A. Variational Inference Derivation for Hidden Markov Models

In this section, we provide the mathematical derivation for the structured variational inference procedure. We focus on the training for Bayesian Hidden Markov Model, in particular the Forward-Backward procedure to complete the description of Algorithm 3. The mathematical details for other types of graphical models depend on the family of such models and should follow similar derivations. Further relevant details on stochastic variational inference can be found in (Hoffman et al., 2013; Johnson & Willsky, 2014; Beal, 2003). d

Settings. Given an arbitrarily ordered set of trajectories $U = \{U_1, \dots, U_K, C\}$, let the coordination mechanism underlying each such U be governed by a true unknown model p, with global parameters θ. We suppress the agent/policy subscript and consider a generic featurized trajectory $x_t = [u_t, c_t]\ \ \forall t$. Let the latent role sequence for the same agent be $z = z_{1:T}$.

At any time t, each agent is acting according to a latent role $z_t \sim \texttt{Categorical}\{\bar{1}, \bar{2}, \dots, \bar{K}\}$, which are the local parameters to the structured model.

Ideally, role and index assignment can be obtained by calculating the posterior $p(z|x, \theta)$, which is often intractable. One way to infer the role assignment is via approximating the intractable posterior $p(z|x, \theta)$ using Bayesian inference, typically via MCMC or mean-field variational methods. Since sampling-based MCMC methods are often slow, we instead aim to learn to approximate $p(z|x, \theta)$ by a simpler distribution q via Bayesian inference. In particular, we employ techniques from stochastic variational inference (Hoffman et al., 2013), which allows for efficient stochastic training on mini-batches that can naturally integrate with our imitation learning subroutine.

Structured Variational Inference for Unsupervised Role Learning. Consider a full probabilistic model:

$$p(\theta, z, x) = p(\theta) \prod_{t=1}^{T} p(z_t|\theta) p(x_t|z_t, \theta)$$

with global latent variables θ, local latent variables $z = \{z_t\}_{t=1}^{T}$. Posterior approximation is often cast as optimizing over a simpler model class $\mathcal{Q}$, via searching for global parameters θ and local latent variables z that maximize the evidence lower bound (ELBO) $\mathcal{L}$:

$$\begin{aligned}\log p(x) &\geqslant \mathbb{E}_q \left[\log p(z, \theta, x)\right] - \mathbb{E}_q \left[\log q(z, \theta)\right] \\ &\triangleq \mathcal{L}\left(q(z, \theta)\right).\end{aligned}$$

Maximizing $\mathcal{L}$ is equivalent to finding $q \in \mathcal{Q}$ to minimize the KL divergence KL $\left(q(z, \theta|x) || p(z, \theta|x)\right)$.

For unsupervised structured prediction problem over a family of graphical model, we focus on the structured mean-field variational family, which factorizes q as $q(z, \theta) = q(z)q(\theta)$ (Hoffman & Blei, 2014) and decomposes the ELBO objective:

$$\begin{aligned}\mathcal{L} = \ &\mathbb{E}_q[\log p(\theta] - \mathbb{E}_q[\log q(\theta] \\ &+ \mathbb{E}_q[\log(p(z, x|\theta)] - \mathbb{E}_q[\log(q(z))]. \qquad (3)\end{aligned}$$

This factorization breaks the dependency between θ and z, but not between single latent states z_t, unlike variational inference for i.i.d data (Kingma & Welling, 2013).

Variational inference optimizes the objective $\mathcal{L}$ typically using natural gradient ascent over global factors $q(\theta)$ and local factors $q(z)$. (Under mean-field assumption, optimization typically proceeds via alternating updates of θ and z.) Stochastic variational inference performs such updates efficiently in mini-batches. For graphical models, structured stochastic variational inference optimizes $\mathcal{L}$ using *natural gradient* ascent over global factors $q(\theta)$ and message-passing scheme over local factors $q(z)$. We assume the prior $p(\theta)$ and complete conditionals $p(z_t, x_t|\theta)$ are conjugate pairs of exponential family, which gives natural gradient of $\mathcal{L}$ with respect to $q(\theta)$ convenient forms (Johnson & Willsky, 2014). Denote the exponential family forms of $p(\theta)$ and $p(z_t, y_t|\theta)$ by:

$$\begin{aligned}\ln p(\theta) &= \langle \eta_\theta, t_\theta(\theta) \rangle - A_\theta(\eta_\theta) \\ \ln p(z_t, x_t|\theta) &= \langle \eta_{zx}(\theta), t_{zx}(z_t, x_t) \rangle - A_{zx}(\eta_{zx}(\theta))\end{aligned}$$

where η_θ and η_{zx} are functions indicating natural parameters, t_θ and t_{zx} are sufficient statistics and $A(\cdot)$ are log-normalizers ((Blei et al., 2017)). Note that in general, different subscripts corresponding to η, t, A indicate different function parameterization (not simply a change in variable value assignment). Conjugacy in the exponential family yields that (Blei et al., 2017):

$$t_\theta(\theta) = [\eta_{zx}(\theta), -A_{zx}(\eta_{zx}(\theta))]$$

and that

$$p(\theta|z_t, x_t) \propto \exp\{\langle \eta_\theta + [t_{zx}(z_t, x_t), 1], t_\theta(\theta) \rangle\} \qquad (4)$$

Conjugacy in the exponential family also implies that the optimal $q(\theta)$ is in the same family (Blei et al., 2017), i.e.

$$q(\theta) = \exp\{\langle \widetilde{\eta}_\theta, t_\theta(\theta) \rangle - A_\theta(\widetilde{\eta}_\theta)\}$$

for some natural parameters $\widetilde{\eta}_\theta$ of $q(\theta)$.

To optimize over global parameters $q(\theta)$, conjugacy in the exponential family allows obtaining convenient expression for the gradient of $\mathcal{L}$ with respect to natural parameters $\widetilde{\eta}_\theta$. The derivation is shown similarly to (Johnson

& Willsky, 2014) and (Blei et al., 2017) - we use simplified notations $\widetilde{\eta} \triangleq \widetilde{\eta}_\theta, \eta \triangleq \eta_\theta, A \triangleq A_\theta$, and $t(z,x) \triangleq \sum_{t=1}^{T} [t_{zx}(z_t,x_t),1]$. Taking advantage of the exponential family identity $\mathbb{E}_{q(\theta)}[t_\theta(\theta)] = \nabla A(\widetilde{\eta})$, the objective $\mathcal{L}$ can be re-written as:

$$\begin{aligned}\mathcal{L} &= \mathbb{E}_{q(\theta)q(z)}\left[\ln p(\theta|z,x) - \ln q(\theta)\right] \\ &= \langle \eta + \mathbb{E}_{q(z)}[t(z,x)], \nabla A(\widetilde{\eta})\rangle - (\langle \widetilde{\eta}, \nabla A(\widetilde{\eta})\rangle - A(\widetilde{\eta}))\end{aligned}$$

Differentiating with respect to $\widetilde{\eta}$, we have that

$$\nabla_{\widetilde{\eta}}\mathcal{L} = \left(\nabla^2 A(\widetilde{\eta})\right)\left(\eta + \mathbb{E}_{q(z)}[t(z,x)] - \widetilde{\eta}\right)$$

The *natural gradient* of $\mathcal{L}$, denoted $\widetilde{\nabla}_{\widetilde{\eta}}$, is defined as $\widetilde{\nabla}_{\widetilde{\eta}} \triangleq \left(\nabla^2 A(\widetilde{\eta})\right)^{-1} \nabla_{\widetilde{\eta}}$. And so the natural gradient of $\mathcal{L}$ can be compactly described as:

$$\widetilde{\nabla}_{\widetilde{\eta}}\mathcal{L} = \eta + \sum_{t=1}^{T} \mathbb{E}_{q(z_t)}\{[t_{zx}(z_t,x_t),1]\} - \widetilde{\eta} \tag{5}$$

Thus a stochastic natural descent update on the global parameters $\widetilde{\eta}_\theta$ proceeds at step n by sampling a mini-batch x_t and taking the global update with step size ρ_n:

$$\widetilde{\eta}_\theta \leftarrow (1-\rho_n)\widetilde{\eta}_\theta + \rho_n(\eta_\theta + b^\top \mathbb{E}_{q*(z_t)}[t(z_t,x_t)]) \tag{6}$$

where b is a vector of scaling factors adjusting for the relative size of the mini-batches. Here the global update assumes optimal local update $q^*(z)$ has been computed. In each step however, the local factors $q^*(z_t)$ are computed with mean field updates and the current value of $q(\theta)$ (analogous to coordinate ascent). In what follows, we provide the derivation for the update rules for Hidden Markov Models, which are the particular instantiation of the graphical model we use to represent the role transition for our multi-agent settings.

Variational factor updates via message passing for Hidden Markov Models. For HMMs, we can view global parameters θ as the parameters of the underlying HMMs such as transition matrix and emission probabilities, while local parameters z govern hidden state assignment at each time step.

Fixing the global parameters, the local updates are based on message passing over the graphical model. The exact mathematical derivation depends on the specific graph structure. The simplest scenario is to assume independence among z_t's, which resembles naive Bayes. We instead focus on Hidden Markov Models to capture first-order dependencies in role transitions over play sequences. In this case, global parameters $\theta = (p_0, P, \phi)$ where $P = [P_{ij}]_{i,j=1}^{K}$ is the transition matrix with $P_{ij} = p(z_t = j|z_{t-1} = i)$, $\phi = \{\phi_i\}_{i=1}^{K}$ are the emission parameters, and p_0 is the initial distribution.

Consider a Bayesian HMM on K latent states. Priors on the model parameters include the initial state distribution p_0, transition matrix P with rows denoted $p_1, \dots, p_K$, and the emission parameters $\phi = \{\phi_i\}_{i=1}^{K}$. In this case we have the global parameters $\theta = (p_0, P, \phi)$. For Hidden Markov Model with observation $x_{1:T}$ and latent sequence $z_{1:T}$, the generative model over the parameters is given by $\phi_i \sim p(\phi)$ (i.i.d from prior), $p_i \sim \text{Dir}(\alpha_i)$, $z_1 \sim p_0, z_{t+1} \sim p_{z_t}$, and $x_t \sim p(x_t|\phi_{z_t})$ (conditional distribution given parameters ϕ). We can also write the transition matrix:

$$P = \begin{bmatrix} p_1 \\ \vdots \\ p_K \end{bmatrix}$$

The Bayesian hierarchical model over the parameters, hidden state sequence $z_{1:T}$, and observation sequence $y_{1:T}$ is

$$\begin{aligned}&\phi_i \overset{\text{iid}}{\sim} p(\phi), p_i \sim \text{Dir}(\alpha_i) \\ &z_1 \sim p_0, z_{t+1} \sim p_{z_t}, x_t \sim p(x_t|\phi_{z_t})\end{aligned}$$

For HMMs, we have a full probabilistic model: $p(z,x|\theta) = p_0(z_1)\prod_{t=1}^{T} p(z_t|z_{t-1},P)p(x_t|z_t,\phi)$. Define the likelihood potential $L_{t,i} = p(x_t|\phi_i)$, the likelihood of the latent sequence, given observation and model parameters, is as follows:

$$\begin{aligned}&p(z_{1:T}|x_{1:T},P,\phi) = \\ &\exp\left(\log p_0(z_1) + \sum_{t=2}^{T} \log P_{z_{t-1},z_t} + \sum_{t=1}^{T} \log L_{t,z_t} - Z\right)\end{aligned} \tag{7}$$

where Z is the normalizing constant. Following the notation and derivation from (Johnson & Willsky, 2014), we denote $p(z_{1:T|x_{1:T},P,\phi}) = \text{HMM}(p_0, P, L)$. Under mean field assumption, we approximate the true posterior $p(P,\phi,z_{1:T}|x_{1:T})$ with a mean field variational family $q(P)q(\phi)q(z_{1:T})$ and update each variational factor in turn while fixing the others.

Fixing the global parameters θ, taking expectation of log of (7), we derive the update rule for $q(z)$ as $q(z_{1:T}) = \text{HMM}(\widetilde{P}, \widetilde{p}_0, \widetilde{L})$ where:

$$\begin{aligned}\widetilde{P}_{j,k} &= \exp\{\mathbb{E}_{q(P)}\ln(P_{j,k})\} \\ \widetilde{p}_{0,k} &= \exp\{\ln \mathbb{E}_{q(p_0)} p_{0,k}\} \\ \widetilde{L}_{t,k} &= \exp\{\mathbb{E}_{q(\phi_k)} \ln(p(x_t|z_t = k))\}\end{aligned}$$

To calculate the expectation with respect to $q(z_{1:T})$, which is necessary for updating other factors, the `Forward-Backward` recursion of HMMs is defined by forward messages F and backward messages B:

$$F_{t,i} = \sum_{j=1}^{K} F_{t-1,j}\widetilde{P}_{j,i}\widetilde{L}_{t,i} \tag{8}$$

$$B_{t,i} = \sum_{j=1}^{K} \widetilde{P}_{i,j}\widetilde{L}_{t+1,j}B_{t+1,j} \tag{9}$$

$$F_{1,i} = p_0(i)$$

$$B_{T,i} = 1$$

As a summary, calculating the gradient w.r.t z yields the following optimal variational distribution over the latent sequence:

$$q^*(z) \propto \exp\Big(\mathbb{E}_{q(P)}[\ln p_0(z_1)] + \sum_{t=2}^{T}\mathbb{E}_{q(P)}[\log P_{z_{t-1},z_t}] + \sum_{t=1}^{T}\mathbb{E}_{q(\phi)}\ln[p(x_t|z_t)]\Big), \tag{10}$$

which gives the local updates for $q^*(z)$, given current estimates of P and ϕ:

$$\widetilde{P}_{j,k} = \exp\left[\mathbb{E}_{q(P)}\ln(P_{j,k})\right] \tag{11}$$

$$\widetilde{p}(x_t|z_t = k) = \exp\left[E_{q(\phi)}\ln p(x_t|x_t = k)\right], \tag{12}$$

for $k = 1, \dots, K$, $t = 1, \dots, T$, and then use $p_0, \widetilde{P}, \widetilde{p}$ to run the forward-backward algorithm to compute the update $q^*(z_t = k)$ and $q^*(z_{t-1} = j, z_t = k)$. The forward-backward algorithm in the local update step takes $O(K^2T)$ time for a chain of length T and K hidden states.

Training to learn model parameters for HMMs. Combining *natural gradient* step with message-passing scheme for HMMs yield specific update rules for learning the model parameters. Again for HMMs, the global parameters are $\theta = (p_0, P, \phi)$ and local variables $z = z_{1:T}$. Assuming the priors on observation parameter $p(\phi_i)$ and likelihoods $p(x_t|\phi_i)$ are conjugate pairs of exponential family distribution for all i, the conditionals $p(\phi_i|x)$ have the form as seen from equation 4:

$$p(\phi_i|x) \propto \exp\{\langle \eta_{\phi_i} + [t_{x,i}(x), 1], t_{\phi_i}(\phi_i)\rangle\}$$

For structured mean field inference, the approximation $q(\theta)$ factorizes as $q(P)q(p_0)q(\phi)$. At each iteration, stochastic variational inference sample a sequence $x_{1:T}$ from the data set (e.g. trajectory from any randomly sampled player) and perform stochastic gradient step on $q(P)q(p_0)q(\phi)$. In order to compute the gradient, we need to calculate expected sufficient statistics w.r.t the optimal factor for $q(z_{1:T})$, which in turns depends on current value of $q(P)q(p_0)q(\phi)$.

Following the notation from (Johnson & Willsky, 2014), we write the prior and mean field factors as

$$p(p_i) = \mathrm{Dir}(\alpha_i), p(\phi_i) \propto \exp\{\langle \eta_{\phi_i}, t_{\phi_i}(\phi_i)\rangle\}$$
$$q(p_i) = \mathrm{Dir}(\widetilde{\alpha}_i), q(\phi_i) \propto \exp\{\langle \widetilde{\eta}_{\phi_i}, t_{\phi_i}(\phi_i)\rangle\}$$

```
Algorithm 5 Coordinated Structure Learning
LearnStructure $\{U_1, \dots, U_K, C, \theta, \rho\} \mapsto q(\theta, z)$
Input: Set of trajectories $U = \{U_k\}_{k=1}^K$. Context $C$
    Previous parameters $\theta = (p_0, \theta^P, \theta^\phi)$, stepsize $\rho$
  $X_k = \{x_{t,k}\}_{t=1}^T = \{[u_{t,k}, c_t]\}\ \forall t, k. X = \{X_k\}_{k=1}^K$
  Local update: Compute $\widetilde{P}$ and $\widetilde{p}$ per equation 11 and 12
    and compute $q(z) = \texttt{Forward-Backward}(X, \widetilde{P}, \widetilde{p})$
  Global update of $\theta$, per equations 16, 17, and 18.
output Updated model $q(\theta, z) = q(\theta)q(z)$
```

Using message passing scheme as per equations (8) and (9), we define the intermediate quantities:

$$\widehat{t}_{x,i} \triangleq \mathbb{E}_{q(z_{1:T})}\sum_{t=1}^{T}\mathbb{I}[z_t = i]t_{x,i}(x_t) = \sum_{t=1}^{T}F_{t,i}B_{t,i}[t_{x,i}(x_t), 1]/Z \tag{13}$$

$$(\widehat{t}_{trans,i})_j \triangleq \mathbb{E}_{q(z_{1:T})}\sum_{t=1}^{T-1}\mathbb{I}[z_t = i, z_{t+1} = j] = \sum_{t=1}^{T-1}F_{t,i}\widetilde{P}_{i,j}\widetilde{L}_{t+1,j}B_{t+1,j}/Z \tag{14}$$

$$(\widehat{t}_{init})_i \triangleq \mathbb{E}_{q(z_{1:T})}\mathbb{I}[z_1 = i] = \widetilde{p}_0 B_{1,i}/Z \tag{15}$$

where $Z \triangleq \sum_{i=1}^K F_{T,i}$ is the normalizing constant, and $\mathbb{I}$ is the indicator function.

Given these expected sufficient statistics, the specific update rules corresponding to the natural gradient step in the natural parameters of $q(P)$, $q(p_0)$, and $q(\phi)$ become:

$$\widetilde{\eta}_{\phi,i} \leftarrow (1 - \rho)\widetilde{\eta}_{\phi,i} + \rho(\eta_{\phi,i} + b^\top\widehat{t}_{x,i}) \tag{16}$$

$$\widetilde{\alpha}_i \leftarrow (1 - \rho)\widetilde{\alpha}_i + \rho(\alpha_i + b^\top\widehat{t}_{trans,i}) \tag{17}$$

$$\widetilde{\alpha}_0 \leftarrow (1 - \rho)\widetilde{\alpha}_0 + \rho(\alpha_0 + b^\top\widehat{t}_{init,i}) \tag{18}$$

B. Experimental Evaluation

B.1. Batch-Version of Algorithm 2 for Predator-Prey

B.2. Visualizing Role Assignment for Soccer

The Gaussian components of latent structure in figure 7 give interesting insight about the latent structure of the demonstration data, which correspond to a popular formation arrangement in professional soccer. Unlike the predator-prey domain, however, the players are sometimes expected to switch and swap roles. Figure 8 displays the tendency that each learning policy k would takes on other roles outside of its dominant mode. Policies indexed $0 - 3$ tend to stay most consistent with the prescribed latent roles. We observe that these also correspond to players with the least variance in their action trajectories. Imitation loss is

Algorithm 6 Multi-Agent Data Aggregation Imitation Learning
`Learn`$(A_1, A_2, \ldots, A_K, C|D)$

```
Input: Ordered actions $A_k = \{a_{t,k}\}_{t=1}^{T}$ $\forall k$, context $\{c_t\}_{t=1}^{T}$
Input: Aggregating data set $D_1, .., D_K$ for each policy
Input: base routine Train$(S, A)$ mapping state to actions
for $t = 0, 1, 2, \ldots, T$ do
    Roll-out $\hat{a}_{t+1,k} = \pi_k(\hat{s}_{t,k})$   $\forall$ agent $k$
    Cross-update for each policy $k \in \{1, \ldots, K\}$
    $\hat{s}_{t+1,k} = \varphi_k\left([\hat{a}_{t+1,1}, \ldots, \hat{a}_{t+1,k}, \ldots, \hat{a}_{t+1,K}, c_{t+1}]\right)$
    Collect expert action $a^*_{t+1,k}$ given state $\hat{s}_{t+1,k}$   $\forall k$
    Aggregate data set $D_k = D_k \cup \{\hat{s}_{t+1,k}, a^*_{t+1,k}\}_{t=0}^{T-1}$
end for
$\pi_k \leftarrow$ Train$(D_k)$
output $K$ new policies $\pi_1, \pi_2, \ldots, \pi_K$
```

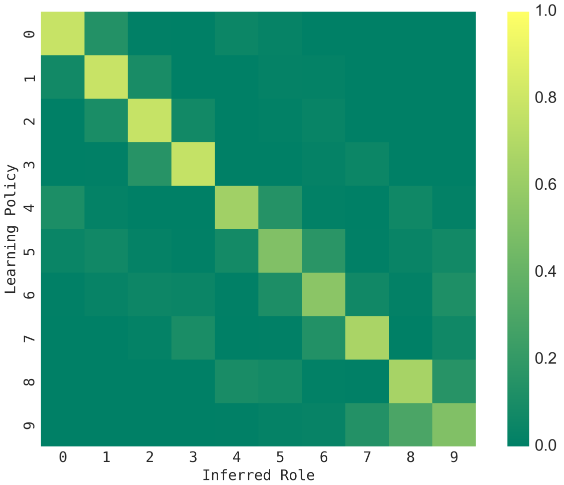

Figure 8. *Role frequency assigned to policy, according to the maximum likelihood estimate of the latent structured model*

generally higher for less consistent roles (e.g. policies indexed $8-9$). Intuitively, entropy regularization encourages a decomposition of roles that result in learning policies as decoupled as possible, in order to minimize the imitation loss.