Comparing utility-based and Pareto-based solution sets in multi-objective normal form games

Patrick Mannion[a;*] **and Roxana Rădulescu**[b;**]

[a]School of Computer Science, University of Galway, Ireland
[b]Artificial Intelligence Lab, Vrije Universiteit Brussel, Belgium

Abstract. A multi-objective multi-agent system (MOMAS) is a flexible group decision making formalism, allowing one to model decision making processes where multiple actors must consider the trade-offs between conflicting objectives. Previous works have advocated for a utility-based approach to MOMAS, where each actor makes decisions in accordance with its own utility function - a function which specifies preferences over objectives. By contrast, many other works in the multi-objective decision making literature adopt the axiomatic or Pareto-based approach, where a set of non-dominated tradeoff solutions called the Pareto optimal set is derived. How exactly utility-based and Pareto-based solution sets relate to each other in MOMAS settings is currently not well understood. In this paper, we use the framework of multi-objective normal form games (MONFGs) to explore the relationship between the solution sets generated by the two approaches. We attempt for the first time in the context of MOMAS to quantify the degree of alignment between individual agent utility functions. We also demonstrate for the first time that situations can exist where none of the Nash equilibria in a MOMAS for a given set of utility functions are Pareto optimal.

1 Introduction

Many decision making processes in the real world involve *multiple agents* making decisions in a distributed manner. Multi-agent systems (MASs) have been used to model a wide variety of domains, including urban and air traffic control [8, 20], autonomous vehicles [7], and energy systems [6], to give a few examples.

A key aspect of many real world applications, including all of the above domains, is that they feature *multiple objectives* that must be optimised simultaneously. The field of multi-objective decision making (MODeM) covers a broad range of approaches that explicitly consider the possible trade-offs between objectives when computing solutions to a problem. Examples include game theoretic approaches [5, 14], as well as algorithms based on reinforcement learning (RL) [11, 17] or planning [3, 4].

The intersection of these two fields, MAS and MODeM, is known as multi-objective multi-agent decision making (MOMADM). Research into MOMADM is still at an early stage, and many open questions remain in the field, as highlighted in a recent survey [15]. Research interest in the field has increased over the years, and more researchers are now exploring the possibilities offered by multi-objective multi-agent systems (MOMAS).

Approaches to computing solutions for MODeM problems generally fall into one of two categories, the *utility-based approach* or the *axiomatic* or *Pareto-based approach* [4]. In the utility-based approach, it is assumed that a decision maker has a utility function that can be used to compute a scalar utility value that represents the decision maker's preference for a given multi-objective vector, thus allowing a total ordering over all possible multi-objective vectors to be computed. By contrast, the axiomatic or Pareto-based approach seeks to compute the Pareto optimal set (or a close approximation of it), and the Pareto optimal set is then presented to the decision maker to allow an appropriate solution to be selected.

In single agent settings where there is only one decision maker, the relationship between the utility-based and Pareto-based approaches is straightforward; if one can compute the true Pareto optimal set for the problem, then it is guaranteed that there will be a solution in this set that maximises the utility of the user for any possible utility function[1].

The relationship between the utility-based and Pareto based approaches becomes much more complex when multiple agents are involved. In MAS, often the desired outcome is to have a stable solution, e.g., a Nash equilibrium [9]. However, it is known from previous research that Nash equilibria do not always exist in MOMAS [16]. When following the utility-based approach, whether a stable solution exists in a MOMAS depends on the particular utility functions being used by each agent in the system. A further complication is that one could take either a system-wide or individual agent perspective when computing Pareto optimal sets in MOMAS.

Little is currently known about the relationship between solution sets derived using the utility-based and Pareto-based approaches in MOMAS, so in this work we present the first investigation into their relationship. We use the framework of multi-objective normal form games (MONFGs) for our analysis and experiments, and introduce a new MONFG specifically to aid our analysis. We also study for the first time in a MOMAS context a non-linear utility function that that can be parameterised with a set of weights or preferences over objectives, as well as attempting for the first time to quantify the degree of alignment between individual agent utility functions in a MOMAS.

Our analysis demonstrates that the specific preferences expressed by individual agents' utility functions have a considerable influence on the range of possible joint strategies that can be Nash equilibria in the system. We also find that for certain combinations of preferences,

* Equal contribution. Email: patrick.mannion@universityofgalway.ie
** Equal contribution. Email: roxana.radulescu@vub.be

[1] Under the very minimal assumption that the utility function is monotonically increasing, see Section 2.1.2.

many Nash equilibria that are not Pareto optimal can appear, which may decrease the probability that learning algorithms converge to Pareto optimal outcomes. Our work demonstrates for the first time that situations can exist where none of the Nash equilibria are contained in the system-wide Pareto optimal set of solutions, implying that the utility-based and Pareto-based approaches can give vastly different sets of solutions for the same problem.

Our results provide the first empirical evidence in support of the arguments in prior works (e.g., [13, 15, 16]), that a utility-based approach should be adopted in preference to a Pareto-based approach when computing solutions in MOMAS, as using the Pareto-based approach alone may make it impossible to find a Nash equilibrium, which is one of the main solution concepts that designers of MAS typically aim to achieve.

2 Background

Here we cover the necessary background to understand the results later in the paper. For a more comprehensive view, we recommend recent surveys on MOMADM [15] and on multi-objective RL [2].

2.1 Multi-Objective Normal Form Games

Definition 2.1 (Multi-objective normal-form game) *An n-person finite multi-objective normal-form game G is a tuple $(N, \mathcal{A}, \mathbf{p})$, with $n \geq 2$ and $d \geq 2$ objectives, where:*

- *$N = \{1, \ldots, n\}$ is a finite set of agents.*
- *$\mathcal{A} = A_1 \times \cdots \times A_n$, where A_i is the finite action set of agent i (i.e., the pure strategies of i). An action (pure strategy) profile is a vector $\mathbf{a} = (a_1, \ldots, a_n) \in \mathcal{A}$.*
- *$\mathbf{p} = (\mathbf{p}_1, \ldots, \mathbf{p}_n)$, where $\mathbf{p}_i \colon \mathcal{A} \to \mathbb{R}^d$ is the vectorial payoff of agent i, given an action profile.*

We can define the set of mixed strategies of player i as the probability distribution over their set of actions: $\Pi_i = P(A_i)$. A mixed-strategy profile is then the Cartesian product of all the individual mixed-strategy sets $\Pi = \Pi_1 \times \ldots \times \Pi_n$. The expected payoff of player i, under a mixed-strategy profile $\boldsymbol{\pi} \in \Pi$ is defined as:

$$\mathbf{p}_i^{\boldsymbol{\pi}} = \mathop{\mathbb{E}}_{\mathbf{a}\sim\boldsymbol{\pi}} \mathbf{p}_i(\mathbf{a}) = \sum_{\mathbf{a}\in\mathcal{A}} \mathbf{p}_i(\mathbf{a}) \prod_{j=1}^{n} \pi_j(a_j) \tag{1}$$

When following the utility-based approach [2], each player has a private (potentially unknown) utility function u_i that represents their preferences over the objective values. Utility functions will be discussed in more detail in Section 2.1.2.

In this paper, for ease of analysis, we only consider MONFGs with two players, two actions and two objectives. We also confine our study to settings where both agents receive the same payoff vector after the game has been played, i.e., $\mathbf{p}_1 = \mathbf{p}_2$. Both agents however have their own individual utility function that represents their preferences over the objectives. When agents receive the same payoffs but have their own utility function this is known as the *team reward individual utility* (TRIU) setting[2] [15].

2.1.1 Optimisation Criteria

To calculate the utility of a payoff vector, there are two choices [2]. One option is to compute the expected value of the payoffs of a joint strategy first and then apply the utility function, leading to the *scalarised expected returns* (SER) optimisation criterion:

$$p_i^u = u(\mathbb{E}[\mathbf{p}_i^{\boldsymbol{\pi}}]) \tag{2}$$

where π is the joint strategy for all the agents in a MONFG.

The second option is to apply the utility function before computing the expectation, leading to the *expected scalarised returns* (ESR) optimisation criterion:

$$p_i^u = \mathbb{E}[u(\mathbf{p}_i^{\boldsymbol{\pi}})]. \tag{3}$$

Which of these optimisation criteria is most appropriate depends on how the agents will interact. SER is the correct criterion if the decision making process will be repeated multiple times and the utility will be calculated based on the expected return vector, whereas ESR is more appropriate if only a single decision will be executed [16]. As we are interested in studying repeated interactions, we opt for the SER criterion in this work.

2.1.2 Utility Functions

As is common in the MODeM literature [2, 15], we make the minimal assumption that the utility functions being used by the agents are monotonically increasing. Formally, a utility function is monotonically increasing if:

$$(\forall o : p_{i,o} \geq p'_{i,o}) \Rightarrow u(\mathbf{p}_i) \geq u(\mathbf{p}'_i) \tag{4}$$

where $p_{i,o}$ is the payoff value for agent i on objective o. In other words, if for all objectives, the payoff of a strategy is greater than or equal to the payoff of another strategy, this relationship should be preserved by the utility function as well. This assumption translates to each agent always wanting to achieve a higher value in each objective.

A linear combination, as shown in Eqn. 5 below, is a widely used canonical example of a utility function:

$$u_i\_linear(\mathbf{w_i}, \mathbf{p}_i) = \sum_{o=1}^{d} w_{i,o} \cdot p_{i,o} \tag{5}$$

where $\mathbf{w_i}$ is a weight vector[3] that has one entry $w_{i,o}$ for each objective, representing the preferences that agent i has for the objectives.

Linear functions, although widely used due to their simplicity, are not as interesting or useful to study as the much broader class of non-linear functions. One reason for this is the well-known fact that linear functions cannot recover solutions in concave regions of the Pareto optimal set [19]. Preferences for real world decision makers are also highly likely to be non-linear, e.g., situations where a minimum value must be achieved on an objective require non-linear utility functions [10].

Note that non-linear utility functions may lead to different optimal strategies under SER and ESR, since a non-linear operation need not return the same result when applied to the vector payoff before or after the expectation (Equations 2 and 3). For linear utility functions, the SER and ESR optimisation criteria are equivalent [16]. If linear utility functions are used in a MONFG, the utility functions can

[2] The assumption that agents are in the TRIU setting is commonplace in prior work on MONFGs (e.g., [14, 16, 17]). This assumption is convenient for our study as we can focus on the effects of the agents' utility functions only, rather than considering the combined effects of different utility functions and different payoff vectors for each agent.

[3] A vector whose coordinates are all non-negative and sum up to 1.

be applied directly to the payoff matrix to create a so-called trade-off game, and standard single-objective solution methods from game theory can be applied.

Examples of non-linear utility functions used previously in the study of MONFGs include simple product and sum of squares functions [16]. Non-linear utility functions with preference parameters have not previously been studied in the context of MONFGs. In order to address this gap and enable us to study the effect of conflicting non-linear preferences, we consider a form of utility function (Eqn. 6) that is well known in the field of multi-attribute utility theory, the Cobb-Douglas (CD) function [1]:

$$u_i\_cd(\mathbf{w_i}, \mathbf{p_i}) = \prod_{o=1}^{d} p_{i,o}^{w_{i,o}} \tag{6}$$

where $\mathbf{w}$ is a weight vector representing an agent's preferences over objectives as before. In Section 3 we present a comprehensive analysis of the differences between the dynamics introduced by parameterised linear and CD utility functions in an example MONFG.

2.1.3 Solution concepts for MONFGs

Since we are contrasting two approaches for MOMAS in this work, we also look at two corresponding solution concepts, namely Nash equilibria for the utility-based approach with known utility functions, and Pareto front for the axiomatic approach.

Definition 2.2 (Nash equilibrium in a MONFG under SER) *A mixed-strategy profile $\boldsymbol{\pi}^{NE}$ is a Nash equilibrium in a MONFG under SER if for all $i \in \{1, ..., N\}$ and all $\pi_i \in \Pi_i$, with Π_i the set of mixed strategies for agent i:*

$$u_i\big[\,\mathbb{E}\,\mathbf{p}_i(\pi_i^{NE}, \boldsymbol{\pi}_{-i}^{NE})\big] \geq u_i\big[\,\mathbb{E}\,\mathbf{p}_i(\pi_i, \boldsymbol{\pi}_{-i}^{NE})\big] \tag{7}$$

i.e. π^{NE} is a Nash equilibrium under SER if no agent can increase the utility of her expected payoffs *by deviating unilaterally from π^{NE}.*

Previous work has shown that NE need not exist in MONFGs under SER [14], this result being emphasised by the equivalence that can be built between MONFGs and continuous games [12].

On the other hand, the Pareto front (PF) leverages the Pareto dominance relation to establish a partial ordering over strategies: a strategy profile π Pareto dominates another strategy profile π' if for all objectives $o : p_o(\pi) \geq p_o(\pi') \wedge \exists o' : p_{o'}(\pi) > p_{o'}(\pi')$.

Definition 2.3 (Pareto front) *The Pareto front is the set containing all Pareto non-dominated strategies, for any possible monotonically increasing (Eqn 4) utility function.*

As noted in [2], multiple strategies can have the same expected payoff, so the set retaining the strategies whose value functions correspond to the PF is called a *Pareto Coverage Set (PCS)*. Furthermore, since we are in a MAS, we will consider the PCS over the mixed-strategy profile Π of all the agents.

2.1.4 Multi-Objective Actor-Critic

As a learning approach for this work we adopt the independent Multi-Objective Actor Critic (MO-AC) proposed by [21], under the SER optimisation criterion (Eqn. 2). We define the SER objective of an agent as:

$$J(\boldsymbol{\theta}) = u\left(\sum_{a \in \mathcal{A}} \pi(a|\boldsymbol{\theta})\boldsymbol{Q}(a)\right) \tag{8}$$

where u is the non-linear utility function, $a \in \mathcal{A}$ is an action available to the agent, π the policy of the agent parameterised by $\boldsymbol{\theta}$ and $\boldsymbol{Q}(a) \in \mathbb{R}^d$ is the multi-objective action value vector that can be learned using a simple stateless Q-learning update rule [14]:

$$\boldsymbol{Q}(a_t) \leftarrow \boldsymbol{Q}(a_t) + \alpha_Q[\boldsymbol{p}_t - \boldsymbol{Q}(a_t)] \tag{9}$$

where α_Q is the learning rate for Q-learning. After the action values have been updated, the objective J can be calculated and analytically derived and we update $\boldsymbol{\theta}$ in the direction of maximising the SER:

$$\boldsymbol{\theta_{t+1}} \leftarrow \boldsymbol{\theta_t} + \alpha_\theta \nabla J(\boldsymbol{\theta}_t) \tag{10}$$

where α_θ is the learning rate for policy update.

3 Analysis

To aid our study of the relationship between utility-based and Pareto-based solution sets, we introduce a new MONFG in this work, which we will refer to as Game20[4]. This game was intentionally designed to be very simple, in order to make this first analysis of the relationship between utility-based and Pareto-based solution sets more straightforward. A key feature of Game20 is that only two of the joint actions in pure strategies are Pareto optimal. The reward vectors for the joint actions (R,L) and (L,R) are Pareto dominated by the reward vectors for the joint actions (L,L) and (R,R) respectively.

	L	*R*
L	$[3, 1]$	$[1, 2]$
R	$[2, 1]$	$[1, 3]$

Table 1: Game20 - a 2-action MONFG. For any joint strategy, both players receive the same payoff vector, i.e., $\mathbf{p_1} = \mathbf{p_2}$. Note that the reward vectors for the joint actions (R,L) and (L,R) are Pareto dominated by the reward vectors for the joint actions (L,L) and (R,R) respectively.

For our study, we consider two types of utility functions: the well-known linear form (Eqn. 5) and the non-linear Cobb-Douglas function (Eqn. 6). To gain an initial understanding of the combinations of preferences where the agents' interests will likely be aligned/non-aligned when both agents use either of these functions, we conducted a correlation analysis, the results of which are shown as heatmaps in Figs. 1 and 2. As far as we are aware, this is the first time that such a correlation analysis has been performed on utility functions with preference/weight parameters in the context of multi-objective multi-agent decision making.

We compute the correlation between utility functions for all combinations of weights on objective 1 ($w_{row,1}$, $w_{col,1}$) for each player, in the range 0.0 (lowest possible preference for objective 1) to 1.0 (highest possible preference for objective 1), where the weight space is discretised at a resolution of 0.1. Therefore, we consider 11 different weights for each player, so the number of distinct pairs of weights is $11 \times 11 = 121$ in total. For each distinct combination of player weights, we calculate the utility for each player over all possible payoff vectors in the range $[1.0, 1.0]$ to $[3.0, 3.0]$, stepping through the objective space at a resolution of 0.01. This resolution yields 40,401 unique multi-objective value vectors, and for each combination of player weights and value vector a utility value is computed using

[4] We pick the number 20 to fit in with the numbering system used for the Ramo library [18] for MONFGs, which at the time of writing already defines games numbered up to 19.

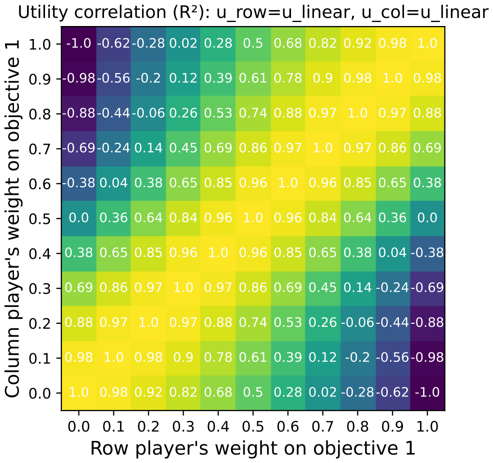

Figure 1: Utility correlation (R^2) heatmap for combinations of weights for objective 1, when both players use a linear function (Eqn. 5).

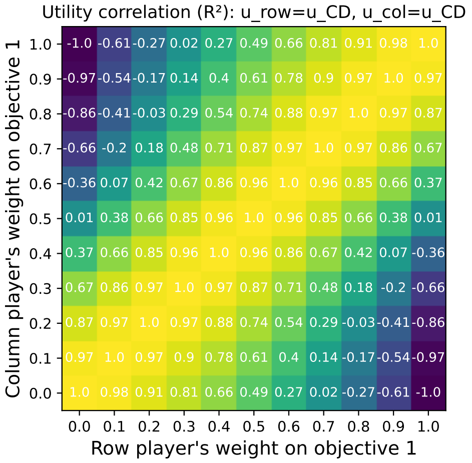

Figure 2: Utility correlation (R^2) heatmap for combinations of weights for objective 1, when both players use a CD function (Eqn. 6).

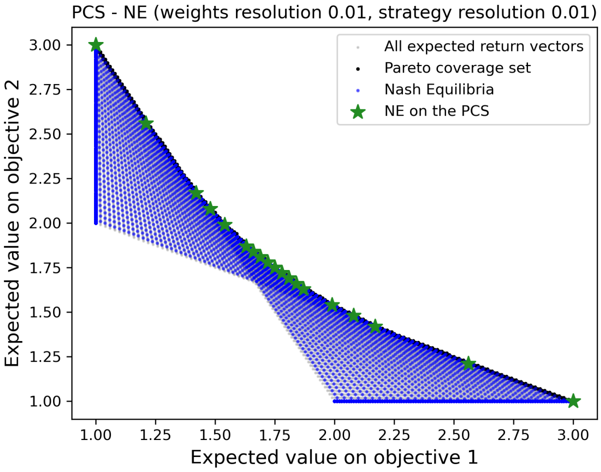

Figure 3: Expected values of Nash equilibria with linear functions.

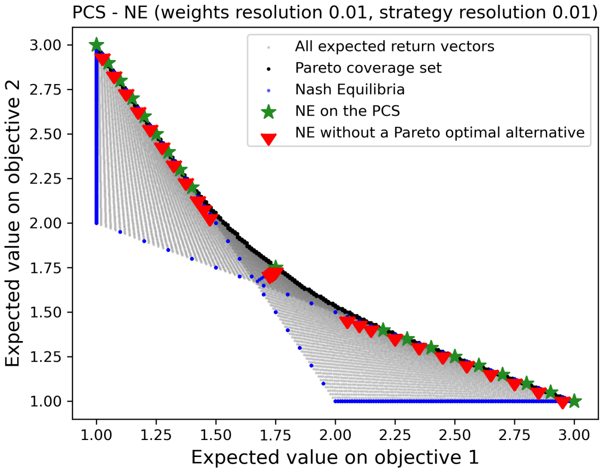

Figure 4: Expected values of Nash equilibria with CD functions.

both the linear and CD functions. Finally, a corresponding correlation value for the utilities over the objective space for each pair of player weights is computed using the built-in R^2 function in scikit-learn, and the correlation value is plotted on the heatmap corresponding to either the linear or CD functions as appropriate.

From Figs. 1 and 2 we observe some general trends. Along the main diagonal of both plots, we can see that when the preferences for objective 1 are matched (e.g., $w_{row,1} = 0.5$, $w_{col,1} = 0.5$), there is a perfect correlation of 1.0 between the players' utilities. By contrast, when the player's preferences are as different as possible (e.g., $w_{row,1} = 0.0$, $w_{col,1} = 1.0$), we observe a perfect negative correlation of -1.0 between the players' utilities.

We will use Figs. 3 and 4 to demonstrate the differences between utility-based and Pareto-based solution sets for both linear and CD utility functions. For both plots, we stepped through the space of all possible joint strategies at a resolution of 0.01, and through the space of possible weights on objective 1 at a resolution of 0.01. This gives a total of 101 possible individual strategies and 101 individual preference vectors for each player, or $101 \times 101 = 10,201$ possible joint strategies and $10,201$ possible preference vectors. In both plots, the full set of expected return vectors is plotted in light grey, where each expected return vector corresponds to one of the 10,201 joint strategies considered. The Pareto coverage set is shown in black on both plots. The `p_prune()` function from the RAMO library [18] was used to generate the PCS, by pruning out all dominated expected return vectors. For both plots the set of all expected return vectors and the Pareto coverage set are identical, as neither of these sets are related to the utility functions used.

When comparing Figs. 3 and 4, it is immediately apparent that there is a much higher density of Nash equilibria (blue points) when linear functions are used by both players, whereas the density of Nash equilibria is much lower in the case of CD utilities. NE in the linear case are very evenly distributed through the space of all expected return vectors, whereas NE in the CD case are focused more towards the extremities of the space of expected return vectors. We also observe that the sets of PCS strategies that can be Nash equilibria are very different for both utility functions, and that the two extreme points on the PCS can be Nash equilibria under certain weight combinations for both utility functions.

The most important difference between Figs. 3 and 4 is the set of Nash equilibria that do not have a Pareto optimal alternative (red inverted triangles)[5]. In the linear case, for any of the considered com-

[5] The discretisation approach that we take here has limitations, in that we consider only a subset of all possible joint strategies. We can use this ap-

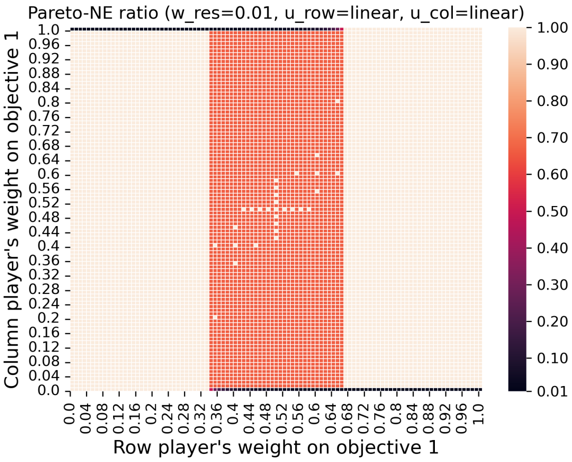

Figure 5: Heatmap of the ratio of computed Nash equilibria that are Pareto optimal when both players use a linear utility function.

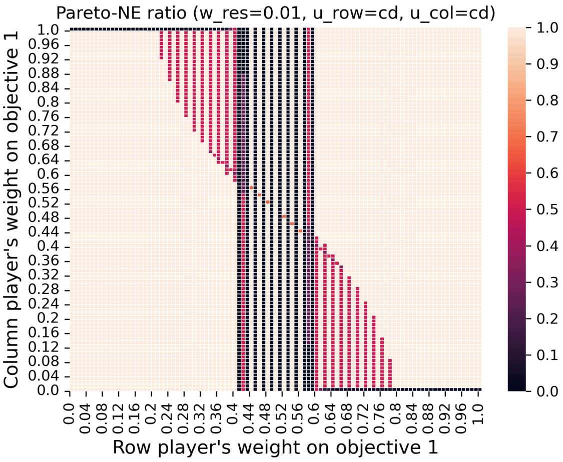

Figure 6: Heatmap of the ratio of computed Nash equilibria that are Pareto optimal when both players use a CD utility function.

binations of weight vectors, there is always at least one NE that is Pareto optimal. By contrast, in the CD case there are combinations of weights where the only NE that are present are not Pareto optimal. In other words, for certain combinations of preferences, searching for joint strategies using a Pareto-based approach alone is insufficient, as it is possible for certain weight combinations that none of the joint strategies that are Pareto optimal will be Nash equilibria, implying that no stable solution could be found using Pareto-based sets alone. We have demonstrated this effect only for one simple MONFG and one form of parameterised non-linear utility functions, however we expect that this effect will also be present in more complex MONFGs as well as for different non-linear utility functions.

Figs. 5 and 6 show heatmaps with the ratios of the number of Nash equilibria that are Pareto optimal to the total number of Nash equilibria for various weight combinations, with the $w_{row,1}$ on the x-axis and $w_{col,1}$ on the y-axis. The heatmaps use a resolution of 0.01 for both the joint strategy space and the weight space $(10, 201$ unique pairs of weights$)$[6], and display persymmetric matrices (i.e., symmetric with respect to the second diagonal). The key difference between these plots is that for the CD utility function there are more regions of the joint weight space characterised by a very low probability of any of the found Nash equilibria being Pareto optimal. In comparison to the linear case, for the CD utility function the ratio can even reach a value of 0, corresponding to the situations identified in Fig. 4 of weight instances with NE without Pareto optimal alternatives. For the linear utility function, the higher density of NE observed in Fig. 3 is reflected in the region where the row player's weight on objective 1 ranges between $0.34 - 0.66$, where 2 out of the 3 NE are Pareto optimal, resulting in a ratio of 0.67. Appendix B presents heatmaps of the number of computed Pareto optimal NE.

One final issue to consider is whether there is any relationship between the correlation of player utilities for a given set of joint preferences and the likelihood that the Nash equilibria for that set of joint preferences are Pareto optimal. In some extreme regions where there is a low or negative correlation between the player's utilities, we see clearly from Figs. 5 and 6 that there is a very low probability that any of the discovered Nash equilibria are Pareto optimal. For example when $w_{row,1} \in [0.0, 0.4]$ and $w_{col,1} = 1.0$, or when $w_{row,1} \in [0.6, 1.0]$ and $w_{col,1} = 0.0$. However, in some instances where the player's utilities are more highly correlated we also observe areas with a low probability of Pareto optimal NE, depending on the specific preference values. It is likely the the probability of Pareto optimal NE depend on other factors besides utility correlation, such as the structure of the payoff matrix and whether the equilibria are in pure or mixed strategies - the interplay of these factors merit further study in future.

4 Experiments

We extend our analysis of Game20 with an empirical evaluation of the learning process, using the MO-AC introduced in Section 2.1.4, available as part of the Ramo library [18]. Each player independently uses MO-AC to learn a *softmax* policy that maximises their individual SER objective. The players interact for 20,000 episodes and use $\alpha_Q = 0.01$ and $\alpha_\theta = 0.05$ as the learning rate values. All the experiments are averaged over 100 trials and are conducted using either the linear (Eqn. 5) or Cobb-Douglas (Eqn. 6) utility functions for both players. We have selected weigh combinations to cover a wide range of scenarios for Game20, according to the different regions identified in Figs. 5 and 6. Tables 2 and 3 from the appendix present the full overview of the experiments, from which we select a few instances and present in more detail below. In each table, w_1 represents Player 1's weight for the first objective, i.e., $(w_{1,1}, w_{1,2}) = (w_1, 1 - w_1)$, while w_2 represents Player 2's weight for the first objective, i.e., $(w_{2,1}, w_{2,2}) = (w_2, 1 - w_2)$.

To complement the analysis from Section 3, we also focus on evaluating the proximity of the learned strategies to the NE of the respective game (i.e., under the considered utility function and weigh values). To this end we employ a straightforward distance metric between the learned strategies, $\boldsymbol{\pi}$, and the strategies under each NE, $\boldsymbol{\pi}_{NE}$, namely, the maximum value of the element-wise difference between $\boldsymbol{\pi}$ and each $\boldsymbol{\pi}_{NE}$. We use 0.02 as a threshold to establish the convergence to a NE, or, in other words, we allow for at most a 0.02 deviation from the NE strategy for each action probability of

proach to make observations about the likelihood of finding Pareto optimal Nash equilibria. We tried smaller resolutions in the strategy space (e.g., 0.001) which still yielded instances of Nash equilibria with no Pareto optimal alternative. In future work we aim to prove mathematically that there are cases where no Pareto optimal Nash equilibria exist.

[6] We note that for the regions pertaining to the row player's weight on objective $1 \in \{0.41, 0.43, 0.45, 0.47, 0.49, 0.51, 0.53, 0.55, 0.57, 0.59\}$ we used a strategy resolution of 0.005, as the previous resolution was insufficient to identify the equilibria in those regions.

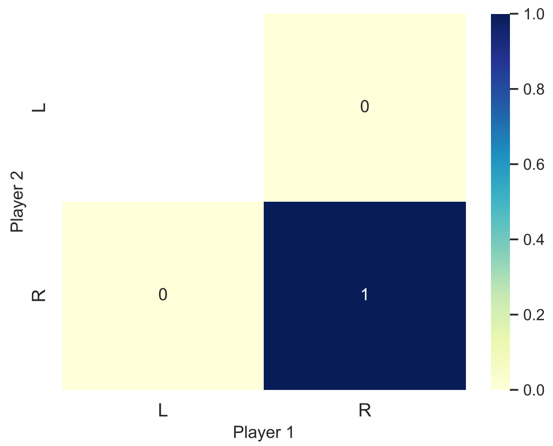

Figure 7: Heatmap over the joint strategy space, averaging the outcomes of the last 5% interactions over 100 runs, with a linear utility function, $w_1 = 0.0$, $w_2 = 0.5$.

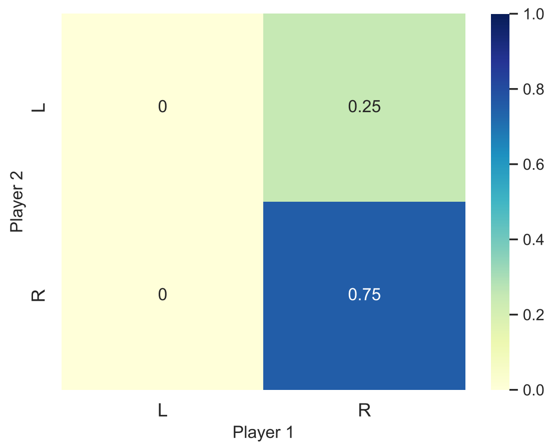

Figure 8: Heatmap over the joint strategy space, averaging the outcomes of the last 5% interactions over 100 runs, CD utility function, $w_1 = 0.0$, $w_2 = 0.5$

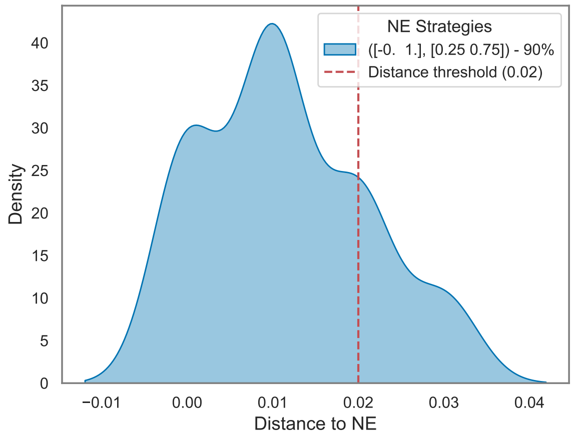

Figure 9: Kernel density estimate plot of the distribution of the distances between learned strategies and NE, with a CD utility function, $w_1 = 0.0$, $w_2 = 0.5$.

each player[7]. A first observation is that when the players' preferences are fully aligned (i.e., $w_1 = w_2 \in \{0.0, 0.5, 1.0\}$), the set of possible NE are fully part of the PCS[8] and independent MO-AC converges with high probability to one of the NE strategies.

For the case in which $w_1 = 0.0$ and $w_2 = 0.5$, we notice an important discrepancy between the linear and CD utility functions. Firstly, under the CD utility only one NE is present, with strategy $[0, 1]$ (i.e., fully action R) for Player 1 and strategy $[0.25, 0.75]$ for Player 2. Agents using independent MO-AC are able to converge to this NE and we can visualise this outcome in a heatmap over the joint strategy space, in Fig. 8, averaged for the last 5% of interactions over the 100 runs. To further visualise the results we present a kernel density estimate (KDE) plot of the distribution of distances (over the 100 trials) between the learned strategies and the NE in Game20, in Fig. 9. Under our distance metric detailed above, with a threshold of 0.02, independent MO-AC converges to the NE in this case in 90% of the runs. Secondly, under the linear utility there is a continuum of strategies, with Player 1 being able to shift his probability distribution from $[1, 0]$ to $0.5, 0.5$, while for Player 2 $[1, 0]$ remains optimal. Next to this continuum, there is one more NE present, namely strategy $[0, 1]$ for both players. Under MO-AC, we remark that the players fully converge to this last NE (Fig. 7).

Also using KDE plots, we can visualise the convergence to NE strategies under the linear utility function (Figs. 10) and the CD utility function (Fig. 11), when $w_1 = 0.62$ (i.e., $(w_{1,1}, w_{1,2}) = (0.62, 0.38)$ for Player 1) and $w_2 = 0.38$ (i.e., $(w_{2,1}, w_{2,2}) = (0.38, 0.62)$ for Player 2). We observe that under linear utility function the players converge to one of the two pure NE (which are also in the PCS) 96% of the time, in comparison to 87% for the CD utility function. We also note that, for both utility functions, the mixed strategy NE (Player 1 $[0.86, 14]$, Player 2 $[0.38, 0.62]$) appears to be challenging to learn using MO-AC, however we did observe runs in the CD utility function case that achieved this outcome.

The final setting we turn our attention to is one in which the players' preferences are fully misaligned, namely $w_1 = 1.0$ and $w_2 = 0$. Under both utility functions, there is again a continuum of NEs, where Player 1 can shift his strategy from fully playing L ($[1, 0]$) to fully playing R ($[0, 1]$), while the optimum strategy for Player 2 is $[0, 1]$ in all cases. The visualise the results in this setting, we use a scatter plot and represent on the x-axis the probability of selecting action L for Player 1 and on the y-axis the probability of selecting action L for Player 2, and plot the joint strategy for each of our 100 runs. In this region of the weight space, both utility functions induce a similar behaviour, with player converging more often towards the cluster of NEs that are not Pareto optimal (i.e., with a higher probability of Player 1 to select L), as it can be observed in Figs. 12 and 13. This result reinforces our hypothesis that when the preferences over the objectives are in strong disagreement, the probability of converging to a Pareto optimal equilibria is negatively affected.

[7] We note that while we focus here on the *learned strategies* and thus on a simple distance metric defined over the strategy space, further analysis is warranted on the landscape in strategy space around the NEs. An alternative is investigating if concepts such as the incentive to deviate and ϵ-NE could provide better grounded insights in the utility space, in this case.

[8] Note that this is not generally true for all instances of our game, under fully aligned preferences, as shown in Figs. 5 and 6.

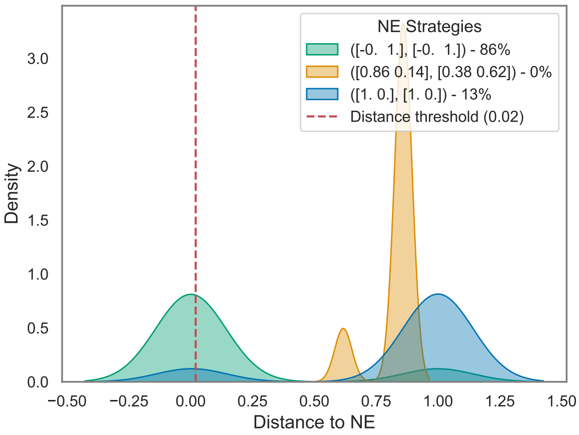

Figure 10: Kernel density estimate plot of the distribution of the distances between learned strategies and NE, with a linear utility function, $w_1 = 0.62$, $w_2 = 0.38$.

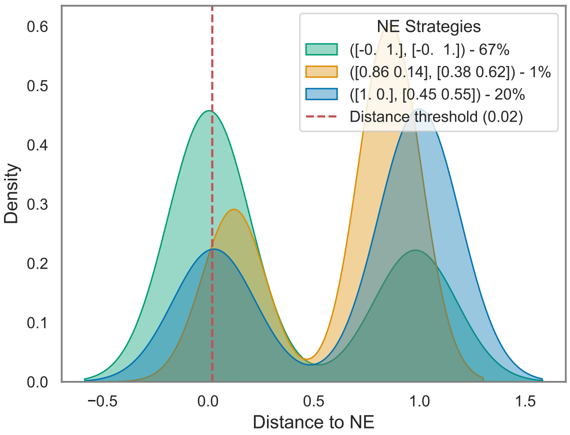

Figure 11: Kernel density estimate plot of the distribution of the distances between learned strategies and NE, with a CD utility function, $w_1 = 0.62$, $w_2 = 0.38$.

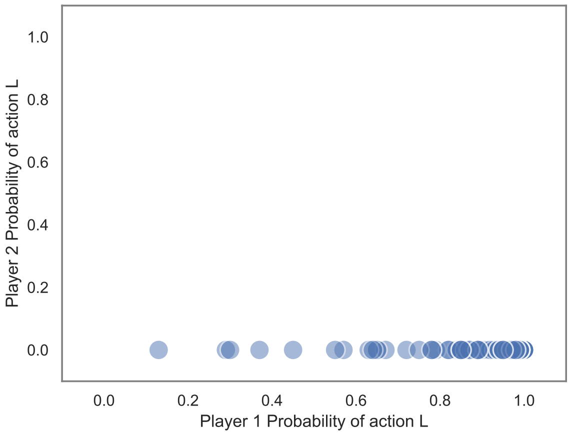

Figure 12: Scatter plot of the learned joint strategies in each of the 100 runs, with a linear utility function, $w_1 = 1.0$, $w_2 = 0.0$.

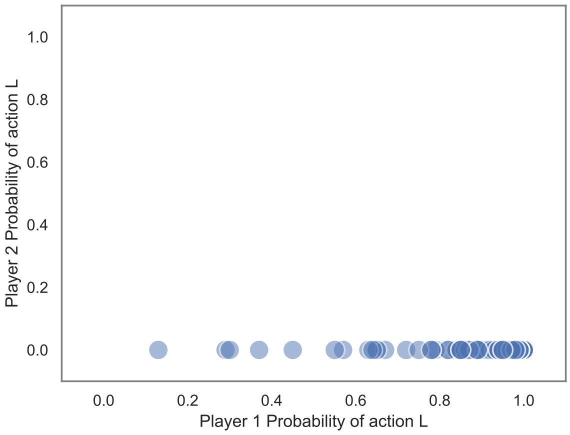

Figure 13: Scatter plot of the learned joint strategies in each of the 100 runs, with a CD utility function, $w_1 = 1.0$, $w_2 = 0.0$

5 Conclusion and Future Work

In this work we contrasted the two potential approaches for MODeM, namely the utility-based and axiomatic approaches, in the context of multi-agent systems. We conducted our analysis on a novel MONFG, Game20 (Table 1). Under the axiomatic approach, we derived the entire Pareto coverage set over the mixed-strategy profiles of all players. Under the utility-based approach, we considered the scalarised expected returns optimisation criterion and derived all the Nash equilibria under parameterised linear and non-linear utility functions.

Using parameterised utility functions allowed us to perform a correlation analysis between the players' preferences and to gain more insight on the impact of this aspect on the learning process and resulting stable solutions. Our analysis demonstrated in for some preference combinations with a very low correlation, the probability of finding a Nash equilibrium that is Pareto optimal is much lower.

Furthermore, we demonstrated that under non-linear utility, for certain preference combinations, the set of NE and the PCS are disjoint, implying that in these situations Pareto-based approaches will not find stable solutions in the joint strategy space.

For the learning experiments, we used independent multi-objective actor critic, and investigated the convergence probabilities to the set of NE, both for Pareto and non-Pareto optimal outcomes. We observed that when preferences are in strong disagreement, convergence to Pareto optimal equilibria was negatively impacted, with players ending up exclusively in dominated equilibria.

We hope that this initial analysis will draw more attention to the importance of studying the alignment between the preferences of the users in multi-objective settings, as this will likely have a strong impact on the number of stable solutions and on the capacity of learning approaches to converge to Pareto optimal outcomes. This will be crucial in future decision-support systems, in domains such as smart grids, logistics, resource management, etc.

In this work we have adopted the stateless MONFG framework for our analysis. Our findings should be extended to sequential settings, e.g., multi-objective stochastic games (MOSGs), to better reflect and translate to real-world problem domains.

Finally, another future research avenue we are interested to pursue is helping agents to converge on the Nash equilibrium that is best in terms of social welfare. This may require optimisation from a global perspective or the use of joint action learners and potentially opponent modelling techniques.

Acknowledgements

Roxana Rădulescu is supported by the Research Foundation – Flanders (FWO), grant number 1286223N. The resources and services used in this work were provided by the VSC (Flemish Supercomputer Center), funded by the FWO and the Flemish Government.

References

[1] Steven Durlauf and Lawrence E Blume, *The new Palgrave dictionary of economics*, Springer, 2016.

[2] Conor F. Hayes, Roxana Rădulescu, Eugenio Bargiacchi, Johan Källström, Matthew Macfarlane, Mathieu Reymond, Timothy Verstraeten, Luisa M. Zintgraf, Richard Dazeley, Fredrik Heintz, Enda Howley, Athirai A. Irissappane, Patrick Mannion, Ann Nowé, Gabriel Ramos, Marcello Restelli, Peter Vamplew, and Diederik M. Roijers, 'A practical guide to multi-objective reinforcement learning and planning', *Autonomous Agents and Multi-Agent Systems*, (2022).

[3] Conor Francis Hayes, Mathieu Reymond, Diederik Marijn Roijers, Enda Howley, and Patrick Mannion, 'Monte carlo tree search algorithms for risk-aware and multi-objective reinforcement learning', *Autonomous Agents and Multi-Agent Systems*, **37**(26), (2023).

[4] Conor Francis Hayes, Diederik Marijn Roijers, Enda Howley, and Patrick Mannion, 'Decision theoretic planning for the expected scalarised returns', in *Proceedings of the 21st International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2022)*, (May 2022).

[5] Ayumi Igarashi and Diederik M Roijers, 'Multi-criteria coalition formation games', in *Algorithmic Decision Theory: 5th International Conference, ADT 2017, Luxembourg, Luxembourg, October 25–27, 2017, Proceedings 5*, pp. 197–213. Springer, (2017).

[6] Junlin Lu, Patrick Mannion, and Karl Mason, 'A multi-objective multi-agent deep reinforcement learning approach to residential appliance scheduling', *IET Smart Grid*, **5**(4), 260–280, (2022).

[7] Wenjie Luo, Cheol Park, Andre Cornman, Benjamin Sapp, and Dragomir Anguelov, 'Jfp: Joint future prediction with interactive multi-agent modeling for autonomous driving', in *Conference on Robot Learning*, pp. 1457–1467. PMLR, (2023).

[8] Patrick Mannion, Jim Duggan, and Enda Howley, 'An experimental review of reinforcement learning algorithms for adaptive traffic signal control', in *Autonomic Road Transport Support Systems*, eds., Leo Thomas McCluskey, Apostolos Kotsialos, P. Jörg Müller, Franziska Klügl, Omer Rana, and René Schumann, 47–66, Springer International Publishing, (2016).

[9] John Nash, 'Non-cooperative games', *Annals of Mathematics*, **54**(2), 286–295, (1951).

[10] David O'Callaghan and Patrick Mannion, 'Exploring the impact of tunable agents in sequential social dilemmas', in *Proceedings of the Adaptive and Learning Agents Workshop (at AAMAS 2021)*, (May 2021).

[11] Mathieu Reymond, Conor F Hayes, Denis Steckelmacher, Diederik M Roijers, and Ann Nowé, 'Actor-critic multi-objective reinforcement learning for non-linear utility functions', *Autonomous Agents and Multi-Agent Systems*, **37**(2), 23, (2023).

[12] Willem Röpke, Carla Groenland, Roxana Rădulescu, Ann Nowé, and Diederik M. Roijers, 'Bridging the gap between single and multi objective games', in *Proceedings of the 2023 International Conference on Autonomous Agents and Multiagent Systems*, AAMAS '23, p. 224–232, Richland, SC, (2023). International Foundation for Autonomous Agents and Multiagent Systems.

[13] Willem Röpke, Diederik M Roijers, Ann Nowé, and Roxana Rădulescu, 'On nash equilibria in normal-form games with vectorial payoffs', *Autonomous Agents and Multi-Agent Systems*, **36**(2), 53, (2022).

[14] Roxana Rădulescu, Patrick Mannion, Diederik Marijn Roijers, and Ann Nowé, 'Equilibria in multi-objective games: A utility-based perspective', in *Proceedings of the Adaptive and Learning Agents Workshop (at AAMAS 2019)*, (May 2019).

[15] Roxana Rădulescu, Patrick Mannion, Diederik Marijn Roijers, and Ann Nowé, 'Multi-objective multi-agent decision making: a utility-based analysis and survey', *Autonomous Agents and Multi-Agent Systems*, **34**(10), (2020).

[16] Roxana Rădulescu, Patrick Mannion, Yijie Zhang, Diederik Marijn Roijers, and Ann Nowé, 'A utility-based analysis of equilibria in multi-objective normal form games', *The Knowledge Engineering Review*, **35**(e32), (2020).

[17] Roxana Rădulescu, Yijie Zhang, Timothy Verstraeten, Patrick Mannion, Diederik Marijn Roijers, and Ann Nowé, 'Opponent learning awareness and modelling in multi-objective normal form games', *Neural Computing and Applications*, **34**(3), (2022).

[18] Willem Röpke. Ramo: Rational agents with multiple objectives. https://github.com/wilrop/mo-game-theory, 2022.

[19] Peter Vamplew, John Yearwood, Richard Dazeley, and Adam Berry, 'On the limitations of scalarisation for multi-objective reinforcement learning of pareto fronts', in *AI 2008: Advances in Artificial Intelligence: 21st Australasian Joint Conference on Artificial Intelligence Auckland, New Zealand, December 1-5, 2008. Proceedings 21*, pp. 372–378. Springer, (2008).

[20] Logan Yliniemi, Adrian K Agogino, and Kagan Tumer, 'Simulation of the introduction of new technologies in air traffic management', *Connection Science*, **27**(3), 269–287, (2015).

[21] Yijie Zhang, Roxana Rădulescu, Patrick Mannion, Diederik M Roijers, and Ann Nowé, 'Opponent modelling for reinforcement learning in multi-objective normal form games', in *Proceedings of the 19th International Conference on Autonomous Agents and MultiAgent Systems*, pp. 2080–2082, (2020).

A Experimental setting overview

Tables 2 and 3 present an overview of all the weight preferences used in our experimental setup: w_1 represents Player 1's weight for the first objective, i.e., $(w_{1,1}, w_{1,2}) = (w_1, 1 - w_1)$, while w_2 represents Player 2's weight for the first objective, i.e., $(w_{2,1}, w_{2,2}) = (w_2, 1 - w_2)$, under either the linear or CD utility function. The tables also include, for each weight combination, the expected payoffs for each NE, the NE joint strategy, as well as the convergence probability to each of the NE of our learning approach, the multi-objective actor critic (MO-AC). In the 'NE expected payoff' column, we highlight in bold the NE outcomes that are also part of the PCS.

We note that in settings in which the players' preferences are aligned, $(w_1 = 0, w_2 = 0)$ or $(w_1 = 1, w_2 = 1)$, there is only one NE present and our approach manages to converge to the NE strategy 100% of the time. On the other end of the alignment spectrum, when the players' preferences are completely opposing $(w_1 = 1, w_2 = 0)$, the convergence rate to NE is still maintained, however the players do not manage to converge to the Pareto optimal NE.

Table 2: Overview of the **linear utility** weight combinations used in the learning evaluation.

w_1	w_2	NE expected payoff	Convergence probability	Strategies	
				Player 1	Player 2
0.0	0.0	$\mathbf{[1, 3]}$	1.0	$[0, 1]$	$[0, 1]$
0.0	0.5	$\mathbf{[3, 1]}$	0.0	$[1, 0]$	$[1, 0]$
		$[2.99, 1]$, ..., $[2.5, 1]$	0.0,...,0.0	$[0.99, 0.01]$, ..., $[0.5, 0.5]$	$[1, 0]$, ..., $[1, 0]$
		$\mathbf{[1, 3]}$	1.0	$[0, 1]$	$[0, 1]$
0.5	0.5	$\mathbf{[3, 1]}$	0.45	$[1, 0]$	$[1, 0]$
		$\mathbf{[1.75, 1.75]}$	0.0	$[0.5, 0.5]$	$[0.5, 0.5]$
		$\mathbf{[1, 3]}$	0.55	$[0, 1]$	$[0, 1]$
0.61	0.48	$\mathbf{[3, 1]}$	0.49	$[1, 0]$	$[1, 0]$
		$[1.61, 1.88]$	0.0	$[0.56, 0.44]$	$[0.39, 0.41]$
		$\mathbf{[1, 3]}$	0.51	$[0, 1]$	$[0, 1]$
0.62	0.38	$\mathbf{[3, 1]}$	0.13	$[1, 0]$	$[1, 0]$
		$[1.7, 1.7]$	0.0	$[0.86, 0.14]$	$[0.38, 0.62]$
		$\mathbf{[1, 3]}$	0.86	$[0, 1]$	$[0, 1]$
1.0	0.0	$[1, 2]$	0.34	$[1, 0]$	$[0, 1]$
		$[1, 2.01]$, ..., $[1, 2.99]$	0.45, ..., 0	$[0.99, 0.01]$, ..., $[0.01, 0.99]$	$[0, 1]$, ..., $[0, 1]$
		$\mathbf{[1, 3]}$	0.0	$[0, 1]$	$[0, 1]$
1.0	0.6	$\mathbf{[3, 1]}$	1.0	$[1, 0]$	$[1, 0]$
		$[1, 2.8]$, ..., $[1, 2.99]$	0.0, ..., 0.0	$[0.2, 0.8]$, ..., $[0.01, 0.99]$	$[1, 0]$, ..., $[1, 0]$
		$\mathbf{[1, 3]}$	0.0	$[1, 0]$	$[1, 0]$
1.0	1.0	$\mathbf{[3, 1]}$	1.0	$[1, 0]$	$[1, 0]$

Table 3: Overview of the **CD utility** weight combinations used in the learning evaluation.

w_1	w_2	NE expected payoff	Convergence probability	Strategies	
				Player 1	Player 2
0.0	0.0	$\mathbf{[1, 3]}$	1.0	$[0, 1]$	$[0, 1]$
0.0	0.5	$\mathbf{[1.25, 1.25]}$	0.90	$[0, 1]$	$[0.25, 0.75]$
0.5	0.5	$\mathbf{[2.5, 1.25]}$	0.39	$[1, 0]$	$[0.75, 0.25]$
		$\mathbf{[1.75, 1.75]}$	0.0	$[0.5, 0.5]$	$[0.5, 0.5]$
		$\mathbf{[1.25, 2.5]}$	0.51	$[0, 1]$	$[0.25, 0.75]$
0.61	0.48	$\mathbf{[2.4, 1.3]}$	0.73	$[1, 0]$	$[0.7, 0.3]$
		$\mathbf{[1.2, 2.6]}$	0.13	$[0, 1]$	$[0.2, 0.8]$
0.62	0.38	$[1.9, 1.55]$	0.2	$[1, 0]$	$[0.45, 0.55]$
		$[1.7, 1.7]$	0.01	$[0.86, 0.14]$	$[0.38, 0.62]$
		$\mathbf{[1, 3]}$	0.67	$[0, 1]$	$[0, 1]$
1.0	0.0	$[1, 2]$	0.34	$[1, 0]$	$[0, 1]$
		$[1, 2.01]$, ..., $[1, 2.99]$	0.45, ..., 0.0	$[0.99, 0.01]$, ..., $[0.01, 0.99]$	$[0, 1]$, ..., $[0, 1]$
		$\mathbf{[1, 3]}$	0.0	$[0, 1]$	$[0, 1]$
1.0	0.6	$\mathbf{[3, 1]}$	1.0	$[1, 0]$	$[1, 0]$
1.0	1.0	$\mathbf{[3, 1]}$	1.0	$[1, 0]$	$[1, 0]$

B Additional results

Figures 14 and 15 show heatmaps of the number of Nash equilibria that are Pareto optimal for all the possible weight combinations under the linear and CD utility functions. For the linear utility function the number of Pareto optimal NE for each weight combination ranges between 1 and 3, while for the CD utility there are some weight combinations where there are no Pareto optimal NE.

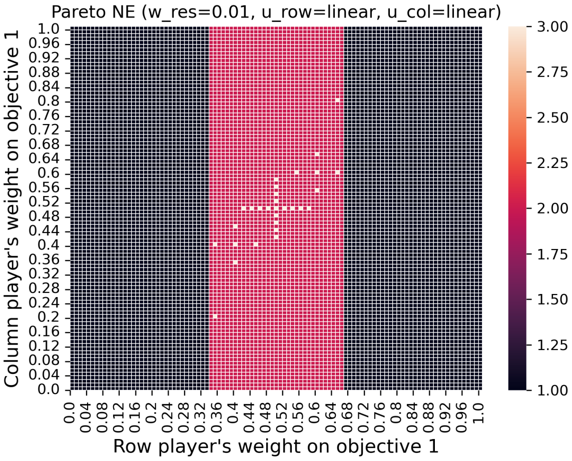

Figure 14: Heatmap of the number of computed Nash equilibria that are Pareto optimal when both players use a linear utility function.

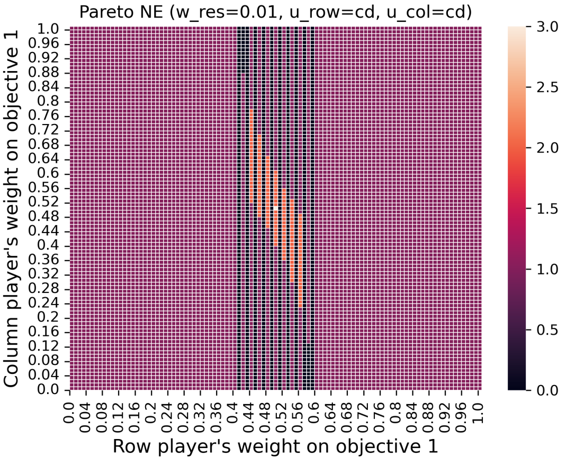

Figure 15: Heatmap of the number of computed Nash equilibria that are Pareto optimal when both players use a CD utility function.