A Game-Theoretic Framework for Managing Risk in Multi-Agent Systems

Oliver Slumbers [1][2]   David Henry Mguni [2]   Stefano B. Blumberg [1]   Stephen McAleer [3]   Yaodong Yang [4]
Jun Wang [1]

Abstract

In order for agents in multi-agent systems (MAS) to be safe, they need to take into account the risks posed by the actions of other agents. However, the dominant paradigm in game theory (GT) assumes that agents are not affected by risk from other agents and only strive to maximise their expected utility. For example, in hybrid human-AI driving systems, it is necessary to limit large deviations in reward resulting from car crashes. Although there are equilibrium concepts in game theory that take into account risk aversion, they either assume that agents are risk-neutral with respect to the uncertainty caused by the actions of other agents, or they are not guaranteed to exist. We introduce a new GT-based Risk-Averse Equilibrium (RAE) that always produces a solution that minimises the potential variance in reward accounting for the strategy of other agents. Theoretically and empirically, we show RAE shares many properties with a Nash Equilibrium (NE), establishing convergence properties and generalising to risk-dominant NE in certain cases. To tackle large-scale problems, we extend RAE to the PSRO multi-agent reinforcement learning (MARL) framework. We empirically demonstrate the minimum reward variance benefits of RAE in matrix games with high-risk outcomes. Results on MARL experiments show RAE generalises to risk-dominant NE in a trust dilemma game and that it reduces instances of crashing by 7x in an autonomous driving setting versus the best performing baseline.

1. Introduction

Game Theory (GT) is a fundamental tool for resolving problems within multi-agent systems (MAS) (Wellman, 2006). Formalising scenarios in MAS as games allows practitioners to model stable outcomes in real-world settings by computing equilibrium solutions. A core challenge for MAS research is building systems that perform effectively while mitigating the risk of adverse events for the agents within the system. In particular, as the level of integration with humans and AI agents increases, it becomes increasingly important to avoid dangerous events for humans i.e. car crashes (Gal & Grosz, 2022). In AI-only systems, risk can come in the form of agents taking low probability yet dangerous strategies (e.g. ϵ-greedy policies outside of training (Mnih et al., 2015)). Human-AI MAS differ markedly from those of simulated systems that involve computerised agents only, however the problem remains the same – humans are prone to execution errors, a trait that seldom applies to computerised agents (Gal & Grosz, 2022). Tackling this challenge requires equilibrium concepts that can account for all possible rewards dependent on other agents' actions and, enable agents to adopt risk-averse strategies in response.

There are two prominent approaches in GT that propose equilibrium concepts that address risk: Trembling Hand Perfect Equilibrium (THPE) (Bielefeld, 1988) and Quantal Response Equilibrium (QRE) (McKelvey & Palfrey, 1995)). However, due to the linearity of expected utility (EU) these concepts can undervalue comparatively large costs with low probability given an agent's tolerance for risk, which undermines their ability to successfully resolve notions of safety and risk in various practical MAS (e.g. crashing has a large negative utility). For example, as demonstrated in Fig.1, a probability of 0.01% of Player 2 *Overtaking* only impacts Player 1's EU by 0.5. Therefore, based on EU, *Overtaking* is preferable for Player 1, however exposing it to the possi-

---
[1]University College London, London, UK [2]Huawei Technologies, London, UK [3]Independent Researcher [4]Peking University, Beijing, China. Correspondence to: Oliver Slumbers <oliver.slumbers.19@ucl.ac.uk>.

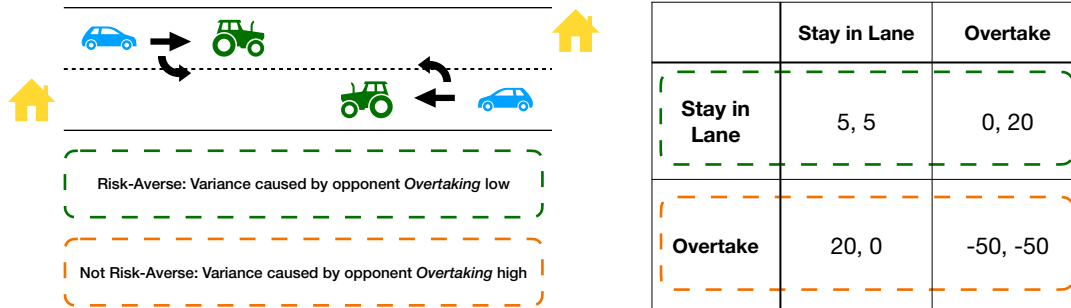|	Stay in Lane	Overtake
Stay in Lane	5, 5	0, 20
Overtake	20, 0	-50, -50

Figure 1. Two cars are rewarded for reaching their destination quickly. They are stuck behind slow tractors but can stay in their lanes and arrive slowly. They can also overtake the tractors to arrive quickly, but if both overtake they will cause large delays, leading to large negative payoffs.

bility of congestion in the middle of the road. Furthermore, it is difficult to control the level of risk-aversion in these frameworks as they are defined only in terms of EU.

Other approaches characterise risk in terms of concave/convex utility functions and study convergence to classical GT equilibria. In Fiat & Papadimitriou 2010, different attitudes to risk are theoretically considered, in particular a risk-averse variant based on including variance in the utility function similar to our proposal. However, they find that almost all of their risk-adjusted games will not have mixed Nash equilibrium. Even if the risk adjusted game does have a mixed Nash equilibrium, small mistakes made by players in interpreting the payoffs of the game will likely cause the equilibrium to be unstable. In contrast, our approach guarantees the existence of a risk-adjusted equilibrium. Furthermore, Fiat & Papadimitriou 2010 only provide theoretical analyses and we are interested in practical solvers to attain a solution.

In this paper we propose a novel equilibrium concept called *Risk-Averse Equilibrium* (RAE). RAE distinguishes itself from THPE and QRE by including the second moment (variance) of the utility function with respect to the strategies of other agents. Unlike these approaches RAE has a risk-aversion parameter explicitly controlling the amount of risk an agent is willing to accept. This places larger emphasis on deviations from expected utility, where variance in RAE values the 0.01% *Overtake* probability in Fig. 1 at 47.6γ (vs. 0.5), where $\gamma > 0$ and generally at minimum 0.1, see Appendix G. We show that RAE can be computed using numerical and iterative methods, unlike the theoretical approaches in (Fiat & Papadimitriou, 2010), which unlock its ability to resolve modelling large-scale MARL games simulating real-world settings. We also show that, for any desired expected utility, RAE minimises the corresponding variance. In other words, RAE reduces highly adverse outcomes caused by the actions of other agents in the system.

To demonstrate the benefits of RAE, we perform a series of experiments in risky matrix games, and two larger-scale MARL problems which are closer to real-world scenarios: a grid-world Stag-Hunt trust dilemma, and an autonomous driving scenario. Our results validate our theoretical advances on risky matrix games by showing that the RAE solutions provide the same EU as our baselines, whilst being safer in providing lower variance. In the Stag-Hunt MARL setting, RAE outperforms the baselines and arrives at the safe solution but the other methods do not. In the autonomous driving setting, RAE reduces crashes 7x in test episodes as compared with the best performing baseline equilibrium.

Our contributions are:

1. We propose RAE, a new solution concept that accounts for expected utility (EU) and utility variance (UVar) in determining a strategy (Sec. 3).

2. We prove that RAE's strategy is the minimum variance solution (Prop. 3.2) and we prove RAE's existence (Thrm. 3.3).

3. We introduce two solvers to compute RAE in small action spaces (Sec 4.1) and large action spaces (Sec. 4.2).

4. We validate RAE on risky matrix games (Sec. 5.1) and in experiments on two MARL settings: a grid-world Stag-Hunt (Sec. 5.2) and autonomous driving scenario (Sec. 5.3), where RAE outperforms state-of-the-art GT approaches THPE, QRE and simple NE.

2. Related Work

Risk in GT: Baselines Harsanyi et al. 1988 introduced risk-dominant Nash equilibria (NE) (Nash, 1951), which, when the strategies of other agents is unknown, leads to the NE with the lowest losses if deviated from. However,

if none of the NE are robust to risk initially, then the risk-dominant strategy amongst NEs also will not be. THPE (Bielefeld, 1988) models risk by having agents 'tremble' through all actions having positive probability in a mixed-strategy. However, iterated deletion of weakly dominated strategies can lead to the THPE strategy being removed from consideration. McKelvey & Palfrey 1995 introduce QRE which accounts for potential errors in strategy selection to more accurately represent human observed strategies. Note, THPE and QRE utilise EU as their risk measure which is insensitive to low probability, high-value deviations as long as they are offset by high probability mean values (Royset, 2022). THPE, QRE and NE are baselines.

Risk in GT: Other Approaches Yekkehkhany et al. 2020 propose a mean-variance equilibrium where the variance relates to reward probabilities, rather than variance caused by the strategies of others as in RAE. Notably, in the model-free machine learning setting reward probabilities are generally not known, and is not within the scope of this work. Fiat & Papadimitriou 2010 consider how NE existence is impacted by non-EU maximising agents, and derive results for multiple risk categorisations, however limit their work to theoretical propositions that do not extend to baselines for this work. Risk in competitive network games (Wardrop, 1952) is widely studied, and is based on a generalisation of the classical selfish routing 'game' (Beckmann et al., 1956) to incorporate uncertain delays. The impact of mean-risk utility frameworks on WE is particularly well studied (Ordóñez & Stier-Moses, 2010; Lianeas et al., 2019; Cherukuri, 2019), however WE is not applicable to non-network games as studied in this work. In terms of NE, the major distinction in risk analysis is between non-atomic (i.e. infinite agents) and atomic (i.e. finite agents) settings. In non-atomic settings Meir & Parkes 2015; Nikolova & Stier-Moses 2012 the marginal impact in terms of risk of each agent on each other is infinitesimal and this leads to a distinctly different analysis than required in the atomic setting of this work (Nikolova & Stier-Moses, 2014). In the atomic setting which parallels more with our setting, Nikolova & Stier-Moses 2014 study a mean-standard deviation model of travel time along a network path, and whilst they are able to show the existence of pure-strategy NE in exogenous risk settings, they are unable to in endogenous risk settings concerned in our work. Piliouras et al. 2016 also study a setting where the risk is largely exogenous, as it is defined by randomised schedulers which controls the ordering through congestion edges in the network, which is not a concept in our setting and therefore can not act as a baseline.

Risk in MARL: RAE fits broadly into risk-sensitive MARL solutions such as: RMIX (Qiu et al., 2021) which optimises decentralised CVaR policies in cooperative risk-sensitive settings, RAM-Q and RA3-Q (Gao et al., 2021) utilise an adversarial approach to promote variance reduction, or risk-sensitive DAPG (Eriksson et al., 2022) which approaches risk in Bayesian games in terms of the CVaR induced by the possible combinations of types in the game. However, as we are specifically concerned with GT-based equilibrium concepts we will not directly compare to these methods.

GT and MARL GT and RL have overlapped in settings where the number of actions in a game becomes too big to write down trivially. For these games with larger action spaces, we consider Policy-Space Response Oracles (PSRO) (Lanctot et al., 2017; McAleer et al., 2020; Perez-Nieves et al., 2021; Feng et al., 2021; McAleer et al., 2022b) which generalises the Double Oracle (DO) (McMahan et al., 2003; Dinh et al., 2022; McAleer et al., 2021) framework from small action spaces to large action spaces by replacing actions with RL policies. In this work, we propose a framework at the overlap between risk-averse GT and risk-averse MARL which involves adaptations to the risk-neutral frameworks mentioned here. In future work we will investigate applying other recent deep RL approaches for finding equilibria (Perolat et al., 2022; McAleer et al., 2022a; Sokota et al.) to our solution concept.

3. Risk-Averse Equilibrium Framework

We introduce RAE and detail its derivation from a mean-variance utility function. We derive key properties of RAE which characterise the two following important benefits: **1.** RAE is a minimum risk solution **2.** The existence of RAE for any finite N-player game (under standard conditions).

3.1. RAE Derivation

We begin by describing the underlying formalism of a normal form game (NFG). An NFG is the standard representation of strategic interaction in GT. A finite n-person NFG is a tuple (N, A, u), where N is a finite set of n players, $A = A^1 \times, ..., \times A^n$ is the joint action profile, with A^i being the actions available to player i, and $u = (u^1, ..., u^n)$ where $u^i : A \rightarrow \mathbb{R}$ is the real-valued utility function for each player. A player plays a mixed-strategy, $\boldsymbol{\sigma}^i \in \Delta_{A^i}$, which is a probability distribution over their possible actions. In Sec. 4.2 we replace atomic actions with neural networks and will therefore re-define our notation to keep clarity between the two game schemes.

The objective of RAE is to provide an equilibrium solution that is robust to other agents taking *any* action. Our approach considers both the EU (mean) and the potential UVar caused by an opponent's strategy, in particular noting that *all* actions may be taken by the opponent as if mistakes (low probability actions) can happen. Whilst low probability actions are not literally mistakes, they are simply a technical device to mimic the idea of a mistake, as described in Bielefeld 1988.

For simplicity, we provide definitions based on playing a symmetric game, such that two players share an action set $\mathcal{A}$, and a utility function u. We extend this to the non-symmetric case in Appendix K.1. Define the utility of action $a_i \in \mathcal{A}$ against action $a_j \in \mathcal{A}$ as $u(a_i, a_j)$ and the full utility matrix as $\mathbf{M}$, where the entry $\mathbf{M}_{i,j}$ refers to $u(a_i, a_j)$ and $\mathbf{M}_i$ refers to $u(a_i, a_j) \ \forall j$, i.e. the vector of utilities that action a_i receives against all other actions. We now define the *expected* utility of the mixed-strategy for player 1 $\boldsymbol{\sigma}$ versus the mixed strategy for player 2 $\boldsymbol{\varsigma}$ as

$$
\begin{aligned}
u(\boldsymbol{\sigma}, \boldsymbol{\varsigma}, \mathbf{M}) &= \sum_{a_i \in A} \sum_{a_j \in A} \sigma(a_i)\varsigma(a_j)u(a_i, a_j) \\
&= \boldsymbol{\sigma}^T \cdot \mathbf{M} \cdot \boldsymbol{\varsigma}.
\end{aligned}
\tag{1}
$$

The weighted co-variance matrix for $\mathbf{M}$ (i.e. the UVar values) is a $|A| \times |A|$ matrix $\boldsymbol{\Sigma}_{\mathbf{M},\boldsymbol{\varsigma}} = [c_{ij}]$ with entries

$$
c_{jk} = \sum_{a_i \in A} \varsigma(a_i)\big(u(a_i, a_j) - \bar{\mathbf{M}}_j\big)\big(u(a_i, a_k) - \bar{\mathbf{M}}_k\big),
\tag{2}
$$

where $\bar{\mathbf{M}}_i = \sum_{k=1}^{|A|} \varsigma_k u(a_i, a_k)$ is the EU for action i given the opponent mixed-strategy $\boldsymbol{\varsigma}$. This is a standard co-variance matrix, but the entries are weighted by the likelihood of them being selected by the opponent. A uniform weighting could be used, however we hypothesise that in terms of minimising UVar it is more appropriate to hedge against the variance caused by higher likelihood actions. Due to the nature of Eq. 5, all actions will, by design, receive positive probability ($> \epsilon$) under our framework and therefore will always provide some weight in the variance calculation, leading to low likelihood high-variance actions still having a large impact. This accounts for the central concept of RAE, that safe play should account for all actions. This allows us to define the mixed-strategy $\boldsymbol{\sigma}$ UVar as:

$$
\begin{aligned}
\text{Var}(\boldsymbol{\sigma}, \boldsymbol{\varsigma}, \mathbf{M}) &= \sum_{k=1}^{|A|} \sum_{n=1}^{|A|} \sigma(a_k)\sigma(a_n)c_{kn} \\
&= \boldsymbol{\sigma}^T \cdot \boldsymbol{\Sigma}_{\mathbf{M},\boldsymbol{\varsigma}} \cdot \boldsymbol{\sigma}.
\end{aligned}
\tag{3}
$$

The total utility function r which considers EU and UVar for mixed-strategy $\boldsymbol{\sigma}$ is,

$$
r(\boldsymbol{\sigma}, \boldsymbol{\varsigma}, \mathbf{M}) = u(\boldsymbol{\sigma}, \boldsymbol{\varsigma}, \mathbf{M}) - \gamma\,\text{Var}(\boldsymbol{\sigma}, \boldsymbol{\varsigma}, \mathbf{M}),
\tag{4}
$$

where $\gamma \in \mathbb{R}$ is the risk-aversion parameter.

Applying Eq. (4) to Fig. (1) we show why the utility function 4 is desirable. Consider two joint strategy profiles, $\boldsymbol{S}_1 = ((1 - \epsilon, 0 + \epsilon), (1 - \epsilon, 0 + \epsilon))$ and a THPE $\boldsymbol{S}_2 = ((0 + \epsilon, 1 - \epsilon), (1 - \epsilon, 0 + \epsilon))$ where $(1 - \epsilon, 0 + \epsilon)$ represents playing *Stay in Lane* with probability $(1 - \epsilon)$. $\epsilon = 0.01$ for this example. Profile $\boldsymbol{S}_1$ receives $u(\boldsymbol{S}_1) \approx 5$ and the THPE profile receives $u(\boldsymbol{S}_2) \approx 20$. However, $\text{Var}(\boldsymbol{S}_1) = 0.32$

and $\text{Var}(\boldsymbol{S}_2) = 47.6$, i.e. the THPE strategy has huge variance for Player 1. Therefore, $r(\boldsymbol{S}_1) \approx 5 - 0.32\gamma$ and $r(\boldsymbol{S}_2) \approx 20 - 47.6\gamma$ and we have for any risk-aversion parameter $\gamma > 0.32$ it is optimal to play $\boldsymbol{S}_1$.

To define RAE, we first define the best-response map:

$$
\begin{aligned}
\boldsymbol{\sigma}^*(\boldsymbol{\varsigma}) &\in \arg\max_{\boldsymbol{\sigma}} r(\boldsymbol{\sigma}, \boldsymbol{\varsigma}, \mathbf{M}) \\
\text{s.t. } &\sigma(a) \geq \epsilon, \forall a \in A \\
&\boldsymbol{\sigma}^T \mathbf{1} = 1,
\end{aligned}
\tag{5}
$$

where due to the quadratic term $\boldsymbol{\sigma}^T \cdot \boldsymbol{\Sigma}_{\mathbf{M},\boldsymbol{\varsigma}} \cdot \boldsymbol{\sigma}$ and the constraints ($\epsilon > 0$), we have a Quadratic Programming (QP) problem. The programme finds $\boldsymbol{\sigma}^*$ such that the total utility is maximised, whilst ensuring no actions are assigned negative probability, and that the probabilities sum to one. Finally, based on Eq. (5) we are able to define RAE:

Definition 3.1 (RAE). A strategy profile $\{\boldsymbol{\sigma}, \boldsymbol{\varsigma}\}$ is a risk-averse equilibrium if both $\boldsymbol{\sigma}$ and $\boldsymbol{\varsigma}$ are risk-averse best responses, in that they satisfy Eq. 5, to each other.

3.2. RAE Properties

The first property of RAE is that, given an opponent's strategy $\boldsymbol{\varsigma}$, one's owns strategy $\boldsymbol{\sigma}$ is the minimum UVar solution available given their desired EU μ_b:

Proposition 3.2. *Given $\boldsymbol{\varsigma}$ and any desired EU μ_b, there exists a γ such that the solution to Eq. 5 receives μ_b with the minimum possible UVar.*

Proof deferred to Appendix A.1. The consequence of this proposition matches with the goal of RAE - to provide a minimum risk solution given the level of risk tolerance of the user. γ is used to select how much EU μ_b is desired and, given the opponent's strategy $\boldsymbol{\varsigma}$, there is no other solution that can provide better risk performance than RAE. The main downside of this is that γ is a hyper-parameter and it may be difficult to know prior to training how it will exactly match to μ_b. We would expect that games that share similar reward distributions would share similar risk-variance trade-offs under the same values of γ. Therefore, one way to select γ is to use restricted and unrestricted versions of environments that share similar reward distributions. One can train on the restricted version more readily, and transfer the γ results to the harder unrestricted version. In Appendix C we show over a range of NFG dimension sizes, with rewards drawn from the same distribution, that EU and Var remain mostly consistent across values of γ and dimension, suggesting the proposed approach is feasible.

Secondly, a common property of most GT equilibrium is that a solution exists, at least in the finite game setting. For RAE, we note the following result in mixed-strategies:
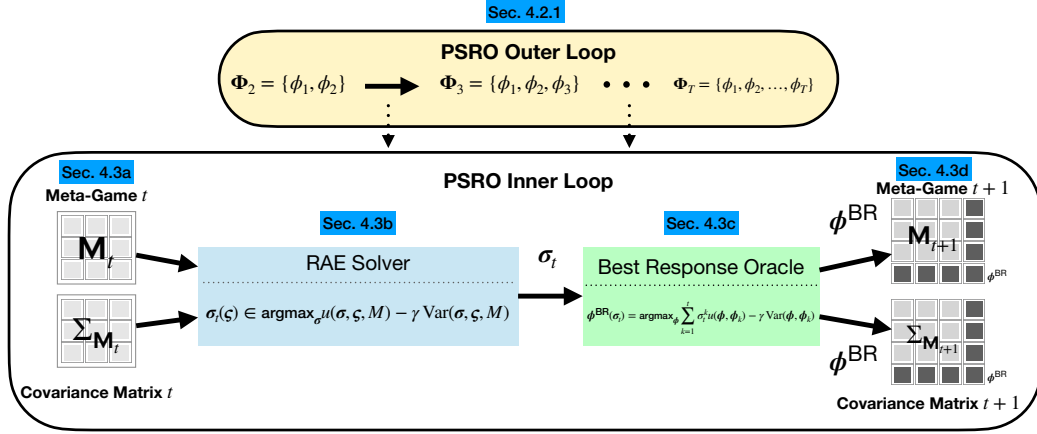
Theorem 3.3. *For any finite N-player game where each*

Figure 2. PSRO-RAE. Each element points to a corresponding subsection in the text, denoted in blue boxes. Note that $u(\cdot)$ and $\mathrm{Var}(\cdot)$ are overloaded to represent utility/variance between distributions over a population or utility/variance between two policies.

player i has a finite k number of pure strategies, $A^i = \{a_1^i, ..., a_k^i\}$, an RAE exists.

We defer the proof of the result to Appendix A.2. By establishing the existence of RAE solutions we have validated the practical relevance of RAE.

4. Finding RAE

This section proposes two solvers to find RAE, with stochastic fictitious play (SFP) for small action-spaces in Sec. (4.1) and with PSRO-RAE for large action-spaces in Sec. 4.2.

4.1. Stochastic Fictitious Play

We start by showing that our total utility function is a form of SFP (Fudenberg & Kreps, 1993) which can find an RAE in small NFGs. SFP has convergence guarantees in a selection of games, e.g. potential games (Monderer & Shapley, 1996a;b) and finite two-player zero-sum games (Robinson, 1951). Furthermore, SFP is also robust empirically in terms of convergence in game classes (Goldberg et al., 2013; Ganzfried, 2020) not listed, and we mirror these observations in Appendix B.

SFP is a learning process where players choose a best response to others time-average strategies. In SFP, a group of $n \geq 2$ players repeatedly play a $n-$player NFG. The state variable is $Z_t \in \Delta_S$, whose components Z_t^i describe the time averages of each player's behaviour,

$$Z_t^i = \frac{1}{t} \sum_{u=1}^{t} \sigma_t^i$$

where $\sigma_t^i \in \Delta_{A^i}$ represents the observed strategy of player i at time-step t. Each player best responds to the time-average strategy of their opponent, Z_t^{-i}, by maximising a perturbed

utility function $\bar{u}$

$$\sigma_{t+1}^i = \arg \max_{\sigma} \bar{u} \tag{6}$$

$$= \arg \max_{\sigma} u^i(\sigma, Z_t^{-i}, \mathbf{M}) - \lambda v^i(\sigma) \tag{7}$$

where $v^i(\sigma) : \Delta_A \to \mathbb{R}$ perturbs u^i such that it is strictly concave (unique global maximum) whilst applying greater than zero probability to all actions.

Proposition 4.1. *Replacing the best-response Eq. (6) with the best-response map Eq. (5) satisfies the conditions of $\bar{u}^i$ for a SFP process.*

Note, SFP does not necessarily converge in all game classes (but is robust empirically, see Appendix B). Therefore, we show that if the SFP process does converge to a strategy then that strategy is an RAE.

Proposition 4.2. *Suppose the SFP sequence $\{Z_t\}$ converges to σ in observed strategies [1], then σ is a risk-averse equilibrium.*

Note for SFP we require a stronger notion of convergence in observed strategies σ_t^i rather than in beliefs Z_t^i, but a converged final σ_t^i is a risk-averse equilibrium.

4.2. PSRO-RAE

For games that can't be displayed in the normal-form, we extend the iterative solution framework PSRO (Lanctot et al., 2017) to RAE, which uses RL policies as proxies for actions. PSRO-RAE approximates equilibria in large games by finding a small representative collection of risk-averse

---

[1]Convergence in time-average Z_t does not imply convergence in the actual strategy taken at each t, but may imply cyclic actual behaviour that results in average behaviour converging.
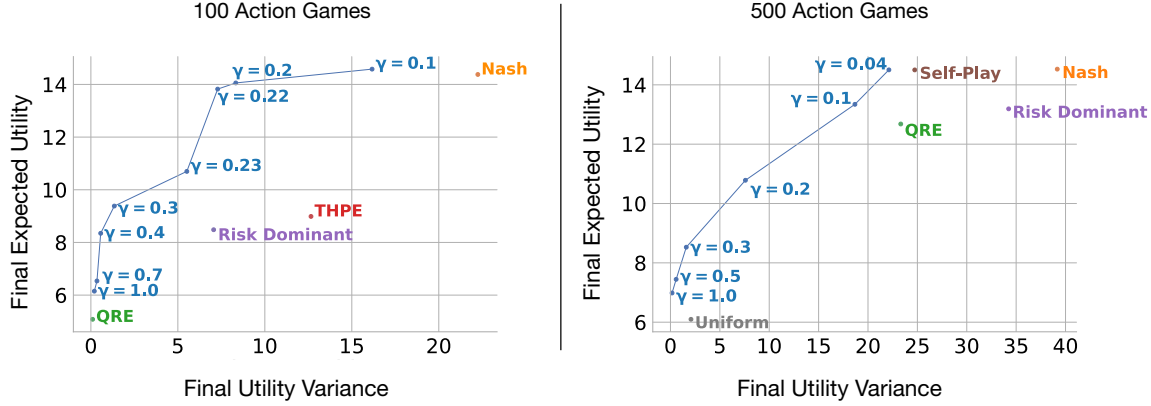
Figure 3. a) SFP on NFGs with 100 actions, b) PSRO on NFGs with 500 actions. Final EU vs. UVar results. RAE values for multiple γ form an *efficient frontier* (Markowitz, 1991) and show that, whilst baselines achieve similar EU they have large UVar solutions. In Fig. a) we exclude the payoff dominant NE result as its UVar was too large, whilst in Fig. b) we exclude the THPE result for the same reason.

RL policies which are sampled by RAE. Whilst PSRO does not have any practical convergence guarantees (in the limit *all* policies may be found and displayed in the normal-form, allowing for an exact solution), PSRO generally finds strong solutions without requiring all potential policies (Perez-Nieves et al., 2021; Feng et al., 2021; McAleer et al., 2022c). We provide a visualisation of the PSRO-RAE process in Fig. (2), and provide an algorithm in Appendix F.

Consider two-player stochastic games G defined by the tuple $\{\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{G}\}$, where $\mathcal{S}$ is the set of states, $\mathcal{A} = \mathcal{A}^1 \times \mathcal{A}^2$ is the joint action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$ is the state-transition function and $G = \{g_1, g_2\}$ is the set of rewards where $g^i : S \times \mathcal{A} \to \mathbb{R}$ is the reward function for player i (we use reward for MARL settings, and utility for NFG settings). An *agent* is a policy ϕ, where a policy is a mapping $\phi : \mathcal{S} \times \mathcal{A} \to [0,1]$ which can be described in both a tabular form or as a neural network. The expected reward between two *agents* is defined to be $M(\phi_i, \phi_j)$ (i.e., in the same manner defined for NFGs in Sec. 3.1), and represents the expected reward to agent ϕ_i against opponent ϕ_j.

4.2.1. PSRO OUTER LOOP

PSRO does $T \in \mathbb{N}^+$ updates on a meta-game $\mathbf{M}$ (an NFG made up of RL policies as actions). At every iteration $t \leq T$, a *player* is defined by a population of fixed *policies* $\mathbf{\Phi}_t = \mathbf{\Phi}_0 \cup \{\phi_1, \phi_2, ..., \phi_t\}$, where $\mathbf{\Phi}_0$ is the initial random policy. For notation convenience, we consider the *single-population* case where players share the same $\mathbf{\Phi}_t$, and refer the reader to Appendix K.2 for the multi-population formulation.

4.2.2. PSRO INNER LOOP

a, d) Meta-Game & Covariance Matrix At the start of the iteration t inner loop, each population has a *meta-game* $\mathbf{M}_t$, a reward matrix between all the *policies* in the popula-

tion, with individual entries $M(\phi_i, \phi_j) \; \forall \phi_i, \phi_j \in \mathbf{\Phi}_t$. In addition, each population also generates a covariance matrix $\mathbf{\Sigma}_{\mathbf{M}_t}$ defined by Eq. 2. At the end of iteration t inner loop, both $\mathbf{M}_t$ and $\mathbf{\Sigma}_{\mathbf{M}_t}$ are updated to include a new policy.

b) Meta Distribution We require a way to select which $\phi_t \in \mathbf{\Phi}_t$ will be used for training opponents. The function f is a mapping $f : \mathbf{M}_t \to [0,1]^t$ which takes as input a meta-game $\mathbf{M}_t$ and outputs a *meta-distribution* $\boldsymbol{\sigma}_t = f(\mathbf{M}_t)$. The output $\boldsymbol{\sigma}_t$ is a probability assignment to each *policy* in the population $\mathbf{\Phi}_t$, the equivalent of a mixed-strategy in a NFG, except actions are now RL policies. We apply RAE (Def. 3.1) as the meta-solver. As ϕ are RL policies then policies are sampled by their respective probability in $\boldsymbol{\sigma}_t$.

c) Best Response Oracle At each epoch $\mathbf{\Phi}_t$ is augmented with a new policy that is a *best-response* (BR) to the meta-distribution $\boldsymbol{\sigma}_t$. The BR oracle aims to optimise the same objective function of that optimised by the meta-distribution. For example, in Vanilla PSRO, the Nash meta-distribution optimises for environment reward and the BR oracle also optimises a new agent in terms of EU. This can be found with any optimisation process such as RL or an evolutionary algorithm. In our setting the meta-distribution optimises two metrics, the EU and the UVar, and therefore, we need a BR oracle that optimises the same dual objective.

In terms of RL quantities, this translates to maximising the expected total reward (i.e. the total of the per-step rewards) whilst minimising the variance of the total reward caused by the sampling of different RL agents from $\boldsymbol{\sigma}_t$.

To achieve this, we follow the approach of (Zhang et al., 2021) that optimises both the total reward and per-step reward variance by solving an augmented MDP where the per-step reward g_t^i is replaced by:
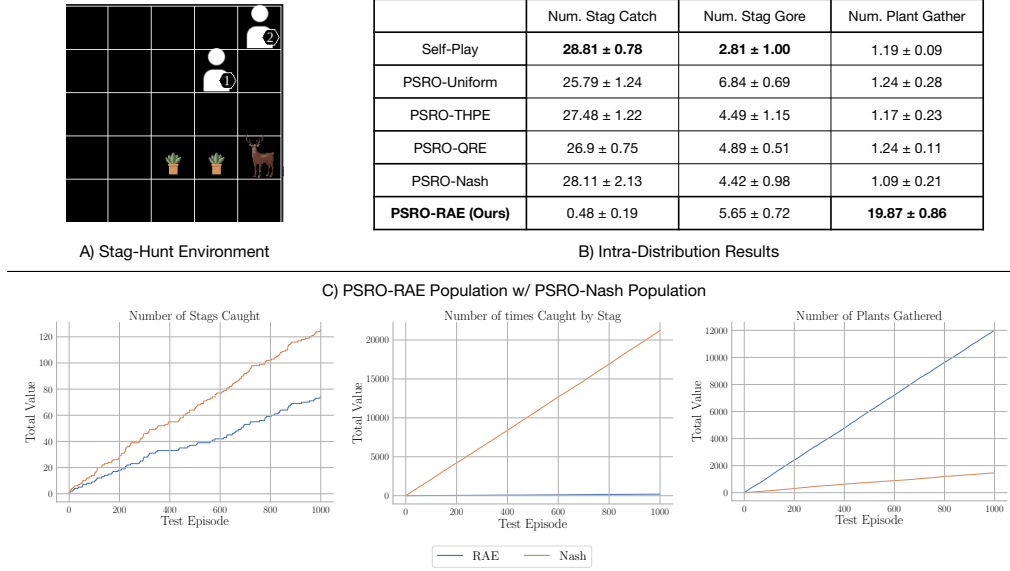
	Num. Stag Catch	Num. Stag Gore	Num. Plant Gather
Self-Play	28.81 ± 0.78	2.81 ± 1.00	1.19 ± 0.09
PSRO-Uniform	25.79 ± 1.24	6.84 ± 0.69	1.24 ± 0.28
PSRO-THPE	27.48 ± 1.22	4.49 ± 1.15	1.17 ± 0.23
PSRO-RAE (Ours)	0.48 ± 0.19	5.65 ± 0.72	19.87 ± 0.86
PSRO-QRE	26.9 ± 0.75	4.89 ± 0.51	1.24 ± 0.11
PSRO-Nash	28.11 ± 2.13	4.42 ± 0.98	1.09 ± 0.21

A) Stag-Hunt Environment B) Intra-Distribution Results

C) PSRO-RAE Population w/ PSRO-Nash Population

Figure 4. Stag-hunt environment results. A) The environment B) Intra-distribution results, e.g. both agents are controlled by a PSRO-RAE population C) Cumulative results over 1000 test episodes for a PSRO-RAE population vs. PSRO-Nash population.

$$\hat{g}_t^i = g_t^i - \lambda (g_t^i)^2 + (2\lambda g_t^i y_i)$$

where $y_i = \frac{1}{T} \sum_{t=1}^{T} g_t^i$ is the average of the per-step rewards during the data collection phase.

We choose the per-step reward as it is an upper bound of the total reward variance (Bisi et al., 2020), therefore reducing per-step variance will also reduce total variance, and is more effective computationally (Zhang et al., 2021). Additionally, as this variance is also with respect to the sampling probability defined by $\boldsymbol{\sigma}_t$ this optimises the correct co-variance matrix which is also weighted by $\boldsymbol{\sigma}_t$.

5. Experiments and Results

We validate RAE through three questions: 1) Does RAE find lower UVar solutions than baselines? 2) Do RAE strategies overlap with risk-dominant NE in some scenarios? 3) Are RAE strategies safe in a safety-critical driving scenario? Full experimental details in Appendix H.

5.1. Does RAE find lower UVar solutions than baselines?

Motivation/Overview Prop. 3.2 shows RAE can find a spectrum of solutions encompassing many EU values, whilst minimising the corresponding UVar. Therefore, if RAE can match the baseline's EU, whilst achieving lower UVar, then RAE is a better solution if safety is of concern.

Baselines NE (including risk/payoff dominant), THPE and QRE introduced in Sec. 2.

Experiment: Matrix Coordination Games NFGs where some actions provide a high EU if other agents select them, but have large costs if not. Other actions have lower coordinated EU but lower costs. These games are designed to highlight the issues of focusing on EU and ignoring UVar.

Results Results in Fig. (3), where (A) represents games with 100 actions solved using SFP (Sec. 4.1), and (B) represents games with 500 actions solved using PSRO (Sec. 4.2. We plot RAE solutions for multiple values of γ to generate a theoretical *efficient frontier*. An efficient frontier shows for values of EU what is the minimum possible UVar. Our results show that, whilst the baselines achieve a diverse range of EU values, they are unable to find the minimum UVar solution which RAE finds. This shows the strong flexibility of our approach, in that it is able to attain any EU that the baselines can achieve, whilst finding a lower UVar solution. This suggests that, if safety is of concern, then RAE is a better choice as any desired EU can be achieved whilst achieving a lower UVar than any of our baselines.

5.2. Are RAE strategies risk-dominant?

Motivation In some scenarios, it is likely that the only viable solutions overlap with the set of NE (for example, in trust dilemmas). Therefore, in these scenarios, if RAE finds safe solutions, we would expect that the RAE solutions would overlap with the risk-dominant NE solutions.

Experiment: MDP Stag-Hunt: We use an MDP-based adaptation of a classic trust dilemma game from

A) Intra-Distribution Results

	Eqm Reward	Eqm Variance	Worst-Case	Num. Crashes	Num. Arrivals
Self-Play	0.97 ± 2.14	1.53 ± 0.12	-4.80 ± 5.92	38.5 ± 4.95	11.50 ± 4.95
PSRO-Uniform	-0.69 ± 2.64	1.705 ± 0.024	-4.80 ± 5.201	39.5 ± 22.81	8.00 ± 2.83
PSRO-THPE	0.34 ± 0.87	1.670 ± 0.09	-7.59 ± 2.140	42 ± 2.16	8.30 ± 2.12
PSRO-QRE	1.60 ± 0.97	1.44 ± 0.13	-2.84 ± 0.94	43.0 ± 2.85	7.00 ± 2.85
PSRO-Nash	0.85 ± 2.14	1.53 ± 0.12	-4.80 ± 5.76	38.5 ± 2.12	11.50 ± 4.95
PSRO-RAE (Ours)	**4.36 ± 2.07**	**0.333 ± 0.004**	**0.10 ± 2.68**	**5.5 ± 0.71**	**46.00 ± 1.41**

B) PSRO-RAE Average Position Heatmap
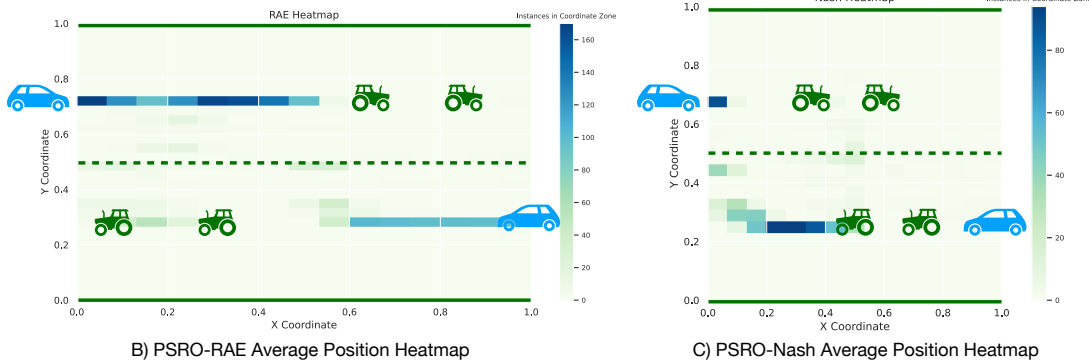

C) PSRO-Nash Average Position Heatmap

Figure 5. A) Results on 50 episodes over 5 seeds for intra-distribution testing, e.g. both agents controlled by PSRO-RAE B) Average position heatmap for PSRO-RAE solution over 200 episodes C) Average position heatmap for PSRO-Nash solution over 200 episodes.

(Peysakhovich & Lerer, 2018) where there is a payoff-dominant equilibrium (chasing the stag) and a risk-dominant equilibrium (gathering plants).

Baselines As this is a MDP task we integrate Nash, Uniform, Self-Play, THPE, QRE, described in Sec. 2 and Appendix I, as the meta-distributions in the PSRO framework, denoted as e.g. PSRO-{Nash}. In this setting we limit our baselines to PSRO-{Variant} algorithms, and do not consider non-population risk-aversion algorithms (standard in PSRO literature). Full details in Appendix I.

Results Fig. (4B) shows all baselines arrive at the payoff-dominant stag catching strategy, whereas RAE arrives at the 'risk-dominant' plant gathering strategy. The baselines solution is risky due to its susceptibility to coordination failure i.e. when only one agent hunts the stag. This could occur frequently, for example, in situations where agents are not able to communicate with each other, or agents do not know each others strategies. For example, we show this by placing a PSRO-RAE population and a PSRO-Nash population into the environment together as co-players, shown in Fig. (4C). The Nash population still attempts to hunt the stag (unable to communicate, has a fixed strategy), but in this case the RAE population is still gathering plants - leading to the Nash population being caught by the stag many times, and the RAE population remaining safe.

5.3. Are RAE strategies safe?

Motivation RAE is designed to avoid strategies that are overly susceptible to other agent's strategies by limiting UVar. We examine how RAE acts in a larger-scale MARL autonomous driving setting where avoiding any large negative outcome (e.g. crashing) is critical, in particular in the presence of strategies that can look overly advantageous unless coordination between agents fails.

Experiment: Autonomous Driving Scenario (Leurent, 2018): MDP recreation of Fig. (1) scenario; two-way traffic with slow-moving vehicles and faster moving agents behind that may be interested in overtaking.

Baselines MDP task, refer to Sec. 5.2.

Results Results are in Fig. (5), with results for a larger-scale representation of the environment (by increasing the number of vehicles from 6 to 25) in Appendix E. In Table (5A) we provide a collection of environment metrics where the average value is based on 50 environment episodes and the standard deviation is from 5 training seeds. In terms of EU and UVar, RAE actually outperforms the baselines considerably, whilst *maintaining strong worst-case performance*. Notably, RAE arrives at a strategy that very rarely crashes, and nearly always arrives at the final destination. The same conclusion can not be drawn for the baselines which often crash and fail to reach the destination. To understand this better, in Fig. (5B, C) we provide position heat-maps of a PSRO-RAE and a PSRO-Nash car respectively. RAE exe-

cutes the safe strategy, i.e. follow behind until all vehicles in the on-coming lane have passed and then proceeds to overtake. This strategy remains sensitive to the risk-element of the environment, overtaking and crashing, which is our desired outcome. On the other hand, the Nash strategy overtakes straight away and nearly always ends up in a crash due to congestion in the middle of the road.

6. Conclusion

We introduce a new risk-averse equilibrium, RAE, based on agents considering the variance of the utility function alongside the expected value. Theoretically, we prove the existence and solvability of RAE and provide methods for arriving at an RAE in both small and large-scale game settings. Empirically, we show that our RAE is able to locate minimum variance solutions for any EU, act as a NE selection method in the presence of risk-dominant NE, and is effective at finding a safe equilibrium in a safety-sensitive autonomous driving environment. Avenues for future work should focus on the limitations of the current RAE approach, namely non-convergence guarantees in certain classes of games and the fact that RAE minimises upside and downside variance, where minimising downside variance only would be a desirable property.

References

Beckmann, M., McGuire, C. B., and Winsten, C. B. Studies in the economics of transportation. 1956.

Bielefeld, R. S. Reexamination of the perfectness concept for equilibrium points in extensive games. In *Models of Strategic Rationality*, pp. 1–31. Springer, 1988.

Bisi, L., Sabbioni, L., Vittori, E., Papini, M., Restelli, M., et al. Risk-averse trust region optimization for reward-volatility reduction. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, pp. 4583–4589, 2020.

Cherukuri, A. Sample average approximation of cvar-based wardrop equilibrium in routing under uncertain costs. In *2019 IEEE 58th Conference on Decision and Control (CDC)*, pp. 3164–3169. IEEE, 2019.

Dinh, L. C., McAleer, S., Tian, Z., Perez-Nieves, N., Slumbers, O., Mguni, D. H., Wang, J., Bou Ammar, H., and Yang, Y. Online double oracle. *TMLR: Transactions on Machine Learning Research*, 2022.

Eriksson, H., Basu, D., Alibeigi, M., and Dimitrakakis, C. Risk-sensitive bayesian games for multi-agent reinforcement learning under policy uncertainty. *arXiv preprint arXiv:2203.10045*, 2022.

Feng, X., Slumbers, O., Wan, Z., Liu, B., McAleer, S., Wen, Y., Wang, J., and Yang, Y. Neural auto-curricula in two-player zero-sum games. *Advances in Neural Information Processing Systems*, 34, 2021.

Fiat, A. and Papadimitriou, C. When the players are not expectation maximizers. In *International Symposium on Algorithmic Game Theory*, pp. 1–14. Springer, 2010.

Fudenberg, D. and Kreps, D. M. Learning mixed equilibria. *Games and economic behavior*, 5(3):320–367, 1993.

Gal, K. and Grosz, B. J. Multi-agent systems: Technical & ethical challenges of functioning in a mixed group. *Daedalus*, 151(2):114–126, 2022.

Ganzfried, S. Fictitious play outperforms counterfactual regret minimization. *arXiv preprint arXiv:2001.11165*, 2020.

Gao, Y., Lui, K. Y. C., and Hernandez-Leal, P. Robust risk-sensitive reinforcement learning agents for trading markets. *arXiv preprint arXiv:2107.08083*, 2021.

Goldberg, P. W., Savani, R., Sørensen, T. B., and Ventre, C. On the approximation performance of fictitious play in finite games. *International Journal of Game Theory*, 42 (4):1059–1083, 2013.

Harsanyi, J. C., Selten, R., et al. A general theory of equilibrium selection in games. *MIT Press Books*, 1, 1988.

Lanctot, M., Zambaldi, V., Gruslys, A., Lazaridou, A., Tuyls, K., Pérolat, J., Silver, D., and Graepel, T. A unified game-theoretic approach to multiagent reinforcement learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, pp. 4193–4206, 2017.

Leurent, E. An environment for autonomous driving decision-making. https://github.com/eleurent/highway-env, 2018.

Lianeas, T., Nikolova, E., and Stier-Moses, N. E. Risk-averse selfish routing. *Mathematics of Operations Research*, 44(1):38–57, 2019.

Markowitz, H. M. Foundations of portfolio theory. *The journal of finance*, 46(2):469–477, 1991.

McAleer, S., Lanier, J., Fox, R., and Baldi, P. Pipeline PSRO: A scalable approach for finding approximate nash equilibria in large games. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

McAleer, S., Lanier, J., Wang, K., Baldi, P., and Fox, R. XDO: A double oracle algorithm for extensive-form games. *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

McAleer, S., Farina, G., Lanctot, M., and Sandholm, T. Escher: Eschewing importance sampling in games by computing a history value function to estimate regret. *arXiv preprint arXiv:2206.04122*, 2022a.

McAleer, S., Lanier, J., Wang, K., Baldi, P., Fox, R., and Sandholm, T. Self-play psro: Toward optimal populations in two-player zero-sum games. *arXiv preprint arXiv:2207.06541*, 2022b.

McAleer, S., Wang, K., Lanctot, M., Lanier, J., Baldi, P., and Fox, R. Anytime optimal psro for two-player zero-sum games. *arXiv preprint arXiv:2201.07700*, 2022c.

McKelvey, R. D. and Palfrey, T. R. Quantal response equilibria for normal form games. *Games and economic behavior*, 10(1):6–38, 1995.

McMahan, H. B., Gordon, G. J., and Blum, A. Planning in the presence of cost functions controlled by an adversary. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pp. 536–543, 2003.

Meir, R. and Parkes, D. Congestion games with distance-based strict uncertainty. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 29, 2015.

Merton, R. C. An analytic derivation of the efficient portfolio frontier. *Journal of Financial and Quantitative Analysis*, 7(4):1851–1872, 1972. doi: 10.2307/2329621.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *nature*, 518(7540): 529–533, 2015.

Monderer, D. and Shapley, L. S. Fictitious play property for games with identical interests. *Journal of economic theory*, 68(1):258–265, 1996a.

Monderer, D. and Shapley, L. S. Potential games. *Games and economic behavior*, 14(1):124–143, 1996b.

Nash, J. Non-cooperative games. *Annals of mathematics*, pp. 286–295, 1951.

Nikolova, E. and Stier-Moses, N. Stochastic selfish routing. *ACM SIGecom Exchanges*, 11(1):21–25, 2012.

Nikolova, E. and Stier-Moses, N. E. A mean-risk model for the traffic assignment problem with stochastic travel times. *Operations Research*, 62(2):366–382, 2014.

Ordóñez, F. and Stier-Moses, N. E. Wardrop equilibria with risk-averse users. *Transportation Science*, 44(1):63–86, 2010.

Perez-Nieves, N., Yang, Y., Slumbers, O., Mguni, D. H., Wen, Y., and Wang, J. Modelling behavioural diversity for learning in open-ended games. In Meila, M. and Zhang, T. (eds.), *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 8514–8524. PMLR, 18–24 Jul 2021. URL https://proceedings.mlr.press/v139/perez-nieves21a.html.

Perolat, J., De Vylder, B., Hennes, D., Tarassov, E., Strub, F., de Boer, V., Muller, P., Connor, J. T., Burch, N., Anthony, T., et al. Mastering the game of stratego with model-free multiagent reinforcement learning. *Science*, 378(6623): 990–996, 2022.

Peysakhovich, A. and Lerer, A. Prosocial learning agents solve generalized stag hunts better than selfish ones. In *Proceedings of the 17th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 2043–2044, 2018.

Piliouras, G., Nikolova, E., and Shamma, J. S. Risk sensitivity of price of anarchy under uncertainty. *ACM Transactions on Economics and Computation (TEAC)*, 5(1):1–27, 2016.

Qiu, W., Wang, X., Yu, R., Wang, R., He, X., An, B., Obraztsova, S., and Rabinovich, Z. Rmix: Learning risk-sensitive policies for cooperative reinforcement learning agents. *Advances in Neural Information Processing Systems*, 34:23049–23062, 2021.

Raffin, A., Hill, A., Gleave, A., Kanervisto, A., Ernestus, M., and Dormann, N. Stable-baselines3: Reliable reinforcement learning implementations. *Journal of Machine Learning Research*, 22(268):1–8, 2021. URL http://jmlr.org/papers/v22/20-1364.html.

Robinson, J. An iterative method of solving a game. *Annals of Mathematics*, 54(2):296–301, 1951. ISSN 0003486X. URL http://www.jstor.org/stable/1969530.

Royset, J. O. Risk-adaptive approaches to learning and decision making: A survey. *arXiv preprint arXiv:2212.00856*, 2022.

Sokota, S., D'Orazio, R., Kolter, J. Z., Loizou, N., Lanctot, M., Mitliagkas, I., Brown, N., and Kroer, C. A unified approach to reinforcement learning, quantal response equilibria, and two-player zero-sum games. In *Deep Reinforcement Learning Workshop NeurIPS 2022*.

Wardrop, J. G. Road paper. some theoretical aspects of road traffic research. *Proceedings of the institution of civil engineers*, 1(3):325–362, 1952.

Wellman, M. P. Methods for empirical game-theoretic analysis. In *AAAI*, pp. 1552–1556, 2006.

Yekkehkhany, A., Murray, T., and Nagi, R. Risk-averse equilibrium for games, 2020.

Zhang, S., Liu, B., and Whiteson, S. Mean-variance policy iteration for risk-averse reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 10905–10913, 2021.

A. Full Proofs

A.1. Proposition 3.2 [Minimum Variance Solution]

The solution to the Eq. 5 provides the same solutions to the following:,

$$\boldsymbol{\sigma}^* \in \arg\min_{\boldsymbol{\sigma}} \boldsymbol{\sigma}^T \cdot \Sigma_{\mathbf{M}} \cdot \boldsymbol{\varsigma}$$
$$\text{s.t. } \boldsymbol{\sigma}^T \cdot \mathbf{M} \cdot \boldsymbol{\sigma} \geq \mu_{\mathsf{b}}$$
$$\sigma(a) \geq 0 \ \forall a \in A$$
$$\boldsymbol{\sigma}^T \mathbf{1} = 1$$

(8)

where $\mu_{\mathsf{b}} \in \mathbb{R}$ is the lowest level of expected return that the actor is willing to accept.

Proof. (Merton, 1972) shows by a Lagrange multiplier argument that the optimisation problem,

$$\boldsymbol{\sigma}^* \in \arg\min_{\boldsymbol{\sigma}} \boldsymbol{\sigma}^T \cdot \Sigma_{\mathbf{M}} \cdot \boldsymbol{\sigma}$$
$$\text{s.t. } \boldsymbol{\sigma}^T \cdot \mathbf{M} \cdot \boldsymbol{\varsigma} \geq \mu_{\mathsf{b}}$$
$$\sigma(a) \geq 0 \ \forall a \in A$$
$$\boldsymbol{\sigma}^T \mathbf{1} = 1$$

(9)

can be rewritten as

$$\boldsymbol{\sigma}^* \in \arg\min_{\boldsymbol{\sigma}} \boldsymbol{\sigma}^T \cdot \Sigma_{\mathbf{M}} \cdot \boldsymbol{\sigma} - \tau\left(\boldsymbol{\sigma}^T \cdot \mathbf{M} \cdot \boldsymbol{\varsigma}\right)$$
$$\text{s.t. } \sigma(a) \geq 0 \ \forall a \in A$$
$$\boldsymbol{\sigma}^T \mathbf{1} = 1$$

(10)

which can be equivalently expressed as,

$$\boldsymbol{\sigma}^* \in \arg\min_{\boldsymbol{\sigma}} -\left(\boldsymbol{\sigma}^T \cdot \mathbf{M} \cdot \boldsymbol{\varsigma} - \lambda\boldsymbol{\sigma}^T \cdot \Sigma_{\mathbf{M}} \cdot \boldsymbol{\sigma}\right)$$
$$\text{s.t. } \sigma(a) \geq 0 \ \forall a \in A$$
$$\boldsymbol{\sigma}^T \mathbf{1} = 1$$

(11)

where $\lambda = \frac{1}{\tau}$.

$\square$

A.2. Theorem 3.3 [RAE Existence]

For any finite N-player game where each player i has a finite k number of pure strategies, $A^i = \{a_1^i, ..., a_k^i\}$, an RAE exists

Proof. We base our proof on Kakutani's Fixed Point Theorem

Lemma (Kakutani Fixed Point Theorem). Let A be a non-empty subset of a finite dimensional Euclidean space. Let $f : A \rightrightarrows A$ be a correspondence, with $x \in A \longmapsto f(x) \subseteq A$, satisfying the following conditions:

1. A is a compact and convex set.

2. $f(x)$ is non-empty for all $x \in A$.

3. $f(x)$ is a convex-valued correspondence: for all $x \in A$, $f(x)$ is a convex set.

4. $f(x)$ has a closed graph: that is, if $\{x^n, y^n\} \to \{x, y\}$ with $y^n \in f(x^n)$, then $y \in f(x)$.

Then, f has a fixed point, that is, there exists some $x \in A$, such that $x \in f(x)$.

We define our best-response function as $B_i(\boldsymbol{\sigma}_{-i}) = \arg\max_{a \in \Delta_i} r^i(a, \boldsymbol{\sigma}_{-i})$ where u_i is defined as in Eq. (5) and by definition s must satisfy all of the properties of a proper mixed-strategy, and the best-response correspondence is $B : \Delta \rightrightarrows \Delta$ such that for all $\boldsymbol{\sigma} \in \Delta$, we have:

$$B(\boldsymbol{\sigma}) = [B_i(\boldsymbol{\sigma}_{-i})]_{i \in N} \tag{12}$$

We show that $B(\boldsymbol{\sigma})$ satisfies the conditions of Kakutani's Fixed Point Theorem

1. Δ *is compact, convex and non-empty.*

   By definition
   $$\Delta = \Pi_{i \in N} \Delta_i \tag{13}$$
   where each $\Delta_i = \{a | \sum_j a_j = 1\}$ is a simplex of dimension $|A^i| - 1$, thus each Δ_i is closed and bounded, and thus compact. Their product set is also compact.

2. $B(\boldsymbol{\sigma})$ *is non-empty.*

   By the definition of $B_i(\boldsymbol{\sigma}_{-i})$ where Δ_i is non-empty and compact, and r^i is a quadratic and hence a polynomial function in a. It is known that all polynomial functions are continuous, we can invoke Weirstrass's Extreme Value Theorem which states

   Lemma. If a real valued-function f is continuous on the closed interval $[a, b]$, then f must attain a maximum and a minimum, each at least once. That is, there exist numbers c and d in $[a, b]$ such that:
   $$f(c) \geq f(x) \geq f(d) \quad \forall x \in [a, b]$$

   Therefore, as Δ_i is non-empty and compact and r^i is continuous in a, $B_i(\boldsymbol{\sigma}_{-i})$ is non-empty, and therefore $B(\boldsymbol{\sigma})$ is also non-empty.

3. $B(\boldsymbol{\sigma})$ *is a convex-valued correspondence.*

   Equivalently, $B(\boldsymbol{\sigma}) \subset \Delta$ is convex if and only if $B_i(\boldsymbol{\sigma}_{-i})$ is convex for all i.

   In order to show that $B_i(\boldsymbol{\sigma}_{-i})$ is convex for all i, we instead show that the Quadratic Programme defined by Eq. (6) is a special case of convex optimisation under certain conditions, and therefore by definition has a feasible set which is a convex set.

   A *convex optimisation problem* is one of the form,

   $$\begin{aligned} \text{minimize} \quad & f_0(x) \\ \text{s.t.} \quad & f_i(x) < 0, \ i = 1, ..., m \\ & a_i^T x = b_i, \quad i = 1, ..., p \end{aligned} \tag{14}$$

   where $f_0, ..., f_m$ are convex functions. The requirements for a problem to be a convex optimisation problem are:

   (a) the objective function must be convex
   (b) the inequality constraint functions must be convex
   (c) the equality constraint functions $h_i(x) = a_i^T x = b_i$ must be affine

   We note that a quadratic form $\mathbf{x}^T \boldsymbol{A} \mathbf{x}$ is convex if $\boldsymbol{A}$ is positive semi-definite, and strictly convex if $\boldsymbol{A}$ is positive definite. In our constrained optimisation, the quadratic term $\boldsymbol{\sigma}^T \Sigma \boldsymbol{\sigma}$ is always guaranteed to be at least convex as Σ, the covariance matrix, is always at least PSD. Therefore, our objective function is convex. Additionally, it is easy to see that our inequality constraint functions are also convex and that our equality constraint function is affine. Therefore, our Quadratic Programme is an instance of a convex optimisation problem.

   Importantly, the feasible set of a convex optimisation problem is convex, since it is the intersection of the domain of the problem

$$\mathcal{D} = \bigcap_{i=0}^{m} \mathbf{dom} f_i, \tag{15}$$

which itself is a convex set.

Therefore, for all members of the feasible set $x, y \in B_i(\boldsymbol{\sigma}_{-i})$ and all $\theta \in [0, 1]$ we have that $\theta x + (1 - \theta)y \in S$ and we have a convex-valued correspondence.

4. $B(\boldsymbol{\sigma})$ *has a closed graph.*

Suppose to obtain a contradiction, that $B(\boldsymbol{\sigma})$ does not have a closed graph. Then, there exists a sequence $(\boldsymbol{\sigma}^n, \hat{\boldsymbol{\sigma}}^n) \to (\boldsymbol{\sigma}, \hat{\boldsymbol{\sigma}})$ with $\hat{\boldsymbol{\sigma}}^n \in B(\boldsymbol{\sigma}^n)$, but $\hat{\boldsymbol{\sigma}} \notin B(\boldsymbol{\sigma})$, i.e. there exists some i such that $\hat{\boldsymbol{\sigma}}_i \notin B_i(\boldsymbol{\sigma}_{-i})$. This implies that there exists some $\boldsymbol{\sigma}_i' \in \Delta_i$ and some $\epsilon > 0$ such that

$$r_i(\boldsymbol{\sigma}_i', \boldsymbol{\sigma}_{-i}) > r_i(\hat{\boldsymbol{\sigma}}_i, \boldsymbol{\sigma}_{-i}) + 3\epsilon. \tag{16}$$

By the continuity of r_i and the fact that $\boldsymbol{\sigma}_{-i}^n \to \boldsymbol{\sigma}_{-i}$, we have for sufficiently large n,

$$r_i(\boldsymbol{\sigma}_i', \boldsymbol{\sigma}_{-i}^n) \geq r_i(\boldsymbol{\sigma}_i', \boldsymbol{\sigma}_{-i}) - \epsilon. \tag{17}$$

and combining the preceding two relations we obtain

$$r_i(\boldsymbol{\sigma}_i', \boldsymbol{\sigma}_{-i}^n) > r_i(\hat{\boldsymbol{\sigma}}_i, \boldsymbol{\sigma}_{-i}) + 2\epsilon \geq r_i(\hat{\boldsymbol{\sigma}}_i^n, \boldsymbol{\sigma}_{-i}^n) + \epsilon \tag{18}$$

where the second relation follows from the continuity of r_i. This contradicts the assumption that $\hat{\boldsymbol{\sigma}}_i^n \in B(\boldsymbol{\sigma}_{-i}^n)$ and completes the proof.

Therefore, $B(\boldsymbol{\sigma})$ satisfies the conditions of Kakutani's Fixed Point Theorem, and therefore if $\boldsymbol{\sigma}^* \in B(\boldsymbol{\sigma}^*)$ then $\boldsymbol{\sigma}^*$ is an equilibrium. $\square$

A.3. Proposition 4.1 [SFP Convergence]

Replacing the best-response Eq. (6) with the best-response map Eq. (5) satisfies the conditions of $\bar{u}^i$ for a SFP process.

Proof. For the perturbed utility function $\bar{u}^i$ to be permissible in an SFP process, there exist two conditions:

1. That there exists a unique global solution to $\bar{u}^i$.

2. That the $\arg \max$ assigns strictly positive probability to all pure strategies.

We let $\bar{u}^i$ be replaced by the best-response map Eq. 5 and show that the 2 conditions noted above are met.

Condition 1 - To show that there exists a global solution to $\bar{u}^i$ we need to show that $\bar{u}^i$ is strictly concave which guarantees a unique global maximum.

As the EU term $u^i(\boldsymbol{\sigma}, Z_t^{-i}\mathbf{M})$ is linear, we therefore require that the perturbation term $v^i(\boldsymbol{\sigma})$ is strictly convex such that the perturbed utility function $\bar{u}^i = \arg \max_{\boldsymbol{\sigma}} u^i(\boldsymbol{\sigma}, Z_t^{-i}, \mathbf{M}) - \lambda v^i(\boldsymbol{\sigma})$ is strictly concave. In A.2 we have already shown that, as long as the covariance matrix $\boldsymbol{\Sigma}_{\mathbf{M}}$ is positive definite, then the quadratic term $\boldsymbol{\sigma}^T \cdot \boldsymbol{\Sigma}_{\mathbf{M}} \cdot \boldsymbol{\sigma}$ is strictly convex. By design, $\boldsymbol{\Sigma}_{\mathbf{M}}$ is strictly convex and therefore $\bar{u}^i$ is also strictly concave and therefore has a unique global maximum.

Condition 2 - By design the QP that solves Eq. 5 is constrained such that all pure strategies receive strictly positive probability (this is by design to induce mistakes in agents). Therefore, the output of the best-response map is always within $\text{int}(\Delta^i)$ and condition 2 is satisfied. We show that our utility measure can be embedded as a version of stochastic fictitious play and therefore can be used to find equilibrium in two-player zero-sum games and potential games.

$\square$

A.4. Proposition 4.2 [SFP is RAE]

Suppose the SFP sequence $\{Z_t\}$ converges to $\boldsymbol{\sigma}$ in the observed strategy sense [2], then $\boldsymbol{\sigma}$ is a Risk-Averse equilibrium.

Proof. Assume the observed strategy has converged to $\boldsymbol{\sigma} = (\boldsymbol{\sigma}^1, \boldsymbol{\sigma}^2)$ and that the strategy is not an RAE. This implies there exists some $\boldsymbol{\sigma}^{i,\prime}$ such that:

$$r^i(\boldsymbol{\sigma}^{i,\prime}, \boldsymbol{\sigma}^{-i}) > r^i(\boldsymbol{\sigma}^i, \boldsymbol{\sigma}^{-i}) \tag{19}$$

However, because $\boldsymbol{\sigma}$ has converged then the SFP sequence $\{Z_t\}$ will also converge such that $\lim_{t\to\infty} Z_t = \boldsymbol{\sigma}$ and because we are in an SFP process it must be the case that:

$$r^i(\boldsymbol{\sigma}^i, \boldsymbol{\sigma}^{-i}) > r^i(\boldsymbol{\sigma}^{i,\prime}, \boldsymbol{\sigma}^{-i}) \quad \forall \boldsymbol{\sigma}^{i,\prime} \in \Delta^i \tag{20}$$

and therefore $\boldsymbol{\sigma}^{i,\prime}$ can not be a best response to $\boldsymbol{\sigma}^{-i}$.

$\square$

---

[2]Convergence in the time-average Z_t does not imply convergence in the actual strategy taken at each t, but may for example imply cyclic actual behaviour that results in average behaviour converging.
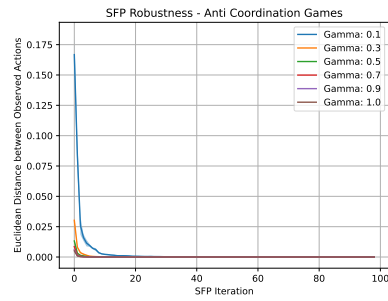
B. SFP Robustness

Figure 6. Euclidean distance between observed actions after each iteration on randomly generated anti-coordination games. A distance of 0 implies that the process has converged.
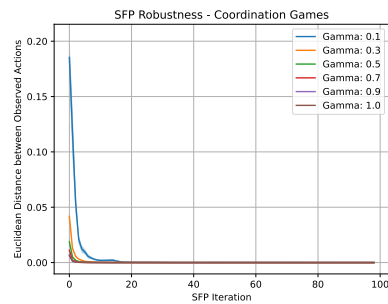

Figure 7. Euclidean distance between observed actions after each iteration on randomly generated coordination games. A distance of 0 implies that the process has converged.
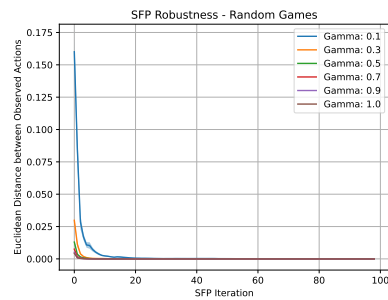

Figure 8. Euclidean distance between observed actions after each iteration on randomly generated games. A distance of 0 implies that the process has converged.

C. γ Generalisation
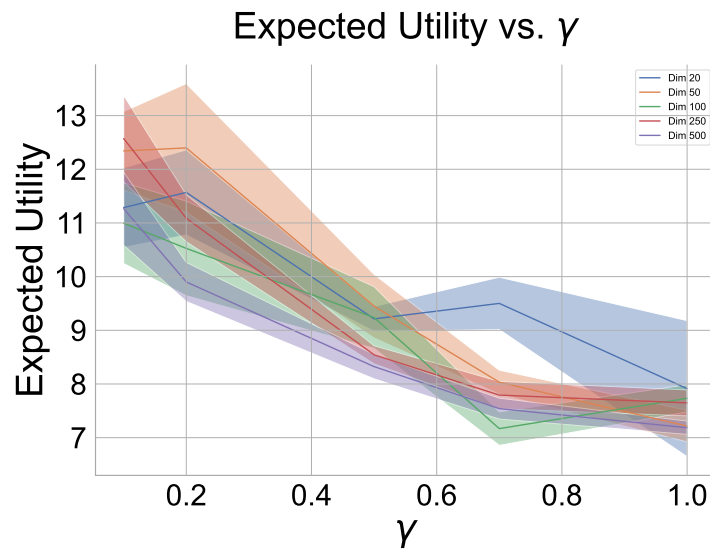
Expected Utility vs. γ

Figure 9. Expected utility values for different γ values. Each line represents a different game dimension, and is intended to show largely similar values in terms of expected utility for a given γ across the dimensions.
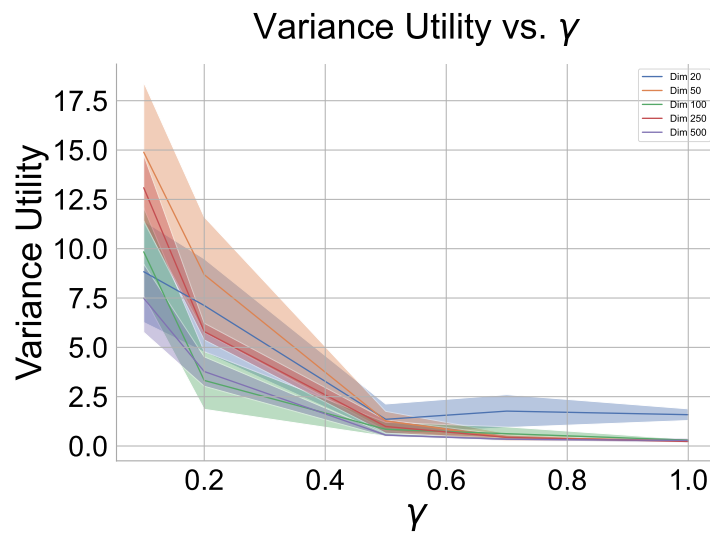
Variance Utility vs. γ

Figure 10. Variance utility values for different γ values. Each line represents a different game dimension, and is intended to show largely similar values in terms of variance utility for a given γ across the dimensions.

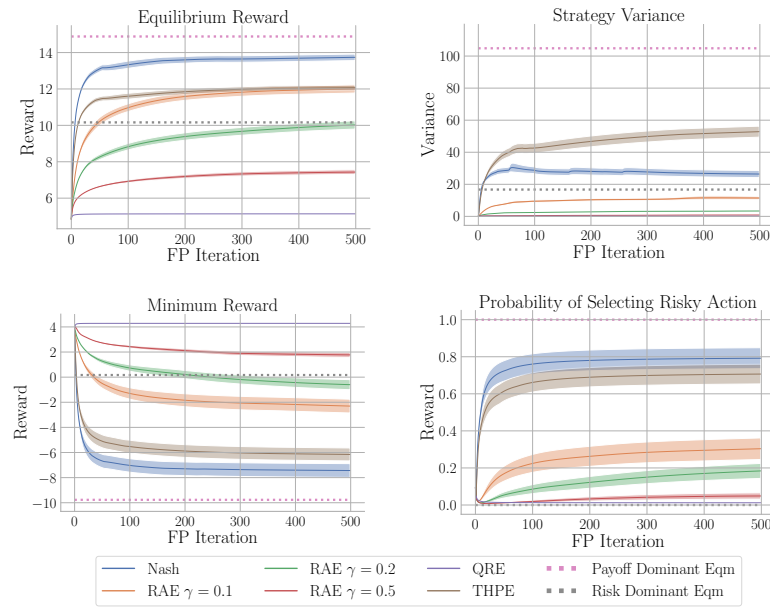## D. Figure 3 Training Curves

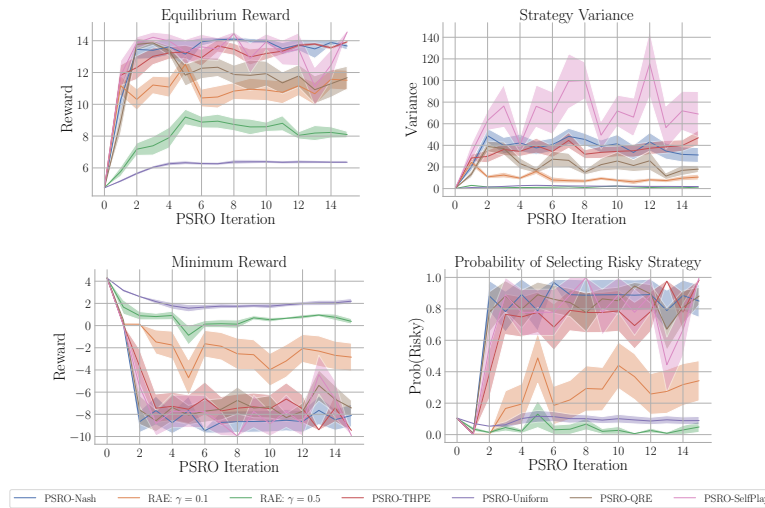*Figure 11.* Training curves over multiple seeds for Fig. 3.

Figure 12. Training curves over multiple seeds for Fig. 3.

E. Large-Scale Driving Results

A) Intra-Distribution Results

	Eqm Reward	Eqm Variance	Worst-Case	Num. Crashes	Num. Arrivals
Self-Play	3.47 ± 0.08	1.20 ± 0.21	-2.05 ± 0.88	50.0 ± 0.00	0.00 ± 0.00
PSRO-Uniform	3.60 ± 0.01	1.14 ± 0.06	1.49 ± 0.17	50.0 ± 0.00	0.00 ± 0.00
PSRO-THPE	3.61 ± 0.13	1.11 ± 0.18	0.99 ± 0.83	50.0 ± 0.00	0.00 ± 0.00
PSRO-QRE	3.65 ± 0.22	0.96 ± 0.23	1.78 ± 0.50	50.0 ± 0.00	0.00 ± 0.00
PSRO-Nash	3.76 ± 0.07	1.46 ± 0.08	1.93 ± 0.03	50.0 ± 0.00	0.00 ± 0.00
PSRO-RAE	**7.14 ± 0.39**	**3.27 ± 0.54**	**-3.48 ± 2.35**	**7.00 ± 2.00**	**43.00 ± 2.00**

Figure 13. Results on large autonomous driving scenario, on 50 episodes over 5 seeds for intra-distribution testing, e.g. both agents controlled by PSRO-RAE.

F. Pseudo-code

Algorithm 1 SFP

1: **Initialise:** Payoff Matrix $\mathbf{M}$, uniform initial time-average strategy Z_0.
2: **for** iteration $t \in \{1, 2, ...\}$ **do**:
3:      Find best-response $\boldsymbol{\sigma}_t$ to Z_t via Eq. 5.
4:      Update time-average strategy Z_t with respect to $\boldsymbol{\sigma}_t$.
5: **Return:** Final time-average strategy Z_T.

Algorithm 2 PSRO-RAE

1: **Initialise:** the policy set $\boldsymbol{\Phi} = \prod_{i \in \mathcal{N}} \boldsymbol{\Phi}^i$, meta-game $\mathbf{M}_0$, co-variance matrix $\boldsymbol{\Sigma}_{\mathbf{M}_T}$
2: **for** iteration $t \in \{1, 2, ...\}$ **do**:
3:      **for** each player $i \in \mathcal{N}$ **do**:
4:          Compute meta-policy $\boldsymbol{\sigma}_t$ by SFP (Alg. 1).
5:          Find new policy by Oracle: $\phi_t^i = \mathcal{O}^i(\boldsymbol{\sigma}_t)$.
6:          Expand $\boldsymbol{\Phi}_{t+1}^i \leftarrow \boldsymbol{\Phi}_t^i \cup \{\phi_t^i\}$.
7:          Update meta-payoff $\mathbf{M}_{t+1}$, co-variance matrix $\boldsymbol{\Sigma}_{\mathbf{M}_{t+1}}$.
8: **Return:** $\boldsymbol{\sigma}_T$ and $\boldsymbol{\Phi}_T$.

G. Hyperparameter Settings

Table 1. Hyper-parameter settings for our experiments.

SETTINGS	VALUE	DESCRIPTION
SFP COORDINATION GAMES		
ACTION DIMENSION	100	NUMBER OF PURE STRATEGIES AVAILABLE
FP ITERATIONS	100	NUMBER OF FP BELIEF UPDATES
TREMBLE PROBABILITY	0.001	PROBABILITY OF TREMBLING TO ANOTHER STRATEGY
QUANTAL TYPE	SOFTMAX	TYPE OF QUANTAL RESPONSE EQUILIBRIUM
# OF SEEDS	50	# TRIALS
PSRO NFG COORDINATION GAMES		
ORACLE METHOD	REINFORCE	SUBROUTINE OF GETTING ORACLES
PSRO ITERATIONS	15	NUMBER OF PSRO ITERATIONS
ACTION DIMENSION	500	NUMBER OF PURE STRATEGIES AVAILABLE
LEARNING RATE	0.005	ORACLE LEARNING RATE
ORACLE EPOCHS	2000	ORACLE TOTAL EPOCHS
ORACLE EPOCH TIMESTEPS	100	TIMESTEPS PER ORACLE EPOCH
RAE GAMMA	0.1, 0.5	VARIANCE AVERSION PARAMETER
METASOLVER	RAE SFP	METASOLVER METHOD
METASOLVER ITERATIONS	100	METASOLVER ITERATIONS
# OF SEEDS	20	# OF TRIALS
STAG-HUNT GRID-WORLD		
ORACLE METHOD	MV-PPO (ZHANG ET AL., 2021)	SUBROUTINE OF GETTING ORACLES
PSRO ITERATIONS	10	NUMBER OF PSRO ITERATIONS
GORE COST	2	COST FOR GETTING CAUGHT BY STAG
PPO HYPERPARAMS	DEFAULT SB3 (RAFFIN ET AL., 2021)	PPO HYPERPARAMETER VALUES
MV-PPO VARIANCE AVERSION	0.15	PPO VARIANCE AVERSION PARAMETER
RAE GAMMA	0.15	VARIANCE AVERSION PARAMETER
METASOLVER	RAE SFP	METASOLVER METHOD
METASOLVER ITERATIONS	100	METASOLVER ITERATIONS
# OF SEEDS	5	# OF TRIALS
TWO-WAY ENVIRONMENT		
ORACLE METHOD	MV-PPO (ZHANG ET AL., 2021)	SUBROUTINE OF GETTING ORACLES
PSRO ITERATIONS	7	NUMBER OF PSRO ITERATIONS
PPO HYPERPARAMS	DEFAULT SB3 (RAFFIN ET AL., 2021)	PPO HYPERPARAMETER VALUES
MV-PPO VARIANCE AVERSION	0.5	PPO VARIANCE AVERSION PARAMETER
RAE GAMMA	0.5	VARIANCE AVERSION PARAMETER
METASOLVER	RAE SFP	METASOLVER METHOD
METASOLVER ITERATIONS	100	METASOLVER ITERATIONS
# OF SEEDS	5	# OF TRIALS

H. Environments

H.1. Randomly Generated NFGs

Figure 14. An example of a 3 action game. Action S_3 (dotted outline) provides a high return assuming successful coordination but high variance in case the opponent does not coordinate correctly.

H.1.1. CHARACTERISTICS

1. High risk high-reward actions. In these strategies if both players play the same action then they receive a high payoff, however if a player takes a different action then a big negative payoff is received. For example, in Fig. 14, S_3 is the high-risk high-reward strategy.

2. Low risk low-reward actions. In these strategies if both players play the same action then they receive a lower payoff, however if a player takes a different action then a big smaller payoff is received. For example, in Fig. 14, S_1 and S_2 are the low-risk low-reward strategies.

H.1.2. GENERATON

We randomly generate coordination games with N actions in the following way:

Algorithm 3 Iterative RAE Generator

1: **Initialise:** Empty $N \times N$ payoff matrix P
2: **for** each action i **do**:
3:      Sample coordination element, $p_{ii} \sim \mathcal{U}(5, 15)$
4:      Set Payoff matrix element $P_{ii} = |p_{ii}|$
5:      **if** $P(X \leq p_{ii}) > 0.9$ **do**
6:          **for** all other actions j **do**
7:              Sample anti-coordination element $p_{ij} \sim \mathcal{U}(-10, 15)$
8:              Set Payoff matrix element $P_{ij} = P_{ji} = p_{ij}$
9:      **else do**
10:         **for** all other actions j **do**
11:             Sample anti-coordination element $p_{ij} \sim \mathcal{U}(0, 10)$
12:             Set Payoff matrix element $P_{ij} = P_{ji} = p_{ij}$
13: **Return:** P.

H.2. Stag Hunt Grid World

Our stag-hunt environment is taken from (Peysakhovich & Lerer, 2018) with minor adjustments to the parameters of the game.

H.2.1. CHARACTERISTICS

Details - 2 Players, 1 Stag, 2 Plants. All spawned in random positions dependent on the seed.

State Space - 5×5 grid-world. Grid spots are marked as: 0 if nothing on them, 1 if Player is on them, 2 if Stag is on them, 3 if a Plant is on them. Players see the full grid-world with the state space being $\mathcal{S} \in \{0, 1, 2, 3\}^{5 \times 5}$.

Action Space - Actions involve movement in the grid-world in the four cardinal directions. $\mathcal{A} = \{\text{left, right, up, down}\}$.

Reward Space - There are 4 different rewards signals in the game:

1. If a Player moves over a Plant they get $r = 2$, and the Plant respawns elsewhere.

2. If both Players move over the Stag at the same time both receive $r = 5$ and the Stag respawns elsewhere.

3. If a Player moves over the Stag on their own, or the Stag moves over them on their own, the Player receives $r = -2$ and the Stag respawns elsewhere on the grid.

4. Otherwise $r = 0$.

The Stag - At each time-step t, the Stag will take one grid-step in the four cardinal directions towards the Player that is closest to it.

H.3. Autonomous Driving Environment

Our driving environment is based on the two-way environment from (Leurent, 2018) where we make modifications to the reward function to introduce a larger factor of risk-aversion into the game. The goal of the controlled drivers is to reach the end of the road (the destination) whilst avoiding crashing and coming into too close contact with other vehicles. Slow moving drivers populate the roads moving at a constant speed of 20.

H.3.1. CHARACTERISTICS

Details - 2 Players heading opposite directions, 6 other cars on road with even split heading each direction. All spawned in random positions dependent on the seed.

State Space - For the state space we use the KinematicObservation in Leurent 2018. The KinematicObservation is a $V \times F$ array that describes a list of V nearby vehicles by a set of features of size F. We use the default feature set in Leurent 2018, which is the x and y coordinate of the V nearby vehicles and their velocities in the x and y direction.

Action Space - For the action space we use the DiscreteAction in Leurent 2018. The DiscreteAction is a uniform quantization of the ContinuousAction which allows the agent to directly set the vehicle kinematics, by controlling the throttle a and the steering angle δ.

Reward Space - There are 4 reward signals in the environment:

1. If the car crashes $r = -2$.

2. If the car arrives at the destination $r = 2$.

3. If the car is travelling at a good speed ([25,30]), $r = 0.2$.

4. If the car comes very close to another car $r = -0.1$

5. Otherwise $r = 0$.

Other Cars - Non-Player cars on the road travel at a constant speed of $v = 20$ and do not change direction. On each road there are 3 spawned NPC cars ahead of the Player cars.

I. Baselines

For NFG tasks we use the baselines: NE (including risk/dominant payoff NE), THPE and QRE introduced in Sec. 2. For MDP tasks we use the baselines: PSRO-{Nash, Uniform, Self-Play, THPE, QRE}. where the brackets refer to the meta-solver used. In the PSRO setting we limit our baselines to algorithms that operate within this framework, and to not consider non-population risk-aversion algorithms (as is standard in PSRO literature).

THPE and QRE - For SFP settings, we solve for these equilibria using Fictitious Play by replacing the best-responses with a trembling hand best-response and a Logit Quantal best-response. Empirically, we clarify that the time-average does converge using FP and therefore we do have a THPE and QRE in the games where we used FP (i.e. smaller games)

For PSRO settings, we treated the new agent best-response for THPE and QRE as vanilla PSRO optimisation agents, i.e. in terms of expected reward. Therefore, it is likely that these baselines are only approximations of the true THPE and QRE, however the lack of a notable solver in these games is an obvious downside of these equilibria.

Nash - We use vanilla Fictitious Play for any Nash equilibrium solving on NFGs.

Uniform - All actions / policies receive equal mixed-strategy probability.

Self-Play - Only the most recent policy in a populations equals probability in the mixed-strategy, i.e. a pure strategy.

J. Compute Architecture

All experiments run on one machine with:

- AMD Ryzen Threadripper 3960X 24 Core

- 1 x NVIDIA GeForce RTX 3090

K. Asymmetric Formulations

In the following section we will show the formulation of Sec. 3 but for the asymmetric case.

K.1. RAE Derivation

Define the utility for player i of action $a_k^i \in \mathcal{A}^i$ against action $a_{k'}^j \in \mathcal{A}^j$ as $u^i(a_k^i, a_{k'}^j)$ and the full utility matrix as $\mathbf{M}^i$, where the entry $\mathbf{M}_{k,k'}^i$ refers to $u^i(a_k^i, a_{k'}^j)$ and $\mathbf{M}_k^i$ refers to $u^i(a_k^i, a_{k'}^j) \; \forall k'$, i.e. the vector of utilities that action a_k^i receives against all other actions. We now define the *expected* utility of the mixed-strategy for player 1 $\boldsymbol{\sigma}^i$ versus the mixed strategy for player 2 $\boldsymbol{\varsigma}^j$ as

$$u^i(\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j, \mathbf{M}^i) = \sum_{a_k^i \in A^i} \sum_{a_{k'}^j \in A^j} \sigma(a_k^i)\varsigma(a_{k'}^j)u^i(a_k^i, a_{k'}^j)$$
$$= \boldsymbol{\sigma}^{i,T} \cdot \mathbf{M}^i \cdot \boldsymbol{\varsigma}^j. \tag{21}$$

The weighted co-variance matrix for $\mathbf{M}^i$ (i.e. the variance of utility values) is a $|A^i| \times |A^i|$ matrix $\boldsymbol{\Sigma}_{\mathbf{M}^i, \boldsymbol{\varsigma}^j} = [c_{k,k'}^i]$ with entries

$$c_{k,k'}^i = \sum_{a_d^j \in A^j} \varsigma^j(a_d^j)\big(u^i(a_d^i, a_k^j) - \bar{\mathbf{M}}_k^i\big)\big(u^i(a_d^i, a_{k'}^j) - \bar{\mathbf{M}}_{k'}^i\big), \tag{22}$$

where $\bar{\mathbf{M}}_k^i = \sum_{k'=1}^{|A^j|} \varsigma(a_{k'}^j)u(a_k^i, a_{k'}^j)$ is the EU for Player i's action k given the opponent mixed-strategy $\boldsymbol{\varsigma}^j$. This allows us to define the mixed-strategy $\boldsymbol{\sigma}^i$ variance utility as:

$$\mathrm{Var}(\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j, \mathbf{M}^i) = \sum_{k=1}^{|A^i|} \sum_{k'=1}^{|A^j|} \sigma(a_k^i)\sigma(a_{k'}^j)c_{k,k'}$$
$$= \boldsymbol{\sigma}^{i,T} \cdot \boldsymbol{\Sigma}_{\mathbf{M}^i, \boldsymbol{\varsigma}^j} \cdot \boldsymbol{\sigma}^i. \tag{23}$$

The final utility function r^i which considers expected and variance utility for mixed-strategy $\boldsymbol{\sigma}^i$ is,

$$r(\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j, \mathbf{M}^i) = u^i(\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j, \mathbf{M}^i) - \gamma^i \,\mathrm{Var}(\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j, \mathbf{M}^i), \tag{24}$$

where $\gamma^i \in \mathbb{R}$ is the risk-aversion parameter.

In order to define RAE, we first define the best-response map:

$$\boldsymbol{\sigma}^{*,}(\boldsymbol{\varsigma}^j) \in \arg\max_{\boldsymbol{\sigma}^i} r^i(\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j, \mathbf{M}^i)$$
$$\text{s.t. } \sigma(a) \geq \epsilon \,, \forall a \in A^i$$
$$\boldsymbol{\sigma}^{i,T}\mathbf{1} = 1, \tag{25}$$

Definition K.1 (RAE). A strategy profile $\{\boldsymbol{\sigma}^i, \boldsymbol{\varsigma}^j\}$ is a risk-averse equilibrium if both $\boldsymbol{\sigma}^i$ and $\boldsymbol{\varsigma}^j$ are risk-averse best responses, in that they satisfy Eq. 25, to each other.

K.2. Multi-population PSRO-RAE

K.2.1. PSRO OUTER LOOP

At every iteration $t \leq T$, a *player* i is defined by a population of fixed *agents* $\boldsymbol{\Phi}_t^i = \boldsymbol{\Phi}_0^i \cup \{\phi_1^i, \phi_2^i, ..., \phi_t^i\}$, where $\boldsymbol{\Phi}_0^i$ is the initial random agent.

K.2.2. PSRO INNER LOOP

a, d) Meta-Game & Covariance Matrix At the start of the iteration t inner loop, each player i has population with a *meta-game* $\mathbf{M}_t^i$, an EU matrix between all the *agents* in its own population and that of the opponent j, with individual entries

$M^i(\phi_k^i, \phi_{k'}^j) \; \forall \phi_k^i \in \Phi_t^i, \phi_{k'}^j \in \Phi_t^j$. In addition, each player i with population also generates a covariance matrix $\Sigma_{\mathbf{M}_t^i}$ defined by Eq. 2. At the end of iteration t inner loop, both $\mathbf{M}_t^i$ and $\Sigma_{\mathbf{M}_t^i}$ are updated to include a new agent.

b) Meta Distribution To use Φ_t^i we require a way to select which $\phi_t^i \in \Phi_t^i$ will be used as training opponents. The function f^i is a mapping $f^i : \mathbf{M}_t^i \to [0,1]^t$ which takes as input a meta-game $\mathbf{M}_t^i$ and outputs a *meta-distribution* $\boldsymbol{\sigma}_t^i = f^i(\mathbf{M}_t^i)$. The output $\boldsymbol{\sigma}_t^i$ is a probability assignment to each *agent* in the population Φ_t^i which is the equivalent of a mixed-strategy in a NFG, except actions are now RL policies. We apply RAE (Def. 3.1) as the meta-solver. As ϕ^i are RL policies then the policies are sampled by their respective probability in $\boldsymbol{\sigma}_t^i$.

c) Best Response Oracle At each epoch Φ_t^i is augmented with a new agent that is a *best-response* (BR) to the meta-distribution $\boldsymbol{\sigma}_t^i$. When selecting the BR oracle one aims to optimise the same objective function of that optimised by the meta-distribution. For example, in Vanilla PSRO the Nash meta-distribution optimises for environment reward and the BR oracle also optimises a new agent in terms of environment reward. This can be found with any optimisation process such as RL or an evolutionary algorithm. In our setting the meta-distribution optimises two metrics, the EU and the variance utility and therefore we similarly need a BR oracles that optimises the same dual objective.

In terms of RL quantities, this translates to maximising the expected total-RL reward (i.e. the total of the per-step rewards) whilst minimising the variance of the total-RL reward caused by the sampling of different RL agents from $\boldsymbol{\sigma}_t^i$.

To achieve this, we follow the approach of (Zhang et al., 2021) that optimises both the total RL-reward and per-step RL-reward variance by solving an augmented MDP where the per-step reward g_t^i is replaced by:

$$\hat{g}_t^i = g_t^i - \lambda(g_t^i)^2 + (2\lambda g_t^i y_i)$$

where $y_i = \frac{1}{T}\sum_{t=1}^T g_t^i$ is the average of the per-step rewards during the data collection phase.

We choose the per-step RL reward as it is an upper bound of the total-RL reward variance (Bisi et al., 2020), therefore reducing per-step variance will also reduce total variance, and is more effective computationally (Zhang et al., 2021). Additionally, as this variance is also with respect to the sampling probability defined by $\boldsymbol{\sigma}_t^i$ this optimises the correct co-variance matrix which is also weighted by $\boldsymbol{\sigma}_t^i$.

L. QRE Failure Case

In the following section we present results on the two-action driving game described in Sec. 1 of the main article and displayed in Fig. 15.
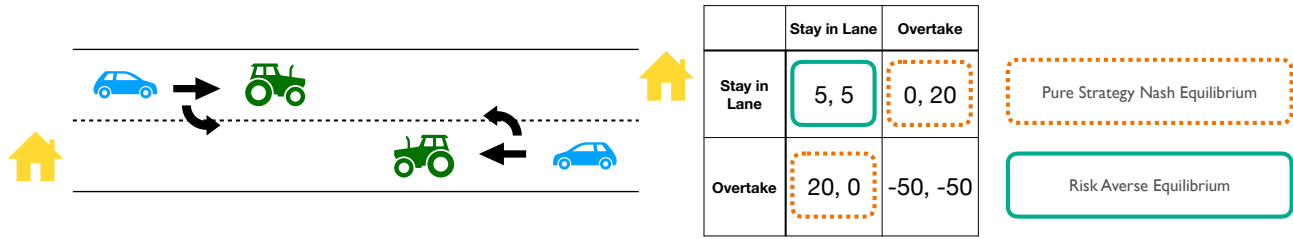

Figure 15. Two-action driving risk game.

We specifically utilise this game to show a failure case of QRE as a risk-sensitive solution. Ideally, a risk-sensitive solution concept would only play the Stay in Lane strategy as the Overtake strategy has far too high potential downside risk.
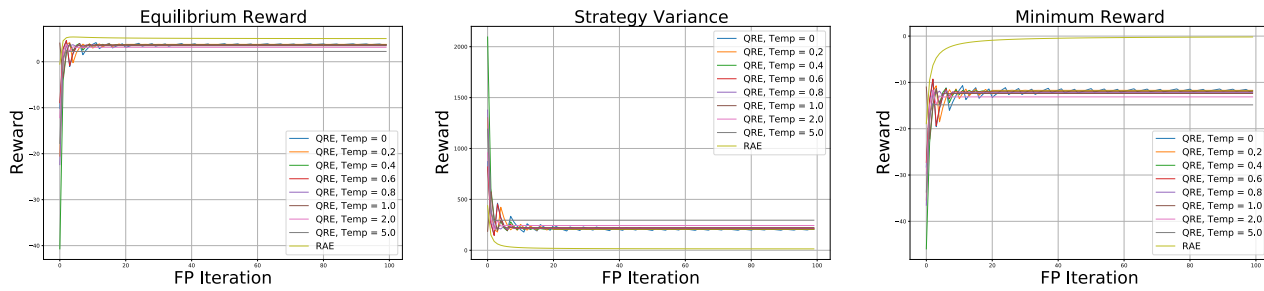

Figure 16. QRE and RAE results on two-action driving game.

As can be seen from the results in Fig. 6, for a large sample of QRE hyperparameters the equilibrium found is high variance with potential poor downside performance. We believe this is because the very large costs of the errors are easily picked up by variance analysis, but not so easily by the setup of QRE.

M. Variance vs. Standard Deviation

It is worth noting that a property of variance is that it is not scale-invariant with relation to the utilities of the game, and one might suggest using standard deviation (STD) instead as it is scale-invariant. However, we choose to stick with variance due to its mathematical properties, notably that because it is quadratic with respect to the strategy probabilities σ QP can be used, whereas STD is not quadratic w.r.t σ and QP cannot be used.