LEARNING TRANSITIONAL SKILLS WITH INTRINSIC MOTIVATION

Anonymous authors
Paper under double-blind review

ABSTRACT

By maximizing an information theoretic objective, a few recent methods empower the agent to explore the environment and learn useful skills without supervision. However, when considering to use multiple consecutive skills to complete a specific task, the transition from one to another cannot guarantee the success of the process due to the evident gap between skills. In this paper, we propose to learn transitional skills (LTS) in addition to creating diverse primitive skills without a reward function. By introducing an extra latent variable for transitional skills, our LTS method discovers both primitive and transitional skills by minimizing the difference of mutual information and the similarity of skills. By considering various simulated robotic tasks, our results demonstrate the effectiveness of LTS on learning both diverse primitive skills and transitional skills, and show its superiority in smooth transition of skills over the state-of-the-art baseline DIAYN.

1 INTRODUCTION

Deep reinforcement learning (DRL) has shown its great effectiveness in learning various reward-driven skills in wide domains, such as performing robotic manipulation tasks (Levine et al. (2016)), playing video games (Mnih et al. (2015)), playing adversarial board games (Silver et al. (2016)) and implementing robot navigation in complex environments (Wang et al. (2018)). Nevertheless, for the majority of real applications, there is no reward in a long term until the agent reaches a goal state (Wu & Chen (2007)), especially in unseen environments. In such cases, DRL has difficulty in carrying out the tasks.

By observing the human intelligence that can explore their surroundings and learn valuable skills without reward, a couple of prior works have been recently proposed to generate useful skills without supervision by embedding the intrinsic motivation into DRL methods (Barto (2013),Ryan & Deci (2000)). Diverse skills can be autonomously acquired without reward by maximizing an information theoretic objective using a maximum entropy policy (DIAYN (Eysenbach et al. (2018)); VIC (Gregor et al. (2016)); DAS (Sharma et al. (2019))). Discovered useful skills may help the exploration in complex environments, and can also serve as primitive skills for hierarchical DRL.

Although discovered useful skills are both distinguishable and diverse, it is still an exceedingly challenge to integrate such skills for a complex task that requires smooth transitions between skills (Lee et al. (2018)). Take the basketball as an example: learning the passing, catching and shooting skills in an isolated way cannot guarantee a score due to the possible failure in the process of transitions between skills. To address this problem, we propose to further learn transitional skills (LTS) without a reward function, where discovered primitive skills that are the same as prior works are also distinguishable and as diverse as possible (Eysenbach et al. (2018)).

More concretely, our LTS method learns both primitive and transitional skills by optimizing an information theoretic objective, where an extra controller of transitional skills is defined except the primitive skill's controller. In such case, the information theoretic objective is the difference of mutual information and the similarity of skills. On four simulated robotic tasks, experimental results show that our LTS can discover both primitive skills and transitional skills, successfully perform the transition between primitive skills that are distinguishable, and achieve a better peformance in comparison to the state-of-the-art baseline DIAYN.

The main contributions of our work are as follows: (1) our proposed LTS can learn both primitive and transitional skills without extrinsic reward, where the primitive skills are distinguishable and diverse, and the transitional skills can accomplish smooth transitions between primitive skills; (2) The discovered skills are in a continuous space rather than a discrete space, which indicates that arbitrary useful skills might be acquired for specific requirements. (3) Extensive experiments are conducted, which demonstrates the effectiveness of our LTS method in learning two categories of skills, performing the transition between primitive skills.

2 PRELIMINARIES

RL: In the standard RL setup, an agent interacts with an environment over discrete time. At time step t, the agent observes the current state s_t and selects an action a_t according to a policy $\pi(a_t|s_t)$. Then, the agent receives a reward r_t and comes to the next state s_{t+1}. The objective of learning is to maximize the discounted return $R = \sum_{t=0}^{\infty} \gamma^t r_t$ of the policy π, where $\gamma \in [0, 1]$ is a discount factor.

Learn Skills with RL: Using the notation from information theory: we introduce two random variables S and A for states and actions, respectively. To discover diverse skills, a latent variable $\Omega \in p(\omega)$ is introduced such that the policy is denoted by $\pi(a_t|s_t, \omega_i)$, where a_t and s_t denote the action and observation state at time stamp t respectively, and ω_i is a sample form distribution $p(\omega)$. We denote that the policy conditioned on a fixed Ω, ω_i, as a "skill". A different ω is input in the policy π to allow the agent to follow different behavioral strategy. Prior works have shown that maximizing the mutual information between the states S and the skills Ω results in distinguishable and diverse skills.

By primarily maximizing the mutual information between the final states S_f and the skills Ω given the initial states S_0,

$$\mathrm{I}(S_f; \Omega|S_0)^1, \tag{1}$$

the variational intrinsic control (VIC) (Gregor et al. (2016)) shows the success of acquiring distinguishable skills from the final states.

Furthermore, in order to enhance the diversity of skills as much as possible, DIAYN primarily maximizes the mutual information between the states S at all the time stamps and the skills Ω (Eysenbach et al. (2018)),

$$\mathrm{I}(S_t; \Omega) + \mathrm{H}[A|S, \Omega]^2, \tag{2}$$

which indicates that different skills visit different states and such diverse skills can be identified distinguishably.

Both VIC and DIAYN successfully discover primitive skills by maximizing the mutual information between the states and the skills. To carry out a complex task that requires a smooth transition between skills, we propose the LTS scheme in this paper to learn both primitive and transitional skills by using an information theoretic objective.

3 METHODOLOGY

In this section, we elaborate our proposed LTS method to discover both primitive and transitional skills without extrinsic reward. We use the same notation as mentioned above: S, A and Ω are random variables for states, actions and primitive skills, respectively. And we define N as the number of primitive skills in this paper, i.e. ω_0, ω_2, ..., ω_{N-1}. Besides, we introduce an extra latent variable $Z_{i,j}$ following the distribution $p(z_{i,j}|\omega_i, \omega_j)$, on which we condition our transitional policy and we refer to a the policy conditioned on a fixed $Z_{i,j}$ as a "transitional skill" from the primitive skill ω_i to ω_j.

[1]The mutual information is denoted as the formation of conditional probability and contains the initial observation s_0: $\mathrm{I}(S_f; \Omega|S_0) = -\sum_{s_f} p(s_f|s_0) + \sum_{\omega, s_f} p^J(s_f|s_0, \omega) \log p^J(s_f|s_0, \omega)$. The controllability distribution $p^C(\omega|s_0)$ maximizes the behavior diversity.

[2]The second term suggests that each skill should act as randomly as possible, aiming improving the exploration.

As in Figure 1, denote by S_i^p and S_j^p the states of the primitive skills ω_i and ω_j. The whole transitional process from one primitive skill ω_i to ω_j is divided into $K-1$ transitions so that the sets of transition states are obtained as $S_{i,j,1}^t, S_{i,j,2}^t, ..., S_{i,j,K-1}^t$, where each set of transition states $S_{i,j,k}^t$, $1 \leq k \leq K-1$, corresponds to a transitional skill $z_{i,j,k}$, as shown in Figure 1. By controlling $z_{i,j,k}$ between ω_i and ω_j, we expect to accomplish the smooth state transition. $S_{i,j,0}^t$ (or S_i^p) corresponds to the starting primitive skill w_i, and $S_{i,j,K}^t$ (or S_j^p) corresponds to the ending primitive skill w_j. For convinence, define by $S^P = \{S | S = S_i^p \cup S_{i,j,k}, \text{ for all i,j,k}\}$ all states corresponding to primitive and transitional skills.

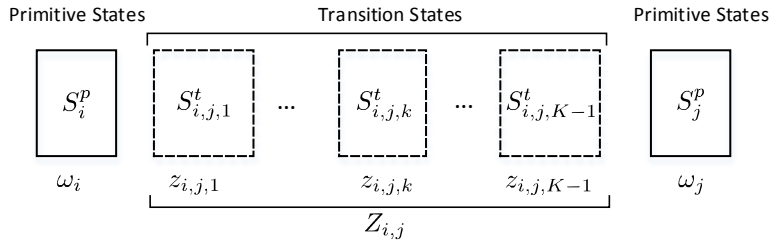


Figure 1: Block diagram of primitive and transition states, and their corresponding skills.

In order to learn both diverse primitive skills and transitional skills, the mutual information between the states $S_{i,j,k} \in S^P$ from $z_{i,j,k}$ and the states S_i^p from the primitive skill Ω_i, denoted by $\mathrm{I}(S_{i,j,k}; S_i^p | z_{i,j,k}, \omega_i, \omega_j)$, will be large at the begining of transition with a small k (e.g. close to 0); conversely, this mutual information will be small at the end of transition with a large k (e.g. close to K).

By defining the similarity $f_{i,j,k}^s$ between ω_i and $z_{i,j,k}$, which is large with a small k but small with a large k, i.e. the same tendency as the muture information $\mathrm{I}(S_{i,j,k}; S_i^p | z_{i,j,k}, \omega_i, \omega_j)$, the objective of our learning problem is to minimize the difference between the mutual information and the similarity

$$\begin{aligned} \mathrm{L}(\theta) &\triangleq \mathbb{E}\left[\left| \mathrm{I}(S_{i,j,k}; S_i^p | z_{i,j,k}, \omega_i, \omega_j) - f_{i,j,k}^s \right| \right] \\ &\approx \mathbb{E}_{\omega_i \sim p(\omega), z_{i,j,k} \sim p(z_{i,j} | \omega_i, \omega_j), s_t \sim \pi(z_{i,j,k})} \left[\left| \log p(\omega_i | s_t) - \log p(s_i^p) - f_{i,j,k}^s \right| \right], \end{aligned} \tag{3}$$

where conditional probability $p(s_i^p | s_{i,j,k}^t)$ is converted to $p(\omega_i | s_t)$ for the reason that the approach could learning diverse primitive skills which means different primitive skills indicate different primitive states. This objective function enables the log-conditional probability $\log p(\omega_i | s_t)$ and the preprocessed similarity $f_{i,j,k}^s$ are of proportional relation. When $z_{i,j,k} = \omega_i$ or $z_{i,j,k} = \omega_j$, diverse primitive skills will be discovered. Otherwise, the transitional skills will be learned for smooth transition from ω_i to ω_j. By defining the degree of divergence $f_{i,j,k}^d = \frac{1 - \alpha \cdot f_{i,j,k}^s}{\alpha}$ (α is a scaling factor) between ω_i and $z_{i,j,k}$, we could transfer minimizing $\mathrm{L}(\theta)$ into maximizing $\mathrm{F}(\theta)$[3]:

$$\mathrm{F}(\theta) \triangleq \mathbb{E}_{\Omega \sim p(\omega), z_{i,j,k} \sim p(z_{i,j} | \omega_i, \omega_j), s_t \sim \pi(z_{i,j,k})} \left[\log p(\omega_i | s_t) + f_{i,j,k}^d \right]. \tag{4}$$

Furthermore, Jensen's Inequality tells us that replacing $p(\omega_i | s_t)$ with $q_\phi(\omega_i | s_t)$ gives us a variational lower bound $\mathrm{G}(\theta, \phi)$ on our objective $\mathrm{L}(\theta)$ (seed Appendix A.3 for derivation):

$$\begin{aligned} \mathrm{F}(\theta) &\geq \mathbb{E}_{\Omega \sim p(\omega), z_{i,j,k} \sim p(z_{i,j} | \omega_i, \omega_j), s_t \sim \pi(z_{i,j,k})} \left[\log q_\phi(\omega_i | s_t) + f_{i,j,k}^s \right] \\ &\triangleq \mathrm{G}(\theta, \phi), \end{aligned} \tag{5}$$

The second term brings the probability of inferring current ω_i into correspondence with the divergence between ω_i and $z_{i,j,k}$. When $z_{i,j,k}$ is close to primitive skills ω_i, the discriminator q_ϕ has a large probability to correctly infer ω_i given s_t with related to $z_{i,j,k}$, and vice versa.

[3]See Appendix A.1 for further analysis.

4 IMPLEMENTATION

4.1 ONE-HOT ENCODING AND HINDSIGHT

In the previous section, we derived a theoretical algorithm for learning transitional skills. In implementation, there are several problems with the use of transition skills $Z_{i,j}$.

One of these is that when $\omega_i \neq \omega_j$ $(i \neq j)$, the discriminator will face with a dilemma: the intersection of transition states and primitives states are not empty, i.e. $S^T_{i,j} \cap S^P \neq \oslash$, leading to a conflict between diversity and transition. An alternative is to use one-hot encoding for primitive skills. In such case, the transitional skill $\hat{z}_{i,j,k}$ also has a different expression. Further analysis could be found in Appendix B.

On the other hand, along with the growth of the number of primitive skills N and transitional skills $K-1$, we have a high training complexity. For improving the efficiency, we utilize the hindsight experience reply mechanism to allow sample-based learning from the sparse reward. In our approach, we calculate the conditional probability given by the discriminator instead of using a single value because $q_\phi(\omega_i|s_t)$ can just guarantee the consistency of $z_{i,j,k}$ with ω_i but ignore the relation with other primitive skills. So we change $q_\phi(\omega_i|s_t)$ to the conditional probability:

$$q_\phi(\Omega, s_t) = [q_\phi(\omega_0|s_t), q_\phi(\omega_1|s_t), ..., q_\phi(\omega_{N-1}|s_t)]^{\mathrm{T}}. \tag{6}$$

By doing so, we consider all primitive skills. Correspondingly, we change the criterion of $f^d_{i,j,k}$ into $\mathbf{f}^d_{i,k}$:

$$\mathbf{f}^d_{i,k} = [f^d_{i,1,k}, f^d_{i,2,k}, ..., f^d_{i,N-1,k}]^{\mathrm{T}}. \tag{7}$$

Consequently, the overall objective takes the form:

$$\mathrm{G}(\theta, \phi) = \frac{1}{N} \cdot \mathrm{E}_{\omega_i \sim p(\omega), s_t \sim \pi(z_{i,k})}(\|q_\phi(\Omega|s_t) + \mathbf{f}^d_{i,k}\|_2), \tag{8}$$

where $z_{i,k} \sim p_z(z|\omega_i) = \{z_{i,k}|z_{i,k} = z_{i,j,k}, 0 \leq j \leq N-1, z_{i,j,k} \in p(z_{i,j}|\omega_i, \omega_j)\}$. Basically, we convert the classification problem (i.e. learning discrete primitive skills) into a regress problem (i.e. learning continuous primitive and transitional skills). When $z_{i_1,k} = \omega_{i_2}$, the regression problem is simplified as a classification problem to learn primitive skills.

Algorithm 1 Learning Transitional Skills (LTS)

while NOT converged **do**
  Sample $\omega_i \sim p(\omega)$
  Sample a skill $z \sim p_z(z|\omega_i)$ and an initial state $s_0 \sim p_0(s)$
  for $t \leftarrow 1$ **to** steps per episode **do**
    Sample an action $a_t \sim \pi_\theta(a_t|s_t, z)$;
    Interact with the environment: $s_{t+1} \sim p(s_{t+1}|s_t, a_t)$;
    Compute $D_t = \|q_\phi(\Omega|s_{t+1}) + \mathbf{f}^d_{i,k}\|$ with the discriminator (ϕ);
    Set the reward for current skill: $r_t = D_t$.
    By using SAC, update the policy (θ) to maximize the discounted return $R = \sum_{t=0}^{\infty} \gamma^t r_t$;
    Update the discriminator (ϕ) to maximize D_t with SGD.
  end for
end while

4.2 ALGORITHM

Using the discriminator for distinguishing primitive and transitional states, we summarize our LTS method in Algorithm 1. At each roll-out, we sample a skill z from a fixed skill distribution $p_z(z|\omega_i)$ given ω_i. After the agent interacts with the environment at time step t, the discriminator calculates the discriminability as

$$D_t = \|q_\phi(\Omega|s_{t+1}) + \mathbf{f}^d_{i,k}\|. \tag{9}$$

As mentioned above, we encode primitive skills using one-hot encoding. And we also constrain $\sum z_{i,j,k} = 1$ so that the similarity $\mathbf{f}^s_{i,k} = z_{i,j,k}$. To improve the exploration, we adopt soft actor critic (SAC) algorithm to train our policy, adding the regularization $\mathbb{E}_{i,j}[\,\mathrm{H}[A|S, Z_{i,j}]\,|]$ to maximize the policy's entropy over actions given states and skills. (see Appendix D for the hyperparameters.)

5 RELATED WORK

Real-world tasks often require diverse behaviors. Wang et al. (2017) notes that building versatile embodied agents capable of performing a wide and diverse set of behaviors is one of the long-standing challenges of AI. And learning continuous control of diverse behaviors in locomotion (Merel et al. (2017); Heess et al. (2017); Peng et al. (2017)) and robotic manipulation (Ghosh et al. (2018); Gu et al. (2017)) is an active research area. In this scenario, although some complex tasks can be solved through extensive reward engineering, undesired behaviors often emerge because of the sparse nature of reward (Riedmiller et al. (2018)). Moreover, training complex skills from scratch is not computationally practical. These issues can be addressed by use of intrinsic motivation (Barto (2013); Chentanez et al. (2005); Singh et al. (2010)), which is a reward-free learning method. Historically, the intrinsic motivation comes from the tendency of organisms to play and explore their environment without any reward (Ryan & Deci (2000), White (1959)).

Another line of work that is conceptually close to our method copes with information theories that are used to drive the agent's exploration. The information gain is a reward based on the reduction of uncertainty on environment's dynamics (Little & Sommer (2013); Oudeyer & Kaplan (2007)), which can also be assimilated to learning progress (Frank et al. (2013); Oudeyer & Kaplan (2007)). This can push agents into unknown areas on the one hand, and prevent them from being attracted to random areas on the other.

Recent work has also applied information theory for skill discovery. VIC (Gregor et al. (2016)) is an optional discovery technique by optimizing a variational lower bound on the mutual information between the context and the final state in a trajectory, conditioned on the initial state. Furthermore, DIAYN (Eysenbach et al. (2018)) maximizes the mutual information between states and skill to achieve diversity and shows the interest as a pre-training for hierarchical reinforcement learning or as an initialization for learning a task. While discriminative embedding reward networks (DISCERN) (Warde-Farley et al. (2018) aim to simultaneously learn a goal-conditioned policy and a goal achievement reward function by maximizing the mutual information between the goal state and the achieved state. Let us notice that the skill space here is discrete, with just one or multiple policies. However, we considered the relationship between different skills during the training process and finally formed a continuous skill space, likely because of inducing a novel latent variable for transitional skills.

In addition, it is important to point out that our skills are transitional with an intrinsically driven approach, which is very different from numerous previous works. While Sharma et al. (2019) discovers predictable behaviors to let the single skill more predictable, it need an external model-predictive-control (MPC) paradigm (Garcia et al. (1989)) to connect skills. Peng et al. (2019) learns reusable motor primitives that can be composed to produce a continuous spectrum of skills. To bridge the gap between skills, Lee et al. (2018) propose a transition policy to get a new smooth skill. In comparison, our method captures intrinsic transition, which is independent from external tasks, and could eliminate the extra fine-tuning process.

6 EXPERIMENTS

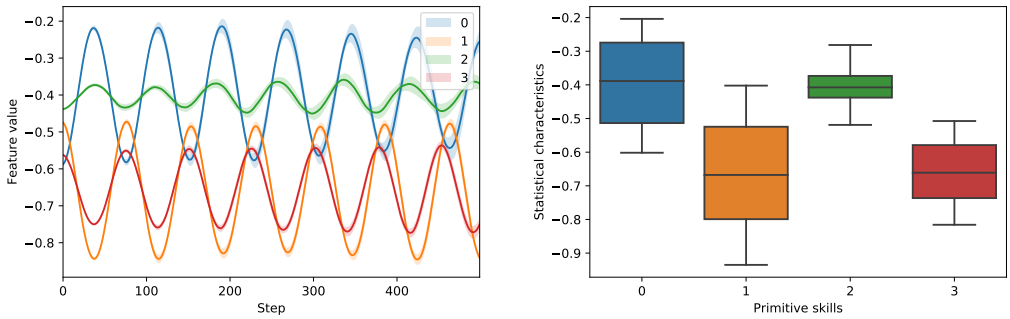
In our experiments, we aim to demonstrate the effectiveness of our approach for learning primitive and transitional skills. We evaluate LTS and compare it to prior works.

6.1 DIVERSITY AND TRANSITIONAL SKILLS

In this section, we provide visualizations and quantitative analysis for our LTS method. Using the MuJoCo (Todorov et al. (2012)) environments from the OpenAI gym [4] as our test bed, we use LTS to address multiple control issues, such as CartPole, MountainCar and Pendulum.

To evaluate different skills in different environment, we extract features from the observation of the agent, e.g. in the MountainCar environment, where we use the variance of the car's altitude as the feature. For more details on the features in other environments, please refer to Appendix C.

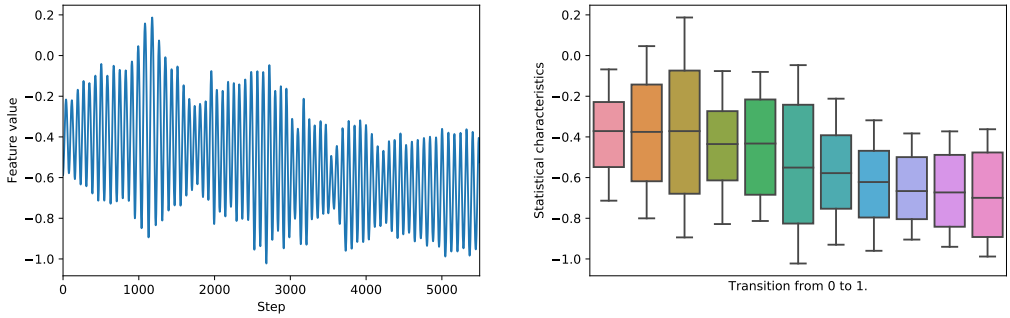
[4] http://gym.openai.com/



(a) Trajectories of 4 skills. (b) Box-plots of 4 skills.

Figure 2: Primitive skills (ω_0, ω_1, ω_2, ω_3) in MountainCar environment.

Figure 2 illustrates the discovered primitive skills in the MountainCar environment. As shown in Figure 2(a), all four skills moves in an periodic manner. Corresponding to all 4 skills in Figure 2(a), the statistical values of features are shown in Figure 2(b) using Box-plot. It is observed that, these four skills have different movement patterns so that these skills are easy to be distinguished. Moreover, we consider a different number of primitive skills and different environments (CarPole, MountainCar and Pendulum) as in Appendix E, from which it is observed that all primitive skills have an evident difference in feature statistics and are easy to be distinguished.



(a) Trajectory of transition. (b) Box-plots of transition.

Figure 3: Transition from skill ω_0 to ω_1 in Mountain-Car.

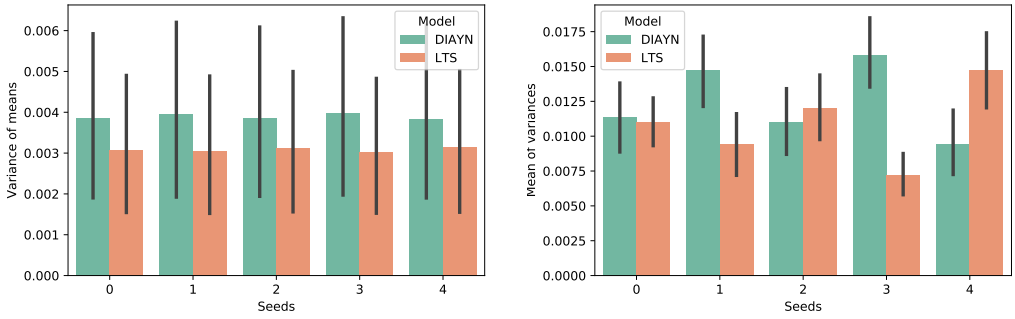
Transition. Furthermore, we use the transitional skill $z_{0,1,k}$ to show the performance of transition from one primitive skill $\omega_0 = [1, 0, 0, 0]$) to another $\omega_1 = [0, 1, 0, 0]$), where the number of transition skills is set as 9.

Figure 3(a) shows the transition of the trajectory, where 9 transition skills are uniformly distributed along the horizontal axis from Step 0 to 5500. Figure 3(b) shows the transition of feature statistics, where the features of two primitive skills and 9 transitional skills are included.

It is observed that the primitive skill ω_0 smoothly changes to ω_1 via 9 transitional skills. More specifically, there exists a slight increment on the amplitude of features in the first three skills, which is followed by consecutive declines until the ending primitive skill ω_1 is discovered. More experiments on skill transition are given in Appendix F. This demonstrates the effectiveness of our LTS method on discovering transitional skills and accomplishing the successful transition between two primitive skills.

6.2 COMPARISON WITH DIAYN

In DIAYN (Eysenbach et al. (2018)), the information regularization implies that learned useful skills could dictate the states that agent visits, by maximizing $\mathrm{I}(S; Z)$. In this subsection, we compare our LTS method with the state-of-the-art DIAYN in terms of learning diverse primitive skills and transitional skills. Experimental results show that LTS achieves an approximate performance on diversity of primitive skills, and a much better performance on skill transition.



(a) Variance of means. (b) Mean of Variances.

Figure 4: Comparison of diversity of primitive skills between LTS and DIAYN.

Diversity: Different skills have different means or variances from their own trajectories. In this subsection, we use (1) *variance of means* and (2) *mean of variances* of the features as metrics to evaluate the diversity of learned primitive skills using LTS and DIAYN.

Figure 4(a) shows the variance of means while Figure 4(b) shows the mean of variances, where the black perpendicular line represents the range of values from a single experiment and we collect all trajectories of various primitive skills. In Figure 4(a), the height of this line depicts the range of variance of means by considering all trajectories of primitive skills. As shown, LTS obtains a lower variance of means in compare to DIAYN because the learned transitional skills from LTS affect the variance of learned primitive skills.

The lower variance of means does not degrade the diversity of learned primitive skills, which can be observed from Figure 4(b). Although there is a large difference between experiments with random seeds, LTS has generally a similar mean of variance as DIAYN, indicating that our method performs similar with the baseline in terms of learning diverse primitive skills.

Transition: We also conduct experiments to evaluate the skill transition of LTS and DIAYN, where we use identical encoding scheme for both and the number of transitional skills is set as 8.

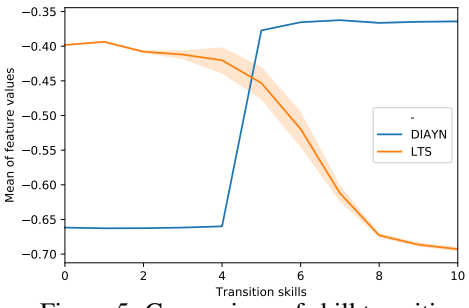


Figure 5: Comparison of skill transition.

Figure 5 shows the complete trajectory of skill transition, where the value denotes the mean of features like the middle bar in Figure 3(b). It is observed that LTS achieves a smooth transition from one primitive skill to another while the transition in DIAYN suffers from one sharp-rising phase and two steady phases indicating a rigid transition process. We control the transition in LTS from ω_0 to ω_1 as in Figure 2(b). However, DIAYN learned different primitive skills so that we cannot select the same starting and ending primitive skills as LTS.

A more comprehensive study including the real path of movement and statistical characteristics is conducted and the results are reported in Appendix G.

6.3 GENERALIZATION

Our LTS method suffers from the high training complexity to learn transitional skills. A reasonable approach to tackle this problem is to use a fixed number of transitional skills. In this experiment, we set this number as 3, indicating that the nonzero element of $z_{i,j,k}$ is from the set $\{0.25, 0.50, 0.75\}$ in the training phase.

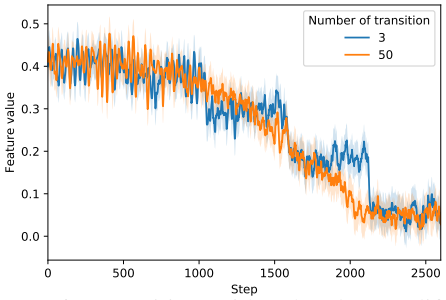


Figure 6: Transition with related to 2 different number transition skills.

Figure 6 shows the skill transition, where the test phase considers 3 and 50 transitional skills and 50 transitional skills are used for evaluating the generalization of our LTS method. It is observed that the transition in blue suffers from severe declines, possibly leading to the failure of the process in practice; fortunately, the transition in red goes through a steady and smooth process from one primitive skill to another. This experiment demonstrate the great capability of LTS in generalization that guarantees the sucess of the skill transition.

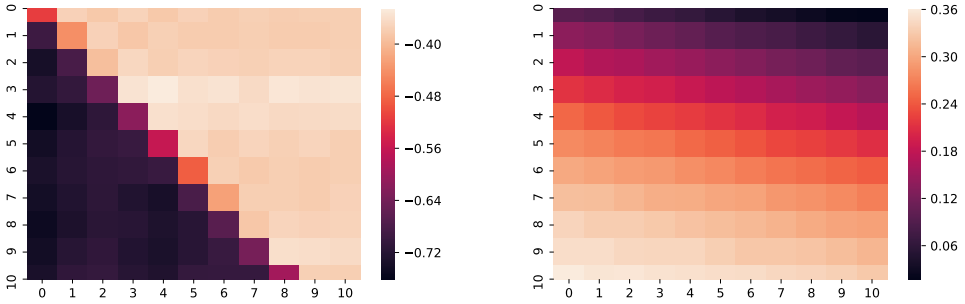


Figure 7: Skill space of DIAYN and LTS.

6.4 EQUIP THE AGENT WITH MORE TRANSITIONAL SKILLS

Figure 7 shows much more skills and corresponding transitions between them, where each small square denotes a different skill, the number of primitive skills is 2 and the number of transitional skills is 98. Two primitive skills $[1, 0, 0, 0]$ and $[0, 1, 0, 0]$ locate at the lower left and upper right squares.

It is observed that our LTS method is able to accomplish a smooth transition between two arbitrary skills, whatever primitive skills or transitional skills defined in this paper; however, DIAYN suffers from a rigid transition in most cases. Furthermore, these results provide us a deep insight that LTS has the ability to learn a larger *continuous* skill space. The agent equipped with such numerous skills is expected to become much more powerful.

7 CONCLUSION

In this paper, we introduce a novel LTS method to learn transitional skills without extrinsic reward by using an extra latent varible. As a result, LTS can discover both primitive skills and transitional skills. Furthermore, LTS achieves a great success in the smooth transition from one primitive skill to another and exhibits its potential in learning a large continuous skill space. Extensive experiments demonstrate the effectiveness of our LTS in the discovery of diverse skills and the smooth transition between skills.

REFERENCES

Andrew G Barto. Intrinsic motivation and reinforcement learning. In *Intrinsically motivated learning in natural and artificial systems*, pp. 17–47. Springer, 2013.

Nuttapong Chentanez, Andrew G Barto, and Satinder P Singh. Intrinsically motivated reinforcement learning. In *Advances in neural information processing systems*, pp. 1281–1288, 2005.

Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *arXiv preprint arXiv:1802.06070*, 2018.

Mikhail Frank, Jrgen Leitner, Marijn Stollenga, Alexander Frster, and Jrgen Schmidhuber. Curiosity driven reinforcement learning for motion planning on humanoids. *Frontiers in Neurorobotics*, 7: 25, 2013.

Carlos E Garcia, David M Prett, and Manfred Morari. Model predictive control: theory and practicea survey. *Automatica*, 25(3):335–348, 1989.

Dibya Ghosh, Avi Singh, Aravind Rajeswaran, Vikash Kumar, and Sergey Levine. Divide-and-conquer reinforcement learning. 2018.

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

Shixiang Gu, Ethan Holly, Timothy Lillicrap, and Sergey Levine. Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates. In *IEEE International Conference on Robotics Automation*, 2017.

Tuomas Haarnoja, Aurick Zhou, Pieter Abbeel, and Sergey Levine. Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. 2018.

Nicolas Heess, T. B. Dhruva, Srinivasan Sriram, Jay Lemmon, and David Silver. Emergence of locomotion behaviours in rich environments. 2017.

Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *Computer Science*, 2014.

Youngwoon Lee, Shao-Hua Sun, Sriram Somasundaram, Edward S Hu, and Joseph J Lim. Composing complex skills by learning transition policies. 2018.

Sergey Levine, Chelsea Finn, Trevor Darrell, and Pieter Abbeel. End-to-end training of deep visuomotor policies. *The Journal of Machine Learning Research*, 17(1):1334–1373, 2016.

D. Y. Little and F. T. Sommer. Learning and exploration in action-perception loops. *Frontiers in Neural Circuits*, 7(7):37, 2013.

Josh Merel, Yuval Tassa, Sriram Srinivasan, Jay Lemmon, Ziyu Wang, Greg Wayne, and Nicolas Heess. Learning human behaviors from motion capture by adversarial imitation. *arXiv preprint arXiv:1707.02201*, 2017.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529, 2015.

Pierre Yves Oudeyer and Frederic Kaplan. What is intrinsic motivation? a typology of computational approaches. *Frontiers in Neurorobotics*, 1(6):6, 2007.

Xue Bin Peng, Glen Berseth, KangKang Yin, and Michiel Van De Panne. Deeploco: Dynamic locomotion skills using hierarchical deep reinforcement learning. *ACM Transactions on Graphics (TOG)*, 36(4):41, 2017.

Xue Bin Peng, Michael Chang, Grace Zhang, Pieter Abbeel, and Sergey Levine. Mcp: Learning composable hierarchical control with multiplicative compositional policies. *arXiv preprint arXiv:1905.09808*, 2019.

Martin Riedmiller, Roland Hafner, Thomas Lampe, Michael Neunert, Jonas Degrave, Tom Van de Wiele, Volodymyr Mnih, Nicolas Heess, and Jost Tobias Springenberg. Learning by playing-solving sparse reward tasks from scratch. *arXiv preprint arXiv:1802.10567*, 2018.

Richard M Ryan and Edward L Deci. Intrinsic and extrinsic motivations: Classic definitions and new directions. *Contemporary educational psychology*, 25(1):54–67, 2000.

Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. *arXiv preprint arXiv:1907.01657*, 2019.

David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529(7587):484, 2016.

Satinder Singh, Richard L Lewis, Andrew G Barto, and Jonathan Sorg. Intrinsically motivated reinforcement learning: An evolutionary perspective. *IEEE Transactions on Autonomous Mental Development*, 2(2):70–82, 2010.

Emanuel Todorov, Tom Erez, and Yuval Tassa. Mujoco: A physics engine for model-based control. In *IEEE/RSJ International Conference on Intelligent Robots Systems*, 2012.

Xin Wang, Qiuyuan Huang, Asli Çelikyilmaz, Jianfeng Gao, Dinghan Shen, Yuan-Fang Wang, William Yang Wang, and Lei Zhang. Reinforced cross-modal matching and self-supervised imitation learning for vision-language navigation. *CoRR*, abs/1811.10092, 2018. URL `http://arxiv.org/abs/1811.10092`.

Ziyu Wang, Josh S Merel, Scott E Reed, Nando de Freitas, Gregory Wayne, and Nicolas Heess. Robust imitation of diverse behaviors. In *Advances in Neural Information Processing Systems*, pp. 5320–5329, 2017.

David Warde-Farley, Tom Van de Wiele, Tejas Kulkarni, Catalin Ionescu, Steven Hansen, and Volodymyr Mnih. Unsupervised control through non-parametric discriminative rewards. *arXiv preprint arXiv:1811.11359*, 2018.

Robert W White. Motivation reconsidered: The concept of competence. *Psychological review*, 66 (5):297, 1959.

Feng Wu and Xiaoping Chen. Solving large-scale and sparse-reward dec-pomdps with correlation-mdps. In *Robot Soccer World Cup*, pp. 208–219. Springer, 2007.

Appendices

A METHODOLOGY

A.1 THE OBJECTIVE FUNCTION

We define $\hat{f}^s_{i,j,k} = \alpha \cdot f^s_{i,j,k} + c$, where a and c are induced to constrain $0 < \hat{f}^s_{i,j,k} < 1$. Based on $\hat{f}^d_{i,j,k} = 1 - \hat{f}^s_{i,j,k}$, we have

$$\begin{aligned}
\alpha \cdot \mathrm{L}(\theta) &= \mathop{\mathbb{E}}_{i,j,k}\Big[\ |\alpha \cdot \mathrm{I}(S_{i,j,k}; S^p_i | z_{i,j,k}, \omega_i, \omega_j) - \hat{f}^s_{i,j,k} + c)|\ \Big] \\
&= \mathop{\mathbb{E}}_{i,j,k}\Big[\ |\alpha \log p(s^p_i|s_t) - \alpha \log p(s^p_i) - \hat{f}^s_{i,j,k} + c|\ \Big] \\
&\approx \mathop{\mathbb{E}}_{i,j,k}\Big[\ |\alpha \log p(\omega_i|s_t) - \alpha \log p(s^p_i) - \hat{f}^s_{i,j,k} + c|)|\ \Big] \\
&= \mathop{\mathbb{E}}_{i,j,k}\Big[\ |1 - (\hat{f}^d_{i,j,k} + \alpha \log p(\omega_i|s_t) - \alpha \log p(s^p_i) + c)|\ \Big],
\end{aligned} \tag{10}$$

Noting that conditional probability $p(s^p_i|s^t_{i,j,k})$ is converted to $p(\omega_i|s_t)$. The reason for this approximation is that the approach could learn the diverse primitive skills. This means different primitive skills represent different primitive states. This objective enables the log-conditional probability $\log p(\omega_i|s_t)$ and the similarity $\hat{f}^s_{i,j,k}$ are of proportional relation. When $(z_{i,j,k} - \omega_i) \cdot (z_{i,j,k} - \omega_j) = 0$, we could learn the diverse primitive skills. When $\hat{f}^s_{i,j,k} \cdot (\hat{f}^s_{i,j,k} - 1) \neq 0$, we evaluate the discrimination $p(\omega_i|s_t)$ using the similarity $\hat{f}^s_{i,j,k}$.

With the help of scaling factor α and constant c, we could keep $\hat{f}^d_{i,j,k} + \alpha \log p(\omega_i|s_t) - \alpha \log p(s^p_i) + c < 1$, so minimizing $L(\theta)$ is equivalent to maximizing

$$\mathrm{F}(\theta) \triangleq \mathop{\mathbb{E}}_{i,j,k}\Big[\ \hat{f}^d_{i,j,k} + \alpha \log p(\omega_i|s_t) + c\ \Big], \tag{11}$$

where we ignore $\mathrm{H}[S^p_i]$.

Defining $f^d_{i,j,k} = \frac{1}{\alpha} \cdot (\hat{f}^d_{i,j,k} + c)$, we have

$$\mathrm{F} = \alpha \cdot (\mathop{\mathbb{E}}_{i,j,k}\big[\log p(\omega_i|s_t) + f^d_{i,j,k}\big]), \tag{12}$$

where the scaling factor could be obtained in the learning rate. So we have

$$f^d_{i,j,k} = \frac{1 - \alpha \cdot f^s_{i,j,k}}{\alpha}. \tag{13}$$

A.2 REMARK

There exists a misleading optimization goal:

$$\begin{aligned}
\mathrm{F}(\theta) &= \mathop{\mathbb{E}}_{i,j,k}(|\mathrm{I}(\Omega; S_{i,j,k}) - f^s_{i,j,k}|) \\
&= \mathop{\mathbb{E}}_{i,j,k}(|\mathrm{H}[\Omega] - \mathrm{H}[\Omega|S_{i,j,k}] - f^s_{i,j,k}|) \\
&\neq \mathbb{E}_{(\omega_i,\omega_j)\sim\Omega, z_{i,j,k}\sim C(\omega_i,\omega_j), s_t\sim\pi(z_{i,j,k})}(|\log p(\omega_i) - \log p(\omega_i|s_t) - f^s_{i,j,k}|).
\end{aligned} \tag{14}$$

A.3 DERIVATION OF THE VARIATION BOUND ON MUTUAL INFORMATION

Here we derive the variational bound:

$$\begin{aligned}
\mathrm{E}_{\omega_i\sim\Omega}(\log p(\omega_i|s_t)) &= \mathrm{E}_{\omega_i\sim\Omega}(\log q_\phi(\omega_i|s_t)) + \alpha KL(p(\omega_i|s_t)|q_\phi(\omega_i|s_t)) \\
&\geq \mathrm{E}_{\omega_i\sim\Omega}(\log q_\phi(\omega_i|s_t))
\end{aligned} \tag{15}$$

B IMPLEMENTATION

B.1 HINDSIGHT AND ONE-HOT ENCODING

We compare the conditional probability given by the discriminator $q_\phi(\omega_i|s_t)$ and the divergence between $z_{i,j,k}$ and specific ω_i. If we want to compare the probability $q_\phi(\omega_{i_1}|s_t)$ and $q_\phi(\omega_{i_2}|s_t)$ in terms of the same transition state s_t controlled by the same skill z ($z = z_{i1,j1,k1} = z_{i2,j2,k2}$) and two different primitive skills ω_{i1} and ω_{i2}, we must wait for the next time step to sample the same z and a different ω in the experience. The efficiency is relatively low. So we utilize the hindsight experience reply mechanism to allow sample-efficient learning form sparse rewards. In our approach, we calculate a distribution of the conditional probability given by the discriminator instead of a single value because that $q_\phi(\omega_i|s_t)$ just constrain the consistency of $z_{i,j,k}$ and ω_i which ignores the consistency to other primitive skills. So we change $q_\phi(\omega_i|s_t)$ to the conditional probability distribution:

$$q_\phi(\Omega, s_t) = [q_\phi(\omega_0|s_t), q_\phi(\omega_1|s_t), ..., q_\phi(\omega_{N-1}|s_t)]^{\mathrm{T}}. \tag{16}$$

By doing this, we could simultaneously constrain the similarity probability distribution given by the discriminator with related to all primitive skills.

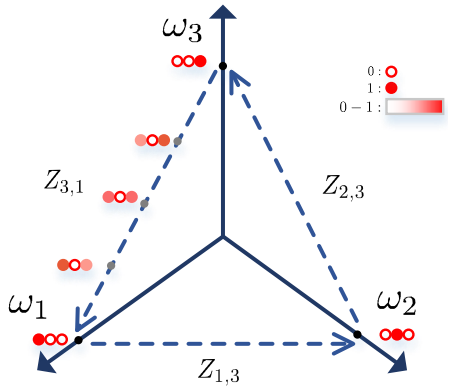


Figure 8: The space of primitive skills and transitional skills.

Moreover, we change the criterion of $f^d_{i,j,k}$ into $f^d_{i,k}$:

$$f^d_{i,k} = [f^d_{i,1,k}, f^d_{i,2,k}, ..., f^d_{i,N-1,k}]^{\mathrm{T}}. \tag{17}$$

Considering transferring two skills from ω_i to ω_j ($\omega_i \neq \omega_j$), the former categorical encoding will cause extra consumption: if $\omega_i - \omega_j \neq \pm 1$, primitive states with related to primitive skills $\{\omega|\omega \in [\omega_i, \omega_j]$ or $[\omega_j, \omega_i]\}$ will occur in the transition states. Assuming $z_{i,j,k} = \omega_i$ for all $z_{i,j,k} \sim p(z_{i,j,k}|\omega_i, \omega_j)$ and $z_{j,i,k} = \omega_j$ for all $z_{j,i,k} \in p(z_{i,j,k}|\omega_j, \omega_i)$, $\hat{f}^s_{i,j,k} = 1$ for all k and $\hat{f}^s_{j,i,k} = 1$ for all k, and the optimization $\mathrm{L}(\theta) = 1 - \alpha \cdot \mathrm{I}(\Omega; S) + c$. Minimizing $\mathrm{F}(\theta)$ is equal to maximizing $\mathrm{I}(\Omega; S)$, which is fully in accordance with DIAYN. Different ω represents different primitive skills because $p(\omega_i|s_t) = 0$ for all $\omega_i \neq \omega$, where $s_t \sim \pi(\omega)$. For the optimal discriminator, there should be $q^*_\phi(\omega_{i_3}|s^{'}_t) = 0$ ($s^{'}_t \sim \pi(\omega_{i_1})$) and $q^*_\phi(\omega_{i_3}|s^{''}_t) = 0$ ($s^{''}_t \sim \pi(\omega_{i_2})$). While if $\omega_{i_1} < \omega_{i_3} < \omega_{i_2}$, $q^*_\phi(\omega_3|s^{'}_t) = 1 - (\omega_{i_3} - \omega_{i_1})$ and $q^*_\phi(\omega_3|s^{''}_t) = 1 - (\omega_{i_2} - \omega_{i_3})$, which is in contrast to the conditional probability of 0. So, we encode $\omega \sim p(\omega)$ with one-hot way:

$$\begin{aligned} \omega_0 &= [1, 0, 0, ..., 0]; \\ \omega_1 &= [0, 1, 0, ..., 0]; \\ &... \\ \omega_{N-1} &= [0, 0, 0, ..., 1]. \end{aligned} \tag{18}$$

And we denote primitive skills and transitional skills as a set $Z^+_{i,j}$:

$$
\begin{aligned}
Z^+_{i,j} &= [\omega_i, z_{i,j,1}, ..., z_{i,j,k}, ..., z_{i,j,K-1}, \omega_j]^{\mathrm{T}} \\
&= \begin{bmatrix}
0 & ... & 1 & ... & 0 & ... & 0 \\
0 & ... & 1-\frac{1}{K} & ... & \frac{1}{K} & ... & 0 \\
.... & & & & & & \\
0 & ... & 1-\frac{k}{K} & ... & \frac{k}{K} & ... & 0 \\
... & & & & & & \\
0 & ... & 0 & ... & 1 & ... & 0
\end{bmatrix}_{(K+1)\times N},
\end{aligned} \tag{19}
$$

where the value of i-th and j-th column keeps decreasing and increasing respectively. Other column always keep 0, which could constrain the incoherence between transition states and other primitive skills. Without causing any misunderstanding, following $z_(i,j,k)$ all comes from $Z^+_{i,j}$. For transition, we assure that the change of $z_{i,j,k}$ only happens on the corresponding dimension, which overcomes the conflict caused by categorical encoding. As show in Fig.8, all transition skills in $Z_{3,1}$ and primitive skills ω_3 are orthogonal. The transition only reflects on the plane defined by the corresponding primitive skills. In fact, there is more than one transitional path, which can be a directed line or any directed curve. As in Figure 7(b), we can find more than one transitional pathes.

B.2 Reward Occupied with KL Divergence

Because that diverse skills (primitive skills) play a vital role, we also add KL divergence ($\mathrm{D}_{KL}(q_\phi(\mathbf{f}^s_{i,k}||\Omega|s_t))$) in the optimization. So the occupied reward is expressed as

$$
r_t = \delta \cdot \|q_\phi(\Omega|s_{t+1}) + \mathbf{f}^d_{i,k}\| - (1-\delta) \cdot \mathrm{D}_{KL}(\cdot q_\phi(\Omega|s_{t+1})||\mathbf{f}^s_{i,k}), \tag{20}
$$

where δ is a scaling factor for controlling the effect of the MSE term and KL divergence, both of them guarantee that the discriminator can distinguish the divergence or similarity between the primitive skills Ω and the states. Because we encode ω with one-hot way and constrain $\sum z_{i,j,k} = 1$, so the similarity $\mathbf{f}^s_{i,k} = z_{i,j,k}$.

C Experimental Environment

The experiments were carried out over three opened reinforcement learning environments (CartPole[5], MountainCar[6], and Pendulum[7]).

C.1 CartPole

In this environment, a pole is attached by an un-actuated joint to a cart, which moves along a frictionless track. The system is controlled by applying a force of +1 or -1 to the cart. The pendulum starts upright, and the goal is to prevent it from falling over by increaseing and reducing the cart's velocity. The episode ends when the pole is more than 15 degrees from vertical, or the cart moves more than 2.4 units from the center.

C.2 MountainCar

A car is on a one-dimensional track, positioned between two "mountains". The goal is to drive up the mountain on the right; however, the car's engine is not strong enough to scale the mountain in a single pass. Therefore, the only way to succeed is to drive back and forth to build up momentum.

C.3 Pendulum

The inverted pendulum swingup problem is a classic problem in the control literature. The problem of the pendulum starts in a random position, and the goal is to swing it up so it stays upright.

[5] https://gym.openai.com/envs/CartPole-v0/

[6] https://gym.openai.com/envs/MountainCar-v0/

[7] https://gym.openai.com/envs/Pendulum-v0/

D HYPERPARAMETERS

For all RL algorithm in our experiments, we use the SAC (Haarnoja et al. (2018)) as implementation framework. The hyperparameters are summed up in the Table 1 and we use ADAM (Kingma & Ba (2014)) optimizer.

Table 1: Parameter setting

Parameters	Description	Value
H	hidden state size	32
layer	layer count	3
epoch	eposide size	12
vf_lr	value network learning rate	1e-5
dc_lr	discriminator network learning rate	5e-4
pi_lr	policy network learning rate	3e-4

E VISUALIZING PRIMITIVE SKILLS

In order to better represent the distinction between skills, we did various experiments and finally determined some optimal observations as feature vector for each skill (see Table 2). The following experiments show that it makes sense to calculate the statistical characteristics of skills' characteristics to represent a skill. Three experiments' performance was shown in Figure 9, Figure 10 and Figure 11.

Table 2: Selection of skill feature.

RL enviroment	Observations	Selected as skill feature
CartPole	0: Cart Position; 1: Cart Velocity 2: Pole Angle 3: Pole Velocity at Tip	2: Pole Angle
MountainCar	0: Position 1: Velocity	0: Position
Pendulum	0: cos(Angle) 1: sin(Angle) 2: speed	1: sin(Angle)

F VISUALIZING TRANSITION PROCESS

For all Cartpole, MountainCar, and Pendulum, we get 4 primitives skills, and control variant z only takes some fixed values (0.25 0.5 0.75) during training. However, at the test phase of transition, z is taken every 0.1. So we can obtain 9 transition skills between any two primitive skills($z = [0, 1]$) (See Figure 12, Figure 13, and Figure 14,).

G TRANSITION COMPARISON

Comprehensive study including motion trail and statistical characteristics on MountainCar was reported in Figure 15.

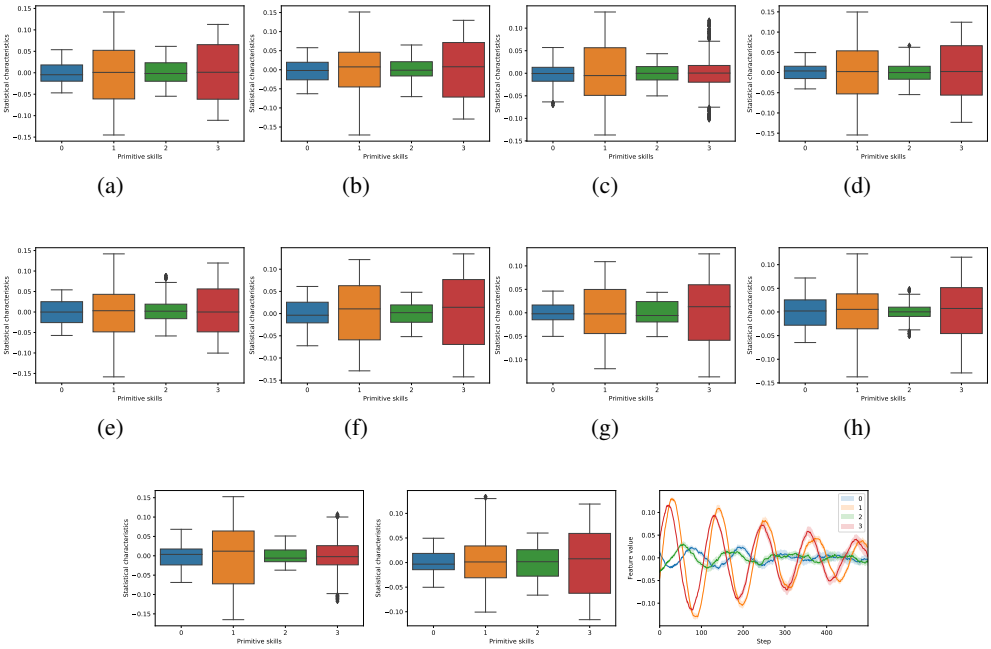


Figure 9: **Cartpole primitive skills**. (a) - (j) stand for 10 random trials with 4 primitive skills for each, and different skills are distinguished by Boxplot. (k) shows the skill in time domain from just one trial.

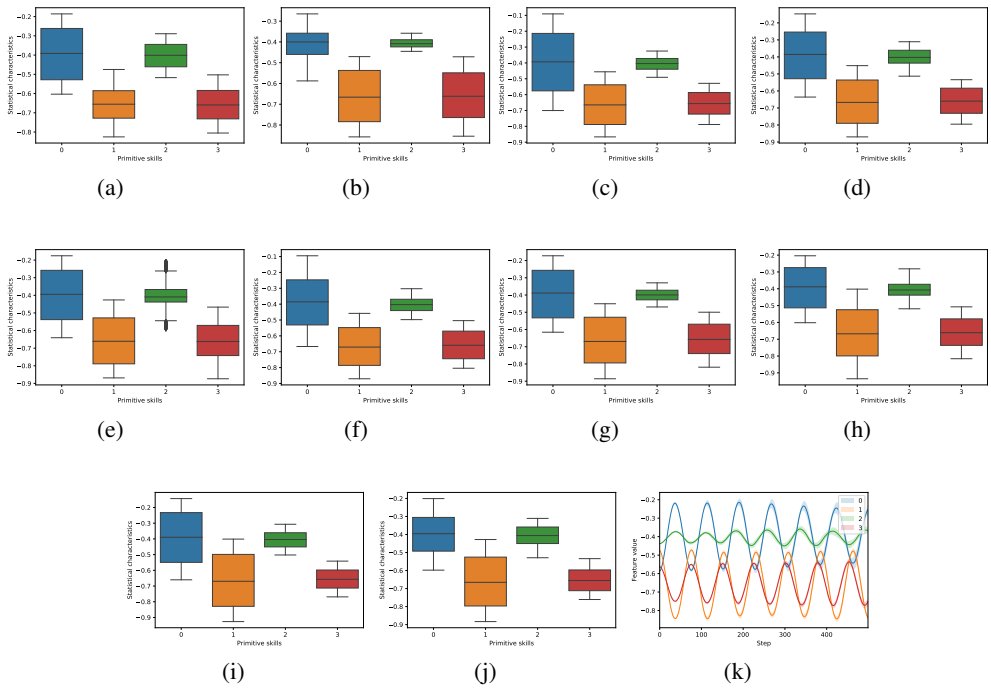


Figure 10: **Mountain Car primitive skills**. (a) - (j) stand for 10 random trials with 4 primitive skills for each, and different skills are distinguished by Boxplot. (k) shows the skill in time domain from just one trial.

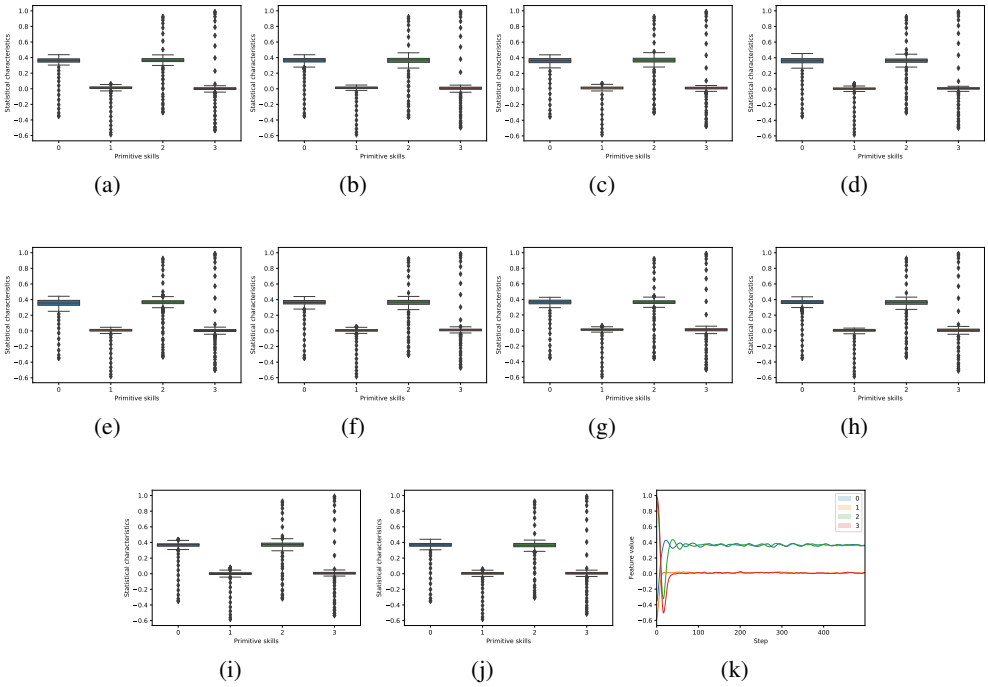


Figure 11: **Pendulum primitives skills**. (a) - (j) stand for 10 random trials with 4 primitive skills for each, and different skills are distinguished by Boxplot. (k) shows the skill in time domain from just one trial.

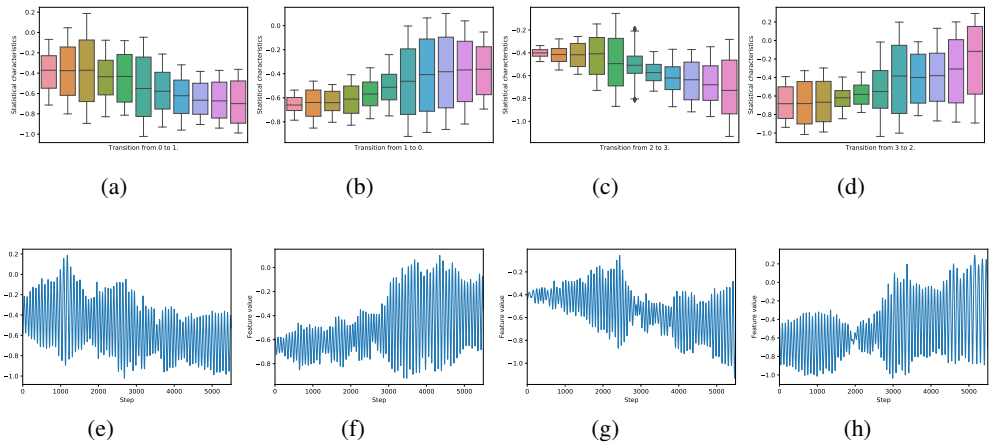
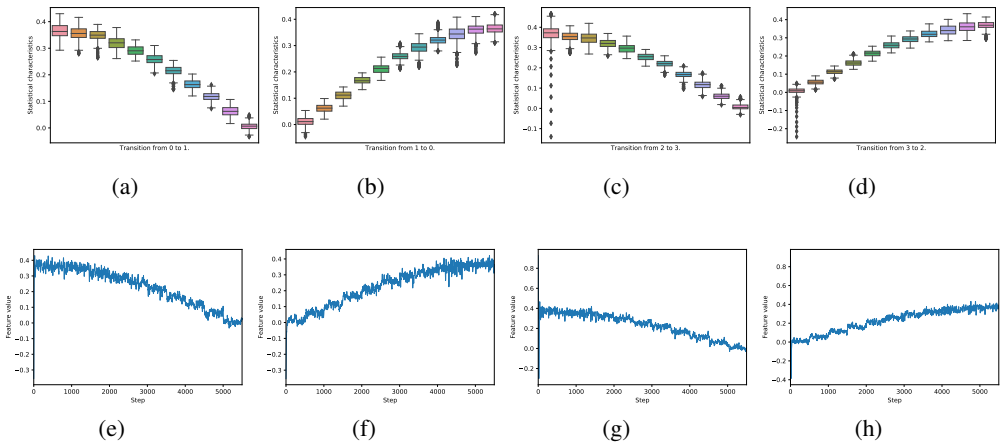


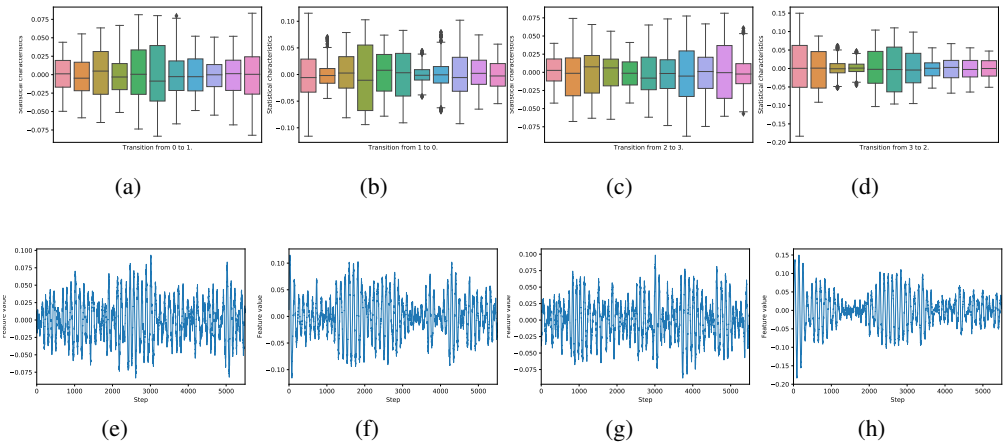
Figure 12: **Mountain Car transition process**. (a) (e), (b) (f), (c) (g), and (d) (g) are from different 4 trials respectively. Each transition skill holds 500 steps and then transfers the final state to the next skill.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 13: **Pendulum transition process**. (a) (e), (b) (f), (c) (g), and (d) (g) are from different 4 trials respectively. Each transition skill holds 500 steps and then transfers the final state to the next skill.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 14: **Cartpole transition process**. (a) (e), (b) (f), (c) (g), and (d) (g) are from different 4 trials respectively. Each transition skill holds 500 steps and then transfers the final state to the next skill.

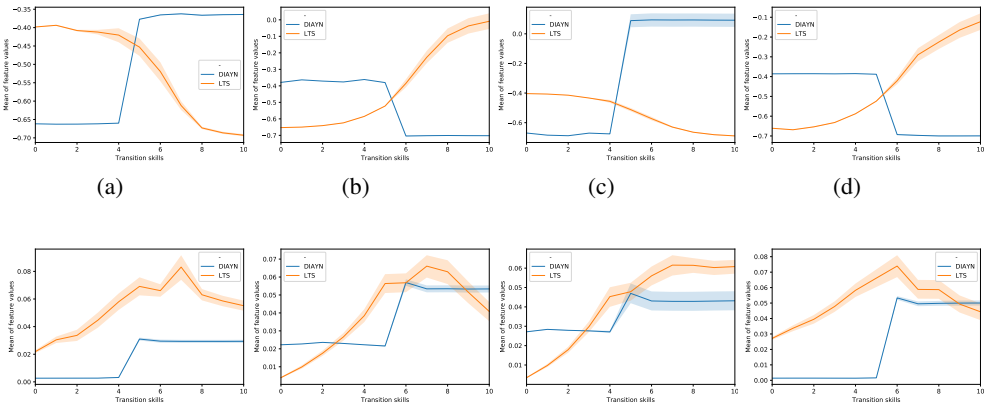


Figure 15: **Transition comparison on MountainCar**. The subgraphs (a),(b),(c),(d) represent the mean of features in terms of the transition skills, and subgraphs (e),(f),(g),(h) represent the variance of features. Compared to DIAYN, LTS performs a continuous transition.