We thank all reviewers for the quality of their reviews. We have grouped in the 1st paragraph the common questions.
All reviewers. **Clarification of experiment goals**. The goal of the experiments is threefold: 1) Check that the calibration targets are reached: for each experiment j we have n_j targets that we want to match simultaneously (table 3 in appendix). We present some of these targets in Fig. 2, and targets for all experiments are presented in appendix Fig. 3-7. The calibrator agent's reward in Fig. 1 (and appendix Fig. 2) quantifies precisely the fit (1=perfect fit, cf. appendix B2). 2) We check empirically in Fig. 1 that the agents rewards converge (supertype 1 and 2), indicating that equilibrium is reached. 3) We check that our algorithm smoothly varies the parameters to be calibrated (supertypes) in Fig. 3 and appendix Fig. 8-12, hence preventing potential reward divergence observed for Bayesian optimization in Fig. 1 (experiment 4). **Extended transitivity assumption**. Our goal is to understand the implications of using a shared policy and in particular the gradient update (3). Due to (4), (3) is in fact self-play for a certain game (Def. 1), which implies that you need a class of games for which self-play converges, which we know is a strong requirement [3]: there, it is said that self-play works for "transitive games", hence we generalize the transitivity assumption in [3] from zero sum to general sum 2-player games. We do not have yet a full picture of the class of extended transitive games, but we provide intuition behind the concept with the following example: assume you play SpaceInvaders but with n players on the screen. At the start, players are dummy and miss enemies. Then, one player becomes smarter and finds a way to hit enemies. Transferring that knowledge to other players will make player 1 worse-off (lower score due to other players hitting enemies), but still better-off than at the beginning where he/she was missing enemies: this game is extended transitive. We believe that extended transitive games are those where there is some reward in the outside world (enemies in SpaceInvaders, customer transactions in our paper) that players can collect by learning *game skill*. **Potential and Bayesian games**. Extended transitive games are not potential games since there is no common potential that gets increased whenever a player increases its utility: indeed, in assumption 1, the second step in moving $(x, x) \to (y, x) \to (y, y)$ may not be an improvement for player 1, however it can be seen as a kind of "2-step potential game" since the utility gets improved over 2 steps based on single player improvement. Our game associated to $\widehat{V}$ implicitly involves types (inside the expectation) as in Bayesian games, however because of the trick of going from n to 2 players (the latter are 2 abstract players in that they are not part of the n agents), the 2nd argument of the function $\widehat{V}(\cdot, \cdot)$ actually ties out $n-1$ agents together with a policy in $\mathcal{X}$, hence we chose to introduce the terminology "shared equilibrium" since the shared nature of the policy is rooted in the definition of the game $\widehat{V}$.
R1. We do not claim that the game where each player i receives a utility of $V_i(\pi_i, \pi_{-i}; \Lambda_i, \Lambda_{-i})$ is symmetric: indeed as you correctly say, it is not, because you would have to permute the supertypes Λ too if you permute the π's. We claim that the 2-player game of Def. 1 with payoff $\widehat{V}$ is symmetric: here we think the confusion comes from our definition of the word "payoff" in L159. Any 2 player symmetric game where $u_i(\pi_1, \pi_2)$ is the utility received by player i satisfies by symmetry $u_1(\pi_1, \pi_2) = g(\pi_1, \pi_2)$ and $u_2(\pi_1, \pi_2) = g(\pi_2, \pi_1)$ for some g which we define as *payoff* in L159. The game of Def. 1 is symmetric by construction since we define it as the symmetric game associated to payoff $g := \widehat{V}$: this is an abstract game in the sense that it has 2 abstract players that are not part of the n agents, the 1st abstract player chooses π_1 and gets $\widehat{V}(\pi_1, \pi_2)$ (cf. L161-167), which is the expected utility received by getting assigned a random supertype using π_1 playing against all other agents using π_2, cf. (4). The 2nd abstract player receives $\widehat{V}(\pi_2, \pi_1)$. We find insightful that the function $\widehat{V}$ emerges naturally out of the gradient update (3), due to (4); Existence + finiteness of all self-play sequences used in the proof of Thm 2 is given by Lemma 2 and its proof.
R2. Benefits of formalizing the problem as RL: the conceptual problem being solved by the calibrator agent is an online search in the supertype space in order to achieve calibration targets. In our approach, supertypes get updated smoothly using properties of specific RL algos (PPO here). Further, one can allow the calibrator to sample N consecutive (batches of B) actions per policy update (instead of 1 in the vanilla version), thus evaluating the right direction to move to in the supertype space based on a sequence of N observations/actions/rewards (we used $N = 3$ in our experiments, cf. L156 of appendix).
R3. Rewards $\mathcal{R}$ can depend on both other agents' states and actions, but not on who plays them: $\mathcal{R}$ has to be invariant w.r.t. permutations of the other agents' states/actions, thus ensuring that the expected reward in (2) only depends on Λ_i: we will add it to the revision. We do not study the mean-field limit but our finite player setting with supertypes naturally allows to group agents under specific distributions of types, thus reducing the number of simulation parameters while keeping heterogeneity in the MAS and allowing coherent scaling w.r.t. the number of agents: this is exploited in sections 3 for calibration, and illustrated in the experiments (yes, Λ_i is defined on the same space $\forall i$, cf. L97). In L167-170 we defined a pure strategy as an element of the game's strategy space $\mathcal{X}$: this is consistent with functional form games of [3]. Thm 1 gives insight on the nature of games for which we get convergence of self-play and states that the endpoint is an $\epsilon-$Nash, where none of the 2 players can improve its utility of more than ϵ (stopping criterion).
R4. We will clarify the experiment section in the revision. Yes, mathematically, we mean the gradient of the function V_{Λ_i} with respect to its first variable, taken at the point (π_θ, π_θ). We should clarify L97 that the space $\mathcal{S}^\Lambda$ is assumed to be a subset of $\mathbb{R}^d$ as we do L234: since (groups of) agents are mapped to supertypes, "moving" a supertype in $\mathbb{R}^d$ as in algo 1 precisely means agents shifting between "fixed" supertypes, but in a continuous space.