We would like to thank the reviewers for their thorough evaluations and for bringing to our attention some missing citations and typos. We answer specific questions raised by the reviewers, below.

Performance w.r.t. number of agents (R1). We discuss how our methods scale with the number of agents in Appendix H. Specifically, Figure 16 shows that our method's benefits hold but that the underlying algorithm (MADDPG in this case) fails to handle many agents; this has also been shown in [12].

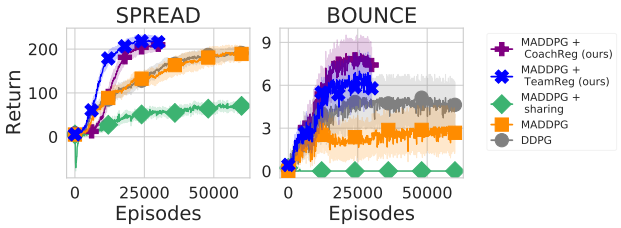
Related work and novelty (R2, R3). (To R2 and R3) We are grateful for bringing to our attention some relevant work in hierarchical RL. Importantly, however, the novelty of CoachReg does not lie in training sub-policies (which are obtained here through a simple and novel masking procedure) but rather in co-evolving synchronized sub-policies across multiple agents. This is indeed closer to the joint exploration work of Mahajan et al. (2019)[1] (as pointed out by R2). Yet, a major difference is that MAVEN's situational-prediction occurs only on the first timestep and requires synchronized random seeds across the agents *at test time*, whereas with CoachReg agents explicitly learn a set of subpolicies, of which they choose one to execute at every timestep. Each agent chooses without using a common sampling procedure and execution is therefore fully decentralized. We will update our manuscript with these more explicit clarifications. ***(To R2)*** Thank you for referring us to Hong et al.[2] whose method is very close to "MADDPG + agent-modelling", the TeamReg ablation that we compare against in Section 7, Figure 5. As we discuss in Related Work (L211-214), agent-modelling (through cross-entropy prediction) is now a widely used MARL component and Hong et al., like [11], uses it as an auxiliary task to learn richer representations. Similarly, TeamReg relies on agent-modelling (L141-146) but our contribution with TeamReg is to instead use it to explicitly influence other agents' behavior toward being predictable rather than just learning a representation (L145-148 and L219-222). To our knowledge, this is a novel contribution and has not been considered in prior work.

Positioning of the paper and missing keywords (R2). While the high level positioning of this work in the Centralized Training Decentralized Execution framework (CTDE) is already made clear throughout the paper (L20, 38, 42, 237, 266, 323), we will highlight it in the abstract as suggested. However, we do not believe that "the planning setting" (usually referring to making use of a transition model rather than the agents model) or "self-play" (where an agent, short of having an opponent to train with, plays against itself) are relevant keywords for our work.

Importance to the broader community, reflection, motivation and transfer (R2). As highlighted by other reviewers, our work makes significant contributions to the research community: at a high level we question the widespread assumption that centralized training always outperforms decentralized training, proposing a definition for coordinated behavior (based on behavior predictability), in order to improve upon it. We propose two (2) novel practical coordination promoting methods that are applicable to any CTDE algorithm and evaluate them on three (3) different baselines based on the prevalent MADDPG algorithm, as well as two (2) ablated versions of our methods.

MADDPG baseline (R3). We disagree with the premise that MADDPG is a weak baseline and argue that the evaluation setting plays a major role in allowing valid and insightful experimental results. Several recent works have pointed out that hyperparameter tuning often plays a fundamental role in determining which algorithms best perform at a given task (Henderson et al. (2018)[3], Colas et al. (2019)[4]). In our work, we make a substantial effort to offer fair and significant comparisons by allowing our three (3) baselines (DDPG, MADDPG, MADDPG + sharing) and two (2) ablations (MADDPG + agent modelling, MADDPG + policy masks) a full hyperparameter tuning, yielding a competitive suite of baselines. To substantiate the importance of such re-tuning, we provide here additional experiments reporting the improvements of our tuned MADDPG over MADDPG with the original hyperparameters configuration from [22]. The improvements are 900 % (SPREAD), 1300 % (BOUNCE), 700 % (COMPROMISE) and 400 % (CHASE) and highlight the important performance gains allowed by our evaluation procedure to the baselines.

Conclusiveness of the results (R4). As requested, we extended the training of the baselines and the inlined figure shows that our methods still outperform them. Additionally, we believe that our evaluation is sound, conclusive and substantiates our claims (as concluded by R1). A key question here is "do the proposed coordination-inducing methods improve performance of the CTDE framework?". We answer this by examining the impact of our proposed ideas on the widely used MADDPG CTDE algorithm and we perform an ablation study to probe each element of our contribution in more detail. We apply a careful experimental methodology (Tables 2 through 11) to both continuous and discrete action environments of varying complexity, requiring *significant* computing resources: e.g., the retraining in Table 1 of our submission alone require 120 CPU-days and prevented us from extending the number of training steps illustrated in this rebuttal.



[1] Mahajan et al., MAVEN : Multi-Agent Variational Exploration (2019) [2] Hong et al. A Deep Policy Inference... (2018)
[3] Henderson et. al., Deep RL that matters. (2018) [4] Colas et al., A Hitchhiker's Guide to Statistical Comparisons of RL ... (2019)