

---

# Nonparametric Bayesian Local Partition Model for Multi-task Reinforcement Learning in POMDPs

---

Anonymous Author(s)

Affiliation

Address

email

## 1 Introduction

We consider the problem of multi-task reinforcement learning (MTRL) in partially observable Markov decision processes (POMDPs). This problem has been addressed in [2], based on a Dirichlet process (DP) prior placed on the parameters of regionalized policy representations (RPRs) across different POMDP tasks. While the DP prior favors clustering of (hence information-sharing across) the POMDP tasks, the clustering is based on the global similarity between tasks. Therefore, two tasks that are partly similar to each other may not share information in an effective manner. In this paper, we extend the work in [2] by replacing the DP prior used there with the nonparametric Bayes *local partition process* (LPP) proposed in [1]. A major advantage arising from this replacement is that the LPP allows simultaneous local and global clustering, and therefore it provides an effective vehicle for sharing information between partially similar tasks. We follow [1, 3] to develop a slice sampler for computing the LPP posterior for MTRL. Experimental results on ten grid-world environments demonstrate that, compared to the DP-based MTRL in [2], the LPP-based MTRL yields significantly improved goal rates and the accumulated reward is also improved according to an approximate estimation.

## 2 Regionalized Policy Representation

The regionalized policy representation [2] is a parametric model which conditions the action-selecting probability directly on the history of past actions and observations, without requiring knowing the underlying POMDP.

**Definition 2.1** A regionalized policy representation is a tuple  $(\mathcal{A}, \mathcal{O}, \mathcal{Z}, W, \mu, \pi)$ . The  $\mathcal{A}$  and  $\mathcal{O}$  are respectively a finite set of actions and observations. The  $\mathcal{Z}$  is a finite set of belief regions. The  $W$  is the belief-region transition function with  $W(z, a, o', z')$  denoting the probability of transiting from  $z$  to  $z'$  when taking action  $a$  in belief region  $z$  results in observing  $o'$ . The  $\mu$  is the initial distribution of belief regions with  $\mu(z)$  denoting the probability of initially being in belief region  $z$ . The  $\pi$  are the region-dependent stochastic policies with  $\pi(z, a)$  denoting the probability of taking action  $a$  in belief region  $z$ .

We denote by  $\Theta = \{\pi, \mu, W\}$  the RPR parameters and enumerate the elements of  $\mathcal{A}$  as  $\mathcal{A} = \{1, 2, \dots, |\mathcal{A}|\}$ , where  $|\mathcal{A}|$  is the cardinality of  $\mathcal{A}$ . Similarly,  $\mathcal{O} = \{1, 2, \dots, |\mathcal{O}|\}$  and  $\mathcal{Z} = \{1, 2, \dots, |\mathcal{Z}|\}$ . A sequence of actions  $(a_0, a_1, \dots, a_T)$  is abbreviated as  $a_{0:T}$ , where the subscripts index discrete time steps. Similarly, a sequence of observations  $(o_1, o_2, \dots, o_T)$  is abbreviated as  $o_{1:T}$ , and a sequence of belief regions  $(z_0, z_1, \dots, z_T)$  abbreviated as  $z_{0:T}$ . A history  $h_t$  of actions executed and observation received up to time step  $t$  is denoted by  $h_t = \{a_{0:t-1}, o_{1:t}\}$ .

### 3 Local Partition Process Model for MTRL

For  $M$  POMDP environments, the empirical value function of the  $m$ -th environment is written as:

$$\hat{V}(\mathcal{D}^{(K_m)}; \Theta_m) \stackrel{def.}{=} \frac{1}{K_m} \sum_{k=1}^{K_m} \sum_{t=0}^{T_{m,k}} \tilde{r}_t^{m,k} p(a_{0:t}^{m,k} | o_{1:t}^{m,k}, \Theta_m) \quad (1)$$

where  $\tilde{r}_t^{m,k} = \frac{\gamma^t r_t^{m,k}}{p^\Pi(a_{0:t}^{m,k} | o_{1:t}^{m,k})}$ ,  $\Theta_m = \{\mu_m, \pi_m, W_m\}$ ,  $m = 1, \dots, M$ ,  $\Pi$  is an arbitrary soft policy. Assuming the agent-environment interactions are episodic, it is proven in [2] that  $\lim_{K_m \rightarrow \infty} \hat{V}(\mathcal{D}^{(K_m)}; \Theta_m)$  is the expected sum of discounted rewards by following the RPR policy parameterized by  $\Theta_m$  for an infinite number of steps.

We rewrite  $\Theta_m = \{\Theta_{mj}, j = 1, \dots, p\}$  with  $p = (|\mathcal{A}||\mathcal{O}| + 1)|\mathcal{Z}| + 1$ , where  $\Theta_{m1} = \mu_m$ ,  $\Theta_{m2} = \pi_m(z = 1, \cdot)$ ,  $\dots$ ,  $\Theta_{m,|\mathcal{Z}|+1} = \pi_m(z = |\mathcal{Z}|, \cdot)$ ,  $\Theta_{m,|\mathcal{Z}|+2} = W_m(z = 1, a = 1, o = 1, \cdot)$ ,  $\dots$ ,  $\Theta_{m, (|\mathcal{A}||\mathcal{O}|+1)|\mathcal{Z}|+1} = W_m(z = |\mathcal{Z}|, a = |\mathcal{A}|, o = |\mathcal{O}|, \cdot)$ .

Let  $G_0$  be the base probability measure. The LPP prior is specified as follows.

$$\Theta_{mj} = \theta_{\gamma_{mj}, j}, \quad m = 1, \dots, M, j = 1, \dots, p, \quad (2)$$

$$\theta_\gamma = \{\theta_{\gamma, j}\}_{j=1}^p \sim G_0, \quad \gamma \in \{0, 1\} \times \{1, 2, \dots, \infty\} \quad (3)$$

$$\gamma_{mj} \sim s_{mj} \delta_{(0, \phi_{m0})} + (1 - s_{mj}) \delta_{(1, \phi_{mj})}, \quad j = 1, \dots, p, \quad (4)$$

$$s_{mj} \sim \eta_j \delta_1 + (1 - \eta_j) \delta_0, \quad \eta_j \sim \text{Beta}(1, \beta), \quad j = 1, \dots, p, \quad (5)$$

$$\phi_{mj} \sim \sum_{h=1}^{\infty} \nu_h \delta_h, \quad \nu_h = \nu_h^* \prod_{l < h} (1 - \nu_l^*), \quad \nu_h^* \sim \text{Beta}(1, \alpha), \quad j = 0, 1, \dots, p, \quad (6)$$

where  $\{\gamma = (0, h) : h = 1, \dots, \infty\}$  indicate global clusters and  $\{\gamma = (1, h) : h = 1, \dots, \infty\}$  indicate local clusters,  $\gamma_{mj} = (l, h)$  indicates that  $\Theta_{mj}$  is allocated to cluster  $(l, h)$ ,  $\phi_{m0}$  is the global cluster indicator and  $\phi_{mj}$ ,  $j > 0$ , is the local cluster indicator, for  $\Theta_{mj}$ . Clearly, as  $\beta \rightarrow 0$ ,  $\eta_j \equiv 1$  and  $s_{mj} \equiv 1$ , which means global sharing is always chosen; in the case, the LPP prior reduces to a Dirichlet process prior  $DP(\alpha G_0)$ .

### 4 Slice Sampling

We develop a slice sampler to compute the posterior. We introduce latent variables  $u_m = \{u_{mj}\}_{j=0}^p$  for  $m = 1, \dots, M$ . Let the complete data joint likelihood be denoted by  $p(\mathcal{D}^{(K_m)}, u, s | \theta_{\gamma_m}, \alpha, \beta)$ , which is proportional to

$$\prod_{m=1}^M \{\hat{V}(\mathcal{D}^{(K_m)}; \theta_{\gamma_m}) \mathbb{1}(u_{m0} < \nu_{\phi_{m0}}) \prod_{j=1}^p \mathbb{1}(u_{mj} < \nu_{\phi_{mj}}) \eta_j^{s_{mj}} (1 - \eta_j)^{(1-s_{mj})}\} \quad (7)$$

where the  $u_{mj} \in (0, 1)$ . The sampling procedure is an iteration of the following steps.

*Step 1.* Draw latent variable  $u_{mj} \sim (0, \nu_{\phi_{mj}})$ , for  $j = 0, 1, \dots, p$ .

*Step 2.* Draw latent variable  $s_{mj} \sim \text{Bern}(p_{mj})$ , with

$$p_{mj} = \frac{\eta_j \hat{V}(\mathcal{D}^{(K_m)}; \theta_{\gamma_m(s_{mj}=1)})}{\eta_j \hat{V}(\mathcal{D}^{(K_m)}; \theta_{\gamma_m(s_{mj}=1)}) + (1 - \eta_j) \hat{V}(\mathcal{D}^{(K_m)}; \theta_{\gamma_m(s_{mj}=0)})} \quad (8)$$

where  $\gamma_m(s_{mj} = l)$  denotes the current value of  $\gamma_m$ , with  $\gamma_{mj}$  replaced with  $(0, \phi_{m0})$  for  $l = 1$  and  $(1, \phi_{mj})$  for  $l = 0$ .

*Step 3.* Draw the stick-breaking variable  $\nu_h^*$  from the conditional density that is proportional to

$$(1 - \nu_h^*)^{(\alpha-1)} \prod_{m=1}^n \prod_{j=0}^p \mathbb{1}(\nu_{\phi_{mj}}^* \prod_{l < \phi_{mj}} (1 - \nu_l^*) > u_{mj}) \quad (9)$$

To implement Step 3, we let  $\phi^* = \max\{\phi_{mj}, m = 1, \dots, M, j = 0, 1, \dots, p\}$ , and then the conditional distribution of  $\nu_h^*$  is  $\text{Beta}(1, \alpha)$  for  $h > \phi^*$ , and it is  $\text{Beta}(1, \alpha)$  that is truncated to have a support  $(a_h, b_h)$  for  $h \leq \phi^*$ , where

$$a_h = \begin{cases} 0, & \text{if } \sum_{m=1}^M \mathbb{1}(\phi_{mj} = h) = 0 \\ \max\left\{\frac{u_{mj}}{\prod_{l < h} (1 - \nu_l^*)} : \phi_{mj} = h, m = 1 \dots M, j = 0 \dots p\right\}, & \text{else} \end{cases} \quad (10)$$

$$b_h = \begin{cases} 1, & \text{if } h = \phi^* \\ 1 - \max\left\{\frac{u_{mj}}{\nu_{\phi_{mj}}^* \prod_{l < \phi_{mj}, l \neq h} (1 - \nu_l^*)} : \phi_{mj} > h, m = 1 \dots M, j = 0 \dots p\right\}, & \text{else} \end{cases} \quad (11)$$

*Step 4.* Draw  $\phi_{mj}$  from the conditional probability that is proportional to  $1(h \in A_{mj})\hat{V}(\mathcal{D}^{(K_m)}; \theta_{\gamma_{m(\phi_{mj}=h)}})$ , where  $\theta_{\gamma_{m(\phi_{mj}=h)}}$  denotes  $\theta_{\gamma_m}$  with the cluster-indicator of its  $j$ -th component equal to  $h$  (see (3) for details),  $A_{mj} = \{h : \nu_h > u_{mj}\}$  is a finite subset of  $\{1, 2, \dots, \infty\}$  obtained by first sampling  $\nu_h^*$  in Step 3, for  $h = 1, \dots, \tilde{\phi}$ , with  $\tilde{\phi}$  the smallest value satisfying  $\sum_{h=1}^{\tilde{\phi}} \nu_h^* \prod_{l < h} (1 - \nu_l^*) > 1 - u^*$ , where  $u^* = \min\{u_{mj}, m = 1, \dots, M, j = 0, 1, \dots, p\}$ .

*Step 5.* Draw  $\theta_{(l,h)}$  from the conditional distribution that is proportional to

$$G_0(\theta_{(l,h)}) \prod_{m=1}^M \hat{V}(\mathcal{D}^{(K_m)}; \theta_{\gamma_m}) \quad (12)$$

*Step 6.* Draw  $\eta_j$  from  $\text{Beta}(1 + \sum_m s_{mj}, \beta + \sum_m (1 - s_{mj}))$ .

*Step 7.* Draw the hyperparameter  $\beta$  from  $Ga(a_\beta + p, b_\beta - \sum_{j=1}^p \log(1 - \eta_j))$ .

*Step 8.* Draw  $\alpha$  from  $Ga(a_\alpha + \tilde{\phi}, b_\alpha - \sum_{h=1}^{\tilde{\phi}} \log(1 - \nu_h^*))$ .

## 5 Experimental Results

We consider the ten maze navigation tasks in [2] and follow the experimental setup used there to replicate the results for comparison. At the beginning of slice sampling, the hyper-parameters of  $a_\alpha, b_\alpha, a_\beta, b_\beta$  in Section 4 are all set to be one. The results are reported in Figure 1.

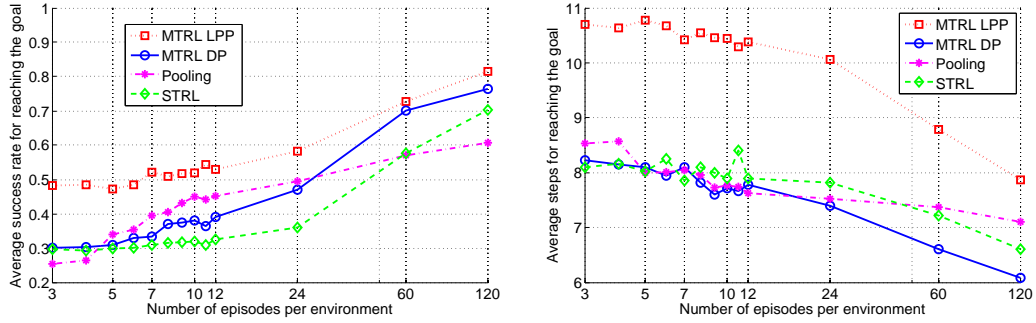


Figure 1: Comparison of the LPP-based MTRL, the DP-based MTRL, single task reinforcement learning (STRL), and the pooling, on the ten maze navigation tasks, with the results of the latter three methods cited from Figure 3 in [2]. The performance of the policy obtained by each policy is evaluated by: (left) the average success rate at which the agent reaches the goal within 15 steps, and (right) the average number of steps that the agent takes to reach the goal. When the agent does not reach the goal within 15 steps, the number of steps is 15. Each performance measure is computed from 1000 instances of policy execution, and is averaged over 20 independent trials.

It is seen that the LPP-based MTRL yields significantly improved goal rates, although the number of steps taken for reaching the goal is increased, compared to the DP-based MTRL in [2]. A more accurate comparison can be made using the accumulated reward. A rough estimate of the average accumulated reward can be obtained by the following formula:  $\text{reward} = \text{discount}^{\text{steps}} \times \text{goal rate}$ . For the present problem, discount is 0.9. Therefore, the average accumulated reward is approximated as  $0.9^{8.2} \times 0.3 = 0.12$  and  $0.9^{11.7} \times 0.5 = 0.14$  for the DP-MTRL and LPP-MTRL, respectively, when the number of episodes is 3. These estimates show that the LPP-MTRL indeed earns better rewards than the DP-MTRL. We leave the exact reward comparison to future work.

## References

- [1] D. B. Dunson. Nonparametric bayes local partition models for random effects. *Biometrika*, 96(2):249–262, 2009.
- [2] H. Li, X. Liao, and L. Carin. Multi-task reinforcement learning in partially observable stochastic environments. *Journal of Machine Learning Research*, 10:1131–1186, 2009.
- [3] S. G. Walker. Sampling the dirichlet mixture model with slices. *Communications in Statistics - Simulation and Computation*, 36(1):45–54, 2007.