S_i} \mathcal{W}_{i,Z_i^P}^k(w' \mid (s,a)[Z_i^P]). \end{aligned}$$

The j -th reward factor of this action is the empirical estimate plus an additional optimistic bonus, i.e.,

$$\min\left\{1, \bar{r}_{j,Z_j^r}^k((s,a)[Z_j^r]) + \epsilon_{Z_j^r}^k((s,a)[Z_j^r])\right\}.$$

Notice that this optimistic MDP is factored, that the number of state-action factors increased by d , and that the scope size increased by only 1. Thus, this method indeed keeps oracle-efficiency.

The naive way to extend this idea to unknown structure is to compute the optimistic MDP for every configuration of consistent scopes, and pick the most optimistic one, i.e., the configuration in which the optimal gain is the biggest. However, this requires exponential number of calls to the oracle.

Instead, we propose to extend the action space even further so the policy can pick the scopes as well as the actions. That is, the extended action space is $\tilde{A}^k = A \times S \times \tilde{Z}_1^k \times \dots \times \tilde{Z}_d^k \times \tilde{\mathcal{R}}_1^k \times \dots \times \tilde{\mathcal{R}}_\ell^k$, and playing action $\tilde{a} = (a, s', Z_1, \dots, Z_d, z_1, \dots, z_\ell)$ in state s corresponds to playing action a , using a reward function according to scopes z_1, \dots, z_ℓ , and using a transition function according to scopes Z_1, \dots, Z_d that puts all the uncertainty in the direction of s' . Formally, for every reward factor j define $\tilde{r}_j^k(\tilde{x}) = \min\{1, \bar{r}_{j,z_j}^k((s,a)[z_j]) + \epsilon_{z_j}^k((s,a)[z_j])\}$, where $\tilde{x} = (s, \tilde{a})$. For every factor i and $w \in S_i$ define

$$\begin{aligned} \tilde{P}_i^k(w|\tilde{x}) & \stackrel{\text{def}}{=} \bar{P}_{i,Z_i}^k(w \mid (s,a)[Z_i]) - \mathcal{W}_{i,Z_i}^k(w \mid (s,a)[Z_i]) \\ & + \mathbb{I}\{w = s'[i]\} \cdot \sum_{w' \in S_i} \mathcal{W}_{i,Z_i}^k(w' \mid (s,a)[Z_i]). \end{aligned} \quad (2)$$

Unfortunately, although this elegant construction looks like a factored MDP, it is in fact not factored. Specifically, the transition and reward functions become non-factored because each factor can now depend on all the factors of the state-action space (this is determined by the policy choosing the scopes), i.e., the scope size is now n . Nevertheless, in

the following section we show that the optimal policy of this optimistic MDP can still be computed by the oracle. To that end, we construct a slightly larger MDP that has the same optimal policy and gain, while being factored with small scopes.

4.2. From Optimistic MDP \tilde{M}^k to Optimistic Factored MDP \hat{M}^k

We construct a factored MDP \hat{M}^k that simulates exactly the dynamics of the optimistic MDP \tilde{M}^k . The idea is to stretch each time step to $2 + \log n$ steps. In the first step the policy chooses a combined action \tilde{a} as described in Section 4.1, in the next $\log n$ steps relevant factors are extracted according to the policy's choices, and in the last step the transition is performed according to \tilde{P}^k (Eq. (2)). Since the relevant factors for the transition were already extracted, this time the scope size remains small.

For \hat{M}^k , we keep the extended action space $\hat{A}^k = \tilde{A}^k$ and extend the state space \hat{S}^k to contain the state, steps counter, the policy's picked scopes and optimistic assignment state, and a "temporary" work space. Formally, $\hat{S}^k = S \times \{0, 1, \dots, \log n + 1\} \times S \times \tilde{Z}_1^k \times \dots \times \tilde{Z}_d^k \times \tilde{\mathcal{R}}_1^k \times \dots \times \tilde{\mathcal{R}}_\ell^k \times \Omega^{(d+\ell)m}$, where S keeps the state, $\{0, 1, \dots, \log n + 1\}$ is a counter of the current step within the actual time step, and $S \times \tilde{Z}_1^k \times \dots \times \tilde{Z}_d^k \times \tilde{\mathcal{R}}_1^k \times \dots \times \tilde{\mathcal{R}}_\ell^k$ keeps the policy's picked scopes and optimistic assignment state. For each factor i (also for each reward factor j) and index $e \in \{1, \dots, m\}$, we have a separate "temporary" work space $\Omega = \omega^n \times \omega^{n/2} \times \dots \times \omega^2 \times \omega$ that allows extracting the e -th element of the scope for the transition of factor i while maintaining small scope sizes. Here, $\omega = (\bigcup_{i=1}^d S_i) \cup (\bigcup_{i=d+1}^n A_i)$ keeps one factor (state or action), so $|\omega| = W$.

A state s in M is mapped to $(s, 0, \perp)^2$ and taking action $\tilde{a} = (a, s', Z_1, \dots, Z_d, z_1, \dots, z_\ell)$ results in a deterministic transition to $(s, 1, s', Z_1, \dots, Z_d, z_1, \dots, z_\ell, \tau)$, where $\tau = (\tau_{i,e}) \in \Omega^{(d+\ell)m}$. The state-action pair (s, a) is copied to each of the work spaces, i.e., $\tau_{i,e} = (s, a, \perp)$. The next $\log n$ steps are used to extract the relevant scopes. The policy has no effect in these steps since a, s' and the chosen scopes are now encoded into the state. In these $\log n$ steps, for each (i, e) , we eliminate half of $\tau_{i,e}$ in each step according to its chosen scope Z_i , until we are left with the e -th factor of $(s, a)[Z_i]$. The elimination steps require scopes of size only 4 since each factor of the next step needs to choose between two factors from the previous step (while considering the scope Z_i chosen by the policy and the counter). The final step performs the transition according to \tilde{P}^k , but notice that now it only requires scopes of size $m + 3$. The reason is that now $(s, a)[Z_i]$ has a fixed location within the state,

²We use \perp to indicate that the rest of the state is irrelevant.

i.e., $(s, a)[Z_i][e]$ is in the last factor of $\tau_{i,e}$ (in addition, the counter, $s'[i]$ and $s[i]$ need to be taken into consideration). At this point the agent also gets the reward \tilde{r}^k , whereas the reward in all other time steps is 0. Similarly to the transitions, the reward scopes are of size $m + 1$ because $(s, a)[z_j]$ has a fixed location (and the counter should also be considered). For more details see [Appendix A](#).

It is easy to see that \widehat{M}^k simulates \widetilde{M}^k exactly, because every $2 + \log n$ steps are equivalent to one step in \widetilde{M}^k . In terms of computational complexity, any planner that is able to solve M can also solve \widehat{M}^k , since it is factored and polynomial in size when compared to M . Indeed, the scope size is $m + 3$ (compared to m), the number of state factors is $3d + \ell + 1 + 2nm(d + \ell)$ (compared to d), the number of action factors is $n + d + \ell$ (compared to $n - d$), the size of each state factor is bounded by $\max\{W, \binom{n}{m}\}$ (compared to W), and finally the size L is replaced with $\max\{L, \binom{n}{m}\}W^2(2 + \log n)$. Given the optimal policy $\hat{\pi}^k$ for \widehat{M}^k , we can easily extract the optimal policy $\hat{\pi}^k(s) = \hat{\pi}^k((s, 0, \perp))$ for \widetilde{M}^k , and the optimistic policy $\pi^k(s) = \hat{\pi}^k(s)[1] = \hat{\pi}^k((s, 0, \perp))[1]$ for the original MDP M .

4.3. Avoiding Large Factors

One shortcoming of the above construction is that the factor size may be significantly larger, i.e., $\binom{n}{m}$ instead of W in the original FMDP. As mentioned before, m is considered to be small, and yet one might prefer to keep the factor size small at the expense of adding a few extra factors and increasing the reward scope size by 1. In what follows, we show that this is indeed possible because each action factor we added for choosing a consistent scope is already factored internally into m factors of size n .

We view the extended action space as $A \times S \times \{1, \dots, n\}^{m(d+\ell)}$ which has $n + m(d + \ell)$ factors of size $\max\{W, n\}$. Similarly, we can view the state space as $2d + 1 + m(d + \ell) + 2nm(d + \ell)$ factors of the same size. Luckily we can still keep the same $m + 3$ scope size, since the consistent scopes are used only in the $\log n$ intermediate steps in which the scope size was 4 and now becomes $m + 3$. However, this gives rise to a new problem: now the policy might choose inconsistent scopes because the action space is not restricted to consistent scopes anymore. To overcome this issue, we enforce the optimal policy in \widehat{M}^k to use only consistent scopes by adding $2(d + \ell)$ binary factors. These factors make sure that any policy that uses an inconsistent scope will never earn a reward.

All the new binary factors start as 1, and we refer to them as bits. When the counter is 0, the i -th bit becomes 0 if the chosen scope for factor i is inconsistent. Similarly, the $(d + j)$ -th bit checks the chosen scope for reward factor j . This requires them to have scope size $m + 2$, and in the next

$\log(d + \ell)$ steps we extract out of them one bit that says if an inconsistent scope was chosen. This is done similarly to the extraction of relevant scopes and requires the counter to reach $\max\{\log(d + \ell), \log n\} + 1$ instead of $\log n + 1$. Finally, when giving a reward in the last step, the reward function also considers the extracted bit and gives 0 reward if it is 0. Since it cannot turn back to 1, this bit ensures that a policy that uses an inconsistent scope has a gain of 0.

4.4. Regret Analysis

In [Appendix B](#) we prove the following regret bound for SLF-UCRL. Here we review the main ideas.

Theorem 1. *Running SLF-UCRL on a factored MDP with unknown structure ensures, with probability at least $1 - \delta$,*

$$\text{Reg}_T(M) = \tilde{O}\left(\sum_{i=1}^d \sum_{Z:|Z|=m} D\sqrt{|S_i||X[Z_i^P \cup Z]|T} + \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z:|Z|=m} \sqrt{|X[Z_j^r \cup Z]|T}\right).$$

In the worst-case regret, this regret bound becomes $\text{Reg}_T(M) = \tilde{O}(\binom{n}{m} d D \sqrt{W L^2 T})$. In comparison to the regret bound of [Xu & Tewari \(2020\)](#) for the known structure case, our bound is worse by only a factor of $\binom{n}{m} \sqrt{L}$. While the exponential dependence in m (hidden already in L) is unavoidable, it is an important open problem whether the multiplicative dependence in $\binom{n}{m}$ is necessary (note that it directly stems from the level of structure knowledge and may be much smaller with some domain knowledge). We believe that the \sqrt{L} factor can be avoided with methods such as the meteorologist algorithm of [Diuk et al. \(2009\)](#), since it comes from our simple structure learning method, i.e., comparing all pairs of scopes $Z \neq Z'$ of size m . Still, it is highly unclear how to incorporate these methods in a regret minimization algorithm. Finally, we stress that ignoring the unknown factored structure leads to regret polynomial in the number of states, which is exponential compared to ours.

Proof sketch. Regret analysis for optimistic algorithms has two main parts: (1) *optimism* - show that the optimal gain in the optimistic model \widetilde{M}^k is at least as large as $\lambda^*(M)$ for all episodes k with high probability; (2) *deviation* - bound the difference between the optimistic policy's gains in M and \widetilde{M}^k .

Optimism follows directly from the consistent scopes definition and standard concentration inequalities. Specifically, since the true scopes are always consistent with high probability, the optimistic policy in the optimistic model maximizes its gain while choosing scopes from a set that contains the true scopes. For the deviation, we need to bound the distance between the true and optimistic dynamics along

the trajectory visited in each episode k . That is, we need to relate $\Delta_t = \|\tilde{P}^k(\cdot|\tilde{x}^t) - P(\cdot|x^t)\|_1$ to the confidence radius ϵ^k , where $\tilde{\pi}^k(s^t) = (a^t, s^t, Z_1^t, \dots, Z_d^t, z_1^t, \dots, z_\ell^t)$, $x^t = (s^t, a^t)$ and $\tilde{x}^t = (s^t, \tilde{\pi}^k(s^t))$. Then, we can sum the confidence radii over $t = 1, \dots, T$ and get the final bound.

To that end, we utilize the transition factorization to bound $\Delta_t \leq \sum_{i=1}^d \|\tilde{P}_i^k(\cdot|\tilde{x}^t) - P_i(\cdot|x^t[Z_i^P])\|_1$. Then, for each i we can use the definition of the optimistic transition function \tilde{P}^k (Eq. (2)) to get

$$\begin{aligned} \Delta_t \lesssim & \sum_{i=1}^d \|\tilde{P}_{i,Z_i^P}^k(\cdot|x^t[Z_i^P]) - P_i(\cdot|x^t[Z_i^P])\|_1 \\ & + \sum_{i=1}^d \|\tilde{P}_{i,Z_i^t}^k(\cdot|x^t[Z_i^t]) - \tilde{P}_{i,Z_i^P}^k(\cdot|x^t[Z_i^P])\|_1. \end{aligned}$$

The first term measures the difference between the empirical and true dynamics on the correct scopes Z_i^P and can therefore be bounded with standard concentration inequalities. For the second term we utilize the fact that the chosen scopes Z_i^t must be consistent. Therefore, we can bound it using Eq. (1) by $\approx \sum_{i=1}^d \sum_{w \in S_i} \epsilon_{i,Z_i^P \cup Z_i^t}^k(w|x^t[Z_i^P \cup Z_i^t]) \lesssim \sum_{i=1}^d \sqrt{|S_i|/N_{Z_i^P \cup Z_i^t}^k(x^t[Z_i^P \cup Z_i^t])}$. \square

5. Factored MDPs with Non-Factored Actions

So far we assumed that both the state and action spaces are factored. While this model is very general, it also requires an oracle that can solve it. However, almost all existing approximate FMDP planners do not address factored action spaces. Moreover, implicitly they assume that the action set is small (or with very unique structure), as they pick a greedy policy with respect to some Q-function estimation.

To make our algorithm more compatible with approximate planners, in this section we do not assume that the action space is factored, and our oracle is limited to such FMDPs. We show that a variant of our algorithm can still achieve similar regret bounds and maintain computational efficiency. This makes our algorithm much more practical than the DORL algorithm of Xu & Tewari (2020). The FMDP definition we adopt assumes that the state space is factored $S = S_1 \times \dots \times S_d$, and that the transition function is factored, only with respect to the state space, in the following manner. The factored reward function is defined similarly, but to simplify presentation, we assume it is known.

Definition 5. Transition function P is called factored over $S = S_1 \times \dots \times S_d$ with scopes Z_1^P, \dots, Z_d^P if there exist functions $\{P_i : S[Z_i^P] \times A \rightarrow \Delta_{S_i}\}_{i=1}^d$ such that

$$P(s' | s, a) = \prod_{i=1}^d P_i(s'[i] | s[Z_i^P], a).$$

We focus on known structure to convey the main ideas, but in Appendix E we show that the methods presented here can be extended to handle unknown structure. The DORL algorithm (Xu & Tewari, 2020) highly relies on the factored action space, because the optimistic MDP is defined using the huge (yet factored) action space $A \times S$ that allows incorporating an optimistic estimate of the dynamics. Instead, we propose to spread the transition across $2 + d$ steps. In the first step the policy picks an action, step $i + 1$ performs the i -th factor optimistic transition, and the last step completes the move.

Formally, the state space of \tilde{M}^k is $\tilde{S} = S \times \{0, 1, \dots, d + 1\} \times A \times S \times \{0, 1\}$, where S keeps the state, $\{0, 1, \dots, d + 1\}$ is a counter of the current step within the actual time step, A keeps the policy's chosen action, another S helps perform the transition, and the last bit validates that the chosen actions are legal. The action space of \tilde{M}^k is $\tilde{A} = A \cup (\bigcup_{i=1}^d S_i)$. The size of \tilde{A} is $\max\{|A|, W\}$ which is exponentially smaller compared to $|A|W^d$ in the original construction of Xu & Tewari (2020).

A state s in M is mapped to $(s, 0, \perp)$ and action $a \in A$ deterministically transitions to $(s, 1, a, \perp)$, while the other actions are not legal at this state. Picking an illegal action turns the last bit to 0 (it starts as 1), canceling all rewards similarly to Section 4.3. In state $(s, i, a, w_1, \dots, w_{i-1}, \perp)$, legal actions are S_i , and action $w \in S_i$ transitions to state $(s, i + 1, a, w_1, \dots, w_{i-1}, w_i, \perp)$ with probability

$$\begin{aligned} & \tilde{P}_{i,Z_i^P}^k(w_i | s[Z_i^P], a) - \mathcal{W}_{i,Z_i^P}^k(w_i | s[Z_i^P], a) \\ & + \mathbb{I}\{w_i = w\} \cdot \sum_{w' \in S_i} \mathcal{W}_{i,Z_i^P}^k(w' | s[Z_i^P], a). \end{aligned}$$

Finally, $(s, d + 1, a, w_1, \dots, w_d, b)$ transitions deterministically to $(s', 0, \perp)$ for $s' = (w_1, \dots, w_d)$.

Similarly to Section 4.1, one can see that the scope size is now $m + 3$, the number of factors is $2d + 3$, the size of each factor is bounded by $\max\{W, |A|, d + 2\}$, and that the number of actions remains small. Thus we can use the limited oracle in order to solve the optimistic MDP. As for the regret bound, it is easy to verify that optimism still holds, but it is not clear that we can still bound the deviation because now the policy in the optimistic model has significantly more ‘‘power’’ – it chooses the uncertainty direction for factor i after the realizations for factors $1, \dots, i - 1$ of the next state are already revealed. Next, we show that it can still be bounded similarly since the actual action of the policy is chosen before the realizations are revealed (the action is chosen in the first of $d + 1$ steps).

To see that, consider an MDP M' that models the exact same process as M but resembles our optimistic MDP as each time step is stretched over $d + 2$ steps. The state space of M' is \tilde{S} like \tilde{M}^k , and taking action $a \in A$ in

state $(s, 0, \perp)$ transitions to state $(s, 1, a, \perp)$. Then, the policy has no effect for $d + 1$ steps and the action is embedded in the state. For every i and $w_i \in S_i$, the probability of transitioning from $(s, i, a, w_1, \dots, w_{i-1}, \perp)$ to $(s, i + 1, a, w_1, \dots, w_{i-1}, w_i, \perp)$ is simply $P_i(w_i | s[Z_i^P], a)$, and finally, $(s, i, a, w_1, \dots, w_d, b)$ transitions to $(w_1, \dots, w_d, 0, \perp)$.

Clearly, playing policy π in M is equivalent to playing policy π' in M' such that $\pi'((s, 0, \perp)) = \pi(s)$. Therefore, $\lambda^*(M') = \frac{\lambda^*(M)}{d+2}$ and we can analyze the regret in M' to obtain a similar regret bound to [Xu & Tewari \(2020\)](#). The full algorithm which we call Non-Factored Actions DORL (NFA-DORL) is found in [Appendix C](#) and the full proof of the following regret bound is found in [Appendix D](#).

Theorem 2. *Running NFA-DORL on a factored MDP with non-factored actions and known structure ensures with probability $1 - \delta$,*

$$\text{Reg}_T(M) = \tilde{O}\left(\sum_i D \sqrt{|S_i| |S[Z_i^P]| |A| T}\right) + \frac{1}{\ell} \sum_j \sqrt{|S[Z_j^r]| |A| T}.$$

6. Lower Bound

In [Appendix F](#) we prove the following lower bound for regret minimization in factored MDPs.

Theorem 3. *Let $d > m > 0$. For any algorithm there exists an FMDP with $3d + \log d$ state factors of size at most $\max\{W + 1, \log d + 2\}$, non-factored action space of size $|A|$, and scope size $1 + \max\{m, \log d\}$, such that*

$$\mathbb{E}[\text{Reg}_T(M)] = \Omega\left(\sqrt{\frac{d}{\log d}} W^m |A| T\right).$$

The proof leverages our techniques (e.g., propagating rewards through multi-step factored transitions) in order to embed dW^m multi-arm bandit (MAB) problems into a factored MDP, and make sure that they must be solved sequentially and not in parallel. It features a sophisticated construction to utilize connections between factors in such a way that in each step the learner gets information on just a single MAB, forcing her to solve all the MABs one by one. Our construction is also the first to feature arbitrary scope size m , while previous constructions simply take d unrelated factors with scope size 1. As a result, our construction highlights the unique hardness that factored structure might introduce.

This is the first lower bound to show that the regret must scale exponentially with the scope size m . Moreover, it improves on previous lower bounds ([Tian et al., 2020](#); [Chen et al., 2021](#)) by a factor of \sqrt{d} , and matches the state-of-the-art regret bound of [Chen et al. \(2021\)](#) in the known structure

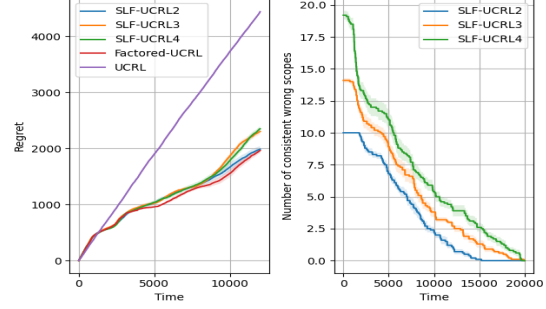


Figure 1. SLF-UCRL performance on circular topology SysAdmin with 4 state factors.

case. Thus, our lower bound is tight, proving that this is indeed the minimax optimal regret for FMDPs with known structure. Yet, two intriguing questions are left open. First, the optimal regret algorithm of [Chen et al. \(2021\)](#) runs in exponential time, and achieving the same regret with an oracle-efficient algorithm seems like a difficult challenge. Second, extending our lower bound to the unknown structure case is another challenging future direction that can advance us towards discovering whether unknown structure indeed introduces additional hardness in terms of optimal regret.

7. Experiments

We test our algorithm on the SysAdmin domain ([Guestrin et al., 2003](#)) – a network of servers connected by some topology, where failing servers affect the probability of their neighbors to fail and the admin chooses which server to reboot at each time step. Our experiments show that the performance of SLF-UCRL is comparable to that of Factored-UCRL ([Osband & Van Roy, 2014](#)) that knows the factored structure in advance, and significantly better than the performance of UCRL ([Jaksch et al., 2010](#)) that completely ignores the factorization. Figure 1 shows that, for circular topology with 4 servers (i.e., 4 state factors and scope size 3), SLF-UCRL eliminates the wrong scopes (right figure), and has similar regret to Factored-UCRL (left figure). “SLF-UCRL i ” refers to i factors whose scope needs to be learned, so SLF-UCRL4 has no knowledge of the structure.

For implementation details and more experiments on different topologies and sizes, see [Appendix G](#).

Acknowledgements

This project has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation program (grant agreement No. 882396), and by the Israel Science Foundation (grant number 993/17).

References

- Abbeel, P., Koller, D., and Ng, A. Y. Learning factor graphs in polynomial time and sample complexity. *Journal of Machine Learning Research*, 7(Aug):1743–1788, 2006.
- Azar, M. G., Osband, I., and Munos, R. Minimax regret bounds for reinforcement learning. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 263–272. JMLR. org, 2017.
- Bartlett, P. L. and Tewari, A. Regal: A regularization based algorithm for reinforcement learning in weakly communicating mdps. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pp. 35–42. AUAI Press, 2009.
- Boutillier, C., Dearden, R., Goldszmidt, M., et al. Exploiting structure in policy construction. In *IJCAI*, volume 14, pp. 1104–1113, 1995.
- Boutillier, C., Dean, T., and Hanks, S. Decision-theoretic planning: Structural assumptions and computational leverage. *Journal of Artificial Intelligence Research*, 11: 1–94, 1999.
- Boutillier, C., Dearden, R., and Goldszmidt, M. Stochastic dynamic programming with factored representations. *Artificial intelligence*, 121(1-2):49–107, 2000.
- Chakraborty, D. and Stone, P. Structure learning in ergodic factored mdps without knowledge of the transition function’s in-degree. In *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, pp. 737–744. Citeseer, 2011.
- Chen, X., Hu, J., Li, L., and Wang, L. Efficient reinforcement learning in factored mdps with application to constrained rl. In *International Conference on Learning Representations*, 2021.
- Delgado, K. V., Sanner, S., and De Barros, L. N. Efficient solutions to factored mdps with imprecise transition probabilities. *Artificial Intelligence*, 175(9-10):1498–1527, 2011.
- Diuk, C., Li, L., and Leffler, B. R. The adaptive k-meteorologists problem and its application to structure learning and feature selection in reinforcement learning. In *Proceedings of the 26th Annual International Conference on Machine Learning*, pp. 249–256, 2009.
- Efroni, Y., Merlis, N., Ghavamzadeh, M., and Mannor, S. Tight regret bounds for model-based reinforcement learning with greedy policies. In Wallach, H. M., Larochelle, H., Beygelzimer, A., d’Alché-Buc, F., Fox, E. B., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, 8-14 December 2019, Vancouver, BC, Canada*, pp. 12203–12213, 2019.
- Fruit, R., Pirotta, M., Lazaric, A., and Ortner, R. Efficient bias-span-constrained exploration-exploitation in reinforcement learning. In *ICML 2018-The 35th International Conference on Machine Learning*, volume 80, pp. 1578–1586, 2018.
- Goldsmith, J., Littman, M. L., and Mundhenk, M. The complexity of plan existence and evaluation in probabilistic domains. In *Proceedings of the Thirteenth conference on Uncertainty in artificial intelligence*, pp. 182–189. Morgan Kaufmann Publishers Inc., 1997.
- Guestrin, C., Koller, D., and Parr, R. Max-norm projections for factored mdps. In *IJCAI*, volume 1, pp. 673–682, 2001.
- Guestrin, C., Patrascu, R., and Schuurmans, D. Algorithm-directed exploration for model-based reinforcement learning in factored mdps. In *ICML*, pp. 235–242. Citeseer, 2002.
- Guestrin, C., Koller, D., Parr, R., and Venkataraman, S. Efficient solution algorithms for factored mdps. *Journal of Artificial Intelligence Research*, 19:399–468, 2003.
- Guo, Z. D. and Brunskill, E. Sample efficient feature selection for factored mdps. *arXiv preprint arXiv:1703.03454*, 2017.
- Hallak, A., Schnitzler, F., Mann, T., and Mannor, S. Off-policy model-based learning under unknown factored dynamics. In *International Conference on Machine Learning*, pp. 711–719, 2015.
- Jaksch, T., Ortner, R., and Auer, P. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(Apr):1563–1600, 2010.
- Jin, C., Allen-Zhu, Z., Bubeck, S., and Jordan, M. I. Is q-learning provably efficient? In *Advances in Neural Information Processing Systems*, pp. 4863–4873, 2018.
- Kearns, M. J. and Koller, D. Efficient reinforcement learning in factored mdps. In Dean, T. (ed.), *Proceedings of the Sixteenth International Joint Conference on Artificial Intelligence, IJCAI 99, Stockholm, Sweden, July 31 - August 6, 1999. 2 Volumes, 1450 pages*, pp. 740–747. Morgan Kaufmann, 1999.
- Koller, D. and Parr, R. Policy iteration for factored mdps. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pp. 326–334. Morgan Kaufmann Publishers Inc., 2000.

- Littman, M. L. Probabilistic propositional planning: Representations and complexity. In *AAAI/IAAI*, pp. 748–754, 1997.
- Osband, I. and Van Roy, B. Near-optimal reinforcement learning in factored mdps. In *Advances in Neural Information Processing Systems*, pp. 604–612, 2014.
- Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., New York, NY, USA, 1st edition, 1994. ISBN 0471619779.
- Rosenberg, A., Cohen, A., Mansour, Y., and Kaplan, H. Near-optimal regret bounds for stochastic shortest path. In *International Conference on Machine Learning*, pp. 8210–8219. PMLR, 2020.
- Sanner, S. and Boutilier, C. Approximate linear programming for first-order mdps. In *Proceedings of the Twenty-First Conference on Uncertainty in Artificial Intelligence*, pp. 509–517, 2005.
- Schuurmans, D. and Patrascu, R. Direct value-approximation for factored mdps. *Advances in Neural Information Processing Systems*, 14:1579–1586, 2001.
- Strehl, A. L. Model-based reinforcement learning in factored-state mdps. In *2007 IEEE International Symposium on Approximate Dynamic Programming and Reinforcement Learning*, pp. 103–110. IEEE, 2007.
- Strehl, A. L., Diuk, C., and Littman, M. L. Efficient structure learning in factored-state mdps. In *AAAI*, volume 7, pp. 645–650, 2007.
- Szita, I. and Lőrincz, A. Optimistic initialization and greediness lead to polynomial time learning in factored mdps. In *Proceedings of the 26th annual international conference on machine learning*, pp. 1001–1008, 2009.
- Talebi, M. S., Jonsson, A., and Maillard, O. Improved exploration in factored average-reward mdps. In *International Conference on Artificial Intelligence and Statistics*, pp. 3988–3996. PMLR, 2021.
- Tian, Y., Qian, J., and Sra, S. Towards minimax optimal reinforcement learning in factored markov decision processes. *Advances in Neural Information Processing Systems*, 33, 2020.
- Xu, Z. and Tewari, A. Reinforcement learning in factored mdps: Oracle-efficient algorithms and tighter regret bounds for the non-episodic setting. *Advances in Neural Information Processing Systems*, 33, 2020.
- Zanette, A. and Brunskill, E. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*, pp. 7304–7312, 2019.

A. The SLF-UCRL Algorithm

Algorithm 3 SLF-UCRL

Input: confidence parameter δ , scope size m , state space $S = \{S_i\}_{i=1}^d$, state-action space $S \times A = X = \{X_i\}_{i=1}^n$.

Initialization

Initialize sets of consistent scopes: $\tilde{\mathcal{R}}_1^0 \leftarrow \dots \leftarrow \tilde{\mathcal{R}}_\ell^0 \leftarrow \tilde{\mathcal{Z}}_1^0 \leftarrow \dots \leftarrow \tilde{\mathcal{Z}}_d^0 \leftarrow \{Z \subseteq \{1, \dots, n\} \mid |Z| = m\}$.

Initialize total visit counters N , in-episode visit counters ν and reward summation variables r :

for $Z \subseteq \{1, \dots, n\}$ such that $m \leq |Z| \leq 2m$, $v \in X[Z]$, $j = 1, \dots, \ell$, $i = 1, \dots, d$, $w \in S_i$ **do**
 $r_{j,Z}(v) \leftarrow N_{i,Z}^0(v, w) \leftarrow \nu_{i,Z}^0(v, w) \leftarrow N_Z^0(v) \leftarrow \nu_Z^0(v) \leftarrow 0$.

end for

Initialize time steps counter: $t \leftarrow 1$, and observe initial state s^1 .

for $k = 1, 2, \dots$ **do**

Start New Episode

Set episode starting time: $t_k \leftarrow t$, and initialize sets of consistent scopes: $\tilde{\mathcal{Z}}_i^k \leftarrow \tilde{\mathcal{Z}}_i^{k-1} \forall i$ and $\tilde{\mathcal{R}}_j^k \leftarrow \tilde{\mathcal{R}}_j^{k-1} \forall j$.

for $Z \subseteq \{1, \dots, n\}$ such that $m \leq |Z| \leq 2m$ and $v \in X[Z]$ **do**

Update visit counters: $\nu_Z^k(v) \leftarrow 0$, $N_Z^k(v) \leftarrow N_Z^{k-1}(v) + \nu_Z^{k-1}(v)$.

for $i = 1, \dots, d$ and $w \in S_i$ **do**

Update visit counters: $\nu_{i,Z}^k(v, w) \leftarrow 0$, $N_{i,Z}^k(v, w) \leftarrow N_{i,Z}^{k-1}(v, w) + \nu_{i,Z}^{k-1}(v, w)$.

Compute empirical transition and reward functions: $\bar{P}_{i,Z}^k(w \mid v) = \frac{N_{i,Z}^k(v, w)}{\max\{N_Z^k(v), 1\}}$; $\bar{r}_{j,Z}^k(v) = \frac{r_{j,Z}(v)}{\max\{N_Z^k(v), 1\}}$.

Set confidence bounds (and set $\mathcal{W}_{i,Z}^k(w \mid v) = \min\{\epsilon_{i,Z}^k(w \mid v), \bar{P}_{i,Z}^k(w \mid v)\}$):

$$\epsilon_{i,Z}^k(w \mid v) = \sqrt{\frac{18 \bar{P}_{i,Z}^k(w \mid v) \log \frac{6dWLt_k}{\delta}}{\max\{N_Z^k(v), 1\}}} + \frac{18 \log \frac{6dWLt_k}{\delta}}{\max\{N_Z^k(v), 1\}} \quad ; \quad \epsilon_Z^k(v) = \sqrt{\frac{18 \log \frac{6dWLt_k}{\delta}}{\max\{N_Z^k(v), 1\}}}$$

end for

end for

Eliminate inconsistent scopes (Algorithm 4).

Construct optimistic MDP \tilde{M}^k and compute optimistic policy π^k (Algorithm 5).

Execute Policy

while $\nu_Z^k((s^t, \pi^k(s^t))[Z]) < N_Z^k((s^t, \pi^k(s^t))[Z]) \forall Z \subseteq \{1, \dots, n\}$ s.t. $m \leq |Z| \leq 2m$ **do**

Play action $a^t = \pi^k(s^t)$, observe next state s^{t+1} and earn reward $r^t = \frac{1}{\ell} \sum_{j=1}^{\ell} r_j^t$.

Update in-episode counters and reward summation variables:

for $Z \subseteq \{1, \dots, n\}$ such that $m \leq |Z| \leq 2m$ and $i = 1, \dots, d$ and $j = 1, \dots, \ell$ **do**

$\nu_Z^k((s^t, a^t)[Z]) \leftarrow \nu_Z^k((s^t, a^t)[Z]) + 1$, $\nu_{i,Z}^k((s^t, a^t)[Z], s^{t+1}[i]) \leftarrow \nu_{i,Z}^k((s^t, a^t)[Z], s^{t+1}[i]) + 1$.

$r_{j,Z}((s^t, a^t)[Z]) \leftarrow r_{j,Z}((s^t, a^t)[Z]) + r_j^t$.

end for

advance time: $t \leftarrow t + 1$.

end while

end for

Algorithm 4 Eliminate Inconsistent Scopes

```

# Eliminate Inconsistent Transition Scopes
for  $i = 1, \dots, d$  and  $Z \in \tilde{\mathcal{Z}}_i^{k-1}$  do
  for  $Z' \subseteq \{1, \dots, n\}$  such that  $|Z'| = m$  and  $v \in X[Z \cup Z']$  and  $w \in S_i$  do
    if  $|\bar{P}_{i,Z \cup Z'}^k(w \mid v) - \bar{P}_{i,Z}^k(w \mid v[Z])| > 2 \cdot \epsilon_{i,Z \cup Z'}^k(w \mid v)$  then
       $\tilde{\mathcal{Z}}_i^k \leftarrow \tilde{\mathcal{Z}}_i^k \setminus \{Z\}$ .
    end if
  end for
end for

# Eliminate Inconsistent Reward Scopes
for  $j = 1, \dots, \ell$  and  $Z \in \tilde{\mathcal{R}}_j^{k-1}$  do
  for  $Z' \subseteq \{1, \dots, n\}$  such that  $|Z'| = m$  and  $v \in X[Z \cup Z']$  do
    if  $|\bar{r}_{j,Z \cup Z'}^k(v) - \bar{r}_{j,Z}^k(v[Z])| > 2 \cdot \epsilon_{Z \cup Z'}^k(v)$  then
       $\tilde{\mathcal{R}}_j^k \leftarrow \tilde{\mathcal{R}}_j^k \setminus \{Z\}$ .
    end if
  end for
end for

```

Algorithm 5 SLF-UCRL Compute Optimistic Policy π^k

Construct MDP: $\widehat{M}^k = (\widehat{S}^k, \widehat{A}^k, \widehat{P}^k, \widehat{r}^k)$.

Define action space: $\widehat{A}^k = A \times S \times \widetilde{Z}_1^k \times \dots \times \widetilde{Z}_d^k \times \widetilde{R}_1^k \times \dots \times \widetilde{R}_\ell^k$.

Define state space: $\widehat{S}^k = S \times \{0, 1, \dots, \log n + 1\} \times S \times \widetilde{Z}_1^k \times \dots \times \widetilde{Z}_d^k \times \widetilde{R}_1^k \times \dots \times \widetilde{R}_\ell^k \times \Omega^{m(d+\ell)}$, where $\Omega = \omega^n \times \omega^{n/2} \times \dots \times \omega^2 \times \omega$ for $\omega = (\bigcup_{i=1}^d S_i) \cup (\bigcup_{i=d+1}^n A_i)$.

Define transition function $\widehat{P}^k(\tilde{s}' | \tilde{s}, \tilde{a}) = \prod_{\tau=1}^{3d+\ell+1+2nm(d+\ell)} \widehat{P}_\tau^k(\tilde{s}'[\tau] | \tilde{s}, \tilde{a})$ as follows:

- The counter factor (factor $d + 1$) counts deterministically modulo $\log n + 2$.
- The action factors (factors $d + 2$ to $3d + \ell + 2$) take the corresponding actions played by the agent when the counter is 0, and otherwise copy the value from the corresponding factor of the previous state.
- For $i = 1, \dots, d$ and $e = 1, \dots, m$, consider $\Omega_{i,e} \in \omega^n \times \omega^{n/2} \times \dots \times \omega^2 \times \omega$ which is the $(i - 1)m + e$ copy of Ω . When the counter is 0 it gets (s, a) , i.e., $\Omega_{i,e} = (s, a, \perp)$. When the counter is 1, we take (s, a) from ω^n and map them to $\omega^{n/2}$ while eliminating half of the factors in consideration with the consistent scope Z_i chosen by the policy (stored in factor $2d + 1 + i$ of the state). This continues for $\log n$ steps until the last ω contains $(s, a)[Z_i][e]$.
- For $j = 1, \dots, \ell$ and $e = 1, \dots, m$, $\Omega_{j,e} \in \omega^n \times \omega^{n/2} \times \dots \times \omega^2 \times \omega$ is the $(d + j - 1)m + e$ copy of Ω . It is handled similarly to the previous item, but considers the reward consistent scope z_j chosen by the policy (stored in factor $3d + 1 + j$ of the state).
- For $i = 1, \dots, d$, the i -th factor is taken from factor i of the previous state when the counter is not $\log n + 1$, and otherwise performs the optimistic transition of factor i . Denote the value in the last factor of $\Omega_{i,e}$ by v_e , the policy's chosen scope by Z_i (stored in factor $2d + 1 + i$ of the state) and the policy's chosen next state direction by s'_i (stored in factor $d + 1 + i$ of the state). Then, the probability that factor i transitions to $w_i \in S_i$ is

$$\begin{aligned} \bar{P}_{i,Z_i}^k(w_i | v_1, \dots, v_m) &= \mathcal{W}_{i,Z_i}^k(w_i | v_1, \dots, v_m) \\ &\quad + \mathbb{I}\{w_i = s'_i\} \cdot \sum_{w \in S_i} \mathcal{W}_{i,Z_i}^k(w | v_1, \dots, v_m). \end{aligned}$$

Define reward function \widehat{r}^k that is always 0 except for the following case. When the counter is $\log n + 1$, for $j = 1, \dots, \ell$, denote by $v_{j,e}$ the last ω in $\Omega_{j,e}$ and by z_j scope chosen by the policy (stored in factor $3d + 1 + j$ of the state). Then, the j -th reward is: $\min\{1, \bar{r}_{j,z_j}^k(v_1, \dots, v_m) + \epsilon_{z_j}^k(v_1, \dots, v_m)\}$.

Compute optimal policy $\hat{\pi}^k$ of \widehat{M}^k using oracle.

Extract optimistic policy: $\pi^k(s) = \hat{\pi}^k((s, 0, \perp))[1]$.

B. Proof of Theorem 1

Remark (Unknown scope size). In this paper we assume that the learner knows a bound m on the scope size in advance. However, in many applications such a bound is not available, and we are required to perform feature selection. Structure learning with unknown scope size was previously studied by [Chakraborty & Stone \(2011\)](#); [Guo & Brunskill \(2017\)](#), but as shown by the latter, it encompasses an inherent difficulty when approached without any additional assumptions. It is an interesting open problem whether additional assumptions are indeed necessary, but here we argue that under the strong assumptions made by previous works, our algorithm keeps a similar regret bound. [Chakraborty & Stone \(2011\)](#) assume that planning with an empirical model with insufficiently large scope size results in ϵ smaller gain than the actual one. In this case, we can keep an estimate \tilde{m} of the scope size and plan twice in each episode, once with \tilde{m} and once with $2\tilde{m}$. If there is a gap of more than ϵ between the gains, we double our estimate. Similarly to the doubling trick used in multi-arm bandit, this adds a constant factor (independent of T) to the regret. [Guo & Brunskill \(2017\)](#) make a similar assumption (but regarding empirical estimates of the transitions) that can be handled similarly.

Remark (Variable scopes sizes). For simplicity, we assume that there is a uniform bound m on the scope sizes of all factors. However, our algorithm readily extends to variable scope sizes, i.e., a bound m_i on the scope size of factor i . Without any changes to the algorithm (just setting different scope sizes for different factors), our algorithm keeps a regret bound of the same order in which the dependence in m is replaced with a dependence in m_i for each factor i .

B.1. Bellman Equations

Define the *bias* of state $s \in S$ as follows,

$$h(M, s) = \mathbb{E} \left[\sum_{t=1}^{\infty} (r(s^t, \pi^*(s^t)) - \lambda^*(M)) \mid s^1 = s \right].$$

The bias vector $h(M, \cdot)$ satisfies the following Bellman optimality equations (see [Puterman \(1994\)](#)),

$$h(M, s) + \lambda^*(M) = r(s, \pi^*(s)) + \sum_{s' \in S} P(s' \mid s, \pi^*(s)) h(M, s') \quad \forall s \in S.$$

We often use the notation $h(s)$ for $h(M, s)$.

B.2. Failure Events

We start by defining the failure events and prove that they occur with probability at most δ . When the failure events do not occur, we say that we are outside the failure event.

- F^r is the event that some empirical estimate of the reward function is far from its expectation. That is, there exist a time t , a reward factor j , a scope Z and a value $v \in X[Z_j^r \cup Z]$ such that

$$|\bar{r}_{j, Z_j^r \cup Z}^t(v) - r_j(v[Z_j^r])| > \epsilon_{Z_j^r \cup Z}^t(v).$$

Notice that the additional scope Z has no influence because the j -th factor only depends on the scope Z_j^r . Thus, by Hoeffding inequality and a union bound the probability of F^r is at most $\delta/5$.

- F^P is the event that some empirical estimate of the transition function is far from its expectation. That is, there exist a time t , a factor i , a scope Z , a value $v \in X[Z_i^P \cup Z]$ and a value $w \in S_i$ such that

$$|\bar{P}_{i, Z_i^P \cup Z}^t(w \mid v) - P_i(w \mid v[Z_i^P])| > \epsilon_{i, Z_i^P \cup Z}^t(w \mid v).$$

Notice that the additional scope Z has no influence because the i -th factor only depends on the scope Z_i^P . Thus, by Bernstein inequality and a union bound the probability of F^P is at most $\delta/5$.

- F_{Az}^r is the event that

$$\sum_{t=1}^T (r(s^t, a^t) - r^t) > 5\sqrt{T \log \frac{10T}{\delta}}.$$

By Azuma inequality the probability of F_{Az}^r is at most $\delta/5$.

- F_{Az}^P is the event that

$$\sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \left(\sum_{s' \in S} P(s' | s^t, a^t) h^k(s') - h^k(s^{t+1}) \right) > 5D \sqrt{T \log \frac{10T}{\delta}},$$

where $h^k(s) = h(\widetilde{M}^k, s)$. By Azuma inequality the probability of F_{Az}^P is at most $\delta/5$.

We define the failure event $F = F^r \cup F^P \cup F_{Az}^P \cup F_{Az}^P$, and by a union bound it occurs with probability at most δ . From now on, we analyze the regret outside the failure events and therefore our regret holds with probability at least $1 - \delta$.

Remark. Notice that outside the failure events the scopes Z_1^P, \dots, Z_d^P and Z_1^r, \dots, Z_ℓ^r are always consistent because:

$$\begin{aligned} |\bar{P}_{i, Z_i^P \cup Z}^t(w | v) - \bar{P}_{i, Z_i^P}^t(w | v[Z])| &\leq |\bar{P}_{i, Z_i^P \cup Z}^t(w | v) - P_i(w | v[Z])| \\ &\quad + |P_i(w | v[Z]) - \bar{P}_{i, Z_i^P}^t(w | v[Z])| \\ &\leq \epsilon_{i, Z_i^P \cup Z}^t(w | v) + \epsilon_{i, Z_i^P}^t(w | v[Z]) \leq 2 \cdot \epsilon_{i, Z_i^P \cup Z}^t(w | v). \end{aligned}$$

B.3. Regret decomposition

Denote $\lambda^* = \lambda^*(M)$ and $\lambda^k = \lambda^*(\widetilde{M}^k)$. Next, we decompose the total regret into the regret in each episode. Then, we further decompose it as follows:

$$\begin{aligned} \text{Reg}_T(M) &= \sum_{t=1}^T (\lambda^* - r^t) \\ &= \sum_{t=1}^T (\lambda^* - r(s^t, a^t)) + \sum_{t=1}^T (r(s^t, a^t) - r^t) \\ &\leq \sum_{t=1}^T (\lambda^* - r(s^t, a^t)) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \end{aligned} \tag{3}$$

$$\begin{aligned} &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} (\lambda^* - r(s^t, a^t)) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \\ &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} (\lambda^* - \lambda^k) \end{aligned} \tag{4}$$

$$\begin{aligned} &\quad + \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} (\lambda^k - r(s^t, \pi^k(s^t))) \\ &\quad + O\left(\sqrt{T \log \frac{T}{\delta}}\right), \end{aligned} \tag{5}$$

where Eq. (3) holds outside the failure event (by event F_{Az}^r). Term (4) is the difference between the optimal gain in the actual MDP and the optimistic MDP, and is bounded by 0 using optimism in [Appendix B.4](#). Term (5) is the deviation of the actual sum of rewards from its expected value in the optimistic MDP, and is bounded by concentration arguments in [Appendix B.5](#).

The theorem then follows from the combination of these two bounds, and because the true MDP M is in the confidence sets of all episodes with probability at least $1 - \delta$, by [Appendix B.2](#).

B.4. Optimism

Lemma 4. For any policy $\pi : S \rightarrow A$ and any vector $h \in \mathbb{R}^{|S|}$, let $\tilde{\pi} : S \rightarrow A \times S \times \widetilde{Z}_1^k \times \dots \times \widetilde{Z}_d^k \times \widetilde{R}_1^k \times \dots \times \widetilde{R}_\ell^k$ be the policy satisfying $\tilde{\pi}(s) = (\pi(s), s^*, Z_1^P, \dots, Z_d^P, Z_1^r, \dots, Z_\ell^r)$ where $s^* = \arg \max_{s \in S} h(s)$. Then, outside the failure event,

$$\sum_{s' \in S} (\tilde{P}^k(s' | s, \tilde{\pi}(s)) - P(s' | s, \pi(s))) h(s') \geq 0 \quad \forall s \in S.$$

Proof. Fix $s \in S$ and denote $x = (s, \pi(s))$. For every $i = 1, \dots, d$ and $w \in S_i$, define $P_i^-(w \mid x[Z_i^P]) = \bar{P}_{i, Z_i^P}^k(w \mid x[Z_i^P]) - \mathcal{W}_{i, Z_i^P}^k(w \mid x[Z_i^P])$, and notice that $P^-(s' \mid x) \leq P(s' \mid x)$ outside the failure event by event F^P . Next, define $\alpha(s' \mid x) \stackrel{\text{def}}{=} \bar{P}^k(s' \mid x) - P(s' \mid x)$ and $\alpha^-(s' \mid x) \stackrel{\text{def}}{=} \bar{P}^k(s' \mid x) - P^-(s' \mid x)$, and notice that $\alpha(s' \mid x) \leq \alpha^-(s' \mid x)$.

Denote $H = \max_{s \in S} h(s)$. By construction of the optimistic transition function,

$$\begin{aligned}
 \sum_{s' \in S} \tilde{P}^k(s' \mid x) h(s') &= \sum_{s' \in S} P^-(s' \mid x) h(s') + H \left(1 - \sum_{s' \in S} P^-(s' \mid x)\right) \\
 &= \sum_{s' \in S} P^-(s' \mid x) h(s') + H \sum_{s' \in S} \alpha^-(s' \mid x) \\
 &= \sum_{s' \in S} (\bar{P}^k(s' \mid x) - \alpha^-(s' \mid x)) h(s') + H \alpha^-(s' \mid x) \\
 &= \sum_{s' \in S} \bar{P}^k(s' \mid x) h(s') + (H - h(s')) \alpha^-(s' \mid x) \\
 &\geq \sum_{s' \in S} \bar{P}^k(s' \mid x) h(s') + (H - h(s')) \alpha(s' \mid x) \\
 &= \sum_{s' \in S} (\bar{P}^k(s' \mid x) - \alpha(s' \mid x)) h(s') + H \alpha(s' \mid x) \\
 &= \sum_{s' \in S} P(s' \mid x) h(s') + H \sum_{s' \in S} \alpha(s' \mid x) = \sum_{s' \in S} P(s' \mid x) h(s').
 \end{aligned}$$

□

Corollary 5. Let $\tilde{\pi}^* : S \rightarrow A \times S \times \tilde{\mathcal{Z}}_1^k \times \dots \times \tilde{\mathcal{Z}}_d^k \times \tilde{\mathcal{R}}_1^k \times \dots \times \tilde{\mathcal{R}}_\ell^k$ be the policy that satisfies $\tilde{\pi}^*(s) = (\pi^*(s), s^*, Z_1^P, \dots, Z_d^P, Z_1^r, \dots, Z_\ell^r)$, where $s^* = \max_{s \in S} h(M, s)$. Then, outside the failure event, $\lambda(\tilde{M}^k, \tilde{\pi}^*, s_1) \geq \lambda^*$ for any starting state s_1 .

Proof. Let $\rho(\cdot) \in \mathbb{R}^{|S|}$ be the vector of stationary distribution for playing policy $\tilde{\pi}^*$ in \tilde{M}^k . By definition of the average reward we have,

$$\begin{aligned}
 \lambda(\tilde{M}^k, \tilde{\pi}^*, s_1) - \lambda^* &= \sum_{s \in S} \rho(s) \tilde{r}^k(s, \tilde{\pi}^*(s)) - \lambda^* \\
 &= \sum_{s \in S} \rho(s) (\tilde{r}^k(s, \tilde{\pi}^*(s)) - \lambda^*) \\
 &\geq \sum_{s \in S} \rho(s) (r(s, \pi^*(s)) - \lambda^*) \\
 &= \sum_{s \in S} \rho(s) \left(h(M, s) - \sum_{s' \in S} P(s' \mid s, \pi^*(s)) h(M, s') \right) \\
 &= \sum_{s \in S} \rho(s) \left(\sum_{s' \in S} \tilde{P}^k(s' \mid s, \tilde{\pi}^*(s)) - \sum_{s' \in S} P(s' \mid s, \pi^*(s)) \right) h(M, s') \geq 0,
 \end{aligned}$$

where the first inequality is by definition of the reward function in \tilde{M}^k and event F^r , and the following equality is by the Bellman equations. The last equality follows because ρ is the stationary distribution of $\tilde{\pi}^*$ is \tilde{M}^k and therefore $\rho(s') = \sum_{s \in S} \rho(s) \tilde{P}^k(s' \mid s, \tilde{\pi}^*(s))$. The final inequality is by [Lemma 4](#). □

B.5. Bounding the Deviation

Denote by $\nu^k(s, a)$ the number of visits to state-action pair (s, a) in episode k , and let $\nu^k(s) = \nu^k(s, \pi^k(s))$ and

$$\Delta_k = \sum_{s \in S} \sum_{a \in A} \nu^k(s, a) (\lambda^k - r(s, a)) = \sum_{s \in S} \nu^k(s) (\lambda^k - r(s, \pi^k(s))).$$

Thus: $(5) = \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} (\lambda^k - r(s^t, \pi^k(s^t))) = \sum_{k=1}^K \Delta_k$.

We now focus on a single episode k . By the Bellman equations in the optimistic model \widetilde{M}^k we have,

$$\begin{aligned} \Delta_k &= \sum_{s \in S} \nu^k(s) (\lambda^k - r(s, \pi^k(s))) \\ &= \sum_{s \in S} \nu^k(s) (\lambda^k - \tilde{r}^k(s, \tilde{\pi}^k(s))) + \sum_{s \in S} \nu^k(s) (\tilde{r}^k(s, \tilde{\pi}^k(s)) - r(s, \pi^k(s))) \\ &= \sum_{s \in S} \nu^k(s) \left(\sum_{s' \in S} \tilde{P}^k(s' | s, \tilde{\pi}^k(s)) h^k(s') - h^k(s) \right) + \sum_{s \in S} \nu^k(s) (\tilde{r}^k(s, \tilde{\pi}^k(s)) - r(s, \pi^k(s))) \\ &= \sum_{s \in S} \nu^k(s) \sum_{s' \in S} h^k(s') (\tilde{P}^k(s' | s, \tilde{\pi}^k(s)) - P(s' | s, \pi^k(s))) \\ &\quad + \sum_{s \in S} \nu^k(s) \left(\sum_{s' \in S} P(s' | s, \pi^k(s)) h^k(s') - h^k(s) \right) + \sum_{s \in S} \nu^k(s) (\tilde{r}^k(s, \tilde{\pi}^k(s)) - r(s, \pi^k(s))) \\ &\leq D \sum_{s \in S} \nu^k(s) \|\tilde{P}^k(\cdot | s, \tilde{\pi}^k(s)) - P(\cdot | s, \pi^k(s))\|_1 \end{aligned} \quad (6)$$

$$+ \sum_{t=t_k}^{t_{k+1}-1} \left(\sum_{s' \in S} P(s' | s^t, a^t) h^k(s') - h^k(s^t) \right) \quad (7)$$

$$+ \sum_{s \in S} \nu^k(s) (\tilde{r}^k(s, \tilde{\pi}^k(s)) - r(s, \pi^k(s))), \quad (8)$$

where $h^k(s) = h(\widetilde{M}^k, s)$, and the last inequality follows from standard arguments (Jaksch et al., 2010) since $h^k(s) \leq D$ similarly to Lemma 3 in (Xu & Tewari, 2020). We now bound each term separately.

Term (7). We can add and subtract $h^k(s^{t+1})$ to term (7), and then when we sum it across all episodes, we obtain a telescopic sum that is bounded by KD for all episode switches, plus a martingale difference sequence bounded by event F_{Az}^P . That is,

$$\sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \left(\sum_{s' \in S} P(s' | s^t, a^t) h^k(s') - h^k(s^t) \right) \leq O\left(D\sqrt{T \log \frac{T}{\delta}} + KD\right).$$

Term (6). Let \lesssim represent \leq up to numerical constants, and denote $x = (s, \pi^k(s))$, $\tilde{x} = (s, \tilde{\pi}^k(s))$ and $\tilde{\pi}^k(s) = (\pi^k(s), s_n^k(s), Z_1^k(s), \dots, Z_d^k(s), z_1^k(s), \dots, z_\ell^k(s))$. We can bound the distance between P and \tilde{P}^k by the sum of distances between P_i and \tilde{P}_i^k (Osband & Van Roy, 2014), i.e.,

$$\begin{aligned} \|\tilde{P}^k(\cdot | \tilde{x}) - P(\cdot | x)\|_1 &\leq \sum_{i=1}^d \|\tilde{P}_i^k(\cdot | x[Z_i^k(s)]) - P_i(\cdot | x[Z_i^P])\|_1 \\ &\leq \sum_{i=1}^d \|\tilde{P}_i^k(\cdot | x[Z_i^k(s)]) - \bar{P}_{i, Z_i^k(s)}^k(\cdot | x[Z_i^k(s)])\|_1 \end{aligned} \quad (9)$$

$$+ \sum_{i=1}^d \|\bar{P}_{i, Z_i^k(s)}^k(\cdot | x[Z_i^k(s)]) - \bar{P}_{i, Z_i^P}^k(\cdot | x[Z_i^P])\|_1 \quad (10)$$

$$+ \sum_{i=1}^d \|\bar{P}_{i, Z_i^P}^k(\cdot | x[Z_i^P]) - P_i(\cdot | x[Z_i^P])\|_1 \quad (11)$$

$$\leq \sum_{i=1}^d \sum_{w \in S_i} \epsilon_{i, Z_i^k(s)}^k(w | x[Z_i^k(s)]) + 4 \cdot \epsilon_{i, Z_i^P \cup Z_i^k(s)}^k(w | x[Z_i^P \cup Z_i^k(s)]) + \epsilon_{i, Z_i^P}^k(w | x[Z_i^P])$$

$$\lesssim \sum_{i=1}^d \sqrt{\frac{|S_i| \log\left(\frac{dLWT}{\delta}\right)}{\max\{N_{Z_i^P \cup Z_i^k(s)}^k(x[Z_i^P \cup Z_i^k(s)]), 1\}}} + \frac{|S_i| \log\left(\frac{dLWT}{\delta}\right)}{\max\{N_{Z_i^P \cup Z_i^k(s)}^k(x[Z_i^P \cup Z_i^k(s)]), 1\}},$$

where term (9) is bounded by the construction of the optimistic MDP, and term (11) is bounded by event F^P . Term (10) is bounded because the policy $\tilde{\pi}^k$ chooses only consistent scopes. Since $Z_i^k(s)$ and Z_i^P are both consistent (outside the failure event), we have that $\bar{P}_{i,Z_i^k(s)}^k$ and $\bar{P}_{i,Z_i^P}^k$ are both close to $\bar{P}_{i,Z_i^P \cup Z_i^k(s)}^k$. Thus, we can bound term (6) as follows

$$\begin{aligned}
 \sum_{k=1}^K (6) &\leq D \sum_{k=1}^K \sum_{s \in S} \nu^k(s) \|\tilde{P}^k(\cdot | s, \tilde{\pi}^k(s)) - P(\cdot | s, \pi^k(s))\|_1 \\
 &\lesssim D \sum_{k=1}^K \sum_{s \in S} \sum_{i=1}^d \nu^k(s) \sqrt{\frac{|S_i| \log(\frac{dLWT}{\delta})}{\max\{N_{Z_i^P \cup Z_i^k(s)}^k(x[Z_i^P \cup Z_i^k(s)]), 1\}}} \\
 &\quad + D \sum_{k=1}^K \sum_{s \in S} \sum_{i=1}^d \frac{\nu^k(s) |S_i| \log(\frac{dLWT}{\delta})}{\max\{N_{Z_i^P \cup Z_i^k(s)}^k(x[Z_i^P \cup Z_i^k(s)]), 1\}} \\
 &\lesssim D \sum_{k=1}^K \sum_{i=1}^d \sum_{Z:|Z|=m} \sum_{v \in X[Z_i^P \cup Z]} \nu_{Z_i^P \cup Z}^k(v) \sqrt{\frac{|S_i| \log(\frac{dLWT}{\delta})}{\max\{N_{Z_i^P \cup Z}^k(v), 1\}}} \\
 &\quad + D \sum_{k=1}^K \sum_{i=1}^d \sum_{Z:|Z|=m} \sum_{v \in X[Z_i^P \cup Z]} \frac{\nu_{Z_i^P \cup Z}^k(v) |S_i| \log(\frac{dLWT}{\delta})}{\max\{N_{Z_i^P \cup Z}^k(v), 1\}} \\
 &\lesssim D \sum_{i=1}^d \sum_{Z:|Z|=m} \sum_{v \in X[Z_i^P \cup Z]} \sqrt{N_{Z_i^P \cup Z}^{K+1}(v) |S_i| \log(\frac{dLWT}{\delta})} + |S_i| \log(\frac{dLWT}{\delta}) \log T \\
 &\lesssim D \sum_{i=1}^d \sum_{Z:|Z|=m} \sqrt{|X[Z_i^P \cup Z]| \sum_{v \in X[Z_i^P \cup Z]} N_{Z_i^P \cup Z}^{K+1}(v) |S_i| \log(\frac{dLWT}{\delta})} \\
 &\quad + D \sum_{i=1}^d \sum_{Z:|Z|=m} \sum_{v \in X[Z_i^P \cup Z]} |S_i| \log(\frac{dLWT}{\delta}) \log T \\
 &\lesssim D \sum_{i=1}^d \sum_{Z:|Z|=m} \sqrt{|X[Z_i^P \cup Z]| |S_i| T \log(\frac{dLWT}{\delta})} \\
 &\quad + D \sum_{i=1}^d \sum_{Z:|Z|=m} |X[Z_i^P \cup Z]| |S_i| \log(\frac{dLWT}{\delta}) \log T,
 \end{aligned}$$

where the third inequality follows from our construction of the episodes as doubling number of visits to some scope-sized state-action pair (specifically, from Lemma 19 in (Jaksch et al., 2010) and Lemma B.18 in (Rosenberg et al., 2020)), the forth inequality follows from Jensen's inequality, and the last one because $\sum_{v \in X[Z_i^P \cup Z]} N_{Z_i^P \cup Z}^{K+1}(v) \leq T$.

Term (8). We can bound the distance between r and \tilde{r}^k by the sum of distances between r_j and \tilde{r}_j^k ,

$$\begin{aligned}
 \tilde{r}^k(s, \tilde{\pi}^k(s)) - r(s, \pi^k(s)) &= \frac{1}{\ell} \sum_{j=1}^{\ell} \tilde{r}_j^k(\tilde{x}[z_j^k(s)]) - r_j(x[Z_j^r]) \\
 &= \underbrace{\frac{1}{\ell} \sum_{j=1}^{\ell} \tilde{r}_j^k(\tilde{x}[z_j^k(s)]) - \bar{r}_j(x[z_j^k(s)])}_{(a)} + \underbrace{\frac{1}{\ell} \sum_{j=1}^{\ell} \bar{r}_j^k(x[z_j^k(s)]) - \bar{r}_j(x[Z_j^r])}_{(b)} \\
 &\quad + \underbrace{\frac{1}{\ell} \sum_{j=1}^{\ell} \bar{r}_j^k(x[Z_j^r]) - r_j(x[Z_j^r])}_{(c)} \\
 &\leq \frac{1}{\ell} \sum_{j=1}^{\ell} \epsilon_{z_j^k(s)}^k(x[z_j^k(s)]) + 4 \cdot \epsilon_{Z_j^r \cup z_j^k(s)}^k(x[Z_j^r \cup z_j^k(s)]) + \epsilon_{Z_j^r}^k(x[Z_j^r]) \\
 &\lesssim \frac{1}{\ell} \sum_{j=1}^{\ell} \sqrt{\frac{\log(\frac{dLWT}{\delta})}{\max\{N_{Z_j^r \cup z_j^k(s)}^k(x[Z_j^r \cup z_j^k(s)]), 1\}}},
 \end{aligned}$$

where (a) is bounded by the construction of the optimistic MDP, and (c) is bounded by event F^r . (b) is bounded because the policy $\tilde{\pi}^k$ chooses only consistent reward scopes. Since $z_j^k(s)$ and Z_j^r are both consistent (outside the failure event), we have that $\tilde{r}_{j, z_j^k(s)}^k$ and $\bar{r}_{j, Z_j^r}^k$ are both close to $\bar{r}_{j, Z_j^r \cup z_j^k(s)}^k$. Thus, we can bound term (8) as follows

$$\begin{aligned}
 \sum_{k=1}^K (8) &= \frac{1}{\ell} \sum_{k=1}^K \sum_{s \in S} \sum_{j=1}^{\ell} \nu^k(s) (\tilde{r}^k(s, \tilde{\pi}^k(s)) - r(s, \pi^k(s))) \\
 &\lesssim \frac{1}{\ell} \sum_{k=1}^K \sum_{s \in S} \sum_{j=1}^{\ell} \nu^k(s) \sqrt{\frac{\log(\frac{dLWT}{\delta})}{\max\{N_{Z_j^r \cup z_j^k(s)}^k(x[Z_j^r \cup z_j^k(s)]), 1\}}} \\
 &\lesssim \frac{1}{\ell} \sum_{k=1}^K \sum_{j=1}^{\ell} \sum_{Z: |Z|=m} \sum_{v \in X[Z_j^r \cup Z]} \nu_{Z_j^r \cup Z}^k(v) \sqrt{\frac{\log(\frac{dLWT}{\delta})}{\max\{N_{Z_j^r \cup Z}^k(v), 1\}}} \\
 &\lesssim \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z: |Z|=m} \sum_{v \in X[Z_j^r \cup Z]} \sqrt{N_{Z_j^r \cup Z}^{K+1}(v) \log(\frac{dLWT}{\delta})} \\
 &\lesssim \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z: |Z|=m} \sqrt{|X[Z_j^r \cup Z]| \sum_{v \in X[Z_j^r \cup Z]} N_{Z_j^r \cup Z}^{K+1}(v) \log(\frac{dLWT}{\delta})} \\
 &\lesssim \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z: |Z|=m} \sqrt{|X[Z_j^r \cup Z]| T \log(\frac{dLWT}{\delta})},
 \end{aligned}$$

where the third inequality follows from our construction of the episodes as doubling number of visits to some scope-sized state-action pair (specifically, from Lemma 19 in (Jaksch et al., 2010) and Lemma B.18 in (Rosenberg et al., 2020)), the forth inequality follows from Jensen's inequality, and the last one because $\sum_{v \in X[Z_j^r \cup Z]} N_{Z_j^r \cup Z}^{K+1}(v) \leq T$.

B.6. Putting Everything Together

Taking the bounds on all the terms, and noting that the failure event occurs with probability at most δ , gives the following regret bound.

$$\begin{aligned}
 \text{Reg}_T(M) &\lesssim \sqrt{T \log \frac{T}{\delta}} + D \sqrt{T \log \frac{T}{\delta}} + KD + \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z:|Z|=m} \sqrt{|X[Z_j^r \cup Z]| T \log \left(\frac{dLWT}{\delta} \right)} \\
 &\quad + D \sum_{i=1}^d \sum_{Z:|Z|=m} \sqrt{|X[Z_i^P \cup Z]| |S_i| T \log \left(\frac{dLWT}{\delta} \right)} \\
 &\quad + D \sum_{i=1}^d \sum_{Z:|Z|=m} \sum_{v \in X[Z_i^P \cup Z]} |S_i| \log \left(\frac{dLWT}{\delta} \right) \log T \\
 &\lesssim \sum_{i=1}^d \sum_{Z:|Z|=m} D \sqrt{|X[Z_i^P \cup Z]| |S_i| T \log \left(\frac{dLWT}{\delta} \right)} \\
 &\quad + \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z:|Z|=m} \sqrt{|X[Z_j^r \cup Z]| T \log \left(\frac{dLWT}{\delta} \right)} \\
 &\quad + \sum_{i=1}^d \sum_{Z:|Z|=m} D |X[Z_i^P \cup Z]| |S_i| \log^2 \left(\frac{dLWT}{\delta} \right) \\
 &\quad + \sum_{Z:|Z|=m} \sum_{Z':|Z'|=m} D |X[Z \cup Z']| \log T \\
 &\lesssim \binom{n}{m} dD \sqrt{L^2 W T \log \left(\frac{dLWT}{\delta} \right)} + \binom{n}{m} dDL^2 W \log^2 \left(\frac{dLWT}{\delta} \right) \\
 &\quad + \binom{n}{m}^2 DL^2 \log T,
 \end{aligned}$$

where the second inequality follows because there are at most $\log T$ episodes for each pair of scopes $Z \neq Z'$ of size m and $v \in X[Z \cup Z']$.

C. The NFA-DORL Algorithm

Algorithm 6 NFA-DORL

Input: confidence parameter δ , scopes $\{Z_i^P\}_{i=1}^d$, reward scopes $\{Z_j^r\}_{j=1}^\ell$, state space $S = \{S_i\}_{i=1}^d$, action space A .

Initialization

Initialize total visit counters N , in-episode visit counters ν and reward summation variables r :

for $a \in A$ and $j = 1, \dots, \ell$ and $v_j \in S[Z_j^r]$ and $i = 1, \dots, d$ and $v_i \in S[Z_i^P]$ and $w \in S_i$ **do**

$r_{j,Z_j^r}(v_j, a) \leftarrow 0, N_{Z_j^r}^0(v_j, a) \leftarrow 0, \nu_{Z_j^r}^0(v_j, a) \leftarrow 0, N_{i,Z_i^P}^0(v_i, a, w) \leftarrow 0, \nu_{i,Z_i^P}^0(v_i, a, w) \leftarrow 0, N_{Z_i^P}^0(v_i, a) \leftarrow 0, \nu_{Z_i^P}^0(v_i, a) \leftarrow 0.$

end for

Initialize time steps counter: $t \leftarrow 1$, and observe initial state s^1 .

for $k = 1, 2, \dots$ **do**

Start New Episode

Set episode starting time: $t_k \leftarrow t$.

for $a \in A$ and $j = 1, \dots, \ell$ and $v_j \in S[Z_j^r]$ and $i = 1, \dots, d$ and $v_i \in S[Z_i^P]$ and $w \in S_i$ **do**

Update visit counters: $\nu_{Z_i^P}^k(v_i, a) \leftarrow 0, \nu_{Z_j^r}^k(v_j, a) \leftarrow 0, \nu_{i,Z_i^P}^k(v_i, a, w) \leftarrow 0, N_{Z_i^P}^k(v_i, a) \leftarrow N_{Z_i^P}^{k-1}(v_i, a) + \nu_{Z_i^P}^{k-1}(v_i, a), N_{Z_j^r}^k(v_j, a) \leftarrow N_{Z_j^r}^{k-1}(v_j, a) + \nu_{Z_j^r}^{k-1}(v_j, a), N_{i,Z_i^P}^k(v_i, a, w) \leftarrow N_{i,Z_i^P}^{k-1}(v_i, a, w) + \nu_{i,Z_i^P}^{k-1}(v_i, a, w).$

Compute empirical transitions and rewards: $\bar{P}_{i,Z_i^P}^k(w|v_i, a) = \frac{N_{i,Z_i^P}^k(v_i, a, w)}{\max\{N_{Z_i^P}^k(v_i, a), 1\}}, \bar{r}_{j,Z_j^r}^k(v_j, a) = \frac{r_{j,Z_j^r}(v_j, a)}{\max\{N_{Z_j^r}^k(v_j, a), 1\}}.$

Set confidence bounds (and set $\mathcal{W}_{i,Z_i^P}^k(w|v_i, a) = \min\{\epsilon_{i,Z_i^P}^k(w|v_i, a), \bar{P}_{i,Z_i^P}^k(w|v_i, a)\}$ where $\tau^k = \log \frac{6dW L t_k}{\delta}$):

$$\epsilon_{i,Z_i^P}^k(w | v_i, a) = \sqrt{\frac{18\bar{P}_{i,Z_i^P}^k(w | v_i, a)\tau^k}{\max\{N_{Z_i^P}^k(v_i, a), 1\}}} + \frac{18\tau^k}{\max\{N_{Z_i^P}^k(v_i, a), 1\}} \quad ; \quad \epsilon_{Z_j^r}^k(v_j, a) = \sqrt{\frac{18\tau^k}{\max\{N_{Z_j^r}^k(v_j, a), 1\}}}$$

end for

Construct optimistic MDP \widetilde{M}^k and compute optimistic policy π^k (Algorithm 7).

Execute Policy

while $\nu_Z^k(s^t[Z], \pi^k(s^t)) < N_Z^k(s^t[Z], \pi^k(s^t))$ for every $Z \in \{Z_1^P, \dots, Z_d^P, Z_1^r, \dots, Z_\ell^r\}$ **do**

Play action $a^t = \pi^k(s^t)$, observe next state s^{t+1} and earn reward $r^t = \frac{1}{\ell} \sum_{j=1}^\ell r_j^t$.

Update in-episode counters and reward summation variables:

for $i = 1, \dots, d$ and $j = 1, \dots, \ell$ **do**

$\nu_{Z_i^P}^k(s^t[Z_i^P], a^t) \leftarrow \nu_{Z_i^P}^k(s^t[Z_i^P], a^t) + 1, \nu_{Z_j^r}^k(s^t[Z_j^r], a^t) \leftarrow \nu_{Z_j^r}^k(s^t[Z_j^r], a^t) + 1.$

$\nu_{i,Z_i^P}^k(s^t[Z_i^P], a^t, s^{t+1}[i]) \leftarrow \nu_{i,Z_i^P}^k(s^t[Z_i^P], a^t, s^{t+1}[i]) + 1.$

$r_{j,Z_j^r}(s^t[Z_j^r], a^t) \leftarrow r_{j,Z_j^r}(s^t[Z_j^r], a^t) + r_j^t.$

end for

advance time: $t \leftarrow t + 1$.

end while

end for

Algorithm 7 NFA-DORL Compute Optimistic Policy π^k

Construct MDP: $\widetilde{M}^k = (\widetilde{S}, \widetilde{A}, \widetilde{P}^k, \widetilde{r}^k)$.

Define action space: $\widetilde{A} = A \cup (\bigcup_{i=1}^d S_i)$.

Define state space: $\widetilde{S} = S \times \{0, 1, \dots, d+1\} \times A \times S \times \{0, 1\}$.

Define reward function for $j = 1, \dots, \ell$:

$$\widetilde{r}_j^k((s, h, a', s', b), a) = \begin{cases} \min\{1, \bar{r}_{j, Z_j^r}^k(s[Z_j^r], a) + \epsilon_{Z_j^r}^k(s[Z_j^r], a)\}, & b = 1, h = 0, a \in A \\ 0, & \text{otherwise} \end{cases}$$

Define transition function $\widetilde{P}^k(\tilde{s}' \mid \tilde{s}, \tilde{a}) = \prod_{\tau=1}^{2d+3} \widetilde{P}_\tau^k(\tilde{s}'[\tau] \mid \tilde{s}, \tilde{a})$ as follows:

- The counter factor (factor $d+1$) counts deterministically modulo $d+2$.
- The action factor (factor $d+2$) takes the action played by the agent when the counter is 0, and otherwise copies the value from the $(d+2)$ -th factor of the previous state.
- The last factor checks that all actions are legal. It starts at 1 and changes to 0 if the taken action a satisfies (1) $a \notin A$ when the counter is 0; (2) $a \notin S_i$ when the counter is i (for $i = 1, \dots, d$).
- For $i = 1, \dots, d$, the i -th factor is taken from factor $i+1+d$ of the previous state when the counter is $d+1$, and otherwise copies the value from the i -th factor of the previous state.
- For $i = 1, \dots, d$, the $(i+2+d)$ -th factor is taken from factor $i+2+d$ of the previous state when the counter is not i , and otherwise performs the optimistic transition of factor i (if the action is not in S_i transition arbitrarily), i.e.,

$$\begin{aligned} \widetilde{P}_{i+2+d}^k(w_i \mid (s, i, a, s', b), w) &= \bar{P}_{i, Z_i^P}^k(w_i \mid s[Z_i^P], a) - \mathcal{W}_{i, Z_i^P}^k(w_i \mid s[Z_i^P], a) \\ &\quad + \mathbb{I}\{w_i = w\} \cdot \sum_{w' \in S_i} \mathcal{W}_{i, Z_i^P}^k(w' \mid s[Z_i^P], a). \end{aligned}$$

Compute optimal policy $\tilde{\pi}^k$ of \widetilde{M}^k using oracle.

Extract optimistic policy: $\pi^k(s) = \tilde{\pi}^k((s, 0, \perp))$.

D. Proof of Theorem 2

The proof relies on the MDP $M' = (\tilde{S}, A, P', r')$ (described in [Section 5](#)) that models M but stretches each time step to $d + 2$ steps. Given a trajectory $(s^t, a^t)_{t=1, \dots, T}$ in M , we map it to a trajectory $(s^{t,h}, a^{t,h})_{t=1, \dots, T, h=0, 1, \dots, d+1}$ in M' as follows:

- $s^{t,0} = (s^t, 0, \perp)$ and $a^{t,0} = a^t$.
- $s^{t,1} = (s^t, 1, a^t, \perp)$ and $a^{t,1}$ is arbitrary.
- $s^{t,i+1} = (s^t, i+1, a^t, s^{t+1}[1], \dots, s^{t+1}[i], \perp)$ for $i = 1, \dots, d$ and $a^{t,i+1}$ is arbitrary.

Moreover, we slightly abuse notation as follows. For a policy π in M , we use the same notation π also for the policy in M' that plays according to π . That is, $\pi(s^{t,0}) = \pi((s^t, 0, \perp)) = \pi(s^t)$ and $\pi(s^{t,h})$ is arbitrary for $h > 0$ as the policy has no effect in these steps.

The failure events for the algorithm are similar to [Appendix B.2](#). Recall that $\lambda^*(M') = \frac{\lambda^*(M)}{d+2}$ and therefore we can write:

$$\begin{aligned}
 \text{Reg}_T(M) &= \sum_{t=1}^T (\lambda^*(M) - r^t) \\
 &= \sum_{t=1}^T (\lambda^*(M) - r(s^t, a^t)) + \sum_{t=1}^T (r(s^t, a^t) - r^t) \\
 &\leq \sum_{t=1}^T (\lambda^*(M) - r(s^t, a^t)) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \\
 &= \sum_{t=1}^T \left(\frac{\lambda^*(M)}{d+2} - r(s^t, a^t)\right) + \sum_{t=1}^T \sum_{h=1}^{d+1} \left(\frac{\lambda^*(M)}{d+2} - 0\right) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \\
 &= \sum_{t=1}^T \sum_{h=0}^{d+1} (\lambda^*(M') - r'(s^{t,h}, a^{t,h})) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \\
 &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} (\lambda^*(M') - r'(s^{t,h}, \pi^k(s^{t,h}))) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \\
 &\leq \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} (\lambda^*(\tilde{M}^k) - r'(s^{t,h}, \pi^k(s^{t,h}))) + O\left(\sqrt{T \log \frac{T}{\delta}}\right) \\
 &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} (\lambda^*(\tilde{M}^k) - \tilde{r}^k(s^{t,h}, \pi^k(s^{t,h}))) \tag{12}
 \end{aligned}$$

$$\begin{aligned}
 &+ \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} (\tilde{r}^k(s^{t,h}, \pi^k(s^{t,h})) - r'(s^{t,h}, \pi^k(s^{t,h}))) \\
 &+ O\left(\sqrt{T \log \frac{T}{\delta}}\right), \tag{13}
 \end{aligned}$$

where the last inequality is by optimism which is proven similarly to [Appendix B.4](#).

Term (13). Notice that the reward is zero when the counter is not 0 and therefore

$$\begin{aligned}
 (13) &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} (\bar{r}^k(s^{t,0}, \pi^k(s^{t,0})) - r'(s^{t,0}, \pi^k(s^{t,0}))) \\
 &\leq \frac{1}{\ell} \sum_{k=1}^K \sum_{s \in S} \sum_{j=1}^{\ell} \nu^k(s) (\bar{r}_{j, Z_j^r}^k(s[Z_j^r], \pi^k(s)) - r_j(s[Z_j^r], \pi^k(s)) + \epsilon_{Z_j^r}^k(s[Z_j^r], \pi^k(s))) \\
 &\leq \frac{1}{\ell} \sum_{k=1}^K \sum_{s \in S} \sum_{j=1}^{\ell} \nu^k(s) \cdot 2\epsilon_{Z_j^r}^k(s[Z_j^r], \pi^k(s)) \\
 &\lesssim \frac{1}{\ell} \sum_{k=1}^K \sum_{j=1}^{\ell} \sum_{v \in S[Z_j^r]} \sum_{a \in A} \nu_{Z_j^r}^k(v, a) \sqrt{\frac{\log \frac{dWLT}{\delta}}{\max\{N_{Z_j^r}^k(v, a), 1\}}} \\
 &\lesssim \frac{1}{\ell} \sum_{j=1}^{\ell} \sqrt{|S[Z_j^r]| |A| T \log \frac{dWLT}{\delta}}.
 \end{aligned}$$

Term (12). By the Bellman equations in the optimistic model \widetilde{M}^k , we can write term (12) as follows

$$\begin{aligned}
 (12) &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} \left(\sum_{s' \in \widetilde{S}} \widetilde{P}^k(s' | s^{t,h}, \pi^k(s^{t,h})) h^k(s') - h^k(s^{t,h}) \right) \\
 &= \sum_{k=1}^K \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} \sum_{s' \in \widetilde{S}} (\widetilde{P}^k(s' | s^{t,h}, \pi^k(s^{t,h})) - P'(s' | s^{t,h}, \pi^k(s^{t,h}))) h^k(s') \\
 &\quad + \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} \left(\sum_{s' \in \widetilde{S}} P'(s' | s^{t,h}, \pi^k(s^{t,h})) h^k(s') - h^k(s^{t,h}) \right) \\
 &\lesssim D \sum_{k=1}^K \sum_{s \in S} \sum_{i=1}^d \sum_{w \in S_i} \nu^k(s) \epsilon_{i, Z_i^P}^k(s[Z_i^P], \pi^k(s), w) \\
 &\quad + \sum_{t=t_k}^{t_{k+1}-1} \sum_{h=0}^{d+1} \left(\sum_{s' \in \widetilde{S}} P'(s' | s^{t,h}, \pi^k(s^{t,h})) h^k(s') - h^k(s^{t,h}) \right) \\
 &\lesssim D \sum_{k=1}^K \sum_{i=1}^d \sum_{v \in S[Z_i^P]} \sum_{a \in A} \nu_{Z_i^P}^k(v, a) \left(\sqrt{\frac{|S_i| \log \frac{dWLT}{\delta}}{\max\{N_{Z_i^P}^k(v, a), 1\}}} + \frac{|S_i| \log \frac{dWLT}{\delta}}{\max\{N_{Z_i^P}^k(v, a), 1\}} \right) \\
 &\quad + KD + D \sqrt{dT \log \frac{dT}{\delta}} \\
 &\lesssim \sum_{i=1}^d D \sqrt{|S_i| |S[Z_i^P]| |A| T \log \frac{dWLT}{\delta}} + \sum_{i=1}^d D |S_i| |S[Z_i^P]| |A| \log^2 \frac{dWLT}{\delta}.
 \end{aligned}$$

The first inequality follows by the definition of P' and \widetilde{P}^k and their factored structure. The second inequality is similar to [Appendix B.5](#), while noting that the bias function in \widetilde{M}^k is bounded by D . The reason is that diameter of \widetilde{M}^k is $D(d+2)$, and that the bias function is always bounded by the diameter times the optimal gain (see [Bartlett & Tewari \(2009\)](#)).

E. Factored MDPs with Non-Factored Actions and Unknown Structure

We now adjust our SLF-UCRL algorithm to cope with non-factored actions. The idea is similar to [Section 5](#) – instead of choosing a factored action that contains the actual action and the optimistic choices for all the consistent scopes, this time step will be stretched across $2 + d(m + 1)$ steps in which the policy makes its choice sequentially. In the first step the policy picks the action, in steps $i(m + 1) - m$ to $i(m + 1) - 1$ it picks a consistent scope for factor i , step $i(m + 1)$ performs the optimistic transition of the i -th factor, and the last step completes the transition.

Thus, the action space of the optimistic MDP \widetilde{M}^k is $\widetilde{A} = A \cup (\bigcup_{i=1}^d S_i) \cup \{1, \dots, d\}$ of size $\max\{|A|, W, d\}$ compared to $|A|W^d n^d$ in our original construction. Moreover, the state space is $\widetilde{S} = S \times \{0, 1, \dots, d(m + 1) + 1\} \times A \times \{1, \dots, d\}^m \times S \times \{0, 1\}$, which is similar to [Section 5](#) up to the new factors $\{1, \dots, d\}^m$ that keep the chosen scope.

As in [Section 5](#), a state s is mapped to $(s, 0, \perp)$ and taking action $a \in A$ transitions to $(s, 1, a, \perp)$ while other actions are not legal. When the counter is between $i(m + 1) - m$ and $i(m + 1) - 1$ the legal actions are $\{1, \dots, d\}$ and the chosen indices are just stored in the state (denote them by Z). Then, the legal actions in state $(s, i(m + 1), a, Z, w_1, \dots, w_{i-1}, \perp)$ are S_i , and picking action $w \in S_i$ transitions to $(s, i(m + 1) + 1, a, Z, w_1, \dots, w_{i-1}, w_i, \perp)$ with probability

$$\bar{P}_{i,Z}^k(w_i \mid s[Z], a) = \mathcal{W}_{i,Z}^k(w_i \mid s[Z], a) + \mathbb{I}\{w_i = w\} \cdot \sum_{w' \in S_i} \mathcal{W}_{i,Z}^k(w' \mid s[Z], a).$$

At this point the validating bit also checks that Z is consistent for factor i , and turns to 0 if not. Finally, we transition from $(s, d(m + 1) + 1, a, Z', w_1, \dots, w_d, b)$ deterministically to $(s', 0, \perp)$, where $s' = (w_1, \dots, w_d) \in S$.

Just like [Section 4.2](#), the transition function of \widetilde{M}^k is no longer factored because some scopes include the entire state-action space. However, as we previously showed, we can overcome this and perform the optimistic transition according to a selected scope while maintaining small scope size by constructing the FMDP \widehat{M}^k with a “temporary” work space Ω^m , where $\Omega = \omega^n \times \omega^{n/2} \times \dots \times \omega^2 \times \omega$. Notice that it is much smaller now because we are not performing the transition for all d factors simultaneously. Thus, the oracle needs to solve an FMDP with scope size $m + 4$, number of factors $2d + m + 3 + 2nm$, size of each factor bounded by $\max\{W, |A|, d(m + 1) + 2, n\}$ and small number of actions.

Finally, a similar construction to [Section 5](#) can be used in order to bound the regret. It involves the MDP M' with state space \widetilde{S} , that stretches each time step of M for $2 + d(m + 1)$ steps but models the exact same process as M .

Theorem 6. *Running NFA-SLF-UCRL on a factored MDP with non-factored actions and unknown structure ensures, with probability at least $1 - \delta$,*

$$\text{Reg}_T(M) = \tilde{O}\left(\sum_{i=1}^d \sum_{Z: |Z|=m} D \sqrt{|S_i| |S[Z_i^P \cup Z]| |A| T} + \frac{1}{\ell} \sum_{j=1}^{\ell} \sum_{Z: |Z|=m} \sqrt{|S[Z_j^r \cup Z]| |A| T}\right).$$

F. Lower Bound

We associate an independent multi-arm bandit (MAB) problem to every tuple $(i, w_1, \dots, w_m) \in \{1, \dots, d\} \times \{1, \dots, W\}^m$. Without loss of generality we assume that the rewards of all the MABs are either 0 or 1.

Now we construct the following factored MDP $M = (S, A, P, R)$, where the state space is $S = \{0, 1, \dots, \log d + 1\} \times \{0, 1\}^{\log d} \times \{0, 1, \dots, W\}^d \times \{0, 1\}^d \times \{0, 1\}^{d/2} \times \dots \times \{0, 1\}^4 \times \{0, 1\}^2$, and the action space is non-factored of size $|A|$. Note that the state space has $3d + \log d$ factors with maximal size $\max\{W + 1, \log d + 2\}$.

The idea is to split the T time steps into blocks of $2 + \log d$ steps. In each block the agent faces a randomly chosen MAB problem (out of the dW^m independent MABs). We make sure that it cannot infer anything about the different MABs, and thus must solve them sequentially. Since the t steps lower bound for each MAB is $\Omega(\sqrt{|A|t})$, and the expected number of times that the agent faces each MAB is $\frac{T}{dW^m(2+\log d)}$, the total regret is

$$\Omega\left(\sum_{i=1}^d \sum_{v \in \{1, \dots, W\}^m} \sqrt{|A| \frac{T}{dW^m(2+\log d)}}\right) = \Omega\left(\sqrt{\frac{d}{\log d} W^m |A| T}\right).$$

We do not make the full formal argument about the relation between the lower bound and the expected number of times we encounter each MAB, but it can be found in the lower bound proof of [Rosenberg et al. \(2020\)](#) for example.

We now continue to define the FMDP that makes the agent face the MABs sequentially. There is only one reward factor. Its scope is the last two bits and the first factor (the counter). It gives a reward of 1 only when the counter is $\log d + 1$ and the last two bits contain a 1. Otherwise the reward is 0.

The transition function is defined as follows:

- The first factor is called the counter factor. It counts deterministically modulo $\log d + 2$.
- The next $\log d$ bits are called the location bits, and they determine the location of the MAB within the state. Each bit j of these $\log d$ location bits is simply changing uniformly at random, i.e., becomes 0 or 1 with probability $1/2$.
- The next d factors are called the value factors, and they give the MAB instance that is encountered by the agent at this time block. The transitions for the i -th value factor are defined as follows. When the counter is 0 denote by $x \in \{1, \dots, d\}$ the integer that the $\log d$ location bits represent. If $x \leq i < x + m$ this factor is chosen uniformly at random from $\{1, \dots, W\}$ and otherwise it is 0. When the counter is larger than 0 this factor is just 0. Note that the scope size for these factors is $\log d + 1$.
- The next d bits are called the reward bits, and they represent the rewards given by the MABs. The transitions of the j -th reward bit is defined as follows. When the counter is 1 denote by (w_1, \dots, w_m) the values of factors j to $j + m - 1$ of the d value factors. If one of them is 0 then the j -th reward bit is zero, and otherwise its value is determined by the reward of MAB (j, w_1, \dots, w_m) . When the counter is not 1 this factor is just 0. Note that the scope size for this factor is $m + 1$. Moreover, this is the only MAB instance that the agent gets any information about, which forces it to solve all the MABs sequentially.
- The final bits $\{0, 1\}^{d/2} \times \dots \times \{0, 1\}^4 \times \{0, 1\}^2$ take the d reward bits and extract whether they contain a 1 or are all 0. Notice that this encodes exactly the reward given by the current MAB. Similarly to the SLF-UCRL algorithm, this can be achieved with scope size 3 (each bit needs to consider two bits from the previous layer and the counter) and within $\log d - 1$ steps. This is done when the counter is $2, \dots, \log d$ and then the last two bits contain a 1 if the answer is yes, and are both 0 if the answer is no.

Remark (Dependence in the diameter). Our main goal in the lower bound was to show that polynomial dependence in the number of factors and exponential dependence in the scope size are necessary. This was not clear from previous lower bounds as they used scopes of size 1, and did not have a dependence on d (because there was an average over factors). Therefore, we did not get a dependence on the diameter D . While getting the dependence in D might be tricky in the average-reward setting, it is straightforward to get a \sqrt{H} dependence in the finite-horizon setting (with horizon H). In the finite-horizon setting our construction is similar such that in each episode one MAB is faced and the agent earns the same reward for $H - (\log d + 2)$ steps (after the reward is chosen in the first $\log d + 2$ steps, the agent has no control and just keeps receiving the same reward). This gives a lower bound of $\Omega\left(\sqrt{\frac{d}{\log d} HW^m |A| T}\right)$ that matches the upper bound of [Chen et al. \(2021\)](#) (up to logarithmic factors), thus proving that this is indeed the minimax optimal regret.

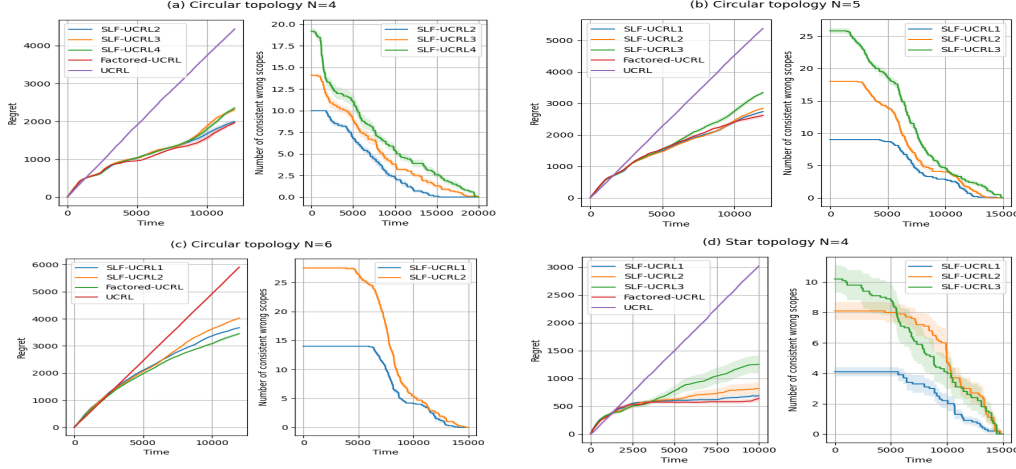


Figure 2. SLF-UCRL performance on SysAdmin domain.

G. Experiments

We perform numerical experiments to support our theoretical claims regarding the SLF-UCRL algorithm. The experiments are performed on the *SysAdmin* domain (Guestrin et al., 2003). This domain consists of N servers that are organized in a graph with a certain topology. Each server is represented by a binary variable that indicates whether or not it is working. At each time step, each server has a chance of failing, which depends on its own status and the status of the servers connected to it. There are $N + 1$ actions: N actions for rebooting a server (after which it works with high probability) and an idle action. In previous work (Guestrin et al., 2003; Xu & Tewari, 2020; Talebi et al., 2021), researchers have performed experiments with two different topologies: A circular topology in which each server is connected to the next server in the circle, and a star topology in which the servers are organized in a tree with three branches.

In each topology, the status of each server depends on at most one other server (and its own status and the action) so the scope size is $m = 3$. The number of state factors is $d = N$, the size of each state factor is $W = 2$, the action space is of size $|A| = N + 1$. Thus the state-action space is of total size $|S \times A| = 2^N(N + 1)$ which is exponential in the number of servers N , while the representation of this FMDP is only polynomial in N .

In our experiments, we set $\delta = 0.01$ and report for each domain the average results over 10 independent experiments (and the standard error in the shaded area). Our code is based on the code of Talebi et al. (2021) which was made publicly available via <https://github.com/aig-upf/dbn-ucrl>. To that code we added a new class called SLFUCRL that implements our algorithm, i.e., maintains sets of consistent scopes (we focus on transitions and assume that the reward scopes are known) and integrates them within the optimistic policy computation. For the planning oracle, we simply solve the full optimistic MDP using extended value iteration (up to some error). We note that for finite-horizon we could solve the optimistic MDP exactly.

Figure 2 shows that in a variety of scenarios the SLF-UCRL algorithm acts as predicted by our theoretical guarantees. In (a),(b),(c) we used the circular topology with $N = 4, 5, 6$ servers, respectively, and in (d) we used the star topology with $N = 4$ servers. We can see that SLF-UCRL eliminates the wrong scopes, and that its regret is comparable to that of the Factored-UCRL algorithm (Osband & Van Roy, 2014) that has full knowledge of the factored structure in advance. Moreover, the regret of SLF-UCRL is significantly better than that of the UCRL algorithm (Jaksch et al., 2010) that simply ignores the existence of a factored structure, demonstrating the importance of learning the structure (as the SLF-UCRL algorithm does). “SLF-UCRL i ” refers to i factors whose scope needs to be learned, demonstrating that additional domain knowledge can be easily integrated into the SLF-UCRL algorithm and help it both in terms of regret and in terms of computational complexity (which does not appear in the graphs).

Note that for experiment (a) we used a slightly stricter threshold (by a factor of 10) to eliminate inconsistent scopes, but then we saw that we can eliminate them faster without eliminating the true scopes. This is why it takes 20000 steps (and not 15000) to eliminate all scopes in experiment (a).