# Discrete Approximate Information States in Partially Observable Environments

Lujie Yang, Kaiqing Zhang, Alexandre Amice, Yunzhu Li, Russ Tedrake

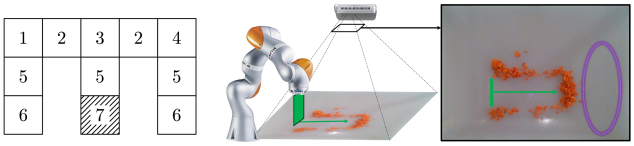
***Abstract*— The notion of approximate information states (AIS) was introduced in [1] as a methodology for learning task-relevant state representations for control in partially observable systems. They proposed particular learning objectives which attempt to reconstruct the cost and next state and provide a bound on the suboptimality of the closed-loop performance, but it is unclear whether these bounds are tight or actually lead to good performance in practice. Here we study this methodology by examining the special case of *discrete* approximate information states (DAIS). In this setting, we can solve for the globally optimal policy using value iteration, allowing us to disambiguate the performance of the AIS objective from the policy search. Going further, for small problems with finite information states, we reformulate the DAIS learning problem as a novel mixed-integer program (MIP) and solve it to its global optimum; in the infinite information states case, we introduce clustering-based and end-to-end gradient-based optimization methods for minimizing the DAIS construction loss. We study DAIS in three partially observable environments and find that the AIS objective offers relatively loose bounds for guaranteeing monotonic performance improvement and is sufficient but not necessary for implementing optimal controllers. DAIS may even prove useful in practice by itself or as part of mixed discrete- and continuous-state representations, due to its ability to represent logical state, to its potential interpretabilty, and to the availability of these stronger algorithms.**

## I. INTRODUCTION

In most autonomous control applications, the agent (or controller) only has access to partial observations of the system state [2]–[5]. Common examples include robot navigation [2], [3] and robotic manipulation [4], [5]. The key to planning and control in such partially observable systems is constructing a *state representation*: a function of the partial observations through which we can predict future performance of future control actions. There is ever-growing literature on representation learning for control in partially observable systems, ranging from the classic state estimation (filtering) in linear systems [6], to the deep learning-based approaches of learning for control from pixels [7]–[12].

Many of these recent approaches to representation learning for control from pixels are built upon *observation reconstruction/prediction*. In particular, they encode the history of observations (high-dimensional images) into lower-dimensional vectors to reconstruct and predict future observations [7]–[9]. Notably, these approaches are *task-agnostic*: the constructed representations are designed to recover all the information in the observations, including information irrelevant to the downstream control tasks. Such irrelevant information may easily distract the control and planning [13]. Moreover, no theoretical guarantees for the control performance were established for these types of representations, and the observations (images) are usually high-dimensional and challenging to reconstruct/predict.

Massachusetts Institue of Technolocy. Emails:{lujie, kaiqing, amice, liyunzhu, russt}@mit.edu




(a) CheeseMaze (b) Object Pile Manipulation

Fig. 1: Examples of tasks DAIS deals with.

On the other hand, when modeled by the framework of partially observable Markov decision processes (POMDPs), the state representation that is sufficient for performance evaluation and optimal control is known to be different from those that are necessary for predicting the observations. Specifically, it is known that the belief state (i.e. the posterior belief of the unobserved state given the action-observation history) is a sufficient statistic for POMDPs, on which the optimal policy can be defined and identified via dynamic programming [14]. In fact, the belief state belongs to a more general notion of *information state* [1], [15] – a function of history which is sufficient to: 1) compute the expected reward; 2) predict its next value. In [1], the authors showed that these two conditions are sufficient for performance evaluation in POMDPs. More importantly, it is also shown in [1] that any state representation that satisfies these two conditions approximately with uniform bounds over all possible observation/action histories, can be used to construct a Markov decision process (MDP) to identify the value function (and thus policy) of the original POMDP, with bounded loss of optimality. In other words, the two aforementioned properties provide rigorous metrics for the quality of a state representation of a POMDP based on its relevance to downstream optimal control. Such representations can thus be viewed as being *task-relevant*.

In this paper, we study a *discrete* approximate information state (DAIS) representation. Specifically, we aim to discover the possible discrete nature of the approximate information state (AIS) in many structured POMDPs, which can potentially represent logical state and improve the interpretability. Moreover, a discrete AIS enables the use of optimal planning methods, e.g., value iteration, to solve the approximate model efficiently. Finally, constructing a discrete AIS facilitates the direct use of the two aforementioned conditions in training, without resorting to the surrogate conditions given in [1] (see Sec. III for

details), which also requires predicting the potentially high-dimensional observations.

Our contributions are summarized as follows. First, we present a framework to construct DAIS without observation prediction, for POMDPs with both finite- and infinite-cardinality belief states. Second, for the finite belief space POMDPs, we propose a mixed integer programming (MIP)-based formulation for constructing the optimal state representation, followed by a novel reformulation technique that yields a globally optimal solution. By solving small problems to optimality, we are able to study the effect of DAIS fitting loss bounds on task performance. For the infinite belief space case, we develop both clustering-based and gradient-based methods and investigate the non-convex DAIS objective independently from the policy. We find that although the original AIS bound in [1] can be relatively loose for guaranteeing monotonic performance improvement, discrete model representations solved with exact value iteration can still yield optimal (or close to optimal in the infinite-belief case) control strategies. Third, we evaluate the effectiveness of DAIS on three benchmark partially observable environments, including a visual-feedback object pile manipulation task in robotics. We also demonstrate the interpretability of DAIS in some examples, and show the numerical advantages of planning over DAIS, compared to existing continuous-space POMDP solvers, e.g., [2].

## II. RELATED WORKS

Besides the most relevant work [1] on AIS discussed above, the other related works are summarized as follows.

**Representation Based on Observation Reconstruction/ Prediction:** Encountered with high-dimensional visual input, model-based reinforcement learning methods typically focus on reconstructing or predicting observations [7], [12], [16] to learn the underlying model for optimal planning. [13] aims to learn invariant representations without reconstruction, which has the closest motivation to ours. However, the framework was focused on the fully-observable settings of MDPs.

**POMDP Solvers:** Various POMDP solvers like pointed-based value iteration [17], [18] and incremental pruning [19] have been proposed to avoid the exponential growth of the value function, which is the major difficulty for solving POMDPs. However, most of the solvers are restricted to discrete spaces and require extensive iterations to update the value function. [2] deals with continuous POMDP but the algorithm is fairly slow to train and is sensitive to model parameters (as observed in Sec. VI).

**State Aggregation:** There are a number of works on state discretization [20], [21] and state aggregation [22] in MDPs. In particular, Givan et al. [23] propose to aggregate MDPs using bisimulation, the strictest partitioning form for preserving most properties. Ferns et al. [24] soften the exact equivalence requirement in bismulation using bisimulation metrics, presenting state aggregation techniques for MDPs which combine "behaviorally similar" states given the distance between their rewards and state distributions. Castro et al. [25] extend the notion of bisimulation metrics to POMDPs. However, they do not provide viable algorithms for computing equivalence and aggregation in belief space. In this work, we formulate simple optimization problems to learn DAIS as an effective discretization of the belief space.

## III. BACKGROUND

In this section, we provide the background for understanding the DAIS framework. We start with describing the POMDP and then introduce the definition of approximate information state, originated from [1].

**Partially Observable Markov Decision Process**: A POMDP is formally defined as a tuple $\langle \mathrm{S}, \mathrm{A}, T, r, \Omega, O, \gamma \rangle$, where S is the set of the states of the world, A is a set of actions that the agent can execute, $T$ is the stochastic transition function $T(s, a, s') = P(s_{t+1} = s' | s_t = s, a_t = a)$, $r(s, a)$ is the reward function, $\Omega$ is a set of possible observations, $O$ is an observation model with $O(s', o) = P(o_t = o | s_{t+1} = s')$ and $\gamma \in [0, 1)$ is the discount factor. The history until time $t$, denoted by $H_t$, is the summary of the past observations and actions, i.e., $H_t = (o_{1:t-1}, a_{1:t-1})$.

**Approximate Information State**: Let $(\epsilon, \delta)$ be positive real numbers, $(\mathrm{X}, \mathcal{G})$ be a measurable space, $d$ denote a probability metric between two probability distributions $\mu, \nu \in \Delta(\mathrm{X})$ (the space of probability measures on X) such as the Wasserstein distance or the Total Variation metrics. An approximate information state $\{Z_t\}_{t=1}^{T}$ is generated by a history compression function $\{\sigma_t : \mathrm{H_t} \to \mathrm{Z}\}_{t \geq 1}$, Markovian update kernel $\hat{P} : \mathrm{Z} \times \mathrm{A} \to \Delta(\mathrm{Z})$ and reward prediction function $\hat{r} : \mathrm{Z} \times \mathrm{A} \to \mathbb{R}$ where $Z_t = \sigma_t(H_t)$ and the following properties are satisfied for any $t$, any realization $h_t$ of $H_t$, and any action $a_t$ of $A_t \in \mathrm{A}$:

**(AP1)** Sufficient to predict the reward $R_t = r(s_t, a_t)$ approximately:

$$| \mathbb{E}[R_t | H_t = h_t, A_t = a_t] - \hat{r}(z_t, a_t) | \leq \epsilon.$$

**(AP2)** Sufficient to predict its Markovian transition approximately: for any Borel subset Y of Z, define $\mu_t(\mathrm{Y}) \coloneqq P(Z_{t+1} \in \mathrm{Y} | H_t = h_t, A_t = a_t)$ , $\nu_t(\mathrm{Y}) \coloneqq \hat{P}(Z_{t+1} \in \mathrm{Y} | z_t, a_t)$,

$$d(\mu_t, \nu_t) \leq \delta.$$

In general, the condition **(AP2)** can be abstract to enforce. [1] has thus proposed the following two surrogate conditions that imply **(AP2)**, which might be easier to enforce.

**(AP2'a)** Evolves deterministically like a state: there exists a measurable update function $\phi : \mathrm{Z} \times \Omega \times \mathrm{A}$ such that

$$z_{t+1} = \phi(z_t, o_t, a_t).$$

**(AP2'b)** Sufficient to predict future observations approximately: for any Borel subset Y of $\Omega$, define $\mu_t^o(\mathrm{Y}) \coloneqq P(O_t \in \mathrm{Y} | H_t = h_t, A_t = a_t), \nu_t^o(\mathrm{Y}) \coloneqq \hat{P}^o(O_t \in \mathrm{Y} | z_t, a_t)$, then

$$d(\mu_t^o, \nu_t^o) \leq \delta^o.$$

We denote the true value function of the history by $V(h_t)$ and the approximation obtained from AIS with dynamic

programming by $\hat{V}(z_t)$, then we have the following bound on the value function approximation error [1, Theorem 9]:

$$|V(h_t) - \hat{V}(z_t)| \leq \alpha, \quad \text{with} \quad \alpha = \frac{\epsilon + \gamma\delta\rho_d(\hat{V})}{1-\gamma},$$

where $\rho_d$ is a constant associated with the chosen probability metric related to the underlying extremization definition of that metric (see [1, Definition 6]). For $f$ defined over a discrete set we have that for the Wasserstein distance $\rho_d(f) = ||f||_{\text{Lip}}$, the Lipschitz semi-norm, and for Total Variation $\rho_d(f) = \frac{\max(f)-\min(f)}{2}$.

The power of AIS is that, by enforcing the conditions approximately with some relaxation error $(\epsilon, \delta)$, the value function of the AIS model, i.e., $\langle \text{Z}, \text{A}, \hat{P}, \hat{r} \rangle$, is also pointwise close to the actual value function of the POMDP, up to an error that can be bounded linearly by $(\epsilon, \delta)$. This provides a principled way to design metrics for representation learning for control in POMDPs, with provable suboptimality guarantees.

In many robotics applications, the observations, i.e., images, are of high dimensions and can be challenging to reconstruct and predict (i.e. to enforce **(AP2'b)**). Hence, we propose to only use **(AP2)**, which becomes tangible and more tractable under the discrete AIS framework. We introduce more details in the next sections.

## IV. PROBLEM FORMULATION

We now present several common types of POMDPs that we aim to address in the ensuing sections.

### A. Finite Belief Space

Some basic POMDP examples with finite state-action and observation spaces also by nature have deterministic transition and observation models. In such models, the set of belief states, i.e., the exact information state, has finite cardinality. In this case, it is thus sensible to design *discrete* AIS. We take the CheeseMaze [26] as an example.

**Example:** (CheeseMaze) The maze environment consists of 11 states (grid cells) and 7 observations (numbers on the grid cells) as shown in Fig. 1. An agent in the maze desires to reach the goal state (shaded cell) where the cheese lands. Its movement in all four directions (north, south, east and west) and the observation functions are deterministic (e.g. $P(o_t = o|s_{t+1} = \text{bottom left cell}) = \mathbb{1}(o = 6)$). The Bayesian update for the belief is:

$$\begin{aligned} b_{t+1}(s') &= f(b_t, a_t, o_t) \qquad (1)\\ &= \frac{P(o_t|s_{t+1} = s') \sum_s P(s_{t+1} = s'|s_t = s, a_t) b_t(s)}{P(o_t|b_t, a_t)}, \end{aligned}$$

where

$$P(o_t|b_t, a_t) = \sum_{s'} P(o_t|s_{t+1} = s') \cdot \sum_s P(s_{t+1} = s'|s_t = s, a_t) b_t(s). \qquad (2)$$

The belief update can be used to form the "belief MDP" for planning by marginalizing over the future observations:

$$P(b_{t+1} = b|b_t, a_t) = \sum_o P(o_t = o|b_t, a_t) \mathbb{1}(b = f(b_t, a_t, o)). \qquad (3)$$

### B. Infinite Belief Space

In general, with stochastic transition dynamics and observation models, there are infinitely many reachable belief states $b_t$ starting from an initial belief $b_0$. We propose to discretize the infinite-cardinality belief state space using the approximate information state conditions in Sec. III. In particular, we focus on two such settings that are common in robotics applications: continuous-state space POMDPs, and vision-feedback control tasks.

*1) Continuous-State POMDPs:* Most existing algorithms for solving model-based POMDPs focus on discrete states while many real-world applications, such as robot navigation and manipulation, are naturally represented using continuous states. Note that in this case, the belief state, i.e., the belief probability over the state space, becomes continuous and has infinite cardinality. We consider the same class of continuous POMDPs as studied in [2]. The dynamics are given by a linear-Gaussian model $P(\cdot|s_t, a_t) = \phi(s_t + f(a_t), \Sigma^{a_t})$, where $\phi$ denotes a Gaussian distribution with mean $s_t + f(a_t)$ and covariance $\Sigma^{a_t}$. The reward $r_{a_t}(s_t)$ is modeled by a linear combination of Gaussian distributions $r_{a_t}(s_t) = \sum_{i=1}^{N_r} w_i^r \phi_i(s_t|\mu_i^{a_t}, \Sigma_i^{a_t})$, where $w_i^r$ are weights, $\phi_i$ are Gaussian distributions and $N_r$ is a predefined number. The observation model $P(o_t|s_{t+1})$ is characterized by a Gaussian mixture model and by assuming uniform $P(s)$ and sampling $N_o$ observation/state pairs: $P(o_t = o|s_{t+1} = s) = \frac{P(o,s)}{P(s)} \propto \sum_{i=1}^{N_o} w_i^o \phi_i(s|s_i^o, \Sigma_i^o)$. The belief state is continuously valued and can be written as a Gaussian mixture:

$$b_t(s) = \sum_{j=1}^{N} w_j \phi_j(s|s_j, \Sigma_j), \qquad (4)$$

where $N$ is the number of Gaussian components, the weights $w_j > 0$ and $\sum_{j=1}^{N} w_j = 1$. Such a representation of belief states is a natural consequence of linear-Gaussian dynamics and Gaussian mixture observation functions because the belief update can then be computed in closed form with all the Gaussian based functions.

*2) Visual-Feedback Control Tasks:* We are interested in visual feedback manipulation tasks with quasi-static dynamics, where the observed images can serve as sufficient statistics of history. Essentially, the high-dimensional raw pixel images with reduced resolution can be viewed as $b_t$, which are mapped into a low-dimensional discrete representations $Z_t$ for reward and transition model prediction. We believe that image reconstruction or observation prediction is not necessary because the low-dimensional DAIS can capture the most essential information for the manipulation tasks.

## V. METHODOLOGY

We aim to learn a set of discrete approximate information states where Z is a finite set and $|\text{Z}| = n_z$ ($n_z$ is a positive integer). The discrete approximate information states can be encoded as one-hot vectors $Z_t$ and their categorical distributions can be represented using vectors $\bar{z}_t \in \Delta(\text{Z})$. In general, it is only tractable to sample from $\mu_t$ as closed form computation is prohibitively difficult. Meanwhile with DAIS,

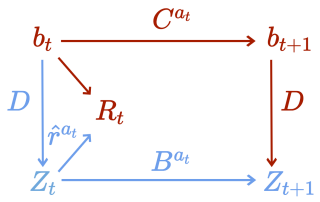



Fig. 2: DAIS learning framework. The red path shows that the current belief $b_t$ propagates to the next time step $b_{t+1}$ under Bayes rule $C^{a_t}$ with action $a_t$, which is then discretized to the DAIS under transformation $D$ at time $t+1$ as $Z_{t+1}$; the blue path demonstrates that the current belief $b_t$ is first discretized to $Z_t$ and then propagates to $Z_{t+1}$ under the learned transition $B^{a_t}$. We aim to minimize the discrepancy between the probability distribution of $Z_{t+1}$ obtained by the two paths as well as the difference between the reward predicted by $b_t$ (i.e. $\hat{r}^{a_t}$) and $Z_t$ (i.e. $R_t$).

we can calculate the distribution $P(Z_{t+1}|h_t, a_t)$ exactly using Bayes rule and belief discretization. Moreover, instead of having to minimize the surrogate loss from samples as in the original AIS framework [1], it is straightforward to encode probability distributions of discrete variables in vectors and measure their distance. Starting from the current belief, we can obtain the probability distribution $\bar{z}_{t+1}$ of the DAIS at the next time step following the two paths depicted in Fig. 2. The red path shows that we can update the current belief $b_t$ to $b_{t+1}$ using Bayes rule and subsequently discretize $b_{t+1}$ to $Z_{t+1}$ using the discretization map $D$:

$$\begin{aligned} P(Z_{t+1}|h_t, a_t) &= P(Z_{t+1}|b_t, a_t) \\ &= \sum_b P(Z_{t+1}|b_{t+1}=b) P(b_{t+1}=b|b_t, a_t) \\ &= \sum_b \mathbb{1}(D(b) = Z_{t+1}) P(b_{t+1}=b|b_t, a_t), \end{aligned} \tag{5}$$

where $Z_{t+1}$ only comes from the discretization of the next time step belief $b_{t+1}$ and is conditionally independent of $b_t$ and $a_t$. Following the red path, we can also define the categorical distributions $\bar{z}_{t+1}$ as

$$\bar{z}_{t+1} = \begin{bmatrix} P(Z_{t+1}=z_1|b_t, a_t) \\ \vdots \\ P(Z_{t+1}=z_{n_z}|b_t, a_t) \end{bmatrix} = \sum_b P(b_{t+1}=b|b_t, a_t) D(b). \tag{6}$$

In general, $D$ is a function that maps belief states, which are in general real-valued functions, to a finite set of categorical variables; in a discrete POMDP with deterministic dynamics and observations, $D$ becomes a projection matrix that projects a large set of belief states down to a much smaller set of DAIS.

Meanwhile, the blue path shows that $Z_{t+1}$ can also be obtained by first mapping $b_t$ to $Z_t$ and then propagating $Z_t$ to the next time step under the learned transition matrix $B^a = [B^a_{ij}]_{i,j\in[n_z]}$, where $B^a_{ij} = \hat{P}(Z_{t+1}=z_i|Z_t=z_j, a_t=a)$:

$$\begin{aligned} P(Z_{t+1}|b_t, a_t) &= \sum_z P(Z_{t+1}|Z_t=z, a_t) P(Z_t=z|b_t) \\ &= \sum_z P(Z_{t+1}|Z_t=z, a_t) \mathbb{1}(D(b_t)=z), \end{aligned} \tag{7}$$

$$\bar{z}'_{t+1} = \hat{P}(Z_{t+1}|Z_t, a_t) D(b_t). \tag{8}$$

We aim to match the probability distribution of the next step DAIS $\bar{z}_{t+1}$ and $\bar{z}'_{t+1}$ obtained by the two procedures as well as the reward predicted by both the belief and DAIS. This framework explicitly avoids predicting observations and is beneficial when the output is high-dimensional (which is common in robotics applications). With the tabular "DAIS MDP", we can run value iteration to obtain the optimal planning policy for the approximate model.

In the following subsections, we first describe how to formulate the finite-belief DAIS learning problem as an MIP and then extend it to the infinite-belief case with gradient-based and clustering-based optimization schemes.

### A. Finite Belief Space

In discrete POMDPs with deterministic dynamics, the number of finite beliefs $n_b$ is bounded. We would like to use a much smaller number ($n_z < n_b$) of DAIS to represent the task-relevant information of the belief optimally, i.e., minimizing the loss that enforces the AIS conditions **(AP1)** and **(AP2)**. Due to the discreteness of the belief space, we can describe each belief state $b_t$ as a one-hot vector $\bar{b}_k$ (where the $k$-th entry is 1), write the belief update as a matrix multiplication and formulate the DAIS learning as a mixed-integer program:

$$\begin{aligned} \min_{\{B^a\}, D, \{\hat{r}^a\}} \quad & \sum_a \sum_{k=1}^{n_b} |r^a_k - \hat{r}^a D \bar{b}_k|^2 + \|B^a D \bar{b}_k - D C^a \bar{b}_k\|^2 \\ \text{s.t.} \quad & D_{ij} \in \{0,1\}, \quad \forall i,j \text{ and } \mathbf{1}^T D = \mathbf{1}^T \\ & B^a_{ij} \geq 0, \quad \forall i,j,a \text{ and } \mathbf{1}^T B^a = \mathbf{1}^T, \quad \forall a, \end{aligned} \tag{9}$$

where we denote the belief MDP transition probability matrix by $C^a = [C^a_{ij}]_{i,j\in[n_b]}$ with $C^a_{ij} = P(\bar{b}_{t+1} = \bar{b}_i | \bar{b}_t = \bar{b}_j, a_t = a)$, the DAIS transition probability matrix by $B^a$, and the projection matrix by $D \in \{0,1\}^{n_z \times n_b}$. $r^a_k = \mathbb{E}[R_t | \bar{b}_t = \bar{b}_k, a_t = a]$ and $\hat{r}^a = [\hat{r}(z_1, a), \cdots, \hat{r}(z_{n_z}, a)]$ is the reward estimation vector with action $a$ for all $z$. $\mathbf{1}$ denotes an all-one vector. The two terms in the objective enforce **(AP1)** and **(AP2)** respectively.

*1) Reformulating Bilinear Optimization Problem:* Notice that the optimization objective is bilinear in $B^a$ and $D$ as well as $\hat{r}^a$ and $D$. Such bilinear objectives are generally intractable for MIP solvers. To make the optimization problem amenable to numerical computation, we use change of variables $Q^a = B^a D, \bar{r}^a = \hat{r}^a D$ and introduce binary auxiliary variables $\{t_{j_1 j_2}\}_{j_1, j_2}$ to reformulate the optimization problem:

$$\min_{\{Q^a\}, D, \{\bar{r}^a\}, \{t_{j_1 j_2}\}} \quad \sum_a \sum_{k=1}^{n_b} |r^a_k - \bar{r}^a \bar{b}_k|^2 + \|Q^a \bar{b}_k - D C^a \bar{b}_k\|^2$$

$$
\begin{aligned}
\text{s.t.}\quad & D_{ij} \in \{0,1\}, \quad \forall i,j \text{ and } \mathbf{1}^T D = \mathbf{1}^T \\
& Q^a_{ij} \geq 0, \quad \forall i,j,a \text{ and } \mathbf{1}^T Q^a = \mathbf{1}^T, \quad \forall a \\
& t_{j_1 j_2} \in \{0,1\}, \quad \forall j_1 \in [n_z], j_2 \in [n_z], j_1 < j_2 \\
& t_{j_1 j_2} - 1 \leq D_{:j_1} - D_{:j_2} \leq 1 - t_{j_1 j_2} \\
& D_{:j_1} + D_{:j_2} \leq 1 + t_{j_1 j_2} \\
& Q^a_{:j_1} - Q^a_{:j_2} \leq 1 - t_{j_1 j_2}, \quad \forall a \\
& (t_{j_1 j_2} - 1)M \leq \bar{r}^a_{:j_1} - \bar{r}^a_{:j_2} \leq (1 - t_{j_1 j_2})M, \quad \forall a \\
& \sum_{j_1, j_2} t_{j_1 j_2} \geq n_b - n_z,
\end{aligned} \tag{10}
$$

where $D_{:j}$ denotes the $j^{\text{th}}$ column of matrix $D$ and $M$ can be set to $\max |r^a_{k_1} - r^a_{k_2}|$. This optimization problem can be efficiently solved to its global optimum using off-the-shelf solvers like Gurobi [27]. The additional constraints on $t_{j_1 j_2}, Q^a, \bar{r}^a$ and $D$ adopt the big-M technique and retains the important structure of $Q^a$ and $\bar{r}^a$ as the multiplication of a matrix and the projection matrix $D$: $D$'s columns are one-hot vectors, and multiplying $B^a$ (resp. $\hat{r}^a$) by $D$ is essentially selecting certain columns of $B^a$ (resp. $\hat{r}^a$) and concatenating them into $Q^a$ (resp. $\bar{r}^a$). The binary auxiliary variables $t_{j_1 j_2}$ specify the connections between $D$'s column selection behavior and $Q^a$ (resp. $\bar{r}^a$)'s columns: when $t_{j_1 j_2} = 1$, $D_{:j_1} = D_{:j_2}$ guarantees $Q^a_{:j_1} = Q^a_{:j_2}$ and $\bar{r}^a_{:j_1} = \bar{r}^a_{:j_2}$ (meaning that $D$ is selecting the same column from $B^a$ for both $j_1$-th and $j_2$-th column of $Q^a$); when $t_{j_1 j_2} = 0$, $D$'s $j_1$-th column is guaranteed to be different from its $j_2$-th column and there are no constraints on $Q^a$ (resp. $\bar{r}^a$)'s corresponding columns.

### B. Infinite Belief Space

We extend our discrete representation learning framework to infinite belief settings. We propose two approaches with function approximations to handle the infinitely many belief states.

*1) Gradient-Based Optimization:* We parametrize the discretization map $D$ as well as the transition and reward estimation models $\{B^a\}_{a\in\mathrm{A}}$ and $\{\hat{r}^a\}_{a\in\mathrm{A}}$ as neural networks with a set of parameters $\theta$ to minimize the DAIS loss in Eq. (11) using end-to-end gradient-based optimization:

$$
\begin{aligned}
\min_{\theta} \quad & \sum_a \sum_t |r^a_t - \hat{r}^a_\theta D_\theta(b_t)|^2 + \|B^a_\theta D_\theta(b_t) - \bar{z}_{t+1}\|^2 \\
\text{s.t.} \quad & [B^a_\theta]_{ij} \geq 0, \quad \forall i,j,a \text{ and } \mathbf{1}^T B^a_\theta = \mathbf{1}^T, \quad \forall a,
\end{aligned} \tag{11}
$$

where $r^a_t = \mathbb{E}[R_t | b_t, a_t = a]$ and $\bar{z}_{t+1}$ is calculated using Eq. (6). Unlike the finite belief setting where there are finitely many $r^a_k$ and $\bar{b}_k$, $r^a_t$ and $b_t$ can be assumed to have a continuous spectrum of values and the loss corresponding to **(AP1)** and **(AP2)** have to be minimized through sampling. In the continuous-state POMDPs with Gaussian mixed models, i.e., the setting in Sec. IV-B 1), the weights, means and covariances of a Gaussian mixture characterizing a belief state are flattened and concatenated into a single vector as the input to the discretization map $D$, which outputs a one-hot vector $Z_t$ as the discrete representation; in visual feedback control tasks, i.e., the setting in Sec. IV-B 2), the images are fed into the discretization map $D$ instantiated by a convolutional neural network followed by categorical reparametrizaiton. In order to allow backpropagation through categorical variables to adjust the parameters of $D$, $\{B^a\}$ and $\{\hat{r}^a\}$ simultaneously, we use the Gumble-Softmax [28] as a continuous approximation to the one-hot vector.

**Algorithm 1** DAIS Learning and Planning

```
Generate data (b_t, a_t, r_t, b_{t+1}) from rollout samples of {a_t, o_t}_{t≥1} using Eq. (1)
if gradient-based optimization then
    Solve Eq. (11) for {B^a}, D, {r̂^a}, using gradient-based solvers
else
    Find D via Total Variation K-means clustering
    Solve Eq. (13) for {B^a}, {r̂^a}
end if
policy, V = value_iteration ({B^a}, {r̂^a})
```

The discretization map $D$ essentially aggregates beliefs into clusters based on **(AP1)** and **(AP2)** loss. The one-hot vector $Z_t$ indicates that the current belief is assigned deterministically to the cluster corresponding to $Z_t$'s non-zero entry. The next-time-step belief $b_{t+1}$ given the current action has the probability of being assigned to the clusters based on the categorical distribution specified by $\bar{z}_{t+1}$.

*2) Clustering-Based Optimization:* Jointly optimizing $D$, $\{B^a\}_{a\in\mathrm{A}}$ and $\{\hat{r}^a\}_{a\in\mathrm{A}}$ as in Eq. (11) is highly nonconvex and generally intractable (note that even the discrete case with deterministic dynamics in the previous section requires convex reparametrization and the MIP reformulation technique). Therefore, we propose to sequentially optimize $D$ followed by $\{B^a\}$ and $\{\hat{r}^a\}$ jointly. Because the expected reward $r^a_t = \int_s r(s,a) b_t(s) ds$ is linear in the belief, aggregating the belief states with small distances to each other helps reduce the loss associated with **(AP1)**. Similarly, because the Bayesian update Eq. (1) is linear in belief, starting from belief states $b_t$ close in probability metrics and executing the same action $a$ result in $b_{t+1}$ close to each other. If these neighboring $b_t$ get mapped to the same DAIS $z_i$ and similar $b_{t+1}$ get mapped to the same DAIS $z_j$, $\bar{z}_{t+1}$ will become a one-hot vector $D(b_{t+1})$ and the second term in Eq. (13)'s objective can be made small with $B^a_{ji} = 1$. Hence, we first find a suitable discretization $D$ via K-means clustering under the total variation-distance metric and then solve a constrained convex optimization problem to minimize the DAIS loss:

$$
\begin{aligned}
\min_{\{B^a\},\{\hat{r}^a\}} \quad & \sum_a \sum_t |r_t - \hat{r}^a z_t|^2 + \|B^a z_t - \bar{z}_{t+1}\|^2 \\
\text{s.t.} \quad & B^a_{ij} \geq 0, \quad \forall i,j,a \text{ and } \mathbf{1}^T B^a = \mathbf{1}^T, \quad \forall a,
\end{aligned} \tag{12}
$$

where $z_t = D(b_t)$ is the one-hot vector representing the clusters obtained by total variation K-means clustering [29] and $\bar{z}_{t+1}$ is again computed using Eq. (6).

### C. Planning

One main advantage of DAIS is that we can run value iteration to obtain the optimal policy for such a representation. This way, we are able to "solve the approximate model exactly". Note that other planning approaches, i.e., policy iteration, Monte-Carlo tree search, may also be used for the DAIS model, but we focus on value iteration for simplicity. The overall DAIS learning and planning pipeline is summarized in Algorithm 1.

## VI. RESULTS

In this section, we validate our discrete representation learning framework for both finite belief space task (CheeseMaze), and infinite belief space tasks: one with continuous-state space (Corridor Navigation) and the other with high-dimensional visual inputs for robotic manipulation (Object Pile Manipulation).

### A. CheeseMaze

We investigate the relationship between learning loss, model performance and DAIS dimension in the CheeseMaze example adapted from [1]. We solve the DAIS optimization program (10) to its global optimum using Gurobi. As baselines comparison, we replace the second term in Eq. (10) that enforces **(AP2)** by losses corresponding to **(AP2'ab)**, **(AP2)+(AP2'a)**, **(AP2)+(AP2'b)** and **(AP2)+(AP2'ab)** respectively. Recall that the DAIS construction loss associated with **(AP1)+(AP2)** gives a concrete bound on the suboptimality of the downstream policy, we plot the DAIS construction loss associated with **(AP1)+(AP2)** for the five different optimization programs in Fig. 3a. We also compute the bounds $\alpha$ on the loss in performance and find them orders of magnitude larger (e.g., 17.6 for $n_z = 11$ and 189.9 for $n_z = 7$) than the empirical value function approximation errors. Although the DAIS fitting loss offers a relatively loose bound on approximation errors and consequently task performance (e.g., purple curve with larger DAIS fitting loss in Fig. 3a can have smaller value function approximation mean squared errors in Fig. 3e compared with the blue curve), Fig. 3c shows that DAIS can still recover the optimal controller computed from the true belief states (dashed line) up to compressed dimension $n_z = 9$ by solving the DAIS model exactly with value iteration. Intuitively, the higher dimensional the discrete representation is, the more capacity it has to capture useful information to model the task. Using only **(AP2)**, we observe that the DAIS loss and value function approximation error decrease monotonically as the maximum DAIS dimension grows (Fig. 3). The AIS loss is zero at 15 states, which is the true cardinality of the belief state, but the value function approximation error reaches zero with just 11 states.

Predicting observations is unnecessary in the regime where the DAIS can retain the optimal sufficient statistics for planning and control. Although we do not encourage observation prediction, this can still be done in this small-scale example. We then observe that in the suboptimal regime where the low-dimensional DAIS has to sacrifice useful information, predicting the output and DAIS' deterministic evolution (i.e., enforcing conditions **(AP2'ab)**) can help decrease the value function approximation error and improve the overall performance (Fig. 3c). Note that the last three optimization programs have redundancy ((**AP2'ab)** imply **(AP2)**), the additional losses can change the objective landscape and thus might offer some numerical advantages for empirical implementation.

**Remark (Interpretability of DAIS in CheeseMaze):** Notably, one main advantage of DAIS is that the learned representation may be readily interpretable, which can be illustrated in this example. For instance, the DAIS algorithm learns to aggregate the three belief states where it is certain about its location at the bottom left cell $b_t(s) = \mathbb{1}(s =$ bottom left cell), certain about its location at the bottom right cell $b_t(s) = \mathbb{1}(s =$ bottom right cell) and uncertain about its location at the bottom left or right cell with probability 0.5 each $b_t(s) = 0.5 \cdot \mathbb{1}(s =$ bottom left cell) $+ 0.5 \cdot \mathbb{1}(s =$ bottom right cell). The aggregation of these three beliefs does not sacrifice information for planning at all because the optimal action at all three belief states is to go north. DAIS also achieves similar aggregation for the three belief states associated with the middle left and right cells without losing any information for optimal planning (as demonstrated by the non-degrading performance and small value function approximation error until $n_z = 11$ in Fig. 3).

**Remark (Minimality of State Representation):** When implementing the optimal controller, it is possible to fully describe the evolution of the CheeseMaze using only 7 controller states by aggregating the belief states with the same optimal action and analyzing their closed-loop transitions. However, this closed-loop state space is insufficient for describing transitions under policies other than the optimal one. As **(AP2)/(AP2'ab)** require the distribution bound to hold for *all* possible histories and actions, for the histories and actions that are not covered by the optimal policy, the error bound $\delta$ (or $\delta^o$) can be vacuous. In other words, this 7-state representation cannot be properly characterized by the AIS framework, implying that the AIS conditions might be only sufficient but not necessary for optimal decision-making. An interesting observation is that if we decrease the weights for the transition loss corresponding to the belief states executing the same optimal action in the handcrafted controller, the optimization program with loss **(AP2)+(AP2'ab)** will be able to find a 8-state DAIS that recovers the optimal policy.

### B. Corridor Navigation

We test the effectiveness of DAIS on the robot corridor navigation task used in [2], which fits in the setting in Sec. IV-B 1). We observe that the clustering-based optimization approach leads to better and more consistent performance than gradient-based optimization in this example, and thus report results from the former in Fig. 3. As an alternative to computing $\bar{z}_{t+1}$ analytically following Eq. 6, we can approximate $P(Z_{t+1}|Z_t, a_t)$ using samples:

$$P(Z_{t+1} = z_i | Z_t = z_j, a_t = a) \approx \frac{\sum_t \mathbb{1}(D(b_{t+1}) = z_i))\mathbb{1}(a_t = a)}{\sum_t \mathbb{1}(D(b_t) = z_j))\mathbb{1}(a_t = a)}.$$

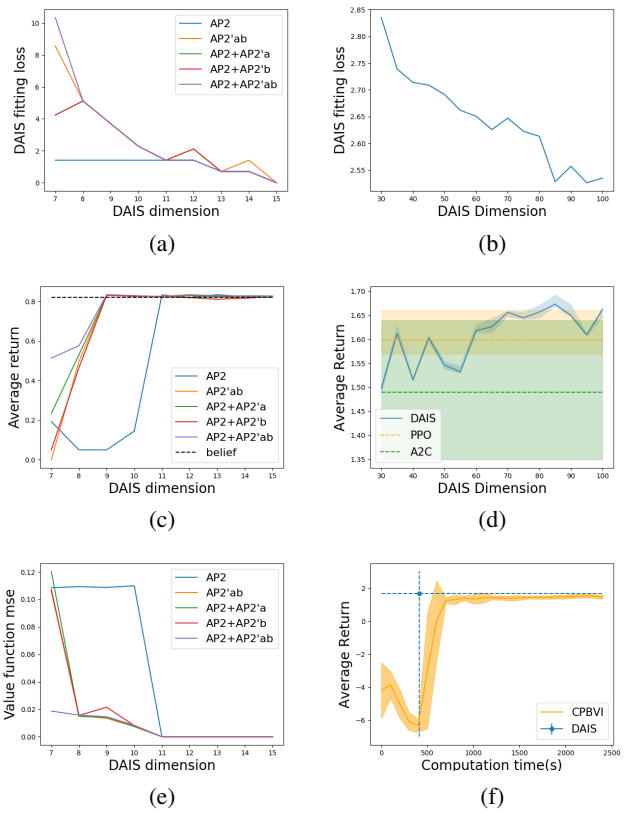



Fig. 3: (left) DAIS loss, task performance from empirical rollouts and value function approximation error vs maximum DAIS dimension in CheeseMaze example. (right) DAIS loss and task performance vs exact DAIS dimension for Corridor Navigation with 10 random seeds. DAIS achieves higher return, more robust performance and much smaller runtime compared to CPBVI.

The optimization program that uses samples to approximate the DAIS transition kernel then becomes:

$$\min_{\{B^a\},\{\hat{r}^a\}} \quad \sum_a \sum_t |r_t - \hat{r}^a z_t|^2 + \|B^a z_t - D(b_{t+1})\|^2$$
$$\text{s.t.} \quad B^a_{ij} \geq 0, \quad \forall i, j, a \quad \text{and} \quad \mathbf{1}^T B^a = \mathbf{1}^T, \quad \forall a. \tag{13}$$

Our method is compared with continuous point-based value iteration (CPBVI), a competitive baseline solution proposed in [2]. CPBVI samples belief points to perform Bellman updates due to the piecewise-linearity of the value function. In contrast, our method first aggregates the belief states into discrete variables and then runs value iteration on this finite set. In our experiments, we observe that CPBVI is extremely sensitive to environmental parameters, takes a long time to train and can fail to converge for certain model parameters. On the contrary, our DAIS does not suffer from convergence issues and consistently achieves higher return and much lower variance.

We also compare DAIS + value iteration against continuous information states + reinforcement learning (RL). We feed the continuous belief states (i.e. the information states with both $\epsilon$ and $\delta$ equal to 0 in **(AP1)** and **(AP2)**) into state-of-the-art RL algorithms such as PPO and A2C to learn a policy. The RL implementation is using Stable-Baselines3 [30] across 10 random seeds. The *discrete* approximate information states incur approximation error for the model representation but enable quick synthesis of the optimal controller for the approximate model; the *continuous* information states achieve zero AIS construction loss but make it much harder to obtain the optimal policy due to function approximation. Fig. 3d shows that for tasks with certain structures (e.g. the innate discreteness in corridor navigation), DAIS can achieve better performance with much lower variance than running PPO and A2C on continuous information states.

As in any K-means approach, convergence to local optima is possible, given different initialization. Moreover, increasing $K$ does not necessarily improve test prediction due to various factors attributable to overfitting [31]. Nevertheless, in Fig. 3b we observe a strong downward trend in the loss as the dimension of the DAIS is allowed to increase. As expected, this downward trend in loss correlates with an upward trend in average return. Though the increase in performance is less dramatic, it is important to note that the higher DAIS dimensions give a stronger a priori bound on the worst case suboptimality even if they achieve roughly the same expected return.

### C. Visual-Feedback Control for Object Pile Manipulation

We are interested in manipulating a pile of objects (e.g., a pile of carrot pieces), whose movement is more "fluid" with interactions among themselves (carrot pieces colliding and pushing each other). We aim to extract such challenging evolution in image space into transition in a discrete representation and follow the setup developed by Suh et al. [5] where the robot manipulator is required to use a flat pusher to push the object pile into a target region. This can be viewed as an example of the setting in Sec. IV-B 2). In total, 2000 trajectories of length 20 are generated with randomly sampled actions (i.e. the pusher's starting and ending locations) in the Pymunk simulator. The greyscale images of the object pieces are downsampled to $32 \times 32$. The images are then fed into a convolutional neural network followed by a Gumble-Softmax [28], [32] as DAIS $Z_t$. The one-hot vector $Z_t$ then goes through a feedforward neural network $B^a$ with softmax as the last layer to output the categorical distribution of DAIS $\bar{z}_{t+1}$ at the next time step. The parameters of all the neural networks are optimized using the end-to-end gradient-based method proposed in Sec. V-B.1). Fig. 4 shows that reasoning in the low-dimensional DAIS space without reconstructing or predicting the high dimensional visual outputs enables the robot manipulator to push the object pile into different target sets including circles, H-shaped and T-shaped regions.

## VII. CONCLUSIONS

In this paper, we evaluate discrete task-relevant representations for planning and control in partially observable environments using AIS. For finite-belief space tasks, we formulate a mixed-integer program

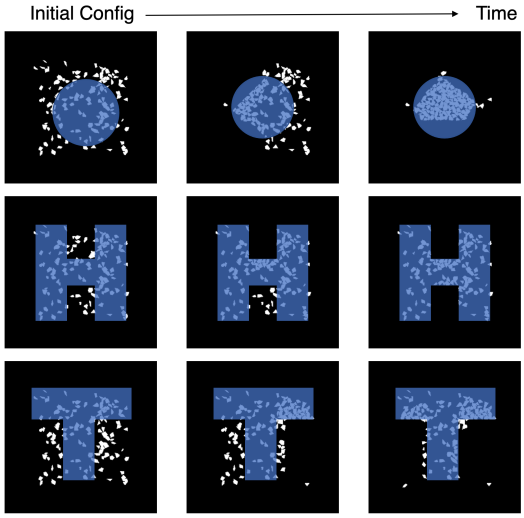



Fig. 4: DAIS performance in object pile manipulation task: the robot manipulator is required to push the object pieces into the blue region.

for solving the globally optimal DAIS in terms of expected reward and Markovian transition prediction; in infinite-belief space tasks, we develop new gradient-based and clustering-based optimization methods to learn the discrete approximate representation. Even the simple finite CheeseMaze example demonstrates that the AIS bounds on closed-loop performance can be loose. However, we posit that DAIS can still be effective due to its ability to extract the most relevant information to accomplish the tasks, which often times can be characterized in a discrete form, especially for control tasks with certain structures. We are interested in validating the effectiveness of DAIS on the real robot and other partially observable robotic control tasks in the future.

## ACKNOWLEDGMENT

The authors would like to thank Jack Umenberger, Alexandre Megretski and Terry Suh for helpful discussions and feedback as well as Boyuan Chen for helping with the RL baselines experiments.

## REFERENCES

[1] J. Subramanian, A. Sinha, R. Seraj, and A. Mahajan, "Approximate information state for approximate planning and reinforcement learning in partially observed systems," *arXiv preprint arXiv:2010.08843*, 2020.

[2] J. M. Porta, N. Vlassis, M. T. Spaan, and P. Poupart, "Point-based value iteration for continuous pomdps," 2006.

[3] A. Foka and P. Trahanias, "Real-time hierarchical POMDPs for autonomous robot navigation," *Robotics and Autonomous Systems*, vol. 55, no. 7, pp. 561–571, 2007.

[4] K. Hsiao, L. P. Kaelbling, and T. Lozano-Perez, "Grasping POMDPs," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*. IEEE, 2007, pp. 4685–4692.

[5] H. Suh and R. Tedrake, "The surprising effectiveness of linear models for visual foresight in object pile manipulation," *arXiv preprint arXiv:2002.09093*, 2020.

[6] R. E. Kalman, "A new approach to linear filtering and prediction problems," *Transactions of the ASME–Journal of Basic Engineering*, vol. 82, no. Series D, pp. 35–45, 1960.

[7] D. Hafner, T. Lillicrap, I. Fischer, R. Villegas, D. Ha, H. Lee, and J. Davidson, "Learning latent dynamics for planning from pixels," in *International Conference on Machine Learning*. PMLR, 2019, pp. 2555–2565.

[8] A. Lee, A. Nagabandi, P. Abbeel, and S. Levine, "Stochastic latent actor-critic: Deep reinforcement learning with a latent variable model," *Advances in Neural Information Processing Systems*, vol. 33, 2020.

[9] D. Yarats, A. Zhang, I. Kostrikov, B. Amos, J. Pineau, and R. Fergus, "Improving sample efficiency in model-free reinforcement learning from images," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 12, 2021, pp. 10 674–10 681.

[10] X. Fu, G. Yang, P. Agrawal, and T. Jaakkola, "Learning task informed abstractions," in *International Conference on Machine Learning*. PMLR, 2021, pp. 3480–3491.

[11] M. Watter, J. T. Springenberg, J. Boedecker, and M. Riedmiller, "Embed to control: A locally linear latent dynamics model for control from raw images," *arXiv preprint arXiv:1506.07365*, 2015.

[12] M. Igl, L. Zintgraf, T. A. Le, F. Wood, and S. Whiteson, "Deep variational reinforcement learning for pomdps," in *International Conference on Machine Learning*. PMLR, 2018, pp. 2117–2126.

[13] A. Zhang, R. T. McAllister, R. Calandra, Y. Gal, and S. Levine, "Learning invariant representations for reinforcement learning without reconstruction," in *International Conference on Learning Representations*, 2020.

[14] K. J. Åström, "Optimal control of markov processes with incomplete state information," *Journal of Mathematical Analysis and Applications*, vol. 10, pp. 174–205, 1965.

[15] P. Kumar and P. Varaiya, *Stochastic Systems: Estimation, Identification, and Adaptive Control*. SIAM, 1986, vol. 75.

[16] A. Zhang, Z. C. Lipton, L. Pineda, K. Azizzadenesheli, A. Anandkumar, L. Itti, J. Pineau, and T. Furlanello, "Learning causal state representations of partially observable environments," *arXiv preprint arXiv:1906.10437*, 2019.

[17] J. Pineau, G. Gordon, S. Thrun, *et al.*, "Point-based value iteration: An anytime algorithm for pomdps," in *IJCAI*, vol. 3. Citeseer, 2003, pp. 1025–1032.

[18] M. T. Spaan and N. Vlassis, "Perseus: Randomized point-based value iteration for pomdps," *Journal of artificial intelligence research*, vol. 24, pp. 195–220, 2005.

[19] A. R. Cassandra, M. L. Littman, and N. L. Zhang, "Incremental pruning: A simple, fast, exact method for partially observable markov decision processes," *arXiv preprint arXiv:1302.1525*, 2013.

[20] D. Bertsekas, "Convergence of discretization procedures in dynamic programming," *IEEE Transactions on Automatic Control*, vol. 20, no. 3, pp. 415–419, 1975.

[21] R. Munos and A. Moore, "Variable resolution discretization in optimal control," *Machine learning*, vol. 49, no. 2, pp. 291–323, 2002.

[22] W. Whitt, "Approximations of dynamic programs, i," *Mathematics of Operations Research*, vol. 3, no. 3, pp. 231–243, 1978.

[23] R. Givan, T. Dean, and M. Greig, "Equivalence notions and model minimization in markov decision processes," *Artificial Intelligence*, vol. 147, no. 1-2, pp. 163–223, 2003.

[24] N. Ferns, P. Panangaden, and D. Precup, "Metrics for finite markov decision processes." in *UAI*, vol. 4, 2004, pp. 162–169.

[25] P. S. Castro, P. Panangaden, and D. Precup, "Equivalence relations in fully and partially observable markov decision processes," in *Twenty-First International Joint Conference on Artificial Intelligence*, 2009.

[26] R. A. McCallum, "Overcoming incomplete perception with utile distinction memory," in *Proceedings of the Tenth International Conference on Machine Learning*, 1993, pp. 190–196.

[27] G. Optimization, "Inc.,"gurobi optimizer reference manual," 2015," 2014.

[28] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with gumbel-softmax," *arXiv preprint arXiv:1611.01144*, 2016.

[29] Y. Chen, T. T. Georgiou, and A. Tannenbaum, "Optimal transport for gaussian mixture models," *IEEE Access*, vol. 7, pp. 6269–6278, 2018.

[30] A. Raffin, A. Hill, M. Ernestus, A. Gleave, A. Kanervisto, and N. Dormann, "Stable baselines3," https://github.com/DLR-RM/stable-baselines3, 2019.

[31] G. Hamerly and C. Elkan, "Learning the k in k-means," *Advances in neural information processing systems*, vol. 16, pp. 281–288, 2004.

[32] C. J. Maddison, A. Mnih, and Y. W. Teh, "The concrete distribution: A continuous relaxation of discrete random variables," *arXiv preprint arXiv:1611.00712*, 2016.