Goal-oriented exploration for reinforcement learning

Jean Tarbouriech

Université de Lille
École Doctorale MADIS

Thèse de Doctorat

Spécialité **Informatique**

présentée par
Jean Tarbouriech

Goal-Oriented Exploration for Reinforcement Learning

Exploration d'états buts pour l'apprentissage par renforcement

sous la direction de **Philippe Preux** et d'**Alessandro Lazaric**,
ainsi que l'encadrement de **Michal Valko**.

Soutenue publiquement le **6 juillet 2022** à **Paris**, devant le jury composé de

Aurélien **Garivier**	Professeur, École Normale Supérieure de Lyon	Rapporteur & Président
Yishay **Mansour**	Professeur, Tel Aviv University	Rapporteur
Doina **Precup**	Professeure associée, McGill University, DeepMind	Examinatrice
Michal **Valko**	Chercheur, DeepMind	Encadrant
Philippe **Preux**	Professeur, Université de Lille, Inria	Directeur de thèse
Alessandro **Lazaric**	Chercheur, Meta AI	Co-directeur de thèse

Centre de Recherche en Informatique, Signal et Automatique de Lille (CRIStAL),
UMR 9189 Équipe Scool, 59650, Villeneuve d'Ascq, France

Meta AI Université de Lille CRIStAL Centre de Recherche en Informatique, Signal et Automatique de Lille Inria

À Pops et Mops.

Remerciements

My first words of acknowledgement are for my advisors Alessandro and Michal. Thank you both for your constant scientific and human support, whether it be to challenge myself even more during the good moments, or to help me overcome the difficult moments. Alessandro, your patience, kindness and clarity not only convinced me to pursue a PhD after my internship with you but also helped me throughout my PhD experience. Michal, I am very grateful for your research drive and useful advice, as well as your care and invitations for social activities which were such a boon during the tough WFH period.

Je remercie également Philippe, mon directeur de thèse, pour sa disponibilité et la liberté qu'il m'a accordée pendant ma thèse. I would also like to express my great gratitude to all my jury members. I thank Aurélien and Yishay for taking the time to review my manuscript, as well as Doina for her thoughtful questions during the defense.

I was very lucky to collaborate with some brilliant researchers and fellow PhD students. Matteo, a warm thank you for your great help throughout my PhD, whether it be to push a paper over the finish line or to boost my confidence during moments of doubt. I also learned a lot from my other collaborators: Evrard, Mohammad, Shekhar, Simon, Runlong, Ludovic, Pierre-Alexandre, Omar, Pierre.

J'en viens à l'exceptionnelle équipe des CIFRE qui a rendu cette expérience de thèse tellement plus agréable à vivre au quotidien. Je remercie notamment Evrard : on s'est cassés la tête sur SSP ensemble (cf. le fameux domac de rue Réaumur à 23h), on s'est plaints ensemble, et surtout on s'est marrés ensemble. Je remercie aussi Pierre-Alexandre et Jean-Baptiste qui ont complété notre quatuor magique de FIFA; un grand merci pour tous les délires partagés. Je salue aussi Guillaume, Stéphane, Gautier, Charlotte, Hubert, Lina, Rui, Baptiste, Laurent, Virginie, Léonard, Louis.

Je remercie également la team lilloise. Tout particulièrement, Omar, merci pour ta bonne humeur de tous les instants et nos moments inoubliables en soirées et voyages. Clin d'œil à nos Jitsi, Haxball et BombSquad pendant les interminables confinements; je salue également mon autre demi-frère de thèse lillois Xuedong. Je pense aussi à Pierre, Réda, Edouard, Antoine, Nathan, Sarah, Dorian, Julien.

De belles amitiés nées avant la thèse ont rendu mon expérience doctorale bien plus sympa. Un merci spécial à Ibrahim, Mhamed, Michel, Julien, Edouard, Samba, Eduardo, Sami, Kevin, Victor, Louis, pour les UrbanSoccer de feu et les soirées de folie.

Ces trois dernières années furent riches, intenses, surprenantes, marquées de plein fouet par les multiples confinements. Je tiens à remercier Akiko qui plus que personne a partagé cette aventure avec moi. Tu as été mon roc pendant ma thèse et je te remercie infiniment pour tout ce que tu m'as apporté.

Pour finir, je n'aurais pas traversé ces années sans ma famille qui m'a entouré, et je tiens à souligner à quel point je vous en suis reconnaissant. Paul et Joseph, merci pour les délires fraternels. Merci à mes parents pour leur affection et encouragements constants : je ne serais pas arrivé jusque là sans vous.

Résumé

Apprendre à atteindre des *buts* est une compétence à acquérir à grande pertinence pratique pour des agents intelligents. Par exemple, ceci englobe de nombreux problèmes de navigation (se diriger vers telle destination), de manipulation robotique (atteindre telle position du bras robotique) ou encore certains jeux (gagner en accomplissant tel objectif). En tant qu'être vivant interagissant avec le monde, je suis constamment motivé par l'atteinte de buts, qui varient en portée et difficulté.

L'*Apprentissage par Renforcement* (AR) est un paradigme prometteur pour formaliser et apprendre des comportements d'atteinte de buts. Un but peut être modélisé comme une configuration spécifique d'états de l'environnement qui doit être atteinte par interaction séquentielle et exploration de l'environnement inconnu. Bien que divers algorithmes en AR dit "profond" aient été proposés pour ce modèle d'apprentissage conditionné par des états buts, les méthodes existantes manquent de compréhension rigoureuse, d'efficacité d'échantillonnage et de capacités polyvalentes. Il s'avère que l'analyse théorique de l'AR conditionné par des états buts demeurait très limitée, même dans le scénario basique d'un nombre fini d'états et d'actions.

Premièrement, nous nous concentrons sur le scénario *supervisé*, où un état but qui doit être atteint en minimisant l'espérance des coûts cumulés est fourni dans la définition du problème. Après avoir formalisé le problème d'apprentissage incrémental (ou "online") de ce modèle souvent appelé Plus Court Chemin Stochastique, nous introduisons deux algorithmes au regret sous-linéaire (l'un est le premier disponible dans la littérature, l'autre est quasi-optimal).

Au delà d'entraîner l'agent d'AR à résoudre une seule tâche, nous aspirons ensuite qu'il apprenne de manière autonome à résoudre une grande variété de tâches, dans l'absence de toute forme de supervision en matière de récompense. Dans ce scénario *non-supervisé*, nous préconisons que l'agent sélectionne lui-même et cherche à atteindre ses propres états buts. Nous dérivons des garanties non-asymptotiques de cette heuristique populaire dans plusieurs cadres, chacun avec son propre objectif d'exploration et ses propres difficultés techniques. En guise d'illustration, nous proposons une analyse rigoureuse du principe algorithmique de viser des états buts "incertains", que nous ancrons également dans le cadre de l'AR profond.

L'objectif et les contributions de cette thèse sont d'améliorer notre compréhension formelle de l'exploration d'états buts pour l'AR, dans les scénarios supervisés et non-supervisés. Nous espérons qu'elle peut aider à suggérer de nouvelles directions de recherche pour améliorer l'efficacité d'échantillonnage et l'interprétabilité d'algorithmes d'AR basés sur la sélection et/ou l'atteinte d'états buts dans des applications pratiques.

Abstract

Learning to reach *goals* is a competence of high practical relevance to acquire for intelligent agents. For instance, this encompasses many navigation tasks ("go to target X"), robotic manipulation ("attain position Y of the robotic arm"), or game-playing scenarios ("win the game by fulfilling objective Z"). As a living being interacting with the world, I am constantly driven by goals to reach, varying in scope and difficulty.

Reinforcement Learning (RL) holds the promise to frame and learn goal-oriented behavior. Goals can be modeled as specific configurations of the environment that must be attained via sequential interaction and exploration of the unknown environment. Although various deep RL algorithms have been proposed for goal-oriented RL, existing methods often lack principled understanding, sample efficiency and general-purpose effectiveness. In fact, very limited theoretical analysis of goal-oriented RL was available, even in the basic scenario of finitely many states and actions.

We first focus on a *supervised* scenario of goal-oriented RL, where a goal state to be reached in minimum total expected cost is provided as part of the problem definition. After formalizing the online learning problem in this setting often known as Stochastic Shortest Path (SSP), we introduce two no-regret algorithms (one is the first available in the literature, the other attains nearly optimal guarantees).

Beyond training our RL agent to solve only one task, we then aspire that it learns to autonomously solve a wide variety of tasks, in the absence of any reward supervision. In this challenging *unsupervised* RL scenario, we advocate to "Set Your Own Goals" (SYOG), which suggests the agent to learn the ability to intrinsically select and reach its own goal states. We derive finite-time guarantees of this popular heuristic in various settings, each with its specific learning objective and technical challenges. As an illustration, we propose a rigorous analysis of the algorithmic principle of targeting "uncertain" goals which we also anchor in deep RL.

The main focus and contribution of this thesis are to instigate a principled analysis of goal-oriented exploration in RL, both in the supervised and unsupervised scenarios. We hope that it helps suggest promising research directions to improve the interpretability and sample efficiency of goal-oriented RL algorithms in practical applications.

Contents

Chapter 1

Introduction

1.1 Context and Scope

As a living being interacting with the world, I am constantly driven by goals to pursue, which can vary in scope and difficulty, from perennial goals of arriving to work on time or finding food, to cognitive goals of winning a game of chess or writing a doctoral thesis, to Utopian goals of scoring a goal in a World Cup final or solving Goldbach's conjecture. How might one define a "goal"? Goal setting theorists Locke and Latham (2002, page 705) suggest:

A goal is the object or aim of an action, for example, to attain a specific standard of proficiency, usually within a specified time limit.

Alternatively, Elliot and Fryer (2008, page 245) propose:

A goal is a cognitive representation of a future object that the organism is committed to approach or avoid.

The acclaimed book "Artificial Intelligence: A Modern Approach" of Russell and Norvig (2002), which defines an agent as "anything that can be viewed as perceiving its environment through sensors and acting upon that environment through actuators", argues that along with "simple reflex", "model-based reflex" and "utility-based" agents,

Goal-based agents [*are one of the*] *four basic kinds of agent programs that embody the principles underlying almost all intelligent systems.* [...] *The agent needs some sort of goal information that describes situations that are desirable* [...] *to choose actions that achieve the goal.*

In this thesis, we study goals and goal-oriented behavior under the light of the mathematical and algorithmic framework of *Reinforcement Learning* (RL). In this paradigm, an agent sequentially interacts with an *unknown environment*, by taking an *action* in the current *state*,

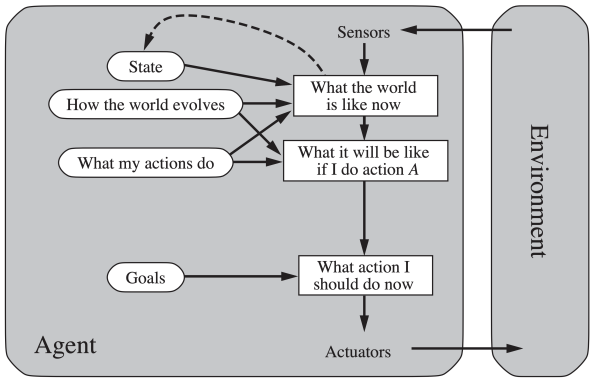


Figure 1.1 – A *goal-based agent*. It keeps track of the world state as well as a set of goals it is trying to achieve, and chooses an action that will (eventually) lead to the achievement of its goals. Figure from Russell and Norvig (2002, Figure 2.13).

transitioning to the next state and (optionally) receiving a *reward* (or incurring a *cost*). A *goal* can be seen as a *specific configuration* of the environment that must be achieved, while optionally maximizing some notion of cumulative reward (or equivalently, minimizing some notion of cumulative cost). In particular, a goal may be represented as a particular target state or a set of target states: we will frame goals as such throughout the thesis and thus use the expression *goal* or *goal state* interchangeably. The agent is said to be *goal-oriented*, or *goal-conditioned*, if its behavior is directly influenced by the goal that it pursues, until the said goal is attained (or abandoned); at which point, another goal (either the same or different) may be considered. The agent must learn how the environment behaves (*exploration*), while learning how to act optimally to reach the goal (*exploitation*). Goals can be separated into two main categories:

- A goal is said to be **extrinsic**, or **supervised**, when it is prescribed by the environment along with a goal-dependent cost function over the state-action space. The objective of the agent is to minimize the total expected cost to reach the goal state. In the special case where costs are uniform, this is equivalent to minimizing the expected time to reach the goal state.

- A goal is said to be **intrinsic**, or **unsupervised**, when it is autonomously set by the agent, which aims to reach it (by optionally minimizing an intrinsically generated cost function). This scenario is particularly relevant when the RL environment does not provide any reward/cost function nor goal to reach, or alternatively provides many of them (i.e., numerous rewards/costs and/or numerous goals), or finally when the agent chooses to ignore them if they are not informative enough.[1] This falls under the umbrella of *unsupervised RL*, which refers to methods where the agent defines alternative and/or complementary objectives that are not directly driven by an extrinsic signal.

[1]For instance, an extrinsic reward/cost signal may be *too* sparse, time-varying or delayed, while an extrinsic goal state may be *too* difficult to reach from the starting state in a reasonable amount of time.

The structure of this thesis will mirror this fundamental distinction: Part I will focus on extrinsic/supervised goals and Part II will focus on intrinsic/unsupervised goals.

The main motivation for studying goal-oriented behavior is its prevalence in practical applications. Take, for instance, many navigation tasks ("go to target X"), robotic manipulation ("attain position Y of the robotic arm"), or game-playing scenarios ("win the game by fulfilling objective Z"). Some representative RL research environments that display a goal-oriented flavor include:

- AntMaze (Fu et al., 2020), a navigation domain simulating a 8-DoF "Ant" quadraped robot to reach a fixed goal location;
- FetchReach (Plappert et al., 2018), a robotic manipulation environment that simulates a 7-DoF Fetch arm that must move the gripper to a target location;
- Breakout, an Atari 2600 game implemented in the Arcade Learning Environment (Bellemare et al., 2013), where the agent is presented with a high-dimensional visual input (210×160 RGB video at 60Hz) and its objective is to destroy all bricks on the screen.

Strikingly, *although goal-oriented RL effectively models many tasks and has garnered increasing empirical attention in deep RL, its rigorous quantitative evaluation and theoretical analysis had remained elusive at the beginning of this PhD thesis in 2019, even in the basic scenario with finitely many states and actions*.

This thesis is motivated by the objective of improving our formal understanding of goal-oriented exploration in RL. By partly filling this gap in the literature, we hope this work helps suggest promising research directions to improve the empirical performance (i.e., sample efficiency) and the underlying principles (i.e., interpretability) of practical RL algorithms in effectively generating and/or solving goal-reaching tasks.

1.2 Reinforcement Learning (RL)

In the traditional RL paradigm, an agent interacts with an environment modeled as a Markov decision process (Puterman, 2014, MDP). At each time step $t \in \mathbb{N}^\star$ (where $\mathbb{N}^\star$ denotes the set of positive integers), the environment is in a *state* $s_t \in \mathcal{S}$ and the agent takes an *action* $a_t \in \mathcal{A}$, where $\mathcal{S}$ and $\mathcal{A}$ denote the sets of possible states and actions, respectively. As a consequence, the agent transitions to a next state $s_{t+1} \in \mathcal{S}$, drawn from a conditional distribution $P(\cdot|s_t, a_t)$ that we call the *transition dynamics* $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, where $\Delta(\mathcal{X})$ denotes the set of probability distributions over a measurable set $\mathcal{X}$. At each time step t, the agent may also receive a *supervised signal* in the form of an instantaneous *cost* $c(s_t, a_t) \in [0, 1]$,[2] otherwise in the *unsupervised* case

[2]While we often consider deterministic costs for simplicity, the extension to stochastic costs drawn i.i.d. from a distribution on $[0, 1]$ with expectation $c(s_t, a_t)$ is rather straightforward.

the agent may generate its own cost function over the state-action space. Equivalently, one may translate costs into *rewards* by simply considering negation. An MDP is thus defined as the tuple $M \triangleq \langle \mathcal{S}, \mathcal{A}, s_0, P, c \rangle$, where $s_0 \in \mathcal{S}$ denotes the initial state.[3] We say that an MDP is *reward-free* (or cost-free) if it is defined as the tuple $M \triangleq \langle \mathcal{S}, \mathcal{A}, s_0, P \rangle$.

An RL agent's actions can be drawn from a state-dependent distribution $\pi(a_t \mid s_t)$, called the *policy* $\pi : \mathcal{S} \to \Delta(\mathcal{A})$. Quantifying the behavior of policies depends on the nature of the interaction that the agent has with the environment, which prescribes the objective that should be optimized. In all generality, the performance of a policy can be informed by a *time-dependent weighting* of the sequence of costs incurred by executing the policy in the environment.

Definition 1.1 (Policy return)**.** *Consider any sequence of weights $\omega : \mathbb{N}^\star \to \mathbb{R}_+$ such that there exists $\Omega \in [0, +\infty]$ that ensures the existence of the following limit*

$$\Omega \triangleq \lim_{T \to +\infty} \sum_{t=1}^{T} \omega(t). \tag{1.1}$$

Then the (possibly infinite) return U^π of a policy π is a random variable defined as the weighted sum of the instantaneous costs

$$U^\pi \triangleq \lim_{T \to +\infty} \sum_{t=1}^{T} \omega(t) \cdot c(s_t, a_t) \tag{1.2}$$

accumulated along a trajectory $\tau = (s_1, a_1, s_2, a_2, \dots)$ induced by the policy $a_t \sim \pi(\cdot|s_t)$ and transition dynamics $s_{t+1} \sim P(\cdot \mid s_t, a_t)$.

Definition 1.1 subsumes the two most common performance criteria considered in the RL literature:

- *Infinite-horizon discounted criterion*: The policy is evaluated by the infinite sum of discounted costs, where the aversion for long-term costs is controlled by a constant $\gamma \in [0, 1)$ called the *discount factor*. The weights are defined as $\omega(t) \triangleq \gamma^{t-1}$ (note that $\Omega = 1/(1-\gamma)$).
- *Finite-horizon criterion*: The policy is evaluated by its cumulative costs over a fixed length of $H \in \mathbb{N}^\star$ time steps called the *horizon*. The weights are defined as $\omega(t) \triangleq \mathbb{1}[t \leq H]$ (note that $\Omega = H$).

A third criterion, called the *infinite-horizon undiscounted criterion*, can also be characterized by Definition 1.1. In this case, the policy is evaluated by its average cumulative cost over an

[3] It is straightforward to extend to the case where the starting state s_0 is sampled from a possibly unknown distribution in $\Delta(\mathcal{S})$.

infinite interaction period, therefore the weights can be defined uniformly as $\omega(t) \triangleq 1/T$ in Equation (1.2).

Given the random variable of the policy return, taking its expectation yields the *value* function which quantifies how well the policy performs on average.

Definition 1.2 (Value functions)**.** *The state value $V^\pi(s)$ of a policy π is the expected return of the policy when starting in state $s \in \mathcal{S}$, i.e.,*

$$V^\pi(s) \triangleq \mathbb{E}\Big[U^\pi \mid s_1 = s\Big],$$

where the expectation is w.r.t. the random trajectory generated by executing π starting from state $s \in \mathcal{S}$. Similarly, the state-action value $Q^\pi(s,a)$ of a policy π is the expected return of the policy when starting in state $s \in \mathcal{S}$ and taking action $a \in \mathcal{A}$, i.e.,

$$Q^\pi(s,a) \triangleq \mathbb{E}\Big[U^\pi \mid s_1 = s, a_1 = a\Big].$$

This allows to define the generic objective of an RL agent: finding an *optimal* policy $\pi^\star$.

Definition 1.3 (Optimality)**.** *If it exists, a policy $\pi^\star$ is said to be optimal if it minimizes the value functions V^π and Q^π in every state and action. If they exist, we also define the optimal value functions $V^\star$ and $Q^\star$ as*

$$\forall s \in \mathcal{S}, \qquad V^\star(s) \triangleq V^{\pi^\star}(s) = \min_\pi V^\pi(s),$$
$$\forall (s,a) \in \mathcal{S} \times \mathcal{A}, \qquad Q^\star(s,a) \triangleq Q^{\pi^\star}(s,a) = \min_\pi Q^\pi(s,a).$$

Both the infinite-horizon discounted and finite-horizon criteria verify some convenient properties. First, due to the boundedness of costs in $[0,1]$, it is easy to see that the value function of *any* policy π starting from *any* state s is bounded, i.e., $V^\pi(s) \leq 1/(1-\gamma)$ and $V^\pi(s) \leq H$, respectively. Moreover, if $\mathcal{S}$ and $\mathcal{A}$ are finite, there always exists an optimal policy $\pi^\star$ that is stationary and deterministic (that is, a mapping from states to actions, i.e., $\pi^\star : \mathcal{S} \to \mathcal{A}$); and when P and c are known, $\pi^\star$ can be computed efficiently using standard planning techniques, e.g., value iteration, policy iteration or linear programming (Bertsekas, 1995; Puterman, 2014).

The RL scenario deals with the more realistic yet challenging setting where P and c are *unknown* to the agent. The high-level learning objective is to achieve a performance as close

and as quickly as possible to the optimal policy $\pi^\star$. Several performance measures have been introduced to evaluate RL algorithms, such as regret and sample complexity. Both criteria have been analyzed by an extensive line of research (representative works include e.g., Kearns and Singh, 2002; Brafman and Tennenholtz, 2002; Kakade, 2003; Strehl and Littman, 2008; Azar et al., 2017; Dann et al., 2017; Jin et al., 2018; Zanette and Brunskill, 2019).

Despite their thorough study (both theoretically and empirically), both the finite-horizon and infinite-horizon discounted settings assume that there exists a fixed *intrinsic horizon* (respectively H and $1/(1-\gamma)$[4]) known to the learning agent. Depending on the convention, H or γ can be explicitly part of the problem definition (i.e., selected by a higher-level agent, e.g., a human), or must be selected by the learning agent to define its optimization objective on the MDP. Either way, in many common RL applications (e.g., objective of accumulating more reward than a specific threshold, or of navigating to a specific state), it is not clear how to define H or γ to ensure that these applications can be adequately solved by optimizing the corresponding finite-horizon or discounted model. On the one hand, setting H (and/or γ) too small will generate a bias in the optimal behavior. On the other hand, setting H (and/or γ) too large will increase the range of the quantities of interest (i.e., value functions), which can lead to numerical instabilities as well as vacuous theoretical guarantees (since the majority of existing regret or sample complexity guarantees explicitly scale with H or $1/(1-\gamma)$).

As a result, carefully presetting an adequate horizon H (and/or γ) is non-trivial and it requires strong task- and environment-dependent prior knowledge. In particular, both the finite-horizon and infinite-horizon discounted criteria may poorly capture tasks where the interaction only ends if a stopping condition is *adaptively* met (i.e., if the stopping condition is not predefined but depends on the online interaction). In particular, *goal-oriented tasks*, where the objective is to minimize the total expected costs to a goal state, have an intrinsic horizon (i.e., the time to reach the goal) that (in most cases) is a *random hitting time* that depends on the agent's behavior and is *unknown* in advance. Hence, they *cannot* be effectively modeled by either the finite-horizon or infinite-horizon (discounted) criterion.

1.3 Goal-Oriented RL

Whether goals are extrinsically or intrinsically generated, formally analyzing goal-oriented behavior in RL requires, at the bare minimum, to define a *goal space* $\mathcal{G}$ which defines the set of possible goals on which the agent may condition its behavior, and a *goal-achievement function* $\Psi : \mathcal{S} \times \mathcal{G} \to \{0, 1\}$ which assesses the achievement of the goal at the agent's current state $s \in \mathcal{S}$. We focus on goals that can be expressed as target features of the state that the agent desires

[4]The quantity $1/(1-\gamma)$ uppers bounds the return of any policy under the discounted setting since the instantaneous costs are in $[0, 1]$ and $\sum_{t=1}^{\infty} \gamma^{t-1} = 1/(1-\gamma)$ for any $0 \leq \gamma < 1$.

to achieve.[5] It is often considered that there exists a known and tractable mapping $\phi : \mathcal{S} \to \mathcal{G}$ that defines a goal representation (which is usually of lower dimensionality than the state space when the latter is high-dimensional). Meanwhile, the goal-achievement function could for example be defined as $\Psi(s, g) \triangleq \mathbb{1}[d(\phi(s), g) \leq \varepsilon]$, for some metric $d : \mathcal{G} \times \mathcal{G} \to \mathbb{R}_+$ (e.g., the Euclidean distance) and a given threshold $\varepsilon \geq 0$, where $\mathbb{1}$ denotes the indicator function. Note that in practice a sparse and binary goal-driven reward signal may be derived from Ψ as $R_g(s) \triangleq \Psi(s, g)$ or $R_g(s) \triangleq \Psi(s, g) - 1$ and used to train the RL agent (Plappert et al., 2018).

Throughout most of the thesis, we consider that the MDP is *finite*, i.e., both $\mathcal{S}$ and $\mathcal{A}$ are finite with cardinalities denoted by S and A, respectively. While this restricts the scope of our investigation, we will see that even in this basic scenario our understanding of goal-oriented RL had remained elusive, thus making it a natural starting point for future research on more applicable settings. Formally, we consider that $\mathcal{G} \subseteq \mathcal{S}$, i.e., goals can be expressed as states of the environment that the agent desires to reach; and that a goal $g \in \mathcal{G}$ is achieved when the current state s coincides with g, i.e., when $\Psi(s, g) \triangleq \mathbb{1}[s = g]$.

According the goal-oriented criterion, the agent's objective is to minimize the expected cumulative costs until the goal state g is reached (the goal and cost function can be either extrinsically or intrinsically generated). This is often called the Stochastic Shortest Path (SSP) objective (Bertsekas, 1995). It can be formalized by instantiating Definition 1.1 with the weights $\omega(t) \triangleq \prod_{i=1}^{t} \mathbb{1}[s_i \neq g]$, which we can equivalently write $\omega(t) \triangleq \mathbb{1}[t \leq \inf\{i \geq 0 : s_{i+1} = g\}]$. This choice of weights captures that the agent's interaction should last *as long as the goal state g has not been reached*. Compared to the finite-horizon or infinite-horizon discounted weights, the SSP weights possess the unique feature of being *random variables* that depend on the trajectory (i.e., the state sequence $(s_t)_{t \geq 1}$) and thus on the policy. Indexing the weights by the policy π and denoting by $\omega_\pi(t)$ the associated random variable, we set $\Omega^\pi \triangleq \lim_{T \to +\infty} \mathbb{E}[\sum_{t=1}^{T} \omega_\pi(t)]$.

SSP is a general criterion which includes both the finite-horizon and discounted criteria as special cases, as we will see in Chapter 2. With such modeling flexibility comes new technical challenges. Indeed, we first notice that there may exist policies π for which $\Omega^\pi = +\infty$; in fact, in environments where the goal is non-trivial to reach, this will be the case for many policies. Moreover, an infinite Ω^π may imply an infinite value function V^π at some states; for example this holds if all non-goal costs are positive (i.e., $c(s, a) > 0$ for all $(s, a) \in \mathcal{S} \setminus \{g\} \times \mathcal{A}$). In addition, there may not exist a policy $\pi^\star$ that minimizes the value function as defined in Definition 1.3. Finally, even if there exists one, it may never reach the goal, i.e., it may hold that $\Omega^{\pi^\star} = +\infty$. These observations give us a glimpse that careful technical attention and extra assumptions are required to properly optimize the goal-oriented criterion, which will be the focus of Chapters 2 and 3.

[5] In all generality, some goals cannot be expressed as target state features, see for instance the survey of Colas et al. (2020, Section 4) for a general typology of goal representations in the RL literature.

Criterion	Finite-horizon	Infinite-horizon discounted	Goal-oriented (a.k.a. SSP)
Weights $\omega(t)$ of policy return	$\mathbb{1}[t \leq H]$	γ^{t-1}	$\prod_{i=1}^{t} \mathbb{1}[s_i \neq g]$ $= \mathbb{1}[t \leq \inf\{i \geq 0 : s_{i+1} = g\}]$
Do the weights require some parameter / prior knowledge?	Horizon H	Discount γ	None (apart from goal state identity g)
Do the weights adapt to the agent's behavior?	No	No	Yes
Intrinsic horizon Ω	H	$\frac{1}{1-\gamma}$	? ($\to +\infty$ for some policies)

Table 1.1 – Characteristics of the weights of policy return (see Definition 1.1) for different performance criteria: finite-horizon, infinite-horizon discounted, and goal-oriented (a.k.a. stochastic shortest path).

1.4 Outline and Contributions

The general research question driving this thesis can be framed as:

Under which learning objectives, environment assumptions and algorithmic designs
can we perform provably efficient exploration in goal-oriented RL,
driven by either extrinsically or intrinsically generated goals?

As a first step in such an endeavour, in Part I we focus on an extrinsically predefined goal state and tackle the unaddressed research question of how to effectively reach it in minimum total expected cost. In Chapter 3, we formalize the online learning problem in the Stochastic Shortest Path setting (online SSP in short) and identify the unique technical challenges that arise, with a particular focus on the regret minimization framework. In Chapter 4, we propose the first no-regret algorithm for online SSP. In Chapter 5, we advance the state-of-the-art for online SSP by designing an algorithm that is simultaneously regret-optimal and fully agnostic to the difficulty of reaching the goal (a.k.a. parameter-free). This result conveys the conceptual message that it is possible to design intelligent agents that are able to adapt to the unknown difficulty of the task at hand (i.e., the goal-reaching horizon) without sacrificing learning performance.

Beyond training our RL agent to solve only one goal-reaching task, in Part II we aspire that it learns to autonomously solve a wide variety of tasks, in the absence of any reward/cost/goal supervision. In this challenging *unsupervised* RL scenario, we advocate to "Set Your Own Goals" — in short, SYOG — which suggests the agent to learn the ability to intrinsically select and reach its own goal states. This general-purpose technique has already been widely studied from an empirical viewpoint, as we review in Chapter 6. Our main contribution is a thorough

and formal analysis of SYOG in various settings, each with its specific exploration objective and technical challenges. At a high level, we demonstrate how SYOG allows to provably efficiently perform either *total*, *local*, or *incremental coverage* of the state space, respectively. ▷ In Chapter 7, we study SYOG in *reward-free reset-free communicating*[6] MDPs. We introduce a new decoupled approach for online RL which isolates the (objective-specific) prescription of state-action samples to collect and the (objective-agnostic) goal-driven collection of the desired samples. We show how this decoupled approach allows us to tackle in a unifying manner a variety of "unsupervised RL" objectives (e.g., cover the state space, estimate the transition dynamics, or learn a set of accurate goal-reaching policies). ▷ In Chapter 8, we examine SYOG in *reward-free resettable*[7] MDPs. We introduce the multi-goal exploration objective, which consists of learning a near-optimal goal-conditioned policy for the (initially unknown) set of goal states that are reachable within a given number of steps in expectation from the starting state. We tackle it by designing an intrinsic goal selection scheme that leverages a measure of uncertainty of the agent's goal-reaching ability to adaptively target goals that are neither too difficult nor too easy. We also investigate, conceptually and empirically, how this idea can be operationalized in deep RL. ▷ Finally, in Chapter 9, we analyze an *incremental*[8] (or *compositional*) version of SYOG in *reward-free resettable* MDPs. Building on the formalism of Lim and Auer (2012), we refine the learning objective and introduce the first algorithm able to learn an incrementally near-optimal goal-conditioned policy. Throughout Part II, we will stress on the dependencies of our theoretical guarantees and compare them between the settings and assumptions, with a closing focus on avoiding a dependence on the total number of states in the learning guarantees.

[6] An MDP is said to be communicating if for any pair of states (s, s'), there exists a policy that can reach s' starting from s with probability 1.

[7] An MDP is said to be resettable if the action space contains a known action a_{reset} that deterministically resets the agent to the starting state s_0.

[8] At a high level, this means focusing on states that satisfy a (a priori unknown) recursive structure: a state is said to be incrementally reachable if it can be attained by a policy going through states that are themselves incrementally reachable. A policy is then said to be incrementally near-optimal if it is near-optimal among the class of policies that go through states that are incrementally reachable (and execute the action a_{reset} in the other states).

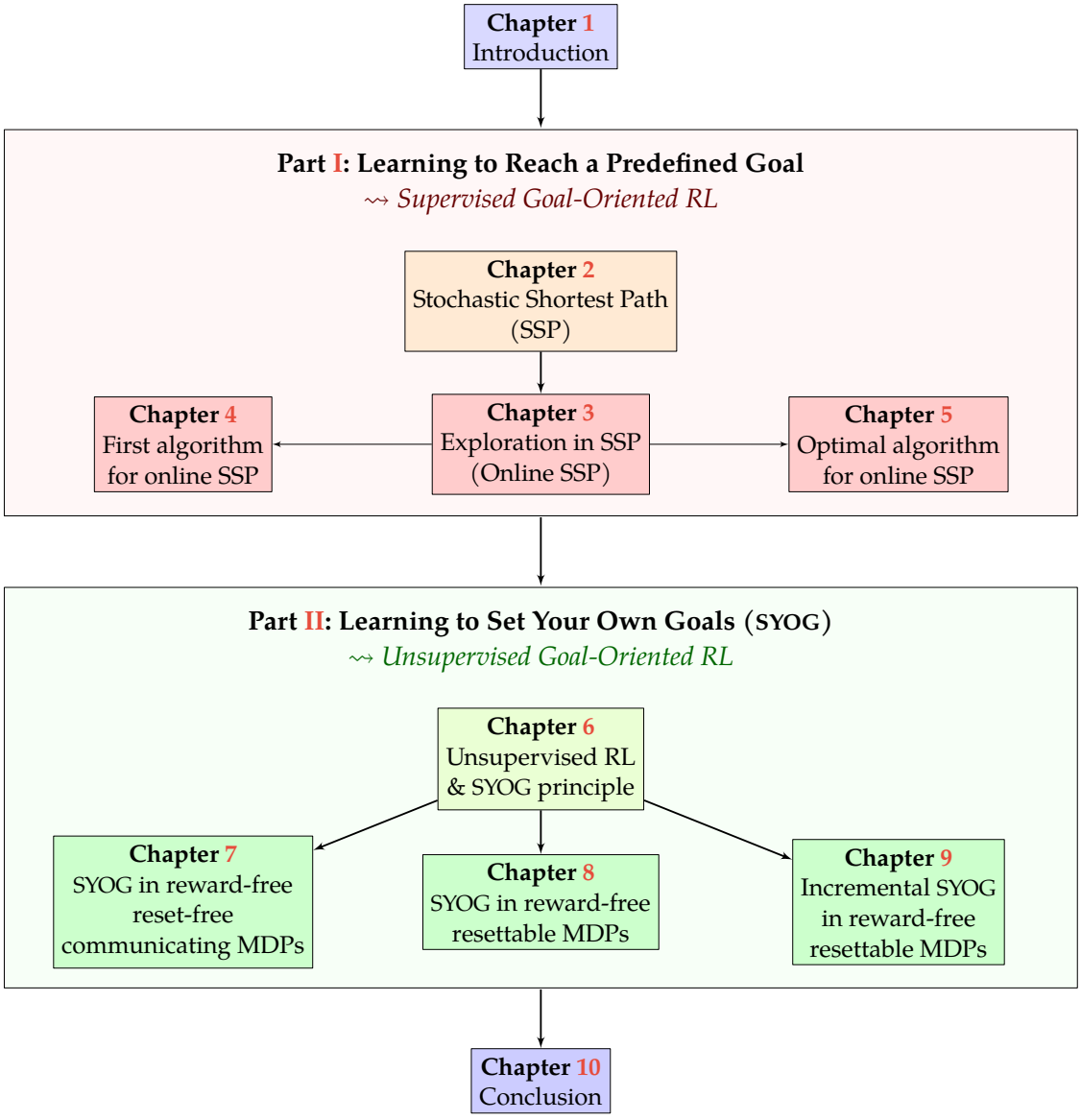


Figure 1.2 – This thesis is structured around the way goal states are generated. We start with the *supervised* scenario of Part I where a goal state to be reached in minimum total expected cost is provided as part of the problem definition. Leveraging its technical findings, we then move to the *unsupervised* scenario of Part II that focuses on learning to autonomously solve a variety of tasks in the absence of any reward supervision, by intrinsically generating and reaching a sequence of goals.

List of publications in international conferences with proceedings

Publications presented in this thesis

- Jean Tarbouriech, Omar Darwiche Domingues, Pierre Ménard, Matteo Pirotta, Michal Valko, Alessandro Lazaric. **Adaptive Multi-Goal Exploration**. In *International Conference on Artificial Intelligence and Statistics* (AISTATS), 2022 (presented in Chapter 8)
- Jean Tarbouriech, Matteo Pirotta, Michal Valko, Alessandro Lazaric. **A Provably Efficient Sample Collection Strategy for Reinforcement Learning**. In *Neural Information Processing Systems* (NeurIPS), 2021 (presented in Chapter 7)
- Jean Tarbouriech*, Runlong Zhou*, Simon S. Du, Matteo Pirotta, Michal Valko, Alessandro Lazaric. **Stochastic Shortest Path: Minimax, Parameter-Free and Towards Horizon-Free Regret**. In *Neural Information Processing Systems* (NeurIPS), 2021 (presented in Chapters 3 and 5)
- Jean Tarbouriech, Matteo Pirotta, Michal Valko, Alessandro Lazaric. **Improved Sample Complexity for Incremental Autonomous Exploration in MDPs**. In *Neural Information Processing Systems* (NeurIPS), 2020 (presented in Chapter 9)
- Jean Tarbouriech, Evrard Garcelon, Michal Valko, Matteo Pirotta, Alessandro Lazaric. **No-Regret Exploration in Goal-Oriented Reinforcement Learning**. In *International Conference on Machine Learning* (ICML), 2020 (presented in Chapters 3 and 4)

Publications discussed in this thesis

- Pierre-Alexandre Kamienny*, Jean Tarbouriech*, Sylvain Lamprier, Alessandro Lazaric, Ludovic Denoyer. **Direct then Diffuse: Incremental Unsupervised Skill Discovery for State Covering and Goal Reaching**. In *International Conference on Learning Representations* (ICLR), 2022 (discussed in Chapter 10)
- Jean Tarbouriech, Matteo Pirotta, Michal Valko, Alessandro Lazaric. **Sample Complexity Bounds for Stochastic Shortest Path with a Generative Model**. In *Algorithmic Learning Theory* (ALT), 2021 (discussed in Chapter 3)
- Jean Tarbouriech, Shubhanshu Shekhar, Matteo Pirotta, Mohammad Ghavamzadeh, Alessandro Lazaric. **Active Model Estimation in Markov Decision Processes**. In *Uncertainty in Artificial Intelligence* (UAI), 2020 (discussed in Chapter 6)
- Jean Tarbouriech, Alessandro Lazaric. **Active Exploration in Markov Decision Processes**. In *International Conference on Artificial Intelligence and Statistics* (AISTATS), 2019 (discussed in Chapter 6)

*denotes equal contribution.

Collaborations not mentioned in this thesis

- Evrard Garcelon, Baptiste Rozière, Laurent Meunier, Jean Tarbouriech, Olivier Teytaud, Alessandro Lazaric, Matteo Pirotta. **Adversarial Attacks on Linear Contextual Bandits**. In *Neural Information Processing Systems* (NeurIPS), 2020

Part I

Online Stochastic Shortest Path: Learning to Reach a Predefined Goal

Overview of Part I:

- **Open research question:** Given an extrinsically predefined goal state in an unknown environment, learn how to effectively reach it while minimizing the total expected costs.
- **Key contribution:** We formalize the setting of exploration in the Stochastic Shortest Path problem (a.k.a. online SSP) and derive two no-regret algorithms.
- **Relevance:** The SSP setting is general (encompassing both finite-horizon and discounted MDPs) and it models numerous RL tasks (e.g., navigation, game playing).

Chapter 2

Stochastic Shortest Path (SSP)

In this chapter, we provide a technical overview of the Stochastic Shortest Path (SSP) problem, where the objective is to minimize the expected cumulative cost to reach a specific goal state. We review some results available in the literature on *planning* in SSP, i.e., on the existence and computation of the optimal policy when all parameters of the SSP model are known. We also show how the two commonly studied settings of finite-horizon MDPs and infinite-horizon discounted MDPs can be cast as special cases of SSP. Lastly, we derive a *simulation lemma* for SSP to illustrate some SSP-specific technical challenges. This chapter lays the technical foundations for the online formulation of SSP that we will introduce and analyze in the remainder of Part I.

Contents

2.1 The SSP model

The stochastic shortest path (SSP) problem was first introduced by Eaton and Zadeh (1962) in the context of pursuit-evasion games and it was then studied thoroughly for the first time by Bertsekas and Tsitsiklis (1991), whose work lays the technical foundations for Chapter 2. As argued by Guillot and Stauffer (2020), "SSP arises naturally in robot motion planning, from maneuvering a vehicle over unfamiliar terrain, steering a flexible needle through human tissue or guiding a swimming micro-robot through turbulent water for instance (Alterovitz et al., 2007). It has also many applications in operations research, artificial intelligence and economics: from inventory control, reinforcement learning to asset pricing (see e.g., White, 1993; Merton, 1973; Bäuerle and Rieder, 2011; Sutton, Barto, et al., 1998)."

Definition 2.1 (SSP-MDP). *An SSP instance is an MDP* $M \triangleq \langle \mathcal{S}, \mathcal{A}, P, c, s_0, g \rangle$*, where* $\mathcal{S}$ *is the finite state space with cardinality* S*,* $\mathcal{A}$ *is the finite action space with cardinality* A*, and* $s_0 \in \mathcal{S}$ *is the initial state. We denote by* $g \notin \mathcal{S}$ *the goal state, and we set* $\mathcal{S}' \triangleq \mathcal{S} \cup \{g\}$ *(and* $S' \triangleq S+1$*). Taking action* a *in state* s *incurs a (instantaneous) cost drawn i.i.d. from a distribution on* $[0,1]$ *with expectation* $c(s,a)$*, and the next state* $s' \in \mathcal{S}'$ *is selected with probability* $P(s'|s,a)$ *(where* $\sum_{s' \in \mathcal{S}'} P(s'|s,a) = 1$*). The goal state* g *is absorbing and zero-cost, i.e.,* $P(g|g,a) = 1$ *and* $c(g,a) = 0$ *for any action* a*, which effectively implies that the agent ends its interaction with* M *once it reaches the goal* g*. Finally, let* $c_{\min} \triangleq \min_{s \in \mathcal{S}, a \in \mathcal{A}} c(s,a) \in [0,1]$ *denote the minimum cost over* $\mathcal{S} \times \mathcal{A}$*.*

Notation. A stationary and deterministic policy $\pi : \mathcal{S} \to \mathcal{A}$ is a mapping from state s to action $\pi(s)$, and we denote by Π the set of such policies. For notational convenience, let $P_{s,a} \triangleq P(\cdot|s,a)$, $P_{s,a,s'} \triangleq P(s'|s,a)$. For any two vectors X, Y of size S', we write their inner product as $XY \triangleq \sum_{s \in \mathcal{S}'} X(s)Y(s)$, we denote $\|X\|_\infty \triangleq \max_{s \in \mathcal{S}'} |X(s)|$, $\|X\|_\infty^{\neq g} \triangleq \max_{s \in \mathcal{S}} |X(s)|$, and if p is a probability distribution on $\mathcal{S}'$, then we define the variance of X w.r.t. p as $\mathbb{V}(p, X) \triangleq \sum_{s \in \mathcal{S}'} p(s)X(s)^2 - (\sum_{s \in \mathcal{S}'} p(s)X(s))^2$.

Objective. The SSP objective is to minimize its expected cumulative cost incurred until the goal is reached. The following definitions ground the performance of a policy in SSP (note that they are equivalent to the definitions given in Chapter 1 with the $\omega(t)$ notation since the goal state is absorbing and zero-cost by Definition 2.1).

Definition 2.2 (Expected cost-to-goal). *The (possibly infinite) expected cost-to-goal — which we call the value function — of a policy* $\pi \in \Pi$ *and its (possibly infinite) associated Q-function are*

defined for any $(s, a) \in \mathcal{S} \times \mathcal{A}$ as

$$V^\pi(s) \triangleq \lim_{T\to\infty} \mathbb{E}\Big[\sum_{t=1}^{T} c_t(s_t, \pi(s_t)) \,\big|\, s_1 = s\Big],$$
$$Q^\pi(s, a) \triangleq \lim_{T\to\infty} \mathbb{E}\Big[\sum_{t=1}^{T} c_t(s_t, \pi(s_t)) \,\big|\, s_1 = s, \pi(s_1) = a\Big],$$

where $c_t \in [0, 1]$ is the (instantaneous) cost incurred at time t at state-action pair $(s_t, \pi(s_t))$, and the expectation is w.r.t. the random sequence of states generated by executing π starting from state $s \in \mathcal{S}$ (and taking action $a \in \mathcal{A}$ in the second case). By definition of the goal, we set $V^\pi(g) \triangleq 0$ and $Q^\pi(g, a) \triangleq 0$ for all policies $\pi \in \Pi$ and actions $a \in \mathcal{A}$.

Definition 2.3 (Expected time-to-goal)**.** *For any policy $\pi \in \Pi$ and state $s \in \mathcal{S}$, let $\tau_\pi(s)$ be the (possibly infinite) hitting time from s to g when executing π, i.e.,*

$$\tau_\pi(s) \triangleq \inf\Big\{t \geq 0 : s_{t+1} = g \,|\, s_1 = s, \pi\Big\}.$$

The (possibly infinite) expected time-to-goal of a policy $\pi \in \Pi$ is then defined for any $s \in \mathcal{S}$ as

$$T^\pi(s) \triangleq \mathbb{E}\Big[\tau_\pi(s)\Big] = \lim_{T\to\infty} \mathbb{E}\Big[\sum_{t=1}^{T} \mathbb{1}[s_t \neq g] \,\big|\, s_1 = s\Big].$$

We now define the standard Bellman operators for the SSP model.

Definition 2.4 (Bellman operators)**.** *For any $V \in \mathbb{R}^S$ and $s \in \mathcal{S}$, we define the policy evaluation Bellman operator $\mathcal{L}^\pi$ for any policy $\pi \in \Pi$ as well as the optimal Bellman operator $\mathcal{L}$ as follows*

$$\mathcal{L}^\pi V(s) \triangleq c(s, \pi(s)) + P_{s,\pi(s)} V, \tag{2.1}$$
$$\mathcal{L} V(s) \triangleq \min_{a\in\mathcal{A}} \Big\{c(s, a) + P_{s,a} V\Big\}. \tag{2.2}$$

Remark 2.5. The conventional fixed point equations $V^\star = \mathcal{L}V^\star$ and $V^\pi = \mathcal{L}^\pi V^\pi$ are referred to as Bellman's equations for the optimal value function and for the value function of π, respectively. They are generally expected to hold in MDP models (it is indeed the case in discounted MDPs or finite-horizon MDPs), yet *this may not be the case in SSP-MDPs*. Before going further in understanding why, we need to review an important SSP-specific characterization of *proper* policies.

2.2 Proper Policies

Definition 2.6 (Proper policies)**.** *A policy π is said to be proper if it reaches the goal g with probability 1 when starting from any state in $\mathcal{S}$. Otherwise, it is said to be improper. We denote by $\Pi_{\text{proper}} \subseteq \Pi$ the set of proper, stationary and deterministic policies.*

Throughout Part I, we make the following basic assumption which ensures that the SSP problem is well-posed.

Assumption 2.7. *For the MDP M there exists at least one proper policy, i.e., $\Pi_{\text{proper}} \neq \emptyset$.*

Properties. We have the following important relations between the quantities of interest in Definitions 2.2 and 2.3, which depend on whether the policy is proper and on the value of the minimum cost $c_{\min}$.

- A proper policy has a finite expected time-to-goal and a finite expected cost-to-goal, i.e.,

$$\forall \pi \in \Pi_{\text{proper}},\ \forall s \in \mathcal{S}, \quad T^\pi(s) < +\infty, \quad V^\pi(s) < +\infty.$$

  This stems from the fact that the Markov chain induced by any proper policy on the MDP M is absorbing with recurrent state g and a finite number of S transient states, hence from standard Markov chain theory (see e.g., Norris, 1998), the expected time until absorption has a finite expectation. The value function is then finite by boundedness of the instantaneous costs.

- An improper policy has an expected time-to-goal with at least one unbounded component, which may also be the case for its value function (e.g., in the special case when all non-goal costs are positive), i.e.,

$$\begin{aligned} &\forall \pi \notin \Pi_{\text{proper}},\ \exists s \in \mathcal{S}, \quad T^\pi(s) = +\infty. \\ &c_{\min} > 0 \quad \Longrightarrow \quad \forall \pi \notin \Pi_{\text{proper}},\ \exists s \in \mathcal{S}, \quad V^\pi(s) = +\infty. \end{aligned}$$

- The expected time-to-goal and cost-to-goal of a policy can be related as follows

$$\forall \pi \in \Pi_{\text{proper}},\ \forall s \in \mathcal{S}, \quad V^\pi(s) \leq T^\pi(s) \leq \frac{V^\pi(s)}{c_{\min}}, \tag{2.3}$$

  where the first inequality always holds since costs are in $[0, 1]$ and the second inequality only holds if $c_{\min} > 0$. This anticipates the key role of $c_{\min}$ in SSP analysis, as we will see in Sections 2.4 to 2.6 and the subsequent chapters.

- From Bertsekas (1995), for any proper policy $\pi \in \Pi_{\text{proper}}$, the operator $\mathcal{L}^\pi$ is a *contraction* w.r.t. some weighted sup-norm $\|\cdot\|_{\infty,\psi_\pi}$ on $\mathbb{R}^S$, defined for some vector $\psi_\pi \in \mathbb{R}^S, \psi_\pi > 0$ by $\|J\|_{\infty,\psi_\pi} \triangleq \max_{s\in\mathcal{S}} |J(s)|/\psi_\pi(s)$. On the other hand, for any improper policy π, $\mathcal{L}^\pi$ is *not* a contraction w.r.t. any norm. Finally, in the special case where all policies are proper, $\mathcal{L}$ is a weighted sup-norm contraction,[1] but in the general case $\mathcal{L}$ may not be a contraction w.r.t. any norm.

In the following definition we introduce the concept of *SSP-diameter* which measures the complexity of navigating to the goal starting from any state. We choose this name to relate to the conventional diameter in infinite-horizon undiscounted MDPs, which measures the complexity of navigating between any two pair of states (Jaksch et al., 2010).

Definition 2.8 (SSP-diameter)**.** *We define the SSP-diameter D as the shortest path between any starting state and the goal state, i.e.,*

$$D \triangleq \max_{s\in\mathcal{S}} \min_{\pi\in\Pi} T^\pi(s).$$

Note that Assumption 2.7 implies that $D < +\infty$.

A simple (yet unsatisfactory) modification of policies to make them all proper. The above discussion shows that a key separation of policies is between those that are proper and improper. While computing a proper policy (there exists at least one from Assumption 2.7) may require some effort, the following lemma shows that it is not difficult to find *some* proper policy, namely the uniform random policy. As an immediate corollary, *any* policy can be made proper by simply injecting enough randomization over the action space (e.g., via an ε-greedy strategy), albeit with possibly very large expected time-to-goal and thus poor goal-reaching behavior.

Lemma 2.9. *For any SSP problem satisfying Assumption 2.7, the uniform random policy is proper.*

Proof. By Assumption 2.7, there exists some proper policy $\mu \in \Pi_{\text{proper}}$. Its expected time-to-goal is upper bounded component-wise, that is, there exists some $m \in \mathbb{N}^\star$ such that $T^\mu(s) \leq m$ for every $s \in \mathcal{S}$. By Markov's inequality, $\mathbb{P}(\tau_\mu(s) > 2m) \leq T^\mu(s)/(2m) \leq 1/2$. Denote by π_{u} the uniform random policy, which selects an action at random based on a uniform

[1] Specifically, when all policies are proper, letting $\psi(s) \triangleq \sup_\pi T^\pi(s)$, it holds that $\mathcal{L}^\pi$ (for any policy π) and $\mathcal{L}$ are contractions w.r.t. the weighted sup-norm $\|\cdot\|_{\infty,\psi}$ with modulus $\alpha \triangleq \max_{s\in\mathcal{S}}(\psi(s)-1)/\psi(s) < 1$, i.e.,

$$\|\mathcal{L}^\pi J - \mathcal{L}^\pi J'\|_{\infty,\psi} \leq \alpha\|J - J'\|_{\infty,\psi} \quad , \qquad \|\mathcal{L}J - \mathcal{L}J'\|_{\infty,\psi} \leq \alpha\|J - J'\|_{\infty,\psi}.$$

probability distribution over the action space. Then π_{u} executes the same action as policy μ for $2m$ consecutive stages with probability $(1/A)^{2m} > 0$. It follows that the probability of reaching the goal state within $2m$ stages by following the uniform random policy is greater than or equal to $(1/A)^{2m}/2$, starting from any state. As a result, we can write

$$T^{\pi_{\text{u}}}(s) = \sum_{n=0}^{+\infty} \mathbb{P}(\tau_{\pi_{\text{u}}}(s) > n) \leq 2m \sum_{j=0}^{+\infty} \mathbb{P}(\tau_{\pi_{\text{u}}}(s) > 2mj) \leq 2m \sum_{j=0}^{+\infty} \left(1 - (1/A)^{2m}/2\right)^j \leq 4mA^{2m},$$

which concludes that the uniform random policy is proper. Nonetheless, note that its goal-reaching behavior can be quite poor, as the bound on its expected time-to-goal scales exponentially in m (as a power of the number of actions A). □

2.3 Two Special Cases of SSP: Finite-Horizon and Discounted MDPs

The general SSP problem features two possibly conflicting objectives — reaching the goal vs. minimizing cost. If we make additional assumptions and only focus on one of these two objectives, we can relate SSP to simpler and more commonly studied settings.

First, we can assume that there exists a *known* upper bound H on the hitting time of *any* policy $\pi \in \Pi$ starting from any state $s \in \mathcal{S}$, i.e., $\pi_\pi(s) \leq H$ almost surely. This is often called the *loop-free* SSP version, which is quite restrictive and fails to hold in many realistic environments (as explained at the end of this section). In this case, the SSP problem is equivalent to the popular *finite-horizon* RL problem, where the objective is to minimize the expected costs accumulated over H steps.

Second, we can assume that all costs are uniform and that all transitions are deterministic (i.e., $P : \mathcal{S} \times \mathcal{A} \to \mathcal{S}$). This scenario is for example captured by a simple deterministic navigation problem, where the state of the agent is its current location, the four actions are deterministically moving 1 step along each of east, west, north or south, and the agent has a goal state g that it is trying to reach as quickly as possible. Then the SSP problem is equivalent to a *discounted* RL problem for any choice of discount factor $\gamma \in (0, 1)$, where the objective is to minimize the expected cumulative discounted costs. Intuitively, the discount factor provides incentive to reach the goal state earlier in the trajectory, hence the optimal behavior in the discounted setting corresponds to finding the shortest path from the initial state to the goal state. This is because the discounted value function of a state given a deterministic policy is $\phi(d, \gamma) \triangleq \sum_{t=0}^{d-1} \gamma^t = (1 - \gamma^d)/(1 - \gamma)$, where d denotes the number of steps required by the policy to reach the goal state, and $\phi(d, \gamma)$ is a monotonically increasing function in d for any $\gamma \in (0, 1)$.

We have thus demonstrated that there exist special cases of SSP that recover both the finite-horizon objective and the discounted objective. Conversely, as alluded to in Chapter 1, any finite-horizon or discounted problem can be cast as an SSP problem. We detail below this claim:

- Any finite-horizon MDP with horizon $H \in \mathbb{N}^\star$ can be interpreted as having a goal state g that is guaranteed to be reached at step $H+1$. Formally, it can be embedded into an SSP problem by extending the state space to $\mathcal{S} \times [H+1]$ and choosing $P((g, H+1)|(s, H), a) = 1$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$.
- Any infinite-horizon discounted MDP with discount factor $\gamma < 1$ can be reduced to an equivalent SSP problem in which, for every state-action pair, there is a probability $(1-\gamma)$ of making a transition to the goal state, with the other transition probabilities normalized.

Observe that for the SSP problems created by these two reductions, all policies are proper, i.e., $\Pi_{\text{proper}} = \Pi$. In fact, the special case of SSP where all policies are proper is easier to study mathematically (Bertsekas, 1995), although this assumption is often unrealistic. Indeed, in many problems it is possible for the agent to "loop" back to states, such as in navigation where cardinal actions can cancel each other's effects.

2.4 On the Optimal Solution in SSP

Equipped with Assumption 2.7 and an additional condition defined below (that all improper policies incur high cost), one can derive the following important properties on proper policies.

Proposition 2.10 (Bertsekas and Tsitsiklis, 1991, Lemma 1)**.** *Suppose that Assumption 2.7 holds and that the following additional condition holds*

(♣) *For every improper policy π' there exists at least one state $s \in \mathcal{S}$ such that $V^{\pi'}(s) = +\infty$.*

Let π be any policy, then

- *If there exists a vector $U : \mathcal{S} \to \mathbb{R}$ such that $U(s) \geq \mathcal{L}^\pi U(s)$ for all $s \in \mathcal{S}$, then π is proper, and $V^\pi(s) \leq U(s)$ for all $s \in \mathcal{S}$.*
- *If π is proper, then its value function V^π is the unique solution to the Bellman equations $V^\pi(s) = \mathcal{L}^\pi V^\pi(s)$ for all $s \in \mathcal{S}$.*

The first property of Proposition 2.10 follows from the *monotonicity* of the operator $\mathcal{L}^\pi$ (Bertsekas, 1995) and the fact that for any arbitrary vector U, $\lim_{i \to +\infty} (\mathcal{L}^\pi)^i U = V^\pi$. It will turn out to be a useful technical tool to prove subsequent results (e.g., Lemmas 2.13 and 2.14). The second property, which shows that a policy is proper if and only if its value function satisfies

the Bellman equations, paves the way for the following important results on the optimal policy in SSP.

Proposition 2.11 (Bertsekas and Tsitsiklis, 1991; Yu and Bertsekas, 2013)**.** *Under the conditions of Proposition 2.10, the policy that minimizes the value function component-wise is stationary, deterministic, and proper; let us denote it by $\pi^\star$. Moreover, $V^\star = V^{\pi^\star}$ is the unique solution of the optimality equations $V^\star = \mathcal{L}V^\star$ and $V^\star(s) < +\infty$ for any $s \in \mathcal{S}$. Finally, the optimal Q-value, denoted by $Q^\star = Q^{\pi^\star}$, is related to the optimal value function, for all $(s,a) \in \mathcal{S} \times \mathcal{A}$, as follows*

$$Q^\star(s,a) = c(s,a) + P_{s,a}V^\star,$$
$$V^\star(s) = \min_{a \in \mathcal{A}} Q^\star(s,a).$$

Under the two conditions of Propositions 2.10 and 2.11, the policy that minimizes the value function is necessarily proper, hence the two objectives (minimizing costs and reaching the goal) coincide. If we relax condition (♣), this may no longer be the case. Indeed, consider the simple scenario where there exists an action $a^\dagger \in \mathcal{A}$ such that $P(s_0|s_0,a^\dagger) = 1$ and $c(s_0,a^\dagger) = 0$. Then, a strategy to incur minimal (namely, zero) total cost would be simply to execute action $a^\dagger$ forever at the initial state s_0, however the goal would never be reached. Since reaching the goal is one of the agent's main objectives, we thus expect the agent to target the *optimal proper policy*, i.e.,

$$\pi^\star \in \arg\min_{\pi \in \Pi_{\text{proper}}} V^\pi.$$

Now, we can handle the second requirement (♣) of Propositions 2.10 and 2.11 as follows (Bertsekas and Yu, 2013). First, the requirement is in particular verified if all instantaneous costs are strictly positive. To deal with the case of non-negative costs, we can introduce a small perturbation $\eta \in (0,1]$ to all costs to yield a new (strictly positive) cost function $c_\eta(s,a) \triangleq \max\{c(s,a),\eta\}$. In this cost-perturbed MDP, the conditions of Propositions 2.10 and 2.11 hold so we get an optimal policy $\pi^\star_\eta$ that is stationary, deterministic and proper and has a finite value function $V^\star_\eta$. Taking the limit as $\eta \to 0$, we have that $\pi^\star_\eta \to \pi^\star$ and $V^\star_\eta \to V^{\pi^\star}$, where $\pi^\star$ is the optimal proper policy in the original model that is also stationary and deterministic, and $V^{\pi^\star}$ denotes its value function (Bertsekas and Yu, 2013). This enables to circumvent the second condition of Propositions 2.10 and 2.11 and only require Assumption 2.7 to hold.

Algorithm 2.1: Value Iteration for SSP (VI-SSP) with precision level η

Input: Goal g, states $\mathcal{S}$, actions $\mathcal{A}$, transitions P, costs c and VI precision level $\eta > 0$.
Set $u_0 \triangleq \mathbf{0}_S$ and $n \triangleq 0$.
Compute $u_1 \triangleq \mathcal{L}u_0$.
while $\|u_{n+1} - u_n\|_\infty > \eta$ **do**
    $u_{n+1} \triangleq \mathcal{L}u_n$.
Set $U \triangleq u_n$ and $\pi(s) \in \arg\min_{a \in \mathcal{A}} \left\{ c(s,a) + P_{s,a}U \right\}$ for any $s \in \mathcal{S}$.
Output: Value vector U and greedy policy π.

2.5 Planning in SSP, with a Focus on Value Iteration

Early work by Bertsekas and Tsitsiklis (1991), followed by a rich line of research (e.g., Bertsekas, 1995; Bonet, 2007; Kolobov et al., 2011; Hansen, 2011; Bertsekas and Yu, 2013; Guillot and Stauffer, 2020), examine the planning problem in SSP, i.e., how to compute an optimal policy when all parameters of the SSP model are known (i.e., transitions P and costs c). Under Assumption 2.7 and the additional condition (♣), the optimal policy is proper, deterministic and stationary and can be computed efficiently using standard planning techniques, e.g., value iteration, policy iteration or linear programming.

Value iteration (VI). Throughout the thesis, our algorithms will compute policies using VI-based planning procedures, hence we focus our discussion on it. Value Iteration (VI) (Bellman, 1966) is a dynamic programming approach that starts with an initial estimate of the states' values, V_0, and iteratively applies the optimal Bellman operator $\mathcal{L}$ to them, i.e., $V_{i+1} \leftarrow \mathcal{L}V_i = \mathcal{L}^{i+1}V_0$. The optimal value function $V^\star$ is the unique fixed point of $\mathcal{L}$ and repeated application of this operator provably forces VI to converge to $V^\star$ irrespectively of the initializing value function V_0, i.e., $\lim_{i\to\infty} V_i = V^\star$ component-wise.

For an RL algorithm to be computationally efficient, we cannot expect it to have a planning procedure that iterates infinitely. As a result, we now study the impact of a *finite* number of iterations on the convergence speed and the suboptimality gap of the VI procedure, which will turn out to be useful throughout the thesis. In what follows, we consider a *positive cost function* lower bounded by $c_{\min} > 0$.

Formally, the procedure VI-SSP (Algorithm 2.1) considers the following inputs: a goal $g \notin \mathcal{S}$, states $\mathcal{S}$, actions $\mathcal{A}$, transitions probabilities P, a cost function c with (non-goal) costs lower bounded by $c_{\min} > 0$, and a VI precision level $\eta > 0$. It outputs an S-sized vector U and a policy π that is greedy w.r.t. the vector U.

Convergence speed. To bound the number of iterations required to reach the termination condition (line 4 in Algorithm 2.1), we can use the following useful result of Bonet (2007).

Proposition 2.12 (Bonet, 2007, Corollary 4.1)**.** *Assume that the SSP instance satisfies Assumption 2.7 and $c_{\min} > 0$. Moreover, assume that the model does not admit self-loops (apart from at the goal), i.e., $P(s, a, s) = 0$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$ (this assumption is a non-restrictive technicality since a model can be "converted" into an equivalent model that satisfies the no-self-loop assumption, see Bertsekas, 1995, page 89, Bonet, 2007, Section 3). Let V be an admissible initial value vector, i.e., $0 \leq V \leq V^\star$. Then applying the value iteration algorithm for SSP achieves a residual of $||V^\star - \mathcal{L}^n V||_\infty \leq \varepsilon$ in a number of iterations n bounded as*

$$n = O\left(\frac{||V^\star||_\infty^2 S^2}{c_{\min}^2} + \left(\log ||V^\star||_\infty + |\log \varepsilon|\right)\frac{||V^\star||_\infty S}{c_{\min}}\right).$$

Combining Proposition 2.12 with the triangle inequality directly ensures that VI-SSP (Algorithm 2.1) is computationally efficient in the sense that it terminates in a number of iterations that is polynomially bounded by $||V^\star||_\infty$, $c_{\min}^{-1}$, S and $|\log \eta|$.

Suboptimality gap. We now seek to bound the suboptimality gap $||V^\star - V^\pi||_\infty$. Note that U is *not* the value function of π, however both quantities can be related according to the following lemma, whose proof is deferred to Section A.1. Note that similar suboptimality bounds have also been derived by Bertsekas (1995) (in the special case of only proper policies) and Hansen (2011).

Lemma 2.13. *Let $(U, \pi) \triangleq$ VI-SSP$(g, \mathcal{S}, \mathcal{A}, P, c, \eta)$ be the solution computed by* VI-SSP *(Algorithm 2.1). The following component-wise inequalities hold*

- $U \leq V^\star \leq V^\pi$.
- *If the* VI *precision level verifies $\eta \leq \frac{c_{\min}}{2}$, then $V^\pi \leq \left(1 + \frac{2\eta}{c_{\min}}\right) U$.*

Combining the two inequalities above gives that if the VI *precision level verifies $\eta \leq \frac{c_{\min}}{2}$, then*

$$||V^\pi - V^\star||_\infty \leq \frac{2\eta ||V^\star||_\infty}{c_{\min}}.$$

2.6 A Simulation Lemma for SSP

One of the most basic and fundamental results in RL is called the simulation lemma, which quantifies how much error we incur in evaluating policies if we build an approximate MDP $\overline{M}$ for the true MDP M. Akin to the existing simulation lemmas in finite-horizon or discounted MDPs (see e.g., Kearns and Singh, 2002), we now derive a simulation lemma tailored to SSP (see proof in Section A.2), with the purpose of showcasing some SSP-specific technical challenges.

Lemma 2.14 (Simulation Lemma for SSP)**.** *Consider any accuracy level $\eta > 0$ and any two models P and $\overline{P} \in \mathcal{P}_\eta^{(P)}$, where $\mathcal{P}_\eta^{(P)}$ represents the set of models "close" to P and is formally defined as follows*

$$\mathcal{P}_\eta^{(P)} \triangleq \left\{ P' \in \mathbb{R}^{S' \times A \times S'} : \forall (s,a) \in \mathcal{S}' \times \mathcal{A},\ P'(\cdot|s,a) \in \Delta(\mathcal{S}'),\ \|P(\cdot|s,a) - P'(\cdot|s,a)\|_1 \leq \eta \right\}.$$

Assume that for each model P and $\overline{P}$, there exists at least one proper policy w.r.t. the goal state g. Consider a known cost function in $[0,1]$ with minimum non-goal cost $c_{\min} > 0$. Consider any policy π that is proper in $\overline{P}$, with value function denoted by $\overline{V}^\pi$, such that the following condition is verified

$$\eta \|\overline{V}^\pi\|_\infty \leq 2c_{\min}. \tag{2.4}$$

Then π is proper in P (i.e., its value function verifies $V^\pi < +\infty$ component-wise), and we have

$$\forall s \in \mathcal{S},\ V^\pi(s) \leq \left(1 + \frac{2\eta \|\overline{V}^\pi\|_\infty}{c_{\min}}\right) \overline{V}^\pi(s),$$

and conversely,

$$\forall s \in \mathcal{S},\ \overline{V}^\pi(s) \leq \left(1 + \frac{\eta \|\overline{V}^\pi\|_\infty}{c_{\min}}\right) V^\pi(s).$$

Combining the two inequalities above yields

$$\|V^\pi - \overline{V}^\pi\|_\infty \leq \frac{7\eta \|\overline{V}^\pi\|_\infty^2}{c_{\min}}.$$

For comparison let us now recall the classical simulation lemma for discounted MDPs.

Proposition 2.15 (Simulation Lemma for discounted MDPs, see e.g., Kearns and Singh, 2002)**.** *Consider any two models P and $\overline{P} \in \mathcal{P}_\eta^{(p)}$ for any $\eta > 0$. Consider as value function in P the expected discounted cumulative reward, i.e., for any policy π and state $s \in \mathcal{S}$, $V^\pi(s) \triangleq \mathbb{E}\left[\sum_{t=1}^{+\infty} \gamma^t r(s_t, \pi(s_t)) \mid s_1 = s\right]$; and $\overline{V}^\pi$ is the value function in $\overline{P}$. Suppose that the instantaneous rewards are known and bounded in $[0, 1]$. Then for any policy π, we have*

$$\|V^\pi - \overline{V}^\pi\|_\infty \leq \frac{\gamma\eta}{2(1-\gamma)^2}.$$

We spell out the key differences between the simulation lemma in the discounted setting (Proposition 2.15) and in SSP (Lemma 2.14), bringing to light the criticalities in the latter setting. Due to the lack of contraction property for the Bellman operators in SSP, we need to take a different path than the discounted analysis; specifically, we cast the error between the models as a translation of the instantaneous costs, which enables us to find a suitable vector to apply the first property of Proposition 2.10. We also observe that the guarantee of Lemma 2.14 requires condition (2.4), which involves both the accuracy η and the value function of π in $\overline{P} \in \mathcal{P}_\eta^{(P)}$. This is due to the fact that in SSP the performance (i.e., value function) of a policy may be arbitrarily bad (even infinite), whereas in discounted MDPs it is always upper bounded by the intrinsic horizon $1/(1-\gamma)$. In Lemma 2.14, such "trajectory length" is captured by the ratio between the infinity norm of the value function of the policy and the minimum cost $c_{\min}$, which upper bounds the expected time-to-goal of the policy starting from any state (cf. Equation (2.3)). This anticipates the key role of the minimum cost $c_{\min}$ in the analysis of the online formulation of the SSP problem, as we will see in the subsequent chapters.

2.7 Extensions

Heuristic search methods. Since VI stores values for the entire state space, it may run out of memory as the size of the SSP model increases. A line of research has focused on deriving methods with smaller memory consumption and faster run-time than VI. Many of them, e.g., LRTDP (Bonet and Geffner, 2003b), LAO* (Hansen and Zilberstein, 2001) or FRET (Kolobov et al., 2011), fall under the heuristic search paradigm, conceptually described by the Find-and-Revise (F&R) framework (Bonet and Geffner, 2003a). F&R algorithms use the knowledge of the initial state and an *admissible heuristic* (i.e., an initial estimate for the value function that does not underestimate the values of any states under $V^\star$) to compute the optimal policy for an SSP problem while avoiding visits to many of the states that are not part of that policy. It could be an interesting direction of future investigation to apply such techniques in the learning

formulation of SSP that we introduce in Chapter 3 so as to design learning algorithms that are more tractable than VI-based ones.

SSP with dead-end states. While Assumption 2.7 is natural, it does not encompass various scenarios that contain at least one *dead-end* state from which reaching the goal g is impossible. To this end, various alternative objectives have been analyzed with VI-based and heuristic search algorithms, such as assigning a finite and fixed penalty for not reaching the goal (Finite-Penalty, Kolobov et al., 2012), maximizing the probability of getting to the goal while ignoring the expected cost (Max-Prob, Kolobov et al., 2011), or minimizing the expected cost among the policies with maximum goal-reaching probability (Min-Cost given Max-Prob, Trevizan et al., 2017). Throughout Part I, we mostly consider that Assumption 2.7 holds, except in Section 4.5 where we discuss a way to relax this assumption in the learning formulation of SSP (using the Finite-Penalty approach). In Part II, we will also study goal-driven objectives in environments that may not satisfy Assumption 2.7.

Chapter 3

Online Stochastic Shortest Path

In this chapter, we formalize for the first time the online learning problem in the SSP setting (a.k.a. online SSP), where both the transition dynamics and the cost function are initially unknown. We structure the agent's interaction in episodes of indefinite length, that terminate (and reset the agent) *if and only if* the goal is reached. We propose an adequate notion of regret to quantify the performance of the learning agent (i.e., how well the behavior of the optimal policy is approximated). We also identify desirable properties for a learning algorithm in online SSP (i.e., minimax-optimal, parameter-free, horizon-free). This chapter lays the foundations for the two no-regret algorithms that we will introduce in Chapters 4 and 5. [1]

Contents

[1] This chapter is based on material from three articles published in the proceedings of the 37^{th} International Conference on Machine Learning (ICML 2020), the 34^{th} Conference on Neural Information Processing Systems (NeurIPS 2021) and the 32^{nd} International Conference on Algorithmic Learning Theory (ALT 2021) (Tarbouriech et al., 2020a; Tarbouriech et al., 2021c; Tarbouriech et al., 2021b).

3.1 Formalizing Exploration in SSP: Minimizing Regret

We formalize the online learning problem in SSP where the agent does not have any prior knowledge of the cost function c or transition function P. Each episode starts at the initial state $s_0 \in \mathcal{S}$ (the extension to any possibly unknown distribution of initial states is straightforward), and ends *only* when the goal state g is reached (note that this may never happen if the agent does not reach the goal). We introduce the following performance metric to capture the high-level objective of approximating as quickly as possible the optimal goal-reaching behavior.

Definition 3.1 (SSP-regret)**.** *We evaluate the performance of the agent after $K \geq 1$ episodes by its SSP-regret, which we define as*

$$R_K \triangleq \sum_{k=1}^{K} \sum_{h=1}^{I^k} c_h^k - K \cdot \min_{\pi \in \Pi_{\text{proper}}} V^\pi(s_0), \tag{3.1}$$

where I^k is the (random) time needed to complete episode k and c_h^k is the cost incurred in the h-th step of episode k when visiting (s_h^k, a_h^k). Also let $T_K \triangleq \sum_{k=1}^{K} I^k$ be the total time elapsed over the K episodes. If there exists k such that I^k is infinite, then we define $R_K = \infty$ and $T_K = \infty$.

A few remarks on the proposed definition of SSP-regret are in order. First, we notice that we consider as optimal comparator the quantity $\min_{\pi \in \Pi_{\text{proper}}} V^\pi$ instead of $\min_{\pi \in \Pi} V^\pi$ as would be done in other settings (e.g., finite-horizon, infinite-horizon). This stems from the fact that in SSP the policy that minimizes the value function may not be proper, yet we expect our agent to reach the goal at each episode in order to terminate it, therefore we compare to the best *proper* policy, as motivated in Chapter 2. On the one hand, the definition of SSP-regret resembles the infinite-horizon undiscounted regret, where the performance of the algorithm is evaluated by the costs accumulated by executing the possibly non-stationary policy executed at episode k denoted by μ_k. At the same time, it incorporates the episodic nature of finite-horizon problems, where the performance of the optimal policy is evaluated by its value function at the initial state. Nonetheless, notice that we cannot use the finite-horizon regret definition, i.e., $\sum_{k=1}^{K} V^{\mu_k}(s_0) - V^\star(s_0)$, where a policy μ_k is chosen at the beginning of the episode and run until its termination. Indeed, the fact that a random realization of μ_k reaches the goal (i.e., $I^k < \infty$) does *not* imply that μ_k is proper; in fact, we have a priori no guarantee that it is proper, yet the execution of a single non-proper policy μ_k would directly lead to an unbounded regret since $V^{\mu_k}(s_0) = +\infty$. Finally, a unique feature of SSP-regret is that it is infinite as soon as one episode is unable to terminate at the goal, which implies that proving that the SSP-regret is linear in K is already non-trivial.

Definition 3.2 (Quantities of interest of the optimal proper policy)**.** *We denote the optimal proper policy by $\pi^\star$, i.e., for all $s \in \mathcal{S}$,*

$$\pi^\star(s) \in \arg\min_{\pi \in \Pi_{\text{proper}}} V^\pi(s).$$

We also set

$$V^\star(s) \triangleq V^{\pi^\star}(s) = \min_{\pi \in \Pi_{\text{proper}}} V^\pi(s),$$
$$Q^\star(s,a) \triangleq Q^{\pi^\star}(s,a) = \min_{\pi \in \Pi_{\text{proper}}} Q^\pi(s,a).$$

Let $B_\star > 0$ bound the values of $V^\star$, i.e.,

$$B_\star \triangleq \max_{s \in \mathcal{S}} V^\star(s).$$

Note that $Q^\star(s,a) \leq 1 + B_\star$. Finally, let $T_\star > 0$ bound the expected time-to-goal of $\pi^\star$, i.e.,

$$T_\star \triangleq \max_{s \in \mathcal{S}} T^{\pi^\star}(s).$$

Since the costs lie in $[0,1]$, we can order the different quantities as

$$B_\star \leq D \leq T_\star < +\infty, \tag{3.2}$$

where their boundedness is guaranteed by Assumption 2.7 (recall the definition of the SSP-diameter D in Definition 2.8). Moreover, in the case of positive costs lower bounded by $c_{\min} > 0$, then $T_\star \leq B_\star / c_{\min}$.

Notation. Throughout Part I, we will report *high-probability* upper bounds on the SSP-regret (Definition 3.1). For any given threshold $\delta \in (0,1)$, we write that $R_K = \widetilde{O}\Big(f(K, S, A, B_\star, T_\star)\Big)$ if there exists polynomial functions $q \triangleq \text{poly}(S, A, B_\star, T_\star)$, $q' \triangleq \text{poly}(K, S, A, B_\star, T_\star, \delta^{-1})$ and absolute constants $\alpha, \beta > 0$ (i.e., independent of the MDP instance) such that with probability at least $1-\delta$, for any $K \geq q$, $R_K \leq \alpha \cdot f(K, S, A, B_\star, T_\star, \delta^{-1}) \cdot \log^\beta q'$; and we similarly write that $R_K = O\Big(g(K, S, A, B_\star, T_\star)\Big)$ if $R_K \leq \alpha \cdot g(K, S, A, B_\star, T_\star, \delta^{-1})$. Finally, we write that $f = \Omega(g)$ if there exists an absolute constant $c > 0$ such that $f \geq c \cdot g$.

3.2 A Special Case: Uniform-Cost Online SSP

In this section, to gain intuition on the unique challenges of online SSP, we focus on the special case of SSP problems with uniform costs. We show that in this case the online SSP problem can be cast as an infinite-horizon problem and that an algorithm such as UCRL2 (Jaksch et al., 2010) can be directly applied and achieve sublinear regret guarantees.

Assumption 3.3 (only in Section 3.2). *The costs $c(s,a)$ are constant (equal to 1 w.l.o.g.) for all $(s,a) \in \mathcal{S} \times \mathcal{A}$.*

Note that under Assumption 3.3, it holds that $B_\star = D = T_\star$.

We introduce the infinite-horizon reward-based MDP $M_\infty \triangleq \langle \mathcal{S}, \mathcal{A}, g, r_\infty, P_\infty, s_0 \rangle$, with reward $r_\infty \triangleq \mathbb{1}_g$ and $P_\infty(\cdot \,|\, s,a) \triangleq P(\cdot \,|\, s,a)$ for $s \neq g$ and $P_\infty(\cdot \,|\, g,a) \triangleq \mathbb{1}_{s_0}$ for all a. In words, the transitions in M_∞ behave as in M and give zero rewards except at g where all actions give a reward of 1 and loop back to s_0 instead of self-looping with probability 1. We show that the solution of M_∞ coincides with solving the original online SSP and we bound the SSP-regret of UCRL2 applied to this problem.

Theorem 3.4. *For any policy $\pi \in \Pi$, let $\rho_\pi \triangleq \lim_{T\to+\infty} \mathbb{E}_\pi\big[\sum_{t=1}^T r_t/T\big]$ be the average reward of π in the MDP M_∞, where r_t denotes the reward received at time t. Under Assumption 3.3, we have*

$$\pi^\star = \arg\min_\pi V^\pi(s_0) = \arg\min_\pi T^\pi(s_0) = \arg\max_\pi \rho_\pi.$$

With probability $1-\delta$, UCRL2 run for any $K \geq 1$ episodes suffers a regret

$$R_K \leq 34(V^\star(s_0)+1)DS\sqrt{AT_K \log\Big(\frac{T_K}{\delta}\Big)}, \tag{3.3}$$

where we recall that $T_K \triangleq \sum_{k=1}^K I^k$ denotes the (random) total time elapsed over the K episodes. Moreover, under the same high-probability event, it holds that

$$T_K \leq 2(V^\star(s_0)+1)K + \widetilde{O}\left(V^\star(s_0)^2 D^2 S^2 A\right). \tag{3.4}$$

Proof sketch and discussion. Solving SSP with uniform costs corresponds to computing the policy that minimizes the expected time to reach the goal g. We first prove that solving its online formulation (which resets the agent once the goal is reached) is equivalent to maximizing the

long-term average reward (or gain) in M_∞. Then, we apply a similar analysis to Jaksch et al. (2010, Theorem 2) to derive an upper bound on the reward-based infinite-horizon regret of UCRL2, which we recall is defined as $R_T^\infty \triangleq T\rho_{\pi^\star} - \sum_{t=1}^T r_t$ for any time $T \geq 1$. Importantly, we refine the conventional regret bound of UCRL2 from $\widetilde{O}(D_\infty S\sqrt{AT})$ to $\widetilde{O}(DS\sqrt{AT})$, i.e., we replace the infinite-horizon diameter $D_\infty \triangleq \max_{s\neq s'\in\mathcal{S}'} \min_{\pi\in\Pi} \mathbb{E}\left[\tau_\pi(s \to s')\right]$ (Jaksch et al., 2010), which measures the longest shortest path between *any* two states.[2] In general $D \leq D_\infty$ and the gap between the two may be arbitrarily large. In fact, Assumption 2.7 does not imply that M_∞ is communicating, which is needed for proving regret bounds for UCRL2 in general MDPs. We show that even when M_∞ is weakly-communicating ($D_\infty = +\infty$) and some states may not be accessible from one another, UCRL2 is able to adapt to the SSP nature of the problem and achieve a bounded regret. We conclude the proof by relating the reward-based infinite-horizon regret of any algorithm to its SSP-regret as $R_K = (V^\star(s_0)+1)R_{T_K}^\infty$, which stems from the fact that $\rho_{\pi^\star} = 1/(V^\star(s_0)+1)$ by Markov chain theory (see Section B.1 for details). □

It is worth mentioning that no assumption is made about the properness of the policies. The key for UCRL2 to manage policies that may never reach the goal state is the construction of *internal* episodes, where policies are interrupted when the number of samples collected in a state-action pair is doubled. This allows UCRL2 to avoid accumulating too much regret when executing non-proper policies (they are eventually stopped) and, at the same time, perform well when the current policy is near-optimal (it is not stopped too early). Note that the stopping condition only relies on the number of samples and it is completely agnostic to the episodic nature of the SSP problem.

While the previous analysis suggests that algorithms for infinite-horizon MDPs could be readily executed in online SSP problems with strong regret guarantees, this is no longer the case when moving to the general setting of non-uniform costs. Indeed, in order to estimate the performance of a stationary policy w.r.t. its value function, we cannot use the average-cost criterion since it does not capture the incentive to reach the goal state. As an illustrative example, consider the deterministic two-state SSP M from Figure 3.1. The optimal SSP policy $\pi^\star$ always selects action a_2 since it has minimal value $V^\star(s_0) = c_{\max}$. The optimal infinite-horizon policy π^∞ always selects action a_1 since it has minimal average cost $\rho_{\pi^\infty} = c_{\min}$, whereas $\rho_{\pi^\star} = c_{\max}/2$. Consequently, running UCRL2 in general SSP may converge to a suboptimal policy and yield linear SSP-regret.

In an attempt to encourage the visit of the goal g, a natural idea could be to add a large reward $\overline{R}$ whenever it is reached. However, this may lead to policies that aim to reach g as fast as possible, completely disregarding the costs accumulated on the trajectory to g. Ideally $\overline{R}$

[2] While the analysis of UCRL2 leverages the fact that the range of the vector v_n computed by extended value iteration is bounded by D_∞, we can show that v_n can in fact be bounded by D in uniform-cost SSP problems. Furthermore, the condition that $D_\infty < +\infty$ is not required for the convergence of extended value iteration, since a sufficient condition for this is that M_∞ is weakly-communicating.

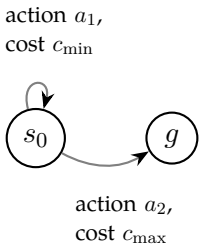


Figure 3.1 – Deterministic two-state SSP M with two available actions: a_1 self-loops on s_0 with cost $c_{\min}$ and a_2 goes from s_0 to g with cost $c_{\max} > 2c_{\min}$.

should be tuned to appropriately balance between the two objectives (minimize costs and reach the goal), yet doing so seems tricky without prior knowledge on the MDP and the optimal policy.

3.3 Three Desirable Properties of an Algorithm for Online SSP

We begin by stating the information-theoretic lower bound on the regret in SSP.

Proposition 3.5 (Rosenberg et al., 2020; Cohen et al., 2021)**.**

- *Let $B_\star \geq 2$, then there exists an SSP problem instance for $S \geq 2$, $A \geq 16$, $K \geq SA$, such that the expected regret of any learner after K episodes satisfies $\mathbb{E}[R_K] \geq \frac{1}{1024} B_\star \sqrt{SAK}$.*
- *Let $B_\star \leq \frac{1}{2}$, then there exists an SSP problem instance for $S \geq 2$, $A \geq 2$, $K \geq B_\star SA$, such that the expected regret of any learner after K episodes satisfies $\mathbb{E}[R_K] \geq \frac{1}{32} \sqrt{B_\star SAK}$.*

We now identify three desirable properties of a learning algorithm for online SSP.

- **Desired property 1: Minimax.** From Proposition 3.5, the information-theoretic lower bound on the regret scales as $\Omega(\sqrt{(B_\star^2 + B_\star)SAK})$.

> *An algorithm for online SSP is (nearly) minimax optimal if its regret is bounded by $\widetilde{O}(\sqrt{(B_\star^2 + B_\star)SAK})$, up to logarithmic factors and lower-order terms.*

- **Desired property 2: Parameter-free.** Another relevant dimension is the amount of prior knowledge required by the algorithm. While the knowledge of S, A, and the cost (or reward) range $[0, 1]$ is standard across regret-minimization settings (e.g., finite-horizon, discounted, average-reward), the complexity of learning in SSP problems may be linked to SSP-specific quantities such as $B_\star$ and $T_\star$.

> *An algorithm for online SSP is parameter-free if it relies neither on $T_\star$ nor $B_\star$ prior knowledge.*

- **Desired property 3: Horizon-free.** A core challenge in SSP is to trade off between minimizing costs and quickly reaching the goal state. This is accentuated when the instantaneous costs are small, i.e., when there is a mismatch between $B_\star$ and $T_\star$. Indeed, while $B_\star \leq T_\star$ always holds since the cost range is $[0, 1]$, the gap between the two may be arbitrarily large (see e.g., the simple example of Section B.2). The lower bound (Proposition 3.5) stipulates that the regret does depend on $B_\star$, while the "time horizon" of the problem, i.e., $T_\star$ should a priori not impact the regret, even as a lower-order term.

> *An algorithm for online SSP is (nearly) horizon-free if its regret depends only logarithmically on $T_\star$.*

Our definition extends the property of so-called horizon-free bounds recently uncovered in finite-horizon MDPs with total reward bounded by 1 (Wang et al., 2020a; Zhang et al., 2021d; Zhang et al., 2021e). These bounds depend only logarithmically on the horizon H, which is the number of time steps by which *any* policy terminates. Such notion of horizon would clearly be too strong in the more general class of SSP, where some (even most) policies may never reach the goal, thus having unbounded time horizon. A more adequate notion of horizon in SSP is $T_\star$, which bounds the *expected* time of the *optimal* policy to terminate the episode starting from any state. The fact that $T_\star$ is (a priori) unknown looks to be a significant difficulty than does not appear in finite-horizon where H is known and fixed.

Remark 3.6. Finally, while the previous properties focus on the learning aspects of the algorithm, another important consideration is computational efficiency. It is desirable that a learning algorithm has run-time complexity at most polynomial in $K, S, A, B_\star$, and $T_\star$. The two algorithms for online SSP proposed in this thesis (Chapters 4 and 5) meet such requirement.

3.4 On Regret-to-PAC in SSP

Assuming that we have access to a no-regret algorithm in SSP with an associated high-probability bound on its SSP-regret, a natural question that arises is whether we readily recover a PAC bound for SSP. Recall that the probably approximately correct (PAC) learning setting for RL provides sample complexity guarantees to find a near-optimal policy at the fixed initial state, i.e., a policy π such that $|V^\star(s_0) - V^\pi(s_0)| \leq \varepsilon$ for a prescribed accuracy level $\varepsilon > 0$ (with high probability). For instance, in finite-horizon MDPs, a regret bound can be converted to a PAC guarantee by selecting as a candidate optimal solution any policy chosen at random out of all episodes (Jin et al., 2018). Unfortunately, we make the observation that this procedure

cannot be applied in SSP. In fact, we recall from Definition 3.1 that the SSP-regret differs from the finite-horizon regret, since at each episode it compares the *empirical* costs accumulated along one trajectory with the optimal value function. As motivated in Section 3.1, a no-regret algorithm may need to change policies within an episode, and moreover none of them may actually be proper (i.e., have bounded value function). As such, it is unclear which policy should be retained or generated as a PAC solution candidate. In fact, it has been shown in the existing regret analyses for SSP (as we will see in Chapters 4 and 5) that explicitly guaranteeing the properness of the executed policies is not an intermediate step that is required to derive the regret bounds.

As a result, we notice that specific attention is required to derive sample complexity bounds for SSP. A first, simplified setting to do so is the *generative model*, which for any state-action pair (s, a) returns a sample drawn from $P(\cdot|s, a)$. In Tarbouriech et al. (2021b), we investigate: *How many calls to the generative model are sufficient to compute a near-optimal policy from any starting state with high probability?* In the interest of conciseness, we omit the details of our sample complexity analysis and refer an interested reader to Tarbouriech et al. (2021b). We point out that the latter bounds are not tight, and it is an interesting future direction to derive tight sample complexity bounds for SSP with a generative model, in a similar vein to the research line in the discounted MDP setting (Azar et al., 2013; Jiang, 2020; Agarwal et al., 2020; Li et al., 2020). Finally, moving beyond the generative model, we argue that deriving sample complexity bounds for fully online SSP adds a layer of complexity. Specifically, we will later prove in Lemma 8.7 that, without an additional assumption on the ability of the agent to take an anytime deterministic "reset" action to the initial state s_0, the sample complexity bounds that we can expect in online SSP are unavoidably worse that those in the generative model case.

Chapter 4

UC-SSP, the First Algorithm for Online SSP

In this chapter, we introduce UC-SSP (**U**pper **C**onfidence for **S**tochastic **S**hortest **P**ath), the first no-regret algorithm for online SSP. It relies on the principle of optimism in the face of uncertainty, and it handles the trade-off between minimizing costs and reaching the goal by crafting a novel stopping rule, such that UC-SSP may interrupt the current policy if it is taking too long to achieve the goal and switch to alternative policies that are designed to rapidly terminate the episode. Excluding the other dependencies, the regret bound of UC-SSP scales as $\widetilde{O}(\sqrt{K/c_{\min}})$ (when costs are lower bounded by $c_{\min} > 0$) or as $\widetilde{O}(K^{2/3})$ (under general non-negative costs), where K denotes the number of episodes. [1]

Contents

[1] This chapter is based on an article published in the proceedings of the 37^{th} International Conference on Machine Learning (ICML 2020) (Tarbouriech et al., 2020a).

4.1 Preliminaries

Additional Technical Tools. We first complement Chapters 2 and 3 with technical tools that will be of use exclusively in this chapter. For any $\pi \in \Pi_{\text{proper}}$, its (almost surely finite) hitting time to reach the goal starting from any state in $\mathcal{S}$ follows a *discrete phase-type distribution*, or in short *discrete PH distribution* (see e.g., Latouche and Ramaswami, 1999, Section 2.5 for an introduction). Indeed, its induced Markov chain is terminating with a single absorbing state g and all the other states are transient. The transition matrix associated to π, denoted by $P_\pi \in \mathbb{R}^{(S+1)\times(S+1)}$, can thus be arranged in the following canonical form

$$P_\pi = \begin{bmatrix} Q_\pi & R_\pi \\ 0 & 1 \end{bmatrix},$$

where $Q_\pi \in \mathbb{R}^{S\times S}$ is the transition matrix between non-absorbing states (i.e., $\mathcal{S}$) and $R_\pi \in \mathbb{R}^S$ is the transition vector from $\mathcal{S}$ to g. Note that Q_π is strictly substochastic ($Q_\pi \mathbb{1} \leq \mathbb{1}$ where $\mathbb{1} \triangleq (1,\dots,1)^T \in \mathbb{R}^S$ and $\exists j$ s.t. $(Q_\pi \mathbb{1})_j < 1$). Denoting by $\mathbb{1}_s$ the S-sized one-hot vector at the position of state $s \in \mathcal{S}$, the following result holds (see e.g., Latouche and Ramaswami, 1999, Theorem. 2.5.3).

Proposition 4.1. *For any $\pi \in \Pi_{\text{proper}}$, $s \in \mathcal{S}$ and $n > 0$,*

$$\mathbb{P}(\tau_\pi(s) > n) = \mathbb{1}_s^\top Q_\pi^n \mathbb{1} = \sum_{s' \in \mathcal{S}} (Q_\pi^n)_{ss'}.$$

Finally, for any $X \in \mathbb{R}^{m\times n}$ we define the ∞-matrix-norm $\|X\|_\infty \triangleq \max_{1\leq i\leq m} \sum_{j=1}^n |X_{ij}|$.

Analysis Road Map. First, we assume that the costs are strictly positive, which ensures that the conditions of Proposition 2.11 hold, thus making it easier analysis-wise.

Assumption 4.2 (for Sections 4.2 to 4.4). *The costs are known, deterministic and positive, i.e., there exist constants $0 < c_{\min} \leq c_{\max}$ such that $c(s,a) \in [c_{\min}, c_{\max}]$ for all $(s,a) \in \mathcal{S} \times \mathcal{A}$.*

Extending the setting to unknown, stochastic costs poses no major difficulty, as long as the learner knows in advance the range of the costs, i.e., the constant $c_{\min}$ and $c_{\max}$. Then, in Section 4.5, we derive a variant of our algorithm that can handle zero costs (i.e., $c_{\min} = 0$) by performing a cost perturbation argument.

Algorithm 4.1: Algorithm UC-SSP

Input: Confidence $\delta \in (0,1)$, costs, $\mathcal{S}', \mathcal{A}$.
Initialization: Set the state-action counter $N_{0,0}(s,a) \triangleq 0$ for any $(s,a) \in \mathcal{S} \times \mathcal{A}$ and the time step $t \triangleq 1$.
Set $k \triangleq 0$. \\ *episode index*
Set $G_{0,0} \triangleq 0$. \\ *number of attempts in phases ②*
while $k < K$ **do**
  Increment $k \mathrel{+}= 1$.
  Set $j \triangleq 0$. \\ *attempts in phase ② of episode k*
  while $s_t \neq g$ **do**
    Set $t_{k,j} \triangleq t$ and counter $\nu_{k,j}(s,a) \triangleq 0$.
    Set $G_{k,j} = G_{k,0} + j$.
    Compute $(\widetilde{\pi}_{k,j}, H_{k,j}) \triangleq \text{EVI}_{\text{SSP}}(k,j)$.
    while $t \leq t_{k,j} + H_{k,j}$ **and** $s_t \neq g$ **do**
      Execute action $a_t = \widetilde{\pi}_{k,j}(s_t)$, observe cost $c(s_t,a_t)$ and next state s_{t+1}.
      Set $\nu_{k,j}(s_t,a_t) \mathrel{+}= 1$ and $t \mathrel{+}= 1$.
    if $s_t \neq g$ **then**
      \\ *Switch to phase ②*
      Set $N_{k,j+1}(s,a) \triangleq N_{k,j}(s,a) + \nu_{k,j}(s,a)$ and increment $j \mathrel{+}= 1$.
  Set $N_{k+1,0}(s,a) \triangleq N_{k,j}(s,a) + \nu_{k,j}(s,a)$ and $G_{k+1,0} \triangleq G_{k,j}$.

4.2 The UC-SSP Algorithm

Recall that the general SSP problem requires (i) to quickly reach the goal state while (ii) at the same time minimizing the cumulative costs. As shown in Section 3.2, if we constrain the costs to be all equal, objectives (i) and (ii) coincide and the SSP problem can for example be addressed using infinite-horizon algorithms.

In Algorithm 4.1 we present UC-SSP (**U**pper **C**onfidence for **S**tochastic **S**hortest **P**ath), the first algorithm for efficient exploration in general SSP problems. At a high level, UC-SSP proceeds through each episode k in a *two-phase* fashion and handles the aforementioned trade-off by crafting a novel stopping rule. In phase ①, UC-SSP executes a policy trying to solve the SSP problem by tackling both objectives (i) and (ii) (i.e., reach the goal while minimizing the cumulative costs). We refer to this first policy as an *attempt* in phase ①. As UC-SSP relies on estimates of the true (unknown) SSP, it may select a non-proper policy that would never reach the goal state and incur an unbounded regret. In order to avoid this situation, if the goal state is not reached after a given *pivot* horizon, the algorithm deems the whole episode as a *failure* and it switches to phase ②, whose only objective is to terminate the episode as fast as possible (i.e., it only considers objective (i) and disregards the costs). Nonetheless, optimizing an estimate of the hitting time (i.e., objective (i)) does not guarantee that the corresponding policy successfully reaches the goal state (i.e., is proper) and multiple *attempts* (i.e., policies) in phase ② may be needed. Similar to phase ①, whenever the goal state is not reached after a

Algorithm 4.2: EVI$_{\text{SSP}}$ planning procedure

Input: Attempt index (k, j) and $N_{k,j}(s, a)$ samples for every (s, a).
if $j = 0$ **then**
  $\varepsilon_{k,0} \triangleq \frac{c_{\min}}{2t_{k,0}}$, $\gamma_{k,j} \triangleq \frac{1}{\sqrt{k}}$.
else
  $\varepsilon_{k,j} \triangleq \frac{1}{2t_{k,j}}$, $\gamma_{k,j} \triangleq \frac{1}{\sqrt{G_{k,j}}}$.
Compute estimates $\widehat{P}_{k,j}$ and confidence set $\mathcal{M}_{k,j}$ with the $N_{k,j}$ samples collected so far.
Define the extended optimal Bellman operator $\widetilde{\mathcal{L}}_{k,j}$ as in Equation (4.1).
\\ *EVI scheme*
Set $m \triangleq 0$, $v_0 \triangleq \mathbf{0}$ (S-sized vector) and $v_1 \triangleq \widetilde{\mathcal{L}}_{k,j} v_0$.
while $\|v_{m+1} - v_m\|_\infty > \varepsilon_{k,j}$ **do**
  $m \mathrel{+}= 1$ and $v_{m+1} \triangleq \widetilde{\mathcal{L}}_{k,j} v_m$.
Set $\widetilde{v}_{k,j} \triangleq v_m$.
Compute $\widetilde{\pi}_{k,j}$ the optimistic greedy policy w.r.t. $\widetilde{v}_{k,j}$.
Compute $\widetilde{P}_{k,j}$ the corresponding optimistic model.
Compute $\widetilde{Q}_{k,j}$ the transition matrix of $\widetilde{\pi}_{k,j}$ in the model $\widetilde{P}_{k,j}$ over $\mathcal{S}$, i.e., for any $(s, s') \in \mathcal{S}^2$,

$$\widetilde{Q}_{k,j}(s, s') \triangleq \sum_{a \in \mathcal{A}} \widetilde{\pi}_{k,j}(a|s) \widetilde{P}_{k,j}(s'|s, a).$$

Compute $H_{k,j} \triangleq \min\left\{n > 1 : \|\widetilde{Q}_{k,j}^{n-1}\|_\infty \leq \gamma_{k,j}\right\}$.
Output: policy $\widetilde{\pi}_{k,j}$ and horizon $H_{k,j}$.

certain *pivot* horizon, the current policy is terminated and a new policy is computed. Phase ② and the overall episode ends when the goal state is eventually reached. Notation-wise, the k-th phase ① is indexed by $(k, 0)$ (note that k coincides with the current number of episodes), while the j-th attempt in the phase ② of episode k is indexed by (k, j) for $j \geq 1$. Moreover, we denote by J_k the number of attempts performed during the phase ② of episode k, and by $G_{k,j}$ the total number of attempts in phases ② up to (and including) attempt (k, j).

Optimistic policies. UC-SSP relies on the principle of *optimism in face of uncertainty*. At each attempt, it executes a policy with either lowest optimistic (cost-weighted) value for an attempt in phase ①, or with lowest optimistic expected hitting time for an attempt in phase ②. At the beginning of any attempt (k, j), the algorithm computes a set of plausible MDPs defined as $\mathcal{M}_{k,j} \triangleq \{\langle \mathcal{S}, \mathcal{A}, c, \widetilde{P}\rangle \mid \widetilde{P}(\cdot|s, a) \in B_{k,j}(s, a)\}$ where $B_{k,j}(s, a)$ is a high-probability confidence set on the transition probabilities of the true MDP M. We set

$$B_{k,j}(s, a) \triangleq \left\{\widetilde{P} \in \mathcal{C} \mid \widetilde{P}(\cdot \mid g, a) = \mathbb{1}_g, \|\widetilde{P}(\cdot \mid s, a) - \widehat{P}_{k,j}(\cdot \mid s, a)\|_1 \leq \beta_{k,j}(s, a)\right\},$$

with $\mathcal{C}$ the S'-dimensional simplex, $\widehat{P}_{k,j}$ the empirical average of transitions prior to attempt (k,j) and

$$\beta_{k,j}(s,a) \triangleq \sqrt{\frac{8S\log\left(2AN_{k,j}^{+}(s,a)\,\delta^{-1}\right)}{N_{k,j}^{+}(s,a)}},$$

where $N_{k,j}^{+}(s,a) \triangleq \max\{1, N_{k,j}(s,a)\}$ with $N_{k,j}$ being the state-action counts prior to attempt (k,j). The construction of $\beta_{k,j}(s,a)$ guarantees that $M \in \mathcal{M}_{k,j}$ with high probability, as shown in the following lemma.

Lemma 4.3. *Introduce the event* $\mathcal{E} \triangleq \bigcap_{k=1}^{+\infty}\bigcap_{j=1}^{J_k}\{M \in \mathcal{M}_{k,j}\}$. *Then* $\mathbb{P}(\mathcal{E}) \geq 1 - \frac{\delta}{3}$.

Once $\mathcal{M}_{k,j}$ has been computed, UC-SSP applies an extended value iteration (EVI) scheme (Algorithm 4.2) to compute a policy with lowest optimistic value (if $j = 0$) or lowest optimistic expected hitting time (if $j \geq 1$). Formally, we define the extended optimal Bellman operator $\widetilde{\mathcal{L}}_{k,j}$ such that for any $v \in \mathbb{R}^{\mathcal{S}}$ and $s \in \mathcal{S}$,

$$\widetilde{\mathcal{L}}_{k,j}v(s) \triangleq \min_{a\in\mathcal{A}}\Big\{c_{k,j}(s,a) + \min_{\widetilde{P}\in B_{k,j}(s,a)}\sum_{y\in\mathcal{S}}\widetilde{P}(y\,|\,s,a)v(y)\Big\}, \tag{4.1}$$

where the costs $c_{k,j}$ depend on the phase as follows

$$c_{k,j}(s,a) \triangleq \begin{cases} c(s,a) & \text{if } j = 0 \\ 1 & \text{otherwise.} \end{cases}$$

As explained by Jaksch et al. (2010, Section 3.1), we can combine all the MDPs in $\mathcal{M}_{k,j}$ into a single MDP $\widetilde{M}$ with extended action set $\mathcal{A}'$. Using the generalization of the SSP results to a compact action set (see e.g., Bertsekas and Yu, 2013), it holds that EVI$_{\text{SSP}}$ converges to a vector denoted by $\widetilde{V}_{k,j}^{\star}$. We have the following component-wise inequalities when the stopping condition of Algorithm 4.2 is met.[2]

Lemma 4.4. *For any attempt* (k,j), *denote by* $\widetilde{v}_{k,j}$ *the output of* EVI$_{\text{SSP}}$ *with operator* $\widetilde{\mathcal{L}}_{k,j}$ *and accuracy* $\varepsilon_{k,j}$. *Then* $\widetilde{\mathcal{L}}_{k,j}\widetilde{v}_{k,j} \leq \widetilde{v}_{k,j} + \varepsilon_{k,j}$. *Furthermore, under the event* $\mathcal{E}$ *we have* $\widetilde{v}_{k,j} \leq V^{\star}$ *if* $j = 0$ *or* $\widetilde{v}_{k,j} \leq \min_{\pi}\mathbb{E}[\tau_{\pi}]$ *otherwise.*

[2]Note that the stopping condition is different from the standard one for VI for average reward MDPs (see e.g., Puterman, 2014; Jaksch et al., 2010) that is defined in span seminorm. Also note that as opposed to standard VI, we do not have guarantees of the type $\|v_n - \widetilde{V}_{k,j}^{\star}\|_{\infty} \leq \varepsilon$ where $\widetilde{V}_{k,j}^{\star} = \widetilde{\mathcal{L}}_{k,j}\widetilde{V}_{k,j}^{\star}$.

The optimistic policy $\widetilde{\pi}_{k,j}$ executed during attempt (k,j) is the greedy policy w.r.t. $\widetilde{v}_{k,j}$. We also denote by $\widetilde{P}_{k,j}$ the optimistic transition probabilities and by $\widetilde{Q}_{k,j}$ the transition matrix of $\widetilde{\pi}_{k,j}$ in $\widetilde{P}_{k,j}$ over the non-goal states $\mathcal{S}$.

The pivot horizon. A crucial aspect for the correct functioning of the algorithm is to carefully select the "pivot" horizon. If the pivot horizon is too small, the algorithm may switch from phase ① to ② too quickly and may perform too many attempts in phase ②. As the policies in phase ② completely disregard the costs, they may lead to suffer large regret. On the other hand, if the pivot horizon is too large and UC-SSP selects a non-proper policy in phase ①, then the regret accumulated during phase ① would be too large.

We select the following length for attempt (k,j)

$$H_{k,j} = \min\left\{n>1 : \|(\widetilde{Q}_{k,j})^{n-1}\|_\infty \leq \frac{\mathbb{1}_{j=0}}{\sqrt{k}} + \frac{\mathbb{1}_{j\geq 1}}{\sqrt{G_{k,j}}}\right\}. \tag{4.2}$$

If $\widetilde{\pi}_{k,j}$ is executed for $H_{k,j}$ steps without reaching g, then attempt (k,j) is said to have *failed* and the next attempt $(k,j+1)$ (necessarily in phase ②) is performed. Otherwise, the attempt is said to have *succeeded*, a new episode begins and the next attempt $(k+1,0)$ (in phase ①) is performed.

Denote by $\widetilde{\tau}_{k,j}$ the hitting time to the goal in the model $\widetilde{P}_{k,j}$ of the policy $\widetilde{\pi}_{k,j}$. We first prove that $\widetilde{\pi}_{k,j}$ is proper in $\widetilde{P}_{k,j}$ by connecting its value function to $\widetilde{v}_{k,j}$, which is finite from Lemma 4.4 (see Section C.1.4 and Equation (C.9)). As a result, $\widetilde{\tau}_{k,j}$ follows a *discrete PH distribution* and plugging Proposition 4.1 into Equation (4.2) entails that

$$\max_{s\in\mathcal{S}} \mathbb{P}(\widetilde{\tau}_{k,j}(s) \geq H_{k,j}) \leq \frac{\mathbb{1}_{j=0}}{\sqrt{k}} + \frac{\mathbb{1}_{j\geq 1}}{\sqrt{G_{k,j}}}.$$

$H_{k,j}$ is thus selected so that the tail probability of the *optimistic* hitting time is small enough, i.e., there is a high probability that $\widetilde{\pi}_{k,j}$ will *optimistically* reach g within $H_{k,j}$ steps. The maximum over $s\in\mathcal{S}$ guarantees this property for any state s from which attempt (k,j) begins (since attempts in phase ② do not necessarily start at s_0).

4.3 Regret Guarantee

As shown below, UC-SSP is the first no-regret learning algorithm in the general SSP setting.

Theorem 4.5. *With overwhelming probability, for any $K\geq 1$, if at each attempt (k,j) EVI$_{SSP}$ is run with accuracy $\varepsilon_{k,j} \triangleq \frac{c_{\min}\mathbb{1}_{j=0}+\mathbb{1}_{j\geq 1}}{2t_{k,j}}$, where $t_{k,j}$ is the time index at the beginning of the attempt,*

then UC-SSP *suffers a regret*

$$R_K = \widetilde{O}\left(c_{\max} DS\sqrt{\frac{c_{\max}}{c_{\min}} ADK} + c_{\max} S^2 AD^2\right).$$

Dependence on K and D. Significantly, under positive costs, UC-SSP achieves an overall rate $\widetilde{O}(\sqrt{K})$ which is optimal w.r.t. the number of episodes K. The bound also illustrates how UC-SSP is able to adapt to the complexity of navigating through the MDP as shown by the dependency on the SSP-diameter D, which measures the longest shortest path to the goal from any state. Interestingly, this is achieved without any prior knowledge either on an upper bound of the optimal value function $V^\star$ (or of the SSP-diameter itself), i.e., UC-SSP is *parameter-free*. We can further inspect the dependency on D by rewriting the regret bound of UC-SSP, which scales as $D^{3/2}\sqrt{K}$ in Theorem 4.5, as $D\sqrt{T_K}$, where T_K is the total number of steps executed until the end of episode of K.[3] As shown in Equation (3.2), up to a factor of $c_{\max}$, the SSP-diameter D is an upper bound on the range of the optimal value function and as such it can be (qualitatively) related to the horizon H in the finite-horizon setting and the diameter D_∞ in the infinite-horizon setting, which bound the range of the optimal value function and bias function respectively.

Dependence on cost range. The multiplicative constant $\frac{c_{\max}}{c_{\min}}$ appearing in the bound quantifies the range of the cost function and accounts for the difference from the uniform-cost setting. Interestingly, the presence of the ratio $\frac{c_{\max}}{c_{\min}}$ implies that the regret bound is not invariant w.r.t. a uniform additive perturbation of all costs. This behavior, which does not appear in the finite- or infinite-horizon settings, stems from the fact that an additive offset of costs may alter the optimal policy in the SSP sense (see Lemma C.5, Section C.2).

While the previous discussion shows that UC-SSP successfully tackles general SSP problems, we can also study its behavior in the limit (and much simpler) cases of uniform-cost and loop-free SSP, and compare its regret to infinite- and finite-horizon algorithms respectively.

Uniform-cost SSP. Under Assumption 3.3, UC-SSP achieves a regret of $\widetilde{O}(DS\sqrt{ADK})$, in contrast with the bound $\widetilde{O}(V^\star(s_0)DS\sqrt{AV^\star(s_0)K})$ of UCRL2 derived in Section 3.2. While in this restricted setting UCRL2 performs better when s_0 is a privileged starting state to reach g compared to the rest of states in $\mathcal{S}$, UC-SSP yields an improvement over UCRL2 whenever $V^\star(s_0) \geq D^{1/3}$. Our experiments in Section C.3 illustrate that UC-SSP suffers smaller regret than UCRL2 in a gridworld with uniform costs, showcasing that UC-SSP manages to better adapt to the goal-oriented structure of the problem.

[3] Even though T_K is a *random* quantity, inspecting the proof (see Section 4.4) provides a bound $T_K \lesssim DK$ for K large enough.

Loop-free SSP. Let us assume that there exists a *known* upper bound H on the hitting time of *any* policy. Then a slight variation of the finite-horizon algorithm UCBVI (Azar et al., 2017) can be applied. While its bound would scale as $\widetilde{O}(\sqrt{HSAT})$ and showcase an improved $\sqrt{S}$-dependency, it would regrettably scale with $\sqrt{H}$ which may be much larger than the D factor appearing in Theorem 4.5 as soon as the hitting times τ_π differ significantly across policies π. Moreover, UC-SSP does not require the prior knowledge of H, as opposed to UCBVI or any other existing algorithm in the finite-horizon or loop-free setting.

The analysis of UC-SSP reveals the crucial role of the pivot horizon in shaping the behavior and performance of the algorithm. In the uniform-cost case, EVI$_{\text{SSP}}$ and standard EVI used in UCRL2 both converge to the same policy. The main difference between the two algorithms consists in the stopping criterion for the execution of the optimistic policy. While UCRL2 applies a generic doubling scheme (i.e., an internal episode is terminated when the number of samples is doubled in at least a state-action pair), UC-SSP leverages the episodic nature of the SSP problem and sets a pivot horizon such that the current policy should successfully terminate with high (optimistic) probability. In the loop-free setting, UCBVI picks a single policy per episode and waits until termination. While all policies are guaranteed to terminate in finite time, the length of the episode may still be very long. On the other hand, UC-SSP goes through different policies within each episode whenever they are taking *too long* to reach the goal state.

4.4 Regret Analysis

In this section, we provide a proof sketch of Theorem 4.5. The SSP-regret introduced in Definition 3.1 can neither be managed by a step-by-step comparison between the algorithmic and optimal performances as in infinite-horizon, nor by an episode-by-episode comparison as in finite-horizon. We thus need to derive a new analysis to handle the specificities of the SSP-regret. Denoting by T_K the total number of steps at the end of episode K, we decompose $T_K = T_{K,1} + T_{K,2}$, with $T_{K,1}$ (resp. $T_{K,2}$) the total time during attempts in phase ① (resp. phase ②). We introduce the *truncated* regret

$$\mathcal{W}_K \triangleq \sum_{k=1}^{K} \left[\Big(\sum_{h=1}^{H_{k,0}} c(s_{k,h}, \widetilde{\pi}_{k,0}(s_{k,h})) \Big) - V^\star(s_0) \right], \tag{4.3}$$

which is obtained by considering the cumulative cost up to $H_{k,0}$ steps rather than for the actual duration of each attempt in phase ①. By assigning a regret of $c_{\max}$ to each step in phase ②, we can then decompose the regret of UC-SSP as

$$R_K \leq \mathcal{W}_K + c_{\max} T_{K,2}. \tag{4.4}$$

This decomposition directly justifies the different nature of the two phases employed by UC-SSP. While phase ① directly tries to minimize $\mathcal{W}_K$, phase ② only needs to keep $T_{K,2}$ under control, which requires executing policies that reach the goal state as quickly as possible.

Bound on $\mathcal{W}_K$. We first bound $\mathcal{W}_K$ by drawing inspiration from techniques in the finite-horizon setting (see e.g., Azar et al., 2017), by successively unrolling the Bellman operator to get a telescopic sum which can be bounded using the Azuma-Hoeffding inequality and a pigeonhole principle.

Lemma 4.6. *Introduce $\Omega_K \triangleq \max_{k\in[K]} H_{k,0}$. With probability at least $1-\delta$,*

$$\mathcal{W}_K = O\left(c_{\max} DS\sqrt{A\Omega_K K \log\left(\frac{\Omega_K K}{\delta}\right)}\right).$$

Bound on Ω_K. On the one hand, since $\mathcal{W}_K$ directly scales with $\sqrt{\Omega_K}$, we must ensure that the lengths of attempts in phase ① are not too long. Ideally, we would set them as relatively tight upper bounds of $V^\star(s_0)$ or D, yet these are critically *unknown*. Instead, in Equation (4.2) we tune the lengths $H_{k,0}$ depending on optimistic quantities (which can be easily computed at the start of each attempt), and prove in the following lemma that they crucially scale as $\widetilde{O}(D)$.

Lemma 4.7. *Under the event $\mathcal{E}$,*

$$\Omega_K \leq \left\lceil 6\frac{c_{\max}}{c_{\min}} D\log(2\sqrt{K})\right\rceil.$$

Proof sketch. Consider a state $y \in \mathcal{S}$ such that

$$\|(\widetilde{Q}_{k,0})^{H_{k,0}-2}\|_\infty = \mathbb{1}_y^\top (\widetilde{Q}_{k,0})^{H_{k,0}-2}\mathbb{1}.$$

From Proposition 4.1, the above is equal to $\mathbb{P}(\widetilde{\tau}_{k,0}(y) \geq H_{k,0}-1)$. To bound it, we apply a corollary of Markov's inequality

$$\mathbb{P}(\widetilde{\tau}_{k,0}(y) \geq H_{k,0}-1) \leq \frac{\mathbb{E}\left[(\widetilde{\tau}_{k,0})^r\right]}{(H_{k,0}-1)^r},$$

for a carefully chosen exponent $r \triangleq \lceil \log(2\sqrt{k})\rceil \geq 1$. We then prove that $\widetilde{\tau}_{k,0}$ follows a discrete PH distribution that satisfies $\mathbb{E}\left[\widetilde{\tau}_{k,0}(s)\right] \leq \frac{2c_{\max}D}{c_{\min}}$ for all $s \in \mathcal{S}$. This leads us to derive an upper

bound on the r-th moment of any hitting time distribution with bounded expectation starting from any state (Lemma C.2, which may be of independent interest). Applying it to $\widetilde{\tau}_{k,0}$ yields

$$\mathbb{E}\left[(\widetilde{\tau}_{k,0})^r\right] \leq 2\left(r\frac{2c_{\max}D}{c_{\min}}\right)^r,$$

which gives on the one hand

$$\|(\widetilde{Q}_{k,0})^{H_{k,0}-2}\|_\infty \leq \frac{2\left(r\frac{2c_{\max}D}{c_{\min}}\right)^r}{(H_{k,0}-1)^r}.$$

On the other hand, the choice of $H_{k,0}$ in Equation (4.2) entails that

$$\frac{1}{\sqrt{k}} < \|(\widetilde{Q}_{k,0})^{H_{k,0}-2}\|_\infty.$$

Combining the two previous inequalities finally provides the desired upper bound on $H_{k,0}$. □

Bound on $T_{K,2}$. On the other hand, since $T_{K,2}$ increases with the number of attempts in phase ②, we must ensure that there are not too many of such attempts and that their lengths can be adequately controlled. In light of this and leveraging the way the length $H_{k,0}$ is constructed in Equation (4.2), we bound the number of failed attempts in phase ① up to episode K, which we denote by F_K.

Lemma 4.8. *With probability at least* $1-\delta$,

$$F_K \leq 2\sqrt{K} + 2\sqrt{2\Omega_K K \log\left(\frac{2(\Omega_K K)^2}{\delta}\right)} + 4S\sqrt{8A\Omega_K K \log\left(\frac{2A\Omega_K K}{\delta}\right)}.$$

Proof sketch. We decompose $F_K = F'_K + F''_K$, where $F'_K \triangleq \sum_{k=1}^K \mathbb{P}(\widetilde{\tau}_{k,0}(s_0) > H_{k,0})$ and $F''_K \triangleq \sum_{k=1}^K \left[\mathbb{1}_{\{\tau_{k,0}(s_0)>H_{k,0}\}} - \mathbb{P}(\widetilde{\tau}_{k,0}(s_0) > H_{k,0})\right]$. A martingale argument and the pigeonhole principle bound F''_K, while the choice of $H_{k,0}$ controls each summand of F'_K. □

Equipped with Lemma 4.8, we proceed in bounding the total duration of the attempts in phase ②.

Lemma 4.9. *With probability at least* $1-\delta$,

$$T_{K,2} = \widetilde{O}\left(DS\sqrt{\frac{c_{\max}}{c_{\min}}ADK} + S^2AD^2\right).$$

Proof sketch. We have $T_{K,2} \leq \Omega'_K G_K$, where we set $\Omega'_K \triangleq \max_{k\in[K]} \max_{j\in[J_k]} H_{k,j}$ and $G_K \triangleq \sum_{k=1}^{K} J_k$ which is the total number of attempts in phase ② up to episode K. Ω'_K can be bounded in a similar way as done in Lemma 4.7. G_K can be bounded by decomposing it as the sum of the number of attempts in phase ② that succeed in reaching g (equal to F_K which is upper bounded by Lemma 4.8) and of the number of attempts in phase ② that fail in reaching g (which can be upper bounded in the same vein as in Lemma 4.8). □

Putting everything together, we obtain Theorem 4.5 by plugging Lemma 4.6, 4.7 and 4.9 into Equation (4.4). Note that while the regret decomposition in the two-phase process of Equation (4.4) has the advantage of making the analysis intuitive and modular, it renders Bernstein techniques less effective in capturing low-variance deviations, as opposed to the analysis of UCBVI and UCRL2B (Fruit et al., 2020) which can shave off a term of $\sqrt{H}$ or $\sqrt{D_\infty}$ for large enough time steps in the finite- and infinite-horizon settings, respectively.

4.5 Relaxation of Assumptions

Although Assumptions 2.7 and 4.2 seem natural in the SSP problem, we design variants of UC-SSP that can handle dead-end states and/or zero costs. We defer to Section C.2 the complete analysis.

Relaxation of Assumption 4.2 ($c_{\min} = 0$). We observe that having $c_{\min} = 0$ renders the bound on Ω_K of Lemma 4.7 vacuous. To circumvent this issue, we introduce an additive perturbation $\eta_{k,0} > 0$ to the cost of each transition in the *optimistic model* of each attempt $(k, 0)$. Our resulting variant of UC-SSP achieves a $\widetilde{O}(K^{2/3})$ regret bound (see Lemma C.6 for the complete bound). The difference in rate ($K^{2/3}$ vs. $\sqrt{K}$) compared to Theorem 4.5 stems from the fact that our procedure of offsetting the costs introduces a bias, which we minimize with the choice of perturbation $\eta_{k,0} = 1/k^{1/3}$.

Relaxation of Assumption 2.7 ($D = +\infty$). If M is non-SSP-communicating, there exists at least one (possibly unknown) *dead-end* state from which reaching the goal g is impossible. This implies that EVI$_{\text{SSP}}$, which operates on the entire state space $\mathcal{S}$, fails to converge since the values at dead-end states are infinite. To tackle this problem, we assume that the agent has prior knowledge on an upper bound $J \geq V^\star(s_0)$ and that it has at any time step the "resetting" ability to transition with probability 1 to s_0 with a cost of J (to prevent it from getting stuck). Equipped with these two assumptions, by optimizing a value function that is *truncated* at J (Kolobov et al., 2012), we prove that a variant of UC-SSP achieves a regret guarantee identical to Theorem 4.5 except that the infinite term D is replaced by J (see Lemma C.4).

4.6 Discussion and Bibliographical Remarks

In this chapter, we presented UC-SSP, the first algorithm for online SSP with sublinear regret guarantees. Note that UC-SSP is parameter-free, as it does not require prior knowledge on the difficulty of reaching the goal. Excluding the other dependencies, the regret bound scales as $\widetilde{O}(\sqrt{K/c_{\min}})$ (when costs are lower bounded by $c_{\min} > 0$) or as $\widetilde{O}(K^{2/3})$ (under general non-negative costs). These bounds display a gap with respect to the lower bound (Proposition 3.5).

A first natural question opened by our work was whether the dependence on $c_{\min}^{-1}$ could be removed in the leading term, which would allow $\widetilde{O}(\sqrt{K})$ regret under general non-negative costs. This was answered positively by the work of Rosenberg et al. (2020), which devised an algorithm that utilizes confidence sets based on the Bernstein concentration inequality (instead of Hoeffding inequality), which enables to be sensitive to the variance of the value function at a next state given some state-action pair visited by the algorithm. Their improved regret bounds scale as $\widetilde{O}(B_\star^{3/2} S\sqrt{AK})$ (or $\widetilde{O}(B_\star S\sqrt{AK})$ if $B_\star$ is known). This still reveals a $\sqrt{B_\star S}$ (resp. $\sqrt{S}$) mismatch with respect to the lower bound $\Omega(B_\star\sqrt{SAK})$ for $B_\star \geq 1$ (Rosenberg et al., 2020). In Chapter 5 we will see how this gap can be closed.

An extension of online SSP is when the costs are allowed to adversarially change, which is also known as *adversarial SSP*. This was recently investigated by Rosenberg and Mansour (2021), Chen et al. (2021c), and Chen and Luo (2021), by building on the online mirror descent framework for online convex optimization. Adversarial SSP poses additional technical challenges, and it remains an open question to derive tight regret guarantees in this setting. Interestingly, the lower bound in adversarial SSP showcases an extra dependence on $T_\star$, as it is $\Omega(\sqrt{DT_\star K} + D\sqrt{SAK})$ for the full information setting and $\Omega(\sqrt{SADT_\star K} + D\sqrt{SAK})$ for the bandit feedback setting (Chen and Luo, 2021).

Chapter 5

EB-SSP, an Optimal Algorithm for Online SSP

In this chapter, we introduce EB-SSP (**E**xploration **B**onus for **S**tochastic **S**hortest **P**ath). The algorithm relies on carefully skewing the empirical transitions and perturbing the empirical costs with an exploration bonus to induce an optimistic SSP problem whose associated value iteration scheme is guaranteed to converge. We prove that EB-SSP achieves the minimax regret rate of $\widetilde{O}(\sqrt{(B_\star + B_\star^2)SAK})$, where we recall that K is the number of episodes, S is the number of states, A is the number of actions, and $B_\star$ bounds the expected cumulative cost of the optimal policy from any state, thus closing the gap with the lower bound. EB-SSP obtains this result while being parameter-free, i.e., it does not require any prior knowledge of $B_\star$, nor of $T_\star$, which bounds the expected time-to-goal of the optimal policy from any state. Furthermore, we illustrate various cases (e.g., positive costs, or general costs when an order-accurate estimate of $T_\star$ is available) where the regret only contains a logarithmic dependence on $T_\star$, thus yielding the first (nearly) horizon-free regret bound beyond the finite-horizon MDP setting. [1]

Contents

[1] This chapter is based on an article published in the proceedings of the 34^{th} Conference on Neural Information Processing Systems (NeurIPS 2021) (Tarbouriech et al., 2021c).

5.1 The EB-SSP Algorithm

We introduce our algorithm EB-SSP (**E**xploration **B**onus for **S**tochastic **S**hortest **P**ath) in Algorithm 5.1. It takes as input the state-action space $\mathcal{S} \times \mathcal{A}$ and confidence level $\delta \in (0, 1)$. For now it considers that an estimate B such that $B \geq \max\{B_\star, 1\}$ is available, and we later handle the case of unknown $B_\star$ (Section 5.5 and Section D.7). As explained in Section 2.4, the algorithm enforces the conditions of Proposition 2.11 to hold by adding a small cost perturbation $\eta \in [0, 1]$ (cf. lines 3, 12 in Algorithm 5.1) — either $\eta = 0$ if the agent is aware that all costs are already positive, otherwise a careful choice of $\eta > 0$ is provided in Section 5.3.

Our algorithm builds on a value-optimistic approach by sequentially constructing optimistic lower bounds on the optimal Q-function and executing the policy that greedily minimizes them. Similar to the MVP algorithm of Zhang et al. (2021d) designed for finite-horizon RL, we adopt the doubling update framework (first proposed by Jaksch et al., 2010): whenever the number of visits of a state-action pair is doubled, the algorithm updates the empirical cost and transition probability of this state-action pair, and computes a new optimistic Q-estimate and optimistic greedy policy. Note that this slightly differs from MVP which waits for the end of its finite-horizon episode to update the policy. In SSP, however, having this delay may yield linear regret as the episode has the risk of never terminating under the current policy (e.g., if it is improper), which is why we perform the policy update instantaneously when the doubling condition is met.

The main algorithmic component lies in how to compute the Q-values (w.r.t. which the policy is greedy) when a doubling condition is met. To this purpose, we introduce a procedure called VISGO, for (**V**alue **I**teration with **S**light **G**oal **O**ptimism). Starting with optimistic values $V^{(0)} = 0$, it iteratively computes $V^{(i+1)} = \widetilde{\mathcal{L}} V^{(i)}$ for a carefully defined operator $\widetilde{\mathcal{L}}$. It ends when a stopping condition is met, specifically once $\|V^{(i+1)} - V^{(i)}\|_\infty \leq \varepsilon_{\text{VI}}$ for a precision level $\varepsilon_{\text{VI}} > 0$ (specified later), and it outputs the values $V^{(i+1)}$ (and Q-values $Q^{(i+1)}$). We now explain how we design $\widetilde{\mathcal{L}}$ and then provide some intuition. Let $\widehat{P}$ and $\widehat{c}$ be the current empirical transition probabilities and costs, and let $n(s, a)$ be the current number of visits to state-action pair (s, a) (and $n^+(s, a) = \max\{n(s, a), 1\}$). We first define transition probabilities $\widetilde{P}$ that are slightly skewed towards the goal w.r.t. $\widehat{P}$, as follows

$$\widetilde{P}_{s,a,s'} \triangleq \frac{n(s,a)}{n(s,a)+1} \widehat{P}_{s,a,s'} + \frac{\mathbb{I}[s' = g]}{n(s,a)+1}. \tag{5.4}$$

Given the estimate B, specific positive constants c_1, c_2, c_3, c_4 and a state-action dependent logarithmic term $\iota_{s,a}$, we then define the exploration bonus function, for any state-action pair

Algorithm 5.1: Algorithm EB-SSP

Input: $\mathcal{S},\ s_0 \in \mathcal{S},\ g \notin \mathcal{S},\ \mathcal{A},\ \delta$.
Input: an estimate B guaranteeing $B \geq \max\{B_\star, 1\}$ (see Section 5.5 and Section D.7 if not available).
Optional input: cost perturbation $\eta \in [0, 1]$.
Specify: Trigger set $\mathcal{N} \leftarrow \{2^{j-1} \ : \ j = 1, 2, \ldots\}$. Constants $c_1 = 6,\ c_2 = 36,\ c_3 = 2\sqrt{2},\ c_4 = 2\sqrt{2}$.
For $(s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}'$, set $N(s, a) \leftarrow 0;\ n(s, a) \leftarrow 0;\ N(s, a, s') \leftarrow 0;\ \widehat{P}_{s,a,s'} \leftarrow 0;$ $\theta(s, a) \leftarrow 0;\ \widehat{c}(s, a) \leftarrow 0;\ Q(s, a) \leftarrow 0;\ V(s) \leftarrow 0$.
Set initial time step $t \leftarrow 1$ and trigger index $j \leftarrow 0$.
for episode $k = 1, 2, \ldots$ **do**
  Set $s_t \leftarrow s_0$
  while $s_t \neq g$ **do**
    Take action $a_t = \arg\min_{a \in \mathcal{A}} Q(s_t, a)$, incur cost c_t and observe next state $s_{t+1} \sim P(\cdot|s_t, a_t)$.
    Set $(s, a, s', c) \leftarrow (s_t, a_t, s_{t+1}, \max\{c_t, \eta\})$ and $t \leftarrow t + 1$.
    Set $N(s, a) \leftarrow N(s, a) + 1$, $\theta(s, a) \leftarrow \theta(s, a) + c$, $N(s, a, s') \leftarrow N(s, a, s') + 1$.
    if $N(s, a) \in \mathcal{N}$ **then**
      \\ *Update triggered:* VISGO *procedure.*
      Set $\widehat{c}(s, a) \leftarrow \mathbb{I}[N(s, a) \geq 2]\frac{2\theta(s,a)}{N(s,a)} + \mathbb{I}[N(s, a) = 1]\theta(s, a)$ and $\theta(s, a) \leftarrow 0$.
      For $s' \in \mathcal{S}'$, set $\widehat{P}_{s,a,s'} \leftarrow N(s, a, s')/N(s, a)$, $n(s, a) \leftarrow N(s, a)$, and $\widetilde{P}_{s,a,s'}$ as in Equation (5.4).
      Set $j \leftarrow j + 1$, $\varepsilon_{\text{VI}} \leftarrow 2^{-j}/(SA)$ and $i \leftarrow 0$, $V^{(0)} \leftarrow 0$, $V^{(-1)} \leftarrow +\infty$.
      For all $(s, a) \in \mathcal{S} \times \mathcal{A}$, set $n^+(s, a) \leftarrow \max\{n(s, a), 1\}$ and $\iota_{s,a} \leftarrow \ln\left(\frac{12SAS'[n^+(s,a)]^2}{\delta}\right)$.
      while $\|V^{(i)} - V^{(i-1)}\|_\infty > \varepsilon_{\text{VI}}$ **do**
        For all $(s, a) \in \mathcal{S} \times \mathcal{A}$, set

$$b^{(i+1)}(s, a) \leftarrow b(V^{(i)}, s, a), \qquad \text{\\ see Equation (5.5) for bonus expression} \tag{5.1}$$

$$Q^{(i+1)}(s, a) \leftarrow \max\left\{\widehat{c}(s, a) + \widetilde{P}_{s,a}V^{(i)} - b^{(i+1)}(s, a),\ 0\right\}, \tag{5.2}$$

$$V^{(i+1)}(s) \leftarrow \min_a Q^{(i+1)}(s, a). \tag{5.3}$$

        Set $V^{(i+1)}(g) = 0$ and $i \leftarrow i + 1$.
      Set $Q \leftarrow Q^{(i)}$, $V \leftarrow V^{(i)}$.

$(s, a) \in \mathcal{S} \times \mathcal{A}$ and vector $V \in \mathbb{R}^{S'}$ such that $V(g) = 0$, as follows

$$b(V, s, a) \triangleq \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a}, V)\iota_{s,a}}{n^+(s, a)}},\ c_2\frac{B\iota_{s,a}}{n^+(s, a)}\right\} + c_3\sqrt{\frac{\widehat{c}(s, a)\iota_{s,a}}{n^+(s, a)}} + c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s, a)}. \tag{5.5}$$

Note that the last term in Equation (5.5) accounts for the skewing of $\widetilde{P}$ w.r.t. $\widehat{P}$ (see Lemma D.5). Given the transitions $\widetilde{P}$ and exploration bonus b, we are ready to define the operator $\widetilde{\mathcal{L}}$ as

$$\widetilde{\mathcal{L}}V(s) \triangleq \max\left\{\min_{a \in \mathcal{A}}\{\widehat{c}(s, a) + \widetilde{P}_{s,a}V - b(V, s, a)\},\ 0\right\}. \tag{5.6}$$

5.2 Properties of VISGO

We see that the operator $\widetilde{\mathcal{L}}$ promotes optimism in two different ways:

(i) On the empirical cost function $\widehat{c}$, via the bonus b in Equation (5.5) that intuitively lowers the costs to $\widehat{c} - b$;

(ii) On the empirical transition function $\widehat{P}$, via the transitions $\widetilde{P}$ in Equation (5.4) that slightly bias $\widehat{P}$ with the addition of a non-zero probability of reaching the goal from *every* state-action pair.

While the first feature **(i)** is standard in finite-horizon approaches, the second **(ii)** is SSP-specific, and is required to cope with the fact that the empirical model $\widehat{P}$ may *not* admit any proper policy, meaning that executing value iteration for SSP on $\widehat{P}$ may diverge. Our simple transition skewing actually guarantees that *all* policies are proper in $\widetilde{P}$, for any fixed and bounded cost function.[2] By decaying the extra goal-reaching probability inversely with $n(s,a)$, we can tightly control the gap between $\widetilde{P}$ and $\widehat{P}$ and ensure that it only accounts for a lower-order regret term, cf. last term of Equation (5.5).

Equipped with these two sources of optimism, as long as $B \geq B_\star$, we are able to prove that a VISGO procedure verifies the following two key properties:

(1) ***Optimism:*** VISGO outputs an optimistic estimator of the optimal Q-function at each iteration step, i.e., $Q^{(i)}(s,a) \leq Q^\star(s,a), \forall i \geq 0$,

(2) ***Finite-time near-convergence:*** VISGO terminates within a finite number of iteration steps (note that the final iterate $V^{(j)}$ approximates the fixed point of $\widetilde{\mathcal{L}}$ up to an error scaling with ε_{VI}).

To satisfy **(1)**, we derive similarly to MVP (Zhang et al., 2021d) a *monotonicity* property for the operator $\widetilde{\mathcal{L}}$, which is achieved by carefully tuning the constants c_1, c_2, c_3, c_4 in the bonus of Equation (5.5). On the other hand, the requirement **(2)** is SSP-specific, since it is not needed in finite-horizon where value iteration requires exactly H backward induction steps. *Without* bonuses, the design of $\widetilde{P}$ would have directly entailed that $\widetilde{\mathcal{L}}$ is contractive and convergent (Bertsekas, 1995). However, our variance-aware exploration bonuses introduce a subtle correlation between value iterates, i.e., b depends on V in Equation (5.5), which leads to a cost function that varies across iterates. By directly analyzing $\widetilde{\mathcal{L}}$, we establish that it is contractive with modulus $\rho \triangleq 1 - \nu < 1$, where $\nu \triangleq \min_{s,a} \widetilde{P}_{s,a,g} > 0$. This *contraction* property guarantees a polynomially bounded number of iterations before terminating, i.e., **(2)**.

[2] In fact this transition skewing implies that an SSP problem defined on $\widetilde{P}$ is equivalent to a discounted RL problem, with a varying state-action dependent discount factor. Also note that for different albeit mildly related purposes, a perturbation trick is sometimes used in regret minimization for average-reward MDPs (e.g., Fruit et al., 2018b; Qian et al., 2019), where a non-zero probability of reaching an arbitrary state at each state-action is added to guarantee that all policies are unichain and that value iteration variants nearly converge in finite-time.

5.3 Regret Analysis

Besides ensuring the computational efficiency of EB-SSP, the properties of VISGO lay the foundations for our regret analysis (Section D.3) to yield the following general guarantee.

Theorem 5.1. *Assume that $B \geq \max\{B_\star, 1\}$ and that the conditions of Proposition 2.11 hold. Then with probability at least $1-\delta$ the regret of* EB-SSP (*Algorithm 5.1 with $\eta = 0$*) *can be bounded by*

$$R_K = O\left(\sqrt{(B_\star^2 + B_\star)SAK}\log\left(\frac{\max\{B_\star,1\}SAT}{\delta}\right) + BS^2A\log^2\left(\frac{\max\{B_\star,1\}SAT}{\delta}\right)\right),$$

with T the accumulated time within the K episodes.

Proof idea. We decompose the regret into three parts: X_1 (error on the optimistic V-values), X_2 (Bellman error) and X_3 (cost estimation error), and among them the major part is X_2. Later, X_1 and X_2 introduce the intermediate quantities X_4 (variance of the optimistic V-values) and X_5 (variance of the differences $V^\star - V$), which are bounded using the recursion technique generalized from Zhang et al. (2021d), where we normalize the values by $1/B_\star$ to avoid an exponential blow-up in the recursions. At a high-level, the key idea is to calculate errors of different orders, $F(1), F(2), \ldots, F(d), \ldots$ (see Lemma D.17 and D.18), and recursively bound $F(i)$'s variance by a sublinear function of $F(i+1)$. Throughout the proof, we bound quantities by solving inequalities that contain the unknown quantities on both sides, such as $X_3 \leq \widetilde{O}(\sqrt{X_3 + C_K})$ or $X_2 \leq \widetilde{O}(\sqrt{X_2 + C_K})$, where the random variable C_K denotes the cumulative cost over the K episodes. Indeed, the analysis at each time step t brings out the instantaneous cost c_t and it is important to combine them so that we can make C_K appear explicitly. Ultimately, we obtain a regret bound scaling as $R_K = \widetilde{O}((\sqrt{B_\star}+1)\sqrt{SAC_K})$. Since the regret in SSP is defined as $R_K = C_K - KV^\star(s_0)$, we obtain a quadratic inequality in C_K, which we solve to eliminate the dependence on the random variable C_K and to get the $\widetilde{O}(\sqrt{(B_\star^2 + B_\star)SAK})$ regret bound. □

Theorem 5.1 is an intermediate result for the regret of EB-SSP, as it depends on the *random and possibly unbounded* total number of steps T executed over K episodes, it requires the possibly restrictive second condition of Proposition 2.11, and it relies on the parameter B being properly tuned. Nonetheless, it already displays interesting properties: **1)** The dependence on T is limited to logarithmic terms; **2)** The parameter B only affects the lower order term, while the main order term naturally scales with the exact range $B_\star$; **3)** Up to dependence on T, the main order term displays minimax optimal dependencies on $B_\star$, S, A, and K.

Throughout the rest of the chapter, we consider for ease of exposition that $B_\star \geq 1$.[3] For simplicity, when tuning the cost perturbations later, we assume as in prior works (e.g., Rosenberg et al., 2020; Chen et al., 2021c; Chen and Luo, 2021) that the total number of episodes K is known to the agent (this knowledge can be eliminated with the standard doubling trick).

5.4 Regret Bounds for Known $B_\star$

First we assume that $B = B_\star$ (i.e., the agent has prior knowledge of $B_\star$) and we instantiate the regret achieved by EB-SSP under various conditions on the SSP model.

5.4.1 Positive Costs

We first focus on the case of positive costs.

Assumption 5.2. *All costs are lower bounded by a constant $c_{\min} > 0$ which is unknown to the agent.*

Assumption 5.2 guarantees that the conditions of Proposition 2.11 hold. Moreover, denoting by C the cumulative cost over K episodes, the total time satisfies $T \leq C/c_{\min}$. By simplifying the bound of Theorem 5.1 as $C \leq B_\star K + R_K \leq O(B_\star S^2 AK \cdot \sqrt{B_\star TSA/\delta})$, we loosely obtain that $T = O(B_\star^3 S^5 A^3 K^2/(c_{\min}^2 \delta))$.

Corollary 5.3. *Under Assumption 5.2, running* EB-SSP (*Algorithm 5.1*) *with $B = B_\star$ and $\eta = 0$ gives the following regret bound with probability at least $1 - \delta$*

$$R_K = O\left(B_\star \sqrt{SAK} \log\left(\frac{KB_\star SA}{c_{\min}\delta}\right) + B_\star S^2 A \log^2\left(\frac{KB_\star SA}{c_{\min}\delta}\right)\right).$$

The bound of Corollary 5.3 only depends polynomially on $K, S, A, B_\star$. We note that $T_\star \leq B_\star/c_{\min}$ and that this upper bound only appears in the logarithms. Under positive costs, the regret of EB-SSP is thus (nearly) **minimax** and **horizon-free**. Furthermore, in Section D.1 we introduce an alternative assumption on the SSP problem (which is weaker than Assumption 5.2) that considers that there are no almost-sure zero-cost cycles. In this case also, the regret of EB-SSP is (nearly) minimax and horizon-free.

[3] Otherwise, all later bounds hold by replacing $B_\star$ with $\max\{B_\star, 1\}$, except for the $B_\star$ factor in the leading term that becomes $\sqrt{B_\star}$. This matches the lower bound of $\Omega(\sqrt{B_\star SAK})$ for $B_\star < 1$ (see Proposition 3.5).

5.4.2 General Costs and $T_\star$ Unknown

We now handle the case of non-negative costs, with no assumption other than Assumption 2.7. We use a cost perturbation argument to generalize the results from positive to general costs (as done in Chapter 4). As reviewed in Section 2.4, this circumvents the second condition of Proposition 2.11 (which holds in the cost-perturbed MDP) and target the optimal proper policy in the original MDP up to a bias scaling with the cost perturbation. Indeed, running EB-SSP with costs $c_\eta(s,a) \leftarrow \max\{c(s,a), \eta\}$ for $\eta \in (0,1]$ gives the bound of Corollary 5.3 with $c_{\min} \leftarrow \eta$, $B_\star \leftarrow B_\star + \eta T_\star$ and an additive bias of $\eta T_\star K$. We then pick η to balance these terms.

Corollary 5.4. *Let $L \triangleq \log\left(KT_\star SA\delta^{-1}\right)$. Running* EB-SSP *(Algorithm 5.1) with $B = B_\star$ and $\eta = K^{-n}$ for **any** choice of constant $n > 1$ gives the following regret bound with probability at least $1-\delta$*

$$R_K = O\Big(nB_\star\sqrt{SAK}L \;+\; \frac{T_\star}{K^{n-1}} + \frac{nT_\star\sqrt{SA}L}{K^{n-1/2}} \;+\; n^2 B_\star S^2 A L^2\Big).$$

This bound can be decomposed as (i) a $\sqrt{K}$ leading term and (ii) an additive term that depends on $T_\star$ and vanishes as $K \to +\infty$ (we omit the last term that does not depend polynomially on either K or $T_\star$). Note that the second term (ii) can be made as small as possible by increasing the choice of exponent n in the cost perturbation, at the cost of the multiplicative constant n in (i). Equipped only with Assumption 2.7, the regret of EB-SSP is thus (nearly) **minimax**, and it may be dubbed as *horizon-vanishing* when K is given in advance, insofar as it contains an additive term that depends on $T_\star$ and that becomes negligible for large values of K (if K is unknown in advance, the application of the doubling trick yields an additive term (ii) scaling as $T_\star$). We now show that the trade-off between (i) and (ii) can be resolved with loose knowledge of $T_\star$ and leads to a horizon-free bound.

5.4.3 General Costs and Order-Accurate Estimate of $T_\star$ Available

We now consider that an order-accurate estimate of $T_\star$ is available. It may be a constant lower-bound approximation away from $T_\star$, or a polynomial upper-bound approximation away from $T_\star$.

Assumption 5.5. *The agent has prior knowledge of a quantity $\overline{T}_\star$ that verifies $\frac{T_\star}{\upsilon} \leq \overline{T}_\star \leq \lambda T_\star^\zeta$ for some unknown constants $\upsilon, \lambda, \zeta \geq 1$. (Note that $\upsilon = \lambda = \zeta = 1$ when $T_\star$ is known.)*

We now tune the cost perturbation η using $\overline{T}_\star$. Specifically, selecting $\eta \triangleq (\overline{T}_\star K)^{-1}$ ensures that the bias satisfies $\eta T_\star K \leq \upsilon = O(1)$. We thus obtain the following guarantee (see Section D.2 for the explicit dependencies on the *constant* terms υ, λ, ζ which only appear as multiplicative and additive factors).

Corollary 5.6. *Under Assumption 5.5, running* EB-SSP (*Algorithm 5.1*) *with* $B = B_\star$ *and* $\eta = (\overline{T}_\star K)^{-1}$ *gives the following regret bound with probability at least* $1 - \delta$

$$R_K = O\left(B_\star \sqrt{SAK} \log\left(\frac{KT_\star SA}{\delta}\right) + B_\star S^2 A \log^2\left(\frac{KT_\star SA}{\delta}\right)\right).$$

This bound depends polynomially on $K, S, A, B_\star$, and only logarithmically on $T_\star$. Thus under general costs with an order-accurate estimate of $T_\star$, EB-SSP's regret is (nearly) **minimax** and **horizon-free**.

5.5 Regret Bounds for Unknown $B_\star$ with Parameter-Free EB-SSP

In this section, we introduce a parameter-free version of EB-SSP that bypasses the requirement of $B \geq B_\star$ (line 2 of Algorithm 5.1). Note that the challenge of not knowing the range of the optimal value function does not appear in finite-horizon MDPs, where the bound H (or 1 for Zhang et al., 2021d) is assumed to be known to the agent. In SSP, if the agent does not have a valid estimate $B \geq B_\star$, then it may design an under-specified exploration bonus which cannot guarantee optimism. The case of unknown $B_\star$ is non-trivial: it appears impossible to properly estimate $B_\star$ (since some states may never be visited) and it is unclear how a standard doubling trick may be used.

Parameter-free EB-SSP initializes a proxy $\widetilde{B} = 1$ and increases it over the learning interaction according to a carefully defined schedule. We need to ensure that the proxy $\widetilde{B}$ does not remain below $B_\star$ for too long, since in this case, the regret may keep growing linearly. Thus, our *first condition* to increase $\widetilde{B}$ is whenever a new episode k begins, specifically we set $\widetilde{B} \leftarrow \max\{\widetilde{B}, \sqrt{k}/(S^{3/2}A^{1/2})\}$, which ensures that $\widetilde{B} \geq B_\star$ for large enough episodes. However, this is not enough: indeed notice that when $\widetilde{B} < B_\star$, the agent may never reach the goal and thus get *stuck* in the episode, so we cannot exclusively rely on the end of an episode as a trigger for increasing $\widetilde{B}$. Our *second condition* to increase $\widetilde{B}$ is to set $\widetilde{B} \leftarrow 2\widetilde{B}$ whenever the cumulative cost exceeds a carefully defined threshold (that depends on $\widetilde{B}$, S, A, δ and the current episode and time indexes k and t, which are all computable quantities). Since the regret is upper bounded by the cumulative cost, this second condition prevents the learner from accumulating too large regret when $\widetilde{B} < B_\star$. Finally, we introduce a *third condition* to increase $\widetilde{B}$ in order to ensure the

computational efficiency, since VISGO may diverge when $\widetilde{B} < B_\star$ (specifically, we track the range of the value $V^{(i)}$ at each VISGO iteration i and if $\|V^{(i)}\|_\infty > \widetilde{B}$, then we terminate VISGO and increase $\widetilde{B} \leftarrow 2\widetilde{B}$). At a high-level, the analysis of the scheme proceeds as follows: we bound the regret by the cumulative cost when $\widetilde{B} < B_\star$ (first regime), and by the regret bound of Theorem 5.1 when $\widetilde{B} \geq B_\star$ (second regime). Note that this two-regime decomposition is only implicit (i.e., at the level of analysis), since the agent is unable to know in which regime it is (since $B_\star$ is unknown). The full pseudo-code and analysis of parameter-free EB-SSP is deferred to Section D.7.

Theorem 5.7 (Extension of Theorem 5.1 to unknown $B_\star$)**.** *Assume the conditions of Proposition 2.11 hold. Then with probability at least $1-\delta$ the regret of parameter-free* EB-SSP (*Algorithm D.1, Section D.7*) *can be bounded by*

$$R_K = O\left(R_K^\star \log\left(\frac{B_\star SAT}{\delta}\right) + B_\star^3 S^3 A \log^3\left(\frac{B_\star SAT}{\delta}\right)\right),$$

where T is the cumulative time within the K episodes and $R_K^\star$ bounds the regret after K episodes of EB-SSP *in the case of known $B_\star$ (i.e., the bound of Theorem 5.1 with $B = B_\star$).*

Theorem 5.7 implies that we can remove the condition of $B \geq \max\{B_\star, 1\}$ in Theorem 5.1, i.e., we make the statement **parameter-free**. Hence, *all* the regret bounds from Section 5.4 in the case of known $B_\star$ (i.e., Corollaries D.2, 5.3, 5.4 and 5.6) still hold up to additional logarithmic and lower-order terms when $B_\star$ is unknown.

5.6 Discussion and Bibliographical Remarks

In this chapter, we presented EB-SSP, a new algorithm for online SSP. It introduces a value-optimistic scheme to efficiently compute optimistic policies for SSP, by both perturbing the empirical costs with an exploration bonus and slightly biasing the empirical transitions towards reaching the goal from *each* state-action pair with positive probability. Under these biased transitions, *all* policies are in fact proper, and the bias is decayed over time in a way that it only contributes to a lower-order regret term. The guarantees of EB-SSP are significant in the following ways:

1. EB-SSP is the first algorithm to achieve the **minimax** regret rate of $\widetilde{O}(B_\star\sqrt{SAK})$ while simultaneously being **parameter-free**: it does not require to know nor estimate $T_\star$, and it is able to bypass the knowledge of $B_\star$ at the cost of only logarithmic and lower-order contributions to the regret. In fact, this result is the first to show that it is possible to devise an adaptive exploration bonus strategy in an RL setting where no prior knowledge of the

"optimal range" is available (this was an open question raised by Qian et al., 2019 whose exploration-bonus-based approach in average-reward MDPs requires prior knowledge of an upper bound on the optimal bias span).

2. EB-SSP is the first algorithm to achieve **horizon-free** regret for SSP in various cases: i) positive costs, ii) no almost-sure zero-cost cycles, and iii) the general cost case when an order-accurate estimate of $T_\star$ is available (i.e., a value $\overline{T}_\star$ such that $T_\star/\upsilon \leq \overline{T}_\star \leq \lambda T_\star^\zeta$ for some unknown constants $\upsilon, \lambda, \zeta \geq 1$ is available). This property is especially relevant if $T_\star$ is much larger than $B_\star$, which can occur in SSP models with very small instantaneous costs. Moreover, EB-SSP achieves its horizon-free guarantees while maintaining the minimax rate. For instance, under general costs when relying on $T_\star$ and $B_\star$, its regret is $\widetilde{O}(B_\star\sqrt{SAK} + B_\star S^2 A)$.[4] Our result is the first to show that the concept of horizon-free regret can be extended beyond finite-horizon MDPs (Wang et al., 2020a; Zhang et al., 2021d; Zhang et al., 2021e). In fact, it is perhaps even more meaningful in the goal-oriented setting, by virtue of the already present distinction between "optimal length" ($T_\star$) and "optimal return" ($B_\star$). Indeed we do not make any extra assumption on the SSP problem, as opposed to the finite-horizon set-up which requires the assumption of different episode length (H) and episode return of *any* trajectory (at most 1) so as to uncover horizon-free properties.

Concurrently to our work, Cohen et al. (2021) proposed an algorithm for online SSP that is interestingly based on a very different algorithmic idea to ours. Whereas we operate at the level of the non-truncated SSP model, they rely on a black-box reduction from SSP to finite-horizon MDPs. Specifically, their approach successively tackles finite-horizon problems with horizon set to $H = \Omega(T_\star)$ and costs augmented by a terminal cost set to $c_H(s) = \Omega(B_\star \mathbb{I}(s \neq g))$, where g denotes the goal state. This finite-horizon construction guarantees that its optimal policy has a similar value function to the optimal policy in the original SSP instance up to a lower-order bias. Their algorithm comes with a regret bound of $O(B_\star\sqrt{SAK}L + T_\star^4 S^2 A L^5)$, with $L = \log(KT_\star SA\delta^{-1})$ (with probability at least $1-\delta$). It achieves a nearly minimax-optimal rate, however it relies on both $T_\star$ and $B_\star$ prior knowledge to tune the horizon and terminal cost in the reduction, respectively.[5][6] In addition, their bound is not horizon-free: indeed, even in the case of known $T_\star$ and $B_\star$, which implies that the conditions of Corollary 5.6 hold, the bound of

[4] We conjecture the optimal problem-independent regret in SSP to be $\widetilde{O}(B_\star\sqrt{SAK} + B_\star SA)$ (by analogy with the conjecture of Menard et al., 2021 for finite-horizon MDPs), which shows the tightness of our bound up to an S lower-order factor.

[5] As mentioned by Cohen et al. (2021, Section 3.1), if $B_\star$ is unknown it may be estimated on the fly using the SSP regret minimization algorithm of Rosenberg et al. (2020) as initial subroutine, see Remark D.29 for more discussion and comparison with our scheme for unknown $B_\star$ for parameter-free EB-SSP.

[6] As mentioned by Cohen et al. (2021, Remark 2), in the case of positive costs lower bounded by $c_{\min} > 0$, their knowledge of $T_\star$ can be bypassed by replacing it with the upper bound $T_\star \leq B_\star/c_{\min}$. However, when generalizing from the $c_{\min}$ case to general costs with a perturbation argument, their regret guarantee worsens from $\widetilde{O}(\sqrt{K} + c_{\min}^{-4})$ to $\widetilde{O}(K^{4/5})$, because of the poor additive dependence on $c_{\min}^{-1}$.

Corollary 5.6 is strictly tighter, since it always holds that $B_\star \leq T_\star$ and the gap between the two may be arbitrarily large (see e.g., Section B.2), especially when some instantaneous costs are very small.

While EB-SSP is computationally efficient (see Section D.6 for details), its poly(K) complexity is a limitation shared by all existing parameter-free algorithms in SSP. On the other hand, the algorithm of Cohen et al. (2021) can obtain a $\log(K)$ computational complexity but only with $T_\star$ prior knowledge: without it, using the upper bound $T_\star \leq B_\star / c_{\min}$, where $c_{\min}^{-1}$ becomes poly(K) when applying the cost perturbation trick, also leads to poly(K) complexity. It is an interesting open question whether it is possible in SSP to have $\log(K)$ computational complexity while staying parameter-free.

All the aforementioned algorithms for online SSP are *model-based*. Chen et al. (2021a) later proposed the first *model-free* algorithm, which is minimax optimal under strictly positive costs. Their analysis relies on a technique called implicit finite-horizon approximation, which approximates the SSP model by a finite-horizon counterpart only in the analysis without explicit implementation. Using this template, they also develop a model-based algorithm, which performs one-step planning (instead of full planning) and exactly matches the regret guarantees of EB-SSP.

While the above algorithms are based on the principle of Optimism in the Face of Uncertainty (OFU), Jafarnia-Jahromi et al. (2021) later developed the first no-regret algorithm for online SSP that is based on Posterior Sampling (also known as Thompson Sampling). Meanwhile, Chen et al. (2022) recently initiated the study of policy optimization for the SSP problem in a range of settings (including stochastic and adversarial environments under full information or bandit feedback). They propose an approximation scheme of SSP that they call Stacked Discounted Approximation (see their discussion in Chen et al., 2022, Section 3), which is interestingly a hybrid of a finite-horizon MDP approximation (Chen and Luo, 2021; Cohen et al., 2021) and a discounted MDP approximation (that we adopt in EB-SSP, see Remark D.7). A take-away message from Chapter 5 and these related works on online SSP is that a pertinent solution to tackle SSP is to consider either implicit or explicit approximations by other MDP models with more convenient analytical and/or computational properties, namely finite-horizon MDPs, discounted MDPs or a combination thereof.

Finally, all the aforementioned algorithms for online SSP are for the tabular setting and their associated regret bounds (unavoidably) scale with S and A. Recently, Vial et al. (2021) and Min et al. (2021) developed the first algorithms for online SSP with *linear function approximation*, respectively linear SSP (where the transition kernel and cost vector are linear in known d-dimensional feature vectors) and linear-mixture SSP (where the transition kernel is parameterized by a linear function over known feature mappings defined on the triplet of state, action, and next state). The regret bounds were then improved by Chen et al. (2021b), who

also derived the first logarithmic instance-dependent expected regret bounds for SSP. Note that these three works still consider that the MDP has a *finite number of states* S. This is to be expected given the way a goal is currently modeled (at the granular level of states), independently of how large the state space is, where it may be very hard to visit specific states. Learning in SSP beyond a finite state space is an interesting direction of future investigation.

Part II

Unsupervised Reinforcement Learning: Learning to Set Your Own Goals

Overview of Part II:

- **Open-ended research question:** In the absence of any reward supervision, how to learn to autonomously and efficiently solve a wide variety of tasks?
- **Key contribution:** We instigate a thorough and formal analysis of the general-purpose principle of SYOG — "Set Your Own Goals" — for various learning objectives.
- **Relevance:** SYOG is a popular heuristic that has already yielded promising empirical results in unsupervised deep RL.

Chapter 6

Overview of Unsupervised RL & SYOG (Set Your Own Goals)

Beyond training our RL agent to solve only one goal-oriented task as in Part I, we now aspire in Part II that it learns to autonomously solve a wide variety of tasks, in the absence of any reward/cost/goal supervision. In this chapter, we review some existing approaches for *unsupervised* RL, both on the empirical and theoretical sides, and we present a general-purpose principle dubbed SYOG — for "**S**et **Y**our **O**wn **G**oals" — which suggests the agent to learn the ability to intrinsically select and reach its own goal states. SYOG has already found wide empirical success in unsupervised deep RL methods. The main contribution of Part II is to instigate a thorough and formal analysis of the SYOG principle in various settings, each with its specific learning objective and set of technical challenges.

Contents

Throughout Part II, we use the term *unsupervised RL* (URL) to describe the setting where the environment does not provide any supervision signal, i.e., the agent is given no reward/cost function nor specific goal to reach. Formally, the agent interacts with a *reward-free* MDP $M \triangleq \langle \mathcal{S}, \mathcal{A}, P, s_0 \rangle$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ denotes the transition probabilities and $s_0 \in \mathcal{S}$ is the initial state. In this case, the conventional RL objective of maximizing cumulative reward cannot be optimized, and must be replaced by alternative objectives intrinsically set by the learning agent.

6.1 High-level Motivations behind URL

Absent reward signal. A first motivation to investigate unsupervised RL is quite "AI-ish", almost existential in nature: imagine a robot deployed in an unknown environment (e.g., a far-away, unexplored planet), with no explicit supervision signal from the environment nor any human intervention. This gives rise to the open-ended question: how should it explore its environment, i.e., driven by which intrinsic objective(s)?

Sparse reward signal. In many RL applications, a pre-specified reward function is available yet it is rarely informative. A sparse reward task is typically characterized by a meagre amount of states in the state space that return a feedback signal. A typical situation is when an agent has to reach a goal and only receives a positive reward signal when it enters the goal state (see e.g., Koenig and Simmons, 1996, Section 4.1 and references therein). This can render the optimization process quite ineffective in converging towards an optimal behavior, and can motivate to augment the sparse extrinsic reward with some carefully constructed *intrinsic rewards* to facilitate the discovery of rewarding states.

Multiple/Varying reward signal. In some settings, there are multiple reward functions of interest, e.g., in constrained RL formulations (Altman, 1999; Achiam et al., 2017; Tessler et al., 2019). To strike a balance between the multiple (possibly conflicting) objectives, reward functions are often iteratively engineered to encourage desired behavior via trial and error. In such cases, repeatedly invoking the same RL algorithm with different reward functions can be quite sample inefficient. In the batch RL setting (Bertsekas and Tsitsiklis, 1995), data collection and planning are explicitly separated, which highlights the potential benefit of performing an initial phase of task-agnostic learning. In hierarchical and multi-task RL (Dietterich, 2000; Tessler et al., 2017; Oh et al., 2017), the agent aims at simultaneously learning a set of skills. In the robotic navigation problem (Rimon and Koditschek, 1992; Kretzschmar et al., 2016), the agent needs to navigate to not only one goal state, but a set of states in the environment.

6.2 Short Review of Empirical Studies of URL

The works of Schmidhuber (1991), Chentanez et al. (2005), Singh et al. (2009), Singh et al. (2010), Oudeyer and Kaplan (2009), and Baranes and Oudeyer (2010) (among others) established computational theories of intrinsic reward signals (and how it might help with downstream learning of tasks). In the Deep Reinforcement Learning (DRL) community, there has been an increasing interest in designing algorithms that can learn without the supervision of a well-designed reward function. Some approaches design intrinsic rewards to drive the learning process, for instance via state visitation counts (Bellemare et al., 2016; Tang et al., 2017), novelty or prediction errors (Houthooft et al., 2016; Pathak et al., 2017; Azar et al., 2019; Badia et al., 2020). Other recent methods perform information-theoretic skill discovery to learn a set of diverse and task-agnostic behaviors (Gregor et al., 2016; Eysenbach et al., 2019; Sharma et al., 2020; Campos et al., 2020; Kamienny et al., 2022). Alternatively, goal-conditioned policies learned by carefully designing the sequence of goals during the learning process are often used to solve sparse reward problems (Ecoffet et al., 2020) and a variety of goal-reaching tasks (Florensa et al., 2018; Colas et al., 2019; Warde-Farley et al., 2019; Pong et al., 2020), as further discussed in Section 6.4.

6.3 Short Review of Theoretical Studies of URL

To the best of our knowledge, Lim and Auer (2012) are the first to propose a formal performance measure accompanied with a theoretical analysis of an unsupervised RL agent. They focus on the restricted class of *incremental autonomous exploration*, where the objective is to identify and learn to reliably reach all the states that are incrementally reliably reachable from a reference starting state s_0 to which the agent can reset (at a high level, the incrementally reliably reachable states are those that admit some unknown order $s_0, s_1, \dots$ such that s_i is reliably reachable by a policy defined only on $s_0, \dots, s_{i-1}$). We defer to Chapter 9 the detailed description of this setting on which we will build.

More recently and contemporarily to this thesis, a growing line of research has focused on analyzing some provably efficient unsupervised RL objectives. We can broadly separate them in the two following classes:

- **"One-shot" unsupervised exploration** (Section 6.3.1): Given a specific desiderata (e.g., behavior or prediction) specified before exploration, the agent should be able to approximate it accurately. Examples include to mimic the behavior of a certain policy $\pi^\dagger$ (measured by some function $F : \Pi \to \mathbb{R}$), or to predict some unknown functions $\{F(s,a)\}_{s,a}$ of state-action pairs whose visits provide possibly noisy observations (e.g., the transition probability $P(\cdot|s,a)$).

- **"End-to-end" finite-horizon unsupervised exploration** (Section 6.3.2): After a reward-free exploration phase, the agent should be able to compute near-optimal policies for some set of possible reward functions revealed only during the subsequent planning phase (either a finite number of them, or any possibly adversarial reward function).

6.3.1 "One-shot" unsupervised exploration

This class of objective can be parametrized in state-action stationary distributions $\lambda \in \Lambda$ and cast as a convex optimization problem

$$\min_{\lambda \in \Lambda} F(\lambda), \tag{6.1}$$

for some convex function $F : \Lambda \to \mathbb{R}$, where the set Λ is defined as

$$\Lambda \triangleq \Big\{ \lambda \in \Delta(\mathcal{S} \times \mathcal{A}) : \ \forall s \in \mathcal{S}, \sum_{b \in \mathcal{A}} \lambda(s, b) = \sum_{s' \in \mathcal{S}, a \in \mathcal{A}} P(s|s', a) \lambda(s', a) \Big\}.$$

Note that this parametrization is used in the dual formulation of reward-based MDP (Puterman, 2014, Section 8). Recall that Λ is a convex set, and that any $\lambda \in \Lambda$ with $\sum_{a \in \mathcal{A}} \lambda(s, a) > 0$ for all $s \in \mathcal{S}$ induces a stationary policy $\pi_\lambda : \mathcal{S} \to \Delta(\mathcal{A})$ defined as

$$\pi_\lambda(a|s) \triangleq \frac{\lambda(s, a)}{\sum_{a \in \mathcal{A}} \lambda(s, a)}.$$

Contemporarily to the works of Hazan et al. (2019) and Cheung (2019), in Tarbouriech and Lazaric (2019) followed by Tarbouriech et al. (2020c) we investigated how to sequentially optimize problems of the type Equation (6.1) by generating a sequence of intrinsic reward signals $r_t(s) \triangleq -\nabla F(\widehat{\lambda}_t)$, where $\widehat{\lambda}_t$ denotes the state-action empirical frequency at time t, i.e., the normalized number of times that action a has been executed in state s after t time steps. The approximation loss, or *regret*, can be defined as $F(\widehat{\lambda}_t) - F(\lambda^\star)$, where $\lambda^\star \in \arg\min_{\lambda \in \Lambda} F(\lambda)$. If F is convex, Lipschitz-continuous and smooth, the analysis can build on the Frank-Wolfe optimization principle. For the interest of conciseness we do not delve into the technical details but rather give an overview of the guarantees. In particular, in the communicating MDP setting with diameter D (recall that D measures the longest shortest path between any two states, see Jaksch et al., 2010), the algorithm of Cheung (2019) comes with the following guarantee

$$F(\widehat{\lambda}_t) - F(\lambda^\star) = \begin{cases} \widetilde{O}\left(t^{-1/2}\right) & \text{if } F \text{ is smooth,} \\ \widetilde{O}\left(t^{-1/3}\right) & \text{if } F \text{ is non-smooth,} \end{cases}$$

where the $\widetilde{O}$ hides polynomial dependencies on MDP-dependent quantities (e.g., S, A, D) and f-dependent quantities (e.g., Lipschitz-continuity constant, smoothness constant).

As illustrated below, there are two main categories of desiderata that can be addressed with this framework: to visit the state(-action) space according to a prescribed behavior (e.g., "MaxEnt"), or to make accurate predictions about state-action outcomes (e.g., "ModEst").

Example ①: Mimic a target distribution. Here the agent receives as input a target state-action distribution $\rho \in \Delta(\mathcal{S} \times \mathcal{A})$, and its objective is to minimize the following mean squared error

$$F(\lambda) \triangleq \frac{1}{SA} \sum_{s,a} \Big(\rho(s,a) - \lambda(s,a)\Big)^2.$$

We note that F is convex, Lipschitz continuous and smooth. For instance, the algorithm of Cheung (2019) in communicating MDPs yields

$$F(\widehat{\lambda}_T) - F(\lambda^\star) = \widetilde{O}\left(\frac{DSA^{1/2}}{T^{1/2}}\right).$$

Example ②: Maximize the state entropy (MaxEnt). Here the agent seeks to learn a policy that induces a distribution over the state space that is as uniform as possible, which can be measured in an entropic sense (Hazan et al., 2019). Let $U \triangleq \{\mu \in \Delta(\mathcal{S}) : \exists \lambda \in \Lambda, \forall s \in \mathcal{S}, \mu(s) = \sum_{a \in \mathcal{A}} \lambda(s,a)\}$, then we can define the function to optimize $F : U \to \mathbb{R}$ as well as an auxiliary function $H_\eta : U \to \mathbb{R}$ for $\eta > 0$ as follows

$$F(\mu) \triangleq \sum_s \mu(s) \log\big(\mu(s)\big), \qquad H_\eta(\mu) \triangleq \sum_s \mu(s) \log\big(\mu(s) + \eta\big).$$

While F is only convex in μ, H_η is convex, Lipschitz continuous and smooth in μ for any $\eta > 0$. The analysis is thus applied to H_η for a carefully selected η such that the bias w.r.t. F is adequately controlled. Let $\mu^\star \in \arg\min_{\mu \in U} F(\mu)$. For instance, the algorithm of Cheung (2019) in communicating MDPs yields

$$F(\widehat{\mu}_T) - F(\mu^\star) = \widetilde{O}\left(\frac{DS^{1/3}}{T^{1/3}} + \frac{DSA^{1/2}}{T^{1/2}}\right).$$

We can also mention that some recent works have empirically studied the MaxEnt problem beyond the tabular case with non-parametric entropy estimation (Mutti et al., 2021; Liu and Abbeel, 2021) or with variations to the entropy objective, such as geometry-awareness (Guo et al., 2021) and Rényi generalization (Zhang et al., 2021a).

Example ③: Accurately estimate the transition model (ModEst). A possible objective to quantify how well the transition dynamics are estimated can be to minimize

$$G_t(\pi) \triangleq \sum_{s,a} \left|\left| \widehat{P}_{\pi,t}(\cdot|s,a) - P(\cdot|s,a) \right|\right|_1,$$

where $\widehat{P}_{\pi,t}$ is the estimate (i.e., empirical average) of the transition dynamics P after t time steps of executing the (possibly non-stationary) policy π. Since directly optimizing the objective function appears highly non-trivial, in Tarbouriech et al. (2020c) we propose to upper bound it with Bernstein's inequality and then reparametrize it in λ, as loosely shown below

$$G_t(\pi) \lesssim B_t(\pi) \triangleq \sum_{s,a} \left(\frac{V(s,a)}{\sqrt{N_{\pi,t}(s,a)}} + \frac{S}{N_{\pi,t}(s,a)} \right) = \frac{1}{\sqrt{t}} \sum_{s,a} \left(\frac{V(s,a)}{\sqrt{\widehat{\lambda}_{\pi,t}(s,a)}} + \frac{1}{\sqrt{t}} \frac{S}{\widehat{\lambda}_{\pi,t}(s,a)} \right),$$

where $N_{\pi,t}(s,a)$ denotes the number of visits to (s,a) after t time steps under policy π, and $V(s,a)$ is a term that depends on the variance of $P(\cdot|s,a)$ which is a priori unknown but can be estimated by an upper confidence bound (Maurer and Pontil, 2009). The objective can thus be cast as optimizing the following function on the Λ space,

$$\min_{\lambda \in \Lambda} F(\lambda) \triangleq \frac{1}{\sqrt{t}} \sum_{s,a} \left(\frac{V(s,a)}{\sqrt{\lambda(s,a)}} + \frac{1}{\sqrt{t}} \frac{S}{\lambda(s,a)} \right).$$

Unfortunately, while F is convex, it has a poorly behaved optimization landscape; in particular it is not Lipschitz continuous. A solution can be to "artificially" make the function well-behaved, by optimizing $\min_{\lambda \in \Lambda_\eta} F(\lambda)$ on a restricted simplex

$$\Lambda_\eta \triangleq \Big\{ \lambda \in \Lambda : \ \forall (s,a) \in \mathcal{S} \times \mathcal{A}, \ \lambda(s,a) \geq \eta \Big\},$$

where $\eta < (SA)^{-1}$ is a small positive constant. Ultimately, we show in Tarbouriech et al. (2020c) that it is possible to obtain a polynomially bounded sample complexity guarantee (i.e., $G_n(\pi) \leq \varepsilon$ for any accuracy level $\varepsilon > 0$), but the analysis requires the strong *ergodicity assumption* as well as the condition $\eta \leq \min_{s,a} \lambda^\star(s,a)$, which implies non-trivial prior knowledge.

Discussion. Consider that we are able to cast our unsupervised objective as minimizing some function $F(\lambda)$ over the space of state-(action) stationary distribution. If F is "optimization-friendly", i.e., convex with bounded gradients, then the Frank-Wolfe-based strategy of feeding the current gradient $-\nabla F(\widehat{\lambda}_t)$ as intrinsic reward yields satisfying results (e.g., example ① and to a lesser extent ②). Unfortunately, many problems do not admit these nice properties (e.g., example ③), which makes such an optimization-based, "first-order" method not ideal. In Chapter 7, we will instead propose a sampling-based, "zero-order" method that will come with

stronger theoretical guarantees. Finally, we point out a limitation of the aforementioned Frank-Wolfe-based approaches, even when F has a well-behaved optimization landscape. Indeed, they may not learn how to effectively reach any state of the environment and thus may not be sufficient to efficiently solve downstream tasks. In other words, there is no theoretical evidence (yet) that they produce *re-usable* policies.

6.3.2 "End-to-end" finite-horizon unsupervised exploration

Another relevant take for theoretical unsupervised RL is the paradigm of Jin et al. (2020) in finite-horizon MDPs with horizon denoted by H. It consists of an *exploration phase* followed by a *planning phase*. In the exploration phase, an agent interacts with the unknown environment without the supervision of reward signals. Afterwards, in the planning phase, without additional environment interaction and only based on its exploration experiences, the agent is required to *compute a near-optimal policy for some revealed reward function*. If the reward function can be designed arbitrarily (including adversarially), the problem is called *reward-free exploration* (RFE) (Jin et al., 2020; Kaufmann et al., 2021; Ménard et al., 2021; Zhang et al., 2021c; Zanette et al., 2020; Wang et al., 2020b; Chen et al., 2021d; Zhang et al., 2021b). If there is only a finite number of possible reward functions that are fixed yet unknown during exploration (i.e., independent of the randomness used in the exploration phase), the problem is called *task-agnostic exploration* (TAE) (Zhang et al., 2020a; Wu et al., 2020; Wu et al., 2021). The agent's performance is measured by the *sample complexity*, i.e., the number of samples that the algorithm needs to collect during the exploration phase in order to complete the planning task near-optimally up a small error $\varepsilon > 0$ (with probability at least $1 - \delta$). The aforementioned works have shown that the minimax sample complexity is

$$\begin{aligned} \widetilde{O}\Big(\mathrm{poly}(H)\, S^2 A \varepsilon^{-2}\Big) \quad & \text{for RFE,} \\ \widetilde{O}\Big(\mathrm{poly}(H)\, \log(N)\, S A \varepsilon^{-2}\Big) \quad & \text{for TAE with } N \text{ possible reward functions.} \end{aligned}$$

Discussion. We notice that the theoretical price to pay for allowing an infinite number of rewards (or adversarial rewards) is the quadratic dependence on S in the sample complexity. While the frameworks of TAE and (even more) RFE yield strong end-to-end guarantees, they are limited to the *finite-horizon* setting and thus do not extend to goal-reaching tasks.

6.4 The SYOG Principle

We now give an overview of the SYOG principle, for "**S**et **Y**our **O**wn **G**oals".

High-level algorithmic structure of SYOG

Alternate between:

- **(GS)** *Goal Selection:* select one or multiple goal states to reach;
- **(PE)** *Policy Execution:* execute an explorative policy conditioned on this goal until it is reached or a (predefined or adaptive) stopping condition is met (and store the experience accumulated over the trajectory).

The SYOG approach can be cast as *intrinsically motivated goal-conditioned RL* (GC-RL, see e.g., Colas et al., 2020, for an excellent survey). In this framework, the agent must learn a *goal-conditioned policy*, which learns a distribution over actions conditioned not only on the current state but also on a goal state that it must reach as quickly as possible (in expectation). For the goal-conditioned policy to be able to reach a variety of goals in the unknown environment, the agent must autonomously set its own goals (via e.g., a curriculum) and learn to effectively reach them. Learning how to execute shortest paths between various (ideally all) pairs of states suggests a thorough understanding of the environment dynamics.

Recently, GC-RL has been extensively studied in the context of deep RL (see e.g., Schaul et al., 2015; Andrychowicz et al., 2017; Florensa et al., 2018; Warde-Farley et al., 2019; Nair et al., 2018; Colas et al., 2019; Zhao et al., 2019; Hartikainen et al., 2020; Ecoffet et al., 2020; Pong et al., 2020; Zhang et al., 2020b; Pitis et al., 2020). GC-RL has notably been shown to be a powerful heuristic to tackle navigation problems (e.g., Florensa et al., 2018), game playing (e.g., Ecoffet et al., 2020, on Montezuma's Revenge) or real-world robotic manipulation tasks (e.g., Pong et al., 2020).

Given the simple and unifying algorithmic structure of alternating between **(GS)** and **(PE)** steps, the core differences between the methods lie in the goal sampling distribution and in ways to take advantage of each policy execution as much as possible to speed up the learning. The specific choice of **(GS)** and **(PE)** steps directly influences the learning speed as well as the quality of the goal-conditioned policy returned by the algorithm. **(PE)** is typically improved by goal relabeling (Andrychowicz et al., 2017) or encoding goal states in lower-dimensional representations (Pong et al., 2020). We now review two popular approaches to prioritize **(GS)**, beyond the canonical uniform goal sampling distribution (Kaelbling, 1993; Schaul et al., 2015; Andrychowicz et al., 2017).

Sampling goals of intermediate difficulty. GoalGan (Florensa et al., 2018) assigns feasibility scores to goals as the proportion of time that the agents successfully reaches it. Based on this data, a generative adversarial network (GAN) is trained to generate goals of intermediate difficulty, whose feasibility scores are contained within an intermediate range. Meanwhile,

Zhang et al. (2020b) perform Value Disagreement based Sampling (VDS) by selecting goals that maximize the disagreement in an ensemble of goal-conditioned value functions. Value functions agree when the goals are too easy (the agent always manages to reach them) or too hard (the agent always fails to reach them) but disagree for goals of intermediate difficulty, on the fringe of the agent's current mastery of the environment. Interestingly, the theoretically grounded goal selection scheme that we introduce in Chapter 8 has similar high-level motivations with VDS, which can be seen as a way of operationalizing our provably efficient approach in deep RL.

Sampling goals to optimize novelty - diversity. Pong et al. (2020), Warde-Farley et al. (2019), and Pitis et al. (2020) bias the selection of goals towards sparsely visited areas of the goal space. For this purpose, they train density models in the goal space. While Pong et al. (2020) and Warde-Farley et al. (2019) target a uniform coverage of the goal space (diversity), Pitis et al. (2020) further skew the distribution of selected goals, effectively maximizing novelty. These algorithms have strong connections with *empowerment*-based methods (e.g., Mohamed and Rezende, 2015; Gregor et al., 2016; Eysenbach et al., 2019; Choi et al., 2021). Indeed, the mutual information (MI) between goals (denoted by the random variable G) and states (denoted by the random variable S) that empowerment methods aim to maximize can be written as $\mathcal{I}(S;G) = \mathcal{H}(G) - \mathcal{H}(G|S)$, where $\mathcal{H}$ denotes the entropy function. As a result, maximizing empowerment can be interpreted as maximizing the entropy of the goal distribution while minimizing the entropy of goals given experienced states. Algorithms that simultaneously learn to sample diverse goals ($\mathcal{H}(G) \uparrow$) and learn to represent goals with variational auto-encoders ($\mathcal{H}(G|S) \uparrow$) can thus be seen as maximizing empowerment.

While these aforementioned SYOG-based approaches effectively leverage deep RL techniques and are able to achieve impressive results in complex domains, they tend to be sample inefficient and driven by heuristics that lack substantial theoretical understanding and guarantees, even when restricted to the tabular case. The focus of Part II is to instigate a thorough and formal analysis of the SYOG principle in various settings, each with its specific technical challenges. We will systematically ask the following questions:

① What is the exact learning objective and how do we measure its achievement (i.e., sample complexity)?

② What upper bound on the sample complexity can we obtain?

③ What are the assumptions on the environment that we require?

④ What are the key algorithmic designs that allow us to establish our guarantee?

Chapter 7

SYOG in Reward-Free Reset-Free Communicating MDPs

In this chapter, we investigate the SYOG principle in reward-free reset-free communicating MDPs. We posit that many unsupervised objectives (i.e., that do not rely on an informative extrinsic reward signal) can be tackled by a *decoupled approach* composed of: **1)** An "objective-specific" algorithm that (adaptively) prescribes *how many* samples to collect *at which* states, as if it has access to a generative model (i.e., a simulator of the environment); **2)** An "objective-agnostic" sample collection exploration strategy responsible for generating the prescribed samples as fast as possible. By casting the latter as a (multi-goal varying) SSP exploration problem, we are able to leverage the techniques developed in Part I. Our decoupled approach allows us to tackle a variety of settings — e.g., model estimation, sparse reward discovery, goal-free cost-free exploration — for which we obtain improved or novel sample complexity guarantees. [1]

Contents

[1] This chapter is based on an article published in the proceedings of the 34^{th} Conference on Neural Information Processing Systems (NeurIPS 2021) (Tarbouriech et al., 2021a).

7.1 Motivation

One of the challenges in *online* reinforcement learning (RL) is that the agent needs to trade off the exploration of the environment and the exploitation of the samples to optimize its behavior. Whenever the agent needs to gather information about a specific region of the Markov decision process (MDP), it must plan for a policy to reach the desired states, despite not having exact knowledge of the environment dynamics. This makes solving the exploration-exploitation problem in RL highly non-trivial and it requires designing a specific strategy depending on the learning objective, such as PAC-MDP learning (e.g., Brafman and Tennenholtz, 2002; Strehl et al., 2009; Wang et al., 2019), regret minimization (e.g., Jaksch et al., 2010; Azar et al., 2017; Jin et al., 2018; Zhang et al., 2020c) or pure exploration (e.g., Jin et al., 2020; Kaufmann et al., 2021; Ménard et al., 2021; Zhang et al., 2020a; Zhang et al., 2021c).

A simpler scenario considered in the literature is to assume access to a *generative model* or *sampling oracle* ($\mathcal{SO}$) (Kearns et al., 2002). Given any state-action pair (s, a), the $\mathcal{SO}$ returns a next state s' drawn from the transition probability $P(\cdot|s, a)$ and a reward $r(s, a)$. In this case, it is possible to focus exclusively on where and how many samples to collect, while disregarding the problem of finding a suitable policy to obtain them. For instance, an $\mathcal{SO}$ can be used to obtain samples from the environment, which are combined with dynamic programming techniques to compute a near-optimal policy. $\mathcal{SO}$-based algorithms can be as simple as prescribing the same amount of samples from each state-action pair (e.g. Kearns et al., 2000; Kearns et al., 2002; Azar et al., 2013; Chen and Wang, 2016; Sidford et al., 2018; Agarwal et al., 2020; Li et al., 2020) or they may adaptively change the sample requirements on different state-action pairs (e.g. Chen et al., 2018; Wang, 2017; Zanette et al., 2019). An $\mathcal{SO}$ is also used in Monte-Carlo planning Szörényi et al., 2014; Grill et al., 2016; Bartlett et al., 2019 which focuses on computing the optimal action at the current state by optimizing over rollout trajectories sampled from the $\mathcal{SO}$. Finally, in multi-armed bandit (Lattimore and Szepesvári, 2020), there are cases where each arm corresponds to a state (or state-action), and "pulling" an arm translates into a call to an $\mathcal{SO}$ (see e.g., the pure exploration setting that we introduced in Tarbouriech and Lazaric, 2019). Unfortunately, while an $\mathcal{SO}$ may be available in domains such as simulated robotics and computer games, this is not the case in the more general *online RL* setting.

In this chapter, we tackle the exploration-exploitation problem in online RL by drawing inspiration from the $\mathcal{SO}$ assumption. Specifically, we define an approach that is decoupled in two parts: **1**) an "objective-specific" algorithm that assumes access to an $\mathcal{SO}$ that (adaptively) prescribes the samples needed to achieve the learning objective of interest, and **2**) an "objective-agnostic" algorithm that takes on the exploration challenge of collecting the samples requested by the $\mathcal{SO}$-based algorithm as quickly as possible.[2]

[2] Alternatively, we can view it as a general approach to take any $\mathcal{SO}$-based algorithm and convert it into an online RL algorithm.

7.2 Problem Definition

We consider that the MDP $M \triangleq \langle \mathcal{S}, \mathcal{A}, P, r, s_0 \rangle$ is finite and *reset-free*, with arbitrary starting state $s_0 \in \mathcal{S}$. Calling an $\mathcal{SO}$ in any state-action pair (s, a) leads to two outcomes: a next state sampled from the transition probability distribution $P(\cdot|s, a) \in \Delta(\mathcal{S})$, and (optionally) a scalar reward $r(s, a) \in \mathbb{R}$. For any policy π and pair of states (s, s'), let $\tau_\pi(s \to s')$ be the (possibly infinite) hitting time from s to s' when executing π, i.e., $\tau_\pi(s \to s') \triangleq \inf\{t \geq 0 : s_{t+1} = s' | \, s_1 = s, \pi\}$, where s_t is the state visited at time step t. We define

$$D_{ss'} \triangleq \min_{\pi \in \Pi} \mathbb{E}\left[\tau_\pi(s \to s')\right], \qquad D_{s'} \triangleq \max_{s \in \mathcal{S} \setminus \{s'\}} D_{ss'}, \qquad D \triangleq \max_{s' \in \mathcal{S}} D_{s'},$$

where $D_{ss'}$ is the shortest-path distance between s and s', $D_{s'}$ is the SSP-diameter of s' (see Chapter 3) and D is the MDP diameter (Jaksch et al., 2010).

We now formalize the problem of simulating an $\mathcal{SO}$ (i.e., to generate the samples prescribed by an $\mathcal{SO}$-based algorithm). At each time step $t \geq 1$, the agent receives a function

$$b_t : \mathcal{S} \times \mathcal{A} \to \mathbb{N},$$

where $b_t(s, a)$ defines the total number of samples that need to be collected at (s, a) by time step t. We consider that $(b_t)_{t \geq 1}$ is an arbitrary sequence with each b_t measurable w.r.t. the filtration up to time t (i.e., it may depend on the samples observed so far).[3] We focus on the objective of designing an *online algorithm* that minimizes the time required to collect the prescribed samples. Since the environment is initially unknown, we need to trade off between exploring states and actions to improve estimates of the dynamics and exploiting current estimates to collect the required samples as quickly as possible. We formally define the performance metric as follows.

Definition 7.1. *For any state-action pair, we denote by* $N_t(s, a) \triangleq \sum_{i=1}^{t} \mathbb{1}_{\{(s_i, a_i) = (s, a)\}}$ *the number of visits to state* s *and action* a *up to (and including) time step* t*. Given a sampling requirement sequence* $b \triangleq (b_t)_{t \geq 1}$ *with* $b_t : \mathcal{S} \times \mathcal{A} \to \mathbb{N}$ *and a confidence level* $\delta \in (0, 1)$*, we define the sample complexity of a learning algorithm* $\mathfrak{A}$ *as*

$$\mathcal{C}(\mathfrak{A}, b, \delta) \triangleq \min\Big\{t > 0 : \mathbb{P}\big(\forall (s, a) \in \mathcal{S} \times \mathcal{A}, \; N_t(s, a) \geq b_t(s, a)\big) \geq 1 - \delta\Big\}.$$

With no additional condition, it is trivial to define problems such that $\mathcal{C}(\mathfrak{A}, b, \delta) = +\infty$ for any algorithm. To avoid this case, we introduce the following assumptions.

[3] Allowing adaptive sampling requirements enables to pair GOSPRL with $\mathcal{SO}$-based algorithms that adjust their requirements *online* as samples are being generated (see e.g., Section 7.4.2).

Assumption 7.2. *The MDP M is communicating with a finite and unknown diameter $D < +\infty$.*

Assumption 7.3. *There exist an unknown and bounded function $\bar{b} : \mathcal{S} \times \mathcal{A} \to \mathbb{N}$ such that the sequence $(b_t)_{t\geq 1}$ verifies: $\forall t \geq 1,\ \forall (s,a) \in \mathcal{S} \times \mathcal{A},\ b_t(s,a) \leq \bar{b}(s,a)$.*

Assumption 7.2 guarantees that whatever state needs to be sampled, there exists at least one policy that can reach it in finite time almost-surely. Assumption 7.3 ensures that the sequence of sampling requirements does not diverge and can thus be fulfilled in finite time. These assumptions guarantee that the problem in Definition 7.1 is well-posed and the sample complexity is bounded.

A variety of problems can be cast under our decoupled approach, in the sense that they can be tackled by solving the problem of Definition 7.1 under a specific instantiation of the sampling requirement sequence $(b_t)_{t\geq 1}$. For instance, consider the problem of covering the state-action space (e.g., to discover a hidden sparse reward), then the requirement is immediately defined as $b_t(s,a) = 1$. In Sections E.8 and 7.4, we review problems where defining b_t can be as simple as computing the sufficient number of samples needed to reach a certain level of accuracy in estimating a quantity of interest (e.g., model estimation) or can be directly extracted from existing literature (e.g., ε-optimal policy learning).

We now provide a simple worst-case lower bound on the sample complexity (details in Section E.3).

Lemma 7.4. *For any $S \geq 1$, there exists an MDP with S states satisfying Assumption 7.2 such that for any sampling requirement $b : \mathcal{S} \to \mathbb{N}$ satisfying Assumption 7.3,*

$$\min_{\mathfrak{A}} \mathcal{C}(\mathfrak{A}, b, \tfrac{1}{2}) = \Omega\Big(\sum_{s\in\mathcal{S}} D_s b(s)\Big).$$

Lemma 7.4 shows that the (possibly non-stationary) policy minimizing the time to collect all samples requires $\Omega(\sum_s D_s b(s))$ time steps in a worst-case MDP. We also notice that when the total sampling requirement B is concentrated on the state $\bar{s}$ for which $D_{\bar{s}} = D$ (i.e., $b(s') = 0$, $\forall s' \neq \bar{s}$), the previous bound reduces to $\Omega(BD)$.

Algorithm 7.1: Algorithm GOSPRL

Input: sampling requirement sequence $(b_t)_{t\geq 1}$ with $b_t : \mathcal{S} \times \mathcal{A} \to \mathbb{N}$ revealed at time t (or anytime before).
Initialize: Set $\mathcal{G}_1 \triangleq \{s \in \mathcal{S} : \exists a \in \mathcal{A}, b_1(s,a) > 0\}$, time step $t \triangleq 1$, counters $N_1(s,a) \triangleq 0$, attempt index $k \triangleq 1$ and attempt counters $U_1(s,a) \triangleq 0$, $\nu_1(s,a) \triangleq 0$.
while $\mathcal{G}_k$ is not empty **do**
    Define the SSP problem M_k with goal states $\mathcal{G}_k$, and compute its optimistic shortest-path policy $\widetilde{\pi}_k$.
    Set `flag` = `True` and counter $\nu_k(s,a) \triangleq 0$.
    while `flag` **do**
        Execute action $a_t \triangleq \widetilde{\pi}_k(s_t)$ and observe next state $s_{t+1} \sim P(\cdot|s_t, a_t)$.
        Increment counters $\nu_k(s_t, a_t)$ and $N_t(s_t, a_t)$.
        if $s_{t+1} \in \mathcal{G}_k$ or $\nu_k(s_t, a_t) > \{U_k(s_t, a_t) \vee 1\}$ **then**
            Set `flag` = `False`.
        Set t += 1.
    if $s_t \in \mathcal{G}_k$ **then**
        Execute an action $a \in \mathcal{A}$ such that $N_t(s_t, a) < b_t(s_t, a)$, observe next state $s_{t+1} \sim P(\cdot|s_t, a)$ and set t += 1.
    Set $U_{k+1}(s,a) \triangleq U_k(s,a) + \nu_k(s,a)$, k += 1.
    Update the set of goal states $\mathcal{G}_k \triangleq \{s \in \mathcal{S} : \exists a \in \mathcal{A}, N_{t-1}(s,a) < b_{t-1}(s,a)\}$.

7.3 Online Learning for Sampling Oracle Simulation with GOSPRL

In this section, we introduce our algorithm for the problem in Definition 7.1, bound its sample complexity and discuss several extensions.

7.3.1 The GOSPRL Algorithm

In Algorithm 7.1 we outline GOSPRL (*Goal-based Optimistic Sampling Procedure for Reinforcement Learning*). At each time step t, GOSPRL receives a sampling requirement $b_t : \mathcal{S} \times \mathcal{A} \to \mathbb{N}$. The algorithm relies on the principle of optimism in the face of uncertainty and proceeds through *attempts* to collect relevant samples. We index the attempts by $k = 1, 2, \ldots$ and denote by t_k the time step at the start of attempt k and by $U_k \triangleq N_{t_k - 1}$ the number of samples available at the start of attempt k. At each attempt, GOSPRL goes through the following steps: **1)** Cast the under-sampled states as goal states and define an associated unit-cost multi-goal SSP instance (with unknown transitions); **2)** Compute an optimistic SSP policy; **3)** Execute the policy until either a goal state is reached or a stopping condition is satisfied; **4)** If a sought-after goal state denoted by g has been reached, execute an under-sampled action (i.e., an action a such that $N_t(g,a) < b_t(g,a)$). The algorithm ends when the sampling requirements are met, i.e., at the first time $t \geq 1$ where $N_t(s,a) \geq b_t(s,a)$ for all (s,a).

Step 1. At any attempt k we begin by defining the set of all under-sampled states

$$\mathcal{G}_k \triangleq \Big\{ s \in \mathcal{S} : \exists a \in \mathcal{A}, N_{t_k-1}(s,a) < b_{t_k-1}(s,a) \Big\}.$$

We then cast the sample collection problem as a goal-reaching objective (see Part I), by constructing a multi-goal SSP problem denoted by $M_k \triangleq \langle \mathcal{S}_k, \mathcal{A}, P_k, c_k, \mathcal{G}_k \rangle$, with:[4]

- $\mathcal{G}_k$ denotes the set of goal states, $\mathcal{S}_k := \mathcal{S} \setminus \mathcal{G}_k$ the set of non-goal states and $\mathcal{A}$ the set of actions.
- The transition model P_k is the same as the original P except for the transitions exiting the goal states which are redirected as a self-loop, i.e., $P_k(s'|s,a) \triangleq P(s'|s,a)$ and $P_k(g|g,a) \triangleq 1$ for any $(s,s',a,g) \in \mathcal{S}_k \times \mathcal{S} \times \mathcal{A} \times \mathcal{G}_k$.
- The cost function c_k is defined as follows: for any $a \in \mathcal{A}$, any goal state $g \in \mathcal{G}_k$ is zero-cost ($c_k(g,a) \triangleq 0$), while the non-goal costs are unitary ($c_k(s,a) \triangleq 1$ for $s \in \mathcal{S}_k$).

According to Proposition 2.11, Assumption 7.2 and the positive non-goal costs c_k entail that solving M_k is a well-posed SSP problem and that there exists an optimal policy that is *proper* (i.e., that eventually reaches one of the goal states with probability 1 when starting from any $s \in \mathcal{S}_k$). Crucially, the objective of collecting a sample from the under-sampled states $\mathcal{G}_k$ coincides with the SSP objective of minimizing the expected cumulative cost to reach a goal state in M_k.

Step 2. Since P_k is unknown, we cannot directly compute the shortest-path policy for M_k. Instead, leveraging the samples collected so far, we apply an extended value iteration scheme for SSP which implicitly skews the empirical transitions $\widehat{P}_k$ towards reaching the goal states. This procedure can be done efficiently as shown in Chapter 4,[5] and it outputs an *optimistic* shortest-path policy $\widetilde{\pi}_k$.

Step 3. $\widetilde{\pi}_k$ is then executed with the aim of quickly reaching an under-sampled state. Along its trajectory, the counter N_t is updated for each visited state-action. Because of the error in estimating the model, $\widetilde{\pi}_k$ may never reach one of the goal states (i.e., it may not be proper in P_k). Thus $\widetilde{\pi}_k$ is executed until either one of the goals in $\mathcal{G}_k$ is reached, or the number of visits is doubled in a state-action pair in $\mathcal{S}_k \times \mathcal{A}$, a standard termination condition first introduced by Jaksch et al. (2010). If a sought-after goal state is reached, the agent executes an under-sampled action according to the current sampling requirements at that state. At the end of each attempt, the statistics (e.g., model estimate) are updated.

The algorithmic design of GOSPRL is conceptually simple and can flexibly incorporate various modifications driven by slightly different objectives or prior knowledge, without altering

[4] If the current state s_{t_k} is under-sampled (i.e., $s_{t_k} \in \mathcal{G}_k$), we duplicate the state and consider it to be both a goal state in $\mathcal{G}_k$ and a non-goal state from which the attempt k starts (and whose outgoing dynamics are the same as those of s_{t_k}), which ensures that the state at the start of each attempt cannot be a goal state.

[5] The only difference is that here we leverage a Bernstein-based construction of confidence intervals, as also done by Rosenberg et al. (2020) (details in Section E.1).

Theorem 7.5. (i) Any non-unit SSP costs can be designed as long as they are positive and bounded: deterring costs may e.g., be assigned to "trap" states with large negative environmental reward that the agent may seek to avoid. (ii) Penalizing the visitation of sufficiently visited states (with costs larger than one) may give the agent incentive to *even out* its sample collection and thus avoid over-sampling some areas of the state space. (iii) It is possible to focus on specific goal states instead of the set of all under-sampled states. In practice, using such a *meta-goal* makes the optimal SSP policy more robust to noise. While the SSP solution to M_k indeed seeks to reach the closest under-sampled state, random transitions may move the agent closer to any other state in $\mathcal{G}_k$ and this would naturally trigger the policy to focus on such closer state. On the other hand, providing the SSP policy with a single goal state may lead to much longer and wasteful attempts. (iv) We remark that if the entire state space is initially under-sampled, any action would produce a "useful" sample and different heuristics can be implemented in prioritizing actions accordingly.

7.3.2 Sample Complexity Guarantee of GOSPRL

Theorem 7.5 establishes the sample complexity guarantee of GOSPRL (Algorithm 7.1).

Theorem 7.5. *Under Assumption 7.2 and 7.3, for any sampling requirement sequence $b = (b_t)_{t\geq 1}$ and any confidence level $\delta \in (0,1)$, the sample complexity of* GOSPRL *is bounded as*

$$\mathcal{C}(\text{GOSPRL}, b, \delta) = \widetilde{O}\Big(\overline{B}D + D^{3/2}S^2A\Big), \tag{7.1}$$

$$\mathcal{C}(\text{GOSPRL}, b, \delta) = \widetilde{O}\Big(\sum_{s\in\mathcal{S}} \big(D_s \bar{b}(s) + D_s^{3/2}S^2A\big)\Big), \tag{7.2}$$

where the $\widetilde{O}$ notation hides logarithmic dependencies on S, A, D, $1/\delta$ and $\bar{b}(s) \triangleq \sum_{a\in\mathcal{A}} \bar{b}(s,a)$ and $\overline{B} \triangleq \sum_{s\in\mathcal{S}} \bar{b}(s)$. Recall that $D_s \leq D$ is the SSP-diameter of state s and captures the difficulty of collecting a sample at state s starting at any other state in the MDP.

We notice that in practice GOSPRL stops at the first *random* step τ at which the sampling requirement $b_\tau(s,a)$ is achieved for all (s,a). Theorem 7.5 provides a worst-case upper bound on the stopping time of GOSPRL using the possibly loose bound $b_\tau(s,a) \leq \bar{b}(s,a)$. On the other hand, in the special case of $b : \mathcal{S} \to \mathbb{N}$ when the requirements are both time-independent (i.e., given as initial input to the algorithm) and action-independent, the actual sampling requirement $b(s)$ (resp. $B \triangleq \sum_{s\in\mathcal{S}} b(s)$) replaces $\bar{b}(s)$ (resp. $\overline{B}$) in the bound. In the following, we consider this case for the ease of exposition.

Proof idea. The key step (see Section E.2 for the full derivation) is to link the sample complexity of GOSPRL to the regret accumulated over the sequence of multi-goal SSP problems M_k gener-

ated across multiple attempts. Indeed, extending Definition 3.1 (on the regret in SSP with a single fixed goal), we can define the regret at attempt k as the gap between the performance of the SSP-optimal policy $\pi_k^\star$ solving M_k (i.e., the minimum expected number of steps to reach any of the states in $\mathcal{G}_k$ starting from s_{t_k}) and the actual number of steps executed by GOSPRL before terminating the attempt. In what follows we build on the SSP regret minimization analysis of Rosenberg et al. (2020), although a similar reasoning holds for the algorithm and analysis used in Chapter 4 or Chapter 5. Specifically, while traditional SSP regret minimization analyses assume that the goal is fixed, we show that it is possible to bound the regret accumulated across different attempts for any arbitrary sequence of goals. The proof is concluded by bounding the cumulative performance of the SSP-optimal policies and it leads to the bound $\widetilde{O}(BD+D^{3/2}S^2A)$ where $B \triangleq \sum_{s\in\mathcal{S}} b(s)$. On the other hand, the refined bound in Equation (7.2) requires a more careful analysis, where we no longer directly translate regret bounds into sample complexity and we rather focus on relating the performance to state-dependent quantities D_s and $b(s)$. Finally, we show that the extension to the general case of time-dependent action-dependent sampling requirements is straightforward and obtain Theorem 7.5. □

Interpretation of Theorem 7.5. We can decompose Equation (7.1) as a linear term in B and a constant term. In the regime of large sample requirements (i.e., large B), the sample complexity thus reduces to $\widetilde{O}(BD)$, which adds at most an extra "cost" factor of D w.r.t. an $\mathcal{SO}$. As this may be loose in many cases, the more refined analysis of Equation (7.2) stipulates a cost of D_s to collect a sample at state s, which better captures the connectivity of the MDP. In fact the lower bound in Lemma 7.4 shows that this cost of D_s is *unavoidable in the worst case*, and that GOSPRL is only constant and logarithmic terms off w.r.t. to the best sample complexity that can be achieved in the worst case. While an extra attempt of refinement would be to avoid being worst-case w.r.t. the starting state in the definition of D_s,[6] this seems particularly challenging as the randomness of the environment makes it hard to control and analyze the sequence of states traversed by the agent.

Optimal solution. GOSPRL targets a *greedy-optimal* strategy, which seeks to sequentially minimize each expected time to reach an under-sampled state. Alternatively, one may wonder if it is possible to design a learning algorithm that approaches the performance of the *exact-optimal* solution, i.e., a (non-stationary) policy explicitly minimizing the number of steps required to fulfill the sampling requirements.[7] Such strategy can be characterized as the optimal policy of an SSP problem for an MDP with state space augmented by the current sampling requirements and goal state corresponding to the case when all desired samples are collected. Even under

[6] For instance, consider a simple deterministic chain with a requirement of one sample per state. If the agent starts on the leftmost state, then a policy that keeps moving right has sample complexity S without extra factor D.

[7] Notice that as illustrated in the lower bound of Lemma 7.4, the exact-optimal and greedy-optimal have the same performance in the worst case.

known dynamics, the computational complexity of computing the optimal policy in this MDP (e.g., via value iteration) is exponential (scaling in B^S). When the dynamics is unknown, it appears highly challenging to obtain any learning algorithm whose performance is comparable to the exact-optimal strategy for any finite sample requirement B.

Beyond Communicating MDPs. In Section E.4 we design an extension of GOSPRL to poorly or weakly communicating environments. In this setting, it is expected to assess online the "reachability" of certain sampling requirements and discard them whenever associated to states that are *too difficult* to reach or unreachable. Given as input a "reachability" threshold L, we derive sample complexity guarantees for our variant of GOSPRL where the (possibly large or infinite) diameter D is fittingly replaced by L.

7.4 Applications of GOSPRL

An appealing feature of GOSPRL is that it can be integrated with techniques that compute the (fixed or adaptive) sampling requirements to readily obtain an online RL algorithm with theoretical guarantees. In this section, we focus on three specific problems where in our decoupled approach the $\mathcal{SO}$-based algorithm is either trivial or can be directly extracted from existing literature, and its combination with the sample collection strategy of GOSPRL yields improved or novel guarantees. Other applications (e.g., PAC-policy learning, diameter estimation, bridging bandits and MDPs) are illustrated in Section E.8.

7.4.1 Sparse Reward Discovery (Treasure)

A number of recent methods focus on the state-space coverage problem, where each state in the MDP needs to be reached as quickly as possible. This problem is often motivated by environments where a one-hot reward signal, called the *treasure*, is hidden and can only be discovered by reaching a specific state and taking a specific action. Not only the environment but also the treasure state-action pair is unknown, and the agent does not receive any side information to guide its search (e.g., a measure of closeness to the treasure). Thus the agent must perform exhaustive exploration to find the treasure.

Definition 7.6. *Given a confidence $\delta \in (0,1)$, the* Treasure *sample complexity of a learning algorithm $\mathfrak{A}$ is defined as*

$$\mathcal{C}_{\textsc{Treasure}}(\mathfrak{A},\delta) \triangleq \min\Big\{t>0 : \mathbb{P}(\forall (s,a)\in\mathcal{S}\times\mathcal{A},\ N_t(s,a)\geq 1)\geq 1-\delta\Big\}.$$

In this case, a $\mathcal{SO}$-based algorithm would immediately solve the problem by collecting one sample from each state-action pair. As a result, we can directly apply GOSPRL for Treasure by simply setting $b(s,a) = 1$ for each (s,a) and from Theorem 7.5 with $B = SA$ we obtain the following guarantee.

Lemma 7.7. GOSPRL *with* $b(s,a) = 1$ *verifies* $\mathcal{C}_{\text{Treasure}}(\text{GOSPRL}, \delta) = \widetilde{O}\left(D^{3/2}S^2A\right)$.

We now compare this result to alternative approaches to the problem, showing that GOSPRL has state-of-the-art guarantee for Treasure (see Section E.6 for details).

- First, reward-free exploration methods (e.g., Jin et al., 2020; Zhang et al., 2021c; Kaufmann et al., 2021; Ménard et al., 2021, see Section 6.3.2) are designed for finite-horizon problems so their guarantees cannot be directly translated to sample complexity for the Treasure problem. Nonetheless, we draw inspiration from their algorithmic principles and analyze a *reward-free* variant of UCRL2 (Jaksch et al., 2010; Fruit et al., 2020). Specifically we consider o/1-UCRL, which runs UCRL by setting a reward of 1 to under-sampled states and 0 otherwise. However, we obtain a Treasure sample complexity for o/1-UCRL of $\widetilde{O}\left(\sum_{s\in\mathcal{S}} D_s^3 S^2 A\right)$, which is always worse than the bound in Lemma 7.7.
- Second, we can adapt the MaxEnt approach (Hazan et al., 2019; Cheung, 2019, see Section 6.3.1) to state-action coverage so that it targets a policy whose stationary state-action distribution λ maximizes $H(\lambda) \triangleq -\sum_{s,a} \lambda(s,a) \log \lambda(s,a)$. While optimizing this entropy does not provably solve Treasure, it encourages us to take a "worst-case" approach w.r.t. the state-action visitations, and rather maximize $F(\lambda) \triangleq \min_{(s,a)\in\mathcal{S}\times\mathcal{A}} \lambda(s,a)$. We show that the learning algorithm of Cheung (2019) instantiated to maximize F yields a Treasure sample complexity of at least $\Omega\left(\min\left\{D^2S^2A/(\omega^\star)^2, D^3/(\omega^\star)^3\right\}\right)$ with $\omega^\star \triangleq \min_\lambda F(\lambda) \le (SA)^{-1}$, which is significantly poorer than Lemma 7.7. In fact, in contrast to MaxEnt-inspired methods that optimize for a single *stationary* policy, GOSPRL realizes a non-stationary strategy that gradually collects the required samples by tackling successive learning problems.

7.4.2 Model Estimation (ModEst)

We now study the problem of accurately estimating the unknown transition dynamics in a reward-free communicating environment. It was discussed in Section 6.3.1 and we refer to it as the *model-estimation* problem, or ModEst for short.

Definition 7.8. *Given an accuracy level* $\eta > 0$ *and a confidence level* $\delta \in (0,1)$, *the* ModEst *sample complexity of an online learning algorithm* $\mathfrak{A}$ *is defined as*

$$\mathcal{C}_{\text{ModEst}}(\mathfrak{A}, \eta, \delta) \triangleq \min\left\{t > 0 : \mathbb{P}(\forall (s,a) \in \mathcal{S}\times\mathcal{A}, \|\widehat{P}_{\mathfrak{A},t}(\cdot|s,a) - P(\cdot|s,a)\|_1 \le \eta) \ge 1-\delta\right\},$$

where $\widehat{P}_{\mathfrak{A},t}$ is the estimate (i.e., empirical average) of the transition dynamics P after t time steps.

Unlike in TREASURE, here the sampling requirements are not immediately prescribed by the problem. To define the $\mathcal{SO}$-based algorithm we first upper-bound the estimation error using an empirical Bernstein inequality and then invert it to derive the amount of samples $b_t(s,a)$ needed to achieve the desired level of accuracy η (see Section E.5). Specifically, letting $\widehat{\sigma}_t^2(s'|s,a) \triangleq \widehat{P}_t(s'|s,a)(1-\widehat{P}_t(s'|s,a))$ be the estimated variance of the transition from (s,a) to s' after t steps, we set

$$b_t(s,a) \triangleq \left\lceil \frac{57(\sum_{s'} \widehat{\sigma}_t(s'|s,a))^2}{\eta^2} \log^2\left(\frac{8e(\sum_{s'} \widehat{\sigma}_t(s'|s,a))^2\sqrt{2SA}}{\sqrt{\delta}\eta}\right) + \frac{24S}{\eta}\log\left(\frac{24S^2A}{\delta\eta}\right)\right\rceil. \tag{7.3}$$

Since the estimated variance changes depending on the samples observed so far, the sampling requirements are *adapted* over time. Given that $\widehat{\sigma}_t^2(s'|s,a) \leq 1/4$, $b_t(s,a)$ is always bounded so Theorem 7.5 provides the following guarantee.

Lemma 7.9. *Let $\Gamma \triangleq \max_{s,a}\|P(\cdot|s,a)\|_0 \leq S$ be the maximal support of $P(\cdot|s,a)$ over the state-action pairs (s,a). Running* GOSPRL *with the sampling requirements in Equation (7.3) yields*

$$\mathcal{C}_{\text{MODEST}}(\text{GOSPRL}, \eta, \delta) = \widetilde{O}\Big(\frac{D\Gamma SA}{\eta^2} + \frac{DS^2A}{\eta} + D^{3/2}S^2A\Big).$$

Lemma 7.9 improves over our Frank-Wolfe-based approach for MODEST reviewed in Section 6.3.1 (example ③) and studied in Tarbouriech et al. (2020c). First, the latter suffers from an inverse dependency on the stationary state-action distribution that optimizes a proxy objective function used in the derivation of their algorithm. Second, while the Frank-Wolfe-based approach requires an ergodicity assumption, Lemma 7.9 is the first sample complexity result for MODEST in the more general communicating setting.

7.4.3 Goal-Free & Cost-Free Exploration in Communicating MDPs

We finally delve into the paradigm of *reward-free exploration* introduced by Jin et al. (2020) and reviewed in Section 6.3.2. While the problem has been exclusively analyzed in the *finite-horizon* setting, here we study the more general and challenging setting of *goal-conditioned* RL. We define the *goal-free cost-free* objective as follows: after the exploration phase, the agent is expected to compute a near-optimal goal-conditioned policy for *any* goal state and *any* cost function

(w.l.o.g. we consider a maximum possible cost $c_{\max} = 1$). Recall from Part I that given a goal state g and costs c, the (possibly unbounded) SSP value function of a policy π is

$$V^{\pi}(s \to g) \triangleq \mathbb{E}\left[\sum_{t=1}^{\tau_\pi(s\to g)} c(s_t, \pi(s_t)) \mid s_1 = s\right].$$

Given a slack parameter $\theta \in [1, +\infty]$, we say that a policy $\widehat{\pi}$ is (ε, θ)-optimal if [8]

$$V^{\widehat{\pi}}(s \to g) \leq \min_{\pi:\, \mathbb{E}[\tau_\pi(s\to g)] \leq \theta D_{s,g}} V^{\pi}(s \to g) + \varepsilon.$$

In this setting, constructing an efficient $\mathcal{SO}$-based algorithm is considerably more complex than Treasure and ModEst. Relying on the sample complexity analysis for the fixed-goal SSP problem with a *generative model* that we derive in Tarbouriech et al. (2021b) (see Section 3.4), we define the (adaptive) number of samples needed in each state-action pair for our online objective. Although the number depends on the unknown diameter, we estimate D using GOSPRL. The resulting sequence of sampling requirements is then fed online to GOSPRL. Combining the sample complexity bound with generative model and the properties of GOSPRL yields the following guarantee (see Section E.7).

Lemma 7.10. *Consider any MDP satisfying Assumption 7.2 and the goal-free cost-free exploration problem with accuracy level $0 < \varepsilon \leq 1$, confidence level $\delta \in (0, 1)$, minimum cost $c_{\min} \in [0, 1]$, slack parameter $\theta \in [1, +\infty]$. We can instantiate* GOSPRL *so that its exploration phase (i.e., number of time steps) is bounded with probability at least $1 - \delta$ by*

$$\widetilde{O}\left(\frac{D^4 \Gamma S A}{\omega \varepsilon^2} + \frac{D^3 S^2 A}{\omega \varepsilon} + \frac{D^3 \Gamma S A}{\omega^2}\right),$$

where $\omega \triangleq \max\{c_{\min}, \varepsilon/(\theta D)\} > 0$ (thus, either $c_{\min} = 0$ or $\theta = +\infty$, but not both simultaneously). Following the exploration phase, the algorithm can compute in the planning phase, for any goal state $g \in \mathcal{S}$ and any cost function c in $[c_{\min}, 1]$, a policy $\widehat{\pi}_{g,c}$ that is (ε, θ)-optimal.

Lemma 7.10 establishes the first sample complexity guarantee for general goal-free, cost-free exploration. While the objective is demanding and the upper bound on the length of the exploration phase can be large, the main purpose of this result is to showcase how GOSPRL can be readily instantiated to tackle a challenging exploration problem for which no existing solution can be easily leveraged. Comparing our analysis to the finite-horizon objective of Jin et al. (2020) reveals two interesting properties:

[8] This reduces to standard ε-optimality for $\theta = +\infty$. We only consider $\theta < +\infty$ in the case of minimum possible cost $c_{\min} = 0$ and it ensures that the algorithm targets proper policies (see Section E.7).

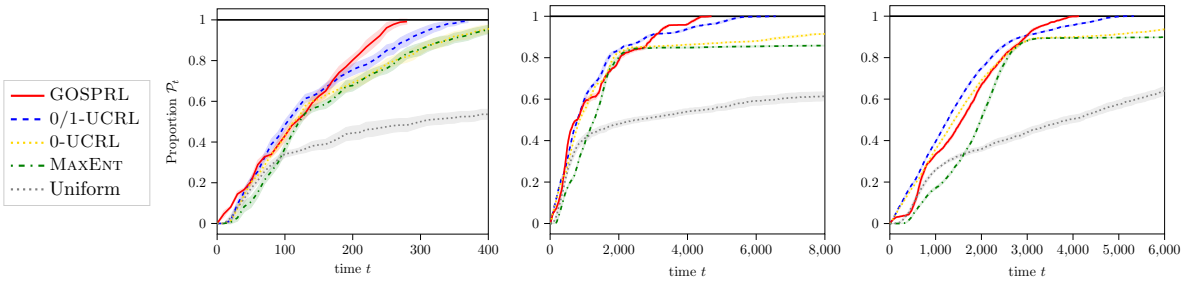


Figure 7.1 – Treasure-10 problem (i.e., with $b(s,a) = 10$): Proportion $\mathcal{P}_t$ of states meeting the requirements at time t, averaged over 30 runs. By definition of the sample complexity, the metric of interest is *not* the rate of increase of $\mathcal{P}_t$ over time but only the time needed to reach the line of success $\mathcal{P}_t = 1$. *Left:* 6-state RiverSwim, *Center:* 24-state corridor gridworld, *Right:* 43-state 4-room gridworld (see Section F.4 for details on the domains).

- ***The goal-free aspect:*** moving from finite-horizon to goal-conditioned renders *unavoidable* both the communicating requirement (Assumption 7.2) and the bound's dependency on the unknown diameter D (which partly captures the role of the known horizon H in the bound of Jin et al., 2020).
- ***The cost-free aspect:*** in contrast to finite-horizon, the value of $c_{\min}$ has an important impact on the type of performance guarantees we can obtain; in particular our analysis distinguishes between positive and non-negative costs (as also done in existing SSP analysis, see Part I).

7.5 Experiments

In this section, we report a preliminary numerical validation of our theoretical findings. While GOSPRL can be integrated in many different contexts, here we focus on the problems where our theory suggests that GOSPRL performs better than state-of-the-art online learning methods.

Treasure-type problem. We consider a Treasure-type problem (Section 7.4.1), where for all (s,a) we set $b(s,a) = 10$ instead of 1 (we call it the Treasure-10 problem).[9] We begin by showing in Figure 7.4 that it is easy to construct a worst-case problem where the sample complexity scales linearly with the diameter, which is consistent with the theoretical discussion in Sections 7.2 and 7.3.

We compare to two heuristics based on UCRL2B (Jaksch et al., 2010; Fruit et al., 2020): o-UCRL, where the reward used in computing the optimistic policy is set proportional to $([N(s,a) - b(s,a)]^+)^{-1/2}$, and o/1-UCRL with reward 1 for undersampled state-action pairs and

[9] Since GOSPRL and our baselines are all based on upper confidence bounds, they tend to display similar behaviors in the initial phases of learning, since the estimates when $N(s,a) = 0$ are similar. As the number of samples required in each state-action increases, the difference between the algorithms' design starts making a real difference in the behavior and eventually their performance. This is why we study here Treasure-10 instead of the Treasure-1 problem for which empirical performance is comparable between learning algorithms.

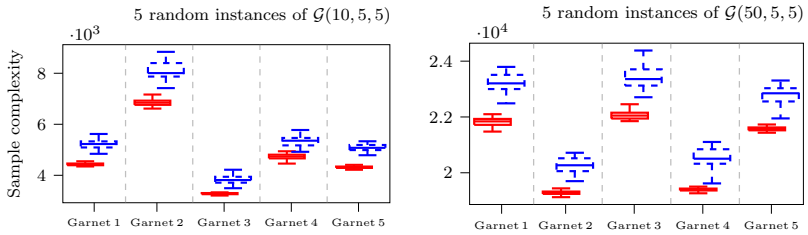


Figure 7.2 – Sample complexity boxplots of GOSPRL (in red) and o/1-UCRL (in blue). Each column represents 30 runs on a randomly generated Garnet $\mathcal{G}(S, A = 5, \beta = 5)$ with randomly generated state-action sampling requirements $b : \mathcal{S} \times \mathcal{A} \to \mathcal{U}(0, 100)$. *Left:* $S = 10$, *Right:* $S = 50$.

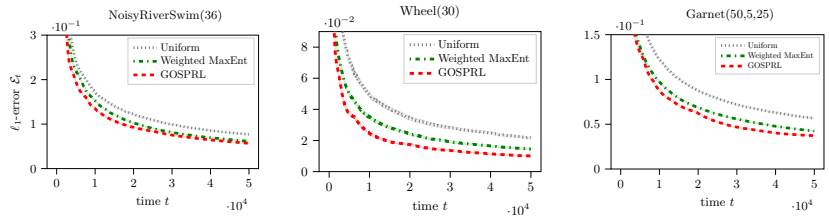


Figure 7.3 – ModEst problem: ℓ_1-error $\mathcal{E}_t \triangleq (SA)^{-1} \cdot \sum_{s,a} \|\widehat{p}_t(\cdot|s,a) - p(\cdot|s,a)\|_1$, averaged over 30 runs. *Left:* NoisyRiverSwim(36), *Center:* Wheel(30), *Right:* Randomly generated Garnet $\mathcal{G}(50, 5, 25)$.

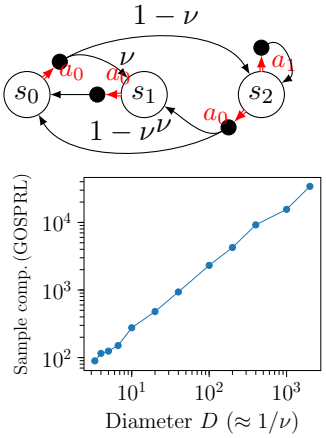


Figure 7.4 – Simple three-state reward-free domain (Fruit et al., 2018b) and Treasure-10 sample complexity of GOSPRL (averaged over 30 runs) as a function of the diameter $D \approx 1/\nu$.

0 otherwise. We also compare with the MaxEnt algorithm (Cheung, 2019) that maximizes entropy over the state-action space, and with a uniformly random baseline policy. We test on the RiverSwim domain (Strehl and Littman, 2008) and various gridworlds (see Section F.4 for details and more results). Figure 7.1 reports the proportion $\mathcal{P}_t$ of states that satisfy the sampling requirements at time t. Our metric of interest is the time needed to collect all required samples, and we see that GOSPRL reaches the $\mathcal{P}_t = 1$ line of success consistently, and faster than o/1-UCRL, while the other heuristics struggle. The steady increase of $\mathcal{P}_t$ illustrates GOSPRL's design to progressively meet the sampling requirements, and not exhaust them state after state.

Random MDPs and sampling requirements. To study the generality of GOSPRL to collect arbitrary sought-after samples, we further compare GOSPRL with o/1-UCRL which is the best heuristic from the previous experiment. We test on a variety of randomly generated *configurations*, that we define as follows: each configuration corresponds to **i)** a randomly generated Garnet environment $\mathcal{G}(S, A, \beta)$ (with S states, A actions and branching factor β, see Bhatnagar et al., 2009), and **ii)** randomly generated requirements $b(s, a) \in \mathcal{U}(0, \overline{U})$, where the maximum budget is set to $\overline{U} = 100$ to have a wide range of possible requirements across each environment. The boxplots in Figure 7.2 provide aggregated statistics on the sample complexity for different configurations. We observe that GOSPRL consistently meets the sampling requirements faster than o/1-UCRL, as well as suffers from lower variance across runs.

ModEst problem. Finally, we empirically evaluate GOSPRL for the ModEst problem (Section 7.4.2). We compare to the fully online WeightedMaxEnt heuristic that we proposed in Tarbouriech et al. (2020c), whose idea is to weigh the state-action entropy components with an optimistic estimate of the next-state transition variance. In Tarbouriech et al. (2020c) we showed that it performed empirically better than our theoretically-grounded Frank-Wolfe-based approach for ModEst reviewed in Section 6.3.1 (example ③). We now test on the two same environments (NoisyRiverSwim and Wheel) that we had considered in Tarbouriech et al. (2020c) for their high level of stochasticity, as well as on a randomly generated Garnet. To facilitate the comparison, we consider a GOSPRL-for-ModEst algorithm where the sampling requirements are computed using a decreasing error η (see Section F.4 for details). We observe in Figure 7.3 that GOSPRL outperforms the WeightedMaxEnt heuristic.

7.6 Discussion

In this chapter, we introduced the online learning problem of simulating a sampling oracle (Section 7.2) and derived the algorithm GOSPRL with its sample complexity guarantee (Section 7.3). We then illustrated how it can be used to tackle in a unifying fashion a variety of applications without having to design a specific online algorithm for each, while at the same time obtaining improved or novel sample complexity guarantees (Section 7.4). Going forward, we believe that GOSPRL can be used as a competitive off-the-shelf baseline when a new application is introduced.

Our sample complexity bounds for the general sample collection problem and its various applications are worst-case and it would be interesting to derive finer problem-dependent bounds, by for instance building on the recent logarithmic instance-dependent expected regret bounds for SSP of Chen et al. (2021b). Another exciting direction of future investigation can be to extend the scope of the chapter beyond the tabular setting. Handling a continuous state space or linear function approximation requires redefining the notion of reaching a specific state (e.g., via adequate discretization or by considering requirements based on the covariance matrix). Studying the SSP problem (Part I) beyond a finite state space may provide insights. On the more algorithmic side, GOSPRL hinges on knowing the sampling requirement function b_t and deriving a shortest-path policy $\widetilde{\pi}$. Interestingly, we can identify algorithmic counterparts to both modules in deep RL. The computation of $\widetilde{\pi}$ can be entrusted to a goal-conditioned network (using e.g., Andrychowicz et al., 2017), while the specification of b_t can be related to goal-sampling selection mechanisms that elect hard-to-reach (Florensa et al., 2018) or rare (Pong et al., 2020) states as goals.

Chapter 8

SYOG in Reward-Free Resettable MDPs

In this chapter, we investigate the SYOG principle in reward-free resettable MDPs. We bypass the communicating assumption required in Chapter 7 by allowing for a reset action to the starting state s_0. As a means of quantifying the agent's ability to efficiently navigate the vicinity of s_0, we introduce the *multi-goal exploration* problem. The objective is to learn a near-optimal goal-conditioned policy for the (initially unknown) set of goal states that are reachable within a given number of steps in expectation from s_0. We achieve this with nearly minimax-optimal sample complexity by designing a novel goal selection scheme, coined AdaGoal, which leverages a measure of uncertainty of the agent's goal-reaching ability in order to adaptively target goals that are neither too difficult nor too easy. We also analyze AdaGoal with linear function approximation, specifically in *linear mixture* MDPs, whose structural assumption on the transition kernel allows us to eliminate the dependence on the total number of states and actions in the sample complexity. Finally, beyond its strong theoretical guarantees, we anchor AdaGoal in goal-conditioned deep reinforcement learning, both conceptually and empirically, by connecting its idea of selecting "uncertain" goals to maximizing value ensemble disagreement. [1]

Contents

[1] This chapter is based on an article published in the proceedings of the 25^{th} International Conference on Artificial Intelligence and Statistics (AISTATS 2022) (Tarbouriech et al., 2022).

8.1 The Multi-Goal Exploration (MGE) Problem

Let $s_0 \in \mathcal{S}$ be a designated initial state in the unknown reward-free MDP. We measure the performance of a policy in navigating the MDP and define the shortest-path distance as follows.

Definition 8.1. *For any policy $\pi \in \Pi$ and a pair of states $(s, s') \in \mathcal{S}^2$, let $V^\pi(s \to s') \in [0, +\infty]$ be the expected number of steps it takes to reach s' starting from s when executing policy π, i.e.,*

$$V^\pi(s \to s') \triangleq \mathbb{E}\big[\inf\{i \geq 0 : s_{i+1} = s'\} \,|\, s_1 = s, \pi, M\big],$$

where the expectation is w.r.t. the random sequence of states generated by executing π in M starting from state s. (Note that $V^\pi(s \to s')$ corresponds to the SSP value function of policy π in the SSP instance with initial state s, goal state s' and unit cost function, see Part I.)

Moreover, for any state $g \in \mathcal{S}$, let $V^\star(s_0 \to g) \in [0, +\infty]$ be the shortest-path distance from s_0 to g, i.e.,

$$V^\star(s_0 \to g) \triangleq \min_{\pi \in \Pi} V^\pi(s_0 \to g).$$

Finally, let $D_0 \in [0, +\infty]$ and $D \in [0, +\infty]$ be respectively the (possibly infinite) s_0-diameter and diameter of the MDP, i.e.,

$$D_0 \triangleq \max_{g \in \mathcal{S}} V^\star(s_0 \to g), \qquad D \triangleq \max_{s,g} V^\star(s \to g).$$

We denote by $\mathcal{G} \subseteq \mathcal{S}$ the *goal space*, which corresponds to the set of goal states that the agent may condition its goal-conditioned policy on (in the absence of prior knowledge on the goal space we simply set $\mathcal{G} = \mathcal{S}$). In environments with arbitrary dynamics, there may be some goal states in $\mathcal{G}$ that are *too* difficult for the agent to reliably reach in a reasonable number of exploration steps, or even completely unreachable from s_0. Consequently, we consider the high-level objective of *learning an accurate goal-conditioned policy for all the goal states that are reliably reachable* from s_0.

Definition 8.2 (Reliably L-reachable goal states $\mathcal{G}_L$)**.** *For any threshold $L \geq 1$, we define a goal state $g \in \mathcal{G}$ to be reliably L-reachable if $V^\star(s_0 \to g) \leq L$, and we denote by $\mathcal{G}_L$ the set of such goal states, i.e.,*

$$\mathcal{G}_L \triangleq \Big\{g \in \mathcal{G} : V^\star(s_0 \to g) \leq L\Big\}.$$

We thus seek to learn a goal-conditioned policy that is accurate in reaching the goals in $\mathcal{G}_L$. A challenge in solving this objective comes from the fact that the set of goals of interest $\mathcal{G}_L$ is initially *unknown* and it has to be discovered online at the same time as learning their corresponding optimal policy. The threshold L can be interpreted as the user's exploration radius of interest around s_0. In the absence of a pre-specified threshold, the agent can build its own curriculum for L to guide its learning process.

Since in environments with arbitrary dynamics the agent may get stuck in a state without being able to return to s_0, we introduce the following "reset" assumption.[2] In Lemma 8.7 we will formally motivate its role in solving our learning objective.

Assumption 8.3. *The action space contains a known action $a_{\text{reset}} \in \mathcal{A}$ such that $P(s_0|s, a_{\text{reset}}) = 1$ for any state $s \in \mathcal{S}$.*

Consider as input an exploration radius $L \geq 1$, an accuracy level $\varepsilon \in (0, 1]$ and a confidence level $\delta \in (0, 1)$. We now formally define our exploration objective.

Definition 8.4 (Multi-Goal Exploration — MGE)**.** *An algorithm is said to be $(\varepsilon, \delta, L, \mathcal{G})$-PAC for* MGE *if*

- *it stops after some (possibly random) number of exploration steps τ that is less than some polynomial in the relevant quantities $(S, A, L, \varepsilon^{-1}, \log \delta^{-1})$ with probability at least $1 - \delta$,*
- *it returns a set of goal states $\mathcal{X}$ and a set of policies $\{\widehat{\pi}_g\}_{g \in \mathcal{X}}$ such that $\mathbb{P}(\mathcal{C}_1 \cap \mathcal{C}_2) \geq 1 - \delta$, where we define the conditions*

$$\mathcal{C}_1 \triangleq \left\{\forall g \in \mathcal{X},\ V^{\widehat{\pi}_g}(s_0 \to g) - V^{\star}(s_0 \to g) \leq \varepsilon\right\},$$
$$\mathcal{C}_2 \triangleq \left\{\mathcal{G}_L \subseteq \mathcal{X} \subseteq \mathcal{G}_{L+\varepsilon}\right\}.$$

The objective is to build an $(\varepsilon, \delta, L, \mathcal{G})$-PAC algorithm for which the MGE *sample complexity, that is the number of exploration steps τ, is as small as possible.*

Remark 8.5. Since the goal set $\mathcal{G}_L$ is *unknown*, it may not be possible to exactly *identify* it within a reasonable number of exploration steps. Thus we allow the learner to output a larger set $\mathcal{X}$ of candidate goals and policies. Nonetheless, we constrain an $(\varepsilon, \delta, L, \mathcal{G})$-PAC algorithm for MGE to return a set $\mathcal{X}$ that is at most contained in the slightly larger set $\mathcal{G}_{L+\varepsilon}$ (i.e., $\mathcal{X} \subseteq \mathcal{G}_{L+\varepsilon}$).

[2] This setting should be contrasted with the finite-horizon setting, where each policy resets automatically after H steps, or assumptions on the MDP dynamics such as ergodicity or bounded diameter, which guarantee that it is always possible to find a policy navigating between any two states.

Remark 8.6. Consider that $\mathcal{G} = \mathcal{S}$. Then the inclusion $\mathcal{G}_L \subseteq \mathcal{S}$ is an equality if $\mathcal{M}$ is communicating (i.e., $D < +\infty$) and if the (unknown) D_0 is lower or equal to L (note that under Assumption 8.3, $D \leq D_0 + 1$).

MGE vs. reset-free MGE. Lemma 8.7 establishes an exponential separation between MGE and reset-free MGE (i.e., MGE without Assumption 8.3). This motivates the use of Assumption 8.3 to solve our learning objective in a reasonable number of exploration steps.

Lemma 8.7. MGE *can be solved in poly*$(S, L, \varepsilon^{-1}, A)$ *steps. On the other hand, there exists an MDP and a goal space where any algorithm requires at least* $\Omega(D)$ *steps to solve reset-free* MGE, *where* D *is exponentially larger than* $L, S, A, \varepsilon^{-1}$.

MGE lower bound. We now give a worst-case lower bound on the MGE problem (details in Appendix F.1).

Lemma 8.8. *For any algorithm that is* $(\varepsilon, \delta, L, \mathcal{G})$*-PAC for* MGE *for any MDP and goal space* $\mathcal{G}$*, there exists an MDP and a goal space where the algorithm requires, in expectation, at least* $\Omega(L^3 SA\varepsilon^{-2})$ *exploration steps to stop.*

Remark 8.9. We can relate the dependencies in Lemma 8.8 with the lower bound of the time steps needed to identify an ε-optimal policy in both γ-discounted MDPs with a generative model — i.e., $\Omega((1-\gamma)^{-3} SA\varepsilon^{-2})$ (Azar et al., 2013) — and online stationary finite-horizon MDPs — i.e., $\Omega(H^3 SA\varepsilon^{-2})$ (Domingues et al., 2021b). This correspondence is not surprising as L captures the "range" of the MGE problem, similar to the effective horizon $1/(1-\gamma)$ or the horizon H. Also recall that, as mentioned in Chapter 2, both discounted MDPs and finite-horizon MDPs are subclasses of goal-oriented MDPs, i.e., SSP-MDPs.

MGE under linear function approximation. Lemma 8.8 shows that the MGE sample complexity must scale with SA in the worst case, which may be prohibitive in the case of large state-action spaces. This motivates us to further analyze MGE under *linear function approximation*. In particular, we focus on the *linear mixture* MDP setting (Ayoub et al., 2020; Zhou et al., 2021), which assumes that the transition probability is a linear mixture of d signed basis measures.

Definition 8.10 (Linear Mixture MDP, Ayoub et al., 2020; Zhou et al., 2021)**.** *The unknown transition probability P is a linear combination of d signed basis measures $\phi_i(s'|s,a)$, i.e., $P(s'|s,a) \triangleq \sum_{i=1}^{d} \phi_i(s'|s,a)\theta_i^\star$. Meanwhile, for any $V : \mathcal{S} \to [0,1]$, $i \in [d]$, $(s,a) \in \mathcal{S} \times \mathcal{A}$, the summation $\sum_{s' \in \mathcal{S}} \phi_i(s'|s,a)V(s')$ is computable. For simplicity, let $\boldsymbol{\phi} \triangleq [\phi_1, \dots, \phi_d]^\top$, $\boldsymbol{\theta}^\star \triangleq [\theta_1^\star, \dots, \theta_d^\star]^\top$ and $\boldsymbol{\psi}_V(s,a) \triangleq \sum_{s' \in \mathcal{S}} \boldsymbol{\phi}(s'|s,a)V(s)$. Without loss of generality, we assume $\|\boldsymbol{\theta}^\star\|_2 \leq B$, $\|\boldsymbol{\psi}_V(s,a)\|_2 \leq 1$ for all $V : \mathcal{S} \to [0,1]$ and $(s,a) \in \mathcal{S} \times \mathcal{A}$.*

8.2 Our AdaGoal Approach

In Algorithm 8.1, we introduce the common algorithmic structure based on AdaGoal. We use it to design AdaGoal-UCBVI that tackles the MGE problem in tabular MDPs, and AdaGoal-UCRL·VTR that tackles the MGE problem in linear mixture MDPs. Both follow the goal-conditioned structure of SYOG described in Section 6.4. The agent sets a horizon of $H = \Omega(L \log L\varepsilon^{-1})$ and splits its learning interaction in algorithmic episodes of length H. At the beginning of each algorithmic episode, it **(GS)** selects a candidate goal state and **(PE)** deploys an explorative (i.e., optimistic) policy conditioned on this goal for H steps before resetting to s_0. It alternates between these two steps until an adaptive stopping rule is met, at which point the algorithm terminates.[3]

□ **(PE) step.** Goal-conditioned finite-horizon $\mathcal{Q}$-functions (Kaelbling, 1993; Schaul et al., 2015) are maintained optimistically. At each episode k and episode step $h \in [H]$, $\mathcal{Q}_{k,h}(s,a,g)$ approximates (from below) the number of (expected) steps required to reach any goal $g \in \mathcal{G}$ starting from any state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$ and executing the optimal goal-reaching policy for $H - h$ steps. Intuitively, the $\mathcal{Q}$-functions will gradually increase, more so for goal states that the agent struggles to reach. This is essentially done by initializing the $\mathcal{Q}$-functions optimistically (i.e., at 0), considering that the cost (i.e., negative reward) is $+1$ (resp. 0) per time step if the conditioned goal is not reached (resp. reached), and carefully subtracting an exploration bonus to maintain optimism. Given a goal $g_k \in \mathcal{G}$ selected at the beginning of episode k, the **(PE)** step simply amounts to deploying an explorative policy conditioned on g_k, that is, a policy $\pi_{k,h}$ that greedily minimizes the current $\mathcal{Q}$-functions, i.e., $\pi_{k,h}(s) \in \arg\min_{a \in \mathcal{A}} \mathcal{Q}_{k,h}(s,a,g_k)$.

□ **(GS) step.** To elect a relevant sequence of candidate goals $(g_k)_{k \geq 1}$, we introduce AdaGoal, an adaptive goal selection scheme based on a simple constrained optimization problem. It

[3] Indeed recall from Definition 8.4 that the algorithm must adaptively decide when to terminate its learning interaction.

Algorithm 8.1: AdaGoal-based algorithmic structure. Blue text denotes AdaGoal-UCBVI specific steps and purple text denotes AdaGoal-UCRL·VTR specific steps.

Input: Exploration radius $L \geq 1$, accuracy level $\varepsilon \in (0, 1]$, confidence level $\delta \in (0, 1)$.
Input: Number of states S, number of actions A.
Input: Dimension of feature mapping d, bound B on ℓ_2-norm of $\boldsymbol{\theta}^\star$.
Input: Goal space $\mathcal{G} \subseteq \mathcal{S}$ (otherwise set $\mathcal{G} = \mathcal{S}$).
Set as horizon $H \triangleq \lceil 5(L+2)\log\left(10(L+2)/\varepsilon\right)/\log(2)\rceil$.
Initialize: algorithmic episode index $k = 1$, distance estimates $\mathcal{D}_1(g) = \mathbb{1}[g \neq s_0]$, error estimates $\mathcal{E}_1(g) = H\mathbb{1}[g \neq s_0]$, goal-conditioned finite-horizon Q-values $\mathcal{Q}_{1,h}(s, a, g) = \mathbb{1}[s \neq g]$, for all $(g, s, a, h) \in \mathcal{G} \times \mathcal{S} \times \mathcal{A} \times [H]$.

while stopping rule (8.1) is not met **do**

i Goal selection rule:

Select as goal state

$$g_k \in \underset{g \in \mathcal{G}}{\arg\max} \quad \mathcal{E}_k(g) \qquad \text{subject to:} \quad \mathcal{D}_k(g) \leq L.$$

ii Policy execution rule:

For a duration of H steps, run the optimistic goal-conditioned policy $\pi^k_{g_k}$ such that at step h, $\pi^k_{g_k,h}(s) \in \arg\min_{a \in \mathcal{A}} \mathcal{Q}_{k,h}(s, a, g_k)$ (note that $\mathcal{D}_k(g_k) = \min_{a \in \mathcal{A}} \mathcal{Q}_{k,1}(s_0, a, g_k)$).

Then execute action a_{reset} and increment episode index $k \mathrel{+}= 1$.

iii Update and check stopping rule:

Update estimates $\mathcal{Q}_k$, $\mathcal{D}_k$, $\mathcal{E}_k$ according to (F.4), (F.5), (F.6) using samples collected so far.

Update estimates $\mathcal{Q}_k$, $\mathcal{D}_k$, $\mathcal{E}_k$ according to (F.26), (F.27), (F.28) using samples collected so far.

Stop the algorithm if

$$\max_{g \in \mathcal{G}:\, \mathcal{D}_k(g) \leq L} \mathcal{E}_k(g) \;\leq\; \varepsilon. \tag{8.1}$$

end

Let $\kappa \triangleq \inf\left\{k \in \mathbb{N} : \max_{g \in \mathcal{G}:\, \mathcal{D}_k(g) \leq L} \mathcal{E}_k(g) \leq \varepsilon\right\}$.

Output: Goal states $\mathcal{X}_\kappa \triangleq \{g \in \mathcal{G} : \mathcal{D}_\kappa(g) \leq L\}$, and for every $g \in \mathcal{X}_\kappa$, a deterministic, non-stationary policy $\widehat{\pi}_g$ that at time step i and state s selects action a according to:

$$\widehat{\pi}_g(a|s, i) \triangleq \begin{cases} \arg\min_{a \in \mathcal{A}} \mathcal{Q}_{\kappa,h}(s, a, g) \\ \quad \text{if } i \equiv h \pmod{H+1} \text{ for } h \in [H], \\ a_{\text{reset}} \\ \quad \text{if } i \equiv 0 \pmod{H+1}. \end{cases}$$

relies on the agent's ability to compute two types of goal-conditioned quantities for any goal $g \in \mathcal{G}$ and episode $k \geq 1$:

- a distance estimate from s_0 to g, denoted by $\mathcal{D}_k(g)$;
- an error of estimating this distance, denoted by $\mathcal{E}_k(g)$.

Conveniently, the distance estimates can be simply instantiated as $\mathcal{D}_k(g) \triangleq \min_{a \in \mathcal{A}} \mathcal{Q}_{k,1}(s_0, a, g)$. Formally, we require the following properties of $\mathcal{D}$ and $\mathcal{E}$ to hold (with high probability):

- **Property 1:** $\mathcal{D}$ is an *optimistic* distance estimate, i.e.,

$$\mathcal{D}_k(g) \leq \mathcal{D}^\star_H(g), \quad \forall k \geq 1, \forall g \in \mathcal{G},$$

  where $\mathcal{D}^\star_H(g) \triangleq \min_\pi \mathbb{E}\big[\omega_\pi(s_0 \to g) \wedge H\big]$ denotes the shortest-path distance from s_0 to g truncated at H steps. Note that $\mathcal{D}^\star_H(g) \in [0, H]$ and that $\lim_{H \to +\infty} \mathcal{D}^\star_H(g) = V^\star(s_0 \to g)$.
- **Property 2:** $\mathcal{E}$ is an *upper bound* on the following error

$$|\mathcal{D}^\star_H(g) - \mathcal{D}_k(g)| \leq \mathcal{E}_k(g), \quad \forall k \geq 1, \forall g \in \mathcal{G}.$$

Given these two goal-conditioned quantities, AdaGoal selects at episode k a candidate goal that solves the following constrained optimization problem:

$$\boxed{\begin{aligned} g_k \in \operatorname*{arg\,max}_{g \in \mathcal{G}} \quad & \mathcal{E}_k(g) && \text{(8.2a)} \\ \text{subject to:} \quad & \mathcal{D}_k(g) \leq L. && \text{(8.2b)} \end{aligned}}$$

Interpretation. The agent sequentially selects a goal state with highest error $\mathcal{E}$ among those whose distance estimate $\mathcal{D}$ is not too large. If the agent is confident that a goal g is either too easy or too hard to reach, it will assign a low error $\mathcal{E}(g)$. As a result, the objective function in (8.2a) adaptively samples goal states on the frontier of the learning process. The constraint in (8.2b), although it is not required for the final sample complexity result, further tightens the goal selection process. Indeed, for any $k \geq 1$, let

$$\mathcal{X}_k \triangleq \{g \in \mathcal{G} : \mathcal{D}_k(g) \leq L\}, \qquad \varepsilon_k \triangleq \max_{g \in \mathcal{X}_k} \mathcal{E}_k(g).$$

Then, if as a warm-up we take the limit $H \to +\infty$, injecting Properties 1 and 2 in (8.2a-8.2b) entails that

$$\mathcal{G}_L \subseteq \mathcal{X}_k \subseteq \mathcal{G}_{L+\varepsilon_k}. \tag{8.3}$$

We thus see that the constraint in (8.2b) does not remove valid goals in $\mathcal{G}_L$ from the set $\mathcal{X}_k$ of candidate goal states to sample. Second, decreasing ε_k has the dual impact of making the set

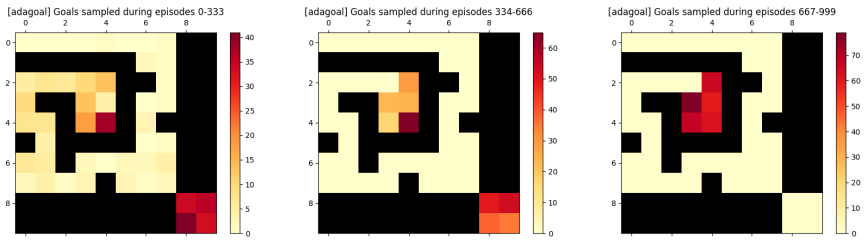


Figure 8.1 – Goal sampling frequency of AdaGoal-UCBVI over 1000 episodes of length $H = 50$ (with $L = 40$). The grid-world has $S = 52$ states, starting state $s_0 = (0, 0)$ (top left), $A = 5$ actions (4 cardinal ones and a_{reset}). The 4 states of the bottom right room can only be accessed from s_0 by any cardinal action with probability $\eta = 0.001$ (their associated $V^\star(s_0 \to \cdot)$ thus scale with η^{-1}).

of candidates goals $\mathcal{X}_k$ closer to $\mathcal{G}_L$ and improving their distance estimates, which motivates the goal selection scheme in (8.2a). In practice, we consider a *finite* truncation H (line 5 in Algorithm 8.1), thus we need to account for the bias $\rho_g \triangleq V^\star(s_0 \to g) - \mathcal{D}_H^\star(g)$, which can be arbitrarily large for goals g that are hard or impossible to reach. Fortunately, our AdaGoal strategy will be able to gradually discard such states, hence the final MGE sample complexity will not pay for such terms. In fact, we will later see that the choice of horizon $H = \Omega(L \log L\varepsilon^{-1})$ ensures that $\rho_g = O(\varepsilon)$ for all the (unknown) goal states of interest $g \in \mathcal{G}_L$.

Choice of $\mathcal{Q}, \mathcal{D}, \mathcal{E}$. A key algorithmic design is how to build and update the goal-conditioned $\mathcal{Q}$-functions and the estimates $\mathcal{D}$ and $\mathcal{E}$. In Appendices F.2 and F.3, we will carefully construct them with exact bonus-based estimates for both tabular MDPs and linear mixture MDPs. As suggested by the algorithms' names, the estimates of the former are inspired from BPI-UCBVI (Ménard et al., 2021), an algorithm for best policy identification in finite-horizon tabular MDPs, while those of the latter are inspired from UCRL-VTR (Ayoub et al., 2020), an algorithm for regret minimization in finite-horizon linear mixture MDPs. Since all our estimates are optimistic, we see that Algorithm 8.1 relies on the principle of optimism in the face of uncertainty *both* for the goal selection and the policy execution. Finally, in Section 8.5, we propose a way to instantiate $\mathcal{Q}, \mathcal{D}, \mathcal{E}$ for a practical implementation in deep RL.

□ **Adaptive stopping rule.** At the end of each algorithmic episode, the estimates are updated using the samples collected so far, and the algorithm checks whether a stopping rule (8.1) based on AdaGoal is triggered, in which case it terminates. This occurs when the errors $\mathcal{E}$ of all the goal states that meet the AdaGoal constraint (8.2b) are below the prescribed accuracy level ε. These states then form the set of *candidate goal states* output by Algorithm 8.1, along with their associated optimistic goal-reaching policies.

Empirical validation. In Figure 8.1 (see Section F.4 for details), we empirically study the sequence of goals selected by AdaGoal-UCBVI during learning. We design a two-room grid-

world with a very small probability of reaching the second room. We see that AdaGoal is able to discard the states from the second room and target as goals the states in the first room that are "furthest" away from s_0, which effectively correspond to the fringe of what the agent can reliably reach.

8.3 Sample Complexity Guarantees

□ **Guarantee for AdaGoal-UCBVI.** We first bound the MGE sample complexity of AdaGoal-UCBVI. For simplicity we consider that $\mathcal{G} = \mathcal{S}$, i.e., the goal space spans the entire state space (the results trivially extend to any $\mathcal{G} \subseteq \mathcal{S}$).

Theorem 8.11. AdaGoal-UCBVI *is* $(\varepsilon, \delta, L, \mathcal{S})$*-PAC for* MGE *and, with probability at least* $1-\delta$*, for* $\varepsilon \in (0, 1/S]$ *its* MGE *sample complexity is of order*[4] $\widetilde{O}(L^3 S A \varepsilon^{-2})$.

Lemma 8.8 and Theorem 8.11 imply that the MGE sample complexity of AdaGoal-UCBVI is nearly minimax optimal for small enough ε and up to logarithmic terms.

In the absence of a pre-specified exploration radius L, the agent can build its own curriculum for L (i.e., design a sequence of increasing L's) to guide its learning. In this case, the total sample complexity is (up to a logarithmic factor) the same of AdaGoal-UCBVI run with the final value of L, as stated below.

Corollary 8.12. *The successive execution of* AdaGoal-UCBVI *for an increasing sequence* $L \in \{2, 2^2, ..., 2^f\}$ *with* $f \in \mathbb{N}^*$ *is* $(\varepsilon, \delta, L_f, \mathcal{S})$*-PAC for* MGE *and, with probability at least* $1-\delta$*, for* $\varepsilon \in (0, 1/S]$ *its* MGE *sample complexity is of order* $\widetilde{O}(L_f^3 S A \varepsilon^{-2})$*, where* $L_f = 2^f$.

Finally, we can investigate the special case where $\mathcal{M}$ is communicating and the objective is to learn an ε-optimal goal-conditioned policy for *every* goal state. Since the s_0-diameter D_0 is unknown, we can use as an initial subroutine the GOSPRL algorithm of Chapter 7 to compute an estimate $\widetilde{D}$ such that $D_0 \leq \widetilde{D} \leq 2D_0$ in $\widetilde{O}(D_0^3 S^2 A)$ time steps (see Lemma E.18). Then we can execute AdaGoal-UCBVI with $L = \widetilde{D}$, which leads to the following guarantee.

Corollary 8.13. *Assume that the MDP* $\mathcal{M}$ *has a finite and unknown* s_0*-diameter* D_0*. Then the above strategy is* $(\varepsilon, \delta, D_0, \mathcal{S})$*-PAC for* MGE *and, with probability at least* $1-\delta$*, for* $\varepsilon \in (0, 1/S]$*, its* MGE *sample complexity is of order* $\widetilde{O}(D_0^3 S A \varepsilon^{-2})$.

We can compare Corollary 8.13 with the approach based on GOSPRL described in Chapter 7 to tackle the different although related problem of cost-free goal-free exploration in *communicating* MDPs (Section 7.4.3), where the agent must find an ε-optimal goal-conditioned policy for any arbitrary starting state, goal state and positive cost function. For small enough ε, the bound of Lemma 7.10 in the unit-cost case scales as $\widetilde{O}(D^4\Gamma SA\varepsilon^{-2})$, where Γ is the branching factor which in the worst case is S. We see that Corollary 8.13 improves over that result by a factor D_0 as well as a factor $\Gamma \leq S$. Although Lemma 7.10 considers a more demanding cost-free objective (i.e., for any positive cost function), it is unable to avoid its superlinear dependence in S when instantiated in the current scenario of unit cost functions, since the design of GOSPRL is to estimate uniformly well the transition kernel.

□ **Guarantee for AdaGoal-UCRL·VTR.** We now bound the MGE sample complexity of Algorithm 8.1 in linear mixture MDPs (Definition 8.10). Since the state space $\mathcal{S}$ may be large, we consider that the known goal space is in all generality a subset of it, i.e., $\mathcal{G} \subseteq \mathcal{S}$, where $G \triangleq |\mathcal{G}|$ denotes the cardinality of the goal space.

Theorem 8.14. *In linear mixture MDPs, for $\varepsilon \in (0, 1]$,* AdaGoal-UCRL·VTR *is $(\varepsilon, \delta, L, \mathcal{G})$-PAC for* MGE *and, with probability at least $1 - \delta$, its* MGE *sample complexity is of order*[5] $\widetilde{O}\left(L^4 d^2 \varepsilon^{-2}\right)$*, where d is the dimension of the feature mapping.*

To the best of our knowledge, Theorem 8.14 yields the first goal-oriented PAC guarantee with linear function approximation. The algorithm's choice of $\mathcal{E}$, $\mathcal{D}$, $\mathcal{Q}$ relies on two regression-based goal-conditioned estimators, one standard "value-targeted" estimator inspired from UCRL-VTR (Ayoub et al., 2020) and one novel "error-targeted" estimator, see Section F.3. We expect that the bound of Theorem 8.14 can be refined using tighter Bernstein-based estimates, for instance inspired from UCRL-VTR$^+$ (Zhou et al., 2021), which we leave as future work. Note that the $\widetilde{O}$ notation in Theorem 8.14 contains a $\log(G)$ factor (which appears when performing a union bound argument over all goals $g \in \mathcal{G}$), and the computational complexity of AdaGoal-UCRL·VTR scales with G (since the algorithm maintains goal-conditioned estimates). We point out that here we still consider that the MDP has a finite number of states S. This is to be expected given the way a goal is currently modeled (at the granular level of states), independently of how large the state space is, where it may be very hard to visit specific states. Learning in single- or multi-goal RL beyond a finite state space is an interesting direction of future investigation. Note that, as mentioned in Section 5.6, existing works on single-goal exploration (i.e., SSP regret minimization) with linear function approximation (Vial et al., 2021; Min et al., 2021; Chen et al., 2021b) also assume that the state space is finite.

8.4 Analysis Overview

The proofs of Theorems 8.11 and 8.14 (see Appendices F.2 and F.3) are decomposed in the same following key steps.

▷ ***Key step ①: Optimism and gap bounds.*** We prove that Properties 1 and 2 hold with high probability, i.e., **(i)** the quantities $\mathcal{D}$ are optimistic estimates of the optimal goal-conditioned finite-horizon value functions $\mathcal{D}_H^\star$ and **(ii)** the quantities $\mathcal{E}$ are valid upper bounds to the goal-conditioned finite-horizon gaps.

▷ ***Key step ②: Bounding the cumulative gap bounds.*** We make explicit a function $f_{\mathcal{M}}$ (depending on the MDP $\mathcal{M}$) that is strictly decreasing in K with $f_{\mathcal{M}}(K) \to_{K\to\infty} 0$, such that with high probability,

$$\sum_{k=1}^{K} \mathcal{D}_H^\star(g_k) - \mathcal{D}_k(g_k) \leq \sum_{k=1}^{K} \mathcal{E}_k(g_k) \leq K \cdot f_{\mathcal{M}}(K), \tag{8.4}$$

for any number of algorithmic episodes K. Specifically, we establish that

$$f_{\mathcal{M}}(K) = \begin{cases} \widetilde{O}\left(\sqrt{KH^2SA} + H^2S^2A\right) & \text{for AdaGoal-UCBVI,} \\ \widetilde{O}\left(\sqrt{KH^3d^2} + H^2d^{3/2}\right) & \text{for AdaGoal-UCRL·VTR.} \end{cases}$$

(8.4) resembles a *no-regret* property of the exploration algorithm that receives as input the sequence of goals $(g_k)_{k\geq 1}$ prescribed by AdaGoal and performs the **(PE)** step. Indeed, intuitively, the aim of the **(PE)** step is to improve the estimation of $\mathcal{D}_k(g_k)$ and make it closer to $\mathcal{D}_H^\star(g_k)$, i.e., to decrease the error $\mathcal{E}_k(g_k)$.

▷ ***Key step ③: Bounding the sample complexity.*** To bound κ the episode index at which Algorithm 8.1 terminates, we combine (8.4) and the termination condition (8.1) to simultaneously lower and upper bound (with high probability) the cumulative errors $\mathcal{E}$ as

$$\varepsilon \cdot (\kappa - 1) \leq \sum_{k=1}^{\kappa-1} \mathcal{E}_k(g_k) \leq (\kappa - 1) \cdot f_{\mathcal{M}}(\kappa - 1).$$

Inverting this functional inequality in κ yields that κ is finite and bounded as

$$\kappa \leq f_{\mathcal{M}}^{-1}(\varepsilon) + 2. \tag{8.5}$$

The sample complexity is $(H+1)\cdot\kappa$ with $H=\widetilde{O}(L)$, thus AdaGoal-UCBVI (resp. AdaGoal-UCRL·VTR) stops in $\widetilde{O}(L^3SA\varepsilon^{-2}+L^3S^2A\varepsilon^{-1})$ (resp. $\widetilde{O}(L^4d^2\varepsilon^{-2})$) time steps, with high probability.

▷ ***Key step ④: Connecting to the original* MGE *objective.*** The key remaining step is to prove that the MGE objective is indeed fulfilled.

Remark 8.15. Algorithm 8.1 relies on a finite-horizon construction, with algorithmic episodes of length H. This relates to the reduction of SSP to finite-horizon studied in some SSP regret minimization works (Cohen et al., 2021; Chen and Luo, 2021), which as reviewed in Part I rely on the idea that an SSP problem can be approximated by a finite-horizon problem if the horizon is large enough w.r.t. $T_\star$, the optimal policy's expected hitting time to the goal starting from any state. Two main differences arise in our MGE setting: (i) first, in these works, the goal state is fixed throughout learning and $T_\star$ is assumed known, whereas we need to deal with goal selection and find the relevant goals of interest while having to discard those that are poorly reachable or unreachable. (ii) Second, these works ensure that the *empirical* goal-reaching performance of the algorithm's non-stationary policy *over the whole learning interaction* is good enough (by definition of the regret objective in SSP). As such, they do not show that the *expected* performance of some *candidate* policy is good enough (i.e., the SSP value function is small enough) – in fact, they do not even explicitly prove that the executed policies are proper. The latter property may actually not be possible to obtain since standard regret-to-PAC conversion may not work in SSP as mentioned in Section 3.4. In our MGE objective, the key difference lies in the availability of the reset action (Assumption 8.3), as we will now see.

Our analysis builds on the following reasoning: given a goal state in $\mathcal{G}_{L+\varepsilon}$, we can find a candidate policy with near-optimal goal-reaching behavior (i.e., SSP value function) by: (i) first computing a near-optimal policy $\widetilde{\pi}$ in the finite-horizon reduction (using the stopping rule (8.1)), (ii) and then expanding $\widetilde{\pi}$ into an infinite-horizon policy via the reset action every H time steps to get our desired candidate policy.

Now, importantly, the above reasoning *only holds* for the goals in $\mathcal{G}_{L+\varepsilon}$, which is an *unknown* set. This is where our AdaGoal strategy comes into the picture, as it provides a simple and computable *sufficient condition* for a goal to belong to $\mathcal{G}_{L+\varepsilon}$.

Lemma 8.16. *With probability at least $1-\delta$, if a goal state $g\in\mathcal{G}$ satisfies $\mathcal{D}_k(g)\le L$ and $\mathcal{E}_k(g)\le\varepsilon$ for an episode $k\ge 1$, then $g\in\mathcal{G}_{L+\varepsilon}$.*

We are now ready to put everything together and prove that AdaGoal-UCBVI and AdaGoal-UCRL·VTR are (ε,δ,L)-PAC for MGE. The candidate goal states are $\mathcal{X}_\kappa\triangleq\{g\in\mathcal{S}:\mathcal{D}_\kappa(g)\le L\}$,

with candidate policies $\widehat{\pi}_g \triangleq (\pi_g^{\kappa+1})^{|H}$. In what follows we reason with high probability. Property 1 ensures that $\mathcal{G}_L \subseteq \mathcal{X}_\kappa$, while Lemma 8.16 entails that $\mathcal{X}_\kappa \subseteq \mathcal{G}_{L+\varepsilon}$. Finally, for any $g \in \mathcal{X}_\kappa$, combining Property 2 and the termination condition (8.1) gives that $\pi_g^{\kappa+1}$ is $\varepsilon/9$-optimal in $\overline{\mathcal{M}}_{g,H}$. As a result, the translation from the finite-horizon to goal-oriented objective (which holds since $g \in \mathcal{G}_{L+\varepsilon}$ and by choice of the horizon $H = \Omega(L \log L\varepsilon^{-1})$, see Lemma F.16) yields that $V_g^{\widehat{\pi}_g}(s_0) \leq V_g^\star(s_0) + \varepsilon$, i.e., $\widehat{\pi}_g$ is ε-optimal for the original SSP objective. This concludes the proofs of Theorems 8.11 and 8.14.

8.5 Operationalizing AdaGoal in Deep RL

In this section, we present a way to operationalize the AdaGoal idea of targeting goals with high uncertainty on the agent's ability in reaching them. We show it can be implemented similar to the deep RL algorithm of Zhang et al. (2020b), and we investigate an aspect that was not considered in the latter paper, which pertains to the capability of AdaGoal to adapt to an unknown target goal set ($\mathcal{G}_L$) given a goal set that is possibly misspecified ($\mathcal{G}$).

First, we notice that $\mathcal{Q}$ and $\mathcal{D}$ in Algorithm 8.1 can be learned in practice with a goal-conditioned value-based neural network (Schaul et al., 2015). Meanwhile, the errors $\mathcal{E}$ can be approximated by the disagreement between an ensemble of goal-conditioned Q-functions. Interestingly, this approach has already been investigated in the deep GC-RL algorithm VDS (Zhang et al., 2020b), which considers a similar goal-proposal module to prioritize goals that maximize the epistemic uncertainty of the Q-function of the current policy, in order to sample goals at the frontier of the set of goals that the agent is able to reach.

Specifically, we take an ensemble $\{Q_j\}_{1\leq j\leq J}$ of J randomly initialized goal-conditioned Q-functions and we instantiate $\mathcal{E}(g)$ as the standard deviation of the ensemble's Q_j values conditioned on goal g. Since computing the maximum over $g \in \mathcal{G}$ in (8.2a) is expensive if the goal space $\mathcal{G}$ is large, the procedure is replaced by first uniformly sampling a set of candidate goals $\{g^{(n)}\}_{n=1}^N \subset \mathcal{G}$, and then selecting a goal $g^{(n)}$ with probability

$$q_n \triangleq \frac{\mathcal{E}(g^{(n)})}{\sum_{n'=1}^N \mathcal{E}(g^{(n')})}, \qquad \mathcal{E}(g) \triangleq \operatorname*{std}_{1\leq j\leq J}\{\min_{a\in\mathcal{A}} Q_j(s_0, a, g)\}.$$

Moreover, approximating $H \approx L$ renders the constraint in (8.2b) always valid so it can be omitted. Hence, this approximation of AdaGoal exactly recovers the goal sampling scheme of Zhang et al. (2020b), which pairs it with Hindsight Experience Replay (HER, Andrychowicz et al., 2017) that performs *uniform* goal sampling.

We consider the multi-goal environments of FetchPush and FetchPickAndPlace, which are sparse-reward simulated robotic manipulation tasks from OpenAI Gym (Plappert et al., 2018). We empirically compare the performance of such an adaptive goal selection with the

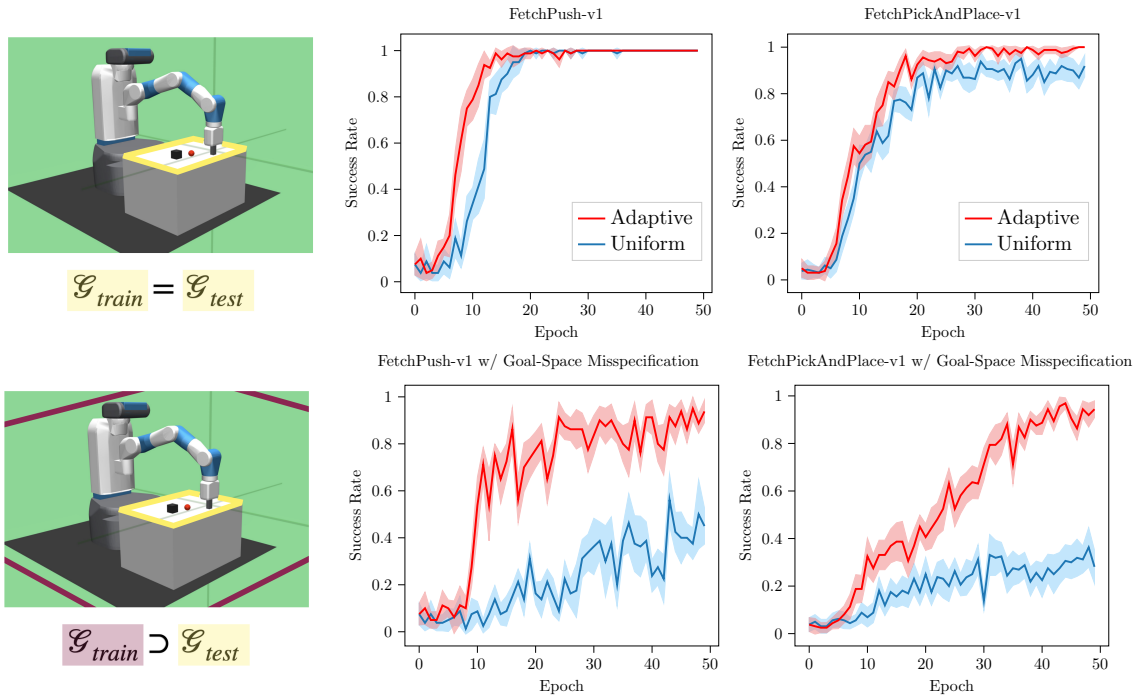


Figure 8.2 – Success rate evaluated on $\mathcal{G}_{\text{test}}$ with the latest policy trained on $\mathcal{G}_{\text{train}}$. The shaded region represents confidence over 5 random seeds. The adaptive goal sampling scheme improves the learning performance over the uniform sampling of HER. This is especially the case in the presence of goal-space misspecification (bottom row), where the training goal space $\mathcal{G}_{\text{train}}$ (delimited in purple) is larger than the test goal space $\mathcal{G}_{\text{test}}$ (delimited in yellow).

performance of HER's uniform goal selection (see Section F.5 for implementation details). We observe in Figure 8.2 (top row) that the adaptive goal sampling scheme outperforms the uniform one of HER, which is consistent with the results of Zhang et al. (2020b).

In the above experimental set-up (which is in fact considered by most deep GC-RL works), the goal space $\mathcal{G}_{\text{train}}$ seen at train time is the same as the goal space $\mathcal{G}_{\text{test}}$ on which the agent is evaluated at test time, i.e., the white rectangular table. In the language of the previous sections, by relating $\mathcal{G}_{\text{train}} \leftrightarrow \mathcal{G}$ and $\mathcal{G}_{\text{test}} \leftrightarrow \mathcal{G}_L$, this means that the environment is considered communicating and $\mathcal{G} = \mathcal{G}_L$. However, in some cases, there may be some *misspecification* in the goal space seen during the learning interaction. This may occur if the agent is unaware of the goals of interest, in which case we have that $\mathcal{G}_L \subsetneq \mathcal{G}$, where we recall that $\mathcal{G}_L$ is a priori unknown. We design an experiment to model this scenario by translating the x-y range of $\mathcal{G}_{\text{train}}$ by a factor of $\lambda \geq 1$. Specifically, denoting by (x_0, y_0, z_0) the center of the table and letting $r \triangleq 0.15$, we leave $\mathcal{G}_{\text{test}}$ unchanged yet we expand $\mathcal{G}_{\text{train}} \supset \mathcal{G}_{\text{test}}$ as

$$\mathcal{G}_{\text{test}} \triangleq \{(x_0 + \mathcal{U}(-r, r), y_0 + \mathcal{U}(-r, r), z_0)\},$$
$$\mathcal{G}_{\text{train}} \triangleq \{(x_0 + \mathcal{U}(-\lambda r, \lambda r), y_0 + \mathcal{U}(-\lambda r, \lambda r), z_0)\},$$

where $\lambda_{\text{FetchPush}} = 10$, $\lambda_{\text{FetchPickAndPlace}} = 5$, and where $\mathcal{U}(a, b)$ denotes the continuous uniform distribution in $[a, b]$. In this scenario, Figure 8.2 (bottom row) shows that an adaptive goal sampling scheme is particularly pertinent. Intuitively, it enables to discard the set of goals $\mathcal{G}_{\text{train}} \setminus \mathcal{G}_{\text{test}}$ that cannot be reached and thus hinder learning when the agent conditions its behavior on them. This empirically corroborates AdaGoal's (theoretically established) ability to adapt to an unknown target goal set ($\mathcal{G}_L$) given a goal set that is possibly misspecified ($\mathcal{G}$).

Chapter 9

Incremental SYOG in Reward-Free Resettable MDPs

In this chapter, we carry on our investigation of the SYOG principle in reward-free resettable MDPs initiated in Chapter 8, by restricting our attention to the *incrementally* reliably reachable states around the reference state s_0, as defined by Lim and Auer (2012). We strenghten their learning objective for incremental autonomous exploration and derive the first algorithm able to learn an incrementally near-optimal goal-conditioned policy. Interestingly, in contrast to Chapter 8, the incremental focus enables to obtain a sample complexity bound that depends only logarithmically on the total number of states. [1]

Contents

[1] This chapter is based on an article published in the proceedings of the 33^{th} Conference on Neural Information Processing Systems (NeurIPS 2020) (Tarbouriech et al., 2020b).

9.1 Incremental Autonomous Exploration

In this chapter, we build on the setting formalized by Lim and Auer (2012) and introduce a more challenging objective for incremental autonomous exploration. We consider that the reward-free MDP has a possibly large state space $\mathcal{S}$, with a known upper bound S on its cardinality, i.e., $|\mathcal{S}| \leq S$.[2] Throughout this chapter, we consider that Assumption 8.3 holds, which we recall below.

Assumption 8.3. *The action space contains a known action $a_{\text{reset}} \in \mathcal{A}$ such that $P(s_0|s, a_{\text{reset}}) = 1$ for any state $s \in \mathcal{S}$.*

We make explicit the states where a policy π takes action a_{reset} in the following definition.

Definition 9.1 (Policy restricted on a subset)**.** *For any $\mathcal{S}' \subseteq \mathcal{S}$, a policy π is restricted on $\mathcal{S}'$ if $\pi(s) = a_{\text{reset}}$ for any $s \notin \mathcal{S}'$. We denote by $\Pi(\mathcal{S}')$ the set of policies restricted on $\mathcal{S}'$.*

We now introduce the following restricted optimality measure.

Definition 9.2 (Restricted optimality)**.** *For any policy π and state $s \in \mathcal{S}$, recall from Definition 8.1 that $V^\pi(s_0 \to s)$ denotes the expected hitting time from s_0 to s following π. Then for any subset $\mathcal{S}' \subseteq \mathcal{S}$, we denote by*

$$V^\star_{\mathcal{S}'}(s_0 \to s) \triangleq \min_{\pi \in \Pi(\mathcal{S}')} V^\pi(s_0 \to s),$$

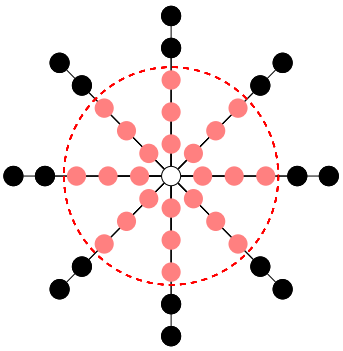
the length of the shortest path to s, restricted to policies resetting to s_0 from any state outside $\mathcal{S}'$.

Note that by definition of $V^\star$ in Definition 8.1, it holds that $V^\star = V^\star_{\mathcal{S}}$. Moreover, Assumption 8.3 entails the following simple optimality ordering.

Lemma 9.3. *For any two sets $\mathcal{X}, \mathcal{Y}$ such that $\mathcal{X} \subseteq \mathcal{Y} \subseteq \mathcal{S}$ and any state $s \in \mathcal{X}$, it holds that $V^\star_{\mathcal{X}}(s_0 \to s) \geq V^\star_{\mathcal{Y}}(s_0 \to s)$.*

As in Chapter 8, the objective of the learning agent is to *control efficiently* the environment in the *vicinity* of the reference state s_0. We say that a state s is controlled if the agent can reliably navigate to it from s_0, that is, there exists an effective SSP policy from s_0 to s.

[2] Lim and Auer (2012) originally considered a countable, possibly infinite state space; however this leads to a technical issue in the analysis of UcbExplore (acknowledged by the authors via personal communication and explained in Section G.2.3), which disappears by considering only finite state spaces.



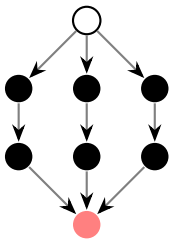


Figure 9.1 – Two environments where the starting state s_0 is in white. *Left:* Each transition between states is deterministic and depicted with an edge. *Right:* Each transition from s_0 to the first layer is *equiprobable* and the transitions in the successive layers are deterministic. If we set $L = 3$, then the states belonging to $\mathcal{S}_L$ are colored in red. As the right figure illustrates, L-controllability is not necessarily linked to a notion of distance between states and an L-controllable state may be achieved by traversing states that are not L-controllable themselves.

Definition 9.4 (L-controllable states)**.** *Given a reference state s_0, we say that a state s is L-controllable if there exists a policy π such that $V^\pi(s_0 \to s) \leq L$. The set of L-controllable states is then defined as*

$$\mathcal{S}_L \triangleq \Big\{ s \in \mathcal{S} : \min_{\pi \in \Pi} V^\pi(s_0 \to s) \leq L \Big\}. \tag{9.1}$$

We illustrate the concept of controllable states in Figure 9.1 for $L = 3$. Interestingly, in the right figure, the black states are not L-controllable. In fact, there is no policy that can directly choose which one of the black states to reach. On the other hand, the red state, despite being in some sense *further* from s_0 than the black states, *does* belong to S_L. In general, there is a crucial difference between the existence of a *random* realization where a state s is reached from s_0 in less than L steps (i.e., black states) and the notion of *L-controllability*, which means that there exists a policy that consistently reaches the state in a number of steps less or equal than L on average (i.e., red state). This explains the choice of the term *controllable* over *reachable*, since a state s is often said to be reachable if there is a policy π with a non-zero probability to eventually reach it, which is a weaker requirement.

Unfortunately, Lemma 8.8 shows that in order to discover all the states in $\mathcal{S}_L$, the learner may require a number of exploration steps that directly scales with the total number of states S, which may be large.[3] To avoid this dependence, we follow Lim and Auer (2012) and constrain the learner to focus on the set of *incrementally controllable* states.

[3] This result can be complemented by those of Lim and Auer (2012) that showed that in order to discover all the states in $\mathcal{S}_L$, the learner may require a number of exploration steps that is *exponential* in L or $|\mathcal{S}_L|$. Intuitively, this negative result is due to the fact that the minimum in Equation (9.1) is over the set of all possible policies, including those that may traverse states that are not in $\mathcal{S}_L$. We refer the reader to Lim and Auer (2012, Section 2.1) for a more formal and complete characterization of this negative result.

Definition 9.5 (Incrementally controllable states $\mathcal{S}_L^{\rightarrow}$)**.** *Let $\prec$ be some partial order on $\mathcal{S}$. The set $\mathcal{S}_L^{\prec}$ of states controllable in L steps w.r.t. $\prec$ is defined inductively as follows:*

- *the initial state s_0 belongs to $\mathcal{S}_L^{\prec}$ by definition,*
- *if there exists a policy π restricted on $\{s' \in \mathcal{S}_L^{\prec} : s' \prec s\}$ with $V^{\pi}(s_0 \to s) \leq L$, then $s \in \mathcal{S}_L^{\prec}$.*

The set $\mathcal{S}_L^{\rightarrow}$ of incrementally L-controllable states is defined as

$$\mathcal{S}_L^{\rightarrow} \triangleq \bigcup_{\prec} \mathcal{S}_L^{\prec},$$

where the union is over all possible partial orders.

By way of illustration, in Figure 9.1 for $L = 3$, it holds that $\mathcal{S}_L^{\rightarrow} = \mathcal{S}_L$ in the left figure, whereas $\mathcal{S}_L^{\rightarrow} = \{s_0\} \neq \mathcal{S}_L$ in the right figure. Indeed, while the red state is L-controllable, it requires traversing the black states, which are not L-controllable.

AX Objectives. We are now ready to formalize two alternative objectives for *Autonomous eXploration* (AX) in MDPs.

Definition 9.6 (AX_L sample complexity, Lim and Auer, 2012)**.** *Fix any exploration radius $L \geq 1$, error threshold $\varepsilon > 0$ and confidence level $\delta \in (0, 1)$. The sample complexity $\mathcal{C}_{\text{AX}_L}(\mathfrak{A}, L, \varepsilon, \delta)$ is defined as the number of time steps required by a learning algorithm $\mathfrak{A}$ to identify a set $\mathcal{K} \supseteq \mathcal{S}_L^{\rightarrow}$ such that with probability at least $1 - \delta$, it has learned a set of policies $\{\pi_s\}_{s \in \mathcal{K}}$ that verifies the following requirement*

$$\forall s \in \mathcal{K},\ V^{\pi_s}(s_0 \to s) \leq L + \varepsilon.$$

Definition 9.7 ($\text{AX}^\star$ sample complexity)**.** *Fix any exploration radius $L \geq 1$, error threshold $\varepsilon > 0$ and confidence level $\delta \in (0, 1)$. The sample complexity $\mathcal{C}_{\text{AX}^\star}(\mathfrak{A}, L, \varepsilon, \delta)$ is defined as the number of time steps required by a learning algorithm $\mathfrak{A}$ to identify a set $\mathcal{K} \supseteq \mathcal{S}_L^{\rightarrow}$ such that with probability at least $1 - \delta$, it has learned a set of policies $\{\pi_s\}_{s \in \mathcal{K}}$ that verifies the following requirement*

$$\forall s \in \mathcal{K},\ V^{\pi_s}(s_0 \to s) \leq V^{\star}_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) + \varepsilon.$$

AX_L is the original objective introduced by Lim and Auer (2012) and it requires the agent to discover all the incrementally L-controllable states as fast as possible.[4] At the end of the learning process, for each state $s \in \mathcal{S}_L^{\rightarrow}$ the agent should return a policy that can reach s from s_0 in at most L steps (in expectation). Unfortunately, this may correspond to a rather poor performance in practice. Consider a state $s \in \mathcal{S}_L^{\rightarrow}$ such that $V^{\star}_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) \ll L$, i.e., the shortest path between s_0 to s following policies restricted on $\mathcal{S}_L^{\rightarrow}$ is much smaller than L. Satisfying AX_L only guarantees that a policy reaching s in L steps is found. On the other hand, objective $\text{AX}^\star$ is more demanding, as it requires learning a near-optimal shortest-path policy for each state in $\mathcal{S}_L^{\rightarrow}$. Since $V^{\star}_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) \leq L$ and the gap between the two quantities may be arbitrarily large, especially for states close to s_0 and far from the fringe of $\mathcal{S}_L^{\rightarrow}$, $\text{AX}^\star$ is a significantly tighter objective than AX_L and it is thus preferable in practice.

Remark 9.8. In the special case where $\mathcal{S}_L^{\rightarrow} = \mathcal{S}_L$, the $\text{AX}^\star$ objective of Definition 9.7 is equivalent to the MGE objective of Definition 8.4 and can thus be effectively solved by the AdaGoal-UCBVI algorithm introduced in Chapter 8, at the expense of having a sample complexity bound directly scaling with the total number of states S (Theorem 8.11). As motivated in the introduction of this chapter, we aim here to remove this dependence. Interestingly, note that the special case of $\mathcal{S}_L^{\rightarrow} = \mathcal{S}_L$ holds when the environment is *deterministic* (i.e., when the next state s_{t+1} is uniquely determined by the current state s_t and action a_t), which covers many standard control or robotic environments (Plappert et al., 2018). This observation motivates the practical relevance of the incremental autonomous exploration objective, despite its quite intrincate definition at first glance.

We say that an exploration algorithm solves an AX problem if its sample complexity $\mathcal{C}_{\text{AX}}(\mathfrak{A}, L, \varepsilon, \delta)$ in Definition 9.6 or 9.7 is polynomial in $|\mathcal{K}|$, A, L, ε^{-1} and $\log(S)$. Notice that requiring a logarithmic dependence on the size of $\mathcal{S}$ is crucial but nontrivial, since the overall state space may be large and we do not want the agent to waste time trying to reach states that are not L-controllable. The dependence on the (algorithmic-dependent and random) set $\mathcal{K}$ can be always replaced using the upper bound $|\mathcal{K}| \leq |\mathcal{S}_{L+\varepsilon}^{\rightarrow}|$, which is implied with high probability by both AX_L and $\text{AX}^\star$ conditions. Finally, notice that the error threshold $\varepsilon > 0$ has a two-fold impact on the performance of the algorithm. First, ε defines the largest set $\mathcal{S}_{L+\varepsilon}^{\rightarrow}$ that could be returned by the algorithm: the larger ε, the bigger the set. Second, as ε increases, the quality (in terms of controllability and navigational precision) of the output policies worsens w.r.t. the shortest-path policy restricted on $\mathcal{S}_L^{\rightarrow}$.

Lim and Auer (2012) designed the UcbExplore algorithm to tackle the AX_L objective. It comes with the following sample complexity bound.

[4] Note that we translated the condition of Lim and Auer (2012) of a relative error of $L\varepsilon$ to an absolute error of ε, to align it with the common formulation of sample complexity in RL.

Proposition 9.9 (Lim and Auer, 2012, Theorem 8).

$$\mathcal{C}_{\mathrm{AX}_L}(\textsc{UcbExplore}, L, \varepsilon, \delta) = \widetilde{O}\left(\frac{L^6 |\mathcal{S}^{\rightarrow}_{L+\varepsilon}| A}{\varepsilon^3}\right).$$

Nonetheless, UcbExplore is unable to tackle the more challenging AX* objective. In the following section, we propose an algorithm that does so.

9.2 The DisCo Algorithm

In this section, we introduce the algorithm DisCo — short for `Discover and Control` — designed to tackle the AX* objective. It is detailed in Algorithm 9.1. Similar to UcbExplore, it maintains a set $\mathcal{K}$ of "controllable" states and a set $\mathcal{U}$ of states that are considered "uncontrollable" *so far*. A state s is tagged as controllable when a policy to reach s in at most $L+\varepsilon$ steps (in expectation from s_0) has been found with high confidence, and we denote by π_s such policy. The states in $\mathcal{U}$ are states that have been discovered as potential members of $\mathcal{S}^{\rightarrow}_L$, but the algorithm has yet to produce a policy to control any of them in less than $L+\varepsilon$ steps. The algorithm stores an estimate of the transition model and it proceeds through rounds, which are indexed by k and incremented whenever a state in $\mathcal{U}$ gets transferred to the set $\mathcal{K}$, i.e., when the transition model reaches a level of accuracy sufficient to compute a policy to control one of the states encountered before. We denote by $\mathcal{K}_k$ (resp. $\mathcal{U}_k$) the set of controllable (resp. uncontrollable) states at the beginning of round k. DisCo stops at a round K when it can confidently claim that all the remaining states outside of $\mathcal{K}_K$ cannot be L-controllable.

At each round, the algorithm uses all samples observed so far to build an estimate of the transition model denoted by $\widehat{P}(s'|s,a) = N(s,a,s')/N(s,a)$, where $N(s,a)$ and $N(s,a,s')$ are counters for state-action and state-action-next state visitations. Each round is divided into two phases. The first is a *sample collection* phase. At the beginning of round k, the agent collects additional samples until $n_k \triangleq \phi(\mathcal{K}_k)$ samples are available at each state-action pair in $\mathcal{K}_k \times \mathcal{A}$ (step ①). A key challenge lies in the careful (and adaptive) choice of the allocation function ϕ, which we report in the statement of Theorem 9.11 (see Equation (G.8) in Section G.1.4 for its exact definition). Importantly, the incremental construction of $\mathcal{K}_k$ entails that sampling at each state $s \in \mathcal{K}_k$ can be done efficiently. In fact, for all $s \in \mathcal{K}_k$ the agent has already confidently learned a policy π_s to reach s in at most $L+\varepsilon$ steps on average (see how such policy is computed in the second phase). The generation of transitions (s,a,s') for $(s,a) \in \mathcal{K}_k \times \mathcal{A}$ achieves two objectives at once. First, it serves as a discovery step, since all observed next states s' not in $\mathcal{U}_k$ are added to it — in particular this guarantees sufficient exploration at the fringe (or border)

Algorithm 9.1: Algorithm DisCo

Input: Actions $\mathcal{A}$, initial state s_0, confidence parameter $\delta \in (0,1)$, error threshold $\varepsilon > 0$, $L \geq 1$ and (possibly adaptive) allocation function $\phi : \mathcal{P}(\mathcal{S}) \to \mathbb{N}$ (where $\mathcal{P}(\mathcal{S})$ denotes the power set of $\mathcal{S}$).

Initialize $k \triangleq 0$, $\mathcal{K}_0 \triangleq \{s_0\}$, $\mathcal{U}_0 \triangleq \{\}$ and a restricted policy $\pi_{s_0} \in \Pi(\mathcal{K}_0)$.

Set $\varepsilon \triangleq \min\{\varepsilon, 1\}$ and `continue` $\triangleq$ `True`.

while `continue` **do**

  Set $k \mathrel{+}= 1$. `//new round`

  `//` ① **`Sample collection on`** $\mathcal{K}$

  For each $(s,a) \in \mathcal{K}_k \times \mathcal{A}$, execute policy π_s until the total number of visits $N_k(s,a)$ to (s,a) satisfies $N_k(s,a) \geq n_k \triangleq \phi(\mathcal{K}_k)$. For each $(s,a) \in \mathcal{K}_k \times \mathcal{A}$, add $s' \sim P(\cdot|s,a)$ to $\mathcal{U}_k$ if $s' \notin \mathcal{K}_k$.

  `//` ② **`Restriction of candidate states`** $\mathcal{U}$

  Compute transitions $\widehat{P}_k(s'|s,a)$ and $\mathcal{W}_k \triangleq \left\{ s' \in \mathcal{U}_k : \exists (s,a) \in \mathcal{K}_k \times \mathcal{A}, \widehat{P}_k(s'|s,a) \geq \frac{1-\varepsilon/2}{L} \right\}$.

  if $\mathcal{W}_k$ is empty **then**

    Set `continue` $\triangleq$ `False`. `//condition STOP1`

  else

    `//` ③ **`Computation of the optimistic policies on`** $\mathcal{K}$

    for each state $s' \in \mathcal{W}_k$ **do**

      Compute $(\widetilde{u}_{s'}, \widetilde{\pi}_{s'}) \triangleq$ $\text{OVI}_{\text{SSP}}\left(\texttt{goal} = s', \texttt{states} = \mathcal{K}_k \cup \{x\}, \texttt{samples} = N_k, \texttt{costs} = 1, \texttt{precision}\ \gamma \triangleq \frac{\varepsilon}{6L}\right)$ (see Algorithm G.1).

    Let $s^\dagger \triangleq \arg\min_{s \in \mathcal{W}_k} \widetilde{u}_s(s_0)$ and $\widetilde{u}^\dagger \triangleq \widetilde{u}_{s^\dagger}(s_0)$.

    if $\widetilde{u}^\dagger > L$ **then**

      Set `continue` $\triangleq$ `False`. `//condition STOP2`

    else

      `//` ④ **`State transfer from`** $\mathcal{U}$ **`to`** $\mathcal{K}$

      Set $\mathcal{K}_{k+1} \triangleq \mathcal{K}_k \cup \{s^\dagger\}$, $\mathcal{U}_{k+1} \triangleq \mathcal{U}_k \setminus \{s^\dagger\}$ and $\pi_{s^\dagger} \triangleq \widetilde{\pi}_{s^\dagger}$.

`//` ⑤ **`Policy consolidation: computation on the final set`** $\mathcal{K}$

Set $K \triangleq k$.

for each state $s \in \mathcal{K}_K$ **do**

  Compute $(\_, \pi_s) \triangleq$ $\text{OVI}_{\text{SSP}}\left(\texttt{goal} = s, \texttt{states} = \mathcal{K}_k \cup \{x\}, \texttt{samples} = N_k, \texttt{costs} = 1, \texttt{precision}\ \gamma \triangleq \frac{\varepsilon}{6L}\right)$.

Output: the states s in $\mathcal{K}_K$ and their corresponding policy π_s.

of the set $\mathcal{K}_k$. Second, it improves the accuracy of the model p in the states in $\mathcal{K}_k$, which is essential in computing near-optimal policies and thus fulfilling the AX* condition.

The second phase does not require interacting with the environment and it focuses on the *computation of optimistic policies*. The agent begins by significantly restricting the set of candidate states in each round to alleviate the computational complexity of the algorithm. Namely, among all the states in $\mathcal{U}_k$, it discards those that do not have a high probability of belonging to $\mathcal{S}_L^{\rightarrow}$ by considering a restricted set $\mathcal{W}_k \subseteq \mathcal{U}_k$ (step ②). In fact, if the estimated probability $\widehat{P}_k$ of reaching a state $s \in \mathcal{U}_k$ from any of the controllable states in $\mathcal{K}_k$ is lower than $(1-\varepsilon/2)/L$,

then no shortest-path policy restricted on $\mathcal{K}_k$ could get to s from s_0 in less than $L+\varepsilon$ steps on average. Then for each state s' in $\mathcal{W}_k$, DisCo computes an optimistic policy restricted on $\mathcal{K}_k$ to reach s'. Formally, for any candidate state $s' \in \mathcal{W}_k$, we define the induced SSP-MDP M'_k with goal state s' as follows (cf. Part I).

Definition 9.10. *We define the SSP-MDP $M'_k \triangleq \langle \mathcal{S}, \mathcal{A}'_k(\cdot), c'_k, P'_k \rangle$ with goal state s', where the action space is such that $\mathcal{A}'_k(s) = \mathcal{A}$ for all $s \in \mathcal{K}_k$ and $\mathcal{A}'_k(s) = \{a_{\text{reset}}\}$ otherwise (i.e., we focus on policies restricted on $\mathcal{K}_k$). The cost function is such that for all $a \in \mathcal{A}$, $c'_k(s', a) = 0$, and for any $s \neq s'$, $c'_k(s, a) = 1$. The transition model is $P'_k(s'|s', a) = 1$ and $P'_k(\cdot|s, a) = P(\cdot|s, a)$ otherwise.*[5]

The solution of M'_k is the optimal SSP policy from s_0 to s' restricted on $\mathcal{K}_k$. Since P'_k is unknown, DisCo cannot compute the exact solution of M'_k, but instead, it executes optimistic value iteration (OVI_{SSP}) for SSP (Rosenberg et al., 2020) to obtain a value function $\widetilde{u}_{s'}$ and its associated greedy policy $\widetilde{\pi}_{s'}$ restricted on $\mathcal{K}_k$ (see Section G.1.1 for more details).

The agent then chooses a candidate goal state $s^\dagger$ for which the value $\widetilde{u}^\dagger \triangleq \widetilde{u}_{s^\dagger}(s_0)$ is the smallest. This step can be interpreted as selecting the optimistically most promising new state to control. Two cases are possible. If $\widetilde{u}^\dagger \leq L$, then $s^\dagger$ is added to $\mathcal{K}_k$ (step ④), since the accuracy of the model estimate on the state-action space $\mathcal{K}_k \times \mathcal{A}$ guarantees that the policy $\widetilde{\pi}_{s^\dagger}$ is able to reach the state $s^\dagger$ in less than $L+\varepsilon$ steps in expectation with high probability (i.e., $s^\dagger$ is incrementally $(L+\varepsilon)$-controllable). Otherwise, we can guarantee that $\mathcal{S}_L^{\rightarrow} \subseteq \mathcal{K}_k$ with high probability. In the latter case, the algorithm terminates and, using the current estimates of the model, it recomputes an optimistic shortest-path policy π_s restricted on the final set $\mathcal{K}_K$ for each state $s \in \mathcal{K}_K$ (step ⑤). This policy consolidation step is essential to identify near-optimal policies restricted on the final set $\mathcal{K}_K$ (and thus on $\mathcal{S}_L^{\rightarrow}$): indeed the expansion of the set of the so far controllable states may alter and refine the optimal goal-reaching policies restricted on it.

Computational Complexity. Note that algorithmically, we do not need to define M'_k (Definition 9.10) over the whole state space $\mathcal{S}$ as we can limit it to $\mathcal{K}_k \cup \{s'\}$, i.e., the candidate state s' and the set $\mathcal{K}_k$ of so far controllable states. As shown in Theorem 9.11, this set can be significantly smaller than $\mathcal{S}$. In particular this implies that the computational complexity of the value iteration algorithm used to compute the optimistic policies is independent from S (see Section G.1.9 for more details).

9.3 Sample Complexity Analysis

We now present our main result: a sample complexity guarantee for DisCo for the AX* objective, which directly implies that AX_L is also satisfied.

Theorem 9.11. *There exists an absolute constant $\alpha > 0$ such that for any $L \geq 1$, $\varepsilon \in (0, 1]$, and $\delta \in (0, 1)$, if we set the allocation function ϕ as*

$$\phi : \mathcal{X} \to \alpha \cdot \left(\frac{L^4 \widehat{\Theta}(\mathcal{X})}{\varepsilon^2} \log^2 \left(\frac{LSA}{\varepsilon\delta} \right) + \frac{L^2 |\mathcal{X}|}{\varepsilon} \log \left(\frac{LSA}{\varepsilon\delta} \right) \right), \tag{9.2}$$

with

$$\widehat{\Theta}(\mathcal{X}) \triangleq \max_{(s,a) \in \mathcal{X} \times \mathcal{A}} \Big(\sum_{s' \in \mathcal{X}} \sqrt{\widehat{P}(s'|s,a)(1 - \widehat{P}(s'|s,a))} \Big)^2,$$

then the algorithm DisCo *(Algorithm 9.1) satisfies the following sample complexity bound for* AX*

$$\mathcal{C}_{\text{AX}^\star}(\text{DisCo}, L, \varepsilon, \delta) = \widetilde{O} \left(\frac{L^5 \Gamma_{L+\varepsilon} S_{L+\varepsilon} A}{\varepsilon^2} + \frac{L^3 S_{L+\varepsilon}^2 A}{\varepsilon} \right), \tag{9.3}$$

where $S_{L+\varepsilon} \triangleq |\mathcal{S}^{\rightarrow}_{L+\varepsilon}|$ and

$$\Gamma_{L+\varepsilon} \triangleq \max_{(s,a) \in \mathcal{S}^{\rightarrow}_{L+\varepsilon} \times \mathcal{A}} \|\{P(s'|s,a)\}_{s' \in \mathcal{S}^{\rightarrow}_{L+\varepsilon}}\|_0 \leq S_{L+\varepsilon}$$

is the maximal support of the transition probabilities $P(\cdot|s,a)$ restricted to the set $\mathcal{S}^{\rightarrow}_{L+\varepsilon}$.

Given the definition of AX*, Theorem 9.11 implies that DisCo

1. terminates after $\mathcal{C}_{\text{AX}^\star}(\text{DisCo}, L, \varepsilon, \delta)$ time steps,
2. discovers a set of states $\mathcal{K} \supseteq \mathcal{S}^{\rightarrow}_{L}$ with $|\mathcal{K}| \leq S_{L+\varepsilon}$,
3. and for each $s \in \mathcal{K}$ outputs a policy π_s which is ε-optimal w.r.t. policies restricted on $\mathcal{S}^{\rightarrow}_{L}$, i.e., $V^{\pi_s}(s_0 \to s) \leq V^{\star}_{\mathcal{S}^{\rightarrow}_{L}}(s_0 \to s) + \varepsilon$.

Note that Equation (9.3) displays only a *logarithmic* dependence on S, the total number of states. This property on the sample complexity of DisCo, along with its S-independent computational complexity, is significant when the state space $\mathcal{S}$ grows large w.r.t. the unknown set of interest $\mathcal{S}^{\rightarrow}_{L}$.

9.3.1 Proof Sketch of Theorem 9.11

While the complete proof is reported in Section G.1, we now provide the main intuition behind the analysis of DisCo.

State Transfer from $\mathcal{U}$ to $\mathcal{K}$ (step ④). Let us focus on a round k and a state $s^\dagger \in \mathcal{U}_k$ that gets added to $\mathcal{K}_k$. For clarity we remove in the notation the round k, goal state $s^\dagger$ and starting state s_0. We denote by v and $\widetilde{v}$ the value functions of the candidate policy $\widetilde{\pi}$ in the true and optimistic model respectively, and by $\widetilde{u}$ the quantity w.r.t. which $\widetilde{\pi}$ is optimistically greedy. We aim to prove that $s^\dagger \in \mathcal{S}^{\rightarrow}_{L+\varepsilon}$ (with high probability). The main chain of inequalities underpinning the argument is

$$v \leq |v - \widetilde{v}| + \widetilde{v} \overset{\text{(a)}}{\leq} \frac{\varepsilon}{2} + \widetilde{v} \overset{\text{(b)}}{\leq} \frac{\varepsilon}{2} + \widetilde{u} + \frac{\varepsilon}{2} \overset{\text{(c)}}{\leq} L + \varepsilon, \tag{9.4}$$

where (c) is guaranteed by algorithmic construction and (b) stems from the chosen level of value iteration accuracy. Inequality (a) has the flavor of a simulation lemma for SSP (see Lemma 2.14), by relating the SSP value function of a same policy between two models (the true one and the optimistic one). Importantly, when restricted to $\mathcal{K}$ these two models are close in virtue of the algorithmic design which enforces the collection of a minimum amount of samples at each state-action pair of $\mathcal{K} \times \mathcal{A}$, denoted by n. Specifically, we obtain that

$$|v - \widetilde{v}| = \widetilde{O}\Big(\sqrt{\frac{L^4 \Gamma_{\mathcal{K}}}{n}} + \frac{L^2 |\mathcal{K}|}{n}\Big), \qquad \text{with} \quad \Gamma_{\mathcal{K}} \triangleq \max_{(s,a) \in \mathcal{K} \times \mathcal{A}} \|\{P(s'|s,a)\}_{s' \in \mathcal{K}}\|_0 \leq |\mathcal{K}|.$$

Note that $\Gamma_{\mathcal{K}}$ is the branching factor restricted to the set $\mathcal{K}$. Our choice of n given in Equation (9.2) is then dictated to upper bound the above quantity by $\varepsilon/2$ in order to satisfy inequality (a).

Termination of the Algorithm. Since $\mathcal{S}^{\rightarrow}_L$ is *unknown*, we have to ensure that none of the states in $\mathcal{S}^{\rightarrow}_L$ are "missed". As such, we prove that with overwhelming probability, we have $\mathcal{S}^{\rightarrow}_L \subseteq \mathcal{K}_K$ when the algorithm terminates at a round denoted by K. There remains to justify the final near-optimal guarantee w.r.t. the set of policies $\Pi(\mathcal{S}^{\rightarrow}_L)$. Leveraging that step ⑤ recomputes the policies $(\pi_s)_{s \in \mathcal{K}_K}$ on the final set $\mathcal{K}_K$, we establish the following chain of inequalities

$$v \leq |v - \widetilde{v}| + \widetilde{v} \overset{\text{(a)}}{\leq} \frac{\varepsilon}{2} + \widetilde{v} \overset{\text{(b)}}{\leq} \frac{\varepsilon}{2} + \widetilde{u} + \frac{\varepsilon}{2} \overset{\text{(c)}}{\leq} V^\star_{\mathcal{K}_K} + \varepsilon \overset{\text{(d)}}{\leq} V^\star_{\mathcal{S}^{\rightarrow}_L} + \varepsilon, \tag{9.5}$$

where (a) and (b) are as in Equation (9.4), (c) leverages optimism and (d) stems from the inclusion $\mathcal{S}^{\rightarrow}_L \subseteq \mathcal{K}_K$.

Sample Complexity Bound. The choice of allocation function ϕ in Equation (9.2) bounds n_K which is the total number of samples required at each state-action pair in $\mathcal{K}_K \times \mathcal{A}$. We then compute a high-probability bound ψ on the time steps needed to collect a given sample, and show that it scales as $\widetilde{O}(L)$. Since the sample complexity is solely induced by the sample collection phase (step ①), it can be bounded by the quantity $\psi\, n_K |\mathcal{K}_K| A$. Putting everything together yields the bound of Theorem 9.11.

9.3.2 Comparison with UcbExplore (Lim and Auer, 2012)

We start recalling the critical distinction that DisCo succeeds in tackling problem AX*, while UcbExplore fails to do so. Nonetheless, in the following we show that even if we restrict our attention to AX$_\mathrm{L}$, for which UcbExplore is designed, DisCo can yield a better sample complexity in many cases. Proposition 9.9 shows that the sample complexity of UcbExplore is linear in $S_{L+\varepsilon}$, while for DisCo the dependence is somewhat worse. In the main-order term $\widetilde{O}(1/\varepsilon^2)$ of Equation (9.3), the bound depends linearly on $S_{L+\varepsilon}$ but also grows with the branching factor $\Gamma_{L+\varepsilon}$, which is not the "global" branching factor but denotes the number of possible next states in $\mathcal{S}^{\rightarrow}_{L+\varepsilon}$ starting from $\mathcal{S}^{\rightarrow}_{L+\varepsilon}$. While in general we only have $\Gamma_{L+\varepsilon} \leq S_{L+\varepsilon}$, in many practical domains (e.g., robotics, user modeling), each state can only transition to a small number of states, i.e., we often have $\Gamma_{L+\varepsilon} = O(1)$ as long as the dynamics is not too "chaotic". While DisCo does suffer from a quadratic dependence on $S_{L+\varepsilon}$ in the second term of order $\widetilde{O}(1/\varepsilon)$, we notice that for any $S_{L+\varepsilon} \leq L^3\varepsilon^{-2}$ the bound of DisCo is still preferable. Furthermore, since for $\varepsilon \to 0$, $S_{L+\varepsilon}$ tends to S_L, the condition is always verified for small enough ε.

Compared to DisCo, the sample complexity of UcbExplore is worse in both ε and L. As stressed in Section 9.1, the better dependence on ε both improves the quality of the output goal-reaching policies as well as reduces the number of incrementally $(L+\varepsilon)$-controllable states returned by the algorithm. It is interesting to investigate why the bound of UcbExplore (Proposition 9.9) inherits a $\widetilde{O}(\varepsilon^{-3})$ dependence. As reviewed in Section G.2, UcbExplore alternates between two phases of state discovery and policy evaluation. The optimistic policies computed by UcbExplore solve a *finite-horizon problem* (with horizon set to H_{Ucb}). However, minimizing the expected time to reach a target state is intrinsically an SSP problem, which is exactly what DisCo leverages. By computing policies that solve a finite-horizon problem (note that it resets every H_{Ucb} time steps), UcbExplore sets the horizon to $H_{\text{Ucb}} \triangleq \lceil L + L^2\varepsilon^{-1} \rceil$, which leads to a policy-evaluation phase with sample complexity scaling as $\widetilde{O}(H_{\text{Ucb}}\varepsilon^{-2}) = \widetilde{O}(\varepsilon^{-3})$. Since the rollout budget of $\widetilde{O}(\varepsilon^{-3})$ is hard-coded into the algorithm, the dependence on ε of UcbExplore's sample complexity cannot be improved by a more refined analysis; instead a different algorithmic approach is required such as the one employed by DisCo.

9.3.3 Goal-Free Cost-Free Incremental Exploration on $\mathcal{S}^{\rightarrow}_{L}$ with DisCo

A compelling advantage of DisCo is that it achieves an accurate estimation of the environment's dynamics restricted to the unknown subset of interest $\mathcal{S}^{\rightarrow}_{L}$. In contrast to UcbExplore which needs to restart its sample collection from scratch whenever L, ε or some transition costs change, DisCo can thus be *robust* to changes in such problem parameters. At the end of its exploration phase in Algorithm 9.1, DisCo is able to perform zero-shot planning to solve other tasks restricted on $\mathcal{S}^{\rightarrow}_{L}$, such as cost-sensitive ones. Indeed in the following we show how the

DisCo agent is able to compute an $\varepsilon/c_{\min}$-optimal policy for *any* SSP problem on $\mathcal{S}_L^{\rightarrow}$ with goal state $s \in \mathcal{S}_L^{\rightarrow}$ (i.e., s is absorbing and zero-cost) and cost function lower bounded by $c_{\min} > 0$.

Corollary 9.12. *There exists an absolute constant $\beta > 0$ such that for any $L \geq 1$, $\varepsilon \in (0, 1]$ and $c_{\min} \in (0, 1]$ verifying $\varepsilon \leq \beta \cdot (L\, c_{\min})$, with probability at least $1 - \delta$, for whatever goal state $s \in \mathcal{S}_L^{\rightarrow}$ and whatever cost function c in $[c_{\min}, 1]$, DisCo can compute (after its exploration phase, without additional environment interaction) a policy $\widehat{\pi}_{s,c}$ whose SSP value function $V^{\widehat{\pi}_{s,c}}$ verifies*

$$V^{\widehat{\pi}_{s,c}}(s_0 \to s) \leq V^{\star}_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) + \frac{\varepsilon}{c_{\min}},$$

where $V^{\pi}(s_0 \to s) \triangleq \mathbb{E}\left[\sum_{t=1}^{\tau_\pi(s_0 \to s)} c(s_t, \pi(s_t)) \mid s_1 = s_0\right]$ is the SSP value function of policy π and $V^{\star}_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) \triangleq \min_{\pi \in \Pi(\mathcal{S}_L^{\rightarrow})} V^{\pi}(s_0 \to s)$ is the optimal SSP value function restricted on $\mathcal{S}_L^{\rightarrow}$.

It is interesting to compare Corollary 9.12 with the reward-free exploration framework introduced by Jin et al. (2020) in finite-horizon MDPs and reviewed in Section 6.3.2. At a high level, the result in Corollary 9.12 can be seen as a counterpart of Jin et al. (2020) beyond finite-horizon problems, specifically in the goal-conditioned setting. Compared to the GOSPRL-based result of Lemma 7.10, Corollary 9.12 does not require the MDP to be communicating and its sample complexity bound does not scale with the (possibly infinite) diameter. While the parameter L defines the horizon of interest for DisCo, resetting after every L steps (as in finite-horizon) would prevent the agent to identify L-controllable states and lead to poor performance. This explains the distinct technical tools used: while Jin et al. (2020) executes finite-horizon no-regret algorithms, DisCo deploys SSP policies restricted on the set of states that it "controls" so far. Algorithmically, both approaches seek to build accurate estimates of the transitions on a specific (unknown) state space of interest: the so-called "significant" states within H steps for Jin et al. (2020), and the incrementally L-controllable states $S_L^{\rightarrow}$ for DisCo. Bound-wise, the cost-sensitive AX* problem inherits the critical role of the minimum cost $c_{\min}$ in SSP problems (see Part I), which is reflected in the accuracy of Corollary 9.12 scaling inversely with $c_{\min}$. Another interesting element of comparison is the dependence on the size of the state space. While the algorithm of Jin et al. (2020) is robust w.r.t. states that can be reached with very low probability, it still displays a *polynomial* dependence on the total number of states S. On the other hand, DisCo has only a *logarithmic* dependence on S, while it directly depends on the number of $(L + \varepsilon)$-controllable states, which shows that DisCo effectively adapts to the state space of interest and it ignores all other states. This result is significant since not only $S_{L+\varepsilon}$ can be arbitrarily smaller than S, but also because the set $\mathcal{S}_{L+\varepsilon}^{\rightarrow}$ itself is initially unknown to the algorithm.

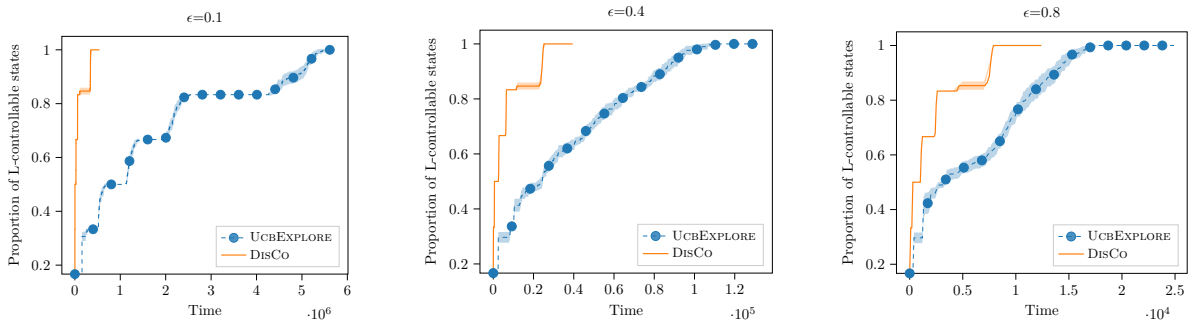


Figure 9.2 – Proportion of the incrementally L-controllable states identified by DisCo and UcbExplore in a confusing chain domain for $L = 4.5$ and $\varepsilon \in \{0.1, 0.4, 0.8\}$. Values are averaged over 50 runs.

9.4 Numerical Simulation

In this section, we provide the first evaluation of algorithms in the incremental autonomous exploration setting. In the implementation of both DisCo and UcbExplore, we remove the logarithmic and constant terms for simplicity. We also boost the empirical performance of UcbExplore in various ways, for example by considering confidence intervals derived from the empirical Bernstein inequality (see Azar et al., 2017) as opposed to the Hoeffding inequality as done by Lim and Auer (2012). We refer the reader to Section G.3 for details on the algorithmic configurations and on the environments considered.

We compare the sample complexity empirically achieved by DisCo and UcbExplore. Figure 9.2 depicts the time needed to identify all the incrementally L-controllable states when $L = 4.5$ for different values of ε, on a confusing chain domain. Note that the sample complexity is achieved soon after, when the algorithm can confidently discard all the remaining states as non-controllable (it is reported in Table G.1 of Section G.3). We observe that DisCo outperforms UcbExplore for any value of ε. In particular, the gap in performance increases as ε decreases, which matches the theoretical improvement in sample complexity from $\widetilde{O}(\varepsilon^{-3})$ for UcbExplore to $\widetilde{O}(\varepsilon^{-2})$ for DisCo. On a second environment — the combination lock problem introduced by Azar et al. (2012) — we notice that DisCo again outperforms UcbExplore, as shown in Section G.3.

Another important feature of DisCo is that it targets the tighter objective $\mathrm{AX}^\star$, whereas UcbExplore is only able to fulfill objective AX_L and may therefore elect suboptimal policies. In Section G.3 we show empirically that, as expected theoretically, this directly translates into higher-quality goal-reaching policies recovered by DisCo.

9.5 Discussion and Bibliographical Remarks

In this chapter, we strengthened the objective of incremental autonomous exploration proposed by Lim and Auer (2012) and derived DisCo, the first algorithm able to learn an incrementally near-optimal goal-conditioned policy for all states in $\mathcal{S}_L^{\rightarrow}$ (i.e., the AX* objective). Due to its model-based nature, DisCo can in fact readily compute an $\varepsilon/c_{\min}$-optimal policy for *any* cost-sensitive SSP problem defined on the $\mathcal{S}_L^{\rightarrow}$ with minimum cost $c_{\min}$. This result serves as a goal-conditioned counterpart to the reward-free exploration framework proposed by Jin et al. (2020) for the finite-horizon setting. A significant feature of the incremental setting is that the sample complexity of DisCo (and UcbExplore) depends only logarithmically on the total number of states.

The main drawback of the algorithmic design of DisCo is its exhaustive and poorly adaptive goal selection scheme to collect relevant samples, which results in the extra $\Gamma_{L+\varepsilon}^{\rightarrow}$ dependence and suboptimal L dependence in the AX* sample complexity bound. A natural future direction to design an algorithm with improved theoretical performance could be to build on the more refined and adaptive AdaGoal algorithmic scheme of Chapter 8.

Note that the AX* setting introduces subtle technical challenges when trying to tighten the theoretical dependences, as recently argued by Cai et al. (2022). Indeed one should build concentration inequalities on $(\widehat{P}_{s,a} - P_{s,a})V_{\mathcal{K}}^\star(\cdot \to g)$ for the goals $g \in \mathcal{K}$, instead of $||\widehat{P}_{s,a} - P_{s,a}||_1$ as done in the analysis of DisCo. However, the set $\mathcal{K}$ of "controllable" states is dependent on the samples collected, therefore $\widehat{P}_{s,a}$ and $V_{\mathcal{K}}^\star$ are interdependent, which makes the analysis more challenging that in the non-incremental case of Chapter 8 where we consider the algorithmic-independent $V^\star$ quantity. Cai et al. (2022) introduced new algorithms with improved sample complexity bounds (e.g., $\widetilde{O}(L^3 S_{2L} A \varepsilon^{-2})$), as well as a lower bound of $\Omega(L^3 S_L A \varepsilon^{-2})$. They connect the AX* problem to a tabular multi-goal SSP problem, which is a special case of the MGE problem that we have introduced in Chapter 8 when all the states of the MDP are reachable within L steps in expectation from the initial state. In this sub-problem, they achieve the same nearly minimax sample complexity of $\widetilde{O}(L^3 S A \varepsilon^{-2})$ as AdaGoal-UCBVI.

There remain some interesting directions for future investigation in the incremental autonomous exploration setting:

- Deriving a minimax-optimal algorithm for the AX problems;
- Eludicating whether a known and finite upper bound on the total number of states is required for the analysis;
- Integrating the (unknown) range of the SSP value functions into the accuracy level, i.e., consider the AX* objective of Definition 9.7 with the requirement that $\forall s \in \mathcal{K},\ V^{\pi_s}(s_0 \to s) \leq V^\star_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s)(1+\varepsilon)$, instead of the currently studied conditions of $V^{\pi_s}(s_0 \to s) \leq V^\star_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) + \varepsilon$ or $V^{\pi_s}(s_0 \to s) \leq V^\star_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s) + L\varepsilon$.

- Extending the problem to continuous state space and function approximation;
- Relaxing the definition of incrementally controllable states and relaxing the performance definition towards allowing the agent to have a non-zero but limited sample complexity of learning a shortest-path policy for any state at test time.

Chapter 10

General Conclusion and Perspectives

10.1 Conclusion on our Contributions

In this thesis, we undertook a formal and thorough investigation of what constitutes provably efficient — and ideally optimal — *goal-oriented exploration for reinforcement learning* (GO-EX-RL). This general scenario encompasses multiple variations in the learning objectives and environment assumptions. In Part I, we focused on a *supervised* scenario of GO-EX-RL. We formalized the online learning problem for Stochastic Shortest Path (SSP), where a goal state to be reached in minimum total expected cost is provided as part of the problem definition. We also presented two no-regret algorithms, each of which extended our understanding of SSP at the time of writing. Then we shifted our attention to the challenging scenario of *unsupervised* GO-EX-RL, where the agent must intrinsically set its own goals and cost functions (SYOG). This general-purpose principle, already widely used in deep RL, had been lacking theoretical underpinning, which was the main motivation and focus of Part II. We rigorously analyzed SYOG in various settings, whose assumptions (e.g., the availability of reset, the incremental restriction) conditioned the dependencies of our theoretical guarantees. We point out that while Part I tackled the traditional regret minimization objective, the metric of interest in Part II was the "pure exploration" sample complexity, which is a particularly relevant scenario in environments where failures are free but samples are costly.

Table 10.1 provides a visual summary of the different results presented in this thesis.

10.2 Perspectives

This thesis only scratches the surface of the foundations of GO-EX-RL. We now discuss some of the (subjectively) most exciting directions for future investigation in GO-EX-RL, both on theoretical and empirical viewpoints.

Part	I	II			
Setting	"Supervised" (a.k.a. SSP)	"Unsupervised"			
Reward Supervision	Specific goal state and cost function	None			
Chapter	Chapters 3 to 5	Chapter 7	Chapter 8	Chapter 8	Chapter 9
Reset Supervision	Reset at goal	None (reset-free)	Anytime reset to initial state		
Extra Assumption	Goal is reachable from any state	Communi-cating	None	Linear mixture	Incremental
Input Parameters	K (optional)	ε	L, ε		
Guarantee type	Regret	Sample complexity			
Algorithm	UC-SSP & EB-SSP	GOSPRL	AdaGoal-UCBVI	AdaGoal-UCRL·VTR	DisCo
Bound poly in	$S, A, B_\star, K$	$S, A, D, \varepsilon^{-1}$	$S, A, L, \varepsilon^{-1}$	d, L, ε^{-1}	$X, A, L, \varepsilon^{-1}$

Table 10.1 – Visual summary of the scope and contributions of this thesis on goal-oriented RL. Notation: $S \triangleq |\mathcal{S}|$ denotes the number of states (known), $A \triangleq |\mathcal{A}|$ denotes the number of actions (known), $B_\star$ bounds the optimal SSP value function from any state (unknown), K denotes the number of SSP episodes (unknown), D denotes the diameter of the MDP (unknown), L denotes the exploration radius around the initial state (known), ε denotes the required accuracy level (known), d denotes the dimension of the feature mapping in the linear mixture MDP (known), $X \triangleq |\mathcal{S}^{\rightarrow}_{L+\varepsilon}|$ denotes the number of incrementally reliably $(L+\varepsilon)$-reachable states (unknown).

Going beyond a finite state space. As motivated in Chapter 1, we focused our study on finitely many states and actions, since even in this basic scenario the formalism of GO-EX-RL had remained under the research radar. However, an important next step is to model a goal beyond a singleton state and thus go beyond a finite state space. For instance, a way to study GO-EX-RL in continuous state spaces could be to adequately discretize the state space (the choice of discretization is a trade-off between precision and computation time and depends on the amount of prior knowledge available). Note that discretization was for example used by Guillot and Stauffer (2020) to model the golfer's problem as an SSP in their study of golf strategy optimization for professional golfers performances estimation on the PGA Tour. When the state space is large, visiting a specific single state can be very difficult, hence it may be necessary to model a goal as a region (e.g., a closed set with a nonempty interior as done by Yershov and LaValle, 2013). Rigorously modeling GO-EX-RL beyond a finite state space is a relevant direction of future investigation to bridge the gap between theory and practice.

Towards horizon-agnostic deep RL algorithms. Existing algorithms in deep RL tackle a goal-reaching task by designing episodes of a *carefully predefined* length H (oftentimes further

adding a fixed discount factor $\gamma < 1$). However, presetting an adequate horizon H is non-trivial and it requires strong task- and environment-dependent prior knowledge. Set H (and/or γ) too small and this will generate a bias in the optimal goal-reaching policy. Set H (and/or γ) too large and the range of value functions will increase, which may lead to numerical instabilities (as well as vacuous theoretical guarantees). In Chapter 5, we have been able to design an algorithm for online SSP that is both minimax-optimal and parameter-free. This theoretical result conveys the conceptual message that it is possible to design intelligent agents that are able to adapt to the unknown difficulty of the task at hand (i.e., the goal-reaching horizon), without sacrificing learning performance. This is promising for improvements in (goal-based) deep RL, where the hope would be to design algorithms that are able to circumvent prior knowledge of the task horizon, for instance *learning when to reset* via a curriculum of increasing episode lengths. In particular, it could be relevant to further investigate (both from a theoretical and practical point of view) how to dynamically select the value of the exploration radius L considered in the formalism of Chapters 8 and 9.

Connecting safe exploration and goal-oriented exploration. The ability of designing a careful sequence of goals and learning when to reset shares high-level similarities with *safe* exploration, which is an important issue to address before RL can be adopted in real-world applications (Amodei et al., 2016). In this case, the agent seeks to minimize safety violations and avoid "unsafe" regions of the state space. This seems related to the *incremental* autonomous exploration formulation (Chapter 9), where "non-controllable" regions of the state space are avoided, as a "controllable" region is gradually expanded around the initial state. Enforcing safe exploration through the angle of goal-driven exploration could be an interesting connection to study.

Designing "goal" curricula. Expanding the concept of *goals* used to drive learning in unsupervised RL, beyond goal *states*, appears very promising. Indeed, some goals may not be expressed as state features, see for instance Colas et al. (2020, Section 4) for a general typology of goal representations in the RL literature. For example, this can include goals represented by a sequence of behaviors to achieve and expressed in language. It also seems intriguing to explore the connections between goal-based RL and the recent studies on Unsupervised Environment Design (Dennis et al., 2020), which seek to fully specify environments (of increasing complexity), rather than just goals within a fixed environment (as for instance done in Chapter 8). I am convinced that designing adaptive *curricula* of increasing "goal" difficulty, for some notion of "goal", is a relevant general idea for training more generally capable RL agents.

Blending goal-based RL and MI-based RL for more sample-efficient unsupervised RL. The broad focus of Part II is the setting of Unsupervised RL (URL), where no reward/cost function

nor goal to reach are provided to the agent. While we restricted our discussion to learning to reach goal states, another popular framework for unsupervised RL is that of learning diverse task-agnostic policies called *skills* via mutual information (MI) maximization (e.g., Gregor et al., 2016; Eysenbach et al., 2019). I also took part in a collaboration exploring this direction (Kamienny et al., 2022). In this work, we learned a growing tree-structured policy that composes directed skills to perform an adaptive and thorough coverage of the state space, and we showed that our method is able to effectively solve sparse-reward (i.e., unknown goal-based) downstream tasks in hard-to-explore continuous navigation and control environments. An interesting feature of our approach is that it does not require prior knowledge on a sensible number of policies, nor on a suitable policy length (i.e., environment diameter).

Trying to smartly combine these two branches of unsupervised RL represents an exciting direction to explore in the near future. On the one hand, we are still lacking a rigorous understanding of what provably constitutes finite-time learning for MI-based objectives, despite the increasing number of empirical works out there, and the finite-time learning guarantees for goal-based RL derived in this thesis could be a relevant source of inspiration. Moreover, there seems to be a whole spectrum on "policy conditioning" that has not yet been fully explored in deep RL, between the two extremes of goal-based RL (grounded conditioning on specific states) and MI-based RL (abstract conditioning on latent variables). In fact, Choi et al. (2021) recently showed that standard goal-conditioned RL is encapsulated by the optimization objective of variational empowerment, and studied the connections between these two principles, which paves the way towards devising goal-conditioned reward functions or developing representation learning techniques[1] in goal-based RL. I believe that further investigating the connections between goal-conditioned RL and MI-based RL may help improve our understanding and the performance of URL.

[1]Indeed, when the goal space is high-dimensional, the problem of learning an adequate lower-dimensional *goal representation* becomes important, although most existing deep RL methods rely on off-the-shelf representation learning (e.g., Nair et al., 2018) optimized prior to or separately from reinforcement learning.

Appendix A

Complements on Chapter 2

A.1 Proof of Lemma 2.13

Let $(U, \pi) \triangleq \text{VI-SSP}(g, \mathcal{S}, \mathcal{A}, P, c, \eta)$ be the solution computed by Algorithm 2.1. The initial vector $u_0 = 0$ verifies $u_0 \leq V^\star$, where it holds that $V^\star = \mathcal{L}V^\star$ by Proposition 2.11. By monotonicity of the operator $\mathcal{L}$ (Bertsekas, 1995), we thus obtain $u_n \leq V^\star$ for any iteration $n \geq 0$, which entails that $U \leq V^\star$.

Moreover, the termination condition implies that for any $s \in \mathcal{S}$, $\mathcal{L}u_n(s) - u_n(s) \leq \eta$, therefore

$$c(s, \pi(s)) + P_{s,\pi(s)}U \leq U(s) + \eta \leq U(s) + \eta \frac{c(s, \pi(s))}{c_{\min}}.$$

We define the vector

$$U' \triangleq \left(1 - \frac{\eta}{c_{\min}}\right)^{-1} U.$$

We see that $c(s, \pi(s)) + P_{s,\pi(s)}U' \leq U'(s)$. Hence, from Proposition 2.10, π is proper and

$$V^\pi(s) \leq U'(s) \leq \left(1 + \frac{2\eta}{c_{\min}}\right) U(s),$$

where in the last inequality we used the assumption that the VI precision level verifies $\eta \leq \frac{c_{\min}}{2}$ and that

$$\forall\, 0 \leq x \leq \frac{1}{2}, \quad \frac{1}{1-x} \leq 1 + 2x. \tag{A.1}$$

A.2 Proof of Lemma 2.14

First, we assume that the policy π is proper in the model $\overline{P}$. This implies that its value function, which we denote by $\overline{V}$, is bounded component-wise. Moreover, from Proposition 2.10, the Bellman equation holds for any $s \in \mathcal{S}$ as follows

$$\overline{V}(s) = c(s, \pi(s)) + \overline{P}_{s,\pi(s)}\overline{V} = c(s, \pi(s)) + P_{s,\pi(s)}\overline{V} + (\overline{P}_{s,\pi(s)} - P_{s,\pi(s)})\overline{V}. \tag{A.2}$$

By successively using Hölder's inequality and that $\overline{P} \in \mathcal{P}_\eta^{(p)}$ and $c(s, \pi(s)) \geq c_{\min}$, we get

$$\overline{V}(s) \geq c(s, \pi(s)) - \eta\|\overline{V}\|_\infty + P(\cdot|s, \pi(s))\overline{V} \geq c(s, \pi(s))\Big(1 - \frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\Big) + P(\cdot|s, \pi(s))\overline{V}.$$

Let us now introduce the vector $V' \triangleq \left(1 - \frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\right)^{-1} \overline{V}$. Then for all $s \in \mathcal{S}$,

$$V'(s) \geq c(s, \pi(s)) + P_{s,\pi(s)}V'.$$

Hence, from Proposition 2.10, π is proper in P, i.e., its associated value function denoted by V is bounded component-wise, and we have

$$V \leq V' \leq \left(1 + 2\frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\right)\overline{V}, \tag{A.3}$$

where the last inequality stems from condition (2.4) and the simple algebraic inequality of Equation (A.1). Conversely, analyzing Equation (A.2) from the other side, we get

$$\overline{V}(s) \leq c(s, \pi(s))\left(1 + \frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\right) + P(\cdot|s, \pi(s))\overline{V}.$$

Let us now introduce the vector $V'' \triangleq \left(1 + \frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\right)^{-1} \overline{V}$. Then

$$V''(s) \leq c(s, \pi(s)) + P_{s,\pi(s)}V''.$$

We then obtain in the same vein as Proposition 2.10 (by leveraging the monotonicity of the Bellman operator $\mathcal{L}^\pi U(s) \triangleq c(s, \pi(s)) + P_{s,\pi(s)}^\top U$) that $V'' \leq V$, and therefore

$$\overline{V} \leq \left(1 + \frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\right)V. \tag{A.4}$$

Combining Equations (A.3) and (A.4) yields component-wise

$$\|V - \overline{V}\|_\infty \leq 2\frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\|\overline{V}\|_\infty + \frac{\eta\|\overline{V}\|_\infty}{c_{\min}}\|V\|_\infty \leq 7\frac{\eta\|\overline{V}\|_\infty^2}{c_{\min}},$$

where the last inequality stems from plugging condition (2.4) into Equation (A.3).

Note that here P and $\overline{P}$ play symmetric roles; we can perform the same reasoning in the case where π is proper in the model P and it would yield an equivalent result by switching the dependencies on V and $\overline{V}$.

Appendix B

Complements on Chapter 3

B.1 Proof of Theorem 3.4

Recall that we introduce the MDP $M_\infty \triangleq \langle \mathcal{S}', \mathcal{A}, r_\infty, P_\infty, s_0 \rangle$, with reward $r_\infty \triangleq \mathbb{1}_g$ and $P_\infty(\cdot \,|\, s, a) \triangleq P(\cdot \,|\, s, a)$ for $s \neq g$ and $P_\infty(\cdot \,|\, g, a) \triangleq \mathbb{1}_{s_0}$ for all a. The SSP problem with uniform costs boils down to minimizing the expected hitting time of the goal state, which according to the following lemma is equivalent to maximizing the long-term average reward (or gain) in M_∞. Recall that for any policy $\pi \in \Pi$, its gain $\rho_\pi(s)$ starting from any $s \in \mathcal{S}$ is defined as

$$\rho_\pi(s) \triangleq \lim_{T \to +\infty} \mathbb{E}_\pi \left[\frac{1}{T} \sum_{t=1}^{T} r_\infty(s_t, \pi(s_t)) \,\middle|\, s \right].$$

Lemma B.1. *Let $\pi_\infty \in \arg\max_\pi \rho_\pi(s)$. Then π_∞ is optimal in the SSP sense and its constant gain ρ_∞ verifies*

$$\rho_\infty = \frac{1}{V^\star(s_0) + 1}.$$

Proof. Let π be a policy such that g is reachable from s_0. Denote by $\mathcal{S}_\pi$ the set of communicating states for policy π in M_∞. Then the underlying Markov chain (restricted to $\mathcal{S}_\pi$) is irreducible with a finite number of states and is thus recurrent positive (see e.g., Brémaud, 2013, Thm. 3.3). Denoting by μ_π its unique stationary distribution, we have almost surely that

$$\rho_\pi(s) = \lim_{T \to +\infty} \mathbb{E}_\pi \left[\frac{\sum_{t=1}^{T} r_t}{T} \right] = \lim_{T \to +\infty} \mathbb{E}_\pi \left[\frac{\sum_{t=1}^{T} \mathbb{1}_{\{s_t = g\}}}{T} \right] \overset{\text{(a)}}{=} \sum_{s \in \mathcal{S}_\pi} \mathbb{1}_{\{s=g\}} \mu_\pi(s) \overset{\text{(b)}}{=} \frac{1}{1 + \mathbb{E}\left[\tau_\pi(s_0)\right]},$$

where (a) comes from the Ergodic Theorem for Markov Chains (see e.g., Brémaud, 2013, Thm. 4.1) and (b) uses the fact that $1/\mu_\pi(g)$ corresponds to the mean return time in state g, i.e., the expected time to reach g starting from g. We conclude with the fact that $V^\pi(s_0) = \mathbb{E}[\tau_\pi(s_0)]$. □

Hence, we can prove that UCRL2 satisfies the following SSP-regret bound.

Lemma B.2. *Under Assumption 3.3, with probability at least $1-\delta$, for any $K \geq 1$, the SSP-regret of* UCRL2 *can be bounded as*

$$R_K \leq 34\,(V^\star(s_0)+1)\,DS\sqrt{AT_K \log\left(\frac{T_K}{\delta}\right)},$$

where we recall that $T_K \triangleq \sum_{k=1}^K I^k$.

Proof. Using the fact that $K = \sum_{t=1}^{T_K} \mathbb{1}_{\{s_t=g\}}$, the SSP-regret can be written as

$$R_K = \sum_{k=1}^K \left[\sum_{t=1}^{I^k} \mathbb{1}_{\{s_t \neq g\}} - V^\star(s_0)\right] = T_K - K - V^\star(s_0)K = T_K - (V^\star(s_0)+1)K.$$

For any $T \geq 1$ denote by R_T^∞ the (reward-based) infinite-horizon total regret of an algorithm after T steps in M_∞, i.e., $R_T^\infty = T\rho^\dagger - \sum_{t=1}^T r_t$ where $\rho^\dagger \triangleq \max_\pi \rho_\pi(s)$ for all $s \in \mathcal{S}$. From Lemma B.1 we have $\rho^\dagger = \rho_\infty$. Moreover, since the rewards satisfy $r_\infty = \mathbb{1}_g$, we have $\sum_{t=1}^{T_K} r_t = K$. Putting everything together yields

$$R_K = T_K - (V^\star(s_0)+1)K = (V^\star(s_0)+1)(T_K\rho^\dagger - K) = (V^\star(s_0)+1)R_{T_K}^\infty.$$

Note that M_∞ is weakly-communicating, where its communicating set of states corresponds to all the states in $\mathcal{S}'$ that are accessible from s_0 with non-zero probability. Although it is weakly-communicating, the specific reward structure, combined with the fact that rewards are necessarily known (since we consider the uniform-cost SSP setting and since the goal state g is assumed to be known), allows to run UCRL2 on this problem (see the Remark at the end of Section B.1 for more detail).

Technically, EVI is guaranteed to converge since the associated extended MDP is weakly-communicating and by Puterman (2014) it is sufficient for convergence of value iteration, see e.g., Puterman (2014, Chapter 9) for finite action space or Schweitzer (1985, Theorem 1) for compact spaces.

From Jaksch et al. (2010, Theorem 2) and using the anytime nature of UCRL2, we have with probability at least $1-\delta$ for any $T > 1$ the following bound on the average-reward regret of

UCRL2 in M_∞,

$$R_T^\infty \leq 34 D_\infty S \sqrt{AT \log(\frac{T}{\delta})},$$

where $D_\infty \triangleq \max_{s \neq s' \in \mathcal{S}'} \min_{\pi \in \Pi^{SD}(M_\infty)} \mathbb{E}\left[\tau_\pi(s \to s')\right]$ is the diameter of M_∞. However, this bound may be vacuous since it depends on D_∞ which may be equal to $+\infty$. By slightly changing the analysis of this result we can obtain an improved dependency on the SSP-diameter D. In particular it is sufficient to prove that for any UCRL2 episode k and for any iteration i of the optimal extended Bellman operator $L_{\mathcal{M}_k}$ (with $h_0 = 0$ and $h_i = (L_{\mathcal{M}_k})^i h_0$), we have that $\mathrm{sp}(h_i) \leq D$ instead of the conventional upper bound D_∞. The remainder of the proof shows this result. It is straightforward that $h_i(g) \geq h_i(s)$ for any $s \in \mathcal{S}$ (this can be proved by recurrence on i using the definition of $h_i = L_{\mathcal{M}_k} h_{i-1}$ and the fact that the reward in $\mathcal{M}_k$ is equal to $\mathbb{1}_g$). Introduce $\underline{s} \in \arg\min_s h_i(s)$ and $\varphi_{\widetilde{M}}(\underline{s} \to g)$ the minimum expected shortest path from $\underline{s}$ to g in any MDP $\widetilde{M}$. Then from Lemma B.4 we have $\mathrm{sp}(h_i) = h_i(g) - h_i(\underline{s}) \leq \varphi_{\mathcal{M}_k}(\underline{s} \to g)$. Since the "true" MDP $M_\infty \in \mathcal{M}_k$, we have $\varphi_{\mathcal{M}_k}(\underline{s} \to g) \leq \varphi_{M_\infty}(\underline{s} \to g)$. Furthermore, $\varphi_{M_\infty}(\underline{s} \to g) = \varphi_M(\underline{s} \to g) \leq D$. Putting everything together, we obtain that $\mathrm{sp}(h_i) \leq D$. We thus have with probability at least $1 - \delta$ for any $T > 1$,

$$R_T^\infty \leq 34 D S \sqrt{AT \log(\frac{T}{\delta})}.$$

□

While we would like to assess the dependency of the regret on the number of episodes K (as in the finite-horizon case), the bound in Lemma B.2 contains the random total number of steps T_K needed to reach K episodes. In light of this, we derive in the following lemma an upper bound of T_K that depends on the quantity of interest K. Plugging it in Lemma B.2 yields the result of Theorem 3.4.

Lemma B.3. *Under the same event for which Lemma B.2 holds with probability at least $1 - \delta$, we have*

$$T_K \leq 2\left(V^\star(s_0) + 1\right) K + \widetilde{O}\left(V^\star(s_0)^2 D^2 S^2 A \log\left(\frac{1}{\delta}\right)\right).$$

Proof. With probability at least $1 - \delta$, we have from the proof of Lemma B.2 that

$$T_K - \left(V^\star(s_0) + 1\right) K \leq 34\left(V^\star(s_0) + 1\right) DS \sqrt{AT_K \log\left(\frac{T_K}{\delta}\right)}.$$

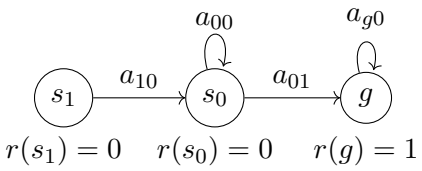


Figure B.1 – A toy example of SSP-communicating ($D = 2$) reward-based MDP.

This implies that

$$T_K \leq 2\left(V^\star(s_0)+1\right) K \underbrace{-T_K + 68\left(V^\star(s_0)+1\right) DS\sqrt{AT_K \log\left(\frac{T_K}{\delta}\right)}}_{\triangleq (y)},$$

where (y) can be bounded using Lemma E.15 (with the constants $a_1 = 68\left(V^\star(s_0)+1\right) DS\sqrt{A}$, $a_2 = \frac{1}{\delta}$ and $a_3 = 1$) as follows

$$(y) \leq \frac{16}{9}\left(68\left(V^\star(s_0)+1\right) DS\sqrt{A}\right)^2 \left[\log\left(\frac{136\left(V^\star(s_0)+1\right) DS\sqrt{Ae}}{\sqrt{\delta}}\right)\right]^2.$$

□

Lemma B.4. *Consider an (extended) MDP $\widetilde{M}$ and define $L_{\widetilde{M}}$ as the associated optimal (extended) Bellman operator (of undiscounted value iteration). Given $h_0 = 0$ and $h_i = (L_{\widetilde{M}})^i h_0$ we have that*

$$\forall s_1, s_2 \in \mathcal{S}', h_i(s_2) - h_i(s_1) \leq r_{\max}\, \varphi_{\widetilde{M}}(s_1 \to s_2),$$

where $\varphi_{\widetilde{M}}(s_1 \to s_2)$ is the minimum expected shortest path from s_1 to s_2 in $\widetilde{M}$ and $r_{\max}$ is the maximal state-action reward.

Proof. The proof follows from the application of the argument of Jaksch et al. (2010, Section 4.3.1). □

Lemma B.5 (Kazerouni et al., 2017, Lemma 8)**.** *For any $x \geq 2$ and $a_1, a_2, a_3 > 0$, the following holds*

$$-a_3 x + a_1\sqrt{x}\log(a_2 x) \leq \frac{16 a_1^2}{9 a_3}\left[\log\left(\frac{2a_1\sqrt{a_2 e}}{a_3}\right)\right]^2.$$

Remark B.6. Consider the reward-based SSP M in Figure B.1. M is SSP-communicating while the associated MDP M_∞ is weakly-communicating since s_1 is transient under every policy. There are just two possible deterministic policies: $\pi_0(s_0) = a_{00}$ and $\pi_1(s_0) = a_{01}$. If rewards are unknown, UCRL2 will periodically alternate between policy π_0 and π_1 without converging to any of the two. This is due to the fact that, in the set of plausible MDPs $\mathcal{M}_k$ there will always be (i.e., $\forall k > 0$) an MDP with arbitrarily small but non-zero transition probability $\tilde{p}$ to state s_1, where, due to maximum uncertainty, there will be a self loop with probability 1 and reward $r_{\max}$ (since $N_k(s_1, a_{10}) \in \{0, 1\}$ depending on the initial state for any k). The probability $\tilde{p}$ will be sometimes higher for action a_{00} and sometimes for a_{01} depending on the counter N_k. This is why UCRL2 will never converge. However, if the rewards are known (which is always the case under Assumption 3.3 and as long as the goal state g is known), after a burn-in phase, it will be clear to UCRL2 that action a_{00} is suboptimal. Even if there is probability $\tilde{p} > 0$ to go to s_1, in s_1 the optimistic behaviour will be to go to g since it is the only one to provide reward. However, this imagined policy is suboptimal since it has an additional step and thus UCRL2 will select π_1. Note that while it is possible to make the MDP stochastic, this will lead to a longer burn-in phase but will not change the behaviour of UCRL2 in the long run.

B.2 $T_\star$ can be arbitrarily larger than $B_\star$, S, A

Here we provide a simple illustration that the inequality $B_\star \leq T_\star$ may be arbitrarily loose, which shows that scaling with $T_\star$ can be much worse than scaling with $B_\star$. Recall that $B_\star$ bounds the total expected cost of the optimal policy starting from any state, and $T_\star$ bounds the expected time-to-goal of the optimal policy from any state.

Let us consider an SSP instance whose optimal policy induces the absorbing Markov chain depicted in Figure B.2. It is easy to see that $B_\star = 1$ and that $T_\star = \Omega(S\, p_{\min}^{-1})$. Hence, the gap between $B_\star$ and $T_\star$ can grow arbitrarily large as $p_{\min} \to 0$.

This simple example illustrates the benefit of having a bound that is (nearly) *horizon-free* (cf. desired property 3 in Section 3.3). Indeed, a bound that is not horizon-free scales polynomially with $T_\star$ and thus with $p_{\min}^{-1}$, which may be arbitrarily large if $p_{\min} \to 0$. In contrast, a horizon-free bound only scales logarithmically with $p_{\min}^{-1}$ and can therefore be much tighter.

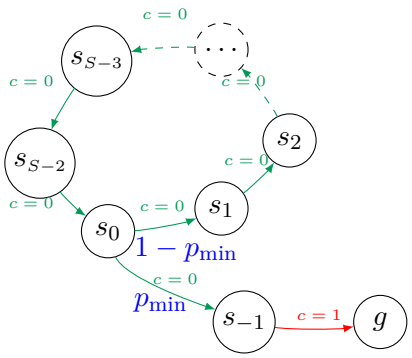


Figure B.2 – Markov chain of the optimal policy of an SSP instance with S states. Transitions in green incur a cost of 0, while the transition in red leading to the goal state g incurs a cost of 1. All transitions are deterministic, apart from the one starting from s_0, which reaches state s_{-1} with probability $p_{\min}$ and state s_1 with probability $1 - p_{\min}$, where $p_{\min} > 0$.

Appendix C

Complements on Chapter 4

C.1 Proofs

C.1.1 Proof of Lemma 4.4

The first inequality comes from the chosen stopping condition. As for the second, since we consider the initial vector $v^{(0)} = 0$, we know that $v^{(0)} \leq \widetilde{V}^{\star}_{k,j}$ with $\widetilde{V}^{\star}_{k,j} = \widetilde{\mathcal{L}}_{k,j}\widetilde{V}^{\star}_{k,j}$. By monotonicity of the operator $\widetilde{\mathcal{L}}_{k,j}$ (Puterman, 2014; Bertsekas, 1995) we obtain $\widetilde{v}_{k,j} \leq \widetilde{V}^{\star}_{k,j}$. If $M \in \mathcal{M}_{k,j}$ and $j = 0$, then $\widetilde{V}^{\star}_{k,j} \leq V^{\star}$. If $M \in \mathcal{M}_{k,j}$ and $j \geq 1$, then all costs are equal to 1 so the optimal value function is $\min_{\pi} \mathbb{E}(\tau_{\pi})$ and hence $\widetilde{V}^{\star}_{k,j} \leq \min_{\pi} \mathbb{E}(\tau_{\pi})$.

C.1.2 Proof of Lemma G.4

The proof is almost identical to the proof of Fruit et al. (2020, Theorem 10) and we report it below for completeness. Recall that we define $\mathcal{M}_{k,j} \triangleq \{\langle \mathcal{S}, \mathcal{A}, c, \widetilde{P}\rangle \mid \widetilde{P} \in B_{k,j}\}$ to be the extended MDP defined by the confidence interval $B_{k,j} \triangleq \{\widetilde{P} \in \mathcal{C} \mid \widetilde{P}(\cdot|g,a) = \mathbb{1}_g$ and $\forall (s,a) \in \mathcal{S} \times \mathcal{A}, \|\widetilde{P}(\cdot|s,a) - \widehat{P}_{k,j}(\cdot|s,a)\|_1 \leq \beta_{k,j}(s,a)\}$, with $\mathcal{C}$ the S'-dimensional simplex and

$$\beta_{k,j}(s,a) \triangleq \sqrt{\frac{8S \log\left(\frac{2AN^{+}_{k,j}(s,a)}{\delta}\right)}{N^{+}_{k,j}(s,a)}}.$$

Furthermore we introduce $B_{k,j}(s,a) \triangleq \{\widetilde{P} \in \mathcal{C} : \|\widetilde{P}(\cdot|s,a) - \widehat{P}_{k,j}(\cdot|s,a)\|_1 \leq \beta_{k,j}(s,a)\}$ (and similarly for $B_{k,j}(s,a,s')$). We want to bound the probability of event $\mathcal{E}^{\mathcal{C}} \triangleq \bigcup_{k=1}^{+\infty} \bigcup_{j=1}^{J_k} \{M \notin \mathcal{M}_{k,j}\}$. As explained by Lattimore and Szepesvári (2020, Chapter 5), when (s,a) is visited for the n-th times, the next state that we observe is the n-th element of an infinite sequence of i.i.d. r.v. lying in $\mathcal{S}'$ with probability density function $P(\cdot|s,a)$. In UCRL2 (Jaksch et al., 2010), the sample

means $\widehat{P}_{k,j}$ and the confidence intervals $B_{k,j}$ are defined as depending on (k,j). Actually, these quantities depend only on the first $N_{k,j}(s,a)$ elements of the infinite i.i.d. sequences that we just mentioned. For the rest of the proof, we will therefore slightly change our notations and denote by $\widehat{P}_n(s'|s,a)$ and $B_n(s'|s,a)$ the sample means and confidence intervals after the first n visits in (s,a). Thus, the random variable that we denoted by $\widehat{P}_{k,j}$ actually corresponds to $\widehat{P}_{N_{k,j}(s,a)}$ with our new notation (and similarly for $B_{k,j}$). This change of notation will make the proof easier.

If $M \notin \mathcal{M}_{k,j}$, then there exists a $k \geq 1$ and $j \geq 0$ s.t. $P(\cdot|s,a) \notin B_{N_{k,j}(s,a)}(s,a)$ for at least one $(s,a,s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}'$. This means that there exists at least one value $n \geq 0$ s.t. $P(s'|s,a) \notin B_n(s,a,s')$. Consequently we have the following inclusion

$$\mathcal{E}^{\mathcal{C}} \subseteq \bigcup_{s,a} \bigcup_{n=0}^{+\infty} \{P(\cdot|s,a) \notin B_n(s,a)\}.$$

Using Boole's inequality we have

$$\mathbb{P}(\mathcal{E}^{\mathcal{C}}) \leq \sum_{s,a} \sum_{n=0}^{+\infty} \mathbb{P}(P(\cdot|s,a) \notin B_n(s,a)).$$

Let us fix a tuple $(s,a) \in \mathcal{S} \times \mathcal{A}$ and define for all $n \geq 0$

$$\varepsilon_n(s,a) \triangleq \sqrt{\frac{2\log\left((2^{S'}-2)5SA(n^+)^2/\delta\right)}{n^+}},$$

where $n^+ \triangleq \max\{n,1\}$. Since $S' = S+1 \leq 2S$, it is immediate to verify that almost surely, $\varepsilon_n(s,a) \leq \beta_n(s,a)$. Using Weissman's inequality (Weissman et al., 2003; Jaksch et al., 2010) we have that for all $n \geq 1$

$$\mathbb{P}\Big(\|P(\cdot|s,a) - \widehat{P}_n(\cdot|s,a)\|_1 \geq \beta_n(s,a)\Big) \leq \mathbb{P}\Big(\|P(\cdot|s,a) - \widehat{P}_n(\cdot|s,a)\|_1 \geq \varepsilon_n(s,a)\Big) \leq \frac{\delta}{5n^2SA}.$$

Note that when $n=0$ (i.e., when there has not been any observation of (s,a)), $\varepsilon_0(s,a) \geq 2$ so $\mathbb{P}(\|P(\cdot|s,a) - \widehat{P}_0(\cdot|s,a)\|_1 \geq \varepsilon_0(s,a)) = 0$ by definition. As a result, we have that for all $n \geq 1$

$$\mathbb{P}\Big(P(\cdot|s,a) \notin B_n(s,a)\Big) \leq \frac{\delta}{5n^2SA},$$

and this probability is equal to 0 if $n=0$. Finally we obtain

$$\mathbb{P}\Big(\exists k \geq 1, \exists j \in [0, J_k], \text{ s.t. } M \notin \mathcal{M}_{k,j}\Big) \leq \sum_{s,a}\left(0 + \sum_{n=1}^{+\infty} \frac{\delta}{5n^2SA}\right) = \frac{\pi^2\delta}{30} \leq \frac{\delta}{3},$$

which concludes the proof.

C.1.3 Proof of Lemma 4.6

For notational ease, in Section C.1.3 we adopt the notation $H_k \triangleq H_{k,0}$, $\widetilde{\pi}_k \triangleq \widetilde{\pi}_{k,0}$, $\varepsilon_k \triangleq \varepsilon_{k,0}$ (i.e., we remove the subscript 0). Furthermore, for any $k \in [K]$ and $h \in [H_k]$, we denote by $s_{k,h}$ the state visited in the h-th step of episode k. Assume from now on that the event $\mathcal{E}$ holds. From Lemma 4.4 we have

$$\begin{aligned}
\mathcal{W}_K &= \sum_{k=1}^{K} \left[\left(\sum_{h=1}^{H_k} c(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \right) - V^\star(s_0) \right] \\
&\leq \sum_{k=1}^{K} \left[\left(\sum_{h=1}^{H_k} c(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \right) - \widetilde{v}_k(s_0) \right] \\
&= \sum_{k=1}^{K} \Theta_{k,1}(s_{k,1}),
\end{aligned}$$

where $s_{k,1} \triangleq s_0$, and for any $k \in [K]$ and $h \in [H_k]$, we introduce

$$\Theta_{k,h}(s_{k,h}) \triangleq \sum_{t=h}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - \widetilde{v}_k(s_{k,h}).$$

For any $h \in [H_k - 1]$, we introduce

$$\Phi_{k,h} \triangleq \widetilde{v}_k(s_{k,h+1}) - \sum_{y \in \mathcal{S}} P(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \widetilde{v}_k(y).$$

We then have

$$\begin{aligned}
\Theta_{k,h}(s_{k,h}) &= \sum_{t=h}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - \widetilde{v}_k(s_{k,h}) \\
&\leq \sum_{t=h}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - \widetilde{\mathcal{L}}_k \widetilde{v}_k(s_{k,h}) + \varepsilon_k \\
&\overset{(a)}{=} \sum_{t=h}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - c(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) - \sum_{y \in \mathcal{S}} \widetilde{P}_k(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \widetilde{v}_k(y) + \varepsilon_k \\
&= \sum_{t=h+1}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - \sum_{y \in \mathcal{S}} [\widetilde{P}_k(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) - P(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \\
&\qquad + P(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h}))] \widetilde{v}_k(y) + \varepsilon_k \qquad \text{(C.1)} \\
&\overset{(b)}{\leq} \sum_{t=h+1}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - \sum_{y \in \mathcal{S}} P(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \widetilde{v}_k(y) \\
&\quad + \| P(\cdot \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) - \widetilde{P}_k(\cdot \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \|_1 \| \widetilde{v}_k \|_\infty + \varepsilon_k
\end{aligned}$$

$$\overset{(c)}{\leq} \sum_{t=h+1}^{H_k} c(s_{k,t}, \widetilde{\pi}_k(s_{k,t})) - \sum_{y\in\mathcal{S}} P(y \mid s_{k,h}, \widetilde{\pi}_k(s_{k,h}))\widetilde{v}_k(y)$$
$$+ 2\beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h}))c_{\max}D + \varepsilon_k \tag{C.2}$$
$$= \Theta_{k,h+1}(s_{k,h+1}) + \widetilde{v}_k(s_{k,h+1}) - \sum_{y\in\mathcal{S}} P(y \mid s_{k,h}, \widetilde{\pi}_k(s_{k,h}))\widetilde{v}_k(y)$$
$$+ 2\beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h}))c_{\max}D + \varepsilon_k \tag{C.3}$$
$$= \Theta_{k,h+1}(s_{k,h+1}) + \Phi_{k,h} + 2\beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h}))c_{\max}D + \varepsilon_k, \tag{C.4}$$

where (a) stems from the fact that $\widetilde{\pi}_k$ is the greedy policy with respect to $(\widetilde{v}_k, \varepsilon_k)$, (b) leverages that $\widetilde{v}_k \geq 0$ component-wise and (c) combines Lemma 4.4 and the fact that $\|V^\star\|_\infty \leq c_{\max}D$. Furthermore, whatever the value of s_{k,H_k} we have

$$\begin{aligned}
\Theta_{k,H_k}(s_{k,H_k}) &= c(s_{k,H_k}, \widetilde{\pi}_k(s_{k,H_k})) - \widetilde{v}_k(s_{k,H_k}) \\
&\leq c(s_{k,H_k}, \widetilde{\pi}_k(s_{k,H_k})) - \widetilde{\mathcal{L}}_k\widetilde{v}_k(s_{k,H_k}) + \varepsilon_k \\
&= c(s_{k,H_k}, \widetilde{\pi}_k(s_{k,H_k})) - c(s_{k,H_k}, \widetilde{\pi}_k(s_{k,H_k})) - \sum_{y\in\mathcal{S}} \widetilde{P}_k(y \mid s_{k,H_k}, \widetilde{\pi}_k(s_{k,H_k}))\underbrace{\widetilde{v}_k(y)}_{\geq 0} + \varepsilon_k \\
&\leq \varepsilon_k.
\end{aligned}$$

By telescopic sum, using Equation (C.4), it holds that

$$\begin{aligned}
\Theta_{k,1}(s_{k,1}) &= \sum_{h=1}^{H_k-1} (\Theta_{k,h}(s_{k,h}) - \Theta_{k,h+1}(s_{k,h+1})) + \Theta_{k,H_k}(s_{k,H_k}) \\
&\leq \sum_{h=1}^{H_k-1} \Phi_{k,h} + 2c_{\max}D \sum_{h=1}^{H_k-1} \beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) + (H_k - 1)\varepsilon_k + \Theta_{k,H_k}(s_{k,H_k}) \\
&\leq \sum_{h=1}^{H_k-1} \Phi_{k,h} + 2c_{\max}D \sum_{h=1}^{H_k-1} \beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) + H_k\varepsilon_k.
\end{aligned}$$

Summing over the episode index k yields

$$\sum_{k=1}^{K} \Theta_{k,1}(s_{k,1}) \leq \underbrace{\sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \Phi_{k,h}}_{\triangleq X_K} + 2c_{\max}D \underbrace{\sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h}))}_{\triangleq Y_K} + \underbrace{\sum_{k=1}^{K} H_k\varepsilon_k}_{\triangleq Z_K}.$$

In order to bound X_K, we can write

$$\mathbb{P}\left(\sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \Phi_{k,h} \geq 2c_{\max}D\sqrt{2\left(\sum_{k=1}^{K} H_k\right)\log\left(\frac{2\left(\sum_{k=1}^{K} H_k\right)^2}{\delta}\right)}\right)$$

$$\leq \sum_{n=1}^{+\infty} \mathbb{P}\left(\sum_{k=1}^{K} \sum_{h=1}^{H_k} \Phi_{k,h} \geq 2c_{\max} D \sqrt{2n \log\left(\frac{2n^2}{\delta}\right)} \bigcap \sum_{k=1}^{K} H_k = n \right)$$
$$\leq \sum_{n=1}^{+\infty} \mathbb{P}\left(\sum_{t=1}^{n} \widetilde{\Phi}_t \geq 2c_{\max} D \sqrt{2n \log\left(\frac{2n^2}{\delta}\right)} \right),$$

where we introduce for any $t > 0$,

$$\widetilde{\Phi}_t = \begin{cases} \Phi_{\widetilde{k}_t, t - Z_t} & \text{if } t > Z_t, \\ \Phi_{\widetilde{k}_t + 1, 1} & \text{otherwise,} \end{cases}$$

where $\widetilde{k}_t = \max\{k \mid \sum_{k'=1}^{k} H_{k'} \leq t\}$ and $Z_t = \sum_{k'=1}^{\widetilde{k}_t - 1} H_{k'} + 1$, i.e., we map a value t to the double index (k, h). Denote by $\mathcal{G}_q$ the history of all random events up to (and including) step h of episode k (i.e., $q = \sum_{k'=1}^{k-1} H_k + h$). We have $\mathbb{E}\left[\Phi_{k,h} | \mathcal{G}_q\right] = 0$ (since $\widetilde{v}_k(g) = 0$), and furthermore the stopping time H_k is selected at the beginning of episode k so it is adapted w.r.t. $\mathcal{G}_q$. Hence, $(\widetilde{\Phi}_t)$ is a martingale difference sequence, such that $|\widetilde{\Phi}_t| \leq 2c_{\max} D$. For any fixed $n > 0$, we thus have from Azuma-Hoeffding's inequality that

$$\mathbb{P}\left(\sum_{t=1}^{n} \widetilde{\Phi}_t \geq 2c_{\max} D \sqrt{2n \log\left(\frac{2n^2}{\delta}\right)} \right) \leq \frac{\delta}{2n^2}.$$

As a result, from a union bound over all possible values of $n > 0$, we have with probability at least $1 - \frac{2\delta}{3}$,

$$\sum_{k=1}^{K} \sum_{h=1}^{H_k - 1} \Phi_{k,h} \leq 2c_{\max} D \sqrt{2 \left(\sum_{k=1}^{K} H_k \right) \log\left(\frac{3 \left(\sum_{k=1}^{K} H_k \right)^2}{\delta} \right)}. \tag{C.5}$$

We now proceed in bounding Y_K using a pigeonhole principle. Denoting by $N^{(1)}$ the counter of samples *only* collected during attempts in phase ①, we get

$$\sum_{k=1}^{K} \sum_{h=1}^{H_k - 1} \sqrt{\frac{1}{N_k^{(1)}(s_{k,h}, \widetilde{\pi}_k(s_{k,h}))}} \leq \sum_{s,a} \sum_{n=1}^{N_K^{(1)}(s,a)} \sqrt{\frac{1}{n}} \leq \sum_{s,a} 2\sqrt{N_K^{(1)}(s,a)}$$
$$\leq 2\sqrt{SA} \sqrt{\sum_{s,a} N_K^{(1)}(s,a)}$$
$$\leq 2\sqrt{SAT_{K,1}}.$$

We have $N_k^+(s,a) \geq N_k^{(1)+}(s,a)$ so by applying the technical Lemma C.1 (and considering that $A \geq 2$ since if $A = 1$ there is no learning problem), we get

$$\beta_k(s,a) = \sqrt{\frac{8S\log\left(\frac{2AN_k^+(s,a)}{\delta}\right)}{N_k^+(s,a)}} \leq \sqrt{\frac{8S\log\left(\frac{2AN_k^{(1)+}(s,a)}{\delta}\right)}{N_k^{(1)+}(s,a)}}.$$

Therefore we obtain

$$\sum_{k=1}^{K}\sum_{h=1}^{H_k-1}\beta_k(s_{k,h},\widetilde{\pi}_k(s_{k,h})) \leq 2S\sqrt{8AT_{K,1}\log\left(\frac{2AT_{K,1}}{\delta}\right)}. \tag{C.6}$$

We finally bound Z_K. We have for any $k \in [K]$, $H_k \leq \Omega_K$ and we select $\varepsilon_k = \frac{c_{min}}{2t_{k,0}}$, hence we have $T_{K,1} \leq \Omega_K K$ and

$$\sum_{k=1}^{K} H_k\varepsilon_k \leq \frac{c_{\min}}{2}\sum_{t=1}^{T_{K,1}}\frac{\Omega_K}{t} \leq \frac{c_{\min}}{2}\Omega_K\left(1+\log(\Omega_K K)\right).$$

Putting everything together, a union bound and Lemma G.4 yields with probability at least $1-\delta$,

$$\begin{aligned}\sum_{k=1}^{K}\left[\left(\sum_{h=1}^{H_k}c(s_{k,h},\widetilde{\pi}_k(s_{k,h}))\right)-\widetilde{v}_k(s_0)\right] \leq\ & 4c_{\max}DS\sqrt{8AT_{K,1}\log\left(\frac{2AT_{K,1}}{\delta}\right)}\\ &+2c_{\max}D\sqrt{2T_{K,1}\log\left(\frac{3T_{K,1}^2}{\delta}\right)}\\ &+\frac{c_{\min}}{2}\Omega_K\left(1+\log(\Omega_K K)\right).\end{aligned}$$

Lemma C.1. *For any constant $c \geq 4$, the function $f(x) \triangleq \sqrt{\frac{\log(cx)}{x}}$ is a non-increasing function for $x \geq 1$.*

Proof. Introduce the function $g(x) \triangleq f(x)^2$. We have $g'(x) = \frac{1-\log(cx)}{x^2} \leq 0$ since $x \geq 1 \geq \frac{e}{c}$. So g is non-increasing, hence by composition of functions, $f = \sqrt{g}$ is also non-increasing. □

Interestingly, the bound of Lemma 4.6 resembles a combination of finite- and infinite-horizon guarantees. On the one hand, we have the standard dependency of finite-horizon problems on the horizon H and number of episodes K. On the other hand, H is no longer bounding the range of the value functions, which is replaced by $c_{\max}D$ as in infinite-horizon problems.

C.1.4 Proof of Lemma 4.7

We start the proof of Lemma 4.7 by deriving a general result — which may be of independent interest — that *upper bounds the moments of any discrete PH distribution*.[1]

Lemma C.2. *Consider an absorbing Markov Chain with state space $\mathcal{Y} \cup \{\overline{y}\}$, a single absorbing state $\overline{y}$ and $|\mathcal{Y}|$ transient states. Denote by $Q \in \mathbb{R}^{Y \times Y}$ the transition matrix within the states in $\mathcal{Y}$ and by $\tau(y) \triangleq \tau(y \to \overline{y})$ the first hitting time of state $\overline{y}$ starting from state y. Suppose that there exists a constant $\lambda \geq 2$ such that for any state $y \in \mathcal{Y}$, we have $\mathbb{E}[\tau(y \to \overline{y})] \leq \lambda$. Then for any $r \geq 1$ and any state $y \in \mathcal{Y}$, we have*

$$\mathbb{E}[\tau(y)^r] \leq 2\,(r\lambda)^r\,.$$

Proof. We first leverage a closed-form expression of the *factorial moments* of discrete PH distributions. For any $r \geq 1$, denoting by $(\tau)_r$ the r-th factorial moment of τ, i.e., $(\tau)_r \triangleq \tau(\tau-1)...(\tau-r+1)$, we have (see e.g., Latouche and Ramaswami, 1999, Equation 2.15) that for any starting state $y \in \mathcal{Y}$,

$$\mathbb{E}[(\tau)_r(y)] = r!\mathbb{1}_y^\top (I-Q)^{-r} Q^{r-1}\mathbb{1}.$$

Recalling that the $\|\cdot\|_\infty$ (resp. $\|\cdot\|_1$) norm of a matrix is equal to its maximum absolute row (resp. column) sum, we have by Hölder's inequality, for any $j \in [r]$,

$$\begin{aligned}
\mathbb{E}[(\tau)_j(y)] &= j!\left\langle (\mathbb{1}_y^\top (I-Q)^{-j})^\top, Q^{j-1}\mathbb{1}\right\rangle \\
&\leq j!\|(\mathbb{1}_y^\top (I-Q)^{-j})^\top\|_1 \|Q^{j-1}\mathbb{1}\|_\infty \\
&= j!\|((I-Q)^{-j})^\top \mathbb{1}_y\|_1 \|Q^{j-1}\mathbb{1}\|_\infty \\
&\leq j!\|((I-Q)^{-j})^\top\|_1 \|\mathbb{1}_y\|_1 \|Q^{j-1}\|_\infty \|\mathbb{1}\|_\infty \\
&\leq j!\|(I-Q)^{-j}\|_\infty \|Q^{j-1}\|_\infty \\
&\leq j!\|(I-Q)^{-1}\|_\infty^j, \qquad \text{(C.7)}
\end{aligned}$$

where the last inequality uses the fact that $\|Q^{j-1}\|_\infty \leq 1$ since the matrix Q^{j-1} is substochastic. There remains to upper bound the quantity $\|(I-Q)^{-1}\|_\infty$. Consider a state

$$z \in \arg\max_{y \in \mathcal{Y}} \sum_{y' \in \mathcal{Y}} (I-Q)^{-1}_{yy'}.$$

[1] Note that while there actually exists a *closed-form* expression of the moments of a continuous PH distribution (see e.g., Latouche and Ramaswami, 1999, Equation 2.13), it does not extend to the discrete case.

By choice of z and non-negativity of the matrix $(I-Q)^{-1}$, we have

$$\|(I-Q)^{-1}\|_\infty = \sum_{y'\in\mathcal{Y}} |(I-Q)^{-1}_{zy'}| = \sum_{y'\in\mathcal{Y}} (I-Q)^{-1}_{zy'} = \mathbb{1}_z^\top (I-Q)^{-1}\mathbb{1} = \sum_{n=0}^{\infty} \mathbb{1}_z^\top Q^n \mathbb{1}.$$

Since $\tau(z)$ follows a discrete PH distribution, we have from Proposition 4.1 that

$$\mathbb{1}_z^\top Q^n \mathbb{1} = \mathbb{P}(\tau(z) > n).$$

Consequently,

$$\|(I-Q)^{-1}\|_\infty = \sum_{n=0}^{\infty} \mathbb{P}(\tau(z) > n) = \mathbb{E}[\tau(z)] \leq \lambda. \tag{C.8}$$

Plugging Equation (C.8) into Equation (C.7) thus yields for any $y \in \mathcal{Y}$,

$$\mathbb{E}\left[(\tau)_j(y)\right] \leq j!\lambda^j.$$

Furthermore, the (raw) moment of a random variable can be expressed in terms of its factorial moments by the following formula (see e.g., Joarder and Mahmood, 1997, Equation 3.1)

$$\mathbb{E}\left[\tau(y)^r\right] = \sum_{j=1}^{r} \left\{ {r \atop j} \right\} \mathbb{E}\left[(\tau)_j(y)\right],$$

where the curly braces denote Stirling numbers of the second kind, i.e.,

$$\left\{ {r \atop j} \right\} \triangleq \frac{1}{j!} \sum_{i=0}^{j} (-1)^{j-i} \binom{j}{i} i^r.$$

Using the upper bound (see e.g., Canfield and Pomerance, 2002, Equation 9)

$$\left\{ {r \atop j} \right\} \leq \frac{j^r}{j!},$$

we obtain

$$\mathbb{E}\left[\tau(y)^r\right] \leq \sum_{j=1}^{r} j^r \lambda^j.$$

We conclude the proof of Lemma C.2 with the fact that

$$\sum_{j=1}^{r} j^r \lambda^j \leq r^r \sum_{j=1}^{r} \lambda^j \leq r^r \lambda \frac{\lambda^r - 1}{\lambda - 1} \leq r^r 2\lambda^r,$$

where the last inequality holds since $\lambda \geq 2$. □

We are now ready to prove Lemma 4.7. For notational ease, in Section C.1.4 we adopt the notation $H_k \triangleq H_{k,0}$, $\widetilde{\pi}_k \triangleq \widetilde{\pi}_{k,0}$, $\varepsilon_k \triangleq \varepsilon_{k,0}$ (i.e., we remove the subscript 0). Denote by $\mathcal{G}_{k-1}$ the history of all random events up to (and including) episode $k-1$. In this section as well as in Section C.1.5, we will write $\mathbb{E}\left[\mathbb{1}_{\{\tau_\pi^P(s) > H_k - 1\}} \,|\, \mathcal{G}_{k-1}\right] = \mathbb{P}(\tau_\pi^P(s) > H_k - 1)$, i.e. the probability $\mathbb{P}$ is only over the randomization of the sequence of states generated by the policy π in the model P starting from state s (i.e., it is conditioned on $\mathcal{G}_{k-1}$, the policy π, the model P and the starting state s).

Suppose that the event $\mathcal{E}$ holds and fix an episode $k \in [K]$. Denote by $\widetilde{Q} \triangleq Q_{\widetilde{\pi}_k}^{\widetilde{P}_k}$ the optimistic transition matrix within $\mathcal{S}$ of policy $\widetilde{\pi}_k$ in the transition model $\widetilde{P}_k$. Also, for any state $s \in \mathcal{S}$, denote by $\widetilde{\tau}(s) \triangleq \tau_{\widetilde{\pi}_k}^{\widetilde{P}_k}(s)$ the hitting time of g starting from s following policy $\widetilde{\pi}_k$ in the transition model $\widetilde{P}_k$. Finally, let $\widetilde{V}_{\widetilde{\pi}_k}(s) \triangleq \mathbb{E}_{\widetilde{P}_k}\left[\sum_{t=1}^{\widetilde{\tau}(s)} c(s_t, \widetilde{\pi}_k(s_t)) \;\middle|\; s_1 = s\right]$ be the value function of policy $\widetilde{\pi}_k$ in the model $\widetilde{P}_k$.

From Lemma 2.13, the choice of EVI precision level $\varepsilon_k \leq \frac{c_{\min}}{2}$ and $\|V^\star\|_\infty \leq c_{\max} D$, it holds that

$$\mathbb{E}[\widetilde{\tau}(s)] \leq \frac{\widetilde{V}_{\widetilde{\pi}_k}(s)}{c_{\min}} \leq \left(1 + \frac{2\varepsilon_k}{c_{\min}}\right) \frac{\widetilde{v}_k(s)}{c_{\min}} \leq \frac{2V^\star(s)}{c_{\min}} \leq \frac{2c_{\max} D}{c_{\min}}. \tag{C.9}$$

Fix any $r \geq 1$ and $s \in \mathcal{S}$. According to a corollary of Markov's inequality (since $x \mapsto x^r$ is a monotonically increasing non-negative function for the non-negative reals), we have

$$\mathbb{P}(\widetilde{\tau}(s) \geq H_k - 1) \leq \frac{\mathbb{E}\left[\widetilde{\tau}(s)^r\right]}{(H_k - 1)^r}.$$

We can apply Lemma C.2 to the discrete PH distribution $\widetilde{\tau}$ with the choice of $\lambda \triangleq \frac{2c_{\max} D}{c_{\min}}$ guaranteed by Equation (C.9). This yields

$$\mathbb{E}\left[\widetilde{\tau}(s)^r\right] \leq 2\left(r \frac{2c_{\max} D}{c_{\min}}\right)^r.$$

Hence, we have

$$\mathbb{P}(\widetilde{\tau}(s) \geq H_k - 1) \leq \frac{2\left(r \frac{2c_{\max} D}{c_{\min}}\right)^r}{(H_k - 1)^r}. \tag{C.10}$$

There exists $y \in \mathcal{S}$ such that

$$\|\widetilde{Q}^{H_k - 2}\|_\infty = \mathbb{1}_y^\top \widetilde{Q}^{H_k - 2} \mathbb{1} = \mathbb{P}(\widetilde{\tau}(y) > H_k - 2) = \mathbb{P}(\widetilde{\tau}(y) \geq H_k - 1), \tag{C.11}$$

where the before-last equality uses Proposition 4.1 applied to $\widetilde{\pi}_k \in \Pi(\langle \mathcal{S}', \mathcal{A}, c, \widetilde{P}_k, y \rangle)$ (the fact that $\widetilde{\pi}_k$ is proper in $\widetilde{P}_k$ stems from Equation (C.9)), while the last equality uses that the hitting time $\widetilde{\tau}(y)$ is an integer. By definition of $H_k \triangleq \min\left\{n > 1 : \|\widetilde{Q}^{n-1}\|_\infty \leq \frac{1}{\sqrt{k}}\right\}$, we have $\|\widetilde{Q}^{H_k-2}\|_\infty > \frac{1}{\sqrt{k}}$. Combining this with Equation (C.10) and (C.11) yields

$$\frac{2\left(r\frac{2c_{\max}D}{c_{\min}}\right)^r}{(H_k-1)^r} > \frac{1}{\sqrt{k}},$$

which implies that

$$H_k - 1 < r\frac{2c_{\max}D}{c_{\min}}\left(2\sqrt{k}\right)^{\frac{1}{r}}.$$

In particular, selecting $r \triangleq \lceil \log(2\sqrt{k}) \rceil$ yields

$$\begin{aligned} H_k - 1 &< \frac{2c_{\max}D}{c_{\min}}\lceil \log(2\sqrt{k}) \rceil (2\sqrt{k})^{\frac{1}{\lceil \log(2\sqrt{k}) \rceil}} \\ &\leq \frac{2c_{\max}D}{c_{\min}}\lceil \log(2\sqrt{k}) \rceil \underbrace{(2\sqrt{k})^{\frac{1}{\log(2\sqrt{k})}}}_{=e}. \end{aligned}$$

Hence,

$$\Omega_K \leq \left\lceil 6\frac{c_{\max}}{c_{\min}} D \log(2\sqrt{K}) \right\rceil.$$

C.1.5 Proof of Lemma 4.8

For notational ease, in Section C.1.5 we adopt the notation $H_k \triangleq H_{k,0}$, $\widetilde{\pi}_k \triangleq \widetilde{\pi}_{k,0}$, $\varepsilon_k \triangleq \varepsilon_{k,0}$ (i.e., we remove the subscript 0).

We denote by $\tau_k(s)$ (resp. $\widetilde{\tau}_k(s)$) the hitting time to the goal of policy π_k in the true model P (resp. in the optimistic model $\widetilde{P}_k$) starting from state s. For any $h \in [H_k]$ we define

$$\Gamma_{k,h}(s_{k,h}) = \mathbb{1}_{\{\tau_k(s_{k,h}) > H_k - h\}} - \mathbb{P}(\widetilde{\tau}_k(s_{k,h}) > H_k - h).$$

Since $F_K = \sum_{k=1}^K \mathbb{1}_{\{\tau_k(s_{k,1}) > H_k - 1\}}$, we have

$$F_K = \sum_{k=1}^K \Gamma_{k,1}(s_{k,1}) + \sum_{k=1}^K \mathbb{P}(\widetilde{\tau}_k(s_0) > H_k - 1).$$

We have for $h \in [H_k - 1]$, $\mathbb{1}_{\{\tau_k(s_{k,h}) > H_k - h\}} = \mathbb{1}_{\{\tau_k(s_{k,h+1}) > H_k - h - 1\}}$ and therefore

$$\Gamma_{k,h}(s_{k,h}) = \mathbb{1}_{\{\tau_k(s_{k,h+1}) > H_k - h - 1\}} - \sum_{y \in \mathcal{S}'} \widetilde{P}_k(y \,|\, s_{k,h}, \widetilde{\pi}_k(s_{k,h}))\mathbb{P}(\widetilde{\tau}_k(y) > H_k - h - 1)$$

$$\leq \mathbb{1}_{\{\tau_k(s_{k,h+1})>H_k-h-1\}} - \sum_{y\in\mathcal{S}'} P(y \mid s_{k,h}, \widetilde{\pi}_k(s_{k,h}))\mathbb{P}(\widetilde{\tau}_k(y) > H_k - h - 1)$$
$$+ 2\beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h}))$$
$$= \Gamma_{k,h+1}(s_{k,h+1}) + \Psi_{k,h} + 2\beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h})),$$

where we define

$$\Psi_{k,h} = \mathbb{P}(\widetilde{\tau}_k(s_{k,h+1}) > H_k - h - 1) - \sum_{y\in\mathcal{S}'} P(y \mid s_{k,h}, \widetilde{\pi}_k(s_{k,h}))\mathbb{P}(\widetilde{\tau}_k(y) > H_k - h - 1).$$

Furthermore, whatever the value of s_{k,H_k} we have

$$\Gamma_{k,H_k}(s_{k,H_k}) = \mathbb{1}_{\{\tau_k(s_{k,H_k})>0\}} - \mathbb{P}(\widetilde{\tau}_k(s_{k,H_k}) > 0) = \mathbb{1}_{\{s_{k,H_k}\neq g\}} - \mathbb{1}_{\{s_{k,H_k}\neq g\}} = 0.$$

By telescopic sum we thus get

$$\Gamma_{k,1}(s_{k,1}) = \sum_{h=1}^{H_k-1} (\Gamma_{k,h}(s_{k,h}) - \Gamma_{k,h+1}(s_{k,h+1})) + \Gamma_{k,H_k}(s_{k,H_k})$$
$$\leq \sum_{h=1}^{H_k-1} \Psi_{k,h} + 2\sum_{h=1}^{H_k-1} \beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h})).$$

Summing over the episode index k yields

$$F_K \leq \sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \Psi_{k,h} + 2\sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) + \sum_{k=1}^{K} \mathbb{P}(\widetilde{\tau}_k(s_0) > H_k - 1).$$

$(\Psi_{k,h})$ is a martingale difference sequence with $|\Psi_{k,h}| \leq 2$, so from Azuma-Hoeffding's inequality, in the same vein as in Equation (C.5), we have with probability at least $1 - \frac{2\delta}{3}$

$$\sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \Psi_{k,h} \leq 2\sqrt{2\left(\sum_{k=1}^{K} H_k\right)\log\left(\frac{3\left(\sum_{k=1}^{K} H_k\right)^2}{\delta}\right)} \leq 2\sqrt{2\Omega_K K \log\left(\frac{3(\Omega_K K)^2}{\delta}\right)}.$$

By the pigeonhole principle (Equation (C.6)), we have

$$\sum_{k=1}^{K}\sum_{h=1}^{H_k-1} \beta_k(s_{k,h}, \widetilde{\pi}_k(s_{k,h})) \leq 2S\sqrt{8A\Omega_K K \log\left(\frac{2A\Omega_K K}{\delta}\right)}.$$

From Proposition 4.1 and Hölder's inequality, we have

$$\sum_{k=1}^{K} \mathbb{P}(\widetilde{\tau}_k(s_0) > H_k - 1) = \sum_{k=1}^{K} \mathbb{1}_{s_0}(Q_{\widetilde{\pi}_k}^{\widetilde{P}_k})^{H_k-1}\mathbb{1} \leq \sum_{k=1}^{K} \|\mathbb{1}_{s_0}\|_1 \|(Q_{\widetilde{\pi}_k}^{\widetilde{P}_k})^{H_k-1}\mathbb{1}\|_\infty \leq \sum_{k=1}^{K} \|(Q_{\widetilde{\pi}_k}^{\widetilde{P}_k})^{H_k-1}\|_\infty.$$

Consequently, by choice of $H_k \triangleq \min\{n > 1 \mid \|(Q^{\widetilde{P}_k}_{\widetilde{\pi}_k})^{n-1}\|_\infty \leq \frac{1}{\sqrt{k}}\}$, we get

$$\sum_{k=1}^{K} \mathbb{P}(\widetilde{\tau}_k(s_0) > H_k - 1) \leq \sum_{k=1}^{K} \frac{1}{\sqrt{k}} \leq 2\sqrt{K}.$$

C.1.6 Proof of Lemma 4.9

Recall that $T_{K,2}$ is the number of time steps during attempts in phase ② up to the end of environmental episode K. We introduce $\Omega'_K \triangleq \max_{k\in[K]} \max_{j\in[J_k]} H_{k,j}$ and $G_K \triangleq \sum_{k=1}^{K} J_k$ which is the total number of attempts in phase ② up to episode K. This means that $T_{K,2} \leq \Omega'_K G_K$. First, by adapting Lemma 4.7 and using that in attempts in phase ② we have $c_{\max} = c_{\min} = 1$, we have under the event $\mathcal{E}$,

$$\Omega'_K \leq \left\lceil 6D \log(2\sqrt{G_K}) \right\rceil. \tag{C.12}$$

We can decompose G_K as the sum of attempts that succeed in reaching g (equal to F_K which is upper bounded by Lemma 4.8) and of those that fail in reaching g, whose number we denote by $F^\dagger_K$. We then have

$$G_K \leq F_K + F^\dagger_K. \tag{C.13}$$

By adapting Lemma 4.8, we have the following high-probability bound, for any value of G_K,

$$F^\dagger_K = O\left(S\sqrt{A\Omega'_K G_K \log\left(\frac{A\Omega'_K G_K}{\delta}\right)}\right). \tag{C.14}$$

Plugging Equation (C.12) and (C.13) into Equation (C.14) yields

$$G_K \leq F_K + O\left(S\sqrt{ADG_K}\log\left(\frac{ADG_K}{\delta}\right)\right).$$

Hence we get

$$G_K \leq 2F_K \underbrace{-G_K + O\left(S\sqrt{ADG_K}\log\left(\frac{ADG_K}{\delta}\right)\right)}_{\triangleq (y)},$$

where (y) can be bounded using the technical Lemma E.15 as follows

$$(y) \leq O\left(S^2 AD\left[\log\left(\frac{SAD}{\sqrt{\delta}}\right)\right]^2\right).$$

Plugging in the result of Lemma 4.8 yields

$$G_K = \widetilde{O}\left(S\sqrt{\frac{c_{\max}}{c_{\min}}ADK\log\left(\frac{K}{\delta}\right)} + S^2AD\log\left(\frac{1}{\delta}\right)\right).$$

This bound can be translated in a bound on $T_{K,2}$ using Equation (C.12) as follows

$$\begin{aligned} T_{K,2} &= O\left(DG_K\log(S\sqrt{G_K})\right) \\ &= \widetilde{O}\left(DS\sqrt{\frac{c_{\max}}{c_{\min}}ADK\log\left(\frac{K}{\delta}\right)}\log(K) + S^2AD^2\log\left(\frac{1}{\delta}\right)\log(K)\right). \end{aligned}$$

C.1.7 Proof of Theorem 4.5

The (possibly non-stationary) policy μ_k that is executed at each episode k can be written as $(\widetilde{\pi}_{k,0}, \widetilde{\pi}_{k,1}, \dots, \widetilde{\pi}_{k,J_k})$. As explained in Section 4.4, by assigning a regret of $c_{\max}$ to each time step during attempts in phase ② (i.e., during the executions of the policies $\widetilde{\pi}_{k,1}, \dots, \widetilde{\pi}_{k,J_k}$), we can decompose the regret of UC-SSP as

$$\begin{aligned} R_K &= \sum_{k=1}^{K}\left[\left(\sum_{h=1}^{I^k} c(s_{k,h}, \mu_k(s_{k,h}))\right) - V^\star(s_0)\right] \\ &\leq \sum_{k=1}^{K}\left[\left(\sum_{h=1}^{H_{k,0}} c(s_{k,h}, \widetilde{\pi}_{k,0}(s_{k,h}))\right) - V^\star(s_0)\right] + c_{\max}T_{K,2}. \end{aligned}$$

Suppose from now on that the event $\mathcal{E}$ is true (this holds with probability at least $1 - \frac{\delta}{3}$). Lemma 4.6 yields that with probability at least $1 - \frac{2\delta}{3}$,

$$\begin{aligned} \sum_{k=1}^{K}\left[\left(\sum_{h=1}^{H_{k,0}} c(s_{k,h}, \widetilde{\pi}_{k,0}(s_{k,h}))\right) - V^\star(s_0)\right] &\leq 4c_{\max}DS\sqrt{8A\Omega_K K\log\left(\frac{2A\Omega_K K}{\delta}\right)} \\ &\quad + 2c_{\max}D\sqrt{2\Omega_K K\log\left(\frac{3(\Omega_K K)^2}{\delta}\right)} \\ &\quad + \frac{c_{\min}}{2}\Omega_K\left(1 + \log(\Omega_K K)\right), \end{aligned}$$

where according to Lemma 4.7,

$$\Omega_K \leq \left\lceil 6\frac{c_{\max}}{c_{\min}}D\log(2\sqrt{K})\right\rceil.$$

On the other hand, Lemma 4.9 yields

$$T_{K,2} = \widetilde{O}\left(DS\sqrt{\frac{c_{\max}}{c_{\min}} ADK \log\left(\frac{K}{\delta}\right)} \log(K) + S^2 AD^2 \log\left(\frac{1}{\delta}\right) \log(K) \right).$$

Putting everything together finally yields that with probability at least $1-\delta$, for any $K \geq 1$,

$$R_K = \widetilde{O}\left(c_{\max} DS\sqrt{\frac{c_{\max}}{c_{\min}} ADK \log\left(\frac{K}{\delta}\right)} \log(K) + c_{\max} S^2 AD^2 \log\left(\frac{1}{\delta}\right) \log(K) \right).$$

C.2 Relaxation of Assumptions

C.2.1 Straightforward extension to unknown, stochastic costs

Although we assume (as in e.g., Azar et al., 2017) that the costs are known and deterministic for ease of exposition, we emphasize that extending the setting to unknown stochastic costs poses no major difficulty. The only requirement is that the learner needs to know in advance the range of the non-goal costs, i.e., the constants $c_{\min}$ and $c_{\max}$. In that case, at the beginning of each attempt $(k,0)$ (i.e., in phase ①), the confidence set $\mathcal{M}_{k,0}$ is not only defined with the confidence interval on the transition probabilities but also with a confidence interval on the costs. Namely, we consider

$$\mathcal{M}_{k,0} \triangleq \left\{ \langle \mathcal{S}, \mathcal{A}, \widetilde{c}, \widetilde{P} \rangle \mid \widetilde{P}(\cdot|s,a) \in B_{k,0}(s,a), \widetilde{c}(s,a) \in B'_{k,0}(s,a) \right\},$$

where $B_{k,0}(s,a)$ is defined as in Section 4.2, and where for any $a \in \mathcal{A}$, $\widetilde{c}(g,a) = 0$ while for any $s \in \mathcal{S}$,

$$B'_{k,0}(s,a) \triangleq [\widehat{c}_{k,0}(s,a) - \beta'_{k,0}(s,a), \widehat{c}_{k,0}(s,a) + \beta'_{k,0}(s,a)] \cap [c_{\min}, c_{\max}],$$

with $\widehat{c}_{k,0}(s,a)$ the empirical costs and

$$\beta'_{k,0}(s,a) \triangleq 2\sqrt{\frac{\log\left(\frac{6SAN^+_{k,0}(s,a)}{\delta}\right)}{N^+_{k,0}(s,a)}}.$$

The analysis on the regret bound of UC-SSP then only adds an additional error term on estimating the transition costs, which is subsumed by the other terms. Consequently, we obtain exactly the same regret bound as in Theorem 4.5.

C.2.2 Relaxation of Assumption 2.7 (i.e., $D = +\infty$)

The requirement that the goal is reachable from any state (Assumption 2.7) is a natural and inherent assumption of the SSP problem as introduced in Bertsekas (1995). However, a reasonable extension is to allow for the existence of (potentially unknown) *dead-end* states, i.e., states from which reaching the goal is impossible. In that case, EVI$_{\text{SSP}}$, which operates of the entire state space $\mathcal{S}$, fails to converge since the values at dead-end states are infinite. Kolobov et al. (2012) propose to put a "cap" on any state's cost by optimizing the *truncated value function*, or Finite-Penalty criterion,

$$V_J^\pi(s) \triangleq \min\{J,\ V^\pi(s)\},$$

where $J > 0$ corresponds to a penalty incurred if a dead-end state is visited. From Kolobov et al. (2012), there exists an optimal policy $\pi_J^\star(s)$ that minimizes $V_J^\pi(s)$ and the optimal truncated value function $V_J^\star$ is a fixed point of the modified Bellman operator $\mathcal{L}_J$ defined as

$$\mathcal{L}_J V(s) \triangleq \min\Big\{J,\ \min_{a\in\mathcal{A}}\Big[c(s,a) + \sum_{y\in\mathcal{S}} P(y|s,a)V(y)\Big]\Big\}.$$

Denote by $\mathcal{S}^{DE} \subsetneq \mathcal{S}$ the set of dead-end states. We replace Assumption 2.7 with the following assumptions.

Assumption C.3. *1)* $s_0 \notin \mathcal{S}^{DE}$. *2)* $V^\star(s_0) < +\infty$ *and an upper bound* J *on* $V^\star(s_0)$ *is known. 3) We augment the action space* $\mathcal{A}$ *with an action* $\overline{a}$ *that causes a transition from any state in* $\mathcal{S}$ *to the target state with probability* 1 *and cost* J *(i.e., we place ourselves in a resetting environment).*

Note that 1) and 3) of Assumption C.3 are required to make the learning problem and the definition of regret sensible (i.e., we have $V^\star(s_0) < +\infty$ and we have the possibility to reset whenever we are stuck in a dead-end state). Moreover, 2) guarantees that $V^\star(s_0) = V_J^\star(s_0)$ and that if we run EVI$_{\text{SSP}}$ on $\mathcal{L}_J$ instead of $\mathcal{L}$, then J is an upper bound on the optimistic value function output by EVI$_{\text{SSP}}$ (instead of $c_{\max}D$ which is vacuous when $D = +\infty$). Note that 2) is tightly related to the requirement of Fruit et al. (2018b) of prior knowledge on an upper bound of the span of the optimal bias function, and that 1) is similar to the assumption of a starting state belonging to the set of communicating states in TUCRL (Fruit et al., 2018a).

With those assumptions at hand, we consider the algorithm UC-SSP-$\mathcal{L}_J$, which differs from UC-SSP in 3 ways: it iterates EVI$_{\text{SSP}}$ on the operator $\mathcal{L}_J$, the length of the k-th phase ① is set to $H_k^{(J)} \triangleq 6\frac{J}{c_{\min}}\log(2\sqrt{k})$, and it executes action $\overline{a}$ at the end of each attempt ① (this means that there is no more phase ②, and the k-th attempt ① exactly corresponds to the k-th environmental episode).

Lemma C.4. *Under Assumption C.3 and 4.2, with probability at least $1-\delta$, the regret of* UC-SSP-$\mathcal{L}_J$ *can be bounded as*

$$R_K(\text{UC-SSP-}\mathcal{L}_J) = O\left(JS\sqrt{A\Omega_K^{(J)}K\log\left(\frac{\Omega_K^{(J)}K}{\delta}\right)}\right),$$

where $\Omega_K^{(J)} \triangleq 6\frac{J}{c_{\min}}\log(2\sqrt{K})$.

Proof. We have

$$\begin{aligned} R_K(\text{UC-SSP-}\mathcal{L}_J) &= \sum_{k=1}^{K}\left[\left(\sum_{h=1}^{I^k} c(s_{k,h},\widetilde{\pi}_k(s_{k,h}))\right) - V_J^\star(s_0)\right] \\ &\leq \sum_{k=1}^{K}\left[\left(\sum_{h=1}^{H_k^{(J)}} c(s_{k,h},\widetilde{\pi}_k(s_{k,h}))\right) - V_J^\star(s_0)\right] + JF_K, \end{aligned}$$

where the double sum can be bounded by

$$O\left(JS\sqrt{A\Omega_K K\log\left(\frac{\Omega_K K}{\delta}\right)}\right)$$

by adapting the proof of Lemma 4.6, since $\widetilde{\pi}_k$ is the greedy policy w.r.t. the optimistic value function $\widetilde{v}_k^{(J)}$ which satisfies both $\widetilde{v}_k^{(J)}(s_0) \leq V_J^\star(s_0)$ and $\|\widetilde{v}_k^{(J)}\|_\infty \leq J$.

Note that the optimistic hitting time $\tau_{\widetilde{\pi}_k}^{\widetilde{P}_k}$ starting from any state in $\mathcal{S}\setminus\mathcal{S}^{DE}$ still follows a discrete PH distribution with $|\mathcal{S}^{DE}|+1$ absorbing states (which can be reduced to a discrete PH distribution with a single absorbing state and with the same distribution of the time to absorption). Consequently, using the same reasoning as in the proof of Lemma 4.7, we can prove that under the event $\mathcal{E}$,

$$\mathbb{P}(\tau_{\widetilde{\pi}_k}^{\widetilde{P}_k}(s_0) \geq H_k^{(J)}) \leq \frac{1}{\sqrt{k}}.$$

Hence we can bound F_K exactly as in Lemma 4.8. We obtain the desired regret bound by using that $\Omega_K^{(J)} \triangleq \max_{k\in[K]} H_K^{(J)} = 6\frac{J}{c_{\min}}\log(2\sqrt{K})$ by choice of $H_k^{(J)}$. □

Interesting future directions in the setting where $D = +\infty$ could be to attempt to remove the need for the prior knowledge J (i.e., weaken Assumption C.3), or to focus on the related problem of maximizing the probability of reaching the goal state while keeping cumulative costs low (see e.g., Kolobov et al., 2012, Section 6).

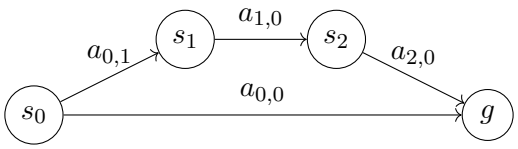


Figure C.1 – SSP instance used in the proof of Lemma C.5.

C.2.3 Relaxation of Assumption 4.2 (i.e., if $c_{\min} = 0$)

The existence of $c_{\min} > 0$ is leveraged in our analysis to bound Ω_K, more specifically in Equation (C.9), which uses that the property of optimism w.r.t. the value functions (i.e., $\widetilde{v}_{k,0} \leq V^\star$ component-wise) yields a "cost-weighted optimism" w.r.t. the expected hitting times, i.e., $\mathbb{E}(\widetilde{\tau}_{k,0}) \leq \frac{2c_{\max}}{c_{\min}} \mathbb{E}(\tau_{\pi^\star})$ component-wise. Yet if zero costs are possible (i.e., $c_{\min} = 0$), then this implication fails to hold.

To circumvent this problem a natural idea is to introduce an additive perturbation $\eta_{k,0} > 0$ to the cost of each transition in the true SSP (note that a small offset of costs to avoid to tricky case of zero costs is also performed by Bertsekas and Yu, 2013). One may hope that this would not affect the behavior of the optimal policy, yet whereas in finite- and infinite-horizon this is indeed the case (i.e., offsetting the costs by a positive constant does not affect the behavior of the optimal policy), Lemma C.5 shows that this property does not hold in the SSP setting.

Lemma C.5. *For any $\eta > 0$, there exists an SSP instance whose optimal policy is different from the one of an identical SSP with all of its transition costs offset by η.*

Proof. Let us consider the SSP from Figure C.1, whose costs are $c(s_0, a_{0,0}) = 4\eta$ and $c(s_0, a_{0,1}) = c(s_1, a_{1,0}) = c(s_2, a_{2,0}) = \eta$. The optimal policy executes action $a_{0,0}$ in state s_0. Yet if the costs are all offset by η, the optimal policy executes action $a_{0,1}$ in state s_0. □

Offsetting the costs thus introduces a bias which should be adequately controlled by the choice of $\eta_{k,0}$. We consider the algorithm UC-SSP-$\mathcal{L}_\eta$, which differs from UC-SSP by introducing an additive perturbation $\eta_{k,0} > 0$ to the cost of each transition in the *optimistic model* for each attempt $(k, 0)$ (i.e., in phase ①), i.e., the algorithm iterates EVI$_{\text{SSP}}$ up to an accuracy of $\varepsilon_{k,0} \triangleq \frac{c_{\max}}{t_{k,0}}$ on the operator $\mathcal{L}_\eta$ defined as

$$\mathcal{L}_\eta V(s) \triangleq \min_{a \in \mathcal{A}} \Big[c(s,a) + \eta + \sum_{y \in \mathcal{S}} P(y|s,a) V(y) \Big],$$

where $\eta > 0$ depends on the episode $k \in [K]$.

Lemma C.6. *If $c_{\min} = 0$, under Assumption 2.7, by selecting $\eta_{k,0} = \frac{1}{k^{1/3}}$, we get with overwhelming probability that*

$$R_K(\text{UC-SSP-}\mathcal{L}_\eta) = \widetilde{O}\left(c_{\max} DS\sqrt{c_{\max} DA}K^{2/3} + T_\star K^{2/3} + c_{\max} DS\sqrt{T_\star AK} \right.$$
$$\left. + T_\star S\sqrt{c_{\max} DA}K^{1/3} + T_\star S\sqrt{T_\star A}K^{1/6} + S^2AD^2\right),$$

where we recall that $T_\star \triangleq \|\mathbb{E}[\tau_{\pi^\star}]\|_\infty$ bounds the hitting time of the optimal policy $\pi^\star$ in the original SSP (i.e., without any cost offset) starting from any state.

Proof. For notational ease, throughout the proof of Lemma C.6 we adopt the notation $\eta_k \triangleq \eta_{k,0}$, $H_k \triangleq H_{k,0}$, $\widetilde{\pi}_k \triangleq \widetilde{\pi}_{k,0}$, $\varepsilon_k \triangleq \varepsilon_{k,0}$ (i.e., we remove the subscript 0).

UC-SSP-$\mathcal{L}_\eta$ modifies the EVI procedure so that it selects a pair $(\widetilde{\pi}_k, \widetilde{P}_k)$ that satisfies for any $s \in \mathcal{S}$,

$$(\widetilde{\pi}_k, \widetilde{P}_k) \in \arg\min_{\widetilde{\pi}, \widetilde{P}} \widetilde{v}^{(\eta)}_{\widetilde{\pi}, \widetilde{P}}(s), \tag{C.15}$$

where

$$\widetilde{v}^{(\eta)}_{\widetilde{\pi}, \widetilde{P}}(s) \triangleq \mathbb{E}_{\widetilde{P}}\left[\sum_{t=1}^{\tau_{\widetilde{\pi}}(s)} c(s_t, \widetilde{\pi}(s_t)) + \eta_k \,\middle|\, s\right] = \mathbb{E}_{\widetilde{P}}\left[\sum_{t=1}^{\tau_{\widetilde{\pi}}(s)} c(s_t, \widetilde{\pi}(s_t)) \,\middle|\, s\right] + \eta_k \mathbb{E}_{\widetilde{P}}\left[\tau_{\widetilde{\pi}}(s)\right],$$

and we let for ease of notation $\widetilde{v}^{(\eta)}_k(s) \triangleq \widetilde{v}^{(\eta)}_{\widetilde{\pi}_k, \widetilde{P}_k}(s)$ and $\widetilde{v}_k(s) \triangleq \mathbb{E}_{\widetilde{P}_k}\left[\sum_{t=1}^{\tau_{\widetilde{\pi}_k}(s)} c(s_t, \widetilde{\pi}_k(s_t)) \,\middle|\, s\right]$.

From Equation (C.15) we have that under the event $\mathcal{E}$, $\widetilde{v}^{(\eta)}_k(s) \le \widetilde{v}^{(\eta)}_{\pi^\star, p}(s)$, or equivalently by expanding,

$$\widetilde{v}^{(\eta)}_k(s) = \widetilde{v}_k(s) + \eta_k \mathbb{E}_{\widetilde{P}_k}\left[\tau_{\widetilde{\pi}_k}(s)\right] \le \mathbb{E}_p\left[\sum_{t=1}^{\tau_{\pi^\star}} c(s_t, \pi^\star(s_t)) + \eta_k \,\middle|\, s\right] = V^\star(s) + \eta_k \mathbb{E}\left[\tau_{\pi^\star}(s)\right]. \tag{C.16}$$

Plugging into Equation (C.16) that $\widetilde{v}_k(s) \ge 0$ and $\|V^\star\|_\infty \le c_{\max} D$ yields

$$\|\mathbb{E}_{\widetilde{P}_k}\left[\tau_{\widetilde{\pi}_k}\right]\|_\infty \le \frac{c_{\max} D}{\eta_k} + T_\star. \tag{C.17}$$

Hence the term $\frac{c_{\max} D}{c_{\min}}$ in Equation (C.9) (and thus in Lemma 4.7) can be replaced by the upper bound in Equation (C.17), which implies that under the event $\mathcal{E}$,

$$\Omega_K \le 6\left(\frac{c_{\max} D}{\eta_K} + T_\star\right)\log(S\sqrt{K}).$$

Furthermore, using Equation (C.16) the regret can be decomposed as

$$\sum_{k=1}^{K}\left[\left(\sum_{h=1}^{I^k} c(s_{k,h},\widetilde{\pi}_k(s_{k,h}))\right) - V^\star(s_0)\right]$$
$$\leq \sum_{k=1}^{K}\left[\left(\sum_{h=1}^{H_k} c(s_{k,h},\widetilde{\pi}_k(s_{k,h}))\right) - \widetilde{v}_k^{(\eta)}(s_0)\right] + T_\star\sum_{k=1}^{K}\eta_k + c_{\max}T_{K,2},$$

where the double sum can be bounded by (excluding lower-order terms)

$$O\left((c_{\max}D + \eta_K T_\star)S\sqrt{A\Omega_K K\log\left(\frac{\Omega_K K}{\delta}\right)}\right),$$

by adapting the proof of Lemma 4.6, since $\widetilde{\pi}_k$ is the greedy policy w.r.t. the optimistic value function $\widetilde{v}_k^{(\eta)}$ which satisfies $\|\widetilde{v}_k^{(\eta)}\|_\infty \leq c_{\max}D + \eta_k T_\star$ from Equation (C.16). Moreover, we can bound $T_{K,2}$ as in Section 4.4 by using Lemma 4.9.

Hence selecting $\eta_k = \frac{1}{k^{1/3}}$ and plugging in the bound on Ω_K yields the desired bound. □

C.2.4 Summary

We report in Table C.1 the regret guarantees of UC-SSP (by isolating the dependencies on K and on D or J), depending on the assumptions made (and the corresponding choices of Bellman operator for EVI$_{\text{SSP}}$). We notice that if $D = +\infty$ and under Assumption C.3, UC-SSP-$\mathcal{L}_J$ satisfies a regret bound where the infinite term D is replaced with the known upper bound $J \geq V^\star(s_0)$. Moreover, UC-SSP-$\mathcal{L}_\eta$ can deal with the existence of zero costs, however the rate worsens from $\sqrt{K}$ (in Theorem 4.5 which requires $c_{\min} > 0$) to $K^{2/3}$, due to the bias introduced by offsetting the costs in the optimistic model. Finally, it is straightforward to combine the two aforementioned variants and derive UC-SSP-$\mathcal{L}_{J,\eta}$ which can handle both $D = +\infty$ (under Assumption C.3) and $c_{\min} = 0$.

Assumptions	**Regret bound**
$c_{\min} > 0$ **(Assumption 4.2) and** $D < \infty$ **(Assumption 2.7)**	$\widetilde{O}(D^{3/2}\sqrt{K})$
$c_{\min} > 0$ **(Assumption 4.2) and** $V^\star(s_0) \leq J$ **w/ RESET (Assumption C.3)**	$\widetilde{O}(J^{3/2}\sqrt{K})$
$c_{\min} = 0$ **and** $D < \infty$ **(Assumption 2.7)**	$\widetilde{O}(D^{3/2}K^{2/3})$
$c_{\min} = 0$ **and** $V^\star(s_0) \leq J$ **w/ RESET (Assumption C.3)**	$\widetilde{O}(J^{3/2}K^{2/3})$

Table C.1 – Regret guarantees of UC-SSP depending on the assumptions made.

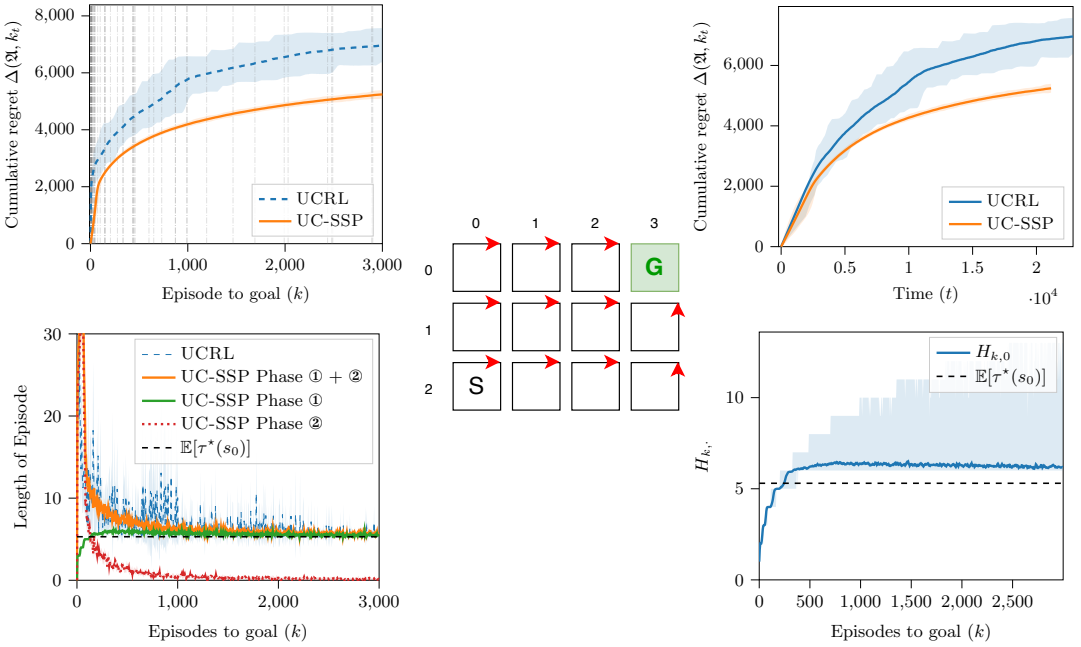


Figure C.2 – Comparison of UC-SSP and UCRL in the case of uniform-cost SSP. The plots are averaged over 200 repetitions. We report the mean and the maximum and minimum value for top line and figure bottom right. For the bottom-left figure, we report the standard deviation of the mean at 96% to simplify the visualization.

C.3 Experiments

In this section, we empirically validate our theoretical findings and perform an ablation study of the algorithms. We consider 3 scenarios: 1) uniform-cost SSP; 2) SSP with $c_{\min} > 0$ and 3) SSP with $c_{\min} = 0$. In all the experiments, we consider the same (3×4) gridworld but we modify the cost function. The agent can move using the cardinal actions (Right, Down, Left, Up). An action fails with probability $p_f = 0.05$. In this case (failure), the agent uniformly follows one of the other directions. Walls are absorbing, i.e., if the action leads against the wall, the agent stays in the current position with probability 1. For example, $P((0,0)|(0,0), right) = \frac{2p_f}{3}$, $P((1,0)|(0,0), right) = \frac{p_f}{3}$ and $P((0,1)|(0,0), right) = 1 - p_f$. If we consider *Up*, we have $P((0,0)|(0,0), Up) = 1$. For the experiments we used the theoretical confidence intervals without constants, i.e., $\beta_{k,j}(s,a) = \sqrt{\frac{SL}{N^+_{k,j}(s,a)}}$ with $L = \log(SAN^+_{k,j}(s,a)/0.1)$. The remaining parameters are set as prescribed by the theory. All the results are averaged over 200 runs.

1) The first experiment aims to compare UCRL2 (Jaksch et al., 2010) and UC-SSP in the case of uniform-cost SSP described in Section 3.2 (see Figure C.2). We set $c(s,a) = 1$ for any $(s,a) \in \mathcal{S} \times \mathcal{A}$, and $c(g,a) = 0$ for all $a \in \mathcal{A}$. We evaluate the algorithms at $K = 3000$ episodes. Figure C.2(top left) shows that the regret of both algorithms is sublinear, as stated

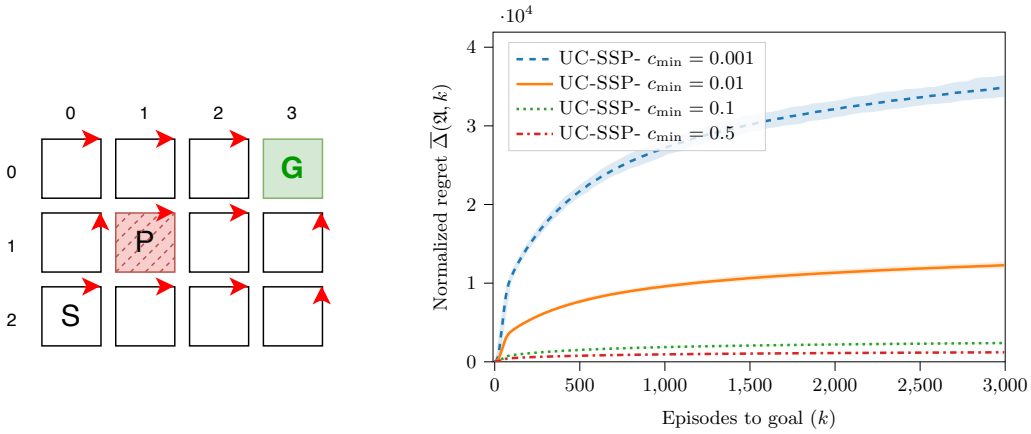


Figure C.3 – Evaluation of the effect of $c_{\min} > 0$ on the regret of UC-SSP. Results are averaged over 200 runs. We report mean value and maximum and minimum observed values.

by the theoretical analysis. Interestingly, the regret of UCRL is higher than the one incurred by UC-SSP. This is possibly due to algorithmic structure of UCRL, which behaves in epochs (or algorithmic episodes) and each epoch ends when the number of visits to some state-action pair is doubled. UCRL computes the policy only at the beginning of an epoch. As shown by the vertical lines in Figure C.2(top left), between each planning step, the agent may reach the goal multiple times. While this can be computationally efficient, the drawback is that UCRL may execute sub-optimal policies for long time. On the other hand, we believe that by planning more often, UC-SSP is able to execute better policies than UCRL. In fact, Figure C.2(bottom left) shows that the time required by UCRL to reach the goal g is often higher than the one of UC-SSP. It also shows that the length of phase ② in UC-SSP quickly goes to zero, meaning that policy executed by UC-SSP is able to quickly reach the goal. Figure C.2(top right) shows that UCRL requires more time (i.e., steps) than UC-SSP to successfully complete 2000 episodes. This test sheds light on the relationship between UCRL and UC-SSP and shows that, despite the good regret guarantees, UCRL may not exploit the specific structure of the SSP problem and poorly performs compared to UC-SSP. Finally, we also plot the estimate of the hitting time computed by UC-SSP (see Figure C.2(bottom right)). As expected, it is a "tight" upper-bound to the expected hitting time of the optimal SSP policy ($\mathbb{E}[\tau_{\pi^\star}(s_0)] = 5.3$), except in the initial episodes where the optimistic model is far away from the true one. In the latter case, the imagined SSP problem has high probability of reaching g from any other state due to the high uncertainty.

2) The second experiment focuses on non-uniform cost. At each step, the agent incurs a cost of $\beta > 0$ except when in $\widetilde{s} = (1, 1) = P$ where the cost is 1. The state $\widetilde{s}$ is considered to be a sand pit and has the effect of slowing down the agent (i.e., higher cost). Formally, $c(s, a) = \beta$ for all $(s, a) \in (\mathcal{S} \setminus \{\widetilde{s}\}) \times \mathcal{A}$, $c(\widetilde{s}, a) = 1$ for all $a \in \mathcal{A}$, and $c(g, a) = 0$ for all $a \in \mathcal{A}$. Clearly, $c_{\min} = \beta > 0$. Note that the optimal SSP policy is the same for all the selected values of β. As before, we evaluate the algorithms at $K = 3000$ episodes. In Figure C.3(right) we show the

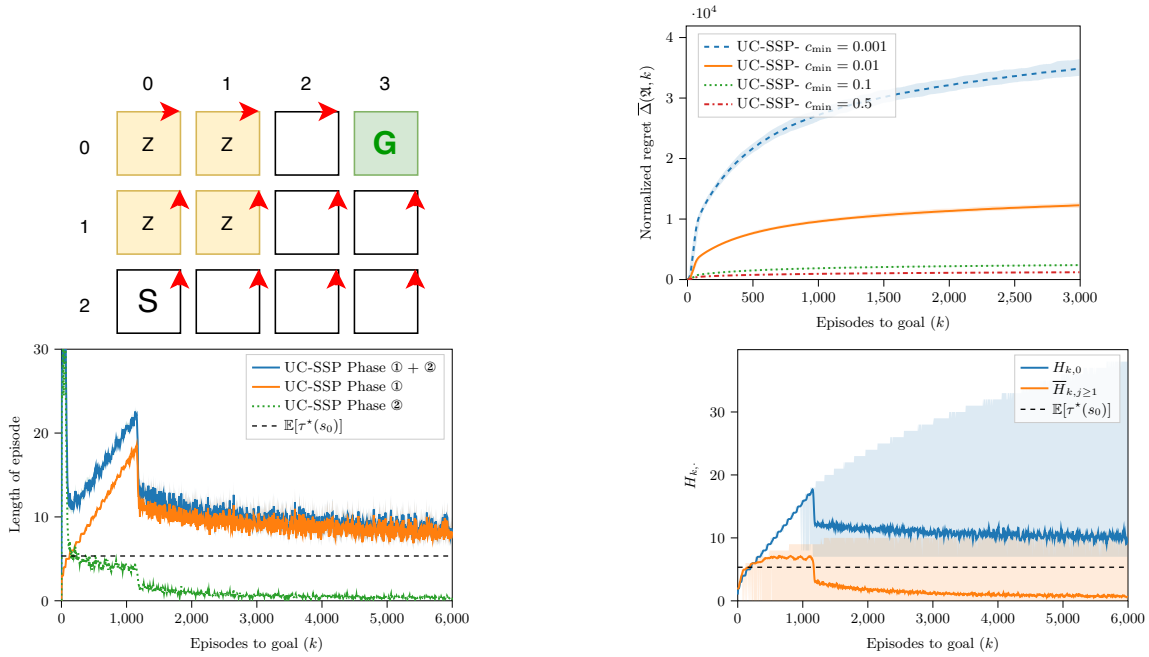


Figure C.4 – Evaluation of UC-SSP for $c_{\min} = 0$. See Figure C.2 for details.

impact of $c_{\min}$ on the regret of UC-SSP. First of all, we show how $c_{\min}$ affects the true solution of the SSP problem. To do so, we run VI on the true model with $\varepsilon = 1.e-10$ and obtain

$$V^\star(s_0|\beta = 0.5) = 2.66, \;\; V^\star(s_0|\beta = 0.1) = 0.55,$$
$$V^\star(s_0|\beta = 0.01) = 0.07, \;\; V^\star(s_0|\beta = 0.001) = 0.02.$$

To remove the impact of the different magnitude of the cost, we consider the normalized regret $\overline{\Delta}(\mathfrak{A}, K) \triangleq \frac{\Delta(\mathfrak{A},K)}{V^\star(s_0)}$. Figure C.3(right) shows that the complexity of the learning problem scales inversely with $c_{\min}$.

3) The final experiment deals with the case $c_{\min} = 0$. We consider the states $(0,0)$, $(0,1)$, $(1,1)$ and $(1,0)$ to have zero cost, see Figure C.4(left). All the other states have cost defined as in experiment 2) with $\beta = 0.4$. Note that there exists loops with zero costs, which means that there exist improper policies with finite V-values. As mentioned in App. C.2.3, in this case we compete against the optimal proper policy (see Figure C.4(top left)). To compute the optimal proper policy and its value V, we use VI with perturbation of $1e-10$ (Bertsekas and Yu, 2013). We evaluate the algorithms at $K = 3000$ episodes. We notice that UC-SSP has sublinear regret as expected. The perturbation of the costs has a large impact on the initial phase of UC-SSP when both uncertainty and perturbation are high. In this case, UC-SSP highly overestimates the hitting time of the optimal policy, leading to the execution of suboptimal policies for a long time (due to Phase ①). Once the perturbation and/or the uncertainty decreases, we notice that the estimated hitting time drops rapidly and approaches the true value. It is also interesting to

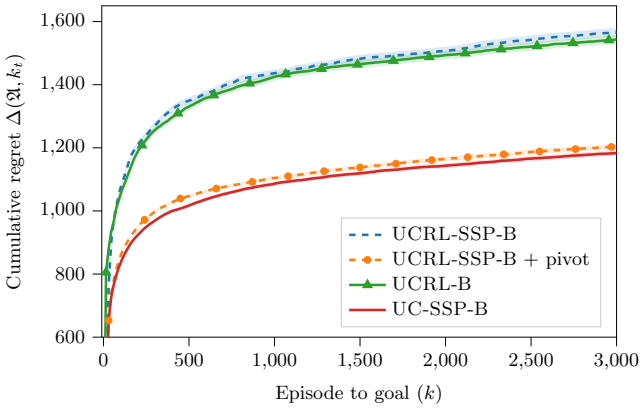


Figure C.5 – Evaluation of the algorithms with Bernstein inequalities and uniform cost. See Figure C.2 for details. We average the results over 200 runs and report the standard deviation of the mean at 96%.

notice that the estimated hitting time of phase ② is never too high. This is due to the fact that phase ② aims to find the policy reaching the goal state in the smallest time.

C.3.1 Bernstein Inequalities

In this section, we provide an evaluation of the proposed algorithm with Bernstein inequalities and perform empirical comparison with later work (Rosenberg et al., 2020). Similar to e.g., Azar et al. (2017) and Fruit et al. (2020), we consider the following concentration inequality of the transition probabilities: $\forall (s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}'$,

$$\left|\widetilde{P}(s'|s,a) - \widehat{P}_{k,j}(s'|s,a)\right| \leq \beta_{k,j}(s,a,s') \approx \sqrt{\frac{\sigma_p^2(s,a,s')L}{N_{k,j}^+(s,a)}} + \frac{L}{N_{k,j}^+(s,a)} \tag{C.18}$$

where $L = \log(SAN_{k,j}^+(s,a)/0.1)$ and $\sigma_p^2(s,a,s') = \widehat{P}_{k,j}(s'|s,a)(1 - \widehat{P}_{k,j}(s'|s,a))$. Optimistic SSP planning can be performed using extended value iteration (as in Alg. 4.2). We thus use the optimistic Bellman operator defined in Equation (4.1) with $B_{k,j}(s,a) \triangleq \{\widetilde{P} \in \mathcal{C} \mid \widetilde{P}(\cdot \mid g, a) = \mathbb{1}_g, |\widetilde{P}(s' \mid s, a) - \widehat{P}_{k,j}(s' \mid s, a)| \leq \beta_{k,j}(s,a,s')\}$.

We compare with UCRL-SSP (Rosenberg et al., 2020). UCRL-SSP is a variant of UCRL2B (Fruit et al., 2020) where the average reward planning is replaced with the SSP planning. When $c_{\min} = 0$, UCRL-SSP leverages the same perturbation idea used by UC-SSP. The cost is then defined as $c(s,a) = \max\{c(s,a), \varepsilon\}$ with $\varepsilon = \frac{S^2 A}{K}$.

The main goal of this section is to empirically show that, despite the $K^{2/3}$ regret bound when $c_{\min} = 0$, UC-SSP is competitive with UCRL-SSP whose regret bound scales as $\sqrt{K}$. We also show the role of the pivot horizon used by UC-SSP.

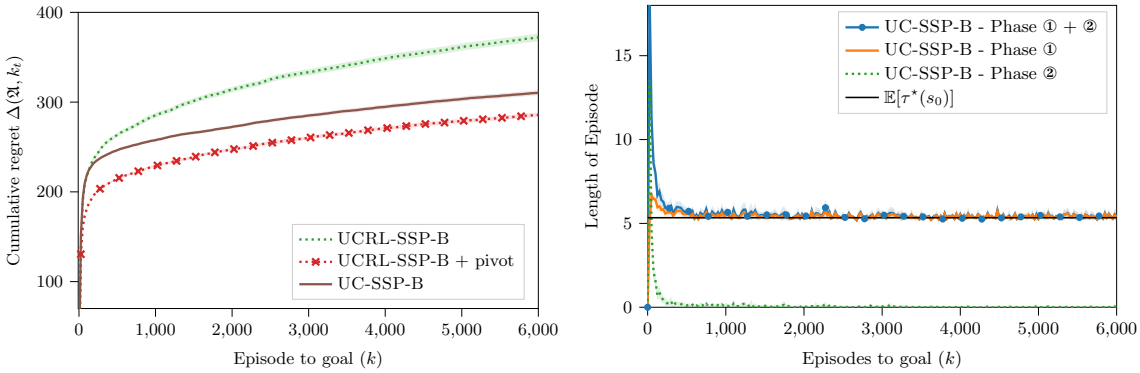


Figure C.6 – Evaluation of the algorithms with Bernstein inequalities and $c_{\min} = 0$. See Figure C.4 for details. Right figure shows the average length of Phase ① and ② for UC-SSP with Bernstein inequalities.

As done in the previous section, we start considering the uniform cost case. Figure C.5 shows that UC-SSP outperforms UCRL-SSP. From Figure C.5 we can see that the lower regret of UC-SSP comes from the use of the pivot horizon. Indeed, when we integrate the pivot horizon idea in UCRL-SSP[2] the algorithms behave similarly. In Figure C.5 we can see that UCRL-SSP behaves as UCRL2B. This is due to the fact that SSP planning is equivalent to average reward planning in this setting (i.e., uniform cost). Furthermore, it shows that, in this domain, UCRL-SSP is not able to leverage the structure of the SSP problem. In contrast, UC-SSP adapts to the SSP problem thanks to the pivot horizon.

The second experiment focuses on the case when $c_{\min} = 0$. As shown in Figure C.6(*left*), UC-SSP has a low regret even in this case. UCRL-SSP achieves the same performance of UC-SSP only when using the pivot horizon as a stopping condition of the algorithmic episode. This shows again that the stopping condition based on pivot horizon allows the algorithms to better adapt to the the SSP structure of this problem. Finally, Figure C.6(*right*) shows that phase ② happens only at the early stages of the learning process. As a consequence, UC-SSP does not suffer additional regret due to phase ② in this domain.

[2]UCRL-SSP uses the same condition of UCRL2B to terminate an algorithmic episode, i.e., when the number of visits to a state-action pair is doubled, the algorithmic episode ends. When using the pivot horizon, we simply limit the number of steps in the algorithmic episode to be at most the pivot horizon (as done for UC-SSP). We also integrated the condition of planning every time the goal state is reached but we didn't observe any significant change in this domain.

Appendix D

Complements on Chapter 5

D.1 An Alternative Assumption on the SSP Problem: No Almost-Sure Zero-Cost Cycles

Here we complement Section 5.4 by introducing an alternative assumption on the SSP problem (which is weaker than Assumption 5.2) and we analyze the regret bound achieved by EB-SSP (under the set-up of Section 5.4). We draw inspiration from the common assumption in the deterministic shortest path setting that the transition graph does not possess any cycle of zero costs (Bertsekas, 1991). In the following we introduce a "stochastic" counterpart of this assumption.

Assumption D.1. *There exist unknown constants $c^\dagger > 0$ and $q^\dagger > 0$ such that:*

$$\mathbb{P}\Bigg(\bigcap_{s' \in \mathcal{S}} \bigcap_{\omega \in \Omega_{s'}} \Big\{ \sum_{i=1}^{|\omega|} c_i \geq c^\dagger \Big\} \Bigg) \geq q^\dagger,$$

where for every state $s' \in \mathcal{S}$ we denote by $\Omega_{s'}$ the set of all possible trajectories in the SSP-MDP that start from state s' and end in state s', and we denote by $c_1, \ldots, c_{|\omega|}$ the sequence of costs incurred during a trajectory ω.

Assumption D.1 is strictly weaker than the assumption of positive costs (Assumption 5.2) and it guarantees that the conditions of Proposition 2.11 hold. Intuitively, it implies that the agent has a non-zero probability of gradually accumulating some positive cost as its trajectory length increases. In particular, under Assumption D.1, any trajectory of length $S+1$ that does not reach the goal must accumulate costs of at least $c^\dagger$ with probability at least $q^\dagger$.

When $z \geq \ln(T/\delta)/q^\dagger \geq \frac{\ln(T/\delta)}{-\ln(1-q^\dagger)}$, it is guaranteed that $(1-q^\dagger)^z \leq \delta/T$. Repeatedly applying this argument means that with probability at least $1-\delta/T$, for $z \geq \ln(T/\delta)/q^\dagger$ it holds that either $\sum_{i=1}^{z(S+1)} c_i \geq c^\dagger$, or the agent has reached the goal in the trajectory indexed by the time steps $[1, z(S+1)]$. Denote $z_0 \triangleq \lceil \ln(T/\delta)/q^\dagger \rceil$. For each episode, divide time steps in it into chunks with length $z_0(S+1)$, with the exception that the last chunk in it may have length less than or equal to $z_0(S+1)$ (just like taking modulo). So in each episode, the agent accumulates cost of at least $c^\dagger$ in each chunk except for the last one, and in the last chunk the agent reaches g. If we define Z as the total number of chunks with cost at least $c^\dagger$ in all episodes, then $Z \geq \frac{T-Kz_0(S+1)}{z_0(S+1)}$. Thus from $C \geq Zc^\dagger$ we have $T \leq O\left(\frac{S\log(T/\delta)}{q^\dagger}\left(\frac{C}{c^\dagger}+K\right)\right) \leq O(S(T/\delta)^{1/4}CK/(q^\dagger c^\dagger))$, with C the cumulative cost. Using the loose bound $C \leq O(B_\star S^2 AK \cdot \sqrt{B_\star TSA/\delta})$ and isolating T (with the same reasoning as in the case of positive costs in Section 5.4) gives that $T \leq O(B_\star^6 S^{14} A^6 K^8/((q^\dagger c^\dagger)^4 \delta^3))$ and thus that $\log T = O(\log(KB_\star SA/(c^\dagger q^\dagger \delta)))$. Plugging this in Theorem 5.1 yields the following.

Corollary D.2. *Under Assumption D.1, running* EB-SSP *(Algorithm 5.1) with $B = B_\star \geq 1$ and $\eta = 0$ gives the following regret bound with probability at least $1-\delta$*

$$R_K = O\left(B_\star\sqrt{SAK}\log\left(\frac{KB_\star SA}{c^\dagger q^\dagger \delta}\right) + B_\star S^2 A \log^2\left(\frac{KB_\star SA}{c^\dagger q^\dagger \delta}\right)\right).$$

The regret bound of Corollary D.2 is (nearly) **minimax** and **horizon-free** (and it can be made **parameter-free** by executing Algorithm D.1 instead of Algorithm 5.1). The bound depends logarithmically on the inverse of the constants $c^\dagger$, $q^\dagger$. We observe that i) it no longer becomes relevant if one constant is exponentially small, ii) spelling out $c^\dagger$, $q^\dagger$ satisfying Assumption D.1 is challenging as they subtly depend on both the cost function and the transition dynamics, although iii) the agent does not need to know nor estimate $c^\dagger$ and $q^\dagger$ to achieve the regret bound of Corollary D.2.

D.2 Full Statement of Corollary 5.6

Here we make explicit the *constant* terms υ, λ, ζ in the regret bound of Corollary 5.6.

Recall that Assumption 5.5 considers that the agent has prior knowledge of a quantity $\overline{T}_\star$ that verifies $T_\star/\upsilon \leq \overline{T}_\star \leq \lambda T_\star^\zeta$ for some unknown constants $\upsilon, \lambda, \zeta \geq 1$ (note that $\upsilon = \lambda = \zeta = 1$ when $T_\star$ is known). Under Assumption 5.5, running EB-SSP (Algorithm 5.1) with $B = B_\star$ and

$\eta = (\overline{T}_\star K)^{-1}$ gives the following regret bound with probability at least $1-\delta$

$$R_K = O\left(\left(B_\star + \frac{\nu}{K}\right)\sqrt{SAK}\zeta \log\left(\frac{\lambda K T_\star SA}{\delta}\right) + \left(B_\star + \frac{\nu}{K}\right) S^2 A \zeta^2 \log^2\left(\frac{\lambda K T_\star SA}{\delta}\right) + \nu\right).$$

D.3 Proof of Theorem 5.1

In this section, we present the proof of Theorem 5.1 (the missing proofs of the intermediate results within the section are deferred to Section D.4). We recall that throughout Section D.3 we analyze Algorithm 5.1 without cost perturbation (i.e., $\eta = 0$) and we assume that **1)** the estimate verifies $B \geq \max\{B_\star, 1\}$ and **2)** the conditions of Proposition 2.11 hold.

D.3.1 High-Probability Event

Definition D.3 (High-probability event)**.** *We define the event* $\mathcal{E} \triangleq \mathcal{E}_1 \cap \mathcal{E}_2 \cap \mathcal{E}_3$*, where*

$$\mathcal{E}_1 \triangleq \left\{\forall (s,a) \in \mathcal{S}\times\mathcal{A}, \forall n(s,a) \geq 1 \,:\, |(\widehat{P}_{s,a} - P_{s,a})V^\star| \leq 2\sqrt{\frac{\mathbb{V}(\widehat{P}_{s,a}, V^\star)\iota_{s,a}}{n(s,a)}} + \frac{14 B_\star \iota_{s,a}}{3n(s,a)}\right\}, \tag{D.1}$$

$$\mathcal{E}_2 \triangleq \left\{\forall (s,a) \in \mathcal{S}\times\mathcal{A}, \forall n(s,a) \geq 1 \,:\, |\widehat{c}(s,a) - c(s,a)| \leq 2\sqrt{\frac{2\widehat{c}(s,a)\iota_{s,a}}{n(s,a)}} + \frac{28\iota_{s,a}}{3n(s,a)}\right\}, \tag{D.2}$$

$$\mathcal{E}_3 \triangleq \left\{\forall (s,a,s') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}', \, \forall n(s,a) \geq 1 \,:\, |P_{s,a,s'} - \widehat{P}_{s,a,s'}| \leq \sqrt{\frac{2P_{s,a,s'}\iota_{s,a}}{n(s,a)}} + \frac{\iota_{s,a}}{n(s,a)}\right\}, \tag{D.3}$$

where $\iota_{s,a} \triangleq \ln\left(\frac{12SAS'[n^+(s,a)]^2}{\delta}\right)$.

Lemma D.4. *It holds that* $\mathbb{P}(\mathcal{E}) \geq 1-\delta$.

Proof. The events $\mathcal{E}_1$ and $\mathcal{E}_2$ hold with probability at least $1-2\delta/3$ by the concentration inequality of Lemma D.20 and by union bound over all $(s,a) \in \mathcal{S}\times\mathcal{A}$. The event $\mathcal{E}_3$ holds with probability at least $1-\delta/3$ by Bennett's inequality (Lemma D.19, anytime version), by Lemma D.26 and by union bound over all $(s,a,s') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}'$. □

D.3.2 Analysis of a VISGO Procedure

A VISGO procedure in Algorithm 5.1 computes iterates of the form $V^{(i+1)} = \widetilde{\mathcal{L}}V^{(i)}$, where $\widetilde{\mathcal{L}}$ is an operator that we define as follows. For any $U \in \mathbb{R}^{S'}$ such that $U(g) = 0$, we set $\widetilde{\mathcal{L}}U(g) \triangleq 0$

and for $s \in \mathcal{S}$ we set $\widetilde{\mathcal{L}}U(s) \triangleq \min_{a \in \mathcal{A}} \widetilde{\mathcal{L}}U(s,a)$, where

$$\widetilde{\mathcal{L}}U(s,a) \triangleq \max\Bigg\{\widehat{c}(s,a) + \widetilde{P}_{s,a}U - \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},U)\iota_{s,a}}{n^+(s,a)}},\ c_2\frac{B\iota_{s,a}}{n^+(s,a)}\right\} \\ - c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)},\ 0\Bigg\}. \tag{D.4}$$

Starting from an optimistic initialization $V^{(0)} = 0$ at each state, we show the following two properties:

- *Optimism:* with high probability, $Q^{(i)}(s,a) \leq Q^\star(s,a), \forall i \geq 0$;
- *Finite-time near-convergence:* Given any error $\varepsilon_{\text{VI}} > 0$, the procedure stops at a *finite* iteration j such that $\|V^{(j)} - V^{(j-1)}\|_\infty \leq \varepsilon_{\text{VI}}$, which implies that the vector $V^{(j)}$ verifies some fixed point equation for $\widetilde{\mathcal{L}}$ up to an error scaling with ε_{VI}.

Properties of the slightly skewed transitions $\widetilde{P}$

Lemma D.5 shows that the bias introduced by replacing $\widehat{P}_{s,a}$ with $\widetilde{P}_{s,a}$ decays inversely with $n(s,a)$, the number of visits to state-action pair (s,a).

Lemma D.5. *For any non-negative vector $U \in \mathbb{R}^{S'}$ such that $U(g) = 0$, for any $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that*

$$\widetilde{P}_{s,a}U \leq \widehat{P}_{s,a}U \leq \widetilde{P}_{s,a}U + \frac{\|U\|_\infty}{n(s,a)+1}, \qquad |\mathbb{V}(\widetilde{P}_{s,a},U) - \mathbb{V}(\widehat{P}_{s,a},U)| \leq \frac{2\|U\|_\infty^2 S'}{n(s,a)+1}.$$

Denote by ν the probability of reaching the goal from any state-action pair in $\widetilde{P}$, i.e.,

$$\nu_{s,a} \triangleq \widetilde{P}_{s,a,g}, \qquad \nu \triangleq \min_{s,a} \nu_{s,a}. \tag{D.5}$$

By construction of $\widetilde{P}$, the quantity ν is strictly positive. This immediately implies the following result.

Lemma D.6. *In the SSP-MDP associated to $\widetilde{P}$ with any bounded cost function, all policies are proper.*

Remark D.7 (Mapping to a discounted problem)**.** In an SSP problem with only proper policies, the (optimal) Bellman operator is usually contractive only w.r.t. a weighted-sup norm (Bertsekas, 1995). Here, the construction of $\widetilde{P}$ entails that any SSP defined on it with fixed bounded costs has a (optimal) Bellman operator that is a sup-norm contraction. In fact, the SSP problem on $\widetilde{P}$ can be cast as a discounted problem with a (state-action dependent) discount factor $\gamma_{s,a} \triangleq 1 - \nu_{s,a} < 1$ (we recall that discounted MDPs are a subclass of SSP-MDPs). Intuitively, at insufficiently visited state-action pairs, the agent behaves optimistically which increases the chance of reaching the goal and terminating the trajectory. Equivalently, we can interpret the agent as being uncertain about its future predictions and it is thus encouraged to act more myopically, which is connected to lowering the discount factor in the discounted RL setting.

Important auxiliary function f and its properties

Lemma D.8 examines an auxiliary function f that plays a key role in the analysis. Indeed, we see that an instantiation of f surfaces in the definition of the operator $\widetilde{\mathcal{L}}$ in Equation (D.4). While the first property (monotonicity) is similar to the one required in Zhang et al. (2021d), the third property (contraction) is SSP-specific and is crucial to guarantee the (finite-time) near-convergence of a VISGO procedure.

Lemma D.8. *Let* $\Upsilon \triangleq \{v \in \mathbb{R}^{S'} : v \geq 0,\ v(g) = 0,\ \|v\|_\infty \leq B\}$. *Let* $f : \Delta^{S'} \times \Upsilon \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ *with* $f(p, v, n, B, \iota) \triangleq pv - \max\left\{c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}},\ c_2\frac{B\iota}{n}\right\}$, *with* $c_1 = 6$ *and* $c_2 = 36$ (*here taking any pair of constants such that* $c_1^2 \leq c_2$ *works*). *Then* f *satisfies, for all* $p \in \Delta^{S'}$, $v \in \Upsilon$ *and* $n, \iota > 0$,

1. $f(p, v, n, B, \iota)$ *is non-decreasing in* $v(s)$, *i.e.,*

$$\forall (v, v') \in \Upsilon^2,\ v \leq v' \implies f(p, v, n, B, \iota) \leq f(p, v', n, B, \iota);$$

2. $f(p, v, n, B, \iota) \leq pv - \frac{c_1}{2}\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} - \frac{c_2}{2}\frac{B\iota}{n} \leq pv - 2\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} - 14\frac{B\iota}{n}$;

3. *If* $p(g) > 0$, *then* $f(p, v, n, B, \iota)$ *is* ρ_p*-contractive in* $v(s)$, *with* $\rho_p \triangleq 1 - p(g) < 1$, *i.e.,*

$$\forall (v, v') \in \Upsilon^2,\ |f(p, v, n, B, \iota) - f(p, v', n, B, \iota)| \leq \rho_p \|v - v'\|_\infty.$$

Optimism of VISGO

We now show that with the bonus defined in Equation (5.1), the Q-function is always optimistic with high probability.

Lemma D.9. *Conditioned on the event $\mathcal{E}$, for any output Q of the* VISGO *procedure (line 22 of Algorithm 5.1) and for any state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that*

$$Q(s,a) \leq Q^\star(s,a).$$

Proof idea. We prove the result by induction on the inner iterations i of VISGO, i.e., $Q^{(i)}(s,a) \leq Q^\star(s,a)$. We use the update of the Q-value (line 5.2), Lemma D.5, the definition of event $\mathcal{E}$ combined with the fact that $B \geq B_\star$, as well as the first two properties of Lemma D.8 applied to $f(\widetilde{P}_{s,a}, V^{(i)}, n^+(s,a), B, \iota_{s,a})$. □

Finite-time near-convergence of VISGO

Warm-up: convergence with no bonuses. For the sake of discussion, let us first examine an idealized case where $n(s,a) \to +\infty$ for all (s,a), which means $b(s,a) = 0$ for all (s,a). In that case, the iterates verify $V^{(i+1)} = \widetilde{\mathcal{L}}^\star V^{(i)}$, where $\widetilde{\mathcal{L}}^\star U(s) \triangleq \min_a \{c(s,a) + \widetilde{P}_{s,a}U\}, \forall U \in \mathbb{R}^S, s \in \mathcal{S}$. Thus $\widetilde{\mathcal{L}}^\star$ is the optimal Bellman operator of the SSP instance $\widetilde{M}$ with transitions $\widetilde{P}$ and cost function c. From Lemma D.6, all policies are proper in $\widetilde{M}$. As a result, the operator $\widetilde{\mathcal{L}}^\star$ is contractive (cf. Remark D.7) and convergent (Bertsekas, 1995).

Convergence with bonuses. In VISGO, however, we must account for the bonuses $b(s,a)$. Setting aside the truncation of each iterate $V^{(i)}$ (i.e., the lower bounding by 0), we notice that a update for $V^{(i+1)}$ can be interpreted as the (truncated) Bellman operator of an SSP problem with cost function $c(s,a) - b^{(i+1)}(s,a)$. However, $b^{(i+1)}(s,a)$ depends on $V^{(i)}$, the previous iterate. This dependence means that the cost function is no longer fixed and the reasoning from the previous paragraph no longer holds. As a result, we directly analyze the properties of the operator $\widetilde{\mathcal{L}}$ that defines the sequence of iterates $V^{(i+1)} = \widetilde{\mathcal{L}} V^{(i)}$ in VISGO, see Equation (D.4)).

Lemma D.10. *The sequence $(V^{(i)})_{i \geq 0}$ is non-decreasing. Combining this with the fact that it is upper bounded by $V^\star$ from Lemma D.9, the sequence must converge.*

While Lemma D.10 states that $\widetilde{\mathcal{L}}$ ultimately converges starting from a vector of zeros, the following result guarantees that it can approximate in finite time its fixed point within any (arbitrarily small) positive component-wise accuracy.

Lemma D.11. *Denote by $\nu > 0$ the probability of reaching the goal from any state-action pair in $\widetilde{P}$, i.e., $\nu \triangleq \min_{s,a} \widetilde{P}_{s,a,g}$. Then $\widetilde{\mathcal{L}}$ is a ρ-contractive operator with modulus $\rho \triangleq 1 - \nu < 1$.*

Proof idea. For any state-action pair (s, a) we can apply the third property (contraction) of Lemma D.8 to the function $f(\widetilde{P}_{s,a}, V^{(i)}, n^+(s,a), B, \iota_{s,a})$. Taking the maximum over (s, a) pairs yields the contraction property of $\widetilde{\mathcal{L}}$. □

Remark D.12. Lemma D.11 guarantees that $\|V^{(i+1)} - V^{(i)}\|_\infty \le \varepsilon_{\text{VI}}$ for $i \ge \frac{\log(\max\{B_\star, 1\}/\varepsilon_{\text{VI}})}{1-\rho}$, which yields the desired property of finite-time near-convergence of VISGO (i.e., it always stops at a finite iteration i). Moreover, by definition of ε_{VI} we have $\log(1/\varepsilon_{\text{VI}}) = O(SA \log(T))$, the (possibly loose) lower bound $1 - \rho = \nu \ge \frac{1}{T+1}$, and there are at most $O(SA \log T)$ VISGO procedures in total, thus we see that EB-SSP has a polynomially bounded computational complexity.

D.3.3 Interval Decomposition and Notation

Interval decomposition. In the analysis we split the time steps into *intervals*. The first interval begins at the first time step, and an interval ends once either (1) the goal state g is reached; (2) or the trigger condition holds (i.e., the visit to a state-action pair is doubled). We see that an update is triggered (line 13 of Algorithm 5.1) whenever condition (2) is met.

Notation. We index intervals by $m = 1, 2, \ldots$ and the length of interval m is denoted by H^m (it is bounded almost surely). The trajectory visited in interval m is denoted by $U^m = (s_1^m, a_1^m, \ldots, s_{H^m}^m, a_{H^m}^m, s_{H^m+1}^m)$, where a_h^m is the action taken in state s_h^m. The concatenation of the trajectories of the intervals up to and including interval m is denoted by $\overline{U}^m$, i.e., $\overline{U}^m = \bigcup_{m'=1}^m U^{m'}$. Moreover, c_h^m denotes the cost in the h-th step of interval m. We use the notation $Q^m(s,a)$, $V^m(s)$, $\widehat{P}_{s,a}^m$, $\widetilde{P}_{s,a}^m$ and $\varepsilon_{\text{VI}}^m$ to denote the values (computed in lines 14-22) of $Q(s,a)$, $V(s)$, $\widehat{P}_{s,a}$, $\widetilde{P}_{s,a}$ and ε_{VI} in the beginning of interval m. Let $n^m(s,a)$ and $\widehat{c}^m(s,a)$ denote the values of $\max\{n(s,a), 1\}$ and $\widehat{c}(s,a)$ used for computing $Q^m(s,a)$. Finally, we set

$$b^m(s,a) \triangleq \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a}, V^m)\iota_{s,a}}{n^m(s,a)}},\ c_2\frac{B\iota_{s,a}}{n^m(s,a)}\right\} + c_3\sqrt{\frac{\widehat{c}^m(s,a)\iota_{s,a}}{n^m(s,a)}} + c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^m(s,a)}.$$

D.3.4 Bounding the Bellman Error

Lemma D.13. *Conditioned on the event $\mathcal{E}$, for any interval m and state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$,*

$$|c(s,a) + P_{s,a}V^m - Q^m(s,a)| \leq \min\left\{\beta^m(s,a), B_\star + 1\right\},$$

where we define

$$\beta^m(s,a) \triangleq 4b^m(s,a) + \sqrt{\frac{2\mathbb{V}(P_{s,a},V^\star)\iota_{s,a}}{n^m(s,a)}} + \sqrt{\frac{2S'\mathbb{V}(P_{s,a},V^\star - V^m)\iota_{s,a}}{n^m(s,a)}} + \frac{3B_\star S'\iota_{s,a}}{n^m(s,a)} + \left(1 + c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m.$$

Proof idea. We use that V^m approximates the fixed point of $\widetilde{\mathcal{L}}$ up to an error scaling with ε_{VI}. We end up decomposing and bounding the difference $P_{s,a}V^m - \widetilde{P}_{s,a}V^m \leq (\widehat{P}_{s,a} - \widetilde{P}_{s,a})V^m + (P_{s,a} - \widehat{P}_{s,a})V^\star + (P_{s,a} - \widehat{P}_{s,a})(V^m - V^\star)$, where the first term is bounded by Lemma D.5 and D.9, while the second and third terms are bounded using the definition of the event $\mathcal{E}$. □

D.3.5 Regret Decomposition

We assume that the event $\mathcal{E}$ defined in Def. D.3 holds. In particular it guarantees that Lemma D.9 and Lemma D.13 hold for all intervals m simultaneously.

We denote by M the total number of intervals in which the first K episodes elapse. For any $M' \leq M$, we denote by $\mathcal{M}_0(M')$ the set of intervals which are among the first M' intervals, and constitute the first intervals in each episode (i.e., either it is the first interval or its previous interval ended in the goal state). We also denote by $K_{M'} \triangleq |\mathcal{M}_0(M')|$, $T_{M'} \triangleq \sum_{m=1}^{M'} H^m$ and $C_{M'} \triangleq \sum_{m=1}^{M'}\sum_{h=1}^{H^m} c_h^m$. Note that K and T are equivalent to K_M and T_M, respectively, and $C_{M'}$ is the cumulative cost in the first M' intervals.

Instead of bounding the regret R_K from Equation (3.1), we bound $\widetilde{R}_{M'} \triangleq C_{M'} - K_{M'}V^\star(s_0)$ for any fixed choice of $M' \leq M$, as done in Rosenberg et al. (2020). We see that $\widetilde{R}_M = R_K$, the true regret within K episodes. To derive Theorem 5.1, we will show that M is finite and instantiate $M' = M$. In the following we do the analysis for arbitrary $M' \leq M$ as it will be useful for the parameter-free case studied in Section D.7 (i.e., when no estimate $B \geq B_\star$ is available).

We decompose $\widetilde{R}_{M'}$ as follows

$$\begin{aligned}
\widetilde{R}_{M'} &\overset{\text{(i)}}{\leq} \sum_{m=1}^{M'}\sum_{h=1}^{H^m} c_h^m - \sum_{m\in\mathcal{M}_0(M')} V^m(s_0), \\
&\overset{\text{(ii)}}{\leq} \sum_{m=1}^{M'}\sum_{h=1}^{H^m} c_h^m + \sum_{m=1}^{M'}\left(\sum_{h=1}^{H^m} V^m(s_{h+1}^m) - V^m(s_h^m)\right) + 2SA\log_2(T_{M'}) \max_{1\leq m\leq M'}\|V^m\|_\infty \\
&\overset{\text{(iii)}}{\leq} \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\left[c_h^m + P_{s_h^m,a_h^m}V^m - V^m(s_h^m)\right] + \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\left[V^m(s_{h+1}^m) - P_{s_h^m,a_h^m}V^m\right] \\
&\quad + 2B_\star SA\log_2(T_{M'}) \\
&\overset{\text{(iv)}}{\leq} \underbrace{\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\left[V^m(s_{h+1}^m) - P_{s_h^m,a_h^m}V^m\right]}_{\triangleq X_1(M')} + \underbrace{\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\beta^m(s_h^m,a_h^m)}_{\triangleq X_2(M')} + \underbrace{\sum_{m=1}^{M'}\sum_{h=1}^{H^m} c_h^m - c(s_h^m,a_h^m)}_{\triangleq X_3(M')} \\
&\quad + 2B_\star SA\log_2(T_{M'}),
\end{aligned}$$

where (i) uses the optimism property of Lemma D.9, (ii) stems from the construction of intervals (Lemma D.15), (iii) uses that $\max_{1\leq m\leq M'}\|V^m\|_\infty \leq B_\star$ (from Lemma D.9), and (iv) comes from Lemma D.13. We now focus on bounding the terms $X_1(M')$, $X_2(M')$ and $X_3(M')$. To this end, we introduce the following useful quantities

$$X_4(M') \triangleq \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathbb{V}(P_{s_h^m,a_h^m},V^m), \qquad X_5(M') \triangleq \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathbb{V}(P_{s_h^m,a_h^m},V^\star - V^m).$$

The $X_1(M')$ term

$X_1(M')$ could be viewed as a martingale, so by taking $c = \max\{B_\star,1\}$ in the technical Lemma D.23, we have with probability at least $1-\delta$,

$$\begin{aligned}
|X_1(M')| \leq\ & 2\sqrt{2X_4(M')(\log_2((\max\{B_\star,1\})^2T_{M'}) + \ln(2/\delta))} \\
& + 5(\max\{B_\star,1\})(\log_2((\max\{B_\star,1\})^2T_{M'}) + \ln(2/\delta)).
\end{aligned}$$

To bound $X_1(M')$, we only need to bound $X_4(M')$.

The $X_3(M')$ term

Taking $c=1$ in the technical Lemma D.23, we have

$$\mathbb{P}\left[|X_3(M')| \geq 2\sqrt{2\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathrm{Var}(s_h^m,a_h^m)(\log_2(T_{M'}) + \ln(2/\delta)) + 5(\log_2(T_{M'}) + \ln(2/\delta))}\right] \leq \delta,$$

where $\mathrm{Var}(s_t, a_t) \triangleq \mathbb{E}[(c_t - c(s_t, a_t))^2]$ (c_t denotes the cost incurred at time step t). By Lemma D.26,

$$\begin{aligned}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathrm{Var}(s_h^m, a_h^m) &\le \sum_{m=1}^{M'}\sum_{h=1}^{H^m} c(s_h^m, a_h^m)\\ &= \sum_{m=1}^{M'}\sum_{h=1}^{H^m}(c(s_h^m, a_h^m) - c_h^m) + C_{M'}\\ &\le |X_3(M')| + C_{M'}.\end{aligned}$$

Therefore we have

$$\mathbb{P}\Big[|X_3(M')| \ge 2\sqrt{2(|X_3(M')| + C_{M'})(\log_2(T_{M'}) + \ln(2/\delta))} + 5(\log_2(T_{M'}) + \ln(2/\delta))\Big] \le \delta,$$

which implies that $|X_3(M')| \le O\left(\log_2(T_{M'}) + \ln(2/\delta) + \sqrt{C_{M'}(\log_2(T_{M'}) + \ln(2/\delta))}\right)$ with probability at least $1 - \delta$.

The $X_2(M')$ term

The full proof of the bound on $X_2(M')$ is deferred to Section D.4.3. Here we provide a brief sketch. First, we bound β^m and apply a pigeonhole principle to obtain

$$\begin{aligned}X_2(M') \le O\Bigg(&\sqrt{SA\log_2(T_{M'})\iota_{M'}X_4(M')} + \sqrt{S^2A\log_2(T_{M'})\iota_{M'}X_5(M')}\\ &+ \sqrt{SA\log_2(T_{M'})\iota_{M'}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\widehat{c}^m(s_h^m, a_h^m)}\\ &+ B_\star S^2A\log_2(T_{M'}) + BS^{3/2}A\log_2(T_{M'})\iota_{M'} + \sum_{m=1}^{M'}\sum_{h=1}^{H^m}(1 + c_1\sqrt{\iota_{M'}/2})\varepsilon_{\mathrm{VI}}^m\Bigg)\end{aligned}$$

with the logarithmic term $\iota_{M'} \triangleq \ln\left(\frac{12SAS'T_{M'}^2}{\delta}\right)$ which is the upper-bound of $\iota_{s,a}$ when considering only time steps in the first M' intervals. The regret contributions of the estimated costs and the VISGO precision errors are respectively

$$\begin{aligned}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\widehat{c}^m(s_h^m, a_h^m) &\le 2SA(\log_2(T_{M'}) + 1) + 2C_{M'},\\ \sum_{m=1}^{M'}\sum_{h=1}^{H^m}(1 + c_1\sqrt{\iota_{M'}/2})\varepsilon_{\mathrm{VI}}^m &= O(SA\log_2(T_{M'})\sqrt{\iota_{M'}}).\end{aligned}$$

To bound $X_4(M')$ and $X_5(M')$, we perform a recursion-based analysis on the value functions normalized by $1/B_\star$. We split the analysis on the intervals, and not on the episodes as done in

Zhang et al. (2021d). In Lemma D.17 and D.18 we establish that with overwhelming probability,

$$X_4(M') \le O\left(B_\star(C_{M'} + X_2(M')) + (B_\star^2 SA + B_\star)(\log_2(T_{M'}) + \ln(2/\delta))\right),$$
$$X_5(M') \le O\left(B_\star^2 SA(\log_2(T_{M'}) + \ln(2/\delta)) + B_\star X_2(M')\right).$$

As a result, we obtain

$$\begin{aligned} X_2(M') \le O\Big(&\sqrt{SAX_4(M')}\bar{\iota}_{M'} + \sqrt{S^2AX_5(M')}\bar{\iota}_{M'} \\ &+ SA\bar{\iota}_{M'}^{3/2} + \sqrt{SAC_{M'}}\bar{\iota}_{M'} + B_\star S^2A\bar{\iota}_{M'}^2 + BS^{3/2}A\bar{\iota}_{M'}^2\Big), \\ X_4(M') \le O&\left(B_\star(C_{M'} + X_2(M')) + (B_\star^2 SA + B_\star)\bar{\iota}_{M'}\right), \\ X_5(M') \le O&\left(B_\star^2 SA\bar{\iota}_{M'} + B_\star X_2(M')\right). \end{aligned}$$

with the logarithmic term $\bar{\iota}_{M'} \triangleq \ln\left(\frac{12SAS'T_{M'}^2}{\delta}\right) + \log_2((\max\{B_\star, 1\})^2 T_{M'}) + \ln\left(\frac{2}{\delta}\right)$. Isolating the $X_2(M')$ term finally yields

$$X_2(M') \le O((\sqrt{B_\star} + 1)\sqrt{SAC_{M'}}\bar{\iota}_{M'} + BS^2A\bar{\iota}_{M'}^2).$$

Putting Everything Together

Ultimately, with probability at least $1 - 6\delta$ we have

$$\begin{aligned} \widetilde{R}_{M'} &\le X_1(M') + X_2(M') + X_3(M') + 2B_\star SA\log_2(T_{M'}) \\ &\le O((\sqrt{B_\star} + 1)\sqrt{SAC_{M'}}\bar{\iota}_{M'} + BS^2A\bar{\iota}_{M'}^2). \end{aligned}$$

Noting that $\widetilde{R}_{M'} = C_{M'} - K_{M'}V^\star(s_0)$, we have

$$\begin{aligned} C_{M'} &\le K_{M'}V^\star(s_0) + O((\sqrt{B_\star} + 1)\sqrt{SAC_{M'}}\bar{\iota}_{M'} + BS^2A\bar{\iota}_{M'}^2), \\ C_{M'} &\overset{\text{(i)}}{\le} \left(O\left((\sqrt{B_\star} + 1)\sqrt{SA}\bar{\iota}_{M'}\right) + \sqrt{K_{M'}V^\star(s_0) + O(BS^2A\bar{\iota}_{M'}^2)}\right)^2 \\ &\le K_{M'}V^\star(s_0) + O\left((\sqrt{B_\star} + 1)\sqrt{V^\star(s_0)SAK_{M'}}\bar{\iota}_{M'} + BS^2A\bar{\iota}_{M'}^2\right) \\ &\le K_{M'}V^\star(s_0) + O\left((B_\star + \sqrt{B_\star})\sqrt{SAK_{M'}}\bar{\iota}_{M'} + BS^2A\bar{\iota}_{M'}^2\right), \end{aligned}$$

where (i) uses Lemma D.28, $V^\star(s_0) \le B_\star$ and $\sqrt{B_\star} + 1 \le O(\sqrt{B_\star + 1}) \le O(\sqrt{B})$. Hence

$$\widetilde{R}_{M'} \le O\left(\sqrt{(B_\star^2 + B_\star)SAK_{M'}}\bar{\iota}_{M'} + BS^2A\bar{\iota}_{M'}^2\right).$$

By scaling $\delta \leftarrow \delta/6$ we have the following important bound

$$\widetilde{R}_{M'} \leq O\Bigg(\sqrt{(B_\star^2 + B_\star)SAK_{M'}}\log\left(\frac{\max\{B_\star,1\}SAT_{M'}}{\delta}\right) + BS^2A\log^2\left(\frac{\max\{B_\star,1\}SAT_{M'}}{\delta}\right)\Bigg). \tag{D.6}$$

The proof of Theorem 5.1 is concluded by taking $M' = M$, where M denotes the number of intervals in which the first K episodes elapse.

D.4 Missing Proofs

D.4.1 Proofs of Lemmas D.5, D.8, D.9, D.10, D.11, D.13

Restatement of Lemma D.5. For any non-negative vector $U \in \mathbb{R}^{S'}$ such that $U(g) = 0$, for any $(s,a) \in \mathcal{S} \times \mathcal{A}$, it holds that

$$\widetilde{P}_{s,a}U \leq \widehat{P}_{s,a}U \leq \widetilde{P}_{s,a}U + \frac{\|U\|_\infty}{n(s,a)+1}, \qquad |\mathbb{V}(\widetilde{P}_{s,a},U) - \mathbb{V}(\widehat{P}_{s,a},U)| \leq \frac{2\|U\|_\infty^2 S'}{n(s,a)+1}.$$

Proof. The proof uses the definition of $\widetilde{P}$ in Equation (5.4) and simple algebraic manipulation. For any $s' \neq g$, we have $\widetilde{P}_{s,a,s'} \leq \widehat{P}_{s,a,s'}$ and $U(s') \geq 0$, as well as $U(g) = 0$, so $\widetilde{P}_{s,a}U \leq \widehat{P}_{s,a}U$, and

$$(\widehat{P}_{s,a} - \widetilde{P}_{s,a})U = \Big(1 - \frac{n(s,a)}{n(s,a)+1}\Big)\widehat{P}_{s,a}U \leq \frac{\|U\|_\infty}{n(s,a)+1}.$$

In addition, for any $s' \in \mathcal{S}'$,

$$|\widetilde{P}_{s,a,s'} - \widehat{P}_{s,a,s'}| \leq \Big|\frac{n(s,a)}{n(s,a)+1} - 1\Big|\widehat{P}_{s,a,s'} + \frac{\mathbb{I}[s'=g]}{n(s,a)+1} \leq \frac{2}{n(s,a)+1}.$$

Therefore we have that

$$\begin{aligned}\mathbb{V}(\widehat{P}_{s,a},U) &= \sum_{s'\in\mathcal{S}'}\widehat{P}_{s,a,s'}(U(s') - \widehat{P}_{s,a}U)^2 \leq \sum_{s'\in\mathcal{S}'}\widehat{P}_{s,a,s'}(U(s') - \widetilde{P}_{s,a}U)^2 \\ &\leq \sum_{s'\in\mathcal{S}'}\left(\widetilde{P}_{s,a,s'} + \frac{2}{n(s,a)+1}\right)(U(s') - \widetilde{P}_{s,a}U)^2 \leq \mathbb{V}(\widetilde{P}_{s,a},U) + \frac{2\|U\|_\infty^2 S'}{n(s,a)+1},\end{aligned}$$

where the first inequality is by the fact that $z^\star = \sum_i p_i x_i$ minimizes the quantity $\sum_i p_i (x_i - z)^2$. Conversely,

$$\begin{aligned}\mathbb{V}(\widetilde{P}_{s,a}, U) &= \sum_{s' \in \mathcal{S}'} \widetilde{P}_{s,a,s'} (U(s') - \widetilde{P}_{s,a} U)^2 \le \sum_{s' \in \mathcal{S}'} \widetilde{P}_{s,a,s'} (U(s') - \widehat{P}_{s,a} U)^2 \\ &\le \sum_{s' \in \mathcal{S}'} \left(\widehat{P}_{s,a,s'} + \frac{2}{n(s,a)+1} \right) (U(s') - \widehat{P}_{s,a} U)^2 \le \mathbb{V}(\widehat{P}_{s,a}, U) + \frac{2 \|U\|_\infty^2 S'}{n(s,a)+1}.\end{aligned}$$

□

Restatement of Lemma D.8. Let $\Upsilon \triangleq \{v \in \mathbb{R}^{S'} : v \ge 0,\ v(g) = 0,\ \|v\|_\infty \le B\}$. Let $f : \Delta^{S'} \times \Upsilon \times \mathbb{R} \times \mathbb{R} \times \mathbb{R} \to \mathbb{R}$ with $f(p, v, n, B, \iota) \triangleq pv - \max\left\{c_1 \sqrt{\frac{\mathbb{V}(p,v)\iota}{n}},\ c_2 \frac{B\iota}{n}\right\}$, with $c_1 = 6$ and $c_2 = 36$ (here taking any pair of constants such that $c_1^2 \le c_2$ works). Then f satisfies, for all $p \in \Delta^{S'}$, $v \in \Upsilon$ and $n, \iota > 0$,

1. $f(p, v, n, B, \iota)$ is non-decreasing in $v(s)$, i.e.,

$$\forall (v, v') \in \Upsilon^2,\ v \le v' \implies f(p, v, n, B, \iota) \le f(p, v', n, B, \iota);$$

2. $f(p, v, n, B, \iota) \le pv - \frac{c_1}{2}\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} - \frac{c_2}{2}\frac{B\iota}{n} \le pv - 2\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} - 14\frac{B\iota}{n}$;

3. If $p(g) > 0$, then $f(p, v, n, B, \iota)$ is ρ_p-contractive in $v(s)$, with $\rho_p \triangleq 1 - p(g) < 1$, i.e.,

$$\forall (v, v') \in \Upsilon^2,\ |f(p, v, n, B, \iota) - f(p, v', n, B, \iota)| \le \rho_p \|v - v'\|_\infty.$$

Proof. The second claim holds by $\max\{x, y\} \ge (x + y)/2, \forall x, y$, by the choices of c_1, c_2 and because both $\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}}$ and $\frac{B\iota}{n}$ are non-negative. To verify the first and third claims, we fix all other variables but $v(s)$ and view f as a function in $v(s)$. Because the derivative of f in $v(s)$ does not exist only when $c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} = c_2\frac{B\iota}{n}$, where the condition has at most two solutions, it suffices to prove that $\frac{\partial f}{\partial v(s)} \ge 0$ when $c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} \ne c_2\frac{B\iota}{n}$. Direct computation gives

$$\begin{aligned}\frac{\partial f}{\partial v(s)} &= p(s) - c_1 \mathbb{I}\left[c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} \ge c_2\frac{B\iota}{n}\right] \frac{p(s)(v(s) - pv)\iota}{\sqrt{n\mathbb{V}(p,v)\iota}} \\ &\ge \min\left\{p(s),\ p(s) - \frac{c_1^2}{c_2 B} p(s)(v(s) - pv)\right\} \\ &\overset{\text{(i)}}{\ge} \min\left\{p(s),\ p(s) - \frac{c_1^2}{c_2} p(s)\right\} \\ &\ge p(s)\left(1 - \frac{c_1^2}{c_2}\right) = 0.\end{aligned}$$

Here (i) is by $v(s) - pv \le v(s) \le B$. For the third claim, we perform a distinction of cases. If $c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} = c_2\frac{B\iota}{n}$, where the condition has at most two solutions, then $f(v) = pv - c_2\frac{B\iota}{n}$,

which corresponds to a ρ_p-contraction since

$$|f(v_1) - f(v_2)| = \left|\sum_{s\in\mathcal{S}} p(s)(v_1(s) - v_2(s))\right| \leq \sum_{s\in\mathcal{S}} p(s)\cdot\|v_1 - v_2\|_\infty = (1-p(g))\|v_1 - v_2\|_\infty.$$

Otherwise $c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} \neq c_2\frac{B\iota}{n}$, then the derivative of f in $v(s)$ exists and it verifies

$$\begin{aligned}
\left\|\frac{\partial f}{\partial v}\right\|_1 &= \sum_{s\in\mathcal{S}}\left|\frac{\partial f}{\partial v(s)}\right| = \sum_{s\in\mathcal{S}}\frac{\partial f}{\partial v(s)} \\
&= \sum_{s\in\mathcal{S}}\left[p(s) - c_1\mathbb{I}\left[c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} \geq c_2\frac{B\iota}{n}\right]\frac{p(s)(v(s)-pv)\iota}{\sqrt{n\mathbb{V}(p,v)\iota}}\right] \\
&= 1 - p(g) - c_1\mathbb{I}\left[c_1\sqrt{\frac{\mathbb{V}(p,v)\iota}{n}} \geq c_2\frac{B\iota}{n}\right]\sqrt{\frac{\iota}{n\mathbb{V}(p,v)}}[pv - (1-p(g))\cdot pv]\} \\
&\leq 1 - p(g).
\end{aligned}$$

In this case, by the mean value theorem we obtain that f is ρ_p-contractive. □

Restatement of Lemma D.9. Conditioned on the event $\mathcal{E}$, for any output Q of the VISGO procedure (line 22 of Algorithm 5.1) and for any state-action pair $(s,a)\in\mathcal{S}\times\mathcal{A}$, it holds that

$$Q(s,a) \leq Q^\star(s,a).$$

Proof. We prove by induction that for any inner iteration i of VISGO, $Q^{(i)}(s,a) \leq Q^\star(s,a)$. By definition we have $Q^{(0)} = 0 \leq Q^\star$. Assume that the property holds for iteration i, then

$$Q^{(i+1)}(s,a) = \max\left\{\widehat{c}(s,a) + \widetilde{P}_{s,a}V^{(i)} - b^{(i+1)}(s,a), 0\right\},$$

where

$$\begin{aligned}
&\widehat{c}(s,a) + \widetilde{P}_{s,a}V^{(i)} - b^{(i+1)}(s,a) \\
&= \widehat{c}(s,a) + \widetilde{P}_{s,a}V^{(i)} - \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(i)})\iota_{s,a}}{n^+(s,a)}}, c_2\frac{B\iota_{s,a}}{n^+(s,a)}\right\} - c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(i)}}{\leq} c(s,a) + \widetilde{P}_{s,a}V^{(i)} - \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(i)})\iota_{s,a}}{n^+(s,a)}}, c_2\frac{B\iota_{s,a}}{n^+(s,a)}\right\} + \frac{28\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&= c(s,a) + f(\widetilde{P}_{s,a},V^{(i)},n^+(s,a),B,\iota_{s,a}) + \frac{28\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(ii)}}{\leq} c(s,a) + f(\widetilde{P}_{s,a},V^\star,n^+(s,a),B,\iota_{s,a}) + \frac{28\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(iii)}}{\leq} c(s,a) + \widetilde{P}_{s,a}V^\star - 2\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^\star)\iota_{s,a}}{n^+(s,a)}} - \frac{14B\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)}
\end{aligned}$$

$$
\begin{aligned}
&\overset{\text{(iv)}}{\leq} c(s,a) + \widehat{P}_{s,a}V^\star - 2\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a}, V^\star)\iota_{s,a}}{n^+(s,a)}} - \frac{14B\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(v)}}{\leq} c(s,a) + P_{s,a}V^\star + 2\sqrt{\frac{\mathbb{V}(\widehat{P}_{s,a}, V^\star)\iota_{s,a}}{n^+(s,a)}} - 2\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a}, V^\star)\iota_{s,a}}{n^+(s,a)}} - (B - B_\star)\frac{14\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(vi)}}{\leq} c(s,a) + P_{s,a}V^\star + 2\sqrt{\frac{|\mathbb{V}(\widehat{P}_{s,a}, V^\star) - \mathbb{V}(\widetilde{P}_{s,a}, V^\star)|\iota_{s,a}}{n^+(s,a)}} - (B - B_\star)\frac{14\iota_{s,a}}{3n^+(s,a)} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(vii)}}{\leq} \underbrace{c(s,a) + P_{s,a}V^\star}_{=Q^\star(s,a)} - (B - B_\star)\left(\frac{14\iota_{s,a}}{3n^+(s,a)} + \frac{2\sqrt{2S'\iota_{s,a}}}{n^+(s,a)}\right) \\
&\leq Q^\star(s,a),
\end{aligned}
$$

where (i) is by definition of $\mathcal{E}_2$ and choice of c_3, (ii) uses the first property of Lemma D.8 and the induction hypothesis that $V^{(i)} \leq V^\star$, (iii) uses the second property of Lemma D.8 and assumption $B \geq \max\{B_\star, 1\}$, (iv) uses Lemma D.5, (v) is by definition of $\mathcal{E}_1$, (vi) uses the inequality $|\sqrt{x} - \sqrt{y}| \leq \sqrt{|x-y|}, \forall x, y \geq 0$, and (vii) uses the second inequality of Lemma D.5 and the choice of c_4. Ultimately,

$$
Q^{(i+1)}(s,a) \leq \max\left\{Q^\star(s,a), 0\right\} = Q^\star(s,a).
$$

□

Restatement of Lemma D.10. The sequence $(V^{(i)})_{i\geq 0}$ is non-decreasing. Combining this with the fact that it is upper bounded by $V^\star$ from Lemma D.9, the sequence must converge.

Proof. We recognize that $V^{(i+1)}(s) \leftarrow \min_a Q^{(i+1)}(s,a)$, with

$$
Q^{(i+1)}(s,a) \leftarrow \max\left\{\widehat{c}(s,a) + \underbrace{f(\widetilde{P}_{s,a}, V^{(i)}, n^+(s,a), B, \iota_{s,a})}_{\triangleq g_{s,a}(V^{(i)})} - c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)},\ 0\right\},
$$

where we introduce the function $g_{s,a}(V) \triangleq f(\widetilde{P}_{s,a}, V, n^+(s,a), B, \iota_{s,a})$ for notational ease as all other parameters (apart from V) will remain the same throughout the analysis.

We prove by induction on the iterations indexed by i that $Q^{(i)} \leq Q^{(i+1)}$. First, it holds that $Q^{(0)} = 0 \leq Q^{(1)}$. Now assume that $Q^{(i-1)} \leq Q^{(i)}$. Then

$$
\begin{aligned}
Q^{(i+1)}(s,a) &= \max\left\{\widehat{c}(s,a) + g_{s,a}(V^{(i)}) - c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)},\ 0\right\} \\
&\geq \max\left\{\widehat{c}(s,a) + g_{s,a}(V^{(i-1)}) - c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} - c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)},\ 0\right\} \\
&= Q^{(i)}(s,a),
\end{aligned}
$$

where the inequality uses the induction hypothesis $V^{(i)} \geq V^{(i-1)}$ and the fact that $g_{s,a}$ is non-decreasing from the first claim of Lemma D.8. □

Restatement of Lemma D.11. Denote by $\nu > 0$ the probability of reaching the goal from any state-action pair in $\widetilde{P}$, i.e., $\nu \triangleq \min_{s,a} \widetilde{P}_{s,a,g}$. Then $\widetilde{\mathcal{L}}$ is a ρ-contractive operator with modulus $\rho \triangleq 1 - \nu < 1$.

Proof. Take any two vectors U_1, U_2, then for any state $s \in \mathcal{S}$,

$$\begin{aligned} |\widetilde{\mathcal{L}}U_1(s) - \widetilde{\mathcal{L}}U_2(s)| &= \left| \min_a \widetilde{\mathcal{L}}U_1(s,a) - \min_a \widetilde{\mathcal{L}}U_2(s,a) \right| \\ &\leq \left| \max_a \left\{ \widetilde{\mathcal{L}}U_1(s,a) - \widetilde{\mathcal{L}}U_2(s,a) \right\} \right|, \end{aligned}$$

and we have that for any action $a \in \mathcal{A}$,

$$\begin{aligned} |\widetilde{\mathcal{L}}U_1(s,a) - \widetilde{\mathcal{L}}U_2(s,a)| &\leq \left| \max\left\{\widehat{c}(s,a) + g_{s,a}(U_1),\ 0\right\} - \max\left\{\widehat{c}(s,a) + g_{s,a}(U_2),\ 0\right\} \right| \\ &\leq \left| g_{s,a}(U_1) - g_{s,a}(U_2) \right| \\ &\overset{\text{(i)}}{\leq} \rho_{s,a} \|U_1 - U_2\|_\infty. \end{aligned}$$

The third claim of Lemma D.8 is employed to justify inequality (i): $g_{s,a}$ is $\rho_{s,a}$-contractive (where $g_{s,a}$ is defined in the proof of Lemma D.10) with modulus

$$\rho_{s,a} \triangleq 1 - \widetilde{P}_{s,a,g} = 1 - \nu_{s,a}.$$

Taking the maximum over (s,a) pairs, $\widetilde{\mathcal{L}}$ is thus ρ-contractive with modulus $\rho \triangleq 1 - \nu < 1$. □

Restatement of Lemma D.13. Conditioned on the event $\mathcal{E}$, for any interval m and state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$,

$$|c(s,a) + P_{s,a}V^m - Q^m(s,a)| \leq \min\left\{\beta^m(s,a), B_\star + 1\right\},$$

where we define

$$\begin{aligned} \beta^m(s,a) \triangleq 4b^m(s,a) &+ \sqrt{\frac{2\mathbb{V}(P_{s,a}, V^\star)\iota_{s,a}}{n^m(s,a)}} + \sqrt{\frac{2S'\mathbb{V}(P_{s,a}, V^\star - V^m)\iota_{s,a}}{n^m(s,a)}} \\ &+ \frac{3B_\star S'\iota_{s,a}}{n^m(s,a)} + \left(1 + c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m. \end{aligned}$$

Proof. First we see that $c(s,a) + P_{s,a}V^m - Q^m(s,a) \leq c(s,a) + P_{s,a}V^\star = Q^\star(s,a) \leq B_\star + 1$ and that $Q^m(s,a) - c(s,a) - P_{s,a}V^m \leq Q^\star(s,a) \leq B_\star + 1$, from Lemma D.9 and the Bellman optimality equation (Proposition 2.11). Now we prove that $|c(s,a) + P_{s,a}V^m - Q^m(s,a)| \leq \beta^m(s,a)$.

Bounding $c(s,a)+P_{s,a}V^m-Q^m(s,a)$. From the VISGO loop of Algorithm 5.1, the vectors Q^m and V^m can be associated to a finite iteration l of a sequence of vectors $(Q^{(i)})_{i\geq 0}$ and $(V^{(i)})_{i\geq 0}$ such that

(i) $Q^m(s,a) \triangleq Q^{(l)}(s,a)$,
(ii) $V^m(s) \triangleq V^{(l)}(s)$,
(iii) $\|V^{(l)}-V^{(l-1)}\|_\infty \leq \varepsilon_{\text{VI}}^m$,
(iv) $b^m(s,a) \triangleq b^{(l+1)}(s,a) = \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(l)})\iota_{s,a}}{n^m(s,a)}},\ c_2\frac{B\iota_{s,a}}{n^m(s,a)}\right\} + c_3\sqrt{\frac{\widehat{c}^m(s,a)\iota_{s,a}}{n^m(s,a)}} + c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^m(s,a)}$.

First, we examine the gap between the exploration bonuses at the final VISGO iterations l and $l+1$ as follows

$$
\begin{aligned}
b^{(l)}(s,a) &\overset{\text{(i)}}{\leq} c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(l-1)})\iota_{s,a}}{n^+(s,a)}} + c_2\frac{B\iota_{s,a}}{n^+(s,a)} + c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} + c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(ii)}}{\leq} c_1\sqrt{2\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(l)})\iota_{s,a}}{n^+(s,a)}} + c_1\sqrt{2\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(l-1)}-V^{(l)})\iota_{s,a}}{n^+(s,a)}} + c_2\frac{B\iota_{s,a}}{n^+(s,a)} \\
&\quad + c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} + c_4\frac{B\sqrt{S'\iota_{s,a}}}{n^+(s,a)} \\
&\overset{\text{(iii)}}{\leq} 2\sqrt{2}b^{(l+1)}(s,a) + c_1\sqrt{\frac{(\varepsilon_{\text{VI}}^m)^2\iota_{s,a}}{2n^+(s,a)}} \\
&\leq 2\sqrt{2}b^{(l+1)}(s,a) + \varepsilon_{\text{VI}}^m c_1\sqrt{\iota_{s,a}/2},
\end{aligned}
$$

where (i) uses $\max\{x,\ y\} \leq x+y$; (ii) uses $\mathbb{V}(P,X+Y) \leq 2(\mathbb{V}(P,X)+\mathbb{V}(P,Y))$ and $\sqrt{x+y} \leq \sqrt{x}+\sqrt{y}$; (iii) uses $x+y \leq 2\max\{x,\ y\}$ and Popoviciu's inequality (Lemma D.21) applied to $V^{(l-1)}-V^{(l)} \in [-\varepsilon_{\text{VI}}^m, 0]$. Moreover, we have that $Q^{(l)}(s,a) \geq \widehat{c}(s,a) + \widetilde{P}_{s,a}V^{(l-1)} - b^{(l)}(s,a)$ from Equation (5.2). Combining everything yields

$$
\begin{aligned}
-Q^m(s,a) &\leq -\widehat{c}(s,a) - \widetilde{P}_{s,a}(V^m-\varepsilon_{\text{VI}}) + \varepsilon_{\text{VI}}c_1\sqrt{\iota_{s,a}/2} + 2\sqrt{2}b^m(s,a) \\
&\leq -\widehat{c}(s,a) - \widetilde{P}_{s,a}V^m + 2\sqrt{2}b^m(s,a) + \left(1+c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m.
\end{aligned}
$$

Therefore, we have

$$
\begin{aligned}
&c(s,a)+P_{s,a}V^m-Q^m(s,a) \\
&\leq c(s,a)+P_{s,a}V^m-\widehat{c}^m(s,a)-\widetilde{P}_{s,a}V^m+2\sqrt{2}b^m(s,a)+\left(1+c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m \\
&\overset{\text{(i)}}{\leq} P_{s,a}V^m-\widehat{P}_{s,a}V^m+\frac{B_\star}{n^m(s,a)+1}+4b^m(s,a)+\left(1+c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m \\
&\leq \underbrace{(P_{s,a}-\widehat{P}_{s,a})V^\star}_{\triangleq Y_1} + \underbrace{(P_{s,a}-\widehat{P}_{s,a})(V^m-V^\star)}_{\triangleq Y_2} + \frac{B_\star}{n^m(s,a)} + 4b^m(s,a) + \left(1+c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m,
\end{aligned}
$$

where (i) comes from Lemma D.5, the event $\mathcal{E}_2$, Lemma D.9 and (loosely) bounding $|c(s,a) - \widehat{c}(s,a)| \leq b^m(s,a)$. It holds under the event $\mathcal{E}_1$ that

$$|Y_1| \leq \sqrt{\frac{2\mathbb{V}(P_{s,a}, V^\star)\iota_{s,a}}{n^m(s,a)}} + \frac{B_\star \iota_{s,a}}{n^m(s,a)}.$$

Moreover, we have

$$\begin{aligned}
|Y_2| &\overset{\text{(i)}}{=} \left|\sum_{s'} (\widehat{P}_{s,a,s'} - P_{s,a,s'})(V^m(s') - V^\star(s') - P_{s,a}(V^m - V^\star))\right| \\
&\leq \sum_{s'} |P_{s,a,s'} - \widehat{P}_{s,a,s'}||V^m(s') - V^\star(s') - P_{s,a}(V^m - V^\star)| \\
&\overset{\text{(ii)}}{\leq} \sum_{s'} \sqrt{\frac{2P_{s,a,s'}\iota_{s,a}}{n^m(s,a)}}|V^m(s') - V^\star(s') - P_{s,a}(V^m - V^\star)| + \frac{B_\star S' \iota_{s,a}}{n^m(s,a)} \\
&\overset{\text{(iii)}}{\leq} \sqrt{\frac{2S'\mathbb{V}(P_{s,a}, V^m - V^\star)\iota_{s,a}}{n^m(s,a)}} + \frac{B_\star S' \iota_{s,a}}{n^m(s,a)},
\end{aligned}$$

where the shift performed in (i) is by $\sum_{s'} P_{s,a,s'} = \sum_{s'} \widehat{P}_{s,a,s'} = 1$; (ii) holds under the event $\mathcal{E}_3$ and Lem. D.9 ($V^m(s) \in [0, B_\star]$); (iii) is by Cauchy-Schwarz inequality.

Bounding $Q^m(s,a) - c(s,a) - P_{s,a}V^m$. If $Q^m(s,a) = Q^{(l)}(s,a) = 0$, then $Q^m(s,a) - \widehat{c}(s,a) - P_{s,a}V^m \leq 0 \leq \min\{\beta^m(s,a), B_\star\}$. Otherwise, we have $Q^m(s,a) = Q^{(l)}(s,a) = \widehat{c}(s,a) + \widetilde{P}_{s,a}V^{(l-1)} - b^{(l)}(s,a)$. Using that $V^m \geq V^{(l-1)}$ (Lemma D.10) and $\widehat{P}_{s,a}V^m \geq \widetilde{P}_{s,a}V^m$ (Lemma D.5), we get

$$\begin{aligned}
Q^m(s,a) - c(s,a) - P_{s,a}V^m &\leq Q^m(s,a) - \widehat{c}(s,a) - P_{s,a}V^m + b^m(s,a) \\
&= \widetilde{P}_{s,a}V^{(l-1)} - b^{(l)}(s,a) - P_{s,a}V^m + b^m(s,a) \\
&\leq \widehat{P}_{s,a}V^m - P_{s,a}V^m + b^m(s,a) \\
&= (\widehat{P}_{s,a} - P_{s,a})V^\star - (\widehat{P}_{s,a} - P_{s,a})(V^\star - V^m) + b^m(s,a) \\
&\leq |Y_1| + |Y_2| + b^m(s,a),
\end{aligned}$$

which can be bounded as above. □

D.4.2 Additional lemmas

Lemma D.14. *Let* $\widetilde{Q}^m(s,a) \triangleq Q^\star(s,a) - Q^m(s,a)$ *and* $\widetilde{V}^m(s) \triangleq V^\star(s) - V^m(s)$. *Then conditioned on the event* $\mathcal{E}$, *we have that for all* (s,a,m,h),

$$\widetilde{V}(s_h^m) - P_{s_h^m,a_h^m}\widetilde{V}(s_{h+1}^m) \leq \beta^m(s_h^m, a_h^m).$$

Proof. We write that

$$\begin{aligned}
\widetilde{V}^m(s_h^m) - P_{s_h^m,a_h^m}\widetilde{V}^m(s_{h+1}^m) &= V^\star(s_h^m) - P_{s_h^m,a_h^m}V^\star + P_{s_h^m,a_h^m}V^m - V^m(s_h^m) \\
&\leq Q^\star(s_h^m, a_h^m) - P_{s_h^m,a_h^m}V^\star + P_{s_h^m,a_h^m}V^m - V^m(s_h^m) \\
&\overset{\text{(i)}}{=} c(s_h^m, a_h^m) + P_{s_h^m,a_h^m}V^m - Q^m(s_h^m, a_h^m) \\
&\overset{\text{(ii)}}{\leq} \beta^m(s_h^m, a_h^m),
\end{aligned}$$

where (i) uses the Bellman optimality equation (Proposition 2.11) and the fact that $V^m(s_h^m) = Q^m(s_h^m, a_h^m)$, and (ii) comes from Lemma D.13. □

Lemma D.15. *For any* $M' \leq M$, *it holds that*

$$\sum_{m=1}^{M'}\left(\sum_{h=1}^{H^m} V^m(s_h^m) - V^m(s_{h+1}^m)\right) - \sum_{m\in\mathcal{M}_0(M')} V^m(s_0) \leq 2SA\log_2(T_{M'}) \max_{1\leq m\leq M'}\|V^m\|_\infty.$$

Proof. We recall that we denote by $\mathcal{M}_0(M')$ the set of intervals among the first M' intervals that constitute the first intervals in each episode. From the analytical construction of intervals, an interval $m < M'$ can end due to one of the following three conditions:

(i) If interval m ends in the goal state, then

$$V^{m+1}(s_1^{m+1}) - V^m(s_{H^m+1}^m) = V^{m+1}(s_0) - V^m(g) = V^{m+1}(s_0).$$

This happens for all the intervals $m+1 \in \mathcal{M}_0(M')$.

(ii) If interval m ends when the count to a state-action pair is doubled, then we replan with a VISGO procedure. Thus we get

$$V^{m+1}(s_1^{m+1}) - V^m(s_{H^m+1}^m) \leq V^{m+1}(s_1^{m+1}) \leq \max_{1\leq m\leq M'}\|V^m\|_\infty.$$

This happens at most $2SA\log_2(T_{M'})$ times.

Combining the three conditions above implies that

$$
\begin{aligned}
&\sum_{m=1}^{M'}\left(\sum_{h=1}^{H^m} V^m(s_h^m) - V^m(s_{h+1}^m)\right) \\
&= \sum_{m=1}^{M'} V^m(s_1^m) - V^m(s_{H^m+1}^m) \\
&= \sum_{m=1}^{M'-1}\left(V^{m+1}(s_1^{m+1}) - V^m(s_{H^m+1}^m)\right) + \underbrace{\sum_{m=1}^{M'-1}\left(V^m(s_1^m) - V^{m+1}(s_1^{m+1})\right)}_{=V^1(s_1^1)-V^{M'}(s_1^{M'})} + V^{M'}(s_1^{M'}) \underbrace{-V^{M'}(s_{H^{M'}+1}^{M'})}_{\leq 0} \\
&\leq \sum_{m=1}^{M'-1}\left(V^{m+1}(s_1^{m+1}) - V^m(s_{H^m+1}^m)\right) + V^1(s_0) \\
&\leq \sum_{m=1}^{M'-1} V^{m+1}(s_0)\mathbb{I}[m+1 \in \mathcal{M}_0(M')] + 2SA\log_2(T_{M'}) \max_{1\leq m\leq M'} \|V^m\|_\infty + V^1(s_0) \\
&= \sum_{m\in\mathcal{M}_0(M')} V^m(s_0) + 2SA\log_2(T_{M'}) \max_{1\leq m\leq M'} \|V^m\|_\infty.
\end{aligned}
$$

□

D.4.3 Full proof of the bound on $X_2(M')$

① First, bound β^m.

Recall that we assume that the event $\mathcal{E}$ holds. From Lemma D.13, we have for any m, s, a,

$$
\begin{aligned}
\beta^m(s,a) = O\Bigg(&\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a}, V^m)\iota_{s,a}}{n^m(s,a)}} + \sqrt{\frac{\mathbb{V}(P_{s,a}, V^\star)\iota_{s,a}}{n^m(s,a)}} + \sqrt{\frac{S\mathbb{V}(P_{s,a}, V^\star - V^m)\iota_{s,a}}{n^m(s,a)}} \\
&+ \sqrt{\frac{\widehat{c}^m(s,a)\iota_{s,a}}{n^m(s,a)}} + \frac{B_\star S\iota_{s,a}}{n^m(s,a)} + \frac{B\sqrt{S}\iota_{s,a}}{n^m(s,a)} + \left(1 + c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\mathrm{VI}}^m\Bigg).
\end{aligned}
$$

Here we interchange S' and S since we use the $O()$ notation. From Lemma D.5 and Lemma D.9, for any m, s, a,

$$
\mathbb{V}(\widetilde{P}_{s,a}, V^m) \leq \mathbb{V}(\widehat{P}_{s,a}, V^m) + \frac{2B_\star^2 S'}{n^m(s,a)+1} < \mathbb{V}(\widehat{P}_{s,a}, V^m) + \frac{2B_\star^2 S'}{n^m(s,a)}.
$$

Under the event $\mathcal{E}_3$, it holds that

$$
\widehat{P}_{s,a,s'} \leq P_{s,a,s'} + \sqrt{\frac{2P_{s,a,s'}\iota_{s,a}}{n^m(s,a)}} + \frac{\iota_{s,a}}{n^m(s,a)} \leq \frac{3}{2}P_{s,a,s'} + \frac{2\iota_{s,a}}{n^m(s,a)}.
$$

Thus, it holds that for any m, s, a,

$$\begin{aligned}\mathbb{V}(\widehat{P}_{s,a}, V^m) &= \sum_{s'} \widehat{P}_{s,a,s'} \left(V^m(s') - \widehat{P}_{s,a} V^m\right)^2 \\ &\overset{\text{(i)}}{\leq} \sum_{s'} \widehat{P}_{s,a,s'} \left(V^m(s') - P_{s,a} V^m\right)^2 \\ &\leq \sum_{s'} \left(\frac{3}{2} P_{s,a,s'} + \frac{2\iota_{s,a}}{n^m(s,a)}\right) \left(V^m(s') - P_{s,a} V^m\right)^2 \\ &\leq \frac{3}{2} \mathbb{V}(P_{s,a}, V^m) + \frac{2B_\star^2 S' \iota_{s,a}}{n^m(s,a)}.\end{aligned}$$

(i) is by the fact that $z^\star = \sum_i p_i x_i$ minimizes the quantity $\sum_i p_i (x_i - z)^2$. As a result,

$$\mathbb{V}(\widetilde{P}_{s,a}, V^m) < \frac{3}{2} \mathbb{V}(P_{s,a}, V^m) + \frac{2B_\star^2 S'}{n^m(s,a)} + \frac{2B_\star^2 S' \iota_{s,a}}{n^m(s,a)}.$$

Utilizing $\mathbb{V}(P, X+Y) \leq 2(\mathbb{V}(P,X) + \mathbb{V}(P,Y))$ with $X = V^\star - V^m$ and $Y = V^m$ and $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$, finally we have

$$\begin{aligned}\beta^m(s,a) \leq O\Bigg(&\sqrt{\frac{\mathbb{V}(P_{s,a}, V^m)\iota_{s,a}}{n^m(s,a)}} + \sqrt{\frac{S\mathbb{V}(P_{s,a}, V^\star - V^m)\iota_{s,a}}{n^m(s,a)}} \\ &+ \sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^m(s,a)}} + \frac{B_\star S \iota_{s,a}}{n^m(s,a)} + \frac{B\sqrt{S}\iota_{s,a}}{n^m(s,a)} + \left(1 + c_1\sqrt{\iota_{s,a}/2}\right)\varepsilon_{\text{VI}}^m\Bigg).\end{aligned}$$

② Second, bound a special type of summation.

Lemma D.16. *Let $w = \{w_h^m \geq 0 \,:\, 1 \leq m \leq M,\ 1 \leq h \leq H^m\}$ be a group of weights, then for any $M' \leq M$,*

$$\sum_{m=1}^{M'} \sum_{h=1}^{H^m} \sqrt{\frac{w_h^m}{n^m(s_h^m, a_h^m)}} \leq O\left(\sqrt{SA\log_2(T_{M'}) \sum_{m=1}^{M'} \sum_{h=1}^{H^m} w_h^m}\right).$$

Proof. For $m \leq M'$, $n^m(s,a) \in \{2^i \,:\, i \in \mathbb{N}, i \leq \log_2(T_{M'})\}$. We can count the occurrences of a fixed value of $n^m(s,a)$ by the doubling property of VISGO: $\forall i, s, a$

$$\sum_{m=1}^{M'} \sum_{h=1}^{H^m} \mathbb{I}[(s_h^m, a_h^m) = (s,a), n^m(s,a) = 2^i] \leq 2^i.$$

Thus

$$\begin{aligned}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\frac{1}{n^m(s_h^m,a_h^m)} &= \sum_{s,a}\sum_{0\le i\le \log_2(T_{M'})}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathbb{I}[(s_h^m,a_h^m)=(s,a), n^m(s,a)=2^i]\frac{1}{2^i}\\ &= \sum_{s,a}\sum_{0\le i\le \log_2(T_{M'})}1\\ &\le SA(\log_2(T_{M'})+1) \qquad \text{(D.7)}\\ &\le O(SA\log_2(T_{M'})).\end{aligned}$$

By Cauchy-Schwarz inequality,

$$\begin{aligned}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\sqrt{\frac{w_h^m}{n^m(s_h^m,a_h^m)}} &\le \sqrt{\left(\sum_{m=1}^{M'}\sum_{h=1}^{H^m}w_h^m\right)\left(\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\frac{1}{n^m(s_h^m,a_h^m)}\right)}\\ &\le O\left(\sqrt{SA\log_2(T_{M'})\sum_{m=1}^{M'}\sum_{h=1}^{H^m}w_h^m}\right).\end{aligned}$$

□

By setting successively $w_h^m = \mathbb{V}(P_{s_h^m,a_h^m},V^m)$, $\mathbb{V}(P_{s_h^m,a_h^m},V^\star-V^m)$ and $\widehat{c}(s_h^m,a_h^m)$, and relaxing $\iota_{s_h^m,a_h^m}$ to its upper-bound $\iota_{M'}=\ln\left(\frac{12SAS'T_{M'}^2}{\delta}\right)$ we have

$$\begin{aligned}X_2(M') \le O\Bigg(&\sqrt{SA\log_2(T_{M'})\iota_{M'}\underbrace{\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathbb{V}(P_{s_h^m,a_h^m},V^m)}_{\triangleq X_4(M')}}\\ &+\sqrt{S^2A\log_2(T_{M'})\iota_{M'}\underbrace{\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathbb{V}(P_{s_h^m,a_h^m},V^\star-V^m)}_{\triangleq X_5(M')}}\\ &+\sqrt{SA\log_2(T_{M'})\iota_{M'}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\widehat{c}(s_h^m,a_h^m)}+B_\star S^2A\log_2(T_{M'})\\ &+BS^{3/2}A\log_2(T_{M'})\iota_{M'}+\sum_{m=1}^{M'}\sum_{h=1}^{H^m}(1+c_1\sqrt{\iota_{M'}/2})\varepsilon_{\text{VI}}^m\Bigg).\end{aligned}$$

③ Third, bound each summation separately.

Regret contribution of the estimated costs. From line 15 in EB-SSP, we have that $\widehat{c}(s,a) \leq \frac{2\theta(s,a)}{N(s,a)}$. Let $\theta^m(s,a)$ denote the value of $\theta(s,a)$ for calculating $\widehat{c}^m$. By definition,

$$\begin{aligned}
\theta^m(s_h^m, a_h^m) = &\sum_{m'=1}^{M'} \sum_{h'=1}^{H^{m'}} \mathbb{I}[(s_h^m, a_h^m) = (s_{h'}^{m'}, a_{h'}^{m'}),\ n^m(s_h^m, a_h^m) = 2n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})]c_{h'}^{m'} \\
&- \mathbb{I}[\text{first occurrence of } (m',h') \text{ such that } (s_h^m, a_h^m) = (s_{h'}^{m'}, a_{h'}^{m'}),\ n^m(s_h^m, a_h^m) = 2n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})]c_{h'}^{m'} \\
&+ \mathbb{I}[\text{first occurrence of } (m',h') \text{ such that } (s_h^m, a_h^m) = (s_{h'}^{m'}, a_{h'}^{m'}),\ n^m(s_h^m, a_h^m) = n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})]c_{h'}^{m'} \\
\leq &\sum_{m'=1}^{M'} \sum_{h'=1}^{H^{m'}} \mathbb{I}[(s_h^m, a_h^m) = (s_{h'}^{m'}, a_{h'}^{m'}),\ n^m(s_h^m, a_h^m) = 2n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})]c_{h'}^{m'} + 1.
\end{aligned}$$

For any $M' \leq M$ we have

$$\begin{aligned}
&\sum_{m=1}^{M'} \sum_{h=1}^{H^m} \widehat{c}^m(s_h^m, a_h^m) \\
&\leq \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \frac{2\theta^m(s_h^m, a_h^m)}{n^m(s_h^m, a_h^m)} \\
&= \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \sum_{m'=1}^{M'} \sum_{h'=1}^{H^{m'}} \mathbb{I}[(s_h^m, a_h^m) = (s_{h'}^{m'}, a_{h'}^{m'}),\ n^m(s_h^m, a_h^m) = 2n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})] \frac{2c_{h'}^{m'}}{n^m(s_h^m, a_h^m)} \\
&\quad + \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \frac{2}{n^m(s_h^m, a_h^m)} \\
&\overset{\text{(i)}}{\leq} \sum_{m'=1}^{M'} \sum_{h'=1}^{H^{m'}} \frac{c_{h'}^{m'}}{n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})} \cdot \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \mathbb{I}[(s_h^m, a_h^m) = (s_{h'}^{m'}, a_{h'}^{m'}),\ n^m(s_h^m, a_h^m) = 2n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})] \\
&\quad + 2SA(\log_2(T_{M'}) + 1) \\
&\leq 2SA(\log_2(T_{M'}) + 1) + \sum_{m'=1}^{M'} \sum_{h'=1}^{H^{m'}} \frac{c_{h'}^{m'}}{n^{m'}(s_{h'}^{m'}, a_{h'}^{m'})} \cdot 2n^{m'}(s_{h'}^{m'}, a_{h'}^{m'}) \\
&= 2SA(\log_2(T_{M'}) + 1) + 2\sum_{m'=1}^{M'} \sum_{h'=1}^{H^{m'}} c_{h'}^{m'} \\
&= 2SA(\log_2(T_{M'}) + 1) + 2C_{M'},
\end{aligned}$$

where (i) comes from Equation (D.7).

Regret contribution of the* VISGO *precision errors. For any $M' \leq M$, denote by $J_{M'}$ the (unknown) total number of triggers in the first M' intervals. For $1 \leq j \leq J_{M'}$, denote by L_j the number of time steps elapsed between the $(j-1)$-th and the j-th trigger. The doubling condition implies that $L_j \leq 2^j SA$ and that there are at most $J_{M'} = O(SA \log_2(T_{M'}/(SA)))$

triggers. Using that Algorithm 5.1 selects as error $\varepsilon_{\mathrm{VI}}^j = 2^{-j}/(SA)$, we have that

$$\begin{aligned}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}(1+c_1\sqrt{\iota_{M'}/2})\varepsilon_{\mathrm{VI}}^m &\le (1+c_1\sqrt{\iota_{M'}/2})\sum_{j=1}^{J_{M'}} L_j\varepsilon_{\mathrm{VI}}^j \\ &\le (1+c_1\sqrt{\iota_{M'}/2})J_{M'} \\ &= O\Big(SA\log_2(T_{M'})\sqrt{\iota_{M'}}\Big).\end{aligned}$$

We proceed with bounding $X_4(M')$ and $X_5(M')$ in Lemmas D.17 and D.18. In the proofs of these two lemmas, we use the following notation for simplicity: for any vector X of size S' and any integer $j \ge 1$, we denote by X^j the vector $[X(1)^j, X(2)^j, \dots, X(S')^j]^\top$.

Lemma D.17. *Conditioned on Lemma D.13, for a fixed $M' \le M$ with probability $1-2\delta$,*

$$X_4(M') \le O\left(B_\star(C_{M'} + X_2(M')) + (B_\star^2 SA + B_\star)(\log_2(T_{M'}) + \ln(2/\delta))\right).$$

Proof. We introduce the normalized value function $\overline{V}^m \triangleq V^m/B_\star \in [0,1]$. Define

$$F(d) \triangleq \sum_{m=1}^{M'}\sum_{h=1}^{H^m}(P_{s_h^m,a_h^m}(\overline{V}^m)^{2^d} - (\overline{V}^m(s_{h+1}^m))^{2^d}),\ G(d) \triangleq \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\mathbb{V}(P_{s_h^m,a_h^m},(\overline{V}^m)^{2^d}).$$

Then $X_4(M') = B_\star^2 G(0)$. Direct computation gives that

$$\begin{aligned}G(d) &= \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\left(P_{s_h^m,a_h^m}(\overline{V}^m)^{2^{d+1}} - (P_{s_h^m,a_h^m}(\overline{V}^m)^{2^d})^2\right) \\ &\overset{\text{(i)}}{\le} \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\left(P_{s_h^m,a_h^m}(\overline{V}^m)^{2^{d+1}} - (\overline{V}^m(s_{h+1}^m))^{2^{d+1}}\right) \underbrace{+\sum_{m=1}^{M'}(\overline{V}^m(s_{H^m+1}^m))^{2^{d+1}}}_{\le M_1'} \\ &\quad + \sum_{m=1}^{M'}\sum_{h=1}^{H^m}\left((\overline{V}^m(s_h^m))^{2^{d+1}} - (P_{s_h^m,a_h^m}\overline{V}^m)^{2^{d+1}}\right) \underbrace{-\sum_{m=1}^{M'}(\overline{V}^m(s_1^m))^{2^{d+1}}}_{\le 0} \\ &\overset{\text{(ii)}}{\le} F(d+1) + M_1' + 2^{d+1}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\max\{\overline{V}^m(s_h^m) - P_{s_h^m,a_h^m}\overline{V}^m,\ 0\} \\ &= F(d+1) + M_1' + \frac{2^{d+1}}{B_\star}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}\max\{Q^m(s_h^m,a_h^m) - P_{s_h^m,a_h^m}V^m,\ 0\} \\ &\overset{\text{(iii)}}{\le} F(d+1) + M_1' + \frac{2^{d+1}}{B_\star}\sum_{m=1}^{M'}\sum_{h=1}^{H^m}(c(s_h^m,a_h^m) + \beta^m(s_h^m,a_h^m))\end{aligned}$$

$$= F(d+1) + M_1' + \frac{2^{d+1}}{B_\star} \sum_{m=1}^{M'} \sum_{h=1}^{H^m} (c_h^m + \beta^m(s_h^m, a_h^m) + (c(s_h^m, a_h^m) - c_h^m))$$
$$\leq F(d+1) + M_1' + \frac{2^{d+1}}{B_\star}(C_{M'} + X_2(M') + |X_3(M')|),$$

where M_1' denotes the number of intervals satisfying $\overline{V}^m(s_{H^m+1}^m) \neq 0$; (i) is by convexity of $f(x) = x^{2^d}$; (ii) is by Lemma D.27; (iii) is by Lemma D.13.

For a fixed d, $F(d)$ is a martingale. By taking $c = 1$ in Lemma D.23, we have

$$\mathbb{P}\left[F(d) > 2\sqrt{2G(d)(\log_2(T_{M'}) + \ln(2/\delta))} + 5(\log_2(T_{M'}) + \ln(2/\delta))\right] \leq \delta.$$

Taking $\delta' = \delta/(\log_2(T_{M'}) + 1)$, using $x \geq \ln(x) + 1$ and finally swapping δ and δ', we have that

$$\mathbb{P}\left[F(d) > 2\sqrt{2G(d)(2\log_2(T_{M'}) + \ln(2/\delta))} + 5(2\log_2(T_{M'}) + \ln(2/\delta))\right] \leq \frac{\delta}{\log_2(T_{M'}) + 1}.$$

Taking a union bound over $d = 1, 2, \ldots, \log_2(T_{M'})$, we have that with probability $1 - \delta$,

$$F(d) \overset{\text{(i)}}{\leq} 2\sqrt{2(2\log_2(T_{M'}) + \ln(2/\delta))} \cdot \sqrt{F(d+1) + 2^{d+1} \cdot \frac{C_{M'} + X_2(M') + |X_3(M')|}{B_\star}}$$
$$+ 5(2\log_2(T_M) + \ln(2/\delta)) + 2\sqrt{2(2\log_2(T_{M'}) + \ln(2/\delta))M_1'}.$$

From Lemma D.25, taking $\lambda_1 = T_{M'},\ \lambda_2 = 2\sqrt{2(2\log_2(T_{M'}) + \ln(2/\delta))},\ \lambda_3 = (C_{M'} + X_2(M') + |X_3(M')|)/B_\star,\ \lambda_4 = 5(2\log_2(T_M) + \ln(2/\delta)) + 2\sqrt{2(2\log_2(T_{M'}) + \ln(2/\delta))M_1'}$, we have that

$$F(1) \leq O\left(\log_2(T_{M'}) + \ln(2/\delta) + \frac{C_{M'} + X_2(M') + |X_3(M')|}{B_\star} + M_1'\right).$$

Hence

$$X_4(M') \leq O\left(B_\star(C_{M'} + X_2(M') + |X_3(M')|) + B_\star^2(\log_2(T_{M'}) + \ln(2/\delta) + M_1')\right).$$

By definition, $M_1' \leq O(SA\log_2(T_{M'}))$ since only those intervals ending by triggering the doubling condition are taken into account. From the bound of $|X_3(M')|$, the following holds with probability $1 - 2\delta$:

$$X_4(M') \leq O\left(B_\star(C_{M'} + X_2(M')) + (B_\star^2 SA + B_\star)(\log_2(T_{M'}) + \ln(2/\delta))\right).$$

Throughout the proof, the inequality $O(\sqrt{xy}) \leq O(x + y)$ is utilized to simplify the bound. □

Lemma D.18. *Conditioned on Lemma D.13, for a fixed $M' \le M$ with probability $1-\delta$,*

$$X_5(M') \le O\left(B_\star^2 SA(\log_2(T_{M'}) + \ln(2/\delta)) + B_\star X_2(M')\right).$$

Proof. We introduce the normalized quantity $\overline{\widetilde{V}}^m \triangleq \widetilde{V}^m / B_\star \in [-1,1]$ (recall the definition in Lemma D.14). Define

$$\widetilde{F}(d) \triangleq \sum_{m=1}^{M'} \sum_{h=1}^{H^m} (P_{s_h^m,a_h^m}(\overline{\widetilde{V}}^m)^{2^d} - (\overline{\widetilde{V}}^m(s_{h+1}^m))^{2^d}),\ \widetilde{G}(d) \triangleq \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \mathbb{V}(P_{s_h^m,a_h^m}, (\overline{\widetilde{V}}^m)^{2^d}).$$

Then $X_5(M') = \widetilde{G}(0) B_\star^2$. Direct computation gives that

$$\begin{aligned}
\widetilde{G}(d) &= \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \left(P_{s_h^m,a_h^m}(\overline{\widetilde{V}}^m)^{2^{d+1}} - (P_{s_h^m,a_h^m}(\overline{\widetilde{V}}^m)^{2^d})^2\right) \\
&\le \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \left(P_{s_h^m,a_h^m}(\overline{\widetilde{V}}^m)^{2^{d+1}} - (\overline{\widetilde{V}}^m(s_{h+1}^m))^{2^{d+1}}\right) \underbrace{+ \sum_{m=1}^{M'} (\overline{\widetilde{V}}^m(s_{H^m+1}^m))^{2^{d+1}}}_{\le \widetilde{M}_1'} \\
&\quad + \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \left((\overline{\widetilde{V}}^m(s_h^m))^{2^{d+1}} - (P_{s_h^m,a_h^m}\overline{\widetilde{V}}^m)^{2^{d+1}}\right) \underbrace{- \sum_{m=1}^{M'} (\overline{\widetilde{V}}^m(s_1^m))^{2^{d+1}}}_{\le 0} \\
&\le \widetilde{F}(d+1) + \widetilde{M}_1' + 2^{d+1} \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \max\{\overline{\widetilde{V}}^m(s_h^m) - P_{s_h^m,a_h^m}\overline{\widetilde{V}}^m,\ 0\} \\
&= \widetilde{F}(d+1) + \widetilde{M}_1' + \frac{2^{d+1}}{B_\star} \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \max\{\widetilde{V}^m(s_h^m) - P_{s_h^m,a_h^m}\widetilde{V}^m,\ 0\} \\
&\overset{\text{(i)}}{\le} \widetilde{F}(d+1) + \widetilde{M}_1' + \frac{2^{d+1}}{B_\star} \sum_{m=1}^{M'} \sum_{h=1}^{H^m} \beta^m(s_h^m, a_h^m) \\
&= \widetilde{F}(d+1) + \widetilde{M}_1' + \frac{2^{d+1}}{B_\star} X_2(M'),
\end{aligned}$$

where $\widetilde{M}_1'$ denotes the number of intervals satisfying $\overline{\widetilde{V}}^m(s_{H^m+1}^m) \ne 0$; (i) come from Lemma D.14.

For a fixed d, $\widetilde{F}(d)$ is a martingale. By taking $c=1$ in Lemma D.23, we have

$$\mathbb{P}\left[\widetilde{F}(d) > 2\sqrt{2\widetilde{G}(d)(\log_2(T_{M'}) + \ln(2/\delta))} + 5(\log_2(T_{M'}) + \ln(2/\delta))\right] \le \delta.$$

Taking $\delta' = \delta/(\log_2(T_{M'})+1)$, using $x \geq \ln(x)+1$ and finally swapping δ and δ', we have that

$$\mathbb{P}\left[\widetilde{F}(d) > 2\sqrt{2\widetilde{G}(d)(2\log_2(T_{M'})+\ln(2/\delta))} + 5(2\log_2(T_{M'})+\ln(2/\delta))\right] \leq \frac{\delta}{\log_2(T_{M'})+1}.$$

Taking a union bound over $d = 1, 2, \ldots, \log_2(T_{M'})$, we have that with probability $1-\delta$,

$$\begin{aligned}\widetilde{F}(d) \leq & 2\sqrt{2(2\log_2(T_{M'})+\ln(2/\delta))} \cdot \sqrt{\widetilde{F}(d+1) + 2^{d+1}\frac{X_2(M')}{B_\star}} \\ & + 5(2\log_2(T_{M'})+\ln(2/\delta)) + 2\sqrt{2(2\log_2(T_{M'})+\ln(2/\delta))\widetilde{M}_1'}.\end{aligned}$$

From Lemma D.25, taking $\lambda_1 = T_{M'}$, $\lambda_2 = 2\sqrt{2(2\log_2(T_{M'})+\ln(2/\delta))}$, $\lambda_3 = X_2(M')/B_\star$, $\lambda_4 = 5(2\log_2(T_{M'})+\ln(2/\delta)) + 2\sqrt{2(2\log_2(T_{M'})+\ln(2/\delta))\widetilde{M}_1'}$, we have that

$$\widetilde{F}(1) \leq O\left(\log_2(T_{M'}) + \ln(2/\delta) + \frac{X_2(M')}{B_\star} + \widetilde{M}_1'\right).$$

Since $V^\star(g) - V^m(g) = 0 - 0 = 0$, similar as bounding M_1', we have $\widetilde{M}_1' \leq O(SA\log_2(T_{M'}))$. Hence with probability $1-\delta$, we have

$$X_5(M') \leq O\left(B_\star^2 SA(\log_2(T_{M'})+\ln(2/\delta)) + B_\star X_2(M')\right).$$

Throughout the proof, the inequality $O(\sqrt{xy}) \leq O(x+y)$ is utilized to simplify the bound. □

④ Finally, bind them together.

Let $\bar{\iota}_{M'} \triangleq \ln\left(\frac{12SAS'T_{M'}^2}{\delta}\right) + \log_2((\max\{B_\star,1\})^2 T_{M'}) + \ln\left(\frac{2}{\delta}\right)$ be the upper bound of all previous log terms.

$$\begin{aligned}X_2(M') &\leq O\Big(\sqrt{SAX_4(M')\bar{\iota}_{M'}} + \sqrt{S^2AX_5(M')\bar{\iota}_{M'}} \\ &\qquad + SA\bar{\iota}_{M'}^{3/2} + \sqrt{SAC_{M'}\bar{\iota}_{M'}} + B_\star S^2 A\bar{\iota}_{M'}^2 + BS^{3/2}A\bar{\iota}_{M'}^2\Big), \\ X_4(M') &\leq O\left(B_\star(C_{M'} + X_2(M')) + (B_\star^2 SA + B_\star)\bar{\iota}_{M'}\right), \\ X_5(M') &\leq O\left(B_\star^2 SA\bar{\iota}_{M'} + B_\star X_2(M')\right).\end{aligned}$$

This implies that

$$\begin{aligned}X_2(M') &\overset{\text{(i)}}{\leq} O\left(\sqrt{B_\star S^2 A\bar{\iota}_{M'}}\cdot\sqrt{X_2(M')} + (\sqrt{B_\star}+1)\sqrt{SAC_{M'}\bar{\iota}_{M'}} + BS^2A\bar{\iota}_{M'}^2\right) \\ &\leq O\left(\max\left\{\sqrt{B_\star S^2 A\bar{\iota}_{M'}}\cdot\sqrt{X_2(M')},\ (\sqrt{B_\star}+1)\sqrt{SAC_{M'}\bar{\iota}_{M'}} + BS^2A\bar{\iota}_{M'}^2\right\}\right),\end{aligned}$$

where (i) uses the assumption $B \geq \max\{B_\star, 1\}$ to simplify the bound. Considering terms in $\max\{\}$ separately, we obtain two bounds:

$$X_2(M') \leq O(B_\star S^2 A \bar{\iota}_{M'}^2),$$
$$X_2(M') \leq O((\sqrt{B_\star}+1)\sqrt{SAC_{M'}}\bar{\iota}_{M'} + BS^2 A\bar{\iota}_{M'}^2).$$

By taking the maximum of these bounds, we have

$$X_2(M') \leq O((\sqrt{B_\star}+1)\sqrt{SAC_{M'}}\bar{\iota}_{M'} + BS^2 A\bar{\iota}_{M'}^2).$$

D.5 Technical Lemmas

Lemma D.19 (Bennett's Inequality, anytime version)**.** *Let $Z, Z_1, \ldots, Z_n$ be i.i.d. random variables with values in $[0, b]$ and let $\delta > 0$. Define $\mathbb{V}[Z] = \mathbb{E}[(Z - \mathbb{E}[Z])^2]$. Then we have*

$$\mathbb{P}\left[\forall n \geq 1,\ \left|\mathbb{E}[Z] - \frac{1}{n}\sum_{i=1}^n Z_i\right| > \sqrt{\frac{2\mathbb{V}[Z]\ln(4n^2/\delta)}{n}} + \frac{b\ln(4n^2/\delta)}{n}\right] \leq \delta.$$

Proof. From Bennett's inequality, if the variables have values in $[0, 1]$, then for a specific $n \geq 1$,

$$\mathbb{P}\left[\left|\mathbb{E}[Z] - \frac{1}{n}\sum_{i=1}^n Z_i\right| > \sqrt{\frac{2\mathbb{V}[Z]\ln(2/\delta)}{n}} + \frac{\ln(2/\delta)}{n}\right] \leq \delta.$$

We then choose $\delta \leftarrow \frac{\delta}{2n^2}$ and take a union bound over all possible values of $n \geq 1$, and the result follows given that $\sum_{n\geq 1}\frac{\delta}{2n^2} < \delta$. To account for the case $b \neq 1$ we apply the result to (Z_n/b). □

Lemma D.20 (Theorem 4 in Maurer and Pontil, 2009, anytime version)**.** *Let $Z, Z_1, \ldots, Z_n$ $(n \geq 2)$ be i.i.d. random variables with values in $[0, b]$ and let $\delta > 0$. Define $\bar{Z} = \frac{1}{n}Z_i$ and $\hat{V}_n = \frac{1}{n}\sum_{i=1}^n (Z_i - \bar{Z})^2$. Then we have*

$$\mathbb{P}\left[\forall n \geq 1,\ \left|\mathbb{E}[Z] - \frac{1}{n}\sum_{i=1}^n Z_i\right| > \sqrt{\frac{2\hat{V}_n\ln(4n^2/\delta)}{n-1}} + \frac{7b\ln(4n^2/\delta)}{3(n-1)}\right] \leq \delta.$$

Lemma D.21 (Popoviciu's Inequality). *Let X be a random variable whose value is in a fixed interval $[a, b]$, then $\mathbb{V}[X] \le \frac{1}{4}(b-a)^2$.*

Lemma D.22 (Lemma 11 in Zhang et al., 2021f). *Let $(M_n)_{n\ge 0}$ be a martingale such that $M_0 = 0$ and $|M_n - M_{n-1}| \le c$ for some $c > 0$ and any $n \ge 1$. Let $\mathrm{Var}_n = \sum_{k=1}^{n} \mathbb{E}[(M_k - M_{k-1})^2 | \mathcal{F}_{k-1}]$ for $n \ge 0$, where $\mathcal{F}_k = \sigma(M_1, \ldots, M_k)$. Then for any positive integer n and any $\varepsilon, \delta > 0$, we have that*

$$\mathbb{P}\left[|M_n| \ge 2\sqrt{2\mathrm{Var}_n \ln(1/\delta)} + 2\sqrt{\varepsilon \ln(1/\delta)} + 2c\ln(1/\delta)\right] \le 2\left(\log_2\left(\frac{nc^2}{\varepsilon}\right) + 1\right)\delta.$$

Lemma D.23. *Let $(M_n)_{n\ge 0}$ be a martingale such that $M_0 = 0$ and $|M_n - M_{n-1}| \le c$ for some $c > 0$ and any $n \ge 1$. Let $\mathrm{Var}_n = \sum_{k=1}^{n} \mathbb{E}[(M_k - M_{k-1})^2 | \mathcal{F}_{k-1}]$ for $n \ge 0$, where $\mathcal{F}_k = \sigma(M_1, \ldots, M_k)$. Then for any positive integer n and $\delta \in (0, 2(nc^2)^{1/\ln 2}]$, we have that*

$$\mathbb{P}\left[|M_n| \ge 2\sqrt{2\mathrm{Var}_n(\log_2(nc^2) + \ln(2/\delta))} + 2\sqrt{\log_2(nc^2) + \ln(2/\delta)} + 2c(\log_2(nc^2) + \ln(2/\delta))\right] \le \delta.$$

Proof. Take $\varepsilon = 1$ and $\delta' = 2(\log_2(nc^2) + 1)\delta$ in Lemma D.22. By $x \ge \ln(x) + 1$, we have

$$\ln(1/\delta) = \ln(2(\log_2(nc^2) + 1)/\delta') = \ln(\log_2(nc^2) + 1) + \ln(2/\delta') \le \log_2(nc^2) + \ln(2/\delta').$$

Hence,

$$\begin{aligned}
&\mathbb{P}\left[|M_n| \ge 2\sqrt{2\mathrm{Var}_n(\log_2(nc^2) + \ln(2/\delta'))} + 2\sqrt{\log_2(nc^2) + \ln(2/\delta')} + 2c(\log_2(nc^2) + \ln(2/\delta'))\right] \\
&\le \mathbb{P}\left[|M_n| \ge 2\sqrt{2\mathrm{Var}_n \ln(1/\delta)} + 2\sqrt{\ln(1/\delta)} + 2c\ln(1/\delta)\right] \\
&\le \delta'.
\end{aligned}$$

By swapping δ and δ' we complete the proof. □

Lemma D.24 (Lemma 11 in Zhang et al., 2021d). *Let $\lambda_1, \lambda_2, \lambda_4 \ge 0$, $\lambda_3 \ge 1$ and $i' = \log_2 \lambda_1$. Let $a_1, a_2, \ldots, a_{i'}$ be non-negative reals such that $a_i \le \lambda_1$ and $a_i \le \lambda_2\sqrt{a_{i+1} + 2^{i+1}\lambda_3} + \lambda_4$ for any $1 \le i \le i'$. Then we have that $a_1 \le \max\{(\lambda_2 + \sqrt{\lambda_2^2 + \lambda_4})^2, \lambda_2\sqrt{8\lambda_3} + \lambda_4\}$.*

Lemma D.25. *Let $\lambda_1, \lambda_2, \lambda_4 \geq 0,\ \lambda_3 \geq 1$ and $i' = \log_2 \lambda_1$. Let $a_1, a_2, \ldots, a_{i'}$ be non-negative reals such that $a_i \leq \lambda_1$ and $a_i \leq \lambda_2 \sqrt{a_{i+1} + 2^{i+1}\lambda_3} + \lambda_4$ for any $1 \leq i \leq i'$. Then we have that $a_1 \leq O(\lambda_2^2 + \lambda_3 + \lambda_4)$.*

Proof. Since $\max\{a,\ b\} \leq a + b$ and $2ab \leq a^2 + b^2$ for any choice of non-negative a and b, we can transform the result of Lemma D.24 into

$$\begin{aligned} a_1 &\leq \max\left\{\left(\lambda_2 + \sqrt{\lambda_2^2 + \lambda_4}\right)^2, \lambda_2\sqrt{8\lambda_3} + \lambda_4\right\} \\ &\leq O\left(\left(\lambda_2 + \sqrt{\lambda_2^2 + \lambda_4}\right)^2 + \lambda_2\sqrt{8\lambda_3} + \lambda_4\right) \\ &\leq O(\lambda_2^2 + \lambda_2^2 + \lambda_4 + \lambda_2^2 + \lambda_3 + \lambda_4) \\ &\leq O(\lambda_2^2 + \lambda_3 + \lambda_4). \end{aligned}$$

□

Lemma D.26. *For random variable $Z \in [0, 1],\ \mathbb{V}[Z] \leq \mathbb{E}[Z]$.*

Proof. $\mathbb{V}[Z] = \mathbb{E}[Z^2] - (\mathbb{E}[Z])^2 \leq \mathbb{E}[Z^2] \leq \mathbb{E}[Z]$. □

Lemma D.27. *For any $a, b \in [0, 1]$ and $k \in \mathbb{N},\ a^k - b^k \leq k \max\{a - b,\ 0\}$.*

Proof. $a^k - b^k = (a - b)\sum_{i=0}^{k-1} a^i b^{k-1-i} \leq \max\{a - b,\ 0\} \cdot \sum_{i=0}^{k-1} 1 = k \max\{a - b,\ 0\}$. □

Lemma D.28. *For $a, b, x \geq 0,\ x \leq a\sqrt{x} + b$ implies $x \leq (a + \sqrt{b})^2$.*

Proof. $x \leq a\sqrt{x} + b \Rightarrow x \leq \left(\frac{a+\sqrt{a^2+b}}{2}\right)^2 \leq (a + \sqrt{b})^2$. □

D.6 Computational Complexity of EB-SSP

Here we complement Remark D.12 on the computational complexity of EB-SSP (Algorithm 5.1).

The computational complexity of a VISGO procedure can be bounded as $O(\frac{S^2 A}{1-\rho} \log(B_\star/\varepsilon_{\text{VI}}))$ (assuming for simplicity that $B_\star \geq 1$, otherwise replace $\max\{B_\star, 1\} \leftarrow B_\star$). By the fact that total number of VISGO procedure is bounded by $O(SA \log T)$, we derive $\log(B_\star/\varepsilon_{\text{VI}}) =$

$O(SA\log(B_\star T))$ by choice of ε_{VI}. As a result, the total computational complexity for EB-SSP is $O(TS^2A \cdot SA\log(B_\star T)\cdot SA\log T)$, which is polynomially bounded and in particular near-linear in T. Also note that T is bounded polynomially w.r.t. K as shown in the various cases of Section 5.4. Indeed, in the case of positive costs lower bounded by $c_{\min} > 0$, Corollary 5.3 entails that $T \le c_{\min}^{-1}KV^\star(s_0) + c_{\min}^{-1}\widetilde{O}(B_\star\sqrt{SAK} + B_\star S^2A)$. In the general cost case, the cost perturbation trick is applied and the minimum cost becomes K^{-n} for Corollary 5.4 or $(\overline{T}_\star K)^{-1}$ for Corollary 5.6, i.e., $c_{\min}^{-1}$ depends polynomially on K.

We note that the analysis of the computational complexity of EB-SSP may likely be refined. Indeed, we see that i) on the one hand, if $n(s,a)$ is small, then the optimistic skewing of $\widetilde{P}_{s,a}$ is not too small so the probability of reaching the goal from (s,a) is not too small (so the associated contraction modulus is bounded away from 1) and ii) on the other hand, if $n(s,a)\to+\infty$, then $\widetilde{P}_{s,a}\to\widehat{P}_{s,a}\to P_{s,a}$, so to the limit we should recover the convergence properties of VI of the optimal Bellman operator under the true model, which by assumption admits a proper policy in P. Thus we see that studying further the "intermediate regime" may bring into the picture the computational complexity of running VI in the true model, yet this is not our main focus here, as our complexity analysis is sufficient to ensure the computational efficiency of EB-SSP.

D.7 Unknown $B_\star$: Parameter-Free EB-SSP

In this section, we relax the assumption that (an upper bound of) $B_\star$ is known to EB-SSP. In Algorithm D.1 we propose a parameter-free EB-SSP that bypasses the requirement $B \ge B_\star$ (line 2 of Algorithm 5.1) to tune the exploration bonus. As in Section 5.3 we consider for ease of exposition that $B_\star \ge 1$. We structure the section as follows: Section D.7.1 presents our algorithm and provides intuition, Section D.7.2 spells out its regret guarantee, and Section D.7.3 gives its proof.

D.7.1 Algorithm and Intuition

Parameter-free EB-SSP (Algorithm D.1) initializes an estimate $\widetilde{B} = 1$ and decomposes the time steps into *phases*, indexed by ϕ. The execution of a phase is reported in the subroutine PHASE (Algorithm D.2). Given any estimate $\widetilde{B}$, a subroutine PHASE has the same structure as Algorithm 5.1, up to two key differences:

- ***Halting due to exceeding cumulative cost.*** PHASE tracks the cumulative cost within the current phase, and terminates whenever it exceeds a threshold C_{bound} that depends on $\widetilde{B}$, S, A, δ and the current episode and time indexes k and t, which are all computable quantities to the agent, see Equation (D.10).

- ***Halting due to exceeding* VISGO *range.*** During each VISGO procedure, PHASE tracks the range of the value function $V^{(i)}$ at each VISGO iteration i, and terminates if $\|V^{(i)}\|_\infty > \widetilde{B}$.

The estimate $\widetilde{B}$ can be incremented in two different ways and speeds:

- ***Doubling increment of* $\widetilde{B}$.** On the one hand, whenever a phase ends (i.e., one of the two halting conditions above is met), $\widetilde{B}$ is doubled ($\widetilde{B} \leftarrow 2\widetilde{B}$).
- ***Episode-driven increment of* $\widetilde{B}$.** On the other hand, at the beginning of each new episode k, the estimate is automatically increased to $\widetilde{B} \leftarrow \max\{\widetilde{B}, \sqrt{k}/(S^{3/2}A^{1/2})\}$.

We now explain the rationale behind our scheme:

- *Reason for episode-driven increment of* $\widetilde{B}$. The fact that $\widetilde{B}$ grows as a function of k implies that at some (unknown) point it will hold that $\widetilde{B} \geq B_\star$ for large enough k. This will enable us to recover the analysis and the regret bound of Theorem 5.1.
- *Reason for doubling increment of* $\widetilde{B}$. The doubling increment comes into play whenever a phase terminates due to an exceeding cumulative cost or VISGO range. At this point, the agent becomes aware that $\widetilde{B}$ is too small and thus it doubles it. It is crucial to allow intra-episode increments of $\widetilde{B}$ to avoid getting *stuck* in an episode with an underestimate $\widetilde{B} < B_\star$.
- *Reason for cumulative cost halting.* The cost threshold C_{bound} is designed so that (w.h.p.) it can be exceeded at most once in the case of $\widetilde{B} \geq B_\star$, and so that it can serve as a tight enough bound on the regret in the case of $\widetilde{B} < B_\star$.
- *Reason for* VISGO *range halting.* The threshold $\widetilde{B}$ on the range of the VISGO value functions is chosen so that (w.h.p.) it is never exceeded in the case of $\widetilde{B} \geq B_\star$, and so that it can serve as a guarantee of finite-time near-convergence of a VISGO procedure (i.e., the contraction property) in the case of $\widetilde{B} < B_\star$.

D.7.2 Regret Guarantee of Parameter-Free EB-SSP

Parameter-free EB-SSP satisfies the following guarantee (which extends Theorem 5.1 to unknown $B_\star$).

Restatement of Theorem 5.7. Assume the conditions of Proposition 2.11 hold. Then with probability at least $1-\delta$ the regret of parameter-free EB-SSP (Algorithm D.1, Section D.7) can be bounded by

$$R_K = O\left(R_K^\star \log\left(\frac{B_\star SAT}{\delta}\right) + B_\star^3 S^3 A \log^3\left(\frac{B_\star SAT}{\delta}\right)\right),$$

where T is the cumulative time within the K episodes and $R_K^\star$ bounds the regret after K episodes of EB-SSP in the case of known $B_\star$ (i.e., the bound of Theorem 5.1 with $B = B_\star$).

As a result, parameter-free EB-SSP is able to circumvent the knowledge of $B_\star$ at the cost of only logarithmic and lower-order terms.

D.7.3 Proof of Theorem 5.7

We begin by defining notations and concepts exclusively used in this section:

- C_t denotes the cumulative cost up to time step t (included) that is accumulated in the execution of the subroutine PHASE in which time step t belongs. Importantly, note that the cumulative cost C_t is initialized to 0 at the beginning of each PHASE (line 5 of Algorithm D.2). Also note that re-planning (i.e., a VISGO procedure) occurs whenever the estimate $\widetilde{B}$ is changed.
- Denote by t_m the time step at the end of the current interval m, and by k_m the episode in which the time step t_m belongs. $\widetilde{B}_m$ denotes the value of $\widetilde{B}$ at time step t_m. C_m denotes C_{t_m}, i.e., the cumulative cost up to interval m (included) in the execution of the PHASE in which interval m belongs.

Unlike EB-SSP of Algorithm 5.1, the parameter-free version has an increasing $\widetilde{B}$ throughout the process. To utilize the regret bounds (Theorem 5.1 and Equation (D.6)) in the case of $\widetilde{B} \geq B_\star$, slight modifications are needed to be applied to the algorithm and some lemmas.

***Modification to* EB-SSP.** Previously, EB-SSP accepted a single value $B \geq \max\{B_\star,\ 1\}$ to compute the bonuses in Equation (5.1). To satisfy the same regret bound when $\widetilde{B}$ changes, we require EB-SSP to accept a series of B_k for $k \in \mathbb{N}^+$, such that $\max\{B_\star,\ 1\} \leq B_k \leq B$ for any k. In any episode k, the analysis simply substitutes B_k for B in Equation (5.1).

Modifications to the proofs of Lemmas D.9, D.10 and D.13. In the original version of the proofs, we proved the lemmas for any update of value functions, without mentioning any time relevant variables. Now since B relies on episode k, the modified proofs need to incorporate the changes. Suppose that we are examining $Q(s,a),\ V(s),\ b(s,a)$ and $\beta(s,a)$ for any state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$ in episode k. Lemma D.9 and Lemma D.10 utilize the property stated in Lemma D.8, and the B in Lemma D.8 is a parameter that is able to vary each time step we utilize Lemma D.8. Thus, in the proofs of Lemma D.9, D.10 and D.13, all the B's are substituted with B_k's to ensure that these lemmas are compatible with our modified setting.

Modification to the proof of bounding β^m in Section D.4.3. Suppose that interval m is in episode k and recall that $B_k \leq B$, then

$$
\begin{aligned}
b^m(s,a) &= \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(l)})\iota_{s,a}}{n^m(s,a)}},\ c_2\frac{B_k\iota_{s,a}}{n^m(s,a)}\right\} + c_3\sqrt{\frac{\widehat{c}^m(s,a)\iota_{s,a}}{n^m(s,a)}} + c_4\frac{B_k\sqrt{S'\iota_{s,a}}}{n^m(s,a)} \\
&\leq O\left(\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(l)})\iota_{s,a}}{n^m(s,a)}} + \frac{B\iota_{s,a}}{n^m(s,a)} + \sqrt{\frac{\widehat{c}^m(s,a)\iota_{s,a}}{n^m(s,a)}} + \frac{B\sqrt{S\iota_{s,a}}}{n^m(s,a)}\right).
\end{aligned}
$$

Combining the above bound of $b^m(s,a)$ with Lemma D.13, we get that the bound of β^m in Section D.4.3 is unchanged.

Equipped with the slight modifications mentioned above, we now derive two key properties on which the analysis of parameter-free EB-SSP relies:

Property 1: Optimism avoids the first halting condition. Let us study any phase starting with estimate $\widetilde{B} \geq B_\star$. From Equation (D.6) (which is the interval-generalization of Theorem 5.1), for a fixed initial state s_0 and a fixed interval m, the cumulative cost can be bounded with probability $1-\delta$ by

$$k_m V^\star(s_0) + x\left(B_\star\sqrt{SAk_m}\log_2\left(\frac{B_\star t_m SA}{\delta}\right) + \widetilde{B}_m S^2 A \log_2^2\left(\frac{B_\star t_m SA}{\delta}\right)\right), \tag{D.8}$$

where $x > 0$ is a large enough absolute constant (which can be retraced in the analysis leading to Equation (D.6)). By scaling $\delta \leftarrow \delta/(2St_m^2)$ for each $m \leq M$, we have the following cumulative cost bound that holds for any initial state in $\mathcal{S}$ and any interval $m \leq M$, with probability $1-\delta$,

$$\begin{aligned} C_m &\leq k_m V^\star(s_0) + x\left(B_\star\sqrt{SAk_m}\log_2\left(\frac{B_\star t_m SA\cdot 2St_m^2}{\delta}\right) + \widetilde{B}_m S^2 A \log_2^2\left(\frac{B_\star t_m SA\cdot 2St_m^2}{\delta}\right)\right) \\ &\leq k_m B_\star + 3x\left(B_\star\sqrt{SAk_m}\log_2\left(\frac{B_\star t_m SA}{\delta}\right) + \widetilde{B}_m S^2 A \log_2^2\left(\frac{B_\star t_m SA}{\delta}\right)\right). \end{aligned}$$

Since we are in the case of $\widetilde{B}_m \geq B_\star$, we have

$$C_m \leq k_m\widetilde{B}_m + 3x\left(\widetilde{B}_m\sqrt{SAk_m}\log_2\left(\frac{\widetilde{B}_m t_m SA}{\delta}\right) + \widetilde{B}_m S^2 A \log_2^2\left(\frac{\widetilde{B}_m t_m SA}{\delta}\right)\right). \tag{D.9}$$

Since costs are non-negative, for any $t \leq t_m$, we have $C_t \leq C_m$ hence C_t must also satisfy the bound of Equation (D.9). There remains to predict the values of $k_m,\ t_m,\ \widetilde{B}_m$, given the current $k_{\text{cur}},\ t_{\text{cur}},\ \widetilde{B}_{\text{cur}}$. The upper bounds for k_m and $\widetilde{B}_m$ are k_{cur} and $\widetilde{B}_{\text{cur}}$ respectively, since they can only be incremented when reaching the goal g, which is a condition for ending the current interval. The upper bound for t_m can be derived using the pigeonhole principle: since $t_{\text{cur}} = \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} n(s,a)$, we know that $2t_{\text{cur}} > \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}(2n(s,a)-1)$. Thus by time step $2t_{\text{cur}}$ there must exist a trigger condition, which is a condition for ending the current interval. Hence, by replacing $k_m \leftarrow k_{\text{cur}}$, $\widetilde{B}_m \leftarrow \widetilde{B}_{\text{cur}}$ and $t_m \leftarrow 2t_{\text{cur}}$ in Equation (D.9), we get, with probability at least $1-\delta$, that the cumulative cost within a phase that starts with $\widetilde{B} \geq B_\star$ has the following anytime upper bound

$$C_{t_{\text{cur}}} \leq k_{\text{cur}}\widetilde{B}_{\text{cur}} + 3x\left(\widetilde{B}_{\text{cur}}\sqrt{SAk_{\text{cur}}}\log_2\left(\frac{2\widetilde{B}_{\text{cur}} t_{\text{cur}} SA}{\delta}\right) + \widetilde{B}_{\text{cur}} S^2 A \log_2^2\left(\frac{2\widetilde{B}_{\text{cur}} t_{\text{cur}} SA}{\delta}\right)\right).$$

Note that this bound corresponds exactly to the cumulative cost threshold C_{bound} in Equation (D.10). This means that with probability at least $1-\delta$, the first halting condition cannot be met in a phase that starts with $\widetilde{B} \geq B_\star$.

Property 2: Optimism avoids the second halting condition. Let us consider the case of $\widetilde{B} \geq B_\star$ whenever the algorithm re-plans (i.e., running VISGO procedure). The proof of Lemma D.9 ensures that at any iteration, $\|V^{(i)}\|_\infty \leq B_\star \leq \widetilde{B}$, so the second halting condition is never met under the same high-probability event as above.

Implications. The two properties above indicate that, if a phase starts with estimate $\widetilde{B} \geq B_\star$, with probability at least $1-\delta$, this phase will never halt due to the two halting conditions (it can only terminate if it completes the final episode K), and Algorithm D.1 will thus never enter a new phase. Due to the doubling increment of $\widetilde{B}$ every time a phase ends, we can therefore bound the total number of phases as $\Phi \leq \lceil \log_2(B_\star) \rceil + 1$.

Analysis. We now split the analysis of the regret contributions of the episodes in two *regimes*. To this end, let $\kappa_\star \triangleq \lceil B_\star^2 S^3 A \rceil$ denote a special episode (note that it is unknown to the learner since it depends on $B_\star$). We consider that the high-probability event mentioned above holds (which is the case with probability at least $1-\delta$). Recall that at the beginning of each episode k, the algorithm sets $\widetilde{B} \leftarrow \max\{\widetilde{B},\ \sqrt{k}/(S^{3/2}A^{1/2})\}$.

① Regret contribution in the first regime (i.e., episodes $k < \kappa_\star$).

We denote respectively by $R_{1\to\kappa_\star}$ and $C_{1\to\kappa_\star}$ the cumulative regret and the cumulative cost incurred by the algorithm before episode $\kappa_\star$ begins. For any phase ϕ, we denote by

- $C_{1\to\kappa_\star}^{(\phi)}$ the cumulative cost incurred during the time steps that are *both* in phase ϕ and in an episode $k < \kappa_\star$;
- $k^{(\phi)}$ the episode when phase ϕ ends;
- $t^{(\phi)}$ the time step when phase ϕ ends;
- $\widetilde{B}^{(\phi)}$ the value of $\widetilde{B}$ at the end of phase ϕ.

Observe that

$$C_{1\to\kappa_\star} = \sum_{\phi=1}^{\Phi} C_{1\to\kappa_\star}^{(\phi)}.$$

Now, by definition of $\kappa^\star$, the episode-driven increment of $\widetilde{B}$ never exceeds $B_\star$, unless $\widetilde{B}$ is already larger or equal to $B_\star$ at the beginning of the phase. But Property 1 ensures that if $\widetilde{B} \geq B_\star$ in the beginning of a phase, then $\widetilde{B}$ will never be doubled afterwards. Hence, we are guaranteed that within the episodes $k < \kappa_\star$, the final value of the estimate $\widetilde{B}$ is at most $2B_\star$.

Since PHASE tracks the cumulative cost at each step using the threshold in Equation (D.10) and since $c_t \leq 1$, by the fact that C_{bound} is monotonously increasing with respect to t, we have that for any phase ϕ,

$$C_{1\to\kappa_\star}^{(\phi)} \leq k^{(\phi)}\widetilde{B}^{(\phi)} + 3x\left(\widetilde{B}^{(\phi)}\sqrt{SAk^{(\phi)}}\log_2\left(\frac{2\widetilde{B}^{(\phi)}t^{(\phi)}SA}{\delta}\right) + \widetilde{B}^{(\phi)}S^2A\log_2^2\left(\frac{2\widetilde{B}^{(\phi)}t^{(\phi)}SA}{\delta}\right)\right) + 1$$

$$\leq \kappa_\star(2B_\star) + 3x\left((2B_\star)\sqrt{SA\kappa_\star}\log_2\left(\frac{2(2B_\star)TSA}{\delta}\right) + (2B_\star)S^2A\log_2^2\left(\frac{2(2B_\star)TSA}{\delta}\right)\right) + 1$$
$$\leq O\left(B_\star^3S^3A + B_\star^2S^2A\log\left(\frac{B_\star TSA}{\delta}\right) + B_\star S^2A\log^2\left(\frac{B_\star TSA}{\delta}\right)\right).$$

In addition, we recall that $\Phi \leq \lceil \log_2(B_\star)\rceil + 1$. Hence, by plugging in the definition of $\kappa^\star$, we can bound the cost (and thus the regret) accumulated over the episodes $k < \kappa_\star$ as follows

$$\begin{aligned} R_{1\to\kappa_\star} \leq C_{1\to\kappa_\star} &\leq \sum_{\phi=1}^{\lceil\log_2(B_\star)\rceil+1} O\left(B_\star^3S^3A + B_\star^2S^2A\log\left(\frac{B_\star TSA}{\delta}\right) + B_\star S^2A\log^2\left(\frac{B_\star TSA}{\delta}\right)\right) \\ &\leq O\Big(B_\star^3S^3A\log(B_\star) + B_\star^2S^2A\log\left(\frac{B_\star TSA}{\delta}\right)\log(B_\star) \\ &\qquad + B_\star S^2A\log^2\left(\frac{B_\star TSA}{\delta}\right)\log(B_\star)\Big) \\ &\leq O\left(B_\star^3S^3A\bar{\iota} + B_\star^2S^2A\bar{\iota}^2 + B_\star S^2A\bar{\iota}^3\right). \end{aligned}$$

② Regret contribution in the second regime (i.e., episodes $k \geq \kappa_\star$).

We denote respectively by $R_{\kappa_\star\to K}$ and $C_{\kappa_\star\to K}$ the cumulative regret and the cumulative cost incurred during the episodes $k \geq \kappa^\star$. By definition of $\kappa^\star$, the episode-driven increment of $\widetilde{B}$ ensures that $\widetilde{B} \geq B_\star$. During this second regime there may be at most two phases: one that started at an episode $k < \kappa_\star$ (i.e., in the first regime) and that overlaps the two regimes, and one starting after that (note that properties 1 and 2 ensure that at this point neither halting condition can end this phase since it started with estimate $\widetilde{B} \geq B_\star$, thus it lasts until the end of the learning interaction). In addition, we can upper bound $\widetilde{B}$ as follows

$$\widetilde{B} \leq \max\left\{2B_\star,\ \frac{2\sqrt{K}}{S^{3/2}A^{1/2}}\right\}.$$

We now introduce a fourth condition of stopping an interval to the analysis performed in Section D.3.3: (4) an interval ends when a subroutine PHASE ends. This implies that the policy always stays the same within an interval when running Algorithm D.1. Condition (4) is met at most once in the second regime.

We now focus on only the second regime: we re-index intervals by $1, 2, \ldots, M'$ and let T_m denote the time step counting from the beginning of $\kappa_\star$ to the end of interval m. To bound $R_{\kappa_\star\to K}$, we need to adapt the proofs in Section D.3.5 and Section D.4.3 to be compatible with our new interval decomposition. Concretely, there are two slight modifications in the analysis of the second regime:

- Statistics: For any statistic (i.e., $N(s,a,s')$, $\theta(s,a)$ and $\widehat{c}(s,a)$ for any $(s,a,s') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}'$), instead of learning from scratch, PHASE reuses all samples collected thus far. This difference

does not affect the regret bound and the probability, since it can be viewed by taking a partial sum of terms in $\widetilde{R}_{M'}$.

- The regret decomposition: In the proof of Lemma D.15, we need to incorporate condition (4) which is met at most once during the second regime. It falls into case (ii) in the proof of Lemma D.15, which thus happens at most $2SA\log_2(T_{M'})+1$ times, and the regret decomposition should be

$$\widetilde{R}_{M'} \le X_1(M') + X_2(M') + X_3(M') + 2B_\star SA\log_2(T_{M'}) + B_\star.$$

Hence by incorporating these slight modifications in the proof of Theorem 5.1, we get probability at least $1-\delta$,

$$\begin{aligned} R_{\kappa_\star \to K} &\le O\left(B_\star\sqrt{SAK}\log\left(\frac{B_\star TSA}{\delta}\right) + S^2A\widetilde{B}_{M'}\log^2\left(\frac{B_\star TSA}{\delta}\right)\right) \\ &\le O\left(B_\star\sqrt{SAK}\log\left(\frac{B_\star TSA}{\delta}\right) + S^2A\frac{\sqrt{K}}{S^{3/2}A^{1/2}}\log^2\left(\frac{B_\star TSA}{\delta}\right)\right) \\ &\le O\left(B_\star\sqrt{SAK}\bar{\iota} + \sqrt{SAK}\bar{\iota}^2\right). \end{aligned}$$

③ Combining the regret contributions in the two regimes.

The overall regret is bounded with probability at least $1-\delta$ by

$$R_K = R_{1\to\kappa_\star} + R_{\kappa_\star\to K} \le O\left(B_\star\sqrt{SAK}\bar{\iota} + \sqrt{SAK}\bar{\iota}^2 + B_\star^3S^3A\bar{\iota} + B_\star^2S^2A\bar{\iota}^2 + B_\star S^2A\bar{\iota}^3\right).$$

There remains to plug in the definition of $\bar{\iota}$. Denote by T the cumulative time within the K episodes and by $R_K^\star$ the regret after K episodes of EB-SSP in the case of known $B_\star$ (i.e., the bound of Theorem 5.1 with $B=B_\star$). Then with probability at least $1-\delta$ the regret of parameter-free EB-SSP can be bounded as

$$\begin{aligned} R_K &= O\left(R_K^\star + \sqrt{SAK}\log^2\left(\frac{B_\star SAT}{\delta}\right) + B_\star^3S^3A\log^3\left(\frac{B_\star SAT}{\delta}\right)\right) \\ &= O\left(R_K^\star\log\left(\frac{B_\star SAT}{\delta}\right) + B_\star^3S^3A\log^3\left(\frac{B_\star SAT}{\delta}\right)\right). \end{aligned}$$

This concludes the proof of Theorem 5.7.

Remark D.29. At a high level, our analysis to circumvent the knowledge of $B_\star$ boils down to the following argument: if the estimate is too small, we bound the regret by the cumulative cost; otherwise if it is large enough, we recover the regret bound under a known upper bound on $B_\star$. Interestingly, this somewhat resembles the reasoning behind the schemes for unknown SSP-diameter D in the adversarial SSP algorithms of Rosenberg and Mansour (2021, App. I) and Chen and Luo (2021, App. E) (recall that $D \triangleq \max_{s\in\mathcal{S}}\min_{\pi\in\Pi_{\text{proper}}} T^\pi(s)$ and that $B_\star \le$

Algorithm D.1: Algorithm for unknown $B_\star$: Parameter-free EB-SSP

Input: $\mathcal{S},\ s_0 \in \mathcal{S},\ g \notin \mathcal{S},\ \mathcal{A},\ \delta$.
Optional input: cost perturbation $\eta \in [0, 1]$.
Set up **global constants:** $\mathcal{S},\ \mathcal{A},\ s_0 \in \mathcal{S},\ g \notin \mathcal{S},\ \eta$.
Set up **global variables:** $t,\ j,\ N(),\ n(),\ \widehat{P},\ \theta(),\ \widehat{c}(),\ Q(),\ V()$.
Set estimate $\widetilde{B} \leftarrow 1$.
Set current starting state $s_{\text{start}} \leftarrow s_0$.
Set $t \leftarrow 1,\ k \leftarrow 1,\ j \leftarrow 0$.
For $(s, a, s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}'$, set $N(s, a) \leftarrow 0;\ n(s, a) \leftarrow 0;\ N(s, a, s') \leftarrow 0;\ \widehat{P}_{s,a,s'} \leftarrow 0;\ \theta(s, a) \leftarrow 0;\ \widehat{c}(s, a) \leftarrow 0;\ Q(s, a) \leftarrow 0;\ V(s) \leftarrow 0$.
Set phase counter $\phi \leftarrow 1$.
while True **do**
    Set $s_{\text{cur}},\ \widetilde{B}_{\text{cur}},\ k_{\text{cur}} \leftarrow$ PHASE $(s_{\text{start}},\ \widetilde{B},\ k)$ (Algorithm D.2).
    \\ PHASE *halts because of* $B_\star$ *underestimation, entering a new* ***phase***
    Set $s_{\text{start}} \leftarrow s_{\text{cur}},\ k \leftarrow k_{\text{cur}},\ \widetilde{B} \leftarrow 2\widetilde{B}_{\text{cur}}$, and increment phase index $\phi \leftarrow \phi + 1$.

$D \leq T_\star$). Note, however, that these schemes change their algorithms' structure: whenever the agent is in a state that is insufficiently visited, it executes the Bernstein-SSP algorithm of Rosenberg et al. (2020) with unit costs until the goal is reached. In other words, these schemes first learn to reach the goal (regardless of the costs) and then focus on minimizing the costs to goal. In contrast, our scheme for unknown $B_\star$ targets the original SSP objective from the start and it does *not* fundamentally alter our algorithm EB-SSP with known $B_\star$. Indeed, the only addition of parameter-free EB-SSP is a *dual tracking* of the cumulative costs and VISGO ranges, and a *careful increment* of the estimate $\widetilde{B}$ in the bonus. Finally, our scheme only adds "horizon-free" lower-order terms (i.e., $B_\star, S, A$) as shown in Theorem 5.7, as opposed to the aforementioned schemes that introduce a lower-order dependence on the SSP-diameter D, which may be much larger than $B_\star$.

Algorithm D.2: Subroutine PHASE

Input: $s_{\text{start}} \in \mathcal{S},\ \widetilde{B},\ k$.
Global constants: $\mathcal{S},\ \mathcal{A},\ s_0 \in \mathcal{S},\ g \notin \mathcal{S},\ \eta$.
Global variables: $t,\ j,\ N(),\ n(),\ \widehat{P},\ \theta(),\ \widehat{c}(),\ Q(),\ V()$.
Specify: Trigger set $\mathcal{N} \leftarrow \{2^{j-1} : j = 1, 2, \ldots\}$. Constants $c_1 = 6,\ c_2 = 36,\ c_3 = 2\sqrt{2},\ c_4 = 2\sqrt{2}$. Large enough absolute constant $x > 0$ (so that Equation (D.8) holds, see Section D.7.3).
Set $C \leftarrow 0$. \\ *Reinitialize cumulative cost tracker*
for episode $k_{\text{cur}} = k, k+1, \ldots$ **do**
  if $\sqrt{k_{\text{cur}}}/(S^{3/2}A^{1/2}) > \widetilde{B}$ **then**
    Set $\widetilde{B} \leftarrow \sqrt{k_{\text{cur}}}/(S^{3/2}A^{1/2})$, and set $j \leftarrow j+1,\ \varepsilon_{\text{VI}} \leftarrow 2^{-j}/(SA)$.
    Info, $Q,\ V \leftarrow$ VISGO $(\widetilde{B},\ \varepsilon_{\text{VI}})$.
    if Info = Fail **then**
      \\ *Second halting condition:* VISGO *range exceeds threshold*
      return $s_t,\ \widetilde{B},\ k_{\text{cur}}$.
  Set $s_t \leftarrow \begin{cases} s_{\text{start}}, & k_{\text{cur}} = k, \\ s_0, & \text{otherwise.} \end{cases}$
  while $s_t \neq g$ **do**
    Take action $a_t = \arg\min_{a \in \mathcal{A}} Q(s_t, a)$, incur cost c_t and observe next state $s_{t+1} \sim P(\cdot|s_t, a_t)$.
    Set $(s, a, s', c) \leftarrow (s_t, a_t, s_{t+1}, \max\{c_t, \eta\})$ and $t \leftarrow t+1$.
    Set $N(s,a) \leftarrow N(s,a)+1$, $\theta(s,a) \leftarrow \theta(s,a)+c$, $C \leftarrow C+c$, $N(s,a,s') \leftarrow N(s,a,s')+1$, and set

$$C_{\text{bound}} \leftarrow k_{\text{cur}}\widetilde{B} + 3x\left(\widetilde{B}\sqrt{SAk_{\text{cur}}}\log_2\left(\frac{2\widetilde{B}tSA}{\delta}\right) + \widetilde{B}S^2A\log_2^2\left(\frac{2\widetilde{B}tSA}{\delta}\right)\right). \tag{D.10}$$

    if $C > C_{\text{bound}}$ **then**
      \\ *First halting condition: cumulative cost exceeds threshold*
      return $s_t,\ \widetilde{B},\ k_{\text{cur}}$.
    if $N(s,a) \in \mathcal{N}$ **then**
      Set $\widehat{c}(s,a) \leftarrow \mathbb{I}[N(s,a) \geq 2]\frac{2\theta(s,a)}{N(s,a)} + \mathbb{I}[N(s,a) = 1]\theta(s,a)$ and $\theta(s,a) \leftarrow 0$.
      For all $s' \in \mathcal{S}$, set $\widehat{P}_{s,a,s'} \leftarrow N(s,a,s')/N(s,a)$, $n(s,a) \leftarrow N(s,a)$, and set $j \leftarrow j+1,\ \varepsilon_{\text{VI}} \leftarrow 2^{-j}/(SA)$.
      Info, $Q,\ V \leftarrow$ VISGO $(\widetilde{B},\ \varepsilon_{\text{VI}})$.
      if Info = Fail **then**
        \\ *Second halting condition:* VISGO *range exceeds threshold*
        return $s_t,\ \widetilde{B},\ k_{\text{cur}}$.

Algorithm D.3: Subroutine VISGO

Inputs: $\widetilde{B},\ \varepsilon_{\text{VI}}$.
Global constants: $\mathcal{S},\ \mathcal{A},\ s_0 \in \mathcal{S},\ g \notin \mathcal{S},\ \eta$.
Global variables: $t,\ j,\ N(),\ n(),\ \widehat{P},\ \theta(),\ \widehat{c}(),\ Q(),\ V()$.
For all $(s,a,s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}'$, set

$$\widetilde{P}_{s,a,s'} \leftarrow \frac{n(s,a)}{n(s,a)+1}\widehat{P}_{s,a,s'} + \frac{\mathbb{I}[s'=g]}{n(s,a)+1}.$$

For all $(s,a) \in \mathcal{S} \times \mathcal{A}$, set $n^+(s,a) \leftarrow \max\{n(s,a),1\},\ \iota_{s,a} \leftarrow \ln\left(\frac{12SAS'[n^+(s,a)]^2}{\delta}\right)$.
Set $i \leftarrow 0$, $V^{(0)} \leftarrow 0$, $V^{(-1)} \leftarrow +\infty$.
while $\|V^{(i)} - V^{(i-1)}\|_\infty > \varepsilon_{\text{VI}}$ **do**
  For all $(s,a) \in \mathcal{S} \times \mathcal{A}$, set

$$b^{(i+1)}(s,a) \leftarrow \max\left\{c_1\sqrt{\frac{\mathbb{V}(\widetilde{P}_{s,a},V^{(i)})\iota_{s,a}}{n^+(s,a)}},\ c_2\frac{\widetilde{B}\iota_{s,a}}{n^+(s,a)}\right\} + c_3\sqrt{\frac{\widehat{c}(s,a)\iota_{s,a}}{n^+(s,a)}} + c_4\frac{\widetilde{B}\sqrt{S'\iota_{s,a}}}{n^+(s,a)}, \tag{D.11}$$

$$Q^{(i+1)}(s,a) \leftarrow \max\left\{\widehat{c}(s,a) + \widetilde{P}_{s,a}V^{(i)} - b^{(i+1)}(s,a),\ 0\right\}, \tag{D.12}$$

$$V^{(i+1)}(s) \leftarrow \min_a Q^{(i+1)}(s,a). \tag{D.13}$$

  Set $V^{(i+1)}(g) \leftarrow 0$ and $i \leftarrow i+1$.
  if $\|V^{(i)}\|_\infty > \widetilde{B}$ **then**
    \\ *Second halting condition:* VISGO *range exceeds threshold*
    return Fail, $Q^{(i)},\ V^{(i)}$.
return Success, $Q^{(i)},\ V^{(i)}$.

Appendix E

Complements on Chapter 7

E.1 Efficient Computation of Optimistic SSP Policy

In this section, we recall how to compute an optimistic stochastic shortest path (SSP) policy using an extended value iteration (EVI) scheme tailored to SSP, as explained in Chapter 4. The only difference here is that we leverage a Bernstein-based construction of confidence intervals, as done by e.g., Rosenberg et al. (2020).

Consider as input an SSP-MDP instance $M^\dagger \triangleq \langle \mathcal{S}^\dagger, \mathcal{A}, c, P, s^\dagger \rangle$, with goal $s^\dagger$, non-goal states $\mathcal{S}^\dagger = \mathcal{S} \setminus \{s^\dagger\}$, actions $\mathcal{A}$, unknown dynamics P, and known cost function with costs in $[c_{\min}, 1]$ where $c_{\min} > 0$. We assume that there exists at least one proper policy (i.e., that reaches the goal $s^\dagger$ with probability one when starting from any state in $\mathcal{S}^\dagger$). Note that in particular such condition is verified under Assumption 7.2. We denote by $N(s,a)$ the current number of samples available at the state-action pair (s,a) and set $N^+(s,a) \triangleq \max\{1, N(s,a)\}$. We also denote by $\widehat{P}$ the current empirical average of transitions: $\widehat{P}(s'|s,a) = N(s,a,s')/N(s,a)$. The algorithm first computes a set of plausible SSP-MDPs defined as

$$\mathcal{M}^\dagger \triangleq \Big\{ \langle \mathcal{S}^\dagger, \mathcal{A}, c, \widetilde{P}, s^\dagger \rangle \mid \widetilde{P}(s^\dagger|s^\dagger,a) = 1,\ \widetilde{P}(s'|s,a) \in \mathcal{B}(s,a,s'),\ \sum_{s'} \widetilde{P}(s'|s,a) = 1 \Big\},$$

where for any $(s,a) \in \mathcal{S}^\dagger \times \mathcal{A}$, $\mathcal{B}(s,a,s')$ is a high-probability confidence set on the dynamics of the true SSP-MDP $M^\dagger$. Specifically, we define the compact sets $\mathcal{B}(s,a,s') \triangleq [\widehat{P}(s'|s,a) - \beta(s,a,s'), \widehat{P}(s'|s,a) + \beta(s,a,s')] \cap [0,1]$, where

$$\beta(s,a,s') \triangleq 2\sqrt{\frac{\widehat{\sigma}^2(s'|s,a)}{N^+(s,a)} \log\left(\frac{2SAN^+(s,a)}{\delta}\right)} + \frac{6\log\left(\frac{2SAN^+(s,a)}{\delta}\right)}{N^+(s,a)},$$

where $\widehat{\sigma}^2(s'|s,a) \triangleq \widehat{P}(s'|s,a)(1-\widehat{P}(s'|s,a))$ is the variance of the empirical transition $\widehat{P}(s'|s,a)$. Importantly, the choice of $\beta(s,a,s')$ guarantees that $M^\dagger \in \mathcal{M}^\dagger$ with high probability. Indeed, let us now spell out the high-probability event. Denote by $\mathcal{E}$ the event under which for any time step $t \geq 1$ and for any state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$ and next state $s' \in \mathcal{S}$, it holds that

$$|\widehat{P}_t(s'|s,a) - P(s'|s,a)| \leq \beta_t(s,a,s'). \tag{E.1}$$

Given the way the confidence intervals are constructed using the empirical Bernstein inequality (see e.g., Audibert et al., 2009; Fruit et al., 2020; Rosenberg et al., 2020), we have $\mathbb{P}(\mathcal{E}) \geq 1-\delta$. Throughout the remainder of the analysis, we will assume that the event $\mathcal{E}$ holds.

Once $\mathcal{M}^\dagger$ has been computed, the algorithm applies an extended value iteration (EVI) scheme to compute a policy with lowest optimistic value. Formally, it defines the extended optimal Bellman operator $\widetilde{\mathcal{L}}$ such that for any vector $\widetilde{v} \in \mathbb{R}^{S^\dagger}$ and non-goal state $s \in \mathcal{S}^\dagger$,

$$\widetilde{\mathcal{L}}\widetilde{v}(s) \triangleq \min_{a \in \mathcal{A}} \Big\{ c(s,a) + \min_{\widetilde{P} \in \mathcal{B}(s,a)} \sum_{s' \in \mathcal{S}^\dagger} \widetilde{P}(s'|s,a)\widetilde{v}(s') \Big\}.$$

We consider an initial vector $\widetilde{v}_0 \triangleq 0$ and set iteratively $\widetilde{v}_{i+1} \triangleq \widetilde{\mathcal{L}}\widetilde{v}_i$. For a predefined VI precision $\mu_{\text{VI}} > 0$, the stopping condition is reached for the first iteration j such that $\|\widetilde{v}_{j+1} - \widetilde{v}_j\|_\infty \leq \mu_{\text{VI}}$. The policy $\widetilde{\pi}$ is then selected to be the optimistic greedy policy w.r.t. the vector $\widetilde{v}_j$. While $\widetilde{v}_j$ is *not* the value function of $\widetilde{\pi}$ in the optimistic model $\widetilde{P}$, which we denote by $\widetilde{V}^{\widetilde{\pi}}$, both quantities can be related according to the following lemma, which stems exactly from Lemma 2.13. We denote by $V^\star$ (resp. $\widetilde{V}^\star$) the optimal value function in the true (resp. optimistic) SSP instance.

Lemma E.1. *Under the event $\mathcal{E}$, the following component-wise inequalities hold:* ***1)*** $\widetilde{v}_j \leq V^\star$, ***2)*** $\widetilde{v}_j \leq \widetilde{V}^\star \leq \widetilde{V}^{\widetilde{\pi}}$, ***3)*** *If the* VI *precision level verifies $\mu_{\text{VI}} \leq \frac{c_{\min}}{2}$, then $\widetilde{V}^{\widetilde{\pi}} \leq \left(1 + \frac{2\mu_{\text{VI}}}{c_{\min}}\right)\widetilde{v}_j$.*

Note that for the purposes of GOSPRL (Algorithm 7.1), the VI precision μ_{VI} can for example be selected as in Chapter 4 equal to $1/(2t_k)$ with t_k the current time step, which only translates in a negligible, lower-order error in the sample complexity result of Corollary E.2.

E.2 Proof of Theorem 7.5

We first focus on the special case where the sampling requirements are time- and action-independent, i.e., $b : \mathcal{S} \to \mathbb{N}$.

Corollary E.2. *Under Assumption 7.2, for any input sampling requirements $b : \mathcal{S} \to \mathbb{N}$ with $B \triangleq \sum_{s\in\mathcal{S}} b(s)$ and for any confidence level $\delta \in (0,1)$,*

$$\mathcal{C}(\text{GOSPRL}, b, \delta) = \widetilde{O}\Big(BD + D^{3/2}S^2A\Big), \tag{E.2}$$

$$\mathcal{C}(\text{GOSPRL}, b, \delta) = \widetilde{O}\Big(\sum_{s\in\mathcal{S}} (D_s b(s) + D_s^{3/2}S^2A)\Big). \tag{E.3}$$

E.2.1 Proof of Corollary E.2

We denote by $\mathcal{E}$ the event — which holds with probability at least $1-\delta$ — such that the empirical Bernstein inequalities stated in Equation (E.1) hold simultaneously for each time step t and each state-action-next-state triplet $(s,a,s') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}$, i.e.,

$$|\widehat{P}_t(s'|s,a) - P(s'|s,a)| \le \beta_t(s,a,s'),$$

where

$$\beta_t(s,a,s') \triangleq 2\sqrt{\frac{\widehat{\sigma}_t^2(s'|s,a)}{N_t^+(s,a)}\log\left(\frac{2SAN_t^+(s,a)}{\delta}\right)} + \frac{6\log\left(\frac{2SAN_t^+(s,a)}{\delta}\right)}{N_t^+(s,a)}.$$

We recall that at the beginning of each episode j, the under-sampled states $\mathcal{G}_j$ are cast as goal states.[1] GOSPRL then constructs an SSP-MDP instance $M_j \triangleq \langle \mathcal{S}_j, \mathcal{A}, P_j, c_j, \mathcal{G}_j\rangle$, where $\mathcal{G}_j$ encapsulates the goal states and $\mathcal{S}_j := \mathcal{S}\setminus\mathcal{G}_j$ the non-goal states. The transition model P_j is the same as the original P except for the transitions exiting the goal states which are redirected as a self-loop, i.e., $P_j(s'|s,a) \triangleq P(s'|s,a)$ and $P_j(g|g,a) \triangleq 1$ for any $(s,s',a,g) \in \mathcal{S}_j\times\mathcal{S}\times\mathcal{A}\times\mathcal{G}_j$. As for the cost function c_j, for any action $a\in\mathcal{A}$, any goal state $g\in\mathcal{G}_j$ is zero-cost (i.e., $c_j(g,a)\triangleq 0$), while the non-goal costs are unitary (i.e., $c_j(s,a)\triangleq 1$ for all $s\in\mathcal{S}_j$).

We now make more explicit the way the SSP optimistic policy is constructed at the beginning of any episode j. Denote by $\widehat{P}_j$ the empirical transitions of the induced SSP-MDP M_j. We consider the following confidence intervals β'_j in the optimistic SSP policy computation from Section E.1

$$\forall (a,s') \in \mathcal{A}\times\mathcal{S}, \quad \forall s\notin\mathcal{G}_j,\ \beta'_j(s,a,s') \triangleq \beta_t(s,a,s'), \quad \forall s\in\mathcal{G}_j,\ \beta'_j(s,a,s') = 0.$$

[1] In the case of state-only requirements, a state s is considered under-sampled if $\sum_{a\in\mathcal{A}} N_{t-1}(s,a) < b(s)$. In the case of state-action requirements, a state s is considered under-sampled if $\exists a\in\mathcal{A}, N_{t-1}(s,a) < b(s,a)$.

We denote by $\widetilde{P}_j$ the optimistic model computed by the EVI scheme with such confidence intervals. Now, denoting by $\mathcal{P}(\mathcal{S})$ the power set of the state space $\mathcal{S}$, we have the following event inclusion

$$\begin{aligned}\mathcal{E} \subseteq \mathcal{E}' \triangleq \Big\{ & \forall j \geq 1,\ \forall \mathcal{G}_j \in \mathcal{P}(\mathcal{S}),\ \forall (a, s') \in \mathcal{A} \times \mathcal{S}, \\ & \forall s \notin \mathcal{G}_j,\ |\widetilde{P}_j(s'|s,a) - \widehat{P}_j(s'|s,a)| \leq \beta'_j(s,a,s'), \\ & \forall s \in \mathcal{G}_j,\ \widetilde{P}_j(s|s,a) = 1 \Big\}.\end{aligned}$$

Indeed, the only transitions that are redirected from P to P_j are those that exit from states in $\mathcal{G}_j$ and they are set to deterministically self-loop, which implies that they do not contain any uncertainty. Note that $\mathcal{E}'$ is the event that we require to hold so that the SSP analysis goes through for *any* considered SSP-MDP M_j. From the inclusion above, we have that the event $\mathcal{E}'$ holds with probability at least $1 - \delta$, and we assume from now on that it holds.

We denote by H_j the length of each episode j, specifically $H_j \triangleq \min_{h \geq 1}\{s_{j,h} \in \mathcal{G}_j\}$, where we denote by $s_{j,h}$ the h-th state visited during episode j. We denote by $\overline{s}_j \triangleq s_{j,H_j}$ the goal state in $\mathcal{G}_j$ that is reached at the end of episode j. Correspondingly, the starting state of each episode j, denoted by $\underline{s}_j$, also varies: if $j = 1$ it is the initial state s_0 of the learning interaction, otherwise it is equal to $\overline{s}_{j-1}$ which is the reached goal state at the end of the previous episode $j - 1$.[2] The important property is that both the starting state $\underline{s}_j$ and the goal states $\mathcal{G}_j$ are measurable (i.e., known and fixed) at the beginning of each episode j.

We define R_J the regret after J episodes as follows

$$R_J \triangleq \sum_{j=1}^{J} \sum_{h=1}^{H_j} c_j(s_{j,h}, a_{j,h}) - \sum_{j=1}^{J} \min_{\pi} V_j^{\pi}(\underline{s}_j), \tag{E.4}$$

where we denote by $V_j^{\pi}(s)$ the value function of a policy π starting from state s in the SSP-MDP instance M_j. We also denote by $\mathcal{C}(\text{GOSPRL}, b)$ the random variable of the total time accumulated by GOSPRL until the sampling requirements b are met.

On the one hand, the regret R_J can be lower bounded almost surely as follows

$$R_J \overset{\text{(a)}}{=} \sum_{j=1}^{J} H_j - \sum_{j=1}^{J} \min_{\pi} \mathbb{E}\left[\tau_{\pi}(\underline{s}_j \to \mathcal{G}_j)\right]$$

[2] This choice of initial state for episodes is when we have state-only sampling requirements. If we instead have state-action requirements, the action taken at each reached goal state matters. In that case, when episode $j - 1$ reaches a goal state $\overline{s}_{j-1}$, the agent takes a relevant action $\overline{a}_{j-1}$ and we then consider that the starting state $\underline{s}_j$ at the next episode j is distributed according to $P(\cdot|\overline{s}_{j-1}, \overline{a}_{j-1})$. The action $\overline{a}_{j-1}$ is naturally specified by the algorithm depending on the current and desired requirements $N(\overline{s}_{j-1}, \cdot)$ and $b(\overline{s}_{j-1}, \cdot)$, i.e., we should select $\overline{a}_{j-1} \in \{a \in \mathcal{A} : N(\overline{s}_{j-1}, a) < b(\overline{s}_{j-1}, a)\}$ with N the state-action counter at the end of episode $j - 1$. We explain in Section F.4 the way we select this action in our experiments.

$$\overset{(b)}{=} \mathcal{C}(\text{GOSPRL}, b) - \sum_{j=1}^{J} \min_{\pi} \mathbb{E}\left[\tau_\pi(\underline{s}_j \to \mathcal{G}_j)\right]$$
$$\overset{(c)}{\geq} \mathcal{C}(\text{GOSPRL}, b) - DB, \tag{E.5}$$

where (a) stems from the fact that all the non-goal costs are unitary, (b) comes from the definition of the index J (i.e., the episode at which all the sampling requirements are met) and (c) combines that $J \leq B$ almost surely and that $\mathbb{E}\left[\tau_\pi(\underline{s}_j \to \mathcal{G}_j)\right] \leq D$ by definition of the diameter D.

On the other hand, retracing the analysis of Rosenberg et al. (2020), the derivation of the regret bound can be easily extended to varying initial states and varying (possibly multiple[3]) goal states across episodes, as long as they are all *known* to the learner at the beginning of each episode (which is our case here). In particular, the high-probability event is $\mathcal{E}' \supseteq \mathcal{E}$ defined above, which holds with probability at least $1 - \delta$. Under this event, we have from Rosenberg et al. (2020, Thm. 2.4) that GOSPRL satisfies

$$R_J = \widetilde{O}\left(DS\sqrt{AJ} + D^{3/2}S^2A\right). \tag{E.6}$$

Combining Equation (E.5) and E.6 yields that with probability at least $1 - \delta$, we have

$$\mathcal{C}(\text{GOSPRL}, b) \leq \widetilde{O}\left(BD + DS\sqrt{AJ} + D^{3/2}S^2A\right).$$

Given that $J \leq B$ almost surely, we get

$$\mathcal{C}(\text{GOSPRL}, b, \delta) \leq \widetilde{O}\left(BD + DS\sqrt{AB} + D^{3/2}S^2A\right). \tag{E.7}$$

We now proceed with a separation of cases. If $B \geq S^2A$, we have $DS\sqrt{AB} \leq BD$. Otherwise, if $B \leq S^2A$, we have $DS\sqrt{AB} \leq D^{3/2}S^2A$. This implies that the second summand in the $\widetilde{O}$ sum in Equation (E.7) can be removed, which yields the first sought-after bound of Equation (E.2).

In order to obtain the second more state-dependent bound of Equation (E.3), the bound of Equation (E.6) is too loose, hence we need to extend the analysis of Rosenberg et al. (2020) to bring out dependencies on $b(s)$ and D_s. In particular, we consider a similar decomposition

[3]Note that the SSP formulation can easily handle multiple goal states. To justify this statement, we make explicit an SSP instance with single goal state that is strictly equivalent to the SSP instance M_j at hand with multiple goals $\mathcal{G}_j$. To do so, we introduce an artificial terminal state λ and define the SSP-MDP Q_j with $\mathcal{S} \cup \{\lambda\}$ states (the non-goals are $\mathcal{S}$ while the unique goal is λ). Its transition dynamics q_j is defined as follows: $q_j(\lambda|\lambda, a) = 1, \forall s \notin \mathcal{G}_j, q_j(s'|s, a) = P_j(s'|s, a)$, and $\forall s \in \mathcal{G}_j, q_j(\lambda|s, a) = 1$. Its cost function is set to the original costs c_j for states not in $\mathcal{G}_j$, and to 0 (or equivalently any constant) for states in $\mathcal{G}_j$, and finally to 0 for the terminal state λ. This construction mirrors the one proposed by Bertsekas in the lecture https://web.mit.edu/dimitrib/www/DP_Slides_2015.pdf (page 25). Note that the SSP instance M_j with multiple goal states $\mathcal{G}_j$ is equivalent to the single-goal SSP instance Q_j. The artificial terminal state λ is not formally necessary; it justifies why having multiple goal states is well-defined from an analysis point of view.

in *epochs* and *intervals* that we carefully adapt for our purposes of varying goal states. The first epoch starts at the first time step and each epoch ends once the number of visits to some state-action pair is doubled. We denote by $\mathcal{G}_m$ the goal states that are considered during interval m and by $D_{\mathcal{G}_m}$ the SSP-diameter of the goal states $\mathcal{G}_m$. The first interval starts at the initial time step and each interval m (with goal states $\mathcal{G}_m$) ends once one of the four following conditions holds: (i) the length of the interval reaches $D_{\mathcal{G}_m}$; (ii) an unknown state-action pair is reached (where a state-action pair (s, a) becomes known if its total number of visits exceeds $\alpha D_{\mathcal{G}_m} S \log(D_{\mathcal{G}_m} SA/\delta)$ for some constant $\alpha > 0$); (iii) the current episode ends, i.e., the a goal state in $\mathcal{G}_m$ is reached; (iv) the current epoch ends, i.e., the number of visits to some state-action pair is doubled. Finally, we denote by H_m the length of each interval m, by M the total number of intervals and by $T_M \triangleq \sum_{m=1}^{M} H_m$ the total time steps. As such, T_M amounts to the sample complexity that we seek to bound. Note that the goal states $\mathcal{G}_m$ are measurable at the beginning of the attempt m. Hence we can extend the reasoning of Rosenberg et al. (2020, Appendix B.2.7 & B.2.8) to varying goal states using the decomposition described above. Assuming throughout that the high-probability events hold, we get[4]

$$T_M = \widetilde{O}\left(\sum_{m \in \mathcal{M}^{(iii)}} D_{\mathcal{G}_m} + S\sqrt{A}\sqrt{\sum_{m=1}^{M} D_{\mathcal{G}_m}^2} + DS^2A \right), \tag{E.8}$$

where $\mathcal{M}^{(iii)}$ is defined as the set of intervals that end according to condition (iii). We now proceed with the following decomposition, which is analogous to Rosenberg et al. (2020, Observation 4.1)

$$\sum_{m=1}^{M} D_{\mathcal{G}_m}^2 \leq \sum_{m: H_m \geq D_{\mathcal{G}_m}} D_{\mathcal{G}_m}^2 + \sum_{m: H_m < D_{\mathcal{G}_m}} D_{\mathcal{G}_m}^2.$$

Using that $D_{\mathcal{G}_m} \leq D$, the first term can be bounded as

$$\sum_{m: H_m \geq D_{\mathcal{G}_m}} D_{\mathcal{G}_m}^2 \leq D \sum_{m: H_m \geq D_{\mathcal{G}_m}} D_{\mathcal{G}_m} \leq D \sum_{m: H_m \geq D_{\mathcal{G}_m}} H_m \leq D \sum_{m=1}^{M} H_m = DT_M.$$

As for the second term, we observe that it removes intervals ending under the condition (i) and thus only accounts for intervals ending under the conditions (ii), (iii) or (iv). We now perform the following key partition of intervals: each interval is categorized depending on the first goal state that ends up being reached at the end or after the considered interval. We call this goal state the *retrospective goal state of the interval*. This retrospective categorization of intervals can be

[4]The intuition behind Equation (E.8) comes from the Cauchy-Schwarz inequality. For instance, let us consider the objective of bounding the quantity $Y \triangleq \sum_m x_m \sqrt{y_m}$, where the (x_m) correspond to the SSP-diameters considered at each interval m and the (y_m) are the summands whose sums are bounded by Rosenberg et al. (2020, Lemma B.16). In the latter work, denoting by $\overline{x}$ the common upper bound on the (x_m), the analysis yields $Y \leq \overline{x} \sum_m \sqrt{y_m} \leq \overline{x}\sqrt{M}\sqrt{\sum_m y_m}$. In contrast, our setting requires to perform the tighter inequality $Y \leq \sqrt{\sum_m x_m^2}\sqrt{\sum_m y_m}$.

performed since it does not appear at an algorithmic level, but only appears at an analysis-level after Equation (E.8) is obtained, in order to simplify it. For any interval m, we denote by s_m its retrospective goal. Likewise, let us denote by M_s (resp. $\mathcal{M}_s$) the number (resp. the set) of intervals with retrospective goal state s. Finally, for any $j \in \{ii, iii, iv\}$, we denote we denote by $M^{(j)}$ (resp. $\mathcal{M}^{(j)}$) the number (resp. the set) of intervals that end according to condition (j), and by $M_s^{(j)}$ (resp. $\mathcal{M}_s^{(j)}$) the number (resp. the set) of intervals with retrospective goal state s that end according to condition (j). We can now write

$$\sum_{m:H_m<D_{\mathcal{G}_m}} D_{\mathcal{G}_m}^2 = \sum_{m\in\mathcal{M}^{(ii)}} D_{\mathcal{G}_m}^2 + \sum_{m\in\mathcal{M}^{(iii)}} D_{\mathcal{G}_m}^2 + \sum_{m\in\mathcal{M}^{(iv)}} D_{\mathcal{G}_m}^2 = \sum_{j\in\{ii,iii,iv\}} \sum_{m\in\mathcal{M}^{(j)}} D_{\mathcal{G}_m}^2.$$

Now, for any $j \in \{ii, iii, iv\}$,

$$\begin{aligned}\sum_{m\in\mathcal{M}^{(j)}} D_{\mathcal{G}_m}^2 = \sum_{m\in\mathcal{M}^{(j)}} \Big(\sum_{s\in\mathcal{S}} \mathbb{1}_{\{s_m=s\}}\Big) D_{\mathcal{G}_m}^2 &= \sum_{s\in\mathcal{S}} \sum_{m\in\mathcal{M}_s^{(j)}} D_{\mathcal{G}_m}^2 \\ &\overset{(a)}{\leq} \sum_{s\in\mathcal{S}} \sum_{m\in\mathcal{M}_s^{(j)}} D_s^2 \\ &= \sum_{s\in\mathcal{S}} M_s^{(j)} D_s^2,\end{aligned}$$

where inequality (a) comes from Lemma E.3 stated later. Moreover, we have

$$M_s^{(ii)} = \widetilde{O}\left(D_s S^2 A\right); \qquad M_s^{(iii)} \leq b(s); \qquad M^{(iv)} \leq 2SA\log(T_M).$$

While the first and third bounds above are similar to those considered by Rosenberg et al. (2020), the key difference lies in the second bound, which leverages that the number of intervals that end in the goal state s is, by definition of our problem, upper bounded by the number of samples required at state s, i.e., $b(s)$. All in all, this implies that

$$\sum_{m:H_m<D_{\mathcal{G}_m}} D_{\mathcal{G}_m}^2 \leq \widetilde{O}\left(\sum_{s\in\mathcal{S}} D_s^3 S^2 A\right) + \sum_{s\in\mathcal{S}} b(s) D_s^2 + \widetilde{O}\left(D^2 SA\right).$$

Moreover, in a similar manner as above, we bound the first term of Equation (E.8) as follows

$$\sum_{m\in\mathcal{M}^{(iii)}} D_{\mathcal{G}_m} = \sum_{s\in\mathcal{S}} M_s^{(iii)} D_s \leq \sum_{s\in\mathcal{S}} D_s b(s).$$

Putting everything together back into Equation (E.8) and simplifying using the subadditivity of the square root, we get

$$T_M = \widetilde{O}\left(\sum_{s\in\mathcal{S}} D_s b(s) + DS^2 A + S\sqrt{ADT_M} + S\sqrt{A}\sqrt{\sum_{s\in\mathcal{S}} b(s) D_s^2} + S^2 A \sum_{s\in\mathcal{S}} D_s^{3/2}\right).$$

Using that $x \leq c_1\sqrt{x} + c_2$ implies $x \leq (c_1 + \sqrt{c_2})^2$ for $c_1 \geq 0$ and $c_2 \geq 0$, we obtain

$$T_M = \widetilde{O}\left(\left[S\sqrt{DA} + \sqrt{\sum_{s\in\mathcal{S}} D_s b(s)} + \sqrt{S\sqrt{A}\sqrt{\sum_{s\in\mathcal{S}} b(s)D_s^2} + \sqrt{S^2A\sum_{s\in\mathcal{S}} D_s^{3/2}}}\right]^2\right). \tag{E.9}$$

We now apply the Cauchy-Schwarz inequality to simplify the third summand

$$S\sqrt{A}\sqrt{\sum_{s\in\mathcal{S}} b(s)D_s^2} \leq \sum_{s\in\mathcal{S}} \sqrt{S^2AD_s}\sqrt{D_s b(s)} \leq \sqrt{\sum_{s\in\mathcal{S}} D_s b(s)}\sqrt{S^2A\sum_{s\in\mathcal{S}} D_s}.$$

Let us introduce $x \triangleq \sqrt{\sum_{s\in\mathcal{S}} D_s b(s)}$ and $y \triangleq \sqrt{S^2A\sum_{s\in\mathcal{S}} D_s^{3/2}}$. Plugging the simplifications into Equation (E.9) finally yields with probability at least $1-\delta$ that $T_M = \widetilde{O}\left((x + \sqrt{xy} + y)^2\right) = \widetilde{O}\left((x+y)^2\right) = \widetilde{O}\left(x^2 + y^2\right)$. Since T_M amounts to the sample complexity, we get the desired bound of Equation (E.3), which reads

$$\mathcal{C}(\text{GOSPRL}, b, \delta) = \widetilde{O}\left(\sum_{s\in\mathcal{S}} D_s b(s) + S^2A\sum_{s\in\mathcal{S}} D_s^{3/2}\right).$$

Lemma E.3. *For any set of goals $\mathcal{G} \subsetneq \mathcal{S}$, we introduce the meta SSP-diameter $D_\mathcal{G} \triangleq \max_{s\in\mathcal{S}\setminus\mathcal{G}} \min_\pi \mathbb{E}\left[\tau_\pi(s \to \mathcal{G})\right]$, where we define $\tau_\pi(s \to \mathcal{G}) \triangleq \min\{t \geq 0 : s_{t+1} \in \mathcal{G} \,|\, s_1 = s, \pi\}$. Then we have*

$$D_\mathcal{G} \leq \min_{s\in\mathcal{G}} D_s.$$

Proof. For any $g \in \mathcal{G}$, $s \in \mathcal{S}\setminus\mathcal{G}$ and policy π, we have $\mathbb{E}\left[\tau_\pi(s \to \mathcal{G})\right] \leq \mathbb{E}\left[\tau_\pi(s \to g)\right]$. In particular, this implies that for any $g \in \mathcal{G}$, $D_\mathcal{G} \leq D_g$, which immediately gives the result. □

E.2.2 From Corollary E.2 to Theorem 7.5

We now consider the general case of possibly action-dependent and time-dependent sampling requirements.

State-action requirements. First, GOSPRL can be easily extended from state requirements $b(s)$ to state-action requirements $b(s,a)$. Indeed, the only difference between these two settings occurs w.r.t. which action the algorithm takes at the end of a given episode (i.e., when a sought-after goal state is reached): for state-action requirements, any under-sampled action is taken (see footnote 2 for details). Bound-wise, the number of times where this scenario occurs is

at most B (since there are at most B episodes), hence the guarantee from Corollary E.2 is unaffected whatever the action executed once a goal state is reached.

Adaptive requirements. GOSPRL can be also easily extended to requirements $(b_t(s,a))_{t\geq 1}$ that vary over time, where b_t may be chosen adaptively depending on the samples observed so far (i.e., b_t is measurable w.r.t. the filtration up to time t). Indeed, the important property required in the derivations of Section E.2.1 that both the starting state and the goal states should be measurable (i.e., known and fixed) at the beginning of each episode still holds. As such, the sample complexity result of Corollary E.2 can be naturally extended by defining $B_\tau \triangleq \sum_{s,a} b_\tau(s,a)$, where τ is the first (random) time step when all the sampling requirements are met. In order for the sample complexity to remain bounded, a sufficient condition is Assumption 7.3. In particular, considering the sequence $b_t(s,a)$ to be upper bounded by a fixed threshold $\overline{b}(s,a)$ for each (s,a), the bound from Corollary E.2 trivially holds with $\overline{B} \triangleq \sum_{s,a} \overline{b}(s,a)$.

E.2.3 Remark

Notice that the "comparator" we are using in the definition of the regret in Equation (E.4) may not be the "global" optimum in terms of sample complexity. Indeed, the optimal sequence of strategies would result in a non-stationary policy $\pi^\star_{\mathcal{C}} \in \arg\min_\pi \mathcal{C}(\pi, b, \delta)$. Yet in our analysis, we compare the algorithmic performance with the larger quantity $\sum_{j=1}^{J} \min_\pi V^\pi_{\mathcal{G}_j}(\underline{s}_j)$, which corresponds to "greedily" minimizing each time to reach an under-sampled state in a sequential fashion. This highlights that GOSPRL does not *track* any optimal sampling allocation or distribution (i.e., it does not seek to "imitate" $\pi^\star_{\mathcal{C}}$), insofar as it discards the effect of traversing other states while reaching an undersampled goal state. While this means that some areas of the state space may be oversampled, GOSPRL is able to devote its full attention to the objective of minimizing the total sample complexity, instead of being mindful to avoid certain areas of the state space which it has already visited. We argue that this is what results in the appealing sample complexity of GOSPRL, whereas other techniques specifically designed to track distributions (via e.g., the Frank-Wolfe algorithmic scheme) struggle to minimize the sample complexity, as explained in Sections E.6 and 7.4.1.

E.3 Lower Bound

In this section, we provide three complementary results that lower bound the sample complexity of the problem of Definition 7.1.

① **First**, as stated in Lemma 7.4, we construct a simple MDP such that for any arbitrary sampling requirements $b(s)$, the (possibly non-stationary) policy minimizing the time to collect all samples has sample complexity of order $\Omega(\sum_{s\in\mathcal{S}} D_s b(s))$. We begin with a useful result.

Lemma E.4. *Let $q \in (0,1)$ and consider the Markov chain M_q with two states x, y whose dynamics p_q are as follows: $p_q(y|x) = q$, $p_q(x|x) = 1-q$ and $p_q(x|y) = 1$. Then M_q is communicating with diameter $D_q \triangleq \frac{1}{q}$. Moreover, denote by T_B the (random) time of the B-th visit to state y starting from any state, and assume that $B \geq 5$. Then with probability at least $\frac{1}{2}$, we have $T_B \geq \frac{B}{2q} + B = \frac{BD_q}{2} + B$.*

Proof. Introduce $X \triangleq \sum_{i=1}^{n} X_i$ where $X_i \sim Ber(q)$ (i.e., it follows a Bernoulli with parameter q) and we set $n \triangleq \frac{B}{2q}$. We have $\mathbb{E}[X] = nq = \frac{B}{2}$. Moreover, the Chernoff inequality entails that

$$\mathbb{P}(X \geq B) = \mathbb{P}(X \geq 2\mathbb{E}[X]) \leq \exp\Big(-\frac{\mathbb{E}[X]}{3}\Big) = \exp\Big(-\frac{B}{6}\Big) \leq \frac{1}{2},$$

where the last inequality holds whenever $B \geq 6\log(2)$. Note that the random variable T_B follows a negative binomial distribution for which each success accounts for two time steps instead of one. This means that with probability at least $\frac{1}{2}$,

$$T_B \geq n + B = \frac{B}{2q} + B = \frac{BD_q}{2} + B.$$

□

Let us now consider a state space $\mathcal{S} \triangleq \{s_1, \dots, s_S\}$ and arbitrary sampling requirements $b : \mathcal{S} \to \mathbb{N}$. We construct a wheel MDP with state space $\mathcal{S} \cup \{s_0\}$, where s_0 is the starting center state. There are $A = S$ actions available and the dynamics P are defined w.r.t. a set $(\varepsilon_i) \in (0,1)^S$ such that $\forall i \in [S]$, $P(s_i|s_0, a_i) = \varepsilon_i$, $P(s_0|s_0, a_i) = 1-\varepsilon_i$, and for every action a, $P(s_0|s_i, a) = 1$. Note that by having such $A = S$ actions, the attempts to collect relevant samples are independent, in the sense that at any $s \in \mathcal{S}$, the learner cannot rely on the attempts performed for the other states $s' \neq s$. Let us assume that $b(s) \geq 6\log(2S)$. From Lemma E.4, for any state $s \in \mathcal{S}$, with probability $1 - \frac{1}{2S}$, the time needed to collect $b(s)$ samples from state s is lower bounded by $\frac{b(s)}{2\varepsilon_i} + b(s)$, and furthermore we have $D_s = \frac{1}{\varepsilon_i} + 1$. Taking a union bound over the S states in $\mathcal{S}$ means that with probability at least $\frac{1}{2}$, the time to collect the required samples is lower bounded by $\sum_{s\in\mathcal{S}} \frac{b(s)(D_s-1)}{2} + b(s)$.

② **Second**, we show that the family of worst-case MDPs is relatively large. In fact, for any MDP with diameter D, we can perform a minor change to its dynamics without affecting the overall diameter and show that when the sampling requirements are concentrated in a single

state, any policy would take at least $\Omega(BD)$ steps to collect all the B samples. More specifically, there exists a class $\mathbb{C}$ of MDPs such that, for each MDP in $\mathbb{C}$, there exists a requirement function b and a finite threshold (that depends on the considered MDP) such that the $\Omega(BD)$ lower bound holds whenever B exceeds this threshold. The class $\mathbb{C}$ effectively encompasses a large number of environments: indeed, take *any* MDP M, then we can find an MDP M' in $\mathbb{C}$ such that M and M' differ in their transitions *only* at one state and have the same diameter. Formally, we have the following statement (proof in Section E.3.1).

Lemma E.5. *Fix any positive natural numbers S, A and D, and any MDP M with $S = |\mathcal{S}|$ states, $A = |\mathcal{A}|$ actions and diameter D. There exists a modification of the transitions of M at only one state which yields an MDP M' with the same diameter D, and there exists a finite integer $W_{\mathfrak{A},M',\delta}$ (depending on $\mathfrak{A}$, M') such that for any total requirement $B \geq W_{\mathfrak{A},M',\delta}$, there exists a function $b^\dagger : \mathcal{S} \to \mathbb{N}$ with $\sum_{s\in\mathcal{S}} b(s) = B$, such that, for any arbitrary starting state, the optimal non-stationary policy $\mathfrak{A}^\star$ needs $\mathcal{C}(\mathfrak{A}^\star, b^\dagger)$ time steps to collect the desired samples in the modified MDP M', where*

$$\mathbb{P}\left(\mathcal{C}(\mathfrak{A}^\star, b^\dagger) > \frac{(B-1)D}{2}\right) \geq \frac{1}{2}.$$

③ **Third**, we note that both results above do not take into account the added difficulty for the agent to have to deal with a learning process. To do so, we can draw inspiration from the lower bound on the expected regret for learning in an SSP problem derived by Rosenberg et al. (2020). Indeed, let us consider a environment M with one state $\overline{s}$ in which all the required samples are concentrated, i.e., $b \triangleq B\mathbb{1}_{\overline{s}}$ with $B \geq SA$. The $S-1$ other states s each contain a special action $a_s^\star$. The transition dynamics P are defined as follows: $P(\overline{s}|s, a_s^\star) = \frac{1}{D_{\overline{s}}}$, $P(s|s, a_s^\star) = 1 - \frac{1}{D_{\overline{s}}}$, $P(\overline{s}|s, a) = \frac{1-\nu}{D_{\overline{s}}}$, $P(s|s, a) = 1 - \frac{1-\nu}{D_{\overline{s}}}$ for any other action $a \in \mathcal{A}\backslash\{a_s^\star\}$, and finally $P(s|\overline{s}, a) = \frac{1}{S-1}$ for any action $a \in \mathcal{A}$, with $\nu \triangleq \sqrt{(S-1)AB}/64$. Recall that $D_{\overline{s}}$ is the SSP-diameter of state $\overline{s}$. The communicating, non-episodic structure of M naturally mimics the interaction of an agent with an SSP problem with goal state $\overline{s}$. Denoting by $\mathcal{C}(\mathfrak{A}, b)$ the (random) time required by any algorithm $\mathfrak{A}$ to collect the b sought-after samples, we obtain from Rosenberg et al. (2020, Thm. 2.7) that

$$\begin{aligned}\mathbb{E}\left[\mathcal{C}(\mathfrak{A}, b = B\mathbb{1}_{\overline{s}})\right] \geq \phi(B) &\triangleq \underbrace{(D_{\overline{s}}+1)B}_{\triangleq\phi_1(B)} + \underbrace{\frac{1}{1024} D_{\overline{s}}\sqrt{(S-1)AB}}_{\triangleq\phi_2(B)} \\ &= \sum_{s\in\mathcal{S}} \left((D_s+1)b(s) + \frac{1}{1024} D_s \sqrt{(S-1)Ab(s)}\right).\end{aligned}$$

This lower bound on the expected time to collect the samples implies in particular that no algorithm can meet the sampling requirements in less than $\widetilde{O}(\phi(B))$ time steps with high

probability. Importantly, note that this result is not contradictory with Corollary E.2. Indeed, as fleshed out in the proof in Section E.2, the upper bound of Corollary E.2 actually contains such a square root term $\phi_2(B)$, yet it is subsumed in the final bound by either the main-order term in $\sum_s b(s)D_s$ or the lower-order term constant w.r.t. B (see Equation (E.7)). We can decompose $\phi(B)$ in two factors: the second term $\phi_2(B)$ comes from the learning process of trying to match the behavior of the optimal policy, while the first term $\phi_1(B)$ stems from the need to navigate through the environment as opposed to the generative model assumption (as such, it is incurred even if the optimal policy is deployed from the start). Part ② of this section actually shows that such a term $\phi_1(B)$ is unavoidable in multiple MDPs.

E.3.1 Proof of Lemma E.5

Here we give the proof of Lemma E.5. For any positive natural numbers S, A, D, we consider any MDP M with S states, A actions and diameter D. We consider

$$(\underline{s}, \overline{s}) \in \arg\max_{s \neq s' \in \mathcal{S}} \left\{ \min_{\pi \in \Pi} \mathbb{E}\left[\tau_\pi(s \to s')\right] \right\}.$$

We modify the transition structure of M, so that $P(\underline{s}|\overline{s}, a) = 1$ for all actions $a \in \mathcal{A}$. Note that the diameter is not affected by this operation. Throughout, whatever the value of B, we will consider the following sampling requirements: $b(s) \triangleq B\mathbb{1}_{\{s=\overline{s}\}}$. We denote by $s_0 \in \mathcal{S}$ the arbitrary starting state of the learning process.

Consider any learning algorithm $\mathfrak{A}$. We denote by π the (possibly non-stationary) policy that is executed by $\mathfrak{A}$. In virtue of Assumption 7.2, we can naturally (and without loss of generality) restrict our attention to a policy π whose expected hitting time to $\overline{s}$ is finite starting from any state in $\mathcal{S}$ — we denote by $\overline{\mu}_\pi$ such an upper bound. We denote by $T_\pi^{(i)}$ the random time required by policy π to collect the i-th sample at state $\overline{s}$, starting from s_0 if $i = 1$ or from $\underline{s}$ if $2 \leq i \leq B$.

Lemma E.6. *The $(T_\pi^{(i)})_{2 \leq i \leq B}$ are i.i.d. sub-exponential random variables whose expectation satisfies $\mu_\pi \triangleq \mathbb{E}\left[T_\pi^{(i)}\right] \geq D$ for all $2 \leq i \leq B$.*

Proof. Consider the SSP problem with unitary costs, starting state $\underline{s}$ and zero-cost, absorbing terminal state $\overline{s}$. As seen in Chapter 2, Assumption 7.2 and the fact that the costs are all positive guarantee that the optimal value function of this SSP problem is achieved by a stationary deterministic policy. This implies that $\min_{\pi' \in \Pi} \mathbb{E}\left[\tau_{\pi'}(\underline{s} \to \overline{s})\right] \leq \mathbb{E}\left[\tau_\pi(\underline{s} \to \overline{s})\right]$, and thus by definition of D and μ_π, we get the inequality $D \leq \mu_\pi$. There remains to prove the sub-exponential nature

of the random variable T_π. For any $\lambda \in \mathbb{R}$, we have

$$\mathbb{E}\left[e^{\lambda(T_\pi - \mu_\pi)}\right] = e^{-\lambda\mu_\pi}\mathbb{E}\left[\sum_{n=0}^{+\infty}\frac{1}{n!}\lambda^n T_\pi^n\right] = e^{-\lambda\mu_\pi}\sum_{n=0}^{+\infty}\frac{1}{n!}\lambda^n\mathbb{E}\left[T_\pi^n\right] \leq 2e^{-\lambda\mu_\pi}\sum_{n=0}^{+\infty}\frac{1}{n!}n^n(\lambda\overline{\mu}_\pi)^n,$$

where the last inequality comes from Lemma C.2 which can be applied to bound the moments $\mathbb{E}\left[T_\pi^n\right] \leq 2(n\overline{\mu}_\pi)^n$, since the random variable T_π satisfies $\mathbb{E}\left[T_\pi(s \to \overline{s})\right] \leq \overline{\mu}_\pi$ for all $s \in \mathcal{S}$ by definition of $\overline{\mu}_\pi$. From Lemma E.7, the series above converges whenever $|\lambda| < \frac{1}{e\overline{\mu}_\pi}$. This proves that T_π is sub-exponential according to the second condition of Definition E.8. □

Lemma E.7. *The series* $\sum_{n=0}^{+\infty}\frac{n^n}{n!}x^n$ *converges absolutely for all* $|x| < \frac{1}{e}$.

Proof. Introduce the summand of the series $a_n(x) \triangleq \frac{n^n}{n!}x^n$. We then have

$$\frac{a_{n+1}(x)}{a_n(x)} = \frac{n!}{(n+1)!}\frac{(n+1)^{n+1}}{n^n}x = \left(1+\frac{1}{n}\right)^n x \xrightarrow[n\to+\infty]{} ex.$$

Hence, for any $|x| < \frac{1}{e}$, we have $|\frac{a_{n+1}(x)}{a_n(x)}| < 1$, which means from d'Alembert's ratio test that the series converges absolutely. □

Since T_π is sub-exponential, from Definition E.8, there exists a pair (σ_π, θ_π) of finite positive parameters that verifies

$$\mathbb{E}\left[e^{\lambda(T_\pi - \mu_\pi)}\right] \leq e^{\frac{\sigma_\pi^2\lambda^2}{2}} \quad \text{for all } |\lambda| < \frac{1}{\theta_\pi}.$$

We now apply the concentration inequality for sub-exponential random variables stated in Proposition E.9.

$$\forall y > \frac{\sigma_\pi^2}{\theta_\pi}, \quad \mathbb{P}\left(\sum_{i=2}^{B} T_\pi^{(i)} \leq \mu_\pi(B-1) - y\right) \leq \exp\left(-\frac{y}{2\theta_\pi}\right).$$

We now fix the integer

$$W_\pi \triangleq 1 + 2\max\left\{\left\lceil\frac{\theta_\pi}{\mu_\pi}\right\rceil, \left\lceil\frac{\sigma_\pi^2}{\theta_\pi\mu_\pi}\right\rceil\right\}.$$

Consider any total sampling requirement $B \geq W_\pi$. Then setting $y \triangleq \frac{\mu_\pi(B-1)}{2} > \frac{\sigma_\pi^2}{\theta_\pi}$ yields

$$\mathbb{P}\left(\sum_{i=2}^{B} T_\pi^{(i)} \leq \frac{\mu_\pi(B-1)}{2}\right) \leq \exp\left(-\frac{\mu_\pi(B-1)}{4\theta_\pi}\right) \leq \frac{1}{2},$$

since we have $B \geq \frac{4\theta_\pi}{\mu_\pi}\log(2)+1$. This implies that with probability at least $\frac{1}{2}$,

$$\sum_{i=1}^{B} T_\pi^{(i)} \geq \sum_{i=2}^{B} T_\pi^{(i)} > \frac{\mu_\pi(B-1)}{2} \geq \frac{(B-1)D}{2},$$

where the last inequality stems from Lemma E.6. As a result, there exists a finite integer $W_{\pi,\delta}$ (depending on π and the environment at hand) such that, for any total sampling requirement $B \geq W_\pi$, the algorithm $\mathfrak{A}$ that executes policy π verifies

$$\mathbb{P}\left(\mathcal{C}(\mathfrak{A}, B\mathbb{1}_{\{\bar{s}\}}) > \frac{(B-1)D}{2}\right) \geq \frac{1}{2},$$

which gives the proof of Lemma E.5.

We recall here the definition of sub-exponential random variables.

Definition E.8 (Wainwright, 2015)**.** *A random variable X with mean $\mu < +\infty$ is said to be sub-exponential if one of the following equivalent conditions is satisfied:*

1. *(Laplace transform condition) There exists $(\sigma,\theta) \in \mathbb{R}^+ \times \mathbb{R}^{+\star}$ such that, for all $|\lambda| < \frac{1}{\theta}$,*

$$\mathbb{E}\left[e^{\lambda(X-\mu)}\right] \leq e^{\frac{\sigma^2\lambda^2}{2}}.$$

2. *There exists $c_0 > 0$ such that $\mathbb{E}\left[e^{\lambda(X-\mu)}\right] < +\infty$ for all $|\lambda| \leq c_0$.*

For any pair (σ,θ) satisfying condition 1, we write $X \sim \textsc{SubExp}(\sigma,\theta)$.

We finally recall a concentration inequality satisfied by sub-exponential random variables.

Proposition E.9 (Wainwright, 2015)**.** *Let $(X_i)_{1\leq i\leq n}$ be a collection of independent sub-exponential random variables such that for all $i \in [n]$, $X_i \sim \textsc{SubExp}(\sigma_i,\theta_i)$ and $\mu_i \triangleq \mathbb{E}[X_i]$. Set $\sigma \triangleq \sqrt{\frac{\sum_{i=1}^n \sigma_i^2}{n}}$ and $\theta \triangleq \max_{i\in[n]}\{\theta_i\}$. The following concentration inequalities hold for any $t \geq 0$,*

$$\mathbb{P}\left(\sum_{i=1}^n X_i - \sum_{i=1}^n \mu_i \geq t\right) \leq \begin{cases} e^{-\frac{t^2}{2n\sigma^2}} & \text{if } 0 \leq t \leq \frac{\sigma^2}{\theta} \\ e^{-\frac{t}{2\theta}} & \text{if } t > \frac{\sigma^2}{\theta} \end{cases},$$

$$\mathbb{P}\left(\sum_{i=1}^n X_i - \sum_{i=1}^n \mu_i \leq -t\right) \leq \begin{cases} e^{-\frac{t^2}{2n\sigma^2}} & \text{if } 0 \leq t \leq \frac{\sigma^2}{\theta} \\ e^{-\frac{t}{2\theta}} & \text{if } t > \frac{\sigma^2}{\theta} \end{cases}.$$

E.4 GOSPRL Beyond the Communicating Setting

Sections E.3 and 7.3.1 demonstrate that the diameter D and/or the SSP-diameters dictate the performance of a sampling procedure in a communicating environment. Indeed, both the GOSPRL upper bound and the worst-case lower bound contain D and/or D_s as a multiplicative factor w.r.t. the total sampling requirement B. However, in many environments, there may exist some states that are hard to reach, or plainly impossible to reach. In that case, the diameter is prohibitively large and even possibly infinite, thus rendering the sample complexity guarantee of Corollary E.2 vacuous. To circumvent this issue, a desirable property of the algorithm would be the ability to assess online the "feasibility" of the sampling requirements, by discarding states that are indeed too difficult to reach. For ease of exposition, we consider throughout Section E.4 the special case of time- and action-independent sampling requirements $b : \mathcal{S} \to \mathbb{N}$ (as explained in Section E.2.2 the extension to the general case of adaptive action-dependent sampling requirements follows straightforwardly).

Formally, we consider any environment that need not be communicating (i.e., it may not satisfy Assumption 7.2). The learning agent receives as input an integer parameter $L \geq 1$, which acts as a reachability threshold that partitions the state space between the states from which we expect sample collection and those that we categorize as too difficult to reach. Specifically, given a sampling requirement $b : \mathcal{S} \to \mathbb{N}$, the desiderata of the agent is to minimize the time it requires, for each state $s \in \mathcal{S}$, to i) either collect the $b(s)$ samples, ii) or discard the sample collection at state s only if there exists a state (accessible from the starting state) that cannot reach s within L steps in expectation. In other words, we do not allow for samples to be discarded if the state is actually below the reachability threshold L. We introduce the following new definition of the sample complexity.

Definition E.10. *Given a reachability threshold $L \geq 1$, sampling requirements $b : \mathcal{S} \to \mathbb{N}$, starting state $s_0 \in \mathcal{S}$ and a confidence level $\delta \in (0, 1)$, the sample complexity of a learning algorithm $\mathfrak{A}$ is defined as*

$$\mathcal{C}(\mathfrak{A}, b, \delta, L, s_0) \triangleq \min \Big\{ t > 0 : \mathbb{P}\Big(\forall s \in \mathcal{S}_L,\ N_t(s) \geq b(s) \wedge I_{\mathfrak{A}}(t) = 1 \Big) \geq 1 - \delta \Big\},$$

where $\mathcal{S}_L \triangleq \{ s \in \mathcal{S} : \max_{\{y \in \mathcal{S} : D_{s_0 y} < +\infty\}} D_{ys} \leq L \}$ and where $I_{\mathfrak{A}}(t)$ corresponds to a Boolean equal to 1 if the algorithm $\mathfrak{A}$ considers at time t that none of the states that remain to be sampled (if there remains any) belong to $\mathcal{S}_L$.[5]

Algorithm GOSPRL-L. We now propose a simple adaptation of GOSPRL to handle this setting, and call the corresponding algorithm GOSPRL-L since it receives as input a reachability threshold L. We split time in *episodes* indexed by j, where the first episode begins at the first time step

and the j-th episode ends when the j-th desired sample is collected. From Corollary E.2 we know that in a communicating environment with diameter D, there exists an absolute constant $\alpha > 0$ (here we exclude logarithmic terms for ease of exposition) such that with probability at least $1 - \delta$, after any j episodes (i.e., after the j-th desired sample is collected), T_j the (total) time step at the end of the j episodes is upper bounded as follows

$$T_j \leq \alpha j D + \alpha j D^{3/2} S^2 A.$$

The key idea is to run GOSPRL and stop its execution if its total duration at some point exceeds a certain threshold depending on L and the current episode. Specifically, in the j-th episode, this threshold is set to $\Phi(j) \triangleq \alpha j L + \alpha j L^{3/2} S^2 A$. If the accumulated duration never exceeds the threshold, the algorithm is naturally run until all the sampling requirements are met.

Lemma E.11. *Consider any reachability threshold $L \geq 1$, starting state $s_0 \in \mathcal{S}$, confidence level $\delta \in (0, 1)$ and sampling requirements $b : \mathcal{S} \to \mathbb{N}$, with $B = \sum_{s \in \mathcal{S}} b(s)$. Then running the algorithm* GOSPRL-L *in any environment yields a sample complexity that can be upper bounded as*

$$\mathcal{C}(\text{GOSPRL-L}, b, \delta, L, s_0) = \widetilde{O}\Big(BL + L^{3/2} S^2 A\Big).$$

Proof. The result is obtained by performing a *reductio ad absurdum* reasoning. We initially make the assumption $\mathcal{H}$ that for all episodes $j \geq 1$, we have $D_{\mathcal{G}_j} \leq L$, where we recall that $D_{\mathcal{G}_j}$ is the SSP-diameter of the goal states $\mathcal{G}_j$ considered during episode j. The condition that is checked at any time step is whether it is smaller or larger than the threshold $\Phi(j) \triangleq \alpha j L + \alpha j L^{3/2} S^2 A$, where j is the current episode. ***i***) In the first case, the total duration is always smaller (or equal) than its threshold and the algorithm performs J episodes until the sampling requirements are met. Since $J \leq B$ and Φ is an increasing function, the sample complexity is bounded by $\Phi(J) \leq \Phi(B) = \widetilde{O}\left(BL + L^{3/2} S^2 A\right)$. ***ii***) In the second case, there exists an episode $j' \geq 1$ and a time step (during that episode) which is larger than the threshold $\Phi(j')$. This implies that with probability at least $1 - \delta$, assumption $\mathcal{H}$ is wrong. Thus there exists an episode $1 \leq j \leq j'$ such that $D_{\mathcal{G}_j} > L$. Since $\mathcal{G}_{j'} \subset \mathcal{G}_j$, we have $D_{\mathcal{G}_j} \leq D_{\mathcal{G}_{j'}}$, thus $D_{\mathcal{G}_{j'}} > L$, which implies from Lemma E.3 that for all $s \in \mathcal{G}_{j'}$, $D_s > L$. Hence the algorithm can terminate and confidently guarantee that none of the states that remain to be sampled belong to S_L. Given that $j' \leq B$, the sample complexity (in the sense of Definition E.10) is bounded by $\Phi(j') \leq \Phi(B) = \widetilde{O}\left(BL + L^{3/2} S^2 A\right)$. □

The algorithm GOSPRL-L requires no computational overhead w.r.t. GOSPRL, as it simply tracks the total duration of GOSPRL and terminates if it exceeds a threshold depending on L. Under the new appropriate definition of sample complexity of Definition E.10, the dependency

in Corollary E.2 on the possibly very large or infinite diameter D is effectively replaced by the reachability threshold L. A large value of L signifies that the sample collection is required at quite difficult-to-reach states, while a small value of L keeps in check the duration of the sampling procedure.

Narrowing the sample collection to states in $\mathcal{S}_L$ may seem at first glance restrictive. Indeed, the presence of states in which the agent may get stuck could disrupt the learning process. However, assume for instance that we consider the canonical assumption made in episodic RL of a *resetting* environment, i.e., an environment that contains a reset action that brings the agent with probability 1 to a reference starting state s_0 (where here we consider that the reset action can be executed at any time step for simplicity). Then we have that $\{s \in \mathcal{S} : \min_\pi \mathbb{E}\left[\tau_\pi(s_0 \to s)\right] \leq L - 1\} \subseteq \mathcal{S}_L$, which shows that numerous states can effectively belong to the set $\mathcal{S}_L$.

Finally, let us delve into the particular case of a weakly communicating MDP, whose state space $\mathcal{S}$ can be partitioned into two subspaces (Puterman, 2014, Section 8.3.1): a communicating set of states (denoted $\mathcal{S}^{\mathrm{C}}$) with each state in $\mathcal{S}^{\mathrm{C}}$ accessible — with non-zero probability — from any other state in $\mathcal{S}^{\mathrm{C}}$ under some stationary deterministic policy, and a (possibly empty) set of states that are transient under all policies (denoted $\mathcal{S}^{\mathrm{T}}$). The sets $\mathcal{S}^{\mathrm{C}}$ and $\mathcal{S}^{\mathrm{T}}$ form a partition of $\mathcal{S}$, i.e., $\mathcal{S}^{\mathrm{C}} \cap \mathcal{S}^{\mathrm{T}} = \emptyset$ and $\mathcal{S}^{\mathrm{C}} \cup \mathcal{S}^{\mathrm{T}} = \mathcal{S}$. Finally, we denote by $D^{\mathrm{C}} < +\infty$ the diameter of the communicating part of M (i.e., restricted to the set $\mathcal{S}^{\mathrm{C}}$), i.e., $D^{\mathrm{C}} \triangleq \max_{s \neq s' \in \mathcal{S}^{\mathrm{C}}} \min_{\pi \in \Pi} \mathbb{E}\left[\tau_\pi(s \to s')\right] < +\infty$. Assume that the starting state s_0 belongs to $\mathcal{S}^{\mathrm{C}}$. We expect the optimal strategy to perform the sample collection at states in $\mathcal{S}^{\mathrm{C}}$ and discard the sample collection at states in $\mathcal{S}^{\mathrm{T}}$. This is what GOSPRL-L does if we have $\mathcal{S}_L = \mathcal{S}^{\mathrm{C}}$, i.e., whenever $D^{\mathrm{C}} \leq L$. Hence, in that setting, the optimal (yet critically unknown) value of the threshold L would be D^{C}.

E.5 Application: Model Estimation (ModEst)

In this section we demonstrate that GOSPRL can be readily applied to tackle the ModEst problem, as well as a "robust" variant called RModEst, both of which are defined as follows. The agent $\mathfrak{A}$ interacts with the environment and, after t time steps, it must return an estimate $\widehat{P}_{\mathfrak{A},t}$ of the transition dynamics, which naturally corresponds to the empirical average of the transition probabilities. The accuracy of the estimate and the corresponding sample complexity are evaluated as follows.

Definition E.12. *Given an accuracy level $\eta > 0$ and a confidence level $\delta \in (0, 1)$, the* ModEst *and* RModEst *sample complexity of an online learning algorithm $\mathfrak{A}$ are defined as*

$$\mathcal{C}_{\textsc{ModEst}}(\mathfrak{A}, \eta, \delta) \triangleq \min\left\{t > 0 : \mathbb{P}\big(\forall (s,a) \in \mathcal{S} \times \mathcal{A},\ \|\widehat{P}_{\mathfrak{A},t}(\cdot|s,a) - P(\cdot|s,a)\|_1 \leq \eta\big) \geq 1 - \delta\right\},$$

$$\mathcal{C}_{\text{RMODEST}}(\mathfrak{A}, \eta, \delta) \triangleq \min\left\{t > 0 : \mathbb{P}\big(\forall (s', s, a) \in \mathcal{S}^2 \times \mathcal{A},\ |\widehat{P}_{\mathfrak{A},t}(s'|s,a) - P(s'|s,a)| \leq \eta\big) \geq 1 - \delta\right\},$$

where $\widehat{P}_{\mathfrak{A},t}$ is the estimate (i.e., empirical average) of the transition dynamics P after t time steps.

We have the following sample complexity guarantees.

Lemma E.13. *Instantiating* GOSPRL *with two different sequences of sampling requirements yields respectively*

$$\mathcal{C}_{\text{RMODEST}}(\text{GOSPRL}, \eta, \delta) = \widetilde{O}\Big(\frac{DSA}{\eta^2} + D^{3/2}S^2A\Big),$$
$$\mathcal{C}_{\text{MODEST}}(\text{GOSPRL}, \eta, \delta) = \widetilde{O}\Big(\frac{D\Gamma SA}{\eta^2} + \frac{DS^2A}{\eta} + D^{3/2}S^2A\Big).$$

Proof. We first focus on the RMODEST objective with desired accuracy level η. From Definition E.12, we would like that, for any state-action pair (s, a) and next state s', the following condition holds:

$$|\widehat{P}_t(s'|s,a) - P(s'|s,a)| \leq \eta. \tag{E.10}$$

From the empirical Bernstein inequality (see e.g., Audibert et al., 2009; Fruit et al., 2020), we have with probability at least $1-\delta$, for any time step $t \geq 1$ and for any state-action pair (s, a) and next state s',

$$|\widehat{P}_t(s'|s,a) - P(s'|s,a)| \leq 2\sqrt{\frac{\widehat{\sigma}_t^2(s'|s,a)}{N_t^+(s,a)}\log\left(\frac{2SAN_t^+(s,a)}{\delta}\right)} + \frac{6\log\left(\frac{2SAN_t^+(s,a)}{\delta}\right)}{N_t^+(s,a)}, \tag{E.11}$$

where $N_t^+(s,a) \triangleq \max\{1, N_t(s,a)\}$ and where the $\widehat{\sigma}_t^2$ are the population variance of transitions, i.e., $\widehat{\sigma}_t^2(s'|s,a) \triangleq \widehat{P}_t(s'|s,a)(1 - \widehat{P}_t(s'|s,a))$. Let us now define, for any $X, Y \geq 0$, the quantity

$$\Phi(X,Y) \triangleq \left\lceil \frac{57X^2}{\eta^2}\left[\log\left(\frac{8eX\sqrt{2SA}}{\sqrt{\delta}\eta}\right)\right]^2 + \frac{24Y}{\eta}\log\left(\frac{24YSA}{\delta\eta}\right)\right\rceil.$$

Using a technical lemma (Lemma E.14), we can prove that condition (E.10) holds whenever the number of samples at the pair (s, a) becomes at least equal to

$$\phi_t^{\text{RMODEST}}(s,a) \triangleq \Phi\left(X, Y\right), \qquad X \triangleq \max_{s' \in \mathcal{S}} \sqrt{\widehat{\sigma}_t^2(s'|s,a)}, \qquad Y \triangleq 1.$$

We thus execute GOSPRL until there exists a time step $t \geq 1$ such that $b_t(s,a) \triangleq \phi_t^{\text{RModEst}}(s,a)$ samples have been collected at each state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$. Although the sampling requirement b_t depends on the time step t, this is not an issue from Section 7.3.2 since for any $s \in \mathcal{S}$ and $t \geq 1$, $b_t(s,a)$ is bounded from above due to the fact that $\widehat{\sigma}_t^2(s'|s,a) \leq \frac{1}{4}$. This means that the total requirement for RModEst is $B_{\text{RModEst}} = \widetilde{O}\left(SA/\eta^2\right)$, which yields the first bound of Lemma E.13.

We now turn to the ModEst objective. GOSPRL collects samples until there exists a time step t such that the number of samples at each pair (s,a) is at least equal to

$$\phi_t^{\text{ModEst}}(s,a) \triangleq \Phi\left(X, Y\right), \qquad X \triangleq \sum_{s' \in \mathcal{S}} \sqrt{\widehat{\sigma}_t^2(s'|s,a)}, \qquad Y \triangleq S.$$

Introducing $\Gamma(s,a) \triangleq \|P(\cdot|s,a)\|_0$ the maximal support of $P(\cdot|s,a)$, we use the following inequality (valid at any time step $t \geq 1$): $\sum_{s' \in \mathcal{S}} \widehat{\sigma}_t(s'|s,a) \leq \sqrt{\Gamma(s,a) - 1}$ (see e.g., Fruit et al., 2020, Lemma 4). This means that the total requirement for ModEst is $B_{\text{ModEst}} = \widetilde{O}\left(\frac{\sum_{s,a} \Gamma(s,a)}{\eta^2} + \frac{S^2 A}{\eta}\right)$. Plugging in the result of Corollary E.2 finally yields the second bound of Lemma E.13 (which corresponds to the statement of Lemma 7.9 in Section 7.4.2). □

Lemma E.14. *For any $x \geq 2$ and $a_1, a_2, a_3, a_4 > 0$ such that $a_3 x \leq a_1 \sqrt{x} \log(a_2 x) + a_4 \log(a_2 x)$, the following holds*

$$x \leq \frac{4a_4}{a_3} \log\left(\frac{2a_4 a_2}{a_3}\right) + \frac{128 a_1^2}{9 a_3^2} \left[\log\left(\frac{4a_1\sqrt{a_2}e}{a_3}\right)\right]^2.$$

Proof. Assume that $a_3 x \leq a_1 \sqrt{x} \log(a_2 x) + a_4 \log(a_2 x)$. Then we have that $\frac{a_3}{2} x \leq -\frac{a_3}{2} x + a_1 \sqrt{x} \log(a_2 x) + a_4 \log(a_2 x)$. From Lemma E.15 we have

$$-\frac{a_3}{2} x + a_1 \sqrt{x} \log(a_2 x) \leq \underbrace{\frac{32 a_1^2}{9 a_3} \left[\log\left(\frac{4a_1\sqrt{a_2}e}{a_3}\right)\right]^2}_{\triangleq a_0}.$$

Thus we have $x \leq \frac{2a_4}{a_3} \log(a_2 x) + \frac{2a_0}{a_3}$ and we conclude the proof using Lemma E.16. □

Lemma E.15 (Kazerouni et al., 2017, Lemma 8)**.** *For any $x \geq 2$ and $a_1, a_2, a_3 > 0$, the following holds*

$$-a_3 x + a_1 \sqrt{x} \log(a_2 x) \leq \frac{16 a_1^2}{9 a_3} \left[\log\left(\frac{2a_1\sqrt{a_2}e}{a_3}\right)\right]^2.$$

Lemma E.16. *Let b_1, b_2 and b_3 be three positive constants such that $\log(b_1 b_2) \geq 1$. Then any $x > 0$ satisfying $x \leq b_1 \log(b_2 x) + b_3$ also satisfies $x \leq 2b_1 \log(2b_1 b_2) + 2b_3$.*

Proof. Assume that $x \leq b_1 \log(b_2 x) + b_3$ and set $y = x - b_3$. If $y \leq b_3$, then we have $x \leq 2b_3$. Otherwise, we can write $y \leq b_1 \log(b_2 y + b_2 b_3) \leq b_1 \log(2b_2 y)$. From Lemma E.17 we have $y \leq 2b_1 \log(2b_1 b_2)$, which concludes the proof. □

Lemma E.17 (Kazerouni et al., 2017, Lemma 9)**.** *Let b_1 and b_2 be two positive constants such that $\log(b_1 b_2) \geq 1$. Then any $x > 0$ satisfying $x \leq b_1 \log(b_2 x)$ also satisfies $x \leq 2b_1 \log(b_1 b_2)$.*

E.6 Application: Sparse Reward Discovery (Treasure Problem)

In this section, we focus on the canonical sampling requirement of the Treasure problem of Section 7.4.1, where each state-action pair must be visited at least once. We illustrate how direct adaptations of existing algorithms are not able to match the guarantees of GOSPRL in Lemma 7.7.

Discussion on finite-horizon or discounted PAC-MDP algorithms. At first glance, an approach to tackle the Treasure problem could be to consider a well-known PAC exploration algorithm such as RMax (Brafman and Tennenholtz, 2002) (the same discussion holds for E3 of Kearns and Singh, 2002). In particular, we can examine the ZeroRMax variant proposed by Jin et al. (2020). Indeed the demarcation between known states and unknown states is an algorithmic principle related to the problem at hand: a state is considered known when the number of times each action has been executed at that state is at least m for a suitably chosen m and its reward is set to 0, while an unknown state receives a reward of 1. The set of known states captures what has been sufficiently sampled (and the empirical estimate of the transitions is used), while the set of unknown states drives exploration to collect additional samples. The central concept for analyzing the sample complexity of the algorithm is the escape probability (i.e., the probability of visiting the unknown states), which, in the case of $m = 1$, would amount exactly to the probability of collecting a required sample in the Treasure problem. However, despite the similarities, ZeroRMax (as well as RF-RL-Explore of Jin et al., 2020) are designed in the infinite-horizon discounted setting or the finite-horizon setting. As such, only a finite number of steps is relevant, and the episode lengths (and resulting sample complexity) directly depend on the discount factor γ or on the horizon H, respectively. Such approach cannot be

employed in the setting of communicating MDPs, where there is no known imposed horizon of the problem, and where the agent must interweave the policy planning and policy execution processes by defining algorithmic episodes. As such, despite bearing high-level similarity with GOSPRL at an algorithmic level, such finite-horizon (or discounted) guarantees cannot be translated to sample complexity for the TREASURE problem.

Leveraging UCRL. We now analyze UCRL2 (Jaksch et al., 2010), an efficient algorithm for reward-dependant exploration in the infinite-horizon undiscounted setting. In order to tackle the TREASURE problem, a first approach could be to consider true rewards of zero everywhere while the uncertainty around the rewards remains, i.e., the algorithm observes as reward $r(s,a) \sim \sqrt{\frac{1}{N^+(s,a)}}$, which corresponds to the usual uncertainty on the rewards (Jaksch et al., 2010), with $N(s,a)$ denoting the number of visits of (s,a) so far. The underlying idea is that as the algorithm visits a state-action pair, its observed reward will decrease, thus favoring the visitation of non-sampled state-action pairs. Yet while this algorithm is fairly intuitive, it appears tricky to directly leverage the analysis of UCRL2 to obtain a guarantee on the time the algorithm requires to solve the TREASURE problem. Indeed, the inspection of the tools used in the regret derivation of UCRL2 does not point out to a step in the analysis which explicitly lower bounds state-action visitations.

Another possibility is to design a non-stationary reward signal to feed to UCRL2. Namely, assigning a reward of 1 if the state is under-sampled and 0 otherwise, corresponds to a sensible strategy (note that this reward signal changes according to the behavior of the algorithm). Yet as explained in Section 3.2, for any SSP problem with unit costs, the SSP-regret bound that is obtained from the analysis of average-reward techniques (by assigning a reward of 1 at the goal state, and 0 everywhere else) is worse than that obtained from the analysis of SSP goal-oriented techniques. This difference directly translates into a worse performance of UCRL2-based approaches for the TREASURE problem. Indeed, retracing the analysis of Section B.1, we obtain that $\widetilde{O}(D_s^3 S^2 A)$ time steps are required to collect a sought-after sample when running the algorithm UCRL2B (Fruit et al., 2020) (which is a variant of UCRL2 that constructs confidence intervals based on the empirical Bernstein inequality rather than Hoeffding's inequality and thus yields tighter regret guarantees). Since the analysis renders the re-use of samples difficult, performing this reasoning for each sought-after state to sample yields a total TREASURE sample complexity of $\widetilde{O}\left(\sum_{s\in\mathcal{S}} D_s^3 S^2 A\right)$, which is always worse than the bound in Lemma 7.7 since $\max_s D_s = D$.

Leveraging MAXENT. At first glance, an alternative and natural approach to visit each state-action pair at least once may be to optimize the MAXENT objective over the state-action space,

i.e., maximize the entropy function H over the stationary state-action distributions $\lambda \in \Lambda$,

$$H(\lambda) \triangleq \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} -\lambda(s,a)\log(\lambda(s,a)).$$

This objective — over the state space, yet the extension to the state-action space is straightforward — was studied by Hazan et al. (2019) in the infinite-horizon discounted setting and by Cheung (2019) in the infinite-horizon undiscounted setting. Following the latter, there exists a learning algorithm such that, with overwhelming probability,

$$H(\lambda^\star) - H(\widetilde{\lambda}_t) = \widetilde{O}\left(\frac{DS^{1/3}}{t^{1/3}} + \frac{DS\sqrt{A}}{\sqrt{t}}\right), \tag{E.12}$$

where $\lambda^\star \in \arg\max_{\lambda\in\Lambda} H(\lambda)$ and $\widetilde{\lambda}_t$ is the empirical state-action frequency at time t, i.e., $\widetilde{\lambda}_t(s,a) = \frac{N_t(s,a)}{t}$. The Treasure sample complexity translates into the first time step $t \geq 1$ such that $\widetilde{\lambda}_t(s,a) \geq \frac{1}{t}$ for all $(s,a) \in \mathcal{S}\times\mathcal{A}$. However, the state-action entropy H corresponds to the sum of a function related to each state-action frequency, and maximizing it provides no guarantee on each summand, i.e., on each state-action frequency. Indeed, assume that there exists a time t such that $\widetilde{\lambda}_t(s,a) \geq \frac{1}{t}$ for all $(s,a) \in \mathcal{S}\times\mathcal{A}$. This implies that $H(\widetilde{\lambda}_t) \geq \frac{SA}{t}\log(t)$. However, the regret bound of Equation (E.12) cannot be leveraged to show that t must necessarily be small enough. Overall, it seems that directly optimizing MaxEnt is unfruitful in guaranteeing the visitation of each state-action pair at least once, and thus in provably enforcing the Treasure objective.

Instead of maximizing MaxEnt, the discussion above encourages us to optimize the "worst-case" summand of the entropy function, by maximizing over Λ the following function

$$F(\lambda) \triangleq \min_{(s,a)\in\mathcal{S}\times\mathcal{A}} \lambda(s,a).$$

It is straightforward to show that F is concave in λ (as the minimum of $S\times A$ concave functions), as well as 1-Lipschitz-continuous w.r.t. the Euclidean norm $\|\cdot\|_2$, i.e.,

$$\forall(\lambda,\lambda') \in \Lambda^2, |F(\lambda) - F(\lambda')| \leq \|\lambda - \lambda'\|_\infty \leq \|\lambda - \lambda'\|_2.$$

However, F is a non-smooth function, therefore the Frank-Wolfe algorithmic design of Hazan et al. (2019) and Cheung (2019) cannot be leveraged. Instead, we propose to use the mirror descent algorithmic design of Cheung (2019, Section 5) that can handle general concave functions. It guarantees that there exists a constant $\beta > 0$ such that, with overwhelming probability (here we exclude logarithmic terms for ease of exposition)

$$F(\lambda^\star) - F(\widetilde{\lambda}_t) \leq \frac{\beta D}{t^{1/3}} + \frac{\beta DS\sqrt{A}}{\sqrt{t}}.$$

Introduce $\omega^\star \triangleq F(\lambda^\star) = \min_{s,a} \lambda^\star(s,a) \in (0, \frac{1}{SA}]$. We then have

$$F(\widetilde{\lambda}_t) \geq \omega^\star - \frac{\beta D}{t^{1/3}} - \frac{\beta D S \sqrt{A}}{\sqrt{t}}. \tag{E.13}$$

Equipped with Equation (E.13), we can easily prove that if

$$t = \Omega\left(\min\left\{\frac{D^2 S^2 A}{(\omega^\star)^2}, \frac{D^3}{(\omega^\star)^3}\right\}\right), \tag{E.14}$$

then $F(\widetilde{\lambda}_t) \geq \frac{1}{t}$, which immediately implies that the Treasure is discovered. This sample complexity result is quite poor compared to Lemma 7.7. In particular, it depends polynomially on $(\omega^\star)^{-1}$, which cannot be smaller than SA.

E.7 Application: Goal-Free Cost-Free Exploration in Communicating MDPs

E.7.1 Reward-Free Exploration in Finite-Horizon MDPs vs. Cost-Free Exploration in Goal-Conditioned RL

Jin et al. (2020) introduced the reward-free framework in the finite-horizon case, which we recall is a special case of a goal-oriented (i.e., SSP) problem where each episode terminates after exactly H steps. The agent receives as input an accuracy level $\varepsilon > 0$, a confidence level $\delta \in (0,1)$, the state and action spaces, and the horizon H, while no knowledge is provided about the transition model P. The learning process is decomposed into two phases. ① *Exploration phase:* The agent first collects trajectories from the MDP without a pre-specified reward function and returns an estimate of the transition model $\widehat{P}$. ② *Planning phase:* The agent receives an arbitrary reward function and is tasked with computing an ε-optimal policy with probability at least $1-\delta$, without any additional interaction with the environment. The objective is to minimize the duration of the exploration phase needed to simultaneously enforce any requested planning guarantee.

In Jin et al. (2020) the reward-free exploration problem is studied for any arbitrary MDP, where there may exist states that are difficult or impossible to reach. The core mechanism in their analysis is to partition the states depending on their ease of being reached within H steps. Specifically, they distinguish between *significant* states, that can be sufficiently visited and whose transition probability can thus be accurately estimated, and *insignificant* states that are too difficult to reach within H steps, but therefore have negligible contribution to any reward optimization.

Interestingly, in the goal-conditioned setting this distinction may no longer be meaningful. By way of illustration, consider any fixed horizon H and the toy environment in Figure E.1. Suppose that the objective is to quickly reach state z (i.e., the goal state is z, the starting state is x and all costs are equal to 1). Even though state y is *insignificant* within H steps (in the finite-horizon sense of Jin et al., 2020, for any positive "significance level"), it is actually crucial in solving the objective, as z can be reached deterministically in 1 step from y. Extrapolating this scenario, in the goal-conditioned setting, we may have an effective horizon of $H = +\infty$ for some goals, which implies that the transition model P must be accurately estimated across the *entire* state-action space to ensure that a near-optimal goal-conditioned policy can be computed.

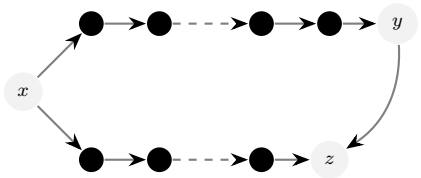


Figure E.1 – The agent starts at state x and reaches z in H steps with probability $1/2$, and y in $H+1$ steps with probability $1/2$. From state y the agent deterministically transitions to state z in 1 step.

Hence the challenges that emerge in the cost-free exploration problem in goal-conditioned RL are orthogonal to the ones in finite-horizon (Jin et al., 2020): a *constraint on the environment is added* (all states must now be reachable, Assumption 7.2), allowing the *removal of the constraint on performance* (which is not limited to H steps anymore) and thus enabling to tackle the more general class of goal-oriented problems.

For a designated goal state $g \in \mathcal{S}$, recall from Part I that the SSP objective is to compute a policy $\pi : \mathcal{S} \to \mathcal{A}$ minimizing the cumulative cost before reaching g. Formally, the (possibly unbounded) SSP value function is defined as

$$V_\pi(s \to g) \triangleq \mathbb{E}\Big[\sum_{t=1}^{\tau_\pi(s\to g)} c(s_t, \pi(s_t)) \;\big|\; s_1 = s\Big],$$

where $\tau_\pi(s \to g) \triangleq \inf\{t \geq 0 : s_{t+1} = g \,|\, s_1 = s, \pi\}$ is the (random) number of steps needed to reach g from s when executing policy π. An optimal policy (if it exists) is denoted by $\pi^\star \in \arg\min_\pi V_\pi(s \to g)$.

Without loss of generality, we consider throughout that the maximum $c_{\max}$ of the cost functions that we intend to consider in the planning phase is equal to 1. On the other hand, the minimum value $c_{\min}$ has a more subtle impact on the type of performance guarantees we can obtain. For any cost function c and any pair of initial and goal states s and g, we introduce a slack parameter $\theta \in [1, +\infty]$ and we say that a policy $\widehat{\pi}$ is (ε, θ)-optimal if [6]

$$V^{\widehat{\pi}}(s \to g) \leq \min_{\pi : \mathbb{E}[\tau_\pi(s\to g)] \leq \theta D_{s,g}} V^\pi(s \to g) + \varepsilon. \tag{E.15}$$

[6] This reduces to standard ε-optimality for $\theta \to \infty$.

We consider this restricted optimality only in the general cost case of $c_{\min} = 0$, where the $(\varepsilon, +\infty)$-optimal policy may not be proper (as seen in Chapter 2). In that case, we are interested in finding the best proper policy, which is what the restricted optimality in Equation (E.15) enables as it constrains the targeted policy to be proper. This consideration is required when translating the performance from the cost-perturbed MDP to the original MDP, which needs constraining the expected goal-reaching time of the targeted policy.

We are now ready to formally define the *goal-free cost-free exploration* problem. It is characterized by an accuracy level $0 < \varepsilon \leq 1$, a confidence level $\delta \in (0, 1)$, a minimum cost $c_{\min} \in [0, 1]$ and a slack parameter $\theta \in [1, +\infty]$ (and we allow either $c_{\min} = 0$ or $\theta = +\infty$, but not both simultaneously). After its exploration phase (whose number of time steps defines the sample complexity of the problem), the agent is expected to be able to compute, with probability at least $1 - \delta$, an (ε, θ)-optimal goal-conditioned policy $\hat{\pi}$ for *any* goal state $g \in \mathcal{S}$ and *any* cost function $c \in [c_{\min}, 1]$, i.e., satisfying Equation (E.15) for all $s \in \mathcal{S}$.

E.7.2 Proof of Lemma 7.10

We show that instantiating GOSPRL for carefully selected sampling requirements $b_t(s, a)$ enables to obtain the guarantee of Lemma 7.10. To do so, we build on the sample complexity analysis of solving a fixed-goal SSP problem with a generative model that we derive in Tarbouriech et al. (2021b). Specifically, we introduce the following sampling requirement function

$$\phi(X, y) \triangleq \alpha \cdot \left(\frac{X^3 \widehat{\Gamma}}{y\varepsilon^2} \log\left(\frac{XSA}{y\varepsilon\delta}\right) + \frac{X^2 S}{y\varepsilon} \log\left(\frac{XSA}{y\varepsilon\delta}\right) + \frac{X^2 \widehat{\Gamma}}{y^2} \log^2\left(\frac{XSA}{y\delta}\right) \right), \tag{E.16}$$

where $\alpha > 0$ is a numerical constant and $\widehat{\Gamma} \triangleq \max_{s,a} \|\widehat{P}(\cdot|s,a)\|_0 \leq \Gamma$ is the largest support of $\widehat{P}$. The sampling requirement function of Equation (E.16) instantiated for specific values of X and y is used to guide the GOSPRL algorithm. Specifically, the analysis distinguishes between two cases: *either* $c_{\min} > 0$ and the cost function considered in the planning phase can be the same as the original one, *or* $c_{\min} = 0$ and all costs incur an additive perturbation of $\varepsilon/(\theta D) > 0$. As stated in Section 7.4.3, we set $\omega \triangleq \max\{c_{\min}, \varepsilon/(\theta D)\}$, which is guaranteed to be positive since we enforce either $c_{\min} = 0$ or $\theta = +\infty$, but not both simultaneously. As such, in Equation (E.16) we define $y \triangleq \omega$ to be equal to the minimum cost of either the true or the perturbed cost function. As for the value of X, we perform the following distinction of cases.

① First let us assume that the learning agent has prior knowledge of the diameter D of the MDP. Then we set $X \triangleq D$. Since our sample complexity analysis of SSP with a generative model in Tarbouriech et al. (2021b) accurately estimates the transition kernel and thus holds for arbitrary cost function in $[\omega, 1]$, we can ensure that collecting at least $\phi(D, \omega)$ samples from each state-action pair provides the ε-optimality cost-free planning guarantee of Lemma 7.10.

The total time required to collect such samples is upper bounded by $DSA\phi(D, \omega^{-1})$, which directly yields the sample complexity guarantee stated in Lemma 7.10.

② Second we show that we can relax the assumption of knowing the diameter D without altering the sample complexity guarantee. To do so, we begin the algorithm by a procedure which computes a quantity $\widehat{D}$ such that $D \leq \widehat{D} \leq D(1+\varepsilon)$ with high probability. From Section E.8.1, this can be done in $\widetilde{O}(D^3S^2A/\varepsilon^2)$ time steps by leveraging GOSPRL. We thus begin the algorithm by running such diameter-estimation subroutine. Crucially, we note that its sample complexity is subsumed in the total sample complexity of Lemma 7.10. Then we simply apply the reasoning in case ① by considering $X \triangleq \widehat{D}$ in the allocation of Equation (E.16) instead of $X = D$. Since $\widehat{D}$ is a sufficiently tight upper bound on D (i.e., $\widehat{D} = O(D)$), we ultimately obtain the same sample complexity guarantee as in case ①.

E.8 Other Applications

In this section, we provide additional applications where GOSPRL can be leveraged to readily obtain an online learning algorithm. We first summarize them here.

Diameter estimation (see Section E.8.1). GOSPRL can be leveraged to estimate the MDP diameter D. In Section E.8.1 we develop a GOSPRL-based procedure that computes an estimate $\widehat{D}$ such that $D \leq \widehat{D} \leq (1+\varepsilon)D$ in $\widetilde{O}(D^3S^2A/\varepsilon^2)$ time steps. This improves on the diameter estimation procedure recently devised by Zhang and Ji (2019) by a multiplicative factor of DS^2. As $\widehat{D}$ provides an upper bound on the optimal bias span $sp(h^\star)$, our procedure may be of independent interest for initializing average-reward regret-minimization algorithms that leverage prior knowledge of $sp(h^\star)$ (as done by e.g., Zhang and Ji, 2019).

PAC-policy learning (see Section E.8.2). One of the most common $\mathcal{SO}$-based settings is the computation of an ε-optimal policy via sample-based value iteration. Since GOSPRL is agnostic to how the sampling requirements are generated, we can easily integrate it with any state-of-the-art $\mathcal{SO}$-based algorithm and directly inherit its properties. For instance, in Section E.8.2 we show that GOSPRL can be easily combined with Bespoke (Zanette et al., 2019) to obtain a competitive online learning algorithm for the policy learning problem. In fact, the sample complexity of the resulting algorithm is only a factor D worse than existing online learning algorithms in the worst case and, leveraging the refined problem-dependent bounds of Bespoke, it is likely to be superior in many MDPs.

Bridging bandits and MDPs with GOSPRL (see Section E.8.3). In multi-armed bandit (MAB) an agent directly collects samples by pulling arms. If we map each arm to a state-action pair, we can see any MAB algorithm as having access to an $\mathcal{SO}$. As such, we can readily turn any bandit algorithm into an RL online linear algorithm by calling GOSPRL to generate the samples needed by the MAB algorithm. Exploiting this procedure, in Section E.8.3 we show

how we can tackle problems such as *best-state identification* and *active exploration* (i.e., state-signal estimation) in the communicating MDP setting, for which no specific online learning algorithm exists yet.

E.8.1 Application: Diameter Estimation

GOSPRL can be leveraged to estimate the diameter D which is a quantity of interest in the average-reward setting. Indeed, D dictates the performance of reward-based no-regret algorithms (Jaksch et al., 2010), and some works assume that an upper bound on the optimal bias span $sp(h^\star)$ is known (e.g., Qian et al., 2019). Since we have $sp(h^\star) \leq r_{\max} D$ (e.g., Bartlett and Tewari, 2009), upper bounding D enables to relax this assumption. Recently, for such purpose of upper bounding $sp(h^\star)$, (Zhang and Ji, 2019) developed an initial procedure based on successive applications of UCRL2 that can compute an estimate $\widehat{D}$ such that $D \leq \widehat{D} \leq (1+\varepsilon)D$ in $\widetilde{O}(D^4 S^4 A/\varepsilon^2)$ time steps (see Zhang and Ji, 2019, Appendix D & Alg. 3 "LD: Learn the Diameter"). In Algorithm E.1 we derive an iterative estimation procedure based on GOSPRL which can compute such upper bound of D faster, namely in $\widetilde{O}(D^3 S^2 A/\varepsilon^2)$ time steps, while simultaneously providing an accurate estimation of the transition dynamics. As such it may be an initial procedure of independent interest for regret-minimization algorithms in the average-reward setting. We define a notation used throughout the section, $\|U\|_\infty^\infty \triangleq \max_{s,s'} U(s \to s')$, which holds for any quantity U that can be naturally mapped to a $\mathcal{S} \times \mathcal{S}$ matrix.

Lemma E.18. *With probability at least* $1-\delta$*, Algorithm E.1:*

- *has a sample complexity bounded by* $\widetilde{O}\left(D^3 S^2 A/\varepsilon^2\right)$,
- *requires at most* $\log_2\left(D(1+\varepsilon)\right)+1$ *inner iterations,*
- *solves the* MODEST *problem for an accuracy level* $\eta > 0$ *and outputs an optimistic* $\mathcal{S} \times \mathcal{S}$ *matrix* $\widetilde{v}$ *such that* $\frac{\varepsilon}{2D} \leq \eta \leq \frac{\varepsilon}{\|\widetilde{v}\|_\infty^\infty}$,
- *outputs a quantity* $\widehat{D} \triangleq \left(1+2\eta\|\widetilde{v}\|_\infty^\infty\right)\|\widetilde{v}\|_\infty^\infty$ *that verifies* $D \leq \widehat{D} \leq \left(1+2\varepsilon(1+\varepsilon)\right)(1+\varepsilon)D$.

Proof. We will assume throughout that the event $\mathcal{E}$ (defined in Section E.1) holds. We now give a useful statement stemming from optimism:

"At any stage of Algorithm E.1, for any given goal state, denote by $\widetilde{v}$ the vector computed using EVI for SSP. Then under the event $\mathcal{E}$, we have component-wise (i.e., starting from any non-goal state): $\widetilde{v} \leq \min_\pi V_p^\pi \leq D$."

To prove this useful statement, we observe that the first inequality stems from Lemma E.1 of Section E.1 while the second inequality uses the definition of the diameter D and the fact that the considered costs are equal to 1.

Algorithm E.1: GOSPRL-based procedure to estimate the diameter

Input: accuracy $\varepsilon > 0$, confidence level $\delta \in (0,1)$.
Set $W \triangleq \frac{1}{2}$ and $\|\widetilde{v}\|_\infty^\infty \triangleq 1$.
while $\|\widetilde{v}\|_\infty^\infty > W$ **do**
  Set $W \leftarrow 2W$.
  Set the accuracy $\eta \triangleq \frac{\varepsilon}{W}$.
  Collect additional samples by running GOSPRL for the ModEst problem with accuracy $\frac{\eta}{2}$ and confidence level δ.
  for each state $s \in \mathcal{S}$ **do**
    Compute a vector $\widetilde{v}(\cdot \to s)$ using EVI for SSP, with goal state s, unit costs and VI precision $\mu_{\text{VI}} \triangleq \frac{\min\{1,\varepsilon\}}{2}$ (see Section E.1).
Output: the quantity $\widehat{D} \triangleq (1 + 2\eta\|\widetilde{v}\|_\infty^\infty)\,\|\widetilde{v}\|_\infty^\infty$.

Now, denote by n the iteration index of the Algorithm E.1 (starting at $n = 1$), so that $W_n = 2^n$. Introduce $N \triangleq \min\{n \geq 1 : \|\widetilde{v}_n\|_\infty^\infty \leq W_n\}$. We have $\|\widetilde{v}_n\|_\infty^\infty \leq D$ at any iteration $n \geq 1$ from the useful statement on optimism above. Since $(W_n)_{n\geq 1}$ is a strictly increasing sequence, Algorithm E.1 is bound to end in a finite number of iterations (i.e., $N < +\infty$), and given that $W_{N-1} \leq \|\widetilde{v}_{N-1}\|_\infty^\infty \leq D$, we get $N \leq \log_2(D) + 1$. Moreover, we have $\|\widetilde{v}_N\|_\infty^\infty \leq W_N$ and $\eta_N = \frac{\varepsilon}{W_N}$, which implies that $\eta_N \leq \frac{\varepsilon}{\|\widetilde{v}_N\|_\infty^\infty}$. Moreover, combining $W_{N-1} \leq D$ and $W_{N-1} = \frac{W_N}{2} = \frac{\varepsilon}{2\eta_N}$ yields that $\frac{\varepsilon}{2D} \leq \eta_N$.

Denote by $\eta \triangleq \eta_N$ the achieved ModEst accuracy at the end of Algorithm E.1. Plugging in the guarantee of Prop. 7.9 yields a sample complexity of

$$\widetilde{O}\Big(\frac{DS^2A}{\eta^2}\Big) = \widetilde{O}\Big(\frac{D^3S^2A}{\varepsilon^2}\Big).$$

Denote by $\widetilde{v} \triangleq \widetilde{v}_N$ the optimistic matrix output by Algorithm E.1. Consider the pair of states $(s_1, s_2) \in \arg\max_{(s,s')} \min_\pi \mathbb{E}\left[\tau_\pi(s \to s')\right]$. Denote by $\widetilde{\pi}$ the greedy policy w.r.t. the vector $\widetilde{v}(\cdot \to s_2)$ in the optimistic model with goal state s_2. Then we have

$$\begin{aligned} D &= \min_\pi \mathbb{E}\left[\tau_\pi(s_1 \to s_2)\right] \mathbb{E}\left[\tau_{\widetilde{\pi}}(s_1 \to s_2)\right] \overset{\text{(a)}}{\leq} (1 + 2\eta\|\mathbb{E}\left[\widetilde{\tau}_{\widetilde{\pi}}\right]\|_\infty^\infty)\, \mathbb{E}\left[\widetilde{\tau}_{\widetilde{\pi}}(s_1 \to s_2)\right] \\ &\overset{\text{(b)}}{\leq} (1 + 2\eta(1+\varepsilon)\|\widetilde{v}\|_\infty^\infty)\,(1+\varepsilon)\widetilde{v}(s_1 \to s_2) \leq (1 + 2\eta(1+\varepsilon)\|\widetilde{v}\|_\infty^\infty)\,(1+\varepsilon)\|\widetilde{v}\|_\infty^\infty \triangleq \widehat{D} \\ &\overset{\text{(c)}}{\leq} (1 + 2\eta(1+\varepsilon)\|\widetilde{v}\|_\infty^\infty)\,(1+\varepsilon)D \overset{\text{(d)}}{\leq} (1 + 2\varepsilon(1+\varepsilon))\,(1+\varepsilon)D, \end{aligned}$$

where (a) corresponds to the SSP simulation lemma (see Lemma 2.14) given that a ModEst accuracy of η is fulfilled, (b) comes from the value iteration precision $\mu_{\text{VI}} \triangleq \frac{\min\{1,\varepsilon\}}{2}$ which implies that $\mathbb{E}\left[\widetilde{\tau}_{\widetilde{\pi}}\right] \leq (1 + 2\mu_{\text{VI}})\widetilde{v} \leq (1+\varepsilon)\widetilde{v}$ component-wise according to Lemma E.1, (c) is implied by the useful statement on optimism given at the beginning of the proof, and finally (d) leverages that $\eta\|\widetilde{v}\|_\infty^\infty \leq \varepsilon$. □

E.8.2 Application: PAC-Policy Learning

One of the most common $\mathcal{SO}$-based settings is the computation of an ε-optimal policy via sample-based value iteration. Since GOSPRL is agnostic to how the sampling requirements are generated, we can easily integrate it with any state-of-the-art $\mathcal{SO}$-based algorithm and directly inherit its properties. For instance, consider the Bespoke algorithm introduced by Zanette et al. (2019). Bespoke proceeds through phases and at the beginning of each phase k, it determines the additional number of samples n_{sa}^{k+1} that need to be generated at each state-action pair (s, a) based on the estimates of the model and reward of the MDP computed so far. Then it simply queries the $\mathcal{SO}$ as needed and it moves to the following phase. In order to turn Bespoke into an online learning algorithm, we can simply replace the query step by running GOSPRL until n_{sa}^{k+1} samples are generated and then move to the next phase. Furthermore, let $b(s, a)$ be the total number of samples required by Bespoke in each state-action pair as stated by Zanette et al. (2019, Theorem 2), then we can directly apply Corollary E.2 and obtain the sample complexity of the online version of Bespoke (Online-Bespoke). As discussed in Section 7.3 the resulting complexity is *at most* a factor D larger than the one of (offline) Bespoke plus an additional term of order $\widetilde{O}(D^{3/2}S^2A)$ independent from the desired accuracy ε. It is interesting to contrast this result with existing online algorithms for this problem. While to the best of our knowledge, there is no algorithm specifically designed for optimal policy learning, we can rely on regret-to-PAC conversion (see e.g., Jin et al., 2018, Section 3.1) to derive sample complexity guarantees for existing regret minimization algorithms and do a qualitative comparison.[7] For instance, we can use Euler (Zanette and Brunskill, 2019) to derive an ε-optimal policy. If we consider a worst-case analysis, Euler achieves the same sample complexity of Bespoke, which in turn matches the lower bound of Azar et al. (2013). As a result, Online-Bespoke would be a factor D suboptimal w.r.t. to Euler. Nonetheless, our $\mathcal{SO}$-to-online learning conversion approach enables Online-Bespoke to directly benefit from the problem-dependent performance of Bespoke, which in many MDPs may outperform the guarantees obtained by using Euler as a online learning algorithm for policy optimization.

E.8.3 Application: Bandit Problems with MDP Dynamics

Algorithmic protocol

The sampling procedure GOSPRL provides an effective way to collect samples for states of the agent's choosing, and can thus be related to the multi-armed bandit setting by mapping arms (in bandits) to states (in MDPs). From Corollary E.2, each state can now be "pulled" within

[7] Regret minimization guarantees are usually provided for the finite-horizon setting, while Bespoke is designed for the discounted setting. Furthermore, the ε-optimality guarantees for $\mathcal{SO}$-based algorithms are typically defined in ℓ_∞ norm, while the regret-to-PAC conversion only provides guarantees on average w.r.t. the initial distribution.

$\widetilde{O}(D)$ time steps (instead of a single time step in the bandit case). This allows to naturally extend some *pure exploration* problems from the bandit setting to the communicating MDP setting. The algorithmic protocol alternates between the two following strategies:

1. the "bandit algorithm" identifies the arm(s), i.e., state(s), from which a sample is desired,
2. GOSPRL is executed to collect a sought-after sample as fast as possible.

To illustrate our decoupled approach we consider the two following problems: best-state identification (Section E.8.3) and reward-estimation, a.k.a. active exploration (Section E.8.3).

Best-state identification

This is the MDP extension of the best-arm identification problem in bandits (Audibert and Bubeck, 2010). Each state $s \in \mathcal{S} \triangleq \{1, \dots, S\}$ is characterized by a reward function r_s. For the sake of simplicity, we assume that the rewards are in $[0, 1]$ and that there is a unique highest-rewarding state $s^\star \triangleq \arg\max_s r_s$. Let $r^\star \triangleq r_{s^\star}$. Consider a budget of n steps. The objective is to bound the probability of error $e_n \triangleq \mathbb{P}(J_n \neq s^\star)$, where J_n is the state from which we desire a sample at step n. For $s \neq s^\star$, we introduce the following suboptimality measure of state s: $\Delta_s \triangleq r^\star - r_s$. We introduce the notation $(i) \in \{1, \dots, S\}$ to denote the i–th best arm (with ties break arbitrarily). The hardness of the task will be characterized by the following quantities $H_1 \triangleq \sum_{s \in \mathcal{S}} \frac{1}{\Delta_s^2}$ and $H_2 \triangleq \max_{s \in \mathcal{S}} s\Delta_{(s)}^{-2}$. These quantities are equivalent up to a logarithmic factor since we have $H_2 \leq H_1 \leq \log(2S) H_2$. A fully connected MDP with known and deterministic transitions amounts to a multi-armed bandit problem of $K \triangleq S$ arms for our problem, thus the Successive Rejects algorithm (Audibert and Bubeck, 2010) directly yields the following bound after j time steps

$$e_j \leq \frac{S(S-1)}{2} \exp\left(-\frac{j - SA}{\overline{\log}(S) H_2}\right), \quad \text{where } \overline{\log}(S) \triangleq \frac{1}{2} + \sum_{i=1}^{S} \frac{1}{i}.$$

In a general MDP, we combine GOSPRL (for the sample collection) with the Successive Rejects algorithm (for deciding which sample to collect). Consider any large enough budget of $n = \Omega(D^{3/2} S^2 A)$ time steps. Denote by j_n the number of time steps during which GOSPRL effectively collects the desired sample stipulated by the Successive Rejects algorithm. Corollary E.2 yields that $n = \widetilde{O}\left(D j_n + D^{3/2} S^2 A\right)$, which means that $j_n = \widetilde{\Omega}\left(\frac{n - D^{3/2} S^2 A}{D}\right)$. Therefore we obtain the following guarantee.

Lemma E.19. *In any unknown communicating MDP with unique highest-rewarding state $s^\star$, combining* GOSPRL *with the* Successive Rejects *algorithm (Audibert and Bubeck, 2010) yields the existence of a polynomial function p such that the probability e_n of wrongly identifying the "best*

state" $s^\star$ at time step n is upper bounded by

$$e_n \leq p(S, A, D, n) \exp\left(-\frac{n - D^{3/2}S^2A}{D\log(S)H_2}\right),$$

which corresponds to an exponential decrease w.r.t. n whenever n is large enough (i.e., after the $D^{3/2}S^2A$ burn-in phase).

Reward estimation (a.k.a. active exploration)

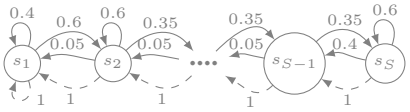
The objective of this problem in bandits (resp. MDPs) is to accurately estimate the mean pay-off (resp. the average reward signal) at each arm (resp. state). Note that this problem was originally studied in the bandit setting (see e.g., Carpentier et al., 2011) and we extended it in ergodic MDPs in Tarbouriech and Lazaric (2019) using a Frank-Wolfe approach. The extension to communicating MDPs remained an open question, and it becomes immediately addressed with GOSPRL. We recall the problem formulation: for a desired accuracy $\varepsilon > 0$, for each state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$ with mean reward $r_{s,a}$ in $[0, 1]$, we seek to output an estimate $\widehat{r}_{s,a}$ such that $|\widehat{r}_{s,a} - r_{s,a}| \leq \varepsilon$. Under the GOSPRL framework, it is sufficient to visit each state-action pair at least $\Omega\left(\varepsilon^{-2}\right)$ times, which directly induces the following sample complexity guarantee.

Lemma E.20. *In any unknown communicating MDP,* GOSPRL *can reach any reward-estimation accuracy $\varepsilon > 0$ with high probability under a sample complexity scaling as*

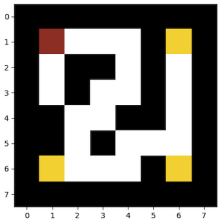
$$\widetilde{O}\left(\frac{DSA}{\varepsilon^2} + D^{3/2}S^2A\right).$$

Comment: Distinction between regret and sample complexity. Note that the results above (Lemma E.19 and E.20) do not provide any guarantee on the *regret* of the corresponding algorithms (which is often the metric of interest in sequential learning). Indeed, our algorithmic approach does not track nor adapt to a notion of optimal performance. Likewise, there remains to derive lower bounds on these problems extended to MDPs, in order to quantity the optimality of our procedure. Nonetheless, our decoupled approach is, to the best of our knowledge, the first method with provably bounded sample complexity that can successfully extend classical bandit problems (such as the two aforementioned ones) to communicating MDPs.

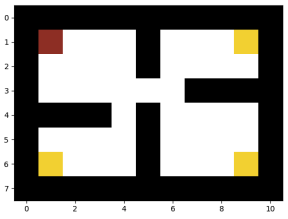
Comment: On the link between MDPs and bandits with a special form of transportation costs. Under the mapping between bandit arms and MDP states, our sampling paradigm has



(a) Reward-free RiverSwim ($S = 6$ states)



(b) Corridor gridworld ($S = 24$ states)



(c) 4-room gridworld ($S = 43$ states)

Figure E.2 – The three domains considered in Figure 7.1. For the gridworlds (b) and (c), the red tile is the starting state, yellow tiles are terminal states that reset to the starting state, and black tiles are reflecting walls (see §"Details on environments").

the effect of casting any MDP as a bandit problem with *transportation costs* between arms. In our setting, the transportation cost from a state to another is unknown, initially unbounded and has to be refined over the learning process (the asymptotically optimal cost amounts to the shortest path distance between the two states). We believe that such a setting of unknown and learnable transportation costs is an interesting formalism to study in the bandit setting, as it may then be applied to the MDP extension and allow for smart algorithms that take into account each transportation cost when proposing the arm/state from which a sample is desired (i.e., in part 1 of the algorithmic protocol given at the beginning of Section E.8.3). For completeness, it is worth mentioning that some papers study various settings of movement/switching costs between arms (see e.g., Dekel et al., 2014; Koren et al., 2017), yet none of these settings can be leveraged for our problem.

E.9 Experiments

This section complements the experiments reported in Section 7.5. We provide details about the algorithmic configurations and the environments as well as additional experiments.

Details on Figure 7.1 and Figure E.3. Figure 7.1 reports, as a function of time t, the proportion $\mathcal{P}_t$ of states that at time t satisfy the sampling requirements of the TREASURE-10 problem (i.e., $b(s,a) = 10$). Formally, $\mathcal{P}_t \triangleq |\{s \in \mathcal{S} : \forall a \in \mathcal{A}, N_t(s,a) \geq b(s,a)\}| \cdot S^{-1}$. As such, all sampling requirements are met as soon as $\mathcal{P}_t = 1$, meaning that the black line $y = 1$ on the y-axis characterizes our objective. Furthermore, we report in Figure E.3 results on additional domains (see below).

Details on environments. The three domains considered in Figure 7.1 are given in Figure E.2. The first one corresponds to a reward-free version of the RiverSwim domain introduced by Strehl and Littman (2008), which is a stochastic chain with 6 states and 2 actions classically used for testing exploration algorithms. The other two domains are gridworlds. In Figure E.3

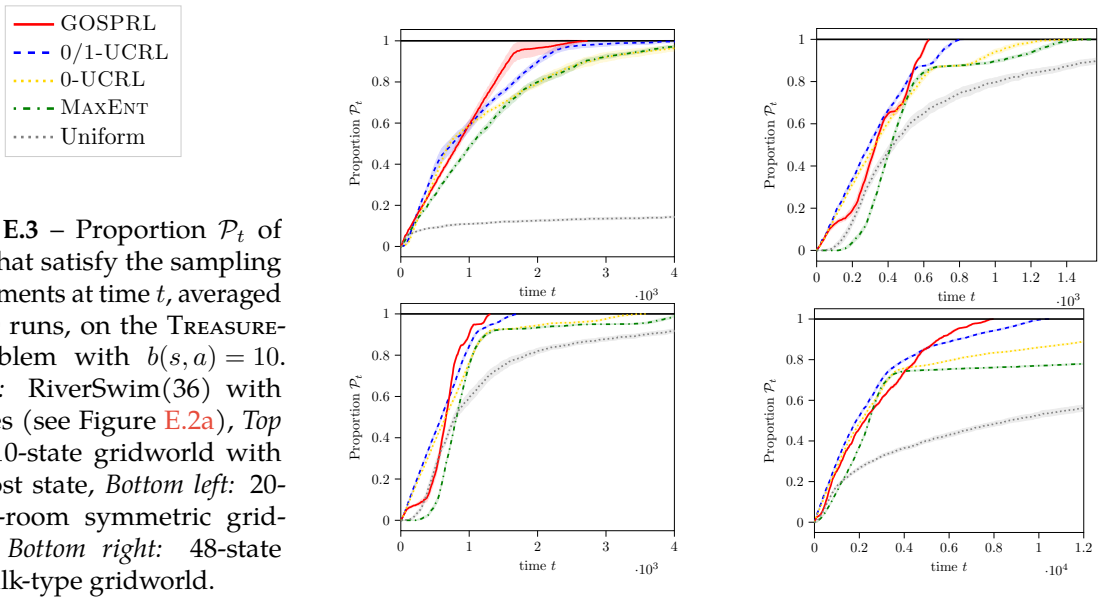
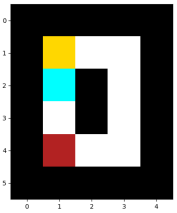
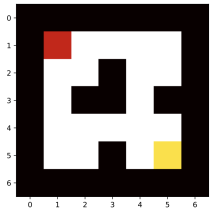


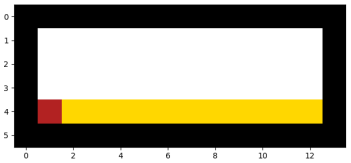
Figure E.3 – Proportion $\mathcal{P}_t$ of states that satisfy the sampling requirements at time t, averaged over 30 runs, on the Treasure-10 problem with $b(s,a) = 10$. *Top left:* RiverSwim(36) with 36 states (see Figure E.2a), *Top right:* 10-state gridworld with high-cost state, *Bottom left:* 20-state 4-room symmetric gridworld, *Bottom right:* 48-state CliffWalk-type gridworld.



(**a**) Gridworld with high-cost state ($S = 10$ states)



(**b**) 4-room symmetric gridworld ($S = 20$ states)



(**c**) CliffWalk-type gridworld ($S = 48$ states)

Figure E.4 – The three gridworlds considered in Figure E.3. The blue tile in (a) is a "trap state" that incurs large negative environmental reward and should thus be avoided as much as possible.

we test on a larger RiverSwim domain with 36 states and three additional gridworlds that are given in Figure E.4. Throughout our experiments, the gridworld domains are defined as follows. The agent can move using the cardinal actions (Right, Down, Left, Up). An action fails with probability $p_f = 0.1$, in which case the agent follows (uniformly) one of the other directions. The starting state is shown in red. Yellow tiles are terminal states that, when reached, deterministically reset to the starting state. The black walls act as reflectors, i.e., if the action leads against the wall, the agent stays in the current position with probability 1. The gridworlds are all reward-free, except the one in Figure E.4a where the blue tile incurs large negative environmental reward: it is thus a *trap state* which should be avoided as much as possible. Finally, in the experiments with the randomly generated Garnet environments and state-action requirements (Figure 7.2), we guarantee the MDPs randomly generated to be communicating by setting $P(s_0|s,a) \geq 0.001$ for every (s,a) and an arbitrary state s_0.

Table E.1 – For the TREASURE-10 problem, we report the quantities BD, $\sum_s b(s)D_s$ and the sample complexity of GOSPRL run with known dynamics (averaged over 30 runs), on the 3 domains of Figure E.2.

Environment	BD	$\sum_s b(s)D_s$	Sample comp. of GOSPRL run with known dynamics P
RiverSwim(6)	1766.7	958.7	249.9
Corridor gridworld(24)	24375.6	13695.2	3156.5
4-room gridworld(43)	27399.7	19048.3	3342.5

Algorithmic details. For all experiments and all considered algorithms, we choose a scaling factor $\alpha_p = 0.1$ of the confidence intervals of the transition probabilities (which enables to speed up the learning, see e.g., Fruit et al., 2018b), as well as a confidence level set to $\delta = 0.1$. Recall that for GOSPRL, in the case of state-only requirements, a state s is considered as under-sampled and is thus a goal state if $\sum_{a\in\mathcal{A}} N(s,a) < b(s)$, while in the case of state-action requirements, a state s is considered as under-sampled if $\exists a \in \mathcal{A}, N(s,a) < b(s,a)$. We consider the following initial phase for GOSPRL (i.e., when all states are under-sampled): we select as goal states those minimizing the "remaining budget" $b(s) - N(s)$ for state-only requirements (or $\sum_{a\in\mathcal{A}} \max\{b(s,a) - N(s,a), 0\}$ for state-action requirements), which has the effect of shortening the length of the initial phase. In the case of state-action requirements, once a sought-after goal state s is reached, GOSPRL selects an under-sampled action a whose gap $b(s,a) - N(s,a)$ is maximized. We note that this design choice can be observed in Figure 7.1 and E.3 where GOSPRL seeks to "even out" its sampling strategy, with a steady increase in $(\mathcal{P}_t)$, instead of exhausting the requirements state after state.

GOSPRL-for-MODEST algorithm. Here we detail the GOSPRL-for-MODEST algorithm used in the MODEST experiment of Figure 7.3. The GOSPRL sampling requirements are computed using a decreasing MODEST accuracy η, which enables the algorithm to be accuracy-agnostic like the WEIGHTEDMAXENT heuristic to which it is compared. GOSPRL-for-MODEST starts at an initial accuracy of $\eta \leftarrow 1$ and iteratively performs the two following steps until the algorithm ends: *i*) it requires a sampling requirement of $b_t^{\text{MODEST}}(s,a) = \alpha_b \cdot \Phi\big(\sum_{s'\in\mathcal{S}} \sqrt{\widehat{\sigma}_t^2(s'|s,a)}\ ,\ S\big)$, where Φ is defined after Equation (7.3) for accuracy η and where $\alpha_b = 0.01$ is a scaling factor to speed up the learning; and *ii*) when the sampling requirements are fulfilled by GOSPRL, it sets $\eta \leftarrow \eta/2$ and goes back to the first step.

Dependencies. For each environment of Figure E.2 on the TREASURE-10 problem (i.e., $b(s,a) = 10$, $B = 10SA$), we compute in Table E.1 the sample complexity of GOSPRL run with known dynamics, to put aside the learning component so that its corresponding sample complexity can be bounded exactly by BD or by $\sum_s b(s)D_s$ according to the analysis in Section 7.3.1. Both bounds are reported in Table E.1: we observe that the second (more state-dependent) quantity is tighter and more preferable than the first. Despite both bounds being loose w.r.t. the actual algorithmic performance, they can effectively capture the difficulty of the

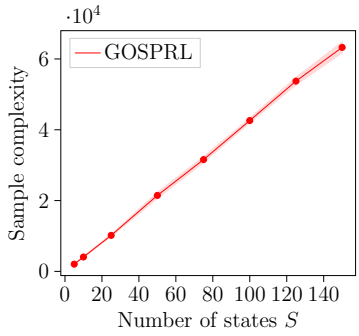


Figure E.5 – Sample complexity of GOSPRL in randomly generated Garnet MDPs for increasing values of S, with all other parameters fixed $(A, \beta, \overline{U})$ as in Figure 7.2. Results are averaged over 5 Garnets, each for 12 runs.

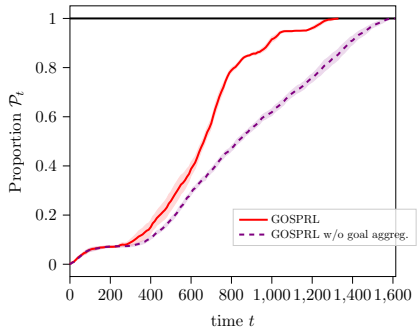


Figure E.6 – Impact of goal aggregation on GOSPRL. Proportion $\mathcal{P}_t$ averaged over 30 runs, on the Treasure-10 problem with $b(s, a) = 10$ on the environment of Figure E.4b.

Table E.2 – Impact of cost shaping on GOSPRL. On the environment of Figure E.4a, sampling requirement are concentrated at the yellow terminal state $y \in \mathcal{S}$, i.e., $b(y, a) = 10$ for all $a \in \mathcal{A}$. Cost-weighted GOSPRL sets a cost of 10 (instead of 1) at the blue trap state during each SSP planning step. Values are averaged over 30 runs.

	GOSPRL (Algorithm 7.1)	Cost-weighted GOSPRL
Sample complexity	253.1	520.0
Visits to trap state	44.6	4.7

problem (in a relative sense where the higher the bounds, the higher the sample complexity). We also recall from Section 7.5 that there exist simple worst-case problems (see e.g., Figure 7.4) where these bounds are tight, i.e., where the sample complexity of GOSPRL (whether the dynamics are known or not) must directly scale with these diameter quantities. Notice that running GOSPRL with known dynamics corresponds to deploying an optimal *greedy* strategy (i.e., by minimizing each time to reach under-sampled states in a sequential fashion), which is likely not the optimal non-stationary solution (which would involve solving a sort of highly difficult, online travelling salesman problem), see Section E.2.3 for additional discussion. Finally, we study the sample complexity of GOSPRL across similar MDPs with increasing number of states to see how that dependence pans out. Figure E.5 reports the sample complexity of GOSPRL in randomly generated Garnet MDPs for increasing values of S. We observe that as expected, the sample complexity scales linearly with S.

Impact of goal aggregation on GOSPRL. GOSPRL iteratively aggregates the undersampled states into a *meta-goal* for which it computes an optimistic goal-oriented policy. While it is possible to focus on specific goal states as mentioned in Section 7.3 without affecting the sample complexity guarantee, performing the goal aggregation leads to shorter and more successful sample collection attempts. We observe in Figure E.6 that this indeed translates into better empirical performance. Indeed, GOSPRL collects the prescribed samples faster than a version

of GOSPRL that selects uniformly at random a single goal state among all undersampled states (i.e., that does not perform goal state aggregation).

Impact of cost shaping on GOSPRL. While GOSPRL in Algorithm 7.1 considers unit costs for each SSP problem it constructs, any non-unit costs can be designed as long as they are positive and bounded. In particular, detering costs may be assigned to trap states with large negative environmental reward that the agent seeks to avoid. To study this, we consider the gridworld of Figure E.4a where the blue tile is a trap state that the agent must avoid as much as possible. For ease of exposition we consider here sampling requirements concentrated at the terminal state in yellow denoted by $y \in \mathcal{S}$, i.e., $b(y, a) = 10$ for any $a \in \mathcal{A}$. We compare GOSPRL with a cost-weighted GOSPRL where a cost of 10 is set at the blue trap state during each SSP planning step. Table E.2 shows that while the sample complexity of cost-weighted GOSPRL is worsened, the number of visits to the undesirable trap state is considerably decreased w.r.t. GOSPRL. This makes sense since the shortest path from the red starting state to the sought-after yellow terminal state goes through the blue trap state, so a trade-off appears between minimizing the sample complexity and visiting undesirable states. This numerical simulation shows that GOSPRL can naturally adjust this trade-off by cost-weighting the successive SSP problems it tackles.

Appendix F

Complements on Chapter 8

F.1 Proofs

Proof of Equation (8.3). Here we take the limit $H \to +\infty$. Let $g \in \mathcal{G}_L$, then Property 1 entails that $\mathcal{D}_k(g) \leq V^\star(s_0 \to g) \leq L$, thus $g \in \mathcal{X}_k$, therefore $\mathcal{G}_L \subseteq \mathcal{X}_k$. Now let $g \in \mathcal{X}_k$, then by Property 2 and definition of ε_k, we have that $V^\star(s_0 \to g) \leq \mathcal{D}_k(g) + \mathcal{E}_k(g) \leq L + \mathcal{E}_k(g) \leq L + \varepsilon_k$, therefore $\mathcal{X}_k \subseteq \mathcal{G}_{L+\varepsilon_k}$. □

Proof of Equation (8.5). Fix any finite episode index $K < \kappa$, where κ denotes the (possibly unbounded) episode index at which AdaGoal-UCBVI terminates. By design of AdaGoal, we have that $\varepsilon \leq \mathcal{E}_k(g_k)$ for every $k \leq K$. We assume that $\kappa > 1$ otherwise the result is trivially true. Define $\kappa' \triangleq \min\{\kappa - 1, K\} \geq 1$. Summing the inequality above yields $\varepsilon \cdot \kappa' \leq \sum_{k=1}^{\kappa'} \mathcal{E}_k(g_k) \leq \kappa' \cdot f_{\mathcal{M}}(\kappa')$. This implies that $\varepsilon \leq f_{\mathcal{M}}(\kappa')$, in which case $f_{\mathcal{M}}^{-1}(\varepsilon) \geq \kappa'$, since $f_{\mathcal{M}}^{-1}$ is strictly decreasing (like $f_{\mathcal{M}}$). Since the last inequality holds for any finite $K < \kappa$, letting $K \to +\infty$ implies that κ is finite and bounded by $f_{\mathcal{M}}^{-1}(\varepsilon) + 2$. □

F.1.1 Proof of Lemma 8.7

We assume throughout that $L \geq 2$ and $\varepsilon \in (0, 1]$. On the one hand, Theorem 8.11 proves that AdaGoal-UCBVI is $(\varepsilon, \delta, L, \mathcal{G})$-PAC for MGE with sample complexity of order $\widetilde{O}(L^3 S A \varepsilon^{-2})$, therefore MGE is solvable in poly$(S, L, \varepsilon^{-1}, A)$ steps (up to poly-log factors). On the other hand, reset-free MGE is a special case of the cost-free goal-free exploration problem in communicating MDPs studied in Section 7.4.3, where we showed that a GOSPRL-based algorithm can solve it in poly$(S, D, \varepsilon^{-1}, A)$ steps (up to poly-log factors). Now we prove that there exists an MDP such that any algorithm requires at least $\Omega(D)$ steps to solve the reset-free MGE problem (i.e., without Assumption 8.3), where the MDP's diameter D can be made exponentially larger than $L, S, A, \varepsilon^{-1}$.

To this end, we design in Figure F.1 a communicating MDP $\mathcal{M}_{a^\dagger}$ that does not satisfy Assumption 8.3, with $A \geq 4$ actions (including a special action $a^\dagger \in \mathcal{A}$) and $S = 5$ states, where $\mathcal{S} \triangleq \{s_0, g, x, \widetilde{s}, \overline{s}\}$, and all states apart from $\overline{s}$ are reliably L-reachable from s_0. We define as goal space $\mathcal{G} \triangleq \{g\}$. We consider the problem of learning an ε-optimal goal-reaching policy with goal state g from the starting state s_0, i.e., finding a policy π such that $V^\pi(s_0 \to g) \leq V^\star(s_0 \to g) + \varepsilon$, which is a sub-problem of the MGE objective.

Given $\eta \in (0,1)$ and an unknown action $a^\dagger \in \mathcal{A}$, we define the following transition probabilities for all $a \in \mathcal{A}$,

$$
\begin{aligned}
&P(\widetilde{s}|s_0,a) \triangleq \eta, && P(x|s_0,a) \triangleq 1-\eta,\\
&P(g|x,a) \triangleq \zeta, && P(x|x,a) \triangleq 1-\zeta,\\
&P(g|\widetilde{s},a) \triangleq \mathbb{1}[a = a^\dagger], && P(\overline{s}|\widetilde{s},a) \triangleq \mathbb{1}[a \neq a^\dagger],\\
&P(g|\overline{s},a) \triangleq \frac{\eta}{2}, && P(\overline{s}|\overline{s},a) \triangleq 1-\frac{\eta}{2},\\
&P(s_0|g,a) \triangleq 1,
\end{aligned}
$$

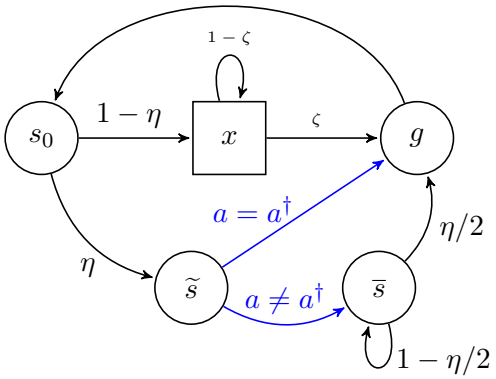


Figure F.1 – Illustration of the MDP instance $\mathcal{M}_{a^\dagger}$.

where we can set any $\zeta = O(1/L)$ such that g is reliably L-reachable from s_0 (i.e., $V^\star(s_0 \to g)$). It is easy to see that the MDP's diameter verifies $D = \alpha\eta^{-1}$ for a constant $\alpha > 0$. Finally, we denote by $\mathcal{F}$ the family of MDPs of the form of Figure F.1 parameterized by $a^\dagger \in \{1, \dots A\}$, i.e., $\mathcal{F} \triangleq \{\mathcal{M}_{a^\dagger}\}_{a^\dagger \in \{1,\dots A\}}$.

We define the event $\mathcal{J}_t$ that state $\widetilde{s}$ has never been visited by the agent by time t (recalling that it initially starts at state s_0), i.e., $\mathcal{J}_t \triangleq \{n^t(\widetilde{s}) = 0\}$ (note that its probability is the same for all MDPs in $\mathcal{F}$). We now fix an MDP $\mathcal{M}_{a^\dagger}$ and denote by $\pi^\star$ its optimal policy, i.e., $\pi^\star \in \arg\min_\pi V^\pi(\cdot \to g)$, which in particular selects action $a^\dagger$ at state $\widetilde{s}$. First, we consider any deterministic algorithm $\mathfrak{A}$ whose candidate policy $\widehat{\pi}$ does not select action $a^\dagger$ when it is in state $\widetilde{s}$. Then it holds that

$$
V^{\widehat{\pi}}(\widetilde{s} \to g) = 1 + \sum_{y \in \mathcal{S}} P(y|s_0, \widehat{\pi}(\widetilde{s})) V^{\widehat{\pi}}(y \to g) \geq 1 + V^{\widehat{\pi}}(\overline{s} \to g) = 1 + \frac{2}{\eta}, \tag{F.1}
$$

$$
V^{\widehat{\pi}}(\widetilde{s} \to g) - V^{\pi^\star}(\widetilde{s} \to g) \geq \frac{2}{\eta},
$$

$$
V^{\pi}(s_0 \to g) = 1 + \eta V^{\pi}(\widetilde{s} \to g) + (1-\eta) V^{\pi}(x \to g), \quad \forall \pi,
$$

$$
V^{\widehat{\pi}}(s_0 \to g) - V^{\pi^\star}(s_0 \to g) = (1-\eta)\big(\underbrace{V^{\widehat{\pi}}(x \to g) - V^{\pi^\star}(x \to g)}_{\geq 0}\big) \tag{F.2}
$$

$$
+ \eta\big(\underbrace{V^{\widehat{\pi}}(\widetilde{s} \to g) - V^{\pi^\star}(\widetilde{s} \to g)}_{\geq 2/\eta}\big) \geq 2 > \varepsilon, \tag{F.3}
$$

thus $\widehat{\pi}$ has a sub-optimality gap larger than ε. This means that under the event $\mathcal{J}_t$, where the algorithm cannot know which is the favorable action $a^\dagger$, it holds that with probability at least $1 - \frac{1}{A} \geq \frac{3}{4}$, any deterministic algorithm's candidate policy $\widehat{\pi}$ does not select the action $a^\dagger$ at state $\widetilde{s}$ thus its value function is not ε-optimal. Second, we note that we can easily extent to the case where $\mathfrak{A}$ outputs stochastic actions at state $\widetilde{s}$. Given that $a^\dagger$ is unknown, in the best case scenario it can set $\widehat{\pi}(\cdot|\widetilde{s}) = 1/A$. Then we can retrace the reasoning above and replace (F.1) with $V^{\widehat{\pi}}(\widetilde{s}) \geq 1 + \frac{A-1}{A}\frac{2}{\eta}$, and thus (F.3) with $V^{\widehat{\pi}}(s_0) - V^{\pi^\star}(s_0) \geq 2\frac{A-1}{A} > 1 \geq \varepsilon$ since $A > 2$, which leads to the same result that $\widehat{\pi}$ is not ε-optimal on at least one of the MDPs in $\mathcal{F}$.

Now we study how the probability of the event $\mathcal{J}_t$ evolves over time t, i.e. we bound the time required to visit $\widetilde{s}$ at least once, which we denote by $\widetilde{T}$. Recall that $\widetilde{s}$ can only be reached with probability η from s_0 following any action. The random variable $\widetilde{T}$ can be seen as an upper bound of a random variable distributed geometrically with success probability η, thus Chernoff's inequality entails that with probability at least $\frac{1}{2}$ we have $\widetilde{T} \geq \frac{1}{9\eta}$, i.e., the event $\mathcal{J}_t$ for $t = \lfloor 1/(9\eta) \rfloor$ holds with probability at least $\frac{1}{2}$.

Putting everything together, there exists an MDP in $\mathcal{F}$ such that with probability at least $\frac{1}{4}$, the number of time steps required to output a candidate policy that is ε-optimal for goal state g is at least $\frac{1}{9\eta} = \Omega(D)$, where η can be made arbitrarily small, so in particular D can be exponentially larger than L. Hence in this MDP instance, no algorithm can solve MGE in poly(L) steps with overwhelming probability. While here we considered $S = 5$ for simplicity, note that the result easily holds for any $S \geq 5$ by replacing the state x in the construction above with a set of $S - 4$ states; the only property that must be still verified is that g remains reliably L-reachable from s_0.

Therefore, for any $L \geq 2, S \geq 5, A \geq 4, \varepsilon \in (0, 1]$, there exists an MDP instance and a goal space for which any algorithm requires at least $\Omega(D)$ time steps to solve reset-free MGE, where the diameter D can be exponentially larger than $L, S, A, \varepsilon^{-1}$, which concludes the proof of Lemma 8.7.

F.1.2 Proof of Lemma 8.8

In the following, we will prove in Lemma F.2 a lower bound on the expected number of exploration steps to find an ε-optimal SSP policy from s_0 for a specific goal state g that is reliably L-reachable from s_0. Such BPI-SSP objective corresponds to our MGE objective with goal space $\mathcal{G} = \{g\}$ and it thus induces a lower bound on the MGE problem, which will conclude the proof of Lemma 8.8.

Notation. We largely follow the notations and definitions of Domingues et al. (2021b). We define an RL algorithm as a history-dependent policy $\boldsymbol{\pi}$ used to interact with the environment. In the BPI setting, where we eventually stop and recommend a policy, an algorithm is defined

as a triple $(\boldsymbol{\pi}, \tau, \widehat{\pi}_\tau)$ where τ is a stopping time and $\widehat{\pi}_\tau$ is a Markov policy recommended after τ time steps. We now write more formally our objective.

Definition F.1 (BPI-SSP). *An algorithm $(\boldsymbol{\pi}, \tau, \widehat{\pi}_\tau)$ is (ε, δ, L)-PAC for best-policy identification in a stochastic shortest path problem satisfying Assumption 8.3 (BPI-SSP) if $V^\star(s_0) \leq L$ and if the policy $\widehat{\pi}_\tau$ returned after τ time steps satisfies, for the initial state s_0,*

$$\mathbb{P}_{\boldsymbol{\pi},\mathcal{M}}\left[V^{\widehat{\pi}_\tau}(s_0) - V^*(s_0) \leq \varepsilon\right] \geq 1 - \delta,$$

where we denote the goal state by g and the SSP value of any policy π at any state s by $V^\pi(s) \triangleq \mathbb{E}_{\boldsymbol{\pi},\mathcal{M}}\left[\inf\{i \geq 0 : s_{i+1} = g\} \,|\, s_1 = s\right]$, and $V^\star(s) \triangleq \min_\pi V^\pi(s)$.

Lemma F.2 (BPI-SSP Lower Bound). *There exist absolute constants $L_0, S_0, A_0, \varepsilon_0, \delta_0$ such that, for any $L \geq L_0$, $A \geq A_0$, $\varepsilon \leq \varepsilon_0$, $\delta \leq \delta_0$ and $S_0 \leq S \leq A^{\frac{L}{3}-2}$, and for any algorithm $(\boldsymbol{\pi}, \tau, \widehat{\pi}_\tau)$ that is (ε, δ, L)-PAC for BPI-SSP in any finite MDP with S states and A actions, there exists an MDP $\mathcal{M}$ with a goal state belonging to $\mathcal{G}_L$ and an absolute constant β such that*

$$\mathbb{E}_{\boldsymbol{\pi},\mathcal{M}}[\tau] \geq \beta \frac{L^3 SA}{\varepsilon^2} \log\left(\frac{1}{\delta}\right).$$

Proof of Lemma F.2.

We first define our family of hard MDPs for $S = 4$ states, and the extension to any S states can be done as in Domingues et al., 2021b as explained later. Consider the hard MDP illustrated in Figure F.2, where all states incur a cost of 1 apart from the goal state s_g (a.k.a. "good" state). The agent stays in the initial state s_0 with probability $1 - q$, and goes to a *decision state* s_d with probability q. For any action a taken in s_d, the agent reaches s_g with probability $1/2$ and a "bad" state s_b with probability $1/2$, except if an action $a^\star$ is chosen, that increases to $1/2 + \widetilde{\varepsilon}$ the probability of reaching s_g. From s_b, the agent returns to the initial state s_0 with probability 1. The goal state s_g is absorbing, and the agent stays there unless the reset action is taken, which brings the agent back to s_0.

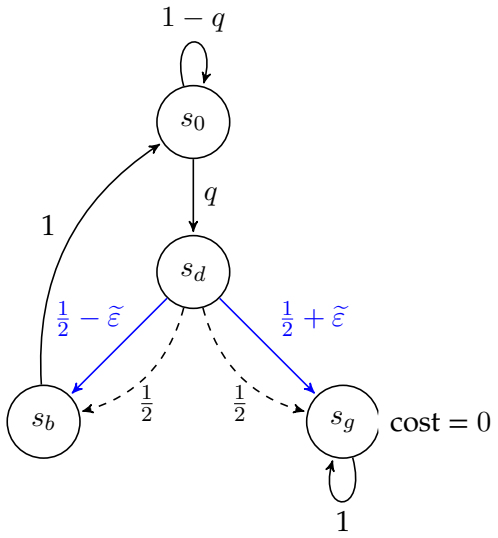


Figure F.2 – Illustration of the hard MDP considered in the proof of Lemma F.2.

Note that the MDP satisfies Assumption 8.3 (the arrows of the reset action from s_d to s_0 and from s_g to s_0 are not represented in Figure F.2 for visual convenience). Moreover, we define the following parameters

$$H \triangleq \lceil \frac{L}{2} - 1 \rceil, \quad q \triangleq 1/H, \quad \widetilde{\varepsilon} \triangleq \frac{\varepsilon}{2(H+1)}.$$

Note that this hard MDP instance is inspired from hard MDPs used in prior lower-bound constructions (see e.g., Lattimore and Hutter, 2012; Domingues et al., 2021b), albeit with slight modifications. Indeed, a key difference with respect to the discounted MDP setting (Lattimore and Hutter, 2012) or the finite-horizon MDP setting (Domingues et al., 2021b) is that in our case, the agent has access to an anytime reset action (Assumption 8.3). This implies that we cannot do as prior works that rely on absorption properties of states in the MDP (e.g., the "good" and "bad" states s_b and s_g) in order to compound errors and add an extra effective horizon term (either $1/(1-\gamma)$ or H) in the sample complexity (i.e., to go from quadratic to cubic). The only absorption property we can rely on here is at the initial state s_0. It turns out that this will be sufficient in our setting to compound error and go from L^2 to L^3 dependence. The intuition for this is that the SSP value function generates a cost of $+1$ at each time step until the goal state is reached, which compounds errors more than the usual reward-based value function in the sparse-reward MDP constructions of Lattimore and Hutter (2012) and Domingues et al. (2021b).

We consider a family of MDPs of the form of Figure F.2, parameterized by $a^\star \in \{1, \dots A\}$, where we denote by $\mathcal{M}_{a^\star}$ the MDP such that $a^\star$ increases the probability by $\widetilde{\varepsilon}$ of reaching the goal state s_g from state s_d. For any policy π we denote its SSP value (with goal state s_g) at state s in $\mathcal{M}_{a^\star}$ by $V^\pi_{a^\star}(s)$.

We also define a reference MDP $\mathcal{M}_0$, where $\widetilde{\varepsilon} = 0$, that is, there is no special action increasing the probability of reaching the goal state s_g. We denote by $\mathbb{P}_{a^\star}[\cdot]$ and $\mathbb{E}_{a^\star}[\cdot]$ the probability measure and the expectation in the MDP $\mathcal{M}_{a^\star}$ by following the algorithm $\boldsymbol{\pi}$ and by $\mathbb{P}_0[\cdot]$ and $\mathbb{E}_0[\cdot]$ the corresponding operators in $\mathcal{M}_0$.

The optimal value function does not depend on the MDP parameter $a^\star$, and for any MDP $\mathcal{M}_{a^\star}$ we get

$$\begin{aligned} V^\star(s_0) &= \frac{1}{q} + \Big(\frac{1}{2} + \widetilde{\varepsilon}\Big) + \Big(1 - \frac{1}{2} - \widetilde{\varepsilon}\Big)\big(1 + V^\star(s_0)\big) \\ \implies V^\star(s_0) &= \Big(\frac{1}{q} + 1\Big)\frac{1}{1/2 + \widetilde{\varepsilon}}. \end{aligned}$$

Note that by our choice of q, it importantly holds that $V^\star(s_0) \leq L$, i.e., $s_g \in \mathcal{G}_L$.

Meanwhile, the value function of the recommended policy $\widehat{\pi}_\tau$ in $\mathcal{M}_{a^\star}$ is

$$V_{a^\star}^{\widehat{\pi}_\tau}(s_0) = \Big(\frac{1}{q}+1\Big)\frac{1}{1/2+\widetilde{\varepsilon}\cdot\widehat{\pi}_\tau(a^\star|s_d)}.$$

As a result,

$$V_{a^\star}^{\widehat{\pi}_\tau}(s_0) - V^\star(s_0) = \Big(\frac{1}{q}+1\Big)\frac{\widetilde{\varepsilon}(1-\widehat{\pi}_\tau(a^\star|s_d))}{(1/2+\widetilde{\varepsilon})(1/2+\widetilde{\varepsilon}\cdot\widehat{\pi}_\tau(a^\star|s_d))} \leq \underbrace{4\Big(\frac{1}{q}+1\Big)\widetilde{\varepsilon}}_{=2\varepsilon}\cdot(1-\widehat{\pi}_\tau(a^\star|s_d)),$$

and thus

$$V_{a^\star}^{\widehat{\pi}_\tau}(s_0) - V^\star(s_0) < \varepsilon \iff \widehat{\pi}_\tau(a^\star|s_d) > \frac{1}{2}.$$

We observe that this analysis of the suboptimality gap of $\widehat{\pi}_\tau$ in terms of SSP value functions can be mapped to the one of Domingues et al. (2021b, Proof of Theorem 7) for their finite-horizon value functions, despite the different MDP constructions. This means that we can retrace the steps of Domingues et al. (2021b, Proof of Theorem 7). In the following, we use the notation $\overset{\text{(D)}}{\geq}$ to signify that the inequality stems from following the same corresponding steps of Domingues et al. (2021b, Proof of Theorem 7). In particular, we similarly define the event

$$\mathcal{E}_{a^\star}^\tau \triangleq \Big\{\widehat{\pi}_\tau(a^\star|s_d) > \frac{1}{2}\Big\},$$

and since the algorithm is assumed to be (ε,δ,L)-PAC for BPI-SSP for any MDP, we have

$$\mathbb{P}_{a^\star}\Big[\mathcal{E}_{a^\star}^\tau\Big] = \mathbb{P}_{a^\star}\Big[V_{a^\star}^{\widehat{\pi}_\tau}(s_0) < V^\star(s_0)+\varepsilon\Big] \geq 1-\delta.$$

We now proceed with lower bounding the expectation of the sample complexity τ in the reference MDP $\mathcal{M}_0$. We define

$$N_{a^\star}^\tau \triangleq \sum_{t=1}^{\tau}\mathbb{1}[S_t = s_d, A_t = a^\star], \quad N^\tau \triangleq \sum_{a^\star} N_{a^\star}^\tau.$$

Following the steps of Domingues et al. (2021b, Proof of Theorem 7), we have that

$$\mathbb{E}_0\Big[N_{a^\star}^\tau\Big]4\widetilde{\varepsilon}^2 \overset{\text{(D)}}{\geq} \mathbb{E}_0\Big[N_{a^\star}^\tau\Big]\mathrm{kl}\Big(\frac{1}{2},\frac{1}{2}+\widetilde{\varepsilon}\Big) \overset{\text{(D)}}{\geq} \Big[\Big(1-\mathbb{P}_0\Big[\mathcal{E}_{a^\star}^\tau\Big]\Big)\log\Big(\frac{1}{\delta}\Big)-\log(2)\Big],$$

thus

$$\mathbb{E}_0\Big[N_{a^\star}^\tau\Big] \geq \frac{1}{4\widetilde{\varepsilon}^2}\Big[\Big(1-\mathbb{P}_0\Big[\mathcal{E}_{a^\star}^\tau\Big]\Big)\log\Big(\frac{1}{\delta}\Big)-\log(2)\Big].$$

Summing all over MDP instances, we obtain following Domingues et al. (2021b, Proof of Theorem 7) that

$$\mathbb{E}_0\left[N^\tau\right] = \sum_{a^\star} \mathbb{E}_0\left[N^\tau_{a^\star}\right] \overset{(D)}{\geq} \frac{A}{8\widetilde{\varepsilon}^2} \log\left(\frac{1}{\delta}\right).$$

While the proof of Domingues et al. (2021b, Theorem 7) can stop at this stage, our proof requires an additional step of linking this back to the sample complexity τ, since the latter is not defined in terms of number of episodes but in terms of number of time steps.

For any $i \leq \tau$, denote by $W_i(s_0 \to s_d)$ the random variable of the number of time steps required to reach s_d starting from s_0 for the i-th time — note importantly that this quantity is independent of the algorithm and of the MDP parameter $a^\star$, and we can write $\mathbb{E}\left[W_i(s_0 \to s_d)\right] = \mathbb{E}\left[W(s_0 \to s_d)\right] = \frac{1}{q}$. Then from Wald's equation we have

$$\mathbb{E}_0[\tau] \geq \mathbb{E}_0\left[\sum_{i=1}^{N^\tau} W_i(s_0 \to s_d)\right] = \mathbb{E}\left[W(s_0 \to s_d)\right] \cdot \mathbb{E}_0\left[N^\tau\right] \geq \frac{1}{q} \cdot \alpha \frac{1}{\widetilde{\varepsilon}^2} A \log\left(\frac{1}{\delta}\right) \geq \alpha' \frac{L^3 A}{\varepsilon^2} \log\left(\frac{1}{\delta}\right),$$

where α and α' are absolute constants.

Finally, as mentioned, the extension to any S to make appear the multiplicative dependence on S can be done by following the steps done in Domingues et al. (2021b, Proof of Theorem 7) (which relies on their Assumption 1, see their Theorem 10 for the relaxed statement). The idea of the construction is to consider not just 1 decision state s_d but $S-3$ of them, where only one of them possesses the favorable action $a^\star$; intuitively this generates SA actions instead of A (see e.g., Lattimore and Szepesvári, 2020, Section 38.7), thus leading to the additional S factor in the sample complexity. □

F.2 DETAILS OF AdaGoal-UCBVI AND ANALYSIS

In this section, we focus on AdaGoal-UCBVI. In Section F.2.1, we introduce useful notation. In Section F.2.2, we define the exact choice of estimates $\mathcal{D}$, $\mathcal{E}$, $\mathcal{Q}$ used by AdaGoal-UCBVI (line 13 of Algorithm 8.1). Then in Section F.2.3, we provide the full proof of Theorem 8.11, by establishing the key steps followed in Section 8.3. Throughout the analysis we consider that $\mathcal{G} = \mathcal{S}$, $\varepsilon \in (0, 1]$ and $\delta \in (0, 1)$.

F.2.1 Notation

Given a goal state $g \in \mathcal{G}$, denote by $\mathcal{M}_g$ the unit-cost SSP-MDP which adds a self-loop at g to the original MDP $\mathcal{M}$, and denote by $P_g($ its transition function and c_g its cost function. Formally, let

$$c_g(s,a) \triangleq \mathbb{1}[s \neq g], \quad P_g(s'|s,a) \triangleq \begin{cases} P(s'|s,a) & \text{if } s \neq g \\ \mathbb{1}[s'=g] & \text{if } s = g. \end{cases}$$

For any (possibly non-stationary) policy $\pi = (\pi_h)_{h \geq 1}$, let V_g^π be its SSP value function (i.e., expected cost-to-go) in $\mathcal{M}_g$, i.e.,

$$V_g^\pi(s_0) \triangleq \mathbb{E}\left[\sum_{h=1}^{+\infty} c_g(s_h,a_h) \mid s_1 = s_0, \pi, \mathcal{M}_g\right],$$

where $a_h \triangleq \pi_h(s_h)$ and $s_{h+1} \sim P_g(s_h,a_h)$. Let $\pi_g^\star \in \arg\min_\pi V_g^\pi$ and $V_g^\star \triangleq V_g^{\pi_g^\star}$. We now define the set of finite-horizon goal-conditioned models.

Definition F.3 (Finite-Horizon Goal-Conditioned Models)**.** *Fix a horizon $H \geq 1$. For any goal state $g \in \mathcal{G}$, denote by $\overline{\mathcal{M}}_{g,H}$ the finite-horizon model that corresponds to starting from state s_0 and interacting for H steps with the original MDP $\mathcal{M}$ in which state g is made absorbing. $\overline{\mathcal{M}}_{g,H}$ admits as cost function $\overline{c}_g \triangleq c_g$ and as transition function $\overline{P}_g \triangleq P_g$.*

Remark F.4. Note that for a given goal state $g \in \mathcal{G}$, the cost function $\overline{c}_g$ is known to the agent, while the MDP transitions $\overline{P}_g$ are unknown with some (known) goal-specific changes w.r.t. the original MDP $\mathcal{M}$ (namely, the self-loop at g). An alternative way of framing the problem is that there is one single MDP with state space $\mathcal{S} \times \mathcal{G}$, i.e., with state variable (s,g).

For a finite-horizon policy $\pi \in \Pi_H$, denote by $\overline{V}_{g,h}^\pi$ its finite-horizon value function at step $1 \leq h \leq H$ in the finite-horizon instance $\overline{\mathcal{M}}_{g,H}$, i.e.,

$$\overline{V}_{g,h}^\pi(s_0) \triangleq \mathbb{E}\left[\sum_{h'=h}^{H} \overline{c}_g(s_{h'},a_{h'}) \mid s_1 = s_0, \pi, \overline{\mathcal{M}}_{g,H}\right].$$

We define the corresponding optimal value function as $\overline{V}_{g,h}^\star \triangleq \min_\pi \overline{V}_{g,h}^\pi$. Observe that $\overline{V}_{g,1}^\star(s_0) = \mathcal{D}_H^\star(g)$ (notation used in Properties 1 and 2 of Section 8.2).

Let (s_i,a_i,s_{i+1}) be the state, action and next state observed by an algorithm at time step i. Let $n^k(s,a) \triangleq \sum_{i=1}^{kH} \mathbb{1}[(s_i,a_i) = (s,a)]$ be the number of times state-action pair (s,a) was visited in the first k episodes and $n^k(s,a,s') \triangleq \sum_{i=1}^{kH} \mathbb{1}[(s_i,a_i,s_{i+1}) = (s,a,s')]$. We define the empirical transitions as $\widehat{P}^k(s'|s,a) \triangleq n^k(s,a,s')/n^k(s,a)$ if $n^k(s,a) > 0$, and $\widehat{P}^k(s'|s,a) \triangleq 1/S$ otherwise. Also, $PX(s,a) \triangleq \mathbb{E}_{s' \sim P(\cdot|s,a)}[X(s')]$ denotes the expectation operator w.r.t. the

transition probabilities P and $\pi_h Y(s) \triangleq Y(s, \pi_h(s))$ denotes the composition with policy π at step h, so that $P\pi_h Z(s,a) \triangleq \mathbb{E}_{s' \sim P(\cdot|s,a)}[Z(s', \pi_h(s'))]$. Finally, we denote the clip function by $\mathrm{clip}(x,y,z) \triangleq \max(\min(x,z),y)$.

Finally, in the analysis we denote by $p^k_{g,h}(s,a)$ the probability of reaching state-action pair (s,a) at step h under policy π^k_g in the true MDP. We also define the pseudo-counts as $\overline{n}^k(s,a) \triangleq \sum_{h=1}^{H} \sum_{l=1}^{k} p^l_{g_l,h}(s,a)$, where $g_l \in \mathcal{S}$ denotes the goal state selected by AdaGoal-UCBVI at the beginning of algorithmic episode l.

F.2.2 Algorithmic Choices of $\mathcal{D}$, $\mathcal{E}$, $\mathcal{Q}$ for AdaGoal-UCBVI

We generalize to our goal-conditioned scenario the estimates used by BPI-UCBVI (Ménard et al., 2021), a recent algorithm designed for Best-Policy Identification (BPI) in finite-horizon non-stationary MDPs. First, we build the optimistic goal-conditioned Q-values and value functions in the finite-horizon models $\overline{\mathcal{M}}_{g,H}$ for $g \in \mathcal{S}$ and $h \leq H$ as follows, $\widetilde{\overline{Q}}^0_{g,h}(s,a) \triangleq \mathbb{1}[s \neq g]$,

$$\widetilde{\overline{Q}}^k_{g,h}(s,a) \triangleq \mathrm{clip}\Bigg(\mathbb{1}[s \neq g] - 3\sqrt{\mathrm{Var}_{\widehat{P}^k}(\widetilde{\overline{V}}^k_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} - 14H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ - \frac{1}{H}\widehat{P}^k(\overline{V}^k_{g,h+1} - \widetilde{\overline{V}}^k_{g,h+1})(s,a) + \widehat{P}^k\widetilde{\overline{V}}^k_{g,h+1}(s,a),\ 0,\ H\Bigg),$$

$$\widetilde{\overline{V}}^k_{g,h}(s) \triangleq \min_{a \in \mathcal{A}} \widetilde{\overline{Q}}^k_{g,h}(s,a), \quad \widetilde{\overline{V}}^k_{g,h}(g) \triangleq 0, \quad \widetilde{\overline{V}}^k_{g,H+1}(s) \triangleq 0,$$

where we define the variance of $\widetilde{\overline{V}}^k_{g,h+1}$ with respect to $\widehat{P}^k(\cdot|s,a)$ as $\mathrm{Var}_{\widehat{P}^k}(\widetilde{\overline{V}}^k_{g,h+1})(s,a) \triangleq \sum_{s'} \widehat{P}^k(s'|s,a)(\widetilde{\overline{V}}^k_{g,h+1}(s') - \widehat{P}^k\widetilde{\overline{V}}^k_{g,h+1}(s,a))^2$, where the quantities $\beta(n,\delta) = \widetilde{O}(S\log(n/\delta))$ and $\beta^\star(n,\delta) = \widetilde{O}(\log(n/\delta))$ are some exploration thresholds, and $\underset{\sim}{\overline{V}}^k_g$ is a pessimistic finite-horizon goal-conditioned value function; see Appendix F.2.3 for the complete definitions. Let $\pi^{k+1}_{g,h}$ be the greedy policy with respect to the lower bounds $\widetilde{\overline{Q}}^k_{g,h}$. We recursively define the functions $\overline{U}^k_g$ for $g \in \mathcal{S}$ and $h \leq H$ as follows, $\overline{U}^0_{g,h}(s,a) \triangleq H\mathbb{1}[s \neq g]$,

$$\overline{U}^k_{g,h}(s,a) \triangleq \mathrm{clip}\Bigg(6\sqrt{\mathrm{Var}_{\widehat{P}^k}(\widetilde{\overline{V}}^k_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 36H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ + \left(1 + \frac{3}{H}\right)\widehat{P}^k\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a), 0, H\Bigg),$$

$$\overline{U}^k_{g,h}(g,a) \triangleq 0, \quad \overline{U}^k_{g,H+1}(s,a) \triangleq 0,$$

We are now ready to define the distance and error estimates of AdaGoal-UCBVI (Algorithm 8.1) as follows

$$\mathcal{Q}_{k,h}(s,a,g) \triangleq \widetilde{Q}^{k-1}_{g,h}(s,a), \tag{F.4}$$

$$\mathcal{D}_k(g) \triangleq \widetilde{V}^{k-1}_{g,1}(s_0), \tag{F.5}$$

$$\mathcal{E}_k(g) \triangleq \pi^k_{g,1}\overline{U}^{k-1}_{g,1}(s_0) + \frac{8\varepsilon}{9}. \tag{F.6}$$

F.2.3 Proof of Theorem 8.11

In this section, we provide the full proof of Theorem 8.11, by establishing the key steps followed in Section 8.3. First, we prove "key steps ①, ②, ③", which focuses on more "standard" technical tools (e.g., high-probability events, variance-aware concentration inequalities); in particular building on the analysis of Ménard et al. (2021) on the sample complexity of BPI in finite-horizon MDPs and extending it to our goal-conditioned scenario. Then we prove "key step ④", which focuses on the technical novelty of the AdaGoal goal selection scheme that is specific to the multi-goal exploration setting.

We begin by stating a simple property that we will rely on throughout the analysis.

Lemma F.5. *For any state-action pair $(s,a) \in \mathcal{S} \times \mathcal{A}$ and goal state $g \in \mathcal{S}$, consider any vector $Y \in \mathbb{R}^S$ such that $Y(g) = 0$, then $PY(s,a) = P_g Y(s,a)$, where we recall that*

$$P_g(s'|s,a) \triangleq \begin{cases} P(s'|s,a) & \text{if } s \neq g \\ \mathbb{1}[s' = g] & \text{if } s = g. \end{cases}$$

Proof. It is easy to see that

$$\begin{aligned} PY(s,a) &= \sum_{s' \in \mathcal{S}\setminus\{g\}} P(s'|s,a)Y(s') + P(g|s,a)\underbrace{Y(g)}_{=0} \\ &= \sum_{s' \in \mathcal{S}\setminus\{g\}} P_g(s'|s,a)Y(s') + P_g(g|s,a)\underbrace{Y(g)}_{=0} \\ &= P_g Y(s,a). \end{aligned}$$

□

Thanks to the above observation, our analysis will not require to handle goal-conditioned (true or empirical) transition probabilities, and will only need to deal with the (true or empirical) transition probabilities of the original MDP $\mathcal{M}$.

□ **Proof of "key step ①"**

Concentration events. Here we define the high-probability event $\mathcal{U}$ on which we condition our statements. We follow the notation of Ménard et al. (2021, Appendix A) and define the three following favorable events: $\mathcal{E}$ the event where the empirical transition probabilities are close to the true ones, $\mathcal{E}^{\mathrm{cnt}}$ the event where the pseudo-counts are close to their expectation, and $\mathcal{E}^\star$ where the empirical means of the optimal goal-conditioned value functions are close to the true ones. Denoting by KL the Kullback-Leibler divergence, we set

$$\begin{aligned}
\mathcal{E} &\triangleq \left\{ \forall k \in \mathbb{N}, \forall (s,a) \in \mathcal{S} \times \mathcal{A} : \ \mathrm{KL}\left(\widehat{p}^k(\cdot|s,a), P(\cdot|s,a)\right) \leq \frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \right\}, \\
\mathcal{E}^{\mathrm{cnt}} &\triangleq \left\{ \forall k \in \mathbb{N}, \forall (s,a) \in \mathcal{S} \times \mathcal{A} : \ n^k(s,a) \geq \frac{1}{2}\bar{n}^k(s,a) - \beta^{\mathrm{cnt}}(\delta) \right\}, \\
\mathcal{E}^\star &\triangleq \Bigg\{ \forall k \in \mathbb{N}, \forall h \in [H], \forall (s,a) \in \mathcal{S} \times \mathcal{A}, \forall g \in \mathcal{S} : \\
&\qquad |(\widehat{P}^k - p)\overline{V}^\star_{g,h+1}(s,a)| \leq \min\Bigg(H, \sqrt{2\mathrm{Var}_p(\overline{V}^\star_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} \\
&\qquad\qquad + 3H\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)} \Bigg) \Bigg\}.
\end{aligned}$$

We define the intersection of these events as

$$\mathcal{U} \triangleq \mathcal{E} \cap \mathcal{E}^{\mathrm{cnt}} \cap \mathcal{E}^\star. \tag{F.7}$$

We prove that for the right choice of the functions β the above event holds with high probability.

Lemma F.6. *For the following choices of functions β,*

$$\begin{aligned}
\beta(n,\delta) &\triangleq \log(3S^2AH/\delta) + S\log\left(8e(n+1)\right), \\
\beta^{\mathrm{cnt}}(\delta) &\triangleq \log\left(3S^2AH/\delta\right), \\
\beta^\star(n,\delta) &\triangleq \log(3S^2AH/\delta) + \log\left(8e(n+1)\right),
\end{aligned}$$

it holds that $\mathbb{P}(\mathcal{U}) \geq 1-\delta$.

Proof. The only difference with respect to the concentration inequalities of Ménard et al. (2021, Appendix A) is that we need to take a union bound over the goal states $g \in \mathcal{S}$ when concentrating our optimal goal-conditioned value functions. We thus set $\delta \leftarrow \delta/S$ in the choices of functions β compared to Ménard et al. (2021). As a result, by Ménard et al. (2021, Theorem 3

& 4 & 5) we have that $\mathbb{P}(\mathcal{E}) \geq 1 - \frac{\delta}{3}$, $\mathbb{P}(\mathcal{E}^{\mathrm{cnt}}) \geq 1 - \frac{\delta}{3}$ and $\mathbb{P}(\mathcal{E}^\star) \geq 1 - \frac{\delta}{3}$, respectively. Applying a union to the above three inequalities, we conclude that $\mathbb{P}(\mathcal{U}) \geq 1 - \delta$. □

We recall the definitions of the functions $\overline{U}_g^k$, $\widetilde{\overline{Q}}_{g,h}^k$ and $\widetilde{\overline{V}}_{g,h}^k$ in Section F.2.2. They rely on the pessimistic finite-horizon goal-conditioned values $\underset{\sim}{\overline{V}}_g^k$ defined as

$$\underset{\sim}{\overline{Q}}_{g,h}^k(s,a) \triangleq \mathrm{clip}\Bigg(\mathbb{1}[s \neq g] + 3\sqrt{\mathrm{Var}_{\widehat{P}^k}(\widetilde{\overline{V}}_{g,h+1}^k)(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 14H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ + \frac{1}{H}\widehat{P}^k(\underset{\sim}{\overline{V}}_{g,h+1}^k - \widetilde{\overline{V}}_{g,h+1}^k)(s,a) + \widehat{P}^k\underset{\sim}{\overline{V}}_{g,h+1}^k(s,a),\ 0,\ H\Bigg),$$

$$\underset{\sim}{\overline{V}}_{g,h}^k(s) \triangleq \min_{a\in\mathcal{A}} \underset{\sim}{\overline{Q}}_{g,h}^k(s,a), \quad \underset{\sim}{\overline{V}}_{g,h}^k(g) \triangleq 0, \quad \underset{\sim}{\overline{V}}_{g,H+1}^k(s) \triangleq 0.$$

Finally, we define the following quantities

$$\mathring{\overline{Q}}_{g,h}^k(s,a) \triangleq \max\Bigg(\mathbb{1}[s \neq g] + p\mathring{\overline{V}}_{g,h+1}(s,a), \\ \mathrm{clip}\Bigg(\mathbb{1}[s \neq g] + 3\sqrt{\mathrm{Var}_{\widehat{P}^k}(\widetilde{\overline{V}}_{g,h+1}^k)(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 14H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ + \frac{1}{H}\widehat{P}^k(\underset{\sim}{\overline{V}}_{g,h+1}^k - \widetilde{\overline{V}}_{g,h+1}^k)(s,a) + \widehat{P}^k\widetilde{\overline{V}}_{g,h+1}^k(s,a),\ 0,\ H\Bigg)\Bigg),$$

$$\mathring{\overline{V}}_{g,h}^k(s) \triangleq \pi_{g,h}^{k+1}\mathring{\overline{Q}}_{g,h}^k(s,a),$$

$$\mathring{\overline{V}}_{g,h}^k(g) \triangleq 0,$$

$$\mathring{\overline{V}}_{g,H+1}^k(s) \triangleq 0.$$

We have the following property, which is the equivalent of Ménard et al. (2021, Lemma 6) and is proved likewise.

Lemma F.7. *On the event $\mathcal{U}$, for all $(s,a,g,h) \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}\times[H]$ and for every episode k, it holds that*

$$\mathring{\overline{Q}}_{g,h}^k(s,a) \geq \max\left(\underset{\sim}{\overline{Q}}_{g,h}^k(s,a), \overline{Q}_{g,h}^{\pi_g^{k+1}}(s,a)\right),$$
$$\mathring{\overline{V}}_{g,h}^k(s) \geq \max\left(\underset{\sim}{\overline{V}}_{g,h}^k(s), \overline{V}_{g,h}^{\pi_g^{k+1}}(s)\right).$$

We now derive "key step ①" by establishing that Properties 1 and 2 hold. Specifically, we show that **(i)** the functions $\widetilde{\overline{V}}_{g,1}$ are optimistic estimates of the optimal goal-conditioned

finite-horizon value functions and **(ii)** the functions $\overline{U}_{g,1}$ serve as valid upper bounds to the goal-conditioned finite-horizon gaps, as shown below.

Lemma F.8. *On the event $\mathcal{U}$, it holds that for every episode k and goal $g \in \mathcal{S}$,*

$$\begin{aligned}\overline{V}_{g,1}^{\pi_{g,1}^{k+1}}(s_0) - \overline{V}_{g,1}^{\star}(s_0) &\leq \overline{V}_{g,1}^{\pi_{g,1}^{k+1}}(s_0) - \widetilde{\overline{V}}_{g,1}^{k}(s_0) \\ &\leq \pi_{g,1}^{k+1}\overline{U}_{g,1}^{k}(s_0).\end{aligned}$$

Proof. On the event $\mathcal{U}$, using Lemma F.7, we upper bound the goal-conditioned gap at episode t as

$$\overline{V}_{g,1}^{\pi_{g,1}^{k+1}}(s_0) - \overline{V}_{g,1}^{\star}(s_0) \leq \overline{V}_{g,1}^{\pi_{g,1}^{k+1}}(s_0) - \widetilde{\overline{V}}_{g,1}^{k}(s_0) \leq \mathring{\overline{V}}_{g,1}^{k}(s_0) - \widetilde{\overline{V}}_{g,1}^{k}(s_0).$$

Next, following the same reasoning as in Ménard et al. (2021, Proof of Lemma 2), we obtain by induction on h that for all state-action pairs (s,a) and goal states g,

$$\mathring{\overline{Q}}_{g,h}^{k}(s,a) - \widetilde{\overline{Q}}_{g,h}^{k}(s,a) \leq \overline{U}_{g,h}^{k}(s,a). \tag{F.8}$$

In particular for the initial layer $h = 1$ and initial state $s = s_0$, we get that

$$\mathring{\overline{V}}_{g,1}^{k}(s_0) - \widetilde{\overline{V}}_{g,1}^{k}(s_0) = \pi_{g,1}^{k+1}(\mathring{\overline{Q}}_{g,1}^{k} - \widetilde{\overline{Q}}_{g,1}^{k})(s_0) \leq \pi_{g,1}^{k+1}\overline{U}_{g,1}^{k}(s_0).$$

□

□ **Proof of "key step ②"**

Lemma F.9. *On the event $\mathcal{U}$, for every goal state $g \in \mathcal{S}$ and episode k, it holds that*

$$\begin{aligned}\pi_{g,1}^{k+1}\overline{U}_{g,1}^{k}(s_0) \leq\ & 24e^{13}H\sqrt{\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\frac{\beta^{\star}(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}} \\ &+ 336e^{13}H^2\sum_{s,a}\left[\sum_{h=1}^{H} p_{g,h}^{k+1}(s,a)\frac{\beta(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}\right]\wedge 1,\end{aligned}$$

where we recall that $p_{g,h}^{k+1}(s,a)$ denotes the probability of reaching (s,a) at step h under policy π_g^{k+1}.

Proof. Similar to Ménard et al. (2021, Steps 1 & 2 in proof of Theorem 2), we begin by upper-bounding $\overline{U}^k_{g,h}(s,a)$ for all (s,a,h,g,k). If $n^k(s,a) > 0$, by definition of $\overline{U}^k_{g,h}$ we have that

$$\overline{U}^k_{g,h}(s,a) \leq 6\sqrt{\mathrm{Var}_{\widehat{p}^k}(\widetilde{\overline{V}}^k_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 36H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \tag{F.9}$$

$$+\left(1+\frac{3}{H}\right)\widehat{p}^k\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a). \tag{F.10}$$

We now replace the empirical transition probabilities with the true ones. Using the Bernstein-type technical inequality of Ménard et al. (2021, Lemma 10) and that $0 \leq \overline{U}^k_{g,h} \leq H$, we get

$$\begin{aligned}(\widehat{P}^k - p)\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a) &\leq \sqrt{2\mathrm{Var}_p(\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1})(s,a)\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)}} + \frac{2}{3}H\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ &\leq \frac{1}{H}p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a) + 3H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)},\end{aligned}$$

where in the last line we used $\mathrm{Var}_p(\pi^{k+1}_{h+1}\overline{U}^k_{g,h+1})(s,a) \leq H\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a)$ and $\sqrt{xy} \leq x+y$ for all $x,y \geq 0$. We then replace the variance of the upper confidence bound under the empirical transition probabilities by the variance of the optimal value function under the true transition probabilities. Using the technical lemmas of Ménard et al. (2021, Lemma 11 & 12) that control the deviation in variances w.r.t. the choice of transition probabilities, we obtain that

$$\begin{aligned}\mathrm{Var}_{\widehat{P}^k}(\widetilde{\overline{V}}^k_{g,h+1})(s,a) &\leq 2\mathrm{Var}_p(\widetilde{\overline{V}}^k_{g,h+1})(s,a) + 4H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ &\leq 4\mathrm{Var}_p(\overline{V}^{\pi_g^{k+1}}_{g,h+1})(s,a) + 4Hp(\widetilde{\overline{V}}^k_{g,h+1} - \overline{V}^{\pi_g^{k+1}}_{g,h+1})(s,a) + 4H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ &\leq 4\mathrm{Var}_p(\overline{V}^{\pi_g^{k+1}}_{g,h+1})(s,a) + 4Hp\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a) + 4H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)},\end{aligned}$$

where we used (F.8) in the last inequality. Next, using $\sqrt{x+y} \leq \sqrt{x} + \sqrt{y}$, $\sqrt{xy} \leq x+y$, and $\beta^\star(n,\delta) \leq \beta(n,\delta)$ leads to

$$\begin{aligned}\sqrt{\mathrm{Var}_{\widehat{p}^k}(\widetilde{\overline{V}}^k_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} &\leq 2\sqrt{\mathrm{Var}_p(\overline{V}^{\pi_g^{k+1}}_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} \\ &\quad + (2H+4H^2)\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} + \frac{1}{H}p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a) \\ &\leq 2\sqrt{\mathrm{Var}_p(\overline{V}^{\pi_g^{k+1}}_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 6H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\ &\quad + \frac{1}{H}p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a).\end{aligned}$$

Combining these two inequalities with (F.10) yields

$$\begin{aligned}
\overline{U}^k_{g,h}(s,a) &\leq 12\sqrt{\mathrm{Var}_p(\overline{V}^{\pi^{k+1}_g}_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 36H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\
&\quad + \frac{6}{H}p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a) + 36H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\
&\quad + \left(1+\frac{3}{H}\right)\frac{1}{H}p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1} + \left(1+\frac{3}{H}\right)3H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\
&\quad + \left(1+\frac{3}{H}\right)p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a) \\
&\leq 12\sqrt{\mathrm{Var}_p(\overline{V}^{\pi^{k+1}_g}_{g,h+1})(s,a)\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}} + 84H^2\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)} \\
&\quad + \left(1+\frac{13}{H}\right)p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a).
\end{aligned}$$

Since by construction, $\overline{U}^k_{g,h}(s,a) \leq H$, we have that for all $n^k(s,a) \geq 0$,

$$\begin{aligned}
\overline{U}^k_{g,h}(s,a) &\leq 12\sqrt{\mathrm{Var}_p(\overline{V}^{\pi^{k+1}_g}_{g,h+1})(s,a)\left(\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1\right)} + 84H^2\left(\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1\right) \\
&\quad + \left(1+\frac{13}{H}\right)p\pi^{k+1}_{g,h+1}\overline{U}^k_{g,h+1}(s,a).
\end{aligned}$$

Unfolding the previous inequality and using $(1+13/H)^H \leq e^{13}$ we get

$$\begin{aligned}
\pi^{k+1}_{g,1}\overline{U}^k_{g,1}(s_0) &\leq 12e^{13}\sum_{h=1}^H\sum_{s,a}p^{k+1}_{g,h}(s,a)\sqrt{\mathrm{Var}_p(\overline{V}^{\pi^{k+1}_g}_{g,h+1})(s,a)\left(\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1\right)} \\
&\quad + 84e^{13}H^2\sum_{h=1}^H\sum_{s,a}p^{k+1}_{g,h}(s,a)\left(\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1\right).
\end{aligned}$$

Using that $\pi^{k+1}_{g,1}\overline{U}^k_{g,1}(s_0) \leq H$, we can clip the above bound as follows

$$\begin{aligned}
\pi^{k+1}_{g,1}\overline{U}^k_{g,1}(s_0) &\leq 12e^{13}\sum_{h=1}^H\sum_{s,a}p^{k+1}_{g,h}(s,a)\sqrt{\mathrm{Var}_p(\overline{V}^{\pi^{k+1}_g}_{g,h+1})(s,a)\left(\frac{\beta^\star(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1\right)} \\
&\quad + 84e^{13}H^2\sum_{s,a}\left[\sum_{h=1}^H p^{k+1}_{g,h}(s,a)\left(\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1\right)\right]\wedge 1. \qquad \text{(F.11)}
\end{aligned}$$

From the technical lemma of Ménard et al. (2021, Lemma 8) that relates counts to pseudo-counts,

$$\frac{\beta(n^k(s,a),\delta)}{n^k(s,a)}\wedge 1 \leq 4\frac{\beta(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1},$$

thus we can replace the counts by the pseudo-counts in (F.11) as

$$\pi_{g,1}^{k+1}\overline{U}_{g,1}^{k}(s_0) \le 24e^{13}\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\sqrt{\mathrm{Var}_p(\overline{V}_{g,h+1}^{\pi_g^{k+1}})(s,a)\frac{\beta^\star(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}} + 336e^{13}H^2\sum_{s,a}\left[\sum_{h=1}^{H} p_{g,h}^{k+1}(s,a)\frac{\beta(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}\right]\wedge 1. \tag{F.12}$$

We now apply the law of total variance (see e.g., Azar et al., 2017 or Ménard et al., 2021, Lemma 7) in order to further upper-bound the first sum in (F.12). In particular, by Cauchy-Schwarz inequality, we obtain

$$\begin{aligned}
&\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\sqrt{\mathrm{Var}_p(\overline{V}_{g,h+1}^{\pi_g^{k+1}})(s,a)\frac{\beta^\star(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}} \\
&\quad\le \sqrt{\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\mathrm{Var}_p(\overline{V}_{g,h+1}^{\pi_g^{k+1}})(s,a)}\sqrt{\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\frac{\beta^\star(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}} \\
&\quad\le \sqrt{\mathbb{E}_{\pi_g^{k+1}}\left[\left(\sum_{h=1}^{H}\mathbb{1}[s_h\neq g]-\overline{V}_{g,1}^{\pi_g^{k+1}}(s_0)\right)^2\right]}\sqrt{\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\frac{\beta^\star(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}} \\
&\quad\le H\sqrt{\sum_{h=1}^{H}\sum_{s,a} p_{g,h}^{k+1}(s,a)\frac{\beta^\star(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}}.
\end{aligned}$$

Plugging this in (F.12) concludes the proof of Lemma F.9. □

We are now ready to derive "key step ②" which controls the cumulative gap bounds, see Equation (8.4).

Lemma F.10. *On the event $\mathcal{U}$, for any number of episodes $K \ge 1$, it holds that*

$$\sum_{k=0}^{K-1}\pi_{g_{k+1},1}^{k+1}\overline{U}_{g_{k+1},1}^{k}(s_0) \le 48e^{13}\sqrt{K}\sqrt{H^2SA\log(HK+1)\beta^\star(K,\delta)} + 1344e^{13}H^2SA\beta(K,\delta)\log(HK+1) + 48e^{13}H^2SA\sqrt{\beta^\star(K,\delta)}.$$

Proof. Plugging in the bound of Lemma F.9 yields

$$\sum_{k=0}^{K-1}\pi_{g_{k+1},1}^{k+1}\overline{U}_{g_{k+1},1}^{k}(s_0)$$

$$
\begin{aligned}
&\leq 24e^{13}H\sum_{k=0}^{K-1}\sqrt{\sum_{h=1}^{H}\sum_{s,a}p^{k+1}_{g_{k+1},h}(s,a)\frac{\beta^\star(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}} \\
&\qquad + 336e^{13}H^2\sum_{k=0}^{K-1}\sum_{s,a}\left[\sum_{h=1}^{H}p^{k+1}_{g,h}(s,a)\frac{\beta(\bar{n}^k(s,a),\delta)}{\bar{n}^k(s,a)\vee 1}\right]\wedge 1 \\
&\leq 24e^{13}H\sqrt{\beta^\star(HK,\delta)}\sum_{k=0}^{K-1}\sqrt{\sum_{s,a}\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}} \\
&\qquad + 336e^{13}H^2\beta(HK,\delta)\sum_{s,a}\sum_{k=0}^{K-1}\left[\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}\right]\wedge 1,
\end{aligned}
$$

where we used that $\beta(.,\delta)$ and $\beta^\star(.,\delta)$ are increasing. We define $\mathcal{J}\triangleq\{k\in[0,K-1]:\bar{n}^k(s,a)<\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)-1\}$. Applying Lemma F.11 gives that

$$
\sum_{k\in\mathcal{J}}\sqrt{\sum_{s,a}\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}}\leq\sum_{s,a}\underbrace{\sum_{k\in\mathcal{J}}\sqrt{\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}}}_{\leq 2H}\leq 2SAH,
$$

$$
\sum_{k\notin\mathcal{J}}\sqrt{\sum_{s,a}\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}}\leq\sqrt{K}\sqrt{\sum_{s,a}\underbrace{\sum_{k\notin\mathcal{J}}\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}}_{\leq 4\log(HK+1)}}\leq 2\sqrt{KSA\log(HK+1)},
$$

$$
\underbrace{\sum_{s,a}\sum_{k=0}^{K-1}\left[\frac{\bar{n}^{k+1}(s,a)-\bar{n}^k(s,a)}{\bar{n}^k(s,a)\vee 1}\right]\wedge 1}_{\leq 4\log(HK+1)}\leq 4SA\log(HK+1).
$$

Putting everything together yields Lemma F.10. Note that as opposed to Ménard et al. (2021), we are in the setting of stationary transition probabilities (and cost functions), which is why we are able to shave a factor H in the main order term of the bound of the cumulative gap bounds (also recall that their sample complexity bound is in terms of exploration episodes and not exploration steps as ours). □

Lemma F.11 (Technical lemma). *For $T\in\mathbb{N}^\star$ and $(u_t)_{t\in\mathbb{N}^\star}$, for any sequence where $u_t\in[0,H]$ for some constant $H>0$ and $U_t\triangleq\sum_{l=1}^{t}u_\ell$, let $\Omega\triangleq\{t\in[0,T]:U_t<u_{t+1}-1\}$ and $\omega\triangleq\max\{t\in\Omega\}$. Then it holds that*

$$
\begin{aligned}
&\sum_{t\in\Omega}\sqrt{\frac{u_{t+1}}{U_t\vee 1}}\leq 2H,\\
&\sum_{t\notin\Omega}\frac{u_{t+1}}{U_t\vee 1}\leq 4\log(U_{T+1}+1),
\end{aligned}
$$

$$\sum_{t=0}^{T}\left[\frac{u_{t+1}}{U_t\vee 1}\right]\wedge 1\le 4\log(U_{T+1}+1).$$

Proof. First, note that for any $t\in\Omega$, $\frac{u_{t+1}}{U_t\vee 1}\ge 1$, therefore

$$\sum_{t\in\Omega}\sqrt{\frac{u_{t+1}}{U_t\vee 1}}\le\sum_{t\in\Omega}\frac{u_{t+1}}{U_t\vee 1}\le\sum_{t\in\Omega}u_{t+1}\le\sum_{t=0}^{\omega}u_{t+1}=U_{\omega+1}=U_\omega+u_\omega\le u_{\omega+1}-1+u_\omega\le 2H.$$

Second, if $t\notin\Omega$, then $2U_t+2\ge U_{t+1}+1$, therefore

$$\frac{u_{t+1}}{U_t\vee 1}\le 4\frac{u_{t+1}}{2U_t+2}\le 4\frac{U_{t+1}-U_t}{U_{t+1}+1},$$

which yields that

$$\sum_{t\notin\Omega}\frac{u_{t+1}}{U_t\vee 1}\le 4\sum_{t\notin\Omega}\frac{U_{t+1}-U_t}{U_{t+1}+1}\le 4\sum_{t=0}^{T}\int_{U_t}^{U_{t+1}}\frac{1}{x+1}\mathrm{d}x\le 4\log(U_{T+1}+1).$$

Third, combining the two cases above and noticing that $4\frac{U_{t+1}-U_t}{U_{t+1}+1}=\frac{4u_{t+1}}{U_t+u_{t+1}+1}\ge\frac{4u_{t+1}}{2u_{t+1}}\ge 1$ for all $t\in\Omega$, it holds that

$$\left[\frac{u_{t+1}}{U_t\vee 1}\right]\wedge 1\le 4\frac{U_{t+1}-U_t}{U_{t+1}+1},$$

thus

$$\sum_{t=0}^{T}\left[\frac{u_{t+1}}{U_t\vee 1}\right]\wedge 1\le 4\sum_{t=0}^{T}\frac{U_{t+1}-U_t}{U_{t+1}+1}\le 4\sum_{t=0}^{T}\int_{U_t}^{U_{t+1}}\frac{1}{x+1}\mathrm{d}x\le 4\log(U_{T+1}+1).$$

□

□ **Proof of "key step ③"**

Lemma F.12. *The* MGE *sample complexity* τ *of algorithm* AdaGoal-UCBVI *can be bounded with probability at least* $1-\delta$ *by*[1]

$$\tau=O\left(\frac{L^3SA}{\varepsilon^2}\cdot\log^3\left(\frac{LSA}{\varepsilon\delta}\right)\cdot\log^3\left(\frac{L}{\varepsilon}\right)+\frac{L^3S^2A}{\varepsilon}\cdot\log^3\left(\frac{LSA}{\varepsilon\delta}\right)\cdot\log^3\left(\frac{L}{\varepsilon}\right)\right).$$

Proof. We assume that the event $\mathcal{U}$ holds and fix a (finite) episode $K < \kappa$, where κ denotes the (possibly unbounded) episode index at which AdaGoal-UCBVI terminates. For any $k \leq K$, denote by g_k the goal selected by AdaGoal-UCBVI at the beginning of episode k, then by design of the stopping rule (8.1) and by choice of error $\mathcal{E}_k$ (F.6), it holds that

$$\varepsilon \leq \pi^k_{g_k,1} \overline{U}^{k-1}_{g_k,1}(s_0) + \frac{8\varepsilon}{9}\,.$$

By summing the previous inequality for all $k \leq K$ and plugging in the bound of Lemma F.10, we get that

$$\begin{aligned}\frac{\varepsilon}{9}K \leq\ & 48e^{13}\sqrt{K}\sqrt{H^2SA\log(HK+1)\beta^\star(K,\delta)} + 1344e^{13}H^2SA\beta(K,\delta)\log(HK+1) \\ & + 48e^{13}H^2SA\sqrt{\beta^\star(K,\delta)}.\end{aligned}$$

We assume that $\kappa > 1$ otherwise the result is trivially true. Defining $\kappa' \triangleq \min\{\kappa - 1, K\}$ and using the definition of β and $\beta^\star$ given in Lemma F.6, we get the following functional inequality in κ'

$$\varepsilon\kappa' \leq x_1\sqrt{\kappa' H^2SA\log(H\kappa')\big(\log(3S^2AH/\delta) + \log(16e\kappa')\big)} + x_2H^2S^2A\log(3S^2AH/\delta)\log^2(H\kappa'),$$

for some absolute constants x_1, x_2. There remains to invert the above inequality to obtain an upper bound on κ'. We use the auxiliary inequality of Lemma F.13 instantiated with scalars $B = x_1\sqrt{H^2SA\log(3S^2AH/\delta)}/\varepsilon$, $C = x_2H^2S^2A\log(3S^2AH/\delta)/\varepsilon$ and $\alpha = 16eH$. This yields that

$$\kappa' \leq O\left(\frac{H^2SA}{\varepsilon^2}\log^3\left(\frac{HSA}{\varepsilon\delta}\right) + \frac{H^2S^2A}{\varepsilon}\log^3\left(\frac{HSA}{\delta\varepsilon}\right)\right). \tag{F.13}$$

Since (F.13) holds for $\kappa' = \min\{\kappa - 1, K\}$ for any finite $K < \kappa$, letting $K \to +\infty$ implies that κ is finite and bounded as in (F.13).

The last step is to relate the above bound on the number of algorithmic episodes κ to the MGE sample complexity of AdaGoal-UCBVI denoted by τ. Since the algorithmic episodes are of length H and separated by a one-step execution of the reset action a_{reset}, it holds that $\tau \leq (H+1)\kappa$. We finally plug in the choice of horizon $H \triangleq \lceil 5(L+2)\log\left(10(L+2)/\varepsilon\right)/\log(2)\rceil$ to conclude the proof of Lemma F.12. □

Lemma F.13 (An auxiliary inequality). *For any positive scalars $B, C \geq 1$ and $\alpha \geq e$, it holds for any $X \geq 2$ that*

$$X \leq B\sqrt{X}\log(\alpha X) + C\log^2(\alpha X) \implies X \leq O(B^2\log^2(\alpha B) + C\log^2(\alpha C)).$$

Proof. On the one hand, assume that $X \leq B\sqrt{X}\log(\alpha X)$, then $\frac{X}{2} \leq -\frac{X}{2} + B\sqrt{X}\log(\alpha X)$. From the technical lemma of Kazerouni et al. (2017, Lemma 8), $-\frac{X}{2} + B\sqrt{X}\log(\alpha X) \leq \frac{32B^2}{9}\left[\log(4B\sqrt{\alpha}e)\right]^2$, thus $X \leq \frac{64B^2}{9}\left[\log(4B\sqrt{\alpha}e)\right]^2$. On the other hand, assume that $X \leq C\log^2(\alpha X)$. Using that $\log(x) \leq x^\beta/\beta$ for all $x \geq 0$, $\beta > 0$, we get $X \leq C(8\alpha^{1/8}X^{1/8})^2 \leq 64C\alpha^{1/4}X^{1/4}$, thus $X \leq (64C)^{4/3}\alpha^{1/3}$, thus $X \leq C\log^2(64\alpha^{4/3}C^{4/3})$. Now, assume that $X \leq B\sqrt{X}\log(\alpha X) + C\log^2(\alpha X)$. Then $X \leq 2\max\{B\sqrt{X}\log(\alpha X), C\log^2(\alpha X)\}$. From above we can bound each term separately, which concludes the proof. □

□ **Proof of "key step ④"** We finally establish "key step ④", which focuses on the technical novelty of the AdaGoal goal selection scheme that is specific to the multi-goal exploration setting.

Recall for any $H \in \mathbb{N}^*$ the definition of the finite-horizon MDP $\overline{\mathcal{M}}_{g,H} = \{H, S, A, \overline{P}_g, \overline{c}_g\}$, where we recall that $\overline{P}_g = P_g$ and $\overline{c}_g = c_g$. We denote by $\overline{\pi}^\star_{g,H}$ the optimal policy in $\overline{\mathcal{M}}_{g,H}$ as well as $\overline{V}_{g,H,h}(s)$ the optimal value function starting from state s at step h. We also define $\overline{P}^{\overline{\pi}_{g,H}}_{g,H}(s_H \neq g | s_1 = s)$ the probability of reaching state g starting from state s with the policy $\pi \in \Pi_H$ in the MDP $\overline{\mathcal{M}}_{g,H}$. When it is clear from the context we drop the dependence on the horizon H in the previous notations.

The following lemma controls the probability of not reaching a goal in $\mathcal{G}_{L+\varepsilon}$ with the optimal policy in the finite-horizon reduction MDP.

Lemma F.14. *For $g \in \mathcal{G}_{L+\varepsilon}$, for all $H \geq 2(L+2)$, for all $s \in \mathcal{S}$,*

$$\overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_1 = s) \leq e^{-\log(2)H/\left(4(L+2)\right)}.$$

Proof. By induction it holds

$$\begin{aligned}
\overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_1 = s) &= \sum_{s' \neq g} \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_{H-M+1} = s' | s_1 = s) \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_{H-M+1} = s') \\
&\leq \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_{H-M+1} \neq g | s_1 = s) \max_{s'} \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_{H-M+1} = s') \\
&\leq \prod_{j=0}^{\lfloor H/M \rfloor} \max_{s'} \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_{H-jM} \neq g | s_{H-(j+1)M+1} = s').
\end{aligned}$$

Then thanks to the Markov inequality and the optimal Bellman equations solved by $\overline{\pi}^\star_{g,H}$ we obtain

$$\begin{aligned}\overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_{H-jM} \neq g | s_{H-(j+1)M+1} = s') &\leq \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_{H-(j+1)M+1} = s') \\ &\leq \frac{\overline{V}^\star_{g,H,H-(j+1)+1}(s')}{M} \\ &= \frac{\overline{V}^\star_{g,(j+1)M,1}(s')}{M} \\ &\leq \frac{\overline{V}^\star_{g,(j+1)M,1}(s')}{M} \leq \frac{V^\star_g(s')}{M} \leq \frac{L+2}{M}\,,\end{aligned}$$

where the last inequality uses the existence of a resetting action (Assumption 8.3) and the fact that $g \in \mathcal{G}_{L+\varepsilon}$ with $\varepsilon \leq 1$. Choosing $M = 2(L+2)$ allows us to conclude

$$\overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_1 = s) \leq e^{-\lfloor H/M \rfloor \log(2)} \leq e^{-\log(2)H/\left(4(L+2)\right)}\,,$$

where in the last inequality we used $\lfloor x \rfloor \geq x/2$ for $x \geq 1$. □

We define the class of non-stationary, infinite-horizon policies that perform the reset action whenever the goal state is not reached after H steps.

Definition F.15 (Resetting policies)**.** *For any π, we denote by $\pi^{|H}$ the non-stationary policy that, until the goal is reached, successively executes the actions prescribed by π for H steps and takes action a_{reset}, i.e., at time step i and state s it executes the following action:*

$$\pi^{|H}(a|s,i) \triangleq \begin{cases} a_{\text{reset}} & \text{if } i \equiv 0 \ (\text{mod } H+1), \\ \pi(a|s,i) & \text{otherwise.} \end{cases}$$

We denote by $\Pi^{|H}$ the set of such resetting policies.

We now establish two key lemmas. First, we show that, equipped with a near-optimal policy for the finite-horizon model $\overline{\mathcal{M}}_{g,H}$, expanding it into an infinite-horizon policy via the reset provides a near-optimal goal-reaching policy in the original MDP $\mathcal{M}_g$ *as long as the goal state g belongs to $\mathcal{G}_{O(L)}$ and the horizon H is large enough.*

Lemma F.16. *For $g \in \mathcal{G}_{L+\varepsilon}$ and $H \geq 5(L+2)\log\left(10(L+2)/\varepsilon\right)/\log(2)$, it holds that*

$$\overline{V}^\star_{g,1}(s_0) \leq V^\star_g(s_0) \leq \overline{V}^\star_{g,1}(s_0) + \frac{\varepsilon}{9}\,,$$

and if a policy $\widetilde{\pi}$ is $\varepsilon/9$-optimal in $\overline{\mathcal{M}}_{g,H}$ then

$$V_g^{\widetilde{\pi}|H}(s_0) \leq V_g^\star(s_0) + \varepsilon\,.$$

Proof. We have trivially $\overline{V}^\star_{g,H,1}(s_0) \leq V_g^\star(s_0)$. Thanks to Lemma F.14 it holds

$$\overline{q}^\star_{g,H} \triangleq \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_{H+1} \neq g | s_1 = s_0) \leq \overline{P}^{\overline{\pi}^\star_{g,H}}_{g,H}(s_H \neq g | s_1 = s_0) \leq \frac{\varepsilon}{10(L+2)} \leq \frac{1}{30}\,. \tag{F.14}$$

Thanks to (F.14) and the definition of a resetting policy, we can conclude that

$$\begin{aligned}
V_g^\star(s_0) \leq V_g^{\overline{\pi}^\star_{g,H}|H}(s_0) &= \overline{V}^\star_{g,H,1}(s_0) + \overline{q}^\star_{g,H}(1 + V_g^{\overline{\pi}^\star_{g,H}|H}(s_0)) \\
&= \overline{V}^\star_{g,H,1}(s_0) + \frac{\overline{q}^\star_{g,H}}{1-\overline{q}^\star_{g,H}}(1 + \overline{V}^\star_{g,H,1}(s_0)) \\
&\leq \overline{V}^\star_{g,H,1}(s_0) + \frac{30}{29}\frac{\varepsilon}{10(L+2)}(1 + L + \varepsilon) \\
&\leq \overline{V}^\star_{g,H,1}(s_0) + \frac{\varepsilon}{9}\,.
\end{aligned}$$

Thus it holds that

$$\overline{V}^\star_{g,H,1}(s_0) \leq V_g^\star(s_0) \leq \overline{V}^\star_{g,H,1}(s_0) + \frac{\varepsilon}{9}\,. \tag{F.15}$$

For the second part of the lemma, first note that

$$\overline{V}^{\widetilde{\pi}}_{g,H,1}(s_0) = \sum_{h=1}^{H} \overline{P}^{\widetilde{\pi}}_{g,H}(s_h \neq g | s_1 = s_0) \geq \overline{V}^{\widetilde{\pi}}_{g,H-L,1}(s_0) + L\overline{P}^{\widetilde{\pi}}_{g,H}(s_H \neq g | s_1 = s_0)\,,$$

where in the inequality we used that $\overline{P}^{\widetilde{\pi}}_{g,H}(s_h \neq g | s_1 = s_0) \geq \overline{P}^{\widetilde{\pi}}_{g,H}(s_H \neq g | s_1 = s_0)$. Using successively the fact that $\widetilde{\pi}$ is $\frac{\varepsilon}{9}$-optimal in $\overline{\mathcal{M}}_{g,H}$, the inequality above and (F.15) we obtain

$$\begin{aligned}
V_g^\star(s_0) + \frac{\varepsilon}{9} &\geq \overline{V}^\star_{g,H,1}(s_0) + \frac{\varepsilon}{9} \\
&\geq \overline{V}^{\widetilde{\pi}}_{g,H,1}(s_0) \\
&\geq \overline{V}^{\widetilde{\pi}}_{g,H-L,1}(s_0) + L\overline{P}^{\widetilde{\pi}}_{g,H}(s_H \neq g | s_0) \\
&\geq V_g^\star(s_0) - \frac{\varepsilon}{9} + L\overline{P}^{\widetilde{\pi}}_{g,H}(s_H \neq g | s_0)\,.
\end{aligned}$$

The previous sequence of inequalities entails that $\overline{P}^{\widetilde{\pi}}_{g,H}(s_H \neq g) \leq (2\varepsilon)/(9L)$. Now we can upper bound the value of the resetting extension of $\widetilde{\pi}$. Indeed, for $\widetilde{q} \triangleq \overline{P}^{\widetilde{\pi}}_{g,H}(s_{H+1} \neq g | s_1 =$

$s_0) \leq \overline{P}^{\widetilde{\pi}}_{g,H}(s_H \neq g | s_1 = s_0)$ we have using that $\widetilde{\pi}$ is $\frac{\varepsilon}{9}$-optimal in $\overline{\mathcal{M}}_{g,H}$ with $g \in \mathcal{G}_{L+\varepsilon}$ that

$$\begin{aligned} V_g^{\widetilde{\pi}|H}(s_0) &= \overline{V}^{\widetilde{\pi}}_{g,H,1}(s_0) + \frac{\widetilde{q}}{1-\widetilde{q}}(1 + \overline{V}^{\widetilde{\pi}}_{g,H,1}(s_0)) \\ &\leq \overline{V}^{\star}_{g,H,1}(s_0) + \frac{\varepsilon}{9} + \frac{2\varepsilon}{9L}\frac{1}{1-2/9}\left(1 + L + \varepsilon + \frac{\varepsilon}{9}\right) \\ &\leq V_g^{\star}(s_0) + \varepsilon\,. \end{aligned}$$

□

The second key lemma that we prove is that any goal state that meets the constraint (8.2b) with small enough error (8.2a) must belong to $\mathcal{G}_{L+\varepsilon}$.

Restatement of Lemma 8.16. *With probability at least $1-\delta$, if a goal state $g \in \mathcal{G}$ satisfies $\mathcal{D}_k(g) \leq L$ and $\mathcal{E}_k(g) \leq \varepsilon$ for an episode $k \geq 1$, then $g \in \mathcal{G}_{L+\varepsilon}$.*

Proof. Consider that the event $\mathcal{U}$ defined in (F.7) holds. Consider a goal state g such that $\mathcal{D}_k(g) \leq L$ and $\mathcal{E}_k(g) \leq \varepsilon$ at an episode $k \geq 1$. Then

$$\begin{aligned} \overline{V}^{\star}_{g,H,1}(s_0) &\overset{\text{(i)}}{\leq} \widetilde{\overline{V}}^{k}_{g,H,1}(s_0) + \pi^{k+1}_{g,1}\overline{U}^{k}_{g,1}(s_0) \\ &\overset{\text{(ii)}}{=} \mathcal{D}_k(g) + \mathcal{E}_k(g) - \frac{8\varepsilon}{9} \\ &\overset{\text{(iii)}}{\leq} L + \frac{\varepsilon}{9}, \end{aligned} \tag{F.16}$$

where (i) comes from Lemma F.8, (ii) stems from the choice of $\mathcal{D}$ and $\mathcal{E}$ estimates and (iii) comes from the conditions on g. Following the steps of the proof of Lemma F.14, we have that

$$\overline{P}^{\overline{\pi}^{\star}_{g,H}}_{g,H}(s_H \neq g | s_1 = s) \leq \prod_{j=0}^{\lfloor H/M \rfloor} \max_{s'} \overline{P}^{\overline{\pi}^{\star}_{g,H}}_{g,H}(s_{H-jM} \neq g | s_{H-(j+1)M+1} = s'),$$

and that

$$\begin{aligned} \overline{P}^{\overline{\pi}^{\star}_{g,H}}_{g,H}(s_{H-jM} \neq g | s_{H-(j+1)M+1} = s') &\leq \frac{\overline{V}^{\star}_{g,(j+1)M,1}(s')}{M} \\ &\leq \frac{\overline{V}^{\star}_{g,H,1}(s')}{M} \\ &\leq \frac{1 + \overline{V}^{\star}_{g,H,1}(s_0)}{M} \\ &\leq \frac{1 + L + \varepsilon/9}{M}, \end{aligned}$$

where the before last inequality uses the existence of a resetting action (Assumption 8.3) and the last inequality uses (F.16). Choosing $M = 2(L+2)$ gives

$$\overline{P}_{g,H}^{\overline{\pi}^\star_{g,H}}(s_H \neq g | s_1 = s) \leq e^{-\log(2)H/\left(4(L+2)\right)}. \tag{F.17}$$

We now follow the steps of the proof of Lemma F.16. Thanks to (F.17) and the choice of H it holds

$$\overline{q}^\star_{g,H} \triangleq \overline{P}_{g,H}^{\overline{\pi}^\star_{g,H}}(s_{H+1} \neq g | s_1 = s_0) \leq \overline{P}_{g,H}^{\overline{\pi}^\star_{g,H}}(s_H \neq g | s_1 = s_0) \leq \frac{\varepsilon}{10(L+2)} \leq \frac{1}{30}. \tag{F.18}$$

Thanks to (F.18) and the definition of a resetting policy, we obtain that

$$\begin{aligned}
V_g^\star(s_0) \leq V_g^{\overline{\pi}^\star_{g,H}|^H}(s_0) &= \overline{V}^\star_{g,H,1}(s_0) + \overline{q}^\star_{g,H}(1 + V_g^{\overline{\pi}^\star_{g,H}|^H}(s_0)) \\
&= \overline{V}^\star_{g,H,1}(s_0) + \frac{\overline{q}^\star_{g,H}}{1-\overline{q}^\star_{g,H}}(1 + \overline{V}^\star_{g,H,1}(s_0)) \\
&\overset{\text{(i)}}{\leq} L + \frac{\varepsilon}{9} + \frac{30}{29}\frac{\varepsilon}{10(L+2)}(1 + L + \frac{\varepsilon}{9}) \\
&\leq L + \frac{2\varepsilon}{9},
\end{aligned}$$

where (i) uses (F.16). Therefore we have that $g \in \mathcal{G}_{L+\varepsilon}$, which concludes the proof. □

We now have all the tools to prove that when AdaGoal-UCBVI terminates, it fulfills the MGE objective of Definition 8.4.

Lemma F.17. *If the algorithm* AdaGoal-UCBVI *stops, it is* (ε, δ, L)*-PAC for* MGE.

Proof. Consider that the event $\mathcal{U}$ defined in (F.7) holds, and that the algorithm AdaGoal-UCBVI has stopped at episode κ. Recall that we define $\mathcal{D}_\kappa(g) \triangleq \widetilde{\overline{V}}^\kappa_{g,1}(s_0)$ and $\mathcal{E}_\kappa(g) \triangleq \pi^{\kappa+1}_{g,1}\overline{U}^\kappa_{g,1}(s_0) + \frac{8\varepsilon}{9}$. Denoting $\mathcal{X}_\kappa \triangleq \{g \in \mathcal{S} : \mathcal{D}_\kappa(g) \leq L\}$, the stopping rule (Equation (8.1)) implies that $\max_{g\in\mathcal{X}_\kappa} \mathcal{E}_\kappa(g) \leq \varepsilon$. We now prove that $\mathcal{G}_L \subseteq \mathcal{X}_\kappa \subseteq \mathcal{G}_{L+\varepsilon}$. On the one hand, it holds that

$$\mathcal{D}_\kappa(g) = \widetilde{\overline{V}}^\kappa_{g,1}(s_0) \leq \overline{V}^\star_{g,1}(s_0) \leq V^\star_g(s_0),$$

which ensures that $\mathcal{G}_L \subseteq \mathcal{X}_\kappa$. On the other hand, consider that $g \in \mathcal{X}_\kappa$, then $\mathcal{D}_\kappa(g) \leq L$ and $\mathcal{E}_\kappa(g) \leq \varepsilon$, which implies that $g \in \mathcal{G}_{L+\varepsilon}$ from Lemma 8.16, therefore $\mathcal{X}_\kappa \subseteq \mathcal{G}_{L+\varepsilon}$.

We now prove that the candidate policies of AdaGoal-UCBVI are near-optimal goal-reaching policies. Consider any $g \in \mathcal{X}_\kappa$. Combining the result of Lemma F.8 and Equation (8.1), we

obtain that

$$\overline{V}_{g,1}^{\pi_{g,1}^{\kappa+1}}(s_0) - \overline{V}_{g,1}^{\star}(s_0) \leq \pi_{g,1}^{\kappa+1}\overline{U}_{g,1}^{\kappa}(s_0) \leq \mathcal{E}_\kappa(g) - \frac{8\varepsilon}{9} \leq \frac{\varepsilon}{9},$$

thus the policy $\pi_g^{\kappa+1}$ is $\frac{\varepsilon}{9}$-optimal in $\overline{\mathcal{M}}_{g,H}$. As a result, denoting by $\widehat{\pi}_g \triangleq (\pi_g^{\kappa+1})^{|H}$ the candidate policy of AdaGoal-UCBVI, we have from Lemma F.16 that

$$V_g^{\widehat{\pi}_g}(s_0) \leq V_g^{\star}(s_0) + \varepsilon\,,$$

i.e., $\widehat{\pi}_g$ is ε-optimal for the original SSP objective. Putting everything together, we have that

$$\mathbb{P}\left(\{\mathcal{G}_L \subseteq \mathcal{X}_\kappa \subseteq \mathcal{G}_{L+\varepsilon}\} \cap \left\{\forall g \in \mathcal{X}_\kappa,\ V^{\widehat{\pi}_g}(s_0 \to g) - V^{\star}(s_0 \to g) \leq \varepsilon\right\}\right) \geq \mathbb{P}(\mathcal{U}) \geq 1 - \delta.$$

which ensures that AdaGoal-UCBVI is (ε, δ, L)-PAC for MGE. □

F.2.4 Putting everything together

Restatement of Theorem 8.11. AdaGoal-UCBVI *is* $(\varepsilon, \delta, L, \mathcal{S})$*-PAC for* MGE *and, with probability at least* $1-\delta$*, for* $\varepsilon \in (0, 1/S]$ *its* MGE *sample complexity is of order*[2] $\widetilde{O}(L^3SA\varepsilon^{-2})$.

Proof. The result comes from combining Lemmas F.12 and F.17. □

F.3 Details of AdaGoal-UCRL·VTR and Analysis

In this section, we provide details on the AdaGoal-UCRL·VTR algorithm and the guarantee of Theorem 8.14 which bounds its MGE sample complexity in linear mixture MDPs. Recall that since the state space $\mathcal{S}$ may be large, we consider that the known goal space is in all generality a subset of it, i.e., $\mathcal{G} \subseteq \mathcal{S}$, where $G \triangleq |\mathcal{G}|$ denotes the cardinality of the goal space.

First of all, we extend the linear mixture definition (Definition 8.10) to handle our multi-goal setting. For any goal $g \in \mathcal{G}$, we define

$$P_g(s'|s,a) \triangleq \langle \phi_g(s'|s,a), \theta_g^\star \rangle,$$

where

$$\theta_g^\star \triangleq \begin{pmatrix} \theta^\star \\ 1 \end{pmatrix} \in \mathbb{R}^{d+1}, \qquad \phi_g(s'|s,a) \triangleq \begin{pmatrix} \mathbb{1}[s \neq g]\phi(s'|s,a) \\ \mathbb{1}[s=g]\mathbb{1}[s'=g] \end{pmatrix} \in \mathbb{R}^{d+1}.$$

[2]The notation $\widetilde{O}$ in Theorem 8.11 hides poly-log terms in $\varepsilon^{-1}, S, A, L, \delta^{-1}$. See Lemma F.12 in Appendix F.2.3 for a more detailed bound that includes the poly-log terms.

We see that by construction,

$$P_g(s'|s,a) = \begin{cases} P(s'|s,a) & \text{if } s \neq g \\ \mathbb{1}[s'=g] & \text{if } s = g. \end{cases}$$

F.3.1 Overview of AdaGoal-UCRL·VTR and Choice of $\mathcal{E}$, $\mathcal{D}$, Q in line 14 of Algorithm 8.1

Here we focus on the specificities of AdaGoal-UCRL·VTR in the linear mixture MDP setting (refer to Section 8.2 for the description of the algorithmic structure that is common to AdaGoal-UCBVI), i.e., we explain how to define the estimates $\mathcal{D}$, $\mathcal{E}$, $\mathcal{Q}$ in line 14 of Algorithm 8.1. At a high level, AdaGoal-UCRL·VTR uses two regression-based goal-conditioned estimators of the unknown parameter vector $\boldsymbol{\theta}_g^\star$ of each goal $g \in \mathcal{G}$:

- *Value-targeted estimator.* The first estimator minimizes a ridge regression problem with the target being the past value functions. This is similar to the UCRL-VTR algorithm for linear mixture MDPs (Ayoub et al., 2020) and follow-up work (e.g., Zhou et al., 2021; Zhang et al., 2021b). This step is used to compute the distance estimates $\mathcal{D}$ (and $\mathcal{Q}$) for AdaGoal.
- *Error-targeted estimator.* The second estimator is novel and minimizes a ridge regression problem with the target being past "error functions", that are computable upper bounds on the goal-conditioned gaps. This step is used to compute the errors $\mathcal{E}$ for AdaGoal.

▷ *Value-targeted estimator.* First, AdaGoal-UCRL·VTR builds a goal-conditioned estimator $\boldsymbol{\theta}_g$ for the unknown parameter vector $\boldsymbol{\theta}_g^\star$ of each goal $g \in \mathcal{G}$, as well as a goal-conditioned covariance matrix $\boldsymbol{\Sigma}_g$ of the feature mappings, which characterizes the uncertainty of the estimator $\boldsymbol{\theta}_g$. Similar to UCRL-VTR, $\boldsymbol{\theta}_g$ is computed as the minimizer to a ridge regression problem with the target being the past value functions, i.e.,

$$\boldsymbol{\theta}_{g,k+1} \leftarrow \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\mathbb{R}^{d+1}} \lambda\|\boldsymbol{\theta}\|^2 + \sum_{k'=1}^{k}\sum_{h=1}^{H}\left(\langle \boldsymbol{\theta}, \boldsymbol{\phi}_{V_{g,k',h}}(s_h^{k'}, a_h^{k'})\rangle - V_{g,k',h}(s_{h+1}^{k'})\right),$$

which has a closed-form solution given in (F.20). Leveraging $\boldsymbol{\theta}_g$ and subtracting an exploration bonus term, AdaGoal-UCRL·VTR builds optimistic goal-conditioned estimators $Q_{g,k,h}(\cdot,\cdot)$ (F.22) and $V_{g,k,h}(\cdot)$ (F.24) for the optimal action-value and value functions $\overline{Q}^\star_{g,h}(\cdot,\cdot)$ and $\overline{V}^\star_{g,h}(\cdot)$. The associated goal-conditioned policy is the greedy policy of the calculated optimistic Q-values (F.23).

▷ *Error-targeted estimator.* The main addition compared to existing works on linear mixture MDPs is that AdaGoal-UCRL·VTR also builds goal-conditioned errors denoted by $U_{g,k,h}$ (F.25) that upper bound the goal-conditioned gaps (defined as the difference between the value function of the current greedy policy and the optimistic value estimates). They rely on an additional

estimator $\mathring{\boldsymbol{\theta}}_{g,k}$ and covariance matrix $\mathring{\boldsymbol{\Sigma}}_{g,k}$ based on the errors $\{U_{g,k',h}\}_{k'\le k-1,h}$, instead of the values $\{V_{g,k',h}\}_{k'\le k-1,h}$ as considered before. Specifically, $\mathring{\boldsymbol{\theta}}_g$ minimizes the ridge regression problem with contexts $\boldsymbol{\phi}_{U_{g,k',h}}(s_h^{k'}, a_h^{k'})$ and targets $U_{g,k',h}(s_{h+1}^{k'})$, i.e.,

$$\mathring{\boldsymbol{\theta}}_{g,k+1} \leftarrow \operatorname*{arg\,min}_{\boldsymbol{\theta}\in\mathbb{R}^{d+1}} \lambda\|\boldsymbol{\theta}\|^2 + \sum_{k'=1}^{k}\sum_{h=1}^{H}\left(\langle \boldsymbol{\theta}, \boldsymbol{\phi}_{U_{g,k',h}}(s_h^{k'}, a_h^{k'})\rangle - U_{g,k',h}(s_{h+1}^{k'})\right),$$

which has a closed-form solution given in (F.21).

▷ *Algorithmic notation and updates.*

Let B be an upper bound of the ℓ_2-norm of $\boldsymbol{\theta}^\star$ (see Definition 8.10) and set as regularization parameter $\lambda \triangleq 1/(B+1)^2$. Also define the confidence radius

$$\beta_k \triangleq H\sqrt{d\log(3(1+kH^3(B+1)^2)/\delta)} + 1. \tag{F.19}$$

At the first episode indexed by $k = 1$, we initialize for every goal $g \in \mathcal{G}$ and $h \in [H]$ the following quantities

$$\boldsymbol{\Sigma}_{g,1,h}, \mathring{\boldsymbol{\Sigma}}_{g,1,h} \triangleq \lambda \boldsymbol{I}, \quad \boldsymbol{b}_{g,1,h}, \mathring{\boldsymbol{b}}_{g,1,h} \triangleq \boldsymbol{0}, \quad \boldsymbol{\theta}_{g,1}, \mathring{\boldsymbol{\theta}}_{g,1,h} \triangleq \boldsymbol{0}, \quad V_{g,1,H+1}(\cdot) \triangleq 0, \quad U_{g,1,H+1}(\cdot) \triangleq 0.$$

We now explain how the various estimates are updated during an episode k with goal state denoted by g_k. Over the trajectory of episode k, given the current state visited at step h denoted by s_h^k, the executed action is denoted by $a_h^k \triangleq \pi_{g_k,h}^k(s_h^k)$ and the next state is denoted by s_{h+1}^k. Then for every goal $g \in \mathcal{G}$ and for $h = 1, \dots, H$, we set

$$\begin{aligned}
\boldsymbol{\Sigma}_{g,k,h+1} &\triangleq \boldsymbol{\Sigma}_{g,k,h} + \boldsymbol{\phi}_{V_{g,k,h}}(s_h^k, a_h^k)\boldsymbol{\phi}_{V_{g,k,h}}(s_h^k, a_h^k)^\top,\\
\boldsymbol{b}_{g,k,h+1} &\triangleq \boldsymbol{b}_{g,k,h} + \boldsymbol{\phi}_{V_{g,k,h}}(s_h^k, a_h^k)V_{g,k,h}(s_{h+1}^k),\\
\mathring{\boldsymbol{\Sigma}}_{g,k,h+1} &\triangleq \mathring{\boldsymbol{\Sigma}}_{g,k,h} + \boldsymbol{\phi}_{U_{g,k,h}}(s_h^k, a_h^k)\boldsymbol{\phi}_{U_{g,k,h}}(s_h^k, a_h^k)^\top,\\
\mathring{\boldsymbol{b}}_{g,k,h+1} &\triangleq \mathring{\boldsymbol{b}}_{g,k,h} + \boldsymbol{\phi}_{U_{g,k,h}}(s_h^k, a_h^k)U_{g,k,h}(s_{h+1}^k),
\end{aligned}$$

and for every goal $g \in \mathcal{G}$, we set

$$\boldsymbol{\Sigma}_{g,k+1,1} \triangleq \boldsymbol{\Sigma}_{g,k,H+1}, \quad \boldsymbol{b}_{g,k+1,1} \triangleq \boldsymbol{b}_{g,k,H+1}, \quad \boldsymbol{\theta}_{g,k+1} \triangleq \boldsymbol{\Sigma}_{g,k+1,1}^{-1}\boldsymbol{b}_{g,k+1,1}, \tag{F.20}$$

$$\mathring{\boldsymbol{\Sigma}}_{g,k+1,1} \triangleq \mathring{\boldsymbol{\Sigma}}_{g,k,H+1}, \quad \mathring{\boldsymbol{b}}_{g,k+1,1} \triangleq \mathring{\boldsymbol{b}}_{g,k,H+1}, \quad \mathring{\boldsymbol{\theta}}_{g,k+1} \triangleq \mathring{\boldsymbol{\Sigma}}_{g,k+1,1}^{-1}\mathring{\boldsymbol{b}}_{g,k+1,1}. \tag{F.21}$$

We proceed by recursively defining for every episode k, goal $g \in \mathcal{G}$ and $h = H, \dots, 1$,

$$Q_{g,k,h}(\cdot,\cdot) \triangleq \operatorname{clip}\left(\mathbb{1}[\cdot \neq g] + \langle \boldsymbol{\theta}_{g,k}, \boldsymbol{\phi}_{V_{g,k,h+1}}(\cdot,\cdot)\rangle - \beta_k\left\|\boldsymbol{\Sigma}_{g,k,1}^{-1/2}\boldsymbol{\phi}_{V_{g,k,h+1}}(\cdot,\cdot)\right\|_2, 0, H\right), \tag{F.22}$$

$$\pi_{g,h}^k(\cdot) \triangleq \operatorname*{arg\,min}_{a\in\mathcal{A}} Q_{g,k,h}(\cdot, a), \tag{F.23}$$

$$V_{g,k,h}(\cdot) \triangleq \min_{a \in \mathcal{A}} Q_{g,k,h}(\cdot, a), \tag{F.24}$$

$$U_{g,k,h}(\cdot) \triangleq \mathrm{clip}\Big(2\beta_k \Big\| \boldsymbol{\Sigma}_{g,k,1}^{-1/2} \boldsymbol{\phi}_{V_{g,k,h+1}}(s, \pi_{g,h}^k(s)) \Big\|_2 + \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi_{g,h}^k(s)), \mathring{\boldsymbol{\theta}}_{g,k} \rangle + \beta_k \Big\| \mathring{\boldsymbol{\Sigma}}_{g,k,1}^{-1/2} \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi_{g,h}^k(s)) \Big\|_2, 0, H\Big). \tag{F.25}$$

▷ *Choice of estimates $\mathcal{D}$, $\mathcal{E}$, $\mathcal{Q}$ of* AdaGoal-UCRL·VTR *(line 14 of Algorithm 8.1):*

$$\mathcal{Q}_{k,h}(s, a, g) \triangleq Q_{g,k,h}(s, a), \tag{F.26}$$

$$\mathcal{D}_k(g) \triangleq V_{g,k,1}(s_0), \tag{F.27}$$

$$\mathcal{E}_k(g) \triangleq U_{g,k,1}(s_0) + \frac{8\varepsilon}{9}. \tag{F.28}$$

F.3.2 Proof sketch of Theorem 8.14

The analysis of AdaGoal-UCRL·VTR follows the same key steps considered in Section 8.3 for the analysis of AdaGoal-UCBVI. We now sketch the AdaGoal-UCRL·VTR equivalent of the various key steps.

First, note that similar to Lemma F.5 in the tabular case, for any state-action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$, goal state $g \in \mathcal{G}$ and vector $Y \in \mathbb{R}^S$ such that $Y(g) = 0$, it holds that $[PY](s, a) = [P_g Y](s, a)$.

We now build the high-probability events. By using the standard self-normalized concentration inequality for vector-valued martingales of Abbasi-Yadkori et al. (2011, Theorem 2), it holds that with probability at least $1 - \delta/3$, for any $k \geq 1$ and $g \in \mathcal{G}$, $\boldsymbol{\theta}_g^\star$ lies in the ellipsoid

$$\mathcal{C}_{g,k} \triangleq \left\{ \boldsymbol{\theta} \in \mathbb{R}^{d+1} : \left\| \boldsymbol{\Sigma}_{g,k,1}^{1/2} (\boldsymbol{\theta}_{g,k} - \boldsymbol{\theta}) \right\|_2 \leq \beta_k \right\}.$$

The proof of the above statement follows the steps of e.g., Zhang et al. (2021b, Lemma A.2), the only slight difference being that we take an additional union bound over all goals $g \in \mathcal{G}$, hence the presence of G in the confidence radius (F.19). Furthermore, following the exact same steps as above and by definition of $\mathring{\boldsymbol{\Sigma}}$ and $\mathring{\boldsymbol{\theta}}$, it holds that with probability at least $1 - \delta/3$, for any $k \geq 1$ and $g \in \mathcal{G}$, $\boldsymbol{\theta}_g^\star$ lies in the ellipsoid

$$\mathcal{C}'_{g,k} \triangleq \left\{ \boldsymbol{\theta} \in \mathbb{R}^{d+1} : \left\| \mathring{\boldsymbol{\Sigma}}_{g,k,1}^{1/2} (\mathring{\boldsymbol{\theta}}_{g,k} - \boldsymbol{\theta}) \right\|_2 \leq \beta_k \right\}.$$

Here the confidence radius is the same as the one in $\mathcal{C}_{g,k}$ since it is chosen to be proportional to the magnitude of the $U_{g,k,h+1}(\cdot)$ function, which lies in $[0, H]$, as does the value function $V_{g,k,h+1}(\cdot)$. In what follows, we assume that the two high-probability events considered above

hold, i.e., that the following event holds (it does so with probability at least $1 - 2\delta/3$)

$$\left\{\forall k \geq 1, \forall g \in \mathcal{G},\ \boldsymbol{\theta}^\star_g \in \mathcal{C}_{g,k} \cap \mathcal{C}'_{g,k}\right\}. \tag{F.29}$$

▷ ***Key step ①: Optimism and gap bounds.*** The optimism property is standard: following e.g., Zhang et al. (2021b, Lemma A.1) (see also Zhou et al., 2021, Lemma C.4), it holds that $Q_{g,k,h}(s,a) \leq \overline{Q}^\star_{g,h}(s,a)$ and $V_{g,k,h}(s) \leq \overline{V}^\star_{g,h}(s)$ for any $(s,a,g) \in \mathcal{S} \times \mathcal{A} \times \mathcal{G}$, $h \in [H]$, $k \geq 1$.

We now depart from a usual regret minimization analysis and examine our errors U, proving that they upper bound the goal-conditioned gaps, formally defined as

$$W_{g,k,h}(s) \triangleq V^{\pi^k_g}_{g,h}(s) - V_{g,k,h}(s).$$

We now prove by induction that $W_{g,k,h}(s) \leq U_{g,k,h}(s)$. The property holds at $H+1$ since $W_{g,k,h}(s) = 0 = U_{g,k,h}(s)$. Assume that $W_{g,k,h+1}(s) \leq U_{g,k,h+1}(s)$, then we start by noticing, similar to Zhou et al. (2021, Equation C.10); Zhang et al. (2021b, Lemma A.1), that

$$W_{g,k,h}(s) \leq 2\beta_k \left\|\boldsymbol{\Sigma}^{-1/2}_{g,k,1} \boldsymbol{\phi}_{V_{g,k,h+1}}(s, \pi^k_{g,h}(s))\right\|_2 + \underbrace{[pV^{\pi^k}_{g,h+1}](s, \pi^k_{g,h}(s)) - [pV_{g,k,h+1}](s, \pi^k_{g,h}(s))}_{\triangleq X}.$$

We bound X as follows

$$\begin{aligned}
X &= [pW_{g,k,h+1}](s, \pi^k_{g,h}(s)) \\
&\overset{\text{(i)}}{\leq} [pU_{g,k,h+1}](s, \pi^k_{g,h}(s)) \\
&= \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s)), \boldsymbol{\theta}^\star_g \rangle \\
&= \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s)), \mathring{\boldsymbol{\theta}}_{g,k} \rangle + \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s)), \boldsymbol{\theta}^\star_g - \mathring{\boldsymbol{\theta}}_{g,k} \rangle \\
&\overset{\text{(ii)}}{\leq} \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s)), \mathring{\boldsymbol{\theta}}_{g,k} \rangle + \left\|\mathring{\boldsymbol{\Sigma}}^{1/2}_{g,k,1}(\mathring{\boldsymbol{\theta}}_{g,k} - \boldsymbol{\theta}^\star_g)\right\|_2 \left\|\mathring{\boldsymbol{\Sigma}}^{-1/2}_{g,k,1} \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s))\right\|_2 \\
&\overset{\text{(iii)}}{\leq} \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s)), \mathring{\boldsymbol{\theta}}_{g,k} \rangle + \beta_k \left\|\mathring{\boldsymbol{\Sigma}}^{-1/2}_{g,k,1} \boldsymbol{\phi}_{U_{g,k,h+1}}(s, \pi^k_{g,h}(s))\right\|_2,
\end{aligned}$$

where (i) comes from the induction hypothesis and because P is a monotone operator w.r.t. the partial ordering of functions, (ii) is by Cauchy-Schwarz, (iii) holds by event (F.29). Finally, using that $W_{g,k,h}(s) \in [0, H]$ and by definition of $U_{g,k,h}(s)$, we conclude that $W_{g,k,h}(s) \leq U_{g,k,h}(s)$.

▷ ***Key step ②: Bounding the cumulative gap bounds.*** We now bound $\sum_{k=1}^K U_{g_k,k,1}(s_0)$. It holds that

$$\begin{aligned}
&U_{g,k,h}(s_{k,h}) - U_{g,k,h+1}(s_{k,h+1}) \\
&\leq 2\beta_k \min\left\{1, \left\|\boldsymbol{\Sigma}^{-1/2}_{g,k,1} \boldsymbol{\phi}_{V_{g,k,h+1}}(s_{k,h}, \pi^k_{g,h}(s_{k,h}))\right\|_2\right\}
\end{aligned}$$

$$+ \beta_k \min\left\{1, \left\|\mathring{\boldsymbol{\Sigma}}_{g,k,1}^{-1/2}\boldsymbol{\phi}_{U_{g,k,h+1}}(s_{k,h}, \pi_{g,h}^k(s_{k,h}))\right\|_2\right\}$$
$$+ \min\Big\{\underbrace{\langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s_{k,h}, \pi_{g,h}^k(s_{k,h})), \mathring{\boldsymbol{\theta}}_{g,k}\rangle - U_{g_k,k,h+1}(s_{k,h+1})}_{\triangleq Y}, H\Big\},$$

where

$$\begin{aligned}
Y &\le |\langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s_{k,h}, \pi_{g,h}^k(s_{k,h})), \boldsymbol{\theta}_g^\star - \mathring{\boldsymbol{\theta}}_{g,k}\rangle| + \langle \boldsymbol{\phi}_{U_{g,k,h+1}}(s_{k,h}, \pi_{g,h}^k(s_{k,h})), \boldsymbol{\theta}_g^\star\rangle - U_{g,k,h+1}(s_{k,h+1}) \\
&\le \left\|\mathring{\boldsymbol{\Sigma}}_{g,k,1}^{1/2}(\mathring{\boldsymbol{\theta}}_{g,k} - \boldsymbol{\theta}_g^\star)\right\|_2 \left\|\mathring{\boldsymbol{\Sigma}}_{g,k,1}^{-1/2}\boldsymbol{\phi}_{U_{g,k,h+1}}(s_{k,h}, \pi_{g,h}^k(s_{k,h}))\right\|_2 \\
&\quad + [P_g U_{g,k,h+1}](s_{k,h}, \pi_{g,h}^k(s_{k,h})) - U_{g,k,h+1}(s_{k,h+1}) \\
&\le \beta_k \left\|\mathring{\boldsymbol{\Sigma}}_{g,k,1}^{-1/2}\boldsymbol{\phi}_{U_{g,k,h+1}}(s_{k,h}, \pi_{g,h}^k(s_{k,h}))\right\|_2 + [P_g U_{g,k,h+1}](s_{k,h}, \pi_{g,h}^k(s_{k,h})) - U_{g,k,h+1}(s_{k,h+1}).
\end{aligned}$$

Therefore we get by telescopic sum

$$\begin{aligned}
\sum_{k=1}^K U_{g_k,k,1}(s_0) &= \sum_{k=1}^K \sum_{h=1}^H \left(U_{g_k,k,h}(s_{k,h}) - U_{g_k,k,h+1}(s_{k,h+1})\right) \\
&\le \underbrace{2\beta_K \sum_{k=1}^K \sum_{h=1}^H \min\left\{1, \left\|\boldsymbol{\Sigma}_{g_k,k,1}^{-1/2}\boldsymbol{\phi}_{V_{g_k,k,h+1}}(s_{k,h}, a_{k,h})\right\|_2\right\}}_{\triangleq Z_1} \\
&\quad + \underbrace{2\beta_K \sum_{k=1}^K \sum_{h=1}^H \min\left\{1, \left\|\mathring{\boldsymbol{\Sigma}}_{g_k,k,1}^{-1/2}\boldsymbol{\phi}_{U_{g_k,k,h+1}}(s_{k,h}, a_{k,h})\right\|_2\right\}}_{\triangleq Z_2} \\
&\quad + \underbrace{\sum_{k=1}^K \sum_{h=1}^H [P_{g_k} U_{g_k,k,h+1}](s_{k,h}, a_{k,h}) - U_{g_k,k,h+1}(s_{k,h+1})}_{\triangleq Z_3}.
\end{aligned}$$

We bound Z_1 and Z_2 using Cauchy-Schwarz and the elliptical potential lemma from linear bandits (Abbasi-Yadkori et al., 2011, Lemma 11), see e.g., Zhang et al. (2021b, Proof of Lemma A.3). This yields

$$\begin{aligned}
Z_1 &\le \sqrt{KH}\sqrt{\sum_{k=1}^K \sum_{h=1}^H \min\left\{1, \left\|\boldsymbol{\Sigma}_{g_k,k,1}^{-1/2}\boldsymbol{\phi}_{V_{g_k,k,h+1}}(s_{k,h}, a_{k,h})\right\|_2^2\right\}} \\
&\le \sqrt{2}\sqrt{KH}\sqrt{\sum_{k=1}^K \sum_{h=1}^H \min\left\{1, \left\|\boldsymbol{\Sigma}_{g_k,k,h}^{-1/2}\boldsymbol{\phi}_{V_{g_k,k,h+1}}(s_{k,h}, a_{k,h})\right\|_2^2\right\}} + 2Hd\log(1 + kH^3/\lambda) \\
&\le \sqrt{2KHd\log(1 + KH^3/(d\lambda)) + 2Hd\log(1 + kH^3/\lambda)},
\end{aligned}$$

and likewise for Z_2. The term Z_3 can be bounded by the Azuma-Hoeffding inequality since its summands form a martingale difference sequence, thus with probability at least $1-\delta/3$, it holds that $Z_3 \leq H\sqrt{2HK\log(3/\delta)}$.

Putting everything together and using that $\beta_K = \widetilde{O}(H\sqrt{d})$, we obtain

$$\sum_{k=1}^{K} U_{g_k,k,1}(s_0) = \widetilde{O}\left(dH^{3/2}\sqrt{K} + H^2 d^{3/2}\right).$$

▷ ***Key step ③: Bounding the sample complexity.*** We follow the reasoning given in Section 8.3. By construction of the stopping rule (8.1), the algorithm terminates at an episode κ that verifies

$$\varepsilon \cdot (\kappa - 1) \leq \sum_{k=1}^{\kappa-1} \mathcal{E}_k(g_k) = \frac{8\varepsilon}{9} \cdot (\kappa - 1) + \widetilde{O}\left(dH^{3/2}\sqrt{\kappa} + H^2 d^{3/2}\right).$$

Solving this functional inequality in κ yields

$$\kappa = \widetilde{O}\left(\frac{H^3 d^2}{\varepsilon^2} + \frac{H^2 d^{3/2}}{\varepsilon}\right).$$

Using that the sample complexity is bounded by $\kappa(H+1)$ and that $H = \widetilde{O}(L)$, we conclude the proof.

▷ ***Key step ④: Connecting to the original* MGE *objective.*** The proof of this step is identical to the one of AdaGoal-UCBVI in Section F.2.3.

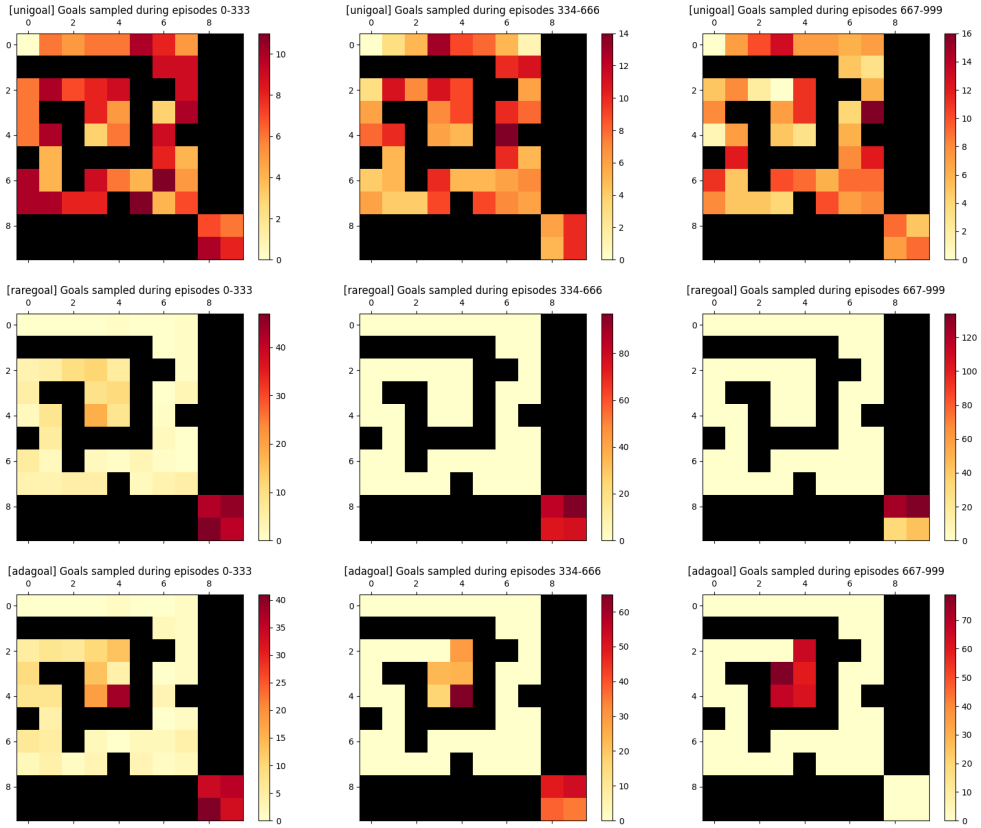


Figure F.3 – Goal sampling frequency of UniGoal-UCBVI (*top row*), RareGoal-UCBVI (*middle row*) and AdaGoal-UCBVI (*bottom row*) over 1000 episodes, split over episodes $0-333$ (*left column*), episodes $334-666$ (*middle column*) and episodes $667-999$ (*right column*). Episodes are of length $H = 50$, the environment is a grid-world with $S = 52$ states, starting state $s_0 = (0, 0)$ (i.e., the top left state), $A = 5$ actions (the 4 cardinal actions plus a_{reset}). The black walls act as reflectors, i.e., if the action leads against the wall, the agent stays in the current position with probability 1. An action fails with probability $p_f = 0.1$, in which case the agent follows (uniformly) one of the other directions. The 4 states of the bottom right room can only be accessed from s_0 by any cardinal action with probability $\eta = 0.001$, thus they are extremely hard to reliably reach as their associated $V^\star(s_0 \to \cdot)$ is very large (scaling with η^{-1}). We select $L = 40$ for AdaGoal, and $\alpha = 0.1$ for RareGoal. For the three methods we follow the practice of Menard et al. (2021, Section 4) and use their proposed simplified form for the exploration bonuses. The experiment is based on the rlberry framework (Domingues et al., 2021a).

F.4 ABLATION OF THE GOAL SELECTION SCHEME & PROOF OF CONCEPT EXPERIMENT

In this section, we single out the role of the adaptive goal selection scheme of AdaGoal, i.e., step ❶ in Algorithm 8.1. For simplicity we focus on the tabular case and consider that $\mathcal{G} = \mathcal{S}$. Keeping the remainder of the AdaGoal-UCBVI algorithm fixed, we compare it to two other ad-hoc goal sampling alternatives:

- UniGoal: the goal state is sampled uniformly in $\mathcal{S} \setminus \{s_0\}$, i.e., with probability

$$p^{\text{UNI}}(g) \triangleq (S-1)^{-1};$$

- RareGoal: the goal state is sampled proportionally to its rarity, i.e., with probability

$$p_\alpha^{\text{RARE}}(g) \triangleq \frac{(n_\alpha^k(g))^{-1}}{\sum_{s \in \mathcal{S} \setminus \{s_0\}} (n_\alpha^k(s))^{-1}},$$

  where $n^k(s) \triangleq \sum_{t=1}^{kH} \mathbb{1}[s_t = s]$ denotes the number of times state s was visited in the first k episodes, and $n_\alpha^k(s) \triangleq \max\{n^k(s), \alpha\}$ for $\alpha \in (0, 1]$.

We can find equivalents of these two goal selection schemes in existing goal-conditioned deep RL methods. The case of a uniform goal sampling distribution prescribed by the environment (i.e., UniGoal) is the most common, see e.g., Schaul et al. (2015) and Andrychowicz et al. (2017). Meanwhile, the goal sampling scheme of Skew-Fit (Pong et al., 2020), a recent state-of-the-art algorithm for deep GC-RL, gives higher sampling weight to rarer goal states, where rarity is measured by a learned generative model. In the tabular case, a goal state's rarity can be characterized by the inverse of its visitation count, which corresponds to RareGoal.

On the one hand, it is straightforward to show that UniGoal achieves a sample complexity of at most $\widetilde{O}(L^3 S^2 A \varepsilon^{-2})$. Intuitively, it pays for an extra S since it may sample goals that are too easy or too hard, in either case they are not very useful for the agent to improve its learning (and there may be in the worst case $S-2$ of such non-informative goal states).

On the other hand, we can see that by design RareGoal relies on the communicating assumption and may require poly$(S, A, D, \varepsilon^{-1})$ samples to learn an ε-optimal goal-conditioned policy on $\mathcal{G}_L$. Here the dependence on the diameter D is somewhat problematic. Indeed, imagine there exist a set of states $\mathcal{S}_{\text{hard}}$ such that $1 \ll V^\star(s_0 \to s) \leq D$ for $s \in \mathcal{S}_{\text{hard}}$ (i.e., very hard to reach states, e.g., by chance due to environment stochasticity). Then throughout the learning process, RareGoal will strive to reach the states in $\mathcal{S}_{\text{hard}}$ and select them as goals, which leads to unsuccessful episodes and a possible waste of samples. Consequently, when goals have varying reachability (e.g., if the environment is highly stochastic), RareGoal suffers from an issue of *goal prioritizing*, i.e., too-hard-to-reach states are given too much goal sampling importance.

Finally, we empirically complement our discussion above on the sequence of goals selected by UniGoal, RareGoal and AdaGoal. We design a simple two-room grid-world with a very small probability of reaching the second room, and illustrate in Figure F.3 the goal sampling frequency of UniGoal-UCBVI, RareGoal-UCBVI and AdaGoal-UCBVI. We see that over the course of the learning interaction, as opposed to the designs of RareGoal and UniGoal, our AdaGoal strategy is able to successfully discard the states from the bottom right room, which have a

negligible probability of being reached. In addition, AdaGoal is able to target as goals the states in the first room that are furthest away from s_0, i.e., those at the center of the "spiral", which effectively correspond to the fringe of what the agent can reliably reach.

F.5 Implementation details of Section 8.5

Here we provide the implementation details of our experiments reported in Section 8.5. Our implementation and hyperparameters of HER (Andrychowicz et al., 2017) are based on the PyTorch open-source codebase of https://github.com/TianhongDai/hindsight-experience-replay, which follows the official implementation of HER. As explained in Section 8.5, we approximate AdaGoal by computing the disagreement (i.e., standard deviation) of an ensemble of J goal-conditioned Q-functions and selecting a goal proportionally to it among N uniformly sampled goals. Then the policy is conditioned on this goal and executed for an episode of length $H = 50$. This recovers the VDS algorithm of Zhang et al. (2020b). We thus follow the implementation details given in the latter paper. In particular, the value ensemble (for goal selection) is treated as a separate module from the policy optimization (for goal-conditioned policy execution). In each training epoch, each Q-function in the ensemble performs Bellman updates with independently sampled mini-batches, and the policy is updated with DDPG (Lillicrap et al., 2015). Each Q-function in the ensemble is trained with its target network, with learning rate 1e-3, polyak coefficient 0.95, buffer size 1e6, and batch size 1000. Finally, we set $J = 3$ and $N = 1000$.

Appendix G

Complements on Chapter 9

G.1 Proof of Theorem 9.11

G.1.1 Computation of the Optimistic Policies

At each round k, for each goal state $s^\dagger \in \mathcal{W}_k$, DisCo computes an optimistic goal-oriented policy associated to the MDP $M'_k(s^\dagger)$ constructed as in Definition 9.10. This MDP is defined over the entire state space $\mathcal{S}$ and restricts the action to the only action a_{reset} outside $\mathcal{K}_k$. We can build an equivalent MDP by restricting the focus on $\mathcal{K}_k$. To this end, we define the following SSP-MDP.

Definition G.1. *Define* $M_k^\dagger(s^\dagger) \triangleq \langle \mathcal{S}_k^\dagger, \mathcal{A}_k^\dagger(\cdot), c_k^\dagger, P_k^\dagger \rangle$ *where* $\mathcal{S}_k^\dagger \triangleq \mathcal{K}_k \cup \{s^\dagger, x\}$ *and* $S_k^\dagger = |\mathcal{S}_k^\dagger| = |\mathcal{K}_k| + 2$. *State* x *is a meta-state that encapsulates all the states that have been observed so far and are not in* $\mathcal{K}_k$. *The action space* $\mathcal{A}_k^\dagger(\cdot)$ *is such that* $\mathcal{A}_k^\dagger(s) = \mathcal{A}$ *for all states* $s \in \mathcal{K}_k$ *and* $\mathcal{A}_k^\dagger(s) = \{a_{\text{reset}}\}$ *for* $s \in \{s^\dagger, x\}$. *The cost function is* $c_k^\dagger(x, a) = 0$ *for any* $a \in \mathcal{A}_k^\dagger(x)$ *and* $c_k^\dagger(s, a) = 1$ *everywhere else. The transition function is defined as* $P_k^\dagger(s^\dagger | s^\dagger, a) = P_k^\dagger(s_0 | x, a) = 1$ *for any* a, $P_k^\dagger(y|s,a) = P(y|s,a)$ *for any* $(s, a, y) \in \mathcal{K}_k \times \mathcal{A} \times (\mathcal{K}_k \cup \{s^\dagger\})$ *and* $P_k^\dagger(x|s,a) = 1 - \sum_{y \in \mathcal{K}_k \cup \{s^\dagger\}} P_k^\dagger(y|s,a)$.

Note that solving $M_k^\dagger$ yields a policy effectively restricted to the set $\mathcal{K}_k$ insofar as we can interpret the meta-state x as $\mathcal{S} \setminus \{\mathcal{K}_k \cup \{s^\dagger\}\}$. Since P is unknown, we cannot construct $M_k^\dagger(s^\dagger)$. Let N_k be the state-action counts accumulated up until now. We denote by $\widehat{P}_k$ the "global" empirical estimates, i.e., $\widehat{P}_k(y|s,a) = N_k(s,a,y)/N_k(s,a)$. Given them, we define the "restricted" empirical estimates $\widehat{P}_k^\dagger$ as follows: $\widehat{P}_k^\dagger(y|s,a) \triangleq \widehat{P}_k(y|s,a)$ for any $(s,a,y) \in \mathcal{K}_k \times \mathcal{A} \times (\mathcal{K}_k \cup \{s^\dagger\})$ and $\widehat{P}_k^\dagger(x|s,a) \triangleq 1 - \sum_{y \in \mathcal{K}_k \cup \{s^\dagger\}} \widehat{P}_k^\dagger(y|s,a)$. Denoting $N_k^+(s,a) \triangleq \max\{1, N_k(s,a)\}$, we then

Algorithm G.1: OVI$_{\text{SSP}}$ planning procedure

Input: goal $s^\dagger$, states $\mathcal{K}_k \cup \{x\}$, samples N_k, precision level $\gamma > 0$.
Output: Value vector $\widetilde{u}^\dagger$ and policy $\widetilde{\pi}^\dagger$.
Estimate transitions probabilities $\widehat{P}_k$ using N_k.
Compute the optimistic SSP-MDP $\widetilde{M}_k^\dagger$ as detailed in Definition G.2.
Compute $(\widetilde{u}_k^\dagger, \widetilde{\pi}_k^\dagger) = \text{VI-SSP}(s^\dagger, \mathcal{K}_k^\dagger \cup \{x\}, \mathcal{A}_k^\dagger, c_k^\dagger, \widetilde{P}_k^\dagger, \gamma)$ (see Algorithm 2.1).

define the following bonuses for any $(s, a, y) \in \mathcal{K}_k \times \mathcal{A} \times (\mathcal{K}_k \cup \{s^\dagger\})$,

$$\beta_k(s,a,y) \triangleq 2\sqrt{\frac{\widehat{P}_k(y|s,a)(1-\widehat{P}_k(y|s,a))}{N_k^+(s,a)} \log\left(\frac{2SAN_k^+(s,a)}{\delta}\right)} + \frac{6\log\left(\frac{2SAN_k^+(s,a)}{\delta}\right)}{N_k^+(s,a)}, \tag{G.1}$$

$$\beta_k(s,a,x) \triangleq \sum_{y \in \mathcal{K}_k \cup \{s^\dagger\}} \beta_k(s,a,y). \tag{G.2}$$

Moreover, we set the uncertainty about the MDP at the meta-state x and at the goal state $s^\dagger$ to 0 by construction (since their outgoing transitions are deterministic, respectively to s_0 and $s^\dagger$).

We now leverage the construction of an optimistic SSP model of Rosenberg et al. (2020).

Definition G.2. *We denote by* $\widetilde{M}_k^\dagger(s^\dagger) = \langle \mathcal{S}_k^\dagger, \mathcal{A}_k^\dagger(\cdot), c_k^\dagger, \widetilde{P}_k^\dagger \rangle$ *the optimistic MDP associated to* $M_k^\dagger(s^\dagger)$ *defined in Definition G.1. Then,* $\forall (s,a) \in \mathcal{K}_k \times \mathcal{A}$,

$$\widetilde{P}_k^\dagger(y|s,a) \triangleq \max\left\{\widehat{P}_k(y|s,a) - \beta_k(s,a,y),\ 0\right\}, \quad \forall y \in \mathcal{K}_k \cup \{x\}, \tag{G.3}$$

$$\widetilde{P}_k^\dagger(s^\dagger|s,a) \triangleq 1 - \sum_{y \in \mathcal{K}_k \cup \{x\}} \widetilde{P}_k^\dagger(y|s,a), \tag{G.4}$$

$$\widetilde{P}_k^\dagger(s^\dagger|s^\dagger,a) = \widetilde{P}_k^\dagger(s_0|x,a) = 1. \tag{G.5}$$

Given this MDP, we can compute the optimistic value vector $\widetilde{u}_k^\dagger$ and policy $\widetilde{\pi}_k^\dagger$ using value iteration for SSP: $(\widetilde{u}_k^\dagger, \widetilde{\pi}_k^\dagger) = \text{VI-SSP}(s^\dagger, \mathcal{K}_k^\dagger \cup \{x\}, \mathcal{A}_k^\dagger, c_k^\dagger, \widetilde{P}_k^\dagger, \gamma)$ with precision level $\gamma = \frac{\varepsilon}{4L}$ (see Algorithm 2.1). We summarize the construction of the optimistic model and the computation of value function and policy in Algorithm G.1 (OVI$_{\text{SSP}}$).

Remark. Note that, given the possibly large number of states in the total environment $\mathcal{S}$, the way we compute the optimistic policies requires the construction of the meta-state x that encapsulates all the states in $\mathcal{S} \setminus \{\mathcal{K}_k \cup \{s^\dagger\}\}$, where $s^\dagger$ is the candidate goal state considered at round k. As a result, the uncertainty on the transitions reaching x needs to be summed over multiple states, as shown in Equation (G.2). This extra uncertainty at a single state in

the induced MDP has the effect of canceling out Bernstein techniques seeking to lower the prescribed requirement of the state-action samples that the algorithm should collect. In turn this implies that such variance-aware techniques would not lead to any improvement in the final sample complexity bound.

G.1.2 High-Probability Event

Lemma G.3. *It holds with probability at least $1-\delta$ that for any time step $t \geq 1$ and for any state-action pair (s,a) and next state s',*

$$|\widehat{p}_t(s'|s,a) - P(s'|s,a)| \leq 2\sqrt{\frac{\widehat{\sigma}_t^2(s'|s,a)}{N_t^+(s,a)} \log\left(\frac{2SAN_t^+(s,a)}{\delta}\right)} + \frac{6\log\left(\frac{2SAN_t^+(s,a)}{\delta}\right)}{N_t^+(s,a)}, \tag{G.6}$$

where $N_t^+(s,a) \triangleq \max\{1, N_t(s,a)\}$ and where $\widehat{\sigma}_t^2$ are the population variance of transitions, i.e., $\widehat{\sigma}_t^2(s'|s,a) \triangleq \widehat{P}_t(s'|s,a)(1-\widehat{P}_t(s'|s,a))$.

Proof. The confidence intervals in Equation (G.6) are constructed using the empirical Bernstein inequality, which guarantees that the considered event holds with probability at least $1-\delta$, see e.g., Fruit et al. (2020). □

Define the set of plausible transition probabilities as

$$C_k^\dagger \triangleq \bigcap_{(s,a)\in\mathcal{S}_k^\dagger\times\mathcal{A}} C_k^\dagger(s,a),$$

where

$$C_k^\dagger(s,a) \triangleq \{\widetilde{p} \in \mathcal{C} \mid \widetilde{p}(\cdot \mid s^\dagger, a) = \mathbb{1}_{s^\dagger}, \widetilde{p}(\cdot \mid x, a) = \mathbb{1}_{s_0}, |\widetilde{p}(s'|s,a) - \widehat{p}_k(s'|s,a)| \leq \beta_k(s,a,s')\},$$

with $\mathcal{C}$ the $S_k^\dagger$-dimensional simplex and $\widehat{p}_k$ the empirical average of transitions.

Lemma G.4. *Introduce the event $\Theta \triangleq \bigcap_{k=1}^{+\infty}\bigcap_{s^\dagger\in\mathcal{W}_k}\{P_k^\dagger \in C_k^\dagger\}$. Then $\mathbb{P}(\Theta) \geq 1-\frac{\delta}{3}$.*

Proof. We have with probability at least $1-\frac{\delta}{3}$ that, for any $y \neq x$, $|P_k^\dagger(y|s,a) - \widehat{P}_k^\dagger(y|s,a)| \leq \beta_k(s,a,y)$ from the empirical Bernstein inequality (see Equation (G.6)), and moreover $|\widehat{P}_k^\dagger(x|s,a) - P_k^\dagger(x|s,a)| = \left|1 - \sum_{y\in\mathcal{K}_k\cup\{s^\dagger\}} P_k^\dagger(y|s,a) - \left(1 - \sum_{y\in\mathcal{K}_k\cup\{s^\dagger\}} \widehat{P}_k^\dagger(y|s,a)\right)\right| \leq \sum_{y\in\mathcal{K}_k\cup\{s^\dagger\}} |P_k^\dagger(y|s,a) - \widehat{P}_k^\dagger(y|s,a)| \leq \beta_k(s,a,x)$. □

Lemma G.5. *Under the event Θ, for any round k and any goal state $s^\dagger \in \mathcal{W}_k$, the optimistic model $\widetilde{P}_k^\dagger$ constructed in Definition G.2 verifies $\widetilde{P}_k^\dagger \in \mathcal{P}_{\eta_k}^{(P_k^\dagger)}$, with $\eta_k \triangleq 4\beta_k(s,a,x)$ where β_k is defined in Equation (G.2).*

Proof. Combining the construction in Definition G.2, the proof of Lemma G.4 and the triangle inequality yields

$$\begin{aligned}\sum_{y\in\mathcal{K}_k\cup\{x\}} |\widetilde{P}_k^\dagger(y|s,a) - P_k^\dagger(y|s,a)| &\leq \sum_{y\in\mathcal{K}_k\cup\{x\}} |\widetilde{P}_k^\dagger(y|s,a) - \widehat{P}_k^\dagger(y|s,a)| + |\widehat{P}_k^\dagger(y|s,a) - P_k^\dagger(y|s,a)| \\ &\leq \sum_{y\in\mathcal{K}_k\cup\{x\}} \beta_k(s,a,y) + 2\beta_k(s,a,x) \\ &\leq 4\beta_k(s,a,x).\end{aligned}$$

□

Throughout the remainder of the proof, we assume that the event Θ holds.

G.1.3 Properties of the Optimistic Policies and Value Vectors

We recall notation. Let us fix any round k and any goal state $s^\dagger \in \mathcal{W}_k$. We denote by $\widetilde{\pi}_k^\dagger$ the greedy policy w.r.t. $\widetilde{u}_k^\dagger(\cdot \to s^\dagger)$ in the optimistic model $\widetilde{P}_k^\dagger$. Let $\widetilde{v}_k^\dagger(s \to s^\dagger)$ be the value function of policy $\widetilde{\pi}_k^\dagger$ starting from state s in the model $\widetilde{P}_k^\dagger$. We can apply Lemma E.1 (indeed, we have $c_{\min} = 1 > 0$ and there exists at least one proper policy to reach the goal state $s^\dagger$ since it belongs to $\mathcal{W}_k$). Moreover, we have that $\widetilde{V}_{\mathcal{K}_k}^\star(s_0 \to s^\dagger) \leq V_{\mathcal{K}_k}^\star(s_0 \to s^\dagger)$ given the way the optimistic model $\widetilde{P}_k^\dagger$ is computed (i.e., by maximizing the probability of transitioning to the goal at any state-action pair), see Rosenberg et al. (2020, Lemma B.12). Hence we get the two following important properties.

Lemma G.6. *For any round k, goal state $s^\dagger \in \mathcal{W}_k$ and state $s \in \mathcal{K}_k \cup \{x\}$, we have under the event Θ,*

$$\widetilde{u}_k^\dagger(s \to s^\dagger) \leq V_{\mathcal{K}_k}^\star(s \to s^\dagger).$$

Lemma G.7. *For any round k, goal state $s^\dagger \in \mathcal{W}_k$ and state $s \in \mathcal{K}_k \cup \{x\}$, we have*

$$\widetilde{v}_k^\dagger(s \to s^\dagger) \leq (1+2\gamma)\widetilde{u}_k^\dagger(s \to s^\dagger).$$

G.1.4 State Transfer from $\mathcal{U}$ to $\mathcal{K}$ (step ④)

We fix any round k and any goal state $s^\dagger \in \mathcal{W}_k$ that is added to the set of "controllable" states $\mathcal{K}$, i.e., for which $\widetilde{u}_k^\dagger(s_0 \to s^\dagger) \leq L$.

Lemma G.8. *Under the event Θ, we have both following inequalities*

$$\begin{cases} v_k^\dagger(s_0 \to s^\dagger) \leq L + \varepsilon, \\ v_k^\dagger(s_0 \to s^\dagger) \leq V_{\mathcal{K}_k}^\star(s_0 \to s^\dagger) + \varepsilon. \end{cases}$$

In particular, the first inequality entails that $s^\dagger \in \mathcal{S}_{L+\varepsilon}^{\to}$, which justifies the validity of the state transfer from $\mathcal{U}$ to $\mathcal{K}$.

Proof. We have

$$\widetilde{v}_k^\dagger(s_0 \to s^\dagger) \overset{\text{(a)}}{\leq} (1+2\gamma)\widetilde{u}_k^\dagger(s_0 \to s^\dagger) \leq \begin{cases} \overset{\text{(b)}}{\leq} L + \frac{\varepsilon}{3} \\ \overset{\text{(c)}}{\leq} V_{\mathcal{K}_k}^\star(s_0 \to s^\dagger) + \frac{\varepsilon}{3}, \end{cases} \tag{G.7}$$

where inequality (a) comes from Lemma G.7, inequality (b) combines the algorithmic condition $\widetilde{u}_k^\dagger(s_0 \to s^\dagger) \leq L$ and the VI precision level $\gamma \triangleq \frac{\varepsilon}{6L}$, and finally inequality (c) combines Lemma G.6 and the VI precision level. Moreover, for any state in $\mathcal{K}_k$,

$$\widetilde{v}_k^\dagger(s \to s^\dagger) \overset{\text{(a)}}{\leq} \widetilde{V}_{\mathcal{K}_k}^\star(s \to s^\dagger) + \frac{\varepsilon}{3} \overset{\text{(b)}}{\leq} \widetilde{V}_{\mathcal{K}_k}^\star(s_0 \to s^\dagger) + 1 + \frac{\varepsilon}{3} \leq \widetilde{v}_k^\dagger(s_0 \to s^\dagger) + 1 + \frac{\varepsilon}{3},$$

where (a) comes from Lemma G.6 and (b) stems from the presence of the a_{reset} action (Assumption 8.3).

We now provide the exact choice of allocation function ϕ in Algorithm 9.1. We introduce

$$\gamma \triangleq \frac{2\varepsilon}{12(L+1+\varepsilon)(L+\frac{\varepsilon}{3})}.$$

(Note that $\gamma = O(\varepsilon/L^2)$.) We set the following requirement of samples for each state-action pair (s,a) at round k,

$$n_k = \phi(\mathcal{K}_k) = \left\lceil \frac{57 X_k^2}{\gamma^2} \left[\log\left(\frac{8e X_k \sqrt{2SA}}{\sqrt{\delta}\gamma} \right) \right]^2 + \frac{24|\mathcal{S}_k^\dagger|}{\gamma} \log\left(\frac{24|\mathcal{S}_k^\dagger| SA}{\delta\gamma} \right) \right\rceil, \tag{G.8}$$

where we define

$$X_k \triangleq \max_{(s,a)\in \mathcal{S}_k^\dagger \times \mathcal{A}} \sum_{s'\in\mathcal{S}_k^\dagger} \sqrt{\widehat{\sigma}_k^2(s'|s,a)},$$

with $\widehat{\sigma}_k^2(s'|s,a) \triangleq \widehat{P}_k^\dagger(s'|s,a)(1-\widehat{P}_k^\dagger(s'|s,a))$ the estimated variance of the transition from (s,a) to s'. Leveraging the empirical Bernstein inequality (Lemma G.3) and perfoming simple algebraic manipulations (see e.g., Kazerouni et al., 2017, Lemma 8 and 9) yields that $\beta_k(s,a,x) \le \gamma$. From Lemma G.5, this implies that $\widetilde{P}_k^\dagger \in \mathcal{P}_\eta^{(P_k^\dagger)}$ with $\eta \triangleq 4\gamma$. We can then apply Lemma 2.14 (whose condition 2.4 is verified), which gives

$$\begin{aligned} v_k^\dagger(s_0 \to s^\dagger) &\le \left(1 + \eta \|\widetilde{v}_k^\dagger(\cdot \to s^\dagger)\|_\infty\right) \widetilde{v}_k^\dagger(s_0 \to s^\dagger) \\ &\le \left(1 + \eta(L+1+\varepsilon)\right) \widetilde{v}_k^\dagger(s_0 \to s^\dagger) \\ &\le \widetilde{v}_k^\dagger(s_0 \to s^\dagger) + \frac{2\varepsilon}{3}, \end{aligned} \tag{G.9}$$

where the last inequality uses that $\eta(L+1+\varepsilon)(L+\frac{\varepsilon}{3}) = \frac{2\varepsilon}{3}$ by definition of γ. Plugging in Equation (G.7) yields the sought-after inequalities. □

G.1.5 Termination of the Algorithm

Lemma G.9 (Variant of Lemma 17 of Lim and Auer, 2012)**.** *Suppose that for every state $s \in \mathcal{S}$, each action $a \in \mathcal{A}$ is executed $b \ge \lceil L \log\left(\frac{3ALS}{\delta}\right) \rceil$ times. Let $\mathcal{S}'_{s,a}$ be the set of all next states visited during the b executions of (s,a). Denote by Λ the complementary of the event*

$$\left\{ \exists (s', s, a) \in \mathcal{S}^2 \times \mathcal{A} : P(s'|s,a) \ge \frac{1}{L} \wedge s' \notin \mathcal{S}'_{s,a} \right\}.$$

Then $\mathbb{P}(\Lambda) \ge 1 - \frac{\delta}{3}$.

Lemma G.10. *Under the event $\Theta \cap \Lambda$, for any round k, either $\mathcal{S}_L^{\rightarrow} \subseteq \mathcal{K}_k$, or there exists a state $s^\dagger \in \mathcal{S}_L^{\rightarrow} \setminus \mathcal{K}_k$ such that $s^\dagger \in \mathcal{W}_k$ and is L-controllable with a policy restricted to $\mathcal{K}_k$. Moreover, $|\mathcal{W}_k| \le 2LA|\mathcal{K}_k|$.*

Proof. Consider a round k such that $\mathcal{S}_L^{\rightarrow} \setminus \mathcal{K}_k$ is non-empty. Due to the incremental construction of the set $\mathcal{S}_L^{\rightarrow}$ (Definition 9.5), there exists a state $s^\dagger \in \mathcal{S}_L^{\rightarrow}$ and a policy restricted to $\mathcal{K}_k$ that can reach $s^\dagger$ in at most L steps (in expectation). Hence there exists a state-action pair $(s,a) \in \mathcal{K}_k \times \mathcal{A}$ such that $P(s^\dagger|s,a) \geq \frac{1}{L}$. Since $\phi(\mathcal{K}_k) \geq \lceil L \log\left(\frac{3ALS}{\delta}\right) \rceil$ samples are available at each state-action pair, according to Lemma G.9, we get that, under the event Λ, $s^\dagger$ is found during the sample collection procedure for the state-action pair (s,a) (step ①), which implies that $s^\dagger \in \mathcal{U}_k$.

Moreover, the choice of allocation function ϕ guarantees in particular that there are more than $\Omega(\frac{4L^2}{\varepsilon^2}\log(\frac{2LSA}{\delta\varepsilon}))$ samples available at each state-action pair $(s,a) \in \mathcal{K}_k \times \mathcal{A}$. From the empirical Bernstein inequality of Equation (G.6), we thus have that $|P(s^\dagger|s,a) - \widehat{P}_k(s^\dagger|s,a)| \leq \frac{\varepsilon}{2L}$ under the event Θ. Consequently we have

$$\widehat{P}_k(s^\dagger|s,a) \geq \frac{1}{L} - |P(s^\dagger|s,a) - \widehat{P}_k(s^\dagger|s,a)| \geq \frac{1-\frac{\varepsilon}{2}}{L},$$

which implies that $s^\dagger \in \mathcal{W}_k$. Furthermore, we can decompose $\mathcal{W}_k$ the following way

$$\mathcal{W}_k = \bigcup_{(s,a)\in\mathcal{K}_k\times\mathcal{A}} \mathcal{Y}_k(s,a),$$

where we introduce the subset

$$\mathcal{Y}_k(s,a) \triangleq \left\{ s' \in \mathcal{U}_k : \widehat{P}_k(s'|s,a) \geq \frac{1-\frac{\varepsilon}{2}}{L} \right\}.$$

We then have

$$1 = \sum_{s'\in\mathcal{S}} \widehat{P}_k(s'|s,a) \geq \sum_{s'\in\mathcal{Y}_k(s,a)} \widehat{P}_k(s'|s,a) \geq \frac{1-\frac{\varepsilon}{2}}{L}|\mathcal{Y}_k(s,a)|.$$

We conclude the proof by writing that

$$|\mathcal{W}_k| \leq \sum_{(s,a)\in\mathcal{K}_k\times\mathcal{A}} |\mathcal{Y}_k(s,a)| \leq \frac{L}{1-\frac{\varepsilon}{2}} A|\mathcal{K}_k| \leq 2LA|\mathcal{K}_k|,$$

where the last inequality uses that $\varepsilon \leq 1$ (from line 3 of Algorithm 9.1). □

Lemma G.11. *Under the event $\Theta \cap \Lambda$, when either condition* STOP1 *or* STOP2 *is triggered (at a round indexed by K), we have $\mathcal{S}_L^{\rightarrow} \subseteq \mathcal{K}_K$.*

Proof. If condition STOP1 is triggered, Lemma G.10 immediately guarantees that $\mathcal{S}_L^{\rightarrow} \subseteq \mathcal{K}_K$ under the event Λ. If condition STOP2 is triggered, we have for all $s \in \mathcal{W}_K$, $\widetilde{u}_s(s_0 \to s) > L$. From Lemma G.6 this means that, under the event Θ, for all $s \in \mathcal{W}_K$, $V^\star_{\mathcal{K}_K}(s_0 \to s) > L$. Hence

none of the states in $\mathcal{W}_K$ can be reached in at most L steps (in expectation) with a policy restricted to $\mathcal{K}_K$. We conclude the proof using Lemma G.10. □

Lemma G.12. *Under the event $\Theta \cap \Lambda$, when* DisCo *terminates at round K, for any state $s \in \mathcal{K}_K$, the policy π_s computed during step ⑤ verifies*

$$V^{\pi_s}(s_0 \to s) \leq \min_{\pi \in \Pi(\mathcal{S}_L^{\to})} V^{\pi}(s_0 \to s) + \varepsilon.$$

Moreover, we have that $\mathcal{S}_L^{\to} \subseteq \mathcal{K}_K \subseteq \mathcal{S}_{L+\varepsilon}^{\to}$.

Proof. Assume that the event $\Theta \cap \Lambda$ holds. Then when the final set $\mathcal{K}_K$ is considered and the new policies are computed using all the samples, Lemma G.8 yields for all $s \in \mathcal{K}_K$,

$$V^{\pi_s}(s_0 \to s) \leq \min_{\pi \in \Pi(\mathcal{K}_K)} V^{\pi}(s_0 \to s) + \varepsilon.$$

Moreover Lemma G.11 entails that $\mathcal{K}_K \supseteq \mathcal{S}_L^{\to}$. This implies from Lemma 9.3 that

$$\min_{\pi \in \Pi(\mathcal{K}_K)} V^{\pi}(s_0 \to s) \leq \min_{\pi \in \Pi(\mathcal{S}_L^{\to})} V^{\pi}(s_0 \to s),$$

which means that $\mathcal{K}_K \subseteq \mathcal{S}_{L+\varepsilon}^{\to}$. □

G.1.6 High Probability Bound on the Sample Collection Phase (step ①)

Denote by K the (random) index of the last round during which the algorithm terminates. We focus on the sample collection procedure for any state $s \in \mathcal{K}_K$. We denote by k_s the index of the round during which s was added to the set of "controllable" states $\mathcal{K}$. To collect samples at state s, the learner uses the shortest-path policy π_s. We say that an attempt to collect a specific sample is a *rollout*. We denote by $Z_K \triangleq |\mathcal{K}_K| A N_K$ the total number of samples that the learner needs to collect. As such, at most Z_K rollouts must take place. Assume that the event Θ holds. Then from Lemma G.12, we have $\mathcal{K}_K \subseteq \mathcal{S}_{L+\varepsilon}^{\to}$. Hence, denoting $S_{L+\varepsilon} \triangleq |\mathcal{S}_{L+\varepsilon}^{\to}|$, we have $Z_K \leq Z_{L+\varepsilon} \triangleq S_{L+\varepsilon} A \Phi(\mathcal{S}_{L+\varepsilon}^{\to})$. The following lemma provides a high-probability upper bound on the time steps required to meet the sampling requirements.

Lemma G.13. *Assume that the event Θ holds. Set*

$$\psi \triangleq 4(L + \varepsilon + 1) \log\left(\frac{6 Z_{L+\varepsilon}}{\delta}\right),$$

and introduce the following event

$$\mathcal{T} \triangleq \Big\{\exists \text{ one rollout (with goal state } s) \text{ s.t. } \tau_{\pi_s}(s_0 \to s) > \psi\Big\}.$$

We have $\mathbb{P}(\mathcal{T}) \leq \frac{\delta}{3}$.

Proof. Assume that the event Θ holds. Leveraging a union bound argument and applying Lemma G.14 to policy π_s which verifies $V^{\pi_s}(s' \to s) \leq L + \varepsilon + 1$ for any $s' \in K_{k_s}$, we get

$$\mathbb{P}(\mathcal{T}) \leq \sum_{rollouts} 2\exp\left(-\frac{\psi}{4(L+\varepsilon+1)}\right) \leq 2Z_{L+\varepsilon}\exp\left(-\frac{\psi}{4(L+\varepsilon+1)}\right) \leq \frac{\delta}{3},$$

where the last inequality comes from the choice of ψ. □

Lemma G.14 (Rosenberg et al., 2020, Lemma B.5)**.** *Let π be a proper policy such that for some $d > 0$, $V^{\pi}(s) \leq d$ for every non-goal state s. Then the probability that the cumulative cost of π to reach the goal state from any state s is more than m, is at most $2e^{-m/(4d)}$ for all $m \geq 0$. Note that a cost of at most m implies that the number of steps is at most $m/c_{\min}$.*

G.1.7 Putting Everything Together: Sample Complexity Bound

The sample complexity of the algorithm is solely induced by the sample collection procedure (step ①). Recall that we denote by K the index of the round at which the algorithm terminates. With probability at least $1 - \frac{2\delta}{3}$, Lemma G.11 holds, and so does the event Θ. Hence the algorithm discovers a set of states $\mathcal{K}_K \supseteq \mathcal{S}_L^{\to}$. Moreover, from Lemma G.12, the algorithm outputs for each $s \in \mathcal{K}_K$ a policy π_s with $\mathbb{E}[\tau_{\pi_s}(s_0 \to s)] \leq V^{\star}_{\mathcal{S}_L^{\to}}(s) + \varepsilon$. Hence we also have $|\mathcal{K}_K| \leq S_{L+\varepsilon} \triangleq |\mathcal{S}^{\to}_{L+\varepsilon}|$.

We denote by $Z_K \triangleq |\mathcal{K}_K| A\, \phi(\mathcal{K}_K)$ the total number of samples that the learner needs to collect. From Lemma G.13, with probability at least $1 - \frac{\delta}{3}$, the total sample complexity of the algorithm is at most ψZ_K, where $\psi \triangleq 4(L+\varepsilon+1)\log\left(\frac{6Z_{L+\varepsilon}}{\delta}\right)$.

Now, from Equation (G.8) there exists an absolute constant $\alpha > 0$ such that DisCo selects as allocation function ϕ

$$\phi : \mathcal{X} \to \alpha \cdot \left(\frac{L^4 \widehat{\Theta}(\mathcal{X})}{\varepsilon^2}\log^2\left(\frac{LSA}{\varepsilon\delta}\right) + \frac{L^2|\mathcal{X}|}{\varepsilon}\log\left(\frac{LSA}{\varepsilon\delta}\right)\right),$$

where

$$\widehat{\Theta}(\mathcal{X}) \triangleq \max_{(s,a)\in\mathcal{X}\times\mathcal{A}} \left(\sum_{s'\in\mathcal{X}} \sqrt{\widehat{P}(s'|s,a)(1-\widehat{P}(s'|s,a))} \right)^2 .$$

The total requirement is $\phi(\mathcal{K}_K)$. Note that from Cauchy-Schwarz's inequality, we have

$$\widehat{\Theta}(\mathcal{K}_K) \leq \Gamma_K \triangleq \max_{(s,a)\in\mathcal{K}_K\times\mathcal{A}} \|\{P(s'|s,a)\}_{s'\in\mathcal{K}_K}\|_0 \leq |\mathcal{K}_K|.$$

Combining everything yields with probability at least $1-\delta$,

$$\psi Z_K = \widetilde{O}\left(\frac{L^5 \Gamma_K |\mathcal{K}_K| A}{\varepsilon^2} + \frac{L^3 |\mathcal{K}_K|^2 A}{\varepsilon} \right).$$

We finally use that $\mathcal{K}_K \subset \mathcal{S}^{\rightarrow}_{L+\varepsilon}$ from Lemma G.12, which implies that

$$\mathcal{C}_{\mathrm{AX}^\star}(\textsc{DisCo}, L, \varepsilon, \delta) = \widetilde{O}\left(\frac{L^5 \Gamma_{L+\varepsilon} S_{L+\varepsilon} A}{\varepsilon^2} + \frac{L^3 S^2_{L+\varepsilon} A}{\varepsilon} \right),$$

where $\Gamma_{L+\varepsilon} \triangleq \max_{(s,a)\in\mathcal{S}^{\rightarrow}_{L+\varepsilon}\times\mathcal{A}} \|\{P(s'|s,a)\}_{s'\in\mathcal{S}^{\rightarrow}_{L+\varepsilon}}\|_0$. This concludes the proof of Theorem 9.11.

G.1.8 Proof of Corollary 9.12

The result given in Corollary 9.12 comes from retracing the analysis of Lemma G.12 and therefore Lemma G.8 by considering non-uniform costs between $[c_{\min}, 1]$ instead of costs all equal to 1. Specifically, Equation (G.9) needs to account for the inverse dependency on $c_{\min}$ of the simulation lemma of Lemma 2.14. This induces the final $\varepsilon/c_{\min}$ accuracy level achieved by the policies output by DisCo. There remains to guarantee that condition 2.4 of Lemma 2.14 is verified. In particular the condition holds if $\eta(L+1+\varepsilon) \leq 2c_{\min}$, where η is the model accuracy prescribed in the proof of Lemma G.8. We see that this is the case whenever we have $\varepsilon = O(Lc_{\min})$ due to the fact that $\eta = \Omega(\varepsilon/L^2)$.

G.1.9 Computational Complexity of DisCo

The overall computational complexity of DisCo can be expressed as $\sum_{k=1}^{K} |\mathcal{W}_k| \cdot C(\mathrm{OVI}_{\mathrm{SSP}})$, where $C(\mathrm{OVI}_{\mathrm{SSP}})$ denotes the complexity of an $\mathrm{OVI}_{\mathrm{SSP}}$ procedure and where we recall that K denotes the (random) index of the last round during which the algorithm terminates. Note that it holds with high probability that $K \leq |S^{\rightarrow}_{L+\varepsilon}|$ and $|\mathcal{W}_k| \leq 2LA|\mathcal{K}_k| \leq 2LA|S^{\rightarrow}_{L+\varepsilon}|$. Moreover $C(\mathrm{OVI}_{\mathrm{SSP}})$ captures the complexity of the value iteration (VI) algorithm for SSP, which was proved by Bonet (2007) to converge in time quadratic w.r.t. the size of the considered state

space (here, $\mathcal{K}_k$) and $\|V^\star\|_\infty/c_{\min}$. Here we have $c_{\min} = 1$, and we can easily prove that in all the SSP instances considered by DisCo, the optimal value function $V^\star$ verifies $\|V^\star\|_\infty = O(L^2)$, due to the restriction of the goal state in $\mathcal{W}_k$ (indeed this restriction implies that there exists a state-action pair in $\mathcal{K}_k \times \mathcal{A}$ that transitions to the goal state with probability $\Omega(1/L)$ in the true MDP). Putting everything together gives DisCo's computational complexity. Interestingly, we notice that while it depends polynomially on $S_{L+\varepsilon}$, L and A, it is independent from S the size of the global state space.

G.2 The UcbExplore Algorithm (Lim and Auer, 2012)

G.2.1 Outline of the Algorithm

The UcbExplore algorithm was introduced by Lim and Auer (Lim and Auer, 2012) to specifically tackle condition AX_L. The algorithm maintains a set $\mathcal{K}$ of "controllable" states and a set $\mathcal{U}$ of "uncontrollable" states. It alternates between two phases of *state discovery* and *policy evaluation*. In a state discovery phase, new candidate states are discovered as potential members of the set of controllable states. Any policy evaluation phase is called a *round* and it relies on an optimistic principle: it attempts to reach an "optimistic" state s (i.e., the easiest state to reach based on information collected so far) among all the candidate states by executing an optimistic policy π_s that minimizes the optimistic expected hitting time truncated at a horizon of $H_{\text{Ucb}} \triangleq \lceil L + L^2\varepsilon^{-1} \rceil$. Within the round of evaluation of policy π_s, the algorithm proceeds through at most $\lambda_{\text{Ucb}} \triangleq \lceil 6L^3\varepsilon^{-3} \log\left(16|\mathcal{K}|^2\delta^{-1}\right) \rceil$ episodes, each of which begins at s_0 and ends either when π_s successfully reaches s or when H_{Ucb} steps have been executed. If the *empirical performance* of π_s is poor (measured through a performance check done after each episode), the round is said to have *failed*. Otherwise, the round is *successful* which means that s is controllable and an acceptable policy (π_s) has been discovered. A failure round leads to selecting another candidate state-policy pair for evaluation, while a success round leads to a state discovery phase which in turn adds more candidate states for the subsequent rounds. Note that UcbExplore is unable to tackle the more challenging objective $\text{AX}^\star$.

G.2.2 Minor Issue and Fix in the Analysis of UcbExplore

The key insight of UcbExplore is to bound the number of *failure rounds* of the algorithm, by lower- and upper-bounding the so-called "regret" contribution of failure rounds, where the regret of a failure round k is defined as

$$\sum_{j=1}^{e_k} \Big[H_{\text{Ucb}} - L - \sum_{i=0}^{\Gamma-1} r_i \Big],$$

where $e_k \leq \lambda_{\text{UCB}}$ is the actual number of episodes executed in round k and where the reward $r_i \in \{0, 1\}$ is equal to 1 only if the state is the goal state. However, upper bounding the regret contribution of failure rounds implies applying a concentration inequality on *only* specific rounds that are chosen given their *empirical performance*. Hence Lim and Auer Lim and Auer, 2012, Lemma 18 improperly use a martingale argument to bound a sum whose summands are chosen in a non-martingale way, i.e., depending on their realization.

To avoid the aforementioned issue, one must upper and lower bound the cumulative regret of the *entire* set of rounds and not *only* the failure rounds in order to obtain a bound on the number of failure rounds. However, this would yield a sample complexity that has a second term scaling as $\widetilde{O}(\varepsilon^{-4})$. Following personal communication with the authors, the fix is to change the definition of regret of a round, making it equal to

$$\sum_{j=1}^{e_k} \widetilde{u}_{H_{\text{UCB}}}(s_0 \to s) - \sum_{i=0}^{H_{\text{UCB}}-1} r_i,$$

where s is the considered goal state and $\widetilde{u}_{H_{\text{UCB}}}(s_0 \to s)$ is the optimistic H_{UCB}-step reward (where the reward is equal to 1 only at state s). With this new definition, it is possible to recover the sample complexity provided by Lim and Auer (2012) scaling as $\widetilde{O}(\varepsilon^{-3})$.

G.2.3 Issue with a Possibly Infinite State Space

Lim and Auer (2012) claim that their setting can cope with a countable, possibly infinite state space. However, this leads to a technical issue, which has been acknowledged by the authors via personal communication and as of now has not been resolved. Indeed, it occurs when a union bound over the unknown set $\mathcal{U}$ is taken to guarantee high-probability statements (e.g., the Lemma 14 or 17 of Lim and Auer, 2012). Yet for each realization of the algorithm, we do not know what the set $\mathcal{U}$, or equivalently $\mathcal{K}$, looks like, hence it is improper to perform a union bound over a set of unknown identity. Simple workarounds to circumvent this issue are to impose a finite state space, or to assume prior knowledge over a finite superset of $\mathcal{U}$. In this paper we opt for the first option. It remains an open and highly non-trivial question as to how (and whether) the framework can cope with an infinite state space.

G.2.4 Effective Horizon of the AX Problem and its Dependency on ε

UCBEXPLORE (Lim and Auer, 2012) designs finite-horizon problems with horizon $H_{\text{UCB}} \triangleq \lceil L + L^2\varepsilon^{-1} \rceil$ and outputs policies that reset every H_{UCB} time steps. In the following we prove that the effective horizon of the AX problem actually scales as $O\left(\log(L\varepsilon^{-1})L\right)$, i.e., only *logarithmically* w.r.t. ε^{-1}. We begin by defining the concept of "resetting" policies as follows.

Definition G.15. *For any $\pi \in \Pi$ and horizon $H \geq 0$, we denote by $\pi^{|H}$ the non-stationary policy that executes the actions prescribed by π and performs the a_{reset} action every H steps, i.e.,*

$$\pi_t^{|H}(a|s) \triangleq \begin{cases} a_{\text{reset}} & \text{if } t \equiv 0 \ (\text{mod } H), \\ \pi(a|s) & \text{otherwise.} \end{cases}$$

We denote by $\Pi^{|H}$ the set of such "resetting" policies.

The following lemma captures the effective horizon H_{eff} of the problem, in the sense that restricting our attention to $\Pi^{|H}(\mathcal{S}_L^{\rightarrow})$ for $H \geq H_{\text{eff}}$ does not compromise the possibility of finding policies that achieve the performance required by $\text{AX}^\star$ (and thus also by AX_L).

Lemma G.16. *For any $\varepsilon \in (0, 1]$ and $L \geq 1$, whenever*

$$H \geq H_{\text{eff}} \triangleq 4(L+1)\lceil \log\left(\frac{4(L+1)}{\varepsilon}\right)\rceil,$$

we have for any $s^\dagger \in \mathcal{S}_L^{\rightarrow}$,

$$\min_{\pi^{|H} \in \Pi^{|H}(\mathcal{S}_L^{\rightarrow})} v_{\pi^{|H}}(s_0 \to s^\dagger) \leq V^\star_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s^\dagger) + \varepsilon.$$

Proof. Consider any goal state $s^\dagger \in \mathcal{S}_L^{\rightarrow}$. Set $\varepsilon' \triangleq \frac{\varepsilon}{2(L+1)} \leq \frac{1}{2}$. Denote by $\pi \in \Pi(\mathcal{S}_L^{\rightarrow})$ the minimizer of $V^\star_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s^\dagger)$. For any horizon $H \geq 0$, we introduce the truncated value function $v_{\pi,H}(s \to s') \triangleq \mathbb{E}\left[\tau_\pi(s \to s') \wedge H\right]$ and the tail probability $q_{\pi,H}(s \to s') \triangleq \mathbb{P}(\tau_\pi(s \to s') > H)$. Due to the presence of the a_{reset} action, the value function of π can be bounded for all states $s \in \mathcal{S}_L^{\rightarrow} \setminus \{s^\dagger\}$ as

$$V^\pi(s \to s^\dagger) \leq V^\star_{\mathcal{S}_L^{\rightarrow}}(s_0 \to s^\dagger) + 1 \leq L + 1.$$

This entails that the probability of the goal-reaching time decays exponentially. More specifically, we have

$$q_{\pi,H}(s_0 \to s^\dagger) \leq 2\exp\left(-\frac{H}{4(L+1)}\right) \leq \varepsilon', \tag{G.10}$$

where the first inequality stems from Lemma G.14 and the second inequality comes from the choice of $H \geq 4(L+1)\lceil \log\left(\frac{2}{\varepsilon'}\right)\rceil$. Furthermore, we have $\tau_\pi(s \to s') \wedge H \leq \tau_\pi(s \to s')$ and thus

$\mathbb{E}\left[\tau_\pi(s \to s') \wedge H\right] \leq \mathbb{E}\left[\tau_\pi(s \to s')\right]$. Consequently,

$$v_{\pi,H}(s_0 \to s^\dagger) \leq V^\pi(s_0 \to s^\dagger) = V^\star_{\mathcal{S}^\to_L}(s_0 \to s^\dagger). \tag{G.11}$$

Now, from Lim and Auer, 2012, Equation 4, the value function of π can be related to its truncated value function and tail probability as follows

$$v_{\pi|H} = \frac{v_{\pi,H} + q_{\pi,H}}{1 - q_{\pi,H}}. \tag{G.12}$$

Plugging Equations (G.10) and (G.11) into Equation (G.12) yields

$$v_{\pi|H}(s_0 \to s^\dagger) \leq \frac{V^\star_{\mathcal{S}^\to_L}(s_0 \to s^\dagger) + \varepsilon'}{1 - \varepsilon'}.$$

Notice that the inequalities $\frac{1}{1-x} \leq 1 + 2x$ and $\frac{x}{1-x} \leq 2x$ hold for any $0 < x \leq \frac{1}{2}$. Applying them for $x = \varepsilon'$ yields

$$\frac{V^\star_{\mathcal{S}^\to_L}(s_0 \to s^\dagger) + \varepsilon'}{1 - \varepsilon'} \leq (1 + 2\varepsilon')V^\star_{\mathcal{S}^\to_L}(s_0 \to s^\dagger) + 2\varepsilon'.$$

From the inequality $V^\star_{\mathcal{S}^\to_L}(s_0 \to s^\dagger) \leq L$ and the definition of ε', we finally obtain

$$v_{\pi|H}(s_0 \to s^\dagger) \leq V^\star_{\mathcal{S}^\to_L}(s_0 \to s^\dagger) + \varepsilon,$$

which completes the proof. □

Lemma G.16 reveals that the effective horizon H_{eff} of the AX problem scales only logarithmically and not linearly in ε^{-1}. This highlights that the design choice in UcbExplore to tackle finite-horizon problems with horizon H_{Ucb} unavoidably leads to a suboptimal dependency on ε in its AX_L sample complexity bound. In contrast, by designing SSP problems and thus leveraging the intrinsic goal-oriented nature of the problem, DisCo can (implicitly) capture the effective horizon of the problem. This observation is at the heart of the improvement in the ε dependency from $\widetilde{O}(\varepsilon^{-3})$ of UcbExplore (Lim and Auer, 2012) to $\widetilde{O}(\varepsilon^{-2})$ of DisCo (Theorem 9.11).

G.3 Experiments

This section complements the experimental findings partially reported in Section 9.4. We provide details about the algorithmic configurations and the environments as well as additional experiments.

G.3.1 Algorithmic Configurations

Experimental improvements to UcbExplore (Lim and Auer, 2012). We introduce several modifications to UcbExplore in order to boost its practical performance. We remove all the constants and logarithmic terms from the requirement for state discovery and policy evaluation (refer to Lim and Auer, 2012, Figure 1). Furthermore, we remove the constants in the definition of the accuracy $\varepsilon' = \varepsilon/L$ used by UcbExplore (while their original algorithm requires ε' to be divided by 8, we remove this constant). We also significantly improve the planning phase of UcbExplore (Lim and Auer, 2012, Figure 2). Their procedure requires to divide the samples into $H := (1 + 1/\varepsilon')L$ disjoint sets to estimate the transition probability of each stage h of the finite-horizon MDP. This substantially reduces the accuracy of the estimated transition probability since for each stage h only $N_k(s,a)/H$ are used. In our experiments, we use all the samples to estimate a stationary MDP (i.e., $\widehat{P}_k(s'|s,a) = N_k(s,a,s')/N_k(s,a)$) rather than a stage-dependent model. Estimating a stationary model instead of bucketing the data is simpler and more efficient since leads to a higher accuracy of the estimated model. To avoid to move too far away from the original UcbExplore, we decided to define the confidence intervals as if bucketing was used. We thus consider $\underline{N}_k(s,a) = N_k(s,a)/H$ for the construction of the confidence intervals. For planning, we use the optimistic backward induction procedure as in Azar et al. (2017). We thus leverage empirical Bernstein inequalities —which are much tighter— rather than Hoeffding inequalities as suggested in Lim and Auer (2012). In particular, we further approximate the bonus suggested in Azar et al. (2017, Algorithm 4) as

$$b_h(s,a) = \sqrt{\frac{Var_{s'\sim\hat{p}_k(\cdot|s,a)}[V_{k,h+1}(s')]}{\underline{N}_k(s,a)\vee 1}} + \frac{(H-h)}{\underline{N}_k(s,a)\vee 1}.$$

For DisCo, we follow the same approach of removing constants and logarithmic terms. We thus use the definition of ϕ as in Theorem 9.11 with $\alpha = 1$ and without log-terms. For planning, we use the procedure described in Section G.1 with $b_k(s,a,s') = \sqrt{\frac{\widehat{P}_k(s'|s,a)(1-\widehat{P}_k(s'|s,a))}{N_k(s,a)\vee 1}} + \frac{1}{N_k(s,a)\vee 1}$. Finally, in the experiments we use a state-action dependent value $\widehat{\Theta}(s,a,\mathcal{K}_k) = \left(\sum_{s'\in\mathcal{K}_k}\sqrt{\widehat{P}_k(s'|s,a)(1-\widehat{P}_k(s'|s,a))}\right)^2$ instead of taking the maximum over (s,a).

Even though we boosted the practical performance of UcbExplore w.r.t. the original algorithm proposed by Lim and Auer (2012) (e.g., the use of Bernstein), we believe it makes the comparison between DisCo and UcbExplore as fair as possible.

G.3.2 Confusing Chain

The *confusing chain* environment referred to in Section 9.4 is constructed as follows. It is an MDP composed of an initial state s_0, a chain of length C (states are denoted by $s_1,\dots,s_C$) and

ε	DisCo	UcbExplore-Bernstein
0.1	374,263 (13,906)	5,076,688 (92,643)
0.2	105,569 (4,645)	636,580 (13,716)
0.4	29,160 (829)	108,894 (2,305)
0.6	15,349 (475)	40,538 (805)
0.8	9,891 (244)	21,270 (441)

Table G.1 – Sample complexity of DisCo and UcbExplore-Bernstein, on the confusing chain domain. Values are averaged over 50 runs and the 95%-confidence interval of the mean is reported in parenthesis.

a set of K confusing states $(s_{C+1}, \dots, s_{C+K})$. Two actions are available in each state. In state s_0, we have a forward action a_0 that moves to the chain with probability p_c ($P(s_1|s_0, a_0) = p_c$ and $P(s_0|s_0, a_0) = 1 - p_c$) and a confusing action that has uniform probability of reaching any confusing state ($P(s_i|s_0, a_1) = 1/K$ for any $i \in \{C+1, \dots, C+K\}$). In the confusing states, all actions move deterministically to the end of the chain ($P(s_C|s_i, a) = 1$ for any $i \in \{C+1, \dots, C+K\}$ and a). In each state of the chain, there is a forward action a_0 that behaves as in s_0 ($P(s_{\min(C,i+1)}|s_i, a_0) = p_c$ and $P(s_i|s_i, a_0) = 1 - p_c$, for any $i \in \{1, \dots, C-1\}$) and a skip action a_1 that moves to m states ahead with probability p_{skip} ($P(s_{\min(C,i+m)}|s_i, a_0) = p_{\text{skip}}$ and $P(s_i|s_i, a_0) = 1 - p_{\text{skip}}$, for any $i \in \{1, \dots, C-1\}$). Finally, $P(s_0|s_c, a) = 1$ for any action a. In our experiments, we set $m = 4$, $p_{\text{skip}} = 1/3$, $p_c = 1$, $C = 5$, $K = 6$, $L = 4.5$.

Sample complexity. We provide in Table G.1 the sample complexity of the algorithms for varying values of ε. As mentioned in Section 9.4, DisCo outperforms UcbExplore for any value of ε, and increasingly so when ε decreases. Figure G.3 complements Figure 9.2 for additional values of ε.

Quality of goal-reaching policies. We now investigate the quality of the policies recovered by DisCo and UcbExplore. In particular, we show that DisCo is able to find the incrementally near-optimal shortest-path policies to any goal state, while UcbExplore may only recover sub-optimal policies. On the confusing chain domain, the intuition is that the set of confusing states makes s_C reachable in just 2 steps but the confusing states are not in the controllable set and thus the algorithms are not able to recover the shortest-path policy to s_C. On the other hand, state s_C is controllable through two policies: 1) the policies π_1 that takes always the forward action a_0 reaches s_C in 5 steps; 2) the policy π_2 that takes the skip action a_1 in s_1 reaches s_C in 4 steps. We observed empirically that DisCo always recovers policy π_1 (i.e., the fastest policy) while UcbExplore selects policy π_2 in several cases. This is highlighted in Table G.2 where we report the expected hitting time of the policies recovered by the algorithms. This finding is not surprising since, as we explain in Section 9.3, UcbExplore is designed to find policies reaching states in *at most* L steps on average, yet it is not able to recover incrementally near-optimal shortest-path policies, as opposed to DisCo.

UcbExplore-Bernstein						
ε	Expected hitting time $v_\pi(s_0 \to s_i)$					
	s_0	s_1	s_2	s_3	s_4	s_5
$0.1, 0.2$	0	1	2	3	4	4
0.4	0	1	2	3	4	4.94 (0.04)
0.6	0	1	2	3.36 (0.11)	4	4.53 (0.07)
0.8	0	1	2	3.38 (0.11)	4.07 (0.07)	4.53 (0.06)

Table G.2 – Expected hitting time of state s_i of the goal-oriented policy π_{s_i} recovered by UcbExplore-Bernstein, on the confusing chain domain. DisCo recovers the optimal goal-oriented policy in all the runs and for all ε. The advantage of DisCo lies in its final policy consolidation step. Values are averaged over 50 runs and the 95%-confidence interval of the mean is reported in parenthesis (it is omitted when equal to 0). This shows that UcbExplore recovers the optimal goal-oriented policy in every run only for ε equal to 0.1 and 0.2.

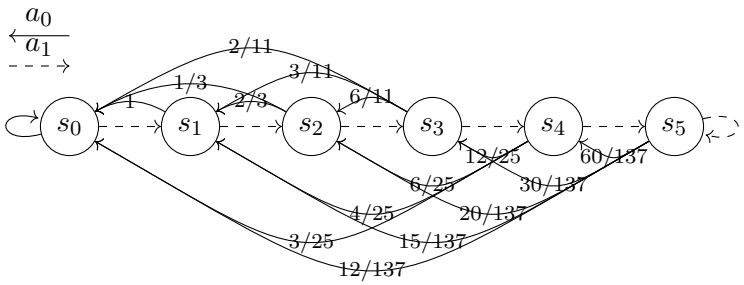


Figure G.1 – Combination lock domain with $S = 6$ states. Expected hitting times from the initial state s_3 are $v_\pi(s_3 \to s) = (2.18, 1.91, 1.64, 0, 1, 2)$. Consider $L = 3$, the set of incrementally L-controllable states is $\mathcal{S}_L^{\rightarrow} = \{s_2, s_3, s_4, s_5\}$. The goal-oriented policy to reach s_4 and s_5 takes always the right action a_1, while the policy for s_2 always selects the left action a_0.

G.3.3 Combination Lock

We consider the combination lock problem introduced by Azar et al. (2012). The domain is a stochastic chain with $S = 6$ states and $A = 2$ actions. In each state s_k, action *right* (a_1) is deterministic and leads to state s_{k+1}, while action *left* (a_0) moves to a state s_{k-l} with probability proportional to $1/(k-l)$ (i.e., inversely proportional to the distance of the states). Formally, we have that

$$n(x_k, x_l) = \begin{cases} \frac{1}{k-l} & \text{if } l < k \\ 0 & \text{otherwise} \end{cases} \qquad \text{and} \qquad P(x_l | x_k, a_0) = \frac{n(x_k, x_l)}{\sum_s n(x_k, s)}.$$

We set the initial state to be at $2/3$ of the chain, i.e., $\lfloor 2N/3 \rfloor$. The actions in the end states are absorbing, i.e., $P(s_0|s_0, a_0) = 1$ and $P(s_{N-1}|s_{N-1}, a_1) = 1$, while the remaining actions behave normally. See Figure G.1 for an illustration of the domain.

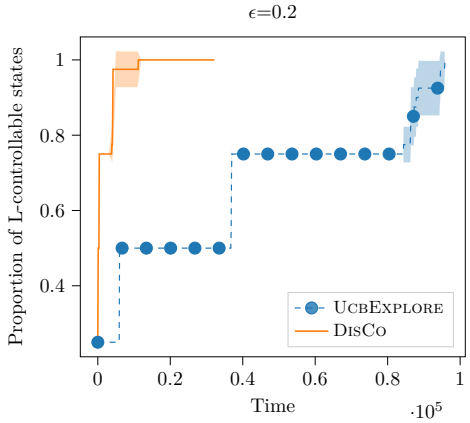


Figure G.2 – Proportion of the incrementally L-controllable states identified by DisCo and UcbExplore in the combination lock domain for $L = 2.7$ and $\varepsilon = 0.2$. Values are averaged over 20 runs.

Sample complexity. We evaluate the two algorithms DisCo and UcbExplore on the combination lock domain, for $\varepsilon = 0.2$ and $L = 2.7$. We further boost the empirical performance of UcbExplore by using N instead of $\underline{N}$ for the construction of the confidence intervals (i.e., we do not account for the data bucketing in Lim and Auer, 2012, see Section G.3.1). To preserve the robustness of the algorithm, we use $\log(|\mathcal{K}_k|^2)/(\varepsilon')^3$ episodes for UcbExplore's policy evaluation phase (indeed we noticed that the removal of the logarithmic term here sometimes leads UcbExplore to miss some states in $\mathcal{S}_L^{\rightarrow}$ in this domain). For the same reason, in DisCo we use the value $\widehat{\Theta}(\mathcal{K}_k) = \max_{s,a} \widehat{\Theta}(s, a, \mathcal{K}_k)$ prescribed by the theoretical algorithm instead of the state-action dependent values used in the previous experiment. We average the experiments over 20 runs and obtain a sample complexity of $30,117$ $(2,087)$ for DisCo and $90,232$ $(2,592)$ for UcbExplore. Figure G.2 reports the proportion of incrementally L-controllable states identified by the algorithms as a function of time. We notice that once again DisCo clearly outperforms UcbExplore.

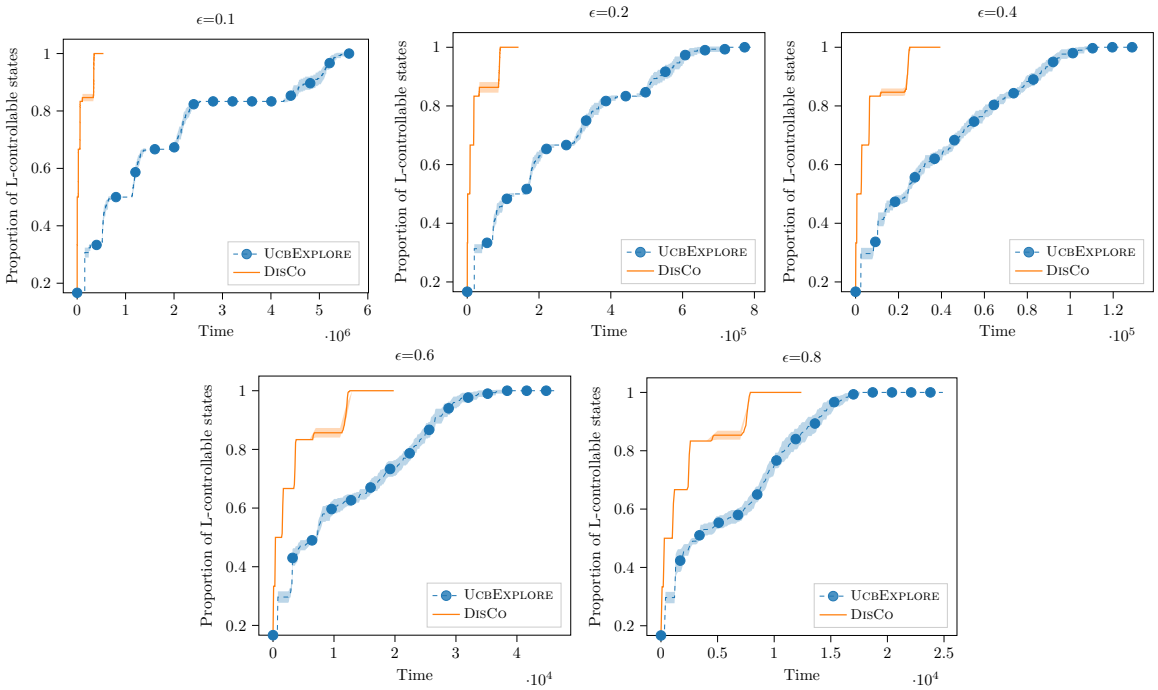


Figure G.3 – Proportion of the incrementally L-controllable states identified by DisCo and UcbExplore on the confusing chain domain for $L = 4.5$ and $\varepsilon \in \{0.1, 0.2, 0.4, 0.6, 0.8\}$. Values are averaged over 50 runs. UcbExplore uses Bernstein confidence intervals for planning.

List of Figures

List of Algorithms

List of Tables

List of References

Abbasi-Yadkori, Yasin, Dávid Pál, and Csaba Szepesvári (2011). Improved algorithms for linear stochastic bandits. *Advances in neural information processing systems* 24, pp. 2312–2320.

Achiam, Joshua, David Held, Aviv Tamar, and Pieter Abbeel (2017). Constrained policy optimization. In *International Conference on Machine Learning*. PMLR, pp. 22–31.

Agarwal, Alekh, Sham Kakade, and Lin F Yang (2020). Model-based reinforcement learning with a generative model is minimax optimal. In *Conference on Learning Theory*. PMLR, pp. 67–83.

Alterovitz, Ron, Thierry Siméon, and Ken Goldberg (2007). The stochastic motion roadmap: A sampling framework for planning with Markov motion uncertainty. In *Robotics: Science and systems*.

Altman, Eitan (1999). *Constrained Markov decision processes*. Vol. 7. CRC Press.

Amodei, Dario, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané (2016). Concrete problems in AI safety. *arXiv preprint arXiv:1606.06565*.

Andrychowicz, Marcin, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, et al. (2017). Hindsight experience replay. In *Advances in neural information processing systems*, pp. 5048–5058.

Audibert, Jean-Yves and Sébastien Bubeck (2010). Best Arm Identification in Multi-Armed Bandits. In *COLT - 23th Conference on Learning Theory*.

Audibert, Jean-Yves, Rémi Munos, and Csaba Szepesvári (2009). Exploration–exploitation tradeoff using variance estimates in multi-armed bandits. *Theoretical Computer Science* 410.19, pp. 1876–1902.

Ayoub, Alex, Zeyu Jia, Csaba Szepesvari, Mengdi Wang, and Lin Yang (2020). Model-based reinforcement learning with value-targeted regression. In *International Conference on Machine Learning*. PMLR, pp. 463–474.

Azar, Mohammad Gheshlaghi, Vicenç Gómez, and Hilbert J Kappen (2012). Dynamic policy programming. *Journal of Machine Learning Research* 13.Nov, pp. 3207–3245.

Azar, Mohammad Gheshlaghi, Rémi Munos, and Hilbert J Kappen (2013). Minimax PAC bounds on the sample complexity of reinforcement learning with a generative model. *Machine learning* 91.3, pp. 325–349.

Azar, Mohammad Gheshlaghi, Ian Osband, and Rémi Munos (2017). Minimax regret bounds for reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 263–272.

Azar, Mohammad Gheshlaghi, Bilal Piot, Bernardo Avila Pires, Jean-Bastian Grill, Florent Altché, and Rémi Munos (2019). World discovery models. *arXiv preprint arXiv:1902.07685*.

Badia, Adrià Puigdomènech, Pablo Sprechmann, Alex Vitvitskyi, Daniel Guo, Bilal Piot, Steven Kapturowski, et al. (2020). Never Give Up: Learning Directed Exploration Strategies. In *International Conference on Learning Representations*.

Baranes, Adrien and Pierre-Yves Oudeyer (2010). Intrinsically motivated goal exploration for active motor learning in robots: A case study. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, pp. 1766–1773.

Bartlett, Peter L and Ambuj Tewari (2009). REGAL: a regularization based algorithm for reinforcement learning in weakly communicating MDPs. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*.

Bartlett, Peter, Victor Gabillon, Jennifer Healey, and Michal Valko (2019). Scale-free adaptive planning for deterministic dynamics & discounted rewards. In *International Conference on Machine Learning*. PMLR, pp. 495–504.

Bäuerle, Nicole and Ulrich Rieder (2011). *Markov decision processes with applications to finance*. Springer Science & Business Media.

Bellemare, Marc G, Yavar Naddaf, Joel Veness, and Michael Bowling (2013). The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research* 47, pp. 253–279.

Bellemare, Marc, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos (2016). Unifying count-based exploration and intrinsic motivation. In *Advances in neural information processing systems*, pp. 1471–1479.

Bellman, Richard (1966). Dynamic programming. *Science* 153.3731, pp. 34–37.

Bertsekas, Dimitri (1991). *Linear network optimization: algorithms and codes*. Mit Press.

— (1995). *Dynamic programming and optimal control*. Vol. 2.

Bertsekas, Dimitri and John N Tsitsiklis (1991). An analysis of stochastic shortest path problems. *Mathematics of Operations Research* 16.3, pp. 580–595.

— (1995). Neuro-dynamic programming: an overview. In *Proceedings of 1995 34th IEEE conference on decision and control*. Vol. 1. IEEE, pp. 560–564.

Bertsekas, Dimitri and Huizhen Yu (2013). Stochastic shortest path problems under weak conditions. *Lab. for Information and Decision Systems Report LIDS-P-2909, MIT*.

Bhatnagar, Shalabh, Richard S Sutton, Mohammad Ghavamzadeh, and Mark Lee (2009). Natural actor–critic algorithms. *Automatica* 45.11, pp. 2471–2482.

Bonet, Blai (2007). On the speed of convergence of value iteration on stochastic shortest-path problems. *Mathematics of Operations Research* 32.2, pp. 365–373.

Bonet, Blai and Hector Geffner (2003a). Faster heuristic search algorithms for planning with uncertainty and full feedback. In *IJCAI*, pp. 1233–1238.

— (2003b). Labeled RTDP: Improving the Convergence of Real-Time Dynamic Programming. In *ICAPS*. Vol. 3, pp. 12–21.

Brafman, Ronen I and Moshe Tennenholtz (2002). R-max-a general polynomial time algorithm for near-optimal reinforcement learning. *Journal of Machine Learning Research* 3.Oct, pp. 213–231.

Brémaud, Pierre (2013). *Markov chains: Gibbs fields, Monte Carlo simulation, and queues*. Vol. 31. Springer Science & Business Media.

Cai, Haoyuan, Tengyu Ma, and Simon Shaolei Du (2022). Near-Optimal Algorithms for Autonomous Exploration and Multi-Goal Stochastic Shortest Path.

Campos, Víctor, Alex Trott, Caiming Xiong, Richard Socher, Xavier Giró Nieto, and Jordi Torres Viñals (2020). Explore, discover and learn: unsupervised discovery of state-covering skills. In *International Conference on Machine Learning*. PMLR, pp. 1317–1327.

Canfield, E Rodney and Carl Pomerance (2002). On the problem of uniqueness for the maximum Stirling number(s) of the second kind. *INTEGERS: Electronic Journal of Combinatorial Number Theory* 2.A01, p. 2.

Carpentier, Alexandra, Alessandro Lazaric, Mohammad Ghavamzadeh, Rémi Munos, and Peter Auer (2011). Upper-confidence-bound algorithms for active learning in multi-armed bandits. In *International Conference on Algorithmic Learning Theory*.

Chen, Liyu, Mehdi Jafarnia-Jahromi, Rahul Jain, and Haipeng Luo (2021a). Implicit Finite-Horizon Approximation and Efficient Optimal Algorithms for Stochastic Shortest Path. *arXiv preprint arXiv:2106.08377*.

Chen, Liyu, Rahul Jain, and Haipeng Luo (2021b). Improved No-Regret Algorithms for Stochastic Shortest Path with Linear MDP. *arXiv preprint arXiv:2112.09859*.

Chen, Liyu and Haipeng Luo (2021). Finding the Stochastic Shortest Path with Low Regret: the Adversarial Cost and Unknown Transition Case. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR, pp. 1651–1660.

Chen, Liyu, Haipeng Luo, and Aviv Rosenberg (2022). Policy Optimization for Stochastic Shortest Path. *arXiv preprint arXiv:2202.03334*.

Chen, Liyu, Haipeng Luo, and Chen-Yu Wei (2021c). Minimax regret for stochastic shortest path with adversarial costs and known transition. In *Conference on Learning Theory*. PMLR, pp. 1180–1215.

Chen, Xiaoyu, Jiachen Hu, Lin F Yang, and Liwei Wang (2021d). Near-Optimal Reward-Free Exploration for Linear Mixture MDPs with Plug-in Solver. *arXiv preprint arXiv:2110.03244*.

Chen, Yichen, Lihong Li, and Mengdi Wang (2018). Scalable Bilinear π Learning Using State and Action Features. *arXiv preprint arXiv:1804.10328*.

Chen, Yichen and Mengdi Wang (2016). Stochastic primal-dual methods and sample complexity of reinforcement learning. *arXiv preprint arXiv:1612.02516*.

Chentanez, Nuttapong, Andrew G Barto, and Satinder P Singh (2005). Intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pp. 1281–1288.

Cheung, Wang Chi (2019). Exploration-exploitation trade-off in reinforcement learning on online markov decision processes with global concave rewards. *arXiv preprint arXiv:1905.06466*.

Choi, Jongwook, Archit Sharma, Honglak Lee, Sergey Levine, and Shixiang Shane Gu (2021). Variational Empowerment as Representation Learning for Goal-Conditioned Reinforcement Learning. In *International Conference on Machine Learning*. PMLR, pp. 1953–1963.

Cohen, Alon, Yonathan Efroni, Yishay Mansour, and Aviv Rosenberg (2021). Minimax regret for stochastic shortest path. *Advances in Neural Information Processing Systems* 34.

Colas, Cédric, Pierre Fournier, Mohamed Chetouani, Olivier Sigaud, and Pierre-Yves Oudeyer (2019). CURIOUS: intrinsically motivated modular multi-goal reinforcement learning. In *International conference on machine learning*. PMLR, pp. 1331–1340.

Colas, Cédric, Tristan Karch, Olivier Sigaud, and Pierre-Yves Oudeyer (2020). Intrinsically motivated goal-conditioned reinforcement learning: a short survey. *arXiv preprint arXiv:2012.09830*.

Dann, Christoph, Tor Lattimore, and Emma Brunskill (2017). Unifying PAC and regret: Uniform PAC bounds for episodic reinforcement learning. *arXiv preprint arXiv:1703.07710*.

Dekel, Ofer, Jian Ding, Tomer Koren, and Yuval Peres (2014). Bandits with switching costs: $T^{2/3}$ regret. In *Proceedings of the forty-sixth annual ACM symposium on Theory of computing*, pp. 459–467.

Dennis, Michael, Natasha Jaques, Eugene Vinitsky, Alexandre Bayen, Stuart Russell, Andrew Critch, et al. (2020). Emergent complexity and zero-shot transfer via unsupervised environment design. *Advances in Neural Information Processing Systems* 33, pp. 13049–13061.

Dietterich, Thomas G (2000). Hierarchical reinforcement learning with the MAXQ value function decomposition. *Journal of artificial intelligence research* 13, pp. 227–303.

Domingues, Omar Darwiche, Yannis Flet-Berliac, Edouard Leurent, Pierre Ménard, Xuedong Shang, and Michal Valko (2021a). *rlberry-A Reinforcement Learning Library for Research and Education*.

Domingues, Omar Darwiche, Pierre Ménard, Emilie Kaufmann, and Michal Valko (2021b). Episodic reinforcement learning in finite mdps: Minimax lower bounds revisited. In *Algorithmic Learning Theory*. PMLR, pp. 578–598.

Eaton, JH and LA Zadeh (1962). Optimal pursuit strategies in discrete-state probabilistic systems.

Ecoffet, Adrien, Joost Huizinga, Joel Lehman, Kenneth O Stanley, and Jeff Clune (2020). First return then explore. *arXiv preprint arXiv:2004.12919*.

Elliot, Andrew J and James W Fryer (2008). The goal construct in psychology. *Handbook of motivation science* 18, pp. 235–250.

Eysenbach, Benjamin, Abhishek Gupta, Julian Ibarz, and Sergey Levine (2019). Diversity is All You Need: Learning Skills without a Reward Function. In *International Conference on Learning Representations*.

Florensa, Carlos, David Held, Xinyang Geng, and Pieter Abbeel (2018). Automatic Goal Generation for Reinforcement Learning Agents. In *International Conference on Machine Learning*, pp. 1515–1528.

Fruit, Ronan, Matteo Pirotta, and Alessandro Lazaric (2018a). Near optimal exploration-exploitation in non-communicating markov decision processes. In *Advances in Neural Information Processing Systems*, pp. 2994–3004.

Fruit, Ronan, Matteo Pirotta, and Alessandro Lazaric (2020). Improved analysis of ucrl2 with empirical bernstein inequality. *arXiv preprint arXiv:2007.05456*.

Fruit, Ronan, Matteo Pirotta, Alessandro Lazaric, and Ronald Ortner (2018b). Efficient bias-span-constrained exploration-exploitation in reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 1578–1586.

Fu, Justin, Aviral Kumar, Ofir Nachum, George Tucker, and Sergey Levine (2020). D4rl: Datasets for deep data-driven reinforcement learning. *arXiv preprint arXiv:2004.07219*.

Gregor, Karol, Danilo Jimenez Rezende, and Daan Wierstra (2016). Variational intrinsic control. *arXiv preprint arXiv:1611.07507*.

Grill, Jean-Bastien, Michal Valko, and Rémi Munos (2016). Blazing the trails before beating the path: Sample-efficient Monte-Carlo planning. In *Advances in Neural Information Processing Systems*, pp. 4680–4688.

Guillot, Matthieu and Gautier Stauffer (2020). The stochastic shortest path problem: a polyhedral combinatorics perspective. *European Journal of Operational Research* 285.1, pp. 148–158.

Guo, Zhaohan Daniel, Mohammad Gheshlaghi Azar, Alaa Saade, Shantanu Thakoor, Bilal Piot, Bernardo Avila Pires, et al. (2021). Geometric entropic exploration. *arXiv preprint arXiv:2101.02055*.

Hansen, Eric A (2011). Suboptimality bounds for stochastic shortest path problems. In *Proceedings of the Twenty-Seventh Conference on Uncertainty in Artificial Intelligence*, pp. 301–310.

Hansen, Eric A and Shlomo Zilberstein (2001). LAO*: A heuristic search algorithm that finds solutions with loops. *Artificial Intelligence* 129.1-2, pp. 35–62.

Hartikainen, Kristian, Xinyang Geng, Tuomas Haarnoja, and Sergey Levine (2020). Dynamical Distance Learning for Semi-Supervised and Unsupervised Skill Discovery. In *International Conference on Learning Representations*.

Hazan, Elad, Sham Kakade, Karan Singh, and Abby Van Soest (2019). Provably Efficient Maximum Entropy Exploration. In *International Conference on Machine Learning*, pp. 2681–2691.

Houthooft, Rein, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel (2016). Variational information maximizing exploration. *Advances in Neural Information Processing Systems (NIPS)*.

Jafarnia-Jahromi, Mehdi, Liyu Chen, Rahul Jain, and Haipeng Luo (2021). Online Learning for Stochastic Shortest Path Model via Posterior Sampling. *arXiv preprint arXiv:2106.05335*.

Jaksch, Thomas, Ronald Ortner, and Peter Auer (2010). Near-optimal Regret Bounds for Reinforcement Learning. *Journal of Machine Learning Research* 11.4.

Jiang, Nan (2020). Notes on Tabular Methods.

Jin, Chi, Zeyuan Allen-Zhu, Sebastien Bubeck, and Michael I Jordan (2018). Is Q-learning provably efficient? *Advances in neural information processing systems* 31.

Jin, Chi, Akshay Krishnamurthy, Max Simchowitz, and Tiancheng Yu (2020). Reward-free exploration for reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 4870–4879.

Joarder, Anwar H and Munir Mahmood (1997). An Inductive Derivation of Stirling Numbers of the Second Kind and their Applications in Statistics.

Kaelbling, Leslie Pack (1993). Learning to achieve goals. In *IJCAI*. Citeseer, pp. 1094–1099.

Kakade, Sham Machandranath (2003). *On the sample complexity of reinforcement learning*. University of London, University College London (United Kingdom).

Kamienny, Pierre-Alexandre, Jean Tarbouriech, Alessandro Lazaric, and Ludovic Denoyer (2022). Direct then Diffuse: Incremental Unsupervised Skill Discovery for State Covering and Goal Reaching. In *International Conference on Learning Representations*.

Kaufmann, Emilie, Pierre Ménard, Omar Darwiche Domingues, Anders Jonsson, Edouard Leurent, and Michal Valko (2021). Adaptive reward-free exploration. In *Algorithmic Learning Theory*. PMLR, pp. 865–891.

Kazerouni, Abbas, Mohammad Ghavamzadeh, Yasin Abbasi, and Benjamin Van Roy (2017). Conservative contextual linear bandits. In *Advances in Neural Information Processing Systems*, pp. 3910–3919.

Kearns, Michael, Yishay Mansour, and Andrew Y Ng (2000). Approximate planning in large POMDPs via reusable trajectories. In *Advances in Neural Information Processing Systems*, pp. 1001–1007.

— (2002). A sparse sampling algorithm for near-optimal planning in large Markov decision processes. *Machine learning* 49.2-3, pp. 193–208.

Kearns, Michael and Satinder Singh (2002). Near-optimal reinforcement learning in polynomial time. *Machine learning* 49.2, pp. 209–232.

Koenig, Sven and Reid G Simmons (1996). The effect of representation and knowledge on goal-directed exploration with reinforcement-learning algorithms. *Machine Learning* 22.1, pp. 227–250.

Kolobov, Andrey, Mausam, and Daniel S Weld (2012). A theory of goal-oriented MDPs with dead ends. In *Proceedings of the Twenty-Eighth Conference on Uncertainty in Artificial Intelligence*. AUAI Press, pp. 438–447.

Kolobov, Andrey, Mausam, Daniel Weld, and Hector Geffner (2011). Heuristic search for generalized stochastic shortest path MDPs. In *Proceedings of the International Conference on Automated Planning and Scheduling*. Vol. 21. 1.

Koren, Tomer, Roi Livni, and Yishay Mansour (2017). Bandits with movement costs and adaptive pricing. In *Conference on Learning Theory*. PMLR, pp. 1242–1268.

Kretzschmar, Henrik, Markus Spies, Christoph Sprunk, and Wolfram Burgard (2016). Socially compliant mobile robot navigation via inverse reinforcement learning. *The International Journal of Robotics Research* 35.11, pp. 1289–1307.

Latouche, Guy and Vaidyanathan Ramaswami (1999). *Introduction to matrix analytic methods in stochastic modeling*. SIAM.

Lattimore, Tor and Marcus Hutter (2012). PAC bounds for discounted MDPs. In *International Conference on Algorithmic Learning Theory*. Springer, pp. 320–334.

Lattimore, Tor and Csaba Szepesvári (2020). *Bandit algorithms*. Cambridge University Press.

Li, Gen, Yuting Wei, Yuejie Chi, Yuantao Gu, and Yuxin Chen (2020). Breaking the Sample Size Barrier in Model-Based Reinforcement Learning with a Generative Model. *Advances in neural information processing systems*.

Lillicrap, Timothy P, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, et al. (2015). Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*.

Lim, Shiau Hong and Peter Auer (2012). Autonomous exploration for navigating in MDPs. In *Conference on Learning Theory*, pp. 40–1.

Liu, Hao and Pieter Abbeel (2021). Behavior from the void: Unsupervised active pre-training. *Advances in Neural Information Processing Systems* 34.

Locke, Edwin A and Gary P Latham (2002). Building a practically useful theory of goal setting and task motivation: A 35-year odyssey. *American psychologist* 57.9, p. 705.

Maurer, Andreas and Massimiliano Pontil (2009). Empirical Bernstein bounds and sample variance penalization. *arXiv preprint arXiv:0907.3740*.

Ménard, Pierre, Omar Darwiche Domingues, Anders Jonsson, Emilie Kaufmann, Edouard Leurent, and Michal Valko (2021). Fast active learning for pure exploration in reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 7599–7608.

Menard, Pierre, Omar Darwiche Domingues, Xuedong Shang, and Michal Valko (2021). UCB Momentum Q-learning: Correcting the bias without forgetting. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR, pp. 7609–7618.

Merton, Robert C (1973). An intertemporal capital asset pricing model. *Econometrica: Journal of the Econometric Society*, pp. 867–887.

Min, Yifei, Jiafan He, Tianhao Wang, and Quanquan Gu (2021). Learning Stochastic Shortest Path with Linear Function Approximation. *arXiv preprint arXiv:2110.12727*.

Mohamed, Shakir and Danilo Jimenez Rezende (2015). Variational information maximisation for intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pp. 2125–2133.

Mutti, Mirco, Lorenzo Pratissoli, and Marcello Restelli (2021). Task-agnostic exploration via policy gradient of a non-parametric state entropy estimate. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 10, pp. 9028–9036.

Nair, Ashvin V, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine (2018). Visual Reinforcement Learning with Imagined Goals. *Advances in Neural Information Processing Systems* 31, pp. 9191–9200.

Norris, James R (1998). *Markov chains*. 2. Cambridge university press.

Oh, Junhyuk, Satinder Singh, Honglak Lee, and Pushmeet Kohli (2017). Zero-shot task generalization with multi-task deep reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 2661–2670.

Oudeyer, Pierre-Yves and Frederic Kaplan (2009). What is intrinsic motivation? A typology of computational approaches. *Frontiers in neurorobotics* 1, p. 6.

Pathak, Deepak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell (2017). Curiosity-driven exploration by self-supervised prediction. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pp. 16–17.

Pitis, Silviu, Harris Chan, Stephen Zhao, Bradly Stadie, and Jimmy Ba (2020). Maximum entropy gain exploration for long horizon multi-goal reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 7750–7761.

Plappert, Matthias, Marcin Andrychowicz, Alex Ray, Bob McGrew, Bowen Baker, Glenn Powell, et al. (2018). Multi-goal reinforcement learning: Challenging robotics environments and request for research. *arXiv preprint arXiv:1802.09464*.

Pong, Vitchyr, Murtaza Dalal, Steven Lin, Ashvin Nair, Shikhar Bahl, and Sergey Levine (2020). Skew-Fit: State-Covering Self-Supervised Reinforcement Learning. In *International Conference on Machine Learning*. PMLR, pp. 7783–7792.

Puterman, Martin L (2014). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

Qian, Jian, Ronan Fruit, Matteo Pirotta, and Alessandro Lazaric (2019). Exploration Bonus for Regret Minimization in Discrete and Continuous Average Reward MDPs. In *Advances in Neural Information Processing Systems*, pp. 4891–4900.

Rimon, Elon and Daniel E Koditschek (1992). Exact Robot Navigation Using Artificial Potential Functions. *Departmental Papers (ESE)*, p. 323.

Rosenberg, Aviv, Alon Cohen, Yishay Mansour, and Haim Kaplan (2020). Near-optimal regret bounds for stochastic shortest path. In *International Conference on Machine Learning*. PMLR, pp. 8210–8219.

Rosenberg, Aviv and Yishay Mansour (2021). Stochastic Shortest Path with Adversarially Changing Costs. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21*, pp. 2936–2942.

Russell, Stuart and Peter Norvig (2002). Artificial intelligence: a modern approach.

Schaul, Tom, Daniel Horgan, Karol Gregor, and David Silver (2015). Universal value function approximators. In *International conference on machine learning*, pp. 1312–1320.

Schmidhuber, Jürgen (1991). A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, pp. 222–227.

Schweitzer, Paul J (1985). On undiscounted Markovian decision processes with compact action spaces. *RAIRO-Operations Research* 19.1, pp. 71–86.

Sharma, Archit, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman (2020). Dynamics-Aware Unsupervised Discovery of Skills. In *International Conference on Learning Representations*.

Sidford, Aaron, Mengdi Wang, Xian Wu, Lin Yang, and Yinyu Ye (2018). Near-optimal time and sample complexities for solving Markov decision processes with a generative model. In *Advances in Neural Information Processing Systems*, pp. 5186–5196.

Singh, Satinder, Richard L Lewis, and Andrew G Barto (2009). Where do rewards come from. In *Proceedings of the annual conference of the cognitive science society*. Cognitive Science Society, pp. 2601–2606.

Singh, Satinder, Richard L Lewis, Andrew G Barto, and Jonathan Sorg (2010). Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Transactions on Autonomous Mental Development* 2.2, pp. 70–82.

Strehl, Alexander L, Lihong Li, and Michael L Littman (2009). Reinforcement learning in finite MDPs: PAC analysis. *Journal of Machine Learning Research* 10.Nov, pp. 2413–2444.

Strehl, Alexander L and Michael L Littman (2008). An analysis of model-based interval estimation for Markov decision processes. *Journal of Computer and System Sciences* 74.8, pp. 1309–1331.

Sutton, Richard S, Andrew G Barto, et al. (1998). *Introduction to reinforcement learning*. Vol. 135. MIT press Cambridge.

Szörényi, Balázs, Gunnar Kedenburg, and Rémi Munos (2014). Optimistic planning in Markov decision processes using a generative model. *Advances in Neural Information Processing Systems* 27, pp. 1035–1043.

Tang, Haoran, Rein Houthooft, Davis Foote, Adam Stooke, Xi Chen, Yan Duan, et al. (2017). # exploration: A study of count-based exploration for deep reinforcement learning. In *Advances in neural information processing systems*, pp. 2753–2762.

Tarbouriech, Jean, Omar Darwiche Domingues, Pierre Ménard, Matteo Pirotta, Michal Valko, and Alessandro Lazaric (2022). Adaptive Multi-Goal Exploration. In *International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 7349–7383.

Tarbouriech, Jean, Evrard Garcelon, Michal Valko, Matteo Pirotta, and Alessandro Lazaric (2020a). No-regret exploration in goal-oriented reinforcement learning. In *International Conference on Machine Learning*. PMLR, pp. 9428–9437.

Tarbouriech, Jean and Alessandro Lazaric (2019). Active exploration in markov decision processes. In *The 22nd International Conference on Artificial Intelligence and Statistics*. PMLR, pp. 974–982.

Tarbouriech, Jean, Matteo Pirotta, Michal Valko, and Alessandro Lazaric (2020b). Improved sample complexity for incremental autonomous exploration in mdps. *Advances in Neural Information Processing Systems* 33, pp. 11273–11284.

— (2021a). A provably efficient sample collection strategy for reinforcement learning. *Advances in Neural Information Processing Systems* 34, pp. 7611–7624.

— (2021b). Sample complexity bounds for stochastic shortest path with a generative model. In *Algorithmic Learning Theory*. PMLR, pp. 1157–1178.

Tarbouriech, Jean, Shubhanshu Shekhar, Matteo Pirotta, Mohammad Ghavamzadeh, and Alessandro Lazaric (2020c). Active model estimation in markov decision processes. In *Conference on Uncertainty in Artificial Intelligence*. PMLR, pp. 1019–1028.

Tarbouriech, Jean, Runlong Zhou, Simon S Du, Matteo Pirotta, Michal Valko, and Alessandro Lazaric (2021c). Stochastic shortest path: Minimax, parameter-free and towards horizon-free regret. *Advances in Neural Information Processing Systems* 34, pp. 6843–6855.

Tessler, Chen, Shahar Givony, Tom Zahavy, Daniel Mankowitz, and Shie Mannor (2017). A deep hierarchical approach to lifelong learning in minecraft. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 31. 1.

Tessler, Chen, Daniel J Mankowitz, and Shie Mannor (2019). Reward Constrained Policy Optimization. In *International Conference on Learning Representations*.

Trevizan, Felipe W, Florent Teichteil-Königsbuch, and Sylvie Thiébaux (2017). Efficient solutions for Stochastic Shortest Path Problems with Dead Ends. In *UAI*.

Vial, Daniel, Advait Parulekar, Sanjay Shakkottai, and R Srikant (2021). Regret Bounds for Stochastic Shortest Path Problems with Linear Function Approximation. *arXiv preprint arXiv:2105.01593*.

Wainwright, Martin (2015). *Course on Mathematical Statistics, chapter 2: Basic tail and concentration bounds. University of California at Berkeley, Department of Statistics*.

Wang, Mengdi (2017). Primal-Dual π Learning: Sample Complexity and Sublinear Run Time for Ergodic Markov Decision Problems. *arXiv preprint arXiv:1710.06100*.

Wang, Ruosong, Simon S. Du, Lin F. Yang, and Sham M. Kakade (2020a). Is Long Horizon RL More Difficult Than Short Horizon RL? In *Advances in Neural Information Processing Systems*.

Wang, Ruosong, Simon S Du, Lin Yang, and Russ R Salakhutdinov (2020b). On Reward-Free Reinforcement Learning with Linear Function Approximation. In *Advances in Neural Information Processing Systems*. Vol. 33, pp. 17816–17826.

Wang, Yuanhao, Kefan Dong, Xiaoyu Chen, and Liwei Wang (2019). Q-learning with UCB Exploration is Sample Efficient for Infinite-Horizon MDP. In *International Conference on Learning Representations*.

Warde-Farley, David, Tom Van de Wiele, Tejas Kulkarni, Catalin Ionescu, Steven Hansen, and Volodymyr Mnih (2019). Unsupervised Control Through Non-Parametric Discriminative Rewards. In *International Conference on Learning Representations*.

Weissman, Tsachy, Erik Ordentlich, Gadiel Seroussi, Sergio Verdu, and Marcelo J Weinberger (2003). Inequalities for the L1 deviation of the empirical distribution. *Hewlett-Packard Labs, Tech. Rep*.

White, Douglas J (1993). A survey of applications of Markov decision processes. *Journal of the operational research society* 44.11, pp. 1073–1096.

Wu, Jingfeng, Vladimir Braverman, and Lin F Yang (2020). Accommodating Picky Customers: Regret Bound and Exploration Complexity for Multi-Objective Reinforcement Learning. *arXiv preprint arXiv:2011.13034*.

— (2021). Gap-Dependent Unsupervised Exploration for Reinforcement Learning. *arXiv preprint arXiv:2108.05439*.

Yershov, Dmitry S and Steven M LaValle (2013). Simplicial Label Correcting Algorithms for continuous stochastic shortest path problems. In *2013 IEEE International Conference on Robotics and Automation*. IEEE, pp. 5062–5067.

Yu, Huizhen and Dimitri Bertsekas (2013). On boundedness of Q-learning iterates for stochastic shortest path problems. *Mathematics of Operations Research* 38.2, pp. 209–227.

Zanette, Andrea and Emma Brunskill (2019). Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*. PMLR, pp. 7304–7312.

Zanette, Andrea, Mykel J Kochenderfer, and Emma Brunskill (2019). Almost Horizon-Free Structure-Aware Best Policy Identification with a Generative Model. In *Advances in Neural Information Processing Systems*, pp. 5626–5635.

Zanette, Andrea, Alessandro Lazaric, Mykel J Kochenderfer, and Emma Brunskill (2020). Provably Efficient Reward-Agnostic Navigation with Linear Value Iteration. In *Advances in Neural Information Processing Systems*. Vol. 33, pp. 11756–11766.

Zhang, Chuheng, Yuanying Cai, Longbo Huang, and Jian Li (2021a). Exploration by maximizing Rényi entropy for reward-free RL framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*. Vol. 35. 12, pp. 10859–10867.

Zhang, Weitong, Dongruo Zhou, and Quanquan Gu (2021b). Reward-Free Model-Based Reinforcement Learning with Linear Function Approximation. In *Thirty-Fifth Conference on Neural Information Processing Systems*.

Zhang, Xuezhou, Yuzhe Ma, and Adish Singla (2020a). Task-agnostic Exploration in Reinforcement Learning. *Advances in Neural Information Processing Systems* 33.

Zhang, Yunzhi, Pieter Abbeel, and Lerrel Pinto (2020b). Automatic Curriculum Learning through Value Disagreement. In *Advances in Neural Information Processing Systems*. Vol. 33, pp. 7648–7659.

Zhang, Zihan, Simon Du, and Xiangyang Ji (2021c). Near Optimal Reward-Free Reinforcement Learning. In *International Conference on Machine Learning*. PMLR, pp. 12402–12412.

Zhang, Zihan and Xiangyang Ji (2019). Regret minimization for reinforcement learning by evaluating the optimal bias function. In *Advances in Neural Information Processing Systems*, pp. 2823–2832.

Zhang, Zihan, Xiangyang Ji, and Simon Du (2021d). Is reinforcement learning more difficult than bandits? a near-optimal algorithm escaping the curse of horizon. In *Conference on Learning Theory*. PMLR, pp. 4528–4531.

Zhang, Zihan, Jiaqi Yang, Xiangyang Ji, and Simon S Du (2021e). Variance-Aware Confidence Set: Variance-Dependent Bound for Linear Bandits and Horizon-Free Bound for Linear Mixture MDP. *arXiv preprint arXiv:2101.12745*.

Zhang, Zihan, Yuan Zhou, and Xiangyang Ji (2020c). Almost Optimal Model-Free Reinforcement Learning via Reference-Advantage Decomposition. *Advances in Neural Information Processing Systems* 33.

— (2021f). Model-Free Reinforcement Learning: from Clipped Pseudo-Regret to Sample Complexity. In *Proceedings of the 38th International Conference on Machine Learning*. PMLR, pp. 12653–12662.

Zhao, Rui, Xudong Sun, and Volker Tresp (2019). Maximum Entropy-Regularized Multi-Goal Reinforcement Learning. In *International Conference on Machine Learning*, pp. 7553–7562.

Zhou, Dongruo, Quanquan Gu, and Csaba Szepesvari (2021). Nearly minimax optimal reinforcement learning for linear mixture markov decision processes. In *Conference on Learning Theory*. PMLR, pp. 4532–4576.