VARIATIONAL INTRINSIC CONTROL REVISITED

Anonymous authors
Paper under double-blind review

ABSTRACT

In this paper, we revisit variational intrinsic control (VIC), an unsupervised reinforcement learning method for finding the largest set of intrinsic options available to an agent. In the original work by Gregor et al. (2016), two VIC algorithms were proposed: one that represents the options explicitly, and the other that does it implicitly. We show that the intrinsic reward used in the latter is subject to bias in stochastic environments, causing convergence to suboptimal solutions. To correct this behavior and achieve the maximal empowerment, we propose two methods respectively based on the transitional probability model and Gaussian Mixture Model. We substantiate our claims through rigorous mathematical derivations and experimental analyses.

1 INTRODUCTION

Variational intrinsic control (VIC) proposed by Gregor et al. (2016) is an unsupervised reinforcement learning algorithm that aims to discover as many *intrinsic* options as possible, i.e., the policies with a termination condition that meaningfully affect the world. The main idea of VIC is to maximize the mutual information between the set of options and final states, called *empowerment*. The maximum empowerment is desirable because it maximizes the information about the final states the agent can achieve with the available options. These options are independent of the extrinsic reward of the environment, so they can be considered as the agent's universal knowledge about the environment.

The concept of empowerment has been introduced in (Klyubin et al., 2005; Salge et al., 2014) along with methods for measuring it based on Expectation Maximization (Arimoto, 1972; Blahut, 1972). They defined the option as a sequence of a fixed number of actions. Yeung (2008) proposed to maximize the empowerment using the Blahut & Arimoto (BA) algorithm, but its complexity increase exponentially with the length of sequence, rendering it impractical for high dimensional and long-horizon options. Mohamed & Rezende (2015) adopted techniques from deep learning and variational inference (Barber & Agakov, 2003) and successfully applied empowerment maximization for high dimensional and long-horizon control. However, this method maximizes the empowerment over *open-loop* options, meaning that the sequence of action is chosen in advance and conducted regardless of the (potentially stochastic) environment dynamics. Obviously this often limits performance severely as the agent cannot properly react to the environment, and it tends to lead to a significant underestimation of empowerment (Gregor et al., 2016).

To overcome this limitation, Gregor et al. (2016) proposed to use the *closed-loop* options where the agent considers transited states while it is generating a sequence of actions. This type of option differs from those in Klyubin et al. (2005), Salge et al. (2014) and Mohamed & Rezende (2015) in that they use *closed-loop* options and have a termination condition, instead of a fixed number of actions. They presented two algorithms: VIC with explicit and implicit options (we will call them *explicit* and *implicit VIC* from here on). Explicit VIC defines a fixed number of options. An option is sampled at the beginning of the trajectory, conditioning the policies of an agent until the termination. In other words, both the state and the sampled option are the input to the policy function of the agent. One clear limitation of explicit VIC is that it requires the preset number of options. This does not only apply to explicit VIC, but also to some recent unsupervised learning algorithms that adopt a discrete option or skill with a predefined set (Machado et al., 2017; Eysenbach et al., 2018). Obviously, preset number of options limits the number of options that agent can learn and hence it limits the maximal level of empowerment. Moreover, choosing a proper number of options is not straightforward, since the maximum of the objective for a given number of options depends on several unknown environmental factors such as the cardinality of the state space and the transi-

tional model. To overcome this issue, Gregor et al. (2016) proposed implicit VIC which defines the option as the trajectory until the termination (Gregor et al., 2016), i.e., the sequence of states and actions until termination. There exist multiple trajectories that lead to the same final state and implicit VIC learns to maximize the mutual information between final state and trajectory by controlling these options (trajectories). As a result, the number of options is no longer limited by the preset number. However, despite using the closed options, implicit VIC fails to achieve maximal empowerment under stochastic environment dynamics, which it aimed for indeed. To recover the maximal empowerment under stochastic dynamics, we revisit variational intrinsic control as follows:

1. We show that the intrinsic reward in implicit VIC suffers from the variational bias in stochastic environments, causing convergence to suboptimal solutions (Section 2).
2. To compensate this bias and achieve the maximal mutual information between final state and trajectory (empowerment), we suggest two modifications of implicit VIC: the environment dynamics modeling with the transitional probability (Section 3) and Gaussian Mixture Model (Section 4).

2 Variational Bias of implicit VIC in stochastic environments

In this section, we derive the variational bias of implicit VIC under stochastic environment dynamics. First, we adopt the definition of termination action and final state from Gregor et al. (2016) for a clear explanation. The termination action terminates the option and yields the final state $s_f = s_t$ independently of the environmental action space. The objective term in VIC, i.e., the mutual information between option Ω and final state s_f, can be written as follows:

$$I(\Omega, s_f|s_0) = -\sum_{\Omega} p(\Omega|s_0) \log p(\Omega|s_0) + \sum_{\Omega, s_f} p(s_f|s_0, \Omega) p(\Omega|s_0) \log p(\Omega|s_0, s_f). \tag{1}$$

Since $p(\Omega|s_0, s_f)$ is intractable, VIC Gregor et al. (2016) derives the variational bound $I^{VB} \leq I$ and optimizes it instead:

$$I^{VB}(\Omega, s_f|s_0) = -\sum_{\Omega} p(\Omega|s_0) \log p(\Omega|s_0) + \sum_{\Omega, s_f} p(s_f|s_0, \Omega) p(\Omega|s_0) \log q(\Omega|s_0, s_f), \tag{2}$$

where $q(\Omega|s_0, s_f)$ is the inference model to be trained. When I^{VB} is maximized, we have $p(\Omega|s_0, s_f) = q(\Omega|s_0, s_f)$ and achieve the maximum I. Explicit VIC has an additional action space of options. An option is sampled at s_0 and then it conditions the policy π until the termination of the option, $\pi(a|s, \Omega)$. Explicit VIC aims to control s_f using Ω while it maximizes the information of s_f.

Unlike explicit VIC, implicit VIC defines the option Ω as the trajectory of an agent, i.e., the sequence of states and actions: $\Omega = (s_0, a_0, s_1, a_1, ..., s_{T-1}, a_{T-1} = a_f, s_T = s_f = s_{T-1})$. Implicit VIC aims to control s_f using its trajectories while it maximizes the information of s_f. In this case, the probability of an option can be decomposed as

$$p(\Omega|s_0) = \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} p(a_t|\tau_t) p(s_{t+1}|\tau_t, a_t) \tag{3}$$

using Bayes rule with $\tau_t = (s_0, a_0, ..., s_t)$. Similarly, $p(\Omega|s_0, s_f)$ can be expressed as

$$p(\Omega|s_0, s_{f|\Omega}) = \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} p(a_t|\tau_t, s_{f|\Omega}) p(s_{t+1}|\tau_t, a_t, s_{f|\Omega}). \tag{4}$$

Note that s_f is replaced by $s_{f|\Omega}$ since it is determined by the given Ω in the following way:

$$p(s_f|\Omega, s_0) = \begin{cases} 1, & \text{if } s_f = s_{f|\Omega} \\ 0, & \text{otherwise} \end{cases}. \tag{5}$$

This is a key characteristic of implicit VIC and is essential for deriving main results of this paper. Using (3), (4) and (5), we can rewrite the mutual information (1) as

$$I(\Omega, s_f|s_0) = \sum_{\Omega} p(\Omega|s_0) \sum_{(\tau_t, a_t, s_{t+1}) \in \Omega} \Big[\log \frac{p(a_t|\tau_t, s_{f|\Omega})}{p(a_t|\tau_t)} + \log \frac{p(s_{t+1}|\tau_t, a_t, s_{f|\Omega})}{p(s_{t+1}|\tau_t, a_t)} \Big]. \tag{6}$$

The intrinsic reward of implicit VIC (Gregor et al., 2016) is given by

$$r_{\Omega}^{IVB} = \sum_{(\tau_t, a_t, s_{t+1}) \in \Omega} \log \frac{q(a_t|\tau_t, s_{f|\Omega})}{p(a_t|\tau_t)}, \tag{7}$$

where $q(a_t|\tau_t, s_f)$ is inference and $p(a_t|\tau_t)$ is policy of an agent. It can be shown that r_{Ω}^{IVB} comes from the first part of (6) (see Appendix A for details). Under deterministic environment dynamics, $\log p(s_{t+1}|\tau_t, a_t, s_{f|\Omega})/p(s_{t+1}|\tau_t, a_t)$ is canceled out since both the nominator and denominator are always 1, turning (6) into the expectation of (7) with variational inference. However, this is not possible under the stochastic environment dynamics and it yields the variational bias b_{Ω}^{VIC} in the intrinsic reward:

$$b_{\Omega}^{VIC} = \sum_{(\tau_t, a_t, s_{t+1}) \in \Omega} \log \frac{p(s_{t+1}|\tau_t, a_t, s_{f|\Omega})}{p(s_{t+1}|\tau_t, a_t)}. \tag{8}$$

In Section 5, we provide the experimental evidence that this variational bias leads to a suboptimal training. Even though the original VIC (Gregor et al., 2016) subtracts $b(s_0)$ from r_{Ω}^{IVB} to reduce the variance of learning, it cannot compensate this bias since it also depends on Ω. In the next section, we analyze the mutual information (1) in more detail under stochastic environment dynamics and define the variational estimate of (1), I^{VE}, for training.

3 Implicit VIC with transitional probability model

In this section, we analyze $I(\Omega, s_f|s_0)$ under stochastic environment dynamics and propose to explicitly model transitional probabilities to fix variational bias. First, for a given option and final state, we define $p_\pi(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_f, s_0)$, $p_\pi(\Omega|s_0)$ and $p_\pi(\Omega|s_0)$ as follows:

$$\begin{aligned} p_\pi(\Omega|s_f, s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} p_\pi(a_t|\tau_t, s_f), \quad & p_\rho(\Omega|s_f, s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} p_\rho(s_{t+1}|\tau_t, a_t, s_f), \\ p_\pi(\Omega|s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} p_\pi(a_t|\tau_t), & p_\rho(\Omega|s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} p_\rho(s_{t+1}|\tau_t, a_t). \end{aligned} \tag{9}$$

Note that $p(\Omega|s_f, s_0) = p_\pi(\Omega|s_f, s_0)p_\rho(\Omega|s_f, s_0)$ is the true distribution of Ω for given s_f where $p_\pi(\Omega|s_f, s_0)$ is policy related part and $p_\rho(\Omega|s_f, s_0)$ is transitional part of $p(\Omega|s_f, s_0)$ and so do $p(\Omega|s_0)$, $p_\pi(\Omega|s_0)$ and $p_\rho(\Omega|s_0)$. Since the variational bias is induced from transitional probabilities, it is necessary to consider them. Hence we model (9) as follows:

$$\begin{aligned} \pi^q(\Omega|s_f, s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} \pi^q(a_t|\tau_t, s_f), \quad & \rho^q(\Omega|s_f, s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} \rho^q(s_{t+1}|\tau_t, a_t, s_f), \\ \pi^p(\Omega|s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} \pi^p(a_t|\tau_t), & \rho^p(\Omega|s_0) &= \prod_{(\tau_t, a_t, s_{t+1}) \in \Omega} \rho^p(s_{t+1}|\tau_t, a_t), \end{aligned} \tag{10}$$

where π^q, ρ^q, π^p and ρ^p are our models to be trained. Since we know the policy of an agent, we have $p_\pi(a_t|\tau_t) = \pi^p(a_t|\tau_t)$. For the other probabilities, they are trained based on our algorithms. Now we can rewrite $I(\Omega, s_f|s_0)$ as below using (9):

$$I(\Omega, s_f|s_0) = \sum_{\Omega, s_f} p(\Omega, s_f|s_0) \Big[\log p_\rho(\Omega|s_f, s_0)p_\pi(\Omega|s_f, s_0) - \log p_\rho(\Omega|s_0)p_\pi(\Omega|s_0) \Big]. \tag{11}$$

Using (10), we define I^{VE} as follows:

$$I^{VE}(\Omega, s_f|s_0) = \sum_{\Omega, s_f} p(\Omega, s_f|s_0) \Big[\log \rho^q(\Omega|s_f, s_0)\pi^q(\Omega|s_f, s_0) - \log \rho^p(\Omega|s_0)\pi^p(\Omega|s_0) \Big]. \tag{12}$$

This is an estimate of the mutual information between Ω and s_f with transitional models. Note that I^{VE} is not a variational lower bound on I unlike I^{VB}, hence the derivation of implicit VIC in Gregor et al. (2016) is not applicable in this case. To tackle this problem, we start from absolute difference, $|I - I^{VE}|$ which can be bounded as

$$\begin{aligned} \left| I - I^{VE} \right| \leq U^{VE} = &\sum_{s_f} p(s_f|s_0) D_{\mathrm{KL}} \big[p_\pi(\cdot|s_f, s_0)p_\rho(\cdot|s_f, s_0) \| \pi^q(\cdot|s_f, s_0)\rho^q(\cdot|s_f, s_0) \big] \\ &+ D_{\mathrm{KL}} \big[p_\pi(\cdot|s_0)p_\rho(\cdot|s_0) \| \pi^p(\cdot|s_0)\rho^p(\cdot|s_0) \big]. \end{aligned} \tag{13}$$

Algorithm 1 Implicit VIC with transitional probability model

Initialize s_0, η, T_{train}, θ^q_π, θ^q_ρ, θ^p_π and θ^p_ρ.
for i_{train} : 1 to T_{train} **do**
  Follow $\pi^p(a_t|\tau_t)$ result in $\Omega = (s_0, a_0, ..., s_f)$.
  $r^{I^{VE}}_\Omega \leftarrow \sum_t [\log \pi^q(a_t|\tau_t, s_f) - \log \pi^p(a_t|\tau_t)]$ ▷ from (15)
  $r^{I^{VE}}_\Omega \leftarrow r^{I^{VE}}_\Omega + \sum_t [\log \rho^q(s_{t+1}|\tau_t, a_t, s_f) - \log \rho^p(s_{t+1}|\tau_t, a_t)]$ ▷ from (15), (*)
  Update each parameter:
    $\theta^p_\pi \leftarrow \theta^p_\pi + \eta r^{I^{VE}}_\Omega \nabla_{\theta^p_\pi} \sum_t \log \pi^p(a_t|\tau_t)$ ▷ from (15)
    $\theta^q_\pi \leftarrow \theta^q_\pi + \eta \nabla_{\theta^q_\pi} \sum_t \log \pi^q(a_t|\tau_t, s_f)$ ▷ from (14)
    $\theta^p_\rho \leftarrow \theta^p_\rho + \eta \nabla_{\theta^p_\rho} \sum_t \log \rho^p(s_{t+1}|\tau_t, a_t)$ ▷ from (14), (*)
    $\theta^q_\rho \leftarrow \theta^q_\rho + \eta \nabla_{\theta^q_\rho} \sum_t \log \rho^q(s_{t+1}|\tau_t, a_t, s_f)$ ▷ from (14), (*)
  end
end for

See Appendix B for the derivation. This upper bound implies that $I^{VE} \rightarrow I$ as ($\pi^q(\Omega|s_f, s_0)$, $\rho^q(\Omega|s_f, s_0)$, $\rho^p(\Omega|s_0)$) $\rightarrow$ ($p_\pi(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_0)$) for all Ω. (Note that $p_\pi(\Omega|s_0)$ is omitted since we know the true value of it.) In other words, our estimate of the mutual information converges to the true value as our estimates (10) converge to the true distribution (9). It makes sense that we can estimate the true value of the mutual information if we know the true distribution. Furthermore, $U^{VE} = 0$ if and only if $(\pi^q(\Omega|s_f, s_0), \rho^q(\Omega|s_f, s_0), \rho^p(\Omega|s_0)) = p_\pi(\Omega|s_f, s_0), p_\rho(\Omega|s_f, s_0), p_\rho(\Omega|s_0))$ for all Ω since U^{VE} is sum of positively weighted KL divergences. This means that minimizing U^{VE} will make (10) converge to (9). If π^q, ρ^q and π^p are parameterized by θ^q_π, θ^q_ρ and θ^p_π, we can obtain $\nabla_{\theta^q_\pi} U^{VE}$, $\nabla_{\theta^q_\rho} U^{VE}$ and $\nabla_{\theta^p_\rho} U^{VE}$ using (5):

$$
\begin{aligned}
\nabla_{\theta^q_\pi} U^{VE} &= -\sum_\Omega p(\Omega|s_0) \nabla_{\theta^q_\pi} \log \pi^q(\Omega|s_{f|\Omega}, s_0), \\
\nabla_{\theta^q_\rho} U^{VE} &= -\sum_\Omega p(\Omega|s_0) \nabla_{\theta^q_\rho} \log \rho^q(\Omega|s_{f|\Omega}, s_0), \\
\nabla_{\theta^p_\rho} U^{VE} &= -\sum_\Omega p(\Omega|s_0) \nabla_{\theta^p_\rho} \log \rho^p(\Omega|s_0),
\end{aligned}
\tag{14}
$$

which can be estimated from sample mean. Once we have ($\pi^q(\Omega|s_f, s_0)$, $\rho^q(\Omega|s_f, s_0)$, $\rho^p(\Omega|s_0)$) $\approx$ ($p_\pi(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_0)$) for all Ω, we can update the policy to maximize I. The gradients, $\nabla_{\theta^p_\pi} I$ and $\nabla_{\theta^p_\pi} I^{VE}$ can be obtained in the following form using (5):

$$
\begin{aligned}
\nabla_{\theta^p_\pi} I &= \sum_\Omega p(\Omega|s_0) \underbrace{\Big[\log p_\pi(\Omega|s_{f|\Omega}, s_0) p_\rho(\Omega|s_{f|\Omega}, s_0) - \log \pi^p(\Omega|s_0) p_\rho(\Omega|s_0)\Big]}_{r^I_\Omega} \nabla_{\theta^p_\pi} \log \pi^p(\Omega|s_0), \\
\nabla_{\theta^p_\pi} I^{VE} &= \sum_\Omega p(\Omega|s_0) \underbrace{\Big[\log \pi^q(\Omega|s_{f|\Omega}, s_0) \rho^q(\Omega|s_{f|\Omega}, s_0) - \log \pi^p(\Omega|s_0) \rho^p(\Omega|s_0)\Big]}_{r^{I^{VE}}_\Omega} \nabla_{\theta^p_\pi} \log \pi^p(\Omega|s_0),
\end{aligned}
\tag{15}
$$

where θ^p_π is parameter of $\pi^p(\Omega|s_0)$ and $p_\pi(\Omega|s_0)$ is replaced by $\pi^p(\Omega|s_0)$ since we know the true value of policy (see Appendix A for details). From (15), we can see that $\nabla_{\theta^p_\pi} I^{VE} \rightarrow \nabla_{\theta^p_\pi} I$ as ($\pi^q(\Omega|s_f, s_0)$, $\rho^q(\Omega|s_f, s_0)$, $\rho^p(\Omega|s_0)$) $\rightarrow$ ($p_\pi(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_f, s_0)$, $p_\rho(\Omega|s_0)$) for all Ω. Note that for deterministic environments, we can omit $\rho^q(\Omega|s_f, s_0)$ and $\rho^p(\Omega|s_0)$ since they are always 1 and it satisfies $I^{VE} = I^{VB}$ of (2) and $\nabla_{\theta^q_\pi} U^{VE} = -\nabla_{\theta^q_\pi} I^{VB}$, which means that maximizing I^{VB} is equivalent to minimizing U^{VE} for θ^q_π (i.e., it is equivalent to the original implicit VIC). Algorithm 1 summarizes the modified implicit VIC with the transitional probability model. The additional steps added to the original implicit VIC are marked with $(*)$ in Algorithm 1. Note that Algorithm 1 is not always practically applicable since it is hard to model $p(s_{t+1}|\tau_t, a_t)$ and $p(s_{t+1}|\tau_t, a_t, s_f)$ when the cardinality of the state space is unknown. (We can not set the number of nodes for *softmax*.) In our experiment of Algorithm 1, we will assume that we know the cardinality of the state space. This allows us to model $p_\rho(s_{t+1}|\tau_t, a_t)$ and $p_\rho(s_{t+1}|\tau_t, a_t, s_f)$ using $softmax$. In the next section, we propose a practically applicable method that avoids this intractability of the cardinality.

4 IMPLICIT VIC WITH GAUSSIAN MIXTURE MODEL

In this section, we propose the alternative method to overcome the limitation of Algorithm 1, modeling the smoothed transitional distributions. This allows us to use the Gaussian Mixture Model (GMM) (Pearson, 1894) and other continuous distributional models. First, we smooth $p(s_{t+1}|\tau_t, a_t, s_f)$ and $p(s_{t+1}|\tau_t, a_t)$ into $f_\sigma(x_{t+1}|\tau_t, a_t, s_f)$ and $f_\sigma(x_{t+1}|\tau_t, a_t)$ with $x_{t+1} = s_{t+1} + z_{t+1}$ and $z_{t+1} \sim \mathcal{N}(\mathbf{0}, \sigma^2 \mathrm{I}_n)$:

$$\begin{aligned} f_\sigma(x_{t+1}|\tau_t, a_t, s_f) &= \sum_{s' \in S(\tau_t, a_t, s_f)} p(s'|\tau_t, a_t, s_f) f_\sigma(x_{t+1} - s'; \mathbf{0}, \sigma^2 \mathrm{I}_n), \\ f_\sigma(x_{t+1}|\tau_t, a_t) &= \sum_{s' \in S(\tau_t, a_t)} p(s'|\tau_t, a_t) f_\sigma(x_{t+1} - s'; \mathbf{0}, \sigma^2 \mathrm{I}_n), \end{aligned} \tag{16}$$

where $S(\tau_t, a_t, s_f) = \{s' | p(s'|\tau_t, a_t, s_f) > 0\}$, $S(\tau_t, a_t) = \{s' | p(s'|\tau_t, a_t) > 0\}$ and n is the dimension of the state. Then, using Gaussian Mixture Model (GMM) (Pearson, 1894), we model (16) as $f^q_\sigma(x_{t+1}|\tau_t, a_t, s_f)$ and $f^p_\sigma(x_{t+1}|\tau_t, a_t)$:

$$\begin{aligned} f^q_\sigma(x_{t+1}|\tau_t, a_t, s_f) &= \sum_{i=1}^{n_{gmm}} w_i(\tau_t, a_t, s_f) f_\sigma(x_{t+1}; \boldsymbol{\mu}_i(\tau_t, a_t, s_f), \sigma^2 \mathrm{I}_n), \\ f^p_\sigma(x_{t+1}|\tau_t, a_t) &= \sum_{i=1}^{n_{gmm}} w_i(\tau_t, a_t) f_\sigma(x_{t+1}; \boldsymbol{\mu}_i(\tau_t, a_t), \sigma^2 \mathrm{I}_n). \end{aligned} \tag{17}$$

Note that if we set $n_{gmm} > \max_{\tau_t, a_t} |S(\tau_t, a_t)|$, (17) can perfectly fit (16). Now using (17), we define I^{VE}_σ, the variational estimate with smoothing as follows:

$$I^{VE}_\sigma = \sum_{\Omega, s_f} p(\Omega, s_f|s_0) \Big[\log \pi^q(\Omega|s_f, s_0) f^q_\sigma(\Omega|s_f, s_0) - \log \pi^p(\Omega|s_0) f^p_\sigma(\Omega|s_0) \Big]. \tag{18}$$

Note that I^{VE}_σ is not a variational lower bound on I, hence we can not also apply derivation of implicit VIC in Gregor et al. (2016). The upper bound on $|I - I^{VE}_\sigma|$ can be obtained as follows:

$$\begin{aligned} &\left| I - I^{VE}_\sigma \right| \le U^{VE}_{\sigma,1} + U^{VE}_{\sigma,2} \quad \text{with} \\ &U^{VE}_{\sigma,1} = \left| \sum_{\Omega, s_f} p(\Omega, s_f|s_0) \log \Big[\frac{p_\rho(\Omega|s_f, s_0)}{p_\rho(\Omega|s_0)} \frac{f^p_\sigma(\Omega|s_0)}{f^q_\sigma(\Omega|s_f, s_0)} \Big] \right|, \\ &U^{VE}_{\sigma,2} = \sum_{s_f} p(s_f|s_0) D_{\mathrm{KL}} \big[p_\pi(\cdot|s_f, s_0) p_\rho(\cdot|s_f, s_0) || \pi^q(\cdot|s_f, s_0) p_\rho(\cdot|s_f, s_0) \big]. \end{aligned} \tag{19}$$

See Appendix C for the derivation. This upper bound implies $U^{VE}_{\sigma,1} \to 0$ as $f^q_\sigma(\Omega|s_f, s_0)/f^p_\sigma(\Omega|s_0) \to p_\rho(\Omega|s_f, s_0)/p_\rho(\Omega|s_0)$ and $U^{VE}_{\sigma,2} \to 0$ as $\pi^q(\Omega|s_f, s_0) \to p_\pi(\Omega|s_f, s_0)$ for all Ω. The correct value of mutual information can be estimated from (18) by minimizing $U^{VE}_{\sigma,1}$ and $U^{VE}_{\sigma,2}$ and we can directly minimize $U^{VE}_{\sigma,2}$ using gradient descent as (14) using (5):

$$\nabla_{\theta^q_\pi} U^{VE}_{\sigma,2} = -\sum_{\Omega} p(\Omega|s_0) \nabla_{\theta^q_\pi} \log \pi^q(\Omega|s_{f|\Omega}, s_0). \tag{20}$$

This can be estimated from the sample mean. Since $U^{VE}_{\sigma,2} = 0$ if and only if $\pi^q(\Omega|s_f, s_0) = p_\pi(\Omega|s_f, s_0)$ for all Ω, this update will make $\pi^q(\Omega|s_f, s_0)$ converge to $p_\pi(\Omega|s_f, s_0)$.

Unlike $U^{VE}_{\sigma,2}$, it is difficult to directly minimize $U^{VE}_{\sigma,1}$ due to the absolute symbol. Therefore, we instead minimize $D_{\mathrm{KL}}[f_\sigma(\cdot|\tau_t, a_t, s_f) \| f^q_\sigma(\cdot|\tau_t, a_t, s_f)]$ and $D_{\mathrm{KL}}[f_\sigma(\cdot|\tau_t, a_t) \| f^p_\sigma(\cdot|\tau_t, a_t)]$ to fit (17) to (16). If f^q_σ and f^p_σ are parameterized by θ^q_ρ and θ^p_ρ, the gradients of KL divergences can be obtained as follows:

$$\begin{aligned} \nabla_{\theta^q_\rho} D_{\mathrm{KL}}[f_\sigma(\cdot|\tau_t, a_t, s_f) \| f^q_\sigma(\cdot|\tau_t, a_t, s_f)] &= -\int_{x_{t+1}} f_\sigma(x_{t+1}|\tau_t, a_t, s_f) \nabla_{\theta^q_\rho} \log f^q_\sigma(x_{t+1}|\tau_t, a_t, s_f), \\ \nabla_{\theta^p_\rho} D_{\mathrm{KL}}[f_\sigma(\cdot|\tau_t, a_t) \| f^q_\sigma(\cdot|\tau_t, a_t)] &= -\int_{x_{t+1}} f_\sigma(x_{t+1}|\tau_t, a_t) \nabla_{\theta^p_\rho} \log f^p_\sigma(x_{t+1}|\tau_t, a_t), \end{aligned} \tag{21}$$

Algorithm 2 Implicit VIC with Gaussian mixture model

Initialize s_0, η, T_{train}, T_{smooth}, θ^q_π, θ^q_ρ, θ^p_π and θ^p_ρ.
for i_{train} : 1 to T **do**
  Follow $\pi^p(a_t|\tau_t)$ result in $\Omega = (s_0, a_0, ..., s_f)$.
  $r_\Omega^{I_\sigma^{VE}} \leftarrow \sum_t [\log \pi^q(a_t|\tau_t, s_f) - \log \pi^p(a_t|\tau_t)]$ ▷ from (22)
  $r_\Omega^{I_\sigma^{VE}} \leftarrow r_\Omega^{I_\sigma^{VE}} + \sum_t [\log f^q_\sigma(s_{t+1}|\tau_t, a_t, s_f) - \log f^p_\sigma(s_{t+1}|\tau_t, a_t)]$ ▷ from (22), (*)
  Update each parameter:
    $\theta^p_\pi \leftarrow \theta^p_\pi + \eta r_\Omega^{I_\sigma^{VE}} \nabla_{\theta^p_\pi} \sum_t \log \pi^p(a_t|\tau_t)$ ▷ from (22)
    $\theta^q_\pi \leftarrow \theta^q_\pi + \eta \nabla_{\theta^q_\pi} \sum_t \log \pi^q(a_t|\tau_t, s_f)$ ▷ from (20)
    $\Delta\theta^p_\rho \leftarrow 0$ ▷ (*)
    $\Delta\theta^q_\rho \leftarrow 0$ ▷ (*)
    for i_{smooth} : 1 to T_{smooth} **do** ▷ (*)
      Sample $(z_1, z_2, ..., z_f), z_i \sim \mathcal{N}(\mathbf{0}, \sigma^2 I_n)$
      $\Delta\theta^p_\rho \leftarrow \Delta\theta^p_\rho + \eta \nabla_{\theta^p_\rho} \sum_t \log f^p_\sigma(s_{t+1} + z_{t+1}|\tau_t, a_t)$ ▷ from (21)
      $\Delta\theta^q_\rho \leftarrow \Delta\theta^q_\rho + \eta \nabla_{\theta^q_\rho} \sum_t \log f^q_\sigma(s_{t+1} + z_{t+1}|\tau_t, a_t, s_f)$ ▷ from (21)
    end for
    $\theta^p_\rho \leftarrow \theta^p_\rho + \Delta\theta^p_\rho / T_{smooth}$ ▷ (*)
    $\theta^q_\rho \leftarrow \theta^q_\rho + \Delta\theta^q_\rho / T_{smooth}$ ▷ (*)
  end
end for

which can be estimated from the sample mean. These updates will make $(f^q_\sigma(\Omega|s_f, s_0), f^p_\sigma(\Omega|s_0))$ converge to $(f_\sigma(\Omega|s_f, s_0), f_\sigma(\Omega|s_0))$. As $(f^q_\sigma(\Omega|s_f, s_0), f^p_\sigma(\Omega|s_0)) \to (f_\sigma(\Omega|s_f, s_0), f_\sigma(\Omega|s_0))$, it satisfies $U^{VE}_{\sigma,1} \to 0$ for finite $\overline{T}$ and $\sigma \ll d_{min}$ where $\overline{T}$ is the average length of trajectories and d_{min} is the minimal distance between two different states (see appendix D for detail). Hence for small enough σ and finite length of trajectories, we can minimize $U^{VE}_{\sigma,1}$ to nearly zero. Once we have $(\pi^q(\Omega|s_f, s_0), f^q_\sigma(\Omega|s_f, s_0), f^p_\sigma(\Omega|s_0)) \approx (p_\pi(\Omega|s_f, s_0), f_\sigma(\Omega|s_f, s_0), f_\sigma(\Omega|s_0))$ after the update by (20) and (21), we get $I^{VE}_\sigma \approx I$. Now, we can obtain $\nabla_{\theta^p_\pi} I^{VE}_\sigma$ using (5):

$$\nabla_{\theta^p_\pi} I^{VE}_\sigma = \sum_\Omega p(\Omega|s_0) \underbrace{\left[\log \frac{\pi^q(\Omega|s_{f|\Omega}, s_0)}{\pi^p(\Omega|s_0)} + \log \frac{f^q_\sigma(\Omega|s_{f|\Omega}, s_0)}{f^p_\sigma(\Omega|s_0)}\right]}_{r_\Omega^{I_\sigma^{VE}}} \nabla_{\theta^p_\pi} \log \pi^p(\Omega|s_0). \tag{22}$$

Note that $\nabla_{\theta^p_\pi} I^{VE}_\sigma \to \nabla_{\theta^p_\pi} I$ as $(\pi^q(\Omega|s_f, s_0),\ f^q_\sigma(\Omega|s_f, s_0)/f^p_\sigma(\Omega|s_0)) \to (p_\pi(\Omega|s_f, s_0),\ p_\rho(\Omega|s_f, s_0)/p_\rho(\Omega|s_0))$. Also, it satisfies $f_\sigma(\Omega|s_f, s_0)/f_\sigma(\Omega|s_0)) \approx p_\rho(\Omega|s_f, s_0)/p_\rho(\Omega|s_0))$ for finite $\overline{T}$ and $\sigma \ll d_{min}$ (see appendix D for detail). Hence it satisfies $\nabla_{\theta^p_\pi} I^{VE}_\sigma \to \nabla_{\theta^p_\pi} I$ as $(f^q_\sigma(\Omega|s_f, s_0),\ f^p_\sigma(\Omega|s_0)) \to (f_\sigma(\Omega|s_f, s_0),\ f_\sigma(\Omega|s_0))$. However, choosing a small enough σ is not straightforward since d_{min} and $\overline{T}$ depend on the environment. Too small σ makes the training of f^q_σ and f^p_σ unstable due to its extreme gradient near the means. Another issue of GMM is the choice of a proper n_{gmm} of (17). We may choose a very large n_{gmm} for the perfect fit but it makes training hard for its complexity. We leave the proper choice of σ and n_{gmm} as future works and use empirically chosen values ($\sigma = 0.25$ and $n_{gmm} = 10$) in this paper. We summarize our method in Algorithm 2. Additional steps added to the original implicit VIC are marked with $(*)$.

5 EXPERIMENTS

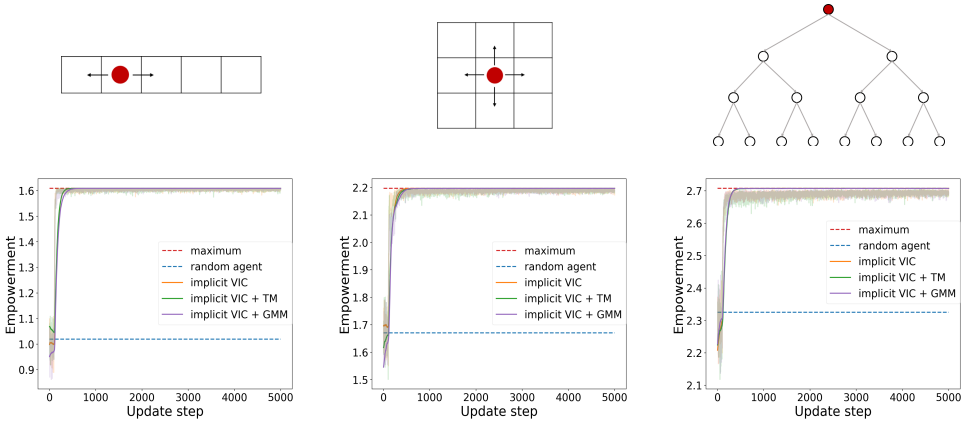
In this section, we evaluate implicit VIC (Gregor et al., 2016), Algorithm 1 and Algorithm 2. We use LSTM (Hochreiter & Schmidhuber, 1997) to encode $\tau_t = (s_0, a_0, ..., s_t)$ into a vector. We conduct experiments on both deterministic and stochastic environments and evaluate results by measuring the mutual information I from samples. To measure I, we rewrite (1) using (5):

$$I(\Omega, s_f|s_0) = -\sum_{s_f} p(s_f|s_0) \log p(s_f|s_0)$$

which is maximized when s_f is distributed uniformly. We estimate $\hat{I}$ using the distribution of s_f from the samples, i.e., $\hat{p}(s_f|s_0)$. We apply exponential moving average (0.99 as a smoothing factor)

to an average of 5 repetitions for estimating $\hat{p}(s_f|s_0)$. We manually set T_{max} (maximum length of a trajectory) for each experiment such that the termination action is the only available action at T_{max}th action. For training GMM of Algorithm 2, transitional models are trained to predict the distribution of $\Delta x_{t+1} = x_{t+1} - s_t$ instead of x_{t+1} since it is easier to learn for model. For the training, we have a warm-up phase which trains the base function $b(s_0)$ in Gregor et al. (2016) and the transitional models. After the warm-up phase, we update the base function, policy and transitional models simultaneously. Please see Appendix E.1 for details on the hyper-parameter settings.

We compare the algorithms in deterministic environments. Note that although (8) is zero for deterministic environments, we still train the transitional models of Algorithm 1 and 2 to show the convergence to the optimum of them. Fig. 1 shows that all three algorithms rapidly achieve the maximal empowerment. Fig. 2 shows the training results of implicit VIC and Algorithm 2 in the 25×25 grid world with 4 rooms used in Gregor et al. (2016). Both implicit VIC and our Algorithm 2 successfully learn passing narrow doors between rooms.



(a) Deterministic 1D world with $T_{max} = 5$.

(b) Deterministic 2D world with $T_{max} = 5$.

(c) Deterministic tree world with $T_{max} = 4$.

Figure 1: **Deterministic environments.** Fig. 1a shows the deterministic 1D world and its training results. The agent can go left right. Fig. 1b shows the deterministic 2D world and its training results. The agent can go left, up, right and down. Fig 1c shows the deterministic tree world and its training results. The agent can go left and right.

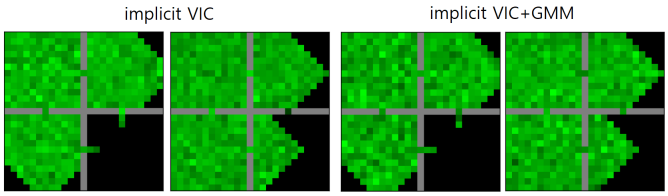
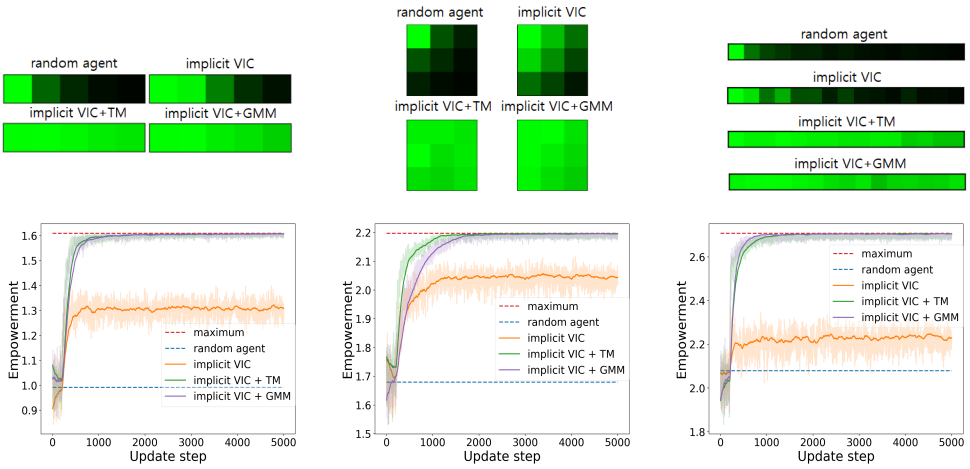


Figure 2: **Deterministic grid world with 4 rooms.** The environment is a 25×25 grid world with 4 rooms. The agent can go left, up, right and down. We set $T_{max} = 25$. Green shows the distribution of s_f.

We compare the algorithms in stochastic environments. Please see appendix E.2 for the details on the stochasticity of the environments. Fig. 3 shows results in simple stochastic environments. We can see that while implicit VIC converges to sub-optimum, our two algorithms achieve the maximal empowerment. In Fig. 4, implicit VIC fails to reach far rooms, however, our Algorithm 2 reaches every room in the environment. From 3 and 4, we can notice that implicit VIC fails to reach far states under the stochastic dynamics. It happens because the variational bias of implicit VIC is accumulated throughout a trajectory, i.e., it gets larger as the length of the trajectory increases. Fig.

4 also shows the results of training with external rewards. The same mixed reward is used as Gregor et al. (2016), $r = r^I + \alpha r^E$ with $\alpha = 30$. For training the random agent, only αr^E is used and the entropy loss was applied for exploration. While the random agent and implicit VIC converge to sub-optimum, our Algorithm 2 achieves the optimal solution.



(a) Stochastic 1D world with $T_{max} = 5$. Algorithm 1 and 2 achieve ∼196% empowerment gain compared to implicit VIC.

(b) Stochastic 2D world with $T_{max} = 5$. Algorithm 1 and 2 achieve ∼142% empowerment gain compared to implicit VIC.

(c) Stochastic tree world with $T_{max} = 4$. Algorithm 1 and 2 achieve ∼406% empowerment gain compared to implicit VIC.

Figure 3: **Estimated empowerment during the training in stochastic environments.** The environments are equal to Fig. 1 except for their stochasticity. Green shows the distribution of s_f.

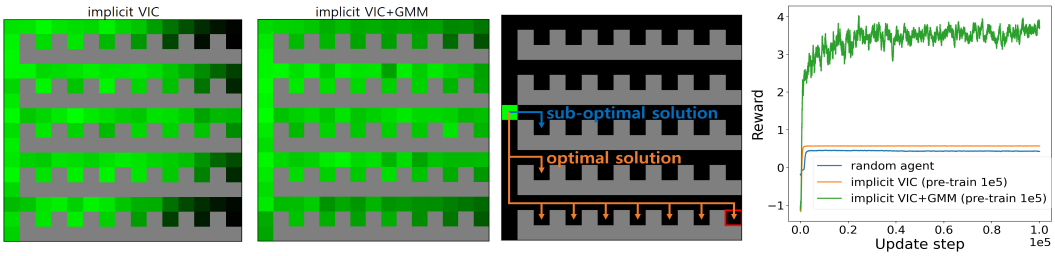


Figure 4: **Stochastic gird world with 35 rooms.** The environment is a 15×15 grid world with 35 rooms (black cells surrounded by gray walls). We set $T_{max} = 25$. The agent can go up, down and right. It starts from (0, 6). Once the agent enters a room, the environment returns done and then the final action is available only. Green shows the distribution of s_f. The external reward is composed of -0.1 for every time step, +1 for entering the normal room and +100 for entering the special room. The sub-optimal solution is reaching the closest room. The optimal solution is trying to reach the special room (the room with the red box) while it enters the closest normal room as soon as possible when it is impossible to reach their due to the stochastic transition.

6 Conclusion

In this work, we revisited variational intrinsic control. We showed that for the VIC with implicit options, the environment stochasticity induces a variational bias in the intrinsic reward, leading to convergence to sub-optimum. To reduce this bias and achieve maximal empowerment, we proposed to model the environment dynamics using either the transitional probability model or the Gaussian mixture model. Evaluations on stochastic environments demonstrated the superiority of our methods over the original VIC algorithm with implicit options (Gregor et al., 2016).

REFERENCES

Suguru Arimoto. An algorithm for computing the capacity of arbitrary discrete memoryless channels. *IEEE Transactions on Information Theory*, 18(1):14–20, 1972.

David Barber and Felix V Agakov. The im algorithm: a variational approach to information maximization. In *Advances in neural information processing systems*, pp. None, 2003.

Richard Blahut. Computation of channel capacity and rate-distortion functions. *IEEE transactions on Information Theory*, 18(4):460–473, 1972.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. In *International Conference on Learning Representations*, 2018.

Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. *arXiv preprint arXiv:1611.07507*, 2016.

Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8): 1735–1780, 1997.

Alexander S Klyubin, Daniel Polani, and Chrystopher L Nehaniv. Empowerment: A universal agent-centric measure of control. In *2005 IEEE Congress on Evolutionary Computation*, volume 1, pp. 128–135. IEEE, 2005.

Marlos C Machado, Marc G Bellemare, and Michael Bowling. A laplacian framework for option discovery in reinforcement learning. In *International Conference on Machine Learning*, pp. 2295–2304, 2017.

Shakir Mohamed and Danilo Jimenez Rezende. Variational information maximisation for intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pp. 2125–2133, 2015.

Karl Pearson. Iii. contributions to the mathematical theory of evolution. *Philosophical Transactions of the Royal Society of London.(A.)*, (185):71–110, 1894.

Christoph Salge, Cornelius Glackin, and Daniel Polani. Empowerment–an introduction. In *Guided Self-Organization: Inception*, pp. 67–114. Springer, 2014.

Raymond W Yeung. The blahut–arimoto algorithms. In *Information Theory and Network Coding*, pp. 211–228. Springer, 2008.

A DERIVATION OF INTRINSIC REWARD FROM MUTUAL INFORMATION

Here we derive the intrinsic reward of VIC by taking the gradient of (1) with respect to the parameter of policy θ. We omit s_0 for simplicity. Note that $p(\Omega), p(s_f)$ and $p(\Omega, s_f)$ can be parameterized by θ since they are all determined by policy. We start by rewriting (1):

$$I(\Omega, s_f) = \sum_{\Omega, s_f} p_\theta(\Omega, s_f)\Big[\log p_\theta(\Omega, s_f) - \log p_\theta(\Omega) p_\theta(s_f)\Big].$$

Then by taking the gradient with respect to θ, we obtain

$$\begin{aligned}
\nabla_\theta I(\Omega, s_f) &= \sum_{\Omega, s_f} p_\theta(\Omega, s_f)\Big[\log p_\theta(\Omega, s_f) - \log p_\theta(\Omega) p_\theta(s_f)\Big] \nabla_\theta \log p_\theta(\Omega, s_f) \\
&\quad + \sum_{\Omega, s_f} p_\theta(\Omega, s_f)\Big[\frac{\nabla_\theta p_\theta(\Omega, s_f)}{p_\theta(\Omega, s_f)} - \frac{\nabla_\theta p_\theta(\Omega)}{p_\theta(\Omega)} - \frac{\nabla_\theta p_\theta(s_f)}{p_\theta(s_f)}\Big] \\
&= \sum_{\Omega, s_f} p_\theta(\Omega, s_f)\Big[\log p_\theta(\Omega, s_f) - \log p_\theta(\Omega) p_\theta(s_f)\Big] \nabla_\theta \log p_\theta(\Omega, s_f) \\
&\quad + \sum_{\Omega, s_f} \Big[\nabla_\theta p_\theta(\Omega, s_f) - p_\theta(s_f|\Omega) \nabla_\theta p_\theta(\Omega) - p_\theta(\Omega|s_f) \nabla_\theta p_\theta(s_f)\Big].
\end{aligned}$$

Using

$$\sum_{\Omega,s_f} \nabla_\theta p_\theta(\Omega, s_f) = 0, \sum_{\Omega,s_f} p_\theta(s_f|\Omega)\nabla_\theta p_\theta(\Omega) = 0, \sum_{\Omega,s_f} p_\theta(\Omega|s_f)\nabla_\theta p_\theta(s_f) = 0,$$

and (5) we have

$$\begin{aligned}\nabla_\theta I(\Omega, s_f) &= \sum_{\Omega,s_f} p_\theta(\Omega, s_f)\Big[\log p_\theta(\Omega, s_f) - \log p_\theta(\Omega)p_\theta(s_f)\Big]\nabla_\theta \log p_\theta(\Omega, s_f) \\ &= \sum_{\Omega} p_\theta(\Omega) r^I_\Omega \nabla_\theta p_\theta(\Omega)\end{aligned} \tag{23}$$

where

$$r^I_\Omega = \log p_\theta(\Omega|s_{f|\Omega}) - \log p_\theta(\Omega).$$

Similarly, we can obtain

$$\begin{aligned}\nabla_\theta I^{VE}(\Omega, s_f) &= \sum_{\Omega,s_f} p_\theta(\Omega, s_f)\Big[\log q(\Omega|s_f) - \log p_\theta(\Omega)\Big]\nabla_\theta \log p_\theta(\Omega, s_f) \\ &= \sum_{\Omega} p_\theta(\Omega) r^{I^{VE}}_\Omega \nabla_\theta p_\theta(\Omega)\end{aligned}$$

where

$$r^{I^{VE}}_\Omega = \log q(\Omega|s_{f|\Omega}) - \log p_\theta(\Omega).$$

Using (3) and (4), we can rewrite r^I_Ω as

$$r^I_\Omega = \sum_{\tau_t,a_t,s_{t+1}\in\Omega} \log \frac{p_\theta(a_t|\tau_t, s_{f|\Omega})p_\theta(s_{t+1}|\tau_t, a_t, s_{f|\Omega})}{p_\theta(a_t|\tau_t)p(s_{t+1}|\tau_t, a_t)}.$$

Since $p_\theta(a_t|\tau_t, s_{f|\Omega})$ is intractable, we may replace it with variational inference $q_\phi(a_t|\tau_t, s_{f|\Omega})$ which result in

$$r^{I^{VE}}_\Omega = \sum_{\tau_t,a_t,s_{t+1}\in\Omega} \log \frac{q_\phi(a_t|\tau_t, s_{f|\Omega})p_\theta(s_{t+1}|\tau_t, a_t, s_{f|\Omega})}{p_\theta(a_t|\tau_t)p(s_{t+1}|\tau_t, a_t)}.$$

For deterministic environment, we have $p_\theta(s_{t+1}|\tau_t, a_t, s_{f|\Omega}) = p(s_{t+1}|\tau_t, a_t) = 1$ and both r^I_Ω and $r^{I^{VE}}_\Omega$ can be reduced into

$$\begin{aligned}r^I_\Omega &= \sum_{\tau_t,a_t,s_{t+1}\in\Omega} \log \frac{p_\theta(a_t|\tau_t, s_{f|\Omega})}{p_\theta(a_t|\tau_t)} \\ r^{I^{VE}}_\Omega &= \sum_{\tau_t,a_t,s_{t+1}\in\Omega} \log \frac{q_\phi(a_t|\tau_t, s_{f|\Omega})}{p_\theta(a_t|\tau_t)} = r^{I^{VB}}_\Omega.\end{aligned}$$

B Derivation of U^{VE}

Here we derive U^{VE} from $|I - I^{VE}|$ with omitted s_0 for simplicity:

$$\begin{aligned}\Big|I - I^{VE}\Big| &= \left|\sum_{\Omega,s_f} p(\Omega, s_f)\Big[\log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p^q_\rho(\Omega|s_f)} - \log \frac{p_\pi(\Omega)p_\rho(\Omega)}{p^p_\pi(\Omega)p^p_\rho(\Omega)}\Big]\right| \\ &\leq \left|\sum_{\Omega,s_f} p(\Omega, s_f) \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p^q_\rho(\Omega|s_f)}\right| + \left|\sum_{\Omega,s_f} p(\Omega, s_f) \log \frac{p_\pi(\Omega)p_\rho(\Omega)}{p^p_\pi(\Omega)p^p_\rho(\Omega)}\right| \\ &= \left|\sum_{\Omega,s_f} p(s_f)p(\Omega|s_f) \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p^q_\rho(\Omega|s_f)}\right| + \left|\sum_{\Omega,s_f} p(\Omega)p(s_f|\Omega) \log \frac{p_\pi(\Omega)p_\rho(\Omega)}{p^p_\pi(\Omega)p^p_\rho(\Omega)}\right|.\end{aligned}$$

Using (5), (9) and (10), we obtain

$$
\begin{aligned}
\left|I - I^{VE}\right| \leq & \left|\sum_{\Omega, s_f} p(s_f)p(\Omega|s_f) \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p^q_\rho(\Omega|s_f)}\right| + \left|\sum_{\Omega, s_f} p(\Omega)p(s_f|\Omega) \log \frac{p_\pi(\Omega)p_\rho(\Omega)}{p^p_\pi(\Omega)p^p_\rho(\Omega)}\right| \\
= & \left|\sum_{\Omega, s_f} p(s_f)p(\Omega|s_f) \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p^q_\rho(\Omega|s_f)}\right| + \left|\sum_{\Omega} p(\Omega) \log \frac{p_\pi(\Omega)p_\rho(\Omega)}{p^p_\pi(\Omega)p^p_\rho(\Omega)}\right| (\because (5)) \\
= & \sum_{s_f} p(s_f) D_{\mathrm{KL}}\left[p_\pi(\cdot|s_f)p_\rho(\cdot|s_f)||p^q_\pi(\cdot|s_f)p^q_\rho(\cdot|s_f)\right] + D_{\mathrm{KL}}\left[p_\pi(\cdot)p_\rho(\cdot)||p^p_\pi(\cdot)p^p_\rho(\cdot)\right] \\
= & U^{VE}.
\end{aligned}
$$

C Derivation of $U^{VE}_{\sigma,1}$ and $U^{VE}_{\sigma,2}$

Here we derive (19). We also omit s_0 for simplicity here. We start from $\left|I - I^{VE}_\sigma\right|$:

$$
\begin{aligned}
\left|I - I^{VE}_\sigma\right| = & \left|\sum_{\Omega, s_f} p(\Omega, s_f)\Big[\log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)f^q_\sigma(\Omega|s_f)} - \log \frac{p_\pi(\Omega)p_\rho(\Omega)}{p^p_\pi(\Omega)f^p_\sigma(\Omega)}\Big]\right| \\
= & \left|\sum_{\Omega, s_f} p(\Omega, s_f)\Big[\log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)f^q_\sigma(\Omega|s_f)} - \log \frac{p_\rho(\Omega)}{f^p_\sigma(\Omega)}\Big]\right| (\because p^p_\pi(\Omega) = p_\pi(\Omega)) \\
= & \left|\sum_{\Omega, s_f} p(\Omega, s_f)\Big[\log \frac{p_\rho(\Omega|s_f)f^p_\sigma(\Omega)}{p_\rho(\Omega)f^q_\sigma(\Omega|s_f)} + \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}\Big]\right| \\
\leq & \left|\sum_{\Omega, s_f} p(\Omega, s_f) \log \frac{p_\rho(\Omega|s_f)f^p_\sigma(\Omega)}{p_\rho(\Omega)f^q_\sigma(\Omega|s_f)}\right| + \left|\sum_{\Omega, s_f} p(\Omega, s_f) \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}\right| \\
= & \left|\sum_{\Omega, s_f} p(\Omega, s_f) \log \frac{p_\rho(\Omega|s_f)f^p_\sigma(\Omega)}{p_\rho(\Omega)f^q_\sigma(\Omega|s_f)}\right| + \left|\sum_{\Omega, s_f} p(s_f)p(\Omega|s_f) \log \frac{p_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}{p^q_\pi(\Omega|s_f)p_\rho(\Omega|s_f)}\right| \\
= & U^{VE}_{\sigma,1} + U^{VE}_{\sigma,2}.
\end{aligned}
$$

D Derivation of the bounds on $I - I_\sigma$

Here we derive the bounds on the estimation error of mutual information with smoothing. First, we derive the upper bound on $f_\sigma(s_{t+1}|\tau_t, a_t)$:

$$
\begin{aligned}
f_\sigma(s_{t+1}|\tau_t, a_t) = & \sum_{s' \in S(\tau_t, a_t)} p(s'|\tau_t, a_t) f_\sigma(s_{t+1} - s'; \mathbf{0}, \sigma^2 \mathbf{I}_n) \\
= & \frac{1}{\sqrt{(2\pi\sigma^2)^n}}\Big(p(s_{t+1}|\tau_t, a_t) + \sum_{s' \neq s_{t+1} \in S(\tau_t, a_t)} p(s'|\tau_t, a_t) \exp\big(-\frac{\left\|s_{t+1} - s'\right\|_2^2}{2\sigma^2}\big)\Big) \\
\leq & \frac{1}{\sqrt{(2\pi\sigma^2)^n}}\Big(p(s_{t+1}|\tau_t, a_t) + \sum_{s' \neq s_{t+1} \in S(\tau_t, a_t)} p(s'|\tau_t, a_t) \exp\big(-\frac{d^2_{min}}{2\sigma^2}\big)\Big) \\
= & \frac{1}{\sqrt{(2\pi\sigma^2)^n}}\Big(p(s_{t+1}|\tau_t, a_t) + (1 - p(s_{t+1}|\tau_t, a_t)) \exp\big(-\frac{d^2_{min}}{2\sigma^2}\big)\Big) \\
\leq & \frac{1}{\sqrt{(2\pi\sigma^2)^n}}\Big(p(s_{t+1}|\tau_t, a_t) + \exp\big(-\frac{d^2_{min}}{2\sigma^2}\big)\Big).
\end{aligned}
$$

Obviously, we have $\frac{1}{\sqrt{(2\pi\sigma^2)^n}} p(s_{t+1}|\tau_t, a_t) \leq f_\sigma(s_{t+1}|\tau_t, a_t)$ which results in:

$$p(s_{t+1}|\tau_t, a_t) \leq \sqrt{(2\pi\sigma^2)^n} f_\sigma(s_{t+1}|\tau_t, a_t) \leq p(s_{t+1}|\tau_t, a_t) + \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right). \tag{24}$$

Similarly, we can obtain the bounds of $f_\sigma(s_{t+1}|\tau_t, a_t, s_f)$ as follows:

$$p(s_{t+1}|\tau_t, a_t, s_f) \leq \sqrt{(2\pi\sigma^2)^n} f_\sigma(s_{t+1}|\tau_t, a_t, s_f) \leq p(s_{t+1}|\tau_t, a_t, s_f) + \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right). \tag{25}$$

Combining (24) and (25), we can obtain

$$\frac{p(s_{t+1}|\tau_t, a_t, s_f)}{p(s_{t+1}|\tau_t, a_t) + \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right)} \leq \frac{f_\sigma(s_{t+1}|\tau_t, a_t, s_f)}{f_\sigma(s_{t+1}|\tau_t, a_t)} \leq \frac{p(s_{t+1}|\tau_t, a_t, s_f) + \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right)}{p(s_{t+1}|\tau_t, a_t, s_f)}. \tag{26}$$

Taking log and using $\log(a+b) \leq \log a + \frac{b}{a}$ for $a, b > 0$ to (26), we have

$$\begin{aligned}\log \frac{p(s_{t+1}|\tau_t, a_t, s_f)}{p(s_{t+1}|\tau_t, a_t)} - \frac{1}{p_{min}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right) \leq \\ \log \frac{f_\sigma(s_{t+1}|\tau_t, a_t, s_f)}{f_\sigma(s_{t+1}|\tau_t, a_t)} \\ \leq \log \frac{p(s_{t+1}|\tau_t, a_t, s_f)}{p(s_{t+1}|\tau_t, a_t)} + \frac{1}{p_{min,f}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right)\end{aligned}$$

which results in

$$-\frac{1}{p_{min,f}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right) \leq \log \frac{p(s_{t+1}|\tau_t, a_t, s_f)}{p(s_{t+1}|\tau_t, a_t)} - \log \frac{f_\sigma(s_{t+1}|\tau_t, a_t, s_f)}{f_\sigma(s_{t+1}|\tau_t, a_t)} \leq \frac{1}{p_{min}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right). \tag{27}$$

Using (9) and (27) we obtain

$$-\frac{T_\Omega}{p_{min,f}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right) \leq \log \frac{p_\rho(\Omega|s_f, s_0)}{p_\rho(\Omega|s_0)} - \log \frac{f_\sigma(\Omega|s_f, s_0)}{f_\sigma(\Omega|s_0)} \leq \frac{T_\Omega}{p_{min}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right).$$

By taking the expectation to each side, we have

$$\begin{aligned}-\frac{T_{max}}{p_{min,f}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right) \leq -\frac{\overline{T}}{p_{min,f}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right) \leq \\ I - I_\sigma \\ \leq \frac{\overline{T}}{p_{min}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right) \leq \frac{T_{max}}{p_{min}} \exp\left(-\frac{d_{min}^2}{2\sigma^2}\right)\end{aligned} \tag{28}$$

with $T_{max} = \max_\Omega T_\Omega$ and $\overline{T} = \sum_{\Omega, s_f} p(\Omega, s_f|s_0) T_\Omega$. This implies that the estimation error of mutual information with smoothing converges to 0 for fine T_{max} or $\overline{T}$ as $\sigma \to 0$. Also, for finite $\overline{T}$ and $\sigma \ll d_{min}$, it satisfies $|I - I_\sigma| \approx 0$.

E EXPERIMENTAL DETAILS

Here we specify experimental details and environment details.

E.1 HYPER-PARAMETERS

Here we show hyper-parameters used in our experiments. We used a learning rate of $1e-4$ for training with external rewards and $1e-3$ otherwise.

Table 1: **Hyper-parameters used for experiments**

Hyper-parameter	**Value**
Optimizer	Adam
Learning rate	1e-3, 1e-4
Betas	(0.9, 0.999)
Weight initialization	Gaussian with std. 0.1 and mean 0
Batch size	128
T_{smooth}	128
σ (GMM)	0.25
n_{gmm} (GMM)	10

E.2 STOCHASTIC ENVIRONMENTS DETAIL

Here we specify the transition tables used in this paper.

Table 2: **Transition table of stochastic environments.**

Action \ Transition	go left	go right
go left	0.7	0.3
go right	0.3	0.7

(a) Transition table of the stochastic 1D world.

Action \ Transition	go left	go up	go down	go right
go left	0.7	0.1	0.1	0.1
go up	0.1	0.7	0.1	0.1
go right	0.1	0.1	0.7	0.1
go down	0.1	0.1	0.1	0.7

(b) Transition table of the stochastic 2D world.

Action \ Transition	go left	go right	no move
go left	0.8	0.0	0.2
go right	0.6	0.2	0.2

(c) Transition table of the stochastic tree world.

Action \ Transition	go up	go down	go right	no move
go up	0.7	0.0	0.0	0.3
go down	0.0	0.7	0.0	0.3
go right	0.0	0.0	0.7	0.3

(d) Transition table of the stochastic grid world with 35 rooms.