Metadata of the chapter that will be visualized in SpringerLink

Book Title	Pattern Recognition and Computer Vision	
Series Title		
Chapter Title	Continuous Exploration via Multiple Perspectives in Sparse Reward Environment	
Copyright Year	2024	
Copyright HolderName	The Author(s), under exclusive license to Springer Nature Singapore Pte Ltd.	

Author	Family Name	**Chen**
	Particle	
	Given Name	**Zhongpeng**
	Prefix	
	Suffix	
	Role	
	Division	School of Artificial Intelligence
	Organization	University of Chinese Academy of Sciences
	Address	Beijing, 100049, China
	Division	
	Organization	Institute of Automation, Chinese Academy of Sciences
	Address	Beijing, 100190, China
	Email	chenzhongpeng2021@ia.ac.cn
Corresponding Author	Family Name	**Guan**
	Particle	
	Given Name	**Qiang**
	Prefix	
	Suffix	
	Role	
	Division	
	Organization	Institute of Automation, Chinese Academy of Sciences
	Address	Beijing, 100190, China
	Email	qiang.guan@ia.ac.cn

Abstract	Exploration is a major challenge in deep reinforcement learning, especially in cases where reward is sparse. Simple random exploration strategies, such as ϵ-greedy, struggle to solve the hard exploration problem in the sparse reward environment. A more effective approach to solve the hard exploration problem in the sparse reward environment is to use an exploration strategy based on intrinsic motivation, where the key point is to design reasonable and effective intrinsic reward to drive the agent to explore. This paper proposes a method called CEMP, which drives the agent to explore more effectively and continuously in the sparse reward environment. CEMP contributes a new framework for designing intrinsic reward from multiple perspectives, and can be easily integrated into various existing reinforcement learning algorithms. In addition, experimental results in a series of complex and sparse reward environments in MiniGrid demonstrate that our proposed CEMP method achieves better final performance and faster learning efficiency than ICM, RIDE, and TRPO-AE-Hash, which only calculate intrinsic reward from a single perspective.

Keywords (separated by '-')	Reinforcement Learning - Exploration Strategy - Sparse Reward - Intrinsic Motivation

Continuous Exploration via Multiple Perspectives in Sparse Reward Environment

Zhongpeng Chen[1,2] and Qiang Guan[2(✉)]

[1] School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China
[2] Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China
{chenzhongpeng2021,qiang.guan}@ia.ac.cn

Abstract. Exploration is a major challenge in deep reinforcement learning, especially in cases where reward is sparse. Simple random exploration strategies, such as ϵ-greedy, struggle to solve the hard exploration problem in the sparse reward environment. A more effective approach to solve the hard exploration problem in the sparse reward environment is to use an exploration strategy based on intrinsic motivation, where the key point is to design reasonable and effective intrinsic reward to drive the agent to explore. This paper proposes a method called CEMP, which drives the agent to explore more effectively and continuously in the sparse reward environment. CEMP contributes a new framework for designing intrinsic reward from multiple perspectives, and can be easily integrated into various existing reinforcement learning algorithms. In addition, experimental results in a series of complex and sparse reward environments in MiniGrid demonstrate that our proposed CEMP method achieves better final performance and faster learning efficiency than ICM, RIDE, and TRPO-AE-Hash, which only calculate intrinsic reward from a single perspective.

Keywords: Reinforcement Learning · Exploration Strategy · Sparse Reward · Intrinsic Motivation

1 Introduction

The goal of reinforcement learning is to train an effective policy, which drives the agent to obtain the maximum cumulative reward in the environment. The generation of the agent's policy primarily depends on the reward provided by the environment. Therefore, whether the agent can learn an effective policy is closely related to its interaction with the environment in order to capture rewards. Classical algorithms like DQN [1] and PPO [2] have demonstrated outstanding performance in the dense reward environment by using simple exploration strategies such as ϵ-greedy, entropy regularization, and Boltzman exploration. However, in the sparse reward environment, these simple exploration strategies face difficulties in guiding the agent to reach the target state and obtain effective reward.

Q. Liu et al. (Eds.): PRCV 2023, LNCS 14427, pp. 1–12, 2024.
https://doi.org/10.1007/978-981-99-8435-0_5

The agent will be unable to update the policy due to long-term inability to receive effective reward. Hence, this paper will concentrate on designing effective exploration strategies to help the agent better resolving hard exploration problem in the sparse reward environment.

In the sparse reward environment, the agent needs to continuously complete a series of correct decisions in order to obtain few effective rewards. However, a simple random exploration strategy is difficult for the agent to explore a trajectory that can complete the task. Exploration strategies based on intrinsic motivation have seen significant development in recent years and can effectively address the exploration challenges faced by the agent in the sparse reward environment. Intrinsic motivation comes from the concept of ethology and psychology [3]. During the learning process, higher organisms often spontaneously explore the unfamiliar and unknown environment without extrinsic stimuli to enhance their ability to survive in the environment. Exploration strategies based on intrinsic motivation formalize the concept of intrinsic motivation as an intrinsic reward by measuring the novelty of the state, thereby utilizing the intrinsic reward to drive the agent to spontaneously explore more unknown space in the sparse reward environment and increase the likelihood of the agent solving tasks. The novelty of a state is related to the number of times the agent accesses the state. Currently, there are two main perspectives for measuring the novelty of a state: global perspective and local perspective, as explained below.

Global Perspective: Use all historical samples collected by the agent from the environment to measure the novelty of a state.

Local Perspective: Evaluate the novelty of a state only using the samples collected in the current episode where the agent is interacting with the environment, without considering historical samples.

The drawback of measuring the novelty of a state from global perspective is that the novelty of the state and its corresponding intrinsic reward gradually decay during the training process. The decaying intrinsic reward cannot drive the agent to continuously explore in the environment. At the same time, measuring the novelty of a state from global perspective alone is not conducive to the agent discovering more novel trajectories. When a novel trajectory appears, it will be diluted by ordinary trajectories in history. Measuring the novelty of a state from local perspective can enable the agent to discover more novel trajectories and access more unknown states within the current episode. However, in the absence of global information, relying solely on local perspective to measure the novelty of a state may blindly and optimistically encourage the agent to explore unknown states.

Taking into account the characteristics of calculating intrinsic reward from different perspectives, we propose a method called CEMP (Continuous Exploration via Multiple Perspectives) that enables the agent to better explore in the sparse reward environment. CEMP calculates intrinsic reward from both global and local perspectives and integrates them to better drive the agent's exploration in the sparse reward environment. To summarize, the main contributions of our work are as follows:

1. We propose a method called CEMP that enables the agent to explore better in the sparse reward environment. This method combines intrinsic reward calculated from different perspectives to obtain a new intrinsic reward, which does not decay gradually during the training process and can continuously drive the agent's exploration in the sparse reward environment. Moreover, our proposed CEMP method can be easily integrated into various existing reinforcement learning algorithms.
2. The majority works that calculate intrinsic reward from local perspective only measure the novelty of a state within the current episode. In contrast, our proposed CEMP method can measure the novelty of a state from local perspective across multiple episodes and can flexibly control the range of the local perspective through parameters.
3. Experimental results in a series of complex and sparse reward environments in MiniGrid show that our proposed CEMP method achieves better final performance and faster learning efficiency compared to methods such as ICM, RIDE, and TRPO-AE-Hash that only calculate intrinsic reward from a single perspective.

2 Related Works

There are two main methods for calculating intrinsic reward from global perspective: state count and prediction error method.

The state count method extends UCB exploration strategy to the high dimensional state environment through pseudo-count or indirect count methods. DQN-PixelCNN [4] indirectly derives the pseudo-count of a state through the density probability model, which is modeled from the raw state space, but it is difficult to directly model a density probability model in the raw state space. To solve this issue, ϕ-EB [5] models a density probability model in a low dimensional feature space of the raw state. TRPO-AE-Hash [6] discretizes the raw state into low dimensional hash code using SimHash [7] and then computes the pseudo-count of a state based on its hash code. DORA [8] constructs an indirect count index E-value, which can become an effective generalization counter of (s_t, s_{t+1}). RND [9] uses the prediction error between the predictor network and the target network as an indirect measure of state count.

The prediction error method uses the prediction error between the next predicted state $\hat{s}_{t+1}$ output by a forward model and the ground-truth s_{t+1} as intrinsic reward. The key aspect of the prediction error method is how to construct a forward model. Dynamic-AE [10] reconstructs the raw state by using an autoencoder and constructs a forward model based on a low-dimensional feature space extracted by the middle layer of the autoencoder. In fact, using the features extracted by the middle layer of the autoencoder to train a forward model is highly susceptible to environmental noise. ICM [11] learns features of the raw state by utilizing an inverse model, which predicts the action a_t from the state (s_t, s_{t+1}) and extracts the intermediate layer output as low-dimensional features to construct a forward model. Inverse model only extracts feature from the raw

state that is related to the agent's actions, which partially reduces the influence of noise from the raw state. Disagreement [12] trains multiple forward models based on random feature subspace and uses the variance of the predicted values of these multiple forward models as intrinsic reward.

The methods introduced above measure the novelty of a state from global perspective, while the methods below measure the novelty of a state from local perspective. DeepCS [13] incentivizes the agent to explore as many unknown states as possible within the same episode by setting intrinsic reward to 1 for unreachable states and 0 for visited states respectively. ECO [14] measures the novelty of a state by the reachability between states in the same episode. RIDE [15] directly uses the difference between two consecutive states as intrinsic reward.

3 Method

3.1 Continuous Exploration via Multiple Perspectives

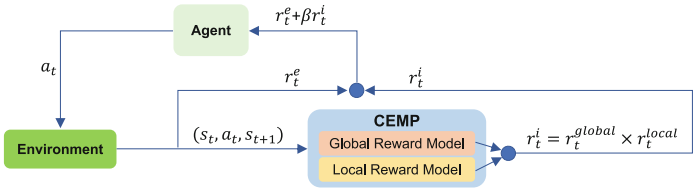

Fig. 1. Our proposed CEMP method: the design details of the Global Reward Model and the Local Reward Model can be found in Sect. 3.2 and 3.3, respectively.

As illustrated in Fig. 1, our proposed CEMP (Continuous Exploration via Multiple Perspectives) method calculates intrinsic reward from both global and local perspectives, and combines the reward from both perspectives by multiplication to obtain the new intrinsic reward r_t^i, as shown in Eq. (1). Next, the final reward that guides the update of the agent's policy is calculated by linearly weighting the new intrinsic reward r_t^i and the extrinsic reward r_t^e provided by the environment in terms of Eq. (2), where β is the weighting coefficient between both rewards.

$$r_t^i = r_t^{global} \times r_t^{local} \tag{1}$$

$$r_t = r_t^e + \beta r_t^i \tag{2}$$

We will use prediction error-based method to design the global intrinsic reward, and use state count method based on hash-discretization to design the local intrinsic reward. The specific reasons for these choices are as follows:

1. The prediction error-based method evaluates the novelty of a state from global perspective by using the prediction error of a forward model modeled with deep neural network, which is well-suited for processing large-scale data and has strong discriminative power when facing with large-scale states, thus, it is suitable for evaluating the novelty of a state from global perspective. However, it is not suitable for evaluating the novelty of a state from local perspective, because the forward model trained with a small dataset may have difficulty learning effective knowledge and distinguishing the novelty of a state due to insufficient data.
2. The state count method based on hash-discretization can quickly and accurately count different states in a small dataset by using the hash code of the raw state. However, in the case with large-scale states, it may not be able to distinguish different states due to the limited number of hash code bits.

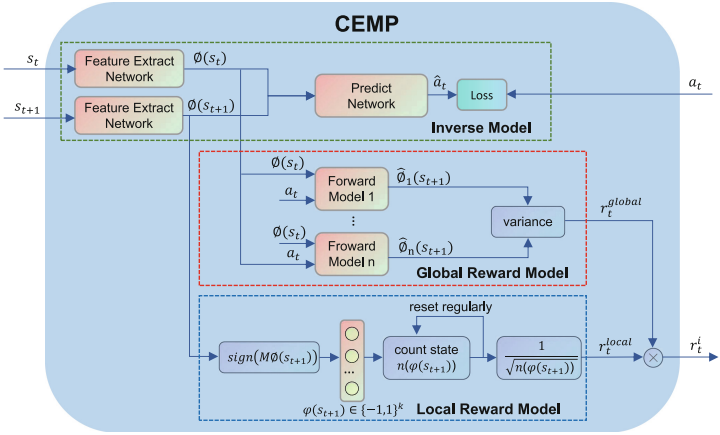

Fig. 2. The design details of the Global Reward Model and the Local Reward Model.

3.2 Global Reward Model

We first use the inverse model adopted in ICM [11] method to extract a low dimensional feature of the raw state, as shown in the green box in Fig. 2. Extracting a low-dimensional feature through the inverse model can capture the feature information related to the agent's actions as much as possible and reduce the impact of noise in the raw state. As shown in the red box in Fig. 2, we then use all the samples collected by the agent in the environment to train n forward models in the inverse feature space ϕ. The variance of the predicted values from these n forward models is then used as the intrinsic reward under global perspective, as shown in Eq. (3). The ensemble of these n forward models can further reduce the influence of state noise.

$$r_t^{global} = variance(\hat{\phi}_1(s_{t+1}), \hat{\phi}_2(s_{t+1}), ..., \hat{\phi}_n(s_{t+1})) \tag{3}$$

3.3 Local Reward Model

As shown in the blue box in Fig. 2, we use feature extract network of the inverse model to extract the inverse feature $\phi(s_{t+1})$ of the raw state s_{t+1}, and then use SimHash method to discretize $\phi(s_{t+1})$ into a k-dimensional binary hash code $\varphi(s_{t+1})$, as shown in Eq. (4), where M is a $k \times d$ matrix sampled from standard normal distribution, d is the dimension of $\phi(s_{t+1})$. Next, we count the state based on its hash code $\varphi(s_{t+1})$. In the counting process, we only count the access times of each state in N consecutive states, and reset the previous count information when the agent enters the next new N consecutive states. Equation (5) provides the relationship between the count of a state and corresponding intrinsic reward under local perspective. The local intrinsic reward r_t^{local} will incentivize the agent to frequently explore more novel states within the current N consecutive states, which is conducive to the agent discovering more novel trajectories.

$$\varphi(s_{t+1}) = sign(M\phi(s_{t+1})) \in \{-1, 1\}^k \tag{4}$$

$$r_t^{local} = \frac{1}{\sqrt{n(\varphi(s_{t+1}))}} \tag{5}$$

Most existing works that calculate intrinsic reward from local perspective only measure the novelty of a state within the same episode, while our proposed CEMP method measures the novelty of a state over N consecutive states, where N is an adjustable parameter. By using a larger N, samples will come from several different episodes. Therefore, the CEMP method provides more flexibility in controlling the range of the local perspective through parameter N.

The intrinsic reward calculated from local perspective does not gradually decay during the training process, providing the agent with a more continuous and stable exploration signal. This can compensate for the drawback of intrinsic reward calculated from global perspective that gradually decays over time, while also improving the agent's ability to discover more novel trajectories.

Our proposed CEMP method can be easily integrated into any reinforcement learning algorithm to enhance the agent's exploration ability in the sparse reward environment.

4 Experiment

4.1 Comparison Algorithms and Evaluation Metrics

We compared our proposed CEMP method with three intrinsic motivation-based exploration strategies: ICM [11], TRPO-AE-Hash [6] and RIDE [15]. ICM and TRPO-AE-Hash calculate intrinsic reward from global perspective, while RIDE calculates intrinsic reward from local perspective. Different reinforcement learning algorithms are used as baseline algorithms in the original papers of ICM, TRPO-AE-Hash, and RIDE. In order to ensure fairness in the comparison experiments, we use PPO as the baseline algorithm to reproduce the three exploration

strategies and compare their performance with our proposed CEMP method. The CEMP method also uses PPO as the baseline algorithm.

We use the mean and standard deviation of the average reward over five different seeds as metrics to compare the performance of different exploration strategies. The mean of the average reward reflects the best performance that each exploration strategy can achieve, while the standard deviation of the average reward reflects the stability of the exploration strategy.

4.2 Network Architectures and Hyperparameters

PPO. In the PPO algorithm we reproduced, the Critic and Actor share a feature extract network, which consists of four linear layers with 256, 128, 64, and 64 nodes, respectively. Both the Critic and Actor consist of one linear layer with 64 nodes. The Critic and Actor take the output of the feature extract network as input and output value estimate and action probability distribution for the current state s_t. Other hyperparameters of the PPO algorithm we reproduced are shown in Table 1.

Table 1. Other hyperparameters of the PPO algorithm we reproduced.

Hyperparameter Name	Value
learning rate	0.0003
value loss weight	0.5
entropy loss weight	0.001
discount factor	0.99
λ, general advantage estimation	0.95
activation function	ReLU
optimizer	Adam

CEMP. As shown in Fig. 2, in our proposed CEMP method, the feature extract network in the inverse model consists of three linear layers with 64, 64, and 128 nodes, respectively. The network used to predict actions in the inverse model consists of one linear layer with 512 nodes. All forward models consist of two linear layers with 512 nodes. Other hyperparameters of CEMP method are summarized in Table 2.

ICM, RIDE, and TRPO-AE-Hash. ICM [11] trains a forward model based on the inverse feature space ϕ extracted by the inverse model. The difference between ICM and the Global Reward Model in CEMP lies in the number of forward model and the way intrinsic reward is calculated. ICM trains only one forward model and uses the prediction error of the forward model as the intrinsic

Table 2. Other hyperparameters of our proposed CEMP method.

Hyperparameter Name	Value
n, number of forward model	5
$step_{max}$, max step of one episode, determined by the environment	/
β, weighting coefficient of intrinsic and extrinsic reward	$10/step_{max}$
k, the dimension of hash code	16
N, adjust the range of local perspective	3200
m, batch size	320
learning rate	0.0003
T_{max}, max step of training	3×10^6
activation function	ReLU
optimizer	Adam

reward, as shown in Eq. (6), while the Global Reward Model in CEMP trains n forward models and uses the variance of the predicted values from n forward models as the intrinsic reward.

$$r_t^i = ||\phi(s_{t+1}) - f(\phi(s_t), a_t)||_2^2 \tag{6}$$

RIDE [15] directly uses the difference between the inverse features of two consecutive states as the intrinsic reward, as shown in Eq. (7). Here, ϕ represents the feature extract network in the inverse model, and $n(\varphi(s_{t+1}))$ is the pseudo-count of s_{t+1} in the current episode. The calculation method of $\varphi(s_{t+1})$ is consistent with the Local Reward Model used in CEMP.

$$r_t^i = \frac{||\phi(s_{t+1}) - \phi(s_t)||_2}{\sqrt{n(\varphi(s_{t+1}))}} \tag{7}$$

TRPO-AE-Hash [6] calculates the intrinsic reward in the same way as the Local Reward Model in the CEMP method. The only difference is that TRPO-AE-Hash omits the step of periodically resetting the count of a state, so it calculates the intrinsic reward from global perspective.

To ensure a fair comparison, all network architectures and hyperparameters used in the ICM, RIDE, and TRPO-AE-Hash methods that we reproduced are kept consistent with those used in our proposed CEMP method.

4.3 Experimental Results

Comparison with Other Methods. We select 9 sparse reward environments from MiniGrid for comparative experiments, which contain a total of 5 different tasks. Among the 5 different tasks, the tasks of DoorKey and LavaCrossing each have 3 environments of gradually increasing difficulty (difficulty ranking: DoorKey-5x5 < DoorKey-8x8 < DoorKey-16x16; LavaCrossingS9N1 < LavaCrossingS9N3 < LavaCrossingS11N5). The learning curves of

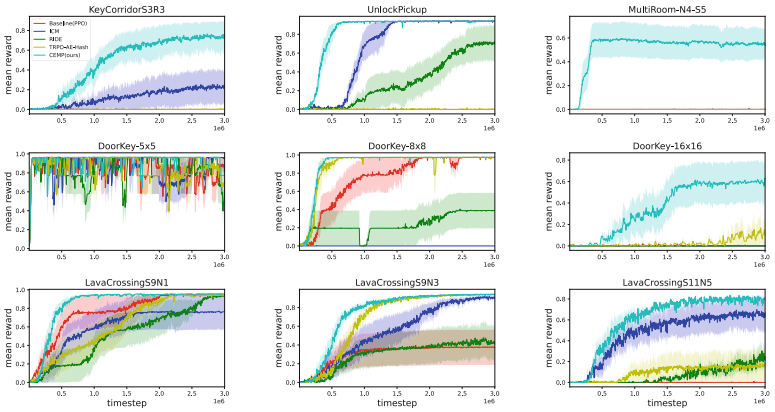

Fig. 3. Performance of each method in different sparse reward environments.

different methods are shown in Fig. 3, from which we can draw the following conclusions:

1. In some environments with lower difficulty levels, such as LavaCrossingS9N1 and DoorKey-5x5, ICM, RIDE, and TRPO-AE-Hash can achieve the same final performance as CEMP. However, our proposed CEMP method has a higher learning efficiency and more stable performance.
2. In the DoorKey and LavaCrossing tasks, as the difficulty increases, the performance of ICM, RIDE, and TRPO-AE-Hash gradually deteriorates, but our proposed CEMP method can still maintain good performance.
3. In more challenging environments, such as MultiRoom-N4-S5 and DoorKey-16x16, ICM, RIDE, and TRPO-AE-Hash are unable to complete the task, but our proposed CEMP method can still perform well and obtain a high average reward.

We evaluate the final policies of the agent obtained by various methods, and the evaluation results are summarized in Table 3. The evaluation metrics in the table are the mean and standard deviation of rewards obtained by each method in five different seeds. It can be seen from Table 3 that our proposed CEMP method has significantly better performance and is more stable than ICM, RIDE, and TRPO-AE-Hash that calculate intrinsic reward from only a single perspective.

Analysis of Local and Global Intrinsic Reward. From Fig. 4, we can see that as the training progresses, the global intrinsic reward gradually decays, while the local intrinsic reward remains in a relatively stable range and does not decay during the training process. Thus, the local intrinsic reward can drive the agent to continuously explore in the sparse reward environment, discover more novel states and trajectories, and compensate for the disadvantage of the global intrinsic reward, which gradually decays and cannot consistently drive the agent to explore.

Table 3. Comprehensive performance of each method in different environments.

mean ± std	PPO	ICM	RIDE	TRPO-AE-Hash	CEMP (ours)
KeyCorridorS3R3	0 ± 0	0.183 ± 0.365	0 ± 0	0 ± 0	$\mathbf{0.733 \pm 0.367}$
UnlockPickup	0 ± 0	0.947 ± 0.006	0.754 ± 0.377	0 ± 0	$\mathbf{0.949 \pm 0.004}$
MultiRoom-N4-S5	0 ± 0	0 ± 0	0 ± 0	0 ± 0	$\mathbf{0.594 \pm 0.298}$
DoorKey-5x5	0.962 ± 0.005	0.964 ± 0.005	0.964 ± 0.005	0.964 ± 0.005	$\mathbf{0.967 \pm 0.003}$
DoorKey-8x8	$\mathbf{0.977 \pm 0.002}$	0 ± 0	0.390 ± 0.477	$\mathbf{0.977 \pm 0.002}$	$\mathbf{0.977 \pm 0.002}$
DoorKey-16x16	0 ± 0	0 ± 0	0 ± 0	0 ± 0	$\mathbf{0.395 \pm 0.484}$
LavaCrossingS9N1	0.954 ± 0.008	0.766 ± 0.383	$\mathbf{0.958 \pm 0.002}$	0.950 ± 0.014	0.951 ± 0.002
LavaCrossingS9N3	0.380 ± 0.466	$\mathbf{0.946 \pm 0.013}$	0.378 ± 0.463	$\mathbf{0.946 \pm 0.011}$	$\mathbf{0.946 \pm 0.015}$
LavaCrossingS11N5	0 ± 0	0.762 ± 0.381	0.193 ± 0.387	0.192 ± 0.384	$\mathbf{0.961 \pm 0.005}$

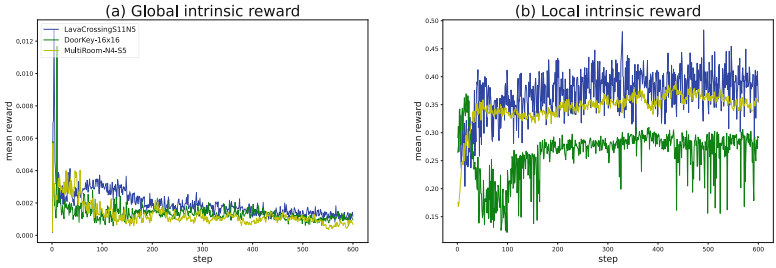

Fig. 4. The intrinsic reward calculated from both global and local perspectives in 3 environments: LavaCrossingS11N5, DoorKey-16x16, and MultiRoom-N4-S5.

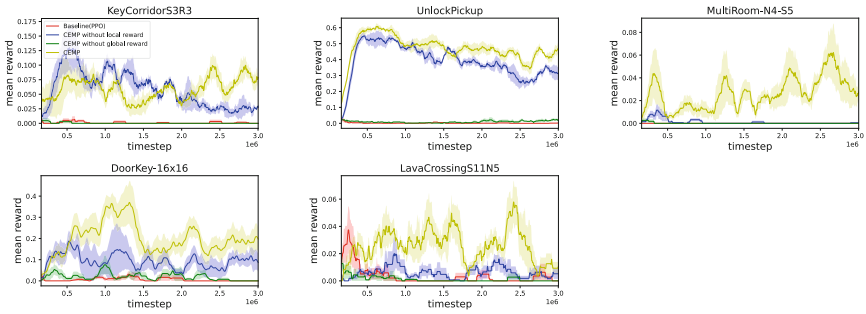

Fig. 5. Without extrinsic reward, only the intrinsic reward is used.

Learning with only Intrinsic Reward. To test whether the intrinsic reward designed from the local and global perspectives in our proposed CEMP method are effective, we only use the intrinsic reward from the local, global, and the fusion of both to guide the agent's learning. Based on the results shown in Fig. 5, we can conclude that our proposed CEMP method can achieve a certain level of performance in the sparse reward environment using only self-generated

intrinsic reward. Moreover, the fusion of the intrinsic reward from the local and global perspectives can achieve better performance than a single intrinsic reward from either the global or local perspective.

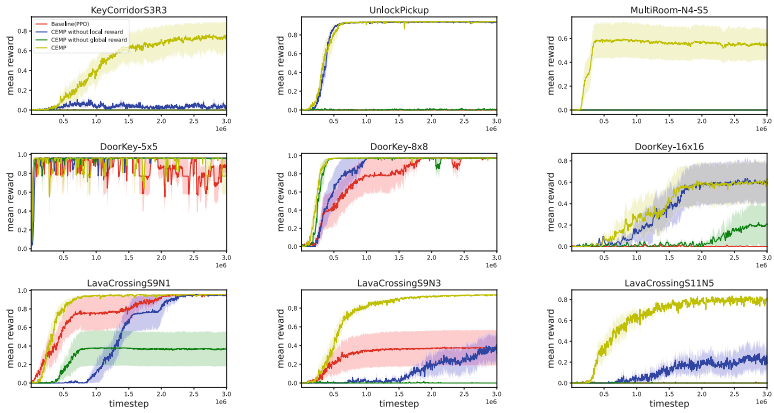

Fig. 6. The ablation experiment on each component of the intrinsic reward.

Ablation Experiment of the Intrinsic Reward. We conducted ablation experiment on each component of the intrinsic reward in our proposed CEMP method. From Fig. 6, we can draw the following conclusions:

1. In all environments, the intrinsic reward that integrates global and local intrinsic reward performs better than using only intrinsic reward calculated from a single perspective.
2. Using only global or local intrinsic reward can also achieve good performance in some environments.

5 Conclusion

This paper proposed a method called CEMP that enables the agent to explore better in the sparse reward environment. The CEMP method contributes a new framework for designing the intrinsic reward from multiple perspectives and can be easily integrated into various existing reinforcement learning algorithms. The experimental results in a series of complex sparse reward environments in MiniGrid demonstrate that our proposed CEMP method can achieve better final performance and faster learning efficiency than algorithms such as ICM [11], RIDE [15], and TRPO-AE-Hash [6] that calculate the intrinsic reward from only a single perspective.

Acknowledgments. This work was supported by the National Defense Science and Technology Foundation Reinforcement Program and the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No.XDA27041001.

References

1. Mnih, V., et al.: Human-level control through deep reinforcement learning. Nature **518**(7540), 529–533 (2015)
2. Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Klimov, O.: Proximal policy optimization algorithms. arXiv preprint arXiv:1707.06347 (2017)
3. Deci, E.L., Ryan, R.M.: Intrinsic Motivation and Self-determination in Human Behavior. Springer, Cham (2013). https://doi.org/10.1007/978-1-4899-2271-7
4. Ostrovski, G., Bellemare, M.G., Oord, A., Munos, R.: Count-based exploration with neural density models. In: International Conference on Machine Learning, pp. 2721–2730. PMLR (2017)
5. Martin, J., Sasikumar, S.N., Everitt, T., Hutter, M.: Count-based exploration in feature space for reinforcement learning. arXiv preprint arXiv:1706.08090 (2017)
6. Tang, H., et al.: # exploration: A study of count-based exploration for deep reinforcement learning. In: Advances in Neural Information Processing Systems, vol. 30 (2017)
7. Charikar, M.S.: Similarity estimation techniques from rounding algorithms. In: Proceedings of the Thiry-Fourth Annual ACM Symposium on Theory of Computing, pp. 380–388 (2002)
8. Choshen, L., Fox, L., Loewenstein, Y.: Dora the explorer: directed outreaching reinforcement action-selection. arXiv preprint arXiv:1804.04012 (2018)
9. Burda, Y., Edwards, H., Storkey, A., Klimov, O.: Exploration by random network distillation. arXiv preprint arXiv:1810.12894 (2018)
10. Stadie, B.C., Levine, S., Abbeel, P.: Incentivizing exploration in reinforcement learning with deep predictive models. arXiv preprint arXiv:1507.00814 (2015)
11. Pathak, D., Agrawal, P., Efros, A.A., Darrell, T.: Curiosity-driven exploration by self-supervised prediction. In: International Conference on Machine Learning, pp. 2778–2787. PMLR (2017)
12. Pathak, D., Gandhi, D., Gupta, A.: Self-supervised exploration via disagreement. In: International Conference on Machine Learning, pp. 5062–5071. PMLR (2019)
13. Stanton, C., Clune, J.: Deep curiosity search: intra-life exploration improves performance on challenging deep reinforcement learning problems. corr abs/1806.00553 (2018) (1806)
14. Savinov, N., et al.: Episodic curiosity through reachability. arXiv preprint arXiv:1810.02274 (2018)
15. Raileanu, R., Rocktäschel, T.: Ride: rewarding impact-driven exploration for procedurally-generated environments. arXiv preprint arXiv:2002.12292 (2020)