ARSM: Augment-REINFORCE-Swap-Merge Gradient for Categorical Variables Supplementary Material

A. Derivation of AR, ARS, and ARSM

A.1. Augmentation of a Categorical Variable

Let us denote $\tau \sim \text{Exp}(\lambda)$ as the exponential distribution, with probability density function $p(\tau \,|\, \lambda) = \lambda e^{-\lambda\tau}$, where $\lambda > 0$ and $\tau > 0$. Its mean and variance are $\mathbb{E}[\tau] = \lambda^{-1}$ and $\text{var}[\tau] = \lambda^{-2}$, respectively. It is well known that, e.g. in Ross (2006), if $\tau_i \sim \text{Exp}(\lambda_i)$ are independent exponential random variables for $i = 1, \ldots, C$, then the probability that τ_z, where $z \in \{1, \ldots, C\}$, is the smallest can be expressed as

$$P\big(z = \arg\min_{i\in\{1,\ldots,C\}} \tau_i\big) = P\left(\tau_z < \tau_i, \ \forall\, i \neq z\right) = \tfrac{\lambda_z}{\sum_{i=1}^C \lambda_i} \ . \tag{17}$$

Note this property, referred to as “exponential racing” in Zhang & Zhou (2018), is closely related to the Gumbel distribution (also known as Type-I extreme-value distribution) based latent-utility-maximization representation of multinomial logistic regression (McFadden, 1974; Train, 2009), as well as the Gumbel-softmax trick (Maddison et al., 2017; Jang et al., 2017). This is because the exponential random variable $\tau \sim \text{Exp}(\lambda)$ can be reparameterized as $\tau = \epsilon/\lambda, \ \epsilon \sim \text{Exp}(1)$, where $\epsilon \sim \text{Exp}(1)$ can be equivalently generated as $\epsilon = -\log u, \ u \sim \text{Uniform}(0, 1)$, and hence we have

$$\arg\min_i \tau_i \overset{d}{=} \arg\min_i\{-\log u_i/\lambda_i\} = \arg\max_i\{\log\lambda_i - \log(-\log u_i)\},$$

where $\tau_i \sim \text{Exp}(\lambda_i)$, “$\overset{d}{=}$” denotes “equal in distribution,” and $u_i \overset{iid}{\sim} \text{Uniform}(0, 1)$; note that if $u \sim \text{Uniform}(0, 1)$, then $-\log(-\log u)$ follows the Gumbel distribution (Train, 2009).

From (17) we know that if

$$z = \arg\min_{i\in\{1,\ldots,C\}} \tau_i \text{ , where } \tau_i \sim \text{Exp}(e^{\phi_i}), \tag{18}$$

then $P(z \,|\, \boldsymbol{\phi}) = e^{\phi_z}/\sum_{i=1}^C e^{\phi_i}$, and hence (18) is an augmented representation of the categorical distribution $z \sim \text{Cat}(\sigma(\boldsymbol{\phi}))$; one may consider $\tau_i \sim \text{Exp}(e^{\phi_i})$ as augmented latent variables, the marginalization of which from $z = \arg\min_{i\in\{1,\ldots,C\}} \tau_i$ leads to $P(z \,|\, \boldsymbol{\phi})$. Consequently, the expectation with respect to the categorical variable of C categories can be rewritten as one with respect to C augmented exponential random variables as

$$\mathcal{E}(\boldsymbol{\phi}) = \mathbb{E}_{z\sim\text{Cat}(\sigma(\boldsymbol{\phi}))}[f(z)] = \mathbb{E}_{\tau_1\sim\text{Exp}(e^{\phi_1}),\ldots,\tau_C\sim\text{Exp}(e^{\phi_C})}[f(\arg\min_i \tau_i)]. \tag{19}$$

Since the exponential random variable $\tau \sim \text{Exp}(e^{\phi})$ can be reparameterized as $\tau = \epsilon e^{-\phi}, \ \epsilon \sim \text{Exp}(1)$, we also have

$$\mathcal{E}(\boldsymbol{\phi}) = \mathbb{E}_{\epsilon_1,\ldots,\epsilon_C \overset{iid}{\sim} \text{Exp}(1)}[f(\arg\min_i \epsilon_i e^{-\phi_i})]. \tag{20}$$

Note as the $\arg\min$ operator is non-differentiable, the widely used reparameterization trick (Kingma & Welling, 2013; Rezende et al., 2014) is not applicable to computing the gradient of $\mathcal{E}(\boldsymbol{\phi})$ via the reparameterized representation in (20).

A.2. REINFORCE Estimator in the Augmented Space

Using REINFORCE (Williams, 1992) on (19), we have $\nabla_{\boldsymbol{\phi}}\mathcal{E}(\boldsymbol{\phi}) = [\nabla_{\phi_1}\mathcal{E}(\boldsymbol{\phi}), \ldots, \nabla_{\phi_C}\mathcal{E}(\boldsymbol{\phi})]'$, where

$$\begin{aligned}\nabla_{\phi_c}\mathcal{E}(\boldsymbol{\phi}) &= \mathbb{E}_{\tau_1\sim\text{Exp}(e^{\phi_1}),\ldots,\tau_C\sim\text{Exp}(e^{\phi_C})}\left[f(\arg\min_i \tau_i)\nabla_{\phi_c}\log\textstyle\prod_{i=1}^C \text{Exp}(\tau_i; e^{\phi_i})\right]\\ &= \mathbb{E}_{\tau_1\sim\text{Exp}(e^{\phi_1}),\ldots,\tau_C\sim\text{Exp}(e^{\phi_C})}[f(\arg\min_i \tau_i)\nabla_{\phi_c}\log\text{Exp}(\tau_c; e^{\phi_c})]\\ &= \mathbb{E}_{\tau_1\sim\text{Exp}(e^{\phi_1}),\ldots,\tau_C\sim\text{Exp}(e^{\phi_C})}[f(\arg\min_i \tau_i)(1 - \tau_c e^{\phi_c})].\end{aligned} \tag{21}$$

Below we show how to merge $\nabla_{\phi_c}\mathcal{E}(\boldsymbol{\phi})$ and $-\nabla_{\phi_j}\mathcal{E}(\boldsymbol{\phi})$ by first re-expressing (21) into an expectation with respect to *iid* exponential random variables, swapping the indices of these random variables, and then sharing common random numbers (Owen, 2013) to well control the variance of Monte Carlo integration.

A.3. Merge of Augment-REINFORCE Gradients

A key observation of the paper is we can re-express the expectation in (21) as

$$\nabla_{\phi_c}\mathcal{E}(\boldsymbol{\phi}) = \mathbb{E}_{\epsilon_1,\dots,\epsilon_C \overset{iid}{\sim} \text{Exp}(1)}[f(\arg\min_i \epsilon_i e^{-\phi_i})(1-\epsilon_c)] \tag{22}$$

Furthermore, we note that $\text{Exp}(1) \overset{d}{=} \text{Gamma}(1,1)$, letting $\epsilon_1,\dots,\epsilon_C \overset{iid}{\sim} \text{Exp}(1)$ is the same (e.g., as proved in Lemma IV.3 of Zhou & Carin (2012)) in distribution as letting

$$\epsilon_i = \pi_i \epsilon, \quad \text{for } i = 1,\dots,C, \quad \text{where } \boldsymbol{\pi} \sim \text{Dirichlet }(\mathbf{1}_C),\ \epsilon \sim \text{Gamma}(C,1),$$

and $\arg\min_i \pi_i e^{-\phi_i} = \arg\min_i \epsilon\pi_i e^{-\phi_i}$. Thus using Rao-Blackwellization (Casella & Robert, 1996), we can re-express the gradient in (21) as

$$\begin{aligned}\nabla_{\phi_c}\mathcal{E}(\boldsymbol{\phi}) &= \mathbb{E}_{\epsilon\sim\text{Gamma}(C,1),\ \boldsymbol{\pi}\sim\text{Dirichlet}(\mathbf{1}_C)}[f(\arg\min_i \epsilon\pi_i e^{-\phi_i})(1-\epsilon\pi_c)]\\ &= \mathbb{E}_{\boldsymbol{\pi}\sim\text{Dirichlet}(\mathbf{1}_C)}[f(\arg\min_i \pi_i e^{-\phi_i})(1-C\pi_c)].\\ &= \mathbb{E}_{\boldsymbol{\pi}\sim\text{Dirichlet}(\mathbf{1}_C)}[f(\arg\min_i \pi_i^{c\leftrightharpoons j} e^{-\phi_i})(1-C\pi_j)],\end{aligned} \tag{23}$$

where $j \in \{1,\dots,C\}$ is an arbitrarily selected reference category, whose selection does not depends on $\boldsymbol{\pi}$ and $\boldsymbol{\phi}$.

Another useful observation of the paper is that the function

$$b(\boldsymbol{\pi},\boldsymbol{\phi},j) = \frac{1}{C}\sum_{m=1}^{C} f(\arg\min_i \pi_i^{m\leftrightharpoons j} e^{-\phi_i})(1-C\pi_j)$$

has zero expectation, as

$$\mathbb{E}_{\boldsymbol{\pi}\sim\text{Dirichlet}(\mathbf{1}_C)}[b(\boldsymbol{\pi},\boldsymbol{\phi},j)] = \mathbb{E}_{\boldsymbol{\pi}\sim\text{Dirichlet}(\mathbf{1}_C)}\left[f(\arg\min_i \pi_i e^{-\phi_i})\sum_{m=1}^{C}\left(\frac{1}{C}-\pi_m\right)\right] = 0. \tag{24}$$

Using $\mathbb{E}[b(\boldsymbol{\pi},\boldsymbol{\phi},j)]$ as the baseline function and subtracting it from (23) leads to (8). We now conclude the proof of Theorem 1 for the AR estimator, and Equation 8 for the ARS estimator. Once the ARS estimator is proved, Theorem 2 for the ARSM estimator directly follows.

Proof of Corollary 3. Note that letting $(u, 1-u) \sim \text{Dir}(1,1)$ is the same as letting $u \sim \text{Uniform}(0,1)$. Thus regardless of whether we choose Category 1 or Category 2 for as the reference category, we have

$$\nabla_{\boldsymbol{\phi}_1}\mathcal{E}(\boldsymbol{\phi}) = \mathbb{E}_{u\sim\text{Uniform}(0,1)}[f(\arg\min(u,\sigma(\phi_1-\phi_2)) - f(\arg\min(1-u,\sigma(\phi_1-\phi_2))](1/2-u) \tag{25}$$

and $\nabla_{\boldsymbol{\phi}_2}\mathcal{E}(\boldsymbol{\phi}) = -\nabla_{\boldsymbol{\phi}_1}\mathcal{E}(\boldsymbol{\phi})$. Denote $\phi = \phi_1 - \phi_2$ and $\eta = \phi_1 + \phi_2$, we have

$$\nabla_{\phi}\mathcal{E}(\boldsymbol{\phi}) = \nabla_{\phi_1}\mathcal{E}(\boldsymbol{\phi})\frac{\partial\phi_1}{\partial\phi} + \nabla_{\phi_2}\mathcal{E}(\boldsymbol{\phi})\frac{\partial\phi_2}{\partial\phi} = \nabla_{\boldsymbol{\phi}_1}\mathcal{E}(\boldsymbol{\phi}).$$

□

B. Fast Computation for the Swap Step

Computing the pseudo actions $z^{c\leftrightharpoons j} = \arg\min_i \pi_i^{c\leftrightharpoons j} e^{-\phi_i}$ due to the swap operations can be efficiently realized: we first compute $o_{ij} = \ln\pi_i - \phi_j$, $z = \arg\min_i(\ln\pi_i - \phi_i)$, and $o_{\min} = \ln\pi_z - \phi_z$; then for $m = 1\dots,C,\ j < m$, compute

$$z^{m\leftrightharpoons j} = \begin{cases} m,\ \text{if } z \notin \{m,j\},\ \min\{o_{mj},o_{jm}\} < o_{\min},\ o_{mj} \le o_{jm};\\ j,\ \text{if } z \notin \{m,j\},\ \min\{o_{mj},o_{jm}\} < o_{\min},\ o_{mj} > o_{jm};\\ \arg\min_i(\ln\pi_i^{m\leftrightharpoons j} - \phi_i),\ \text{if } z \in \{m,j\};\\ z,\ \text{otherwise;}\end{cases}$$

and let $z^{j\leftrightharpoons j} = z$ for all j, and $z^{m\leftrightharpoons j} = z^{j\leftrightharpoons m}$ for all $j > m$.

C. ARSM for Multivariate, Hierarchical, and Sequential Categorical Variables

C.1. ARSM for Multivariate Categorical Variables

Proposition 5 (AR, ARS, and ARSM for multivariate categorical)**.** *Denote $\boldsymbol{z} = (z_1, \dots, z_K)$, where $z_k \in \{1, \dots, C\}$, as a K dimensional vector of C-way categorical variables. Denote $\boldsymbol{\Pi} = (\boldsymbol{\pi}_1, \dots, \boldsymbol{\pi}_K) \in \mathbb{R}^{C \times K}$ as a matrix obtained by concatenating K column vectors $\boldsymbol{\pi}_k = (\pi_{k1}, \dots, \pi_{kC})'$, and $\boldsymbol{\Phi} = (\boldsymbol{\phi}_1, \dots, \boldsymbol{\phi}_K) \in \mathbb{R}^{C \times K}$ by concatenating $\boldsymbol{\phi}_k = (\phi_{k1}, \dots, \phi_{kC})'$. With the multivariate AR estimator, the gradient of*

$$\mathcal{E}(\boldsymbol{\Phi}) = \mathbb{E}_{\boldsymbol{z} \sim \prod_{k=1}^K \text{Cat}(z_k; \sigma(\boldsymbol{\phi}_k))}[f(\boldsymbol{z})] \tag{26}$$

with respect to ϕ_{kc} is expressed as

$$\begin{aligned} \nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi}) &= \mathbb{E}_{\boldsymbol{\Pi} \sim \prod_{k=1}^K \text{Dir}(\boldsymbol{\pi}_k; \mathbf{1}_C)}[f(\boldsymbol{z})(1 - C\pi_{kc})], \\ z_k &: = \arg\min\nolimits_{i \in \{1, \dots, C\}} \pi_{ki} e^{-\phi_{ki}}. \end{aligned} \tag{27}$$

Denoting $\boldsymbol{j} = (j_1, \dots, j_K)$, where $j_k \in \{1, \dots, C\}$ is a randomly selected reference category for dimension k, the multivariate ARS estimator is expressed as

$$\begin{aligned} \nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi}) &= \mathbb{E}_{\boldsymbol{\Pi} \sim \prod_{k=1}^K \text{Dir}(\boldsymbol{\pi}_k; \mathbf{1}_C)}[f_\Delta^{c \leftrightharpoons \boldsymbol{j}}(\boldsymbol{\Pi})(1 - C\pi_{kj_k})], \\ f_\Delta^{c \leftrightharpoons \boldsymbol{j}}(\boldsymbol{\Pi}) &: = f(\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}}) - \tfrac{1}{C} \sum\nolimits_{m=1}^C f(\boldsymbol{z}^{m \leftrightharpoons \boldsymbol{j}}), \\ \boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}} &: = (z_1^{c \leftrightharpoons j_1}, z_2^{c \leftrightharpoons j_2}, \dots, z_K^{c \leftrightharpoons j_K}), \\ z_k^{c \leftrightharpoons j_k} &: = \arg\min\nolimits_{i \in \{1, \dots, C\}} \pi_{ki}^{c \leftrightharpoons j_k} e^{-\phi_{ki}}. \end{aligned} \tag{28}$$

Setting $\boldsymbol{j} = j\mathbf{1}_K$ and averaging over all $j \in \{1, \dots, C\}$, the multivariate ARSM estimator is expressed as

$$\nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi}) = \mathbb{E}_{\boldsymbol{\Pi} \sim \prod_{k=1}^K \text{Dir}(\boldsymbol{\pi}_k; \mathbf{1}_C)} \big[\sum\nolimits_{j=1}^C f_\Delta^{c \leftrightharpoons (j\mathbf{1}_K)}(\boldsymbol{\Pi})(\tfrac{1}{C} - \pi_{kj}) \big]. \tag{29}$$

Note to obtain $\nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi})$ for all k and c based on the ARS estimator in (28), we only need to evaluate $f(\boldsymbol{z}^{1 \leftrightharpoons \boldsymbol{j}}), \dots, f(\boldsymbol{z}^{C \leftrightharpoons \boldsymbol{j}})$. Thus regardless of how large K is, to obtain a single Monte Carlo sample estimate of the true gradient, one needs to evaluate the reward function $f(\cdot)$ as few as zero time, which happens when the number of unique vectors in $\{\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}}\}_{c=1,C}$ is one, and as many as C times, which happens when all $\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}}$ are different from each other. Similarly, if the ARSM estimator in (29) is used, the number of times one needs to evaluate $f(\cdot)$ is between zero and $C(C-1)/2 + 1$. In the multivariate setting where $\boldsymbol{z} \in \{1, \dots, C\}^K$, we often choose a relatively small C, such as $C = 10$, but allows K to be as large as necessary, such as $K = 100$. Thus even C^K, the number of unique $\boldsymbol{z}$'s, could be enormous when K is large, both the ARS and ARSM estimators remain computationally efficient; this differs them from estimators, such as the one in Titsias & Lázaro-Gredilla (2015), that are not scalable in the dimension K.

C.2. ARSM for Categorical Stochastic Networks

Let us construct a T-categorical-stochastic-layer network as

$$\begin{aligned} q_{\boldsymbol{\Phi}_{1:T}}(\boldsymbol{z}_{1:T} \,|\, \boldsymbol{x}) = \prod\nolimits_{t=1}^T q(\boldsymbol{z}_t \,|\, \boldsymbol{\Phi}_t),\ \boldsymbol{\Phi}_t := \mathcal{T}_{\boldsymbol{w}_t}(\boldsymbol{z}_{1:t-1}), \\ q(\boldsymbol{z}_t \,|\, \boldsymbol{\Phi}_t) := \prod\nolimits_{k=1}^{K_t} \text{Cat}(z_{tk}; \sigma(\boldsymbol{\phi}_{tk})), \end{aligned} \tag{30}$$

where $\boldsymbol{z}_0 := \boldsymbol{x}$, $\boldsymbol{z}_t := (z_{t1}, \dots, z_{tK_t})' \in \{1, \dots, C\}^{K_t}$ is a K_t-dimensional C-way categorical vector at layer t, $\boldsymbol{\phi}_{tk} := (\phi_{tk1}, \dots, \phi_{tkC})' \in \mathbb{R}^C$ is the parameter vector for dimension k at layer t, $\boldsymbol{\Phi}_t := \big(\boldsymbol{\phi}_{t1}, \dots, \boldsymbol{\phi}_{tK_t}\big) \in \mathbb{R}^{C \times K_t}$, and $\mathcal{T}_{\boldsymbol{w}_t}(\cdot)$ represents a function parameterized by $\boldsymbol{w}_t$ that deterministically transforms $\boldsymbol{z}_{t-1}$ to $\boldsymbol{\Phi}_t$. In this paper, we will define $\mathcal{T}_{\boldsymbol{w}_t}(\cdot)$ with a neural network.

Proposition 6. *For the categorical stochastic network defined in (30), the ARSM gradient of the objective*

$$\mathcal{E}(\boldsymbol{\Phi}_{1:T}) = \mathbb{E}_{\boldsymbol{z}_{1:T} \sim q_{\boldsymbol{\Phi}_{1:T}}(\boldsymbol{z}_{1:T} \,|\, \boldsymbol{x})} [f(\boldsymbol{z}_{1:T})] \tag{31}$$

with respect to $\boldsymbol{w}_t$ can be expressed as $\nabla_{\boldsymbol{w}_t} \mathcal{E}(\boldsymbol{\Phi}_{1:T}) = \nabla_{\boldsymbol{w}_t} \big(\sum_{k=1}^{K_t} \sum_{c=1}^C (\nabla_{\phi_{tkc}} \mathcal{E}(\boldsymbol{\Phi}_{1:T}))\phi_{tkc} \big)$, where

$$\nabla_{\phi_{tkc}} \mathcal{E}(\boldsymbol{\Phi}_{1:T}) = \mathbb{E}_{\boldsymbol{\Pi}_t \sim \prod_{k=1}^{K_t} \text{Dir}(\boldsymbol{\pi}_{tk}; \mathbf{1}_C)} \big[\sum\nolimits_{j=1}^C f_{t\Delta}^{c \leftrightharpoons j}(\boldsymbol{\Pi}_t) \big(\tfrac{1}{C} - \pi_{tkj}\big) \big], \tag{32}$$

where $\boldsymbol{\pi}_{tk} = (\pi_{tk1}, \ldots, \pi_{tkC})'$ *is the Dirichlet distributed probability vector for dimension* k *at layer* t *and*

$$\begin{aligned}
f_{t\Delta}^{c\leftrightharpoons j}(\boldsymbol{\Pi}_t) &:= f(Z_t^{c\leftrightharpoons j}) - \tfrac{1}{C}\textstyle\sum_{m=1}^{C} f(Z_t^{m\leftrightharpoons j}),\\
Z_t^{c\leftrightharpoons j} &:= \{\boldsymbol{z}_{1:t-1}, \boldsymbol{z}_{t:T}^{c\leftrightharpoons j}\},\ \ \boldsymbol{z}_{1:t-1} \sim q_{\boldsymbol{\Phi}_{1:t-1}}(\boldsymbol{z}_{1:t-1} \,|\, \boldsymbol{x}),\\
\boldsymbol{z}_t^{c\leftrightharpoons j} &:= (z_{t1}^{c\leftrightharpoons j}, \ldots, z_{tK_t}^{c\leftrightharpoons j})',\\
z_{tk}^{c\leftrightharpoons j} &:= \arg\min\nolimits_{i\in\{1,\ldots,C\}} \pi_{tki}^{c\leftrightharpoons j} e^{-\phi_{tki}},\\
\boldsymbol{z}_{t+1:T}^{c\leftrightharpoons j} &\sim q_{\boldsymbol{\Phi}_{t+1:T}}(\boldsymbol{z}_{t+1:T} \,|\, \boldsymbol{z}_{1:t-1}, \boldsymbol{z}_t^{c\leftrightharpoons j}).
\end{aligned}$$

C.3. Proofs

Below we show how to generalize Theorem 2 for a univariate categorical variable to Proposition 5 for multivariate categorical variables, and Proposition 6 for hierarchical multivariate categorical variables.

Proof of Proposition 5. For the expectation in (26), since z_k are conditionally independent given $\boldsymbol{\phi}_k$, we have

$$\nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi}) = \mathbb{E}_{\boldsymbol{z}_{\setminus k} \sim \prod_{k'\neq k} \text{Discrete}(z_{k'};\sigma(\boldsymbol{\phi}_{k'}))} \big[\nabla_{\phi_{kc}} \mathbb{E}_{z_k \sim \text{Cat}(\sigma(\boldsymbol{\phi}_k))}[f(\boldsymbol{z})]\big]. \tag{33}$$

Using Theorem 2 to compute the gradient in the above equation directly leads to

$$\nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi}) = \mathbb{E}_{\boldsymbol{z}_{\setminus k} \sim \prod_{k'\neq k} \text{Discrete}(z_{k'};\sigma(\boldsymbol{\phi}_{k'}))} \Big\{ \mathbb{E}_{\boldsymbol{\pi}_k \sim \text{Dirichlet}(\mathbf{1}_C)} \Big[\big(f(\boldsymbol{z}_{\setminus k}, \boldsymbol{z}_k^{c\leftrightharpoons j}) - \frac{1}{C}\sum_{m=1}^{C} f(\boldsymbol{z}_{\setminus k}, \boldsymbol{z}_k^{m\leftrightharpoons j})\big)(1 - C\pi_{kj}) \Big] \Big\}, \tag{34}$$

The term inside $[\cdot]$ of (34) can already be used to estimate the gradient, however, in the worst case scenario that all the elements of $\{\boldsymbol{z}_k^{c\leftrightharpoons j}\}_{j=1,C}$ are different, it needs to evaluate the function $f(\boldsymbol{z}_{\setminus k}, \boldsymbol{z}_k^{c\leftrightharpoons j})$ for $j = 1, \ldots, C$, and hence C times for each k and KC times in total. To reduce computation and simplify implementation, exchanging the order of the two expectations in (34), we have

$$\nabla_{\phi_{kc}} \mathcal{E}(\boldsymbol{\Phi}) = \mathbb{E}_{\boldsymbol{\pi}_k \sim \text{Dirichlet}(\mathbf{1}_C)} \left\{ (1 - C\pi_{kj}) \mathbb{E}_{\boldsymbol{z}_{\setminus k} \sim \prod_{k'\neq k} \text{Discrete}(z_{k'};\sigma(\boldsymbol{\phi}_{k'}))} \left[f(\boldsymbol{z}_{\setminus k}, \boldsymbol{z}_k^{c\leftrightharpoons j}) - \frac{1}{C}\sum_{m=1}^{C} f(\boldsymbol{z}_{\setminus k}, \boldsymbol{z}_k^{m\leftrightharpoons j}) \right] \right\} \tag{35}$$

Note that

$$\begin{aligned}
&\mathbb{E}_{\boldsymbol{z}_{\setminus k} \sim \prod_{k'\neq k} \text{Discrete}(z_{k'};\sigma(\boldsymbol{\phi}_{k'}))} [f(\boldsymbol{z}_{\setminus k}, \boldsymbol{z}_k^{c\leftrightharpoons j})]\\
&= \mathbb{E}_{\boldsymbol{\epsilon}_{\setminus k} \sim \prod_{k'\neq k} \prod_{i=1}^{C} \text{Exp}(\epsilon_{k'i};e^{\phi_{k'i}})} \big[f\big((z_{k'} = \arg\min\nolimits_{i\in\{1,\ldots,C\}} \epsilon_{k'i} e^{-\phi_{k'i}})_{k'\neq k},\ \boldsymbol{z}_k^{c\leftrightharpoons j}\big)\big]\\
&= \mathbb{E}_{\boldsymbol{\epsilon}_{\setminus k} \sim \prod_{k'\neq k} \prod_{i=1}^{C} \text{Exp}(\epsilon_{k'i};e^{\phi_{k'i}})} \big[f\big((z_{k'} = \arg\min\nolimits_{i\in\{1,\ldots,C\}} \epsilon_{k'i}^{(c\leftrightharpoons j)} e^{-\phi_{k'i}})_{k'\neq k},\ \boldsymbol{z}_k^{c\leftrightharpoons j}\big)\big]\\
&= \mathbb{E}_{\boldsymbol{\Pi}_{\setminus k} \sim \prod_{k'\neq k} \text{Dirichlet}(\boldsymbol{\pi}_{k'};\mathbf{1}_C)} \big[f\big((z_{k'} = \arg\min\nolimits_{i\in\{1,\ldots,C\}} \pi_{k'i}^{(c\leftrightharpoons j)} e^{-\phi_{k'i}})_{k'\neq k},\ \boldsymbol{z}_k^{c\leftrightharpoons j}\big)\big]\\
&= \mathbb{E}_{\boldsymbol{\Pi}_{\setminus k} \sim \prod_{k'\neq k} \text{Dirichlet}(\boldsymbol{\pi}_{k'};\mathbf{1}_C)} \big[f\big(\boldsymbol{z}_1^{c\leftrightharpoons j}, \ldots, \boldsymbol{z}_K^{c\leftrightharpoons j}\big)\big]
\end{aligned}$$

Plugging the above equation into (35) leads to a simplified representation as (29) shown in Proposition 5, with which, regardless of the dimensions C, we draw $\boldsymbol{\Pi} = \{\boldsymbol{\pi}_1, \ldots, \boldsymbol{\pi}_K\}$ once to produce correlated $\boldsymbol{z}^{c\leftrightharpoons j}$'s, and evaluate the function $f(\cdot)$ at most C times. □

Proof of Proposition 6. For multi-layer stochastic network $q_{\boldsymbol{\Phi}_{1:T}}(\boldsymbol{z}_{1:T} \,|\, \boldsymbol{x}) = q_{\boldsymbol{\Phi}_1}(\boldsymbol{z}_1 \,|\, \boldsymbol{x}) \Big[\prod_{t=1}^{T-1} q_{\boldsymbol{\Phi}_{t+1}}(\boldsymbol{z}_{t+1} \,|\, \boldsymbol{z}_t)\Big]$, the gradient of the t-th layer parameter $\boldsymbol{\Phi}_t$ is

$$\nabla_{\boldsymbol{\Phi}_t} \mathcal{E}(\boldsymbol{\Phi}_{1:T}) = \mathbb{E}_{\boldsymbol{z}_{1:t-1} \sim q(\boldsymbol{z}_{1:t-1}|\boldsymbol{x})} \nabla_{\boldsymbol{\Phi}_t} \mathbb{E}_{q(\boldsymbol{z}_t|\boldsymbol{z}_{t-1})} f_t(\boldsymbol{z}_{1:t})$$

where $f_t(\boldsymbol{z}_{1:t}) = \mathbb{E}_{q(\boldsymbol{z}_{t+1:T}|\boldsymbol{z}_t)}[f(\boldsymbol{z}_{1:T})]$. To compute the ARSM gradient estimator, first draw a single sample $\boldsymbol{z}_{1:t-1} \sim q(\boldsymbol{z}_{1:t-1} \,|\, \boldsymbol{x})$ if $t > 1$ and compute the pseudo action vector for the t-th layer according to Proposition 5 as

$$z_{tk}^{c\leftrightharpoons j} := \arg\min\nolimits_{i\in\{1,\ldots,C\}} \pi_{tki}^{c\leftrightharpoons j} e^{-\phi_{tki}}$$

for $c, j \in \{1, \ldots, C\}$. For each pseudo action vector $\boldsymbol{z}_t^{c\leftrightharpoons j}$, sample $\boldsymbol{z}_{t+1:T}^{c\leftrightharpoons j} \sim q(\boldsymbol{z}_{t+1:T} \,|\, \boldsymbol{z}_t^{c\leftrightharpoons j})$ and compute $f_t(\boldsymbol{z}^{c\leftrightharpoons j}) = f(\boldsymbol{z}_{1:t-1}, \boldsymbol{z}_{t:T}^{c\leftrightharpoons j})$. Replacing $f(\boldsymbol{z}^{c\leftrightharpoons j})$ in Proposition 5 with the $f_t(\boldsymbol{z}^{c\leftrightharpoons j})$ leads to the gradient estimator in Proposition 6. □

Proof of Proposition 4. We first write the objective function $J(\boldsymbol{\theta})$ in terms of the intermediate parameters $\boldsymbol{\phi}_t = \mathcal{T}_{\boldsymbol{\theta}}(\boldsymbol{s}_t)$, and then apply the chain rule to obtain the policy gradient $\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta})$. Since

$$J(\boldsymbol{\phi}_{0:\infty}) = \mathbb{E}_{\mathcal{P}(\boldsymbol{s}_0)\prod_{t=0}^{\infty}\mathcal{P}(\boldsymbol{s}_{t+1}\,|\,\boldsymbol{s}_t,a_t)\text{Cat}(a_t;\sigma(\boldsymbol{\phi}_t))}\left[\sum_{t=0}^{\infty}\gamma^t r(\boldsymbol{s}_t,a_t)\right]$$

we have

$$\begin{aligned}
J(\boldsymbol{\phi}_{0:\infty}) &= \mathbb{E}_{\mathcal{P}(\boldsymbol{s}_0)[\prod_{t'=0}^{t-1}\mathcal{P}(\boldsymbol{s}_{t'+1}\,|\,\boldsymbol{s}_{t'},a_{t'})\text{Cat}(a_{t'};\sigma(\boldsymbol{\phi}_{t'}))]}\left\{\mathbb{E}_{a_t\sim\text{Cat}(\sigma(\boldsymbol{\phi}_t))}\left[\sum_{t'=0}^{t-1}\gamma^{t'}r(s_{t'},a_{t'})+\gamma^t Q(\boldsymbol{s}_t,a_t)\right]\right\}\\
&= \mathbb{E}_{\mathcal{P}(\boldsymbol{s}_0)[\prod_{t'=0}^{t-1}\mathcal{P}(\boldsymbol{s}_{t'+1}\,|\,\boldsymbol{s}_{t'},a_{t'})\text{Cat}(a_{t'};\sigma(\boldsymbol{\phi}_{t'}))]}\left\{\mathbb{E}_{a_t\sim\text{Cat}(\sigma(\boldsymbol{\phi}_t))}\left[\sum_{t'=0}^{t-1}\gamma^{t'}r(s_{t'},a_{t'})\right]\right\}\\
&\quad+\mathbb{E}_{\mathcal{P}(\boldsymbol{s}_0)[\prod_{t'=0}^{t-1}\mathcal{P}(\boldsymbol{s}_{t'+1}\,|\,\boldsymbol{s}_{t'},a_{t'})\text{Cat}(a_{t'};\sigma(\boldsymbol{\phi}_{t'}))]}\left\{\mathbb{E}_{a_t\sim\text{Cat}(\sigma(\boldsymbol{\phi}_t))}\left[\gamma^t Q(\boldsymbol{s}_t,a_t)\right]\right\},
\end{aligned}\tag{36}$$

where $Q(\boldsymbol{s}_t,a_t)$ is the discounted action-value function defined as

$$Q(\boldsymbol{s}_t,a_t) := \mathbb{E}_{\prod_{t'=t}^{\infty}\text{Cat}(a_{t'+1};\sigma(\boldsymbol{\phi}_{t'+1}))\mathcal{P}(\boldsymbol{s}_{t'+1}\,|\,\boldsymbol{s}_{t'},a_{t'})}\left[\sum_{t'=t}^{\infty}\gamma^{t'-t}r(\boldsymbol{s}_{t'},a_{t'})\right].$$

The first summation term in (36) can be ignored for computing $\nabla_{\boldsymbol{\phi}_t}J(\boldsymbol{\phi}_{0:\infty})$, and the second one can be re-expressed as

$$\mathbb{E}_{\mathcal{P}(\boldsymbol{s}_t\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})\mathcal{P}(\boldsymbol{s}_0)}\left\{\mathbb{E}_{a_t\sim\text{Cat}(\sigma(\boldsymbol{\phi}_t))}\left[\gamma^t Q(\boldsymbol{s}_t,a_t)\right]\right\},\tag{37}$$

where $\mathcal{P}(\boldsymbol{s}_t\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})$ is the marginal form of the joint distribution $\prod_{t'=0}^{t-1}\mathcal{P}(\boldsymbol{s}_{t'+1}\,|\,\boldsymbol{s}_{t'},a_{t'})\text{Cat}(a_{t'};\sigma(\boldsymbol{\phi}_{t'}))$. Applying Theorem 2 to (37), we have

$$\begin{aligned}
\nabla_{\boldsymbol{\phi}_{tc}}J(\boldsymbol{\phi}_{0:\infty}) &= \mathbb{E}_{\mathcal{P}(\boldsymbol{s}_t\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})\mathcal{P}(\boldsymbol{s}_0)}\left\{\gamma^t\nabla_{\boldsymbol{\phi}_{tc}}\mathbb{E}_{a_t\sim\text{Cat}(\sigma(\boldsymbol{\phi}_t))}\left[Q(\boldsymbol{s}_t,a_t)\right]\right\}\\
&= \mathbb{E}_{\mathcal{P}(\boldsymbol{s}_t\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})\mathcal{P}(\boldsymbol{s}_0)}\left\{\gamma^t\mathbb{E}_{\boldsymbol{\varpi}_t\sim\text{Dir}(\mathbf{1}_C)}\left[g_{tc}\right]\right\},
\end{aligned}\tag{38}$$

where

$$\begin{aligned}
g_{tc} &:= \sum_{j=1}^{C} f_{t\Delta}^{c\leftrightharpoons j}(\boldsymbol{\varpi}_t)\left(\frac{1}{C}-\varpi_{tj}\right),\\
f_{t\Delta}^{c\leftrightharpoons j}(\boldsymbol{\varpi}_t) &:= Q(s_t,a_t^{c\leftrightharpoons j})-\frac{1}{C}\sum_{m=1}^{C}Q(s_t,a_t^{m\leftrightharpoons j}),\\
a_t^{c\leftrightharpoons j} &:= \arg\min\nolimits_{i\in\{1,\dots,C\}}\varpi_{ti}^{c\leftrightharpoons j}e^{-\phi_{ti}}.
\end{aligned}$$

Applying the chain rule, we obtain the gradient as

$$\begin{aligned}
\nabla_{\boldsymbol{\theta}}J(\boldsymbol{\theta}) &= \sum_{t=0}^{\infty}\sum_{c=1}^{C}\frac{\partial J(\boldsymbol{\phi}_{0:\infty})}{\partial\phi_{tc}}\frac{\partial\phi_{tc}}{\partial\boldsymbol{\theta}}\\
&= \sum_{t=0}^{\infty}\sum_{c=1}^{C}\mathbb{E}_{\mathcal{P}(\boldsymbol{s}_0)\mathcal{P}(\boldsymbol{s}_t\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})}\left\{\gamma^t\mathbb{E}_{\boldsymbol{\varpi}_t\sim\text{Dir}(\mathbf{1}_C)}\left[g_{tc}\right]\nabla_{\boldsymbol{\theta}}\phi_{tc}\right\}\\
&= \sum_{t=0}^{\infty}\mathbb{E}_{\mathcal{P}(\boldsymbol{s}_0)\mathcal{P}(\boldsymbol{s}_t\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})}\left\{\gamma^t\mathbb{E}_{\boldsymbol{\varpi}_t\sim\text{Dir}(\mathbf{1}_C)}\left[\nabla_{\boldsymbol{\theta}}\sum_{c=1}^{C}g_{tc}\phi_{tc}\right]\right\}\\
&= \mathbb{E}_{\boldsymbol{s}_t\sim\rho_\pi(\boldsymbol{s})}\left\{\mathbb{E}_{\boldsymbol{\varpi}_t\sim\text{Dir}(\mathbf{1}_C)}\left[\nabla_{\boldsymbol{\theta}}\sum_{c=1}^{C}g_{tc}\phi_{tc}\right]\right\},
\end{aligned}\tag{39}$$

where $\rho_\pi(\boldsymbol{s}) := \sum_{t=0}^{\infty}\gamma^t\mathcal{P}(\boldsymbol{s}_t=\boldsymbol{s}\,|\,\boldsymbol{s}_0,\pi_{\boldsymbol{\theta}})$ is the unnormalized discounted state visitation frequency. This concludes the proof of the ARSM policy gradient estimator. The proof of the ARS policy gradient estimator can be similarly derived, omitted here for brevity. □

D. Additional Figures and Tables

Table 3. The constructions of variational auto-encoders. The following symbols "→", "]",)", and "⇝" represent deterministic linear transform, leaky rectified linear units (LeakyReLU) (Maas et al., 2013) nonlinear activation, softmax nonlinear activation, and discrete stochastic activation, respectively, in the encoder; their reversed versions are used in the decoder.

	One layer	Two layers
Encoder	784→512]→256]→200)⇝200	784→512]→256]→200)⇝200 → 200) ⇝200
Decoder	784⇜(784←[512←[256←200	784⇜(784←[512←[256←200 ⇜ (200 ← 200

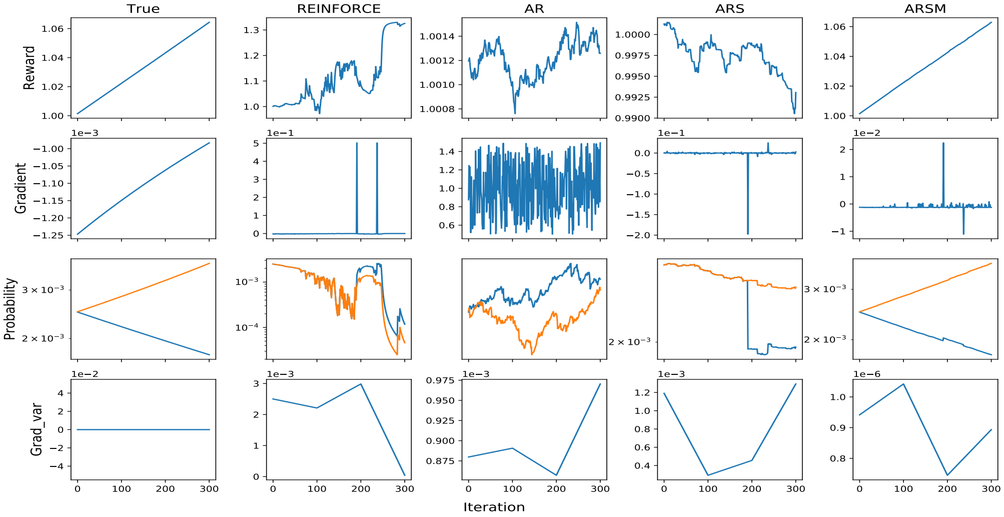


Figure 4. Analogous plots to these in Figure 1, obtained with $C = 1,000$.

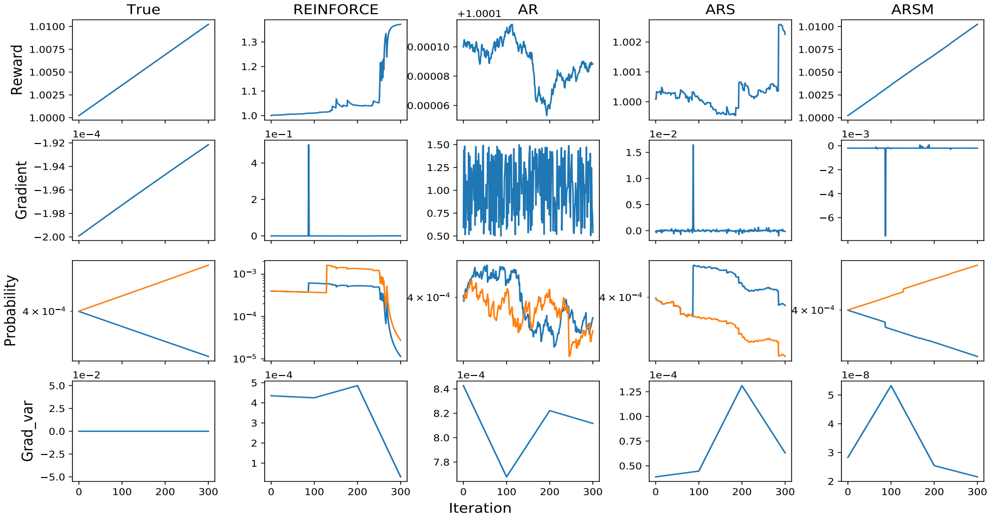


Figure 5. Analogous plots to these in Figure 1, obtained with $C = 10,000$.

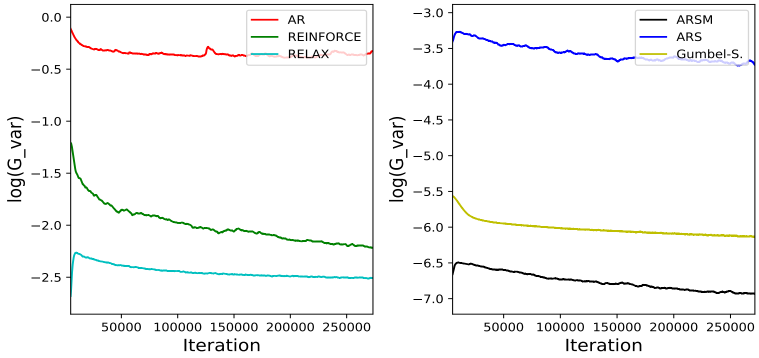
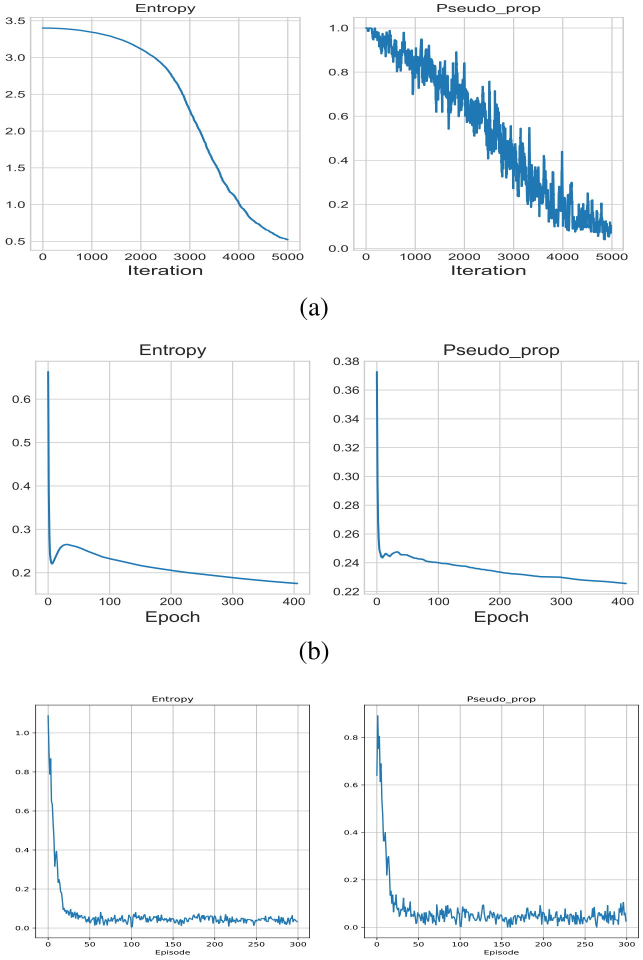


Figure 6. Trace plots of the log variance of the gradient estimators for categorical VAE on MNIST. The variance is estimated by exponential moving averages of the first and second moments with a decay factor of 0.999. The variance is averaged over all elements of the gradient vector.



(a)

(b)

(c)

Figure 7. The entropy of latent categorical distributions and the number of distinct pseudo actions, which differ from their corresponding true actions, both decrease as the training progresses. We plot the average entropy for $\{z_{tk}\}$ for all $t = 1 : T$ and $k = 1 : K$. The pseudo action proportion for the k-th categorical random variable at the t-th stochastic layer is calculated as the number of unique values in $\{z_{tk}^{c \leftrightharpoons j}\}_{c=1:C, j=1:C} \setminus z_{tk}$ divided by $C - 1$, the maximum number of distinct pseudo actions that differ from the true action z_{tk}. We plot the average pseudo action proportion for $\{z_{tk}\}$ for all $t = 1 : T$ and $k = 1 : K$. Subplots (a), (b), and (c) correspond to the Toy data ($T = K = 1$, $C = 30$), VAE with a single stochastic layer ($T = 1$, $K = 20$, $C = 10$), and Acrobot RL task ($0 \leq T \leq 500$, $K = 1$, $C = 3$); other settings yield similar trace plots.

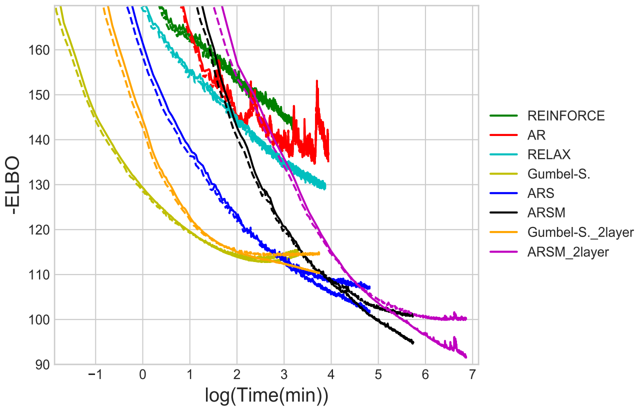


Figure 8. Plots of $-$ELBOs (nats) on binarized MNIST against wall clock times on NVIDIA Tesla V100 GPU (analogous ones against training iterations are shown in Figure 2). The solid and dash lines correspond to the training and testing respectively (best viewed in color).

E. Algorithm

Algorithm 1 ARS/ARSM gradient for K-dimensional C-way categorical vector $\boldsymbol{z} = (z_1, \cdots, z_K)$, where $z_k \in \{1, \dots, C\}$.

```
input  : Reward function $f(\boldsymbol{z}; \boldsymbol{\theta})$ parameterized by $\boldsymbol{\theta}$;
output : Distribution parameter $\boldsymbol{\Phi} = (\boldsymbol{\phi}_1, \cdots, \boldsymbol{\phi}_K) \in \mathbb{R}^{C \times K}$ and reward function parameter $\boldsymbol{\theta}$ that maximize the expected reward as
         $\mathcal{E}(\boldsymbol{\Phi}, \boldsymbol{\theta}) := \mathbb{E}_{\boldsymbol{z} \sim \prod_{k=1}^{K} \text{Cat}(z_k; \sigma(\boldsymbol{\phi}_k))}[f(\boldsymbol{z}; \boldsymbol{\theta})]$;
Initialize $\boldsymbol{\Phi}$ and $\boldsymbol{\theta}$ randomly;
while not converged do
    Sample $\boldsymbol{\pi}_k \sim \text{Dirichlet}(\mathbf{1}_C)$ for $k = 1, \dots, K$;
    Let $z_k = \arg\min_{i \in \{1,\dots,C\}} (\ln \pi_{ki} - \phi_{ki})$ for $k = 1, \dots, K$ to obtain the true action vector $\boldsymbol{z} = (z_1, \dots, z_k)$;
    if Using the ARS estimator then
        Using a single reference vector $\boldsymbol{j} = (j_1, \dots, j_K)$ for the variable-swapping operations, where all $j_k$ are uniformly at random
        selected from $\{1, \dots, C\}$;
        for $c = 1, \dots, C$ (in parallel) do
            Let $z_k^{c \leftrightharpoons j_k} = \arg\min_{i \in \{1,\dots,C\}} (\ln \pi_{ki}^{c \leftrightharpoons j_k} - \phi_{ki})$ for $k = 1, \dots, K$;
            Denote $\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}} = (z_1^{c \leftrightharpoons j_1}, \dots, z_K^{c \leftrightharpoons j_K})$ as the $c$th pseudo action vector;
        end
        Let $\bar{f} = \frac{1}{C} \sum_{c=1}^{C} f(\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}})$
        Let $g_{\phi_{kc}} = \big(f(\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}}) - \bar{f}\big)(1 - C\pi_{kj_k})$ for all $(k, c) \in \{(k, c)\}_{k=1:K,\ c=1:C}$;
    end
    if Using the ARSM estimator then
        Initialize the diagonal of reward matrix $F \in \mathbb{R}^{C \times C}$ with $f(\boldsymbol{z})$, which means letting $F_{cc} = f(\boldsymbol{z})$ for $c = 1, \dots, C$;
        for $(c, j) \in \{(c, j)\}_{c=1:C,\ j<c}$ (in parallel) do
            Let $\boldsymbol{j} = j\mathbf{1}_K$, which means $j_k \equiv j$ for all $k \in \{1, \dots, K\}$;
            Let $z_k^{c \leftrightharpoons j} = \arg\min_{i \in \{1,\dots,C\}} (\ln \pi_{ki}^{c \leftrightharpoons j} - \phi_{ki})$ for $t = 1, \dots, K$;
            Denote $\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}} = (z_1^{c \leftrightharpoons j}, \dots, z_K^{c \leftrightharpoons j})$ as the $(c, j)$th pseudo action vector;
            Let $F_{cj} = F_{jc} = f(\boldsymbol{z}^{c \leftrightharpoons \boldsymbol{j}})$;
        end
        Let $\bar{F}_{\cdot j} = \frac{1}{C} \sum_{c=1}^{C} F_{cj}$ for $j = 1, \dots, C$;
        Let $g_{\phi_{kc}} = \sum_{j=1}^{C} (F_{cj} - \bar{F}_{\cdot j})(\frac{1}{C} - \pi_{kj})$ for all $(t, c) \in \{(t, c)\}_{k=1:K,\ c=1:C}$;
    end
    $\boldsymbol{\Phi} = \boldsymbol{\Phi} + \rho_\phi \{g_{\boldsymbol{\phi}_{kc}}\}_{k=1:T,\ c=1:C}$,     with step-size $\rho_\phi$;
    $\boldsymbol{\theta} = \boldsymbol{\theta} + \eta_\theta \nabla_{\boldsymbol{\theta}} f(\boldsymbol{z}; \boldsymbol{\theta})$,     with step-size $\eta_\theta$
end
*Note if the categorical distribution parameter $\boldsymbol{\Phi}$ itself is defined by neural networks with parameter $\boldsymbol{w}$, standard backpropagation can be
applied to compute the gradient with $\frac{\partial \mathcal{E}(\boldsymbol{\Phi},\boldsymbol{\theta})}{\partial \boldsymbol{w}} = \frac{\partial \mathcal{E}(\boldsymbol{\Phi},\boldsymbol{\theta})}{\partial \boldsymbol{\Phi}} \frac{\partial \boldsymbol{\Phi}}{\partial \boldsymbol{w}} \approx \nabla_{\boldsymbol{w}} \big( \sum_{k=1}^{K} \sum_{c=1}^{C} g_{\boldsymbol{\phi}_{kc}} \phi_{kc} \big)$.
```

Algorithm 2 ARS/ARSM gradient for T layer K-dimensional C-way categorical vector $\boldsymbol{z}_t = (z_{t1}, \cdots, z_{tK})$, where $t \in \{1, \dots, T\}$, $z_{tk} \in \{1, \dots, C\}$.

```
input  : Reward function $f(\boldsymbol{z}_{1:T}; \boldsymbol{\theta})$ parameterized by $\boldsymbol{\theta}$;
output : Distribution parameter $\boldsymbol{\Phi}_t = (\boldsymbol{\phi}_{t1}, \cdots, \boldsymbol{\phi}_{tK})' \in \mathbb{R}^{K \times C}$ and parameter $\boldsymbol{\theta}$ that maximize the expected reward as $\mathcal{E}(\boldsymbol{\Phi}_{1:T}, \boldsymbol{\theta}) := \mathbb{E}_{\boldsymbol{z} \sim q_{\boldsymbol{\Phi}_1}(\boldsymbol{z}_1 \mid \boldsymbol{x})[\prod_{t=1}^{T-1} q_{\boldsymbol{\Phi}_{t+1}}(\boldsymbol{z}_{t+1} \mid \boldsymbol{z}_t)])}[f(\boldsymbol{z}; \boldsymbol{\theta})]$; $q_{\boldsymbol{\Phi}_t}(\boldsymbol{z}_t \mid \boldsymbol{z}_{t-1}) = \prod_{k=1}^{K} \text{Categorical}(z_{tk} | \sigma(\boldsymbol{\phi}_{tk}(\boldsymbol{z}_{t-1})))$;

Initialize $\boldsymbol{\Phi}_{1:T}$ and $\boldsymbol{\theta}$ randomly;
while not converged do
    for t = 1 : T do
        Sample $\boldsymbol{\pi}_{tk} \sim \text{Dirichlet}(\mathbf{1}_C)$ for $k = 1, \dots, K$; Let $z_{tk} = \arg\min_{i \in \{1,\dots,C\}}(\ln \pi_{tki} - \phi_{tki})$ for $k = 1, \dots, K$ to obtain the true action vector $\boldsymbol{z}_t = (z_{t1}, \dots, z_{tK})$;
        if Using the ARS estimator then
            Let $\boldsymbol{j}_t = (j_{t1}, \dots, j_{tK})$, where $j_{tk} \in \{1, \dots, C\}$ is a randomly selected reference category for dimension $k$ at layer $t$.
            for $c = 1, \dots, C$ (in parallel) do
                Let $z_{tk}^{c \leftrightharpoons j_{tk}} := \arg\min_{i \in \{1,\dots,C\}} \pi_{tki}^{c \leftrightharpoons j_{tk}} e^{-\phi_{tki}}$ for $k = 1, \dots, K$;
                Denote $\boldsymbol{z}_t^{c \leftrightharpoons \boldsymbol{j}_t} = (z_{t1}^{c \leftrightharpoons j_{t1}}, \dots, z_{tK}^{c \leftrightharpoons j_{tK}})$ as the $c$th pseudo action vector;
            end
            Let $\bar{f}_t = \frac{1}{C} \sum_{c=1}^{C} f(\boldsymbol{z}_t^{c \leftrightharpoons \boldsymbol{j}_t})$
            Let $g_{\phi_{tkc}} = \big(f(\boldsymbol{z}_t^{c \leftrightharpoons \boldsymbol{j}_t}) - \bar{f}_t\big)(1 - C\pi_{kj_{tk}})$ for all $(k, c) \in \{(k, c)\}_{k=1:K,\ c=1:C}$;
        end
        if Using the ARSM estimator then
            Let $F^{(t)} \in \mathbb{R}^{C \times C}$
            If $t > 1$, sample $\boldsymbol{z}_{1:t-1} \sim q(\boldsymbol{z}_{1:t-1} | \boldsymbol{x})$ ;
            for $(c, j) \in \{(c, j)\}_{c=1:C,\ j \leq c}$ (in parallel) do
                Let $\boldsymbol{j} = j\mathbf{1}_K$, which means $j_k \equiv j$ for all $k \in \{1, \dots, K\}$;
                Let $z_{tk}^{c \leftrightharpoons j} := \arg\min_{i \in \{1,\dots,C\}} \pi_{tki}^{c \leftrightharpoons j} e^{-\phi_{tki}}$ for all $k \in \{1, \dots, K\}$;
                Denote $\boldsymbol{z}_t^{c \leftrightharpoons j} = (z_{t1}^{c \leftrightharpoons j}, \dots, z_{tK}^{c \leftrightharpoons j})$ as the $(c, j)$th pseudo action vector;
                If $t < T$, sample $\boldsymbol{z}_{t+1:T}^{c \leftrightharpoons j} \sim q(\boldsymbol{z}_{t+1:T} | \boldsymbol{z}_t^{c \leftrightharpoons j})$;
                Let $F_{cj}^{(t)} = F_{jc}^{(t)} = f(\boldsymbol{z}_{1:t-1}, \boldsymbol{z}_{t:T}^{c \leftrightharpoons j})$;
                Let $\bar{F}_{\cdot j}^{(t)} = \frac{1}{C} \sum_{c=1}^{C} F_{cj}^{(t)}$ for $j = 1, \dots, C$;
                Let $g_{\phi_{tkc}} = \sum_{j=1}^{C} (F_{cj}^{(t)} - \bar{F}_{\cdot j}^{(t)})(\frac{1}{C} - \pi_{kj})$ for all $(k, c) \in \{(k, c)\}_{k=1:K,\ c=1:C}$;
            end
        end
        $\boldsymbol{\Phi}_t = \boldsymbol{\Phi}_t + \rho_{\boldsymbol{\Phi}_t} \{g_{\phi_{tkc}}\}_{k=1:K,\ c=1:C}$,     with step-size $\rho_{\boldsymbol{\Phi}_t}$;
    end
    $\boldsymbol{\theta} = \boldsymbol{\theta} + \eta_\theta \nabla_{\boldsymbol{\theta}} f(\boldsymbol{z}; \boldsymbol{\theta})$,     with step-size $\eta_\theta$
end
```

Algorithm 3 ARSM policy gradient for reinforcement learning with a discrete-action space of C actions.

```
input  : Maximum number of state-pseudo-action rollouts $S_{\max}$ allowed in a single iteration;
output : Optimized policy parameter $\boldsymbol{\theta}$;
while not converged do
    Given a random state $\boldsymbol{s}_0$ and environment dynamics $\mathcal{P}(\boldsymbol{s}_{t+1} \mid a_t, \boldsymbol{s}_t)$, we run an episode till its termination (or a predefined number of steps) by sampling a true-action trajectory $(a_0, \boldsymbol{s}_1, a_1, \boldsymbol{s}_2, \ldots)$ given policy $\pi_{\boldsymbol{\theta}}(a_t \mid \boldsymbol{s}_t) := \text{Cat}(a_t; \sigma(\boldsymbol{\phi}_t)),\ \ \boldsymbol{\phi}_t := \mathcal{T}_{\boldsymbol{\theta}}(\boldsymbol{s}_t)$, where we sample each $a_t$ by first sampling $(\varpi_{t1}, \ldots, \varpi_{tc}) \sim \text{Dir}(\mathbf{1}_C)$ and then letting $a_t = \arg\min_{i \in \{1,\ldots,C\}} (\ln \varpi_{ti} - \phi_{ti})$;
    Record the termination time step of the episode as $T$, and set the rollout set as $H = []$ and $S_0 = 0$;
    for $t \in$ RandomPermute$(0, \ldots, T)$ do
        Let $A_t = \{(c, j)\}_{c=1:C,\ j<c}$
        Initialize $a_t^{c \rightleftharpoons j} = a_t$ for all $c$ and $j$;
        for $(c, j) \in A_t$ (in parallel) do
            Let $a_t^{c \rightleftharpoons j} = a_t^{j \rightleftharpoons c} = \arg\min_{i \in \{1,\ldots,C\}} (\ln \varpi_{ti}^{c \rightleftharpoons j} - \phi_{ti})$
        end
        Let $S_t = \text{unique}(\{a_t^{c \rightleftharpoons j}\}_{c,j}) \backslash a_t$, which means $S_t$ is the set of all unique values in $\{a_t^{c \rightleftharpoons j}\}_{c,j}$ that are different from the true action $a_t$; Denote the cardinality of $S_t$ as $|S_t|$, where $0 \leq |S_t| \leq C - 1$ ;
        if $S_0 + |S_t| \leq S_{\max}$ then
            $S_0 = S_0 + |S_t|$
            Append $t$ to $H$
        else
            break
        end
    end
    for $t \in H$ (in parallel) do
        Initialize $R_{tmj} = \hat{Q}(\boldsymbol{s}_t, a_t) = \sum_{t'=t}^{T} \gamma^{t'-t} r(\boldsymbol{s}_{t'}, a_{t'})$ for all $m, j \in \{1, \ldots, C\}$   for $k \in \{1, \ldots, |S_t|\}$ (in parallel) do
            Let $\tilde{a}_{tk} = S_t(k)$ be the $k$th unique pseudo action at time $t$;
            Evaluate $\hat{Q}(\boldsymbol{s}_t, \tilde{a}_{tk})$, which in this paper is set as $r(\boldsymbol{s}_t, \tilde{a}_{tk}) + \gamma \sum_{t'=t+1}^{\infty} \gamma^{t'-(t+1)} r(\tilde{\boldsymbol{s}}_{t'}, \tilde{a}_{t'})$, where $(\boldsymbol{s}_t, \tilde{a}_{tk}, \tilde{\boldsymbol{s}}_{t+1}, \tilde{a}_{t+1}, \ldots)$ is a state-pseudo-action rollout generated by taking pseudo action $\tilde{a}_{tk}$ at state $\boldsymbol{s}_t$ and then following the environment dynamics and policy $\pi_{\boldsymbol{\theta}}$;
            Let $R_{tmj} = \hat{Q}(\boldsymbol{s}_t, \tilde{a}_{tk})$ for all $(m, j)$ in $\{(m, j) : a_t^{m \rightleftharpoons j_t} = \tilde{a}_{tk}\}$;
        end
    end
    Esimate the ARSM policy gradient as
        $$\nabla_{\boldsymbol{\theta}} J(\boldsymbol{\theta}) \approx \nabla_{\boldsymbol{\theta}} \left\{ \sum_{t \in H} \sum_{c=1}^{C} \left[ \sum_{j=1}^{C} \left( R_{tcj} - \frac{1}{C} \sum_{m=1}^{C} R_{tmj} \right) \left( \frac{1}{C} - \varpi_{tj} \right) \right] \phi_{tc} \right\},$$
    $\boldsymbol{\theta} = \boldsymbol{\theta} + \eta_\theta J(\boldsymbol{\theta}),$     with step-size $\eta_\theta$;
end
```