We thank the reviewers for their thoughtful feedback and suggestions. We are encouraged to read that they found our work clearly written (R1, R2, R3, R4) and they appreciated the simplicity (R2, R4), generality (R1) as well as convincing performance (R1, R2, R4) of our method.

Reviewer 1

3.1 The fully-cooperative setting assumes that agents share the same rewards in each time step. In contrast, we allow agents to use individual reward functions (line 79), but assume that the environment is such that the local policy gradients of agents provide useful learning directions for all agents (lines 99-103). Intuitively, this means that rewards are in a sense symmetrical between agents, in that actions that work for one agent also work for another agent if they swapped positions. Hence, SEAC could also be applied in some competitive tasks. We use examples of such environments in our evaluation. For example, most LBF tasks are not fully cooperative (Fig. 3efg) in the sense that agents do not share rewards. A food captured by a single agent can deprive other agents of rewards (competition). We provide a theoretical grounding of our symmetry assumption in Section C of the supplementary material.

3.2 & 3.4 We agree that the citations on self- and population-play, especially with respect to exploration, are relevant. We will investigate this relation further and include respective related work in Section 2.

Reviewer 2

3.1 Hyperparameters for MADDPG and QMIX were optimised using a grid search over learning rate, exploration rate, batch sizes (and more) with the grid centred on the hyperparameters used in the original papers and parameter performance tested in all used environments. Thus, MADDPG/QMIX used individually-tuned hyperparameters and were not limited to the hyperparameters and network architectures tuned on the SEAC/SNAC/IAC algorithms. This clarification was missing from the submitted version, and we now include it in Section 5.3.

3.2 We compared against MADDPG and QMIX (as well as IAC/SNAC baselines) which, in our experience, show good performance across different tasks. We will look into the cited work from ICML 2019.

Reviewer 3

8. *[Why CDTE algorithms underperform SEAC and baselines]* MADDPG and QMIX rely on the global state during training, which is not always be desirable: This global state (often approximated through a concatenation of agents' observations) grows with respect to the number of agents, which leads to large networks that are harder to train, especially in the absence of a dense reward signal. This is apparent in RWARE, where observations are high-dimensional and rewards are very sparse, and both QMIX and MADDPG do not learn efficiently (Tab. 1). We offer an alternative CTDE approach, SEAC, in which the centralised training is exploited without growing network sizes.

8. For a fair comparison we remain consistent with the original implementations of QMIX and MADDPG. Indeed, MADDPG/QMIX and other methods could also use experience sharing. We consider this generality a strength of our approach (e.g. see discussion in lines 124-130). Our experiments on methods using experience replay (Section D of supplementary material) also indicated a meaningful improvement, although not as significant as for SEAC.

8. Thank you for pointing us to the cited paper which we now include in the related work section.

Reviewer 4

3.Q1 Indeed, the likelihood ratios could tend to zero which can zero-out the respective gradients. However, our experiments indicate that the IS weights stay in a desirable range (Fig 5: [IS weights centered around 1.0+-0.5]) indicating that agents learn similar but not identical policies (lines 226-227). We observe that *small* differences in policies can have a significant impact in the overall coordination of agents in the tested environments. In RWARE (Fig. 2a), optimal behaviours are very similar (when seeing a requested shelf, pick it up and deliver it) but small differences, e.g. some agents giving way to others in corridors, allow for a much better overall efficiency. Using identical policies (SNAC) leads to agents colliding and clustering in narrow places, disrupting each other. Similarly, in LBF (Fig. 2c) agents must learn similar policies (go towards the same food locations) but their policies must collect the food from different neighbouring cells to avoid collision with other agents. In that sense, the reviewer's intuition is not wrong, but we consider environments in which agents can coordinate effectively without requiring greatly dissimilar policies.

3.Q2 We agree that MADDPG is more general in this regard, however we would like to point out that SEAC allows for some flexibility with experience sharing beyond fully-homogeneous tasks as long as our symmetry assumption holds between some agents (see answer to Reviewer 1, 3.1). For example, learning in a predator-prey environment with multiple preys and multiple predators could make use of SEAC by sharing experience only within preys and predators, respectively. This could be trivially achieved by setting the lambdas to 1 for agents in the same role, and 0 for other agents. Learning lambda values as part of the RL process could be the subject of future work, and might strongly relate to prior work in MARL with role assignment and similar ideas.