We thank the reviewers for all of these valuable comments. We provide point by point responses below.

To Reviewer #1

Q1: About the significance and contribution. A: The technical part of `LIIR` is indeed inspired by what was proposed in [16], while we think the considered problem and the definitions of the individual intrinsic reward and proxy value are new for MARL research and the reported results could contribute to the MARL domain. Moreover, we think formulating the intrinsic reward learning problem into bi-level optimization is new from the perspective of meta-gradient learning, which is the key to make the extension to the multi-agent case natural.

Q2: "Another paper...'Optimal rewards for cooperative agents'..." A: We have carefully read the paper and we think the method differs from ours that it does not consider the centralized learning and decentralized execution architecture and the learning of its intrinsic reward is integrated with the update of the policy while we cast the intrinsic reward learning as a meta-gradient learning problem. We will provide more discussions in the revision.

Q3: "...why the authors did not choose all the tasks used in the COMA paper..." A: We think 8M, 2S3Z and 3S5Z are more challenging tasks compared to 5M, 5W and 2D3Z. We also noticed that 2S3Z and 3S5Z were studied in QMIX [33] which had been demonstrated to be superior than COMA. Therefore, we chose a mixture of the scenarios of these used in COMA and those used in QMIX. Actually, all these settings are based on the SMAC framework.

Q4: "...deeper analyses of the learned intrinsic reward..." A: Thanks for the comments. According to your suggestion, we have collected all the $r_{in}(s, a)$'s for the action $a =$'attack' when the corresponding HP's are lower than a percent of 50% from 100 test episodes, and we compute their cosine similarity coefficient (a value in [-1, 1]). The averaged cosine similarity is 0.55 for 2S3Z and 0.67 for 3M, showing that when the HP is low, the intrinsic reward generally shows a low value for taking 'attack' action as well. We will include these discussions in the revision.

Q5: "...more analysis/explanation...more convincing results, maybe in other domains." A: Thanks for the comments. We think 3S5Z is the most complicated task among the four settings, and in 3S5Z agents might act more diversely and hence `LIIR` could perform much better. We will perform more explanations for these experimental results. For other domains, per the suggestion, we have designed a new game named *1D Pursuit* to provide a fast evaluation of the generality of `LIIR`. In *1D pursuit*, two agents are assigned with two initial integers x and y, and the agents could take actions from $\{+1, -1, 0\}$ to increase, decrease or keep their values to approach a target value z. The team reward is set to be inversely proportional to $|z - x| + |z - y|$. We find that `LIIR` could easily assign a reasonable intrinsic reward for each agent. Specifically, we denote actions approaching (moving away from) the target as "good" ("bad") actions, and we plot the histogram of the intrinsic reward distributions from 100 episodes in Fig. 1 (d). The figure shows that `LIIR` can learn reasonable intrinsic reward for the agents. So we think `LIIR` is a general approach.

Q6: "Is the idea well suited for the competitive scenarios?" A: Applying `LIIR` to competitive MARL scenarios is very interesting and it should not be a complicated extension. For example, under competitive settings, there should also exists a global score measuring the game status of all the agents, and one can design an intrinsic reward function for each of these competitive agents to differentiate their gains (which might not be symmetric). We are interested to investigate such scenarios in the future work.

To Reviewer #2

Q1: "The readability and reproducibility can certainly be improved..." **A:** Thanks for your comments. We followed the parameter settings in COMA and QMIX, so we omitted some details on describing the experiment. We will enrich these information in the revision. Specifically, we fix the learning rates α and β to be 5×10^{-4} in all experiments. Following [16], λ is set to be 0.01. We set the batch size as 32. An overview of the network architecture is shown in Fig. 1. Codes will be released for reproducing all the results.

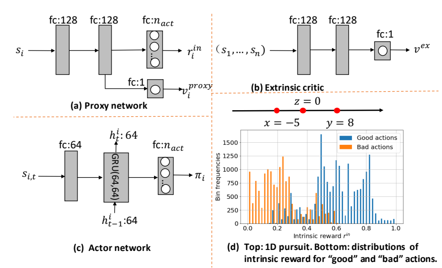

Figure 1: Network architecture and new results on the 1D pursuit game.

To Reviewer #3

Q1: "...the learned IR curves do overlap..." A: Most of the agents might have similar observations so their intrinsic rewards are similar, while we have performed more analyses of the learned intrinsic reward. Please refer to the response to Q4 of Reviewer #1.

Q2: "...straightforward application...any differences in implementation..." A: In the considered MARL problem, we have to define each agent an individual intrinsic reward. An important difference compared to [16] is that for MARL problem the original objective is maximizing over the expected extrinsic team return, and a direct connection between the extrinsic team return and the individual intrinsic reward functions is not straightforward. In `LIIR`, we build proxy value functions for the agents and connect them with the team return via the bi-level optimization problem.

Q3:"...what is meant by 'share the same policy'...How is λ tuned..." A: "share the same policy" indicates the agents share the same policy parameters. Each agent may have different partial observations, so the output actions are also different. In decentralized settings, IAC differs from Central-V that their value functions are distinctly defined and Central-V uses a centralized critic while IAC uses independent critics. Following [16], the parameter λ is set as 0.01. We indeed tried different choices of λ while we found that the results did not differ much.

Q4: "...could be strengthened by considering more domains..." A: Thanks for the suggestion. We have studied another task for evaluating the generality of `LIIR`. Please refer to the response to Q5 of Reviewer #1.