ENS DE LYON

THÈSE
pour l'obtention du grade de Docteur, délivré par
l'ÉCOLE NORMALE SUPÉRIEURE DE LYON

École Doctorale N° 512
InfoMaths - Informatique et Mathématiques de Lyon

Discipline : MATHEMATIQUES (Mathématiques)

Soutenue publiquement le 11 Mars 2026, par :

Alexandre Marthe

Gestion du risque dans les Processus de Décision Markoviens : Approche Distributionnelle et Front Entropique

Risk-Sensitive Planning in Markov Decision Processes: Distributional Perspective and Entropic Front

---

Devant le jury composé de :

Vianney PERCHET, Professeur des universités, ENSAE	Rapporteur
Marc G. BELLEMARE, Adjunct Professor, Montréal & McGill Univ.	Rapporteur
Rémi MUNOS, Directeur Recherche, Meta	Examinateur
Claire VERNADE, Full Professor, University of Technology Nuremberg	Co-encadrante de thèse
Aurélien GARIVIER, Professeur des universités, ENS de Lyon	Directeur de thèse

Contents

Publications

This thesis is based on the following publications:

- Alexandre Marthe, Aurélien Garivier, and Claire Vernade. Beyond average return in markov decision processes. *Advances in Neural Information Processing Systems*, 36:56488–56507, 2023, also presented at EWRL 2023.

- Alexandre Marthe, Samuel Bounan, Aurélien Garivier, and Claire Vernade. Efficient risk-sensitive planning via entropic risk measures. *arXiv preprint arXiv:2502.20423*, 2025, presented at EWRL 2025.

- Alexandre Marthe, Mehrasa Ahmadipour, Aurélien Garivier, and Claire Vernade. Statistical complexity of the entropic risk measure. Work in progress, 2026.

Chapter 3 covers the results of Marthe et al. [2023] as well as additional insights and discussions not present in the original paper. Chapter 4 covers the results of Marthe et al. [2025]. Some work of this chapter are due to a collaboration with Samuel Bounan, intern student at ENS de Lyon in Summer of 2024. Finally, Chapter 5 presents novel results from Marthe et al. [2026] that have not been published yet. Some of those results are due to a collaboration with Mehrasa Ahmadipour, postdoctoral researcher at ENS de Lyon.

Abstract

Standard approaches in Markov Decision Processes (MDPs) typically focus on maximizing the *expected* return. Yet, many real-world applications require a consideration of risk that extends beyond the average outcome. This thesis investigates risk-sensitive sequential decision-making, aiming to optimize functionals of the return distribution beyond simple expectation.

The distributional approach has raised significant hope in this domain by allowing for the capture of the full return distribution. This framework theoretically facilitates the handling of complex risk metrics such as Value-at-Risk (VaR), Conditional Value-at-Risk (CVaR), and the Entropic Risk Measure (EntRM). This thesis rigorously investigates the capabilities and limitations of this approach, specifically studying which risk measures can be effectively optimized using dynamic programming.

Despite the promise of the distributional perspective, we uncover fundamental theoretical barriers. We characterize the set of risk measures amenable to dynamic programming and demonstrate that it is much narrower than previously assumed. In particular, we show that only a specific class of risk measures, the Entropic Risk Measure family, can be exactly optimized using standard dynamic programming recursion.

However, this family proves to be crucial, as it appears naturally in the approximation of other important risk measures. Building on this insight, we propose a unified planning framework. This method leverages the full spectrum of risk-sensitive behaviors offered by the entire EntRM family (the *Optimality Front*), for which we prove key structural properties. Inspired by these properties, we develop an algorithm called DOLFIN (Distributional Optimality Front Iteration) to approximately solve otherwise intractable objectives (VaR, CVaR, Threshold Probabilities) via Generalized Policy Improvement.

Finally, we investigate the problem of learning the EntRM under uncertainty to ensure reliable decision-making in environments with unknown dynamics. We derive statistical concentration bounds for its estimation and provide the first analysis of learning the EntRM for a full range of risk parameters simultaneously.

Résumé

Les approches classiques des Processus de Décision Markoviens (MDP) se concentrent généralement sur la maximisation du retour *espéré*. Pourtant, de nombreuses applications du monde réel nécessitent une prise en compte du risque allant au-delà du résultat moyen. Cette thèse étudie la prise de décision séquentielle sensible au risque, visant à optimiser des fonctionnelles de la distribution des retours au-delà de la simple espérance.

Un espoir a été soulevé par l'approche distributionnelle, qui permet de capturer l'intégralité de la distribution du retour. Cette approche offre, en théorie, un moyen d'aborder plus aisément des métriques de risque telles que la Value-at-Risk (VaR), la Conditional Value-at-Risk (CVaR) et la Mesure de Risque Entropique (EntRM). Cette thèse examine rigoureusement les capacités et les limites de cette approche, en étudiant quelles mesures de risque peuvent être efficacement optimisées par programmation dynamique.

Malgré les promesses de la perspective distributionnelle, nous mettons en lumière des barrières théoriques fondamentales. Nous caractérisons l'ensemble des mesures de risque se prêtant à une optimisation par programmation dynamique et montrons qu'il est beaucoup plus restreint qu'on ne le supposait auparavant. En particulier, seule une classe spécifique de mesures de risque, la famille des Mesures de Risque Entropiques, peut être optimisée de manière exacte via la programmation dynamique.

Cette famille s'avère cependant cruciale, car elle apparaît naturellement dans l'approximation d'autres mesures de risque importantes. Sur la base de ce constat, nous proposons un cadre d'optimisation unifié appelé DOLFIN (Distributional Optimality Front Iteration). Cette méthode exploite le spectre complet des comportements sensibles au risque offerts par la famille EntRM (le Front d'Optimalité), pour lequel nous prouvons des propriétés structurelles. Inspirés par ces propriétés, nous développons un algorithme permettant de résoudre approximativement des objectifs autrement intraitables (VaR, CVaR, Probabilités de Seuil) via le principe d'Amélioration de Politique Généralisée (Generalized Policy Improvement).

Enfin, nous étudions le problème de l'apprentissage de l'EntRM sous incertitude afin de permettre une prise de décision fiable dans des environnements à la dynamique inconnue. Nous dérivons des bornes de concentration statistiques pour son estimation et fournissons la première analyse de l'apprentissage de l'EntRM pour une plage de paramètres de risque simultanément.

1

Introduction

1.1 Context and Motivations

Every day, individuals and autonomous agents must select sequences of actions to achieve long-term goals, often in contexts where the consequences of current decisions only manifest in the future. For example, a clinician prescribing a treatment plan may only observe therapeutic effects weeks or months later. This problem of *sequential decision-making* constitutes a fundamental challenge in fields ranging from robotics and operations research to healthcare. Historically, the optimization of such trajectories has been the central focus of *Stochastic Optimal Control Theory*, which provides rigorous mathematical tools for planning when the system dynamics are fully known [Bertsekas, 2012].

However, real-world applications are rarely governed by fully known models. Returning to our example, a clinician often treats a patient without knowing exactly how their specific physiology will respond to a drug; the precise probabilistic outcomes are unknown a priori. To address such environments with unknown dynamics, *Reinforcement Learning* (RL) [Sutton et al., 1998, Lattimore and Szepesvári, 2020] has emerged as the dominant computational paradigm. The optimal strategies are learned through trial and error, allowing RL agents to discover effective policies even in complex, uncertain environments. This framework has achieved remarkable success in highly complex domains, from mastering game strategies to controlling autonomous vehicles [Mnih et al., 2015, Silver et al., 2018, Bellemare et al., 2020, Perolat et al., 2022].

Both Control Theory and RL share a common mathematical formalization: the *Markov Decision Process* (MDP) [Bellman, 1957, Puterman, 2014]. In this framework, an agent interacts with a stochastic environment over a sequence of time steps. At each step, the agent observes the current state, selects an action, and receives a reward along with a transition to a new state according to probabilistic dynamics. As the environment is stochastic, the standard objective is to maximize the *expected value* of the cumulative reward.

Yet, in many high-stakes applications, maximizing the average performance may be insufficient. In a clinical trial, a treatment maximizing the average recovery rate might be unacceptable if it causes severe side effects in a small subset of patients. Similarly, an autonomous vehicle optimizing for average travel time might increase the probability of collisions in rare but critical scenarios. In safety-critical domains, decision-makers must be *risk-sensitive*: they should be concerned not only with the average of outcomes but also with the variability and tail events [Garcıa and Fernández, 2015].

To induce risk-sensitive behavior, two primary methods exist. The first one, *reward shaping* [Silver et al., 2021, Ng et al., 1999], modifies the reward signal to align with the desired behaviors. While reasonable, this approach can be problematic: designing scalar rewards that capture precise constraints is notoriously difficult and may lead to unintended behaviors [Bowling et al., 2023, Amodei et al., 2016]. The second approach, which is also the one adopted in this thesis, is to modify the *optimization objective* itself. The environment is fixed and we optimize explicit *risk measures* (e.g., the variance or quantiles) of the return distribution instead of the expectation.

Optimizing risk measures in MDPs, however, introduces significant theoretical and computational challenges. The standard theory of Dynamic Programming for solving MDPs efficiently relies heavily on the linearity of the expectation [Puterman, 2014]. In practice, most risk measures do not satisfy that property, leading to a breakdown of the basic results such as the *Bellman Optimality Principle*. Solving it instead requires considering more complex algorithms that are often computationally intractable [Bäuerle and Rieder, 2014, Li et al., 2022].

In this context, *Distributional Reinforcement Learning* (DistRL) offers a promising path forward. By modeling the full probability distribution of returns rather than just a simple scalar such as the expectation [Bellemare et al., 2023], DistRL inherently captures the information required to evaluate those risk measures. Several works have proposed algorithms to optimize specific risk measures using the distributional framework [Dabney et al., 2018a, Achab and Neu, 2021, Lim and Malik, 2022]. Yet, those methods use either strong assumptions or lack any theoretical guarantees.

The objective of this thesis is precisely to explore the theoretical foundations and limitations of risk-sensitive objectives in MDPs, through the lens of this distributional perspective.

1.2 Outline and Overview of the Contributions

This thesis is divided into 5 chapters, including this one. Chapter 2 introduces the necessary background material: the MDP framework, the distributional perspective, and an introduction to risk measures. Then, the main contributions of this thesis are presented in the three following chapters. Chapter 3 studies how risk measures can be evaluated and optimized through dynamic programming with or without the distributional framework. Chapter 4 focuses on the Entropic Risk Measure (EntRM) family. It studies the set of optimal policies for the whole EntRM family and shows how it can be used to perform risk-sensitive planning for any risk measures. Finally, Chapter 5 focuses on statistical estimation of the EntRM.

Markov Decision Processes and the Notion of Risk

Chapter 2 introduces the three important notions studied in this thesis: MDPs and classical dynamic programming (Section 2.1), the distributional framework (Section 2.2), and the notion of risk measures in MDPs (Section 2.3).

Markov Decision Processes In Section 2.1, we present the undiscounted finite-horizon MDP with expected return framework, illustrated with the Windy Cliff environment. The key results are the Bellman Optimality Principle, along with the Bellman equation and the associated dynamic programming algorithms for both policy evaluation and policy optimization.

The Distributional Perspective We then introduce in Section 2.2 the distributional framework for MDPs. We explain how the previous Bellman equations and dynamic programming algorithms can be adapted to compute not only the expected return, but the full return distribution.

Policy Evaluation extends naturally to this setting (Section 2.2.1), but the Policy Optimization problem is more subtle as there is no canonical notion of *optimal distribution* (Section 2.2.2). It requires a criterion to compare distributions. For the expectation operator, it provides a Distributional Policy Optimization algorithm that outputs an optimal policy and its return distribution with optimal mean. It is however unclear at this point whether this algorithm can be used to optimize other functionals over distributions.

As distributions are infinite-dimensional objects, Section 2.2.3 presents two common parametrizations to approximate them in practice: the quantile and categorical parametrizations. We explain how to adapt distributional dynamic programming to work with those approximations.

Risk Measures Finally, Section 2.3 introduces the notion of (static) risk measures, which are functionals mapping $\varphi : \mathcal{P}(\mathbb{R}) \to \mathbb{R}$, from distributions to real values. We present in Section 2.3.1 a few classical risk measures which coincide with those studied in this thesis: the Entropic Risk Measure (EntRM) and more generally the Expected Utilities, and the most widely used quantiles-based risk measures: the Value at Risk (VaR) and the Conditional Value at Risk (CVaR). We give some counter-examples to show that the optimization of these risk measures in MDPs is more complicated than the expected return case, mainly because the *Bellman Optimality Principle* may not hold anymore. The following subsections focus on each of these classes of risk measures, presenting their definitions, properties, and optimization methods in MDPs.

Section 2.3.2 focuses on the EntRM family: $X \mapsto \frac{1}{\beta} \log \mathbb{E}[\exp(\beta X)]$, with $\beta \in \mathbb{R}$ the risk parameter. This family of risk measures is of great importance since they are also optimized by classical dynamic programming in standard MDPs. They give a natural extension of the expectation in our MDP framework.

Section 2.3.3 extends this to the more general Expected Utilities class of risk measures, of the form $X \mapsto \mathbb{E}[f(X)]$ for some utility function f. We explain that this broader class is harder to optimize and requires considering another state variable: the stock, representing the cumulated reward up to the current time step. This added variable makes the policy non-Markov and can lead to an exponential complexity.

Finally, Section 2.3.4 focuses on the most used risk measures in practice: the quantile-based risk measures, VaR and CVaR. VaR at level $\alpha \in (0, 1)$ is defined as the α-upper quantile of the distribution, while CVaR at level α is the expectation of the worst α-fraction of the outcomes. The VaR can be optimized through a method similar to the one used for Expected Utilities, augmenting the state space (Section 2.3.4.1). However, CVaR optimization is even more complicated even though it can also be related to expected utilities (Section 2.3.4.2).

Theoretical Limitations of Risk Measures in MDPs

Chapter 3 addresses a fundamental question: which risk measures can be evaluated and optimized using dynamic programming principles? We explore this in two contexts: the standard scalar value function approach and the richer distributional framework.

Policy Evaluation We first investigate the evaluation of risk metrics for a fixed policy (Section 3.1). For *Exact Evaluation*, we review in Section 3.1.1 the concept of *Bellman Closedness* from Rowland et al. [2019]. This concept characterizes the sets of statistics that can be computed exactly through standard dynamic programming. Previous works focus on the discounted MDP setting, so we explain how the results translate to the undiscounted finite-horizon setting we consider. In particular, only the power-exponential statistics are Bellman closed among the expected utilities.

We then turn to the distributional framework for *Approximate Evaluation* (Section 3.1.2). While DistRL theoretically allows evaluating any statistic, practical implementations rely on parametrized approximations (e.g., quantiles). Our main contribution here is a theoretical bound on the approximation error for a general class of Lipschitz statistics (Theorem 3.2). We prove that for metrics such as the CVaR, the error induced by the distributional approximation decays linearly with the resolution of the parametrization. We also analyze the tightness of this bound (Section 3.1.4), providing examples where the error is maximized.

Policy Optimization The optimization problem (Section 3.2) poses significantly greater challenges. We first clarify in Section 3.2.1 the link between optimization without distributions and *Bellman Closedness*. We explain that we may not hope to optimize a risk measure through dynamic programming without distributions unless it belongs to a Bellman closed set of statistics.

Then, Section 3.2.2 addresses distributional policy optimization. While Distributional RL provides the means to compute distributions, it does not automatically enable the optimization of arbitrary functionals over them. Our key contribution in this section is the formalization of *Bellman Optimization*, the property of a risk measure to be maximized by a greedy, recursive distributional algorithm, and its characterization. We identify two necessary conditions for this property, that we call *Independence* and *Translation*. Building on these properties, we prove a strict characterization result (Theorem 3.5): among continuous functionals, only the family of Exponential Utilities is Bellman Optimizable. This result is important as it formally establishes that popular risk measures like the Value-at-Risk or Conditional Value-at-Risk cannot be optimized exactly using standard dynamic programming, even within the distributional framework. This limitation motivates the algorithmic approach developed in the following chapter.

General Risk-Sensitive Planning through the Entropic Risk Measure

Chapter 4 bridges the gap between the EntRM being the only tractable risk measure for dynamic programming, and the practical need to optimize more interpretable metrics like VaR or CVaR. We propose a unified framework for risk-sensitive planning that leverages EntRM optimal policies as proxies for more general risk measures.

A Unified Framework We begin by establishing the connection between the EntRM and other risk measures (Section 4.1). Using Chernoff bounds and the concept of Entropic Value-at-Risk (EVaR), we show that optimizing metrics like VaR or CVaR can be approximated by a relaxed problem involving the EntRM optimal policies. This approximation motivates the application of the *Generalized Policy Improvement* (GPI) principle: after computing a set of optimal policies for various risk preferences (EntRM with different β), we can evaluate them on the desired risk-sensitive objective and achieve excellent performance even for intractable risk measures.

Structural Insights: The Optimality Front To solve this relaxed problem efficiently, we investigate the structure of the set of all EntRM-optimal policies as β varies, which we term the *Optimality Front* (Section 4.2). We prove a fundamental structural property (Proposition 4.4): the mapping from risk parameter to optimal policy is piecewise constant. We derive theoretical bounds on the location of the *breakpoints* where the optimal policy changes (Theorem 4.2) and give some insights on how the optimal policies evolve with risk preference.

Computing the Optimality Front Building on these structural insights, we introduce our main algorithmic contribution: DOLFIN (Distributional Optimality Front Iteration) in Section 4.3. Unlike naive grid-search approaches, DOLFIN efficiently computes the Optimality Front by iteratively identifying breakpoints. By applying the General Policy Improvement (GPI) principle over this finite set of policies, we obtain a flexible planning algorithm capable of addressing various risk-sensitive objectives.

Experimental Validation Finally, in Section 4.4, we empirically demonstrate the effectiveness of our approach. We show that DOLFIN outperforms state-of-the-art baselines in both computational efficiency and solution quality on standard environments such as Inventory Management and Windy Cliff.

Learning the Entropic Risk Measure

Having established how to plan with the EntRM when the model is known, Chapter 5 shifts focus to the problem of *learning* the EntRM from data in uncertain environments. This is crucial for bridging the gap between planning and Reinforcement Learning.

Fixed-Parameter Estimation We first address the fundamental statistical problem of estimating the EntRM from i.i.d. samples for a single, fixed risk parameter β (Section 5.1). We analyze the plug-in estimator and derive rigorous, non-asymptotic concentration bounds. Specifically, we provide Chernoff-type bounds for bounded random variables, quantifying the convergence rate and enabling the construction of confidence intervals.

Uniform Estimation and EVaR Recognizing that our planning framework relies on the entire spectrum of risk parameters, we extend our analysis to the simultaneous estimation of the EntRM over a continuous range of β (Section 5.2). We derive the first uniform concentration bounds for this problem, assuming discrete variables. Finally, we apply these results to the estimation of the EVaR, showing how uniform bounds on the EntRM translate into guarantees for the EVaR.

2

Markov Decision Processes and the Notion of Risk

Contents

2.1 Markov Decision Processes and Dynamic Programming

In sequential decision-making, an agent interacts with an environment over time. At each step, it observes the current situation, selects an action, and receives feedback in the form of a reward. The choice of action not only influences the immediate outcome but also affects the future evolution of the environment.

Consider, for instance, a bicycle store managing its inventory over a finite sales season. Each day, the manager observes the current stock level and decides how many bicycles to order from the supplier. Orders are fulfilled overnight, but customer demand for bicycles the next day is uncertain; it varies stochastically from day to day. If the store has enough stock to meet the demand, it earns revenue for each bicycle sold. Otherwise, it may lose potential sales to competitors and disappoint customers. At the same time, holding excess inventory incurs storage costs.

The manager's objective is to maximize the store's total profit over the season. However, this involves a delicate trade-off: ordering too few bicycles leads to missed sales, while ordering too many leads to high storage costs and unsold inventory. The manager must make these ordering decisions sequentially, taking into account the current stock level, the uncertain future demand, and the remaining time in the sales season.

This type of decision problem involves sequential choices under uncertainty where outcomes unfold over time and depend on both current decisions and random events. It is naturally modeled using the framework of *Markov Decision Processes* (MDPs).

2.1.1 Markov Decision Processes

Sequential decision-making is formalized using the framework of Markov Decision Processes (MDPs), introduced by Bellman [1957]. An MDP models the interaction between an agent and a stochastic environment over discrete time steps, in which the agent must select actions to maximize cumulative rewards. Formally, an MDP is defined as follows.

Definition 2.1. A Markov Decision Process (MDP) is defined as a tuple $\mathcal{M} = (\mathcal{X}, \mathcal{A}, p, r, H, x_0)$ where:

- $\mathcal{X}$ is the *state space*, containing all possible configurations of the environment.
- $\mathcal{A}$ is the *action space*, containing all possible actions the agent may take.
- $p : \mathcal{X} \times \mathcal{A} \to \mathcal{P}(\mathcal{X})$ is the *transition kernel*. We denote by $p(x' \mid x, a)$ the probability of transitioning to x' from x after taking action a.

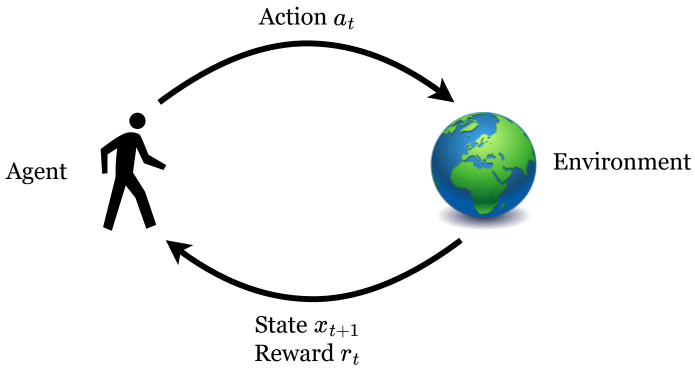

Figure 2.1: Illustration of a Markov Decision Process. The agent interacts with the environment by choosing actions, leading to rewards and changes in the environment.

- $r : \mathcal{X} \times \mathcal{A} \times \mathcal{X} \to [-1, 1]$ is the *reward function*, assigning a scalar reward to each transition. We write $r(x, a, x')$ for the reward obtained after choosing action a in state x and reaching state x'.
- $H \in \mathbb{N}$ is the *planning horizon*, indicating the number of time steps the agent can interact with the environment.
- $x_0 \in \mathcal{X}$ is the *initial state*, where the agent starts its interaction with the environment.

The Cliff Environment. To illustrate the elements of an MDP, we introduce a concrete example that will serve as a running case study throughout the thesis: the Windy Cliff environment [Sutton et al., 1998]. This environment is represented in Figure 2.2. In this stylized gridworld, the agent starts at the bottom-left of a rectangular grid and aims to reach a goal state located at the bottom-right. However, a cliff runs along part of the grid's edge: stepping into it results in a terminal negative outcome. Additionally, strong winds may push the agent off-course, which can be modeled by the transition kernel.

- The state $s \in \mathcal{X}$ corresponds to the current position of the agent on the grid. Here, $\mathcal{X} = [L] \times [W]$ where L is the width and W is the height of the grid.
- The action $a \in \mathcal{A}$ corresponds to one of four directions: up, down, left, or right. Thus, $\mathcal{A} = \{\text{up}, \text{down}, \text{left}, \text{right}\}$.
- The transition kernel models the movement and the wind. For instance, attempting to move north may lead to unintended movement to the east with some probability. Here,

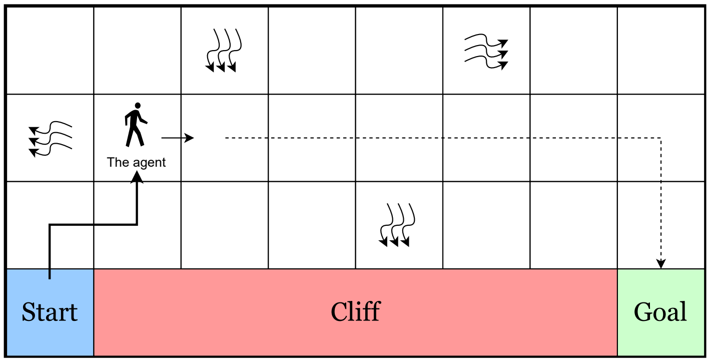

Figure 2.2: The Windy Cliff environment. The agent starts on the left (blue) and aims to reach the goal on the right (green) while avoiding the cliff (red area). Wind introduces stochasticity into transitions. In this example, $L = 8$ and $W = 4$.

- $p((x, y+1) \mid (x, y), \text{up}) = 0.7$
- $p((x+1, y) \mid (x, y), \text{up}) = p((x-1, y) \mid (x, y), \text{up}) = p((x, y-1) \mid (x, y), \text{up}) = 0.1$.

When the agent tries to move outside the grid, it remains in the same position. When the agent reaches the goal or falls into the cliff, it remains in that state until the end.

- The reward is $+1$ when the agent reaches the goal ($r(\cdot, \cdot, \text{goal}) = 1$) and -1 when it falls into the cliff ($r(\cdot, \cdot, \text{cliff}) = -1$). The agent receives an additional -0.01 reward penalty at every step, which encourages the agent to reach the goal quickly.
- The horizon defines the number of steps the agent has to reach the goal. It can be set arbitrarily. In this example, we set $H = 30$.
- The initial state $x_0 = (0, 0)$ is the bottom-left corner of the grid, where the agent starts its journey.

Trajectories. An agent interacting with an MDP generates a stochastic process over states, actions, and rewards. A realization of this process is called a *trajectory*, and takes the form

$$(x_0, a_0, r_0, x_1, a_1, r_1, \dots, x_{H-1}, a_{H-1}, r_{H-1}, x_H),$$

where x_0 is the initial state, a_t is the action taken at time t, x_{t+1} is the resulting state, and $r_t = r(x_t, a_t, x_{t+1})$ is the corresponding reward.

Histories. The sequence of all observations and actions up to time t is called the *history*, and is denoted $h_t = (x_0, a_0, r_0, \dots, x_t)$. We denote by $\mathcal{H}_t$ the set of all possible histories up to time t, so that $\mathcal{H}_t \subseteq (\mathcal{X} \times \mathcal{A} \times [-1, 1])^t \times \mathcal{X}$.

2.1.2 Decision Rules, Policies and the Return

Having formalized the environment as an MDP, we now turn to the agent. At each timestep, the agent must decide which action to take based on its observations so far. To model this behavior, we introduce the notions of *decision rules* and *policies*, which specify how the agent selects actions over time.

Definition 2.2. A *decision rule* at time t is a function $d_t : \mathcal{H}_t \to \mathcal{P}(\mathcal{A})$ that maps a history h_t to a distribution over actions. We write $d_t(a \mid h_t)$ for the probability of selecting action a given history h_t.

We distinguish several subclasses of decision rules:

- A decision rule is said to be *Markov* if it only depends on the current state: $\forall h_t,\ d_t(h_t) = d_t(x_t)$.
- It is *deterministic* if it selects an action with probability one: $\forall h_t,\ d_t(h_t)$ is a Dirac distribution.

We denote by D_{M}, D_{D}, and D_{MD} the sets of Markov, deterministic, and Markov-deterministic decision rules, respectively.

Definition 2.3. A *policy* is a sequence of decision rules $\pi = (\pi_0, \pi_1, \dots, \pi_{H-1})$, where π_t is the decision rule applied at time t.

A policy is said to be:

- *Markov* if it is only made out of Markov decision rules: $\forall t,\ \pi_t \in D_{\mathrm{M}}$.
- *Deterministic* if it is only made out of deterministic decision rules: $\forall t,\ \pi_t \in D_{\mathrm{D}}$.
- *Stationary* if the same Markov decision rule is used at every step: $\pi = (d, d, \dots, d)$ for some $d \in D_{\mathrm{M}}$. For Markov policies, we will use the same notation π for both the policy and the associated decision rule.

We write Π_{MD} and Π_{SD} for the sets of Markov-deterministic and stationary-deterministic policies, respectively.

Some Cliff policies. Consider the Cliff environment introduced earlier. There are multiple reasonable policies to choose from, depending on the agent's objectives and its *risk tolerance*. One natural option is to follow a path along the cliff edge: this allows the agent to reach the goal in fewer steps, but at the cost of a higher probability of falling into the cliff due to the wind. Alternatively, the agent may choose a longer but safer path along the outer boundary of the environment, thereby avoiding the risk of falling but incurring more negative rewards due to the longer trajectory.

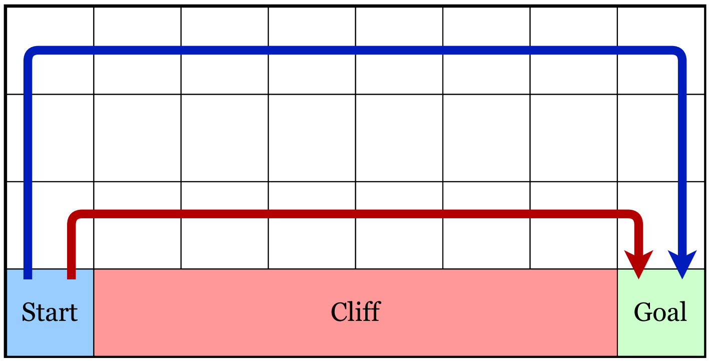

Figure 2.3: Two trajectories in the Windy Cliff environment. The red path is fast but risky, while the blue path is safe but slower.

In the policy described here, the agent chooses its next action solely based on the current position without taking the remaining time into account; this corresponds to a stationary policy. However, in some cases, this may not be the best approach. When the agent has only a few steps remaining before the trajectory ends, and is too far from the goal to reach it in time, it may be beneficial to adopt a different behavior.

In such situations, a time-dependent (i.e., non-stationary) policy may be more appropriate. The agent may decide to give up on reaching the goal and focus instead on minimizing the risk of falling into the cliff, even if it means not making progress toward the out-of-reach objective.

About the notion of state space. As explained, the state space $\mathcal{X}$ is intended to capture all relevant information about the environment at a given time. However, it is possible to incorporate additional information into the state representation, such as the remaining time steps before the horizon, or the entire history of observations and actions until a certain time (i.e., at a time t, the agent's state could be defined as $s_t = h_t$). Doing so can have some advantages which will be discussed in Section 2.3.3.2. Most notably, any policy becomes stationary and Markov in such an augmented state space.

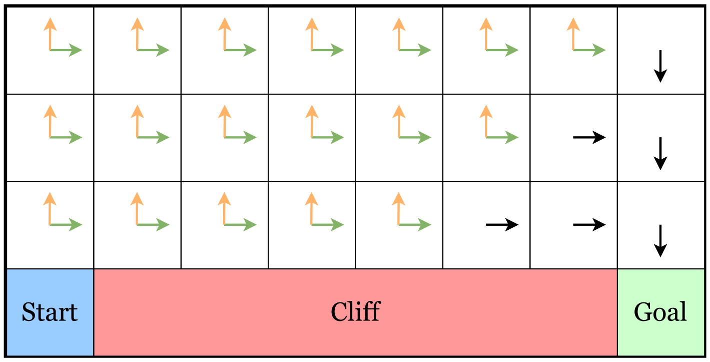

Figure 2.4: Illustration of a non-stationary policy in the Windy Cliff environment. In green is the decision rule at $t = 0$ while yellow arrows represent the decision rule at $t = H - 3$. In black are the actions that are common for both policies. With only three time steps remaining and the goal out of reach, the agent chooses to stay away from the cliff. This policy depends explicitly on the remaining time.

The Return. To understand the performance of different policies, we need a measure of how successful a policy is over time. This is usually captured by the notion of *Return*, which is the sum of the rewards collected by the agent over the course of a trajectory.

Definition 2.4. The *return* of a policy π, denoted by $\mathcal{R}^\pi$, is the random variable defined as:

$$\mathcal{R}^\pi = \sum_{t=0}^{H-1} r(X_t, A_t, X_{t+1}),$$

where the random trajectory $(X_0, A_0, X_1, \ldots, X_H)$ is generated by following policy π in the MDP, that is, $A_t \sim \pi_t(\cdot \mid \mathcal{H}_t)$ and $X_{t+1} \sim p(\cdot \mid X_t, A_t)$, starting from x_0.

For $t < H$ and when π is a policy that does not depend on history prior to time t (such as a Markov policy), we also define the *partial return*, for all $x \in \mathcal{X}$, $a \in \mathcal{A}$, by:

$$\mathcal{R}_t^\pi(x) = \sum_{s=t}^{H-1} r(X_s, A_s, X_{s+1}), \quad X_t = x$$

and

$$\mathcal{R}_t^\pi(x, a) = \sum_{s=t}^{H-1} r(X_s, A_s, X_{s+1}), \quad X_t = x, A_t = a$$

with $A_s \sim \pi_s(\cdot \mid X_t, A_t, R_t, \ldots, X_s)$ and $X_{s+1} \sim p(\cdot \mid X_s, A_s)$. Note that $\mathcal{R}_0^\pi(x_0) = \mathcal{R}^\pi$.

These partial returns are well-defined for Markov policies, as the decision rule at each step depends only on the current state and not on the past history. They will play a central role in the development of Dynamic Programming algorithms, which rely on recursively computing values from intermediate states or state-action pairs.

The Return in the Cliff Environment. To illustrate this, consider the two policies from Figure 2.3 for which we plotted the return distribution in Figure 2.5. It can be observed through the return that the risky policy may achieve higher return, but also lead to higher chances of failure. The safe policy, on the other hand, is less likely to fall into the cliff, but may accumulate less reward overall. The return distribution helps understand this trade-off between risk and reward.

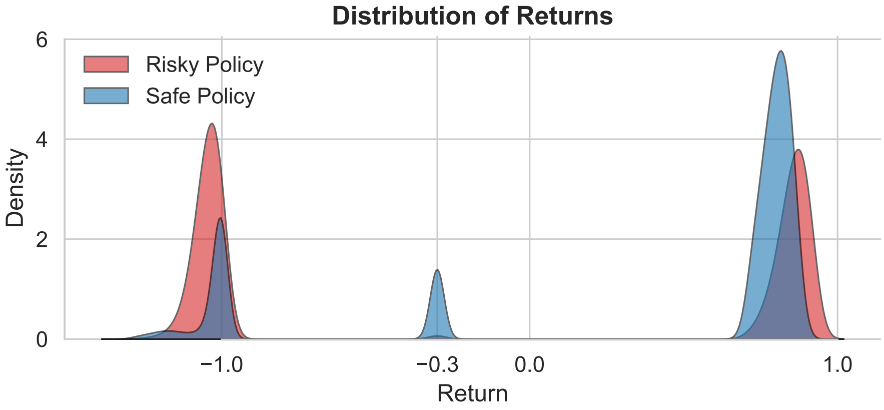

Figure 2.5: (Smoothed) Return distribution in the Cliff environment for the two policies mentioned in Figure 2.3. The risky policy (red) can achieve higher returns but exhibits high chances of failure. The safe policy (blue) has lower chances of failure but achieves lower returns.

The Expected Return. The return serves as a measure of the performance of a policy over a single trajectory. However, since the return is a random variable, comparing two policies requires a criterion to evaluate their respective return distributions. The most natural and commonly used approach is to consider their expected return, $\mathbb{E}[\mathcal{R}^\pi]$, which reflects the average performance of the policy across repeated interactions with the environment.

Definition 2.5. The *expected return* of a policy π is defined as

$$\mathbb{E}[\mathcal{R}^\pi] = \mathbb{E}_{\substack{A_t \sim \pi(\cdot|\mathcal{H}_t) \\ X_{t+1} \sim p(\cdot|X_t, A_t)}} \left[\sum_{t=0}^{H-1} r(X_t, A_t, X_{t+1}) \right].$$

In the remainder of this section, we focus on the standard objectives of evaluating and maximizing the expected return, which serves as the foundation for most classical results in Markov Decision Processes. We will only consider Markov policies, as we will see they are sufficient to achieve optimality.

2.1.3 Policy Evaluation for the Expected Return

Policy evaluation is the process of computing the expected return of a given policy. This is a crucial step in MDPs, as it allows us to evaluate the performance of different policies.

The Value Function. We start by defining the *value function* of a policy π, which quantifies the expected return starting from a given state and time, when following the policy thereafter.

Definition 2.6. Let $\pi = (\pi_0, \dots, \pi_{H-1})$ be a Markov policy. The *value function* of π at time t and state $x \in \mathcal{X}$ is defined as

$$V_t^\pi(x) = \mathbb{E}_\pi\left[\sum_{s=t}^{H-1} r(X_s, A_s, X_{s+1}) \,\middle|\, X_t = x\right] = \mathbb{E}\left[\mathcal{R}_t^\pi(x)\right],$$

where $\mathbb{E}_\pi$ denotes the expectation taken over the randomness of both the policy and the environment, that is, over the trajectory $(X_s, A_s)_{s\geq t}$ generated by:

$$A_s \sim \pi_s(\cdot \mid X_s), \quad X_{s+1} \sim p(\cdot \mid X_s, A_s).$$

We recover the expected return of a policy π as the value function starting at state x_0 and time $t = 0$: $\mathbb{E}[\mathcal{R}^\pi] = V_0^\pi(x_0)$.

Action-Value Function. Similarly, we define the *action-value function*, or *Q-function*, which quantifies the expected return when starting from a given state and action at time t, and then following π thereafter.

Definition 2.7. Let π be a Markov policy. The *action-value function* of π at time t is defined as:

$$Q_t^\pi(x, a) = \mathbb{E}_\pi\left[\sum_{s=t}^{H-1} r(X_s, A_s, X_{s+1}) \,\middle|\, X_t = x, A_t = a\right] = \mathbb{E}\left[\mathcal{R}_t^\pi(x, a)\right].$$

The value function and action-value function are related through the distribution of actions of π_t:

$$V_t^\pi(x) = \mathbb{E}_{a\sim\pi_t(\cdot|x)}\left[Q_t^\pi(x, a)\right]. \tag{2.1}$$

Bellman Equations. The value function and the action-value function satisfy recursive equations, known as the *Bellman equations* [Bellman, 1957].

For all $x \in \mathcal{X}$, $a \in \mathcal{A}$, and $t < H$:

$$V_t^\pi(x) = \mathbb{E}_{a\sim\pi_t(\cdot|x),\, x'\sim p(\cdot|x,a)}\left[r(x,a,x') + V_{t+1}^\pi(x')\right], \tag{2.2}$$

$$Q_t^\pi(x,a) = \mathbb{E}_{x'\sim p(\cdot|x,a),\, a'\sim\pi_{t+1}(\cdot|x')}\left[r(x,a,x') + Q_{t+1}^\pi(x',a')\right]. \tag{2.3}$$

The intuition is simple: the expected return starting from state x at time t is equal to the expected immediate reward plus the expected return starting from the next state x' at time $t+1$.

These equations are among the most important in the MDP literature as they form the basis of most algorithms. The first algorithm we consider is the *Policy Evaluation Algorithm* (Algorithm 1) [Sutton et al., 1998], which computes the expected return of a policy using dynamic programming. This algorithm runs in $\mathcal{O}(H|\mathcal{X}|^2|\mathcal{A}|)$ time.

Algorithm 1 Finite-Horizon Policy Evaluation

```
Require: MDP $(\mathcal{X}, \mathcal{A}, p, r, H)$, Markov policy $\pi = (\pi_0, \dots, \pi_{H-1})$
Ensure: Value function $V_t^\pi(x)$ for all $x \in \mathcal{X}$ and $t \in \{0, \dots, H\}$
Initialize $V_H^\pi(x) \leftarrow 0$ for all $x \in \mathcal{X}$
for $t = H-1 \to 0$ do
    for all $x \in \mathcal{X}$ do
        $V_t^\pi(x) \leftarrow \sum_{a\in\mathcal{A}} \pi_t(a \mid x) \sum_{x'\in\mathcal{X}} p(x' \mid x,a)\left[r(x,a,x') + V_{t+1}^\pi(x')\right]$
    end for
end for
output $V^\pi$
```

Action-value function evaluation. A similar algorithm can be used to compute the action-value function, by replacing the inner line of the loop with

$$Q_t^\pi(x,a) \leftarrow \sum_{x'\in\mathcal{X}} p(x' \mid x,a)\left[r(x,a,x') + \sum_{a'\in\mathcal{A}} \pi_{t+1}(a' \mid x')Q_{t+1}^\pi(x',a')\right] \tag{2.4}$$

iterated over all $a \in \mathcal{A}$. The action-value function will mainly be used for optimization algorithms.

History-dependent Policies. There also exists an algorithm to compute the value function of a history-dependent policy [Puterman, 2014]. This algorithm is similar to the Policy Evaluation Algorithm (Algorithm 1), but it iterates over all histories instead of states. Such an algorithm usually has a complexity of $\mathcal{O}(H|\mathcal{H}|^2)$, where $|\mathcal{H}|$ is the number of histories up to time H. This is usually intractable, as the number of histories grows exponentially with the horizon.

2.1.4 Planning for the Expected Return

Planning, in the context of MDPs, refers to the process of computing an optimal policy that maximizes the expected return. In this thesis, we will also use the term *policy optimization* to refer more broadly to the maximization of alternative objectives beyond the expected return.

Formally, planning consists of solving the following optimization problem:

$$\max_{\pi \in \Pi} \mathbb{E}[\mathcal{R}^\pi]$$

We denote by π^* a policy that achieves it, i.e., $V_0^{\pi^*}(x_0) = \max_\pi \mathbb{E}[\mathcal{R}^\pi]$. In general, there may not be a unique optimal policy.

This problem appears challenging at first, due to the maximization over the space of all history-dependent policies. However, a fundamental result guarantees that this maximum is always achieved by a Markov policy.

Proposition 2.1 (Puterman [2014])**.** For any MDP $(\mathcal{X}, \mathcal{A}, p, r, H, x_0)$, there exists a Markov deterministic policy $\pi^* \in \Pi_{MD}$ such that

$$V_0^{\pi^*}(x_0) = \max_{\pi \in \Pi} \mathbb{E}[\mathcal{R}^\pi].$$

This result ensures that we can restrict our attention to Markov (and even Markov deterministic) policies without loss of optimality when optimizing the expected return. We will see later that this may not be the case for other objectives.

Optimal Value Functions. Given this restriction, we now define the *optimal value function* and the *optimal action-value function* as the pointwise maximum of the value and Q-functions over all Markov policies.

Definition 2.8. For any $x \in \mathcal{X}$, $a \in \mathcal{A}$, and $t < H$, we define:

$$V_t^*(x) = \max_{\pi \in \Pi_{MD}} V_t^\pi(x),$$
$$Q_t^*(x, a) = \max_{\pi \in \Pi_{MD}} Q_t^\pi(x, a).$$

The optimal expected return is recovered with $V_0^{\pi^*}(x_0) = V_0^*(x_0)$.

Similarly to the value function, the optimal (action-)value function satisfies the *Optimal Bellman Equation*:

$$V_t^*(x) = \max_{a \in \mathcal{A}} \mathbb{E}_{X' \sim p(\cdot|x,a)} \left[r(x, a, X') + V_{t+1}^*(X') \right]. \tag{2.5}$$

$$Q_t^*(x, a) = \mathbb{E}_{X' \sim p(\cdot|x,a)} \left[r(x, a, X') + \max_{a' \in \mathcal{A}} Q_{t+1}^*(X', a') \right]. \tag{2.6}$$

A first crucial implication of the Optimal Bellman Equation is the *Bellman Optimality Principle*. It states that if a policy π^* is optimal, then for any starting state x and time t, the policy also optimizes the return starting from x at time t. Formally,

Proposition 2.2 (Bellman Optimality Principle)**.** Let $\pi^* \in \Pi$ be an optimal policy. Then, for all $x \in \mathcal{X}$ and $t < H$, if $\Pr_{\pi^*}(X_t = x) > 0$, then

$$V_t^{\pi^*}(x) = V_t^*(x), \qquad Q_t^{\pi^*}(x,a) = Q_t^*(x,a), \quad \forall a \in \mathcal{A}.$$

Or, as R. E. Bellman himself stated,

> "An optimal policy has the property that whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision."
>
> — Bellman [1957]

Value Iteration. Following the same principle of the Policy Evaluation Algorithm, this time using the Optimal Bellman Equation, the optimal value function and the optimal action-value function can be computed using dynamic programming. We call this algorithm *Value Iteration*. See Algorithm 2.

Algorithm 2 Finite-Horizon Value Iteration

```
Require: MDP $(\mathcal{X}, \mathcal{A}, p, r, H)$
Ensure: Optimal action-value function $Q_t^*(x,a)$ and greedy policy $\pi_t^*(x)$ for all $x \in \mathcal{X}$,
    $a \in \mathcal{A}$, and $t \in \{0, \dots, H\}$
Initialize $Q_H^*(x,a) \leftarrow 0$, $V_H^*(x) \leftarrow 0$ for all $x \in \mathcal{X}$, $a \in \mathcal{A}$
for $t = H - 1 \to 0$ do
    for all $x \in \mathcal{X}$ do
        for all $a \in \mathcal{A}$ do
            $Q_t^*(x,a) \leftarrow \sum_{x' \in \mathcal{X}} p(x' \mid x,a) \left[r(x,a,x') + V_{t+1}^*(x')\right]$
        end for
        $V_t^*(x) \leftarrow \max_{a \in \mathcal{A}} Q_t^*(x,a)$
        $\pi_t^*(x) \leftarrow \arg\max_{a \in \mathcal{A}} Q_t^*(x,a)$
    end for
end for
output $V^*$, $Q^*$, $\pi^*$
```

This algorithm also runs in $\mathcal{O}(H|\mathcal{X}|^2|\mathcal{A}|)$ time, similar to the Policy Evaluation Algorithm (Algorithm 1). Optimizing a policy has the same complexity as evaluating it. This is a key feature of MDPs, which makes them particularly suitable for planning.

Cliff optimal policy. To illustrate this notion of optimal policy, we plot in Figure 2.6 the optimal policy for the Cliff environment at different timesteps. This policy finds the best trade-off between the risky and safe policies mentioned in Figure 2.3. As we can see, the policy is not stationary: it changes as the time steps get closer to the horizon H. We observe the phenomenon mentioned in Figure 2.4.

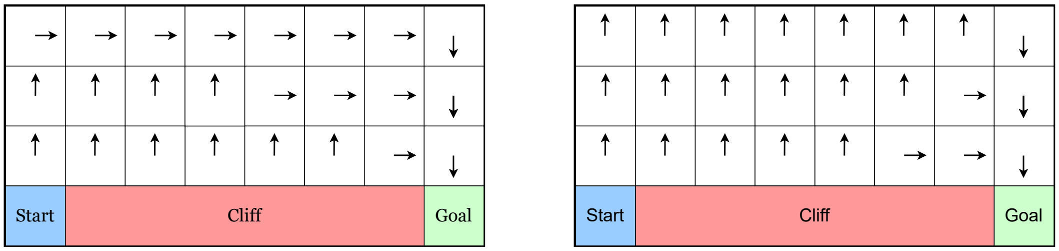

Figure 2.6: Optimal policy in the Cliff environment at different time steps. On the left, the policy at time $t = 0$. On the right, the policy at time $t = H - 3$. The agent's behavior changes as it approaches the goal, becoming more cautious to avoid falling into the cliff when far away from the goal, as mentioned with Figure 2.4.

We also report in Table 2.1 the expected return of the different policies in the Cliff environment.

Table 2.1: Expected return of different policies in the Cliff environment.

Policy	Safe Policy	Risky Policy	Optimized Policy
Expected Return	0.379	-0.105	**0.409**

Greedy Policies. The optimal policy can be interpreted as a greedy policy with respect to the action-value function. The Optimal Bellman Equation (2.6) shows that the optimal action-value function is obtained by taking the maximum over all actions at each step. Thus, the greedy policy $\pi_t^*(x) = \arg\max_{a \in \mathcal{A}} Q_t^*(x, a)$ is optimal. In particular, $V_t^*(x) = \max_{a \in \mathcal{A}} Q_t^*(x, a)$. Conversely, any policy that follows the greedy action with respect to its action-value function is an optimal policy:

Proposition 2.3. Let π be a policy that follows the greedy action with respect to its action-value function, i.e., for all $x \in \mathcal{X}$ and $t < H$, $\pi_t(x) \in \arg\max_{a \in \mathcal{A}} Q_t^\pi(x, a)$. Then, π is an optimal policy.

This result can be used as a test to verify whether a given policy is optimal. If the policy does not follow the greedy action, then it cannot be optimal. If it does, then the policy is optimal.

2.2 The Distributional Perspective in MDPs

In the previous section, we presented the standard framework where the return is summarized by a single scalar: its expectation. While this representation is sufficient to derive an optimal policy maximizing the expected return, it discards higher-order information, thereby failing to capture the intrinsic stochasticity of the environment and the variability of outcomes.

In this section, we adopt the broader *distributional perspective*, which preserves this information by propagating the entire probability distribution of the return.

All the results in this section can be found in Bellemare et al. [2023].

2.2.1 Computing distributions

The idea of studying return distributions was first explored theoretically in the work of Sobel [1982] and Chung and Sobel [1987], and was then reintroduced by Morimura et al. [2010] who were the first to propose practical ways of computing them. It was later popularized by Bellemare et al. [2017], which introduced a practical and theoretically grounded algorithm (C51) for approximating the return distribution in deep RL. This line of work has since led to several breakthroughs in value-based reinforcement learning [Bellemare et al., 2017, Hessel et al., 2018, Schwarzer et al., 2023].

The first objective is to compute the return distribution, which is defined as

$$\eta_0^\pi(x_0) = \mathcal{L}\left(\mathcal{R}_0^\pi(x_0)\right) \tag{2.7}$$

where $\mathcal{L}$ denotes the distribution of a random variable, seen as a probability measure over the real numbers. In this thesis, to avoid measure theory considerations, an equality in distribution will be verified as long as the two sides have the same cumulative distribution function (CDF):

$$\eta_1 \overset{d}{=} \eta_2 \iff F_{\eta_1} = F_{\eta_2},$$

with F_η being the cumulative distribution function of η. In the following, we will write $\eta_1 = \eta_2$ instead of $\eta_1 \overset{d}{=} \eta_2$ to simplify the notation.

Random Variable Bellman Equations. A compelling argument for studying return distributions is that the recursive structure of the expected return generalizes naturally to the return *as a random variable*. Specifically, the return $\mathcal{R}_t^\pi(x)$ satisfies a stochastic recursion that mirrors the standard Bellman equation, but in the space of random variables.

Proposition 2.4 (Random Variable Bellman Equation)**.** Let π be a Markov policy. Then, for all $t < H$ and all $x \in \mathcal{X}$, the return $\mathcal{R}_t^\pi(x)$ satisfies the following identity:

$$\mathcal{R}_t^\pi(x) = r(x, A, X') + \mathcal{R}_{t+1}^\pi(X'), \tag{2.8}$$

where $A \sim \pi_t(\cdot \mid x)$ and $X' \sim p(\cdot \mid x, A)$.

This equation follows directly from the definition of the return:

$$\mathcal{R}_t^\pi(x) := \sum_{s=t}^{H-1} r(X_s, A_s, X_{s+1}), \quad X_t = x, A_s \sim \pi_s(\cdot \mid X_s), X_{s+1} \sim p(\cdot \mid X_s, A_s)$$

Importantly, this equation is not an expectation; it holds at the level of random variables.

From Random Variables to Distributions. Given the random variable Bellman equation (Proposition 2.4), we now describe how to translate equations in random variables to equations in distributions.

Proposition 2.5 (Operations on Distributions)**.** Let X be a real-valued random variable, and let $r \in \mathbb{R}$. Then:

$$\mathcal{L}(X + r) = \tau_r \mathcal{L}(X),$$

where τ_r represents a translation (pushforward) of the measure by r (i.e., $\tau_r \eta(C) = \eta(C - r)$ for any measurable set C).

Let Z be a discrete random variable on a finite set $\mathcal{Z}$ with law $p(z) = P(Z = z)$, and let $(G(z))$ be a family of independent real-valued random variables indexed by $z \in \mathcal{Z}$. Then:

$$\mathcal{L}(G(Z)) = \sum_{z \in \mathcal{Z}} p(z) \mathcal{L}(G(z)),$$

Distributional Bellman Equations. We start by defining the *Value Distribution* or *Return Distribution* as the distribution of the partial return:

Definition 2.9 (Value Distribution)**.** Let π be a Markov policy. Then, for all $t < H$ and all $x \in \mathcal{X}$, the return distributions $\eta_t^\pi(x)$ and $\eta_t^\pi(x, a)$ are defined as the distribution of the partial returns:

$$\eta_t^\pi(x) = \mathcal{L}(\mathcal{R}_t^\pi(x)), \quad \eta_t^\pi(x, a) = \mathcal{L}(\mathcal{R}_t^\pi(x, a)).$$

The value distribution $\eta_t^\pi(x)$ verifies the following recursive equation:

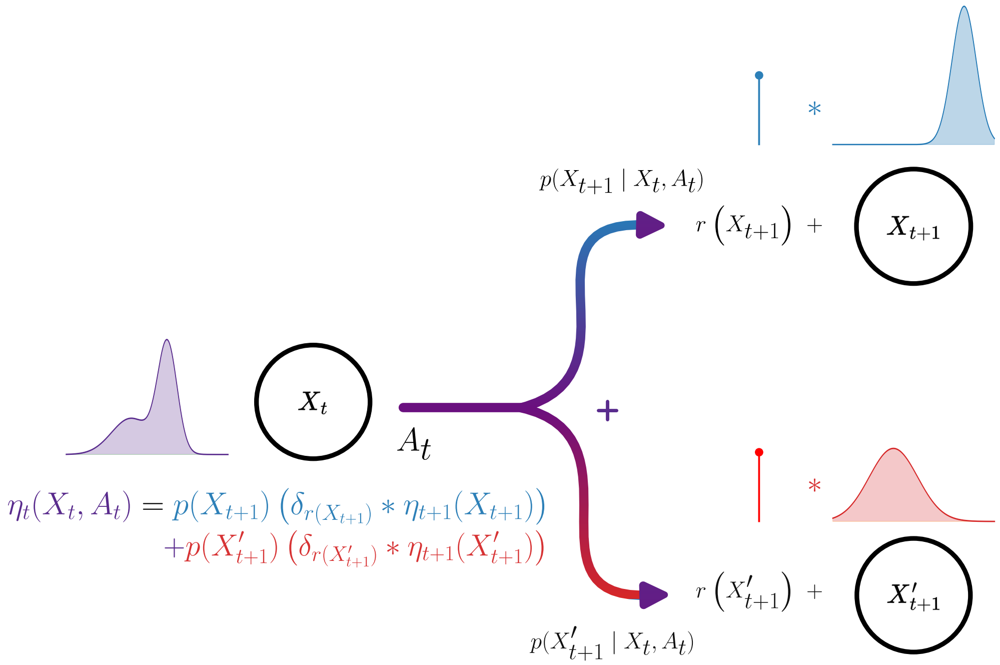
*
$p(X_{t+1} \mid X_t, A_t)$
$r\left(X_{t+1}\right) +$
X_{t+1}
X_t
A_t
+
$\eta_t(X_t, A_t) = p(X_{t+1})\left(\delta_{r(X_{t+1})} * \eta_{t+1}(X_{t+1})\right)$
$+p(X'_{t+1})\left(\delta_{r(X'_{t+1})} * \eta_{t+1}(X'_{t+1})\right)$
*
$r\left(X'_{t+1}\right) +$
X'_{t+1}
$p(X'_{t+1} \mid X_t, A_t)$

Figure 2.7: Illustration of the distributional Bellman Equation.

Proposition 2.6 (Distributional Bellman Equation)**.** Let π be a Markov policy. Then, for all $t < H$ and all $x \in \mathcal{X}$, the return distribution $\eta_t^\pi(x)$ satisfies

$$\eta_t^\pi(x) = \sum_{a \in \mathcal{A}} \pi_t(a \mid x) \sum_{x' \in \mathcal{X}} p(x' \mid x, a) \left(\tau_{r(x,a,x')} \eta_{t+1}^\pi(x')\right). \tag{2.9}$$

Remark 2.1 (Random variable vs distribution notation)**.** Depending on the source, both the random variable and the distributional formulations can be found in the literature. The random variable notation has the advantage of having simpler equations, more similar to the expected return. Conversely, the distribution notation is less intuitive, but is more explicit with the operations done in practice. In this thesis, we will favor the distribution notation as it is closest to practical implementations.

Distributional Policy Evaluation. As in the expected return setting, the full distribution of returns can be computed under a fixed policy using dynamic programming. This leads to the *Distributional Policy Evaluation Algorithm*, presented in Algorithm 3 which iteratively computes the distributions $\eta_t^\pi(x)$ for each state and timestep.

The algorithm is similar to the standard policy evaluation algorithm (Algorithm 1). The only difference is that it maintains the full distribution of the return instead of the expectation. The update step is adapted accordingly using the distributional Bellman

Algorithm 3 Distributional Policy Evaluation

```
Require: MDP $(\mathcal{X}, \mathcal{A}, p, r, H)$, Markov policy $\pi = (\pi_0, \dots, \pi_{H-1})$
Ensure: Return distributions $\eta_t^\pi(x)$ for all $x \in \mathcal{X}$ and $t \in \{0, \dots, H\}$
Initialize $\eta_H^\pi(x) \leftarrow \delta_0$ for all $x \in \mathcal{X}$        {Zero reward after episode ends}
for $t = H - 1 \to 0$ do
    for all $x \in \mathcal{X}$ do
        Initialize $\eta_t^\pi(x) \leftarrow 0$        {Zero measure}
        for all $a \in \mathcal{A}$ do
            for all $x' \in \mathcal{X}$ do
                $\eta_t^\pi(x) \leftarrow \eta_t^\pi(x) + \pi_t(a \mid x) \cdot p(x' \mid x, a) \cdot \tau_{r(x,a,x')} \eta_{t+1}^\pi(x')$
            end for
        end for
    end for
end for
output $\eta^\pi$
```

equation (2.9). The distribution $\eta_H^\pi(x)$ is initialized to the Dirac measure at 0, meaning that at the end of the episode, the partial return is 0 with probability 1. By contrast, for $t < H$, the distribution $\eta_t^\pi(x)$ is initialized to the zero measure.

On the Tractability of Distributional Evaluation. The Distributional Policy Evaluation Algorithm requires maintaining the full distribution of returns for each state and timestep. In general, computing arbitrary return distributions is intractable: the problem is known to be NP-hard in the worst case [Cooper, 1990]. Even in our finite-horizon setting with discrete state and action spaces and deterministic rewards, the space of all possible return distributions remains infinite-dimensional. The size of the support can grow exponentially with the horizon due to the combinatorial number of possible trajectories.

To mitigate this, most practical algorithms restrict themselves to a tractable class of distributions to approximate the true return distributions. These approximations are further discussed in Section 2.2.3, along with their theoretical and empirical trade-offs.

2.2.2 Policy Optimization under Return Distributions

We now consider the *Distributional Policy Optimization problem*. While policy evaluation extends naturally to the distributional setting, policy optimization is considerably more subtle: there is no canonical notion of an "optimal" distribution of returns. There is no natural total order over distributions. This raises a fundamental question: *what does it mean to optimize a distribution?*

To give this problem meaning, one must impose an ordering over distributions. The most common approach is to project each distribution onto a scalar via a *risk measure* φ (such as the expectation, or a quantile) and optimize this scalar value instead. This scalarization reduces the problem to:

$$\arg\max_{\eta^\pi} \varphi(\eta_0^\pi(x_0)), \tag{2.10}$$

where $\eta_0^\pi(x_0)$ denotes the return distribution induced by policy π starting from x_0.

The different risk measures of interest will be studied in Section 2.3, but one can take the expectation as a simple example. In this distributional setting, we aim not only to compute $\varphi(\eta^\pi)$ but also to recover the full distribution η^π corresponding to an optimal policy. Also, here there may not be a single optimal distribution, but rather a set of distributions that are optimal for the same functional φ. For instance, two different return distributions may have the same expectation but differ in variance or higher moments.

It is therefore natural to consider adapting dynamic programming techniques to this setting by adapting the Optimal Bellman Equation (2.5) with distributions.

Distributional Policy Optimization Algorithm. The canonical algorithm for distributional policy optimization proceeds similarly to the standard value iteration algorithm (Algorithm 8), but operates on return distributions. At each step, it updates the distributional value function using the distributional Bellman equation and selects actions that maximize the scalar functional φ applied to the updated distributions.

Algorithm 4 Distributional Policy Optimization

```
Require: MDP $(\mathcal{X}, \mathcal{A}, p, r, H)$, scalar functional $\varphi : \mathcal{P}(\mathbb{R}) \to \mathbb{R}$
Ensure: Markov Policy $\pi = (\pi_0, \dots, \pi_{H-1})$ and distributions $\eta_t(x)$ for all $x \in \mathcal{X}$, $t < H$
Initialize $\eta_H(x) \leftarrow \delta_0$ for all $x \in \mathcal{X}$        // zero return at final step
for $t = H - 1 \to 0$ do
    for all $x \in \mathcal{X}$ do
        for all $a \in \mathcal{A}$ do
            $\eta_t(x, a) \leftarrow \sum_{x' \in \mathcal{X}} p(x' \mid x, a) \left( \tau_{r(x,a,x')} \eta_{t+1}(x') \right)$
        end for
        $\pi_t(x) \in \arg\max_{a \in \mathcal{A}} \varphi\left(\eta_t(x, a)\right)$
        $\eta_t(x) \leftarrow \eta_t(x, \pi_t(x))$
    end for
end for
output Policy $\pi$, distributions $\eta$
```

This algorithm was first introduced by Bellemare et al. [2017] in the case where φ is the expectation. It was then extended with various risk measures [Dabney et al., 2018a].

About the correctness of the algorithm. This approach is not guaranteed to find an optimal policy for all objectives. One key limitation stems from the restriction to Markov policies. As we will explain in Section 2.3, some risk measures may require history-dependent policies to achieve optimality. Moreover, the greedy selection step may not always preserve optimality. It is only proven for the expected return [Bellemare et al., 2017] and the entropic risk measure ($X \mapsto \frac{1}{\beta} \log \mathbb{E}[e^X]$, [Liang and Luo, 2024], Section 2.3.2) that the algorithm outputs both an optimal policy and a correct return distribution under that policy. The general case is studied in Chapter 3.

The expected return. The specific case of φ being the expectation is the most studied [Bellemare et al., 2023] and the one that led to state-of-the-art RL algorithms such as C51 [Bellemare et al., 2017] and Rainbow [Hessel et al., 2018]. We begin by stating the optimal distributional Bellman equation, which provides a recursive characterization of the return distributions under an optimal policy.

Proposition 2.7 (Distributional Optimal Bellman Equation)**.** Let π^* be an optimal deterministic Markov policy for the expected return. Then, for all $t < H$ and all $x \in \mathcal{X}$, $\exists\, a^*_{t+1}(x') \in \arg\max_{a \in \mathcal{A}} E[\eta^{\pi^*}_{t+1}(x', a)]$ such that

$$\eta^{\pi^*}_t(x) = \sum_{x' \in \mathcal{X}} p(x' \mid x, \pi^*(x)) \left(\tau_{r(x,\pi^*(x),x')} \eta^{\pi^*}_{t+1}(x', a^*_{t+1}(x')) \right).$$

This equation has two main consequences. First, it implies that the optimality principle still holds with distributions. Second, it also implies that Algorithm 4 outputs an optimal distribution for the expected return.

Corollary 2.1 (Distributional Optimality Principle)**.** Let π^* be an optimal Markov policy for the expected return. Then, for all $t < H$ and all $x \in \mathcal{X}$,

$$E[\eta^{\pi^*}_t(x)] = \max_{\pi} E[\eta^{\pi}_t(x)],$$

The relevance of Distributional Policy Optimization. In this section, we introduced a method for obtaining optimal distributions. This procedure works similarly as in the non-distributional case, choosing actions greedily only on the statistical functional φ[1]. While having access to the full distribution of returns seems promising, it is not always clear what is its true benefit.

[1] This gives an intuition on why the distributional value iteration works. The iterative process of choosing actions is actually the same.

In the RL setting, it was popularized by the series of works from Bellemare et al. [2017] which showed improvements in the performance of RL algorithms (optimizing the expected return). Yet theses studies were only experimental. The theoretical work of Lyle et al. [2019] showed that it can only perform worse in the tabular and linear function approximation settings, suggesting that it is only relevant in the deep (i.e. non-linear) RL setting. The experimental study from Farebrother et al. [2024] further investigates this question, showing that it seems to rely on the difference in loss function. Some theoretical work also used distributional algorithms to improve state-of-the-art regret bounds for both the expected return [Wang et al., 2023] and the entropic risk measure [Liang and Luo, 2024]. Yet, they do not prove that such bounds cannot be reached with non-distributional algorithms. The distributional methods have also been used to improve the exploration process of the algorithms [Mavrin et al., 2019, Tang and Agrawal, 2018].

When it comes to risk-sensitive objectives, several works have used distributional algorithms to optimize various risk measures [Dabney et al., 2018a, Lim and Malik, 2022], but they either use strong assumptions or lack any theoretical guarantees. The goal of this thesis is to provide a better understanding of the theoretical properties and limitations of distributional policy optimization for risk-sensitive objectives.

2.2.3 Approximating Distributions

As discussed before, return distributions in MDPs are, in general, infinite-dimensional objects. Even in the tabular setting, the number of possible return values grows exponentially with the planning horizon H [Bellemare et al., 2023].

Therefore, representing and manipulating these distributions exactly can become intractable, and approximate representations should be employed. A wide variety of parametric families have been proposed for this purpose, such as Gaussian and Laplace distributions [Morimura et al., 2012], Gaussian mixtures [Choi et al., 2019], and kernel density estimators [Nguyen-Tang et al., 2021]. However, in practice, two classes of approximations have emerged as particularly effective in reinforcement learning: the *categorical representation* [Bellemare et al., 2017] and the *quantile representation* [Dabney et al., 2018b].

The generic Dynamic Programming algorithm is presented in Algorithm 5. Basically, after each dynamic programming update, the resulting distribution is projected onto the chosen family of distributions using the projection operator Π.

Categorical Representation. The categorical representation approximates a distribution by a finite weighted sum of Dirac masses (atoms) located at fixed support points.

Algorithm 5 Distributional Dynamic Programming with Projection

```
Require: MDP $(\mathcal{X}, \mathcal{A}, p, r, H)$
Ensure: Return distributions $\eta_t(x)$ for all $x \in \mathcal{X}$ and $t \in \{0, \dots, H\}$. Projection operator $\Pi$
Initialize $\eta_H(x) \leftarrow \delta_0$ for all $x \in \mathcal{X}$        {Zero reward after episode ends}
for $t = H - 1 \to 0$ do
    for all $x \in \mathcal{X}$ do
        Initialize $\eta_t(x) \leftarrow 0$        {Zero measure}
        for all $a \in \mathcal{A}$ do
            for all $x' \in \mathcal{X}$ do
                Update $\eta_t(x)$ as a function of $\eta_{t+1}(x')$ using the Bellman equation of choice
                $\eta_t(x) \leftarrow \Pi(\eta_t(x))$        {Project onto the chosen family of distributions}
            end for
        end for
    end for
end for
output $\eta$
```

Formally, for a given number of atoms $n + 1$ and fixed support $\{z_0, z_1, \dots, z_n\}$ with

$$z_i = z_{\text{min}} + i \cdot \Delta, \quad \Delta = \frac{z_{\text{max}} - z_{\text{min}}}{n},$$

the space of approximated distributions is:

$$\mathcal{P}_{\text{cat}} = \left\{ \sum_{i=0}^{n} p_i \delta_{z_i} \,\middle|\, p_i \geq 0, \ \sum_{i=0}^{n} p_i = 1 \right\},$$

where δ_{z_i} is the Dirac measure at z_i. The hyperparameters of the representation are the number of atoms n and the support range $[z_{\text{min}}, z_{\text{max}}]$.

To project a general distribution η onto this set, Bellemare et al. [2017] minimize the *Cramér-2 metric* ℓ_2 between the original and the projected distribution [Dabney et al., 2018b]:

$$\Pi_{\text{cat}}\eta = \arg\min_{\bar{\eta} \in \mathcal{P}_{\text{cat}}} \ell_2(\bar{\eta}, \eta).$$

where $\Pi_{\text{cat}} : \mathcal{P}(\mathbb{R}) \to \mathcal{P}_{\text{cat}}$ is the projection operator and

$$\ell_2(\bar{\eta}, \eta) = \sqrt{\int_{\mathbb{R}} \left(F_{\bar{\eta}}(z) - F_{\eta}(z)\right)^2 dz}, \tag{2.11}$$

with $F_{\bar{\eta}}$ and F_{η} being the cumulative distribution functions of $\bar{\eta}$ and η, respectively.

This projection can be written in closed form using the following property.

Proposition 2.8 (Categorical Projection [Bellemare et al., 2017])**.** The operator Π_{cat} is linear on distributions. Furthermore, its value on a Dirac measure δ_x is given by:

$$\Pi_{\text{cat}}(\delta_x) = \begin{cases} \delta_{z_0} & x \leq z_0 \\ \frac{z_{i+1}-x}{z_{i+1}-z_i}\delta_{z_i} + \frac{x-z_i}{z_{i+1}-z_i}\delta_{z_{i+1}} & z_i < x < z_{i+1} \\ \delta_{z_n} & x \geq z_n \end{cases} \tag{2.12}$$

Hence, for a discrete distribution $\eta = \sum_{i=0}^{n} p_i \delta_{x_i}$, the projection $\Pi_{\text{cat}}\eta$ can be computed using

$$\Pi_{\text{cat}}\eta = \sum_{i=0}^{n} p_i \ \Pi_{\text{cat}}(\delta_{x_i}). \tag{2.13}$$

and the formula in Proposition 2.8.

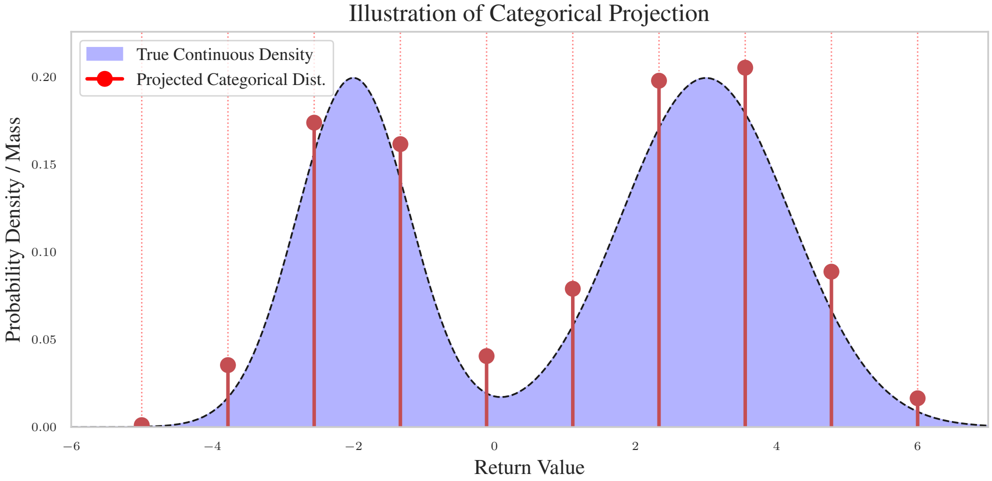

Figure 2.8: Illustration of the categorical projection operator. The probability mass from the continuous distribution that falls between two atoms is split proportionally between its nearest neighbors, effectively discretizing the distribution onto the fixed grid while preserving the mean.

This projection has the desirable property of preserving the expectation of the original distribution, making it particularly suitable for expected return optimization.

Proposition 2.9 ([Bellemare et al., 2017])**.** Let η be a distribution with support in $[z_{\min}, z_{\max}]$ and finite expectation $\mathbb{E}_\eta[X]$. Then,

$$\mathbb{E}_{\Pi_{\text{cat}}\eta}[X] = \mathbb{E}_\eta[X].$$

This property ensures that algorithm 5 outputs a distribution with optimal expected return (in the case φ is the expectation), even if the distribution is not the true return distribution but only an approximation.

Another important property is the following bound on the distance between the original and projected distributions:

Proposition 2.10 ([Rowland et al., 2018])**.** Let $\eta \in \mathcal{P}(\mathbb{R})$ be a distribution with finite support in $[z_{\min}, z_{\max}]$. Then,

$$\ell_2(\Pi_{\text{cat}}\eta, \eta) \leq \frac{z_{\max} - z_{\min}}{n}.$$

This bound shows that the approximation error decreases with the number of atoms n, which is a desirable property for practical applications.

Nonetheless, a key limitation of this representation is its fixed support: the distribution is constrained to lie within $[z_{\min}, z_{\max}]$. In our MDP setting, the support of the return is linear in H, the horizon. To obtain a better distributional approximation with a fixed number of atoms n, the support of the parametrization must be changed for each timestep t.

This representation underpins the C51 algorithm [Bellemare et al., 2017], as well as several deep RL extensions such as Rainbow [Hessel et al., 2018].

Quantile Representation. In contrast, the quantile representation uses fixed weights and variable support locations. It approximates a distribution as:

$$\mathcal{P}_{\text{qr}} = \left\{ \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i} \;\middle|\; x_i \in \mathbb{R} \right\},$$

where each atom x_i is a learnable parameter representing a quantile location and n is the hyperparameter representing the number of quantiles.

While there can be several ways to project distributions onto this representation space, a natural one is to use the minimization of the *Wasserstein-1 distance* W_1 (also equal to the Cramér-1 ℓ_1 distance) between the target distribution and its approximation [Dabney et al., 2018b].

$$\Pi_{\text{qr}}\eta = \arg\min_{\bar{\eta} \in \mathcal{P}_{\text{qr}}} W_1(\bar{\eta}, \eta).$$

where $\Pi_{\text{qr}} : \mathcal{P}(\mathbb{R}) \to \mathcal{P}_{\text{qr}}$ is the projection operator and

$$W_1(\bar{\eta}, \eta) = \int_{\mathbb{R}} |F_{\bar{\eta}}(z) - F_{\eta}(z)| \, dz. \tag{2.14}$$

This projection also has a more explicit form using the quantiles of the distribution (hence its name):

Proposition 2.11 (Quantile Projection [Dabney et al., 2018b])**.** Let η be a distribution with cumulative distribution function F_η. Let $\bar{\Theta} = \arg\min_{\bar{\eta} \in \mathcal{P}_{\text{qr}}} W_1(\bar{\eta}, \eta)$. Then,

$$\bar{\Theta} = \left\{ \bar{\eta} = \frac{1}{n} \sum_{i=1}^{n} \delta_{x_i} \;\middle|\; \forall i, \; x_i \in F_\eta^{-1}\left(\frac{2i-1}{2n}\right) \right\}$$

This representation allows for unbounded support and greater flexibility in representing sharp or asymmetric distributions. Similarly to the categorical case, it is also possible to bound the approximation error:

Proposition 2.12 ([Dabney et al., 2018b]). Let $\eta \in \mathcal{P}(\mathbb{R})$ be a discrete distribution with support on $[z_{\text{min}}, z_{\text{max}}]$. Then,

$$W_1(\Pi_{\text{qr}}\eta, \eta) \leq \frac{z_{\text{max}} - z_{\text{min}}}{2n}.$$

This also ensures that the approximation error decreases with the number of atoms n.

This distribution representation has also been used successfully, such as in the work of Fawzi et al. [2022].

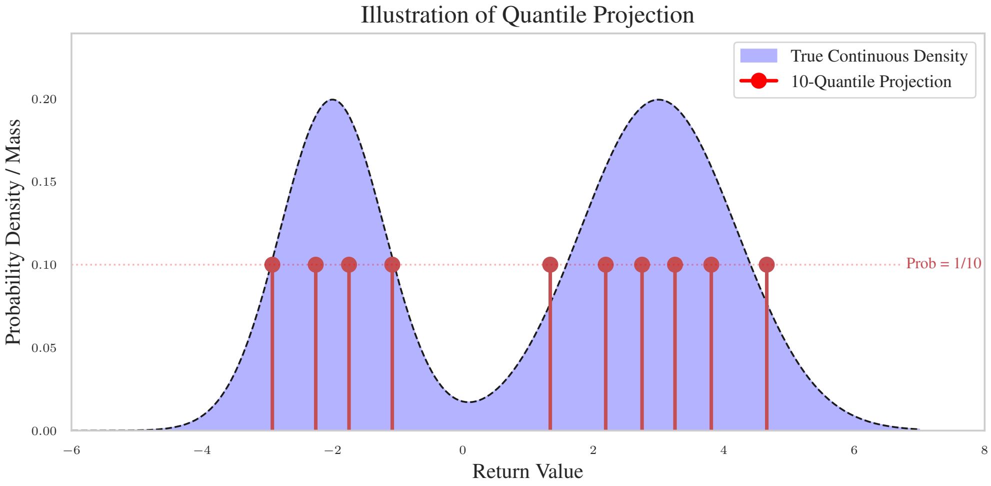

Figure 2.9: Illustration of the quantile projection operator. Visually, the atoms cluster densely in regions where the true probability density is high, and are sparse in the tails, thereby dividing the total probability mass into n equal portions.

Comparison and Trade-offs. The categorical and quantile representations form a natural duality: the former uses fixed support and variable weights, while the latter uses fixed weights and variable support. The categorical method preserves the expectation and provides better control over boundedness, while the quantile method better captures distributional geometry under the Wasserstein metric and adapts more easily to skewed, heavy-tailed or highly concentrated distributions.

In practice, the choice between them may depend on the optimization objective, computational constraints, and the expected shape of the return distribution.

2.3 Risk Measures in MDPs

The standard objective in Markov Decision Processes is to maximize the expected return. This objective enjoys powerful theoretical and computational properties: the Bellman equations hold, the optimality principle applies, and there always exists a Markovian optimal policy. Moreover, dynamic programming algorithms can be used to efficiently compute the optimal policy. However, the expected return may fail to capture the true preferences of a decision-maker in many situations.

Consider, for instance, a physician trying to minimize the chances of losing a patient rather than maximizing the expected life expectancy, or a farmer who wishes to maximize the probability of a successful harvest rather than the expected yield. In both cases, the expected return fails to capture what truly matters to the decision-maker. These considerations motivate the study of more general *risk-sensitive objectives*, which aim to encode preferences beyond the mean.

2.3.1 Risk-Sensitive Objectives in MDPs

As we previously illustrated in the Cliff example (see Figure 2.3), there may exist several plausible policies depending on the user's attitude towards risk, ranging from conservative ones to more risky ones. Each may be considered optimal with respect to a different performance criterion. Rather than optimizing the expectation, we may instead optimize statistics like variance, quantiles, tail probabilities, or other distributional features.

Risk Measures. A *risk measure* is a functional φ that maps a random variable to a real value [Föllmer and Schied, 2011]. In this thesis, we consider the specific class of *static* risk measures, which depend only on the distribution of the random variable. Formally, let Φ denote the set of all static risk measures $\varphi : \mathcal{P}(\mathbb{R}) \to \mathbb{R} \cup \{\pm\infty\}$. For any real random variable X, we let $\eta_X = \mathcal{L}(X)$, and denote $\varphi(X) = \varphi(\eta_X)$ by slight abuse of notation.

The optimization problem considered in this section is thus to find a policy π^* that maximizes the risk measure of the return:

$$\max_{\pi \in \Pi} \varphi(\eta_\pi) = \max_{\pi \in \Pi} \varphi(\mathcal{R}^\pi).$$

Numerous risk measures have been studied in the literature, each capturing different notions of risk. We here present a non-exhaustive list of commonly used risk-sensitive criteria:

- **Entropic Risk Measure:**

  $$\varphi(X) = \frac{1}{\beta} \log \mathbb{E}[\exp(\beta X)]$$

  This criterion penalizes variance in a smooth, parameterized way, depending on the sign and magnitude of β. It was first introduced in MDPs by Howard and Matheson [1972].

- **Value-at-Risk (VaR$_\alpha$):**

  $$\varphi(X) = \inf \{\tau \in \mathbb{R} \mid \Pr(X \leq \tau) > \alpha\}$$

  Measures the worst-case outcome not exceeded with probability α. It corresponds to the α-quantile of the return distribution [Li et al., 2022].

- **Conditional Value-at-Risk (CVaR$_\alpha$):**

  $$\varphi(X) = \mathbb{E}[X \mid X \leq \mathrm{VaR}_\alpha(X)]$$

  Also known as Expected Shortfall or Average Value-at-Risk (AVaR), it is the average of the worst α% of outcomes [Chow et al., 2015].

- **Threshold probability objective:**

  $$\varphi(X) = -\Pr(X \leq \tau)$$

  Maximizes the probability of exceeding a fixed reward threshold τ [White, 1993]; it is also known as the *target level criterion* [Bouakiz and Kebir, 1995]. The threshold probability objective can be seen as a dual to VaR, as minimizing the probability of exceeding a threshold is equivalent to maximizing the VaR at a certain level.

A Fundamental Difficulty. Changing the optimization criterion comes at a cost: the classical theory of MDPs no longer applies. In particular, the Bellman equations and the optimality principle may fail, and the set of Markov policies may no longer be sufficient. Optimal policies may need to depend on the full history.

To illustrate this, we introduce two concrete examples: one based on the power utility $\varphi(X) = \mathbb{E}[X^3]$, where the optimal policy is non-Markov, and one based on quantile criterion (VaR) with $\alpha = 0.25$, where the optimality principle fails.

Expected Utilities optimal policies may not be Markov For this example, we consider the functional $\varphi(X) = \mathbb{E}[X^3]$. Consider the MDP illustrated on Figure 2.10.

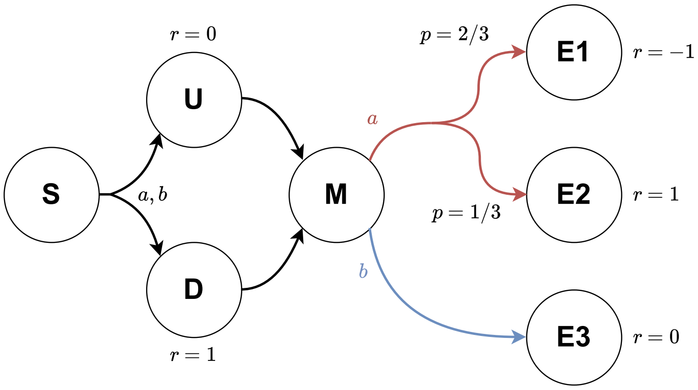

Figure 2.10: Counter-example for the criterion $\varphi(X) = \mathbb{E}[X^3]$. In this MDP, the optimal policy is non-Markov. The best policy in state `M` is to choose action `a` if the agent went through `D` before, or `b` if the agent went through `U` instead. The reason is because $\mathbb{E}\left[(\mathcal{R}_2(M,a))^3\right] = -\frac{1}{3} < \mathbb{E}\left[(\mathcal{R}_2(M,b))^3\right] = 0$, but $\mathbb{E}\left[(\mathcal{R}_2(M,a)+1)^3\right] = \frac{8}{3} > \mathbb{E}\left[(\mathcal{R}_2(M,b)+1)^3\right] = 1$.

The agent first goes stochastically to either `U` or `D` and then to `M` independently of the policy. They receive a reward of either 0 or 1 along the way. Then the choice of the policy matters. $\mathcal{R}_2(M,a)$ and $\mathcal{R}_2(M,b)$ are the random partial returns obtained after reaching state `M` and taking `a` and `b` respectively. We observe that $\eta_2(M,a) := \mathcal{L}(\mathcal{R}_2(M,a)) = \frac{1}{3}\delta_1 + \frac{2}{3}\delta_{-1}$ and $\eta_2(M,b) := \mathcal{L}(\mathcal{R}_2(M,b)) = \delta_0$. For *a Markov policy,*

$$\eta^\pi = \left(\frac{1}{2}\delta_0 + \frac{1}{2}\delta_1\right) * \left(\pi(a \mid M)\eta_2(M,a) + \pi(b \mid M)\eta_2(M,b)\right).$$

Where $*$ is the convolution operation. The objective is $\max_{\pi\in\Pi}\mathbb{E}[(\mathcal{R}^\pi)^3]$, that is we want to optimize the third moment of the return. The computation yields:

- **Policy favoring action `a`**: $\mathbb{E}[(\mathcal{R}^{\pi_a})^3] = \frac{7}{6}$
- **Policy favoring action `b`**: $\mathbb{E}[(\mathcal{R}^{\pi_b})^3] = \frac{1}{2}$

Now consider the history-dependent policy π_h satisfying $\pi(a|M, r_0 = 1) = 1$ and $\pi(b \mid M, r_0 = 0) = 1$. This policy depends explicitly on the reward observed at time $t = 0$, and thus is not Markov. If the reward is 0, then the policy will choose action `b` when in state `M`. Conversely, if the reward is 1, then the policy will choose action `a`. Under this policy, the return distribution becomes

$$\eta^{\pi_h} = \frac{1}{2}\left(\delta_1 * \eta_2(M,a)\right) + \frac{1}{2}\left(\delta_0 * \eta_2(M,b)\right)$$

and the objective value is

- **History-dependent policy** π_h: $\mathbb{E}[(\mathcal{R}^{\pi_h})^3] = \frac{8}{6}$

This value is higher than the best achievable value by any Markov policy. Hence, the optimal policy for this MDP and objective is not Markov. The main phenomenon at play here is that $\mathbb{E}\left[(\mathcal{R}_2(M,a))^3\right] = -\frac{1}{3} < \mathbb{E}\left[(\mathcal{R}_2(M,b))^3\right] = 0$, but $\mathbb{E}\left[(\mathcal{R}_2(M,a)+1)^3\right] = \frac{8}{3} > \mathbb{E}\left[(\mathcal{R}_2(M,b)+1)^3\right] = 1$. In essence, adding a fixed reward changes the preference. This behavior will be studied in more depth in Chapter 3.

Value at Risk does not satisfy the optimality principle For this example, we consider the functional $\varphi(X) = \mathrm{VaR}_{0.25}(X)$, the lower quartile of the distribution. Consider the MDP illustrated in Figure 2.11.

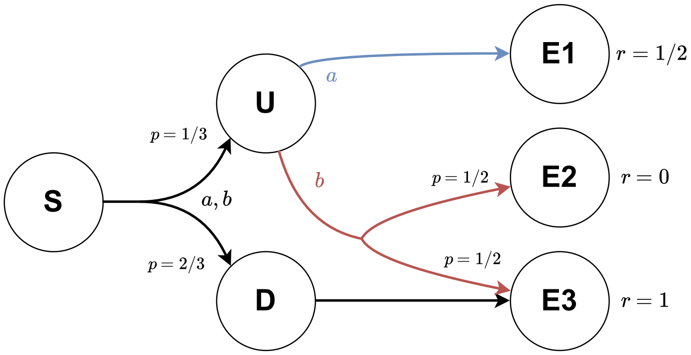

Figure 2.11: Counter-example for Value-at-Risk with quantile level $\alpha = 0.25$ ($\mathrm{VaR}_{0.25}$). Here, the optimality principle does not hold. An agent starting from `U` would prefer action `a`, but when starting from `S` they prefers action `b` when reaching `U` instead. This is because $\mathrm{VaR}_{0.25}[\eta^\pi(U,a)] = 0.5 > \mathrm{VaR}_{0.25}[\eta^\pi(U,b)] = 0$, but $\mathrm{VaR}_{0.25}[\frac{1}{3}\eta^\pi(U,a) + \frac{2}{3}\eta^\pi(D)] = 0.5 < \mathrm{VaR}_{0.25}[\frac{1}{3}\eta^\pi(U,b) + \frac{2}{3}\eta^\pi(D)] = 1$.

The agent starts at state `S` and then stochastically goes to either `U`, with probability 1/3, or `D`, with probability 2/3, independently of the action. They receive no reward at this point. If they reach `D`, then any action goes to `E3` and receives a reward of 1. At state `U`, the agent chooses between actions `a` and `b`. Action `a` leads deterministically to `E1` with a reward of 0.5. Action `b` goes to `E3` with probability 0.5, receiving a reward of 1 and goes to `E2` also with probability 0.5 but receiving no reward. The associated return distributions are $\eta^\pi(U,b) = \frac{1}{2}\delta_0 + \frac{1}{2}\delta_1$, and $\eta^\pi(U,a) = \delta_{1/2}$. At state `U`, the distribution of return is

$$\eta^\pi(U) = \pi(a|U)\eta^\pi(U,a) + \pi(b|U)\eta^\pi(U,b)\ .$$

Computing the Value at Risk at level 0.25 of both actions, we have

- **Start: U, Action `a`**: $\mathrm{VaR}_{0.25}[\eta^\pi(U,a)] = 0.5$
- **Start: U, Action `b`**: $\mathrm{VaR}_{0.25}[\eta^\pi(U,b)] = 0$

thus the best action here is `a`. Now consider the full random return. It is given by

$$\eta^\pi(S) = \frac{1}{3}\eta^\pi(U) + \frac{2}{3}\delta_1 \ .$$

We observe that

- **Start: S, Action `a`**: $\mathrm{VaR}_{0.25}[\eta^\pi(S,a)] = 0.5$
- **Start: S, Action `b`**: $\mathrm{VaR}_{0.25}[\eta^\pi(S,b)] = 1$

Hence, the optimal policy at state `S` is to choose action `b` at state `U`. This demonstrates a violation of the Bellman optimality principle Corollary 2.1, which states that the action optimal in a subproblem should remain optimal when part of a larger problem. The optimal action at state `U` is `a` when the agent starts from this same state `U`, but is `b` when the agent starts from `S` instead. In this counter-example, the preference changes when mixing the distributions with a third fixed distribution. This phenomenon will also be studied in more depth in Chapter 3.

The two phenomena observed in these examples highlight two important properties needed for the risk measure to satisfy the Bellman optimality principle and allow for the existence of a Dynamic Programming algorithm. We will prove in Chapter 3 that both are satisfied simultaneously by only a specific class of risk measures, the *entropic risk measures.*

Overview of the Section. The remainder of this section focuses on three major families of risk measures:

- **Entropic Risk Measures**, based on exponential utilities, with strong properties and efficient algorithms.
- **Expected Utilities**, a broader class of risk measures, but with weaker theoretical guarantees and higher complexity algorithms.
- **Quantile-based Measures**, focusing on VaR and CVaR which are quite popular but pose challenges for optimization.

Each of these classes comes with its own theoretical properties and limitations. Our goal is to understand how to evaluate and optimize them in the context of finite-horizon MDPs.

2.3.2 The Entropic Risk Measure

Entropic Risk Measures (EntRM) are a widely studied and analytically tractable risk measure family [Föllmer and Schied, 2011]. Their risk parameter $\beta \in \mathbb{R}$ captures a trade-off between the *risk* of the distribution (i.e., the lower tail of the distribution), and its *gain* (i.e., the upper tail of the distribution). Importantly, this risk measure can be computed efficiently in MDPs through dynamic programming, which makes it a good choice to use for risk-sensitive planning [Howard and Matheson, 1972]. Overall, it offers a practical generalization of the expectation enabling risk-sensitive objectives.

2.3.2.1 Definition and Properties

Definition 2.10 (Entropic Risk Measure)**.** Let X be a real-valued random variable. The *Entropic Risk Measure* of X with risk sensitivity parameter $\beta \in \mathbb{R}$ is defined as:

$$\mathrm{EntRM}_\beta[X] = \begin{cases} \frac{1}{\beta} \log \mathbb{E}\left[e^{\beta X}\right] & \text{if } \beta \neq 0, \\ \mathbb{E}[X] & \text{if } \beta = 0. \end{cases} \tag{2.15}$$

The entropic risk measure can be seen as a distorted mean, with exponential weights on the rewards. It is finite whenever the moment generating function $\mathbb{E}[e^{\beta X}]$ is. It includes the expectation (recovered as the limit case $\beta \to 0$), and provides a smooth interpolation between risk-seeking and risk-averse behavior:

- For $\beta > 0$, the EntRM emphasizes the upper tail of the distribution, favoring high-reward outcomes: this corresponds to a *risk-seeking* attitude.
- For $\beta < 0$, it emphasizes the lower tail, penalizing adverse outcomes and reflecting *risk aversion.*

This behavior can be illustrated through the simple example of Gaussian random variables with equal means but different variances.

Example 2.1 (EntRM for Gaussian Variables with Equal Means)**.** Let $X_1 \sim \mathcal{N}(\mu, \sigma_1^2)$ and $X_2 \sim \mathcal{N}(\mu, \sigma_2^2)$ with $\sigma_1 > \sigma_2$. Both random variables have the same mean μ, but X_1 has higher variance. Their entropic risk measures are given by:

$$\mathrm{EntRM}_\beta[X_i] = \mu + \frac{\beta \sigma_i^2}{2}, \quad i = 1, 2.$$

This implies:

- When $\beta = 0$, $\mathrm{EntRM}_\beta[X_1] = \mathrm{EntRM}_\beta[X_2] = \mu$: the EntRM recovers the expectation and does not distinguish the distributions.

- When $\beta > 0$, $\text{EntRM}_\beta[X_1] > \text{EntRM}_\beta[X_2]$: the EntRM favors high variance: this corresponds to a **risk-seeking** behavior.
- When $\beta < 0$, $\text{EntRM}_\beta[X_1] < \text{EntRM}_\beta[X_2]$: the EntRM penalizes variance, preferring X_2 with lower spread: this illustrates **risk aversion**.

Limit Behavior. For β values close to 0, the EntRM behaves similarly to the expectation. For very large β, the EntRM can be viewed as a smoothed approximation of the maximum (when $\beta \to +\infty$) or the minimum (when $\beta \to -\infty$) of the support of X. More precisely, the following limits hold:

- As $\beta \to 0$, a Taylor expansion yields:

  $$\text{EntRM}_\beta[X] = \mathbb{E}[X] + \frac{\beta}{2}\text{Var}[X] + o(\beta),$$

  so EntRM recovers the expectation for $\beta = 0$. For small values of β, we understand the role of the risk parameter β, rewarding a high variance for $\beta > 0$ and penalizing it for $\beta < 0$.
- As $\beta \to +\infty$, the EntRM converges to the essential supremum of X (assuming bounded support):

  $$\text{EntRM}_\beta[X] \to_{\beta \to +\infty} \operatorname{ess\,sup} X.$$

  Conversely, as $\beta \to -\infty$, it converges to the essential infimum:

  $$\text{EntRM}_\beta[X] \to_{\beta \to -\infty} \operatorname{ess\,inf} X.$$

These limits are well illustrated by the simple examples of Bernoulli random variables.

Example 2.2 (EntRM of a Bernoulli Random Variable)**.** Let $X \sim \text{Bern}(p)$, i.e., $X = 1$ with probability $p \in (0, 1)$ and $X = 0$ with probability $1 - p$. Then, for $\beta \neq 0$,

$$\text{EntRM}_\beta[X] = \frac{1}{\beta} \log\left((1 - p) + pe^\beta\right). \tag{2.16}$$

This expression illustrates the behavior of EntRM at the extremes:

- As $\beta \to 0$, we recover the expectation: $\text{EntRM}_\beta[X] \to p$.
- As $\beta \to +\infty$, the dominant term becomes pe^β, and $\text{EntRM}_\beta[X] \to 1$. We recover the **essential supremum**.
- Similarly, as $\beta \to -\infty$, the dominant term is $(1 - p)$, and $\text{EntRM}_\beta[X] \to 0$. We recover the **essential infimum**.

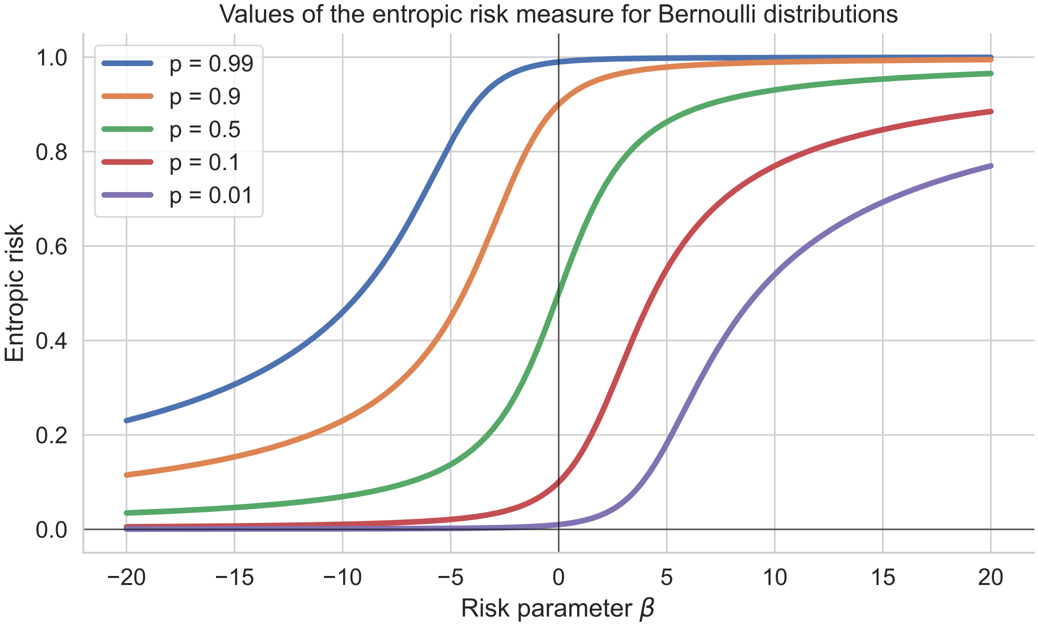

Figure 2.12: Entropic Risk Measure of $X \sim \mathrm{Bern}(p)$ as a function of β, for different values of p. Note the interpolation between 0 and 1 as β varies.

Properties. The entropic risk measure satisfies two main important properties:

- **additivity**: Let X, Y be independent random variables. Then,

$$\mathrm{EntRM}_\beta[X + Y] = \mathrm{EntRM}_\beta[X] + \mathrm{EntRM}_\beta[Y]. \tag{2.17}$$

  As a special case, for any constant $c \in \mathbb{R}$, $\mathrm{EntRM}_\beta[X + c] = \mathrm{EntRM}_\beta[X] + c$.

- **Tower Property**: For any random variables X, Y,

$$\mathrm{EntRM}_\beta[X] = \mathrm{EntRM}_\beta[\mathrm{EntRM}_\beta[X \mid Y]], \tag{2.18}$$

  where $\mathrm{EntRM}_\beta[X \mid Y] = \frac{1}{\beta} \log \mathbb{E}[e^{\beta X} \mid Y]$.

These two properties are shared with expectation and are used to prove the Bellman equation. Also benefitting from those two properties, the entropic risk measure verifies a similar equation. Before turning to this, we discuss another way to write the EntRM objective that simplifies some computations.

The Exponential Utility. Consider any $\beta \neq 0$. The function $x \mapsto \operatorname{sign}(\beta) e^{\beta x}$ is increasing. Therefore, composing the EntRM with this function does not change the optimization problem: for any two random variables X and Y, we have

$$\mathrm{EntRM}_\beta[X] \leq \mathrm{EntRM}_\beta[Y] \Leftrightarrow \operatorname{sign}(\beta)\mathbb{E}[e^{\beta X}] \leq \operatorname{sign}(\beta)\mathbb{E}[e^{\beta Y}]. \tag{2.19}$$

This risk measure is called the *exponential utility*. It is often easier to manipulate, as it does not involve a logarithm. This logarithm is however important to rescale the exponential entropy and obtain the additivity and limits mentioned above.

Remark 2.2 (Sign Convention)**.** Most of the literature on risk measures focuses only on the risk-averse case, i.e., $\beta < 0$, as the goal is usually to reduce the probability of lowest outcomes and minimize the variance. Because of this, many authors define the EntRM with a negative sign, i.e., $\mathrm{EntRM}_\beta[X] = -\frac{1}{\beta}\log \mathbb{E}[e^{-\beta X}]$ to have a positive risk parameter β instead (see [Hau et al., 2023b] for instance). This definition is equivalent to ours by replacing β with $-\beta$. As we consider here both the risk-averse and risk-seeking cases, we prefer to use the definition of Definition 2.10 to avoid overloading the notation with negative signs.

2.3.2.2 Dynamic Programming for the Entropic Risk Measure

We here consider the EntRM in the context of MDPs. We show that the EntRM satisfies similar properties as the expectation. Mainly it satisfies Bellman equations, the optimality principle, and can be optimized through dynamic programming. In this section, all policies will be Markov.

First, we introduce the EntRM value function. Like in the expected return case, we define the value function as the EntRM of partial returns. We aim at giving a recursive formula to compute this value function, and deriving a dynamic programming algorithm to evaluate and optimize policies.

Entropic Risk of a Policy. Let π be a Markov policy. The EntRM (action-)value function and the optimal EntRM (action-)value function are defined as:

$$V^\pi_{t,\beta}(x) := \mathrm{EntRM}_\beta[\mathcal{R}^\pi_t(x)], \qquad Q^\pi_{t,\beta}(x,a) := \mathrm{EntRM}_\beta[\mathcal{R}^\pi_t(x,a)], \tag{2.20}$$

$$V^*_{t,\beta}(x) := \sup_{\pi\in\Pi_M} V^\pi_{t,\beta}(x), \qquad Q^*_{t,\beta}(x,a) := \sup_{\pi\in\Pi_M} Q^\pi_{t,\beta}(x,a). \tag{2.21}$$

The objective we now consider is the optimization of the entropic risk measure of the return, i.e.,

$$\pi^*_\beta \in \arg\max_{\pi\in\Pi_M} \mathrm{EntRM}_\beta[\mathcal{R}^\pi]. \tag{2.22}$$

EntRM Bellman Equation. The EntRM (action-)value function can indeed be computed recursively using the following Bellman equation:

Proposition 2.13 (EntRM Bellman Equation)**.** Let π be a Markov policy. Then the entropic (action-)value function satisfies:

$$\begin{aligned} V^{\pi}_{H,\beta}(x) &= 0, \qquad & V^{\pi}_{t,\beta}(x) &= \mathrm{EntRM}_{\beta}[r(x, A, X') + V^{\pi}_{t+1,\beta}(X')]. \\ Q^{\pi}_{H,\beta}(x,a) &= 0, \qquad & Q^{\pi}_{t,\beta}(x,a) &= \mathrm{EntRM}_{\beta}[r(x, a, X') + Q^{\pi}_{t+1,\beta}(X', A')]. \end{aligned}$$

Similarly, the optimal (action-)value function satisfies

$$\begin{aligned} V^{*}_{H,\beta}(x) &= 0, \qquad & V^{*}_{t,\beta}(x) &= \max_{a\in\mathcal{A}} \mathrm{EntRM}_{\beta}[r(x, a, X') + V^{*}_{t+1,\beta}(X')], \\ Q^{*}_{H,\beta}(x,a) &= 0, \qquad & Q^{*}_{t,\beta}(x,a) &= \mathrm{EntRM}_{\beta}[r(x, a, X') + \max_{a'\in\mathcal{A}} Q^{*}_{t+1,\beta}(X', a')]. \end{aligned}$$

While involving the EntRM, these Bellman equations have a similar structure as in the expected return setting. Their proof follows the same lines, relying on the tower property and additivity of the EntRM. Because of their similarity, the consequences are also similar: the optimal policy is deterministic and the optimality principle holds.

Proposition 2.14 (EntRM Optimality Principle)**.**

1. There exists a deterministic Markov policy $\pi^*_\beta \in \Pi_{MD}$ that is optimal for the EntRM objective (2.22), i.e.,

$$\pi^*_\beta \in \arg\max_{\pi\in\Pi_M} \mathrm{EntRM}_{\beta}[\mathcal{R}^{\pi}].$$

2. The optimality principle holds: let π^*_β be optimal, Then, for all $x \in \mathcal{X}$, and $t < H$, if $\Pr(X_t = x) > 0$, then

$$V^{\pi^*_\beta}_{t,\beta}(x) = V^{*}_{t,\beta}(x), \qquad Q^{\pi^*_\beta}_{t,\beta}(x,a) = Q^{*}_{t,\beta}(x,a), \quad \forall a \in \mathcal{A}.$$

Dynamic Programming Algorithm. The Bellman equations allow to derive dynamic programming algorithms to compute both the value function of the policy and the optimal EntRM value function. The algorithms are similar to the expected return case, using the adapted Bellman equations. We provide the pseudo-code in Algorithm 6 and Algorithm 7. To simplify the notation, the policy is assumed to be deterministic in the case of Policy Evaluation, but the algorithm works similarly for stochastic policies. The complexity of those algorithms is $O(H|\mathcal{X}|^2|\mathcal{A}|)$, similar to the expected return case.

Algorithm 6 EntRM Policy Evaluation

```
Require: MDP $(\mathcal{X}, \mathcal{A}, P, r, H)$, risk parameter $\beta \in \mathbb{R}$, policy $\pi \in \Pi_{MD}$.
Ensure: EntRM value function $V_t^\pi(x)$ for all $x \in \mathcal{X}$ $t \in [H]$.
Initialize $V_H^\pi(x) \leftarrow 0$ for all $x \in \mathcal{X}$
for $t = H - 1$ to $0$ do
    for all $x \in \mathcal{X}$ do
        $V_t^\pi(x) \leftarrow \frac{1}{\beta} \log \left( \sum_{x' \in \mathcal{X}} p\left(x' \mid x, \pi_t(x)\right) e^{\beta[r(x,\pi_t(x),x') + V_{t+1}^\pi(x')]} \right)$
    end for
end for
output $V^\pi$
```

Algorithm 7 EntRM Value Iteration

```
Require: MDP $(\mathcal{X}, \mathcal{A}, P, r, H)$, risk parameter $\beta \in \mathbb{R}$
Ensure: $Q_t^*(x, a)$ for all $x \in \mathcal{X}$, $a \in \mathcal{A}$, $t \in [H]$, $\pi_\beta^*$.
Initialize $Q_H^*(x, a) \leftarrow 0$ for all $x \in \mathcal{X}$ and $a \in \mathcal{A}$.
for $t = H - 1$ to $0$ do
    for all $x \in \mathcal{X}$ do
        for all $a \in \mathcal{A}$ do
            $Q_t^*(x, a) \leftarrow \frac{1}{\beta} \log \left( \sum_{x' \in \mathcal{X}} p\left(x' \mid x, a\right) e^{\beta[r(x,a,x') + \max_{a' \in \mathcal{A}} Q_{t+1}^*(x', a')]} \right)$
            $\pi_t^*(x) \leftarrow \arg\max_{a \in \mathcal{A}} Q_t^*(x, a)$
        end for
    end for
end for
output $Q^*$ and $\pi_\beta^*$.
```

Distributional Dynamic Programming The optimization of EntRM by distributional dynamic programming was first studied by Liang and Luo [2024]. As the natural extension of the expected return in MDPs, it has been shown to be optimized by Distributional Policy Optimization (Algorithm 4) with the EntRM functional:

Proposition 2.15. Let π_β^* be an optimal deterministic Markov policy for the Entropic Risk Measure of the return, with risk parameter $\beta \in \mathbb{R}$. Then, for all $t < H$ and all $x \in \mathcal{X}$, $\exists\, a_{t+1}^*(x') \in \arg\max_{a \in \mathcal{A}} \mathrm{EntRM}_\beta[\eta_{t+1}^{\pi_\beta^*}(x', a)]$ such that

$$\eta_t^{\pi_\beta^*}(x) = \sum_{x' \in \mathcal{X}} p(x' \mid x, \pi_\beta^*(x)) \left(\tau_{r(x, \pi_\beta^*(x), x')} \# \eta_{t+1}^{\pi_\beta^*}(x', a_{t+1}^*(x')) \right).$$

Also,

$$\forall\, t < H,\ \forall\, x \in \mathcal{X}, \qquad \mathrm{EntRM}_\beta[\eta_t^{\pi_\beta^*}(x)] = \max_\pi \mathrm{EntRM}_\beta[\eta_t^\pi(x)],$$

2.3.3 Certainty Equivalents and Expected Utilities

What are the risk measures that would match most closely the way rational humans perceive and handle risk? This question has been extensively studied [Yaari, 1987, Tversky and Kahneman, 1992, Wu and Gonzalez, 1996], in particular by Von Neumann and Morgenstern [1944] who established 4 *rationality* axioms that a decision maker should satisfy. They proved the *Von Neumann-Morgenstern utility theorem*, which states that a rational decision maker acts as if they were maximizing a risk measure among the family of *Expected Utilities*. Expected Utilities form a large family of risk measures that include the Entropic Risk Measure and the usual expectation. They display favorable properties for optimizing them in MDPs. We give a brief introduction to these families of risk measures, and how to optimize them in MDPs.

2.3.3.1 An introduction to Certainty Equivalents and Expected Utilities

Definition 2.11. Let $f : \mathbb{R} \to \mathbb{R}$ be non-decreasing. We call *Expected Utility (EU)* the risk measure E_f defined as:

$$E_f[X] = \mathbb{E}[f(X)]$$

where X is any real random variable. Its value may be infinite.

Furthermore, assume f is strictly increasing and continuous. We call *Certainty Equivalent (CE)* the risk measure U_f defined as:

$$U_f[X] = f^{-1}(\mathbb{E}[f(X)])$$

With $f(x) = x$, we recover the usual expectation. When $f(x) = \text{sign}(\beta)\exp(\beta x)$, we recover respectively the exponential utility and the EntRM.

Behavior. Depending on the definition of f, the optimization of such risk measure can lead to either a risk-averse or a risk-seeking behavior (or a mix of both). We can consider two specific cases:

- **Concave utility function** ($f'' < 0$): the objective is risk-averse.
- **Convex utility function** ($f'' > 0$): the objective is risk-seeking.

This behavior can be illustrated through the Taylor expansion of the risk measure [Bäuerle and Rieder, 2014]: under regularity assumptions, we have

$$U_f[X] \approx \mathbb{E}[X] + \frac{1}{2}\frac{f''(\mathbb{E}[X])}{f'(\mathbb{E}[X])}\mathbb{V}(X) \tag{2.23}$$

Hence, if f is concave, the second term is negative and the risk measure penalizes the variance of X. Conversely, if f is convex, the second term is positive and the risk measure favors high-variance distributions. The ratio $-\frac{f''(x)}{f'(x)}$ is called the *Arrow-Pratt measure of absolute risk aversion*. Notably this ratio is constant if and only if f is an exponential function ($f(x) = e^{\beta x}$ for some $\beta \in \mathbb{R}$), which corresponds to the entropic risk measure [Föllmer and Schied, 2011].

Some important examples.

- **The power functions** $f(x) = x^\gamma$ for $\gamma \in \mathbb{N}$ odd. The expected utility is the γ-th moment of X:
$$E_f[X] = \mathbb{E}[X^\gamma] \tag{2.24}$$
It is convex for $x > 0$ (risk-seeking) and concave for $x < 0$ (risk-averse). The resulting certainty equivalent is the γ-mean of the random variable X: $U_f[X] = \mathbb{E}[X^\gamma]^{\frac{1}{\gamma}}$.

- **The exponential function** $f(x) = \text{sign}(\beta)e^{\beta x}$ for $\beta \neq 0$. The resulting certainty equivalent is the EntRM. It is concave for $\beta < 0$ (risk-averse) and convex for $\beta > 0$ (risk-seeking).

- **The step function** $f(x) = \mathbb{1}_{x \geq c}$. The resulting expected utility is the probability of being above the threshold c:
$$E_f[X] = \Pr[X \geq c] \tag{2.25}$$

- **The rectified linear unit** $f(x) = \frac{1}{\alpha}\max(0, c - x)$. The resulting expected utility is the following:
$$E_f[X] = \frac{1}{\alpha}\mathbb{E}[c - X]_+ \tag{2.26}$$
This is linked to the Conditional Value at Risk $\text{CVaR}_\alpha(X)$ [Rockafellar et al., 2000] (see Section 2.3.4.2). It is convex for all $\alpha > 0$.

Von Neumann-Morgenstern Expected Utility Theorem. The axioms of rationality introduced by Von Neumann and Morgenstern [1944] can be seen as an order $\preceq$ on the set of probability distributions $\mathcal{P}(\mathbb{R})$, where $\eta_1 \preceq \eta_2$ means that the decision maker prefers the distribution of rewards η_2 to the distribution η_1. The axioms are the following:

- **Completeness**: the order is total: either $\eta_1 \preceq \eta_2$, or $\eta_2 \preceq \eta_1$, or the decision maker is indifferent between both outcomes $\eta_1 \sim \eta_2$.

- **Transitivity**: The preference is transitive: for any three distributions η_1, η_2 and η_3, if $\eta_1 \preceq \eta_2$ and $\eta_2 \preceq \eta_3$, then $\eta_1 \preceq \eta_3$.

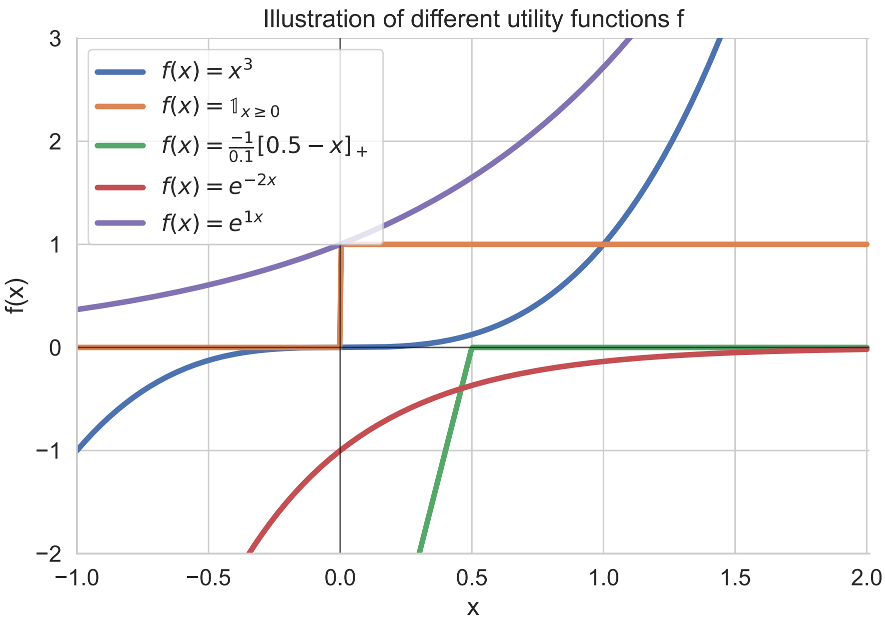

Figure 2.13: Illustration of different utility functions.

- **Independence**: for any three distributions η_1, η_2 and η_3, if $\eta_1 \preceq \eta_2$, then for any $p \in [0,1], p\eta_1 + (1-p)\eta_3 \preceq p\eta_2 + (1-p)\eta_3$.
- **Continuity**: for any three distributions η_1, η_2 and η_3, if $\eta_1 \preceq \eta_2$ and $\eta_2 \preceq \eta_3$, then there exists a probability $p \in (0,1)$ such that $p\eta_1 + (1-p)\eta_3 \sim \eta_2$.

The Von Neumann-Morgenstern Expected Utility Theorem states that if a decision maker's preferences satisfy these four axioms, then there exists a non-decreasing function $f : \mathbb{R} \to \mathbb{R}$ such that for any two distributions η_1 and η_2, we have $\eta_1 \preceq \eta_2$ if and only if $E_f[\eta_1] \leq E_f[\eta_2]$. In other words, the decision maker behaves as if they were maximizing an expected utility.

To give a counterexample, in Section 2.3 we introduced an MDP where the optimal policy for the VaR did not verify the Bellman optimality principle. The proof revolved around the fact that we had distributions $\eta^\pi(x,a), \eta^\pi(x,b), \delta_r$ and probability p such that $\mathrm{VaR}_\alpha[\eta^\pi(x,a)] > \mathrm{VaR}_\alpha[\eta^\pi(x,b)]$, yet $\mathrm{VaR}_\alpha[p \cdot \eta^\pi(x,a) + (1-p) \cdot \delta_r] < \mathrm{VaR}_\alpha[p \cdot \eta^\pi(x,b) + (1-p) \cdot \delta_r]$. This violates the independence axiom, and therefore the VaR cannot be represented as an expected utility.

While these axioms have been widely accepted as a reasonable description of rational behavior, they have also been criticized for not accurately describing actual human behavior in most situations (see [Tversky and Kahneman, 1992] for instance). The most famous counterexample of these axioms is the Allais paradox [Allais, 1990], and can be illustrated by the following example. A decision maker is first given the choice between two lotteries:

- Lottery A: a sure gain of 10 million dollars.
- Lottery B: a 90% chance to win 15 million dollars, and a 10% chance to win nothing.

Most people choose Lottery A, preferring the sure gain over the risky option. Next, the decision maker is given a second choice between two different lotteries:

- Lottery A': a 10% chance to win 10 million dollars, and a 90% chance to win nothing.
- Lottery B': a 9% chance to win 15 million dollars, and a 91% chance to win nothing.

Most people choose Lottery B', preferring the higher potential gain despite the lower probability of winning. Such specific choices violate the independence axiom: lottery A' and B' can be seen as a mixture of lottery A and B with a 90% chance of winning nothing, so the preference should not change.

Remark 2.3. Similarly to the EntRM and the exponential utility, optimizing upon an expected utility E_f is equivalent to optimizing upon the certainty equivalent U_f. Indeed, if f is strictly increasing, then f^{-1} is also strictly increasing. Then, for any two random variables X and Y, we have

$$E_f[X] \leq E_f[Y] \Leftrightarrow U_f[X] \leq U_f[Y].$$

2.3.3.2 Expected Utilities in MDPs

We now turn to the problem of optimizing an expected utility in the context of MDPs. The results presented here were mainly developed by Bäuerle and Rieder [2014]. Let f be a non-decreasing function, and consider the following objective:

$$\max_{\pi\in\Pi} E_f\left[\mathcal{R}^\pi\right] = \max_{\pi\in\Pi} \mathbb{E}_\pi\left[f(\mathcal{R}^\pi)\right]. \tag{2.27}$$

In this subsection, we do not assume that f is strictly increasing or continuous. Such assumptions are unnecessary for the dynamic programming approach presented below, and relaxing them allows us to cover important cases such as the step function and the rectified linear function introduced earlier, both of which are not strictly increasing (and in the case of the step function, not continuous either).

The first difficulty is that, for general f, optimal policies may not be Markov (see the example with $f(x) = x^3$ in Section 2.3). This contrasts with the cases of the standard expected return and of the entropic risk measure, where Markov policies are sufficient for optimality. Characterizing the set of function f for which Markov policies are sufficient is an important problem that we tackle in Section Chapter 3. For now, we focus on the general case.

Stock-augmented policies. Although general optimal policies may depend on the entire history, it turns out that we can restrict ourselves to the smaller class of policies called *stock-augmented policies* [Kreps, 1977, Bäuerle and Rieder, 2014]. These are policies that depend on the history only through the current state and the accumulated reward (called *stock*) so far.

Definition 2.12 (Stock-augmented policies)**.** Let $i \in [H]$ and $h = (x_0, a_0, r_0, \dots, x_i) \in \mathcal{H}_i$. The *stock* of h is defined as

$$\operatorname{stock}(h) = \sum_{t=0}^{i-1} r_t .$$

The set of stock-augmented policies is then defined as

$$\Pi_{\mathrm{St}} = \left\{ \pi \in \Pi \;\middle|\; \forall i \in [H],\ \forall h, h' \in \mathcal{H}_i,\ \operatorname{stock}(h) = \operatorname{stock}(h') \implies \pi_i(\cdot \mid h) = \pi_i(\cdot \mid h') \right\}.$$

This restriction is without loss of generality: there is always an optimal policy that depends on the history only through the current state and the stock.

Theorem 2.2. There exists an optimal policy $\pi^* \in \Pi_{\mathrm{St}}$ such that

$$E_f\left[\mathcal{R}^{\pi^*}\right] = \max_{\pi \in \Pi} E_f[\mathcal{R}^{\pi}] .$$

EU value functions. We now introduce the value functions associated with expected utilities. Since stock-augmented policies are necessary and sufficient, these value functions naturally depend on both the current state and this stock.

Definition 2.13 (EU value functions)**.** Let $\pi \in \Pi_{\mathrm{St}}$. For all $t \in [H]$, $x \in \mathcal{X}$, and $c \in \mathbb{R}$, the expected utility value function is defined as

$$V_{t,f}^{\pi}(x, c) = \mathbb{E}_{\pi}[f(\mathcal{R}_t^{\pi} + c)] .$$

The corresponding optimal value function is

$$V_{t,f}^{*}(x, c) = \max_{\pi \in \Pi_{\mathrm{St}}} V_{t,f}^{\pi}(x, c).$$

The optimization objective is recovered as $V_{0,f}^{*}(x_0, 0)$.

Bellman equations. These value functions satisfy the following recursive relations:

$$V^{\pi}_{H,f}(x, c) = f(c), \tag{2.28}$$

$$V^{\pi}_{t,f}(x, c) = \mathbb{E}_{\pi}[V^{\pi}_{t+1,f}(X_{t+1}, c + r(x, A_t, X_{t+1}))], \quad \forall t < H. \tag{2.29}$$

Where $\mathbb{E}_{\pi}$ denotes the expectation over $A_t \sim \pi_t(\cdot \mid x, c)$ and $X_{t+1} \sim p(\cdot \mid x, A_t)$. This equation follows directly from the decomposition

$$\mathcal{R}^{\pi}_t + c = (\mathcal{R}^{\pi}_{t+1} + r_t) + c = \mathcal{R}^{\pi}_{t+1} + (r_t + c),$$

together with the tower property of conditional expectation. Similarly, the optimal value function satisfies

$$V^{*}_{H,f}(x, c) = f(c), \tag{2.30}$$

$$V^{*}_{t,f}(x, c) = \max_{a \in \mathcal{A}} \mathbb{E}_{X_{t+1} \sim p(\cdot|x,a)}[V^{*}_{t+1,f}(X_{t+1}, c + r(x, a, X_{t+1}))], \quad \forall t < H. \tag{2.31}$$

These equations extend the classical Bellman equations to the setting of expected utilities, at the price of augmenting the state space with the stock variable c.

Stock-augmented MDPs. An alternative yet equivalent viewpoint for handling stock-augmented policies is to reformulate the problem as a standard Markov Decision Process over an augmented state space. Specifically, we define the *stock-augmented MDP* $\mathcal{M}' = (\mathcal{X}', \mathcal{A}, p', r', H, x'_0)$, where:

- $\mathcal{X}' = \mathcal{X} \times \mathbb{R}$ is the augmented state space. The second component stores the accumulated reward (or *stock*) up to time t.
- $\mathcal{A}' = \mathcal{A}$: the action space remains unchanged.
- The transition function p' is defined as:

$$p'((x_{t+1}, c_{t+1}) \mid (x_t, c_t), a_t) = \begin{cases} p(x_{t+1} \mid x_t, a_t) & \text{if } c_{t+1} = c_t + r(x_t, a_t, x_{t+1}), \\ 0 & \text{otherwise.} \end{cases}$$

- The reward function is defined as 0 for all transitions except at the terminal time step H, where it is given by the utility function applied to the accumulated stock:

$$r'((x_t, c_t), a_t, (x_{t+1}, c_{t+1})) = 0, \quad \forall t < H - 1, \tag{2.32}$$

$$r'((x_{H-1}, c_{H-1}), a_{H-1}, (x_H, c_H)) = f(c_H). \tag{2.33}$$

We note that in this MDP, the reward is not stationary since it depends on the time step t, as opposed to the original MDP $\mathcal{M}$.

- The horizon and initial state remain unchanged: $H' = H$, $x'_0 = (x_0, 0)$.

This construction ensures that the dynamics of the original MDP are preserved, while tracking the accumulated return through the augmented state variable. Each trajectory in $\mathcal{M}$ corresponds to a trajectory in $\mathcal{M}'$, and vice versa. The key benefit of this formulation is that it reduces the optimization over stock-augmented policies in $\mathcal{M}$ to a standard Markov policy optimization problem in $\mathcal{M}'$: since both the original state and the stock are incorporated in the augmented state, the policy only depends on the augmented state. Therefore, standard dynamic programming algorithms can be directly applied to compute an optimal policy for the expected utility objective [Bäuerle and Rieder, 2014].

Value Iteration We present in Algorithm 8 the corresponding value iteration algorithm. Here, we denote R_t the set of possible stock values at time t, that is $R_t = \{\text{stock}(h) \mid h \in \mathcal{H}_t\}$. As it is not feasible to compute exactly for all values of the stock $c \in \mathbb{R}$, we only compute the value function for the reachable stock values $c \in R_t$ at each time step t. The set R_t is finite since there are a finite number of possible rewards at each step and a finite horizon. This algorithm works similarly as the previous ones except now the value function depends on this stock, and we need to loop over all possible stock values at each time step.

Algorithm 8 Expected Utility Value Iteration

```
Require: MDP $(\mathcal{X}, \mathcal{A}, P, r, H)$, utility function $f : \mathbb{R} \to \mathbb{R}$.
Ensure: $V_t^*(x, c), Q_t^*(x, c, a)$ for all $x \in \mathcal{X}$, $c \in R_t$, $a \in \mathcal{A}$, $t \in [H]$ and $\pi_t^*(x, c)$ for all
    $x \in \mathcal{X}$, $c \in R_t, t \in [H]$.
Initialize $V_H^*(x, c) \leftarrow f(c)$ for all $x \in \mathcal{X}$ and $c \in R_H$.
for $t = H - 1$ to $0$ do
    for all $x \in \mathcal{X}$ do
        for all $c \in R_t$ do
            for all $a \in \mathcal{A}$ do
                $Q_t^*(x, c, a) \leftarrow \sum_{x' \in \mathcal{X}} p(x'|x, a)[V_{t+1}^*(x', c + r(x, a, x'))]$
            end for
            $V_t^*(x, c) \leftarrow \max_{a \in \mathcal{A}} Q_t^*(x, c, a)$
            $\pi_t^*(x, c) \in \arg\max_{a \in \mathcal{A}} Q_t^*(x, c, a)$
        end for
    end for
end for
output $V^*$ and $\pi^*$.
```

Complexity. Both Algorithm 8 and this reformulation highlight a computational challenge: the complexity scales with the size of the number of possible stock values in R_t. Indeed, the time complexity of the value iteration algorithm is $O\left(\sum_{t=0}^{H-1} |R_t| \cdot |\mathcal{X}|^2 \cdot |\mathcal{A}|\right)$. We can also bound $|R_t|$ to give a more explicit complexity bound. Let $R = \{r(x, a, x') \mid (x, a, x') \in \mathcal{X} \times \mathcal{A} \times \mathcal{X}\} \subset \mathbb{R}$ be the finite set of all possible immediate rewards, whose size is at most $|\mathcal{X}|^2 |\mathcal{A}|$ in the worst case. Then, the number of possible stock values at t is upper bounded the number of multisets (with repetition) of size t drawn from R. This number is given by: $|R_t| \leq \binom{|R|+t-1}{t}$, which grows polynomially in $|R|$ and thus in $|\mathcal{X}|$ and $|\mathcal{A}|$, but exponentially in t. Therefore, the time complexity of solving the stock-augmented MDP using dynamic programming can be bounded by $O\left(\sum_{t=0}^{H-1} \binom{|R|+t-1}{t} \cdot |\mathcal{X}|^2 \cdot |\mathcal{A}|\right)$ which is exponential in the horizon H in the general case.

Complexity with Structured Rewards. In most situations, the complexity can be reduced by exploiting structure in the reward function. For instance, if all rewards lie on a uniform grid of size n over a bounded interval (e.g., $[-1, 1]$), then the set of reachable return values is a subset of a regular discrete lattice. In that case, the number of possible stocks is at most nH, and the complexity becomes $O\left(n^2 \cdot |\mathcal{X}|^2 \cdot |\mathcal{A}| \cdot H^3\right)$, which is polynomial in H, n, and the size of the MDP. This illustrates that the difficulty of solving expected utility problems in MDPs comes from the explosion in the number of possible stock values.

A relevant set of MDPs where this complexity is mitigated is when the rewards may only be received once, like in a *terminal state*. Consider for example the cliff environment (Figure 2.2) where the agent does not receive the -0.01 reward penalty at each step. The agent only receives a reward when it reaches the goal (+1) or falls off the cliff (-1), and then stays stationary without receiving rewards. In such a case, along a trajectory, the stock always remains zero until the terminal state and there will be no added complexity compared to the classical setting. Notably, the optimal policy will be Markov.

In Chapter 4, we will study a way to mitigate this complexity for certain utilities, using an approximation of the objective with the entropic risk measure.

2.3.4 Quantile Based Risk Measures

When making decisions under uncertainty, it is natural to reason in terms of quantiles of possible outcomes. For instance, one may ask: “what are the 95% most likely values of my random variable?” Such questions typically arise when we wish to control risk. A decision-maker concerned with safety may aim to maximize the 5% worst outcomes of a treatment, or equivalently, to ensure that with 95% confidence, the outcome exceeds a given threshold. This threshold corresponds to a quantile of the outcome distribution, leading to the notion of the *Value at Risk* (VaR).

Value at Risk. The Value at Risk of a random variable X at level $\alpha \in [0, 1]$ corresponds to its upper α-quantile:

$$\mathrm{VaR}_\alpha(X) := \inf\{\tau \in \mathbb{R} : P(X \leq \tau) > \alpha\}. \tag{2.34}$$

It represents the smallest value that the outcome exceeds with probability $1 - \alpha$.

VaR is one of the most widely used risk measures due to its simplicity and interpretability. It is central to financial regulation, notably through the Basel II Accords [Berkowitz and O'Brien, 2002], and has also found applications in diverse domains such as reliability engineering [DeCandia et al., 2007] and epidemiology [Wei et al., 2019].

However, VaR suffers from important limitations. It provides only a single point estimate of the distribution, offering no information about the severity of losses beyond this point, and can thus underestimate tail risk. This lack of sensitivity to extreme events was one of the reasons cited for its inadequacy during the 2008 financial crisis [Nocera, 2009].

To address these limitations, the *Conditional Value at Risk* (CVaR) was introduced as an extension of VaR that accounts for the entire tail beyond the VaR threshold.

Conditional Value at Risk. The *Conditional Value at Risk*, also known as the *Average Value at Risk* (AVaR) [Bäuerle and Ott, 2011] or *Expected Shortfall* [Acerbi and Tasche, 2002], is defined as the expected value of the worst α fraction of outcomes. It is therefore more sensitive to the magnitude of extreme events. Formally, for a random variable X and level $\alpha \in [0, 1)$,

$$\mathrm{CVaR}_\alpha(X) := \frac{1}{\alpha} \int_0^\alpha \mathrm{VaR}_\gamma(X)\, d\gamma \ . \tag{2.35}$$

Equivalently, CVaR admits the following integral and optimization representations:

$$\mathrm{CVaR}_\alpha(X) = \sup_{\tau \in \mathbb{R}} \left\{ \tau - \frac{1}{\alpha} \mathbb{E}[\tau - X]_+ \right\}. \tag{2.36}$$

The variational form (2.36), due to Rockafellar et al. [2000], is particularly useful as it expresses CVaR as an expected utility. Importantly, this supremum is attained for the value $\tau = \mathrm{VaR}_\alpha(X)$. Furthermore, the CVaR is always upper bounded by the VaR:

$$\mathrm{CVaR}_\alpha(X) \leq \mathrm{VaR}_\alpha(X) \ , \tag{2.37}$$

and for $\alpha = 1$, it matches the expectation: $\mathrm{CVaR}_1(X) = \mathbb{E}[X]$.

CVaR possesses strong theoretical properties. It is a coherent risk measure [Artzner et al., 1999], meaning that it satisfies a specific set of axioms, notably the subadditivity (in contrast to the VaR) and the positive homogeneity (in opposition to the EntRM), which are sought-after properties in risk-management. Because of these properties, CVaR

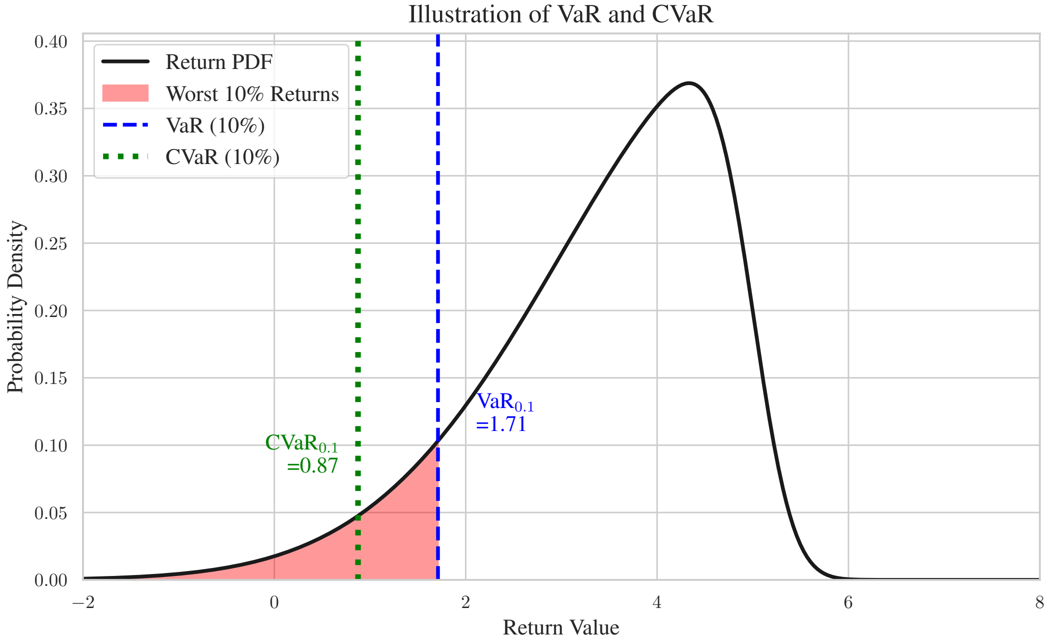

Figure 2.14: Illustration of Value at Risk (VaR) and Conditional Value at Risk (CVaR). The VaR at level α corresponds to the α-quantile of the distribution (dashed line). The CVaR at level α corresponds to the average of all values below the VaR (shaded area). CVaR is more sensitive to the tail of the distribution than VaR.

is often preferred in practice and has been applied in domains such as finance, inventory control, and supply-chain management (see [Filippi et al., 2020] for a comprehensive survey). It has also replaced VaR as the standard regulatory risk measure under the Basel III Accords [Basel Committee on Banking Supervision, 2013].

VaR convention. Several conventions exist in the literature regarding the definition of VaR and CVaR. Many authors define VaR as the lower α-quantile, which corresponds to the smallest value exceeded with probability at least $1 - \alpha$ [Rockafellar et al., 2000, Bäuerle and Ott, 2011, Ahmadi-Javid, 2012]. This definition is particularly common in finance, where VaR is often interpreted as a loss measure that one tries to minimize. In that setting, the CVaR is often defined as the upper tail of the distribution. In the MDP setting, we consider a return maximization problem, which explains why we prefer using here a *reversed* convention, where VaR is defined as the upper quantile and CVaR as the lower tail [Hau et al., 2023a, Acerbi and Tasche, 2002]. Both conventions are equivalent up to a sign change $X' \leftarrow -X$ of the random variable considered.

2.3.4.1 VaR Dynamic Programming in MDPs

We here present how to optimize VaR in MDPs using a state augmentation technique. As illustrated in Section 2.3, VaR does not satisfy the Bellman optimality principle, and therefore cannot be optimized using the usual Dynamic Programming. Here again it will be necessary to use a state augmentation technique. The augmentation will not be on the stock but on the quantile level. We will need to keep track of all the quantile levels in $[0, 1]$ at every step, and the policy will depend on a quantile level that will vary during the process. To give an intuition on this varying quantile, consider the following example.

Remark 2.4. Computing all the α-quantiles of a distribution is equivalent to computing the quantile function F_X^{-1}, which itself is equivalent to computing the cumulative distribution function F_X and thus equivalent to computing the full distribution.

The quantile level to optimize varies over time. Consider a simple example with distributions illustrated in Figure 2.15. At time $t = 0$, from state x_0, taking action a leads to state x_1 with probability 0.5 and to state x_2 with probability 0.5 too. Assume the return from x_1 onward $R_1(x_1) = \eta_1$ is bounded in $[0, 1]$ and the return from x_2 onward $R_1(x_2) = \eta_2$ is bounded in $[1, 2]$. Then $R_0(x_0, a) = \eta_3$ is a mixture of these two distributions, with 50% of the mass in $[0, 1]$ and 50% in $[1, 2]$. In this case, for any $\alpha < 0.5$, the α-quantile of η_3 is in $[0, 1]$ and thus corresponds to the $\frac{\alpha}{0.5} = 2\alpha$-quantile of η_1. On the other hand, for any $\alpha \geq 0.5$, the α-quantile of η_3 is in $[1, 2]$ and thus corresponds to the $(\alpha - 0.5)/0.5 = 2\alpha - 1$-quantile of η_2. Hence, optimizing on the initial quantile level α at step $t = 0$ leads to optimizing on a different quantile level at step $t = 1$, depending on the state reached.

Computing quantiles of mixtures. Computing quantiles of mixtures of distribution is important because, in an MDP, the return at a given step is a mixture of the returns from all following states (by the distributional Bellman equation). In the example above, the distributions have disjoint supports, making the quantile of the mixture easy to derive. In general, the formula is more complex. Li et al. [2022] (revised by [Hau et al., 2023a]) showed that the quantiles (and thus, VaR) of a mixture of distributions can be computed as follows:

Lemma 2.3 (VaR of a mixture, [Li et al., 2022]). Let $\eta_1, \ldots, \eta_n$ be n distributions and $p_1, \ldots, p_n$ be n probabilities summing to 1. Let $\eta = \sum_{i=1}^n p_i \eta_i$ be the mixture distribution. Then, for any $\alpha \in [0, 1]$,

$$\mathrm{VaR}_\alpha[\eta] = \max_{\mathbf{q}} \min_{i \in [n]} \mathrm{VaR}_{q_i}(\eta_i),$$

where $\mathbf{q} \in [0, 1]^n$ is such that $\sum_{i=1}^n p_i q_i \leq \alpha$.

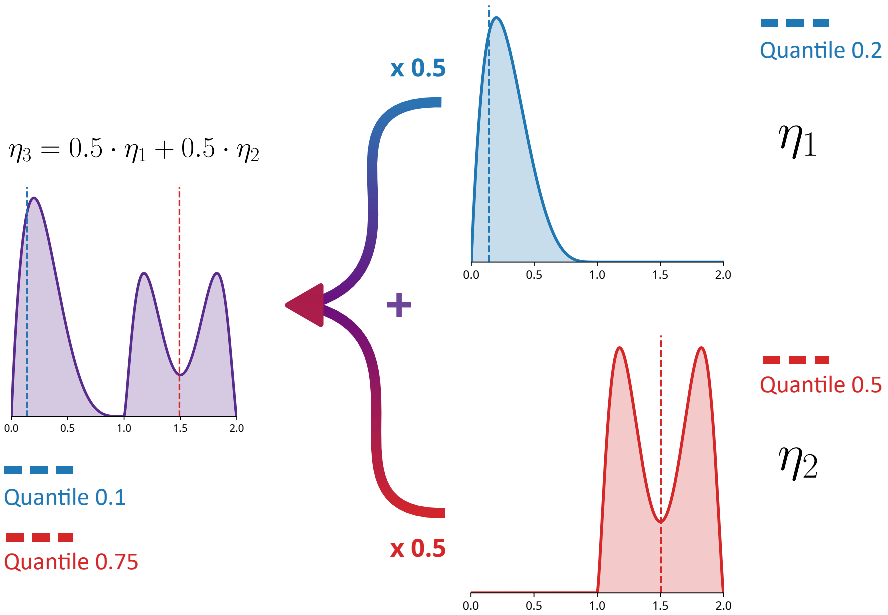

Figure 2.15: Example of quantile levels varying over time. The distribution at step $t = 0$ is a mixture of the two distributions at step $t = 1$. The quantile level at step $t = 1$ depends on the state reached.

Intuitively, $\mathbf{q}$ represents the quantile levels of each component distribution η_i that will be mixed to obtain the quantile level α of the mixture. The constraint $\sum_{i=1}^n p_i q_i \leq \alpha$ ensures that the quantile levels q_i are chosen so that the overall quantile level is at most α. In particular for continuous distributions, by writing $\tau = \mathrm{VaR}_\alpha[\eta]$, the maximum is reached for $\mathbf{q}$ such that $\forall i, q_i = \int_{-\infty}^{\tau} d\eta_i$.

Li et al. [2022] presents an algorithm to solve this optimization problem for all $\alpha \in [0, 1]$ simultaneously but only in the case of discrete distributions. Their solution consists of simply computing the quantile function of the mixture distribution which is equivalent to computing the mixture of distributions (see Remark 2.4) and can be done easily for discrete distributions.

The following alternative formula is used by Hau et al. [2024]:

Lemma 2.4. Let $\eta_1, \ldots, \eta_n$ be n distributions and $p_1, \ldots, p_n$ be n probabilities summing to 1. Let $\eta = \sum_{i=1}^n p_i \eta_i$ be the mixture distribution. Let $q(i, \alpha) = \mathrm{VaR}_\alpha[\eta_i]$ be the quantile value of each component distribution. Then, for any $\alpha \in [0, 1]$,

$$\mathrm{VaR}_\alpha[\eta] = \mathrm{VaR}_\alpha[q(X, U)]$$

where X is a random variable with values in $[n]$ and $\Pr(X = i) = p_i$ and U is a uniform random variable on $[0, 1]$ independent of X.

The formula is less explicit than the previous one for computing the quantile, but solving it relies on the same principle: $q(i, U) \sim \eta_i$ when $U \sim \mathcal{U}([0,1])$ (see for example [Wasserman, 2013]) and thus $q(X, U) \sim \eta$. Hence the computation in practice also amounts to computing the full distribution of η and then computing its quantile. The advantage of this formula is that it is more adapted to settings where the model is not known and can only be sampled (which is the setting considered in [Hau et al., 2024]).

The complexity of computing such quantile functions is just the complexity of computing the mixture of distributions. For discrete distributions where their support is finite, then the complexity is linear in the size of the supports of the distributions: $O\left(\sum_i |\text{support}(\eta_i)|\right)$.

VaR Bellman equations. Computing the quantile function of the return amounts to computing its full distribution, which will be studied in more detail in Section 2.2. Here, we only focus on the policy optimization problem:

$$\max_{\pi} \mathrm{VaR}_\alpha[\mathcal{R}^\pi]. \tag{2.38}$$

We define the optimal quantile value function similarly as in other problems:

$$V_t^*(x, \alpha) = \max_{\pi} \mathrm{VaR}_\alpha[\mathcal{R}_t^\pi(x)] \tag{2.39}$$

$$Q_t^*(x, \alpha, a) = \max_{\pi} \mathrm{VaR}_\alpha[\mathcal{R}_t^\pi(x, a)]. \tag{2.40}$$

Leveraging Lemma 2.3, we can write the following dynamic programming equations [Li et al., 2022, Hau et al., 2023a]:

$$Q_t^*(x, \alpha, a) = \max_{\mathbf{q}} \min_{x'} r(x, a, x') + V_{t+1}^*(x', q_{x'}) \tag{2.41}$$

$$V_t^*(x, \alpha) = \max_{a} Q_t^*(x, \alpha, a) \tag{2.42}$$

where $\mathbf{q} \in [0,1]^{\mathcal{X}}$ is such that $\sum_{x'} P(x'|x, a) q_{x'} \leq \alpha$. Hence the optimal α-quantile is the α-quantile of a mixture of specific distributions $(\eta_{x'})_{x' \in \mathcal{X}}$ that each satisfy $\mathrm{VaR}_\alpha[\eta_{x'}] = r(x, a, x') + V_{t+1}^*(x', \alpha)$. Alternatively, $\eta_{x'} = \mathcal{L}\left(r(x, a, x') + V_{t+1}^*(x', U)\right)$ for $U \sim \mathcal{U}([0,1])$.

This formula also explains how the quantile level α will vary over time. Let $\mathbf{q}$ be the optimal solution of the optimization problem. Then, the action of the following step at state x' will be taken with respect to the quantile level $q_{x'}^*$.

Lemma 2.4 can also be used to obtain a similar dynamic programming equation [Hau et al., 2024]:

$$Q_t^*(x, \alpha, a) = \mathrm{VaR}_\alpha\left[r(x, a, X') + \max_{a'} Q_{t+1}^*(X', U, a')\right]. \tag{2.43}$$

However this formula is less explicit and does not provide a clear intuition on how the quantile level varies over time.

Complexity of computing the optimal quantile. Computing the optimal quantile value function is done through a value iteration algorithm using the above Bellman equations. It is a H-step backward induction, where two operations are performed at each step: first computing the quantile function $Q_t^*(x, \cdot, a)$ for each state x and action a, then computing the quantile function $V_t^*(x, \cdot)$ from the Q-functions for all states x.

The first part amounts to computing the mixtures of distributions as explained above. The complexity of this operation hence depends on the size of the supports and of the number of distributions involved. In our finite MDPs, the support will always be finite, so the algorithm is tractable. As in Section 2.3.3.2, we denote R_t the possible return values at step t, such that the support of these distributions is of size at most $|R_t|$, and there are $|\mathcal{X}|$ distributions to consider. The complexity of computing the mixture is thus $O(|\mathcal{X}| \cdot |R_{t+1}|)$. Iterating over all actions and states, the total complexity of this part is $O(|\mathcal{A}| \cdot |\mathcal{X}|^2 \cdot |R_{t+1}|)$.

The second part amounts to taking the maximum over actions for each quantile level. As the support is finite, the quantile function is piecewise constant, with the breakpoints being the elements of the support. Hence, the maximum only has to be computed once between each pair of breakpoints. The complexity of that computation is thus $O(|R_{t+1}| \cdot |\mathcal{A}|)$. Iterating over all states, the total complexity of this part is $O(|\mathcal{A}| \cdot |\mathcal{X}| \cdot |R_{t+1}|)$.

Hence, each step of the value iteration algorithm has a complexity of $O(|\mathcal{A}| \cdot |\mathcal{X}|^2 \cdot |R_{t+1}|)$. Over H steps, the total complexity is thus $O(|\mathcal{A}| \cdot |\mathcal{X}|^2 \cdot H \cdot |R_0|)$. This complexity is the same as for expected utilities, see Section 2.3.3.2 for a discussion about the size of the supports R_t.

Duality with the Probability threshold. Having the same complexity is natural because of the duality between quantiles and probability thresholds ($X \mapsto \Pr(X \leq \tau)$, the expected utility associated to $X \mapsto \mathbb{1}_{X \leq \tau}$). Indeed, the quantile function of a distribution is the inverse of its cumulative distribution function. Computing one is thus equivalent to computing the other.

Remark 2.5. When solving for the optimal VaR, the algorithms do so for all the quantiles. This lead to a obtaning a quantile function $F^{-1}(\alpha) = V_0^*(x_0, \alpha)$ for all $\alpha \in [0, 1]$. This function is associated to a distribution that, by definition, stochasticly dominates the return distribution of any (potentially history dependent) policy. However, this is in general not the return distribution of any policy. The optimal policy obtained through the value iteration algorithm depends on the initial quantile, and different initial quantiles may lead to different policies. For example, if it was the case, that same policy would be optimal for any expected utilities. In most MDPs considered in this thesis, different objectives will lead to different optimal policies, which is why risk-sensitive objectives are relevant in the first place.

2.3.4.2 CVaR optimization in MDPs

Similarly to the VaR, evaluating the CVaR of the return of a policy relies solely on computing the full distribution of the return, which is studied in Section 2.2. We present here the different results proposed in the literature to optimize CVaR in MDPs. First, we recall the objective. For $\alpha \in [0, 1]$, the risk-sensitive objective is:

$$\max_{\pi} \mathrm{CVaR}_{\alpha}[\mathcal{R}^{\pi}]. \tag{2.44}$$

In a similar fashion as for VaR or the expected utilities, CVaR does not satisfy the Bellman optimality principle [Bäuerle and Ott, 2011] and thus cannot be optimized using standard Dynamic Programming.

Primal representation. Bäuerle and Ott [2011] suggest using the optimization formulation of CVaR (2.36) (also called *primal representation*) to reformulate the problem as a joint optimization over policies π and threshold τ:

$$\begin{aligned} \max_{\pi} \mathrm{CVaR}_{\alpha}[\mathcal{R}^{\pi}] &= \max_{\pi} \sup_{\tau \in \mathbb{R}} \left\{ \tau - \frac{1}{\alpha} \mathbb{E}[\tau - \mathcal{R}^{\pi}]_{+} \right\} & (2.45) \\ &= \sup_{\tau \in \mathbb{R}} \left\{ \tau + \max_{\pi} \frac{-1}{\alpha} \mathbb{E}[\tau - \mathcal{R}^{\pi}]_{+} \right\}. & (2.46) \end{aligned}$$

The relevant part of this reformulation is that the maximization over policies is only on an expected utility, which we know how to optimize using stock-augmented states (see Section 2.3.3.2). However, this policy optimization depends on the threshold τ, on which we also optimize. Nevertheless, Bäuerle and Ott [2011] shows that there exists a fixed τ^* such that optimizing the expected utility with respect to this threshold leads to an optimal policy for the CVaR optimization problem:

Theorem 2.5. Let $\alpha \in (0, 1)$. For any $\tau \in \mathbb{R}$, let π^*_{τ} be an optimal policy for the expected utility maximization problem:

$$\max_{\pi} -\mathbb{E}[\tau - \mathcal{R}^{\pi}]_{+}. \tag{2.47}$$

Then, there exists $\tau^* \in \mathbb{R}$ such that the policy $\pi^*_{\tau^*}$ is optimal for the CVaR_{α} maximization problem.

As the optimal policy for the CVaR is also the optimal policy for an expected utility, this theorem implies that the set of policy to consider is that of stock-augmented policies (see Theorem 2.2).

However, there is no clear way to find the right threshold τ^*. The function $\tau \mapsto \tau - \max_{\pi} \mathbb{E}[\tau - \mathcal{R}^{\pi}]_{+}$ is for instance not necessarily concave [Bäuerle and Ott, 2011]. Several approximation schemes have been given to find the optimal policy [Bäuerle and Ott, 2011, Lim and Malik, 2022, Yu et al., 2017, Bellemare et al., 2023], either by computing values of τ on a grid or by computing iteratively $\pi^*_{\tau_t}$ and $\tau_{t+1} = \mathrm{VaR}_\alpha[\mathcal{R}^{\pi_{\tau_t}}]$. It is possible to control the approximation error of these schemes, but to the best of our knowledge there exists no exact algorithm to find the true optimal policy in finite time.

Dual representation. Another approach introduced by Chow et al. [2015] uses a dual representation of CVaR [Artzner et al., 1999]:

$$\mathrm{CVaR}_\alpha[X] = \inf_{Q \ll P, \frac{dQ}{dP} \leq \frac{1}{\alpha}} \mathbb{E}_Q[X] \tag{2.48}$$

This representation allows to express the CVaR recursively using the tower property of the expectation [Pflug and Pichler, 2016]:

$$\mathrm{CVaR}_\alpha[r(S,A,S')] = \inf_{\xi \in \mathcal{Z}_C} \sum_{s \in \mathcal{X}} \xi_s \, \mathrm{CVaR}_{\alpha \xi_s \hat{p}_s^{-1}}[r(s,A,S') \mid S = s] \tag{2.49}$$

with $S \sim \hat{p}$ and $\mathcal{Z}_C = \{\xi \in \Delta_{\mathcal{X}} : \forall s, \ \alpha \cdot \xi_s \leq \hat{p}_s\}$. However, when designing an optimal Bellman equation, they claim that the identity

$$\max_{\pi \in \Pi} \mathrm{CVaR}_\alpha^{A \sim \pi(S)}[r(S,A,S')] = \min_{\xi \in \mathcal{Z}_C} \sum_{s \in \mathcal{X}} \xi_s \left(\max_{d \in \Delta_{\mathcal{A}}} \mathrm{CVaR}^{A \sim d}_{\alpha \xi_s \hat{p}_s^{-1}}[r(s,A,S')] \right) \tag{2.50}$$

holds (they permute the $\inf_\xi$ and $\max_\pi$ operators), and make an algorithm relying on it. Hau et al. [2023a] proved that this identity is in general not true, even for a very simple MDP. Their counter-example invalidates the optimality of the algorithm of Chow et al. [2015] and many others that built on it (see the discussion in Hau et al. [2023a] for the affected works).

Other approaches. Some work designed algorithms to optimize CVaR with assumptions on the space of policies. Lim and Malik [2022] provide an algorithm under the assumption that the optimal policy is Markov, while Achab and Neu [2021] provide an algorithm that outputs the best policy among a set of policies with the same expectation. In general, finding the optimal CVaR policy remains an open problem.

3

Theoretical Limitations of Risk Measures in MDPs

Contents

Despite recent progress, the full understanding of the abilities and limitations of the distributional framework to compute other risk measures remains incomplete, with the underlying theory yet to be fully understood.

In this chapter, we explore policy evaluation and policy optimization algorithms for undiscounted MDPs with general functionals of the return. We explicitly delimit the possibilities offered by dynamic programming as well as the distributional framework.

This chapter specifically addresses two questions:

(i) How accurately can we evaluate statistics using the distributional framework?

(ii) Which risk measures can be exactly optimized through dynamic programming?

Addressing question (i), we refer to Rowland et al. [2019]'s results on Bellman closedness and provide their adaptation to undiscounted MDPs. We then prove upper bounds on the approximation error of policy evaluation using distributional techniques and corroborate these bounds with practical experiments. For question (ii), we draw a connection between Bellman closedness and Policy Optimization. We then utilize the distributional framework to identify two key properties held by *optimizable* risk measures.

Our main contribution is a characterization of the families of utilities that verify these two properties (Theorem 3.5). This result gives a comprehensive answer to question (ii) and closes an important open issue in the theory of MDPs. It shows in particular that DistRL does not seem to extend the class of risk measures for which policy optimization is possible beyond what is already allowed by classical dynamic programming.

3.1 Policy Evaluation: (Distributional) Dynamic Programming

In Section 2.2.1, we explained how the distributional framework could be used to compute the full distribution of the return for a given policy by dynamic programming. Theoretically, this directly provides a policy evaluation algorithm for any (law-invariant) risk measure. The method can simply be described in two steps:

1. Compute the distribution of the return η^π for the policy π by dynamic programming (Algorithm 3).

2. Compute the risk measure of interest $\varphi(\eta^\pi)$ by applying its definition to the obtained distribution.

This highlights the power of the distributional approach: it solves the policy evaluation problem for any risk measure in a unified way. However, in practice, this approach still faces the challenge of the exponential complexity of computing the exact distribution (see discussion in Section 2.2.1). This exponential complexity is unnecessary for some risk measures, such as the expectation of the return, or the Entropic Risk Measure family (see Section 2.3.2), when the policy is Markov. Sobel [1982] also showed that a regular (i.e. polynomial time) dynamic programming algorithm could be used to compute the variance of the return. These examples can be seen as special cases of approximate distributional dynamic programming, where the distributions are represented by a single statistic (Expectation, or Entropic Risk) for the expected return and Entropic Risk Measure, or by two statistics (first and second moment) for the variance [Sobel, 1982][1]. The natural question is, for which risk measures is there a representation of the return distributions (with a finite number of statistics) that allows performing exact policy evaluation? This problem was studied by Rowland et al. [2019] for the discounted MDP setting. In the following section, we mention how their results apply in the undiscounted, finite-horizon setting. For the remainder of this section, we only consider Markov policies as they are the class of policies associated with regular dynamic programming.

3.1.1 Exact Evaluation with Parametrized Distributions

The previous policy evaluation problem can be extended to a finite family of statistics: instead of recursively evaluating a single statistic, we evaluate several at the same time. The reason to do this is illustrated by the case of the variance of the return. Dynamic programming algorithms without state-augmentation cannot compute the variance of the return alone: at timestep t, $\mathbb{V}[R_t^\pi]$ cannot be expressed solely in terms of $\mathbb{V}[R_{t+1}^\pi]$ and the reward at time t. However, it can be expressed as a function of both the expectation $\mathbb{E}[R_{t+1}^\pi]$ and the variance $\mathbb{V}[R_{t+1}^\pi]$ (and the reward at time t) [Sobel, 1982]. Indeed, the second moment of the return verifies:

$$\mathbb{E}[(R_t^\pi(x))^2] = \mathbb{E}\left[r(x,A,X')^2\right] + 2\mathbb{E}\left[R_{t+1}^\pi(X')r(x,A,X')\right] + \mathbb{E}\left[R_{t+1}(X')^2\right] , \quad (3.1)$$

which is a function of $\mathbb{E}[R_{t+1}^\pi]$ and $\mathbb{E}[(R_{t+1}^\pi)^2]$. Using the relation $\mathbb{V}[R] = \mathbb{E}[R^2] - \mathbb{E}[R]^2$, we can recursively compute both the expectation and the variance of the return by dynamic programming without needing any other state-augmentation. Such a method has the same complexity as the usual dynamic programming, and thus improves upon the exponential complexity of the distributional approach. Thus, for the variance, it is important to consider the pair of statistics (first and second moment) to perform exact policy evaluation.

We now tackle the characterization of families of statistics that verify this property.

[1] There is a slight distinction between distribution representations (here), and parametrized distributions (as mentioned in Section 2.2.3). Parametrized distributions are true distributions in $\mathcal{P}(\mathbb{R})$, from which one can sample; distribution representations are not.

Bellman Closedness To formalize this idea of recursively computable family of statistics, Rowland et al. [2019] defines the notion of *Bellman closedness*:

Definition 3.1 (Bellman closedness [Rowland et al., 2019])**.** A set of statistics $\{\varphi_1, \dots \varphi_K\}$ is said to be *Bellman closed* if for each $(x, a, t) \in \mathcal{X} \times \mathcal{A} \times [H]$, the statistics $\varphi_{1:K}(\eta_t^\pi(x, a))$ can be expressed in closed form in terms of the random variables $r(x, a, x'), p(x'|x, a)$, and $\varphi_{1:K}(\eta_{t+1}^\pi(x')),\ x' \in \mathcal{X}$, independently of the MDP.

In simpler terms, a set of statistics is Bellman closed if we can derive a Bellman equation and dynamic programming algorithm that computes those statistics of the return for any policy, by computing recursively only those values. For instance, we saw in Section 2.3.4.1 that computing recursively the quantiles may need the entire quantile function (i.e., an infinite number of statistics). Hence, there is no general finite family of statistics including quantiles that are Bellman closed.

The general characterization of Bellman closed families of statistics is still an open problem to this day, but Rowland et al. [2019] provided a partial answer, giving the characterization among the set of expected utilities. Their original work only considers the discounted MDP setting, but their proof can directly be adapted to the undiscounted, finite-horizon setting. For completeness, we provide the adapted statement and proof below.

Theorem 3.1 (Rowland et al. [2019], adapted to undiscounted MDPs)**.** Let $m \in \mathbb{N}$ and $\{\varphi_1, \dots, \varphi_m\}$ be a set of statistics such that each φ_i is an expected utility with characteristic function f_i. If the set $\{\varphi_1, \dots, \varphi_m\}$ is Bellman closed, then there exist $\lambda \in \mathbb{R}, \mathbf{A} \in \mathbb{R}^{m \times m}$ an invertible matrix, and $\mathbf{b} \in \mathbb{R}^m$ such that:

$$\forall i \in [m], \quad f_i(x) = \sum_{j=1}^{m} A_{ij} x^{j-1} e^{\lambda x} + b_i \ .$$

Said differently, the statistics are all affine combinations of the functions $x \mapsto x^i \exp(\lambda x)$ for $i \in [m]$.

Sufficient condition Let $\lambda \in \mathbb{R}$ and $m \in \mathbb{N}$. Consider the expected utility φ_n with characteristic function $f_n(x) = x^n e^{\lambda x}$ for all $n \in [m]$. We start by showing that the set of statistics $\{\varphi_1, \dots, \varphi_m\}$ is Bellman closed.

$$\begin{aligned}
\varphi_n(\eta_t^\pi(x,a)) &= \mathbb{E}\left[\mathcal{R}_t^\pi(x,a)^n e^{\lambda\mathcal{R}_t^\pi(x,a)}\right] \\
&= \mathbb{E}\left[(r(x,a,X') + \mathcal{R}_{t+1}^\pi(X'))^n e^{\lambda(r(x,a,X')+\mathcal{R}_{t+1}^\pi(X'))}\right] \\
&= \mathbb{E}_{X'}\left[\mathbb{E}_{\mathcal{R}_{t+1}}[(r(x,a,X') + \mathcal{R}_{t+1}^\pi(X'))^n e^{\lambda(r(x,a,X')+\mathcal{R}_{t+1}^\pi(X'))} \mid X' = x']\right] \\
&= \sum_{x'} p(x'|x,a)\mathbb{E}[(r(x,a,x') + \mathcal{R}_{t+1}^\pi(x'))^n e^{\lambda(r(x,a,x')+\mathcal{R}_{t+1}^\pi(x'))}] \\
&= \sum_{x'} p(x'|x,a)\mathbb{E}\Big[\sum_{k=0}^{n}\binom{n}{k} r(x,a,x')^{n-k}e^{\lambda r(x,a,x')}\mathcal{R}_{t+1}^\pi(x')^k e^{\lambda\mathcal{R}_{t+1}^\pi(x')}\Big] \\
&= \sum_{x'} p(x'|x,a)\sum_{k=0}^{n}\binom{n}{k} r(x,a,x')^{n-k}e^{\lambda r(x,a,x')}\mathbb{E}\Big[\mathcal{R}_{t+1}^\pi(x')^k e^{\lambda\mathcal{R}_{t+1}^\pi(x')}\Big] \\
&= \sum_{x'} p(x'|x,a)\sum_{k=0}^{n}\binom{n}{k} r(x,a,x')^{n-k}e^{\lambda r(x,a,x')}\varphi_k(\eta_{t+1}^\pi(x')).
\end{aligned}$$

This shows that each φ_n can be expressed as a function of $\varphi_{1:n}$ at the next step, and thus the set $\{\varphi_1, \ldots, \varphi_m\}$ is Bellman closed. The proof follows that of Bellemare et al. [2023].

Necessary condition [Bellemare et al., 2023] Let $\{\varphi_1, \ldots, \varphi_m\}$ be a Bellman closed set of expected utilities with characteristic functions $f_1, \ldots, f_m$. We consider here a simple setting. We have an MDP with a state x that transitions deterministically to state y with associated reward. Let $t < H$ be arbitrary. The return distribution from state y, $\eta_{t+1}(y) = \nu$, is also arbitrary. We have $\eta_t(x) = \delta_r * \eta_{t+1}(y)$, where r is the reward obtained when transitioning from x to y. By assumption on the statistics,

$$\forall i, \quad \varphi_i(\eta_{t+1}(y)) = \mathbb{E}_{Z\sim\nu}[f_i(Z)] \quad \text{and} \quad \varphi_i(\eta_t(x)) = \mathbb{E}_{Z\sim\nu}[f_i(r+Z)]\ .$$

By Bellman closedness, there exist functions $g_i : \mathbb{R}^m \times \mathbb{R} \to \mathbb{R}$ such that:

$$\forall i, \quad \varphi_i(\eta_t(x)) = g_i\left(\Phi(\eta_{t+1}(y)), r\right)$$

where $\Phi(\eta_{t+1}(y)) = (\varphi_1(\eta_{t+1}(y)), \ldots, \varphi_m(\eta_{t+1}(y)))$, for any distribution ν.

The proof uses two arguments. (1) using the expected utility assumption, we show that the functions g_i must be affine in their first argument. (2) The affine property implies that for any $i \in [m]$, $f_i(r + \cdot)$ must be an affine combination of all $f_j(\cdot)$, $j \in [m]$. This leads to a functional equation on the f_i's, whose solutions are known to be of the form stated in the theorem.

1. **Affine property of g_i:** In vector spaces, a function g is affine if and only if for any $\alpha \in \mathbb{R}$ and any $\mathbf{u}, \mathbf{v} \in \mathbb{R}^m$, $g(\alpha\mathbf{u} + (1-\alpha)\mathbf{v}) = \alpha g(\mathbf{u}) + (1-\alpha)g(\mathbf{v})$ (the

function preserves barycenters). Let $u, v \in \mathbb{R}^m$ such that there exists distributions ν_1, ν_2 with $u = \Phi(\nu_1)$ and $v = \Phi(\nu_2)$. Let $\alpha \in [0, 1]$ and consider the distribution $\nu_\alpha = \alpha\nu_1 + (1-\alpha)\nu_2$. By linearity of the expectation in the distribution:

$$\begin{aligned} g_i\left(\alpha\mathbf{u} + (1-\alpha)\mathbf{v}, r\right) &= \mathbb{E}_{Z\sim\nu_\alpha}[f_i(r+Z)] \\ &= \alpha\mathbb{E}_{Z\sim\nu_1}[f_i(r+Z)] + (1-\alpha)\mathbb{E}_{Z\sim\nu_2}[f_i(r+Z)] \\ &= \alpha g_i(\mathbf{u}, r) + (1-\alpha)g_i(\mathbf{v}, r)\ . \end{aligned}$$

Hence, g_i is affine in its first argument for any $i \in [m]$.

2. **Characterizing the g_is**: The fact that g_i is affine in $\mathbf{u}$ means there exists $b_{i0}(r), \dots, b_{im}(r)$ such that:

$$g_i(u, r) = b_{i0}(r) + \sum_{k=1}^{m} b_{ik}(r)u_k$$

For $z \in \mathbb{R}$, consider now the distribution $\nu = \delta_z$. Then, $\varphi_j(\nu) = f_j(z)$ for all $j \in [m]$. Thus,

$$\begin{aligned} f_i(r+z) &= \varphi_i(\eta_t(x)) \\ &= g_i\left(\Phi(\eta_{t+1}(y)), r\right) \\ &= b_{i0}(r) + \sum_{k=1}^{m} b_{ik}(r) f_k(z)\ . \end{aligned}$$

Hence, for any $r \in \mathbb{R}$. $f_i(r+\cdot)$ is an affine combination of the f_k's. The only finite-dimensional, translation-invariant function vector spaces have been studied by Engert [1970], and can only be of the form stated in the theorem.

The previous result applies to expected utilities only. However, we can extend it to other family of statistics obtained by invertible transformations of a Bellman closed set.

Proposition 3.1. Let $f : \mathbb{R}^m \to \mathbb{R}^m$ be invertible and $\varphi'_i = f_i(\varphi_1, \dots, \varphi_m)$ for all $i \in [m]$. If $\{\varphi_1, \dots, \varphi_m\}$ is Bellman closed, then $\{\varphi'_1, \dots, \varphi'_m\}$ is also Bellman closed.

In particular, any invertible function of the functions $x \mapsto x^i \exp(\lambda x)$ for $i \in [m]$ form a Bellman closed set of statistics. This applies for example for the EntRM, which is the exponential utility ($x \to \exp(\lambda x)$) composed with the invertible function $x \mapsto \frac{1}{\lambda}\log(x)$.

Idea of the proof Consider $\{\varphi'_1, \dots, \varphi'_m\}$ such that there exists $f : \mathbb{R}^m \to \mathbb{R}^m$ invertible with $\varphi'_i = f_i(\varphi_1, \dots, \varphi_m)$ for all $i \in [m]$. Then, since $\{\varphi_1, \dots, \varphi_m\}$ is Bellman closed, they verify $\Phi(\eta_t) = g(\Phi(\eta_{t+1}))$ for some function g. We can express each φ'_i as a function of $\varphi'_{1:m}$ at the next step by inverting f and applying the Bellman equations of $\varphi_{1:m}$: $\Phi'(\eta_t) = f(g(f^{-1}(\Phi'(\eta_{t+1}))))$. Hence, $\{\varphi'_1, \dots, \varphi'_m\}$ is also Bellman closed.

The last result considers non-invertible transformations of Bellman closed statistics. Intuitively, any function of Bellman closed statistics should also be Bellman closed by just applying the Bellman equations of the original statistics and then applying the function. Formally:

Proposition 3.2. Let $\{\varphi_1, \dots, \varphi_m\}$ be a Bellman closed set of statistics and $f : \mathbb{R}^m \to \mathbb{R}$ be any function. Consider the statistic $\varphi' = f(\varphi_1, \dots, \varphi_m)$. Then $\{\varphi_1, \dots, \varphi_m, \varphi'\}$ is also Bellman closed.

This can apply for instance for $X \to \mathbb{E}[X]^2$, which is not Bellman closed by itself, because of the non-injectivity of the function $x \mapsto x^2$. Yet, when combined with the expectation, it becomes Bellman closed.

Idea of the proof The idea is similar to the previous proof. Since $\{\varphi_1, \dots, \varphi_m\}$ is Bellman closed, we have $\Phi(\eta_t) = g(\Phi(\eta_{t+1}))$ for some function g. Then, we can express φ' at time t as a function of $\varphi_{1:m}$ at time $t+1$ by composing f and g: $\varphi'(\eta_t) = f(g(\Phi(\eta_{t+1})))$. Hence, the set $\{\varphi_1, \dots, \varphi_m, \varphi'\}$ is also Bellman closed.

Other Bellman Closed statistics Unfortunately, the previous result only provides a partial characterization of Bellman closed statistics. There are known Bellman closed statistics that cannot be expressed from the exp-moment functions, for instance $\operatorname{ess\,sup} : \nu \mapsto \sup\{x : \nu(x) > 0\}$, or $\operatorname{ess\,inf}$. They are Bellman closed because of the following relations:

$$\operatorname{ess\,sup}(\eta_t^\pi(x,a)) = \sup_{x' \text{ s.t. } p(x,a,x')>0} \left[r(x,a,x') + \operatorname{ess\,sup}(\eta_{t+1}^\pi(x'))\right] \tag{3.2}$$

$$\operatorname{ess\,inf}(\eta_t^\pi(x,a)) = \inf_{x' \text{ s.t. } p(x,a,x')>0} \left[r(x,a,x') + \operatorname{ess\,inf}(\eta_{t+1}^\pi(x'))\right] \ . \tag{3.3}$$

By applying the proposition on non-invertible functions to $\operatorname{ess\,inf}$, and considering the function $\mathrm{pos} : \nu \mapsto \mathbb{1}(\nu([0,\infty)) = 1)$, the set $\{\operatorname{ess\,inf}, \mathrm{pos}\}$ is also Bellman closed. Indeed, $\mathrm{pos} = \mathbb{1}(\operatorname{ess\,inf} \geq 0)$.[2]

In general, the full characterization of Bellman closed statistics remains an open problem to this day.

[2] We mention this statistic because Pires et al. [2025] claimed it was a special case of statistic only computable and optimizable with distributional dynamic programming. We show here that the claim is incorrect.

3.1.2 Approximate Evaluation with General Parametrized Distributions

Some important risk measures such as the CVaR or the quantiles are not known to belong to any Bellman-closed set and hence cannot be exactly computed without an exponential complexity. However, the distributional framework offers natural ways to approximate the distribution of the returns, as mentioned in Section 2.2.3. In this section, we study how well those approximations of distributions translate into approximations of the risk measures of interest. The case of quantiles was already studied by Rowland et al. [2019]. Here we consider a large class of statistics, the W_1-Lipschitz statistics. We focus on the quantile parametrization as it provides better properties and guarantees than the categorical one [Dabney et al., 2018b]. The case of the categorical parametrization is discussed at the end of the section.

Bounding the change in distribution In Algorithm 5, the projection is applied at each step, each time deviating further from the true distribution. It is then important to understand how much each step loses. We measure this loss as the Wasserstein distance between the exact and approximate distributions. We start by proving a bound on this distance.

Proposition 3.3. Let π be a policy and η^π the associated Q-value distributions. Let $\hat{\eta}^\pi$ be the Q-value distributions obtained by dynamic programming (Algorithm 5) using the quantile projection Π_{qr} with resolution N. Then,

$$\sup_{(x,a,t)\in\mathcal{X}\times\mathcal{A}\times[H]} W_1(\hat{\eta}_t^\pi(x,a), \eta_t^\pi(x,a)) \leq \frac{H^2}{2N} .$$

This result shows that the loss of information due to the parametrization may only grow quadratically with the horizon. The proof consists of summing the projection bound in Proposition 2.12 at each projection step, and using the non-expansion property of the Bellman operator [Bellemare et al., 2017]. The details can be found in Section 3.1.3. The quadratic dependence in H is verified experimentally (see Figure 3.2). Note that this bound is only relevant when $N > H/4$ as $\mathrm{supp}(\eta) \subseteq [-H, H]$ and the Wasserstein distance is always upper bounded by the size of the support (here $2H$).

Lipschitz statistics The key question is then to understand how such error translates into our estimation problem when we apply the function of interest to the approximate distribution. To this end, we consider a general class of statistics, the W_1-Lipschitz statistics, i.e., statistics φ for which there exists $L > 0$ such that for any distributions ν_1, ν_2:

$$|\varphi(\nu_1) - \varphi(\nu_2)| \leq L W_1(\nu_1, \nu_2) . \tag{3.4}$$

Conveniently, many statistics of interest fall into this category, starting with the expected utility family with Lipschitz expected utility function f.

Proposition 3.4. Let φ be a statistic of the form $\varphi = \mathbb{E}[f(\cdot)]$ where f is L-Lipschitz on its domain. Then, φ is W_1-Lipschitz with Lipschitz constant L.

The proof is a direct application of the Kantorovich-Rubinstein duality [Villani, 2003]. This includes in particular any expected utility with continuous f as a continuous function on a closed interval is always Lipschitz. For instance, the exponential utility is $e^{|\beta|H}$-Lipschitz on $[-H, H]$.

Another important family of W_1-Lipschitz statistics are the *distorted means*:

Example 3.1 (Distorted means)**.** Let $\beta : [0,1] \to [0,1]$ be a continuous non-decreasing function such that $\beta(0) = 0$ and $\beta(1) = 1$. The *distorted mean* with distortion function β is defined for any distribution ν as:

$$\varphi_\beta(\nu) = \int_0^1 F_\nu^{-1}(p)\beta'(p)dp \ ,$$

where F_ν^{-1} is the quantile function of ν.

The idea is to put different weights on the quantiles of the distribution. For instance, when β is the identity function, the distorted mean is the expectation. Importantly, for $\beta(p) = \min(1, p/\alpha)$, the distorted mean corresponds to the CVaR_α. Indeed, this corresponds to computing the mean of the distribution with weight $1/\alpha$ on the lower α-tail and weight 0 elsewhere (illustrated in Figure 2.14). Distorted means will not be studied further in this thesis, but they form a quite important class of risk measures. Similar to Expected Utilities, they verify their own axioms of *rationality* [Yaari, 1987] and can be optimized in MDPs with stock-augmented policies [Bastani et al., 2022].

Proposition 3.5. Let φ_β be a distorted mean with distortion function β. Then, φ_β is W_1-Lipschitz with Lipschitz constant $L = \sup_{p\in[0,1]}|\beta'(p)|$.

When $\beta(p) = \min(1, p/\alpha)$, we have $\beta'(p) = \dfrac{1}{\alpha}\mathbb{1}\{p < \alpha\}$. In particular, the CVaR_α is $1/\alpha$-Lipschitz.

Bounding the estimation error Combining the previous results, we can now bound the estimation error when evaluating such Lipschitz statistics with the approximate distribution obtained by Algorithm 5. This property allows us to prove a maximal upper bound on the estimation error for those two families.

Theorem 3.2. Let π be a Markov policy. Let η^π be the Q-value return distribution associated with π. Let $\hat{\eta}^\pi$ be the approximate return distribution computed with

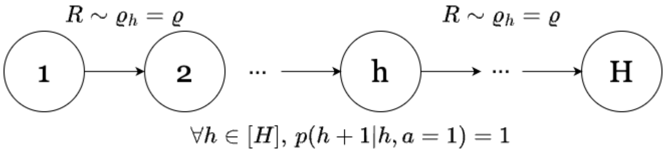
$R \sim \varrho_h = \varrho$
$R \sim \varrho_h = \varrho$
1
2
...
h
...
H
$\forall h \in [H],\, p(h+1|h, a=1) = 1$

Figure 3.1: A Chain MDP of length H with deterministic transition and identical reward distribution for each state.

Algorithm 5, for the projection Π_{qr} with resolution N. Let φ be a Lipschitz statistic with Lipschitz constant L. Then:

$$\sup_{x,a,t} |\varphi(\hat{\eta}_t^\pi(x,a)) - \varphi(\eta_t^\pi(x,a))| \leq \frac{LH^2}{2N} \ .$$

Note that depending on the choice of statistic, the Lipschitz coefficient L may also depend on H. For instance, the Lipschitz constant of the exponential utility depends exponentially on H. For the CVaR_α, however, L is constant and only depends on $\alpha \in (0,1)$.

Experiment: empirical validation of the bounds on a simple MDP We consider a simple Chain MDP environment of length $H = 70$ equal to the horizon (see Figure 3.1) [Rowland et al., 2019], with a single action leading to the same discrete reward distribution for every step. We consider a Bernoulli reward distribution $\mathcal{B}(0.5)$ for each state so that the number of atoms for the return only grows linearly[3] with the number of steps, which allows us to compute the exact distribution easily.

We compare the distributions obtained with exact dynamic programming and the approximate distribution obtained by Algorithm 5, with a quantile projection with resolution $N = 1000$. Note that even at early stages, when the true distribution has fewer atoms than the resolution, the exact and approximate distributions differ due to the weights of the atoms in the quantile projection. Figure 3.2 (Right) reports the Wasserstein distance between the two distributions: the cumulative projection approximation error (dashed blue), the true error between the current exact and approximate distributions (solid blue) and the theoretical bound (red). Fundamentally, the proof of Prop. 3.3 upper bounds the distance between distributions by the cumulative projection error so we plot this quantity to help validate it.

We also empirically validate Theorem 3.2 by computing the CVaR_α for $\alpha \in \{0.1, 0.25\}$, corresponding respectively to distorted means with Lipschitz constants $L = \{10, 4\}$. We compute these statistics for both distributions and report the maximal error together with the theoretical bound, re-scaled[4] by a factor 2. Figure 3.2 (Left)

[3] At round $t \in [H]$, the support of the return is $\{0, 1, ..., t\}$, hence t atoms.

[4] Scaling by a constant factor allows us to show the corresponding quadratic trends better.

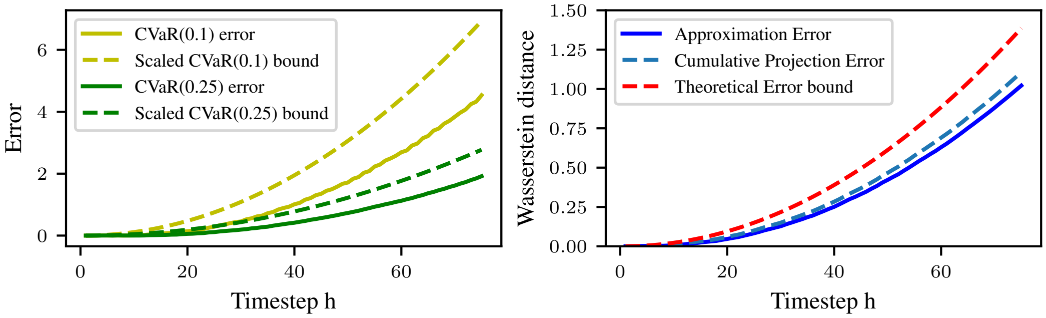

Figure 3.2: Left: Validation of Theorem 3.2 on CVaR_α together with the scaled upper bound (see main text for discussion): the quadratic dependence in H is verified. Right: Validation of Proposition 3.3: The cumulative projection error (dashed blue) is the sum of the projection errors at every timestep, and matches the true approximation error (solid blue). The theoretical upper bound (dashed red) also matches closely the true error.

shows an impressive correspondence of the theory and the empirical results despite a constant multiplicative gap.

About the Categorical Approximation While the results above mention only the quantile parametrization, similar results can easily be obtained for the categorical parametrization. Indeed, the proof of Proposition 3.3 relies on the triangle inequality, on the non-expansion of the Bellman operator, and on the projection error bound. Using the Cramér distance, the projection error for the categorical projection is similar (see Proposition 2.10) [Rowland et al., 2018]. Furthermore, the Bellman operator is also non-expansive for the Cramér distance [Rowland et al., 2018], and the Cramér distance trivially verifies the triangle inequality. Hence, Proposition 3.3 can be adapted to:

$$\sup_{(x,a,t)\in\mathcal{X}\times\mathcal{A}\times[H]} \ell_2(\hat{\eta}_t^\pi(x,a), \eta_t^\pi(x,a)) \leq \frac{2H^2}{N} .$$

Moreover, it is possible to bound the Wasserstein distance with the Cramér distance, using the Cauchy-Schwarz inequality. For two distributions ν_1, ν_2 supported on an interval of length Δ, $W_1(\nu_1,\nu_2) \leq \sqrt{\Delta}\ell_2(\nu_1,\nu_2)$ (see the proof in Section 3.1.3). This implies that W_1-Lipschitz statistics are also ℓ_2-Lipschitz. If the statistic φ is L-Lipschitz for the W_1 distance, then it is also $\sqrt{2H}L$-Lipschitz for the ℓ_2 distance. Combining those results, we obtain a similar bound as in Theorem 3.2 for the categorical parametrization:

$$\sup_{x,a,t}|\varphi(\hat{\eta}_t^\pi(x,a)) - \varphi(\eta_t^\pi(x,a))| \leq \frac{\sqrt{2}LH^{5/2}}{N} .$$

3.1.3 Proofs

Proof of Proposition 3.3 We recall the statement of Proposition 3.3: Let π be a policy and η^π the associated Q-value distributions. Assume the reward is bounded on an interval of length Δ_R ($\Delta_R = 2$ here as the rewards are in $[-1, 1]$). Let $\hat{\eta}^\pi$ be the Q-value distributions obtained by dynamic programming (Algorithm 5) using the quantile projection Π_{qr} with resolution N. Then,

$$\forall t, \qquad \sup_{(x,a)\in\mathcal{X}\times\mathcal{A}} W_1(\hat{\eta}^\pi_t(x,a), \eta^\pi_t(x,a)) \leq H^2 \frac{\Delta_R}{2N} \, .$$

To simplify the notation, we denote $\overline{W}_1(\hat{\eta}, \eta) := \sup_{(x,a)\in\mathcal{X}\times\mathcal{A}} W_1(\hat{\eta}(x,a), \eta(x,a))$.

Proof. First recall that for any Q-value distribution $(\eta_t)_{t\in[H]}$, with the return bounded on an interval of length $\Delta_{\eta_t} \leq 2(H-t)\Delta_R$, and Π_{qr} the projection operator of interest with resolution N, we have the following bound on the projection estimation error due to Rowland et al. [2019] (Proposition 2.12):

$$\overline{W}_1(\Pi_{\mathrm{qr}}\eta_t, \eta_t) \leq \frac{\Delta_{\eta_t}}{2N} \leq (H-t)\frac{\Delta_R}{N} \, . \tag{3.5}$$

At a fixed step $t \in [H]$, we have the following inequality:

$$\begin{aligned} \overline{W}_1(\hat{\eta}^\pi_t, \eta^\pi_t) &= \overline{W}_1(\Pi_{\mathrm{qr}}\mathcal{T}^\pi\hat{\eta}^\pi_{t+1}, \mathcal{T}^\pi\eta^\pi_{t+1}) \\ &\leq \overline{W}_1(\Pi_{\mathrm{qr}}\mathcal{T}^\pi\hat{\eta}^\pi_{t+1}, \mathcal{T}^\pi\hat{\eta}^\pi_{t+1}) + \overline{W}_1(\mathcal{T}^\pi\hat{\eta}^\pi_{t+1}, \mathcal{T}^\pi\eta^\pi_{t+1}) \qquad (3.6) \\ &\leq (H-t)\frac{\Delta_R}{N} + \overline{W}_1(\hat{\eta}^\pi_{t+1}, \eta^\pi_{t+1}) \, . \qquad (3.7) \end{aligned}$$

Where $\mathcal{T}^\pi$ is the Bellman operator associated to the distributional Bellman equation (Proposition 2.6). (3.6) is due to the triangle inequality with $\mathcal{T}^\pi\hat{\eta}^\pi_{t+1}$ as the middle term. In (3.7), the first term comes from applying Equation (3.5) to the first term of the previous line. The second term is a consequence of the non-expansive property of the Bellman operator [Bellemare et al., 2017]:

$$\overline{W}_1(\mathcal{T}\eta_1, \mathcal{T}\eta_2) \leq \overline{W}_1(\eta_1, \eta_2) \, .$$

Using it recursively starting from $t = 0$, and using the fact that $\hat{\eta}^\pi_H = \eta^\pi_H$ we get:

$$\overline{W}_1(\hat{\eta}^\pi_0, \eta^\pi_0) \leq H\frac{\Delta_R}{N} + \overline{W}_1(\hat{\eta}^\pi_1, \eta^\pi_1) \leq (H + H - 1)\frac{\Delta_R}{N} + \overline{W}_1(\hat{\eta}^\pi_2, \eta^\pi_2) \leq \ldots \leq H^2\frac{\Delta_R}{2N} \, .$$

□

Proof of Proposition 3.4 We recall the statement: Let φ be an expected utility associated to f and let L_f be its Lipschitz coefficient. Let ν_1, ν_2 be return distributions. Then:

$$|\varphi(\nu_1) - \varphi(\nu_2)| \leq L W_1(\nu_1, \nu_2)\ .$$

Proof. The Kantorovich-Rubinstein duality [Villani, 2003] states that:

$$W_1(\nu_1, \nu_2) = \frac{1}{L_f} \sup_{||g||_L \leq L_f} \left(\int g \ \mathrm{d}\nu_1 - \int g \ \mathrm{d}\nu_2 \right) , \tag{3.8}$$

where $||\cdot||_L$ is the Lipschitz norm. We then immediately get:

$$L_f W_1(\nu_1, \nu_2) \geq \left| \int f \ \mathrm{d}\nu_1 - \int f \ \mathrm{d}\nu_2 \right| = |\varphi(\nu_1) - \varphi(\nu_2)|\ . \tag{3.9}$$

□

Proof of Proposition 3.5 The statement is the following: Let φ be a distorted mean associated to weight function β, that is Lipschitz with constant L_β. Let ν_1, ν_2 be return distributions. Then:

$$|\varphi(\nu_1) - \varphi(\nu_2)| \leq L_\beta W_1(\nu_1, \nu_2)\ .$$

Proof. φ is a distorted mean. There exists β such that $\varphi(\nu) = \int_0^1 \beta'(\tau) F_\nu^{-1}(\tau) \mathrm{d}\tau$. Let L_β be its Lipschitz coefficient. Thus:

$$\begin{aligned} |\varphi(\nu_1) - \varphi(\nu_2)| &= \left| \int_0^1 \beta'(\tau) \left(F_{\nu_1}^{-1} - F_{\nu_2}^{-1}(\tau) \right) \mathrm{d}\tau \right| \\ &\leq ||\beta'||_\infty \int_0^1 \left| F_{\nu_1}^{-1}(\tau) - F_{\nu_2}^{-1}(\tau) \right| \mathrm{d}\tau \\ &\leq L_\beta W_1(\nu_1, \nu_2)\ . \end{aligned}$$

□

Proof of Theorem 3.2 The theorem states that for the projection Π_{qr} with resolution N, and any Lipschitz statistic φ with Lipschitz constant L, the difference between the computed statistic with the approximate distribution from Algorithm 5 and the exact distribution is bounded as:

$$\sup_{x,a,t} |\varphi(\hat{\eta}_t^\pi(x,a)) - \varphi(\eta_t^\pi(x,a))| \leq \frac{LH^2}{2N}\ .$$

Proof. We just need to combine Proposition 3.3 and the definition of Lipschitz statistics (Eq. 3.4):

$$\begin{aligned} \sup_{x,a,t} |\varphi(\hat{\eta}_t^\pi(x,a)) - \varphi(\eta_t^\pi(x,a))| &\leq L \sup_{x,a,t} W_1(\hat{\eta}_t^\pi(x,a), \eta_t^\pi(x,a)) \\ &\leq L \frac{H^2}{2N}\ . \end{aligned}$$

□

Proof for the Categorical Projection The adaptation of Proposition 3.3 is straightforward by replacing Equation (3.5) with the projection error bound for the categorical projection from Proposition 2.10, and the W_1 distance by the ℓ_2 distance. We here only prove the bound between the Wasserstein and Cramér distance:

Proposition 3.6. Let ν_1, ν_2 be two distributions supported on an interval of length Δ. Then:

$$W_1(\nu_1, \nu_2) \leq \sqrt{\Delta}\ell_2(\nu_1, \nu_2) \ .$$

Proof. For this proof, we assume the distributions have support on $[-H, H]$, to simplify. We have:

$$\begin{aligned} W_1(\nu_1, \nu_2) &= \int_{-H}^{H} |F_{\nu_1}(x) - F_{\nu_2}(x)| \mathrm{d}x \\ &\leq \sqrt{\int_{-H}^{H} 1} \cdot \sqrt{\int_{-H}^{H} |F_{\nu_1}(x) - F_{\nu_2}(x)|^2 \mathrm{d}x} \quad \text{(Cauchy-Schwarz inequality)} \\ &= \sqrt{2H}\ell_2(\nu_1, \nu_2) \ . \end{aligned}$$

□

3.1.4 About the tightness of Theorem 3.2

In this section, we discuss the tightness of the bound provided by Theorem 3.2. To provide simpler examples, we will here drop the assumption on the stationarity of the MDP and on the reward being deterministic. This means that both the reward and the transition functions may depend on the timestep, and that the reward may be stochastic ($r_t(x, a) \sim \rho_{x,a}$). The upper bound provided by Theorem 3.2 is mainly based on Proposition 3.3. The latter is obtained by summing, for every step, the projection bound by Rowland et al. [2019] (Proposition 2.10). Thus, achieving the bound would first require to find a problem instance for which, at every step, the projection bound is tight. Then it would require to verify that the total error is the sum of those projection errors.

The experiment in Figure 3.2 already shows that summing the total error is very close to the sum of the projection errors. However, in that example, the projection error bound is not reached after the first step. In the following, we exhibit an MDP for which the projection bound is tight at every timestep.

First, let us consider a family of distributions for which the projection error is tight:

Proposition 3.7. Let $N \in \mathbb{N},\ \Delta \in \mathbb{R}^+$. Consider $z_i = \frac{i\Delta}{N}$. The distribution

$$\nu_{N,\Delta} = \frac{1}{2N} \sum_{i=0}^{N-1} (\delta_{z_i} + \delta_{z_{i+1}})$$

has a support of length Δ and verifies $W_1(\nu_{N,\Delta}, \Pi_{\mathrm{qr}}\nu_{N,\Delta}) = \frac{\Delta}{2N}$.

Proof. The cumulative distribution function (CDF) of the distribution $\nu_{N,\Delta}$ is

$$F_{\nu_{N,\Delta}}(x) = \begin{cases} 0 & x < 0 = z_0 \ , \\ \frac{2i+1}{2N} = \tau_i & z_i \le x < z_{i+1} \ , \\ 1 & x \ge \Delta = z_N \ . \end{cases}$$

We write $\tau_i = \frac{2i+1}{2N}$, so that $\forall i \in [N-1], F_{\nu_{N,\Delta}}(z_i) = \tau_i$. We now explicitly describe the projection of $\nu_{N,\Delta}$ and the Wasserstein distance relative to it. A possible quantile projection is $\Pi_{\mathrm{qr}}\nu_{N,\Delta} = \frac{1}{N}\sum_{i=0}^{N-1} \delta_{z_i}$, and

$$\begin{aligned} W_1(\nu_{N,\Delta}, \Pi_{\mathrm{qr}}\nu_{N,\Delta}) &= \int_0^1 |F_\nu^{-1}(w) - F_{\Pi_{\mathrm{qr}}\nu}^{-1}(w)| \mathrm{d}w \\ \\ &= \sum_{i=0}^{N-1} \int_{\frac{i}{N}}^{\frac{i+1}{N}} |F_\nu^{-1}(w) - \underbrace{F_{\Pi_{\mathrm{qr}}\nu}^{-1}(w)}_{z_i}| \mathrm{d}w \\ &= \sum_{i=0}^{N-1} \int_{\frac{i}{N}}^{\tau_i} |\underbrace{F_\nu^{-1}(w)}_{z_i} - z_i| \mathrm{d}w + \int_{\tau_i}^{\frac{i+1}{N}} |\underbrace{F_\nu^{-1}(w)}_{z_{i+1}} - z_i| \mathrm{d}w \\ &= \sum_{i=0}^{N-1} \frac{1}{2N} \underbrace{(z_{i+1} - z_i)}_{\frac{\Delta}{N}} \\ &= \sum_{i=0}^{N-1} \frac{\Delta}{2N^2} \\ &= \frac{\Delta}{2N} \end{aligned}$$

□

The value distributions of the following timesteps are obtained by applying two operators: the Bellman operator and the projection operator. Here we will consider a MDP with only one state. We need to find such operators so that $\Pi_{\mathrm{qr}}\mathcal{T}\nu_{N_1\Delta_1} = \nu_{N_2\Delta_2}$, where the Bellman operator simply consists of the added reward distribution.

Proposition 3.8. Let $N \in \mathbb{N},\ \Delta \in \mathbb{R}^+$. Consider $\varrho = \frac{1}{2}(\delta_0 + \delta_{\frac{\Delta}{N-1}})$. Then there exists a quantile projection operator Π_{qr} such that

$$\varrho * \Pi_{\mathrm{qr}}(\nu_{N,\Delta}) = \nu_{N,(\Delta + \frac{\Delta}{N-1})}$$

Proof. We consider $\Pi_{\text{qr}}(\nu_{N,\Delta}) = \frac{1}{N}\sum_{i=0}^{N-1}\delta_{\frac{i\Delta}{N-1}}$. Since $\forall i$, $\frac{i\Delta}{N-1} \leq z_i < \frac{(i+1)\Delta}{N-1}$, we have $F(z_i) = \tau_i$, verifying that it is indeed a valid quantile projection.

Hence,

$$\begin{aligned}
\varrho * \Pi_{\text{qr}}(\nu_{N,\Delta}) &= \frac{1}{2}(\delta_0 + \delta_{\frac{\Delta}{N-1}}) * \left(\frac{1}{N}\sum_{i=0}^{N-1}\delta_{\frac{i\Delta}{N-1}}\right) \\
&= \frac{1}{N}\sum_{i=0}^{N-1}\left(\frac{1}{2}(\delta_0 + \delta_{\frac{\Delta}{N-1}}) * \delta_{\frac{i\Delta}{N-1}}\right) \\
&= \frac{1}{2N}\sum_{i=0}^{N-1}\left(\delta_{\frac{i\Delta}{N-1}} + \delta_{\frac{(i+1)\Delta}{N-1}}\right) \\
&= \nu_{N,(\Delta+\frac{\Delta}{N-1})}
\end{aligned}$$

The last equality comes down to noticing that $\frac{i\Delta}{N-1} = \frac{i}{N}(\Delta + \frac{\Delta}{N-1})$. □

Here, we take advantage of the fact that the quantile projection is not unique (see discussion in Section 2.2.3). By choosing the adapted projection, it is then possible to obtain one of those distributions at every timestep.

Corollary 3.3. Let $N \in \mathbb{N}$, $\Delta_0 \in \mathbb{R}^+$. Consider the sequence $\Delta_{t+1} = \Delta_t(1 + \frac{1}{N-1})$ and $\varrho_t = \frac{1}{2}(\delta_0 + \delta_{\frac{\Delta_t}{N-1}})$. Consider Π_{qr} as in Prop. 3.8. Consider the MDP with only one state x and action a, reward distribution ϱ_t, horizon H. Consider $\hat{\eta}_t$ the value distributions obtained through dynamic programming with quantile projection. Then, at any timestep t, the return has support on $[0, \Delta_t]$ and the error induced by the projection operator matches the bound in Proposition 3.3:

$$W_1(\Pi_{\text{qr}}\eta_t, \eta_t) = \frac{\Delta_t}{2N}$$

While the projection error is maximal in such instance, it was verified experimentally that the bound in Prop. 3.3 was still not tight. This comes from the fact that the Bellman operator is not a non-contraction in this case, and that the triangle inequality used to sum the projection error is not tight either.

We hence found that every inequality used in proving Theorem 3.2 can be tight, but there does not seem to exist an instance for which all the inequalities are tight at the same time, meaning that this bound would never be reached exactly.

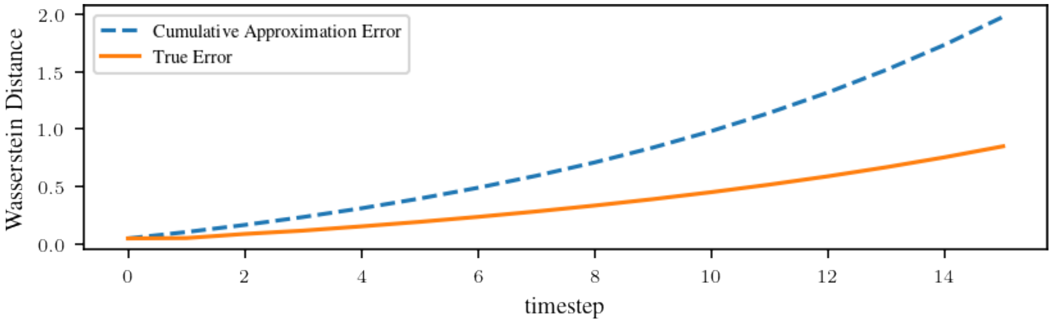

Figure 3.3: Evaluation of the Wasserstein Distance between the true value distribution and the approximate one, in the MDP described in Corollary 3.3

.

3.2 Policy Optimization: (Distributional) Dynamic Programming

The previous section studied the policy evaluation problem for general statistics of the return. We now turn to the policy optimization problem: finding an optimal policy $\pi^\star \in \arg\max_{\pi\in\Pi} \varphi(\eta_0^\pi(x_0))$ for a given statistic φ.

3.2.1 Non-Distributional Policy Optimization

In MDPs, even though the policy evaluation and policy optimization problems are fundamentally distinct problems, they are solved in a similar fashion, using Bellman equations. The recipe is as follows: we define the value functions and show that they verify a recursive formula (the Bellman equation). For the policy evaluation problem, the equations are of the form:

$$Q_t^\pi(x,a) = f(r(x,a,\cdot), V_{t+1}^\pi) \ , \tag{3.10}$$

$$V_t^\pi(x) = Q_t^\pi(x, \pi_t(x)) \ . \tag{3.11}$$

For the policy optimization problem, the equations become:

$$Q_t^*(x,a) = f(r(x,a,\cdot), V_{t+1}^*) \ , \tag{3.12}$$

$$V_t^*(x) = \max_a Q_t^*(x,a) \ . \tag{3.13}$$

The only difference lies in how the value function at time t is computed from the Q-function: in the evaluation problem, the action is selected according to the policy π, while in the optimization problem, the action is selected to maximize the value. This can be seen as the policy optimization problem being the evaluation of the optimal policy, when we know that this policy is greedy with respect to the Q-function. This form of equation is also verified for the Entropic Risk Measure (see Proposition 2.13), but also for expected utilities and quantiles (see Section 2.3.3.2 and Section 2.3.4.1) up to an augmentation of the state variable x (in both cases, the optimization is still of the form $V_t^*(x,c) = \max_a Q_t^*(x,c,a)$).

The takeaway from this is that the policy optimization problem is solved similarly to a policy evaluation problem. Thus, if a risk measure cannot be evaluated through a Bellman equation, it won't be optimized either. Hence we can use the results from the previous sections on Bellman Closedness: if a risk measure is not Bellman closed, then it cannot be optimized through dynamic programming. By "Bellman closed" here, we mean that the set $\{\varphi\}$ is Bellman closed.

The reason we consider singleton sets here, instead of sets of arbitrary size like in the previous section, is the very reason why the second moment $\mathbb{E}[\cdot^2]$ is in a Bellman closed set but cannot be optimized by dynamic programming. The recursive equation to compute the second moment depends on both the first and second moments: $\mathbb{E}[\mathcal{R}_t^2] = f\left(\mathbb{E}[\mathcal{R}_{t+1}], \mathbb{E}[\mathcal{R}_{t+1}^2]\right)$, where f is increasing in both arguments. Yet, both the first and second moments may not be optimal at time $t+1$ for the same policy. Hence $\max_\pi \mathbb{E}[(\mathcal{R}_t^\pi)^2] \leq f\left(\max_\pi \mathbb{E}[\mathcal{R}_{t+1}^\pi], \max_\pi \mathbb{E}[(\mathcal{R}_{t+1}^\pi)^2]\right)$ and in general the equality may not hold. Optimizing two different risk measures at the same time may not be possible, hence it is not clear how to use Bellman closedness of sets of size greater than one to derive policy optimization equations.

In particular, we know from Theorem 3.1 that, among expected utilities, only the expectation and the family of exponential utilities are Bellman closed (singleton sets). Thus, only these expected utilities can verify an optimal Bellman equation. However, as in policy evaluation, there can be other risk measures that can still verify an optimal Bellman equation, such as the functional ess sup: if we define

$$Q_t^*(x,a) = \max_\pi \operatorname{ess\,sup}(\eta_t^\pi(x,a)) \ , \tag{3.14}$$

$$V_t^*(x) = \max_\pi \operatorname{ess\,sup}(\eta_t^\pi(x)) \ , \tag{3.15}$$

then we have the optimal Bellman equation:

$$Q_t^*(x,a) = \max_{x' \in \mathcal{X} \text{ s.t. } p(x,a,x')>0} r(x,a,x') + V_{t+1}^*(x') \tag{3.16}$$

$$V_t^*(x) = \max_a Q_t^*(x,a) \ . \tag{3.17}$$

We indeed find in practice the same risk measures that can be evaluated through Bellman equations without state augmentation.

3.2.2 Distributional Policy Optimization

Without distributions, policy evaluation and policy optimization seem to only be solvable by regular dynamic programming for the same risk measures. We also saw that the distributional framework gave a general way to evaluate any risk measure, at the cost of the complexity of computing the distributions. Can we obtain such a result for the policy optimization problem? We recall here the distributional policy optimization algorithm (see Algorithm 4) from Section 2.2.2.

(Algorithm 4) Distributional Policy Optimization

```
Require: MDP $(\mathcal{X}, \mathcal{A}, p, r, H)$, scalar functional $\varphi : \mathcal{P}(\mathbb{R}) \to \mathbb{R}$
Ensure: Markov Policy $\pi = (\pi_0, \dots, \pi_{H-1})$ and distributions $\eta_t(x)$ for all $x \in \mathcal{X}, t < H$
Initialize $\eta_H(x) \leftarrow \delta_0$ for all $x \in \mathcal{X}$        // zero return at final step
for $t = H - 1 \to 0$ do
    for all $x \in \mathcal{X}$ do
        for all $a \in \mathcal{A}$ do
            $\eta_t(x, a) \leftarrow \sum_{x' \in \mathcal{X}} p(x' \mid x, a) \left(\tau_{r(x,a,x')} \eta_{t+1}(x')\right)$
        end for
        $\pi_t(x) \in \arg\max_{a \in \mathcal{A}} \varphi\left(\eta_t(x, a)\right)$
        $\eta_t(x) \leftarrow \eta_t(x, \pi_t(x))$
    end for
end for
output Policy $\pi$, distributions $\eta$
```

Link with regular Policy Optimization This algorithm outputs the optimal policy for the Expectation [Bellemare et al., 2017] (also see Section 2.2.2), and Liang and Luo [2024] also proved that it works for the Entropic Risk Measure. This is because the algorithm is a natural extension of the regular policy optimization dynamic programming (not accounting for state augmentation). Indeed, following the usual recipe (see Section 3.2.1), the algorithm evaluates the risk measure and chooses the best action: $V_t^*(x) = \max_a \varphi(\eta_t^*(x, a))$, where the distribution is computed recursively $\eta_t^*(x, a) = \sum_{x'} p(x' \mid x, a) \tau_{r(x,a,x')} \eta_{t+1}^*(x')$ so that the optimal Q-value function matches $\varphi(\eta_t^*(x, a)) = Q_t^*(x, a)$. Compared to the Expectation (Section 2.1.4) and the Entropic Risk Measure (Section 2.3.2.2), the computation for the optimal policy is the same: evaluate recursively $\varphi(\eta_t^*(x, a))$ and the optimal action is chosen by maximizing this quantity $\pi^*(x) = \arg\max_a \varphi(\eta_t^*(x, a))$. The only difference between the two methods lies in how $\varphi(\eta_t^*(x, a))$ is computed: either through a direct recursive relation (Expectation (Equation (2.5)), EntRM (Proposition 2.13)) or through the full distribution $\eta_t^*(x, a)$

recursive relation. The advantage with the distributional approach is that there is no need for the recursive relation between the value functions, which changes depending on the risk measure. There is a single unified Bellman (distributional) equation. Compared to the discussion in the previous subsection, the risk measure does not need to be Bellman closed.

Correctness of Distributional Policy Optimization Unfortunately, the policy optimization problem is much more complex than the policy evaluation one when considering general risk measures. We can simply show that this algorithm will not optimize any risk measure, as opposed to what happens in the policy evaluation case. The important point to notice is that Algorithm 4 always outputs a Markov policy. However, as discussed in Section 2.3, some risk measures require history-dependent policies to be optimized. Thus, for such functionals, Algorithm 4 cannot output an optimal policy in general. This is the case, for instance, for the variance or the quantiles (see Section 2.3).

Characterizing the correctness of Algorithm 4 We are interested here in characterizing all the risk measures that can be optimized through Algorithm 4, i.e. for which the algorithm outputs an optimal policy. As discussed above, this is necessary for a regular policy optimization DP algorithm to hold. We formalize this idea with the new concept of *Bellman Optimizable* risk measures:

Definition 3.2 (Bellman Optimizable risk measure)**.** A risk measure φ is called *Bellman optimizable* if, for any MDP $\mathcal{M}$, the Pseudo-Algorithm 4 outputs an optimal return distribution $\eta = \eta^*$ that verifies:

$$\forall x, a, h, \quad \varphi(\eta_h^*(x,a)) = \sup_\pi \varphi(\eta_h^\pi(x,a)) \ . \tag{3.18}$$

This definition is equivalent to the satisfaction of an optimal distributional Bellman equation:

Proposition 3.9. A Bellman Optimizable risk measure φ verifies, for any $\mathcal{X}, \mathcal{A}$ finite, $r \in \mathbb{R}^{\mathcal{X}}, p \in \mathcal{P}(\mathcal{X}), \eta \in \mathcal{P}(\mathbb{R})^{\mathcal{X}\times\mathcal{A}}$

$$\sup_{(a_x)\in\mathcal{A}^{\mathcal{X}}} \varphi\left(\sum_x p(x)\tau_{r(x)}\eta(x,a_x)\right) = \varphi\left(\sum_x p(x)\tau_{r(x)}\eta(x,a_x^*)\right)$$

with $a_x^\star \in \arg\max_a \varphi(\eta(x,a))$.

This property amounts to saying that, in any MDP,

$$\eta_t^*(x,a) = \sum_{x'} p(x' \mid x,a)\tau_{r(x,a,x')}\eta_{t+1}^*(x', a_{x'}^*) \ , \tag{3.19}$$

which is what the algorithm assumes and how policy optimization works in practice. Proposition 3.9 is a quite strong requirement and we will try to give some intuition on it. There are mainly two aspects to it. On the right-hand side, the choice of action a_x^* is done solely based on η, and is independent of (1) the probabilities p and (2) the rewards r. Whatever the probabilities and rewards are, choosing the actions that maximize $\varphi(\eta)$ for each state x should also maximize φ of the full distribution. This is illustrated in Figure 3.4 for the Expectation.

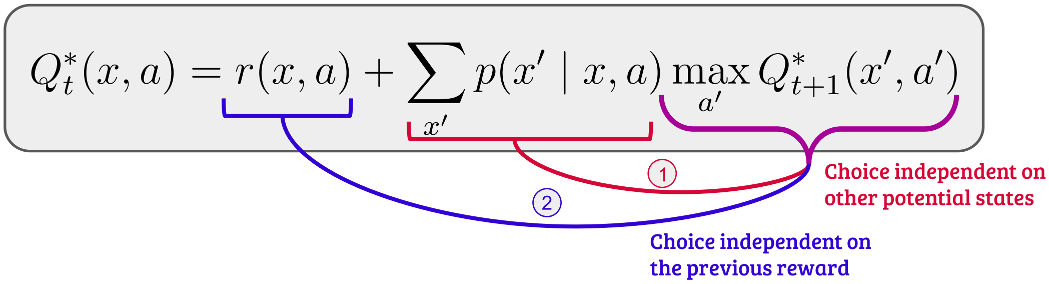
$Q_t^*(x,a) = r(x,a) + \sum_{x'} p(x' \mid x, a) \max_{a'} Q_{t+1}^*(x', a')$
1
2
Choice independent on other potential states
Choice independent on the previous reward

Figure 3.4: Illustration of the Bellman Optimizable property for the Expectation. In here $\mathbb{E}[\mathcal{R}_t(x,a)]$ is maximized by choosing at the next step the actions that maximize $\mathbb{E}[\mathcal{R}_{t+1}(x',a')]$ for each next state x', independently of the probabilities $p(x'|x,a)$ and rewards $r(x,a,x')$.

This analysis can be formalized onto the following lemma.

Lemma 3.4. A Bellman Optimizable risk measure φ satisfies the two following properties:

1. Independence Property: If $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$ are such that $\varphi(\nu_1) \geq \varphi(\nu_2)$, then

$$\forall \nu_3 \in \mathscr{P}(\mathbb{R}), \forall \lambda \in [0,1], \quad \varphi(\lambda\nu_1 + (1-\lambda)\nu_3) \geq \varphi(\lambda\nu_2 + (1-\lambda)\nu_3)\ .$$

2. Translation Property: Let τ_c denote the translation on the set of distributions: $\tau_c\delta_x = \delta_{x+c}$. If $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$ are such that $\varphi(\nu_1) \geq \varphi(\nu_2)$, then

$$\forall c \in \mathbb{R}, \quad \varphi(\tau_c\nu_1) \geq \varphi(\tau_c\nu_2)\ .$$

The Independence property relates to the independence in the convex combination of distributions with the probabilities $p(x' \mid x, a)$ of reaching the state x', while the Translation property relates to the independence of the reward $r(x,a,x')$ before reaching x'. Fundamentally, the properties follow from the Markov nature of policies optimized this way: the choice of the action in each state should be independent of other states and rely only on the knowledge of the next-state value distribution.

The expectation satisfies both properties, thanks to its linearity (illustrated in Figure 3.4), and so does the Entropic Risk Measure [Liang and Luo, 2024]. The Entropic Risk Measure is indeed linear for constants (see Equation (2.17)), and its exponential utility form ensures that the independence property is also verified (see Section 2.3.3).

Independence property and Expected Utilities The Independence property is indeed closely related to expected utilities [Von Neumann and Morgenstern, 1944]. As mentioned in Section 2.3.3, any expected utility verifies this property, but most importantly, it is one of the conditions of the Expected Utility Theorem (also known as the Von Neumann Morgenstern theorem). Along with a continuity condition, it implies that any risk measure φ verifying those properties can be reduced to an expected utility. This means that for any such risk measure φ, there exists f continuous such that $\forall \nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$, we have $\varphi(\nu_1) > \varphi(\nu_2) \iff U_f(\nu_1) > U_f(\nu_2)$ [Von Neumann and Morgenstern, 1944, Grandmont, 1972].

Characterization of Bellman Optimizable risk measures The Von Neumann-Morgenstern theorem directly narrows down the family of Bellman optimizable risk measures to utilities. The next task is therefore to identify all the expected utilities that satisfy the second property. We demonstrate that, apart from the mean and exponential utilities, no other W_1-continuous functional satisfies this property.

Theorem 3.5. The only W_1-*continuous* Bellman Optimizable risk measures of the cumulative return are exponential utilities $\varphi(X) = \text{sign}(\beta)\mathbb{E}[\exp(\beta X)]$ for $\beta \in \mathbb{R}$, with the special case of the expectation $\mathbb{E}[X]$ when $\beta = 0$.

We provide two proofs for this theorem. The first one is the original one from [Marthe et al., 2023]. It is based on our intuition of the properties, but requires that the utility function f is twice differentiable. The second one is less intuitive but does not make any assumptions and makes the link with the proof of the Bellman Closedness characterization of expected utilities (Theorem 3.1).

The intuition on the first proof is that the Translation property imposes strong regularity constraints on the utility function f associated to the expected utility $\varphi = E_f$. A distribution can be seen as a convex combination: $\varphi(\nu) = \sum_i p_i f(x_i)$ for $\nu = \sum_i p_i \delta_{x_i}$. The Translation property implies that a local equality or inequality becomes a global one when shifting the points x_i by a constant c. For instance, if $f(x_1) > f(x_2)$, then for any c, $f(x_1 + c) > f(x_2 + c)$ (choose $\nu_1 = \delta_{x_1}$ and $\nu_2 = \delta_{x_2}$). Similarly, if $\frac{1}{2}(f(x_1) + f(x_2)) > f(\frac{x_1+x_2}{2})$, then for any c, $\frac{1}{2}(f(x_1 + c) + f(x_2 + c)) > f\left(\frac{x_1+x_2}{2} + c\right)$. Overall, assuming differentiability of the utility function f, we can find a differential equation on f that only admits affine or exponential solutions.

If $E_f = \mathbb{E}[f(\cdot)]$ is an expected utility and ψ is a monotonous scalar mapping, $\psi(E_f(\cdot))$ is an equivalent utility: one should understand in the previous theorem that a W_1-*continuous* Bellman Optimizable risk measure is equivalent to $E_{\text{sign}(\beta)\exp(\beta\cdot)}(\cdot)$ for some $\beta \in \mathbb{R}$. This is the same as saying that it is equivalent to optimize either the EntRM or the Exponential Utility (see discussion in Section 2.3.2.1). The full proof of Theorem 3.5 is provided in Section 3.2.3.

Limitations of Theorem 3.5 We make a few important observations. First, this result shows that algorithms using Bellman updates to optimize any continuous risk measure other than the exponential utility cannot guarantee optimality. Then, the theorem does not apply to non-continuous functionals, but Lemma 3.4 still does. For instance, the quantiles are not W_1-continuous so Theorem 3.5 does not apply, but it is easy to prove that they do not verify the Independence Property and thus are not Bellman Optimizable. For non-continuous risk measures, $\operatorname{ess\,sup}$ is an example of Bellman Optimizable functional[5] as it can already be optimized without distributions.

A troubling example An interesting case is the functional mentioned in Section 3.1.1 $\varphi(\nu) = \mathbb{1}\{\nu([0, \infty)) = 1\}$. It is easy to see that the set $\{\varphi\}$ is not Bellman closed. Indeed, consider the two distributions $\eta_1 = \delta_{0.5}$ and $\eta_2 = \delta_{1.5}$. We have $\varphi(\eta_1) = 1$ and $\varphi(\eta_2) = 1$. Then consider adding the reward $r = -1$. We obtain $\varphi(\tau_r \eta_1) = \varphi(\delta_{-0.5}) = 0$ and $\varphi(\tau_r \eta_2) = \varphi(\delta_{0.5}) = 1$. Hence, the values of $\varphi(\eta)$ and r are not enough to compute $\varphi(\tau_r \eta)$, so the set cannot be Bellman closed. Similarly, this same example proves that this functional is not Bellman optimizable, as it does not verify the Translation property: $\varphi(\eta_1) \geq \varphi(\eta_2)$ but $\varphi(\tau_{-1}\eta_1) < \varphi(\tau_{-1}\eta_2)$.

Yet, this functional is a non-decreasing function of $\operatorname{ess\,inf}$. Indeed, $\varphi(\nu)$ is equal to 1 if and only if $\operatorname{ess\,inf}(\nu) \geq 0$. We already mentioned that the set $\{\varphi, \operatorname{ess\,inf}\}$ is Bellman closed. Also, optimizing the $\operatorname{ess\,inf}$ also optimizes φ by composition with an increasing function. As $\operatorname{ess\,inf}$ can be optimized through regular dynamic programming, so does the set $\{\operatorname{ess\,inf}, \varphi\}$. Hence this is an example of a functional that cannot be optimized on its own, but can be optimized when combined with another functional. This is similar to how the variance cannot be evaluated on its own, but can be evaluated when combined with the expectation. The difference lies in the fact that both φ and $\operatorname{ess\,inf}$ can be optimized by the same policy (here the policy optimizing $\operatorname{ess\,inf}$ also optimizes φ), while the expectation and variance may not be optimized by the same policy in general. Please note that the problem of optimizing φ can be reduced to optimizing the $\operatorname{ess\,inf}$ alone, so it does not contradict our argument in Section 3.2.1 about requiring Bellman closedness of singleton sets.

3.2.3 Proofs

Proof of Proposition 3.9

Proof. We start from the definition of Bellman Optimizable risk measure. For any MDP

[5] It can be argued that the $\operatorname{ess\,sup}$ aligns with the exponential utility as the limit case for $\beta = \infty$.

$\mathcal{M}$, the Pseudo-Algorithm 4 outputs an optimal return distribution $\eta = \eta^*$ that verifies:

$$\forall x, a, h, \quad \varphi(\eta^*_h(x,a)) = \sup_\pi \varphi(\eta^\pi_h(x,a)) \ .$$

By definition of the algorithm, we have:

$$\eta^*_h(x,a) = \sum_{x'} p(x' \mid x,a) \tau_{r(x,a,x')} \eta^*_{h+1}(x', a^*_{x'}) \ ,$$

with $a^\star_{x'} \in \arg\max_a \varphi(\eta^*_{h+1}(x',a))$. Hence:

$$\forall x, a, h, \quad \varphi\left(\sum_{x'} p(x' \mid x,a) \tau_{r(x,a,x')} \eta^*_{h+1}(x', a^*_{x'})\right) = \varphi(\eta^*_h(x,a)) \ .$$

By definition of the algorithm and using the fact that $\eta^*_h(x,a)$ is optimal, we also have:

$$\varphi(\eta^*_h(x,a)) = \sup_{(a_{x'})_{x'}} \varphi\left(\sum_{x'} p(x' \mid x,a) \tau_{r(x,a,x')} \eta^*_{h+1}(x', a_{x'})\right) \ ,$$

where the supremum is taken over all possible choices of actions $(a_{x'})$ for each next state x'. Otherwise, we could obtain a better policy. Thus, we obtain:

$$\begin{aligned}\forall x, a, h, \quad & \varphi\left(\sum_{x'} p(x' \mid x,a) \tau_{r(x,a,x')} \eta^*_{h+1}(x', a^*_{x'})\right) \\ & \qquad = \sup_{(a_{x'})_{x'}} \varphi\left(\sum_{x'} p(x' \mid x,a) \tau_{r(x,a,x')} \eta^*_{h+1}(x', a_{x'})\right) \ .\end{aligned}$$

This is exactly the statement of the proposition. □

The proof of Theorem 3.5 result is then divided into several parts. First we show that Bellman optimizable functionals verify the two properties of Lemma 3.4 (Independence and Translation). Then, using those properties, we prove that Bellman optimizable risk measures can only be exponential utilities (Theorem 3.5). Using the known fact that exponential utilities are bellman optimizable, we obtain the full characterization.

Proof of Lemma 3.4

Proof. **To prove that each property is necessary**, we use a proof by contradiction, and exhibit MDPs where the algorithm is not optimal when the property is not verified.

Independence Property Let φ be a Bellman optimizable risk measure that does not satisfy the Independence property. That is, there exists $\nu_1, \nu_2, \nu_3 \in \mathscr{P}(\mathbb{R})$ and $\lambda \in [0,1]$ such that $\varphi(\nu_1) \geq \varphi(\nu_2)$ but $\varphi(\lambda\nu_1 + (1-\lambda)\nu_3) < \varphi(\lambda\nu_2 + (1-\lambda)\nu_3)$.

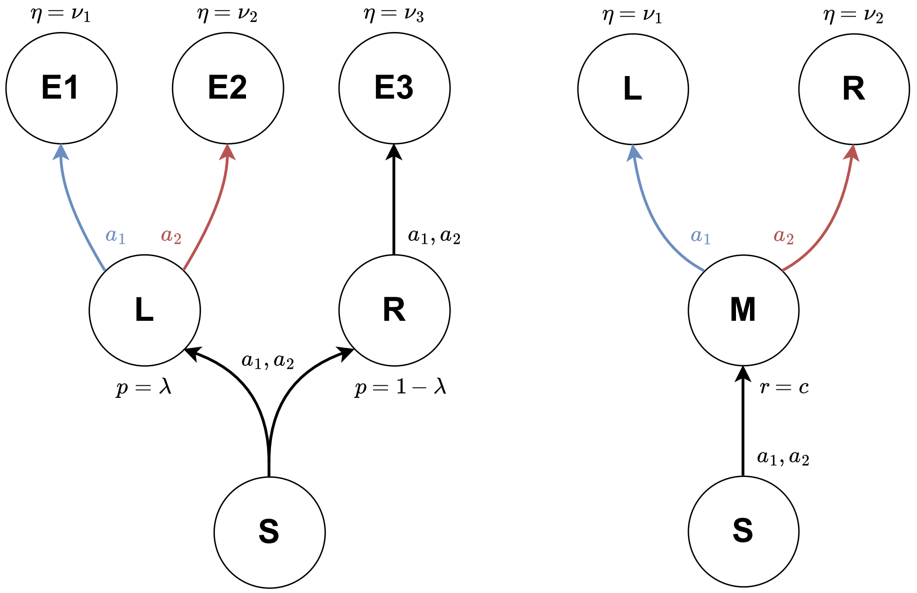

Figure 3.5: Left: Independence Property Counter Example. Right: Translation Property Counter Example. If the respective property is not verified, the distributional policy optimization algorithm does not output an optimal policy in the MDP above.

Then consider the MDP in Fig.3.5 (left) with horizon $H = 2$ corresponding to the depth of the tree: The agent starts in `S` and must take 2 actions, a unique but random and non-rewarding one (a_0) and a final deterministic step (a_1 or a_2) to a rewarding state. Thus, by construction, the optimal strategy is (a_0, a_2) that leads to `E2` with probability λ (and `E3` with probability $1 - \lambda$). The true optimal distribution at `S` state is $\eta_0^* = \lambda\nu_2 + (1-\lambda)\nu_3$. We compute the distributions output by the algorithm:

$$\begin{aligned}
H = 2: &\qquad \eta_2(\mathrm{E1}, a^*) = \delta_0, \quad \eta_2(\mathrm{E2}, a^*) = \delta_0, \quad \eta_2(\mathrm{E3}, a^*) = \delta_0 \\
H = 1: &\qquad \eta_1(\mathrm{L}, a^* = \arg\max_a s(\nu_a)) = \nu_1, \quad \eta_1(\mathrm{R}, a^* = a_1, a_2) = \nu_3 \\
H = 0: &\qquad \eta_0(\mathrm{S}, a^* = a_0) = \lambda\nu_1 + (1-\lambda)\nu_3
\end{aligned}$$

The output return distribution η_0 is not the true optimal η_0^* for φ so the algorithm is incorrect which is a contradiction as φ is assumed to be Bellman optimizable. Hence the property is needed.

Translation Property Let φ be a Bellman optimizable risk measure that does not verify the Translation Property, *i.e.* there exists $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$, $c \in \mathbb{R}$ such that $\varphi(\nu_1) \geq \varphi(\nu_2)$ but $\varphi(\tau_c\nu_1) < \varphi(\tau_c\nu_2)$. Then consider MDP in Fig.3.5 (right). The optimal strategy is again (a_0, a_2) by construction. The algorithm output the following distribution:

$$\begin{aligned}
H = 2: \quad & \eta_2(\mathrm{L}, a^*) = \delta_0, \quad \eta_2(\mathrm{R}, a^*) = \delta_0 \\
H = 1: \quad & \eta_1(\mathrm{M}, a^*) = \nu_1 \\
H = 0: \quad & \eta_0(\mathrm{S}, a^*) = \tau_c \nu_1
\end{aligned}$$

So here again, the algorithm does not output an optimal distribution for φ, hence the necessity of the property.

This proof shows that both properties are necessary, but not that they are sufficient. The other implication could be proven, but the proof would be unnecessary as those properties are enough to restrict to only 1 class of function for which we already know is bellman optimizable.

□

First part of the proof of Theorem 3.5 The first part consists in applying the Von Neumann-Morgenstern theorem to reduce the problem to expected utilities. Indeed, the theorem requires 4 axioms: Completeness, Transitivity, Continuity and Independence, as mentioned in Section 2.3.3. The Completeness and Transitivity axioms are trivially verified by any risk measure: by definition, a risk measure induces a total order on distributions, using the natural order of $\mathbb{R}$. The Continuity axiom is verified using the W_1-continuity assumption of the theorem. Finally, the Independence axiom is verified using the Independence Property from Lemma 3.4. Hence, by the Von Neumann-Morgenstern theorem, there exists a continuous function $f : \mathbb{R} \to \mathbb{R}$ such that for any $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$,

$$\varphi(\nu_1) \geq \varphi(\nu_2) \iff U_f(\nu_1) \geq U_f(\nu_2)\ . \tag{3.20}$$

Hence the problem is equivalent to optimizing the expected utility associated to f. Now we consider such expected utility φ_f associated to a continuous function f. Assuming the Translation property, we have for any $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R}), c \in \mathbb{R}$:

$$\int f(x)\mathrm{d}\nu_1(x) = \int f(x)\mathrm{d}\nu_2(x) \implies \int f(x+c)\mathrm{d}\nu_1(x) = \int f(x+c)\mathrm{d}\nu_2(x)\ . \tag{3.21}$$

We aim to show that this implies that f is either affine or exponential.

We provide two proofs here, the first one is the original from [Marthe et al., 2023], and the second is a new one making the link with the proof of Theorem 3.1 about the characterization of Bellman Closed utilities.

Second part of the proof of Theorem 3.5 (first proof)

Proof. In this proof, we assume that f is two times differentiable, i.e. $f \in C^2(\mathbb{R})$.

If f is constant, the theorem holds trivially. Suppose f is not constant. There exists a point x_0 such that $f'(x_0) \neq 0$. By translation invariance, we may assume $x_0 = 0$

without loss of generality. Since $f \in C^2$ and $f'(0) \neq 0$, the Inverse Function Theorem guarantees that f is locally invertible in a neighborhood of 0, and its inverse is also C^2.

For sufficiently small h, we define $\phi(h) = f^{-1}\left(\frac{1}{2}(f(0) + f(h))\right)$. This implies:

$$f(\phi(h)) = \frac{1}{2}(f(0) + f(h)) \ .$$

Note that $\phi \in C^2$ and $\phi(0) = f^{-1}(f(0)) = 0$.

Consider the probability distributions $\mu_1 = \frac{1}{2}(\delta_0 + \delta_h)$ and $\mu_2 = \delta_{\phi(h)}$. By definition of $\phi(h)$, we have:

$$U_f(\mu_1) = \frac{1}{2}(f(0) + f(h)) = f(\phi(h)) = U_f(\mu_2) \ .$$

The Translation property implies that this equality is preserved under any translation $x \in \mathbb{R}$. Therefore, $U_f(\mu_1 * \delta_x) = U_f(\mu_2 * \delta_x)$, which gives the functional equation:

$$\frac{1}{2}(f(x) + f(x+h)) = f(x + \phi(h)), \quad \forall x \in \mathbb{R} \ . \tag{3.22}$$

Differentiating (3.22) with respect to h gives:

$$\frac{1}{2} f'(x+h) = \phi'(h) f'(x + \phi(h)) \ . \tag{3.23}$$

Evaluating (3.23) at $x = 0$ and $h = 0$ (using $\phi(0) = 0$), we obtain $\frac{1}{2} f'(0) = \phi'(0) f'(0)$. Since $f'(0) \neq 0$, we deduce that $\phi'(0) = \frac{1}{2}$.

Differentiating (3.23) again with respect to h gives:

$$\frac{1}{2} f''(x+h) = \phi''(h) f'(x + \phi(h)) + [\phi'(h)]^2 f''(x + \phi(h)) \ . \tag{3.24}$$

Letting $h \to 0$ in (3.24) and substituting $\phi'(0) = \frac{1}{2}$ and $\phi(0) = 0$, we find:

$$\begin{aligned} &\frac{1}{2} f''(x) = \phi''(0) f'(x) + \left(\frac{1}{2}\right)^2 f''(x) \\ \iff &\frac{1}{4} f''(x) = \phi''(0) f'(x) \ . \end{aligned}$$

We distinguish two cases based on the value of $\phi''(0)$:

Case 1: $\phi''(0) = 0$. The equation reduces to $f''(x) = 0$. Thus, f is affine: $f(x) = ax + b$.

Case 2: $\phi''(0) \neq 0$. Let $\lambda = 4\phi''(0)$. The differential equation becomes $f''(x) = \lambda f'(x)$. The general solution is of the form:

$$f(x) = c_1 e^{\lambda x} + c_2 \ .$$

Consequently, f must be either affine or exponential (up to affine transformations). □

Second part of the proof of Theorem 3.5 (second proof) We first start by proving the following lemma:

Lemma 3.6. Let $g(x) = f(x) - f(0)$. There exists a mapping $\alpha : \mathbb{R} \to \mathbb{R}$ such that for any $x, c \in \mathbb{R}$:

$$g(x+c) = \alpha(c)g(x) + g(c) \ . \tag{3.25}$$

Proof. Define for $c \in \mathbb{R}$, $\Psi_c(\nu) = \int f(x+c)d\nu(x)$. We observe that Ψ is affine: for any $\nu_1, \nu_2 \in \mathscr{P}(\mathbb{R})$ and $\lambda \in [0,1]$,

$$\begin{aligned}\Psi_c(\lambda\nu_1 + (1-\lambda)\nu_2) &= \int f(x+c)d(\lambda\nu_1 + (1-\lambda)\nu_2)(x) \\ &= \lambda \int f(x+c)d\nu_1(x) + (1-\lambda)\int f(x+c)d\nu_2(x) \\ &= \lambda\Psi_c(\nu_1) + (1-\lambda)\Psi_c(\nu_2) \ .\end{aligned}$$

Now define $H_c : \operatorname{Im} f \to \operatorname{Im} f$ by

$$H_c(\Psi_0(\nu)) = \Psi_c(\nu) \ . \tag{3.26}$$

We verify that H_c is well-defined: if $\Psi_0(\nu_1) = \Psi_0(\nu_2)$, then by the translation property,

$$\begin{aligned}\Psi_0(\nu_1) = \Psi_0(\nu_2) &\implies \int f(x)d\nu_1(x) = \int f(x)d\nu_2(x) \\ &\implies \int f(x+c)d\nu_1(x) = \int f(x+c)d\nu_2(x) \\ &\implies \Psi_c(\nu_1) = \Psi_c(\nu_2) \ .\end{aligned}$$

Also, H_c is affine: for any $y_1, y_2 \in \operatorname{Im} f$ and $\lambda \in [0,1]$, let ν_1, ν_2 such that $\Psi_0(\nu_i) = y_i$.

$$\begin{aligned}H_c(\lambda y_1 + (1-\lambda)y_2) &= H_c(\lambda\Psi_0(\nu_1) + (1-\lambda)\Psi_0(\nu_2)) \\ &= H_c(\Psi_0(\lambda\nu_1 + (1-\lambda)\nu_2)) \\ &= \Psi_c(\lambda\nu_1 + (1-\lambda)\nu_2) \\ &= \lambda\Psi_c(\nu_1) + (1-\lambda)\Psi_c(\nu_2) \\ &= \lambda H_c(\Psi_0(\nu_1)) + (1-\lambda)H_c(\Psi_0(\nu_2)) \\ &= \lambda H_c(y_1) + (1-\lambda)H_c(y_2) \ .\end{aligned}$$

Where the second and fourth equalities come from the affinity of Ψ_c and Ψ_0. Hence, H_c is affine, so there exists $\alpha(c), \gamma(c) \in \mathbb{R}$ such that for any $y \in \operatorname{Im} f$,

$$H_c(y) = \alpha(c)y + \gamma(c) \ . \tag{3.27}$$

In particular, for any $x, c \in \mathbb{R}$, we have:

$$f(x+c) = \Psi_c(\delta_x) = H_c(\Psi_0(\delta_x)) = H_c(f(x)) = \alpha(c)f(x) + \gamma(c) \ .$$

Going back to g, we have:

$$\begin{aligned} g(x+c) &= f(x+c) - f(0) \\ &= \alpha(c) f(x) + \gamma(c) - f(0) \\ &= \alpha(c)\left(f(x) - f(0)\right) + \underbrace{\alpha(c) f(0) + \gamma(c)}_{=f(c)} - f(0) \\ &= \alpha(c) g(x) + g(c)\ . \end{aligned}$$

□

We can now directly prove Theorem 3.5 by using the same argument as for the Bellman Closedness characterization (Theorem 3.1) in the 1-dimensional case: for any $c \in \mathbb{R}$, the function $g(\cdot + c)$ is an affine transformation of $g(\cdot)$, which implies by Engert [1970] that g is either affine or an affine transformation of an exponential, and so is f.

3.3 Discussions

Bellman Closedness in finite-horizon MDPs A question we raise is whether Bellman closedness is always the right notion to characterize the risk measures that can be evaluated through dynamic programming. The issue comes from the specific setting of discounted MDPs. The original result of Rowland et al. [2019] shows that, in this case, only the moments are Bellman closed in analogy with Theorem 3.1. While indeed $\mathrm{EntRM}_\beta[\eta_t]$ cannot be expressed as a function of $\mathrm{EntRM}_\beta[\eta_{t+1}]$, it can be expressed as a function of $\mathrm{EntRM}_{\gamma\cdot\beta}[\eta_{t+1}]$ [Chung and Sobel, 1987, Hau et al., 2023b]. Thus, it can be efficiently evaluated by dynamic programming by computing at timestep t the value of $\mathrm{EntRM}_{\gamma^{H-t}\cdot\beta}[\eta_t]$. The issue is that the functional changes with time, which is not accounted for by Bellman closedness. Hence, there are some functionals that can be evaluated through dynamic programming, but do not fit into any Bellman closed set.

Importance of the distributional framework Importantly, while in theory, the distributional framework provides the most general framework for optimizing policies via dynamic programming, our result shows that in fact, the only utilities that can be exactly and efficiently optimized do not require resorting to DistRL. This certainly does not question the very purpose of DistRL, which has been shown to play important roles in practice (see discussion in Section 2.2.2).

Extension to the stock-augmented state space As explained in Section 2.3.3.2, a way to solve the policy optimization problem for a broader class of functionals is to augment the state space with the current stock. In these augmented-state MDPs, the optimal policy becomes Markov. It is natural to wonder if the results of this section can be extended to those augmented MDPs. This is indeed what Pires et al. [2025] study, building on our results. The way the stock-augmented recursion formula works makes it so the reward can be assumed to be always zero in the Bellman equation (all the information is contained in the transition function with the stock being part of the state). They leverages it to show that only the independence property (there, called *indifference to mixtures*) is enough for a risk measure to be optimizable through dynamic programming in the stock-augmented MDP. Using a similar argument as with Theorem 3.5, among the W_1-continuous risk measures, only the expected utilities are optimizable in this framework. However, they also fail to find any risk measure that can only be optimized through this distributional framework. They mention $\varphi(\nu) = \mathbb{1}\{\operatorname{supp} \nu \subset [0, \infty)\}$ as a possible candidate, but we debunked this earlier in this chapter.

3.4 Conclusion

In this chapter, we investigated the theoretical limits of the distributional framework regarding the evaluation and optimization of general statistical functionals in undiscounted MDPs. Our analysis reveals a fundamental dichotomy between the flexibility of policy evaluation and the rigidity of policy optimization when using dynamic programming.

Regarding policy evaluation, the distributional framework is theoretically powerful as it enables the computation of the full return distribution for any Markov policy via dynamic programming. This means policy evaluation can be solved for any law-invariant risk measure. However, exact DP is confined to a Bellman closed set of statistics, primarily affine combinations of exp-moment functions. For all other complex risk measures, approximation is validated: we established that for any Lipschitz statistic (such as CVaR_α), the approximation error remains bounded and scales only quadratically with the horizon.

Conversely, for optimization, we established a strong impossibility result. We demonstrated that to be optimizable by dynamic programming, a functional must satisfy two strict properties. In particular, we proved that only the expectation and exponential utilities satisfy these conditions. This implies that Distributional RL does not extend the class of exact, optimizable risk measures beyond what is already possible with classical dynamic programming.

Consequently, exact optimization of risk measures outside this narrow family is unattainable through distributional Bellman updates. This validates the necessity of alternative approaches. In the next chapter, we leverage these findings to propose a new approximate optimization strategy: using the Entropic Risk Measure family to compute tractable risk-sensitive policies, and distributional techniques to evaluate them on general risk measures.

4

General Risk Sensitive Planning through the Entropic Risk Measure

Contents

Despite their practical interest and interpretability, the common quantile risk metrics (VaR and CVaR) and the Threshold Probability cannot be efficiently optimized using standard dynamic programming (see Section 2.3.4, Section 2.3.3, Section 3.2). Our work addresses precisely this gap by connecting the Threshold Probability, the VaR and the CVaR metrics to the moment-generating function of the return of a policy. In fact, in the context of MDPs, these two optimization problems can be approximated by the Entropic (Exponential) Risk Measure [Howard and Matheson, 1972], the unique functional admitting a dynamic programming decomposition [Ben-Tal and Teboulle, 2007, Föllmer and Schied, 2011] (also see Section 3.2). Yet, despite this computational appeal, practical usage of EntRM can be hindered by interpretability concerns, especially around selecting the risk-tolerance parameter β.

We propose a unifying framework for risk-sensitive planning in MDPs through the study of the EntRM. Rather than directly optimizing the latter as a proxy for other target metrics, we prove that optimal policies evolve in a structured manner as the risk parameter changes, which allows us to derive an efficient algorithm to compute all the optimal policies for the EntRM, a set that we call *the Optimality Front*. We show how this set can then be used to optimize tail-focused risk measures through the *Generalized Policy Improvement* principle [Barreto et al., 2020] (see Section 4.1). We demonstrate the consequence of this new method on the optimization of the Threshold Probability, the VaR and CVaR via an empirical study on the Cliff environment and an inventory management problem.

4.1 A Unified Framework for Risk-Sensitive Planning

The Threshold Probability and VaR/CVaR objectives are all related to the tail probabilities of the return distribution. These tail probabilities can be approximated with the help of exponential moments of the distribution by Chernoff bound:

$$\Pr(X \leq T) \quad \leq \quad \inf_{\beta \leq 0} \exp(-\beta T)\,\mathbb{E}[e^{\beta X}]. \tag{4.1}$$

Exponential moments are the core of the Entropic Risk Measure and we explain in this section how Inequality (4.1) leads to proxies for the risk metrics introduced above.

From (C)VaR to EVaR. Solving for the right-hand side of (4.1) to equal α, Ahmadi-Javid [2012] introduced the *Entropic Value at Risk (EVaR)* as a proxy for the VaR defined by

$$\mathrm{EVaR}_\alpha[X] = \sup_{\beta<0} \mathrm{EntRM}_\beta[X] - \frac{1}{\beta}\log(\alpha).$$

EVaR has been shown to be an approximation of the VaR and CVaR, due to $\mathrm{VaR}_\alpha[X] \geq \mathrm{CVaR}_\alpha[X] \geq \mathrm{EVaR}_\alpha[X]$ [Ahmadi-Javid, 2012]. It is a *coherent* risk measure, and its use for approximating VaR for bandit algorithms was already noted by Maillard [2013] and has received growing interest recently in MDPs [Ni and Lai, 2022, Hau et al., 2023b, Su et al., 2024]. The related proxy for VaR and CVaR is

$$\max_{\pi} \text{(C)VaR}_{\alpha}\left[R^{\pi}\right] \geq \sup_{\beta<0} \max_{\pi} \text{EntRM}_{\beta}[R^{\pi}] - \frac{\log(\alpha)}{\beta}, \tag{4.2}$$

where the sup and max can be swapped as the policy space is finite.

A proxy for the Threshold Probability. Using a similar idea, we derive a proxy for the Threshold Probability:

$$\max_{\pi} \; -\Pr(R^{\pi} \;\leq\; T) \geq \sup_{\beta<0} \max_{\pi} -e^{-\beta T}\,\mathbb{E}[e^{\beta R^{\pi}}]. \tag{4.3}$$

More details can be found in Section 4.1.3. To the best of our knowledge, this metric has been little studied so far in the setting of MDPs, and this simple approximation scheme seems novel.

Quality of the approximations. The quality of the deviation bound used to derive the proxies above depends on the tail of the distributions and is known to be more accurate for distributions with light tails [Vershynin, 2018]. In MDPs, the return of a policy is more concentrated around its mean when there is a rich and bounded reward signal along the trajectory, leading to thinner tails and tighter EVaR approximations.

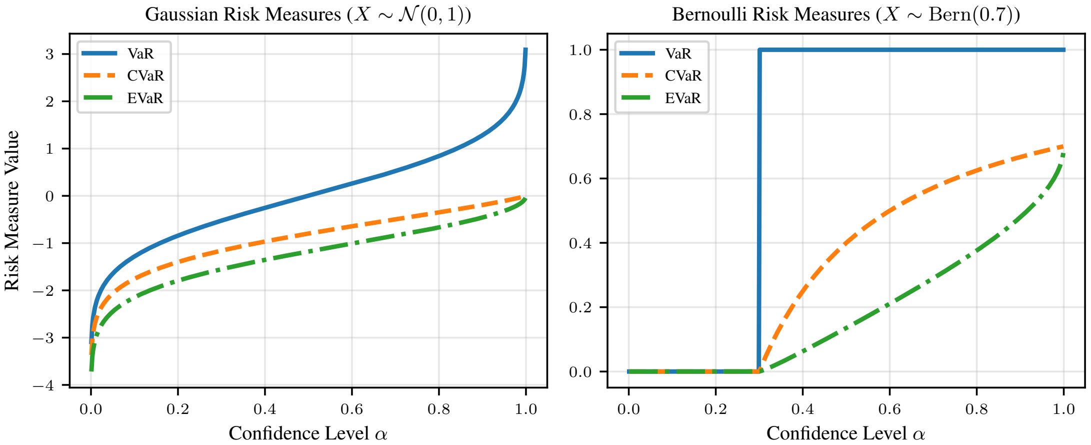

Figure 4.1: Illustration of the values of VaR, CVaR, and EVaR for various distributions. EVaR is a tighter approximation of VaR and CVaR when the reward signal is rich (left) compared to sparse (right).

Relaxed optimization problems. Having derived proxy targets, a natural idea is to attempt to optimize them *instead of the original risk measure*. We comment on these proxy optimization problems and propose below a better and more general method that builds on these intermediate solutions.

For a given parameter β, we denote π^*_β the optimal policy for the EntRM_β[1]:

$$\pi^*_\beta := \arg\max_{\pi\in\Pi} \mathrm{EntRM}_\beta[R^\pi]\,. \tag{4.4}$$

For all objectives, as suggested by the proxy derivations above, risk-sensitive optimization problems can be reduced to finding the right parameter β:

$$\beta^*_{\mathrm{(C)VaR}} = \arg\sup_{\beta<0} \mathrm{EntRM}_\beta[R^{\pi^*_\beta}] - \frac{1}{\beta}\log(\alpha), \quad \beta^*_{\mathrm{TP}} = \arg\sup_{\beta<0} -e^{-\beta T}\,\mathbb{E}[e^{\beta R^{\pi^*_\beta}}], \tag{4.5}$$

with the corresponding $\pi^*_{\beta^*}$ as the optimal policy for the optimized metric.

Both proxy problems are effectively optimizing the EntRM, inducing important properties of the resulting optimal policies.

Proposition 4.1 (Hau et al. [2023b]). For each proxy problem in (4.5), there exists $\beta < 0$ such that the optimal policy is also optimal for the EntRM with parameter β. The optimal policies are deterministic and Markov.

Compared to the initial problems where optimal policies are usually not Markov, the transformation using the EntRM allows finding optimal policies that are easier to compute and implement. However, the proxies in (4.5) now involve an optimization over a *continuous* range of values of $\beta < 0$, and for each value, the return of the optimal policy π^*_β must be computed. While we know efficient algorithms to compute this quantity for one fixed β, optimizing over a continuous range is significantly harder and the only known approaches use discretization schemes [Hau et al., 2023b]. See Section 4.1.1 for more details.

Our main contribution is to show that this continuous search problem can be done efficiently, and more importantly, that the set of all optimal policies $(\pi^*_\beta)_{\beta<0}$ is nicely structured and can be stored to further optimize the original risk measure rather than their proxy.

4.1.1 Grid-Based Optimization of the Risk Parameter

EVaR optimization A natural first idea to optimize the EVaR in MDPs is to discretize β on a grid. Hau et al. [2023b] show that they can get an ε-approximation of the optimal

[1] Recall that this optimal policy can be efficiently computed by DP.

policy for the EVaR problem by considering a specific discretization where the values of β follow the sequence[2]:

$$\beta_0 = 0, \qquad \beta_1 = -\frac{2\varepsilon}{H^2}, \qquad \beta_{i+1} = \frac{\beta_i \cdot \log(\alpha)}{\log(\alpha) - \beta_i \cdot \varepsilon}.$$

where the last value β_K satisfies $\beta_K = \max\left\{\beta_i \mid \beta_i \leq \frac{\log(\alpha)}{\varepsilon}\right\}$. They show that the number of computed points can be bounded by $O(H^2 \frac{\log(H/\varepsilon)}{\varepsilon^2})$. Using the fact that each policy can be computed in $O(|\mathcal{S}|^2\,|\mathcal{A}|\,H)$ time, the overall complexity is $O(|\mathcal{S}|^2\,|\mathcal{A}|\,H^3 \frac{\log(H/\varepsilon)}{\varepsilon^2})$.

Threshold Probability optimization We prove a similar result for the Chernoff approximation of the Threshold Probability problem.

Proposition 4.2. Let R be the return of an MDP bounded between $m < 0$ and $M > 0$. Assume that there exists $a < 0$ and $p > 0$ with the property that for any policy π, $\Pr(R \leq a) \geq p$. Then, solving the proxy problem with accuracy ε on β and $\beta_{\min} = \log(1/p)/a$, finds a policy π that satisfies

$$\Pr(R^{\pi^*} \leq 0) \leq \tilde{B} \quad \text{and} \quad B \leq \tilde{B} \leq \exp\left(-\varepsilon \frac{mM}{M-m}\right) B\,,$$

where $B = \sup_{\beta<0} \max_\pi -e^{-\beta T}\,\mathbb{E}[e^{\beta R^\pi}]$ is the Threshold Probability proxy.

Setting $m = -H, M = H$, one needs to compute $\frac{H\,\beta_{\min}}{2\,\log(1+\varepsilon)}$ policies to obtain a value within a factor $(1+\varepsilon)$ of the optimum. Given that computing an optimal policy for EntRM involves a complexity of $O(|\mathcal{S}|^2\,|\mathcal{A}|\,H)$, the overall complexity to achieve an approximation ratio of $(1+\varepsilon)$ is $O\Big(\frac{H^2\,|\mathcal{S}|^2\,|\mathcal{A}|\,\beta_{\min}}{\log(1+\varepsilon)}\Big)$.

Note that this result translates to any value of threshold other than 0, by just translating the rewards of the MDP so that the threshold becomes equivalent to 0. In general we cannot give a bound on $\beta_{\min}$ without further knowledge of the MDP, but we expect it to scale with H, so as to match the complexity of the EVaR proxy.

As expected, those bounds show that the complexity explodes as the grid is refined to obtain more accurate approximations. However, this method does not use the knowledge about the structure of the set of policies $\Pi_\beta = \{\pi^*_\beta \mid \beta \leq 0\}$, and will tend to compute the same optimal policies over and over again.

We show in Section 4.2 that Π_β is actually piecewise constant, and knowing this structure it is possible to design slightly more efficient algorithms: consider the EntRM

[2] Their results only address the discounted MDP setting, with the convention $\beta \geq 0$. We give here the adaptation using only the change of convention for the quantile risk measures and the undiscounted setting.

optimal policies $\pi_1, \ldots, \pi_K$ and the corresponding optimality intervals $I_1, \ldots, I_K$. We have

$$\begin{aligned}\max_{\pi} \; -\Pr(R^\pi \leq T) &\geq \max_{\beta<0} -\mathbb{E}[e^{\beta(R^{\pi^*_\beta}-T)}] \\ &= \max_{k} \; \max_{\beta \in I_k} -\mathbb{E}[e^{\beta(R^{\pi_k}-T)}].\end{aligned}$$

and

$$\begin{aligned}\max_{\pi} \; \text{(C)VaR}_\alpha[R^\pi] &\geq \sup_{\beta<0} \text{EntRM}_\beta[R^{\pi^*_\beta}] + \frac{1}{\beta}\log(\alpha) \\ &= \max_{k} \max_{\beta \in I_k} \text{EntRM}_\beta[R^{\pi_k}] + \frac{1}{\beta}\log(\alpha).\end{aligned}$$

These problems are reduced to simpler optimization problems on small intervals that can be solved using gradient methods. Indeed, the first one, for the Threshold Probability, is concave [Boyd and Vandenberghe, 2004], and the second one is quasiconcave[3] [Hau et al., 2023b]. The analysis on the structure of Π_β and how to compute it is studied in the following sections.

4.1.2 Generalized Policy Improvement

The key observation is that the optimal policy $\pi^*_{\beta^*}$ for the proxy target (4.5) need not be the best one for the original risk measure. In general, there may be $\beta' \neq \beta^*$ such that $\pi^*_{\beta'}$ achieves better performance on the original risk measure (see the experiments in Section 4.4.2).

Conveniently, optimizing the proxies already requires sweeping through all β values and computing all the optimal policies for EntRM (4.4), $\Pi^* = \{\pi^*_\beta \,|\, \beta \leq 0\}$, that we call the *Optimality Front*. While previous works discard this set during the optimization process [Hau et al., 2023b], we propose to store it, and apply the *Generalized Policy Improvement* principle (GPI) proposed by Barreto et al. [2020]. Namely, multiple policies are computed using tractable objectives (here, EntRM_β for various β), and then the one performing best under the original, possibly intractable, risk criterion is selected. Our original approach is to leverage the Optimality Front such that for any risk measure ρ, we compute the optimal policy as

$$\max_{\pi \in \Pi^*} \rho(R^\pi) \quad \text{with} \quad \Pi^* = \{\pi^*_\beta \mid \beta \leq 0\}\,. \tag{4.6}$$

In Section 4.2, we give all the elements to justify that the Optimality Front of the EntRM is indeed a suitable set of policies to perform GPI. Mainly, we show that it can

[3] It becomes a concave problem after using the change of variable $\beta \leftarrow \frac{1}{\beta}$ and thus can still be optimized efficiently

be computed efficiently and that it is *small* in general because there is a bounded number of β values for which the optimal policy changes. Importantly, it can be computed once and reused across multiple downstream risk objectives, such as VaR, CVaR, threshold-based criteria, or any arbitrary risk-sensitive objective. Evaluating each candidate policy under the target risk measure can be done efficiently using distributional planning (see Section 3.1), which provides access to the full return distribution of each π^*_β.

4.1.3 Proofs

EVaR The *Entropic Value at Risk* was proposed as a tighter exponential-based bound on VaR and CVaR. We rewrite here the derivation of Ahmadi-Javid [2012] to obtain EVaR as a proxy for VaR.

Proof. By the Chernoff inequality, for any $\beta < 0$ we have:

$$\Pr(X \leq \ell) \ \leq \ \mathbb{E}[e^{\beta X}] \exp(-\beta\, \ell).$$

Solving the equation $\mathbb{E}[e^{\beta X}] \exp(-\beta\, \ell) \ = \ \alpha$ for ℓ, we obtain

$$\ell \ = \ a_X(\beta, \alpha) \ = \ \frac{1}{\beta} \log(\mathbb{E}[e^{\beta X}]) \ - \ \frac{1}{\beta} \log(\alpha).$$

Hence, for any $\beta < 0$ and $\alpha \in (0, 1]$, the inequality

$$\Pr(X \leq a_X(\beta, \alpha)) \ \leq \ \alpha$$

implies

$$a_X(\beta, \alpha) \ \leq \ \mathrm{VaR}_\alpha(X).$$

Hence,

$$\mathrm{EVaR}_\alpha(X) \ = \ \sup_{\beta<0} a_X(\beta, \alpha) \ \leq \ \mathrm{VaR}_\alpha(X).$$

□

Probability Threshold Problem. The initial problem is

$$\max_\pi \ -\Pr(R^\pi \ \leq \ T).$$

Using the Chernoff bound, and a few manipulations, we obtain

$$\begin{aligned}\max_\pi \ -\Pr(R^\pi \leq T) &\geq \max_\pi \ \max_{\beta<0} -\mathbb{E}[e^{\beta(R^\pi - T)}] \\ &= \max_{\beta<0} \ \max_\pi -\mathbb{E}[e^{\beta(R^\pi - T)}] \\ &= \max_{\beta<0} -\mathbb{E}[e^{\beta\left(R^{\pi^*_\beta} - T\right)}]\end{aligned}$$

where the inversion between the maximum on the policy and the maximum on β is justified by the fact that there is only a finite number of policies.

Value at Risk Problem. The initial problem is

$$\max_{\pi} \ (\mathrm{C})\mathrm{VaR}_{\alpha}[R^{\pi}].$$

Similar manipulations lead to

$$\begin{aligned}
\max_{\pi} \ (\mathrm{C})\mathrm{VaR}_{\alpha}[R^{\pi}] \geq \max_{\pi} \ \mathrm{EVaR}_{\alpha}[R^{\pi}] \\
&= \max_{\pi} \sup_{\beta<0} \mathrm{EntRM}_{\beta}[R^{\pi}] + \frac{1}{\beta}\log(\alpha) \\
&= \sup_{\beta<0} \max_{\pi} \mathrm{EntRM}_{\beta}[R^{\pi}] + \frac{1}{\beta}\log(\alpha) \\
&= \sup_{\beta<0} \mathrm{EntRM}_{\beta}[R^{\pi^*_{\beta}}] + \frac{1}{\beta}\log(\alpha)
\end{aligned}$$

Proof of Proposition 4.2 We denote by $M_{\mathcal{R}}(\beta) = \mathbb{E}[\exp(\beta\mathcal{R})]$ the moment generating function of the random variable $\mathcal{R}$. We first prove the following lemma.

Lemma 4.1. Suppose that $\mathcal{R}$ is bounded between m and M, then for a fixed step size $\varepsilon > 0$, for $k \in \mathbb{N}$ and $\beta \in [\varepsilon k, \varepsilon(k+1)]$, noting $b_k = M_{\mathcal{R}}(-\varepsilon k)$, we have

$$M_{\mathcal{R}}(-\beta) \geq \exp(-\varepsilon M) b_k \quad \text{and} \quad M_{\mathcal{R}}(-\beta) \geq \exp(\varepsilon m) b_{k+1},$$

and if $m < 0$ and $M > 0$,

$$\frac{M_{\mathcal{R}}(-\beta)}{\min\{b_k, b_{k+1}\}} \geq \exp\left(\varepsilon \frac{mM}{M-m}\right) \max\left\{\left(\frac{b_k}{b_{k+1}}\right)^{\frac{m}{M-m}}, \left(\frac{b_k}{b_{k+1}}\right)^{\frac{M}{M-m}}\right\} \geq \exp\left(\varepsilon\frac{mM}{M-m}\right)$$

otherwise the ratio is greater or equal to 1.

Proof. First we have

$$\begin{aligned}
M_{\mathcal{R}}(-\beta) &= \mathbb{E}[\exp(-(\beta - \varepsilon k)\mathcal{R}) \exp(-\varepsilon k \mathcal{R})] \\
&\geq \exp(-(\beta - \varepsilon k)M)\mathbb{E}[\exp(-\varepsilon k\mathcal{R})] \geq \exp(-\varepsilon M) b_k, \\
M_{\mathcal{R}}(-\beta) &= \mathbb{E}[\exp((-\beta + \varepsilon(k+1))\mathcal{R}) \exp(-\varepsilon(k+1)\mathcal{R})] \\
&\geq \exp((\varepsilon(k+1) - \beta)m)\mathbb{E}[\exp(-\varepsilon(k+1)\mathcal{R})] \geq \exp(\varepsilon m) b_{k+1}.
\end{aligned}$$

To obtain the following inequality it is enough to consider the intersection of the functions $t \mapsto \exp(-tM)b_k$ and $t \mapsto \exp((\varepsilon - t)m)b_{k+1}$ on $t \in [0, \varepsilon]$: as $M_{\mathcal{R}}(-\beta)$ follows both left decreasing and right increasing constraint, the minimum value is on the intersection. □

We can now give a proof of proposition 4.2.

Proof. Let π_c and β_c be the policy and the β optimizing the Chernoff bound B. First we can consider that $\beta_c < \log(1/p)/a$. Indeed if not we can choose $\beta_c = 0$ instead:

$$\begin{aligned} M_{\mathcal{R}}(-\beta_c) &= \mathbb{E}[\exp(-\beta_c\mathcal{R})\mathbb{1}\{\mathcal{R} \leq a\}] + \mathbb{E}[\exp(-\beta_c\mathcal{R})\mathbb{1}\{\mathcal{R} > a\}] \\ &\geq \exp(-\beta_c a)\Pr[\mathcal{R} \leq a] + \exp(-\beta_c M)\Pr[\mathcal{R} \geq a] \\ &\geq \exp(-\beta_c a)p \geq 1 = M_{\mathcal{R}}(0) \quad \text{using the hypothesis on } \beta_c. \end{aligned}$$

Then let $k \in \mathbb{N}$ be such that $\varepsilon k \leq \beta_c < \varepsilon(k+1)$, suppose $M_{\mathcal{R}^{\pi_c}}(-\varepsilon k) \leq M_{\mathcal{R}^{\pi_c}}(-\varepsilon(k+1))$ without loss of generality. We have

$$\tilde{B} \leq \min_{\pi} M_{\mathcal{R}^{\pi}}(-\varepsilon k) \leq M_{\mathcal{R}^{\pi_c}}(-\varepsilon k) \leq \exp\left(\varepsilon\frac{-mM}{M-m}\right) M_{\mathcal{R}^{\pi_c}}(\beta_c) \leq \exp\left(\varepsilon\frac{-mM}{M-m}\right) B,$$

using lemma 4.1. □

4.2 Structural Insights into Entropic Risk Measures

Optimizing EntRM for an entire range of risk parameters requires understanding the structure of EntRM optimal policies. We give here a series of results to help characterize it. Furthermore, understanding how a small perturbation of the risk parameter can influence the optimal policy is also helpful from the point of view of interpretability and robustness [Bäuerle et al., 2024].

4.2.1 The Optimality Front

We first start by defining the Optimality Front, that is the set of all optimal policies for EntRM as the risk parameter β varies over $\mathbb{R}$. We give a few informal definitions and properties giving the intuition of the concept before discussing the technical subtleties and giving more formal results better suited to our needs.

Definition 4.1 (Optimality Front (informal))**.** For any MDP, the *optimality front* is defined as $\Gamma = (\pi_k, I_k)_k$, where $(I_k)_k$ is a partition of $\mathbb{R}$ and where π_k is the optimal policy of EntRM for all risk tolerance parameters $\beta \in I_k$.

This definition is justified by the following property, formalizing the intuition that a small perturbation of the risk parameter typically does not change the optimal policy.

Proposition 4.3 (Piecewise constant optimal policies (informal))**.** The Optimality Front Γ contains a finite set of policies. Each policy in Γ is optimal on a finite union of closed intervals. In other words, the mapping $\beta \mapsto \pi^*_\beta$ is piecewise constant.

Definition 4.2 (Breakpoints (informal))**.** The *breakpoints* of the Optimality Front Γ are the values of the risk parameter β where the optimal policy changes, i.e., the endpoints of the intervals I_k.

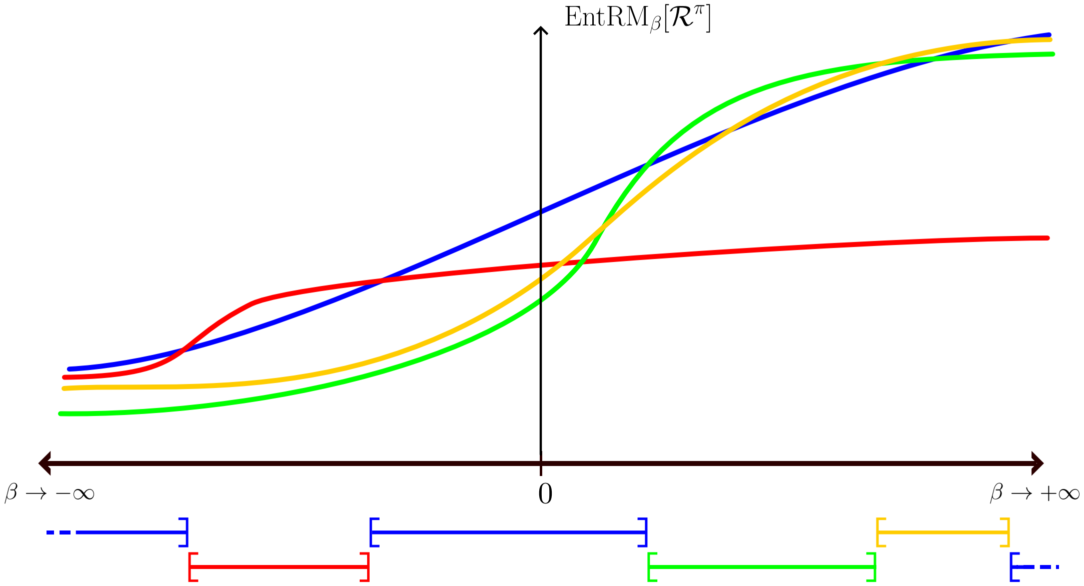

Figure 4.2: Illustration of the Optimality Front. Each color represents a different policy. The lines represent the function $\beta \mapsto \mathrm{EntRM}_\beta[\mathcal{R}^\pi]$ for each policy. The optimal policy at each β is the one with the highest value. The optimality intervals are represented as the colored intervals below.

Shortcomings of Definition 4.1 The optimality front is not well-defined in this way. A first issue is that, for a specific β, the optimal policy may not be unique. This could happen if two different policies have the same return distribution. In general, there might be different optimal policies that could lead to also different optimality intervals. For the optimality front to be well-defined, we would need uniqueness.

A second issue arises from the fact that a policy might be optimal for a unique value of β, and not an interval of length > 0. To match with the definition of *breakpoints*, we would like to ensure that the optimality intervals are always of length > 0, and that two optimality intervals may only overlap at their endpoints. For both those reasons, we introduce the following notions.

Formalizing the Optimality Front We say that two policies π_1 and π_2 are equivalent, denoted $\pi_1 \sim \pi_2$, if the associated return distributions are equal: $\pi_1 \sim \pi_2 \iff \mathcal{R}^{\pi_1} \overset{d}{=}$

$\mathcal{R}^{\pi_2}$. It is an *equivalence relation* (in the sense of mathematical binary relations) so it is possible to consider the quotient set $\bar{\Pi} = \Pi/\sim$ of equivalence classes. For $\bar{\pi} \in \bar{\Pi}$, the return distribution $R^{\bar{\pi}}$ is well-defined. Considering the equivalence classes allows to avoid the issue of having several optimal policies on any given intervals of the risk parameter, instead, all such policies are in the same equivalence class $\bar{\pi}$.

For $\bar{\pi} \in \bar{\Pi}$, we consider $O_{\bar{\pi}} = \{\beta \in \mathbb{R} \text{ s.t. } \mathrm{EntRM}_\beta[\mathcal{R}^{\bar{\pi}}] = \sup_\pi \mathrm{EntRM}_\beta[\mathcal{R}^\pi]\}$ the set of values for which $\bar{\pi}$ is optimal, and $I_{\bar{\pi}} = \overline{\mathrm{int}\, O_{\bar{\pi}}}$, i.e., the closure of its interior (so isolated points are removed).

Definition 4.3 (Optimality front). The optimality front Γ is defined as $\Gamma = (\bar{\pi}_k, I_k)_{k\in[1,K]}$ where $I_k = I_{\bar{\pi}_k}$ and $(\bar{\pi}_k)$ is the set of policy classes such that $I_k \neq \emptyset$.

With this definition, the optimality front is uniquely defined, up to permutations. Computing it amounts to computing *one* representative policy of each optimal class $\bar{\pi}_k$, and the corresponding optimality interval I_k.

Proposition 4.4. The Optimality Front is unique, and finite. Also, $\forall k, I_k$ is a finite union of closed intervals in $\mathbb{R}$ with no isolated points, and $\bigcup_k I_k = \mathbb{R}$. Finally, for $k_1 \neq k_2$, $I_{k_1} \cap I_{k_2} = \partial I_{k_1} \cap \partial I_{k_2}$.

Using this result, we can now formally define the breakpoints of the optimality front.

Definition 4.4. The breakpoints of the optimality front are defined as the finite set of points $\bigcup_{\bar{\pi}\in\bar{\Pi}} \partial I_{\bar{\pi}}$. The breakpoints are the points where the optimality intervals $I_{\bar{\pi}_k}$ meet. We denote by $\mathcal{B} = \{\beta_1, \ldots, \beta_K\}$ the set of breakpoints.

4.2.2 Some properties of the Optimality Front

The optimality front and the breakpoints are now well-defined and we can study their properties.

Lower bounding the optimality interval A first important property that we will use later is that we can give simple lower bounds on the length of the intervals of optimality. These bounds only depend on the *Generalized Advantage function*, defined as the difference in EntRM between following the optimal policy and taking a different action at a given state and time step.

Theorem 4.2. Let $\beta \in \mathbb{R}$ be such that there is a unique deterministic policy π^*_β optimizing EntRM_β. Define the Generalized Advantage function:

$$A^\pi_{h,\beta}(x,a) \;=\; \mathrm{EntRM}_\beta[\mathcal{R}^\pi_h(x)] \;-\; \mathrm{EntRM}_\beta[\mathcal{R}^\pi_h(x,a)].$$

Then, define the optimality gap as the smallest scaled advantage function over the entire MDP:

$$\Delta = \begin{cases} \frac{|\beta|}{2} \min_{h,x} \min_{a \neq \pi^*_{\beta,h}(x)} \frac{1}{H-h} A^{\pi^*_\beta}_{h,\beta}(x,a) & \text{if } \beta \neq 0 \\ 2 \min_{h,x} \min_{a \neq \pi^*_{\beta,h}(x)} \frac{1}{(H-h)^2} A^{\pi^*_\beta}_{h,\beta}(x,a) & \text{if } \beta = 0. \end{cases}$$

Then, for all $\beta' \in [\beta - \Delta, \beta + \Delta]$, the optimal policy for $\text{EntRM}_{\beta'}$ remains π^*_β.

The first bound is relevant when the risk parameter is not too small, as it scales with β. For large values of β, it balances the small optimality gap (remember that $\text{EntRM}_\beta[R^\pi] \to \operatorname{ess\,inf} R^\pi$ when $\beta \to -\infty$, so the gaps tend to 0). The degeneracy at $\beta = 0$ is circumvented by the second bound.

This theorem can be relatively useful as it allows us to estimate the size of the optimality intervals once the optimal policy is computed, without additional computational complexity.

Bounding the largest breakpoints The number of breakpoints is finite, but their values can be quite large. Giving the exact value of the largest breakpoints is difficult in general, but we can bound it. This bound depends on the return distribution of the optimal policy when $\beta \to -\infty$ (i.e., the policy optimizing the worst-case return).

Theorem 4.3. Consider $\pi_{\text{inf}} = \lim\limits_{\beta \to -\infty} \pi^*_\beta$. Consider the return $\mathcal{R}$ taking value in $x_1 \leq \ldots \leq x_n$ evenly spaced. We write $a_i(\pi) = \Pr(\mathcal{R}^{\pi_{\text{inf}}} = x_i) - \Pr(\mathcal{R}^\pi = x_i)$. Then, the lowest breakpoint β_{inf} satisfies

$$\beta_{\text{inf}} \geq \frac{-\log\left(1 + \max_{\pi \neq \pi_{\text{inf}}} \max_{i>0} \frac{|a_i(\pi)|}{|a_0(\pi)|}\right)}{|x_2 - x_1|} .$$

Similarly, consider $\pi_{\text{sup}} = \lim\limits_{\beta \to +\infty} \pi^*_\beta$. We write $b_i(\pi) = \Pr(\mathcal{R}^{\pi_{\text{sup}}} = x_i) - \Pr(\mathcal{R}^\pi = x_i)$. Then, the largest breakpoint β_{sup} satisfies

$$\beta_{\text{sup}} \leq \frac{\log\left(1 + \max_{\pi \neq \pi_{\text{sup}}} \max_{i<n} \frac{|b_i(\pi)|}{|b_n(\pi)|}\right)}{|x_2 - x_1|} .$$

This result is mostly theoretical, as computing the bound requires to know the return distributions of all possible Markov policies. However, it also shows that the largest breakpoint cannot be arbitrarily large. Another interesting point we prove is about what π_{inf} and π_{sup} optimize. We know from Section 2.3.2 that they do optimize the worst-case and best-case return respectively, but we can be more precise. For a policy π, consider the *lexicographic return* defined as the vector of probabilities of the return taking its possible values, sorted in increasing order: $L(\pi) = (P(\mathcal{R}^\pi = x_1), \ldots, P(\mathcal{R}^\pi = x_n))$, and the reverse lexicographic return defined as $L_{\text{rev}}(\pi) = (P(\mathcal{R}^\pi = x_n), \ldots, P(\mathcal{R}^\pi = x_1))$. Then $\pi_{\text{inf}} = \arg\min_\pi L(\pi)$ and $\pi_{\text{sup}} = \arg\max_\pi L_{\text{rev}}(\pi)$ (where the min and max are taken in the lexicographic sense).

Theoretical Bound on the number of breakpoints While we know that there is a finite number of optimal policies and breakpoints, the exact number is problem-dependent and hard to characterize precisely. We give the following theoretical upper bound:

Proposition 4.5. Let n be the number of possible values of R^π. The number of breakpoints B satisfies

$$B \leq n \cdot |\mathcal{A}|^{2|\mathcal{X}|H}$$

This bound is also mostly of theoretical interest, as it is very loose in practice. Yet, it shows that the number of breakpoints cannot be arbitrarily large.

Breakpoints are only local changes Breakpoints are the values of β where the optimal policy changes. What we show in practice is that the change in the optimal policy is generally only local: it only affects a single state at a single timestep.

Proposition 4.6. Consider a randomly generated MDP with reward functions and transitions $(r_t(x, a))_{t,x,a}$ and $(p_t(x'|a, x))_{t,x,a,x'}$ generated from, say, independent uniform distributions. With probability 1, for each breakpoint $\beta \in \mathcal{B}$ there is a single state-horizon pair for which the optimal action changes: if π^1 is optimal for $\beta \in [\beta_1, \beta_2]$ and π^2 is optimal for $\beta \in [\beta_2, \beta_3]$, then there exists a unique state x and time step t such that $\pi^1_t(x) \neq \pi^2_t(x)$.

This result can give an idea on how the number of breakpoints will scale in practice with the parameter of the MDP. Take for example the Cliff environment [Sutton et al., 1998] (presented in Section 2.1.1). In that MDP, there are two main tendencies depending on the risk parameter β: either the agent takes the safe path, far from the cliff, or it takes the risky path close to the cliff. In most states and actions, there will be two actions possible, depending on the value of β: go straight, or go around the cliff. This result indicates that most likely, for each state and timestep, the change of optimal action will happen in a slightly different risk parameter β and thus lead to a different breakpoint. Hence, the number of breakpoints should scale linearly with the horizon and the number of states. This analysis is in practice closer to the experimental results we obtained, compared to the exponential theoretical bound of Proposition 4.5. The illustration of the Optimality Front of the MDP is given in Section 4.4 with Figure 4.7.

4.2.3 Proofs

For each result of this section, we recall the statement before giving the proof.

Proposition 4.4 The Optimality Front is unique and finite. Moreover, for each k, I_k is a finite union of closed intervals in $\mathbb{R} \cup \{\pm\infty\}$ with no isolated points, and $\bigcup_k I_k = \mathbb{R} \cup \{\pm\infty\}$. Finally, for distinct indices $k_1 \neq k_2$, the sets intersect only at their boundaries: $I_{k_1} \cap I_{k_2} \subseteq \partial I_{k_1} \cap \partial I_{k_2}$.

Proof of Proposition 4.4. We first establish that the Optimality Front is well-defined. Let $\bar{\pi} \in \bar{\Pi}$. By definition of the equivalence relation, for any $\pi_1, \pi_2 \in \bar{\pi}$, the return distributions are identical: $\mathcal{R}^{\pi_1} \overset{d}{=} \mathcal{R}^{\pi_2}$. Hence, the objective function $\beta \mapsto \mathrm{EntRM}_\beta[\mathcal{R}^\pi]$ depends only on the class $\bar{\pi}$, not the specific representative. It follows that the optimality sets $O_{\bar{\pi}}$ and $I_{\bar{\pi}}$ are well-defined. Since the set of equivalence classes $\bar{\Pi}$ is uniquely determined by the underlying MDP, the Optimality Front $\Gamma = \{(\bar{\pi}, I_{\bar{\pi}}) \mid \bar{\pi} \in \bar{\Pi}, I_{\bar{\pi}} \neq \emptyset\}$ is unique.

We now prove the structural properties of the intervals. Recall that in a finite-horizon MDP with finite state and action spaces, the number of deterministic policies is finite. Thus, the quotient set $\bar{\Pi}$ is finite.

We restrict our analysis to $\beta \leq 0$ (the case $\beta > 0$ follows by symmetry). Optimizing $\mathrm{EntRM}_\beta[\mathcal{R}]$ for $\beta < 0$ is equivalent to maximizing the exponential utility the moment generating function $M_{\bar{\pi}}(\beta) = \mathbb{E}[e^{\beta \mathcal{R}^{\bar{\pi}}}]$.

Consider any pair of distinct classes $\bar{\pi}_1, \bar{\pi}_2 \in \bar{\Pi}$ with $\bar{\pi}_1 \neq \bar{\pi}_2$. By definition of the classes, their return distributions are distinct: $\mathcal{R}^{\bar{\pi}_1} \overset{d}{\neq} \mathcal{R}^{\bar{\pi}_2}$. The moment generating function uniquely characterizes the distribution of a bounded random variable [Durrett, 2019]. Therefore, the function $D(\beta) = M_{\bar{\pi}_1}(\beta) - M_{\bar{\pi}_2}(\beta)$ is not identically zero.

Since the returns are discrete random variables with finite number of values (because of deterministic rewards and finite horizon), $M_{\bar{\pi}}(\beta)$ is a finite sum of exponential functions and is therefore analytic on $\mathbb{R}$. Hence, the difference $D(\beta)$ is also analytic. An analytic function that is not identically zero can only have isolated zeros [Rudin, 1987].

Using Theorem 4.3, we restrict our attention to the compact interval $B_{rest} = [\beta_{\min}, 0]$. Since $D(\beta)$ has only isolated zeros, the set of points where the two curves intersect, $\{\beta \in B_{rest} \mid M_{\bar{\pi}_1}(\beta) = M_{\bar{\pi}_2}(\beta)\}$, is finite. Let $\mathcal{Z}$ be the union of all such intersection points for all pairs in $\bar{\Pi}$. Since $\bar{\Pi}$ is finite, $\mathcal{Z}$ is finite.

The set $\mathcal{Z}$ partitions B_{rest} into a finite number of open intervals. On any such interval (β_j, β_{j+1}), the functions $\{M_{\bar{\pi}}\}_{\bar{\pi} \in \bar{\Pi}}$ do not intersect. By continuity, their order is preserved throughout the interval. Thus, there exists a single class $\bar{\pi}^*$ such that $M_{\bar{\pi}^*}(\beta) < M_{\bar{\pi}'}(\beta)$ for all $\bar{\pi}' \neq \bar{\pi}^*$ and all $\beta \in (\beta_j, \beta_{j+1})$.

Hence, for any $\bar{\pi}$, the set $O_{\bar{\pi}}$ consists of a finite union of these open intervals plus potentially some points from $\mathcal{Z}$. The definition $I_{\bar{\pi}} = \overline{\operatorname{int} O_{\bar{\pi}}}$ removes isolated optimality points (where a policy might be optimal only at a crossing point) and takes the closure of the intervals. Therefore, $I_{\bar{\pi}}$ is a finite union of closed intervals.

Finally, since for every β there exists at least one optimal policy, the union of optimality sets covers the domain. The "regularization" (taking the interior and closure) ensures that intervals overlap only at their endpoints (the points in $\mathcal{Z}$), which are the breakpoints of the Optimality Front. □

Theorem 4.2 Let $\beta \in \mathbb{R}$ be such that there is a unique deterministic policy π^*_β optimizing EntRM_β. Define the Generalized Advantage function:

$$A^\pi_{h,\beta}(x,a) \;=\; \mathrm{EntRM}_\beta[\mathcal{R}^\pi_h(x)] \;-\; \mathrm{EntRM}_\beta[\mathcal{R}^\pi_h(x,a)].$$

Then, define the optimality gap as the smallest scaled advantage function over the entire MDP:

$$\Delta = \begin{cases} \frac{|\beta|}{2}\, \min_{h,x} \min_{a \neq \pi^*_{\beta,h}(x)} \frac{1}{H-h}\, A^{\pi^*_\beta}_{h,\beta}(x,a) & \text{if } \beta \neq 0 \\ 2\, \min_{h,x} \min_{a \neq \pi^*_{\beta,h}(x)} \frac{1}{(H-h)^2}\, A^{\pi^*_\beta}_{h,\beta}(x,a) & \text{if } \beta = 0. \end{cases}$$

Then, for all $\beta' \in [\beta - \Delta, \beta + \Delta]$, the optimal policy for $\mathrm{EntRM}_{\beta'}$ remains π^*_β.

We start by deriving bounds on the growth of the EntRM when the risk parameter is changed slightly.

Proposition 4.7. Let X be a random variable, $\beta \in \mathbb{R}^-$, and $0 < \varepsilon < |\beta|$. We assume X is bounded in $[r_{\min}, r_{\max}]$. Then:

$$\mathrm{EntRM}_\beta[X] \leq \mathrm{EntRM}_{\beta+\varepsilon}[X] \leq \frac{\beta}{\beta+\varepsilon}\mathrm{EntRM}_\beta[X] + \frac{\varepsilon}{\beta+\varepsilon} r_{\min}, \tag{4.7}$$

$$\frac{\beta}{\beta-\varepsilon}\mathrm{EntRM}_\beta[X] - \frac{\varepsilon}{\beta-\varepsilon} r_{\min} \leq \mathrm{EntRM}_{\beta-\varepsilon}[X] \leq \mathrm{EntRM}_\beta[X] \tag{4.8}$$

Proof. In the following, $0 < \varepsilon < |\beta|$. We write $X = \sum \mu_i \delta_{x_i}$, with $r_{\min} = \min_i x_i$.

$$\begin{aligned}
\mathrm{EntRM}_{\beta+\varepsilon}[X] &= \frac{1}{\beta+\varepsilon} \ln\left(\sum \mu_i e^{(\beta+\varepsilon)x_i}\right) \\
&= \frac{1}{\beta+\varepsilon} \ln\left(\sum \mu_i e^{\beta x_i} e^{\varepsilon x_i}\right) \\
&\leq \frac{1}{\beta+\varepsilon} \ln\left(e^{\varepsilon r_{\min}} \sum \mu_i e^{\beta x_i}\right) \quad (\text{since } \frac{1}{\beta+\varepsilon} < 0) \\
&= \frac{1}{\beta+\varepsilon} \ln\left(\sum \mu_i e^{\beta x_i}\right) + \frac{\varepsilon}{\beta+\varepsilon} r_{\min} \\
&= \frac{\beta}{\beta+\varepsilon}\mathrm{EntRM}_\beta[X] + \frac{\varepsilon}{\beta+\varepsilon} r_{\min},
\end{aligned}$$

and

$$\begin{aligned}
\mathrm{EntRM}_{\beta-\varepsilon}[X] &= \frac{1}{\beta-\varepsilon}\ln\left(\sum \mu_i e^{(\beta-\varepsilon)x_i}\right)\\
&= \frac{1}{\beta-\varepsilon}\ln\left(\sum \mu_i e^{\beta x_i} e^{-\varepsilon x_i}\right)\\
&\geq \frac{1}{\beta-\varepsilon}\ln\left(e^{-\varepsilon r_{\min}} \sum \mu_i e^{\beta x_i}\right) \quad \left(\text{since } \frac{1}{\beta-\varepsilon} < 0\right)\\
&= \frac{1}{\beta-\varepsilon}\ln\left(\sum \mu_i e^{\beta x_i}\right) - \frac{\varepsilon}{\beta-\varepsilon} r_{\min}\\
&= \frac{\beta}{\beta-\varepsilon}\mathrm{EntRM}_{\beta}[X] - \frac{\varepsilon}{\beta-\varepsilon} r_{\min}.
\end{aligned}$$

The other sides of the inequalities are obtained using the monotony of the function $\beta \mapsto \mathrm{EntRM}_{\beta}[X]$ [Ahmadi-Javid, 2012]. We assumed X to be a discrete random variable, but the proof can be extended to continuous random variables by using integrals instead of the sum. The case $\beta > 0$ is similar by symmetry, the only difference being the bound depends on $r_{\max}$ instead of $r_{\min}$. □

The next theorem is a special case of Theorem 4.4 for the case of a single state. While not necessary to prove the main result, the result is used in the algorithm *FindBreaks*, see Section 4.3.2.

Theorem 4.4 (Interval of Action Optimality)**.** Let $r_{\min}$ (resp. $r_{\max}$) be the minimum (resp. maximum) achievable reward, and $\Delta R = r_{\max} - r_{\min}$. Suppose we have actions $(a_{(i)})_i$ ordered so that

$$U_\beta^1 = \mathrm{EntRM}_\beta[R(a_{(1)})] \ > \ U_\beta^2 = \mathrm{EntRM}_\beta[R(a_{(2)})] \ \geq \ \dots \ \geq \ U_\beta^n = \mathrm{EntRM}_\beta[R(a_{(n)})].$$

In particular, $a_{(1)}$ is the unique optimal action and $a_{(2)}$ is the second-best. Define $\Delta U = U_\beta^1 - U_\beta^2$. Then:

- If $\beta \neq 0$, for all $\beta' \in \left[\beta\left(1 - \frac{\Delta U}{U_2 - r_{\min}}\right), \beta\left(1 + \frac{\Delta U}{U_1 - r_{\min}}\right)\right]$ and for all $i \geq 2$, we have
$$\mathrm{EntRM}_{\beta'}[R(a_{(1)})] \ > \ \mathrm{EntRM}_{\beta'}[R(a_{(i)})].$$
- If $\beta = 0$, then for all $\beta' \in \left[-\frac{8\,\Delta U}{\Delta R^2}, \frac{8\,\Delta U}{\Delta R^2}\right]$ and all $i \geq 2$, we have
$$\mathrm{EntRM}_{\beta'}[R(a_{(1)})] \ > \ \mathrm{EntRM}_{\beta'}[R(a_{(i)})].$$

In particular, the action $a_{(1)}$ remains strictly optimal for all β' in the specified range.

Proof of Theorem 4.4. Assume $\beta \neq 0$.By hypothesis, $U_\beta^1 > U_\beta^2$. We aim to show that if β' remains within the specified range around β, action $a_{(1)}$ remains strictly optimal for $\mathrm{EntRM}_{\beta'}$.

Case $\beta' > \beta, \beta \neq 0$. Write $\beta' = \beta + \varepsilon$ with $\varepsilon > 0$. Assume

$$\varepsilon < \beta \, \frac{\Delta U}{r_{\min} - U_\beta^1}.$$

From the previously established bounds (see Proposition 4.7), we know

$$U_\beta^1 \;\leq\; \mathrm{EntRM}_{\beta+\varepsilon}[R(a_{(1)})] \;=\; U_{\beta'}^1,$$

and

$$\mathrm{EntRM}_{\beta+\varepsilon}[R(a_{(2)})] \;=\; U_{\beta'}^2 \;\leq\; \frac{\beta}{\beta+\varepsilon} U_\beta^2 \;+\; \frac{\varepsilon}{\beta+\varepsilon} r_{\min}.$$

Thus, showing

$$U_\beta^1 \;>\; \frac{\beta}{\beta+\varepsilon} U_\beta^2 \;+\; \frac{\varepsilon}{\beta+\varepsilon} r_{\min}$$

implies $U_{\beta'}^1 > U_{\beta'}^2$. Rewriting, we get

$$\beta(U_\beta^1 - U_\beta^2) \;<\; \varepsilon\,(r_{\min} - U_\beta^1),$$

i.e.

$$\beta \, \frac{\Delta U}{r_{\min} - U_\beta^1} \;>\; \varepsilon.$$

The changes of sign in the inequality are due to the fact that $\beta + \varepsilon < 0$ and $U_\beta^1 > r_{\min}$.This is exactly our assumption on ε. Hence $a_{(1)}$ remains strictly better than $a_{(2)}$ at $\beta' = \beta + \varepsilon$, and by extension, better than all other actions.

Case $\beta' < \beta, \beta \neq 0$. The similar argument holds using the two inequalities

$$U_\beta^2 \;\geq\; \mathrm{EntRM}_{\beta-\varepsilon}[R(a_{(2)})] \;=\; U_{\beta'}^2,$$

and

$$\mathrm{EntRM}_{\beta-\varepsilon}[R(a_{(1)})] \;=\; U_{\beta'}^1 \;\geq\; \frac{\beta}{\beta-\varepsilon} U_\beta^1 \;-\; \frac{\varepsilon}{\beta-\varepsilon} r_{\min}.$$

Case $\beta = 0$. This second part of Theorem 4.4, when $\beta = 0$, uses *Hoeffding's lemma* (see e.g. Massart [2007]):

$$\forall \lambda \in \mathbb{R}, \quad \mathbb{E}[\exp(\lambda X)] \leq \exp\left(\lambda \mathbb{E}[X] + \frac{\lambda^2 \Delta R^2}{8}\right). \tag{4.9}$$

Hoeffding's lemma yields

$$\mathrm{EntRM}_\beta[X] \geq \mathbb{E}[X] - \beta \frac{\Delta R^2}{8}. \tag{4.10}$$

We can then proceed similarly as the previous proof. Consider $0 > \beta' > \frac{8\Delta U}{\Delta R^2}$. Using the previous equation and the fact that $U^1_\beta \geq E[R(a_1)]$, we have

$$U^1_{\beta'} - U^2_{\beta'} \geq \mathbb{E}[R(a_1)] - \mathbb{E}[R(a_2)] - \beta' \frac{\Delta R^2}{8} \geq \Delta U - \frac{8\Delta U}{\Delta R^2}\frac{\Delta R^2}{8} = 0 \; ,$$

hence $U^1_\beta > U^2_\beta$. This concludes the proof of Theorem 4.4. □

we can now prove Theorem 4.2. We use the same principle as in the proof of Theorem 4.4 and the fact that the optimal policy is always greedy with respect to itself to show that the optimal policy remains the same. The inequalities from Proposition 4.7 are adapted by replacing $r_{\min}$ by $h - H$ (for timestep h, the return is in $[h-H, H-h]$ as the reward is bounded by $[-1,1]$ at each step). The value $r_{\min} - U^1_\beta$ is replaced by its upper bound $2(H-h)$ to obtain a symetric bound and simplify the formula.

Proof. (of Theorem 4.2) Let us address the case where $\beta < 0$ and $\varepsilon > 0$. By the principle of optimality, we have:

$$\forall h, x, \quad \pi^*_{h,\beta}(x) = \arg\max_a \mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,a)] \; .$$

Let $\varepsilon < \Delta$ (with $\varepsilon < |\beta|$). We then obtain the inequalities:

$$\begin{aligned}
\forall h, x, a, \quad & \mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,a)] \;\leq\; \mathrm{EntRM}_{\beta+\varepsilon}[\mathcal{R}^{\pi^*}_h(x,a)] \\
\text{and} \quad & \mathrm{EntRM}_{\beta+\varepsilon}[\mathcal{R}^{\pi^*}_h(x,a)] \;\leq\; \frac{\beta}{\beta+\varepsilon}\mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,a)] + \frac{\varepsilon}{\beta+\varepsilon}(H-h) \; .
\end{aligned}$$

By the definition of ε, and following the same method as for Theorem 4.4 we get $\forall h, x, \forall a \neq \pi^*_h(x)$,

$$\frac{\beta}{\beta+\varepsilon}\mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,a)] + \frac{\varepsilon}{\beta+\varepsilon}(H-h) \;\leq\; \mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,\pi^*_h(x))].$$

Therefore,

$$\begin{aligned}
\mathrm{EntRM}_{\beta'}[\mathcal{R}^{\pi^*}_h(x,a)] &\leq \frac{\beta}{\beta+\varepsilon}\mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,a)] + \frac{\varepsilon}{\beta+\varepsilon}(H-h) \\
&\leq \mathrm{EntRM}_\beta[\mathcal{R}^{\pi^*}_h(x,\pi^*_h(x))] \\
&\leq \mathrm{EntRM}_{\beta'}[\mathcal{R}^{\pi^*}_h(x,\pi^*_h(x))] \; .
\end{aligned}$$

Thus, π^* remains greedy with respect to itself and remains the optimal policy.

The cases $\beta = 0$ and $\varepsilon < 0$ follow similarly, using the associated inequalities. □

Theorem 4.3 Consider $\pi_{\inf} = \lim_{\beta\to-\infty}\pi^*_\beta$. Consider the return $\mathcal{R}$ taking value in $x_1 \leq \ldots \leq x_n$ evenly spaced. We write $a_i(\pi) = P(R^{\pi_{\inf}} = x_i) - \Pr(R^\pi = x_i)$. Then, the lowest breakpoint $\beta_{\inf}$ verifies

$$\beta_{\inf} \geq \frac{-\log\left(1 + \max_{\pi\neq\pi_{\inf}} \max_{i>0} \frac{|a_i(\pi)|}{|a_0(\pi)|}\right)}{\min_{i\neq j}|x_i - x_j|} \; .$$

Proof. (of Theorem 4.3)

We write $\Delta R = R_{\max} - R_{\min}$. This mean that the values of the return are of the form $R_i = R_{\min} + i\frac{\Delta R}{n}$ for $i \in \{0, \ldots, n\}$.

With this natural assumption, the problem of finding β such that,

$$\mathrm{EntRM}_\beta[R^\pi] = \mathrm{EntRM}_\beta[R^{\pi'}]$$

can be transformed into a problem of finding the roots of a polynomial.

$$\begin{aligned}
\mathrm{EntRM}_\beta[R^\pi] = \mathrm{EntRM}_\beta[R^{\pi'}] &\implies \mathbb{E}[\exp(\beta R^\pi)] = \mathbb{E}[\exp(\beta R^{\pi'})] \\
&\iff \sum_{i=0}^{n} \mu_i \exp(\beta R_i) = \sum_{i=0}^{n} \mu'_i \exp(\beta R'_i) \\
&\iff \sum_{i=0}^{n} a_i \exp(\beta R_i) = 0 \\
&\iff \sum_{i=0}^{n} a_i \exp\Big(\beta\, i\, \frac{\Delta R}{n}\Big) = 0 \\
&\iff \sum_{i=0}^{n} a_i\, X^{-i} = 0\,. \\
&\iff \sum_{i=0}^{n} a_{n-i}\, X^{i} = 0\,.
\end{aligned}$$

Where $X = \exp\Big(-\beta\, \frac{\Delta R}{n}\Big)$. The first implication is not an equivalence because of the case $\beta = 0$, where the exponential form (rhs) is always equal to 1 and thus the equality is always verified. The last implication is verified by multiplying by X^n because of that same fact that 0 is already a root of the equation.

Hence, if X is a solution, β verifies $\beta = -\frac{\log(X)}{\frac{\Delta R}{n}}$.

Cauchy's bound on the size of the largest polynomial root [Cauchy, 1828] claims that the largest root verifies

$$1 + \max_{i>0} \frac{|a_i|}{|a_0|}.$$

The lowest breakpoints corresponds to the largest breakpoint solution between the $\pi_{\inf}$ and any other policy. Hence, the lowest breakpoint verifies

$$\beta_{\inf} \geq \frac{-\log\left(1 + \max_{\pi \neq \pi_{\inf}} \max_i \frac{|a_i(\pi)|}{|a_0(\pi)|}\right)}{\frac{\Delta R}{n}}\,.$$

The case for the highest breakpoint is done similarly. □

This proof uses Cauchy's bound on the size of the largest polynomial root as it is simple to write and an efficient bound. Tighter bound have been developed in the litterature that could also be used here. See Akritas et al. [2008] for example.

Using this polynomial formulation, it is also possible to derive a theoretical bound on the smallest distance between breakpoints. Mignotte's separation bound [Collins, 2001] gives a lower bound on the distance between two roots of the polynomial, as a function of the coefficient of such polynomial. This bound can then be transfered to a bound on the distance between breakpoints. This kind of bound could be useful to choose the necessary precision for the computation of the breakpoints, but are intractable to compute and the obtained values are too small to be relevant in practice.

Alternative bound on the largest breakpoints We provide here an alternative bound that does not assume any structure on the rewards.

Proof. **First case, $\beta_{\max}$:** Consider the finite space of possible rewards $X = \{x_1 > \cdots > x_k\}$ of size k. Consider the n actions $a_1, \dots, a_n$ associated to random rewards $R_1, \dots, R_n$. We write $\mu^i = \sum_{j=1}^{k} \mu^i_j \delta_{x_i}$ the distributions of those rewards.

We reorder the action/distribution with $\mu^1 > \cdots > \mu^n$ where we write $\mu^i > \mu^j$ if $(\mu^i_1, \dots, \mu^i_k) > (\mu^j_1, \dots, \mu^j_k)$ for the lexical order. Here we assume every distribution to be distinct pairwise. Note that this reordering corresponds to the order of the best distributions for $\beta \to \infty$. Let $l = \min_i \mu^1_i \neq \mu^2_i$. We hence have $\sum_{j=1}^{l-1} \mu^1_j e^{\beta x_j} = \sum_{j=1}^{l-1} \mu^2_j e^{\beta x_j}$.

For all $\beta > 0$, we write

$$E_\beta(\mu^1) = \sum_{j=1}^{k} \mu^1_j e^{\beta x_j} > \sum_{j=1}^{l} \mu^1_j e^{\beta x_j}$$

$$E_\beta(\mu^2) = \sum_{j=1}^{k} \mu^2_j e^{\beta x_i} < \sum_{j=1}^{l} \mu^2_j e^{\beta x_j} + \left(\sum_{j=l+1}^{k} \mu^2_j \right) e^{\beta x_{l+1}}$$

We can now show that

$$\begin{aligned} E_\beta(\mu^1) > E_\beta(\mu^2) &\Longleftarrow \sum_{j=1}^{l} \mu_j^1 e^{\beta x_j} > \sum_{j=1}^{l} \mu_j^2 e^{\beta x_j} + \left(\sum_{j=l+1}^{k} \mu_j^2 \right) e^{\beta x_{l+1}} \\ &\Longleftrightarrow \mu_l^1 e^{\beta x_l} > \mu_l^2 e^{\beta x_l} + \left(\sum_{j=l+1}^{k} \mu_j^2 \right) e^{\beta x_{l+1}} \\ &\Longleftrightarrow (\mu_l^1 - \mu_l^2) e^{\beta x_l} > \left(\sum_{j=l+1}^{k} \mu_j^2 \right) e^{\beta x_{l+1}} \\ &\Longleftrightarrow \log(\mu_l^1 - \mu_l^2) + \beta x_l > \log \left(\sum_{j=l+1}^{k} \mu_j^2 \right) + \beta x_{l+1} \\ &\Longleftrightarrow \beta > \frac{\log \left(\sum_{j=l+1}^{k} \mu_j^2 \right) - \log(\mu_l^1 - \mu_l^2)}{x_l - x_{l+1}} = \beta_{\max} \end{aligned}$$

By definition of our re-ordering, for other distributions $\mu^i,\ i \geq 2$, we also have

$$E_\beta(\mu^i) = \sum_{j=1}^{k} \mu_j^i e^{\beta x_i} < \sum_{j=1}^{l} \mu_j^2 e^{\beta x_j} + \left(\sum_{j=l+1}^{k} \mu_j^2 \right) e^{\beta x_{l+1}}$$

Hence,

$$\forall \beta > \beta_{\max},\ \forall i \geq 2, \quad E_\beta(\mu^1) > E_\beta(\mu^i) \tag{4.11}$$

and a_1 is the single optimal action for $\beta > \beta_{\max}$

Second case, $\beta_{\min}$: The reasoning is the same, but a few modifications need to be made. We consider the reordering of the actions such that $\mu^i > \mu^j \Leftrightarrow (\mu_k^i, \dots, \mu_1^i) < (\mu_k^j, \dots, \mu_1^j)$. Remark the change of order to adapt to the fact thac, for negative values of β, the EntRM tries to minimize the probabilities of the lowest rewards. We then consider $l = \max_i \mu_i^1 \neq \mu_i^2$. We hence have $\sum_{j=1}^{l} \mu_j^1 e^{\beta x_j} = \sum_{j=1}^{l} \mu_j^2 e^{\beta x_j}$. Computation is then similar, and we obtain

$$\beta_{\min} = -\frac{\log \left(\sum_{j=l+1}^{k} \mu_j^2 \right) - \log(\mu_l^1 - \mu_l^2)}{x_l - x_{l+1}} \tag{4.12}$$

□

Proposition 4.5 Let n the number of possible values of R^π. The number of breakpoints B verifies

$$B \leq n \cdot |\mathcal{A}|^{2|X|H}$$

Proof. (of Proposition 4.5)

We first consider this lemma, on the number of roots of an exponential sum.

Lemma 4.5 ([Tossavainen, 2007])**.** Let $f_n(x) = \sum_0^n b_i k_i^x$, $b_i \in \mathbb{R}, k_i > 0$. Then f_n has at most $n-1$ roots.

This lemma allows for bounding the number of values of the risk parameter β for which two different policies have the same entropic risk.

Proposition 4.8. Consider two distributions $\mu_1 \neq \mu_2$ with support on $X = \{x_1, \dots, x_n\} \subset \mathbb{R}$ of size n. Consider $X_1 \sim \mu_1, X_2 \sim \mu_2$ random variables following those distributions. Consider $\mathcal{B} = \{\beta \in \mathbb{R} \mid \mathrm{EntRM}_\beta(X_1) = \mathrm{EntRM}_\beta(X_2)\}$. Then:

$$|\mathcal{B}| \leq n-1$$

The proposition is straightforward by considering the exponential form of the EntRM, which is a sum of exponentials.

Finally we conclude by considering that there are $|\mathcal{A}|^{|X|H}$ deterministic markovian policies for a specific MDP, and thus $|\mathcal{A}|^{2|X|H}$ pairs of policies. The breakpoints are values for which the entropic risk of two policies is equal, and thus are included in the union of the "breakpoints" of all pairs of policies. By Proposition 4.8, the number of breakpoints is at most $(n-1)|\mathcal{A}|^{2|X|H}$. □

This proposition hat no reason to be tight in general. In the conclusion we consider the number of point of equality between any two policies, but those points cannot all be breakpoints. Only the points where one of the two policy is optimal count as breakpoints. This problem is reduced to finding the number of components of the function $h(\beta) = \max\{f_\pi(\beta)\}_{\pi \in \Pi}$, where $f_\pi(\beta) = \mathrm{EntRM}_\beta[R^\pi]$. This combinatorial problem has been studied before (see Lemma 2.4 in Atallah [1985]), but to the best of our knowledge, no better explicit formula has been found.

Proposition 4.6 Consider a random MDP with reward functions and transitions $(r_h(x,a))_{h,x,a}$ and $(p_h(x|a,x'))_{h,x,a,x'}$ generated from, say, independent uniform distributions. With probability 1, for each breakpoint $\beta \in \mathcal{B}$ there is a single state-horizon pair for which the optimal action changes: if π^1 is optimal for $\beta \in [\beta_1, \beta_2]$ and π^2 is optimal for $\beta \in [\beta_2, \beta_3]$, then there exists a unique state x and time step t such that $\pi^1_h(x) \neq \pi^2_h(x)$. We start with a lemma:

Lemma 4.6. Let Y be a random variable with continuous law. Let $X_1, \dots, X_n$ be random variables with Y independant from $(X_1, \dots, X_n)$. Then,

$$\Pr(Y = f(X_1, \dots, X_n) \mid X_1, \dots, X_n) = 0$$

Proof. This is a special case of Exercise 2.1.5 in Durrett [2019]. It comes down to writing the definition of conditional probabilities and computing the integrals with Fubini's theorem. □

Proof of Proposition 4.6. Let π be a policy. Assume the probability transitions are fixed and only the rewards are random. We consider, for $x, h, a, a' \in \mathcal{X} \times [H] \times \mathcal{A} \times \mathcal{A}$ the set $\mathcal{B}^{x,h}_{a,a'}$ of parameters β such that $\mathrm{EntRM}_\beta[R^\pi_h(x,a)] = \mathrm{EntRM}_\beta[R^\pi_h(x,a')]$. We aim to show that the sets $(\mathcal{B}^{x,h}_{a,a'})_{x,h,a,a'}$ have no element in common pairwise, with probability 1.

Consider orders on both $\mathcal{X}$ and $\mathcal{A}$, and consider the associated lexigocraphic order of $[H,1] \times \mathcal{X} \times \mathcal{A}$. We proceed by induction on this order.

Case $t = H$. Let $x_1, x_2 \in \mathcal{X}$, $a_1, a_1', a_2, a_2' \in \mathcal{A}$, with $(x_1, a_1, a_1') \neq (x_2, a_2, a_2')$ (the triplets are different, but some elements of the triplets can be equal). Consider known $\mathcal{B}^{x_1,H}_{a_1,a_1'}$.

$$\begin{aligned}
\forall \beta \in \mathbb{R}, \quad &\Pr\left(\mathrm{EntRM}_\beta[R^\pi_H(x_2,a_2)] = \mathrm{EntRM}_\beta[R^\pi_H(x_2,a_2')] \;\middle|\; r_H(x_1,a_1), r_H(x_1,a_1')\right) \\
&= \Pr\left(r_H(x_2,a_2) = r_H(x_2,a_2') \;\middle|\; r_H(x_1,a_1), r_H(x_1,a_1')\right) \\
&= 0.
\end{aligned}$$

Because $r_H(x_2,a_2)$ and $r_H(x_2,a_2')$ are continuous random variables and at least one of the two is not conditioned on, the probability that they are equal is zero according to Lemma 4.6. It is in particular true for all $\beta \in \mathcal{B}^{x_1,H}_{a_1,a_1'}$. Using the union bound, we get that $\mathcal{B}^{x_1,H}_{a_1,a_1'}$ and $\mathcal{B}^{x_2,H}_{a_2,a_2'}$ have no element in common with probability 1. By considering elements in order and conditioning on the previously *observed* rewards, the induction is verified for $t = H$.

Case $t < H$. Let $u = h_0, x_0, a_0$. Consider all breakpoints observed before

$$\mathcal{B} = \bigcup_{\substack{(h,x,a)<u \\ (h,x,a')\leq u}} \mathcal{B}^{x,h}_{a,a'}.$$

By induction, all of them are disjoint pairwise with probability 1.

$$\begin{aligned}
&\forall \beta \in \mathcal{B}, \quad \Pr\left(\mathrm{EntRM}_\beta[R^\pi_{h_0}(x_0,a_0)] = \mathrm{EntRM}_\beta[R^\pi_{h_0}(x_0,a'_0)] \;\middle|(r_h(x,a))_{(h,x,a)\le u}\right)\\
&\quad = \Pr\Big(r_{h_0}(x_0,a_0) + \mathrm{EntRM}_\beta[R^\pi_{h_0+1}(X_0)] = r_{h_0}(x_0,a'_0) + \mathrm{EntRM}_\beta[R^\pi_{h_0+1}(X'_0)]\\
&\quad \Big|\; (r_h(x,a))_{(h,x,a)\le u}\Big)\\
&\quad = \Pr\left(r_{h_0}(x_0,a_0) = f\left((r_h(x,a))_{(h,x,a)\le u}\right) \;\middle|\; (r_h(x,a))_{(h,x,a)\le u}\right)\\
&\quad = 0,
\end{aligned}$$

where

$$f\left((r_h(x,a))_{(h,x,a)\le u}\right) = r_{h_0}(x_0,a'_0) + \mathrm{EntRM}_\beta[R^\pi_{h_0+1}(X'_0)] - \mathrm{EntRM}_\beta[R^\pi_{h_0+1}(X_0)]$$

and using Lemma 4.6. The induction is then proved.

As there is a finite number of policy, this result remains when considering the set of all policies. To conclude, notice that a breakpoint β_0 is a value of risk parameter for which two different action a_1, a_2 have the same expected return for some state x and timestep h. Hence, $\beta_0 \in \mathcal{B}^{x,h}_{a_1,a_2}$. With probability 1, those set are pairwise disjoint, meaning a single action changes in β_0. As this proof works for any value of the probability transitions, the result also remains for p random. □

4.3 Computing the Optimality Front

The first step towards computing the Optimality Front is to *find the breakpoints.* We first show that a direct approach is not feasible, but instead we can exploit Theorem 4.2. Then, we present our algorithm, *Distributional Optimality Front Iteration* (DOLFIN), based on (distributional) policy optimization.

4.3.1 Exact Computation of the Breakpoints

Breakpoints mark the transition between distinct intervals of optimality. According to Proposition 4.4, at these transition points, there must be at least two distinct policies that are simultaneously optimal. This condition yields a system of equations characterizing the breakpoints.

Proposition 4.9. Let π^1 and π^2 be optimal policies on the adjacent intervals $[\beta_1, \beta_b]$ and $[\beta_b, \beta_2]$. Then, the breakpoint β_b satisfies:

$$\forall h, x, \quad \mathrm{EntRM}_{\beta_b}[\mathcal{R}^{\pi^1}_h(x)] \;=\; \mathrm{EntRM}_{\beta_b}[\mathcal{R}^{\pi^2}_h(x)]\ .$$

Since the policies π^1 and π^2 generally differ in only a single state-horizon pair (see Proposition 4.6), the equality holds trivially for most h, x. For the states where the policies differ, one could theoretically compute the breakpoint β_b by solving the corresponding equation. However, we argue that this approach is computationally intractable in practice due to several challenges outlined below.

Evaluation Difficulty The first issue is that there is no easy way to compute the function $\beta \mapsto \mathrm{EntRM}[\mathcal{R}_h^\pi(x, \pi(x))]$ without having access to the closed-form distribution of the return for this policy, or to perform a Policy Evaluation step for each candidate value of β. In both cases, this quickly becomes computationally expensive.

Combinatorial Difficulty A second challenge is that the system of equations in Proposition 4.9 assumes we already know which two policies are adjacent on the optimality front. Suppose we know the policies π_1, π_2 are optimal for some β_1 and β_2, and we have access to the two functions $\beta \mapsto \mathrm{EntRM}[\mathcal{R}_h^{\pi^i}(x)]$ for $i = 1, 2$. Solving the equations with π_1 and π_2 outputs a risk parameter β_b for which one policy becomes superior to the other. However, there could also be a third (or more) policy π_3 which is optimal for some values of β between β_1 and β_2 (see Figure 4.3). Computing the optimality front would require computing the breakpoints between π_1 and π_3, and between π_3 and π_2. To correctly identify the true breakpoint, one would theoretically need to compute

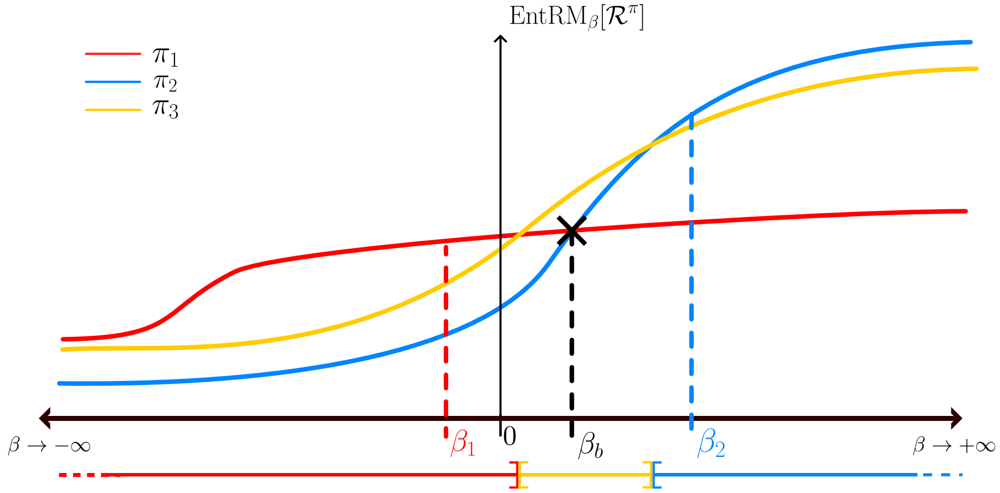

Figure 4.3: Illustration of the issues arising when trying to compute the breakpoints directly from the equations. Here, three policies π_1, π_2, π_3 are optimal for different intervals of β. Knowing only the optimal policies in β_1 and β_2 and trying to compute the intersection between them would output β_b, which is not a breakpoint because π_3 is optimal in between π_1 and π_2. The breakpoints are the intersections between π_1 and π_3 and between π_2 and π_3.

the intersection of π^1 with all other possible policies $\pi \in \Pi_{\mathrm{MD}}$ and select the smallest solution $\beta > \beta_1$. The complexity of considering every single policy makes this direct approach infeasible for all but the simplest MDPs.

Algebraic Difficulty Finally, the equations involve sums of exponential functions (Dirichlet polynomials), which, to the best of our knowledge, lack closed-form analytic solutions or efficient numerical solvers. Identifying the correct breakpoint requires finding *all* roots of these functions, a task complicated by the fact that the number of intersections is generally difficult to bound a priori (see Figure 4.2 for policies having several intersections). While efficient numerical methods exist for regular polynomials [Kobel et al., 2016], and Dirichlet polynomials can be reduced to regular polynomials via a non-linear transformation under specific assumptions (see the proof of Theorem 4.3), this transformation can introduce significant approximation errors when mapping the solutions back to the original problem.

4.3.2 Interval-Based Approximation for Action Selection

Instead, we propose to exploit Theorem 4.2 to build an incremental approach that finds the breakpoints one after the other. We call this procedure *FindBreaks*. Here the pseudocode of *FindBreaks* is designed to handle only non-positive values of the risk parameter. In practice, it also works with positive values by adding the optimality gaps instead of subtracting them. We study first the simpler case of a single state with multiple actions (a_i) and stochastic rewards $R(a_i)$ before expanding to the full MDP case.

Theorem 4.2[4] implies that evaluating the Entropic risk at a given point allows us to identify an interval of β values over which the optimal policy does not change. This fact can be utilized to "jump" over β values. The process as follows. At every given state and timestep, assume the Q-value for each action $(\mathrm{EntRM}[R(x,a)])_{a\in\mathcal{A}}$ is known. Start with $\beta = 0$ and iterate the following steps:

1. Evaluate the Generalized Advantage function and the optimality gaps (see Theorem 4.2).

2. Use the optimality gaps to get a lower bound $\beta - \Delta$ on the next breakpoint, and "jump": $\beta \leftarrow \beta - \Delta$.

At some point, when getting close to a breakpoint, the increments Δ will get closer to 0. Then, use a minimal increment ε until the optimal policy changes.

[4] In practice, the values of the increments Δ will actually come from Theorem 4.4 instead, which provides slightly better bounds.

Algorithm 9 FindBreaks: Computing all optimal actions

```
Require: Precision $\varepsilon > 0$, random rewards $(R(a_i))_i$, interval $I$.
Compute $a^* = \arg\max_a \mathrm{EntRM}_0(R(a))$ {Initial optimal action}
Compute $A = \min_{a \neq a^*} \mathrm{EntRM}_0(R(a^*)) - \mathrm{EntRM}_0(R(a))$, $\Delta R = r_{\max} - r_{\min}$ {Advantage and reward range}
Initialize $\beta = -\frac{8A}{\Delta R^2}$, $\beta_{\mathrm{old}} = 0$ {Initial parameter values}
Initialize $\beta_\ell = 0$, $a_{\beta_{\mathrm{old}}} = a^*$
while $\beta_{\min} < \beta < \beta_{\max}$ do
    Compute $\mathrm{EntRM}_\beta(R(a))$ for each $a$ {Evaluate risks for all actions}
    $a_\beta = \arg\max_a \mathrm{EntRM}_\beta(R(a))$ {Best action at $\beta$}
    if $a_\beta \neq a_{\beta_{\mathrm{old}}}$ then
        Add $[\beta_\ell, \beta]$ to $\mathcal{I}$ and $a_\beta$ to $\Pi^*$ {Update intervals and front}
        $\beta_\ell \leftarrow \beta$ {Update lower bound of next interval}
    end if
    $a_{\beta_{\mathrm{old}}} \leftarrow a_\beta$ {Update last optimal action}
    $\Delta U = \min_{a \neq a_\beta}(\mathrm{EntRM}_\beta(R(a_\beta)) - \mathrm{EntRM}_\beta(R(a)))$ {Smallest optimality gap}
    $\beta \leftarrow \beta - \max\{\beta\frac{\Delta U}{\Delta R}, \varepsilon\}$ {Decrease $\beta$ for negative values}
end while
Add $[\beta_{\min}, \beta_\ell]$ to $\mathcal{I}$ {Add the last interval}
Ensure: Optimality Front $\Gamma = (\Pi^*, \mathcal{I})$
```

4.3.3 Recursive Construction for MDPs

The *FindBreaks* procedure can also be adapted to MDPs. Instead of computing the minimum advantage function on a single state, we have to consider all the advantage functions for every state and timestep (see Theorem 4.2). This new approach already offers some advantages over to the grid search method of Hau et al. [2023b]. In their approach, they use an accuracy parameter ε and the number of grid points computed scales with $O\left(\frac{\log(1/\varepsilon)}{\varepsilon^2}\right)$. In contrast, our method gives fixed intervals that do not depend on any parameter. Hence, for very small values of ε, combining it with our method can significantly improve the complexity by removing redundant computations (see Section 4.4.2.3 for a practical example).

However, in large MDPs, this method can lead to some performance issues. There may be states where two different actions lead to almost indistinguishable return distributions. In this case, the advantage function and thus the optimality gap are close to 0, and the increments Δ become very small. Even if for all other states the advantage function is large, the algorithm only considers the smallest optimality gap over the entire MDP. This leads to very small increments and a very long computation time.

Recursive Computation of Breakpoints This issue can be mitigated by benefiting from the MDP structure. Just as optimal policies are computed recursively in Dynamic Programming, breakpoints can be propagated backward. Specifically, we can process each state separately and then merge the breakpoints across states. This can be seen as applying *FindBreaks* recursively to single states, where the reward for each action $r(x, a)$ is the return distribution of this action-value $\mathcal{R}_h(x, a)$.

However, this return distribution depends on the optimal policy at the next timestep, π^*_{h+1}, which itself varies with β. $\mathcal{R}_h(x, a)$ remains fixed only within the optimality intervals of π^*_{h+1}. To address this, we must apply *FindBreaks* separately within each interval determined by the breakpoints at timestep $h + 1$. On any such interval, the future optimal policy is constant, ensuring that $\mathcal{R}_h(x, a)$ is a fixed distribution. This logic leads to the following recursive property of the breakpoints.

Proposition 4.10. Let $\mathcal{B}^h$ and Γ^h be, respectively, the set of breakpoints and the optimality front when the MDP starts at timestep h. Let $\mathcal{B}\left((R_i)_i, I\right)$ be the set of breakpoints for the MDP with a single state and reward distributions $(R_i)_i$ (which are not assumed to be deterministic).

$$\mathcal{B}^h = \mathcal{B}^{h+1} \cup \left(\bigcup_{x\in\mathcal{X}, (\pi_k^h, I_k^h)\in\Gamma^h} \mathcal{B}\left([\mathcal{R}_h^{\pi_k^h}(x, a)]_a, I_k^h \right) \right)$$

This may not be the optimal way to compute the breakpoints but it exploits all the structure of the problem: both the regularity of the exponential functions and the recursive properties of the MDP optimization allow speeding up the process. The general question of characterizing optimality for this problem is a challenging open problem we leave for future work. Our final risk-sensitive optimization algorithm below is fully modular and could integrate any other breakpoint-search algorithm.

Distributional Planning Another key insight is to store in memory the distribution of the return recursively. While not strictly necessary, it offers several benefits. First, it simplifies the recursive computation of breakpoints: having direct access to the distribution of action-value returns $\eta_h(x, a) = \mathcal{L}(\mathcal{R}_h(x, a))$ enables the direct computation of the EntRM for any β, without having to recompute $\mathrm{EntRM}_\beta[\mathcal{R}_h(x, a)]$ for every β value recursively. Second, having direct access to the distribution of returns for every policy in the Optimality Front allows us to evaluate directly any risk measure ρ on these policies, as required for Generalized Policy Improvement.

DOLFIN Algorithm Combining insights from Dynamic Programming, the approximation of Optimality Intervals and the use of return distributions, we derive the following algorithm to compute the Optimality Front up to a desired accuracy.

Algorithm 10 DOLFIN – **D**istributional **O**ptima**l**ity **F**ront **I**teratio**n**

```
Require: Precision $\varepsilon \in (0,1)$; MDP $\mathcal{M}(\mathcal{X}, \mathcal{A}, p, r, H)$ parameters.
Select lower bound $\beta_{\min}$ {Computed or handpicked}
$\mathcal{I}_H \leftarrow [\beta_{\min}, 0]$ {Starting interval}
$\nu_H(x) \leftarrow \delta_0$ {Optimal return distribution at timestep $H$}
for $h = H$ downto $0$ do
    for all $x \in \mathcal{X}$ do
        for all $I \in \mathcal{I}_h$ do
            $\eta_h^I(x,a) \leftarrow \sum_{x'} p(x' \mid x,a)\, \tau_{r(x,a,x')} \nu_{h+1}^I(x')$ {Return distributions}
            $\{\mathcal{J}, (a_j^*)_{j \in \mathcal{J}}\} \leftarrow$ FindBreaks$(\varepsilon, (\eta_h^I(x,a))_a, I)$ {Apply Algorithm 9 on $(\eta_h^I(x,a))_a$ as reward distributions}
            for all $j \in \mathcal{J}$ do
                Add $j$ to $\mathcal{I}_{h-1}$ {Update intervals for next timestep}
                $\nu_h^j(x) \leftarrow \eta_h^I(x, a_j^*)$ {Store optimal return distribution}
            end for
        end for
    end for
end for
Ensure: $\Gamma = (\pi_k, I_k)_k$, $(\eta_0^k)_k$ {Optimality Front, distributions}
```

DOLFIN outputs the Optimality Front Γ as well as the return distributions of each optimal policy and it remains to solve (4.6): $\max_k \rho(\pi_k)$, following the Generalized Policy Improvement principle discussed above. In the experimental section below, we simply call this combined optimization the *Optimality Front* method.

About the number of iterations Let B be the number of breakpoints in the optimality front of the MDP, the number of calls to FindBreaks is bounded by $\Theta(|\mathcal{X}|HB)$ and thus heavily depends on the number of policies in the Optimality Front. For a low number, only a few calls will be made and only a few Q-value evaluations will have to be computed. Using the empirical observations on the number of breakpoints (see Section 4.4.1), the number of calls to FindBreaks can be estimated to be in the order of $(|\mathcal{X}|H)^2$. The total complexity ultimately depends on the complexity of FindBreaks. An empirical study of our implementation can also be found in Section 4.4.1. In practice we observe that FindBreaks runs in $O(|\mathcal{A}|f(1/\varepsilon))$ time, where ε is the minimum optimality gap, with some sublinear f.

About the cost of distributional planning The algorithm presented here uses distributions. Computing whole distributions of returns can be very costly, as the support of the distribution can grow exponentially with the horizon. It would also not improve on the complexity of optimizing risk measures such as VaR for which the bottleneck is the need for whole distributions. However, this algorithm still presents several benefits: first, it is modular and can be adapted to any other method of computing the breakpoints. Even without distributions, it improves on previous methods such as [Hau et al., 2023b], as argued earlier. Second, the distributional approach is well-studied and approximation techniques can be used to bound this complexity with limited losses (see Section 2.2.3 and Section 3.1.2). Third, when scaling to large problems with function approximation, the cost of computing distributions becomes negligible as many state-of-the-art algorithms already use it [Schwarzer et al., 2023, Wang et al., 2024, Hafner et al., 2023]. Finally, all the complexity relies on the computation of the Optimality Front, which can be done once and reused for multiple risk measures using the GPI principle. This is, to the best of our knowledge, the first algorithm to offer such a general and reusable solution to risk-sensitive planning in MDPs with many different objectives.

4.3.4 Proofs

Proof of Proposition 4.9

Proof. The result follows from the continuity of the function $\beta \mapsto \mathrm{EntRM}_\beta[R^\pi]$. By assumption, π^1 is optimal on $[\beta_1, \beta_b]$ and π^2 is optimal on $[\beta_b, \beta_2]$. Therefore, for any state x and timestep h, we have:

$$\mathrm{EntRM}_\beta[R_h^{\pi^1}(x)] \geq \mathrm{EntRM}_\beta[R_h^{\pi^2}(x)] \quad \forall \beta \in [\beta_1, \beta_b],$$
$$\mathrm{EntRM}_\beta[R_h^{\pi^2}(x)] \geq \mathrm{EntRM}_\beta[R_h^{\pi^1}(x)] \quad \forall \beta \in [\beta_b, \beta_2].$$

By continuity, taking the limit as $\beta \to \beta_b$ yields equality:

$$\mathrm{EntRM}_{\beta_b}[R_h^{\pi^1}(x)] = \mathrm{EntRM}_{\beta_b}[R_h^{\pi^2}(x)].$$

Since we assume that no other policy dominates these two in the neighborhood of β_b, this equality characterizes the breakpoint. □

Proof of Proposition 4.10

Proof. Let $\beta_0 \in \mathcal{B}^h$. If $\beta_0 \in \mathcal{B}^{h+1}$, the inclusion is trivial. Assume now that $\beta_0 \in \mathcal{B}^h \backslash \mathcal{B}^{h+1}$. This implies that the optimal policy for timesteps $t > h$ is constant in a neighborhood of β_0. Specifically, there exists an optimality interval $I_k^h \in \Gamma^{h+1}$ (from the problem starting at $h+1$) such that $\beta_0 \in I_k^h$.

Let π^I be a representative optimal policy associated with this interval. By definition, for all $\beta \in I_k^h$, the optimal future policy is fixed to π^I. Consequently, for any state x and action a, the return distribution $\mathcal{R}_h^{\pi^I}(x, a)$ is independent of β on this interval.

The optimization problem at timestep h restricted to $\beta \in I_k^h$ thus reduces to a single-state problem where the "rewards" are the distributions $[\mathcal{R}_h^{\pi^I}(x,a)]_a$. Since β_0 is a breakpoint at h, the optimal action at h must change at β_0. Therefore, β_0 must be a breakpoint of this induced single-state problem:

$$\beta_0 \in \mathcal{B}\left([\mathcal{R}_h^{\pi^I}(x,a)]_a, I_k^h\right).$$

The result follows by taking the union over all states and optimality intervals. □

4.4 Numerical Experiments

The numerical experiments are divided into two parts. First, Section 4.4.1 focuses on the single-state setting to study the number of breakpoints in practice and evaluate the performance of our *FindBreaks* algorithm compared to a naive grid search. Then, in Section 4.4.2, we evaluate the Optimality Front approach on small tabular risk-sensitive planning tasks, comparing its performance against existing baselines.

4.4.1 Number of breakpoints and FindBreaks performance

This section focuses on the single-state setting. We first conduct an empirical study on the number of breakpoints in simple environments, and then evaluate the performance of *FindBreaks* (Algorithm 9) compared to a naive grid search to find the breakpoints.

Empirical study on the number of breakpoints We conducted experiments in the single-state setting to determine whether the theoretical bound on the number of breakpoints is reached in practice. We generated numerous random reward distributions and used our algorithm to find the number of breakpoints. We performed two experiments: one evaluating how the number of breakpoints evolves with respect to the number of actions, and the other with respect to the number of atoms in the reward distributions.

All the distributions considered have support in $[0,1]$ with atoms evenly spaced over this interval. To generate these distributions, we treat each as an element of $[0,1]^n$, where the sum of the elements equals 1, representing a point on the n-dimensional simplex. The distributions are thus generated by uniformly sampling a point on this simplex.

For the first experiment, we fixed the number of atoms to 10 and varied the number of actions from 5 to 50. For the second experiment, we fixed the number of actions to 10 and varied the number of atoms from 5 to 50 as well. For each action, the reward distribution was generated randomly as previously described. In the solving algorithm, we searched for breakpoints for β values in the range $[-15, 15]$ with a precision of 0.01.

For each plot, we generated 100 independent problems and displayed a histogram of the number of breakpoints found across these 100 problems.

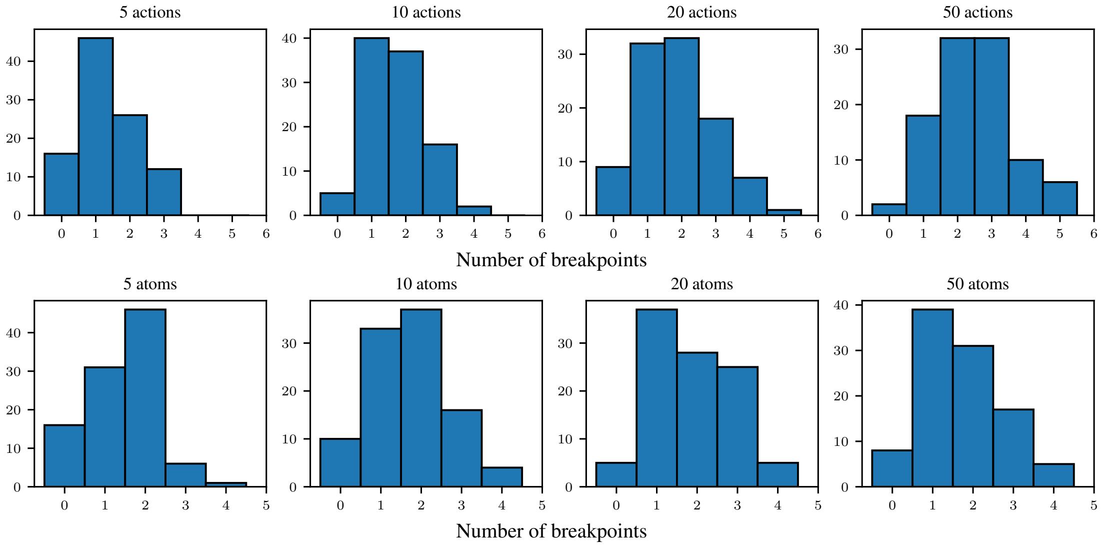

Figure 4.4: Illustration of the evolution of the number of breakpoints when the number of actions (Top) and atoms in the distributions (Bottom) is increasing. While the number of actions and atoms is multiplied by 10, the average number of breakpoints increases by less than a factor of 2.

The results are shown in Figure 4.4. We observe that the number of breakpoints increases according to the studied parameters, with an average number of breakpoints of 1.25, 1.9, 2.19, and 2.36 for the first experiment, and 1.42, 1.66, 1.93, and 2.05 for the second. However, this increase is far from the theoretical bound established in Proposition 4.5, showing sublinear growth in practice. This experiment confirms the efficiency of our algorithm, whose complexity is strongly tied to the number of breakpoints.

Empirical Evaluation and Performance Analysis of FindBreaks A simple simulation illustrates the behavior of *FindBreaks* (Algorithm 9). We consider two actions, a_1 and a_2, with reward distributions $\varrho(a_1) = \frac{1}{2}\delta_0 + \frac{1}{2}\delta_1$ and $\varrho(a_2) = \frac{99}{100}\delta_0 + \frac{1}{100}\delta_2$. Action a_1 is better in expectation (lower risk) but action a_2 can achieve a higher reward with small probability and should only outperform action a_1 for large risk parameters. Figure 4.5 illustrates this: the functions $\mathrm{EntRM}(\mathcal{R}(a))$ are plotted for $\beta > 0$ [5]. As soon as the risk parameter β is large enough (specifically, $\beta > 3.9$), action a_2 becomes optimal.

[5] This experiment is designed to test the transition to a risky action, so it is only relevant to observe the Optimality Front for $\beta > 0$.

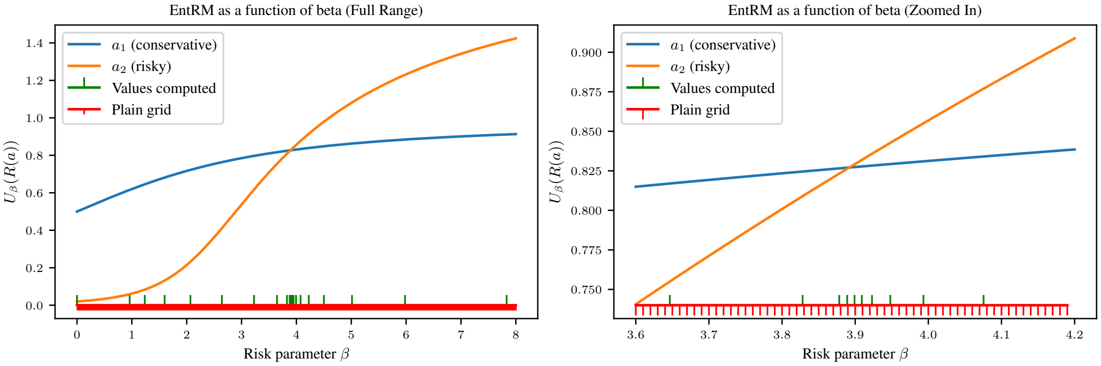

Figure 4.5: Utility functions on a single-state problem: conservative (blue) and risky (red) actions, with respective ranges of optimality intersecting at $\beta_{\text{bp}} = 3.9 \pm 10^{-2}$. In red, we show the 800 values of β tested with a naive grid; in green the 22 values tested by *FindBreaks* to identify the breakpoint. Right-hand figure zooms around β_{bp}.

FindBreaks was executed on this example for $\beta \in [0, 8]$, with a precision of $\varepsilon = 10^{-2}$, but our theoretical upper bound[6] is $\beta_{\max} = \ln(100) = 4.6$[7]. The green markers correspond to the values of β where the algorithm computed the EntRM, while the red markers represent the values that would be computed using a plain grid search with precision ε (here, a regular grid, as opposed to [Hau et al., 2023b]). As expected, we observe that the intervals shrink near the breakpoint, but grow significantly larger as we move away from these regions. Figure 4.5 (Right) zooms in on the interval $\beta \in [3.6, 4.2]$ to better visualize the concentration of intervals. Around $\beta = 3.9$, we observe that a few intervals are indeed capped by the maximal precision.

In this simple example, the naive grid uses 800 evaluations while *FindBreaks* only requires 22, with an *efficiency ratio* of $800/22 = 36$. To better quantify this gain, we run another experiment on random problems with 8 actions and reward function supported on 20 atoms in $[0, 1]$ (hence with possibly much more than 1 breakpoint). On Figure 4.6, for each level of precision $\varepsilon \in [10^{-3}, 10^{-1}]$, we compare the number of evaluations required by *FindBreaks* with the naive $1/\varepsilon$ obtained with a plain grid search (with a regular grid). We report the average gain over 20 random problems and observe efficiency ratios up to 35 for high-precision ε values.

The performance of this algorithm is closely tied to the number of breakpoints. The more breakpoints there are, the smaller the intervals will tend to be, which in turn reduces the algorithm's efficiency compared to a plain grid search. Therefore, the number of breakpoints is a critical factor. This section shows that the number of breakpoints

[6] We show a larger upper bound for visualization purposes. Our algorithm only evaluates 3 values past this limit so it does not hurt performance significantly.

[7] This bound is computed with the alternative method detailed in Section 4.2.3.

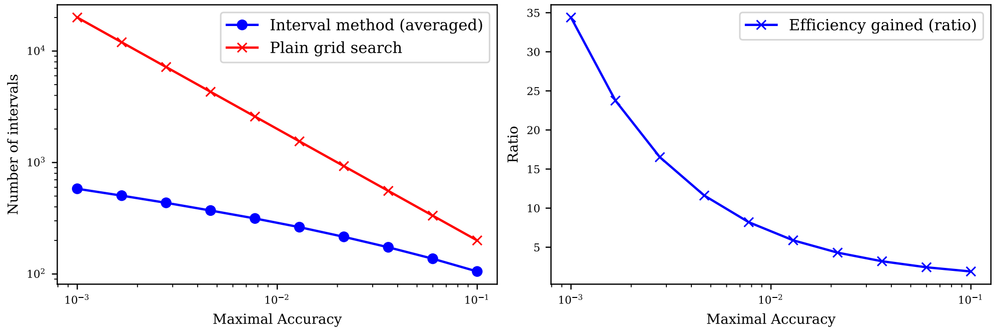

Figure 4.6: Performance gained using *FindBreaks* over a regular grid search with accuracy ε. (Left) Number of evaluations, (Right) efficiency ratio (red over blue).

grows much more slowly than the theoretical bound. This allows our algorithm to be more efficient in practice than predicted by theory.

4.4.2 The Optimality Front for Risk-Sensitive Objectives

This section evaluates the effectiveness and efficiency of the Optimality Front approach on small tabular risk-sensitive planning tasks. Our goal is primarily to demonstrate the performance of our method compared to existing baselines mentioned earlier, namely the EVaR optimization approach of Hau et al. [2023b] (*Proxy Optimization* thereafter) and the *Nested Risk Measure* [Bäuerle and Glauner, 2022] implementing an approximate (invalid) value iteration algorithm. All the experiments in this section are of relatively small scale and run on a standard laptop in a few seconds.

Environments We test our method on three different settings, the Inventory Management MDP [Bellman et al., 1955, Scarf et al., 1960], which is a standard model for an important logistics problem, the Cliff environment [Sutton et al., 1998], which we mainly use to visualize the Optimality Front, and a small synthetic environment on which we evaluate the efficiency gain of our FindBreaks algorithm compared to the grid search of Hau et al. [2023b].

Optimality Front For our method, *Optimality Front*, we executed DOLFIN only once, with accuracy $\varepsilon = 10^{-2}$ on the chosen environment. It outputs a set of return distributions for all EntRM-optimal policies. We then computed all the metrics following the GPI principle (Equation (4.6)) by applying the functional of choice on all returned optimal distributions and selecting the optimal value.

4.4.2.1 Inventory Management Results

In the Inventory Management problem, the goal is to maximize the profit of a store selling one extensive good. The store has a strict maximal capacity of $M = 10$. At each time step, the state of the store is its number of available goods, $x_t \in [M]$, and it can buy (action) a quantity $a_t \in [M]$ of new goods. The reward obtained is the profit minus the costs: $r_t = [f(D_t, x_t, a_t) - C_m(x_t) - C_c(a_t)]/4M$, where D_t is the random demand modeled by a binomial $D_t \sim B(0.5, M)$, $f(D_t, x_t, a_t) = 4\min(D_t, x_t + a_t)$ is the sales profit, $C_m(x_t) = 1x_t$ is the maintenance cost, and $C_c(a_t) = 3 + 2a_t$ is the order cost. We considered a horizon $H = 10$ with $x_1 = 0$. Optimal policies in the Inventory Management MDP can be parametrized by two thresholds (s_t, S_t) [Scarf et al., 1960]: at time t, if the stock x_t is less than s_t, then the agent should buy goods so that they have a stock of exactly S_t, i.e. $a_t^* = S_t - x_t$.

For the Inventory Management problem described above, with our choice of rewards and costs, the Optimality Front contained 18 different optimal policies.

Threshold Probability In Table 4.1, we optimize policies to minimize the probability that the return falls below a threshold $T < \mu^*$ where μ^* is the optimal mean return. For this problem, we were able to compute the non-Markov optimal policies via Dynamic Programming on the augmented state space which is small enough. We report the estimated probability of falling below the imposed threshold.

We observe that using the Optimality Front method outperforms the risk-neutral optimal policy by up to a factor of 10. On the other hand, *Nested Risk Measure* fails to find a good policy and performs worse than the risk-neutral one. While *Proxy Optimization* achieves reasonable results, this experiment shows that Generalized Policy Improvement on the Optimality Front performs better.

Table 4.1: Evaluation of $P(\mathcal{R}^\pi \leq T)$ for Inventory Management.

T/μ^*	0.25	0.33	0.5	0.66	0.75
Optimality Front	$\mathbf{1.26e^{-5}}$	$\mathbf{8.40e^{-5}}$	$\mathbf{3.26e^{-3}}$	$\mathbf{3.91e^{-2}}$	$\mathbf{8.78e^{-2}}$
Proxy Optimization	$2.33e^{-5}$	$1.18e^{-4}$	$3.28e^{-3}$	$\mathbf{3.91e^{-2}}$	$\mathbf{8.78e^{-2}}$
Risk neutral optimal	$1.11e^{-4}$	$4.24e^{-4}$	$5.73e^{-3}$	$4.62e^{-2}$	$9.77e^{-2}$
Nested Prob. Thresh.	$1.54e^{-3}$	$8.37e^{-3}$	1	1	1
Optimal value	$6.71e^{-8}$	$6.29e^{-7}$	$4.48e^{-5}$	$7.85e^{-4}$	$3.59e^{-3}$

The true optimal value here is significantly better than what any Markov policy can achieve, especially for low thresholds. This is due to the density and high randomness of the reward that strongly affect the distance to the threshold at each step. In goal-oriented

MDPs, with scarce reward, this gap mostly vanishes (ex. see Cliff in Section 4.4.2.2).

Value at Risk family In Table 4.2, we maximize lower quantiles of the return, aiming for policies with thin tails again, but through a different optimization objective. We do not have access to the optimal non-Markovian policy for this problem, as computing it led to numerical issues. However, we refer to Hau et al. [2023b] who demonstrated that their proxy optimization approach outperforms traditional methods. We observe that for both VaR and CVaR, DOLFIN achieves slightly better or comparable performance to their state-of-the-art *Proxy Optimization*.

Table 4.2: Evaluation of (C)VaR$_\alpha[\mathcal{R}^\pi]$ for Inventory Management.

Risk Measure	VaR				CVaR			
Risk parameter α	0.05	0.1	0.2	0.5	0.05	0.1	0.2	0.5
Optimality Front	**1.25**	**1.33**	**1.45**	**1.65**	**1.14**	**1.21**	**1.30**	**1.45**
Proxy Optimization	1.22	1.30	**1.45**	**1.65**	1.13	1.20	1.30	**1.45**
Risk neutral optimal	1.22	**1.33**	1.43	**1.65**	1.11	1.19	1.28	1.44
Nested Risk Measure	0.88	0.95	1.05	1.28	0.75	0.84	0.95	1.12

The values in Table 4.1 and Table 4.2 reveal that the improvements of our *Optimality Front* method over the *Proxy Optimization* become less significant as the level of risks decreases.

4.4.2.2 Cliff Environment results

Here we consider the Cliff grid world [Sutton et al., 1998] illustrated in Figure 4.7. The agent starts in the blue state. At each step, they have a small probability (0.1 here) of moving to another random direction. Due to these random transitions, it is risky to walk too close to the cliff (bottom, in red, negative reward $-\frac{1}{2}$), and conservative policies will prefer to walk further away to reach the goal (in green). The horizon is fixed at $H = 15$, so in principle, the agent has enough time to reach the end using the safe path. The reward when the goal is reached at step h is $1 - \frac{h}{2H}$, which encourages the agent to reach it as fast as possible. Note that the reward function is slightly different from the original Cliff environment in Section 2.1.1. Here, there is no per-step negative reward for each step taken, the reward only happens when the goal is reached or when falling into the cliff. This simplifies the computation of expected utilities objectives, as the stock is always 0 until the goal is reached, making optimal policies Markovian. The only downside is the reward no longer being stationary.

Figure 4.7 illustrates the Optimality Front, where the optimal policies for different values of β are shown. For high values of β (e.g. $\beta = 10$), the agent takes the risky path, while for low values of β (e.g. $\beta = -5$), the agent takes the safe path. One can even observe that for extremely negative values of β (e.g. $\beta = -10$) the agent prefers to stay away from the cliff, not even trying to reach the goal (see purple arrow in top right corner pointing up).

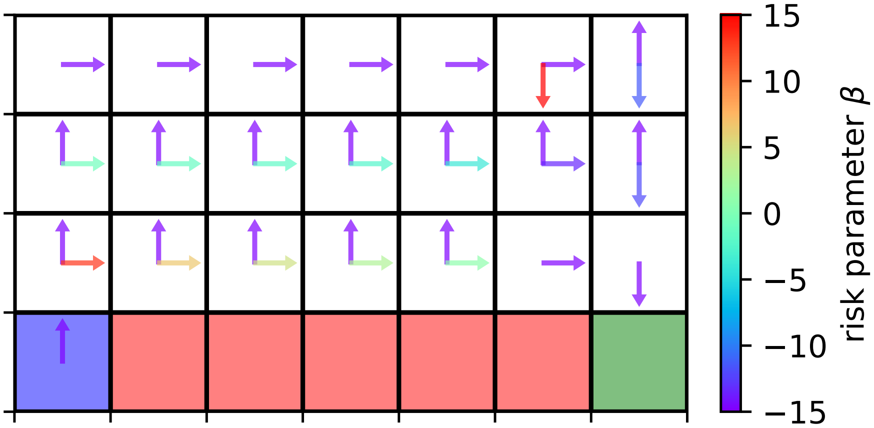

Figure 4.7: Optimal policies for different values of β in Cliff. The blue state is the starting point, the red states are the cliff (negative reward), and the green state is the goal (positive reward). The color of arrows indicates the lowest value of the risk parameter β for which the action is optimal. For high β values, the agent takes the risky path (green-blue), while for low β values, it takes the safe path (violet). For very low β values, it prefers to stay away from the cliff, not even trying to reach the goal (see purple arrow in top right corner pointing up).

Table 4.3: Evaluation of $P(\mathcal{R}^\pi \leq T)$ for Cliff.

T	-0.5	0
Optimality Front	$\mathbf{3.72e^{-2}}$	$\mathbf{4.65e^{-2}}$
Proxy Optimization	$3.85e^{-2}$	$4.67e^{-2}$
Risk neutral optimal	$4.50e^{-2}$	$4.84e^{-2}$
Nested Prob. Thresh.	1	1
Optimal value	$\mathbf{3.72e^{-2}}$	$\mathbf{4.65e^{-2}}$

For the Threshold Probability problem, we only consider 2 values of the threshold, -0.5 corresponding to falling into the cliff, and 0 corresponding to not reaching the goal. For the first threshold, the objective is to find the policy that is least likely to fall, while for the second it is to find the policy with the most chances of reaching the goal.

Table 4.3 confirms that our *Optimality Front* method performs better than other methods for the Threshold Probability. An important remark is that, here, the real

Table 4.4: Evaluation of (C)VaR$_\alpha[\mathcal{R}^\pi]$ for Cliff.

Risk Measure	VaR				CVaR			
Risk parameter α	0.05	0.1	0.2	0.5	0.05	0.1	0.2	0.5
Optimality Front	**0.53**	**0.63**	**0.70**	**0.76**	**−0.37**	**0.11**	**0.38**	**0.58**
Proxy Optimization	0.00	0.6	**0.70**	**0.76**	−0.39	−0.08	0.38	0.58
Risk neutral optimal	**0.53**	**0.63**	**0.70**	**0.76**	−0.43	0.08	0.37	0.58
Nested Risk Measure	−0.5	−0.5	−0.5	−0.5	−0.5	−0.5	−0.5	−0.5

optimal value is reached by the Optimality Front. Similar performances are observed for the CVaR in Table 4.4. For the VaR, the gain in performance is limited, which is explained by the scarcity of rewards in the environment (the small changes in the return distribution do not change the value of the VaR).

4.4.2.3 Optimality Front and Efficiency

We also plot in Figure 4.8 and Figure 4.9 the efficiency ratio of using FindBreaks compared to a regular grid, similar to Figure 4.6. This figure highlights the polynomial gain in performance of using DOLFIN and FindBreaks when one wants to compute all the breakpoints up to a specific accuracy.

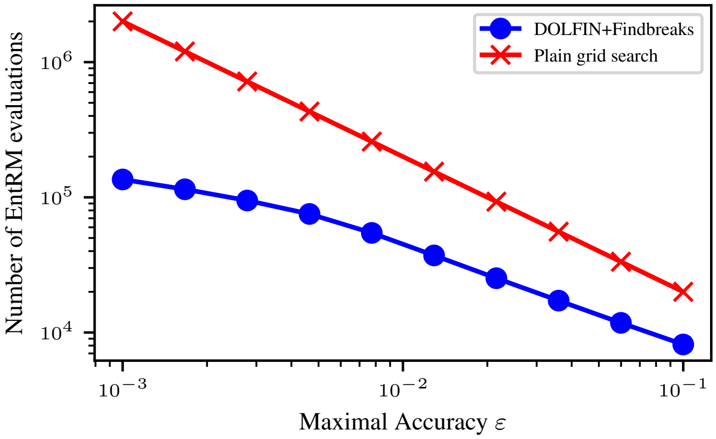

Figure 4.8: Performance gained using Algorithm 10 over a regular grid, on Inventory Management.

Compared to Inventory Management, the performance gained in the Cliff environment for computing the Optimality Front is much better, with a ratio up to a factor 100, as seen in Figure 4.9.

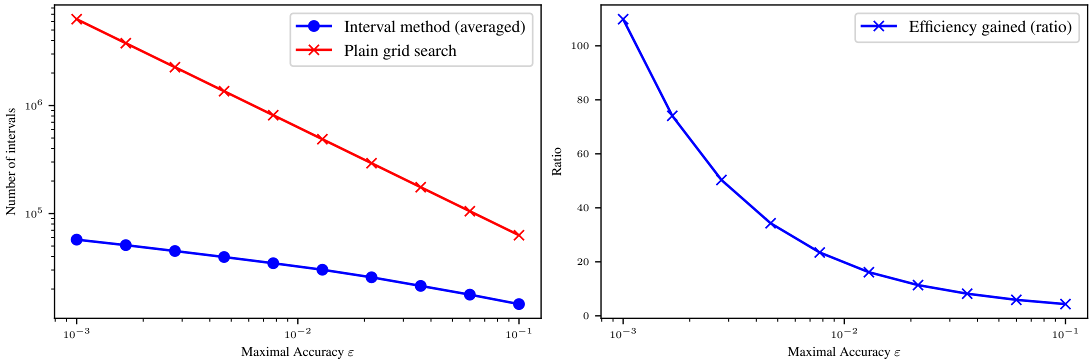

Figure 4.9: Performance gained using Algorithm 10 on Cliff. (Left) Number of evaluations, (Right) efficiency ratio (red over blue).

We also consider a synthetic MDP, illustrated in Figure 4.10, with $|X|= 4$ states and $|\mathcal{A}|= 2$ actions. We set the horizon to $H = 10$. The transitions are stochastic, and the rewards are dense (most transitions lead to a different reward), leading to complex return distributions. We used this MDP to compare the grid search suggested by Hau et al. [2023b] with the same grid augmented with the *FindBreaks* algorithm to prevent recomputing the values of the same optimal policy. This environment is particularly interesting for this experiment. First, it is small enough to allow us to run the experiment with limited compute in a reasonable time. Second, it is complex enough to have a high diversity of risk-sensitive behaviors: we found 16 different policies in the optimality front.

We vary the accuracy parameter ε[8] and report the number of evaluations required to reach this accuracy. We observe that our approach greatly improves the efficiency of the method, especially for small ε.

4.5 Limitations and Discussions

Beyond the Planning Setting. This study considers only the *planning* setting, when the environment dynamics are fully known, and only small-scale problems are considered. The extension to more general settings is natural yet challenging because of the new challenges that arise. Optimizing the EntRM in the reinforcement learning setting has been studied before [Borkar, 2002], but many questions remain and it is still a current topic of research [Su et al., 2025]. The sample complexity is not well understood yet, especially considering several values of the risk parameter β [Liang and Luo, 2024,

[8] This is not the same ε as in the *FindBreaks* algorithm for the accuracy on breakpoints, it is the ε of Hau et al. [2023b] to compute an ε-approximation of EVaR.

ε	Grid ([Hau et al., 2023b])	Grid + FindBreaks	Ratio
1.000	3	3	1.0
0.100	117	117	1.0
0.060	321	320	1.003
0.030	1281	675	1.90
0.010	11514	1411	8.16
0.006	31982	1817	17.60
0.003	127923	2371	53.95

Table 4.5: Number of evaluations of the EntRM required with accuracy ε using the Hau et al. [2023b] method (Grid) vs. their approach and removing the redundant computation using *FindBreaks* (Grid + FindBreaks). *FindBreaks* can save up to an order of magnitude of evaluations for small ε.

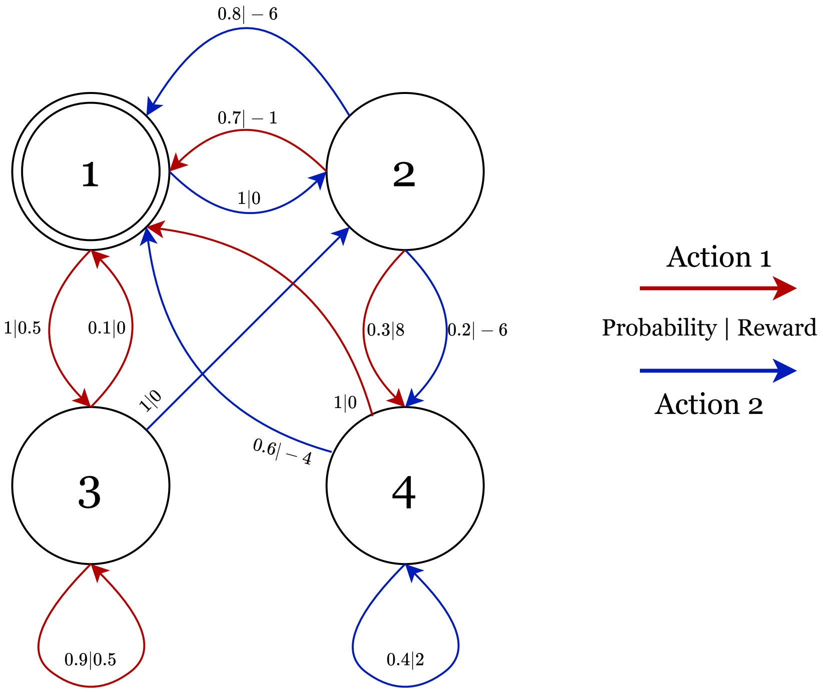

Figure 4.10: Synthetic MDP with $|X|= 4, |\mathcal{A}|= 2$, with stochasticity in the transitions and dense rewards. The arrow color indicates action associated with that transition. The agent starts on state 1.

Mortensen and Talebi, 2025, Su et al., 2025]. Larger-scale environments would require using function approximation. In this case, computing the exact Optimality Front may no longer be tractable, but the principles of Generalized Policy Improvement and of the *FindBreaks* algorithm still hold.

Approximation tightness. The approximations used in Section 4.1 are entirely dependent on the return distribution and hard to control precisely in general. We are not aware of distribution-dependent measures of the tightness of EVaR, and it is unclear to us whether it is possible to prove any relevant error bounds for our *Optimality Front* in a general class of MDPs.

Complexity of the algorithm. The theoretical complexity of DOLFIN is a function of the number of breakpoints. Those breakpoints arise from the MDP dynamics and their number cannot be expressed simply as a function of the main parameters of the MDP ($\mathcal{X}$, $\mathcal{A}$ or H). The only bound we can obtain is highly pessimistic and does not enable us to provide meaningful and problem-dependent complexity results. Empirical results show a clear gain on a synthetic problem, but it is not yet fully understood how these gains scale with the size of the MDP.

4.6 Conclusion

We propose a unified framework for optimizing risk-sensitive objectives in Markov Decision Processes. Leveraging the computational advantages of the Entropic Risk Measure (EntRM), we provide an efficient algorithm for computing the Optimality Front, a family of policies that are optimal on a range of risk tolerance values. This also enables the approximation of key metrics such as Threshold Probability, Values at Risk and Conditional Values at Risk. Our algorithm demonstrates significant practical benefits in both efficiency and policy quality. This approach not only enhances risk-sensitive planning but also provides a versatile tool for tackling a variety of decision-making problems under uncertainty. It remains to extend this framework beyond planning, to learning with unknown transition probabilities and reward distributions.

5

Learning the Entropic Risk Measure

Contents

The Entropic Risk Measure plays a central role in risk-sensitive decision-making, as illustrated by the previous chapters. These chapters investigated the planning setting, where the dynamics of the environment are known. In practice, such algorithms are used in reinforcement learning settings, where the model is unknown and must be learned from data. In such contexts, estimating the Entropic Risk Measure accurately from samples becomes crucial [Sutton et al., 1998, Lattimore and Szepesvári, 2020].

The EntRM in the RL setting has been investigated in the literature [Borkar, 2002], with some very recent results highlighting the ongoing efforts to better understand these problems [Mortensen and Talebi, 2025, Su et al., 2025, Liang and Luo, 2024]. These works address various settings and problems, but focus on estimating the EntRM for a single, fixed risk parameter β. This limitation overlooks a key strength of the EntRM framework highlighted in Chapter 4: the ability to capture a full spectrum of risk-sensitive behaviors by varying β. To fully exploit the potential of the EntRM for general risk-sensitive learning, we must therefore address the problem of estimating it not just pointwise, but across the entire range of risk parameters. The goal of this chapter is to bridge this gap by investigating the statistical estimation of the EntRM, with a specific focus on uniform convergence over multiple values of β simultaneously.

We first establish statistical guarantees for the plug-in EntRM estimator under a fixed risk parameter β, deriving non-asymptotic concentration bounds for bounded random variables. We then address the problem of uniform estimation for discrete random variables, providing the first concentration results that hold simultaneously across a continuous range of β. Finally, we explain how these results can be applied to the estimation of the EVaR.

5.1 EntRM Estimation with Fixed Risk Parameter

In this section, we study the problem of estimating the Entropic Risk Measure from i.i.d. samples. We start by investigating the estimation of a single Entropic Risk Measure for a fixed risk parameter β. Then, we extend our analysis to the more challenging problem of estimating the Entropic Risk Measure uniformly over a range of risk parameters β.

Problem setup We observe an i.i.d. sequence $\{X_i\}_{i=1}^n$ drawn from a distribution P_X on $\mathbb{R}$. Given a risk parameter $\beta \in \mathbb{R}$, we write the entropic risk measure as

$$\mu_\beta \;=\; \mathrm{EntRM}_\beta[X] \;=\; \frac{1}{\beta} \log \mathbb{E}[e^{\beta X}], \tag{5.1}$$

where the expectation is taken under $X \sim P_X$. Throughout this section, we assume that the moment generating function $\mathbb{E}[e^{\beta X}]$ exists and is finite for any β.

Given a range of values for β and an i.i.d. sample $X_1^n = (X_1, \dots, X_n)$, we are interested in constructing estimators that approximate μ_β from data. In particular, we want an estimator that works for every fixed β. That is,

$$\forall \beta \in \mathbb{R}, \qquad |\hat{\mu}_\beta^n \; - \; \mu_\beta| \; \leq \; \varepsilon_n(\beta), \tag{5.2}$$

where $\varepsilon_n(\beta) \to 0$ as $n \to \infty$.

The plug-in estimator A natural way to estimate μ_β is to replace the expectation in (5.1) by its empirical counterpart.

Definition 5.1 (plug-in estimator)**.** Let $\beta \in \mathbb{R}$ and let $(X_1, \dots, X_n)$ be i.i.d. samples drawn from a distribution P_X on $\mathbb{R}$. The *plug-in estimator* of μ_β is defined by

$$\hat{\mu}_\beta^n \;=\; \frac{1}{\beta} \log\!\left(\frac{1}{n} \sum_{i=1}^{n} e^{\beta X_i} \right).$$

For small $|\beta|$, we may use the Taylor expansion

$$e^{\beta X_i} = 1 + \beta X_i + O(\beta^2),$$

which yields

$$\hat{\mu}_\beta^n \;=\; \frac{1}{n} \sum_{i=1}^{n} X_i + O(\beta).$$

Hence, the estimator reduces to the sample mean when β is close to 0, as expected from the fact that μ_β converges to the mean of X as $\beta \to 0$ (see Section 2.3.2.1).

Remark 5.1 (Bias of the plug-in estimator)**.** The plug-in estimator is biased. Let $Z_i = e^{\beta X_i}$ and $Z = e^{\beta X}$. Then

$$\hat{\mu}_\beta^n \;=\; \frac{1}{\beta} \log \hat{m}_n, \qquad \hat{m}_n = \frac{1}{n} \sum_{i=1}^{n} Z_i,$$

while

$$\mu_\beta \;=\; \frac{1}{\beta} \log m, \qquad m = \mathbb{E}[Z].$$

When $\beta > 0$, by concavity of the logarithm and Jensen's inequality,

$$\mathbb{E}[\hat{\mu}_\beta^n] = \frac{1}{\beta}\, \mathbb{E}[\log \hat{m}_n] \;\leq\; \frac{1}{\beta} \log \mathbb{E}[\hat{m}_n] = \frac{1}{\beta} \log m = \mu_\beta,$$

so $\hat{\mu}_\beta^n$ is negatively biased. When $\beta < 0$, the inequality is reversed, and $\hat{\mu}_\beta^n$ is positively biased.

Consistency via the law of large numbers We first show that the plug-in estimator is consistent for every fixed β.

Proposition 5.1 (Consistency of the plug-in estimator)**.** Assume that $\mathbb{E}[e^{\beta X}] < \infty$ for a fixed $\beta \in \mathbb{R}$. Then

$$\hat{\mu}_\beta^n \xrightarrow[n\to\infty]{\text{a.s.}} \mu_\beta .$$

In particular, $\hat{\mu}_\beta^n$ is consistent for every fixed β.

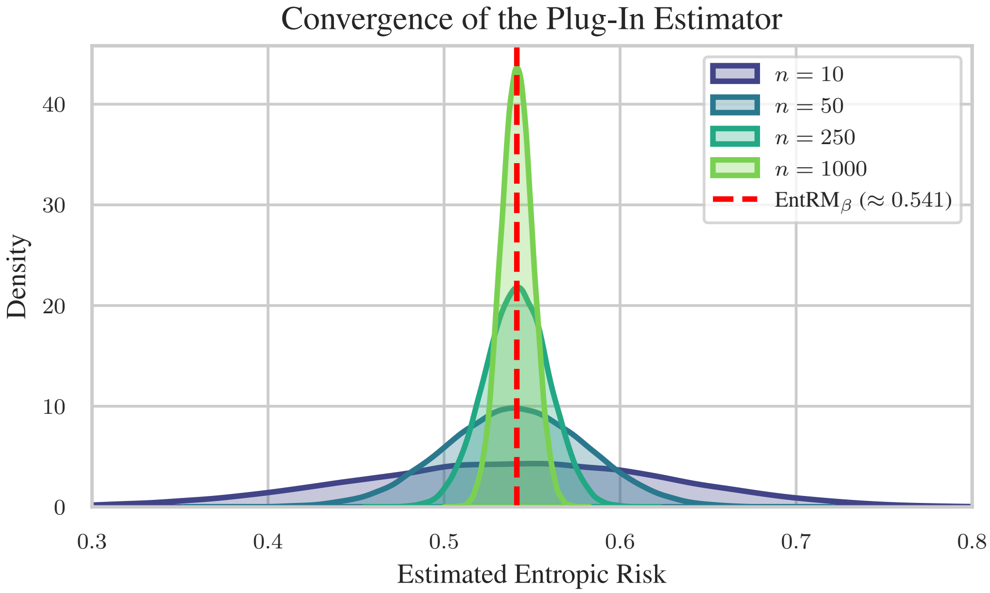

Figure 5.1: Illustration of the convergence in distribution of the plug-in estimator $\hat{\mu}_\beta^n$ to the true Entropic Risk Measure μ_β as the number of samples n increases. Here, $X \sim U[0,1]$ and $\beta = 1$. The plot shows the distribution of the estimates for different values of n.

Beyond almost sure convergence, we are interested in non-asymptotic deviation bounds that quantify the speed of convergence and allow us to build confidence intervals.

Exponential tilting and its limitations A standard way to provide concentration inequalities in the mean estimation setting is to apply Chernoff's method to the sample mean. Here, we consider the transformed variables $Z_i = e^{\beta X_i}$, for all $i \in [n]$, so that

$$\hat{\mu}_\beta^n = \frac{1}{\beta} \log \hat{m}_n, \qquad \hat{m}_n = \frac{1}{n} \sum_{i=1}^{n} Z_i.$$

and the problem is reduced to estimating the sample mean of the variables Z_i. Chernoff's approach relies on the existence of the moment generating function of Z, namely

$$M_Z(t) = \mathbb{E}[e^{tZ}] = \mathbb{E}[e^{te^{\beta X}}]$$

for $t > 0$. This requirement is significantly stronger than the existence of $\mathbb{E}[e^{\beta X}]$ alone: we now need finiteness of $\mathbb{E}[e^{te^{\beta X}}]$ for some nonzero t.

Remark 5.2 (Failure for Gaussian variables). When X is Gaussian, $Z = e^{\beta X}$ is log-normal. For any $t > 0$,

$$\mathbb{E}[e^{te^{\beta X}}] = \int_{\mathbb{R}} e^{te^{\beta x}} \phi(x; \mu_x, \sigma_x^2)\, dx$$

diverges [Heyde, 1963], where $\phi(\cdot; \mu_x, \sigma_x^2)$ is the Gaussian density. Hence, the moment generating function $M_Z(t)$ is infinite for all $t > 0$, and exponential tilting cannot be applied. In this case, Chernoff-type bounds based on $Z_i = e^{\beta X_i}$ are vacuous, even though μ_β itself is perfectly well defined:

$$\mu_\beta = \frac{1}{\beta} \log \mathbb{E}[e^{\beta X}] = \frac{1}{\beta} \log \left(\exp\Big(\mu_x \beta + \frac{1}{2} \sigma_x^2 \beta^2 \Big) \right) = \mu_x + \frac{1}{2} \sigma_x^2 \beta,$$

The discussion above highlights that Chernoff bounds for the plug-in estimator are only available under additional integrability assumptions on $e^{\beta X}$.

A general Chernoff bound Assume that the cumulant generating function (CGF) of $Z = e^{\beta X}$,

$$\psi_Z(t) \ = \ \log \mathbb{E}[e^{tZ}],$$

is finite in a neighborhood of 0. Let ψ_Z^* denote its Legendre transform:

$$\psi_Z^*(y) \ = \ \sup_{t \in \mathbb{R}} \{ ty - \psi_Z(t) \}.$$

Theorem 5.1 (General Chernoff bound). Let $\beta \in \mathbb{R}$ and assume that the cumulant generating function of $Z = e^{\beta X}$ is finite in a neighborhood of 0. Then, for any $\varepsilon > 0$,

$$\Pr\left(\hat{\mu}_\beta^n \ - \ \mu_\beta \ > \ \varepsilon \right) \ \le \ \exp\Big(-n\, \psi_Z^*(e^{\beta\varepsilon}\, \mathbb{E}[e^{\beta X}]) \Big). \tag{5.3}$$

This result gives a general deviation bound for the plug-in estimator, but is difficult to exploit directly. The function ψ_Z^* depends on the full distribution of $Z = e^{\beta X}$, and explicit expressions are only available in special cases. We next focus on the important case of bounded variables, where more explicit bounds can be derived.

Bernoulli variables Bernoulli variables are simple random variables for which the bound can be computed more explicitly. We obtain the following result:

Lemma 5.2 (Chernoff bound for Bernoulli variables). Let $X \sim \text{Bernoulli}(p)$ and $\beta \in \mathbb{R}$. Then, for any $0 < \varepsilon < 1 - \mu_\beta$,

$$\Pr\left(\frac{1}{\beta} \log \frac{1}{n} \sum_{i=1}^{n} e^{\beta X_i} \ > \ \mu_\beta + \varepsilon \right) \ \le \ \exp(-n\, \mathrm{kl}(q \,\|\, p)), \tag{5.4}$$

where $\mathrm{kl}(q\|p) = q\log\frac{q}{p} + (1-q)\log\frac{1-q}{1-p}$ denotes the binary Kullback–Leibler divergence and

$$q = \frac{\mathbb{E}[e^{\beta(X+\varepsilon)}] - 1}{e^{\beta} - 1}\,.$$

This bound is closely related to the classical Chernoff bound for estimating the mean of Bernoulli variables[1], with the usual term $p+\varepsilon$ replaced by the transformed term q that depends on β. The interpretation of q will be made clearer with the following result.

Bounded variables The case where X is bounded is particularly important as it covers a large range of practical applications. Conveniently, we can generalize the result obtained for Bernoulli variables to any bounded variable, leading to the following theorem.

Theorem 5.3 (Chernoff bound for bounded variables). Consider X with support contained in $[0,1]$.

Then, for any $\beta \in \mathbb{R}$ and any $0 < \varepsilon < 1 - \mu_\beta$,

$$\begin{aligned}\Pr\left(\frac{1}{\beta}\log\frac{1}{n}\sum_{i=1}^{n} e^{\beta X_i} \;>\; \mu_\beta + \varepsilon\right) \;&\leq\; \exp\left(-n\,\mathrm{kl}\left(\frac{e^{\beta(\mu_\beta+\varepsilon)}-1}{e^{\beta}-1}\,\middle\|\,\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}\right)\right),\\ &=\; \exp(-n\,\mathrm{kl}(g_\beta(\mu_\beta+\varepsilon)\,\|\,g_\beta(\mu_\beta)))\,,\end{aligned}$$

where $g_\beta(x) = \frac{e^{\beta x}-1}{e^{\beta}-1}$.

The proof can be found in Section 5.3 and works by bounding the Chernoff bound by that of a well-chosen Bernoulli variable. This concentration bound can also be obtained more directly by applying the Chernoff–Hoeffding bound to the variable $\frac{e^{\beta X}-1}{e^{\beta}-1}$ which is supported on $[0,1]$ whenever X is supported on $[0,1]$. However, the full proof gives insights that will turn out to be useful.

Remark 5.3. For a fixed β, g_β corresponds to the transformation that maps the entropic risk of a Bernoulli random variable to its mean: if $X \sim \mathrm{Ber}(p)$ and $\mathrm{EntRM}_\beta[X] = \mu_\beta$, then $\forall\beta, g_\beta(\mu_\beta) = p$.

Corollary 5.4 (Simplified concentration bound). Consider X with support contained in $[0,1]$.

Then, for any $\beta \in \mathbb{R}$ and any $0 < \varepsilon < 1 - \mu_\beta$,

$$\Pr\left(\frac{1}{\beta}\log\frac{\sum_{i=1}^{n} e^{\beta X_i}}{n} \;>\; \mu_\beta + \varepsilon\right) \;\leq\; \exp\Big(-2n\left(g_\beta(\mu_\beta+\varepsilon) - g_\beta(\mu_\beta)\right)^2\Big)\,,$$

[1] As a reminder, the Chernoff–Hoeffding bound provides the bound $\Pr(\hat{\mu} > \mu + \varepsilon) \leq \exp(-n\,\mathrm{kl}(\mu+\varepsilon\|\mu))$ [Boucheron et al., 2003].

This bound is obtained by applying Pinsker's inequality on the KL divergence in Theorem 5.3. We note the analogy with the Hoeffding bound for the sample mean as we recover it when $\beta \to 0$ since $\lim\limits_{\beta\to 0} g_\beta(x) = x$ (and thus $\lim\limits_{\beta\to 0} g_\beta(\mu_\beta+\varepsilon) - g_\beta(\mu_\beta) = \varepsilon$).

Remark 5.4. The inequalities and proofs provide valuable insights for generalizing these results to any expected utilities. Indeed, consider a monotonic utility function $f : [0,1] \to \mathbb{R}$. We are interested in estimating the uncertainty equivalent $\mu_f = f^{-1}\mathbb{E}[f(X)]$. The plug-in estimator is $\hat{\mu}_f^n = f^{-1}\left(\frac{1}{n}\sum_{i=1}^n f(X_i)\right)$. Following the same ideas as in the proof of Theorem 5.3, it is possible to obtain the following concentration bound:

Assume X has support contained in $[0,1]$ and $f : [0,1] \to \mathbb{R}$ is a monotonic function. Then, for any $0 < \varepsilon < 1 - \mu_f$,

$$\Pr\left(\hat{\mu}_f^n \ > \ \mu_f + \varepsilon\right) \ \le \ \exp(-n\,\mathrm{kl}\left(g_f(\mu_f+\varepsilon)\,\|\,g_f(\mu_f)\right)), \tag{5.5}$$

where $g_f(x) = \frac{f(x)-f(0)}{f(1)-f(0)}$.

5.2 Uniform EntRM and EVaR estimation

For a fixed random variable X, the whole function $\beta \mapsto \mu_\beta$ is fixed and characterized by the distribution of X. Estimating several values of μ_β for varying β should not be done independently, as the estimations are highly correlated. In this section, we investigate uniform concentration bounds on the estimation of μ_β for a range of values of β. We will also see how this can be used to estimate the Entropic Value at Risk (EVaR), defined as $\mathrm{EVaR}[X] = \sup_{\beta<0} \mu_\beta - \frac{1}{\beta}\log(\alpha)$

Concentration bounds for extreme values of β The first part of the problem relies on understanding how the concentration bounds derived in Theorem 5.3 and corollary 5.4 behave when β varies. The first thing we notice is that the concentration bounds can become vacuous when β goes to extreme values.

Example 5.1 (Uniform random variable)**.** We show that there can be no positive lower bound for the function $\beta \mapsto g_\beta(\mu_\beta+\varepsilon) - g_\beta(\mu_\beta)$. Suppose that $X \sim \mathcal{U}([0,1])$. Then, for every β

$$\mathbb{E}[\exp(\beta X)] = \frac{e^\beta - 1}{\beta}$$

We can compute that

$$g_\beta(\mu_\beta+\varepsilon) - g_\beta(\mu_\beta) = \underbrace{\frac{e^{\beta\varepsilon}-1}{e^\beta-1}}_{\to 1}\underbrace{\frac{e^\beta-1}{\beta}}_{\to 0} \to 0 \quad \text{as } \beta \to \pm\infty.$$

Thus, the bound in Corollary 5.4 becomes vacuous for large values of β. There is no positive lower bound for the function $\beta \mapsto g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta)$ if $X \sim \mathcal{U}([0,1])$.

Similarly, we can also show that $\lim\limits_{\beta \to \pm\infty} \mathrm{kl}\left(g_\beta(\mu_\beta + \varepsilon) \mid\mid g_\beta(\mu_\beta)\right) = 0$ for $X \sim \mathcal{U}([0,1])$, making the bound in Theorem 5.3 also vacuous for large values of β. We include the proof in Section 5.4.

This is not the expected behavior as we know that for $\beta \to -\infty$, μ_β converges to the essential infimum of X. In the case of $\hat{\mu}_\beta$, this corresponds to the essential infimum of the empirical distribution, which corresponds to the minimum of the sample: $\lim\limits_{\beta \to -\infty} \hat{\mu}_\beta = \min_i X_i$. Hence, we would expect the concentration bound to converge to that of the minimum of the sample,

$$\Pr\left(\min_i X_i > \operatorname{ess\,inf} X + \varepsilon\right) = \Pr\left(\forall i, X_i > \varepsilon\right) = \Pr(X \geq \varepsilon)^n ,$$

when $\operatorname{ess\,inf} X = 0$ as for the uniform distribution.

We actually show that this bound is indeed recovered from the Chernoff bound:

Proposition 5.2. Let X be a random variable with support in $[0,1]$ such that $\operatorname{ess\,inf} X = 0$. Then, the Chernoff method gives the following limit:

$$\begin{aligned}\lim_{\beta \to -\infty} \Pr(\hat{\mu}_\beta \geq \mu_\beta + \varepsilon) &\leq \lim_{\beta \to -\infty} \inf_{\lambda > 0} \exp(\lambda \mathbb{E}[e^{\beta X}] e^{\beta\varepsilon} + n \log\left(\mathbb{E}\left[e^{-\lambda e^{\beta X}}\right]\right)) \\ &\leq \Pr(X \geq \varepsilon)^n .\end{aligned}$$

This issue comes from the approximation made when deriving the concentration bound. The loss appears when we get the general bound for variables on $[0,1]$ by bounding the kl divergence by that of a well-chosen Bernoulli variable.

Indeed, to prove such a bound, we use the fact that for any variable X with support in $[0,1]$, and $\mu_\beta = \mathrm{EntRM}_\beta[X]$, we can consider the Bernoulli variable $Y_\beta \sim \mathcal{B}(g_\beta(\mu_\beta))$ which satisfies $\mathbb{E}[\exp(\beta X)] = \mathbb{E}[\exp(\beta Y_\beta)]$ and $\mathbb{E}[\exp(\lambda \exp(\beta X))] \leq \mathbb{E}[\exp(\lambda \exp(\beta Y_\beta))]$ for any $\lambda > 0$. We thus bound the concentration of $\hat{\mu}_\beta$ by that of the estimator of the Bernoulli variable Y_β. However, when $\beta \to -\infty$, the variable Y_β converges to a Bernoulli variable with parameter $1 - \Pr(X = 0)$ ($= 1$ if X is continuous). For such variable, $\Pr(X \geq \varepsilon) = 1$, which explains why the bound becomes trivial.

The case of discrete variables Discrete variables (precisely, variables for which both $P(X = 0)$ and $P(X = 1)$ are strictly positive) behave better as we can show that the term $g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta)$ is bounded below by a positive constant independent of β, which can give us a concentration bound independent of β.

Lemma 5.5. Let $\varepsilon > 0$. Consider X such that $P(X = 0) > 0$ and $P(X = 1) > 0$. The function $\beta \mapsto g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta)$ is bounded below and

$$i(X, \varepsilon) = \inf_\beta g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta) > 0$$

.

This result directly results in a bound independent of β using Corollary 5.4.

Corollary 5.6.

$$\forall \beta, \qquad \Pr\left[\hat{\mu}_\beta \; > \; \mu_\beta + \varepsilon\right] \; \leq \; \exp\Big(-2n \cdot i(X, \varepsilon)^2\Big)$$

This quantity $i(X, \varepsilon)$ is particularly interesting as it helps us quantify the uniform convergence of $\hat{\mu}_\beta$ to μ_β. However, its value depends on the distribution of X in a non-trivial way, as illustrated by the following example.

Example 5.2 (Bernoulli random variable)**.** We show that the lower bound of the function $\beta \mapsto g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta)$ depends on the distribution of the random variable. Suppose that $X \sim \mathcal{B}(p)$, where p will be chosen later. Then,

$$\begin{aligned} g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta) &= \frac{e^{\beta\varepsilon} - 1}{e^\beta - 1}(1 - p + pe^\beta) && (5.6) \\ &= \frac{e^{\beta\varepsilon} - 1}{e^\beta - 1} + p(e^{\beta\varepsilon} - 1) && (5.7) \end{aligned}$$

Note that $\frac{e^{\beta\varepsilon}-1}{e^\beta-1} \to 0$ when $\beta \to +\infty$. Let $\delta > 0$ and choose β such that $\frac{e^{\beta\varepsilon}-1}{e^\beta-1} < \frac{\delta}{2}$. Then, choose $p \in (0, 1)$, such that $(e^{\beta\varepsilon} - 1)p < \frac{\delta}{2}$. Then,

$$g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta) < \frac{\delta}{2} + \frac{\delta}{2} = \delta.$$

Since the choice of δ was arbitrary, we deduce that the lower bound of the function $\beta \mapsto g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta)$ depends on the law of the random variable, in this case, it depends on the value of p.

Remark 5.5. In the Bernoulli setting, we can do better and use the kl version directly, benefiting from the fact that $g_\beta(\mu_\beta) = p$ for all β. The better bound becomes

$$\Pr\left[\hat{\mu}_\beta \; > \; \mu_\beta + \varepsilon\right] \; \leq \; \exp(\, - n \, \mathrm{kl}\,(p + i(X, \varepsilon) \;||\; p)\,)$$

Finally, we can derive a uniform concentration bound:

Theorem 5.7. Under the assumptions of Lemma 5.5,

$$\Pr\left(\sup_\beta |\hat{\mu}_\beta - \mu_\beta| > \; \varepsilon\right) \; \leq \; 2\exp(\, - 2n \cdot i(X, \varepsilon)^2)$$

This theorem uses the Dvoretzky–Kiefer–Wolfowitz inequality to first bound the empirical cumulative distribution function uniformly, and then uses the Lipschitz property of the exponential utility with respect to the cdf and Corollary 5.6 to conclude.

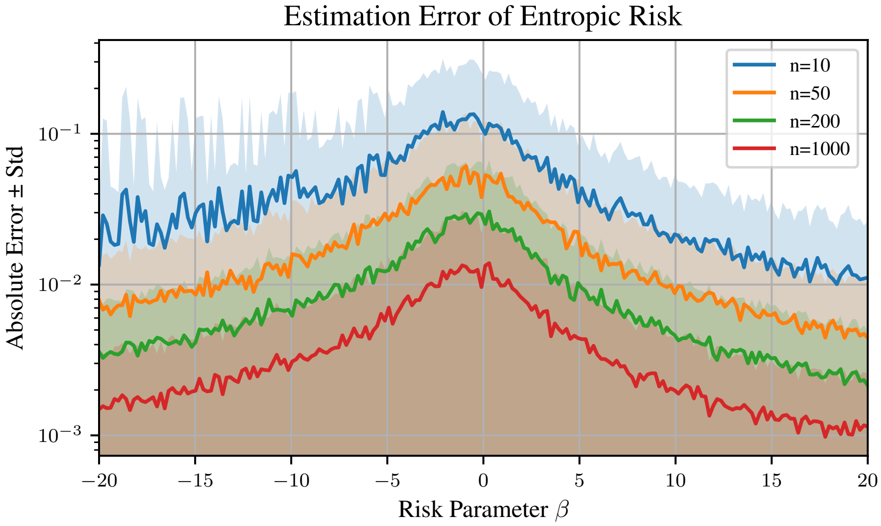

Figure 5.2: Convergence of the function $\beta \mapsto \hat{\mu}_\beta$ to $\beta \mapsto \mu_\beta$ for a Bernoulli variable with parameter $p = 0.6$. The different curves represent the estimation for different sample sizes n. We can see that the convergence is uniform over all values of β.

EVaR estimation We recall the definition we use for EVaR:

Definition 5.2. For X a random variable, we write $\mu_\beta = U_\beta(X)$.

$$\mathrm{EVaR}_\alpha[X] = \sup_{\beta<0} \mu_\beta - \frac{1}{\beta} \log \alpha$$

As suggested in Section 4.1, EVaR is an important risk measure establishing a connection between entropic risk measure and VaR/CVaR. The principal issue is needing to optimize over all values of $\beta < 0$ to compute it.

We give here a few results showing how to estimate EVaR from the estimation of the EntRM. First, uniform estimation of the EntRM directly gives a bound on the estimation of EVaR:

Proposition 5.3. Consider $\widehat{\mathrm{EVaR}}_\alpha = \sup\limits_{\beta<0} \hat{\mu}_\beta - \frac{1}{\beta} \log \alpha$ where $\hat{\mu}_\beta$ is our usual estimator. Then,

$$|\widehat{\mathrm{EVaR}}_\alpha - \mathrm{EVaR}_\alpha| \leq \sup_{\beta<0} |\hat{\mu}_\beta - \mu_\beta|$$

This directly gives a bound for the EVaR of discrete variables using Theorem 5.7.

Theorem 5.8. For any variable X such that $P(X = 0) > 0$, and for any $\varepsilon > 0$, we have

$$\Pr\left(|\mathrm{EVaR}_\alpha - \widehat{\mathrm{EVaR}}_\alpha| > \varepsilon\right) \leq 4 \exp(-2n \cdot i(X, \varepsilon)^2)$$

This result however does not work for general variables as the example of the Uniform variable in Example 5.1 shows that the uniform concentration bound does not hold in general. We are unfortunately unable to provide an explicit bound that holds for all variables, but we can provide a bound that holds by restricting the range of β to only a finite interval.

Estimation on a finite interval While the function $\beta \mapsto g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta)$ may not be bounded above 0 when β goes to $-\infty$, it does remain bounded on any finite interval $[B, 0]$ with $B < 0$.

Proposition 5.4. Let $\varepsilon > 0$ and $B < 0$. Consider any random variable X with support in $[0, 1]$. it holds that

$$\forall \beta \in [B, 0], \varepsilon > 0, \quad g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta) > \frac{Be^B}{e^B - 1}\,.$$

Thus,

$$\Pr(\sup_{B \leq \beta < 0} |\hat{\mu}_\beta - \mu_\beta| > \varepsilon) \leq 2 \exp\left(-4n \left[\frac{Be^B}{e^B - 1}\varepsilon\right]^2\right)$$

This bound ensures an exponential decay of the estimation error when estimating the EntRM on a finite interval of β. A similar bound can be derived for $B > 0$, where the constant becomes $\frac{B}{e^B - 1}$. In both cases, the constant scales exponentially with $-|B|$, which leads to very slow convergence when $|B|$ is large.

Truncated EVaR estimation We now try to apply this result to the estimation of EVaR. We consider the truncated version of EVaR where the optimization over β is only done on a finite interval $[B, 0]$ for some $B < 0$.

Consider

- $\mathrm{EVaR}_\alpha^B = \sup_{B \leq \beta < 0} \mu_\beta - \frac{1}{\beta} \log \alpha$, the *truncated* EVaR, and
- $\widehat{\mathrm{EVaR}}_\alpha^B = \sup_{B \leq \beta < 0} \hat{\mu}_\beta - \frac{1}{\beta} \log \alpha$, the estimator.

We can state the following results relating the truncated EVaR to the regular EVaR and their estimations:

Proposition 5.5. Proposition 5.3 still holds for the truncated version:

$$|\widehat{\mathrm{EVaR}}_\alpha^B - \mathrm{EVaR}_\alpha^B| < \sup_{B < \beta < 0} |\hat{\mu}_\beta - \mu_\beta| \tag{5.8}$$

Also, we can bound the difference with the truncated version

$$\mathrm{EVaR}_\alpha^B \leq \mathrm{EVaR}_\alpha \leq \mathrm{EVaR}_\alpha^B + \frac{1}{B}\log(\alpha) \tag{5.9}$$

And we can combine both results to obtain the following bound:

$$|\mathrm{EVaR}_\alpha - \widehat{\mathrm{EVaR}}_\alpha^B| \leq \sup_{B<\beta<0}\{|\hat{\mu}_\beta - \mu_\beta|\} + \frac{1}{B}\log(\alpha) \tag{5.10}$$

With such result, we can derive confidence bounds for the estimation of EVaR. The bound will depend on the choice of $B < 0$, trading-off the approximation error due to truncation and the estimation error due to the concentration bound.

Theorem 5.9. For any $1 > \delta > 0$, we have,

$$\Pr\left(\mathrm{EVaR}_\alpha > \inf_{B<0} \widehat{\mathrm{EVaR}}_\alpha^B + \frac{1}{B}\log(\alpha) + \frac{e^B - 1}{Be^B}\sqrt{\frac{\log(2/\delta)}{4n}}\right) \leq \delta$$

However, because of both the scaling of the constant $\frac{e^B-1}{Be^B}$ in Proposition 5.4 and the additive term $\frac{1}{B}\log(\alpha)$ in Proposition 5.5, this bound is not very practical as it requires a very large number of samples to be non-vacuous for reasonable values of B and α. While still converging to 0 as n goes to infinity, the convergence of the confidence interval is very slow, and not in a $1/\sqrt{n}$ fashion as one would expect from usual concentration bounds. From the experimental results, it seems that the rate of convergence cannot be expressed as the inverse of a polynomial in n, confirming the impracticality of this bound.

5.3 Proofs of Section 5.1

In all of the following proofs, we will only consider the case where $\beta > 0$. The case where $\beta < 0$ can be treated similarly each time with only minor modifications, but is omitted to avoid redundancy. The case $\beta = 0$ is not addressed as all the results reduce to the classical results for the sample mean.

Proof of Proposition 5.1. Let $Z_i := e^{\beta X_i}$ and $Z := e^{\beta X}$. By assumption, $\mathbb{E}[Z] < \infty$. By the strong law of large numbers,

$$\hat{m}_n = \frac{1}{n}\sum_{i=1}^{n} Z_i \xrightarrow[n\to\infty]{\text{a.s.}} \mathbb{E}[Z] = \mathbb{E}[e^{\beta X}].$$

The function $g : (0, \infty) \to \mathbb{R}$ defined by $g(m) := \frac{1}{\beta}\log m$ is continuous. By the continuous mapping theorem,

$$\hat{\mu}_\beta^n = g(\hat{m}_n) \xrightarrow[n\to\infty]{\text{a.s.}} g(\mathbb{E}[e^{\beta X}]) = \mu_\beta,$$

which proves the claim. □

Proof of Theorem 5.1

Proof of Theorem 5.1. We start from the formal definition of the error between our estimator and the true value of the β-mean function and apply the Chernoff bound:

$$\begin{aligned}
&\Pr\left[\frac{1}{\beta}\log\frac{\sum_{i=1}^{n} e^{\beta X_i}}{n} \;-\; \frac{1}{\beta}\log\mathbb{E}[e^{\beta X}] > \varepsilon\right]\\
&= \Pr\left[\sum_{i=1}^{n} e^{\beta X_i} \;>\; n\cdot e^{\beta\varepsilon}\cdot\mathbb{E}[e^{\beta X}]\right]\\
&\le \inf_{t>0}\exp\Big(-t\,e^{\beta\varepsilon}\,n\,\mathbb{E}[e^{\beta X}]\Big) \;\times\; \mathbb{E}\left[\exp\Big(t\sum_{i=1}^{n} e^{\beta X_i}\Big)\right]\\
&= \inf_{t>0}\exp\left(-t\,e^{\beta\varepsilon}\,n\,\mathbb{E}[e^{\beta X}] \;+\; \log\mathbb{E}\Big[\exp\Big(t\sum_{i=1}^{n} e^{\beta X_i}\Big)\Big]\right)\\
&\overset{(a)}{=} \inf_{t>0}\exp\left(-n\Big(t\,e^{\beta\varepsilon}\mathbb{E}[e^{\beta X}] - \underbrace{\log\mathbb{E}[\exp(t\,e^{\beta X})]}_{\psi_Z(t)}\Big)\right)
\end{aligned}$$

where (a) follows by the i.i.d. assumption. For notational convenience, define the new random variable $Z = e^{\beta X}$. Let $\psi_Z(t)$ be the cumulant generating function (CGF) of Z. Rather than finding the infimum, we focus on the equivalent problem of maximizing the term inside the exponent:

$$\sup_t\Big(t\,e^{\beta\varepsilon}\mathbb{E}[e^{\beta X}] - \psi_Z(t)\Big) = \psi_Z^*\Big(e^{\beta\varepsilon}\mathbb{E}[e^{\beta X}]\Big)$$

Finally, we obtain the following upper bound for our estimator:

$$\Pr\left[\frac{1}{\beta}\log\frac{\sum_{i=1}^{n} e^{\beta X_i}}{n} \;-\; \frac{1}{\beta}\log\mathbb{E}[e^{\beta X}] \;>\; \varepsilon\right] \;\le\; \exp\Big(-n\,\psi_Z^*\Big(e^{\beta\varepsilon}\mathbb{E}[e^{\beta X}]\Big)\Big),$$

□

Proof of Lemma 5.2 We begin by proving an additional lemma that will be useful in the derivation.

Lemma 5.10. Let $X \sim \mathrm{Ber}(p)$. For any $\beta \in \mathbb{R}, \varepsilon > 0, n \in \mathbb{N}$, the following holds:

$$\inf_{\lambda>0}\left\{\exp\Big(-\lambda\,n\,e^{\beta\varepsilon}\,\mathbb{E}[e^{\beta X}]\Big)\Big(\mathbb{E}\left[\exp(\lambda e^{\beta X})\right]\Big)^n\right\} \le \exp(-n\,\mathrm{kl}(q\|p)) \qquad (5.11)$$

where $q = \frac{\mathbb{E}[e^{\beta(X+\varepsilon)}]-1}{e^{\beta}-1}$.

Proof. Let $Y = e^{\beta X}$. Since $X \sim \mathrm{Ber}(p)$, the variable Y takes values in $\{1, e^{\beta}\}$ with probabilities $1-p$ and p respectively. Define $C = e^{\beta\varepsilon}\mathbb{E}[Y]$. The expression to be minimized in (5.11) can be written as $\exp(n\Psi(\lambda))$, where:

$$\Psi(\lambda) = -\lambda C + \log \mathbb{E}[e^{\lambda Y}] = -\lambda C + \log(pe^{\lambda e^{\beta}} + (1-p)e^{\lambda}).$$

Minimizing the LHS of (5.11) is equivalent to minimizing $\Psi(\lambda)$. Furthermore, the CGF of a random variable is always convex, hence Ψ is convex. Differentiating $\Psi(\lambda)$ with respect to λ yields:

$$\Psi'(\lambda) = -C + \frac{pe^{\beta}e^{\lambda e^{\beta}} + (1-p)e^{\lambda}}{pe^{\lambda e^{\beta}} + (1-p)e^{\lambda}}.$$

Setting $\Psi'(\lambda) = 0$ to find the optimal λ^*, and factoring out e^{λ} from the ratio, we obtain:

$$C = \frac{pe^{\beta}e^{\lambda(e^{\beta}-1)} + (1-p)}{pe^{\lambda(e^{\beta}-1)} + (1-p)}.$$

Let $z = e^{\lambda(e^{\beta}-1)}$. Solving the equation for z gives:

$$z = \frac{1-p}{p} \cdot \frac{C-1}{e^{\beta} - C}.$$

Recall the definition $q = \frac{\mathbb{E}[e^{\beta(X+\varepsilon)}]-1}{e^{\beta}-1} = \frac{C-1}{e^{\beta}-1}$. We observe that $C - 1 = q(e^{\beta} - 1)$ and $e^{\beta} - C = (e^{\beta} - 1)(1-q)$. Substituting these into the expression for z, we find the optimal value z^*:

$$z^* = \frac{1-p}{p}\frac{q}{1-q}.$$

To obtain the final bound, we substitute λ^* back into $\Psi(\lambda)$. Note that $\lambda = \frac{\log z}{e^{\beta}-1}$, which implies $\lambda(1-C) = -q \log z$. We rewrite $\Psi(\lambda)$ as:

$$\begin{aligned}\Psi(\lambda) &= -\lambda C + \log(e^{\lambda}(pz + 1 - p)) \\ &= \lambda(1-C) + \log(pz + 1 - p).\end{aligned}$$

Evaluating at z^*:

$$\begin{aligned}\Psi(\lambda^*) &= -q \log z^* + \log\left(p \cdot \frac{1-p}{p}\frac{q}{1-q} + 1 - p\right) \\ &= -q \log\left(\frac{1-p}{p}\frac{q}{1-q}\right) + \log\left(\frac{1-p}{1-q}\right) \\ &= -q \log \frac{q}{p} - q \log \frac{1-p}{1-q} + \log \frac{1-p}{1-q} \\ &= -q \log \frac{q}{p} - (1-q) \log \frac{1-q}{1-p} \\ &= -\mathrm{kl}(q \| p).\end{aligned}$$

Exponentiating $n\Psi(\lambda^*)$ concludes the proof. □

We now prove Lemma 5.2 using the above lemma.

Proof of Lemma 5.2. We start from the formal definition of the error between our estimator and the entropic risk and apply the Chernoff bound:

$$\begin{aligned}
\Pr\left[\frac{1}{\beta}\log\frac{\sum_{i=1}^{n} e^{\beta X_i}}{n} \;-\; \frac{1}{\beta}\log\mathbb{E}[e^{\beta X}] \;>\; \varepsilon\right]& \\
&= \Pr\left[\sum_{i=1}^{n} e^{\beta X_i} \;>\; n e^{\beta\varepsilon}\mathbb{E}[e^{\beta X}]\right] \\
&\le \inf_{\lambda>0} \exp\Big(-\lambda\, n\, e^{\beta\varepsilon}\,\mathbb{E}[e^{\beta X}]\Big) \;\times\; \mathbb{E}\left[\exp\Big(\lambda\sum_{i=1}^{n} e^{\beta X_i}\Big)\right] \\
&\overset{(a)}{=} \inf_{\lambda>0} \exp\Big(-\lambda\, n\, e^{\beta\varepsilon}\,\mathbb{E}[e^{\beta X}]\Big) \times \Big(\mathbb{E}\left[\exp(\lambda e^{\beta X})\right]\Big)^{n} \\
&\overset{(b)}{\le} \exp(-n\,\mathrm{kl}(q\|p))
\end{aligned}$$

with (a) following from the i.i.d. assumption and (b) from Lemma 5.10. □

Proof of Theorem 5.3 We first prove a lemma that will allow us to reduce the problem to the Bernoulli case.

Lemma 5.11 (Bernoulli domination)**.** Let X be a random variable with support contained in $[0,1]$. Write $\mu_\beta = \frac{1}{\beta}\log\mathbb{E}[\exp(\beta X)]$. Consider $X_b \sim \mathrm{Ber}\left(\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}\right)$. Then, the following holds:

$$\forall\beta,\lambda, \qquad \mathbb{E}\left[e^{\lambda e^{\beta X}}\right] \le \mathbb{E}\left[e^{\lambda e^{\beta X_b}}\right] \tag{5.12}$$

$$\forall\beta, \qquad \mathbb{E}[e^{\beta X}] = \mathbb{E}[e^{\beta X_b}] \tag{5.13}$$

Proof. For the first part, we use the convexity of the function $x \mapsto e^{\lambda x}$,

$$\forall x\in[1,e^{\beta}], \quad e^{\lambda x} \le \left(1-\frac{x-1}{e^{\beta}-1}\right)e^{\lambda} + \frac{x-1}{e^{\beta}-1}e^{\lambda e^{\beta}}.$$

Replacing x by $e^{\beta X}$, we obtain

$$e^{\lambda e^{\beta X}} \le \left(1-\frac{e^{\beta X}-1}{e^{\beta}-1}\right)e^{\lambda} + \frac{e^{\beta X}-1}{e^{\beta}-1}e^{\lambda e^{\beta}}.$$

Finally, taking the expectation of both sides,

$$\begin{aligned}
\mathbb{E}\left[e^{\lambda e^{\beta X}}\right] &\leq \left(1-\frac{\mathbb{E}\left[e^{\beta X}\right]-1}{e^{\beta}-1}\right)e^{\lambda}+\frac{\mathbb{E}\left[e^{\beta X}\right]-1}{e^{\beta}-1}e^{\lambda e^{\beta}}\\
&=\left(1-\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}\right)e^{\lambda}+\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}e^{\lambda e^{\beta}}\\
&=\mathbb{E}\left[e^{\lambda e^{\beta X_b}}\right].
\end{aligned}$$

For the second part, we have

$$\begin{aligned}
\mathbb{E}[e^{\beta X_b}] &= \left(1-\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}\right)+\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}e^{\beta}\\
&=\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}(e^{\beta}-1)+1\\
&=e^{\beta\mu_\beta}\\
&=\mathbb{E}[e^{\beta X}].
\end{aligned}$$

□

We now prove Theorem 5.3.

Proof of Theorem 5.3. Using the same Chernoff bound technique as before, and using the Chernoff bound for Bernoulli variables,

$$\begin{aligned}
&\Pr\left[\frac{1}{\beta}\log\frac{\sum_{i=1}^{n}e^{\beta X_i}}{n}\;-\;\frac{1}{\beta}\log\mathbb{E}[e^{\beta X}]\;>\;\varepsilon\right]\\
&=\Pr\left[\sum_{i=1}^{n}e^{\beta X_i}\;>\;n\cdot e^{\beta\varepsilon}\cdot\mathbb{E}[e^{\beta X}]\right]\\
&\leq\inf_{\lambda>0}\exp\Big(-\lambda\,e^{\beta\varepsilon}\,n\,\mathbb{E}[e^{\beta X}]\Big)\;\times\;\mathbb{E}\left[\exp\Big(\lambda\sum_{i=1}^{n}e^{\beta X_i}\Big)\right]\\
&=\inf_{\lambda>0}\exp\Big(-\lambda\,e^{\beta\varepsilon}\,n\,\mathbb{E}[e^{\beta X}]\Big)\;\times\;\Big(\mathbb{E}\left[\exp\Big(\lambda e^{\beta X}\Big)\right]\Big)^{n}\\
&\leq\inf_{\lambda>0}\exp\Big(-\lambda\,e^{\beta\varepsilon}\,n\,\mathbb{E}[e^{\beta X_b}]\Big)\;\times\;\Big(\mathbb{E}\left[\exp\Big(\lambda e^{\beta X_b}\Big)\right]\Big)^{n}\\
&\overset{(Lem. 5.10)}{\leq}\exp\left(-n\,\mathrm{kl}\left(\frac{\mathbb{E}[e^{\beta(X_b+\varepsilon)}]-1}{e^{\beta}-1}\middle\|\mathbb{E}[X_b]\right)\right)\\
&=\exp\left(-n\,\mathrm{kl}\left(g_\beta(\mu_\beta+\varepsilon)\|g_\beta(\mu_\beta)\right)\right)
\end{aligned}$$

where the last equality holds since

$$\mathbb{E}[X_b]=\frac{e^{\beta\mu_\beta}-1}{e^{\beta}-1}=g_\beta(\mu_\beta)$$

and

$$\mathbb{E}[e^{\beta(X_b+\varepsilon)}] = e^{\beta\varepsilon}\mathbb{E}[e^{\beta X_b}] = e^{\beta\varepsilon}e^{\beta\mu_\beta} = e^{\beta(\mu_\beta+\varepsilon)}$$

so that

$$\frac{\mathbb{E}[e^{\beta(X_b+\varepsilon)}]-1}{e^\beta-1} = \frac{e^{\beta(\mu_\beta+\varepsilon)}-1}{e^\beta-1} = g_\beta(\mu_\beta+\varepsilon).$$

□

Proof of Corollary 5.4 The proof is a direct application of Pinsker's inequality on the KL divergence [Lattimore and Szepesvári, 2020] in Theorem 5.3. Pinsker's inequality states that for any two Bernoulli distributions with parameters $p, q \in [0,1]$,

$$\mathrm{kl}(q\|p) \geq 2(q-p)^2.$$

Applying this inequality in Theorem 5.3 concludes the proof.

5.4 Proofs of Section 5.2

The KL bound may be vacuous at the limit We prove here the statement in Section 5.2 claiming that the kl bound for a uniform distribution is vacuous when $\beta \to -\infty$.

Proof. Consider X following the uniform distribution on $[0,1]$. Recall that $\mathbb{E}[e^{\beta X}] = \frac{e^\beta-1}{\beta}$. Then,

$$\begin{aligned} g_\beta(\mu_\beta) &= \frac{1}{\beta} + \frac{1}{1-e^\beta} \underset{\beta\to-\infty}{\to} 1 \\ g_\beta(\mu_\beta+\varepsilon) &= \frac{e^{\beta\varepsilon}}{\beta} + \frac{1}{1-e^\beta} \underset{\beta\to-\infty}{\to} 1 \end{aligned}$$

We then consider

$$\mathrm{kl}(g_\beta(\mu_\beta+\varepsilon) \mid\mid g_\beta(\mu_\beta)) = g_\beta(\mu_\beta+\varepsilon)\log\left(\frac{g_\beta(\mu_\beta+\varepsilon)}{g_\beta(\mu_\beta)}\right) + (1-g_\beta(\mu_\beta+\varepsilon))\log\left(\frac{1-g_\beta(\mu_\beta+\varepsilon)}{1-g_\beta(\mu_\beta)}\right)$$

For the left part we have

$$\underbrace{g_\beta(\mu_\beta+\varepsilon)}_{\to 1}\underbrace{\log\left(\frac{\overbrace{g_\beta(\mu_\beta+\varepsilon)}^{\to 1}}{\underbrace{g_\beta(\mu_\beta)}_{\to 1}}\right)}_{\to 0} \to 0$$

The right part requires careful analysis as we have the indeterminate limit $\frac{0}{0}$ inside the logarithm.

$\frac{1}{1-e^{\beta}} = 1 + O(e^{\beta})$, hence $1 - g_\beta(\mu_\beta) = \frac{1}{-\beta} + O(e^{\beta})$ and $1 - g_\beta(\mu_\beta + \varepsilon) = \frac{e^{\beta\varepsilon}}{-\beta} + O(e^{\beta})$. Thus,

$$\log\left(\frac{1 - g_\beta(\mu_\beta + \varepsilon)}{1 - g_\beta(\mu_\beta)}\right) = \log(e^{\beta\varepsilon} + O(\beta e^{\beta})) = \beta\varepsilon + O(\beta e^{\beta(1-\varepsilon)}).$$

From this, we obtain

$$\begin{aligned}(1 - g_\beta(\mu_\beta + \varepsilon)) \log\left(\frac{1 - g_\beta(\mu_\beta + \varepsilon)}{1 - g_\beta(\mu_\beta)}\right) &= \left(\frac{e^{\beta\varepsilon}}{-\beta} + O(e^{\beta})\right)(\beta\varepsilon + O(\beta e^{\beta(1-\varepsilon)}))\\ &= \varepsilon e^{\beta\varepsilon} + O(\beta e^{\beta})\\ &\to 0\end{aligned}$$

□

Proof of Proposition 5.2

Proof of Proposition 5.2. We want to bound the quantity

$$\lim_{\beta\to-\infty} \inf_{\lambda>0} \exp\left(\lambda \mathbb{E}[e^{\beta X}] e^{\beta\varepsilon} + n \log\left(\mathbb{E}\left[e^{-\lambda e^{\beta X}}\right]\right)\right)$$

For any $\beta < 0$, let $\lambda_\beta = -\frac{e^{-\beta\varepsilon}}{\beta}$. We consider the term inside the exponential:

$$\begin{aligned}&\lambda_\beta n \mathbb{E}[e^{\beta X}] e^{\beta\varepsilon} + n \log \mathbb{E}[e^{-\lambda_\beta e^{\beta X}}]\\ &\qquad = \frac{n}{|\beta|}\mathbb{E}[e^{\beta X}] + n \log \mathbb{E}[e^{\frac{1}{\beta} e^{\beta(X-\varepsilon)}}]\\ &\qquad = \underbrace{\frac{n}{|\beta|}\mathbb{E}[e^{\beta X}]}_{\text{(a)}} + n \log\left(\underbrace{\mathbb{E}[e^{\frac{1}{\beta} e^{\beta(X-\varepsilon)}} \mathbb{1}\{X < \varepsilon\}]}_{\text{(b)}} + \underbrace{\mathbb{E}[e^{\frac{1}{\beta} e^{\beta(X-\varepsilon)}} \mathbb{1}\{X \geq \varepsilon\}]}_{\text{(c)}}\right)\end{aligned}$$

We now compute the limit of each term separately. For the first **(a)** term, we have

$$\mathbb{E}[e^{\beta X}] = \underbrace{\mathbb{E}[e^{\beta X} \mathbb{1}\{X = 0\}]}_{=P(X=0)} + \mathbb{E}[e^{\beta X} \mathbb{1}\{X \neq 0\}] \,.$$

By the dominated convergence theorem, $\mathbb{E}[e^{\beta X} \mathbb{1}\{X \neq 0\}] \to 0$ when $\beta \to -\infty$. Hence, $\mathbb{E}[e^{\beta X}] \to P(X = 0)$ and

$$\text{(a)} = \frac{n}{|\beta|}\mathbb{E}[e^{\beta X}] \to 0.$$

Next we consider term **(b)**. Note that when $X < \varepsilon$, $\frac{1}{\beta}e^{\beta(X-\varepsilon)} \to -\infty$. Hence, $e^{\frac{1}{\beta}e^{\beta(X-\varepsilon)}}\mathbb{1}\{X < \varepsilon\} \to 0$. By the dominated convergence theorem, we have

$$\mathbb{E}[e^{\frac{1}{\beta}e^{\beta(X-\varepsilon)}}\mathbb{1}\{X < \varepsilon\}] \to 0 \ .$$

Finally, we consider **(c)**. When $X \geq \varepsilon$, $\frac{1}{\beta}e^{\beta(X-\varepsilon)} \to 0$. Hence, $e^{\frac{1}{\beta}e^{\beta(X-\varepsilon)}}\mathbb{1}\{X \geq \varepsilon\} \to 1$. By the dominated convergence theorem, we have

$$\mathbb{E}[e^{\frac{1}{\beta}e^{\beta(X-\varepsilon)}}\mathbb{1}\{X \geq \varepsilon\}] \to P(X \geq \varepsilon).$$

Combining these results, we have

$$\begin{aligned}
&\lim_{\beta\to-\infty} \inf_{\lambda>0} \exp\left(\lambda\mathbb{E}[e^{\beta X}]e^{\beta\varepsilon} + n\log\left(\mathbb{E}\left[e^{-\lambda e^{\beta X}}\right]\right)\right) \\
&\quad \leq \lim_{\beta\to-\infty} \exp\left(\frac{n}{\beta}\mathbb{E}[e^{\beta X}] + n\log\left(\mathbb{E}[e^{\frac{1}{\beta}e^{\beta(X-\varepsilon)}}\mathbb{1}\{X < \varepsilon\}] + \mathbb{E}[e^{\frac{1}{\beta}e^{\beta(X-\varepsilon)}}\mathbb{1}\{X \geq \varepsilon\}]\right)\right) \\
&\quad = \exp(n\log P(X \geq \varepsilon)) = P(X \geq \varepsilon)^n
\end{aligned}$$

□

Proof of Lemma 5.5

Proof of Lemma 5.5. First, notice that $g_\beta : x \mapsto \frac{e^{\beta x}-1}{e^\beta - 1}$ is strictly increasing in x for any $\beta \in \mathbb{R}$. Hence, $\forall\beta, g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta) > 0$. Furthermore, we can compute the limits when $\beta \to +\infty$ and $\beta \to -\infty$. We start with $g_\beta(\mu_\beta)$:

$$\begin{aligned}
g_\beta(\mu_\beta) &= \frac{e^{\beta\mu_\beta} - 1}{e^\beta - 1} = \frac{\mathbb{E}[e^{\beta X}] - 1}{e^\beta - 1} = \frac{\mathbb{E}[e^{\beta X}\mathbb{1}\{X = 1\}] + \mathbb{E}[e^{\beta X}\mathbb{1}\{X \neq 1\}] - 1}{e^\beta - 1} \\
&= \frac{P(X = 1)e^\beta + \mathbb{E}[e^{\beta X}\mathbb{1}\{X \neq 1\}] - 1}{e^\beta - 1} \\
&= P(X = 1) + \frac{\mathbb{E}[e^{\beta X}\mathbb{1}\{X \neq 1\}] - 1}{e^\beta - 1} \\
&\underset{\beta\to+\infty}{=} P(X = 1) + O(\underbrace{\mathbb{E}[e^{\beta(X-1)}\mathbb{1}\{X \neq 1\}]}_{\to 0}) \\
&\underset{\beta\to+\infty}{=} P(X = 1)
\end{aligned}$$

and

$$\begin{aligned}
g_\beta(\mu_\beta) &= \frac{e^{\beta\mu_\beta}-1}{e^\beta-1} = \frac{\mathbb{E}[e^{\beta X}]-1}{e^\beta-1} = \frac{\mathbb{E}[e^{\beta X}\mathbb{1}\{X=0\}]+\mathbb{E}[e^{\beta X}\mathbb{1}\{X\neq 0\}]-1}{e^\beta-1}\\
&= \frac{P(X=0)+\mathbb{E}[e^{\beta X}\mathbb{1}\{X\neq 0\}]-1}{e^\beta-1}\\
&= \underbrace{\frac{P(X=0)-1}{e^\beta-1}}_{\to 1-P(X=0)\text{ when }\beta\to-\infty} + \underbrace{\frac{\mathbb{E}[e^{\beta X}\mathbb{1}\{X\neq 0\}]}{e^\beta-1}}_{\to 0\text{ when }\beta\to-\infty}\\
&\underset{\beta\to-\infty}{=} 1-P(X=0)
\end{aligned}$$

We remark that these limits are consistent with the fixed value $g_\beta(\mu_\beta)=p$ for Bernoulli variables, and $P(X=1)=1-P(X=0)=p$ in that case.

For $g_\beta(\mu_\beta+\varepsilon)$, we have

$$\begin{aligned}
g_\beta(\mu_\beta+\varepsilon) &= \frac{e^{\beta(\mu_\beta+\varepsilon)}-1}{e^\beta-1} = \frac{\mathbb{E}[e^{\beta(X+\varepsilon)}]-1}{e^\beta-1}\\
&\geq \frac{e^{\beta(1+\varepsilon)}P(X=1)-1}{e^\beta-1}\\
&\underset{\beta\to+\infty}{=} +\infty
\end{aligned}$$

since $P(X=1)>0$ by assumption. Finally,

$$\begin{aligned}
g_\beta(\mu_\beta+\varepsilon) &= \frac{e^{\beta(\mu_\beta+\varepsilon)}-1}{e^\beta-1} = \frac{\mathbb{E}[e^{\beta(X+\varepsilon)}]-1}{e^\beta-1}\\
&\underset{\beta\to-\infty}{=} 1
\end{aligned}$$

Hence, the function $\beta\mapsto g_\beta(\mu_\beta+\varepsilon)-g_\beta(\mu_\beta)$ has non-zero limits on both sides. As it is also continuous, there exists β_0 such that the infimum is reached for that specific β_0. In particular, we have $i(X,\varepsilon)=\inf_{\beta\in\mathbb{R}} g_\beta(\mu_\beta+\varepsilon)-g_\beta(\mu_\beta)>0$. □

Proof of Corollary 5.6

Proof of Corollary 5.6. It boils down to a simple application of Lemma 5.5 in Corollary 5.4. For any X bounded in $[0,1]$, $\beta\in\mathbb{R}$ and $\varepsilon>0$, we have

$$\begin{aligned}
\Pr\left[\hat{\mu}_\beta > \mu_\beta+\varepsilon\right] &\leq \exp(-2n(g_\beta(\mu_\beta+\varepsilon)-g_\beta(\mu_\beta))^2)\\
&\leq \exp(-2n\cdot i(X,\varepsilon)^2)
\end{aligned}$$

□

Proof of Theorem 5.7

Proof. We begin by recalling the Dvoretzky–Kiefer–Wolfowitz (DKW) inequality [Massart, 1990], which provides a concentration bound for the empirical cumulative distribution function F_n around the true CDF F. For any $\eta > 0$:

$$\Pr\Big(\|F_n - F\|_\infty > \eta\Big) \leq 2\exp(-2n\eta^2). \tag{5.14}$$

Let $\varepsilon > 0$. We aim to bound the probability of the event $\Big\{\sup_\beta(\hat{\mu}_\beta - \mu_\beta) \geq \varepsilon\Big\}$.

First, we establish a deterministic implication between the distance of the CDFs and the error in the entropic risk. Let $\hat{E}_\beta = e^{\beta\hat{\mu}_\beta}$ and $E_\beta = e^{\beta\mu_\beta}$. Recall the Lipschitz property of the exponential utility with respect to the Kolmogorov distance [Liang and Luo, 2024]:

$$|\hat{E}_\beta - E_\beta| \leq |e^\beta - 1| \cdot \|F_n - F\|_\infty. \tag{5.15}$$

Assume that the empirical CDF satisfies the condition $\|F_n - F\|_\infty \leq i(X,\varepsilon)$. By the definition of $i(X,\varepsilon)$, we have:

$$\|F_n - F\|_\infty \leq \inf_{\beta'}\left[g_{\beta'}(\mu_{\beta'} + \varepsilon) - g_{\beta'}(\mu_{\beta'})\right] \leq g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta),$$

for any fixed β. Recalling that $g_\beta(y) = \frac{e^{\beta y}-1}{e^\beta - 1}$, this inequality can be rewritten for $\beta > 0$ as:

$$\|F_n - F\|_\infty \leq \frac{e^{\beta(\mu_\beta+\varepsilon)} - e^{\beta\mu_\beta}}{e^\beta - 1} = \frac{e^{\beta\varepsilon} - 1}{e^\beta - 1}E_\beta.$$

Multiplying by $(e^\beta - 1)$ (which is positive for $\beta > 0$) and combining with (5.15), we obtain:

$$\hat{E}_\beta - E_\beta \leq |\hat{E}_\beta - E_\beta| \leq |e^\beta - 1| \cdot \|F_n - F\|_\infty \leq (e^{\beta\varepsilon} - 1)E_\beta.$$

Rearranging the terms yields $\hat{E}_\beta \leq e^{\beta\varepsilon}E_\beta$, or equivalently $e^{\beta\hat{\mu}_\beta} \leq e^{\beta(\mu_\beta+\varepsilon)}$. Taking the logarithm and dividing by β gives $\hat{\mu}_\beta \leq \mu_\beta + \varepsilon$.

The same logic holds for $\beta < 0$ and for $\mu_\beta - \hat{\mu}_\beta \geq \varepsilon$, with appropriate sign adjustments in the intermediate steps, and leading to the same conclusion. Thus, the condition $\|F_n - F\|_\infty \leq i(X,\varepsilon)$ implies that $\sup_\beta|\hat{\mu}_\beta - \mu_\beta| \leq \varepsilon$.

Finally, using the contrapositive of this implication and the DKW inequality (5.14) with $\eta = i(X,\varepsilon)$, we conclude:

$$\begin{aligned}\Pr\left(\sup_\beta|\hat{\mu}_\beta - \mu_\beta| > \varepsilon\right) &\leq \Pr\left(\|F_n - F\|_\infty > i(X,\varepsilon)\right)\\ &\leq 2\exp(-2n \cdot i(X,\varepsilon)^2).\end{aligned}$$

□

Proof of Proposition 5.5

Proof of Proposition 5.5. We prove separately the three equations:

Equation (5.8): The proof is similar as in Proposition 5.3.

$$\begin{aligned}|\widehat{\mathrm{EVaR}}_\alpha - \mathrm{EVaR}_\alpha| &= |\sup_{B<\beta<0}\{\hat{\mu}_\beta - \frac{1}{\beta}\log\alpha\} - \sup_{B<\beta<0}\{\mu_\beta - \frac{1}{\beta}\log\alpha\}| \\ &\leq \sup_{B<\beta<0}|\hat{\mu}_\beta - \frac{1}{\beta}\log\alpha - \mu_\beta + \frac{1}{\beta}\log\alpha| \\ &= \sup_{B<\beta<0}|\hat{\mu}_\beta - \mu_\beta|\end{aligned}$$

Equation (5.9): The proof comes from Lemma D.6 in [Hau et al., 2023b].

We write $h(\beta) = \mu_\beta - \frac{1}{\beta}\log\alpha$ so that $\mathrm{EVaR}_\alpha = \sup_{\beta<0} h(\beta)$ and $\mathrm{EVaR}_\alpha^B = \sup_{B<\beta<0} h(\beta)$.

$$\begin{aligned}\sup_{\beta<B} h(\beta) - h(B) &= \sup_{\beta<B}\mu_\beta - \frac{1}{\beta}\log\alpha - \mu_B + \frac{1}{B}\log\alpha \\ &\leq \sup_{\beta<B}\mu_\beta - \mu_B + \frac{1}{B}\log\alpha \\ &\leq \frac{1}{B}\log\alpha\end{aligned}$$

where the first inequality is due to the fact that $-\frac{1}{\beta}\log\alpha < 0$ and the second because $\mu_\beta - \mu_B < 0$ since $\beta < B$ and $\beta \mapsto \mu_\beta$ is a non-decreasing function [Hau et al., 2023b].

Hence,

$$\begin{aligned}\mathrm{EVaR}_\alpha &= \sup_{\beta<0}\mu_\beta - \frac{1}{\beta}\log\alpha \\ &= \max\{\sup_{\beta<B} h(\beta), \sup_{B\leq\beta<0} h(\beta)\} \\ &= \max\{\sup_{\beta<B} h(\beta), \mathrm{EVaR}_\alpha^B\} \\ &\leq \max\{h(B) + \frac{1}{B}\log\alpha, \mathrm{EVaR}_\alpha^B\} \\ &\leq \max\{\mathrm{EVaR}_\alpha^B + \frac{1}{B}\log\alpha, \mathrm{EVaR}_\alpha^B\} \\ &= \mathrm{EVaR}_\alpha^B + \frac{1}{B}\log\alpha\end{aligned}$$

The other side of the inequality is trivial by definition of EVaR and EVaR^B. □

Proof of Proposition 5.4

Proof of Proposition 5.4. The derivative of the function g_β with respect to x is given by $g'_\beta(x) = \frac{\beta e^{\beta x}}{e^\beta - 1}$. For $\beta < 0$, we observe that $\frac{\beta}{e^\beta - 1} > 0$ and that $x \mapsto e^{\beta x}$ is strictly decreasing. Consequently, g'_β is a decreasing function on $[0, 1]$, attaining its minimum at $x = 1$. Thus, we have:

$$\inf_{x\in(0,1)} g'_\beta(x) = g'_\beta(1) = \frac{\beta e^\beta}{e^\beta - 1}.$$

Then, by the Mean Value Theorem,

$$g_\beta(\mu_\beta + \varepsilon) - g_\beta(\mu_\beta) \geq \varepsilon \cdot \inf_{x\in(0,1)} g'_\beta(x) = \varepsilon \frac{\beta e^\beta}{e^\beta - 1}.$$

Substituting this lower bound into the concentration inequality of Corollary 5.6 gives the desired result. □

Proof of Theorem 5.9

Proof of Theorem 5.9. Fix $B < 0$ and define the constant $C(B) = \frac{Be^B}{e^B - 1} > 0$.

First, we have

$$2\exp(-4n[C(B)\varepsilon]^2) = \delta \iff \varepsilon = \frac{1}{C(B)}\sqrt{\frac{\log(2/\delta)}{4n}}.$$

Applying Proposition 5.4, we obtain the uniform concentration bound for the EntRM on the interval $[B, 0)$:

$$\Pr\left(\sup_{B\leq\beta<0} |\hat{\mu}_\beta - \mu_\beta| > \frac{1}{C(B)}\sqrt{\frac{\log(2/\delta)}{4n}}\right) \leq \delta. \tag{5.16}$$

Recall Proposition 5.5, which bounds the error of the truncated EVaR estimator:

$$|\mathrm{EVaR}_\alpha - \widehat{\mathrm{EVaR}}{}^B_\alpha| \leq \sup_{B\leq\beta<0} |\hat{\mu}_\beta - \mu_\beta| + \frac{1}{|B|}\log(\alpha).$$

Combining this with (5.16) yields:

$$\Pr\left(|\mathrm{EVaR}_\alpha - \widehat{\mathrm{EVaR}}{}^B_\alpha| > \frac{1}{|B|}\log(\alpha) + \frac{1}{C(B)}\sqrt{\frac{\log(2/\delta)}{4n}}\right) \leq \delta.$$

Since this bound holds for any fixed $B < 0$ chosen independently of the samples, we may choose the value of B that minimizes the upper bound to obtain the tightest confidence interval. □

5.5 Conclusion

In this chapter, we investigated the statistical estimation of the Entropic Risk Measure for bounded random variables. We derived non-asymptotic concentration inequalities for the plug-in estimator by adapting the classical Chernoff–Hoeffding bounds. Importantly, we extended this analysis to uniform estimation of the EntRM over a continuous range of risk parameters. To the best of our knowledge, this is the first work providing concentration guarantees for the entire entropic risk function. Our results demonstrate that the EntRM can be estimated uniformly over β with a sample complexity comparable to pointwise estimation. Furthermore, we leveraged these uniform bounds to establish the first concentration results for the Entropic Value at Risk, a risk measure that inherently requires optimization over the risk parameter. These results pave the way for designing reinforcement learning agents capable of optimizing policies across a continuum of risk preferences simultaneously.

References

Carlo Acerbi and Dirk Tasche. On the coherence of expected shortfall. *Journal of banking & finance*, 26(7):1487–1503, 2002.

Mastane Achab and Gergely Neu. Robustness and risk management via distributional dynamic programming. *arXiv preprint arXiv:2112.15430*, 2021.

Amir Ahmadi-Javid. Entropic value-at-risk: A new coherent risk measure. *Journal of Optimization Theory and Applications*, 155(3):1105–1123, 2012.

AGG Akritas, AWW Strzebonski, and PSS Vigklas. Improving the performance of the continued fractions method using new bounds of positive roots. *Nonlinear Analysis: Modelling and Control*, 13(3):265–279, 2008.

Maurice Allais. Allais paradox. In *Utility and probability*, pages 3–9. Springer, 1990.

Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete problems in ai safety. *arXiv preprint arXiv:1606.06565*, 2016.

Philippe Artzner, Freddy Delbaen, Jean-Marc Eber, and David Heath. Coherent measures of risk. *Mathematical finance*, 9(3):203–228, 1999.

Mikhail J Atallah. Some dynamic computational geometry problems. *Computers & Mathematics with Applications*, 11(12):1171–1181, 1985.

André Barreto, Shaobo Hou, Diana Borsa, David Silver, and Doina Precup. Fast reinforcement learning with generalized policy updates. *Proceedings of the National Academy of Sciences*, 117(48):30079–30087, 2020.

Basel Committee on Banking Supervision. Minimum capital requirements for market risk: Fundamental review of the trading book. Technical report, Bank for International Settlements, October 2013.

Osbert Bastani, Jason Yecheng Ma, Estelle Shen, and Wanqiao Xu. Regret bounds for risk-sensitive reinforcement learning. *Advances in Neural Information Processing Systems*, 35:36259–36269, 2022.

Nicole Bäuerle and Alexander Glauner. Markov decision processes with recursive risk measures. *European Journal of Operational Research*, 296(3):953–966, 2022.

Nicole Bäuerle and Jonathan Ott. Markov decision processes with average-value-at-risk criteria. *Mathematical Methods of Operations Research*, 74(3):361–379, 2011.

Nicole Bäuerle and Ulrich Rieder. More risk-sensitive markov decision processes. *Mathematics of Operations Research*, 39(1):105–120, 2014.

Nicole Bäuerle, Marcin Pitera, and Łukasz Stettner. Blackwell optimality and policy stability for long-run risk-sensitive stochastic control. *SIAM Journal on Control and Optimization*, 62(6):3172–3194, 2024.

Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International conference on machine learning*, pages 449–458. PMLR, 2017.

Marc G Bellemare, Salvatore Candido, Pablo Samuel Castro, Jun Gong, Marlos C Machado, Subhodeep Moitra, Sameera S Ponda, and Ziyu Wang. Autonomous navigation of stratospheric balloons using reinforcement learning. *Nature*, 588(7836): 77–82, 2020.

Marc G Bellemare, Will Dabney, and Mark Rowland. *Distributional reinforcement learning*. MIT Press, 2023.

Richard Bellman, Irving Glicksberg, and Oliver Gross. On the optimal inventory equation. *Management Science*, 2(1):83–104, 1955.

Richard Ernest Bellman. *Dynamic Programming*. Princeton University Press, 1957.

Aharon Ben-Tal and Marc Teboulle. An old-new concept of convex risk measures: The optimized certainty equivalent. *Mathematical Finance*, 17(3):449–476, 2007.

Jeremy Berkowitz and James O'Brien. How accurate are value-at-risk models at commercial banks? *The journal of finance*, 57(3):1093–1111, 2002.

Dimitri Bertsekas. *Dynamic programming and optimal control: Volume I*, volume 4. Athena scientific, 2012.

Vivek S Borkar. Q-learning for risk-sensitive control. *Mathematics of operations research*, 27(2):294–311, 2002.

Mokrane Bouakiz and Youcef Kebir. Target-level criterion in markov decision processes. *Journal of Optimization Theory and Applications*, 86(1):1–15, 1995.

Stéphane Boucheron, Gábor Lugosi, and Olivier Bousquet. Concentration inequalities. In *Summer school on machine learning*, pages 208–240. Springer, 2003.

Michael Bowling, John D Martin, David Abel, and Will Dabney. Settling the reward hypothesis. In *International Conference on Machine Learning*, pages 3003–3020. PMLR, 2023.

Stephen P Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

Augustin Louis Baron Cauchy. *Exercices de mathématiques*, volume 3. De Bure Frères, 1828.

Yunho Choi, Kyungjae Lee, and Songhwai Oh. Distributional deep reinforcement learning with a mixture of gaussians. In *2019 international conference on robotics and automation (ICRA)*, pages 9791–9797. IEEE, 2019.

Yinlam Chow, Aviv Tamar, Shie Mannor, and Marco Pavone. Risk-sensitive and robust decision-making: a cvar optimization approach. *Advances in neural information processing systems*, 28, 2015.

Kun-Jen Chung and Matthew J Sobel. Discounted mdp's: Distribution functions and exponential utility maximization. *SIAM journal on control and optimization*, 25(1): 49–62, 1987.

George E Collins. Polynomial minimum root separation. *Journal of Symbolic Computation*, 32(5):467–473, 2001.

Gregory F Cooper. The computational complexity of probabilistic inference using bayesian belief networks. *Artificial intelligence*, 42(2-3):393–405, 1990.

Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. In *International conference on machine learning*, pages 1096–1105. PMLR, 2018a.

Will Dabney, Mark Rowland, Marc Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018b.

Giuseppe DeCandia, Deniz Hastorun, Madan Jampani, Gunavardhan Kakulapati, Avinash Lakshman, Alex Pilchin, Swaminathan Sivasubramanian, Peter Vosshall, and Werner Vogels. Dynamo: Amazon's highly available key-value store. *ACM SIGOPS operating systems review*, 41(6):205–220, 2007.

Rick Durrett. *Probability: theory and examples*, volume 49. Cambridge university press, 2019.

Martin Engert. Finite dimensional translation invariant subspaces. *Pacific Journal of Mathematics*, 32(2):333–343, 1970.

Jesse Farebrother, Jordi Orbay, Quan Vuong, Adrien Ali Taïga, Yevgen Chebotar, Ted Xiao, Alex Irpan, Sergey Levine, Pablo Samuel Castro, Aleksandra Faust, et al. Stop regressing: Training value functions via classification for scalable deep rl. *arXiv preprint arXiv:2403.03950*, 2024.

Alhussein Fawzi, Matej Balog, Aja Huang, Thomas Hubert, Bernardino Romera-Paredes, Mohammadamin Barekatain, Alexander Novikov, Francisco J R. Ruiz, Julian Schrittwieser, Grzegorz Swirszcz, et al. Discovering faster matrix multiplication algorithms with reinforcement learning. *Nature*, 610(7930):47–53, 2022.

Carlo Filippi, Gianfranco Guastaroba, and Maria Grazia Speranza. Conditional value-at-risk beyond finance: a survey. *International Transactions in Operational Research*, 27(3):1277–1319, 2020.

Hans Föllmer and Alexander Schied. *Stochastic finance: an introduction in discrete time*. Walter de Gruyter, 2011.

Javier Garcıa and Fernando Fernández. A comprehensive survey on safe reinforcement learning. *Journal of Machine Learning Research*, 16(1):1437–1480, 2015.

Jean-Michel Grandmont. Continuity properties of a von neumann-morgenstern utility. *Journal of economic theory*, 4(1):45–57, 1972.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models. *arXiv preprint arXiv:2301.04104*, 2023.

Jia Lin Hau, Erick Delage, Mohammad Ghavamzadeh, and Marek Petrik. On dynamic programming decompositions of static risk measures in markov decision processes. *Advances in Neural Information Processing Systems*, 36:51734–51757, 2023a.

Jia Lin Hau, Marek Petrik, and Mohammad Ghavamzadeh. Entropic risk optimization in discounted mdps. In *International Conference on Artificial Intelligence and Statistics*, pages 47–76. PMLR, 2023b.

Jia Lin Hau, Erick Delage, Esther Derman, Mohammad Ghavamzadeh, and Marek Petrik. Q-learning for quantile mdps: A decomposition, performance, and convergence analysis. *arXiv preprint arXiv:2410.24128*, 2024.

Matteo Hessel, Joseph Modayil, Hado Van Hasselt, Tom Schaul, Georg Ostrovski, Will Dabney, Dan Horgan, Bilal Piot, Mohammad Azar, and David Silver. Rainbow: Combining improvements in deep reinforcement learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Chris C Heyde. On a property of the lognormal distribution. *Journal of the Royal Statistical Society Series B: Statistical Methodology*, 25(2):392–393, 1963.

Ronald A Howard and James E Matheson. Risk-sensitive markov decision processes. *Management science*, 18(7):356–369, 1972.

Alexander Kobel, Fabrice Rouillier, and Michael Sagraloff. Computing real roots of real polynomials... and now for real! In *Proceedings of the 2016 ACM International Symposium on Symbolic and Algebraic Computation*, pages 303–310, 2016.

David M Kreps. Decision problems with expected utility criteria, ii: Stationarity. *Mathematics of Operations Research*, 2(3):266–274, 1977.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Xiaocheng Li, Huaiyang Zhong, and Margaret L Brandeau. Quantile markov decision processes. *Operations research*, 70(3):1428–1447, 2022.

Hao Liang and Zhi-Quan Luo. Bridging distributional and risk-sensitive reinforcement learning with provable regret bounds. *Journal of Machine Learning Research*, 25 (221):1–56, 2024.

Shiau Hong Lim and Ilyas Malik. Distributional reinforcement learning for risk-sensitive policies. *Advances in Neural Information Processing Systems*, 35:30977–30989, 2022.

Clare Lyle, Marc G Bellemare, and Pablo Samuel Castro. A comparative analysis of expected and distributional reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 4504–4511, 2019.

Odalric-Ambrym Maillard. Robust risk-averse stochastic multi-armed bandits. In *Algorithmic Learning Theory: 24th International Conference, ALT 2013, Singapore, October 6-9, 2013. Proceedings 24*, pages 218–233. Springer, 2013.

Alexandre Marthe, Aurélien Garivier, and Claire Vernade. Beyond average return in markov decision processes. *Advances in Neural Information Processing Systems*, 36: 56488–56507, 2023.

Alexandre Marthe, Samuel Bounan, Aurélien Garivier, and Claire Vernade. Efficient risk-sensitive planning via entropic risk measures. *arXiv preprint arXiv:2502.20423*, 2025.

Alexandre Marthe, Mehrasa Ahmadipour, Aurélien Garivier, and Claire Vernade. Statistical complexity of the entropic risk measure. Work in progress, 2026.

Pascal Massart. The tight constant in the dvoretzky-kiefer-wolfowitz inequality. *The annals of Probability*, pages 1269–1283, 1990.

Pascal Massart. *Concentration inequalities and model selection: Ecole d'Eté de Probabilités de Saint-Flour XXXIII-2003*. Springer, 2007.

Borislav Mavrin, Hengshuai Yao, Linglong Kong, Kaiwen Wu, and Yaoliang Yu. Distributional reinforcement learning for efficient exploration. In *International conference on machine learning*, pages 4424–4434. PMLR, 2019.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.

Tetsuro Morimura, Masashi Sugiyama, Hisashi Kashima, Hirotaka Hachiya, and Toshiyuki Tanaka. Nonparametric return distribution approximation for reinforcement learning. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 799–806, 2010.

Tetsuro Morimura, Masashi Sugiyama, Hisashi Kashima, Hirotaka Hachiya, and Toshiyuki Tanaka. Parametric return density estimation for reinforcement learning. *arXiv preprint arXiv:1203.3497*, 2012.

Oliver Mortensen and Mohammad Sadegh Talebi. Entropic risk optimization in discounted mdps: Sample complexity bounds with a generative model. *arXiv preprint arXiv:2506.00286*, 2025.

Andrew Y Ng, Daishi Harada, and Stuart J Russell. Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pages 278–287, 1999.

Thanh Nguyen-Tang, Sunil Gupta, and Svetha Venkatesh. Distributional reinforcement learning via moment matching. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pages 9144–9152, 2021.

Xinyi Ni and Lifeng Lai. Evar optimization for risk-sensitive reinforcement learning. *IEEE Transactions on Information Theory*, 2022.

Joe Nocera. Risk mismanagement. *The New York Times*, 2(Jan), 2009.

Julien Perolat, Bart De Vylder, Daniel Hennes, Eugene Tarassov, Florian Strub, Vincent de Boer, Paul Muller, Jerome T Connor, Neil Burch, Thomas Anthony, et al. Mastering the game of stratego with model-free multiagent reinforcement learning. *Science*, 378 (6623):990–996, 2022.

Georg Ch Pflug and Alois Pichler. Time-consistent decisions and temporal decomposition of coherent risk functionals. *Mathematics of Operations Research*, 41(2):682–699, 2016.

Bernardo Ávila Pires, Mark Rowland, Diana Borsa, Zhaohan Daniel Guo, Khimya Khetarpal, André Barreto, David Abel, Rémi Munos, and Will Dabney. Optimizing return distributions with distributional dynamic programming. *arXiv preprint arXiv:2501.13028*, 2025.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming.* John Wiley & Sons, 2014.

R Tyrrell Rockafellar, Stanislav Uryasev, et al. Optimization of conditional value-at-risk, 2000.

Mark Rowland, Marc Bellemare, Will Dabney, Rémi Munos, and Yee Whye Teh. An analysis of categorical distributional reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 29–37. PMLR, 2018.

Mark Rowland, Robert Dadashi, Saurabh Kumar, Rémi Munos, Marc G Bellemare, and Will Dabney. Statistics and samples in distributional reinforcement learning. In *International Conference on Machine Learning*, pages 5528–5536. PMLR, 2019.

Walter Rudin. *Real and complex analysis*. McGraw-Hill, Inc., 1987.

Herbert Scarf, K Arrow, S Karlin, and P Suppes. The optimality of (s, s) policies in the dynamic inventory problem. *Optimal pricing, inflation, and the cost of price adjustment*, pages 49–56, 1960.

Max Schwarzer, Johan Samir Obando Ceron, Aaron Courville, Marc G Bellemare, Rishabh Agarwal, and Pablo Samuel Castro. Bigger, better, faster: Human-level atari with human-level efficiency. In *International Conference on Machine Learning*, pages 30365–30380. PMLR, 2023.

David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. A general reinforcement learning algorithm that masters chess, shogi, and go through self-play. *Science*, 362(6419):1140–1144, 2018.

David Silver, Satinder Singh, Doina Precup, and Richard S Sutton. Reward is enough. *Artificial intelligence*, 299:103535, 2021.

Matthew J Sobel. The variance of discounted markov decision processes. *Journal of Applied Probability*, 19(4):794–802, 1982.

Xihong Su, Marek Petrik, and Julien Grand-Clément. Evar optimization in mdps with total reward criterion. In *Seventeenth European Workshop on Reinforcement Learning*, 2024.

Xihong Su, Marek Petrik, and Julien Grand-Clément. Risk-averse total-reward mdps with erm and evar. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 20646–20654, 2025.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Yunhao Tang and Shipra Agrawal. Exploration by distributional reinforcement learning. *arXiv preprint arXiv:1805.01907*, 2018.

Timo Tossavainen. The lost cousin of the fundamental theorem of algebra. *Mathematics Magazine*, 80(4):290–294, 2007.

Amos Tversky and Daniel Kahneman. Advances in prospect theory: Cumulative representation of uncertainty. *Journal of Risk and uncertainty*, 5(4):297–323, 1992.

Roman Vershynin. *High-dimensional probability: An introduction with applications in data science*, volume 47. Cambridge university press, 2018.

C. Villani. *Topics in Optimal Transportation*. Graduate studies in mathematics. American Mathematical Society, 2003. ISBN 9780821833124.

John Von Neumann and Oskar Morgenstern. Theory of games and economic behavior: 60th anniversary commemorative edition. In *Theory of games and economic behavior*. Princeton university press, 1944.

Kaiwen Wang, Kevin Zhou, Runzhe Wu, Nathan Kallus, and Wen Sun. The benefits of being distributional: Small-loss bounds for reinforcement learning. *Advances in neural information processing systems*, 36:2275–2312, 2023.

Shengjie Wang, Shaohuai Liu, Weirui Ye, Jiacheng You, and Yang Gao. Efficientzero v2: Mastering discrete and continuous control with limited data. *arXiv preprint arXiv:2403.00564*, 2024.

Larry Wasserman. *All of statistics: a concise course in statistical inference*. Springer Science & Business Media, 2013.

Ying Wei, Rebecca D Kehm, Mandy Goldberg, and Mary Beth Terry. Applications for quantile regression in epidemiology. *Current Epidemiology Reports*, 6(2):191–199, 2019.

Douglas J White. Minimizing a threshold probability in discounted markov decision processes. *Journal of mathematical analysis and applications*, 173(2):634–646, 1993.

George Wu and Richard Gonzalez. Curvature of the probability weighting function. *Management science*, 42(12):1676–1690, 1996.

Menahem E Yaari. The dual theory of choice under risk. *Econometrica: Journal of the Econometric Society*, pages 95–115, 1987.

Pengqian Yu, William B Haskell, and Huan Xu. Dynamic programming for risk-aware sequential optimization. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 4934–4939. IEEE, 2017.