Distributionally Robust Optimization for Hybrid Systems

Verteilungsrobuste Optimierung für hybride Systeme
Master thesis by Yannick Eich
Date of submission: September 22, 2021

1. Review: Prof. Heinz Köppl
2. Review: M.Sc. Hany Abdulsamad
3. Review: M.Sc. Joe Watson
4. Review: Prof. Jan Peters
Darmstadt

TECHNISCHE
UNIVERSITÄT
DARMSTADT

Erklärung zur Abschlussarbeit gemäSS §22 Abs. 7 und §23 Abs. 7 APB der TU Darmstadt

Hiermit versichere ich, Yannick Eich, die vorliegende Masterarbeit ohne Hilfe Dritter und nur mit den angegebenen Quellen und Hilfsmitteln angefertigt zu haben. Alle Stellen, die Quellen entnommen wurden, sind als solche kenntlich gemacht worden. Diese Arbeit hat in gleicher oder ähnlicher Form noch keiner Prüfungsbehörde vorgelegen.

Mir ist bekannt, dass im Fall eines Plagiats (§38 Abs. 2 APB) ein Täuschungsversuch vorliegt, der dazu führt, dass die Arbeit mit 5,0 bewertet und damit ein Prüfungsversuch verbraucht wird. Abschlussarbeiten dürfen nur einmal wiederholt werden.

Bei der abgegebenen Thesis stimmen die schriftliche und die zur Archivierung eingereichte elektronische Fassung gemäSS §23 Abs. 7 APB überein.

Bei einer Thesis des Fachbereichs Architektur entspricht die eingereichte elektronische Fassung dem vorgestellten Modell und den vorgelegten Plänen.

Darmstadt, 22. September 2021

Y. Eich

Abstract

In this thesis, we extend the Guided Policy Search framework to the class of Markov Jump Linear Systems. As common in stochastic hybrid systems, the forward pass leads to an exponentially growing number of assumptions about the state distribution. We present two methods of reducing the number of assumptions in a robust manner. Therefore, we adjust the state assumptions to the worst-case distribution that is close to the original distribution. This approach is similar to changing the Markov chain, that describes the discrete state in the Markov Jump Linear Systems. We use this similarity and provide a procedure to find a policy that is robust with respect to changes in the Markov chain. Finally, we compare the robust policy with the policy that is optimal on the nominal Markov chain.

Acknowledgments

First of all, I would like to thank Hany Abdulsamad and Joe Watson for supervising this thesis. They invested a lot of time in me, guiding me with their ideas and discussing methods in detail.

I want to thank Prof. Jan Peters for giving me the option to study at IAS. His fascinating online course gave me insights into the basics of robot learning, while the Oberseminar presented many interesting current research topics.

Lastly, I want to thank Prof. Heinz Köppl for agreeing to co-supervise my thesis and offering his support.

Contents

List of Algorithms

List of Figures

Acronyms

Notation	Description
DP	Dynamic Programming
DRO	Distributionally Robust Optimization
GPB	Generalized Pseudo-Bayesian
GPS	Guided Policy Search
IMM	Interacting Multiple Model
KL	Kullback-Leibler divergence
MJLS	Markov Jump Linear Systems
PWA	Piecewise Affine Systems

1. Introduction

The class of nonlinear dynamical systems remains a major challenge in classical control theory [1] and reinforcement learning [2]. Classical control theory provides a lot of methods such as feedback linearization [3] or backstepping [4], but those work only on limited classes of nonlinear systems. In contrast to linear control theory, there is no notion of a general solution. In reinforcement learning those issues are generally tackled by learning large nonlinear policies iteratively [5].

A different approach tries to model nonlinear systems with hybrid systems. The idea is to use hybrid switching models to decompose nonlinear systems into simpler segments. This strategy has been studied in control theory [6] and machine learning [7].

Ultimately, our goal is to have a complete framework to optimize nonlinear systems using the concepts of hybrid systems. Therefore, we want to learn hybrid switching models from data that describe the dynamics of a nonlinear system reasonably well. We hope to find policies to those hybrid models that outperform traditional data-driven approaches.

This thesis focuses on the control of stochastic hybrid systems. As solutions to optimal control so far exist only for limited classes of stochastic hybrid systems [8], we focus on the simple class of Markov Jump Linear Systems (MJLS). Even if the expressive power of these models is limited, they have difficulties that are common in hybrid systems. The way we tackle those problems can provide insight on how to handle more sophisticated hybrid models.

A major challenge with stochastic hybrid systems is the exponentially growing number of assumptions about the state, when simulating over multiple time steps. Typically, approximations are needed to keep the number of assumptions within bounds. As information is lost by using these approximations, the goal of this thesis is to find a robust way to tackle the exponential growth of the assumptions.

In the following we give a short introduction on hybrid systems, MJLS, and the optimal control of MJLS. We then show how the challenges of MJLS are dealt with in the research area of tracking. Subsequently, we introduce the concepts of distributionally robust optimization. Chapter 2 provides a different procedure to find the optimal control for MJLS by introducing an optimization problem that is solved iteratively. This new formulation of finding the optimal control forms the basis for the methods in the next chapters. In chapter 3 we build on the ideas used in tracking to find a robust manner to deal with the exponential growth of MJLS. In chapter 4 we extend the optimization framework from chapter 2 to a minimax optimization to find a distributionally robust policy. Finally, in chapter 5 we draw a conclusion.

1.1. Hybrid Systems and Optimal Control

Hybrid systems describe the dynamics and the interaction of continuous and discrete variables in a common framework [9]. Most interesting to us are the classes of switching systems, where there exist a set of dynam-

ical models and a discrete variable called mode that defines which model is used to describe the continuous state. A simple example are Piecewise Affine Systems (PWA) [10]. PWA consist of a set of affine systems that partition the state and input space in polyhedral regions. The mode that determines the affine system is therefore described by the continuous state and the action. Due to the occurrence of continuous and discrete variables, optimal control problems are solved by mixed-integer techniques [11][12].

Adding uncertainty to hybrid systems makes the problem of finding an optimal control much more difficult. As [8] shows, there are only a very limited number of subclasses of stochastic hybrid systems, where solutions to optimal control have been found. One of those is the class of Markov Jump Linear Systems (MJLS) [13]. MJLS are switching systems where the dynamics of the discrete mode are described by a finite state Markov chain $P_t(\mathbf{z}'|\mathbf{z})$. For each mode there consists a linear Gaussian model, that describes the evolution of the continuous state $\mathbf{s}$ given an action $\mathbf{a}$:

$$\mathbf{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) = \mathcal{N}(\mathbf{A}_t^z\mathbf{s} + \mathbf{b}_t^z\mathbf{a} + \mathbf{c}_t^z|\mathbf{\Sigma}_{dyn}^z)$$

It can be shown [14] that with a quadratic cost the stochastic optimal control can be computed in closed-form by applying Dynamic Programming (DP) [15]. In the following we refer to the assumptions of model and cost as LQG assumptions. Instead of a cost, we use the reward $R_t(\mathbf{s},\mathbf{a})$. Using the principle of optimality, DP decomposes the complex problem of finding the policy that maximizes the expected reward into simpler subproblems that can be solved recursively. This recursion can be described by the dependence of the state value function $V_t(\mathbf{s},\mathbf{z})$ and the state-action value function $Q_t(\mathbf{s},\mathbf{z},\mathbf{a})$. $V_t(\mathbf{s},\mathbf{z})$ is defined as the expected reward-to-go given the current state following policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$, whereas $Q_t(\mathbf{s},\mathbf{z},\mathbf{a})$ is the expected reward-to-go given the current state, choosing action $\mathbf{a}$ before following policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$. The relation between $V_t(\mathbf{s},\mathbf{z})$ and $Q_t(\mathbf{s},\mathbf{z},\mathbf{a})$ is expressed by the Bellman equations:

$$Q_t(\mathbf{s},\mathbf{z},\mathbf{a}) = R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} P_t(\mathbf{z}'|\mathbf{z})\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})V_{t+1}(\mathbf{s}',\mathbf{z}')d\mathbf{s}',$$

$$V_t(\mathbf{s},\mathbf{z}) = \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})Q_t(\mathbf{s},\mathbf{z},\mathbf{a})d\mathbf{a},$$

with $V_T(\mathbf{s},\mathbf{z}) = R_T(\mathbf{s})$. Using this description, finding the optimal policy breaks down to maximizing the state-action value function each time step:

$$\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) = \underset{\mathbf{a}}{\text{argmax}}\ Q_t(\mathbf{s},\mathbf{z},\mathbf{a}).$$

Using the aforementioned assumptions, the quadratic reward leads to quadratic value functions and Gaussian policies.

1.2. Multiple Model Approach in Tracking

In the research area of tracking the multiple model approach [16] is used, when dealing with switching systems consisting of r models. It provides a Bayesian framework to calculate the probabilities of a state at time k, given measurements up to k. The optimal solution consists of a Gaussian mixture with an exponentially increasing number of terms, as the solution has to be conditioned on each possible mode sequence.

To tackle the exponential increasing number of possible sequences, suboptimal algorithms are necessary. The simplest approach is called pruning, where after each time step only the most probable N sequences are kept.

A more sophisticated approach is the Generalized Pseudo-Bayesian (GPB) method. The idea is to combine similar sequences. Therefore, sequences that differ several time steps ago are merged, as their outcome will generally be more similar than sequences that differ in recent time steps [17]. The first-order GPB regards only the discrete state in the last time step. At the end of each time step r hypotheses are merged into a single hypothesis. GPB2 considers the history of the last two time steps. All sequences that differ in older modes are combined. Therefore, each time step r^2 hypotheses are merged into r hypotheses. The increasing order of the GBP approach comes with better approximations of the optimal approach, but also with an increasing computational cost.

The Interacting Multiple Model (IMM) filter approximates GPB2, but requires only to keep track of r hypothesis each time step [18]. The idea is to use a different timing for merging the hypotheses. While in GPB methods, the merging is done after the measurement update, leading to an increasing number of hypotheses, the IMM mixes the hypotheses at the beginning of each cycle, so that the number of assumption does not increase.

As the IMM performs much better than GPB1 and almost as well as GPB2 [18], it seems to be the best tradeoff between computational cost and accuracy.

The IMM and GPB filter use moment matching to collapse mixture of Gaussians into a single Gaussian. We are interested in methods that reduce the number of mixture terms to a given number. A typical strategy is to repeatedly merge two components until the given number is reached. Well-known methods differ in the way they choose the components to merge. The idea in Runnalls' algorithm [19] is to choose the two components, so that the Kullback-Leibler divergence (KL) [20] between the new mixture and the original distribution is minimal. As there is no closed-form solution for the KL between mixtures of Gaussians, the paper proposes an upper bound that can be used as a criterion. In [21], this method is compared with more sophisticated approaches. The evaluations show that Runnalls' algorithm is reasonably good and has much lower computational cost than the compared methods.

1.3. Distributionally Robust Optimization

Distributionally Robust Optimization (DRO) is a framework that combines the idea of robust optimization and stochastic optimization [22]. Consider the problem of minimizing a loss function $\mathcal{L}$ with respect to a decision variable $\mathbf{x}$, where there is uncertainty about the parameters $\mathbf{w}$. The idea of robust optimization is to think of the worst-case values the parameters can take and then minimize the loss function:

$$\min_{\mathbf{x}} \max_{\mathbf{w}} \mathcal{L}(\mathbf{x}, \mathbf{w}).$$

By optimizing the worst-case scenario, it is guaranteed that with the actual unknown parameters $\mathbf{w}$ the loss cannot be higher when choosing the found decision variable. However, this risk-averse decision is often too conservative, which means that it leads to poor values in the other scenarios [23]. Additionally, this approach does not use all the information available about the parameters, but only the information about the worst-case.

In contrast, stochastic optimization uses all the information available about the parameters. Assume the uncertainty of the parameters can be described by a distribution $p(\mathbf{w})$. Then the goal of stochastic optimization is to minimize the expected loss under that distribution:

$$\min_{\mathbf{x}} \mathbb{E}_{\mathbf{w}\sim p} \left[\mathcal{L}(\mathbf{x}, \mathbf{w})\right].$$

This risk-neutral decision often leads to bad outcomes for extreme cases of $\mathbf{w}$.

DRO proposes an optimization problem that combines these two approaches. The idea is to minimize the loss function as in stochastic optimization, but with the worst-case distribution from a set of distributions $\mathcal{P}$:

$$\min_{\mathbf{x}} \max_{p \in \mathcal{P}} \mathbb{E}_{\mathbf{w} \sim p} \left[\mathcal{L}(\mathbf{x}, \mathbf{w}) \right].$$

The set of distributions $\mathcal{P}$ is called ambiguity set. With the ambiguity set, the uncertainty about the distributions can be described. When the distribution of the parameter is learned from data, similar distributions can be included in the ambiguity set to make the optimization robust to errors or biases in the data. With this idea knowledge about the parameters can be included, even if this knowledge is not perfect.

2. Guided Policy Search for Markov Jump Linear Systems

In contrast to the exact solution proposed in [14], we solve the optimal control for MJLS iteratively. Typically, iterative methods in stochastic optimal control are used for finding optimal control for nonlinear dynamical systems. Methods like Differential Dynamic Programming [24] and Iterative Linear Quadratic Gaussian [25] repeatedly linearize the dynamics around each point of the state trajectory, solve linear optimal control problems and apply the new locally optimal policies to get a new trajectory.

Our work builds on Guided Policy Search (GPS) [26] that follows this iterative scheme. Instead of linearizing the points on the trajectory, the linearized models are directly learned from data. Additionally, GPS introduces a KL constraint on the trajectory update. This constraint ensures, that the linearized models describing the system only locally are not exploited. Even when working with linear systems, this limitation in the step size will be useful, as it is crucial when trying to find a robust control. In [27] the optimization problem is rewritten and it is shown that the KL on the trajectory is equivalent to an expected KL on the policy over the state distribution. We extend this approach to switching systems by adding the discrete state $\mathbf{z}$ and the Markov chain $P_t(\mathbf{z}'|\mathbf{z})$.

2.1. Optimization Problem

To find the optimal control, we iteratively solve the following optimization problem where the dynamics are assumed to be known, starting with a policy $q_t(\mathbf{a}|\mathbf{s},\mathbf{z})$ and searching for the new policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$:

$$\underset{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{\operatorname{argmax}} \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s}, \tag{2.1}$$

$$\text{s.t} \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} = 1 \qquad \forall \mathbf{s}, \forall \mathbf{z}, \forall t < T, \tag{2.2}$$

$$\sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_{t-1}(\mathbf{s},\mathbf{z}) \pi_{t-1}(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_{t-1}(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) P_{t-1}(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} = \mu_t(\mathbf{s}',\mathbf{z}') \qquad \forall \mathbf{s}', \forall \mathbf{z}', \forall t > 1, \tag{2.3}$$

$$\mu_1(\mathbf{s},\mathbf{z}) = p_1(\mathbf{s},\mathbf{z}) \qquad \forall \mathbf{s}, \forall \mathbf{z}, \tag{2.4}$$

$$\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} d\mathbf{a} d\mathbf{s} \leq \epsilon. \tag{2.5}$$

The objective function (2.1) aims to find the policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$ that maximizes the cumulative reward of the state trajectory under the initial conditions (2.4) and the dynamics (2.3). The constraint (2.5) limits the policy update by enforcing the new policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$ to be close to the previous policy $q_t(\mathbf{a}|\mathbf{s},\mathbf{z})$.

The optimization problem is solved by using the method of Lagrangian multipliers. The primal problem can be formulated by adding a Lagrangian multiplier for each constraint:

$$
\begin{aligned}
L(\pi_t, \mu_t, V_t, \lambda_t, \alpha_t) = & \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s}, \mathbf{a}) \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}, \mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
& + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \lambda_t(\mathbf{s}, \mathbf{z}) \left(1 - \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a}\right) d\mathbf{s} \\
& + \sum_{\mathbf{z}} \int_{\mathbf{s}} V_1(\mathbf{s}, \mathbf{z}) \left(p_1(\mathbf{s}, \mathbf{z}) - \mu_1(\mathbf{s}, \mathbf{z})\right) d\mathbf{s} \\
& + \sum_{t=2}^{T} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}', \mathbf{z}') \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_{t-1}(\mathbf{s}, \mathbf{z}) \pi_{t-1}(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_{t-1}(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) P_{t-1}(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \\
& - \sum_{t=2}^{T} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}', \mathbf{z}') \mu_t(\mathbf{s}', \mathbf{z}') d\mathbf{s}' \\
& + \alpha \left(\epsilon - \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z})}{q_t(\mathbf{a}|\mathbf{s}, \mathbf{z})} d\mathbf{a} d\mathbf{s}\right).
\end{aligned}
$$

Solving

$$
\frac{\partial L}{\partial \pi_t} = 0, \qquad \frac{\partial L}{\partial \lambda_t} = 0,
$$

gives an expression for the optimal policy:

$$
\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \propto \exp\left(\frac{Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})}{\alpha}\right), \tag{2.6}
$$

where

$$
Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a}) = \alpha \log\left(q_t(\mathbf{a}|\mathbf{s}, \mathbf{z})\right) + R_t(\mathbf{s}, \mathbf{a}) + \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) P_t(\mathbf{z}'|\mathbf{z}) d\mathbf{s}'. \tag{2.7}
$$

The dual problem is obtained by plugging the policy back in the primal problem:

$$
\begin{aligned}
L(\mu_t, V_t, \alpha) = & \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}, \mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} - \sum_{t=1}^{T} \sum_{\mathbf{z}} \int_{\mathbf{s}} V_t(\mathbf{s}, \mathbf{z}) \mu_t(\mathbf{s}, \mathbf{z}) d\mathbf{s} \\
& + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \alpha \log \int_{\mathbf{a}} \exp\left(\frac{Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})}{\alpha}\right) d\mathbf{a} d\mathbf{s} + \alpha\epsilon.
\end{aligned}
$$

Using the duality of this optimization [28], the original problem can be solved by minimizing the dual function

with respect to a positive α. Therefore, the partial derivatives are computed:

$$\frac{\partial L}{\partial \mu_t} = \begin{cases} -V_T(\mathbf{s}, \mathbf{z}) + R_T(\mathbf{s}) & , t = T \\ -V_t(\mathbf{s}, \mathbf{z}) + \alpha \log \int_{\mathbf{a}} \exp\left(\frac{Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})}{\alpha}\right) d\mathbf{a} & , t < T \end{cases} \tag{2.8}$$

$$\frac{\partial L}{\partial V_t} = \begin{cases} -\mu_1(\mathbf{s}, \mathbf{z}) + p_1(\mathbf{s}, \mathbf{z}) & , t = 1 \\ -\mu_t(\mathbf{s}, \mathbf{z}) + \sum_{\hat{\mathbf{z}}} \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \pi_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} & , t > 1 \end{cases} \tag{2.9}$$

$$\frac{\partial L}{\partial \alpha} = \epsilon - \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z})}{q_t(\mathbf{a}|\mathbf{s}, \mathbf{z})} d\mathbf{a} d\mathbf{s}. \tag{2.10}$$

Setting the partial derivatives (2.8) and (2.9) to zero leads to a forward pass for the state $\mu_t(\mathbf{s}, \mathbf{z})$ and a backward pass for the Lagrangian multiplier $V_t(\mathbf{s}, \mathbf{z})$ as optimality conditions:

$$V_t(\mathbf{s}, \mathbf{z}) = \begin{cases} R_T(\mathbf{s}) & , t = T \\ \alpha \log \int_{\mathbf{a}} \exp\left(\frac{Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})}{\alpha}\right) d\mathbf{a} & , t < T \end{cases} \tag{2.11}$$

$$\mu_t(\mathbf{s}, \mathbf{z}) = \begin{cases} p_1(\mathbf{s}, \mathbf{z}) & , t = 1 \\ \sum_{\hat{\mathbf{z}}} \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \pi_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{z}}, \hat{\mathbf{a}}) P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} & , t > 1. \end{cases} \tag{2.12}$$

By inserting the optimality conditions the dual simplifies to:

$$L(\mu_t, V_t, \alpha) = \sum_{\mathbf{z}} \int_{\mathbf{s}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} + \alpha\epsilon \tag{2.13}$$

The Lagrangian variable α needs to be optimized with gradient methods using (2.10) until the dual (2.13) converges. The full derivation can be found in appendix A.1.

2.2. Interpretation

Equations (2.7) and (2.11) show great resemblance to the Bellman optimality equations. The $\log \int \exp$ (logSumExp) operator is a smooth approximation of the maximum function over the action $\mathbf{a}$. The α term on the inside and outside of the logSumExp operator is the smoothness parameter. As α goes to zero, the operator converges to the maximum function. Therefore, we interpret the Lagrangian multiplier $V_t(\mathbf{s}, \mathbf{z})$ as the optimal state value function. By following this interpretation $Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})$ is the state action value function, where the reward is augmented by $\alpha \log q_t(\mathbf{a}|\mathbf{s}, \mathbf{z})$, a term that punishes actions of low probability in the previous policy. This is a consequence of constraint (2.5) making the new policy stay close to the old one. The policy (2.6) is a softmax function with the state value function $Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})$ as input and α as a temperature parameter.

For a given α we solve an approximation of the Bellman optimality equations. This approximation limits the policy updates. When α goes to zero, the approximation converges to the equation for the optimal value function, as the logSumExp operator converges to the maximum function and the augmentation of the reward

vanishes. The whole approach is very similar to the simulated annealing technique where the optimization problem is approximated by an easier optimization problem. The solution to the approximation is repeatedly used as initial guess for the next iteration while the approximation converges to the original problem [29].

2.2.1. LQG Assumptions

Using the assumptions of Markov Jump Linear systems and quadratic reward leads to closed-form solutions for the backward pass and the forward pass. Starting with a Gaussian policy $q_t(\mathbf{a}|\mathbf{s}, \mathbf{z})$, the augmented reward will be quadratic as the logarithm of a Gaussian is a quadratic function. The quadratic reward yields a quadratic state value function $V_t(\mathbf{s}, \mathbf{z})$ and a quadratic state action value function $Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})$. As the policy $\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z})$ is the softmax of the quadratic state action value function, the new policy will again be Gaussian. A complete derivation of the closed-form solutions under LGQ assumptions can be found in the appendix A.2.

2.2.2. Drawbacks

Even though there are closed-form solutions, there is still a major issue. As can be seen in the forward pass (2.12), the number of Gaussians grows in each time step due to the sum over the discrete state $\mathbf{z}$. This exponential growth leads quickly to high computational cost. In this form, the algorithm is therefore only feasible for a short horizon.

Figure 2.1 highlights the effect of the growing number of mixture terms over time. It shows the trajectories of a MJLS with 2 discrete states and 2 continuous states with an applied optimal policy. The blue lines visualize the probability density functions of the continuous state each time step, while the red dots show samples of the trajectory. The number of mixture terms grow exponentially over time, leading to $2^7 = 128$ mixture terms in the final time step. Due to the optimal control, most terms are brought to the origin, making it seem like only one mixture term in the end.

2.3. Implementation

Applying the iterative style of GPS to find the optimal control for MJLS, we solve the presented optimization multiple times. After each iteration, the solution $\pi_t(\mathbf{s}, \mathbf{z})$ is used as initial policy $q(\mathbf{s}, \mathbf{z})$ for the next iteration. In each optimization problem we initialise α and start with the backward pass (2.11) to get the value function $V_t(\mathbf{s}, \mathbf{z})$ and the policy $\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z})$. Using this policy, the state distributions $\mu_t(\mathbf{s}, \mathbf{z})$ are computed with the forward pass (2.12). With the state distribution and the policy, we are able to calculate the gradient of α. With this gradient we do an update on α. Instead of using gradient descent, we use a bisection method. Algorithm 1 shows the pseudocode of this optimization.

When the MJLS is iteratively learned from data, an adequate step size needs to be chosen that makes sure that the local model is valid for the trajectory. When the MJLS model is globally valid, this is not the case. A big step size can be chosen, so that the optimal control is found fast. It would also be possible to directly use the exact solution proposed in [14] in that case. But as mentioned before, when extending this method for finding a robust policy, a limit on the policy update is crucial.

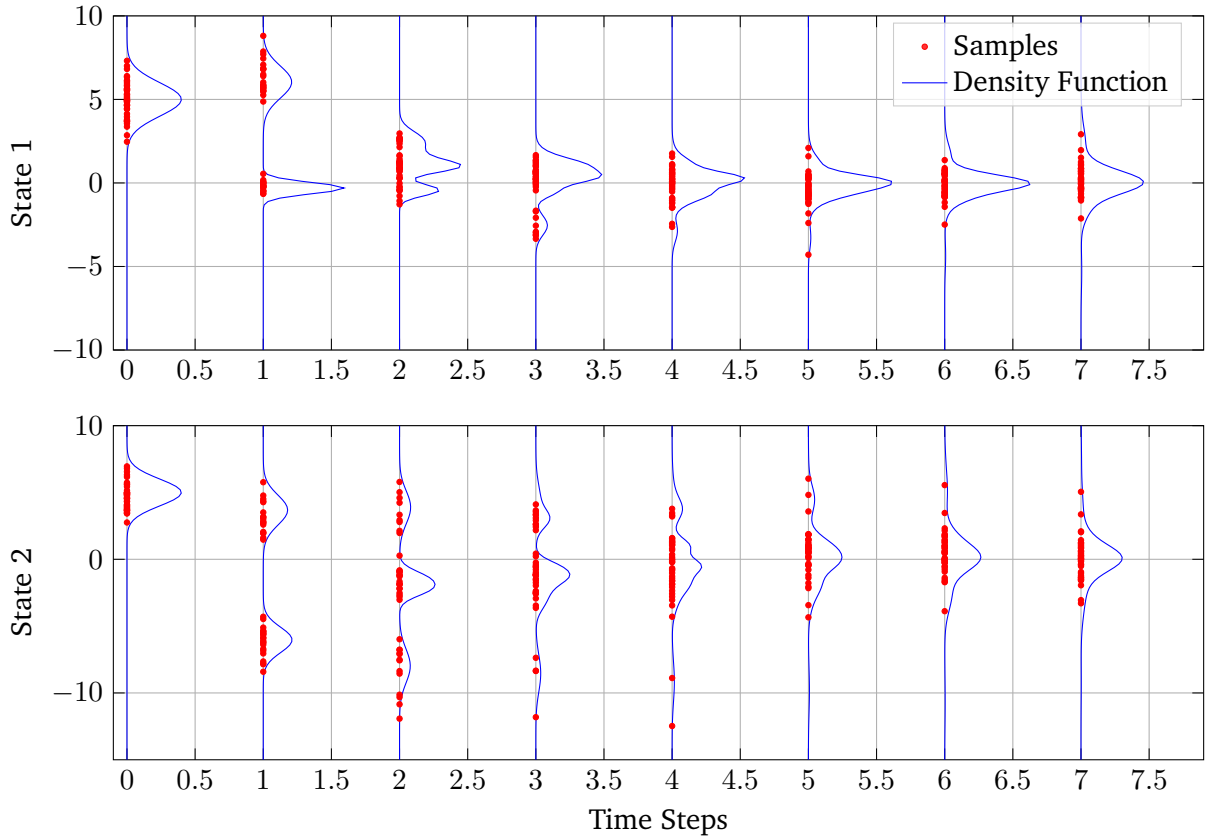

Figure 2.1.: Trajectory of a random MJLS with applied optimal control. The number of mixture components increases exponentially with time. Due to the optimal control the uncertainty decreases in the final time steps as the mixture terms are brought to the origin.

input : T ; /* time horizon */
$\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})$; /* linear dynamics */
$P_t(\mathbf{z}'|\mathbf{z})$; /* Markov chain */
$\mu_1(\mathbf{s},\mathbf{z})$; /* initial state distribution */
$q_t(\mathbf{a}|\mathbf{s},\mathbf{z})$; /* initial policy */
$R_t(\mathbf{s},\mathbf{z})$; /* reward function */

output : $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$; /* optimal policy */

initialize : $\alpha, \alpha_{\text{min}}, \alpha_{\text{max}}, L_{\text{opt}}$

while $L(\mu_t, V_t, \alpha)$ *not at minimum* **do**

/* compute value function and policy using equation (2.11) */
$[V_t(\mathbf{s},\mathbf{z}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})] \leftarrow$ backward_pass$(\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}), R_t(\mathbf{s},\mathbf{a}), \alpha)$;

/* compute the state distribution using equation (2.12) */
$\mu_t(\mathbf{s}) \leftarrow$ forward_pass$(\mu_1(\mathbf{s},\mathbf{z}), \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}))$;

/* update dual value using equation (2.13) */
$L(\mu_t, V_t, \alpha) \leftarrow$ update_dual$(\mu_1(\mathbf{s},\mathbf{z}), V_1(\mathbf{s},\mathbf{z}), \alpha, \epsilon)$;

/* compute dual gradient with respect to α using equation (2.10) */
$\frac{\partial L}{\partial \alpha} \leftarrow$ dual_alpha_gradient$(\mu_t(\mathbf{s}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}), \epsilon)$;

/* bisection method to find optimal α */
if $L(\mu_t, V_t, \alpha) < L_{opt}$ **then**
  $\alpha_{\text{opt}} \leftarrow \alpha$;
  $L_{\text{opt}} \leftarrow L(\mu_t, V_t, \alpha)$;
  if $\frac{\partial L}{\partial \alpha} < 0$ **then**
    /* α too small */
    $\alpha_{\text{min}} \leftarrow \alpha$;
    $\alpha \leftarrow \sqrt{\alpha_{\text{min}} \cdot \alpha_{\text{max}}}$;
  else
    /* α too large */
    $\alpha_{\text{max}} \leftarrow \alpha$;
    $\alpha \leftarrow \sqrt{\alpha_{\text{min}} \cdot \alpha_{\text{max}}}$;
else
  $\alpha_{\text{min}} \leftarrow \alpha$;
  $\alpha \leftarrow \sqrt{\alpha_{\text{min}} \cdot \alpha_{\text{max}}}$;

Algorithm 1: Pseudocode of Guided Policy Search applied to Markov Linear Jump Systems

3. Robustifying Assumptions About the State Distribution

As described in the last chapter, one major issue when dealing with hybrid systems is the exponentially growing number of hypotheses about the state in the forward pass. In the research are of tracking, there are several solutions to tackle this issue. The simplest approach is called pruning, where after each time step the hypotheses with the lowest probabilities are dropped. Other methods involve the idea of merging similar hypotheses to reduce the number of assumptions. No matter which method is used, there will be a loss of information when the number of hypotheses is decreasing. In tracking this loss of information is moderated by measurements, which add new information and adjust the assumptions. As no measurements are available, we would like to have a different principle to adjust the assumptions. Typically we are more interested in bad outcomes that lead to higher costs. We propose to use information about cost and dynamics to robustify the hypotheses before using pruning and merging techniques. Therefore, we adjust the state distributions to the worse in the sense of a cost function.

This chapter focuses on two different methods that make the assumptions about the state more robust. Two mechanisms are presented that change a mixture of Gaussians to a mixture with higher cost. We then analyze how those methods in combination with the merging and pruning filters used in tracking affect the forward pass and the routine to find the optimal control.

3.1. Robustifying Mixture of Gaussians

3.1.1. Optimization Problem

The goal of the first method is to find the distribution $q(\mathbf{s})$ in a KL-ball around the given distribution $\mu(\mathbf{s})$, that maximizes a cost function $C(\mathbf{s})$. This leads to solving the following optimization problem:

$$\begin{aligned} \max_{q(\mathbf{s})} & \int_{\mathbf{s}} C(\mathbf{s}) q(\mathbf{s}) d\mathbf{s}, \\ \text{s.t.} & \int_{\mathbf{s}} q(\mathbf{s}) d\mathbf{s} = 1, \\ & \int_{\mathbf{s}} q(\mathbf{s}) \log \frac{q(\mathbf{s})}{\mu(\mathbf{s})} d\mathbf{s} \leq \epsilon. \end{aligned} \tag{3.1}$$

The Lagrangian can be formulated by using the method of Lagrangian multipliers:

$$L(q, \alpha, \lambda) = \int_{\mathbf{s}} C(\mathbf{s}) q(\mathbf{s}) d\mathbf{s} - \alpha \left(\int_{\mathbf{s}} q(\mathbf{s}) \log \frac{q(\mathbf{s})}{\mu(\mathbf{s})} d\mathbf{s} - \epsilon \right) - \lambda \left(\int_{\mathbf{s}} q(\mathbf{s}) d\mathbf{s} - 1 \right).$$

Solving

$$\frac{\partial L}{\partial q} = 0, \qquad \frac{\partial L}{\partial \lambda} = 0,$$

leads to the following expression for the new state distribution:

$$q(\mathbf{s}) \propto \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s}) \right).$$

By inserting this back into the Lagrangian, the dual function is obtained:

$$L(\alpha) = \alpha \left(\epsilon + \log \int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s}) \right) d\mathbf{s} \right). \tag{3.2}$$

Using the principle of duality [28], the original problem is solved by minimizing the dual function with respect to a positive α. Therefore a gradient descent method is used with the partial derivative:

$$\frac{\partial L}{\partial \alpha} = \epsilon - \int_{\mathbf{s}} q(\mathbf{s}) \log \frac{q(\mathbf{s})}{\mu(\mathbf{s})} d\mathbf{s}. \tag{3.3}$$

It can be shown that for each ϵ there exists an α, that is the solution to the gradient descent. As we are interested in a closed-form solution to the optimization problem, we solve this problem with a fixed α instead of a fixed ϵ. With this change of the hyperparameter that we choose, we trade off the intuitive interpretability of ϵ that defines the KL for a faster solution, as we avoid gradient descent.

Using the assumptions of a quadratic cost function $C(s) = \mathbf{s}^T \mathbf{C} \mathbf{s} + \mathbf{s}^T \mathbf{c} + c$ and a mixture of Gaussians $\mu(\mathbf{s}) = \sum_i w_i \mathcal{N}(\mathbf{s} | \tau_i, \mathbf{\Sigma}_i)$ it can be shown (see Appendix B) that $q(\mathbf{s})$ will also be a mixture of Gaussians with the following form:

$$q(\mathbf{s}) = \sum_i q_i \mathcal{N}\left(\mathbf{s} | (\mathbf{\Sigma}_i^{-1} - \frac{2}{\alpha}\mathbf{C})^{-1} (\frac{\mathbf{c}}{\alpha} + \mathbf{\Sigma}_i^{-1} \tau_i), (\mathbf{\Sigma}_i^{-1} - \frac{2}{\alpha}\mathbf{C})^{-1} \right). \tag{3.4}$$

By analyzing the covariance term, it can be seen that the existence of the solution depends on α. A correct covariance matrix has to be positive semidefinite, therefore the following condition needs to be satisfied:

$$\mathbf{\Sigma}_i^{-1} - \frac{2}{\alpha}\mathbf{C} \succ 0. \tag{3.5}$$

For positive semidefinite $\mathbf{\Sigma}_i$ and $\mathbf{C}$ there will always be a small positive α that dissatisfies condition (3.5). This is a major drawback that comes from the change of hyperparameters. For every ϵ the corresponding α would satisfy the condition.

3.1.2. Existence of the Worst-Case Distribution

In the following we give a short explanation, why this issue arises, when changing the optimization problem from a fixed ϵ to a fixed α. Choosing a value for α is the same as changing the optimization problem (3.1) to:

$$\max_{q(\mathbf{s})} \int_{\mathbf{s}} C(\mathbf{s})q(\mathbf{s})d\mathbf{s} - \alpha \int_{\mathbf{s}} q(\mathbf{s}) \log \frac{q(\mathbf{s})}{\mu(\mathbf{s})} d\mathbf{s},$$
$$\text{s.t.} \int_{\mathbf{s}} q(\mathbf{s})d\mathbf{s} = 1.$$

Using a quadratic cost and Gaussian distributions $q(\mathbf{s}) = \mathcal{N}(\mathbf{s}|\mathbf{a}, \mathbf{A})$ and $\mu(\mathbf{s}) = \mathcal{N}(\mathbf{s}|\tau, \mathbf{\Sigma})$, the objective changes to:

$$\mathbf{a}^T\mathbf{C}\mathbf{a} + \mathbf{a}^T\mathbf{c} + c - \frac{\alpha}{2}\left(\log|\mathbf{\Sigma}| - \log|\mathbf{A}| + tr(\mathbf{\Sigma}^{-1}\mathbf{A}) + (\tau - \mathbf{a})^T\mathbf{\Sigma}^{-1}(\tau - \mathbf{a}) - n\right). \tag{3.6}$$

Notice that the quadratic cost function is convex, while the negative KL is concave. When maximizing this, the concave part needs to outweigh the convex part, so that the objective becomes concave. For a convex function, there would not be a solution, as the function grows to infinity. The objective becomes concave, when

$$\mathbf{C} - \frac{\alpha}{2}\mathbf{\Sigma}^{-1} \prec 0.$$

As a consequence, condition (3.5) makes sure that the objective of the simplified optimization problem with a fixed α becomes concave.

3.1.3. Example

As an example, we analyze how the distribution $q(\mathbf{s})$ changes for different α, given a one-dimensional mixture of Gaussians $\mu(\mathbf{s}) = 0.5\mathcal{N}(\mathbf{s}| - 5, 1) + 0.3\mathcal{N}(\mathbf{s}|0, 1) + 0.2\mathcal{N}(\mathbf{s}|5, 1)$ and the cost function $C(\mathbf{s}) = \mathbf{s}^2 + \mathbf{s} + 1$. To satisfy the condition (3.5), α needs to be higher than 2. Figure 3.1 compares the resulting distributions for different α values. The plots visualize that for high α the distribution $q(\mathbf{s})$ stays very close to the original distribution $\mu(\mathbf{s})$, since high values of α correspond to a small KL. For smaller values of α two effects can be seen: Each Gaussian increases its cost by an increasing covariance and a mean that moves further away from the minimum of the cost function. Secondly, with decreasing α the optimization gives more weight to the Gaussians with higher cost. When α converges to the limit set by condition (3.5), distribution $q(\mathbf{s})$ diverges leading to a cost that rises to infinity.

3.1.4. Optimistic Mixture of Gaussians

Interestingly, with a small change in the optimization formulation, the problem can be changed from a robust pessimistic search to an optimistic one. By turning the maximization into a minimization, we search for the distribution with the smallest cost in some KL-ball around the given distribution $\mu(\mathbf{s})$. This formulation leads to the same equations, but with negative α instead of positive ones. For negative α the condition (3.5) is always satisfied, when $\mathbf{\Sigma}_i$ and $\mathbf{C}$ are positive semidefinite. This is a consequence of the objective (3.6) being a convex function for all negative α.

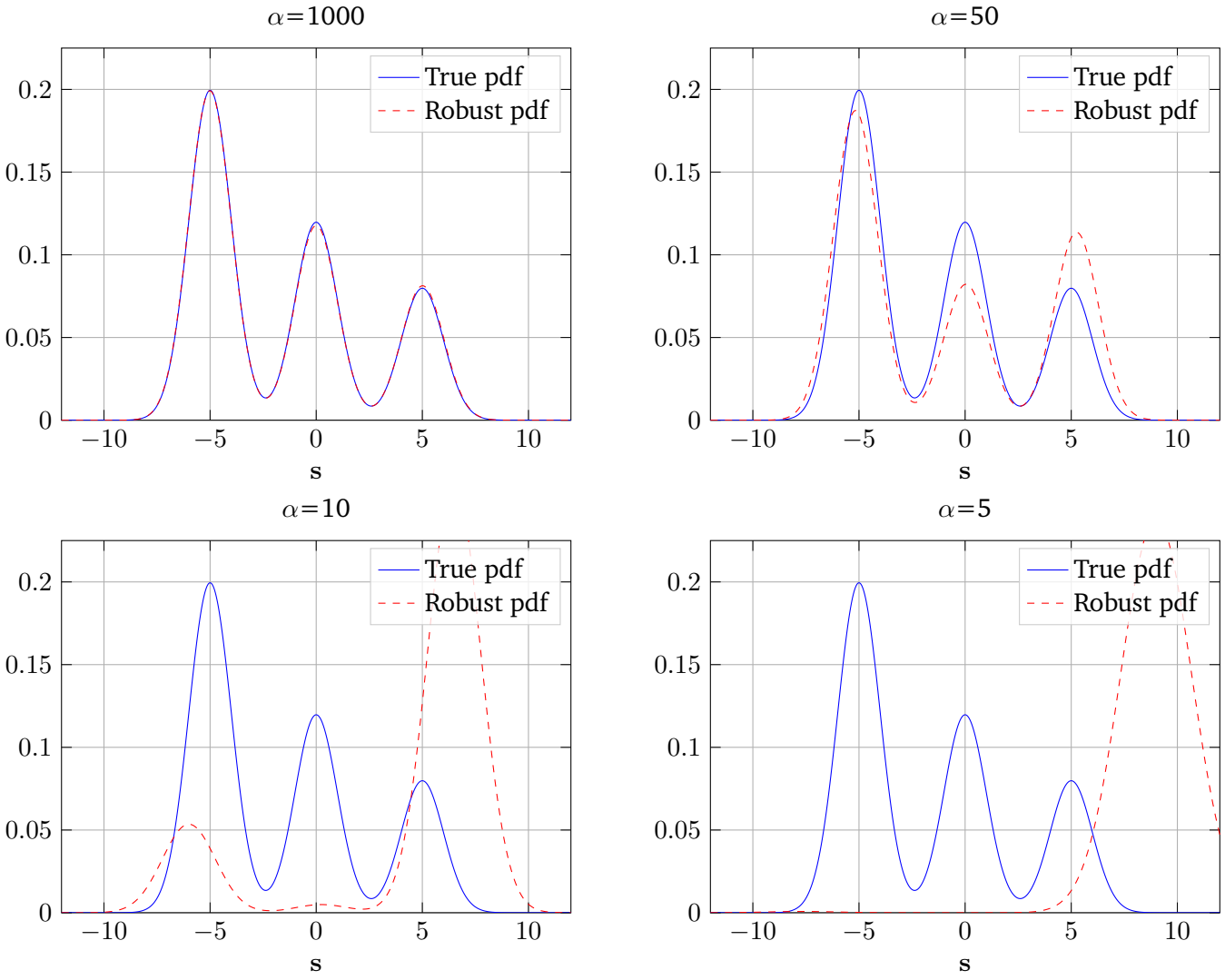

Figure 3.1.: Comparison of robust state distributions with different α values. For high α values the robust distribution is close to the original distribution as high α values correspond to a small KL. For decreasing α, the terms move to higher cost states and the weights of the higher cost terms increase.

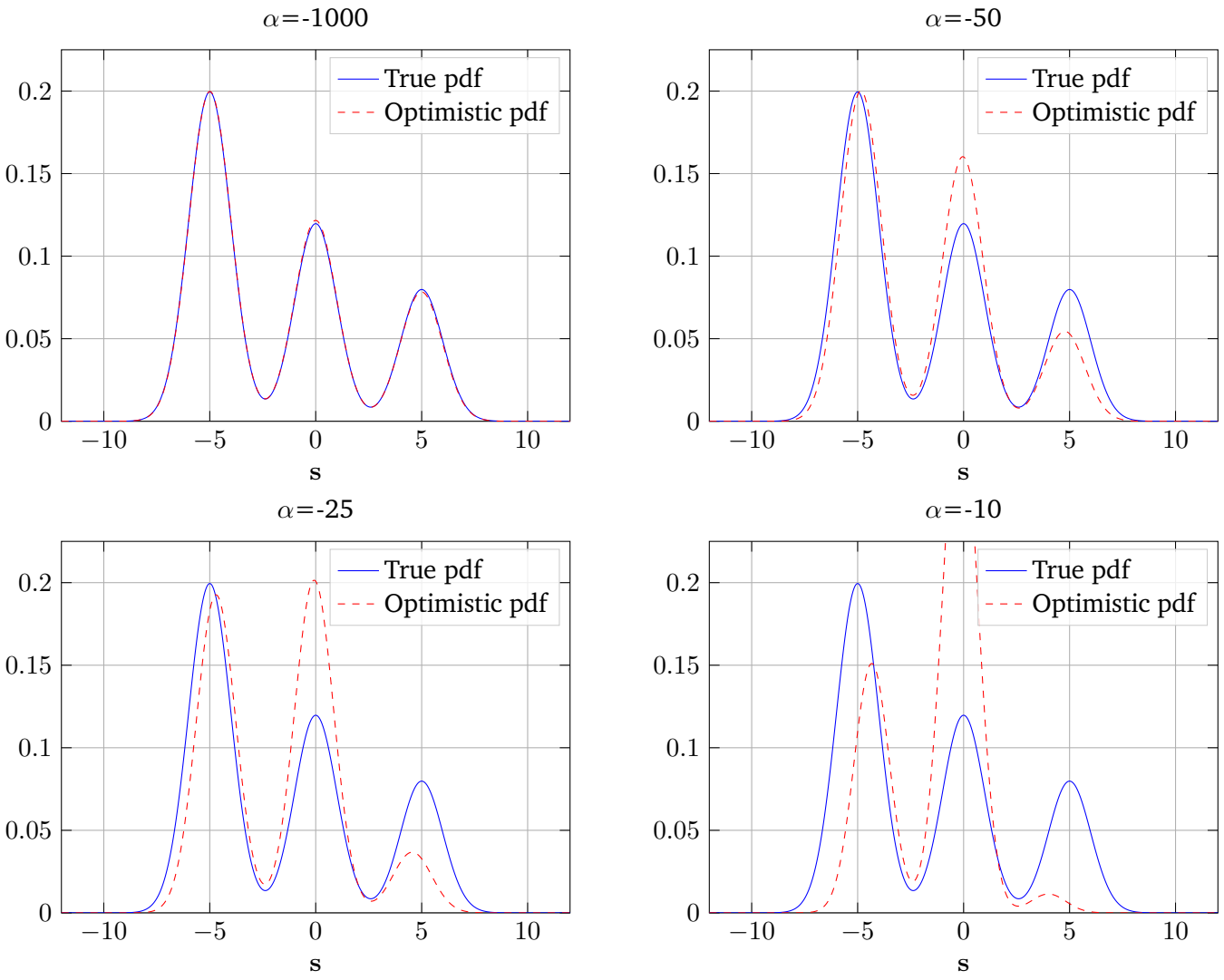

Figure 3.2.: Comparison of optimistic state distributions with different α values. For high absolute α values the optimistic distribution is close to the original distribution as high α values correspond to a small KL. For decreasing α, the terms move to the minimum of the cost function and the weights of the lower cost terms increase.

Figure 3.2 shows how the optimistic distribution changes with different α. It can again be seen that for high absolute α values the optimistic distribution is close to the given distribution as those α values correspond to small KL. When α goes to zero, the KL-ball grows bigger and therefore the distribution goes to the minimum of the cost function.

3.1.5. Conclusion

There are two issues using this method to find a robust distribution. First, condition (3.5) has to be satisfied. Secondly, while moving closer to the limit set by the condition, the cost of the distributions will rise to infinity. As this rate of going to infinity is different for each given distribution and cost function, it is hard to intuitively choose the hyperparameter α.

3.2. Robustifying Mixture Weights

3.2.1. Optimization Problem

To tackle the issues mentioned, we propose a different approach to make the assumptions robust. Instead of trying to find the distribution with the highest cost within a KL-ball, the mixture terms are kept and we search for a new distribution of their weights:

$$\begin{aligned} \max_{v_i} & \int_{\mathbf{s}} C(\mathbf{s}) \sum_i v_i \mu_i(\mathbf{s}) d\mathbf{s}, \\ \text{s.t.} & \sum_i v_i = 1, \\ & \sum_i v_i \log \frac{v_i}{w_i} \leq \epsilon. \end{aligned}$$

The optimization is very similar to the one mentioned before. We start again by formulating the Lagrangian:

$$L(v_i, \alpha, \lambda) = \int_{\mathbf{s}} C(\mathbf{s}) \sum_i v_i \mu_i(\mathbf{s}) d\mathbf{s} - \alpha \sum_i v_i \log\left(\frac{v_i}{w_i}\right) - \epsilon - \lambda \left(\sum_i v_i - 1\right).$$

Solving

$$\frac{\partial L}{\partial v_i} = 0, \qquad \frac{\partial L}{\partial \lambda} = 0,$$

gives the following expression for the new weights:

$$v_i \propto w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s}\right)\right]. \tag{3.7}$$

Plugging this into the Lagrangian yields the dual function:

$$L(\alpha) = \alpha \left(\epsilon + \log \sum_i w_i \exp \left(\frac{1}{\alpha} \int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s} \right) \right).$$

The original problem is solved by minimizing the dual problem with respect to a positive α [28]. Therefore a gradient descent method is used with the partial derivative:

$$\frac{\partial L}{\partial \alpha} = \epsilon - \sum_i v_i \log \frac{v_i}{w_i}.$$

The full derivation is shown in appendix C. As before, we avoid the gradient descent on α, by choosing α as the hyperparameter instead of the KL ϵ. In contrast to the previous method, the solution to (3.7) does always exist. For a quadratic cost and the mixture terms being Gaussian $\mu_i = \mathcal{N}(\mathbf{s}|\tau_i, \mathbf{\Sigma}_i)$ the integral resolves to :

$$\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s} = \int_{\mathbf{s}} (\mathbf{s^T C s} + \mathbf{c} + c) \mathcal{N}(\mathbf{s}|\tau_i, \mathbf{\Sigma}_i) = \tau_i^T \mathbf{C} \tau_i + \tau_i^T + c + \mathrm{Tr}(\mathbf{C\Sigma}_i).$$

Figure 3.3 shows how the mixture changes for different positive α values using the same original distribution and cost function. Like the previous method, for high α values the mixture stays close to the given mixture, as high α values correspond to a small KL. When α decreases, the weights of the mixture terms with higher cost increase, while the weights of the lower cost terms decrease.

3.2.2. Conclusion

The idea of robustifying the weights of the mixture terms does not have the same issues as the previous method. As pointed out before, the existence of the solution does not depend on α. Also, there is a limit on how much the mixture can change. When α converges to zero, the solution will have a weight of one for the mixture term with the highest cost and zeros for the other terms. Since these issues are avoided, choosing the hyperparameter α is much simpler.

3.2.3. Optimistic Mixture Weights

Like the previous method, the problem can easily be changed to find the mixture weights, that lead to the minimal cost. By turning the maximization into a minimization, the equations stay the same, but the α is restricted to negative values.

Figure 3.4 shows how the weights change for different negative α values. For high absolute α values the weights stay close to the original values. When α decreases, the weight of the mixture terms with lower cost increase.

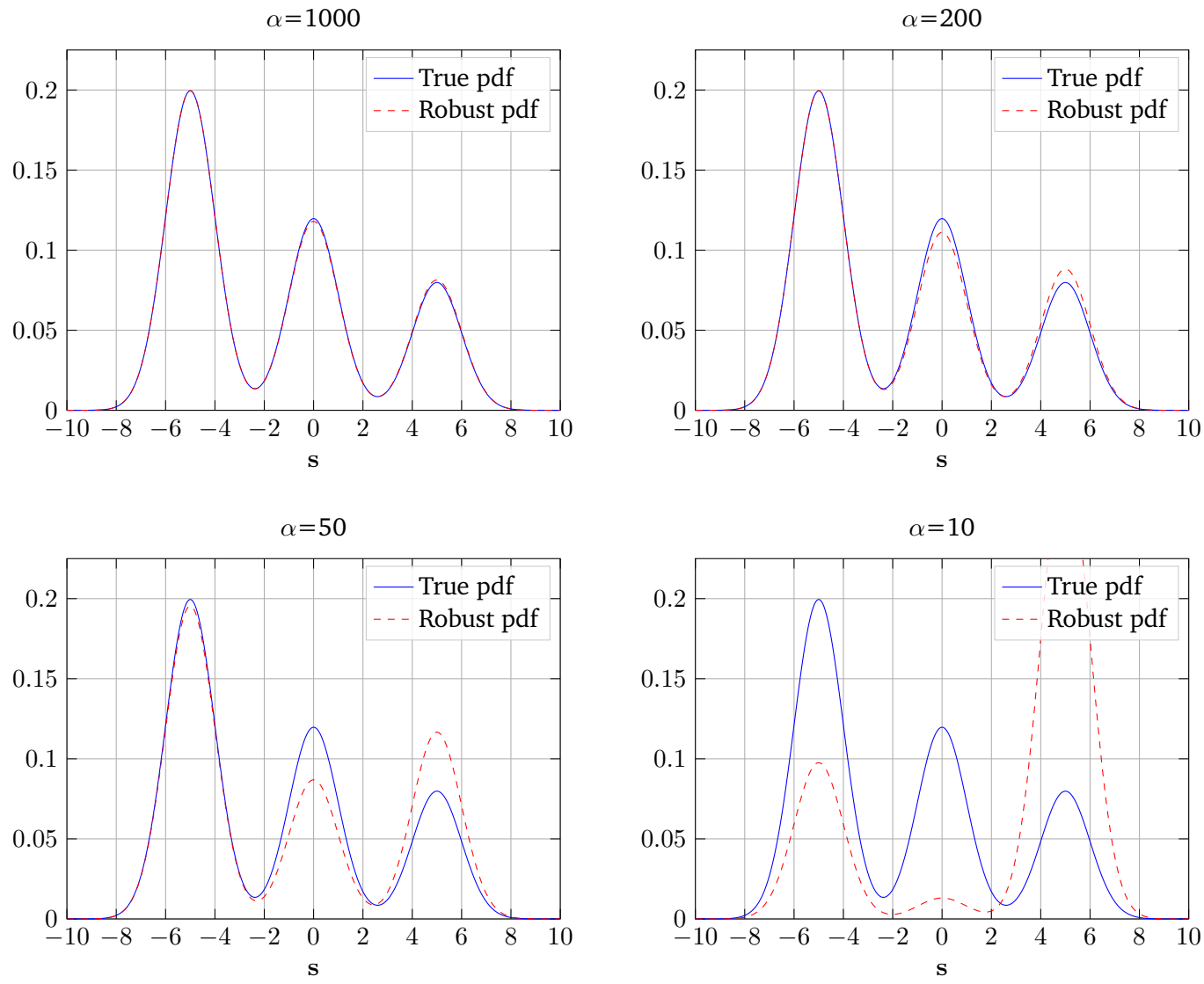

Figure 3.3.: Comparison of robust mixture weights with different α values. For high α values the robust weights are close to the original weights as high α values correspond to a small KL. For decreasing α, the weights of the higher cost terms increase.

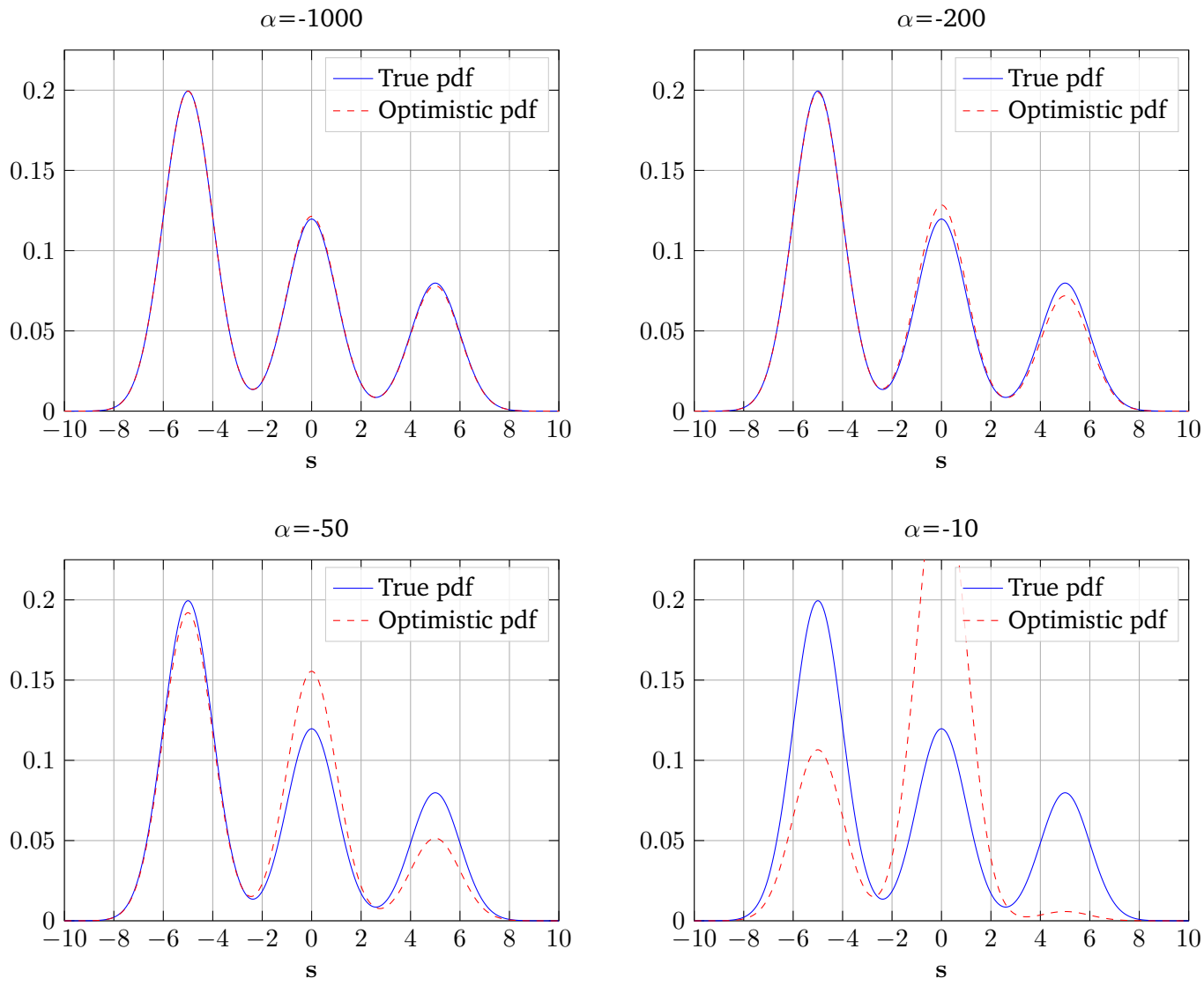

Figure 3.4.: Comparison of optimistic mixture weights with different α values. For high absolute α values the robust weights are close to the original weights as high α values correspond to a small KL. For decreasing α, the weights of the lower cost terms increase.

3.3. Robust Forward Pass

Our goal is to tackle the exponential growth of the forward pass in a robust way. Therefore, we use the method of robustifying the weights each time we use merging methods and thereby lose information. We do the merging similarly to IMM, but instead of merging a mixture of Gaussians to a single term, we choose a number k of terms we want to keep track of for each discrete state. Each time the number of terms exceeds k, we adjust the weights and merge the new terms to a mixture of k components with Runnalls' algorithm [19]. With the hyperparameter α we can decide on the level of robustness. Algorithm 2 shows the pseudocode for this robust forward pass.

input : T ; /* time horizon */
$\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})$; /* linear dynamics */
$P_t(\mathbf{z}'|\mathbf{z})$; /* Markov chain */
$\mu_1(\mathbf{s},\mathbf{z})$; /* initial state distribution */
$\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$; /* policy */
$C_t(\mathbf{s},\mathbf{z})$; /* cost function */
α; /* hyperparameter */
Z; /* number of discrete states */
k; /* limit on mixture terms */

output : $q_t(\mathbf{s},\mathbf{z})$; /* new distribution */

$q_1(\mathbf{s},\mathbf{z}) \leftarrow \mu_1(\mathbf{s},\mathbf{z})$;

for $t \leftarrow 1$ **to** $T-1$ **do**

  /* one step of forward pass using equation (2.12) */
  $\mu_{t+1}(\mathbf{s}',\mathbf{z}') \leftarrow$ forward_pass$(q_t(\mathbf{s},\mathbf{z}), \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}))$;

  for $\mathbf{z} \leftarrow 1$ **to** Z **do**

    if *size*$(\mu_{t+1}(\mathbf{s}',\mathbf{z}')) > k$ **then**

      /* adjust weights by equation (3.7) */
      $q_{t+1}(\mathbf{s},\mathbf{z}) \leftarrow$ adjust_weights$(\mu_{t+1}(\mathbf{s},\mathbf{z}), C(\mathbf{s},\mathbf{z}), \alpha)$;

      /* Use runnals algorithm to reduce the size of the mixture */
      $q_{t+1}(\mathbf{s},\mathbf{z}) \leftarrow$ runnals$(q_{t+1}(\mathbf{s},\mathbf{z}), k)$;

    else

      $q_{t+1}(\mathbf{s},\mathbf{z}) \leftarrow \mu_{t+1}(\mathbf{s},\mathbf{z})$;

Algorithm 2: Pseudocode of the robust forward pass

For the cost function we have several options. The simplest cost function would be to use the cost (negative reward) of the state and the input, given the control law $\mathbf{C}(\mathbf{s}) = \int_{\mathbf{a}} -R(\mathbf{s},\mathbf{a})\pi(\mathbf{a}|\mathbf{s})d\mathbf{a}$. This cost function, however, would not use information about the dynamics of the system. To include information about the dynamics, it is possible to do a backward pass over a chosen horizon to take future costs into account. If available, the best choice is the value function as it includes by definition the expected total cost of the trajectory of the state.

In the following we want to evaluate the robust forward pass on a random MJLS, given the optimal control and the value function. For that we sample trajectories with applied optimal control and evaluate the cost of

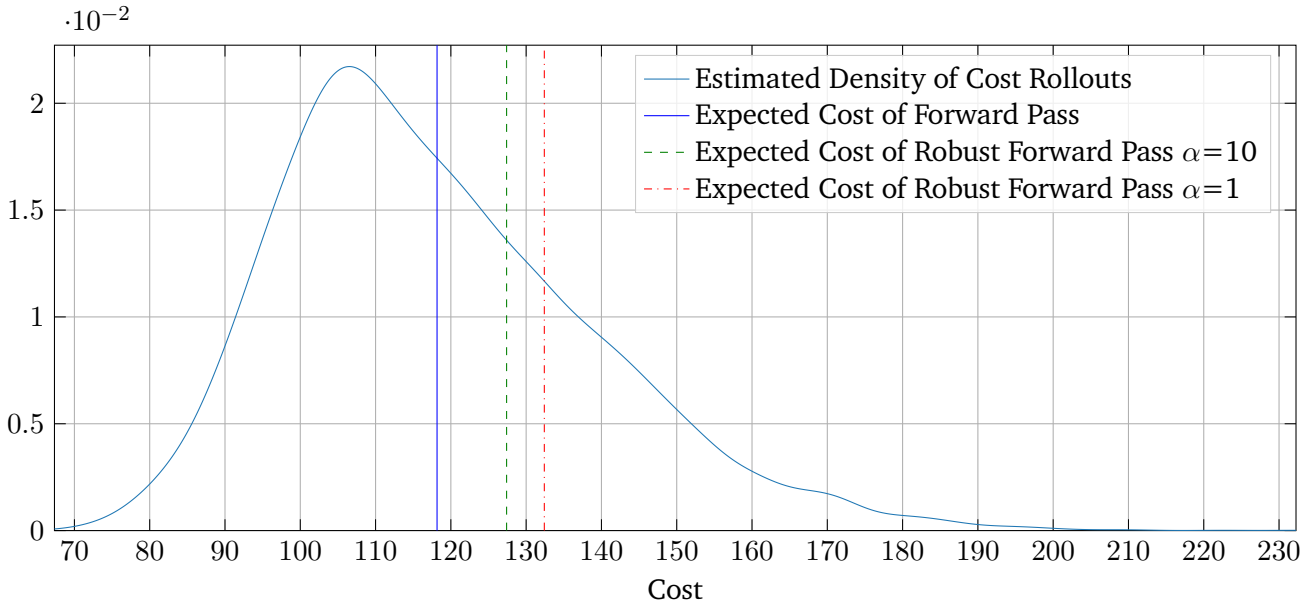

Figure 3.5.: Comparison of the cost of trajectory samples and the expected cost of robust forward passes with different α values. For high α values the expected cost of the robust forward pass is close to the expected cost of the normal forward pass. For decreasing α values the expected cost increases.

those samples. We now compare the expected cost under the robust forward pass with the samples. Figure 3.5 shows the estimated density of the cost samples. The vertical lines mark the expected cost of the different forward passes. The plot visualizes that the expected costs of the robust forward passes are higher than the cost of the normal forward pass. This is because each time step the weights are changed to the worse with respect to the cost function. When α decreases, the cost increases. As there is a lower limit on how much each distribution can change, there is also a limit on the cost. The robust forward pass with $\alpha = 1$ is very close to that limit. Smaller α values would not increase the expected cost. As can be seen, this limit on the cost is not the worst-case for the samples. This is because the method only effects the weights, not the uncertainties that come from dynamics or control. If we would have used the previous method from section 3.1, there would not be a limit on the cost, as there was no limit on the distributions. Figure 3.6 shows similar behavior with optimistic forward passes.

Having a different forward pass can be interpreted as having a different dynamic model. This new dynamic model changes the weights of the mixture terms in a robust way that leads to a higher cost. As the only term in the dynamic model that influences the weights of the Gaussian mixture is the Markov chain that updates the discrete state, this method is similar to having a worst-case Markov chain.

3.3.1. Effect of Robust Forward Pass on GPS routine

We now want to evaluate how the robust forward pass affects the GPS routine, when exchanging it with the traditional forward pass. Due to the different forward pass, the trajectory of the state will be different, given the same controller. This leads to a different α, as its gradient depends on the state. As the backward pass is influenced by α, the policy and value function in the next iteration are different than in the conventional GPS. But as long as α goes to zero, the backward pass converges to the bellman optimality equations and therefore

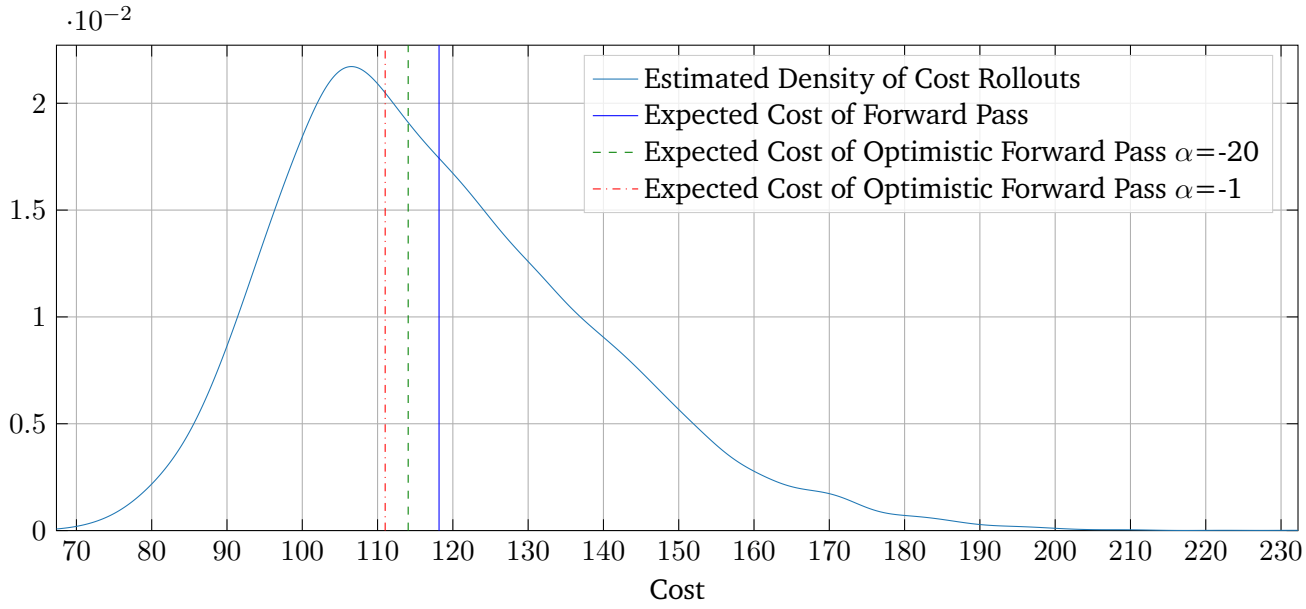

Figure 3.6.: Comparison of the cost of trajectory samples and the expected cost of optimistic forward passes with different α values. For high absolute α values the expected cost of the robust forward pass is close to the expected cost of the normal forward pass. For decreasing α values the expected cost decreases.

yields the same optimal control as the approach with the normal forward pass. Therefore our intuition is that this approach does not effect the solution, but only the rate of convergence.

We tested this on 50 random MJLS by comparing the normal GPS routine with the altered routine. On all systems both approaches converged to the same solutions for the optimal control. In all tested cases the altered routine took more or the same amount of iterations to converge.

The robust forward pass can be seen as a robust prediction since it yields a higher cost state distribution. Using the GPS routine with this robust forward pass, however, does not result in a different optimal controller that is robust to these changes. This is because the new forward pass can be interpreted as having a different dynamic model. If we want the policy to adapt to the changes, the forward pass and the backward pass need to consider the new model. By only changing the forward pass this new dynamic model is not included in the backward pass. Instead of changing the forward pass after the derivation of the GPS routine, the forward pass constraint in the formulation of the GPS optimization problem should be changed. This would induce the new dynamic model in the backward pass.

In the following we observe a different approach, where we use the mentioned similarity of this method compared to changing the Markov chain.

4. Distributionally Robust Trajectory Optimization

The last chapter mentioned that adjusting the weights of the mixture terms is very similar to changing the Markov chain that defines the state update for the discrete state $\mathbf{z}$. This change of the dynamics of the system can lead to a higher cost of the trajectories. Ultimately, we want a controller that adapts to these changes and minimizes the cost in a robust way. Therefore, we formulate a minimax optimization problem, where the expected reward is minimized with respect to the new Markov chain $D_t(\mathbf{z}'|\mathbf{z})$ and maximized with respect to the policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$. As the Markov chain $D_t(\mathbf{z}'|\mathbf{z})$ defines the distribution of $\mathbf{z}'$, this can be seen as a distributionally robust optimization. As ambiguity set, we choose all Markov chains that lie in a KL-ball around the nominal distribution $P_t(\mathbf{z}'|\mathbf{z})$. This way it is possible to include known information about the parameters. By the size of the KL-ball we can define how much we trust the nominal behavior.

The following optimization problem is similar to the one in [30]. This work proposes a distributional robust control with respect to changes in the linear dynamics.

4.1. Optimization Problem

First the optimization problem from chapter 2 is extended by adding the KL constraint on the Markov chain and the minimization with respect to $D_t(\mathbf{z}'|\mathbf{z})$. This leads to the following optimization problem:

$$
\begin{aligned}
\min_{D_t(\mathbf{z}'|\mathbf{z})} \max_{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})} & \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s}, \\
\text{s.t.} & \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_{t-1}(\mathbf{s},\mathbf{z}) \pi_{t-1}(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_{t-1}(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) D_{t-1}(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} = \mu_t(\mathbf{s}',\mathbf{z}') \qquad \forall \mathbf{s}', \forall \mathbf{z}', \forall t > 1, \\
& \mu_1(\mathbf{s},\mathbf{z}) = p_1(\mathbf{s},\mathbf{z}) \qquad \forall \mathbf{s}, \forall \mathbf{z}, \\
& \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) = 1 \qquad , \forall \mathbf{z}, \forall t < T, \\
& \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \leq \delta, \\
& \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} d\mathbf{a} d\mathbf{s} \leq \epsilon.
\end{aligned}
$$

This optimization problem is solved in an alternate fashion, where we repeatedly fix $D_t(\mathbf{z}'|\mathbf{z})$ to get a policy update by solving the maximization in chapter 2 and then fix the policy $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$ and search for the worst-case

distribution $D_t(\mathbf{z}'|\mathbf{z})$ by solving:

$$\begin{aligned}
&\min_{D_t(\mathbf{z}'|\mathbf{z})} \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s}, \\
&\quad \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_{t-1}(\mathbf{s},\mathbf{z})(\mathbf{z}) \pi_{t-1}(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_{t-1}(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) D_{t-1}(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} = \mu_t(\mathbf{s}',\mathbf{z}') \qquad \forall \mathbf{s}', \forall \mathbf{z}', \forall t > 1, \\
&\quad \mu_1(\mathbf{s},\mathbf{z}) = p_1(\mathbf{s},\mathbf{z}) \qquad \forall \mathbf{s}, \forall \mathbf{z}, \\
&\quad \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) = 1 \qquad \forall t < T, \forall \mathbf{z}, \\
&\quad \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \leq \delta.
\end{aligned}$$

As the maximization is solved in chapter 2, we focus on the minimization. The primal problem is formulated by using the method of Lagrangian multipliers:

$$\begin{aligned}
L(\mu_t, D_t, \beta_t, V_t, \alpha) = &\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}.\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
&+ \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) D_t(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}', \\
&- \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}',\mathbf{z}') \mu_t(\mathbf{s}',\mathbf{z}') d\mathbf{s}' \\
&+ \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s},\mathbf{z}) p_1(\mathbf{s},\mathbf{z}) d\mathbf{s} - \int_{\mathbf{s}} \sum_{\mathbf{z}} V_T(\mathbf{s},\mathbf{z}) \mu_T(\mathbf{s},\mathbf{z}) d\mathbf{s} \\
&+ \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \beta_t(\mathbf{z}) \left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) - 1 \right) \\
&+ \alpha \left(\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} - \delta \right).
\end{aligned}$$

Solving

$$\frac{\partial L}{\partial \beta_t} = 0, \qquad \frac{\partial L}{\partial D_t} = 0,$$

gives an expression for the worst-case Markov chain:

$$D_t(\mathbf{z}'|\mathbf{z}) \propto P_t(\mathbf{z}'|\mathbf{z}) \exp \left[\frac{-1}{\alpha} \left(\int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') d\mathbf{s} \right) \right],$$

where

$$Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') = \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) d\mathbf{a} d\mathbf{s}'.$$

By plugging in the solution of the primal problem, the dual problem is obtained:

$$
\begin{aligned}
L(\mu_t, V_t, \alpha) = &\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s}, \mathbf{a}) \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a} d\mathbf{s} - \sum_{t=1}^{T} \sum_{\mathbf{z}} \int_{\mathbf{s}} V_t(\mathbf{s}, \mathbf{z}) \mu_t(\mathbf{s}, \mathbf{z}) d\mathbf{s} \\
&+ \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}, \mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
&+ \sum_{t=1}^{T-1} \sum_{\mathbf{z}} -\alpha(\delta + \log \sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z})) Q_t(\mathbf{s}, \mathbf{z}, \mathbf{z}') d\mathbf{s} \right] \right).
\end{aligned}
$$

Using the principle of duality [28], the original problem is solved by maximizing the dual problem. Therefore we calculate the partial derivatives:

$$
\frac{\partial L}{\partial \mu_t} = \begin{cases} R_T(\mathbf{s}, \mathbf{a}) - V_T(\mathbf{s}, \mathbf{z}) & , t = T \\ \int_{\mathbf{a}} R_t(\mathbf{s}, \mathbf{a}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a} - V_t(\mathbf{s}, \mathbf{z}) + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) Q_t(\mathbf{s}, \mathbf{z}, \mathbf{z}') & , t < T \end{cases} \tag{4.1}
$$

$$
\frac{\partial L}{\partial V_t} = \begin{cases} -\mu_1(\mathbf{s}, \mathbf{z}) + p_1(\mathbf{s}, \mathbf{z}) & , t = 1 \\ -\mu_t(\mathbf{s}, \mathbf{z}) + \sum_{\hat{\mathbf{z}}} D_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \pi_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} & , t > 1 \end{cases} \tag{4.2}
$$

$$
\frac{\partial L}{\partial \alpha} = \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) - \delta \right). \tag{4.3}
$$

Setting the partial derivatives (4.1) and (4.2) to zero leads to a forward pass for the state $\mu_t(\mathbf{s}, \mathbf{z})$ and a backward pass for the Lagrangian multiplier $V_t(\mathbf{s}, \mathbf{z})$ as optimality conditions:

$$
\mu_t(\mathbf{s}, \mathbf{z}) = \begin{cases} p_1(\mathbf{s}, \mathbf{z}) & , t = 1 \\ \sum_{\hat{\mathbf{z}}} D_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \pi_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} & , t > 1 \end{cases} \tag{4.4}
$$

$$
V_t(\mathbf{s}, \mathbf{z}) = \begin{cases} R_T(\mathbf{s}, \mathbf{a}) & , t = T \\ \int_{\mathbf{a}} R_t(\mathbf{s}, \mathbf{a}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a} + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) Q_t(\mathbf{s}, \mathbf{z}, \mathbf{z}') & , t < T \end{cases}. \tag{4.5}
$$

Plugging the optimality conditions back in the Lagrangian the dual simplifies to:

$$
L(\mu_t, V_t, \beta_t, \alpha, D_t) = \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} + \alpha \left(\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) - \delta \right). \tag{4.6}
$$

Using (4.3) we can maximize (4.6) with gradient methods until convergence. The full derivation of the optimization problem is shown in appendix D.1.

We can again interpret the Lagrangian multiplier $V_t(\mathbf{s}, \mathbf{z})$ as state value function. $Q_t(\mathbf{s}, \mathbf{z}, \mathbf{z}')$ represents the state action value function, where the "action" is the discrete state in the next time step $\mathbf{z}'$. Instead of a policy choosing an action given the state, we find the Markov chain $D_t(\mathbf{z}'|\mathbf{z})$, that chooses the next discrete state given the current state. This describes the adversary choosing the worst-case discrete state to minimize the reward.

4.2. Implementation

4.2.1. Worst-Case Distribution

Using the LQG assumptions, we get closed-form solutions for the forward pass and the backward pass. For the full derivation, see appendix D.2. In contrast to the minimization in chapter 2, there is an issue concerning the backward pass and the forward pass of the solution. In order to calculate $D_t(\mathbf{z}'|\mathbf{z})$, the state distribution is needed in the backward pass. However, to calculate the state distribution in the forward pass, the Markov chain $D_t(\mathbf{z}'|\mathbf{z})$ is needed. To tackle this circular dependence we use a barycentric interpolation scheme as proposed in [30] .

Therefore, we start by calculating the state distribution given the Markov chain $P_t(\mathbf{z}'|\mathbf{z})$. Using this state distribution we then do the backward pass to get the new Markov chain $D_t(\mathbf{z}'|\mathbf{z})$. Then we interpolate and repeat the procedure with the new Markov chain. When converging, that means using a Markov chain in the forward pass, that leads to the same one as a result of the backward pass, the optimality conditions (4.4) and (4.5) are guaranteed. Note that the same procedure can be done with an interpolation of the state distribution instead of the Markov chain.

For the interpolation between two Markov chains the following optimization problem is solved:

$$\begin{aligned}\min_{H_t(\mathbf{z}'|\mathbf{z})} \lambda \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \log\left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})}\right) + (1-\lambda) \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \log\left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})}\right) \\ \text{s.t.} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) = 1, \qquad \forall \mathbf{z}, t < T,\end{aligned}$$

which leads to the interpolated Markov chain

$$H_t(\mathbf{z}'|\mathbf{z}) = \frac{D_t(\mathbf{z}'|\mathbf{z})^\lambda P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}}{\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})^\lambda P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}}. \tag{4.7}$$

See appendix E for the derivation of the solution to the optimization problem.

Using the barycentric interpolation scheme, the forward pass and the backward pass are repeatedly solved to get the Markov chain update. With this new distribution the gradient (4.3) can be calculated and α can be a updated with a bisection method. Algorithm 3 shows the pseudocode for the optimization.

4.2.2. Minimax Optimization

We now use Algorithm 1 and 3 in an alternate fashion to solve the minimax optimization problem. It is important to limit the KL-bound of the policy step. Too high policy updates can cause oscillations and will prevent the algorithm from converging. Algorithm 4 provides the pseudocode for the opimization problem.

input : T ; /* time horizon */
$\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})$; /* linear dynamics */
$P_t(\mathbf{z}'|\mathbf{z})$; /* nominal Markov chain */
$\mu_1(\mathbf{s},\mathbf{z})$; /* initial state distribution */
$\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$; /* policy */
$R_t(\mathbf{s},\mathbf{z})$; /* reward function */
δ, λ; /* KL bound and interpolation parameter */

output : $D_t(\mathbf{z}'|\mathbf{z})$; /* worst-case distribution */

initialize : α

while $L(\mu_t, V_t, \alpha, P_t, D_t)$ *not at maximum* **do**

$H_t(\mathbf{z}'|\mathbf{z}) \leftarrow P_t(\mathbf{z}'|\mathbf{z})$;

while $KL(D||H)$ <*threshold* **do**

/* compute the state distribution using equation (4.2) */
$\mu_t(\mathbf{s}) \leftarrow$ r_forward_pass$(\mu_1(\mathbf{s},\mathbf{z}), \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), H_t(\mathbf{z}'|\mathbf{z}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}))$;

/* compute value function and Markov chain using equation (4.1) */
$[V_t(\mathbf{s},\mathbf{z}), D_t(\mathbf{z}'|\mathbf{z})] \leftarrow$ r_backward_pass$(\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), \mu_t(\mathbf{s}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}), R_t(\mathbf{s},\mathbf{a}), \alpha)$;

/* compute the interpolated distribution using equation (4.7) */
$H_t(\mathbf{z}'|\mathbf{z}) \leftarrow$ barycentric$(H_t(\mathbf{z}'|\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), \lambda)$;

/* update dual value with equation (4.6) */
$L(\mu_t, V_t, \alpha, D_t, P_t) \leftarrow$ r_update_dual$(\mu_1(\mathbf{s},\mathbf{z}), V_1(\mathbf{s},\mathbf{z}), \alpha, \delta, D_t, P_t)$;

/* compute dual gradient with respect to α using equation (4.3) */
$\frac{\partial L}{\partial \alpha} \leftarrow$ r_dual_alpha_gradient$(\mu_t(\mathbf{s}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}), \delta)$;

/* bisection method to find optimal α */
$\alpha \leftarrow$ bisection$(L(\mu_t, V_t, \alpha, D_t, P_t), \frac{\partial L}{\partial \alpha})$;

Algorithm 3: Pseudocode for finding the worst-case distribution given a policy

input : T ; /* time horizon */
$\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})$; /* linear dynamics */
$P_t(\mathbf{z}'|\mathbf{z})$; /* nominal Markov chain */
$\mu_1(\mathbf{s},\mathbf{z})$; /* initial state distribution */
$q_t(\mathbf{a}|\mathbf{s},\mathbf{z})$; /* initial policy */
$R_t(\mathbf{s},\mathbf{z})$; /* reward function */
$\delta, \epsilon, \lambda$; /* KL bounds and interpolation parameter */

output : $\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$; /* optimal robust policy */
$D_t(\mathbf{z}'|\mathbf{z})$; /* worst-case distribution */

while *not converged* **do**
  initialize : α
  while $L(\mu_t, V_t, \alpha, P_t, D_t)$ *not at maximum* **do**
    $H_t(\mathbf{z}'|\mathbf{z}) \leftarrow P_t(\mathbf{z}'|\mathbf{z})$;
    while $KL(D||H)$ <*threshold* **do**
      /* compute the state distribution using equation (4.2) */
      $\mu_t(\mathbf{s}) \leftarrow$ r_forward_pass$(\mu_1(\mathbf{s},\mathbf{z}), \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), H_t(\mathbf{z}'|\mathbf{z}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}))$;
      /* compute value function and Markov chain using equation (4.1) */
      $[V_t(\mathbf{s},\mathbf{z}), D_t(\mathbf{z}'|\mathbf{z})] \leftarrow$ r_backward_pass$(\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), \mu_t(\mathbf{s}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}), R_t(\mathbf{s},\mathbf{a}), \alpha)$;
      /* compute the interpolated distribution using equation (4.7) */
      $H_t(\mathbf{z}'|\mathbf{z}) \leftarrow$ barycentric$(H_t(\mathbf{z}'|\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), \lambda)$;
    /* update dual value using equation (4.6) */
    $L(\mu_t, V_t, \alpha, D_t, P_t) \leftarrow$ r_update_dual$(\mu_1(\mathbf{s},\mathbf{z}), V_1(\mathbf{s},\mathbf{z}), \alpha, \delta, D_t, P_t)$;
    /* compute dual gradient w.r.t. α using equation (4.3) */
    $\frac{\partial L}{\partial \alpha} \leftarrow$ r_dual_alpha_gradient$(D_t(\mathbf{z}'|\mathbf{z}), P_t(\mathbf{z}'|\mathbf{z}), \delta)$;
    /* bisection method to find optimal α */
    $\alpha \leftarrow$ bisection$(L(\mu_t, V_t, \alpha, D_t, P_t), \frac{\partial L}{\partial \alpha})$;
  initialize : α
  while $L(\mu_t, V_t, \alpha)$ *not at minimum* **do**
    /* compute value function and policy using equation (2.11) */
    $[V_t(\mathbf{s},\mathbf{z}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})] \leftarrow$ backward_pass$(\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), D_t(\mathbf{z}'|\mathbf{z}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}), R_t(\mathbf{s},\mathbf{a}), \alpha)$;
    /* compute the state distribution using equation (2.12) */
    $\mu_t(\mathbf{s}) \leftarrow$ forward_pass$(\mu_1(\mathbf{s},\mathbf{z}), \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}), D_t(\mathbf{z}'|\mathbf{z}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}))$;
    /* update dual value with equation (2.13) */
    $L(\mu_t, V_t, \alpha) \leftarrow$ update_dual$(\mu_1(\mathbf{s},\mathbf{z}), V_1(\mathbf{s},\mathbf{z}), \alpha, \epsilon)$;
    /* compute dual gradient w.r.t. α using equation (2.10) */
    $\frac{\partial L}{\partial \alpha} \leftarrow$ dual_alpha_gradient$(\mu_t(\mathbf{s}), \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}), q_t(\mathbf{a}|\mathbf{s},\mathbf{z}), \epsilon)$;
    /* bisection method to find optimal α */
    $\alpha \leftarrow$ bisection$(L(\mu_t, V_t, \alpha), \frac{\partial L}{\partial \alpha})$;
  $q_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \leftarrow \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$;

Algorithm 4: Pseudocode for the minimax optimization

4.3. Evaluation

In the following we want to evaluate our solutions on a random MJLS with two discrete states and two continuous states. Therefore, we calculate the optimal policy of the nominal distribution $P_t(\mathbf{z}'|\mathbf{z})$ and the robust policy that is optimal on the worst-case distribution $D_t(\mathbf{z}'|\mathbf{z})$ with $\delta = 3$. First, we compare the policies on the nominal distribution. The optimal control leads to an expected cost of 381.43, while the robust control leads to an expected cost of 404.07. Figure 4.1 compares the trajectories of the different policies applied to

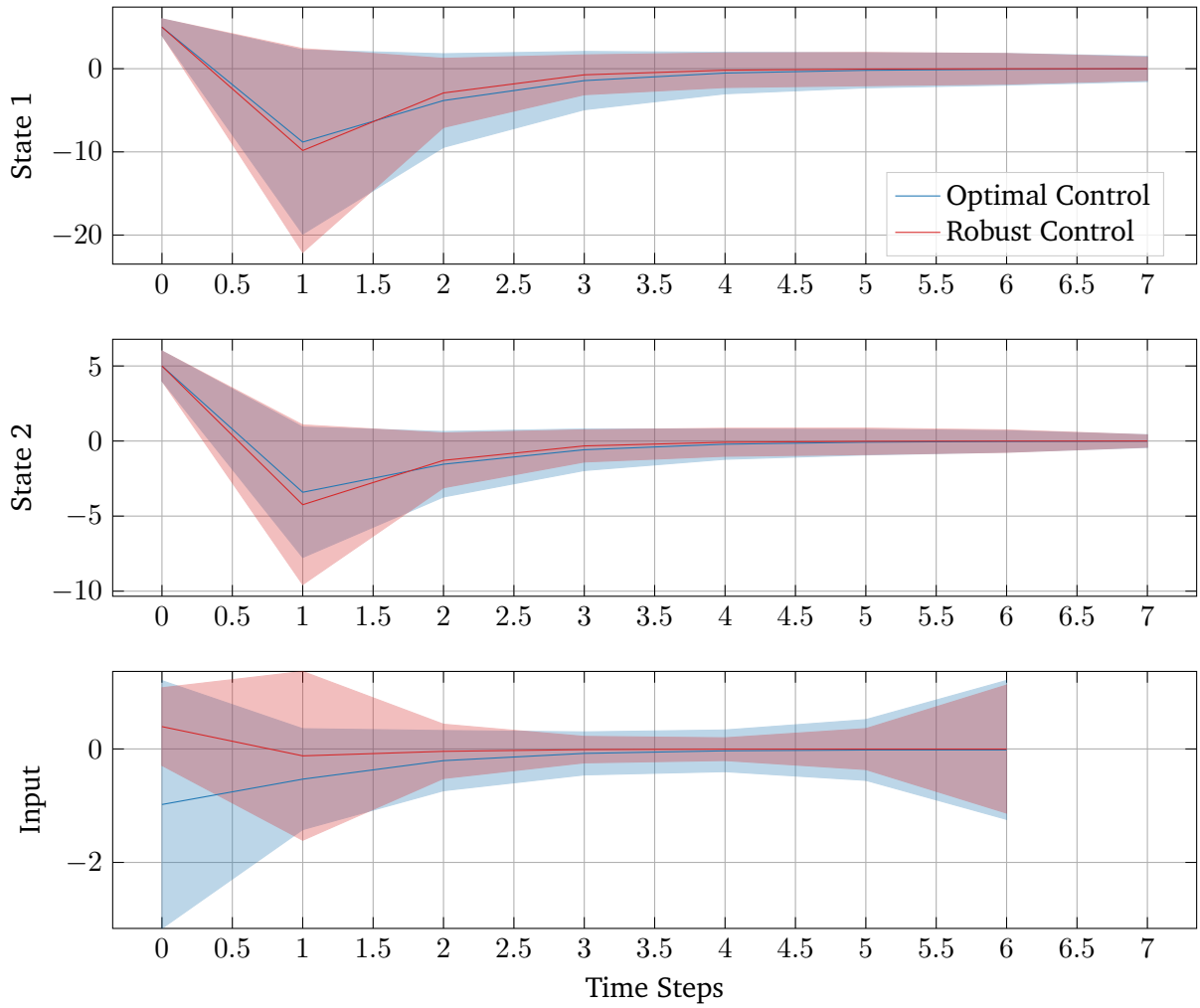

Figure 4.1.: Trajectories of the optimal policy and the robust policy evaluated on the nominal behavior. The robust policy brings the state faster to the origin, but due the higher cost in time step 1, it has higher total cost than the optimal policy

the system with nominal behavior. For both policies the typical behavior of controlled MJLS can be seen. In the beginning uncertainty rises, due to the different dynamic models for each discrete state, but after some time steps the controller reduces the uncertainty, when bringing the state to the origin. The robust policy reaches the target a little faster, but due to the higher cost time step one, it performs worse. In general, the trajectories are very similar. The biggest difference lies in the input of the first time steps.

Second, we compare the policies on the worst-case distribution. The optimal control leads to an expected cost of 506.37, while the robust control leads to an expected cost of 442.92. Figure 4.2 shows the trajectories

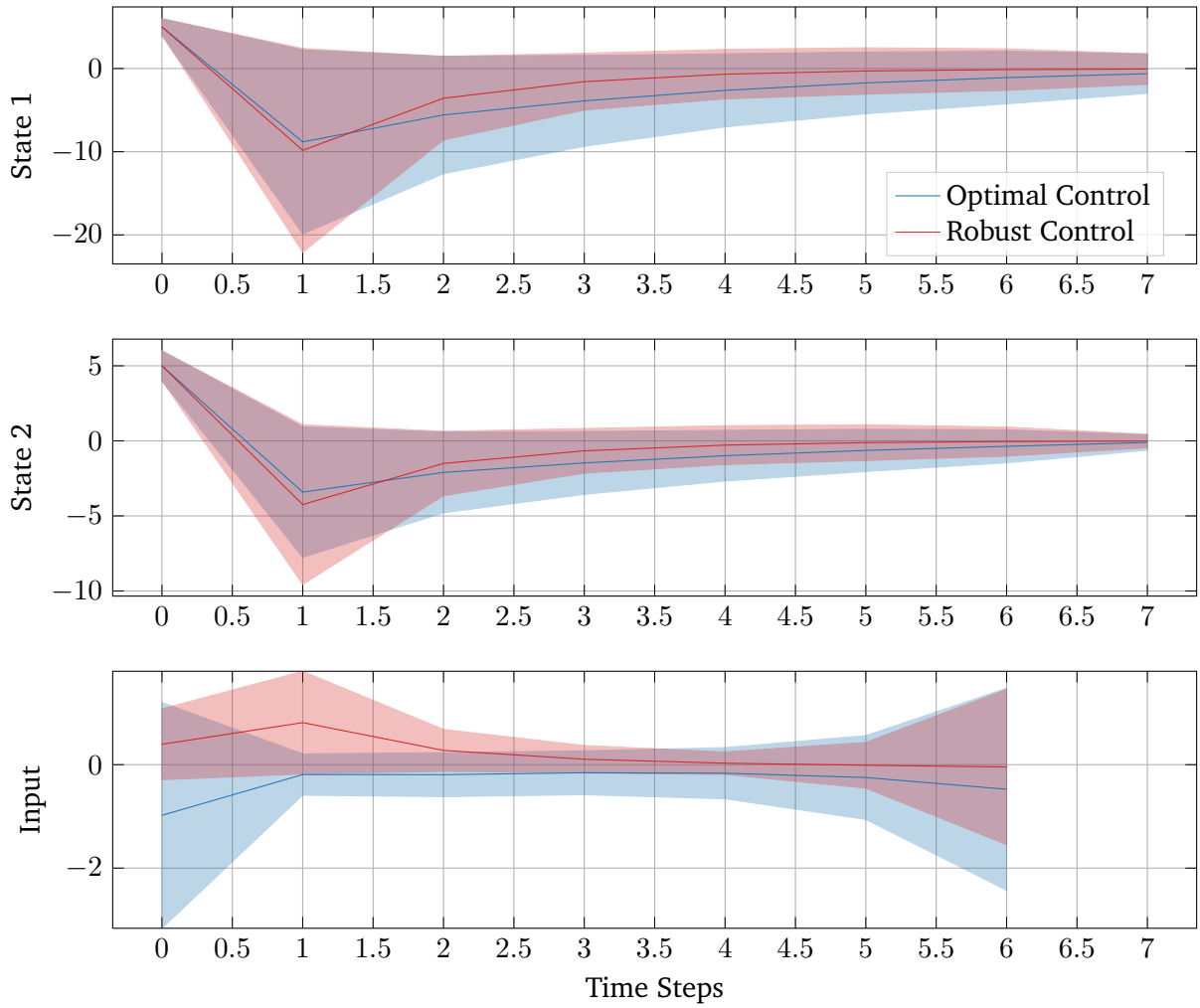

Figure 4.2.: Trajectories of the optimal policy and the robust policy evaluated on the worst-case distribution. The robust policy again has a higher cost in time step 1, but as it brings the state much faster to the origin, it has a better performance than the optimal policy.

of the system with worst-case behavior with the same initial conditions. The state distribution at time step 1 is the same as with the nominal behavior, because the worst-case behavior only influences the discrete state in the first state update. Therefore, the robust trajectory has again a higher cost in the first time step. In the following time steps, the robust trajectory moves very fast to the origin, while the optimal controller converges slowly. The figure shows, that the robust controller is better suited for the worst-case behavior.

Figure 4.3 shows the KL between the nominal Markov chain $P_t(\mathbf{z}'|\mathbf{u})$ and $D_t(\mathbf{z}'|\mathbf{u})$ over time. As the KL constraint is over the total horizon, the adversary can freely choose how to allocate the KL over the time steps. The plot shows a typical allocation of the KL, where the change is highest in the beginning and decreases over time. An early change in the dynamics will have the biggest impact on the state distribution over time. If we would only have a terminal cost, the allocation would probably be higher in the later steps of the trajectory. In the last time step the KL is zero, as the last Markov chain does not affect the final continuous state.

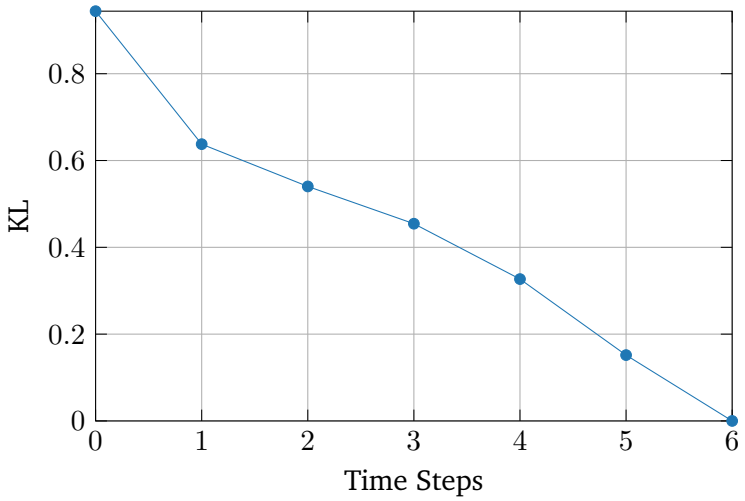

Figure 4.3.: Allocation of the KL over the time steps. Most of the KL is allocated in the early time steps, as a change of the dynamics in the beginning has the highest influence on the total cost.

The comparison between robust and optimal control seems unfair, as the robust control was optimized on the worst-case behavior. Therefore, we compare both control laws on a set of different parameters. For that we use again the barycentric interpolation scheme (4.7) to inter- and extrapolate between the Markov chains. Figure 4.4 visualizes the expected cost of the controllers on the different distributions. The cost of both controllers rises, when the KL grows. The optimal control starts with a lower cost when very close to the nominal distribution, but the robust control is much less affected by the changes in the dynamics and therefore surpasses the performance of the optimal control very early on. Even for high KL the robust control still leads to a good performance. The figure also shows that the expected cost converges for both controllers, when the KL increases. This behavior comes from the fact that there is a limit, how much the discrete distributions can change. This effect is similar to the one that occurs when robustifying the mixture weights as described in the previous chapter. At this limit the distributionally robust optimization changes basically to a robust optimization, as the ambiguity set contains the worst-case distribution.

Finally, we evaluate the distributionally robust control on a variety of systems. Therefore we compare the optimal and the robust policy on 70 randomly generated MJLS. Figure 4.5 shows the relative difference of the reward by the robust policy over the optimal policy. For all systems, the robust controller performs better on the system with worst-case distribution, while the opposite is true on the nominal distribution as they were optimized for each scenario. The plots visualize, that for most systems the difference between the two policies is insignificant, while for some systems the difference is large compared to the others. As the KL between the nominal distribution and the worst-case distribution was the same for all systems, these big differences come from the system dynamics.

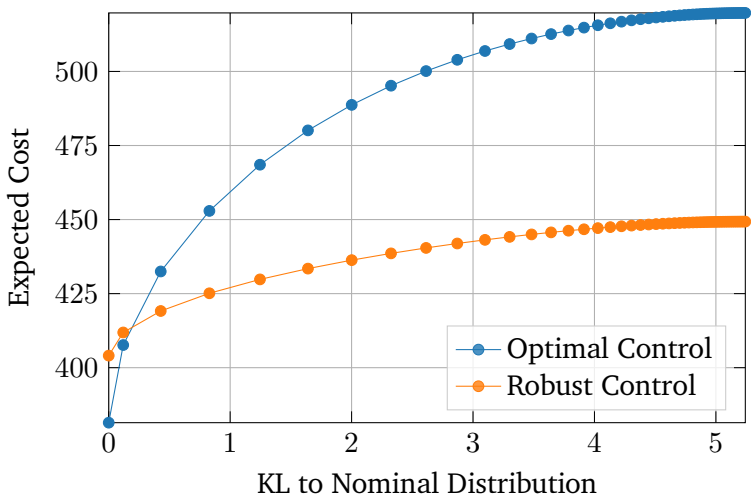

Figure 4.4.: Comparison of robust policy and optimal policy on distributions with different KL to the nominal distribution. For small KL the optimal policy performs better, but the robust policy outperforms the optimal policy very early on and is less affected by an increasing KL.

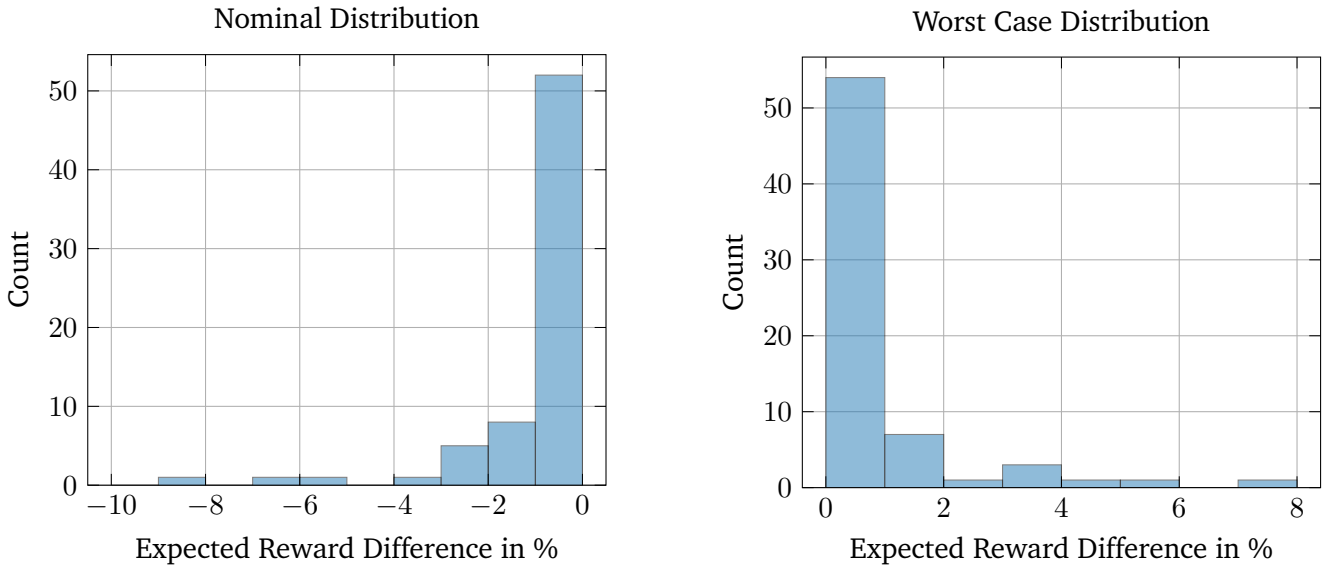

Figure 4.5.: Relative increase in expected trajectory reward by robust over nominal policy evaluated on the nominal distribution and the worst-case distribution for 70 random MJLS. The optimal policy always outperforms the robust policy on the nominal distribution, while the robust policy yields a better performance on the worst-case distribution. For most systems the difference is very small.

5. Conclusion

In this work, we extended the GPS framework to MJLS. Therefore, we formulated an iterative optimization problem that can be used to find the optimal control to a model learned from data. We presented the closed-form solutions and the pseudocode for the resulting algorithm. However, the closed-form solutions come with an exponential growth in the forward pass, which is typical for stochastic hybrid systems.

To tackle this in a robust, way we presented a method that combines the redistribution of the mixture weights with merging techniques. The solution to this method is a robust forward pass that yields a robust prediction about the state distribution. We discussed the effect of using this robust forward pass instead of the traditional forward pass in the GPS routine and saw that it does not lead to a policy that adapts to the changes. To get a robust policy, the optimization problem of the GPS needs to be reformulated to include the change of the forward pass in the constraints.

Instead, we followed a different approach using the similarity between the change in the forward pass and the change in the Markov chain $P_t(\mathbf{z}'|\mathbf{z})$ that describes the dynamics of the discrete state. We formulated a distributionally robust optimization problem that leads to a robust controller with respect to changes in the discrete state. Finally, we evaluated the robust policy on random MJLS and saw that it outperforms the optimal policy of the nominal behavior not only for worst-case distributions, but also when there are only small changes to the nominal dynamics.

The work presented here serves as a starting point for working on solutions for more sophisticated stochastic hybrid models. Future work could be to use the presented methods on nonlinear jumping models. Therefore linearized models along the trajectory are needed to solve the problems for optimal and robust control iteratively. The major challenge in this approach is to find linear models that describe the dynamics locally well enough, when having multiple assumptions about the state.

Besides that, the problem of reformulating the GPS optimization problem with the robust forward pass as constraint needs to be researched. The resulting robust policy could be used to further investigate the similarities to the here proposed distributionally robust controller.

Bibliography

[1] H. K. Khalil, *Nonlinear systems; 3rd ed.* Upper Saddle River, NJ: Prentice-Hall, 2002.

[2] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *The International Journal of Robotics Research*, vol. 32, no. 11, pp. 1238–1274, 2013.

[3] B. Charlet, J. Lévine, and R. Marino, "On dynamic feedback linearization," *Systems & Control Letters*, vol. 13, no. 2, pp. 143–151, 1989.

[4] P. Kokotovic, "The joy of feedback: nonlinear and adaptive," *IEEE Control Systems Magazine*, vol. 12, pp. 7–17, June 1992.

[5] J. Schulman, S. Levine, P. Abbeel, M. Jordan, and P. Moritz, "Trust region policy optimization," in *Proceedings of the 32nd International Conference on Machine Learning* (F. Bach and D. Blei, eds.), vol. 37 of *Proceedings of Machine Learning Research*, (Lille, France), pp. 1889–1897, PMLR, 07–09 Jul 2015.

[6] D. Liberzon, *Switching in systems and control*. Springer Science & Business Media, 2003.

[7] S. Linderman, M. Johnson, A. Miller, R. Adams, D. Blei, and L. Paninski, "Bayesian Learning and Inference in Recurrent Switching Linear Dynamical Systems," in *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics* (A. Singh and J. Zhu, eds.), vol. 54 of *Proceedings of Machine Learning Research*, pp. 914–922, PMLR, 20–22 Apr 2017.

[8] J. Lygeros and M. Prandini, "Stochastic hybrid systems: A powerful framework for complex, large scale applications," *European Journal of Control*, vol. 16, p. 583594, 11 2010.

[9] R. Goebel, R. G. Sanfelice, and A. R. Teel, *Hybrid dynamical systems*. Princeton University Press, 2012.

[10] F. Borrelli, A. Bemporad, and M. Morari, *Predictive Control for Linear and Hybrid Systems*. Cambridge University Press, 2017.

[11] T. Marcucci and R. Tedrake, "Mixed-integer formulations for optimal control of piecewise-affine systems," in *Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control*, HSCC '19, (New York, NY, USA), p. 230239, Association for Computing Machinery, 2019.

[12] A. Bemporad and M. Morari, "Control of systems integrating logic, dynamics, and constraints," *Automatica*, vol. 35, no. 3, pp. 407–427, 1999.

[13] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, *Discrete-time Markov jump linear systems*. Springer Science & Business Media, 2006.

[14] M. D. Fragoso, "Discrete-time jump lqg problem," *International Journal of Systems Science*, vol. 20, no. 12, pp. 2539–2545, 1989.

[15] R. E. Bellman, *Dynamic Programming*. USA: Dover Publications, Inc., 2003.

[16] Y. Bar-Shalom, X. Li, and T. Kirubarajan, "Estimation with applications to tracking and navigation: Theory, algorithms and software," 2001.

[17] K. Murphy, "Switching kalman filters," 1998.

[18] H. Blom and Y. Bar-Shalom, "The interacting multiple model algorithm for systems with markovian switching coefficients," *IEEE Transactions on Automatic Control*, vol. 33, no. 8, pp. 780–783, 1988.

[19] A. Runnalls, "Kullback-leibler approach to gaussian mixture reduction," *Aerospace and Electronic Systems, IEEE Transactions on*, vol. 43, pp. 989 – 999, 08 2007.

[20] S. Kullback and R. A. Leibler, "On information and sufficiency," *Ann. Math. Statist.*, vol. 22, no. 1, pp. 79–86, 1951.

[21] D. Crouse, P. Willett, K. Pattipati, and L. Svensson, "A look at gaussian mixture reduction algorithms," pp. 1 – 8, 08 2011.

[22] H. Rahimian and S. Mehrotra, "Distributionally robust optimization: A review," 2019.

[23] D. Bertsimas and M. Sim, "The price of robustness," *Operations Research*, vol. 52, pp. 35–53, 02 2004.

[24] D. Mayne, "A second-order gradient method for determining optimal trajectories of non-linear discrete-time systems," *International Journal of Control*, vol. 3, no. 1, pp. 85–95, 1966.

[25] E. Todorov and W. Li, "A generalized iterative lqg method for locally-optimal feedback control of constrained nonlinear stochastic systems," in *Proceedings of the 2005, American Control Conference, 2005.*, pp. 300–306 vol. 1, 2005.

[26] S. Levine and P. Abbeel, "Learning neural network policies with guided policy search under unknown dynamics.," in *NIPS*, vol. 27, pp. 1071–1079, Citeseer, 2014.

[27] H. Abdulsamad, O. Arenz, J. Peters, and G. Neumann, "State-regularized policy search for linearized dynamical systems," in *Proceedings of the International Conference on Automated Planning and Scheduling (ICAPS)*, 2017.

[28] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge University Press, 2004.

[29] S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *SCIENCE*, vol. 220, no. 4598, pp. 671–680, 1983.

[30] H. Abdulsamad, T. Dorau, B. Belousov, J.-J. Zhu, and J. Peters, "Distributionally robust trajectory optimization under uncertain dynamics via relative-entropy trust regions," in *arXiv*, 2021.

A. Derivation of GPS for MJLS

A.1. Optimization Problem

$$
\begin{aligned}
&\underset{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{\operatorname{argmax}} \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
&\quad \text{s.t} \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} = 1 \qquad \forall \mathbf{s}, \forall \mathbf{z}, \forall t < T \\
&\quad \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_{t-1}(\mathbf{s},\mathbf{z}) \pi_{t-1}(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_{t-1}(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) P(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} = \mu_t(\mathbf{s}',\mathbf{z}') \qquad \forall \mathbf{s}', \forall \mathbf{z}', \forall t > 1 \\
&\quad \mu_1(\mathbf{s},\mathbf{z}) = p_1(\mathbf{s},\mathbf{z}) \qquad \forall \mathbf{s}, \forall \mathbf{z} \\
&\quad \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} d\mathbf{a} d\mathbf{s} \leq \epsilon
\end{aligned}
$$

Primal Problem

$$
\begin{aligned}
L(\pi_t, \mu_t, V_t, \lambda_t, \alpha) = &\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
&+ \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \lambda_t(\mathbf{s},\mathbf{z}) d\mathbf{s} - \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \lambda_t(\mathbf{s},\mathbf{z}) \left(\int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} \right) d\mathbf{s} \\
&+ \sum_{\mathbf{z}} \int_{\mathbf{s}} V_1(\mathbf{s},\mathbf{z}) \left(p_1(\mathbf{s},\mathbf{z}) \right) d\mathbf{s} - \sum_{\mathbf{z}} \int_{\mathbf{s}} V_T(\mathbf{s},\mathbf{z}) \left(\mu_T(\mathbf{s},\mathbf{z}) \right) d\mathbf{s} \\
&+ \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \left(\sum_{\mathbf{z}} \left[\int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) P_t(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} \right] \right) \\
&- \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}',\mathbf{z}') \mu_t(\mathbf{s}',\mathbf{z}') d\mathbf{s}' \\
&+ \alpha \left(\epsilon - \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} d\mathbf{a} d\mathbf{s} \right)
\end{aligned}
$$

$$\begin{aligned}\frac{\partial L}{\partial \pi_t} =& R_t(\mathbf{s},\mathbf{a})\mu_t(\mathbf{s},\mathbf{z}) - \lambda_t(\mathbf{s},\mathbf{z}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'}\mu_t(\mathbf{s},\mathbf{z})V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}' \\ & - \alpha\mu_t(\mathbf{s},\mathbf{z})\log\frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} - \alpha\mu_t(\mathbf{s},\mathbf{z})\end{aligned}$$

$$\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) = q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) - \frac{\lambda_t(\mathbf{s},\mathbf{z})}{\mu_t(\mathbf{s},\mathbf{z})} + \sum_{\mathbf{z}'}\int_{\mathbf{s}'}V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}' - \alpha\right)\right]$$

Dual
Inserting policy back in Lagrangian.

$$\begin{aligned}L(\pi_t,\mu_t,V_t,\lambda_t,\alpha) =& \sum_{\mathbf{z}}\int_{\mathbf{s}}\mu_T(\mathbf{s},\mathbf{z})R_T(\mathbf{s})d\mathbf{s} \\ & + \sum_{t=1}^{T-1}\sum_{\mathbf{z}}\int_{\mathbf{s}}\lambda_t(\mathbf{s},\mathbf{z})d\mathbf{s} \\ & + \sum_{\mathbf{z}}\int_{\mathbf{s}}V_1(\mathbf{s},\mathbf{z})\left(p_1(\mathbf{s},\mathbf{z})\right)d\mathbf{s} - \sum_{\mathbf{z}}\int_{\mathbf{s}}V_T(\mathbf{s},\mathbf{z})\left(\mu_T(\mathbf{s},\mathbf{z})\right)d\mathbf{s} \\ & - \sum_{t=1}^{T-1}\sum_{\mathbf{z}'}\int_{\mathbf{s}'}V_t(\mathbf{s}',\mathbf{z}')\mu_t(\mathbf{s}',\mathbf{z}')d\mathbf{s}' \\ & + \alpha(\epsilon + \sum_{t=1}^{T-1}1)\end{aligned}$$

Solve for λ_t

$$1 = \int_{\mathbf{a}}\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})d\mathbf{a}$$

$$1 = \int_{\mathbf{a}}q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) - \frac{\lambda_t(\mathbf{s},\mathbf{z})}{\mu_t(\mathbf{s},\mathbf{z})} + \sum_{\mathbf{z}'}\int_{\mathbf{s}'}V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}' - \alpha\right)\right]d\mathbf{a}$$

$$1 = \int_{\mathbf{a}}q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'}V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]\exp\left(-\frac{\lambda_t(\mathbf{s},\mathbf{z})}{\mu_t(\mathbf{s},\mathbf{z})\alpha} - 1\right)d\mathbf{a}$$

$$\exp\left(\frac{\lambda_t(\mathbf{s},\mathbf{z})}{\mu_t(\mathbf{s},\mathbf{z})\alpha} + 1\right) = \int_{\mathbf{a}}q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'}V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a}$$

$$\lambda_t(\mathbf{s},\mathbf{z}) = \mu_t(\mathbf{s},\mathbf{z})\alpha\left(-1 + \log\int_{\mathbf{a}}q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'}V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a}\right)$$

Inserting λ back in π -> normalized exponential

$$
\begin{aligned}
\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) =& q_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) - \frac{\lambda_t(\mathbf{s},\mathbf{z})}{\mu_t(\mathbf{s},\mathbf{z})} + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}' - \alpha\right)\right] \\
=& \exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) - \alpha\log\int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a}\right.\right. \\
&\left.\left. + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right] q_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \\
=& q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right] \\
& \exp\left[-\log\int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a}\right] \\
=& q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\frac{\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]}{\int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)d\mathbf{a}\right]}
\end{aligned}
$$

Inserting λ in Lagrangian

$$
\begin{aligned}
& L(\mu_t, V_t, \alpha) \\
&= \sum_{\mathbf{z}}\int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z})R_T(\mathbf{s})d\mathbf{s} \\
&+ \sum_{\mathbf{z}}\int_{\mathbf{s}} V_1(\mathbf{s},\mathbf{z})p_1(\mathbf{s},\mathbf{z})d\mathbf{s} - \sum_{\mathbf{z}}\int_{\mathbf{s}} V_T(\mathbf{s},\mathbf{z})\mu_T(\mathbf{s},\mathbf{z})d\mathbf{s} \\
&- \sum_{t=1}^{T-1}\sum_{\mathbf{z}}\int_{\mathbf{s}} V_t(\mathbf{s},\mathbf{z})\mu_t(\mathbf{s},\mathbf{z})d\mathbf{s} \\
&+ \alpha\epsilon \\
&+ \sum_{t=1}^{T-1}\sum_{\mathbf{z}}\int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z})\alpha\log\int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a}d\mathbf{s}
\end{aligned}
$$

Derivatives

$$
\begin{aligned}
\frac{\partial L}{\partial \mu_t} =& - V_t(\mathbf{s},\mathbf{z}) + \alpha\log\int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s},\mathbf{z})\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a} \\
=& - V_t(\mathbf{s},\mathbf{z}) + \alpha\log\int_{\mathbf{a}}\exp\left[\frac{1}{\alpha}\left(\alpha\log q_t(\mathbf{a}|\mathbf{s},\mathbf{z}) + R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]d\mathbf{a}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial V_t} =& - \mu_t(\mathbf{s},\mathbf{z}) + \sum_{\hat{\mathbf{z}}}\int_{\hat{\mathbf{s}}}\mu_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{z}})\int_{\hat{\mathbf{a}}}\frac{q_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}},\hat{\mathbf{z}})\exp\left[\frac{1}{\alpha}\left(R_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{a}}) + \sum_{\mathbf{z}}\int_{\mathbf{s}} V_t(\mathbf{s},\mathbf{z})\mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})P_{t-1}(\mathbf{z}|\hat{\mathbf{z}})d\mathbf{s}\right)\right]}{\int_{\mathbf{a}} q_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}},\hat{\mathbf{z}})\exp\left[\frac{1}{\alpha}\left(R_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{a}}) + \sum_{\mathbf{z}}\int_{\mathbf{s}} V_t(\mathbf{s},\mathbf{z})\mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})P_{t-1}(\mathbf{z}|\hat{\mathbf{z}})d\mathbf{s}\right)\right]} \\
& \ldots\mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})P_{t-1}(\mathbf{z}|\hat{\mathbf{z}})d\hat{\mathbf{a}}d\hat{\mathbf{s}} \\
=& - \mu_t(\mathbf{s},\mathbf{z}) + \sum_{\hat{\mathbf{z}}}\int_{\hat{\mathbf{s}}}\int_{\hat{\mathbf{a}}}\pi_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}},\hat{\mathbf{z}})\mu_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{z}})\mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{z}},\hat{\mathbf{a}})P_{t-1}(\mathbf{z}|\hat{\mathbf{z}})d\hat{\mathbf{a}}d\hat{\mathbf{s}}
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial \alpha} =& \epsilon + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \log \int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \exp \left[\frac{1}{\alpha} \left(R_t(\mathbf{s}, \mathbf{a}) + \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) P_t(\mathbf{z}'|\mathbf{z}) d\mathbf{s}' \right) \right] d\mathbf{a} d\mathbf{s} \\
& - \frac{1}{\alpha} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \left(R_t(\mathbf{s}, \mathbf{a}) + \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) P_t(\mathbf{z}'|\mathbf{z}) d\mathbf{s}' \right) d\mathbf{a} d\mathbf{s} \\
=& \epsilon - \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z})}{q_t(\mathbf{a}|\mathbf{s}, \mathbf{z})} d\mathbf{a} d\mathbf{s}
\end{aligned}
$$

Optimality Conditions

$$
V_t(\mathbf{s}, \mathbf{z}) = \begin{cases} R_T(\mathbf{s}) & t = T \\ \alpha \log \int_{\mathbf{a}} \exp \left[\frac{1}{\alpha} \left(\alpha \log q_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) + R_t(\mathbf{s}, \mathbf{a}) + \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \mathcal{P}_t(\mathbf{s}', \mathbf{z}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{s}' \right) \right] d\mathbf{a} & t < T \end{cases}
$$

$$
\mu_t(\mathbf{s}, \mathbf{z}) = \begin{cases} p_1(\mathbf{s}, \mathbf{z}) & t = 1 \\ \sum_{\hat{\mathbf{z}}} \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \pi_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{z}}, \hat{\mathbf{a}}) P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} & t > 1 \end{cases}
$$

final dual formulation

$$
L(\mu_t, V_t, \alpha) = + \sum_{\mathbf{z}} \int_{\mathbf{s}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} + \alpha \epsilon
$$

A.2. LQG Assumptions

$$
\begin{aligned}
\mathcal{P}(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) =& \mathcal{N}(\mathbf{s}'|\mathbf{A}^z \mathbf{s} + \mathbf{b}^z \mathbf{a} + \mathbf{c}^z, \mathbf{\Sigma}_{s'}^z) \\
q_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) =& \mathcal{N}(\mathbf{a}|\mathbf{K}_t^{q,z} \mathbf{s} + \mathbf{k}_t^{q,z}, \mathbf{\Sigma}_{a,t}^{q,z}) \\
R_t(\mathbf{s}, \mathbf{a}) =& (\mathbf{g} - \mathbf{s})^T \mathbf{M}_t (\mathbf{g} - \mathbf{s}) + \mathbf{a}^T \mathbf{H}_t \mathbf{a} \\
V_t(\mathbf{s}, \mathbf{z}) =& \mathbf{s}^T \mathbf{V}_t^z \mathbf{s} + \mathbf{s}^T \mathbf{v}_t^z + v_t^z \\
\mu_t(\mathbf{s}, \mathbf{z}) =& \sum_i w_{t,i}(\mathbf{z}) \mathcal{N}(\mathbf{s}|\tau_{t,i}, \mathbf{\Sigma}_{t,i})
\end{aligned}
$$

Compute augmented reward

$$
\begin{aligned}
r_t(\mathbf{s}, \mathbf{a}, \mathbf{z}) =& R_t(\mathbf{s}, \mathbf{a}) + \alpha \log q_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \\
=& (\mathbf{g} - \mathbf{s})^T \mathbf{M}_t (\mathbf{g} - \mathbf{s}) + \mathbf{a}^T \mathbf{H}_t \mathbf{a} - \alpha \log \sqrt{\left| 2\pi \mathbf{\Sigma}_{a,t}^{q,z} \right|} - \frac{\alpha}{2} (\mathbf{a} - \mathbf{K}_t^{q,z} \mathbf{s} - \mathbf{k}_t^{q,z})^T (\mathbf{\Sigma}_a^{q,z})^{-1} (\mathbf{a} - \mathbf{K}_t^{q,z} \mathbf{s} - \mathbf{k}_t^{q,z}) \\
=& \mathbf{s}^T \mathbf{R}_{ss,t}^z \mathbf{s} + \mathbf{a}^T \mathbf{R}_{aa,t}^z \mathbf{a} + \mathbf{a}^T \mathbf{R}_{sa,t}^{zT} \mathbf{s} + \mathbf{s}^T \mathbf{R}_{sa,t}^z \mathbf{a} + \mathbf{s}^T \mathbf{r}_{s,t}^z + \mathbf{a}^T \mathbf{r}_{a,t}^z + r_{0,t}^z
\end{aligned}
$$

Where

$$
\begin{aligned}
\mathbf{R}^z_{ss,t} =& \mathbf{M}_t - \frac{\alpha}{2}(\mathbf{K}^{q,z}_t)^T(\mathbf{\Sigma}^{q,z}_{a,z})^{-1}\mathbf{K}^{q,z}_t \\
\mathbf{R}^z_{aa,t} =& \mathbf{H}_t - \frac{\alpha}{2}(\mathbf{\Sigma}^{q,z}_{a,t})^{-1} \\
\mathbf{R}^z_{sa,t} =& \frac{\alpha}{2}(\mathbf{K}^{q,z}_t)^T(\mathbf{\Sigma}^{q,z}_{a,t})^{-1} \\
\mathbf{r}^z_{s,t} =& -\alpha(\mathbf{K}^{q,z}_t)^T(\mathbf{\Sigma}^{q,z}_{a,t})^{-1}\mathbf{k}^{q,z}_t - 2\mathbf{M}_t\mathbf{g} \\
\mathbf{r}^z_{a,t} =& \alpha(\mathbf{\Sigma}^{q,z}_{a,t})^{-1}\mathbf{k}^{q,z}_t \\
r^z_{0,t} =& \mathbf{g}^T\mathbf{M_t}\mathbf{g} - \alpha\log\sqrt{\left|2\pi\mathbf{\Sigma}^{q,z}_{a,t}\right|} - \frac{\alpha}{2}(\mathbf{k}^{q,z}_t)^T(\mathbf{\Sigma}^{q,z}_{a,t})^{-1}\mathbf{k}^{q,z}_t
\end{aligned}
$$

Compute Q-Function

$$
\begin{aligned}
Q_t(\mathbf{s},\mathbf{a},\mathbf{z}) =& r_t(\mathbf{s},\mathbf{a},\mathbf{z}) + \mathbb{E}_{\mathcal{P}}\left[V_{t+1}(\mathbf{s}',\mathbf{z}')\right] \\
=& \mathbf{s}^T\mathbf{R}^z_{ss,t}\mathbf{s} + \mathbf{a}^T\mathbf{R}^z_{aa,t}\mathbf{a} + \mathbf{a}^T\mathbf{R}^{zT}_{sa,t}\mathbf{s} + \mathbf{s}^T\mathbf{R}^z_{sa,t}\mathbf{a} + \mathbf{s}^T\mathbf{r}^z_{s,t} + \mathbf{a}^T\mathbf{r}^z_{a,t} + r^z_{0,t} \\
&+ \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left((\mathbf{A}^z_t\mathbf{s} + \mathbf{b}^z_t\mathbf{a} + \mathbf{c}^z_t)^T\mathbf{V}^{z'}_{t+1}(\mathbf{A}^z_t\mathbf{s} + \mathbf{b}^z_t\mathbf{a} + \mathbf{c}^z_t)\right. \\
&\left.+ \operatorname{Tr}(\mathbf{V}^{z'}_{t+1}\mathbf{\Sigma}^z_{s',t}) + (\mathbf{v}^{z'}_{t+1})^T(\mathbf{A}^z_t\mathbf{s} + \mathbf{b}^z_t\mathbf{a} + \mathbf{c}^z_t) + v^{z'}_{t+1}\right) \\
=& \mathbf{s}^T\mathbf{Q}^z_{ss,t}\mathbf{s} + \mathbf{a}^T\mathbf{Q}^z_{aa,t}\mathbf{a} + \mathbf{a}^T\mathbf{Q}^{zT}_{sa,t}\mathbf{s} + \mathbf{s}^T\mathbf{Q}^z_{sa,t}\mathbf{a} + \mathbf{s}^T\mathbf{Q}^z_{s,t} + \mathbf{a}^T\mathbf{Q}^z_{a,t} + Q^z_{0,t}
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{Q}^z_{ss,t} =& \mathbf{R}^z_{ss,t} + \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left(\mathbf{A}^{zT}\mathbf{V}^{z'}_{t+1}\mathbf{A}^z\right) \\
\mathbf{Q}^z_{aa,t} =& \mathbf{R}^z_{aa,t} + \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left(\mathbf{b}^{zT}\mathbf{V}^{z'}_{t+1}\mathbf{b}^z\right) \\
\mathbf{Q}^z_{sa,t} =& \mathbf{R}^z_{sa,t} + \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left(\mathbf{A}^{zT}\mathbf{V}^{z'}_{t+1}\mathbf{b}^z\right) \\
\mathbf{Q}^z_{s,t} =& \mathbf{r}^z_{s,t} + \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left(2\mathbf{A}^{zT}\mathbf{V}^{z'}_{t+1}\mathbf{c}^z + \mathbf{A}^{zT}\mathbf{v}^{z'}_{t+1}\right) \\
\mathbf{Q}^z_{a,t} =& \mathbf{r}^z_{a,t} + \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left(2\mathbf{b}^{zT}\mathbf{V}^{z'}_{t+1}\mathbf{c}^z + \mathbf{b}^{zT}\mathbf{v}^{z'}_{t+1}\right) \\
Q^z_{0,t} =& r^z_{0,t} + \sum_{\mathbf{z}'} P(\mathbf{z}'|\mathbf{z})\left(v^{z'}_{t+1} + \operatorname{Tr}(\mathbf{V}^{z'}_{t+1}\mathbf{\Sigma}^z_{s'}) + \mathbf{v}^{z'}_{t+1}\mathbf{c}^z + \mathbf{c}^{zT}\mathbf{V}^{z'}_{t+1}\mathbf{c}^z\right)
\end{aligned}
$$

$$
\mathbf{x} = \begin{pmatrix}\mathbf{s}\\ \mathbf{a}\end{pmatrix} \qquad \mathbf{W}^z_t = \begin{pmatrix}-2\mathbf{Q}^z_{ss,t} & -2\mathbf{Q}^z_{sa,t}\\ -2\mathbf{Q}^{zT}_{sa,t} & -2\mathbf{Q}^z_{aa,t}\end{pmatrix} \qquad \mathbf{w}^z_t = \begin{pmatrix}\mathbf{Q}^z_{s,t}\\ \mathbf{Q}^z_{a,t}\end{pmatrix} \qquad w^z_t = Q^z_{0,t}
$$

$$
Q_t(\mathbf{s},\mathbf{a},\mathbf{z}) = -\frac{1}{2}\mathbf{x}^T\mathbf{W}^z_t\mathbf{x} + \mathbf{x}^T\mathbf{w}^z_t + w^z_t
$$

$$\exp\left[\frac{Q_t(\mathbf{s},\mathbf{z},\mathbf{a})}{\alpha}\right] = \exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{x}^T\mathbf{W}_t^z\mathbf{x} + \mathbf{x}^T\mathbf{w}_t^z + w_t^z\right)\right]$$
$$= \mathcal{N}\left[\mathbf{x}|\frac{\mathbf{w}_t^z}{\alpha}, \frac{\mathbf{W}_t^z}{\alpha}\right] \frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]}$$

Policy

$$\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})$$
$$= q_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \frac{\exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right)\right]}{\int_{\mathbf{a}} q_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \exp\left[\frac{1}{\alpha}\left(R_t(\mathbf{s},\mathbf{a}) + \sum_{\mathbf{z}'}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}')\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})P_t(\mathbf{z}'|\mathbf{z})d\mathbf{s}'\right) d\mathbf{a}\right]}$$
$$= \frac{\exp\left[\frac{1}{\alpha}Q_t(\mathbf{s},\mathbf{a},\mathbf{z})\right]}{\int_{\mathbf{a}} \exp\left[\frac{1}{\alpha}Q_t(\mathbf{s},\mathbf{a},\mathbf{z}\right] d\mathbf{a}}$$
$$= \frac{\mathcal{N}\left[x|\frac{\mathbf{w}_t^z}{\alpha}, \frac{\mathbf{W}_t^z}{\alpha}\right] \frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]}}{\int_{\mathbf{a}} \mathcal{N}\left[x|\frac{\mathbf{w}_t^z}{\alpha}, \frac{\mathbf{W}_t^z}{\alpha}\right] \frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]} d\mathbf{a}}$$
$$= \frac{\mathcal{N}\left[\mathbf{s}|\frac{1}{\alpha}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z), -2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,T}\right] \mathcal{N}\left[\mathbf{a}|\frac{1}{\alpha}(\mathbf{Q}_{a,t}^z + 2\mathbf{Q}_{sa,t}^{z,T}\mathbf{s}), \frac{-2\mathbf{Q}_{aa,t}^z}{\alpha}\right]}{\int_{\mathbf{a}} \mathcal{N}\left[\mathbf{s}|\frac{1}{\alpha}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z), -2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,T}\right] \mathcal{N}\left[\mathbf{a}|\frac{1}{\alpha}(\mathbf{Q}_{a,t}^z + 2\mathbf{Q}_{sa,t}^{z,T}\mathbf{s}), \frac{-2\mathbf{Q}_{aa,t}^z}{\alpha}\right] d\mathbf{a}}$$
$$= \mathcal{N}(\mathbf{a}| - \frac{1}{2}(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z - (\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{zT}\mathbf{s}, -\frac{\alpha}{2}(\mathbf{Q}_{aa,t}^z)^{-1})$$
$$= \mathcal{N}(\mathbf{a}|\mathbf{k}_t^{\pi,z} + \mathbf{K}_t^{\pi,z}\mathbf{s}, \mathbf{\Sigma}_{a,t}^{\pi,z})$$

Compute V-Function

$$
\begin{aligned}
& V_t(\mathbf{s}, \mathbf{z}) \\
=& \alpha \log \int_{\mathbf{a}} \exp\left[\frac{1}{\alpha} Q_t(\mathbf{s}, \mathbf{z}, \mathbf{a})\right] d\mathbf{a} \\
=& \alpha \log \int_{\mathbf{a}} \mathcal{N}\left[x | \frac{\mathbf{w}_t^z}{\alpha}, \frac{\mathbf{W}_t^z}{\alpha}\right] \frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]} d\mathbf{a} \\
=& \alpha \log \left[\frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]}\right. \\
& \left. \int_{\mathbf{a}} \mathcal{N}\left[\mathbf{s} | \frac{1}{\alpha}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z), -2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,T}\right] \mathcal{N}\left[\mathbf{a} | \frac{1}{\alpha}(\mathbf{Q}_{a,t}^z + 2\mathbf{Q}_{sa,t}^{z,T}\mathbf{s}), \frac{-2\mathbf{Q}_{aa,t}^z}{\alpha}\right] d\mathbf{a}\right] \\
=& \alpha \log \left[\frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]} \mathcal{N}\left[\mathbf{s} | \frac{1}{\alpha}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z), -2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,T}\right]\right] \\
=& \alpha \log \left[\frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]}\right. \\
& \left. \mathcal{N}\left(\mathbf{s} | \left(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,T}\right)^{-1}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z), \alpha\left(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,T}\right)^{-1}\right)\right] \\
=& \alpha \log \left[\frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]}\right] \\
& - \alpha \log \sqrt{\left|2\pi\alpha\left(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^z\right)^{-1}\right|} \\
& - \mathbf{s}^T(-\mathbf{Q}_{ss,t}^z + \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^z)\mathbf{s} + \mathbf{s}^T(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z) \\
& - \frac{1}{2}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z)^T(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^z)^{-1}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z) \\
=& \mathbf{s}^T\mathbf{V}_t^z\mathbf{s} + \mathbf{s}^T\mathbf{v}_t^z + v_t^z
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{V}_t^z =& (\mathbf{Q}_{ss,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{zT}) \\
\mathbf{v}_t^z =& (\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z)
\end{aligned}
$$

$$
\begin{aligned}
v_t^z =& \alpha \log \left[\frac{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}}{\exp\left[\frac{1}{\alpha}\left(-\frac{1}{2}\mathbf{w}_t^z(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z - w_t^z\right)\right]} \right] - \alpha \log \sqrt{\left|2\pi\alpha\left(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^z\right)^{-1}\right|} \\
& - \frac{1}{2}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z)^T(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^z)^{-1}(\mathbf{Q}_{s,t}^z - \mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z) \\
=& -\alpha \log \left[\frac{\sqrt{\left|2\pi\alpha\left(-2\mathbf{Q}_{ss,t}^z + 2\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^z\right)^{-1}\right|}}{\sqrt{|2\pi\alpha(\mathbf{W}_t^z)^{-1}|}} \right] \\
& + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + \frac{1}{2}\mathbf{w}_t^{z,T}(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z + w_t^z \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + \frac{1}{2}\mathbf{w}_t^{z,T}(\mathbf{W}_t^z)^{-1}\mathbf{w}_t^z + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) \\
& + \frac{1}{2}\begin{pmatrix}\mathbf{Q}_{s,t}^z \\ \mathbf{Q}_{a,t}^z\end{pmatrix}^T \begin{pmatrix}-2\mathbf{Q}_{ss,t}^z & -2\mathbf{Q}_{sa,t}^z \\ -2\mathbf{Q}_{sa,t}^{zT} & -2\mathbf{Q}_{aa,t}^z\end{pmatrix}^{-1} \begin{pmatrix}\mathbf{Q}_{s,t}^z \\ \mathbf{Q}_{a,t}^z\end{pmatrix} \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) \\
& + \frac{1}{2}\begin{pmatrix}\mathbf{Q}_{s,t}^z \\ \mathbf{Q}_{a,t}^z\end{pmatrix}^T \begin{pmatrix}-\frac{1}{2}(\mathbf{V}_{ss,t}^z)^{-1} & \frac{1}{2}(\mathbf{V}_{ss,t}^z)^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1} \\ \frac{1}{2}(\mathbf{Q}_{aa,t}^{z,T})^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{V}_{ss,t}^z)^{-1} & -\frac{1}{2}(\mathbf{Q}_{aa,t}^z)^{-1} - \frac{1}{2}(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,t}(\mathbf{V}_{ss,t}^z)^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\end{pmatrix}\begin{pmatrix}\mathbf{Q}_{s,t}^z \\ \mathbf{Q}_{a,t}^z\end{pmatrix} \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) \\
& - \frac{1}{4}\mathbf{Q}_{s,t}^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{Q}_{s,t}^z + \frac{1}{4}\mathbf{Q}_{s,t}^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z \\
& + \frac{1}{4}\mathbf{Q}_{a,t}^{z,T}(\mathbf{Q}_{aa,t}^{z,T})^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{V}_t^z)^{-1}\mathbf{Q}_{s,t}^z - \frac{1}{4}\mathbf{Q}_{a,t}^{z,T}((\mathbf{Q}_{aa,t}^z)^{-1} + (\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{sa,t}^{z,t}(\mathbf{V}_t^z)^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1})\mathbf{Q}_{a,t}^z \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) \\
& - \frac{1}{4}\mathbf{Q}_{a,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z - \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{Q}_{s,t}^z + \frac{1}{4}(\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{Q}_{sa,t}^z(\mathbf{Q}_{aa,t}^z)^{-1})\mathbf{Q}_{a,t}^z \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^z + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) - \frac{1}{4}\mathbf{Q}_{a,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z - \frac{1}{4}\mathbf{v}_t^{z,T}(\mathbf{V}_t^z)^{-1}\mathbf{v}_t^{z,T} \\
=& -\frac{\alpha}{2}\log\left[\frac{\left|(-2\mathbf{V}_t^z)^{-1}\right|}{|(\mathbf{W}_t^z)^{-1}|}\right] + w_t^z + \frac{\alpha N_a}{2}\log(2\pi\alpha) - \frac{1}{4}\mathbf{Q}_{a,t}^z(\mathbf{Q}_{aa,t}^z)^{-1}\mathbf{Q}_{a,t}^z
\end{aligned}
$$

alpha update

$$
\begin{aligned}
&\sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) D_{\mathrm{KL}}(\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})||q_t(\mathbf{a}|\mathbf{s},\mathbf{z}))d\mathbf{s} \\
=&\sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} d\mathbf{a} d\mathbf{s} \\
=&\sum_{\mathbf{z}} \int_{\mathbf{s}} \sum_i w_{t,i}(\mathbf{z}) \mathcal{N}(\mathbf{s}|\tau_{t,i}, \mathbf{\Sigma}_{t,i}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \log \frac{\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})}{q_t(\mathbf{a}|\mathbf{s},\mathbf{z})} d\mathbf{a} d\mathbf{s} \\
=&\sum_{\mathbf{z}} \int_{\mathbf{s}} \sum_i w_{t,i}(\mathbf{z}) \mathcal{N}(\mathbf{s}|\tau_{t,i}, \mathbf{\Sigma}_{t,i}) \left(\frac{1}{2} \log \frac{|\mathbf{\Sigma}_{a,t}^{q,z}|}{|\mathbf{\Sigma}_{a,t}^{\pi,z}|} + \frac{1}{2} \operatorname{Tr}\left((\mathbf{\Sigma}_{a,t}^{q,z})^{-1} \mathbf{\Sigma}_{a,t}^{\pi,z} \right) - \frac{1}{2} N_a \right. \\
&\left. + \frac{1}{2} ((\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z})\mathbf{s} + (\mathbf{k}_t^{q,z} - \mathbf{k}_t^{\pi,z}))^T (\mathbf{\Sigma}_{a,t}^{q,z})^{-1} ((\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z})\mathbf{s} + (\mathbf{k}_t^{q,z} - \mathbf{k}_t^{\pi,z})) \right) d\mathbf{s} \\
=&\sum_{\mathbf{z}} w_{t,i}(\mathbf{z}) \left(\frac{1}{2} \log \frac{|\mathbf{\Sigma}_{a,t}^{q,z}|}{|\mathbf{\Sigma}_{a,t}^{\pi,z}|} + \frac{1}{2} \operatorname{Tr}\left((\mathbf{\Sigma}_{a,t}^{q,z})^{-1} \mathbf{\Sigma}_{a,t}^{\pi,z} \right) \right. \\
&- \frac{1}{2} N_a + \frac{1}{2} \operatorname{Tr}\left((\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z})^T (\mathbf{\Sigma}_{a,t}^{q,z})^{-1} (\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z}) \mathbf{\Sigma}_{t,i} \right) \\
&+ \frac{1}{2} (\tau_{t,i})^T (\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z})^T (\mathbf{\Sigma}_{a,t}^{q,z})^{-1} (\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z}) (\tau_{t,i}) \\
&- (\tau_{t,i})^T (\mathbf{K}_t^{q,z} - \mathbf{K}_t^{\pi,z})^T (\mathbf{\Sigma}_{a,t}^{q,z})^{-1} (-\mathbf{k}_t^{q,z} + \mathbf{k}_t^{\pi,z}) \\
&\left. + \frac{1}{2} (-\mathbf{k}_t^{q,z} + \mathbf{k}_t^{\pi,z})^T (\mathbf{\Sigma}_{a,t}^{q,z})^{-1} (-\mathbf{k}_t^{q,z} - + \mathbf{k}_t^{\pi,z}) \right)
\end{aligned}
$$

state update

$$
\begin{aligned}
&\mu_t(\mathbf{s},\mathbf{z}) \\
=&\sum_{\hat{\mathbf{z}}} \int_{\hat{\mathbf{s}}} \mu_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{z}}) \int_{\hat{\mathbf{a}}} \pi_{t-1}(\mathbf{a}|\hat{\mathbf{s}},\hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}}) P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} \\
=&\sum_{\hat{\mathbf{z}}} P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}} \mu_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{z}}) \int_{\hat{\mathbf{a}}} \mathcal{N}(\mathbf{s}|\mathbf{A}_{t-1}^{\hat{z}} \hat{\mathbf{s}} + \mathbf{b}_{t-1}^{\hat{z}} \hat{\mathbf{a}} + \mathbf{c}_{t+1}^{\hat{z}}, \mathbf{\Sigma}_{s',t-1}^{\hat{z}}) \mathcal{N}(\hat{\mathbf{a}}_{t-1}|\mathbf{k}_{t-1}^{\pi,\hat{z}} + \mathbf{K}_{t-1}^{\pi,\hat{z}} \mathbf{s}, \mathbf{\Sigma}_{a,t-1}^{\pi,\hat{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} \\
=&\sum_{\hat{\mathbf{z}}} P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}} \mu_{t-1}(\mathbf{s},\hat{\mathbf{z}}) \mathcal{N}(\mathbf{s}|\mathbf{A}_{t-1}^{\hat{z}} \hat{\mathbf{s}} + \mathbf{b}_{t-1}^{\hat{z}} (\mathbf{k}_{t-1}^{\pi,\hat{z}} + \mathbf{K}_{t-1}^{\pi,\hat{z}} \mathbf{s}) + \mathbf{c}_{t+1}^{\hat{z}}, \mathbf{\Sigma}_{s',t-1}^{\hat{z}} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{\Sigma}_{a,t-1}^{\pi,\hat{z}} \mathbf{b}_{t-1}^{\hat{z},T}) d\hat{\mathbf{s}} \\
=&\sum_{\hat{\mathbf{z}}} P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \sum_i w_{t-1,i}(\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}} \mathcal{N}(\hat{\mathbf{s}}|\tau_{t-1,i}, \mathbf{\Sigma}_{t-1,i}) \\
&\mathcal{N}(\mathbf{s}|\mathbf{A}_{t-1}^{\hat{z}} \hat{\mathbf{s}} + \mathbf{b}_{t-1}^{\hat{z}} (\mathbf{k}_{t-1}^{\pi,\hat{z}} + \mathbf{K}_{t-1}^{\pi,\hat{z}} \mathbf{s}) + \mathbf{c}_{t+1}^{\hat{z}}, \mathbf{\Sigma}_{s',t-1}^{\hat{z}} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{\Sigma}_{a,t-1}^{\pi,\hat{z}} {\mathbf{b}_{t-1}^{\hat{z}}}^T) d\hat{\mathbf{s}} \\
=&\sum_{\hat{\mathbf{z}}} P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \sum_i w_{t-1,i}(\hat{\mathbf{z}}) \mathcal{N}\left(\mathbf{s}|\mathbf{c}_{t-1}^{\hat{z}} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{k}_{t-1}^{\pi,\hat{z}} + (\mathbf{A}_{t-1}^{\hat{z}} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{K}_{t-1}^{\pi,\hat{z}}) \tau_{t-1,i}, \right. \\
&\left. \mathbf{\Sigma}_{s,t-1} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{\Sigma}_{a,t-1}^{\pi,\hat{z}} \mathbf{b}_{t-1}^{\hat{z}T} + (\mathbf{A}_{t-1}^{\hat{z}} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{K}_{t-1}^{\pi,\hat{z}}) \mathbf{\Sigma}_{t-1,i} (\mathbf{A}_{t-1}^{\hat{z}} + \mathbf{b}_{t-1}^{\hat{z}} \mathbf{K}_{t-1}^{\pi,\hat{z}})^T \right)
\end{aligned}
$$

dual

$$
\begin{aligned}
L(\mu_t, V_t, \alpha) &= \sum_{\mathbf{z}} \int_{\mathbf{s}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} + \alpha\epsilon \\
&= \sum_{\mathbf{z}} \sum_{i} w_{1,i}(\mathbf{z}) \int_{\mathbf{s}} \mathcal{N}(\mathbf{s}|\tau_{1,i}, \mathbf{\Sigma}_{1,i})(\mathbf{s}^T \mathbf{V}_1^z \mathbf{s} + \mathbf{s}^T \mathbf{v}_1^z + v_1^z) d\mathbf{s} + \alpha\epsilon \\
&= \sum_{\mathbf{z}} \sum_{i} w_{1,i} \left((\tau_{1,i})^T \mathbf{V}_1^z (\tau_{1,i}) + (\tau_{1,i})^t \mathbf{v}_1^z + v_1^z + \mathrm{Tr}(\mathbf{V}_1^z \mathbf{\Sigma}_{1,i}) \right) + \alpha\epsilon
\end{aligned}
$$

B. Robustify Gaussians

$$\begin{aligned} \max & \int_{\mathbf{s}} C(\mathbf{s}) d\mathbf{s} - \alpha \int_{\mathbf{s}} q(\mathbf{s}) \log \frac{q(\mathbf{s})}{\mu(\mathbf{s})} d\mathbf{s} \\ \text{s.t.} & \int_{\mathbf{s}} q(\mathbf{s}) d\mathbf{s} = 1 \end{aligned}$$

$$L = \int_{\mathbf{s}} C(\mathbf{s}) q(\mathbf{s}) d\mathbf{s} - \alpha \int_{\mathbf{s}} q(\mathbf{s}) \log \frac{q(\mathbf{s})}{\mu(\mathbf{s})} d\mathbf{s} + \lambda (1 - \int_{\mathbf{s}} q(\mathbf{s}) d\mathbf{s})$$

$$\frac{\partial L}{\partial q} = \int_{\mathbf{a}} C(\mathbf{s}) - \alpha (\log(\frac{q(\mathbf{s})}{\mu(\mathbf{s})}) + 1) - \lambda = 0$$

$$q(\mathbf{s}) = \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} \left[C(\mathbf{s}) - \lambda - \alpha\right]\right)$$

$\frac{\partial L}{\partial \lambda} = 0$:

$$\begin{aligned} 1 &= \int_{\mathbf{s}} q(\mathbf{s}) d\mathbf{s} \\ 1 &= \int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} \left[C(\mathbf{s}) - \lambda - \alpha\right]\right) d\mathbf{s} \\ \exp(\frac{\lambda}{\alpha} + 1) &= \int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right) d\mathbf{s} \\ \lambda &= \alpha \left(-1 + \log\left(\int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right) d\mathbf{s}\right)\right) \end{aligned}$$

plug λ back in q:

$$\begin{aligned} q(\mathbf{s}) &= \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} \left[C(\mathbf{s}) - \alpha \left(-1 + \log\left(\int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right)\right)\right) - \alpha\right]\right) \\ q(\mathbf{s}) &= \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right) \exp\left(-\log\left(\int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right)\right)\right) \\ q(\mathbf{s}) &= \frac{\mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right)}{\int_{\mathbf{s}} \mu(\mathbf{s}) \exp\left(\frac{1}{\alpha} C(\mathbf{s})\right)} \end{aligned}$$

For Mixture of gaussians:

$$
\begin{aligned}
q(\mathbf{s}) &= \frac{\sum_i w_i \mu_i(\mathbf{s}) \exp(\frac{1}{\alpha} C(\mathbf{s}))}{\int_s \sum_i w_i \mu_i(\mathbf{s}) \exp(\frac{1}{\alpha} C(\mathbf{s})) d\mathbf{s}} \propto \sum_i w_i \exp\left[\frac{1}{\alpha}(C(\mathbf{s}) + \alpha \log \mu_i(\mathbf{s})\right] \\
&= \sum_i w_i \exp\left[\frac{1}{\alpha}(C(\mathbf{s}) + \alpha \log \mathcal{N}(\mathbf{s}|\tau_i, \mathbf{\Sigma}_i)\right] \\
&= \sum_i w_i \exp\left[\frac{1}{\alpha}\left(\mathbf{s}^T \hat{\mathbf{C}} \mathbf{s} + \mathbf{s}^T \hat{\mathbf{c}} + \hat{c} + \alpha\left(-\frac{1}{2}\log|2\pi\mathbf{\Sigma}_i| - \frac{1}{2}(\mathbf{s}-\tau_i)^T \mathbf{\Sigma}_i^{-1}(\mathbf{s}-\tau_i)\right)\right)\right] \\
&= \sum_i w_i \exp\left[-\frac{1}{2}\mathbf{s}^T \underbrace{(\mathbf{\Sigma}_i^{-1} - \frac{2}{\alpha}\hat{\mathbf{C}})}_{\mathbf{A}_i} \mathbf{s} + \mathbf{s}^T \underbrace{(\frac{\hat{\mathbf{c}}}{\alpha} + \mathbf{\Sigma}_i^{-1}\tau_i)}_{\mathbf{a}_i} + \frac{\hat{c}}{\alpha} - \frac{1}{2}\log|2\pi\mathbf{\Sigma}_i| - \frac{1}{2}\tau_i^T \mathbf{\Sigma}_i^{-1}\tau_i\right] \\
&= \sum_i w_i \mathcal{N}[\mathbf{s}|\mathbf{a}_i, \mathbf{A}_i] \left|2\pi\mathbf{A}_i^{-1}\right|^{\frac{1}{2}} \exp\left(\frac{1}{2}\mathbf{a}_i^T \mathbf{A}_i^{-1}\mathbf{a}_i\right) \exp\left(\frac{\hat{c}}{\alpha}\right) \exp\left(-\frac{1}{2}\tau_i \mathbf{\Sigma}_i^{-1}\tau_i\right) \frac{1}{|2\pi\mathbf{\Sigma}_i|^{\frac{1}{2}}} \\
&= \sum_i w_i \mathcal{N}(\mathbf{s}|\mathbf{A}_i^{-1}\mathbf{a}_i, \mathbf{A}_i^{-1}) \left|2\pi\mathbf{A}_i^{-1}\right|^{\frac{1}{2}} \exp\left(\frac{1}{2}\mathbf{a}_i^T \mathbf{A}_i^{-1}\mathbf{a}_i\right) \exp\left(\frac{\hat{c}}{\alpha}\right) \exp\left(-\frac{1}{2}\tau_i \mathbf{\Sigma}_i^{-1}\tau_i\right) \frac{1}{|2\pi\mathbf{\Sigma}_i|^{\frac{1}{2}}} \\
&= \sum_i v_i \mathcal{N}(\mathbf{s}|\mathbf{A}_i^{-1}\mathbf{a}_i, \mathbf{A}_i^{-1})
\end{aligned}
$$

C. Robustify Weights

$$\begin{aligned}
\max & \int_{\mathbf{s}} C(\mathbf{s}) \sum_i v_i \mu_i(\mathbf{s}) d\mathbf{s} \\
\text{s.t.} & \sum_i v_i = 1 \\
& \sum_i v_i \log\left(\frac{v_i}{w_i}\right) d\mathbf{s} \leq \epsilon
\end{aligned}$$

$$L = \int_{\mathbf{s}} C(\mathbf{s}) \sum_i v_i \mu_i(\mathbf{s}) d\mathbf{s} - \alpha \left(\sum_i v_i \log\left(\frac{v_i}{w_i}\right) d\mathbf{s} - \epsilon \right) - \lambda \left(\sum_i v_i - 1 \right)$$

$$\frac{\partial L}{\partial v_i} = \int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s} - \alpha \left(\log\left(\frac{v_i}{w_i}\right) + 1 \right) - \lambda = 0$$

$$v_i = w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s} - \lambda - \alpha\right)\right]$$

$$\begin{aligned}
1 &= \sum_i v_i \\
1 &= \sum_i w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s} - \lambda - \alpha\right)\right] \\
1 &= \sum_i w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s}\right)\right] \exp\left(\frac{-\lambda - \alpha}{\alpha}\right) \\
\exp\left(\frac{\lambda}{\alpha} + 1\right) &= \sum_i w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s}\right)\right] \\
\lambda &= \alpha(-1 + \log\left(\sum_i w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s}\right)\right]\right)
\end{aligned}$$

$$v_i = \frac{w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s}\right)\right]}{\sum_i w_i \exp\left[\frac{1}{\alpha}\left(\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s}\right)\right]}$$

$$\int_{\mathbf{s}} C(\mathbf{s}) \mu_i(\mathbf{s}) d\mathbf{s} = \int_s (\mathbf{s^T C s} + \mathbf{s^T} + c) \mathcal{N}\left(\mathbf{s} | \tau_i, \mathbf{\Sigma}_i\right) d\mathbf{s} = \tau_i^T \mathbf{C} \tau_i + \tau_i \mathbf{c} + c + tr(\mathbf{C \Sigma}_i)$$

D. Derivation of Worst-Case Distribution

D.1. Optimization Problem

$$
\begin{aligned}
\min_{D_t(\mathbf{z}'|\mathbf{z},\mathbf{z})} & \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s},\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
\text{s.t.} & \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_{t-1}(\mathbf{s},\mathbf{z}) \pi_{t-1}(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_{t-1}(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) D_{t-1}(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} = \mu_t(\mathbf{s}',\mathbf{z}') \qquad \forall \mathbf{s}' \forall \mathbf{z}', \forall t > 1 \\
& \mu_1(\mathbf{s},\mathbf{z}) = p_1(\mathbf{s},\mathbf{z}) \qquad \forall \mathbf{s}, \forall \mathbf{z} \\
& \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) = 1 \qquad , \forall \mathbf{z}, \forall t < T \\
& \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \leq \delta
\end{aligned}
$$

Formulate Lagrangian:

$$
\begin{aligned}
L(\mu_t, V_t, \beta_t, \alpha, D_t) = & \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}.\mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
& + \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s},\mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) D_t(\mathbf{z}'|\mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \\
& - \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}',\mathbf{z}') \mu_t(\mathbf{s}',\mathbf{z}') d\mathbf{s}' \\
& + \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s},\mathbf{z}) p_1(\mathbf{s},\mathbf{z}) d\mathbf{s} - \int_{\mathbf{s}} \sum_{\mathbf{z}} V_T(\mathbf{s},\mathbf{z}) \mu_T(\mathbf{s},\mathbf{z}) d\mathbf{s} \\
& + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \beta_t(\mathbf{z}) \left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) - 1 \right) \\
& + \alpha \left(\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} - \delta \right)
\end{aligned}
$$

Solve Primal problem:

$$\frac{\partial L}{\partial D_t(\mathbf{z}'|\mathbf{z})} = \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' + \beta_t(\mathbf{z}) + \alpha \left(\log \left(\frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) + 1 \right) = 0$$

$$D_t(\mathbf{z}'|\mathbf{z}) = P_t(\mathbf{z}'|\mathbf{z}) \exp \left[\frac{-1}{\alpha} \left(\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' + \beta_t(\mathbf{z}) + \alpha \right) \right]$$

Insert $D_t(\mathbf{z}'|\mathbf{z})$ back in Lagrangian to obtain dual:

$$\begin{aligned} L(\mu_t, V_t, \beta_t, \alpha) = & \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s}, \mathbf{a}) \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}, \mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\ & - \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}', \mathbf{z}') \mu_t(\mathbf{s}', \mathbf{z}') d\mathbf{s}' \\ & + \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} - \int_{\mathbf{s}} \sum_{\mathbf{z}} V_T(\mathbf{s}, \mathbf{z}) \mu_T(\mathbf{s}, \mathbf{z}) d\mathbf{s} \\ & + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \beta_t(\mathbf{z})(-1) \\ & + (\sum_{t=1}^{T-1} \sum_{\mathbf{z}} -\alpha) - \alpha\delta \end{aligned}$$

Calculate β:

$$\begin{aligned} 1 = & \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \\ 1 = & \sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left[\frac{-1}{\alpha} \left(\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' + \beta_t(\mathbf{z}) + \alpha \right) \right] \\ \exp(\frac{\beta}{\alpha} + 1) = & \sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left[\frac{-1}{\alpha} \left(\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right) \right] \\ \beta_t(\mathbf{z}) = & \alpha \left(-1 + \log \left(\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left[\frac{-1}{\alpha} \left(\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right) \right] \right) \right) \end{aligned}$$

Insert β back in Lagrangian:

$$
\begin{aligned}
& L(\mu_t, V_t, \alpha) \\
=& \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \int_{\mathbf{a}} R_t(\mathbf{s}, \mathbf{a}) \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) d\mathbf{a} d\mathbf{s} + \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_T(\mathbf{s}, \mathbf{z}) R_T(\mathbf{s}) d\mathbf{s} \\
& - \sum_{t=1}^{T-1} \sum_{\mathbf{z}'} \int_{\mathbf{s}'} V_t(\mathbf{s}', \mathbf{z}') \mu_t(\mathbf{s}', \mathbf{z}') d\mathbf{s}' \\
& + \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} - \int_{\mathbf{s}} \sum_{\mathbf{z}} V_T(\mathbf{s}, \mathbf{z}) \mu_T(\mathbf{s}, \mathbf{z}) d\mathbf{s} \\
& + \sum_{t=1}^{T-1} \sum_{z} -\alpha (\log \sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right] \right) - \alpha \delta
\end{aligned}
$$

Derivation of partial derivatives:

$$
\begin{aligned}
\frac{\partial L}{\partial \alpha} =& - \delta + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} - \log \left(\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right] \right) \right) \\
& - \alpha (\sum_{\mathbf{z}'} \frac{P_t(\mathbf{z}'|\mathbf{z}) \exp \left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right] \right)}{\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right] \right)} \\
& \frac{1}{\alpha^2} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}') \\
=& - \delta + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} - \log \left(\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right] \right) \right) \\
& - \frac{1}{\alpha} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \\
=& \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{\exp \left(-\frac{1}{\alpha} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right)}{\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp \left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \right] \right)} \right) \right) \\
=& \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) \right) - \delta
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial V_t(\mathbf{s}, \mathbf{z})} =& - \mu_t(\mathbf{s}, \mathbf{z}) - \sum_{\hat{\mathbf{z}}} \alpha \left(\frac{P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \exp \left(-\frac{1}{\alpha} \int_{\mathbf{s}} V_t(\mathbf{s}, \mathbf{z}) \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \pi_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} d\mathbf{s} \right)}{\sum_{\hat{\mathbf{z}}} P_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \exp \left(-\frac{1}{\alpha} \int_{\mathbf{s}} V_t(\mathbf{s}, \mathbf{z}) \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \pi_{t-1}(\hat{\mathbf{a}}|\hat{\mathbf{s}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} d\mathbf{s} \right)} \right. \\
& \left. ... \frac{-1}{\alpha} \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \pi_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} \right) \\
=& - \mu_t(\mathbf{s}, \mathbf{z}) + \sum_{\hat{\mathbf{z}}} \left(D_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}} \int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}}, \hat{\mathbf{z}}) \pi_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) \mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}}, \hat{\mathbf{a}}, \hat{\mathbf{z}}) d\hat{\mathbf{a}} d\hat{\mathbf{s}} \right)
\end{aligned}
$$

$$
\begin{aligned}
\frac{\partial L}{\partial \mu_t(\mathbf{s},\mathbf{z})} &= \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a})\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})d\mathbf{a} - V_t(\mathbf{s},\mathbf{z}) \\
&\quad - \alpha(\sum_{\mathbf{z}'} \frac{P_t(\mathbf{z}'|\mathbf{z}) \exp\left(\frac{-1}{\alpha}\left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \int_{\mathbf{s}}\int_{\mathbf{a}} \mu_t(\mathbf{s},\mathbf{z})\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})d\mathbf{a}d\mathbf{s}d\mathbf{s}'\right]\right)}{\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left(\frac{-1}{\alpha}\left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \int_{\mathbf{s}}\int_{\mathbf{a}} \mu_t(\mathbf{s},\mathbf{z})\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})d\mathbf{a}d\mathbf{s}d\mathbf{s}'\right]\right)} \\
&\quad \frac{-1}{\alpha}\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})d\mathbf{a}d\mathbf{s}') \\
&= \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a})\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})d\mathbf{a} - V_t(\mathbf{s},\mathbf{z}) + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})\mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z})d\mathbf{a}d\mathbf{s}'
\end{aligned}
$$

Derivatives (Results only):

$$
\begin{aligned}
\frac{\partial L}{\partial \mu_t} &= \begin{cases} R_T(\mathbf{s},\mathbf{a}) - V_T(\mathbf{s},\mathbf{z}) & ,t = T \\ \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a})\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})d\mathbf{a} - V_t(\mathbf{s},\mathbf{z}) + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') & ,t < T \end{cases} \\
\frac{\partial L}{\partial V_t} &= \begin{cases} -\mu_1(\mathbf{s},\mathbf{z}) + p_1(\mathbf{s},\mathbf{z}) & ,t = 1 \\ -\mu_t(\mathbf{s},\mathbf{z}) + \sum_{\hat{\mathbf{z}}} D_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}}\int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{z}})\pi_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})\mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})d\hat{\mathbf{a}}d\hat{\mathbf{s}} & ,t > 1 \end{cases} \\
\frac{\partial L}{\partial \alpha} &= \sum_{t=1}^{T-1}\sum_{\mathbf{z}}\left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log\left(\frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})}\right) - \delta\right).
\end{aligned}
$$

Optimality conditions:

$$
\begin{aligned}
\mu_t(\mathbf{s},\mathbf{z}) &= \begin{cases} p_1(\mathbf{s},\mathbf{z}) & ,t = 1 \\ \sum_{\hat{\mathbf{z}}} D_{t-1}(\mathbf{z}|\hat{\mathbf{z}}) \int_{\hat{\mathbf{s}}}\int_{\hat{\mathbf{a}}} \mu_{t-1}(\hat{\mathbf{s}},\hat{\mathbf{z}})\pi_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})\mathcal{P}_{t-1}(\mathbf{s}|\hat{\mathbf{s}},\hat{\mathbf{a}},\hat{\mathbf{z}})d\hat{\mathbf{a}}d\hat{\mathbf{s}} & ,t > 1 \end{cases} \\
V_t(\mathbf{s},\mathbf{z}) &= \begin{cases} R_T(\mathbf{s},\mathbf{a}) & ,t = T \\ \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a})\pi_t(\mathbf{a}|\mathbf{s},\mathbf{z})d\mathbf{a} + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') & ,t < T \end{cases}.
\end{aligned}
$$

Put back in Lagrangian

$$
\begin{aligned}
& L(\mu_t, V_t, \beta_t, \alpha, D_t) \\
=& -\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s}' d\mathbf{s} \\
& + \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} \\
& - \alpha(\delta + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \log \sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}'\right]\right) \\
=& \alpha \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \frac{-1}{\alpha} \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \mu_t(\mathbf{s}, \mathbf{z}) \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}' \\
& + \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} \\
& - \alpha(\delta + \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \log \sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left(\frac{-1}{\alpha} \left[\int_{\mathbf{s}'} V_{t+1}(\mathbf{s}', \mathbf{z}') \int_{\mathbf{s}} \int_{\mathbf{a}} \mu_t(\mathbf{s}, \mathbf{z}) \pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) d\mathbf{a} d\mathbf{s} d\mathbf{s}'\right]\right) \\
=& \int_{\mathbf{s}} \sum_{\mathbf{z}} V_1(\mathbf{s}, \mathbf{z}) p_1(\mathbf{s}, \mathbf{z}) d\mathbf{s} + \alpha \left(\sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \log\left(\frac{D_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})}\right) - \delta\right)
\end{aligned}
$$

D.2. LQG Assumptions

The forward pass is the same as in the maximization. Here we focus on the backward pass.

$$
\begin{aligned}
\mathcal{P}_t(\mathbf{s}'|\mathbf{s}, \mathbf{a}, \mathbf{z}) =& \mathcal{N}(\mathbf{s}'|\mathbf{A}_t^z \mathbf{s} + \mathbf{b}_T^z \mathbf{a} + \mathbf{c}_T^z, \mathbf{\Sigma}_{s',t}^z) \\
\pi_t(\mathbf{a}|\mathbf{s}, \mathbf{z}) =& \mathcal{N}(\mathbf{a}|\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z, \mathbf{\Sigma}_{a,t}^z) \\
R_t(\mathbf{s}, \mathbf{a}) =& (\mathbf{g} - \mathbf{s})^T \mathbf{M}_t (\mathbf{g} - \mathbf{s}) + \mathbf{a}^T \mathbf{H}_t \mathbf{a} \\
\mu_t(\mathbf{s}, \mathbf{z}) =& \sum_i w_{t,i}(\mathbf{z}) \mathcal{N}(\mathbf{s}|\tau_{t,i}, \mathbf{\Sigma}_{t,i})
\end{aligned}
$$

Starting with $V_T(\mathbf{s}, \mathbf{z}) = R_T(\mathbf{s})$, we can calculate Q_T and then D_T. Using these we can calculate V_{T-1} and repeat. In the following the closed-form solution for these three functions are calculated.

Compute Q-Function

$$
\begin{aligned}
Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') =& \int_{\mathbf{s}'} V_{t+1}(\mathbf{s}',\mathbf{z}') \int_{\mathbf{a}} \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) \mathcal{P}_t(\mathbf{s}'|\mathbf{s},\mathbf{a},\mathbf{z}) d\mathbf{a} d\mathbf{s}' \\
=& \int_{\mathbf{s}'} \left(\mathbf{s}'^T \mathbf{V}_{t+1}^{z'} \mathbf{s}' + \mathbf{s}'^T \mathbf{v}_{t+1}^{z'} + v_{t+1}^{z'} \right) \int_{\mathbf{a}} \mathcal{N}(\mathbf{a}|\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z, \mathbf{\Sigma}_{a,t}^z) \mathcal{N}(\mathbf{s}'|\mathbf{A}_t^z \mathbf{s} + \mathbf{b}_t^z \mathbf{a} + \mathbf{c}_t^z, \mathbf{\Sigma}_{s',t}^z) d\mathbf{a} d\mathbf{s}' \\
=& \int_{\mathbf{s}'} \left(\mathbf{s}'^T \mathbf{V}_{t+1}^{z'} \mathbf{s}' + \mathbf{s}'^T \mathbf{v}_{t+1}^{z'} + v_{t+1}^{z'} \right) \mathcal{N}(\mathbf{s}'|\mathbf{A}_t^z \mathbf{s} + \mathbf{b}_t^z (\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z) + \mathbf{c}_t^z, \mathbf{\Sigma}_{s',t}^z + \mathbf{b}_t^z \mathbf{\Sigma}_{a,t}^z \mathbf{b}_t^{z,T}) d\mathbf{s}' \\
=& (\mathbf{A}_t^z \mathbf{s} + \mathbf{b}_t^z (\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z) + \mathbf{c}_t^z)^T \mathbf{V}_{t+1}^{z'} (\mathbf{A}_t^z \mathbf{s} + \mathbf{b}_t^z (\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z) + \mathbf{c}_t^z) + (\mathbf{v}_{t+1}^{z'})^T (\mathbf{A}_t^z \mathbf{s} + \mathbf{b}_t^z (\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z) + \mathbf{c}_t^z) \\
& + \mathrm{Tr}(\mathbf{V}_{t+1}^{z'} (\mathbf{\Sigma}_{s',t}^z + \mathbf{b}_t^z \mathbf{\Sigma}_{a,t}^z \mathbf{b}_t^{z,T})) + v_{t+1}^{z'} \\
=& \mathbf{s}^T \mathbf{Q}_t^{z,z'} \mathbf{s} + \mathbf{s}^T \mathbf{q}_t^{z,z'} + q_t^{z,z'}
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{Q}_t^{z,z'} =& (\mathbf{A}_t^z + \mathbf{b}_t^z \mathbf{K}_t^z)^T \mathbf{V}_{t+1}^{z'} (\mathbf{A}_t^z + \mathbf{b}_t^z \mathbf{K}_t^z) \\
\mathbf{q}_t^{z,z'} =& 2(\mathbf{A}_t^z + \mathbf{b}_t^z \mathbf{K}_t^z)^T \mathbf{V}_{t+1}^{z'} (\mathbf{b}_t^z \mathbf{k}_t^z + c_t^z) + (\mathbf{A}_t^z + \mathbf{b}_t^z \mathbf{K}_t^z)^T \mathbf{v}_{t+1}^{z'} \\
q_t^{z,z'} =& (\mathbf{b}_t^z \mathbf{k}_t^z + c_t^z)^T \mathbf{V}_{t+1}^{z'} (\mathbf{b}_t^z \mathbf{k}_t^z + c_t^z) + (\mathbf{b}_t^z \mathbf{k}_t^z + c_t^z) \mathbf{v}_{t+1}^{z'} + \mathrm{Tr}(\mathbf{V}_{t+1}^{z'} (\mathbf{\Sigma}_{s',t}^z + \mathbf{b}_t^z \mathbf{\Sigma}_{a,t}^z \mathbf{b}_t^{z,T})) + v_{t+1}^{z'}
\end{aligned}
$$

Compute $D_t(\mathbf{z}'|\mathbf{z})$

$$
\begin{aligned}
D_t(\mathbf{z}'|\mathbf{z}) =& \frac{P_t(\mathbf{z}'|\mathbf{z}) \exp\left[\frac{-1}{\alpha} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') d\mathbf{s}\right]}{\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left[\frac{-1}{\alpha} \int_{\mathbf{s}} \mu_t(\mathbf{s},\mathbf{z}) Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') d\mathbf{s}\right]} \\
=& \frac{P_t(\mathbf{z}'|\mathbf{z}) \exp\left[\frac{-1}{\alpha} \int_{\mathbf{s}} (\sum_i w_{t,i}(\mathbf{z}) \mathcal{N}(\mathbf{s}|\tau_{t,i}, \mathbf{\Sigma}_{t,i})) (\mathbf{s}^T \mathbf{Q}_t^{z,z'} \mathbf{s} + \mathbf{s}^T \mathbf{q}_t^{z,z'} + q_t^{z,z'}) d\mathbf{s}\right]}{\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left[\frac{-1}{\alpha} \int_{\mathbf{s}} (\sum_i w_{t,i}(\mathbf{z}) \mathcal{N}(\mathbf{s}|\tau_{t,i}, \mathbf{\Sigma}_{t,i})) (\mathbf{s}^T \mathbf{Q}_t^{z,z'} \mathbf{s} + \mathbf{s}^T \mathbf{q}_t^{z,z'} + q_t^{z,z'}) d\mathbf{s}\right]} \\
=& \frac{P_t(\mathbf{z}'|\mathbf{z}) \exp\left[\frac{-1}{\alpha} \sum_i w_{t,i}(\mathbf{z}) \left(\tau_{t,i}^T \mathbf{Q}_t^{z,z'} \tau_{t,i} + \tau_{t,i}^T \mathbf{q}_t^{z,z'} + q_t^{z,z'} + \mathrm{Tr}(\mathbf{\Sigma}_{t,i} \mathbf{Q}_t^{z,z'})\right)\right]}{\sum_{\mathbf{z}'} P_t(\mathbf{z}'|\mathbf{z}) \exp\left[\frac{-1}{\alpha} \sum_i w_{t,i}(\mathbf{z}) \left(\tau_{t,i}^T \mathbf{Q}_t^{z,z'} \tau_{t,i} + \tau_{t,i}^T \mathbf{q}_t^{z,z'} + q_t^{z,z'} + \mathrm{Tr}(\mathbf{\Sigma}_{t,i} \mathbf{Q}_t^{z,z'})\right)\right]}
\end{aligned}
$$

Compute V

$$
\begin{aligned}
V_t(\mathbf{s},\mathbf{z}) =& \int_{\mathbf{a}} R_t(\mathbf{s},\mathbf{a}) \pi_t(\mathbf{a}|\mathbf{s},\mathbf{z}) d\mathbf{a} + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') \\
=& \int_{\mathbf{a}} \left((\mathbf{g}-\mathbf{s})^T \mathbf{M}_t (\mathbf{g}-\mathbf{s}) + \mathbf{a}^T \mathbf{H}_t \mathbf{a} \right) \mathcal{N}(\mathbf{a}|\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z, \mathbf{\Sigma}_{a,t}^z) d\mathbf{a} + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) Q_t(\mathbf{s},\mathbf{z},\mathbf{z}') \\
=& (\mathbf{g}-\mathbf{s})^T \mathbf{M}_t (\mathbf{g}-\mathbf{s}) + (\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z)^T \mathbf{H}_t (\mathbf{K}_t^z \mathbf{s} + \mathbf{k}_t^z) + \mathrm{Tr}(\mathbf{H_t} \mathbf{\Sigma}_{a,t}^z) \\
& + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) (\mathbf{s}^T \mathbf{Q}_t^{z,z'} \mathbf{s} + \mathbf{s}^T \mathbf{q}_t^{z,z'} + q_t^{z,z'}) \\
=& \mathbf{s}^T \mathbf{V}_t^z \mathbf{s} + \mathbf{s}^T \mathbf{v}_t^z + v_t^z
\end{aligned}
$$

where

$$
\begin{aligned}
\mathbf{V}_t^z =& \mathbf{M}_t + (\mathbf{K}_t^z)^T \mathbf{H}_t \mathbf{K}_t^z + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \mathbf{Q}_t^{z,z'} \\
\mathbf{v}_t^z =& -2\mathbf{M}_t \mathbf{g} + 2\mathbf{K}_t^{z,T} \mathbf{H}_t \mathbf{k}_t^z + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) \mathbf{q}_t^{z,z'} \\
v_t^z =& \mathbf{g}^T \mathbf{M}_t \mathbf{g} + \mathbf{k}_t^{z,T} \mathbf{H}_t \mathbf{k}_t^z + \mathrm{Tr}(\mathbf{H_t} \mathbf{\Sigma}_{a,t}^z) + \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z}) q_t^{z,z'}
\end{aligned}
$$

E. Barycentric Interpolation

$$\begin{aligned}\min_{H_t(\mathbf{z}'|\mathbf{z})} \lambda \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})} \right) + (1-\lambda) \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) \\ \text{s.t.} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) = 1, \qquad \forall \mathbf{z}, t < T,\end{aligned}$$

$$L = \lambda \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})} \right) + (1-\lambda) \sum_{t=1}^{T-1} \sum_{\mathbf{z}} \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) + \beta_t(\mathbf{z})(\sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) - 1)$$

$$\begin{aligned}\frac{\partial L}{\partial H} &= \lambda \left(\log \frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})} + 1 \right) + (1-\lambda) \left(\log \frac{H_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} + 1 \right) + \beta_t(\mathbf{z}) \\ &= \lambda \left(\log \frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})} \right) + (1-\lambda) \left(\log \frac{H_t(\mathbf{z}'|\mathbf{z})}{P_t(\mathbf{z}'|\mathbf{z})} \right) + \beta_t(\mathbf{z}) \\ &= \log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})^{\lambda} H_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}}{D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}} \right) + \beta_t(\mathbf{z}) \\ &= \log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}} \right) + \beta_t(\mathbf{z})\end{aligned}$$

Setting to zero

$$\begin{aligned}\log \left(\frac{H_t(\mathbf{z}'|\mathbf{z})}{D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}} \right) &= -\beta_t(\mathbf{z}) \\ H_t(\mathbf{z}'|\mathbf{z}) &= D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda} \exp(-\beta_t(\mathbf{z}))\end{aligned}$$

Calculating beta:

$$
\begin{aligned}
1 &= \sum_{\mathbf{z}'} H_t(\mathbf{z}'|\mathbf{z}) \\
1 &= \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda} \exp(-\beta_t(\mathbf{z})) \\
\exp(\beta_t(\mathbf{z})) &= \sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda} \\
\beta_t(\mathbf{z}) &= \log\left(\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}\right)
\end{aligned}
$$

back in H:

$$
H_t(\mathbf{z}'|\mathbf{z}) = \frac{D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}}{\sum_{\mathbf{z}'} D_t(\mathbf{z}'|\mathbf{z})^{\lambda} P_t(\mathbf{z}'|\mathbf{z})^{1-\lambda}}
$$