Learning in automated negotiation

Renting, B.M.

II

Evaluating Learning Negotiation Agents

5

6

Analysis of Learning Agents in Automated Negotiation

This chapter is under review at the time of writing.

6.1 Introduction

The Automated Negotiating Agents Competition (ANAC) was first organised in 2010 to support the development and benchmarking of automated negotiating agents [10]. Since 2017, ANAC has been extended with a number of additional leagues that each focus on a more specialised challenge, such as the game of Diplomacy [81], supply chain environments [6], the game of Werewolves [6] or negotiations between agents and humans [106]. The main league, which focuses on more classical agent-based negotiations, has since then been called Automated Negotiation League (ANL). In ANL, the participating agents are designed to bargain with other agents over a collective agreement in scenarios with conflicting interests.

Over the years, the main league of ANAC has evolved to incorporate new challenges, such as multi-lateral negotiation, preference elicitation and large intractable solution spaces. In most earlier editions, the negotiations were considered largely single-shot sessions, in which the agents would be re-initialised for every new negotiation, making it impossible for them to use any knowledge from previous interactions. However, in some future applications of negotiating agents like the ones provided before, it is imaginable that agents would encounter opponents multiple times, making the negotiation a repeated game. In such scenarios, it is also realistic that agents would use information from previous encounters in order to optimise their performance. This adds a learning dynamic between negotiation sessions to the negotiating agents. Agents generally also learn within a single negotiation session (e.g., for opponent preference estimation), but in this work, we exclusively mean agents that learn over multiple negotiation sessions when we refer to "learning agents". We intended to study such learning behaviour further using the ANL.

It is beneficial for a negotiating agent to implement a measure of adaptivity to the environment in which it carries out negotiation. Negotiation strategies can be adapted depending on the characteristics of the negotiation scenario and the opponent. For example, a negotiation scenario in which the preferences of agents are strongly conflicting might require an agent to behave differently than a scenario in which preferences are largely overlapping. Also, if an opponent drives a hard bargain, it might not be smart to adopt a cooperative strategy, as this risks being extorted. We have seen agents that successfully adapt to negotiation scenarios [76] and opponents [137, 123], but not yet in environments where opponents are also learning.

We set the challenge of the 2021 and 2022 editions of ANL with the goal to study learning negotiating agents better. The challenge was to improve performance by learning and adapting to the behaviour of the other agents submitted to ANL. This article provides an overview of learning agents in the history of ANL in general and the submissions and results of the 2022 edition of ANL, held in conjunction with the International Joint Conference on Artificial Intelligence (IJCAI) 2022, specifically. We consider the competition and its design part of the novelty of this work, which we now extend with a thorough analysis. We aim to answer the following questions:

1. Can we design a negotiation competition where participants manage to submit strategies including learning mechanisms?

2. Given that the negotiation games are general sum, do agents that perform well in the social welfare performance criteria also perform well in terms of individual utility and vice versa?

3. Do learning strategies benefit negotiation agents?

4. What is the effect of the negotiation scenario generator on the performance of the agents?

5. Does the standard approach of averaging performance, used in earlier editions of ANL, provide a robust ranking of agents?

We have analysed to which extent the agents are sensitive to the characteristics of the given negotiation scenarios. We observed that agents perform noticeably better in scenarios with strong mutually beneficial outcomes or a high variance of utility over the outcomes for both agents. Furthermore, we have analysed the results in depth and explored to what extent the learning algorithms positively affected the agents' negotiation performance.

We draw three main conclusions. Firstly, we conclude that agents that apply learning techniques clearly outperform those that do not, which shows that learning can improve the performance of a negotiating agent. Secondly, however, we also observe that a naïve strategy that does not learn at all outperforms all other agents when we look at the results from a game-theoretical perspective, forming an empirical Nash equilibrium. Finally, we conclude that the current approach of ranking agents through average scores is not sufficiently robust and that there is no clear alternative to ranking the agents. We hope this work serves as a useful starting point of this last issue within the automated negotiation agents community.

6.2 RELATED WORK

6.2.1 THE AUTOMATED NEGOTIATING AGENTS COMPETITION

The annual Automated Negotiating Agents Competition (ANAC) was first organized in 2010. The first three editions of ANAC were focused on the simplest scenario only, in which two agents negotiate with each other over a domain with linear utility functions [16]. In these tournaments, each negotiation session was completely independent of previous sessions, so the agents were not allowed to learn from previous encounters. This changed in 2013 when the option was added for agents to store information between sessions and, hence, to learn and evolve over the course of the tournament [1], but opponents were anonymous. In 2014, this option was removed again, and the focus shifted to very large domains, where the number of possible deals was of the order 10^{30} and in which the utility functions were non-linear [58]. From 2015 onward, the competition returned to smaller domains and linear utility but focused on multi-lateral negotiations involving more than two agents at a time [57]. In the 2017 and 2018 editions, for the second time, the opportunity was provided to the agents to maintain an internal state and to learn from previous encounters with opponents. However, this was limited to a single negotiation setting, which was repeated six times with the same opponent and

scenario. In 2019 and 2020, the focus was on bilateral negotiations again, but this time with *partially* known utility functions simulating an agent estimating human preferences [6]. Then, in 2021, the preferences of the agents were again represented by linear utility functions and the option to learn from previous negotiation sessions was re-introduced, similar to the setting adopted in 2013. The difference between the 2021 and 2022 editions and the 2013 edition is that in 2013, opponents were anonymous; this means that they were not able to adapt to specific opponents.

6.2.2 Learning agents in automated negotiation

As mentioned previously, there are multiple opportunities for learning in the context of automated negotiation. Within a single negotiation, the stream of proposals received from the opponent contains information about the preferences and tactics of opponents [12]. In repeated encounters, agreements and observations of previous encounters with the opponent can also be used to reason about the opponent's tactics. It is important to note that learning and adapting to opponents can benefit all agents involved in a negotiation rather than merely improve individual performance. Specifically, adapting to opponents can improve the chances of reaching an agreement and finding mutually beneficial (i.e., Pareto efficient) outcomes in settings where preferences are partially aligned.

Another option is offline learning, where the performance of an agent is optimised in a controlled environment through training on a given set of agents and negotiation scenarios. A distinction can be made in the way these agents are trained. Some take an approach where the behaviour of the agent is parameterised, and these parameters are optimised either through reinforcement learning (see Chapter 5) or other algorithmic optimisers (see Chapter 3). Others take an algorithm selection approach as we discussed in Chapter 4.

6.2.3 Learning agents in ANL

Many agents in the history of ANL have implemented a form of preference estimation, which attempts to learn opponent preferences. Accurate preference estimation helps find mutually beneficial outcomes and thus potentially improves performance. The simple frequency model that was part of the SmithAgent [59] submitted in the 2010 edition of ANAC is often used; this model estimates preferences based on the frequency an opponent has offered an outcome. Besides frequency models, Bayesian models based on Bayesian inference are also commonly seen [69, 159]. Baarslag et al. [11] created an overview of preference estimation methods that have been applied in ANL and demonstrated that frequency models show better performance than Bayesian models. As frequency models are also performant and conceptually easy, most agents in the ANL competition implement a frequency model.

Aside from learning opponent preferences, learning opponent strategy can also help improve performance. However, few agents in the past of ANL have implemented such a mechanism, with one notable exception. In the 2012 edition, an agent adopted an algorithm selection approach using previously submitted agents called the MetaAgent [77]. An offline-trained classifier was then used to

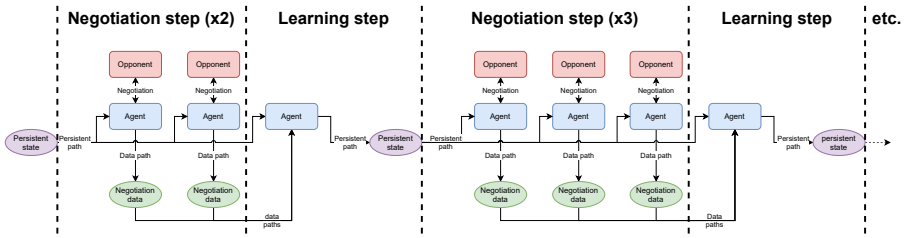

Figure 6.1: ANL tournament flow for the 2021 edition. The learning and negotiating phases were strictly separated. As agents were killed after execution, strategy-specific settings were stored in a persistent state that was fed to the agent at initialisation. During the negotiation phase, agents were allowed to store observation data, but not to update their strategy settings. Learning from this data and modifying the strategy settings of the agent was only allowed during the learning phases.

select an agent based on manually designed features of the negotiation scenario and the opponent's first few observations. This agent was also submitted to ANAC 2013.

6.3 ANL 2021

The ANL is intended to stimulate the advance of research in automated negotiation. Every year, a specific topic on the research agenda is chosen as the basis for the challenge. As mentioned before, we challenged the participants to learn in repeated negotiation games, where we tried to restrict the agents as little as possible in their learning methods. The 2021 and 2022 editions of ANL were organised around this topic.

In the 2021 edition, we decided to use a fixed set of negotiation scenarios such that every agent would encounter every other agent on the exact same set. This makes the competition fair in the sense that the impact on agent performance of using a different set of scenarios per agent is eliminated. We designed a complex data-saving structure that participants could use for learning purposes. The complexity was required to prevent potential unfair play caused by repeated use of the same negotiation scenarios.

All submitted agents would negotiate against each other on identical negotiation scenarios concurrently. Without restrictions, this could lead to unfair play by agents saving data on the negotiation scenario that they would face once more against another agent. The designed competition flow provides us control of the location where agents could save their data. Only after all negotiation sessions on the same scenario finished, the agents were given access to their data and a chance to use it for learning and changing its behaviour. This mechanism made the agents blind to their history until we allowed them to access it and thus prevented unfair play; it is visualised in Figure 6.1.

In retrospect, this structure added too much complexity to the competition for participants, causing a lower-than-expected number of submissions. Moreover, only few of the submissions implemented a learning mechanism, failing the goal of the competition. The ANL 2021 edition received 8 submissions, of which 2 imple-

mented a learning mechanism.[1] This prevented us from performing a meaningful analysis of such learning negotiation agents. Our main insight from ANL 2021 was to keep future editions as simple as possible from the perspective of participants.

In the 2022 ANL edition, we simplified the rules by allowing agents to save and load data files in a specifically provided directory without any restrictions. This resulted in more participants. Fair play was ensured by never repeating negotiation scenarios, while the number of negotiation rounds played was massively scaled up to minimise the stochastic impact caused by the randomly generated negotiation scenarios. We also moved from Java to Python as the default implementation language in order to allow for the use of the plethora of machine learning packages available in Python.

6.4 COMPETITION SETUP OF ANL 2022

Participants of ANL must design and submit an agent that can perform bilateral negotiation with other submitted agents following a finite-horizon Alternating Offers Protocol (AOP) [132]. We used GeniusWeb[2] as a platform for the negotiations, which is a software package that was specifically designed as a test-bed for agent-based negotiation.

This section describes the competition setup and the specifics of the negotiation games that are played between agents.

6.4.1 NEGOTIATION SCENARIO

The negotiation scenarios used in this competition follow the format described in Section 2.1. The utility functions are considered private information to the agents, making the negotiation an imperfect information game and exclusively categorical issues are used.

RANDOM GENERATION

In editions of ANL prior to 2021, the negotiation scenarios used to be manually designed. However, the challenge we set this year includes never repeating a negotiation scenario, which requires many such scenarios. The negotiation domains and utility functions in ANAC 2022 were randomly generated to accommodate this.

Outcome space To generate the negotiation scenarios, a goal outcome space size between 200 and 10 000 is sampled uniformly at random. The number of issues between 4 and 10 is also randomly sampled uniformly. Finally, the number of values per issue must be set so that the product of the number of values per issue is close to the goal size of the outcome space. This is done by distributing the number of values per issue according to a Dirichlet distribution, which creates a vector of random values summing up to 1. The probability density function of the Dirichlet distribution is

$$f(x_1, \cdots, x_n \mid \alpha_1, \cdots, \alpha_n) = \frac{1}{C} \cdot \prod_{i=1}^{n} x_i^{\alpha_i - 1} \tag{6.1}$$

[1] https://tracinsy.ewi.tudelft.nl/pubtrac/GeniusWebThirdParties
[2] https://ii.tudelft.nl/GeniusWeb/

where C is a normalising constant. We set all parameters α_i in this distribution to 1, i.e. $(\alpha_1, \cdots, \alpha_n) = \mathbf{1}_n$ such that the individual values are sampled from a uniform distribution. The full generation method is provided as pseudocode in Algorithm 6.1.

Algorithm 6.1 Outcome space generation

$g \leftarrow$ random integer $\in [200, 10\,000]$
while true **do**
  $m \leftarrow$ random integer $\in [4, 10]$
  $\boldsymbol{x} \leftarrow Dirichlet(\mathbf{1}_m)$
  $\boldsymbol{x} \leftarrow \boldsymbol{x} \cdot \left(\frac{g}{\prod_{b=1}^{m} x_b}\right)^{\frac{1}{m}}$ ▷ After 5: $\prod_{b=1}^{m} x_b = g$
  for $b \leftarrow 1$ to m **do**
    $|\Omega_b| \leftarrow \max\{round(x_b), 2\}$
  if $\prod_{b=1}^{m} |\Omega_b| - g < 0.1 \cdot g$ **then**
    break
$\Omega_1, \cdots, \Omega_m \leftarrow create\_values(\{|\Omega_1|, \cdots, |\Omega_m|\})$
$\Omega \leftarrow \{\Omega_1 \times \cdots \times \Omega_m\}$
return Ω

Utility functions Only bilateral negotiations are considered in this competition, so two utility functions that express preferences over the outcome space must be generated. The utility is obtained through a linear weighted sum of the values per issue, with weight factors $w(b)$ for every issue. These weights are again sampled from a Dirichlet distribution parameterized by $(\alpha_1, \cdots, \alpha_m) = \mathbf{1}_m$. Finally, for each issue b, the scores of the values ω_b within that issue are also sampled from a Dirichlet distribution and scaled to the range $[0, 1]$. These scores are expressed through the value weight function $w_b(\omega_b)$.

6.4.2 ANL 2022 CHALLENGE

In 2022, all submitted agents repeatedly negotiated against each other in one-on-one negotiation sessions with a deadline of 60 seconds in wall clock time to ensure a finite horizon. Failing to reach an agreement resulted in 0 utility for both agents involved in the negotiation. The negotiation scenarios were randomly generated and are likely always different in terms of size, number of issues, and utility functions.

The challenge in 2022 was to learn from previous encounters with other agents. The name of the opponent was made known to the agent. Agents were allowed to save any data files in a provided directory while encountering every other submitted agent 50 times throughout the tournament. One challenging part was effectively using information extracted from previous encounters with the same opponents, while the negotiation scenarios changed between each negotiation session.

6.4.3 EVALUATION

Agents were ranked based on two performance measures: individual utility and social welfare, both averaged over all negotiation sessions. Social welfare is the sum

Table 6.1: Computing hardware and resources per negotiation session.

Description	Type	Quantity
CPU	Intel® Xeon® CPU E5-2620 v4	2 cores
Memory	RDIMM DDR4-2400	10GB
OS	CentOS 7.9.2009	-

of utilities obtained by both agents involved in the negotiation session and is thus identical for both agents. Prize money was to be awarded to the two best-performing agents according to each of these performance measures. This resulted in multiple optimisation criteria for the participants of this competition. Maximising individual utility is selfish, and maximising social welfare could be considered social.

6.4.4 SIMULATION SPECIFICS

Every submitted agent encountered every other agent 50 times sequentially. For each negotiation session, a new negotiation scenario was generated randomly (see Section 6.4.1). As preferences over the negotiation domains can be unbalanced and might favour one of the agents, we decided to repeat the full tournament once more while switching the utility functions. The storage directory of every agent was completely erased when we restarted the tournament with switched utility functions to rule out the possibility of foul play. To further reduce the stochastic influence in the results, we repeated the previously mentioned procedure 5 times for the competition and 25 times for the analysis provided later in this article. The 19 submissions received required us to run a total of $19 \cdot 18 \cdot 50 \cdot 5 = 85\,500$ and $19 \cdot 18 \cdot 50 \cdot 25 = 427\,500$ negotiation sessions, respectively.

The sessions were run parallelized on a compute node. Details of the hardware and resources per session can be found in Table 6.1. The speed of the system is important as negotiation sessions are run with a wall-clock deadline. We ensured that no agent would ever face the same opponent concurrently so that the sequential encounter requirement of our challenge was satisfied. The participants were notified of potential file race issues due to parallel negotiation sessions and were suggested to save files based on the name of their current opponent to avoid this.

6.5 SUBMISSIONS TO ANL 2022

An overview of the submissions is provided in Table 6.2. The competition received a total of 20 submissions, of which 1 was invalid, resulting in a total of 19 agents that participated in the competition. The code of these agents can be found on the GeniusWeb webpage[3].

6.5.1 LEARNING CAPABILITIES

As mentioned earlier, the challenge was designed to encourage participants to develop learning methods implemented by providing the agents with a directory

[3] https://tracinsy.ewi.tudelft.nl/pubtrac/GeniusWebThirdParties

Table 6.2: Overview of agents that were submitted to ANL 2022

Name	Affiliation	Learning
Agent007	Bar Ilan University	
Agent4410	College of Management Academic Studies	
AgentFish	Tokyo University of Agriculture and Technology	
AgentFO2	Tokyo University of Agriculture and Technology	x
BIUagent	Bar Ilan University	
ChargingBoul	University of Tulsa	x
CompromisingAgent	Bar Ilan University	x
DreamTeam109Agent	College of Management Academic Studies	x
GEAAgent	College of Management Academic Studies	
LearningAgent	Bar Ilan University	x
LuckyAgent2022	Babol Noshirvani University of Technology	x
MiCROAgent	IIIA-CSIC	
PinarAgent	Siemens	x
ProcrastinAgent	University of Tulsa	x
RGAgent	Bar Ilan University	
SmartAgent	College of Management Academic Studies	x
SuperAgent	Bar Ilan University	x
ThirdAgent	College of Management Academic Studies	
Tjaronchery10Agent	College of Management Academic Studies	x

to save and load data. When we refer to learning, we mean changing behaviour between sessions based on previously recorded information. Preference estimation of an opponent within a negotiation session could also be considered learning, but we do not refer to it as such in this article. As opponents are repeatedly encountered during the competition, observations about their past behaviour could be exploited to improve negotiation capabilities. However, not all submitted agents implemented such a mechanism, making their strategies single-shot-based. Table 6.2 indicates which agents implemented a learning mechanism using the storage location to save data. As can be seen, more than half of the agents actually implemented a learning mechanism. We were successful in designing a competition that enables participants to actually implement such a mechanism. The effects of the implemented learning mechanisms are studied in more detail in Table 6.6.2.

6.5.2 SUBMITTED AGENT STRATEGIES

This section describes the strategies of selected agents: AgentFO2, DreamTeam109Agent, SuperAgent, Tjaronchery10Agent, and MiCROAgent. In general, the behaviour of the submitted agents can be considered a black box due to heavy manual design and parameter tuning. These agents were selected because they implemented an intuitively describable mechanism, and the respective participants submitted a report with their agent code. We summarize the main components based on these reports, which can be found in the repository of submitted agents[4].

[4] https://tracinsy.ewi.tudelft.nl/pubtrac/GeniusWebThirdParties

AgentFO2 This agent tries to reason over the opponent based on the Hamming distance between offers that it received. It classifies opponents as time-dependent converters, i.e., agents that concede based on the time towards the deadline, random agents, or other strategies. It then applies a time-dependent strategy while changing the minimum utility goal depending on the classified opponent strategy. This minimal utility to aim for is based on historical observations of the opponent.

DreamTeam109Agent This agent focuses on obtaining high utility first, and if that does not work, tries to minimize negotiation sessions that end in no agreement. It keeps track of the speed of the opponent in the past and tries to estimate how many rounds still can be played. If it is likely that this is the last round, then it simply accepts the offer. It also maintains a percentage of top outcomes it accepts per opponent and increases this percentage if past sessions result in low utility. The reasoning is that a low utility could result from the agent accepting a bad offer and that utility could be improved if it was more lenient towards the opponent in accepting offers with higher utility but were out of the top percentage pool.

SuperAgent This agent splits the negotiation session into timeslots and saves the average self-utility and average estimated opponent utility of all received offers in this timeslot for future use. It uses these values as utility thresholds for generating offers in the corresponding timeslot by demanding the average obtained utility as a minimum threshold and making offers above the opponent's threshold at the end of the session. The friendly behaviour at the end of the negotiation session is randomized to prevent opponents from exploiting it.

Tjaronchery10Agent This agent also adopts a time-dependent conceding strategy but does not concede in the first few encounters with an opponent. It attempts to force opponents to accept bad offers from them by being a hardliner. However, if this strategy does not appear fruitful after 3 sessions with an opponent, the strategy towards this opponent is modified to be slightly more conceding. These modifications can be repeated.

MiCROAgent MiCROAgent is an implementation of the recently introduced MiCRO strategy [40]. It is a very simple strategy that employs no form of learning or opponent modelling. It sorts all possible outcomes in order of decreasing individual utility and then proposes them in this order as long as the opponent also keeps making new proposals. That is, whenever the opponent makes a *new* proposal, MiCRO replies by proposing the next offer from its list. Whenever the opponent repeats an offer it has already made before, MiCRO replies by also proposing a (random) offer it has already proposed before. MiCRO accepts an offer from the opponent when that offer is better than or equal to the next offer that MiCRO will make. The idea is that it is a tit-for-tat-like strategy that assumes no knowledge about the utility function of the opponent. The agent always makes the smallest possible concession whenever it notices that the opponent is making a concession,

Table 6.3: Top 5 agents of the submitted agents to the ANL. Both prize categories are displayed. Here, individual utility is the average individual utility that agents obtained over all their negotiation sessions, and social welfare is the sum of the utilities of agents averaged over all negotiation sessions. Note that these are the original competition results, which differ from the results in the paper for reasons described in Section 6.6.1.

Agent	Individual Utility	Rank
DreamTeam109Agent	0.7247	1st
ChargingBoul	0.7238	2nd
SuperAgent	0.7040	3rd
CompromisingAgent	0.6857	4th
RGAgent	0.6819	5th

Agent	Social Welfare	Rank
DreamTeam109Agent	1.4605	1st
Agent007	1.4564	2nd
CompromisingAgent	1.4563	3rd
AgentFish	1.4396	4th
Agent4410	1.3993	5th

regardless of the magnitude of that concession. A negotiation between two such agents guarantees a Pareto-efficient agreement.

6.6 RESULTS & ANALYSIS

In this section, the agents are thoroughly empirically evaluated using multiple approaches. The 2021 edition results of ANL showed that agent scores depend on the other submitted agents, which we explore further. We answer the question of the influence of the learning mechanism on the performance of the negotiating agents, as this was the ANL challenge of the competition. Finally, we perform a game theoretical analysis of the competition and see how that relates to the official results of the competition.

6.6.1 DIFFERENCES TO ACTUAL COMPETITION

The results presented in this section are not fully in line with the actual results of the competition. The "LuckyAgent2022" was underperforming during the competition because of bugs. We allowed a resubmission of this agent to be included in this article. As this also affects the performance of other agents, the ranking of agents differs slightly compared to the official ranking presented after the competition. Finally, the entire competition was rerun to gather additional results that we use in our analysis presented in the following. The actual competition winners can be found in Table 6.3.

6.6.2 TOURNAMENT RESULTS

The top part of Figure 6.2 shows the tournament results. The agents are sorted based on the average utility obtained during the tournament, where the leftmost agent is the best-performing agent. A more elaborate results table is provided in Table 6.4. Notice that there are no large differences between the individual utility scores, as the difference between the maximum and minimum scores is only 0.2. The difference in social welfare score is much more apparent, with a maximum difference of nearly 0.6. We emphasise that most of the top-performing agents implemented a learning mechanism.

The results also generally show a higher social welfare score for agents with higher individual utility. Still, it is not evidently true that a higher individual utility

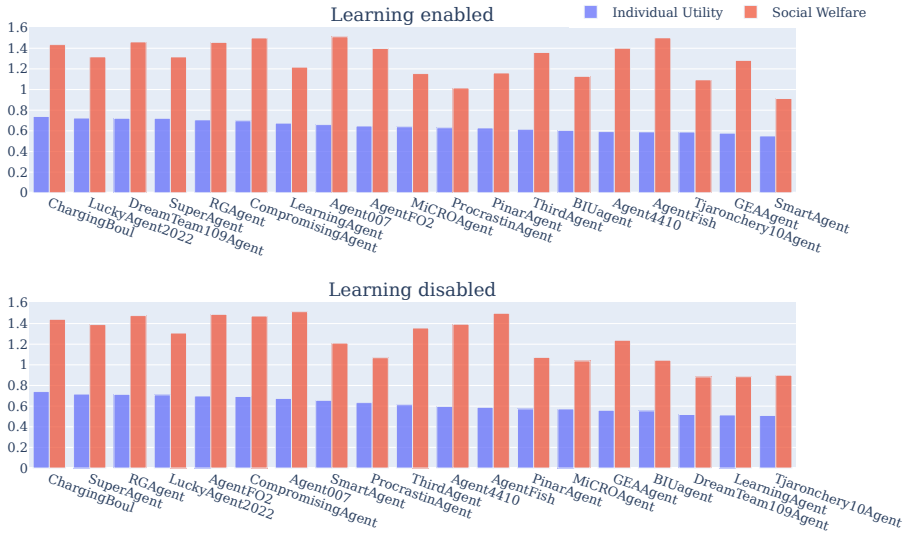

Figure 6.2: Results of two tournaments where learning is **enabled** (top) and learning is **disabled** (bottom). Both individual utility and social welfare are presented and agents are sorted from highest individual utility to lowest.

also leads to higher social welfare. Table 6.4 shows that the top-performing agent in utility fails to reach an agreement in more than 10% of the negotiation session. This clearly indicates wasted potential, as finding an agreement will always result in a higher utility than finding no agreement. On the contrary, the two highest-scoring agents in social welfare have a near 100% success rate in finding agreements yet do not obtain the highest utility. Being more selfish apparently leads to a higher utility at the cost of a lower agreement ratio. Paradoxically, a lower agreement ratio also, in turn, leads to lower utility. In terms of our research question, our empirical results indicate that agents that perform well in utility also perform well in social welfare, but that top performance in one of the categories tends to correlate with a lower score in the other.

IMPACT OF LEARNING

To determine to what extent the ability to learn influences the performance of the agents, we ran another experiment in which the agents' learning capability was disabled. This is achieved by emptying the storage directory of the agents after every negotiation, returning them to their initial state. The agents have no knowledge about previous negotiation sessions when initialised for all negotiation sessions. The results are found in the lower part of Figure 6.2.

The utility and ranking of every agent are different compared to the tournament where learning is enabled. This is more apparent for some agents, e.g. DreamTeam109Agent. We visualise this difference in ranking between a tournament where learning is enabled and a tournament where learning is disabled in Figure 6.3. We see that the top four highest-ranking agents when learning is enabled

Table 6.4: More results from the Automated Negotiation League (ANL) where **bold** is best and underline is worst. The results are averaged over all the negotiation sessions that each agent participated in. Distances are calculated by taking the Euclidian distance between the utilities of two outcomes. The Pareto front is the set of outcomes that are not strictly dominated by other outcomes in terms of utility. The Nash bargaining solution [111] is the outcome that maximizes the product of utilities. The Kalai-Smorodinsky bargaining solution [83] is the outcome on the Pareto front that is closest to equal utility.

Agent	Learning	**Individual Utility**	**Opponent Utility**	**Nash Product**	**Social Welfare**	**Number of Offers**	**Distance to Pareto Front**	**Distance to Nash Bargaining Solution**	**Distance to Kalai-Smorodinsky Bargaining Solution**	**Distance to Max Social Welfare**	**Agreement Ratio**
ChargingBoul	x	**0.739**	0.696	0.570	1.435	4286	0.113	0.275	0.261	0.282	0.891
LuckyAgent2022	x	0.724	0.593	0.522	1.317	2500	0.186	0.357	0.342	0.362	0.814
DreamTeam109Agent	x	0.721	0.740	0.540	1.461	**4733**	0.069	0.306	0.294	0.311	0.945
SuperAgent	x	0.721	0.595	0.522	1.316	4669	0.185	0.361	0.347	0.365	0.812
RGAgent		0.707	0.750	**0.593**	1.457	1036	0.111	**0.246**	**0.240**	**0.252**	0.886
CompromisingAgent	x	0.698	0.802	0.567	1.500	997	0.051	0.277	0.271	0.282	0.953
LearningAgent	x	0.675	0.542	0.487	1.217	1026	0.248	0.425	0.410	0.429	0.741
Agent007		0.660	0.851	0.549	**1.511**	2144	**0.036**	0.280	0.272	0.285	0.990
AgentFO2	x	0.647	0.751	0.553	1.398	2448	0.137	0.298	0.282	0.305	0.873
MiCROAgent		0.640	0.515	0.464	1.155	4620	0.291	0.451	0.426	0.458	0.709
ProcrastinAgent	x	0.630	0.385	0.364	1.015	3792	0.336	0.598	0.576	0.601	0.658
Pinar_Agent	x	0.628	0.532	0.467	1.161	3865	0.292	0.438	0.407	0.446	0.718
ThirdAgent		0.615	0.744	0.530	1.359	915	0.154	0.328	0.310	0.335	0.858
BIU_agent		0.604	0.523	0.461	1.128	1245	0.316	0.459	0.438	0.465	0.682
Agent4410		0.593	0.807	0.522	1.400	1195	0.114	0.336	0.322	0.340	0.900
AgentFish		0.590	**0.911**	0.530	1.501	1894	0.033	0.300	0.287	0.304	**0.997**
Tjaronchery10Agent	x	0.589	0.505	0.425	1.094	775	0.319	0.515	0.496	0.518	0.681
GEAAgent		0.577	0.705	0.497	1.282	31	0.207	0.375	0.360	0.379	0.814
SmartAgent	x	0.551	0.362	0.344	0.913	1238	0.418	0.644	0.620	0.647	0.574

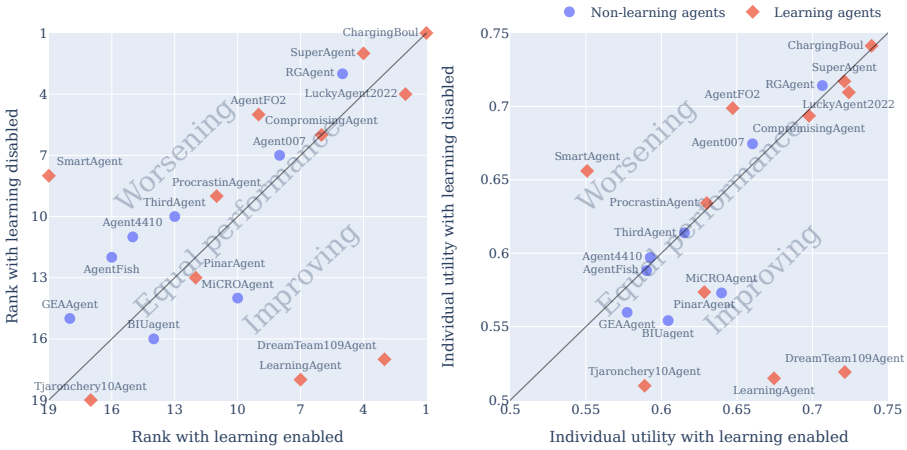

Figure 6.3: Ranking (left) and individual utility (right) comparison of a tournament where learning was disabled and one where learning was enabled. Agents in the lower right improved their ranking when learning was enabled.

are, in fact, agents that learn between sessions. One of them actually made a significant jump from 17th position to 3rd. On the contrary, we also see that SmartAgent performed significantly worse when learning was allowed. Finally, one might expect the non-learning agents to be on the equal performance line as their behaviour is agnostic to disabling the learning capabilities. However, their individual utility is affected by opponents being more capable of finding agreements with them.

PERFORMANCE CONVERGENCE OF LEARNING AGENTS

One problem when evaluating a group of learning agents is that their strategies continuously change, and their scores may not converge. This makes a given ranking dependent on the number of iterations a tournament is run, impacting its robustness. We analyse whether this behaviour can indeed be observed for the agents that were submitted to the competition. To do so, instead of the competition tournament of 50 rounds, a tournament of 1,000 rounds was run for a total of 342 000 negotiation sessions. We report the moving average of the individual utility of the agents against the number of rounds played in Figure 6.4. The window size used is 100 rounds to smooth out the stochastic influence of the random negotiation scenario generator.

Figure 6.4 shows that the ranking and individual utility of the agents keeps changing in the later rounds but that the differences are minor and reasonably stable, but still influence the ranking. Before round 400, differences are more pronounced. We also clearly see a difference between agents with and without learning mechanisms, where the latter exhibit more stable behaviour. The learning mechanisms do not always work out to the benefit of the agents, which we also saw in Figure 6.3; especially AgentFO2 stands out in terms of worsening performance as the rounds progress.

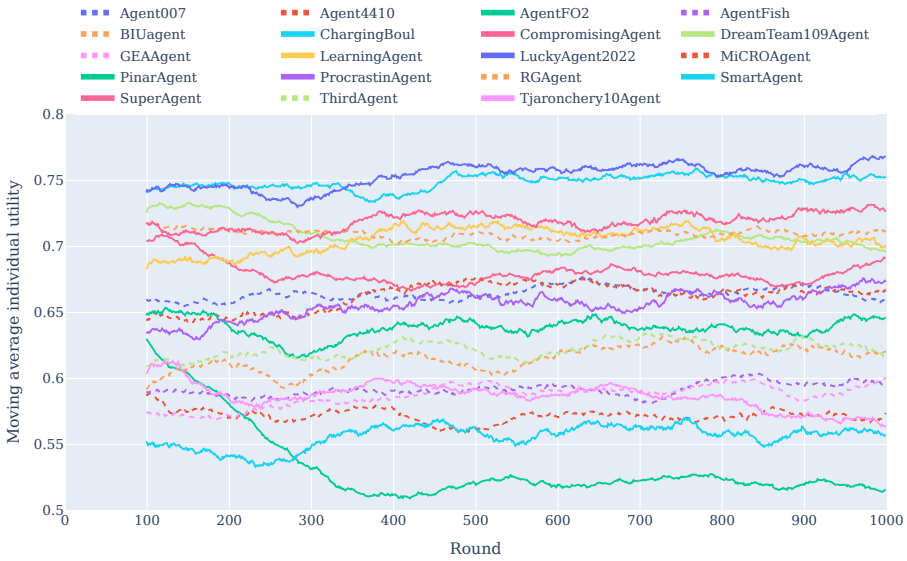

Figure 6.4: Results of a tournament of 1 000 rounds. The moving average of the individual utility with a window size of 100 rounds is visualised. Agents without a learning mechanism are indicated with a dotted line.

6

IMPACT OF GROUP COMPOSITION

We observed during this competition that the composition of the group of agents also influences the final ranking. To explore this further, we evaluated the performance of the submitted agents in every possible group of minimum size 2. Out of the 19 submitted agents, we created all possible 524 268 sub-tournaments (Equation 6.2). Separately running all these tournaments would have been computationally intractable, so we obtained the results naïvely by filtering the result from a full tournament. To the best of our knowledge, none of the participating agents' behaviour is influenced by this naïve approach, as the agents only reason about the current opponent they are facing and their history with that agent. We counted the ranking of every agent in all of the sub-tournaments for both individual utility and social welfare. Both results are plotted in Figure 6.5.

$$\sum_{i=2}^{19} \binom{19}{i} = 524\,268 \tag{6.2}$$

As we can see in the heatmaps, there was a chance for all submitted agents to win the tournament, depending on which opponents were also submitted. This chance was low, but greater than zero, for the agents that obtained a low ranking in the full tournament. This observation is more pronounced for the higher-ranking agents, as chances to win a sub-tournament are much more similar. This means that, at least in part, winning the competition was a matter of chance, depending on the submitted opponents.

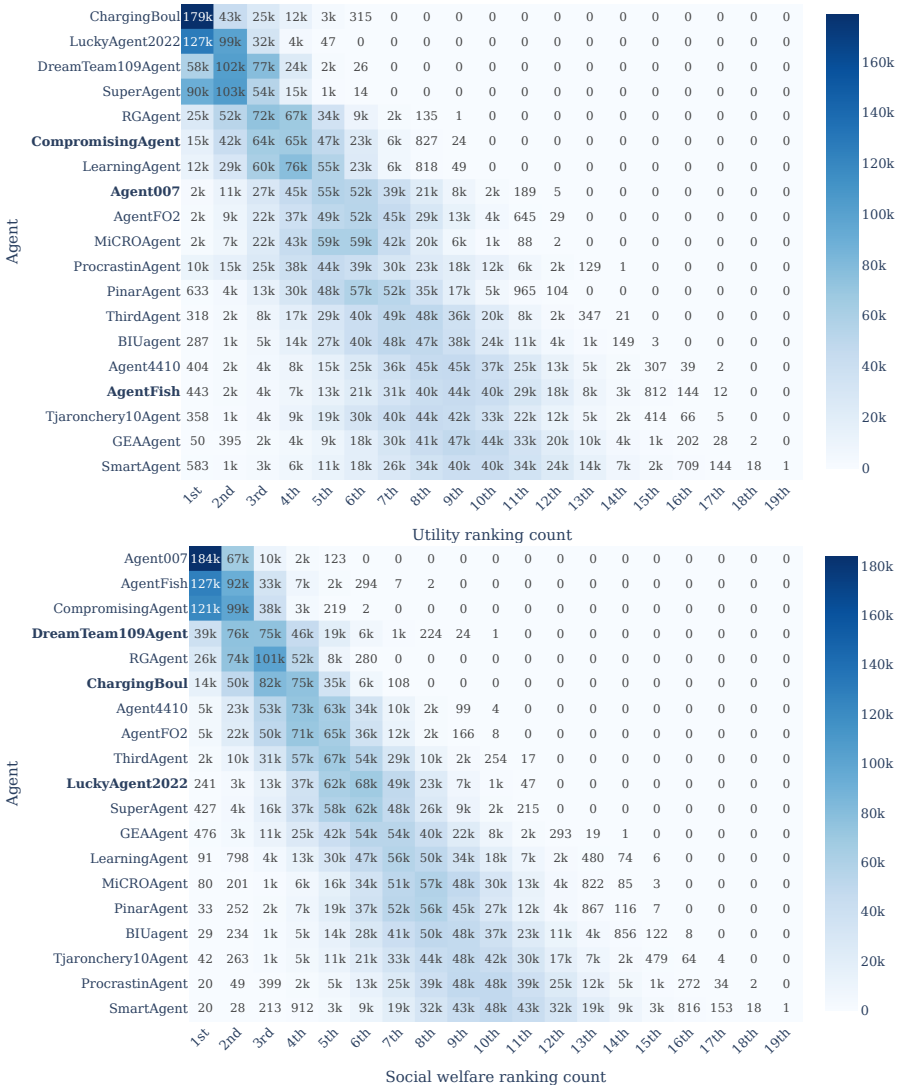

Figure 6.5: Heatmap of the number of times an agent obtained a certain rank in average individual utility (top) and social welfare (bottom). Results are counted over a total of 524 267 tournament setups that could be created with the submitted agents. The agents are sorted based on their ranking in average individual utility (top) and social welfare (bottom) in Figure 6.2. The top three agents in the other category are boldfaced.

In fact, simply averaging the performance of the agents provides a reasonable ranking under the assumption that all opponents are equally likely to be encountered. One could argue that this is unlikely to be the case as, especially after obtaining these tournament results, the underperforming agents are not likely to be used. This degrades the value of the obtained ranking. Therefore, we take a step towards analysing the performance of the agents beyond average scores in a tournament.

6.6.3 GAME-THEORETICAL ANALYSIS OF THE AGENTS

In the following, we evaluate the agents from a more game-theoretical point of view through empirical game theory analysis [158]. Specifically, we construct a meta-game of the underlying negotiation game where agents must select one of the ANL 2022 competition agents to negotiate on their behalf. This would automatically mitigate the previously described issue that underperforming agents are unlikely to be used in practice and could be more in line with a realistic scenario. We analyse which agents are likely to be picked and whether Nash equilibria can be found in such a meta-game.

This analysis assumes that agents are perfectly rational, which may be too strong an assumption for real-world applications. However, our previous evaluation method is also based on an unrealistic assumption that all opponents are equally likely to be encountered. We argue that tournament evaluation and game-theoretical evaluation both have their advantages and their disadvantages. For the same reason, other authors also performed game-theoretical evaluations [160, 10, 28].

We averaged the bilateral result of every agent against every opponent separately and combined these results into a matrix. This matrix can then be seen as the payoff matrix of a symmetric normal-form game in which the two players choose one of the agents as their strategy. Note that, in order to obtain the full matrix, we also need to have, for each agent, the score it would obtain when negotiating against itself, while the ANL 2022 tournament did not involve self-play. We therefore repeated the tournament, but this time including self-play, to obtain those scores. The payoff matrix U we obtained is displayed in Table 6.5. For readability, we multiplied the scores and their standard errors by 1000. Each entry $U_{A,B}$ represents the average individual utility obtained by agent A when playing against agent B (averaged over 2500 negotiation sessions).

NASH EQUILIBRIA

The meta-game has two pure Nash equilibria: SuperAgent against SuperAgent $U_{4,4}$, and MiCROAgent against MiCROAgent $U_{10,10}$. Of these two equilibria, MiCROAgent against MiCROAgent achieves the highest payoff for both players and is therefore preferred.

We performed several statistical tests to verify critical results in Table 6.5. First, we verified both pure Nash equilibria by checking whether the respective agent playing against itself actually results in the highest average individual utility. That is, for each agent $B \in \{\text{MiCROAgent}, \text{SuperAgent}\}$ we performed a one-sided Welch t-test against the null hypothesis that $\bar{U}_{A,B} \geq \bar{U}_{B,B}$ for each opponent A (with $A \neq B$)

Table 6.5: Results of game-theoretical evaluation. Each cell displays the average score of the agent indicated in the row header, along with its standard error, obtained against the agent in the corresponding column. The scores and standard errors are multiplied by 1000 for readability. In each column, the highest score is indicated in boldface.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**
1 ChargingBoul	809	674	705	684	863	798	602	881	884	764	490	**725**	**918**	622	**978**	943	560	785	427
	± 3	± 5	± 5	± 5	± 3	± 3	± 7	± 2	± 3	± 4	± 6	± 5	± 3	± 8	± 1	± 2	± 7	± 6	± 7
2 LuckyAgent2022	813	671	828	690	832	929	605	935	808	526	499	593	861	511	966	964	500	777	397
	± 5	± 7	± 4	± 7	± 5	± 2	± 8	± 1	± 6	± 8	± 8	± 8	± 5	± 9	± 2	± 1	± 8	± 7	± 8
3 DreamTeam109Agent	797	**698**	743	744	852	916	**659**	955	732	623	435	619	821	478	946	957	569	739	445
	± 5	± 5	± 5	± 5	± 4	± 3	± 5	± 1	± 7	± 5	± 6	± 6	± 5	± 7	± 2	± 1	± 6	± 5	± 6
4 SuperAgent	816	674	802	**767**	776	929	598	954	862	607	**547**	555	766	502	955	966	534	737	402
	± 5	± 7	± 4	± 5	± 7	± 2	± 8	± 1	± 5	± 7	± 7	± 8	± 7	± 9	± 2	± 0	± 9	± 8	± 8
5 RGAgent	761	679	736	619	817	746	589	864	803	591	515	692	851	663	830	930	**573**	**811**	**465**
	± 3	± 5	± 3	± 6	± 3	± 2	± 6	± 2	± 5	± 7	± 6	± 6	± 3	± 7	± 4	± 1	± 7	± 5	± 7
6 CompromisingAgent	787	644	610	641	**879**	790	585	807	**892**	613	395	640	819	**694**	791	980	558	792	434
	± 4	± 4	± 5	± 5	± 3	± 5	± 5	± 4	± 3	± 5	± 5	± 6	± 5	± 5	± 6	± 0	± 6	± 6	± 6
7 LearningAgent	733	623	885	630	759	957	620	985	808	426	444	506	668	525	645	979	452	734	382
	± 8	± 8	± 3	± 8	± 8	± 2	± 8	± 1	± 7	± 8	± 8	± 9	± 9	± 9	± 9	± 0	± 8	± 8	± 8
8 Agent007	702	631	529	573	780	797	462	791	821	622	393	651	698	650	907	931	558	760	424
	± 3	± 5	± 5	± 5	± 3	± 3	± 5	± 5	± 3	± 5	± 5	± 4	± 6	± 6	± 3	± 2	± 6	± 5	± 5
9 AgentFO2	724	631	565	618	726	682	565	746	734	745	474	631	847	557	799	850	549	553	389
	± 3	± 5	± 6	± 5	± 5	± 4	± 6	± 2	± 5	± 4	± 5	± 6	± 3	± 8	± 6	± 2	± 7	± 8	± 7
10 MiCROAgent	**822**	577	865	657	728	935	500	916	847	**820**	104	497	719	336	894	875	383	655	206
	± 4	± 9	± 5	± 8	± 8	± 2	± 9	± 1	± 3	± 2	± 6	± 9	± 7	± 9	± 2	± 2	± 9	± 9	± 8
11 ProcrastinAgent	683	587	916	688	747	**981**	540	**995**	781	85	377	87	605	322	874	**1000**	444	733	272
	± 8	± 9	± 5	± 8	± 8	± 2	± 9	± 1	± 8	± 5	± 9	± 5	± 10	± 9	± 7	± 0	± 10	± 8	± 8
12 PinarAgent	750	582	830	555	762	744	512	880	784	488	90	384	758	432	857	881	433	704	267
	± 5	± 8	± 5	± 8	± 6	± 6	± 9	± 2	± 5	± 8	± 5	± 9	± 7	± 9	± 4	± 2	± 9	± 7	± 8
13 ThirdAgent	614	569	683	553	739	693	515	595	753	612	363	598	648	671	708	731	524	730	422
	± 4	± 5	± 5	± 6	± 4	± 5	± 7	± 5	± 4	± 7	± 7	± 6	± 6	± 7	± 5	± 4	± 8	± 6	± 7
14 BIUagent	641	503	744	512	716	851	531	896	627	308	321	416	712	420	798	898	478	703	227
	± 8	± 9	± 7	± 9	± 7	± 4	± 9	± 3	± 8	± 8	± 9	± 9	± 7	± 9	± 5	± 1	± 9	± 7	± 8
15 Agent4410	472	513	541	532	737	670	496	639	595	643	377	602	798	660	766	777	478	752	384
	± 5	± 5	± 5	± 5	± 5	± 6	± 7	± 5	± 6	± 5	± 5	± 5	± 5	± 6	± 4	± 4	± 8	± 6	± 7
16 AgentFish	540	532	504	509	644	493	519	646	708	715	389	638	696	614	796	772	543	700	438
	± 5	± 6	± 6	± 6	± 5	± 5	± 5	± 5	± 4	± 4	± 5	± 5	± 5	± 5	± 2	± 4	± 6	± 5	± 5
17 Tjaronchery10Agent	685	529	898	501	668	811	504	959	723	261	293	322	607	387	548	956	527	652	297
	± 8	± 9	± 4	± 9	± 8	± 7	± 9	± 1	± 8	± 7	± 7	± 8	± 9	± 8	± 9	± 1	± 8	± 8	± 7
18 GEAAgent	630	548	758	514	682	614	522	669	537	457	525	537	668	563	631	787	511	717	240
	± 5	± 6	± 4	± 6	± 5	± 5	± 6	± 5	± 7	± 7	± 6	± 6	± 5	± 7	± 6	± 3	± 7	± 5	± 7
19 SmartAgent	556	477	**921**	492	611	890	456	994	545	181	281	274	578	232	608	994	452	370	219
	± 9	± 9	± 5	± 10	± 9	± 6	± 9	± 1	± 9	± 7	± 8	± 8	± 9	± 8	± 9	± 0	± 10	± 9	± 8

and set a maximum significance level of $p = 0.05$ to reject the null hypothesis. Note that we use $\bar{U}$ to denote the *true* expected utility that agent A would obtain against agent B, whereas U represents the *measured* average individual utility. We found that the highest p-value for this hypothesis was $4 \cdot 10^{-34}$ for MiCROAgent and $6 \cdot 10^{-4}$ for SuperAgent. Each of these p-values still needs to be multiplied by 18, to take into account that for only one of the 18 opponents, the null hypothesis needs to be true to reject our conclusion, but then they still stay well below the threshold of $p = 0.05$, so our claims that MiCROAgent and SuperAgent both form a Nash equilibrium are statistically significant. Furthermore, we inverted this test to verify that none of the other agents forms a Nash equilibrium when playing against itself. That is, for each agent $B \notin \{\text{MiCROAgent}, \text{SuperAgent}\}$, and each agent A (with $A \neq B$) we performed a one-sided Welch t-test against the null hypothesis that $\bar{U}_{A,B} \leq \bar{U}_{B,B}$. Indeed, for each agent B we found at least one opponent A for which the p-value for this hypothesis was far below 0.05. Therefore, we can reject the hypothesis that $\bar{U}_{A,B} \leq \bar{U}_{B,B}$ for *all* opponents A.

Apart from pure Nash equilibria, we also found 21 *mixed* Nash equilibria using the Gambit software package (v.16.2) [134]. However, for each of these mixed equilibria, the payoff was lower than for the two pure equilibria. The top mixed equilibrium found has a probability of 66% for SuperAgent and 34% for MiCROAgent. In this mixed equilibrium, both players receive an expected utility of 0.712, which is significantly lower than the utility they would achieve if they both played SuperAgent (0.767) or if they both played MiCROAgent (0.820).

We note that MiCROAgent does not perform well in the tournament evaluation but does perform strongly in the game-theoretical evaluation. A quick analysis shows that MiCROAgent works particularly well against competitive opponents and less so against weaker opponents. As the game-theoretical approach depends on selecting the best possible response, it emphasises results obtained against stronger opponents. As mentioned earlier, the average scoring used in the tournament evaluation is based on the assumption that all opponents are equally likely to be encountered, which could be considered unrealistic. This also suggests that learning is especially beneficial in the presence of weaker agents that can be exploited. If only stronger agents are present, the learning agents may lose their advantage over a simpler approach such as MiCROAgent.

6.6.4 ANALYSIS OF THE NEGOTIATION SCENARIOS

The characteristics of the randomly generated negotiation scenarios can have a significant influence on the performance of the submitted agents [70]. We analyze the scenarios based on characteristics often used in the automated negotiation literature: the opposition, distribution, and balance scores described in the sections below. The cumulative distribution functions of these metrics for the negotiation scenarios used in this paper are visualised in Figure 6.6 and Figure 6.7. The maximum social welfare and maximum Nash product are included in Figure 6.6.

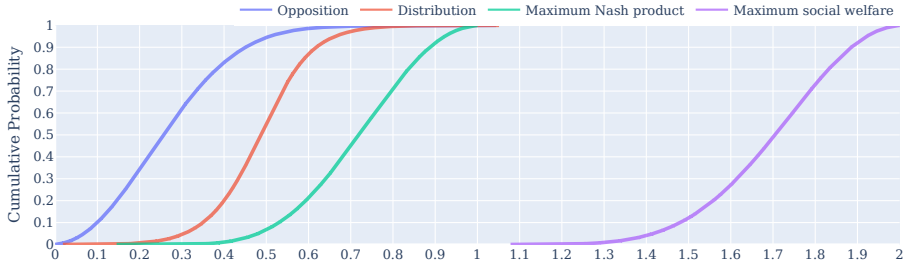

Figure 6.6: Cumulative distribution functions of the characteristics of the randomly generated negotiation scenarios used in this paper.

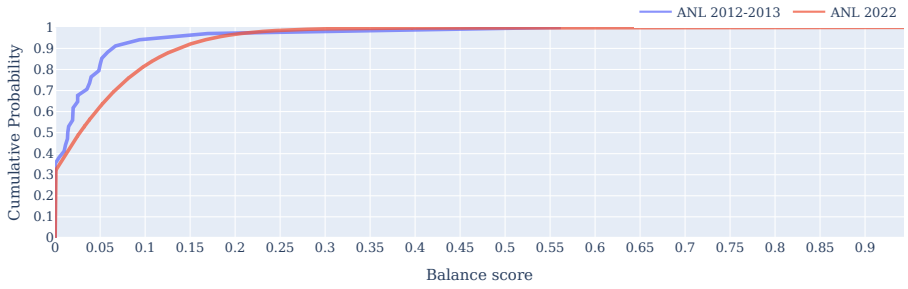

Figure 6.7: Cumulative distribution functions of the balance scores of the negotiation scenarios of the ANL 2012-2013 and ANL 2022 editions. The scenarios used for ANAC 2022 are less balanced than those used for ANAC 2012/2013. For example, we see that 80% of the ANAC 2012/2013 scenarios had a balance score less than or equal to 0.050, while among the ANAC 2022, this fraction was only 62%

Table 6.6: Average individual utility and success rate for all agents, compared between negotiation scenarios with low, medium and high opposition.

	low opposition $s_{opp} < 0.20$	medium opposition $0.20 \leq s_{opp} \leq 0.325$	high opposition $0.325 < s_{opp}$
Av. util. all sessions	0.814	0.647	0.471
Av. util. sessions with agreement	0.877	0.784	0.690
Agreement ratio	0.93	0.83	0.68

OPPOSITION

A commonly used measure to quantify the competitiveness of a negotiation scenario is the opposition value, which indicates how easy it is to find agreements that are satisfactory to both agents [10]. Before calculating the opposition value, the subset of Pareto efficient outcomes Ω_p must be extracted. An outcome is Pareto efficient if no other outcome improves the utility of at least one of the agents while not decreasing the utility for the others (Equation 6.3). From this Pareto-efficient set, we select the Kalai-Smorodinsky bargaining solution [83] (Equation 6.4) and use it to calculate the opposition based on Equation 6.5.

$$\Omega_p = \left\{ \omega \in \Omega \mid \neg\exists \omega' \in \Omega : \begin{bmatrix} (u_A(\omega') > u_A(\omega) \wedge u_B(\omega') \geq u_B(\omega)) \\ \vee \\ (u_A(\omega') \geq u_A(\omega) \wedge u_B(\omega') > u_B(\omega)) \end{bmatrix} \right\} \tag{6.3}$$

$$\omega_{Kalai} \in \arg\min_{\omega \in \Omega_p} |u_A(\omega) - u_B(\omega)| \tag{6.4}$$

$$s_{opp}(\Omega) = \sqrt{(1 - u_A(\omega_{Kalai}))^2 + (1 - u_B(\omega_{Kalai}))^2} \tag{6.5}$$

We split the negotiation scenarios into three roughly equal-sized categories with low, medium and high opposition values. The results of this analysis are displayed in Table 6.6. We observe that the lower the opposition values, the better the agents perform in terms of agreement rate and utility obtained from the agreement. Another interesting observation is that the opposition value has a noticeable influence on the ranking of the agents. Most notably, DreamTeamAgent109 ends in seventh place on the scenarios with low opposition, while it ends in first place on the scenarios with high opposition.

DISTRIBUTION

The same analysis was carried out for the distribution value, $s_{dist}(\Omega)$, which is defined in Equation 6.6; it is the average Euclidian distance in utility of every outcome to its closest Pareto-efficient outcome. The higher this value, the more difficult it becomes to find an agreement that is (close to) Pareto efficient. Estimating the preferences of the opponent accurately is essential in negotiation scenarios with a high distribution value.

$$s_{dist}(\Omega) = \sum_{\omega \in \Omega} \left[\min_{\omega' \in \Omega_p} \sqrt{(u_A(\omega) - u_A(\omega'))^2 + (u_B(\omega) - u_B(\omega'))^2} \right] \tag{6.6}$$

Table 6.7: Average individual utility and success rate for all agents, compared between negotiation scenarios with low, medium and high distribution values.

	low distribution $s_{dist}, 0.450$	medium distribution $0.450 \leq s_{dist} \leq 0.525$	high distribution $0.525, s_{dist}$
Av. util. all sessions	0.516	0.656	0.777
Av. util. sessions with agreement	0.720	0.793	0.857
Agreement ratio	0.72	0.83	0.91

The results are displayed in Table 6.7. Again, we clearly observe a difference in the performance of the agents depending on the distribution value. The higher the distribution value, the higher the score of the agents, both in terms of the quality of the deals made and the percentage of negotiations that end with a deal.

These results may initially seem surprising since we argued that finding Pareto-efficient outcomes in scenarios with high distribution is harder. However, if a negotiation scenario has a high distribution, the outcomes are also more scattered throughout the utility space and may, in turn, contain outcomes with high utility for both agents. A change in the ranking of the agents is also observed. Most notably, DreamTeamAgent109 ends in second place on scenarios with low distribution, while it ends in seventh place on scenarios with a high distribution.

BALANCE SCORE

Recent work stated that the negotiation scenarios used for ANAC 2012 and 2013, also used in many later editions of ANAC, were too simplistic [40]. They showed that many of these scenarios display a certain type of symmetry, which makes them easy to tackle by a naïve strategy called the MiCRO strategy (which also participated in ANL 2022, see Section 6.5.2). This was demonstrated by the fact that MiCRO was able to outperform some of the best agents in these scenarios, even though MiCRO is a much simpler strategy that does not apply any form of opponent modelling or learning [40].

To quantify this symmetry, De Jonge [40] defined the notion of the "balance values" of a negotiation scenario, which are the individual utilities of the outcome ω_b that two agents would agree upon if they both apply the MiCRO strategy. They showed that many of the ANAC 2012 and 2013 negotiation scenarios are "balanced", meaning that the balance values lie very close to the Nash Bargaining Solution [111] (NBS). De Jonge [40] argued that a more versatile set of negotiation scenarios should be used to test agents.

We performed the same analysis to see to what extent the scenarios used in ANAC 2022 are balanced. Similar to De Jonge [40], the balancedness of a negotiation scenario is measured by comparing the balance values to the utilities associated with the optimal outcome. However, we do not consider the NBS as the optimal outcome but rather argue that the optimal solution is the one that maximizes social welfare.

The motivation for using the Maximum Social Welfare Solution (MSWS) instead of the NBS as the optimal outcome is twofold. First, maximizing social welfare was explicitly one of the goals of the competition. Second, even if the goal is to maximize

individual utility, each scenario has two utility functions, u_A and u_B, randomly assigned to the two agents. Maximising the expected individual utility before being assigned a utility function would equal agreeing to the MSWS in advance (see Jonge [80] for a more detailed discussion).

$$s_{bal}(\Omega) = \max_{\omega \in \Omega} \{u_A(\omega) + u_B(\omega)\} - (u_A(\omega_b) + u_B(\omega_b)) \tag{6.7}$$

We define the *balance score* s_{bal} in Equation 6.7, where ω_b is the outcome that would be obtained by 2 MiCRO strategies. The utilities are assumed to be normalized to fall within the range [0,1]. The lower the balance score, the closer the balance values are to the MSWS. If the balance score is exactly 0, the balance values coincide with the MSWS, which in turn means that in such a negotiation scenario the agreement made between two MiCRO agents would always be exactly the MSWS.

The result of the analysis is shown in Figure 6.7. We observe that the negotiation scenarios of ANL 2022 are less balanced than those of ANAC 2012 and 2013 and should therefore be preferred for research. The fact that we see different values between the two types of scenarios can be explained by the fact that the ANAC 2012/2013 scenarios were handcrafted by participants, while the scenarios of ANL 2022 were randomly generated.

However, ANL 2022 scenarios still seem to display a high degree of balance. Around half of the scenarios have a balance score of 0.025 or less, and for around one-third of the scenarios, the balance score is exactly 0. It remains an open question why exactly the randomly generated scenarios are so balanced and how this compares to real-world negotiation scenarios.

We recalculated the scores of all the agents while only counting negotiation sessions involving negotiation scenarios with low balance scores ($s_{bal} = 0$), with medium balance scores ($0 < s_{bal} \leq 0.05$) and with high balance scores ($0.05 < s_{bal}$). The balance scores were chosen such that each category contained roughly one-third of all scenarios.

While we expected that MiCRO would perform better on the balanced scenarios, we noticed that this was actually the case for *all* agents. The agents make better and more agreements on the balanced scenarios, while the opposite is true for the unbalanced scenarios. The results are summarized in Table 6.8. The differences are lower than in Table 6.6 and Table 6.7, suggesting that the opposition value and the distribution value are better indicators of the level of difficulty of a scenario than the balance score.

Finally, not much difference was observed in the outcome of the tournament. The final ranking in the tournament evaluation remains more or less the same, with a few agents moving one or two positions up or down the ranking.

6.7 DISCUSSION

The extensive analysis we performed using the agents that were submitted to ANL 2022 led to insights into their behaviour that we will now discuss. We will discuss some general observations first, and after that, we will discuss two core topics more in-depth.

Table 6.8: Average individual utility and success rate for all agents, compared between negotiation scenarios with low, medium, and high balance scores.

	low balance score $s_{bal} = 0$	medium balance score $0 < s_{bal} \leq 0.050$	high balance score $0.050 < s_{bal}$
Av. util. all sessions	0.689	0.652	0.611
Av. util. sessions with agreement	0.817	0.796	0.774
Agreement ratio	0.84	0.82	0.79

Regarding the negotiation scenarios used, we observed that the randomly generated scenarios from ANL 2022 are slightly less balanced than the handcrafted scenarios used in many of the earlier editions of ANL. This indicates that the randomly generated scenarios are slightly more challenging than the handcrafted ones. We also showed that the agents performed better on the balanced scenarios, but this did not significantly influence the outcome of the tournament. On the other hand, the more classical notions, such as opposition and distribution of the outcome space, correlated more strongly with the performance of the agents.

We can calculate from Table 6.4 that 18.4% of the negotiation sessions ended without agreement. This is undesirable, as it lowers the social welfare obtainable by the group, which is to no one's benefit. However, competitive strategies that cause these failures to reach an agreement are beneficial for individual utility, which we observed more often in the history of ANL, and which is also a general observation in partially cooperative (general-sum) games, such as the Prisoner's Dilemma. It would, therefore, seem useful to investigate the design of mechanisms that push agents towards cooperating strategies [112].

6.7.1 LEARNING IN NEGOTIATION AGENTS

We have outlined and analysed agents that learn from repeated encounters with opponents and presented the results of a competition between such learning agents. The main focus of the competition was to assess the strength of algorithms that can learn from previous encounters in groups with other learning agents and use this knowledge to adapt their strategies to individual opponents. Since not every participant submitted an agent that implemented such a learning approach, we were able to compare the results of learning and non-learning agents and showed that the learning agents indeed performed better than the non-learning agents for the challenge set in the competition.

When the learning capabilities are disabled, some learning agents drop substantially in performance. While this suggests that learning is beneficial, it could also indicate that these agents became dependent on their learning mechanism. Ideally, agents should be more robust and perform well in single-shot and repeated encounters with opponents. This could have been achieved by evaluating agents in the competition based on their single-shot performance, which we believe is an interesting idea for future competitions.

A notable observation from this competition was that the agent that scored highest in the tournament, ChargingBoul, did not produce the best response against

itself. If one is to select one of the agents that participated in the competition, choosing the winner may not be the best option. Instead, we observed that two agents formed a pure Nash equilibrium when playing against themselves (MiCRO and SuperAgent), of which MiCRO scored the highest. This is remarkable, as MiCRO is a naïve strategy that does not use any form of learning or opponent modelling. However, while MiCRO was the strongest participant in the game-theoretical sense, it only ended in 10th place in the tournament evaluation. We can explain this based on the fact that MiCRO fails to exploit weaker opponents and/or fails to make agreements with them.

To our knowledge, none of the agents let the group composition influence their strategy. Still, it greatly impacts overall performance within a competition setting, and we believe that this could be an interesting topic for future research.

6.7.2 RANKING NEGOTIATION AGENTS

Our analysis of the competition results used various methods to analyse the performance of negotiation agents. One is based on average performance, the default method in the automated negotiation community, and one is based on Nash equilibria found using empirical game theory in a meta-game.

The ranking based on average performance strongly depends on the submitted agents, as shown in Figure 6.6.2. Surely, this is the case in any AI competition, but it is even more apparent in agent-versus-agent competitions. In such competitions, a single added agent influences the score of all the other agents instead of merely contributing another score that is added to the ranking. Intuitively, in a group of defecting strategies, a conceding strategy is needed to win a tournament, as some utility is better than no utility, and the defecting opponents will also not obtain agreements with each other. Conversely, a hardheaded strategy is needed to win in a group of conceding strategies, as it will exploit all opponents, obtaining the highest utility.

The game theoretic analysis did not give us a full ranking but a selection of strategies that form Nash equilibria in the meta-game. Such equilibria rely on flawless rational agents that all know the full structure of the game, which is a disputable assumption. The equilibria also depend on the strategy set included, but many more strategies will likely exist than were submitted to this competition. There can be more than one pure equilibrium, which makes it unclear which equilibrium is played. Equilibria can be unfair to one of the agents involved. The last two points apply, for example, to the game of chicken. All in all, there are many counterarguments for analysing agent performance based on Nash equilibria.

We attempted to find a ranking method that would be a natural fit for this competition but were unsuccessful. We considered Elo ranking, which assumes that relative skill is transitive and is sensitive to copies of the same strategy [21]; unfortunately, both of these assumptions are problematic for ranking negotiation agents. Nash Averaging [21] is also a popular ranking method but can only deal with zero-sum games and is sensitive to the set of included agents [91]. In contrast, our setting is a general sum game, and we attempt to avoid methods sensitive to the set of included agents. On the other hand, α-rank [114] is suitable for general

sum games. It uses replicator dynamics to find a dynamical solution concept and extracts a ranking based on that. However, the main motivation behind this method was computational tractability, which is not an issue in our case, and the obtained ranking depends on the α parameter setting. We attempted to obtain a stable ranking using this method but were unhappy with the sensitivity to the α parameter. We checked ranking methods based on social choice theory and voting [91], but these methods consider ordinal pairwise comparisons, where our meta-game is cardinal. Finally, the ranking method that we believe comes closest to our needs has been proposed by Marris et al. [103] and was developed to rank N-player general sum games. However, it proposes a ranking based on the (Coarse) Correlated Equilibrium obtained through the maximum entropy objective. Agents require an external correlation device to be able to optimise for a correlated equilibrium, which is a requirement that might not be realistic for negotiation games.

This leaves us without a convincing ranking method for negotiation agents in competitions like ANL 2022. We see the development of such a ranking method as a worthwhile yet highly non-trivial undertaking that is beyond the scope of our work presented here. We hope that our discussion here draws attention to this important topic and serves as a good starting point for discussion within the negotiation community, as much of the recently published work is still benchmarked on the average performance of a given agent.

6.8 CONCLUSION

In this chapter, we discussed learning agents in iterated negotiation games and provided an in-depth analysis of groups of such agents competing in a tournament. We utilised ANL to obtain a diverse set of learning negotiating agents by making learning over repeated games the challenge of the 2021 and 2022 editions. We ran experiments with these agents and extensively analysed the results. To the best of our knowledge, this is the first analysis of learning negotiating agents competing in a tournament.

The agents were designed for the performance metric we set for the competition. Regarding this metric, we found that the agents equipped with a learning mechanism performed better than those that did not, and we conclude that the challenge we set was successful. We also observed that complex strategies are sometimes outperformed by relatively naïve strategies, such as the MiCRO strategy, which managed to outperform every other agent in being a pure Nash equilibrium action in a strategy selection meta-game.

Agents that performed well in the competition show more competitive (defecting) behaviour, despite this sometimes causing failures to reach an agreement and thus hurting both their own utility and the social welfare of the group. We should aim to prevent such behaviour in negotiation games if we care about social welfare.

We showed that the ranking of the agents depends on the submitted opponents when using simple average performance-based ranking methods. Such ranking methods assume that opponents are equally likely to be encountered. We attempted other methods to evaluate agents and showed that the performance of agents does not transfer across these evaluation methods. All ranking methods in this paper are

based on assumptions that can be disputed, and obtained rankings vary depending on the chosen method. This is not unexpected, but still unsatisfactory; as discussed in Section 6.7, this suggests room for further improvements in evaluation criteria and the automated negotiation competition.

To conclude, we designed and analysed a first negotiation competition with a focus on learning across negotiation sessions. We made significant progress in setting up a framework for analysing the performance of learning negotiating agents. Despite this advancement, we note that challenges remain in obtaining a definitive answer to the question of what a good performing negotiating agent is. There might not be a single answer to this question, as it also depends on a conscious choice of which objectives are important and the environment the agent is in. We should push for more (empirical) research into diverse adaptive and learning negotiating agents to gain a more robust understanding of the performance of such agents.