LAWRENCE
LIVERMORE
NATIONAL
LABORATORY

LLNL-CONF-867347

The Innate Curiosity in the Multi-Agent Transformer

A. S. Williams, A. O. Maguire, B. C. Soper, D. M. Merl

July 26, 2024

IEEE International Conference on Machine Learning and Applications
Miami, FL, United States
December 18, 2024 through December 20, 2024

Disclaimer

The Innate Curiosity in the Multi-Agent Transformer

Arthur S. Williams
Lawrence Livermore National Laboratory
Livermore, USA
williams323@llnl.gov

Alister Maguire
Lawrence Livermore National Laboratory
Livermore, USA
maguire7@llnl.gov

Braden Soper
Lawrence Livermore National Laboratory
Livermore, USA
soper3@llnl.gov

Daniel Merl
Lawrence Livermore National Laboratory
Livermore, USA
merl1@llnl.gov

***Abstract*—Curiosity is a cognitive mechanism that drives one's intrinsic need to understand the unknown. This intrinsic drive is responsible for guiding the acquisition of knowledge about novel stimuli. Curiosity, akin to thirst and hunger, is considered an evolved motivational mechanism promoting self-beneficial actions. In the context of Multi-Agent Reinforcement Learning (MARL), curiosity encourages exploration by capturing the novelty of an environmental state as an intrinsic reward signal. For cooperative MARL tasks, the Multi-Agent-Transformer (MAT) is one of the state-of-the-art models. However, its performance on sparse reward tasks requiring collaboration is uncertain. This paper explores MAT's performance on the grid-world environment *Multi-Robot Warehouse*. We integrated an Intrinsic Curiosity Module (ICM) for exploration and our results suggests that MAT does not need ICM to learn on sparse environments.**

***Index Terms*—Reinforcement Learning, Transformer, Intrinsic Curiosity Module, Multi-Agent Reinforcement Learning, Exploration**

I. Introduction

Curiosity is a cognitive mechanism responsible for driving the intrinsic need to understand the unknown. The intrinsic desire to understand the unknown is responsible for guiding us toward acquiring knowledge about novel stimuli within our environment. Additionally, curiosity can be described as a perceived gap in one's knowledge and understanding, which results in cognitive deprivation [11], [15]. It is suggested that curiosity is an evolved motivational mechanism that promotes self-beneficial actions, similar to thirst, which motivates drinking water, or hunger, which motivates eating [11], [15]. These concepts can be leveraged in reinforcement learning environments that require agent exploration of novel states. Furthermore, crafting an exploration strategy can be more arduous in the multi-agent setting, especially where cooperation and coordination are required.

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344 and was supported by the LLNL-LDRD Program under Project No. 22-SI-001. LLNL-CONF-867347.

In cooperative Multi-Agent Reinforcement Learning (MARL) tasks, agents learn coordinated strategies for the purpose of reaching a common goal. One of the state-of-the-art models in MARL on cooperative tasks is the Multi-Agent-Transformer (MAT) [27]. Transformers in reinforcement learning (RL) have been gaining traction in the field because of their ability to better handle long temporal horizons compared to recurrent neural networks (RNNs) [18], [24]. Previous RL models adapted the use transformers in an offline setting, where trajectories are stored in a data set. Recently, some transformer-based RL models have transitioned to being online, which includes MAT. The Multi-Agent Advantage Decomposition theorem is leveraged by MAT, allowing for monotonic improvement of action selection. Additionally, MAT can utilize Centralized Training with Centralized Execution without exponential growth due to the sequential action selection process for all agents. However, there is still a question on whether the MAT is capable of performing well on sparse reward tasks that require coordination and exploration.

Curiosity based exploration has been demonstrated in the single agent case and multi-agent case. The Intrinsic Curiosity Module (ICM) [19] captured intrinsic motivation by measuring how novel an environmental state is by training a model to predict the transitioned state (s_{t+1}) given the previous state (s_t) and action (a_t) at time t. The error between the actual next state (s_{t+1}) and predicted next state ($\hat{s}_{t+t}$) results in an intrinsic reward signal (r_t^i). Agents are encouraged to seek out novel states because the model's predictive abilities are low for non-frequented states, thus producing a large predictive error/intrinsic reward signal. Additionally, as uncertainty of environmental states approach zero, agents are able to employ exploitative strategies.

For online MARL, MAT is one of the state-of-the-art models because of its ability to transform the processing of agents' actions as a sequence problem. Also, MAT makes use of

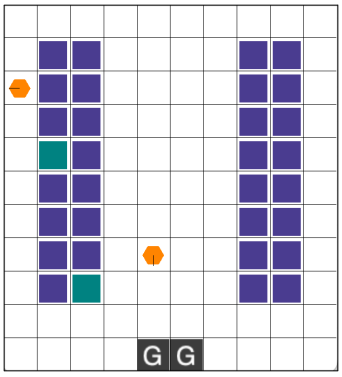

Fig. 1: The Multi-Robot Warehouse environment. The orange hexagons are the robots. The blue squares represent unrequested shelves and green squares represent requested shelves. The black squares are the designated locations to bring the requested shelves.

the multi-agent advantage decomposition theorem [12], [13], which guarantees a monotonic improvement in action selection. However, is the multi-agent advantage decomposition theorem enough to overcome the difficulties of environments that are collaborative and have sparse rewards? While MAT has performed well on collaborative MARL tasks [27], to our knowledge, MAT has yet to be tested on sparse reward environments that require exploration. In this paper, we investigate the performance of MAT on the grid-world environment *Multi-Robot Warehouse* [17]. This environment has many variations that will allow us to explore different sparsity levels within an environment. Our findings show that MAT performs similar to Multi-Agent PPO (MAPPO) [28] with ICM, which suggests that MAT's innate exploration mechanisms are on par with curiosity based methods. Furthermore, MAT does not require an additional network to train, thereby reducing training complexity. The main contributions of this paper are as follows:

1) We investigate whether MAT could benefit from adding ICM for solving sparse reward environments.
2) We investigate the impact that environment size and task difficulty has on learning performance. We compare MAT with and without ICM to MAPPO with and without ICM on the *Multi-Robot Warehouse* environment.
3) We offer a simple solution for extending ICM to multi-agent domains, which requires minimal adjustments to the single-agent case.

II. Background

In this section, we present background material on Markov Decision Processes and Multi-Agent Reinforcement Learning. Additionally, we discuss the role of transformers in reinforcement learning and the use of intrinsic curiosity as a solution for solving sparse reward environments. Markov Decision Processes (MDPs) are a foundation of reinforcement learning, and Multi-Agent Reinforcement Learning extends MDPs to account for multiple agents. Furthermore, the role of transformers in reinforcement learning gives further context to the importance of the research problem this paper aims to address.

A. Markov Decision Processes (MDPs)

Markov Decision Processes (MDPs) are mathematical frameworks used to formalize decision making problems where an agent interacts with an environment that provides feedback in the form of a reward signal. MDPs are defined by the tuple (S, A, P, R, γ), where the state space $S = \{s_1, s_2, \cdots, s_n\}$, the action space A represents the actions across all states, the state transition probabilities $P : S \times A \times S \rightarrow [0, 1]$ that defines the probability of transitioning from $s_t \in S$ to $s_{t+1} \in S$ given action $a_t \in A$. A scalar reward from the environment $R : S \times A \times S \rightarrow \mathbb{R}$ is provided to the agent on each timestep t. The discount factor $\gamma \in [0, 1]$ is a scalar that weights current rewards higher than those received in the future. In an MDP, the agent selects actions based on the current state determined by a policy $\pi(s)$, and the environment responds by transitioning to a new state and providing a reward signal. The agent's goal is to learn an optimal policy $\pi^*(s)$, which is a mapping from states to actions, that maximizes the expected cumulative reward. Solving MDPs forms the basis for various reinforcement learning algorithms and applications.

B. Multi-Agent Reinforcement Learning (MARL)

In Multi-Agent Reinforcement Learning (MARL), multiple agents interact with an environment and learn to coordinate their actions to achieve individual or collective objectives. MARL scenarios introduce complexities due to the interdependencies between agents' actions and the non-stationary environment. Coordinating multiple agents in dynamic environments requires sophisticated algorithms that can handle the challenges posed by simultaneous learning and decision making among diverse agents. MARL is an extension of the MDP framework and can be defined by the tuple $(N, S, \{A^i\}_{i\in\{1,\cdots,N\}}, P, \{R^i\}_{i\in\{1,\cdots,N\}}, \gamma)$ where N is the number of agents, S is the shared environment state for all agents, A^i is agent i's set of actions with the joint action space $A = A_1 \times A_2 \times \cdots \times A_N$. The transition probabilities $P : S \times A \rightarrow S$ with $P(s'|s, \mathbf{a})$ is the probability of transitioning from state s' to s given joint action $\mathbf{a}$ $\forall A_{i\in[1,N]}$. Each agent i receives a reward $R_i : S \times A \rightarrow \mathbb{R}$.

C. Transformers in Reinforcement Learning

Transformers, initially proposed in natural language processing tasks [24], have gained prominence in reinforcement learning [9]. These models utilize self-attention mechanisms to capture long-range dependencies in sequential data, making them suitable for RL tasks that involve processing complex, non-local information. Integrating Transformers in RL architectures has shown significant improvements in handling large state spaces, enhancing the capability to capture intricate patterns and dependencies in diverse environments [16]. The Decision Transformer [6] framed RL as a sequence learning problem and leveraged a causally masked transformer to

learn trajectories that achieve a specified return. It is trained on offline data that consists of trajectories collected from different environments. Following the Decision Transformer, many works explored the effectiveness of transformers for offline RL [5], [10], [23], [25], [26], [30]. Additionally, agents with the ability to solve multiple RL tasks utilizing offline data emerged [8], [14], [20]. What happens when the task or environment does not have a dataset of trajectories? Online RL allows the agent to learn from directly interacting with the environment, as opposed to collected data as in the offline case. Esslinger et al. [7] showed that you can couple a transformer with Deep RL for online learning. Furthermore, Zheng et al. [29] introduced the Online Decision Transformer (ODT) as a method of blending offline pretraining with online finetuning. However, unlike ODT, MAT does not require finetuning and is fully online.

D. Intrinsic Curiosity in Reinforcement Learning

Intrinsic Curiosity is a concept in RL where agents are driven to explore the environment. Instead of relying solely on external rewards, agents are motivated by curiosity-driven learning objectives. This approach encourages agents to explore unfamiliar or challenging situations, leading to more efficient learning and adaptation to novel environments [2], [3]. Intrinsic curiosity mechanisms have been pivotal in enhancing exploration strategies, enabling agents to discover informative states and actions, thereby accelerating the learning process. The Intrinsic Curiosity Module (ICM) [19] predicts the next state transition of the agent to produce a curiosity bonus that is added to the reward. Additionally, Burda et al. introduced random network distillation as a way to mitigate the noisy TV problem [4], where an agent gets attracted to the entropy of the environment similar to the white noise of a TV.

ICM is comprised of a forward model and an inverse model. The forward model takes as input action a_t and encoded state $\phi(s_t)$ and predicts the next encoded state $\hat{\phi}(s_{t+1})$. Furthermore, the forward model trains a neural network to learn a function f as follows:

$$\hat{\phi}(s_{t+1}) = f(\phi(s_t), a_t)) \tag{1}$$

Additionally, the intrinsic reward r_t^i is produced by taking the L2 squared distance between the predicted encoded state $\hat{\phi}(s_{t+1})$ and the target encoded state $\phi(s_{t+1})$,

$$r_t^i = \frac{\eta}{2}\|\hat{\phi}(s_{t+1}) - \phi(s_{t+1})\|_2^2 \tag{2}$$

where η is a scaling factor. The inverse model takes as input $\phi(s_t)$ and $\phi(s_{t+1})$ and outputs action $\hat{a}_t$, which is the predicted action taken during the state transition and is defined as:

$$\hat{a_t} = g(\phi(s_t), \phi(s_{t+1})) \tag{3}$$

where the function g is learned by a neural network. The main purpose of the inverse model is to aid in leaning feature representations that contain only information relevant to the action that was performed. In addition to intrinsic rewards, the agent receives rewards r_t^e from the environment. The agent's goal is to maximize the sum of both rewards $r_t = r_t^i + r_t^e$.

III. Method

The Intrinsic Curiosity Module for MARL requires modifying the forward model and inverse dynamics model such that the observations and actions of each agent are utilized for feature prediction and action prediction. We will show how the model networks are defined to accommodate these adaptations. Furthermore, this section details the environments used for the experiments, along with the implementation and training details.

A. Intrinsic Curiosity for Multiple Agents

ICM was originally designed for the single agent case. We extended ICM to work for the multi-agent case by allowing each agent to utilize a shared network for both the forward and inverse model. Each agent computes their own intrinsic curiosity, based on the agent's individual observation. The forward model computes the curiosity as follows:

$$\hat{\phi}(s_{t+1}^i) = f(\phi(s_t^i), a_t^i)), \forall i \in [1, N] \tag{4}$$

where $\hat{\phi}(s_{t+1}^i)$ is the predicted embedded next state s_{t+1} of agent i, f is the forward model, and N is the number of agents. The intrinsic reward signal is then computed as:

$$I_t^i = \frac{\eta}{2}\|\hat{\phi}_{t+1}^i - \phi_{t+1}^i\|_2^2, \forall i \in [1, N] \tag{5}$$

where I_t^i is the intrinsic reward at time t for agent i, and η is a scaling factor. More details on ICM is provided in Algorithm 1.

Algorithm 1 Multi-Agent ICM

Require: Environment E, scaling factor η, max steps T, number of agents N
Randomly Initialize forward model f_θ, policy model p_φ and feature encoder h_ω
for $t = 0$ to T **do**
  for agent $i = 1$ to N **do**
    $\pi_\varphi^i \leftarrow p_\varphi(s_t)$
    Observe $o_t^i = \{s_t^i, a_t^i, R_t^i, s_{t+1}^i\} \sim \pi_\varphi^i$ from E
    $\phi_t^i \leftarrow h_\omega(s_t^i)$
    $\phi_{t+1}^i \leftarrow h_\omega(s_{t+1}^i)$
    $\hat{\phi}_{t+1}^i \leftarrow f_\theta(\phi_t^i, a_t^i)$
    $I_t^i \leftarrow \frac{\eta}{2}\|\hat{\phi}_{t+1}^i - \phi_{t+1}^i\|_2^2$ {Calculate Intrinsic Reward}
    $reward_t^i = R_t^i + I_t^i$
  end for
end for

B. Environments

The Multi-Robot Warehouse environment simulates robots moving requested goods to a designated location within a warehouse [17]. This environment simulates a real-world application that involves robots picking up shelves and delivering them to a workstation where humans unload the contents. The robot then returns the self back to an empty location. The action space is discrete and consists of 4 actions: Turn Left, Turn Right, Forward, Load/Unload Shelf. The observation of

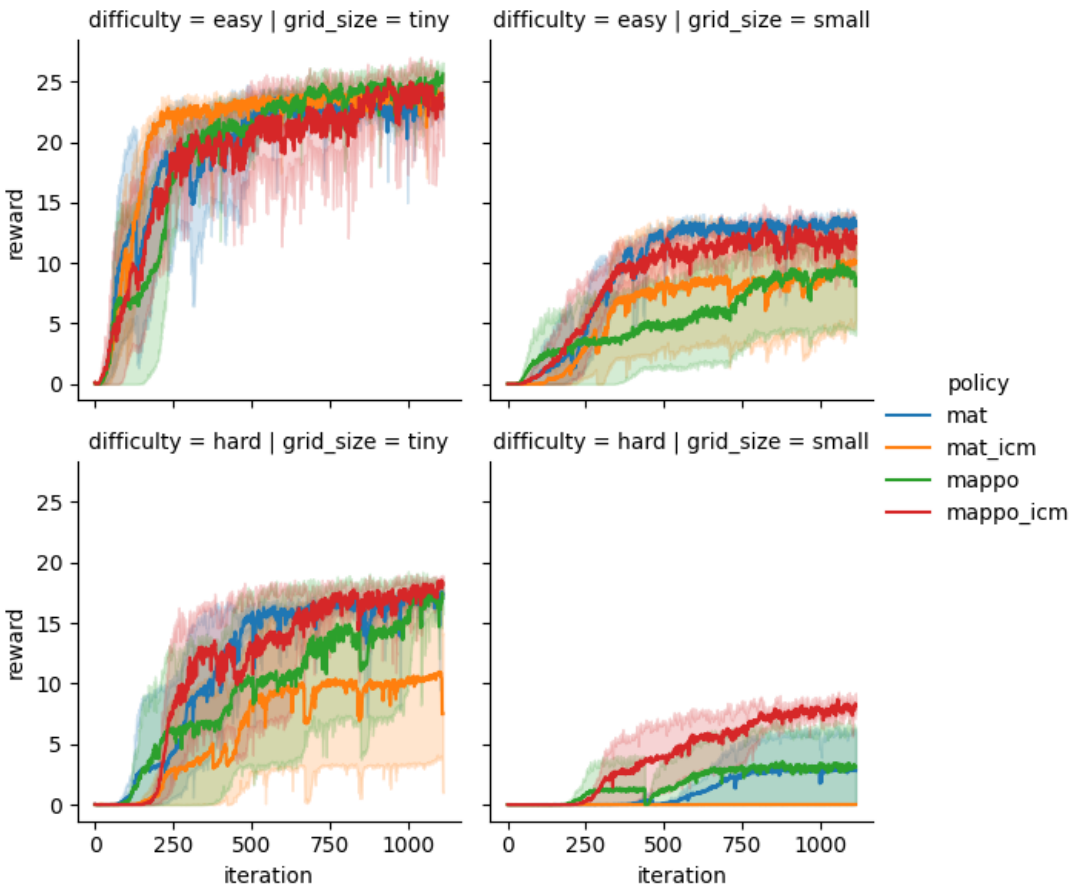

Fig. 2: Multi-Robot Warehouse with 2 Agents. MAT performs on par with MAPPO with ICM on all except the configuration where the difficult is hard and the grid size is small (10x11).

each agent is a 3x3 partially observable window centered on the agent. If an entity is within the window the following can be observed: the location, the rotation and whether the agent is carrying a shelf; the location and rotation of other robots; and shelves and whether they are currently in the request queue. The difficulty level of the environment is determined by the number of requested shelves R relative to the number of agents N. By default $R = N$ but there is an easy and hard variation where $R = 2N$ and $R = N/2$, respectively. Sparse variations of Multi-Robot Warehouse consist of small R on larger grid sizes. In Figure 1, we see an example of the 10x11 environment with 2 agents.

C. Implementation and Training Details

This section presents the network architecture and training details for the models MAT, MAPPO, and ICM. The architecture for MAT consists of 2 transformer blocks, 1 transformer head, and an embedding dimension of 128 for both the actor and critic networks. The MAPPO network architecture is multi-layer perceptron (MLP) configured with 2 hidden layers of dimension 256, each with a ReLU activation function, for both the actor and critic. The MAPPO implementation utilizes parameter sharing, which shares both the policy and value function parameters across all agents. For MAT and MAPPO we use generalized advantage estimation (GAE) [21], [28] to approximate the advantage function. Additionally, ICM uses a 2 layer MLP with a ReLU activation function and a hidden dimension of 64. All models were trained in a distributed manner across a compute node with 36 ranks/processors. The common hyper-parameters are given in Table I. The entropy coefficient, ppo clip, gradient clip, discount factor gamma, ppo epochs, and optimizer hyper-parameter values were determined based on recommended values [1], [22]. Furthermore, we obtained the actor lr, critic lr, ICM scaling factor and ICM lr hyper-parameter values by performing a grid-search and selecting the values that achieved the highest cumulative reward.

IV. Results

This section presents the results from the experiments that are designed to test MAT's ability to learn in collaborative sparse reward environments. We evaluated the performance of MAT with and without ICM on multiple variations of the *Multi-Robot Warehouse* environment. Additionally, MAPPO with and without ICM was evaluated and used as a baseline. The *Multi-Robot Warehouse* environment can become more

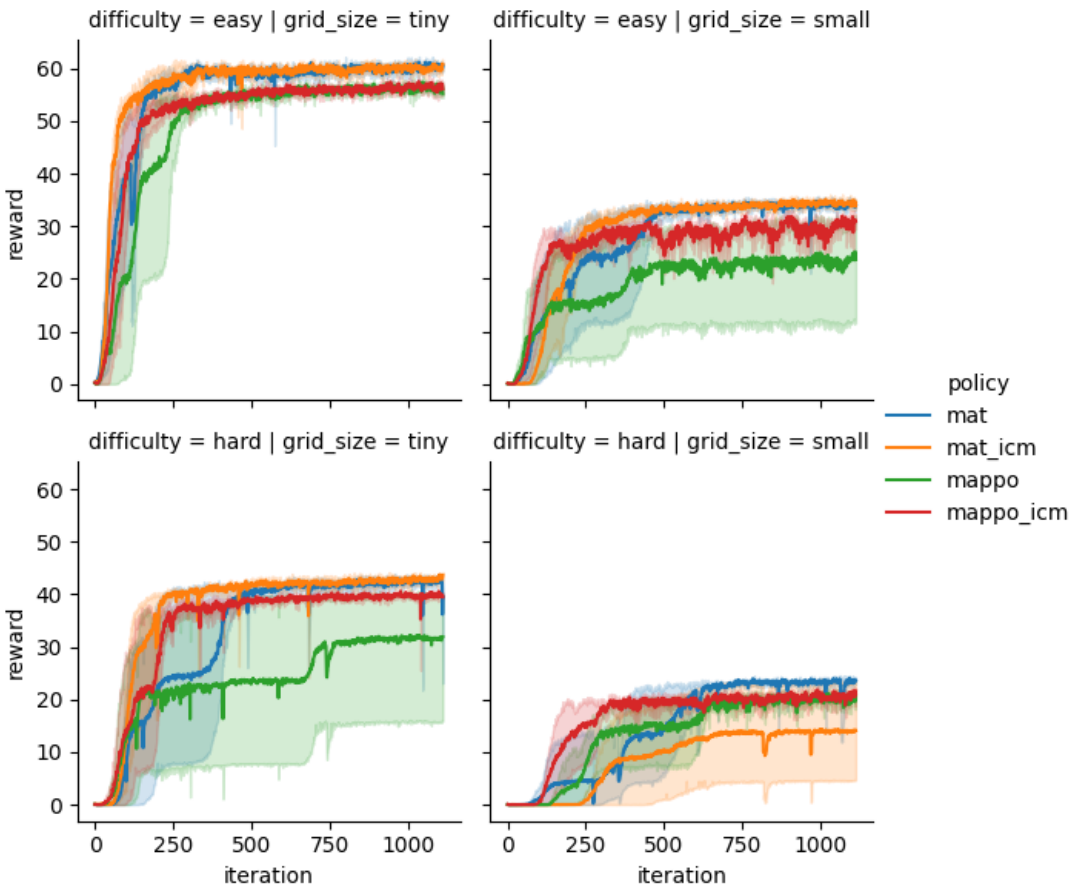

Fig. 3: Multi-Robot Warehouse with 4 Agents. MAT performs on par with MAPPO with ICM on all of the configurations. Additionally, an increase in agents allowed MAT to take advantage of its superior cooperative capabilities among agents. This is evident by the change in reward curve where the difficult is hard and the grid size is small (10x11) from Figure 2 to the current figure.

TABLE I: The hyper-parameters for MAT and MAPPO for Multi-Robot Warehouse.

common hyper-parameter	value
actor lr	2e-3
critic lr	2e-3
ppo epochs	10
ppo clip	0.2
entropy coef	0.01
batch size	500
gradient clip	0.5
gamma	0.99
optimizer	Adam
ICM scaling factor	1e-3
ICM lr	1e-4

sparse as the grid size increases and the number of shelve requests are small. The metric used for evaluation is the cumulative rewards obtained by all agents in the environment.

The environment hyper-parameters that we varied for the *Multi-Robot Warehouse* include: the difficulty, grid size, and number of agents. We conducted 5 training runs with different seeds for each environment configuration per policy model and computed the 95% confidence intervals. Tiny refers to a grid size of 10x11 and small refers to a grid size of 10x20. When the difficulty is easy and the grid size is tiny, the differences between the policies are negligible for both the 2 agent and 4 agent case(see Figures 2 and 3). On the 2 agent variation, MAT with ICM under-performed compared to the other policies. However, on the 4 agent variation, MAT with ICM performed well on all but when the difficulty is hard and the grid size is small. MAT without ICM performed as good or better than other policies on all variations except for when the difficulty is hard and grid size is small.

V. Discussion

In this section, we present an analysis of our results. We discuss the impact of reward sparsity and how this impacts training results. We also argue that ICM is not needed for MAT due to MAT's innate exploration.

A. The Impact of Reward Sparsity

As the size of the grid-world increases and the number of shelve requests are small, the rewards in the environment

become more sparse. We observe that ICM helps MAPPO in all circumstances for both the 4 agent and 2 agent case. When the difficulty is easy and the grid size is tiny, we can only see a negligible improvement for MAPPO. This makes sense, because ICM helps with exploration in sparse reward settings. However, ICM doesn't always increase performance with MAT. This issue is more pronounce with 2 agents vs 4 agents. This could be attributed to MAT's ability to scale more robustly as the number of agents increase.

B. MAT Does Not Need Curiosity

MAT performs as well as MAPPO with ICM and outperforms base MAPPO on all the tasks that contain sparsity. This shows that MAPPO requires ICM to achieve the same level of performance that is innately attributed to MAT. Additionally, MAT leverages the Multi-Agent Advantage Decomposition Theorem to achieve monotonic improvement in action selection among agents. We hypothesize that the utilization of this theorem, coupled with processing agents as a sequence, accounts for the performance gap between MAT and MAPPO. It is our conclusion that MAT does not require the addition of ICM to improve exploration.

C. ICM with MAT

Sparse environmental rewards caused MAT to not learn as efficiently compared to MAPPO with ICM. This suggests that, even though MAT performs well on these tasks that require some exploration, there is still room for improvement. In addition, since ICM adds a performance boost for MAPPO, we may need to make additional augmentations to ICM to account for how the transformer processes the agents.

VI. Conclusion

Curiosity allows us to seek out novel stimuli without the need for external motivation. In reinforcement learning, we can capture this concept by utilizing ICM. The ICM mechanism is needed on environments that give sparse feedback to the agent. In a multi-agent system, this issue can be compounded because of the addition of collaboration. MAT is one the state-of-the-art models for solving multi-agent collaborative environments. It was our goal to test whether MAT could benefit from adding ICM for solving sparse environments. Our results show that MAT without ICM outperformed MAPPO and performed similar to MAPPO with ICM. This means MAT does not need the addition of ICM to learn in these sparse environments. Furthermore, MAT with ICM performed worst than base MAT on several environment variations. This suggests ICM is an ineffective additive for MAT in regards to exploration. However, there is room for improvement for MAT for learning sparse environments more efficiently. Therefore, future work will address how to improve learning efficiency for MAT on sparse environments.

References

[1] M. Andrychowicz, A. Raichuk, P. Stańczyk, M. Orsini, S. Girgin, R. Marinier, L. Hussenot, M. Geist, O. Pietquin, M. Michalski, S. Gelly, and O. Bachem, "What Matters In On-Policy Reinforcement Learning? A Large-Scale Empirical Study," Jun. 2020. [Online]. Available: https://arxiv.org/abs/2006.05990v1

[2] N. Botteghi, B. Sirmacek, M. Poel, C. Brune, and R. Schulte, "CURIOSITY-DRIVEN REINFORCEMENT LEARNING AGENT for MAPPING UNKNOWN INDOOR ENVIRONMENTS," in *ISPRS Annals of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, vol. 5. Copernicus, Jun. 2021, pp. 129–136, iSSN: 2194-9042 Issue: 1. [Online]. Available: https://research.utwente.nl/en/publications/curiosity-driven-reinforcement-learning-agent-for-mapping-unknown

[3] Y. Burda, H. Edwards, D. Pathak, A. Storkey, T. Darrell, and A. A. Efros, "Large-Scale Study of Curiosity-Driven Learning."

[4] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "EXPLORATION BY RANDOM NETWORK DISTILLATION."

[5] Y. Chebotar, Q. Vuong, K. Hausman, F. Xia, Y. Lu, A. Irpan, A. Kumar, T. Yu, A. Herzog, K. Pertsch, K. Gopalakrishnan, J. Ibarz, O. Nachum, S. A. Sontakke, G. Salazar, H. T. Tran, J. Peralta, C. Tan, D. Manjunath, J. Singh, B. Zitkovich, T. Jackson, K. Rao, C. Finn, and S. Levine, "Q-Transformer: Scalable Offline Reinforcement Learning via Autoregressive Q-Functions," in *Proceedings of The 7th Conference on Robot Learning*. PMLR, Dec. 2023, pp. 3909–3928, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v229/chebotar23a.html

[6] L. Chen, K. Lu, A. Rajeswaran, K. Lee, A. Grover, M. Laskin, P. Abbeel, A. Srinivas, and I. Mordatch, "Decision Transformer: Reinforcement Learning via Sequence Modeling," in *Advances in Neural Information Processing Systems*, vol. 34. Curran Associates, Inc., 2021, pp. 15 084–15 097.

[7] K. Esslinger, R. Platt, and C. Amato, "Deep Transformer Q-Networks for Partially Observable Reinforcement Learning," Nov. 2022, arXiv:2206.01078 [cs]. [Online]. Available: http://arxiv.org/abs/2206.01078

[8] S. Hu, Z. Fan, L. Shen, Y. Zhang, Y. Wang, and D. Tao, "HarmoDT: Harmony Multi-Task Decision Transformer for Offline Reinforcement Learning," May 2024, arXiv:2405.18080 [cs]. [Online]. Available: http://arxiv.org/abs/2405.18080

[9] S. Hu, L. Shen, Y. Zhang, Y. Chen, and D. Tao, "On Transforming Reinforcement Learning With Transformers: The Development Trajectory," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–20, 2024, conference Name: IEEE Transactions on Pattern Analysis and Machine Intelligence. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/10546317

[10] M. Janner, Q. Li, and S. Levine, "Offline Reinforcement Learning as One Big Sequence Modeling Problem," Nov. 2021, arXiv:2106.02039 [cs]. [Online]. Available: http://arxiv.org/abs/2106.02039

[11] C. Kidd and B. Y. Hayden, "The psychology and neuroscience of curiosity," *Neuron*, vol. 88, no. 3, pp. 449–460, Nov. 2015. [Online]. Available: https://www.ncbi.nlm.nih.gov/pmc/articles/PMC4635443/

[12] J. G. Kuba, R. Chen, M. Wen, Y. Wen, F. Sun, J. Wang, and Y. Yang, "Trust Region Policy Optimisation in Multi-Agent Reinforcement Learning," Apr. 2022, arXiv:2109.11251 [cs]. [Online]. Available: http://arxiv.org/abs/2109.11251

[13] J. G. Kuba, M. Wen, Y. Yang, L. Meng, S. Gu, H. Zhang, D. H. Mguni, and J. Wang, "Settling the Variance of Multi-Agent Policy Gradients," Apr. 2022, arXiv:2108.08612 [cs]. [Online]. Available: http://arxiv.org/abs/2108.08612

[14] K.-H. Lee, O. Nachum, M. S. Yang, L. Lee, D. Freeman, S. Guadarrama, I. Fischer, W. Xu, E. Jang, H. Michalewski, and I. Mordatch, "Multi-Game Decision Transformers," *Advances in Neural Information Processing Systems*, vol. 35, pp. 27 921–27 936, Dec. 2022.

[15] G. Loewenstein, "The psychology of curiosity: A review and reinterpretation," *Psychological Bulletin*, vol. 116, no. 1, pp. 75–98, 1994, place: US Publisher: American Psychological Association.

[16] T. Ni, M. Ma, B. Eysenbach, and P.-L. Bacon, "When Do Transformers Shine in RL? Decoupling Memory from Credit Assignment," *Advances in Neural Information Processing Systems*, vol. 36, pp. 50 429–50 452, Dec. 2023.

[17] G. Papoudakis, F. Christianos, L. Schäfer, and S. V. Albrecht, "Benchmarking multi-agent deep reinforcement learning algorithms in cooperative tasks," in *Proceedings of the Neural Information Processing*

Systems Track on Datasets and Benchmarks (NeurIPS), 2021. [Online]. Available: http://arxiv.org/abs/2006.07869

[18] E. Parisotto, F. Song, J. Rae, R. Pascanu, C. Gulcehre, S. Jayakumar, M. Jaderberg, R. L. Kaufman, A. Clark, S. Noury, M. Botvinick, N. Heess, and R. Hadsell, "Stabilizing Transformers for Reinforcement Learning," in *Proceedings of the 37th International Conference on Machine Learning*. PMLR, Nov. 2020, pp. 7487–7498, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v119/parisotto20a.html

[19] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven Exploration by Self-supervised Prediction," in *Proceedings of the 34th International Conference on Machine Learning*. PMLR, Jul. 2017, pp. 2778–2787, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v70/pathak17a.html

[20] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg, T. Eccles, J. Bruce, A. Razavi, A. Edwards, N. Heess, Y. Chen, R. Hadsell, O. Vinyals, M. Bordbar, and N. de Freitas, "A Generalist Agent," Nov. 2022, arXiv:2205.06175 [cs]. [Online]. Available: http://arxiv.org/abs/2205.06175

[21] J. Schulman, P. Moritz, S. Levine, M. Jordan, and P. Abbeel, "High-Dimensional Continuous Control Using Generalized Advantage Estimation," Oct. 2018, arXiv:1506.02438 [cs]. [Online]. Available: http://arxiv.org/abs/1506.02438

[22] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms," Aug. 2017, arXiv:1707.06347 [cs]. [Online]. Available: http://arxiv.org/abs/1707.06347

[23] J. Shang, K. Kahatapitiya, X. Li, and M. S. Ryoo, "StARformer: Transformer with State-Action-Reward Representations for Visual Reinforcement Learning," in *Computer Vision – ECCV 2022*, S. Avidan, G. Brostow, M. Cissé, G. M. Farinella, and T. Hassner, Eds. Cham: Springer Nature Switzerland, 2022, pp. 462–479.

[24] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is All you Need," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017.

[25] K. Wang, H. Zhao, X. Luo, K. Ren, W. Zhang, and D. Li, "Bootstrapped Transformer for Offline Reinforcement Learning," *Advances in Neural Information Processing Systems*, vol. 35, pp. 34 748–34 761, Dec. 2022.

[26] Y. Wang, C. Yang, Y. Wen, Y. Liu, and Y. Qiao, "Critic-Guided Decision Transformer for Offline Reinforcement Learning," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 14, pp. 15 706–15 714, Mar. 2024, number: 14. [Online]. Available: https://ojs.aaai.org/index.php/AAAI/article/view/29499

[27] M. Wen, J. G. Kuba, R. Lin, W. Zhang, Y. Wen, J. Wang, and Y. Yang, "Multi-Agent Reinforcement Learning is a Sequence Modeling Problem," Oct. 2022, arXiv:2205.14953 [cs]. [Online]. Available: http://arxiv.org/abs/2205.14953

[28] C. Yu, A. Velu, E. Vinitsky, J. Gao, Y. Wang, A. Bayen, and Y. Wu, "The Surprising Effectiveness of PPO in Cooperative, Multi-Agent Games," Nov. 2022, arXiv:2103.01955 [cs]. [Online]. Available: http://arxiv.org/abs/2103.01955

[29] Q. Zheng, A. Zhang, and A. Grover, "Online Decision Transformer," in *Proceedings of the 39th International Conference on Machine Learning*. PMLR, Jun. 2022, pp. 27 042–27 059, iSSN: 2640-3498. [Online]. Available: https://proceedings.mlr.press/v162/zheng22c.html

[30] Z. Zhuang, D. Peng, J. Liu, Z. Zhang, and D. Wang, "Reinformer: Max-Return Sequence Modeling for Offline RL," Jun. 2024, arXiv:2405.08740 [cs]. [Online]. Available: http://arxiv.org/abs/2405.08740