Negotiation Agent Report – Group 14

Introduction

Negotiation agents are autonomous systems designed to strategically interact with opponents in a competitive setting to reach an agreement that maximizes utility.

Our agent, Group14, is designed to negotiate efficiently by leveraging **machine learning**, **utility-based decision-making**, and **game-theoretic principles**.

The **goal** of our agent is to maximize long-term gains while adapting to different negotiation environments. The report provides a detailed breakdown of our agent's structure, strategies, and performance evaluations.

High-Level Overview of the Agent

The agent's logic is implemented through several key functions, each responsible for different aspects of decision-making:

- **on_preferences_changed()** – Computes and stores all **rational outcomes** (offers that exceed the agent's reservation value).
- **__call__()** – The main decision function that determines whether to **accept, reject, or counter an offer**.
- **acceptance_strategy()** – Evaluates incoming offers based on **utility and negotiation time**.
- **bidding_strategy()** – Selects the **best counteroffer** based on precomputed utility values.
- **update_opponent_reserved_value()** – Uses **NN model** to estimate the opponent's reservation value.
- **value_iteration()** – Computes long-term optimal utilities using **dynamic programming**.

Key Design Considerations

- **Efficiency**: The agent **precomputes utility values** to speed up decision-making.
- **Adaptability**: It learns the opponent's behavior using **machine learning**.
- **Strategic Decision-Making**: The **value iteration algorithm** allows the agent to consider **future rewards** rather than making impulsive decisions.

Together, these components form a **robust, data-driven negotiation strategy** that improves as negotiations progress.

Acceptance Strategy

The acceptance strategy is designed to balance **utility maximization** with **negotiation efficiency**. The agent considers the following factors when deciding whether to accept an offer:

1. **Utility of the Offer**

   The agent computes the utility of the incoming offer using its utility function (ufun). If the utility of the offer is high enough, the agent accepts it.

2. **Reservation Value**

   The agent rejects any offer that provides a utility lower than its reservation value.

3. **Time Pressure**

   As time progresses, the agent lowers its acceptance threshold to avoid deadlocks. This ensures that the agent is more flexible towards the end of the negotiation.

Key Features of the Acceptance Strategy:

✓ **Time-Dependent Threshold**

- The function gradually decreases the acceptance threshold as the deadline approaches.
- Early in the negotiation, the agent is more selective.
- As time runs out, the agent becomes more willing to accept offers.

✓ **Prevents Unnecessary Concessions**

- The agent rejects offers that do not meet its minimum acceptable value.

✓ **Optimized Decision-Making**

- The agent evaluates offers quickly and efficiently using precomputed utilities.

Bidding Strategy

The bidding strategy is designed to **maximize utility** while keeping offers **acceptable to the opponent**. The agent follows a structured approach:

1. **First Offer:**

   The agent does not start with the absolute best offer for itself.

Instead, it selects a high-utility rational outcome that is not immediately rejected by the opponent. This helps the agent avoid being perceived as uncooperative.

2. **Counteroffers:**

   If an offer is rejected, the agent proposes a new offer using a value iteration algorithm.

   The new offer is both beneficial to the agent and acceptable to the opponent. Offers are selected based on precomputed utility values.

3. **Opponent Considerations:**

   The agent avoids proposing offers that the opponent is unlikely to accept. It uses opponent modeling to estimate the opponent's reservation value. Offers that fall below the opponent's estimated threshold are excluded.

Key Features of the Bidding Strategy:

✓ **First Offer is Reasonable:**

- The agent does not start with an extreme demand but still ensures it benefits.
- This reduces the risk of early rejection and keeps negotiations productive.

✓ **Counteroffers are Rational & Efficient:**

- The agent proposes the best available outcome instead of random offers.
- This increases the likelihood of a mutually beneficial agreement.

✓ **Opponent-Aware Strategy:**

- Uses opponent utility estimation to avoid unrealistic offers.
- Ensures offers are within the opponent's acceptable range.

✓ **Optimized for Long-Term Gains:**

- The value iteration algorithm selects offers that maximize expected utility.
- This balances short-term and long-term benefits.

Value Iteration Algorithm

Value Iteration is a **dynamic programming algorithm** used to compute long-term expected utility for each possible outcome. Instead of choosing offers based on immediate rewards, the agent optimizes decisions over multiple negotiation rounds.

The algorithm works as follows:

1. Initialize Utility Values – The agent starts with an initial estimate of the utility for all possible rational outcomes.
2. Iterative Updates – The agent repeatedly refines its estimates by applying the Bellman equation, which updates the value of each outcome based on its immediate utility (reward) and the expected future reward.
3. Convergence Check – The agent repeats the updates until the utility values stabilize. This ensures that further updates will not significantly alter the outcome rankings.
4. Optimal Counteroffers – Once the values have converged, the agent uses them to determine the best counteroffers.

Algorithm advantages:

- ✓ Avoids Short-Sighted Decisions – Instead of just maximizing immediate utility, the agent considers how an offer might influence future negotiations.
- ✓ Strategic Counteroffers – The agent selects offers that not only benefit itself but also have a higher chance of leading to a better overall deal.
- ✓ Handles Complex Negotiations Efficiently – Using precomputed values, the agent quickly determines the best counteroffer, reducing response time.

Sources: Artificial Intelligence: A Modern Approach (S. Russell and P. Norvig) – Chapter 17 Page 618, Article, Paper 1, Paper 2.

Reservation Value Model

In negotiations, understanding the opponent's **reservation value** (the minimum utility they are willing to accept) helps in making smarter offers. Our agent estimates this value dynamically using machine learning.

The agent predicts the **minimum utility the opponent is willing to accept** by analyzing past offers. This helps refine counteroffers and speeds up agreement.

1. **Feature Extraction:**
   The agent collects data about each received offer, including:
   Self-utility, opponent-utility, time factor (progress of negotiation) and game-theory metrics (Nash, Pareto optimality, Kalai scores).

2. **Predicting Reservation Value:**
   A Neural Network (or Random Forest) is trained on these features. The model predicts the opponent's reservation value based on past offers. This prediction is continuously updated to improve accuracy.

3. **Refining Counteroffers:**
   The agent avoids offering values below the opponent’s estimated threshold.
   This reduces rejections and leads to faster agreements.

Key Features of the Reservation Value model:

✓ **Learns & Adapts**
The agent continuously refines its estimate based on incoming offers.

✓ **Prevents Inefficient Offers**
Avoids offering values too low, reducing rejections.

✓ **Improves Negotiation Efficiency**
More accurate predictions lead to faster agreements.

Existing vs. New Techniques

Our agent combines established negotiation methods with new innovations to enhance adaptability and efficiency.

Existing Techniques:

1. Utility-Based Acceptance Strategy – uses a dynamic threshold that lowers over time, a common approach in negotiation agents.
2. Opponent Modeling with Machine Learning – predicts the opponent's reservation value using Neural Networks (or Random Forests).
3. Value Iteration for Offer Selection – a well-known reinforcement learning technique to compute long-term optimal utilities.

New Contributions:

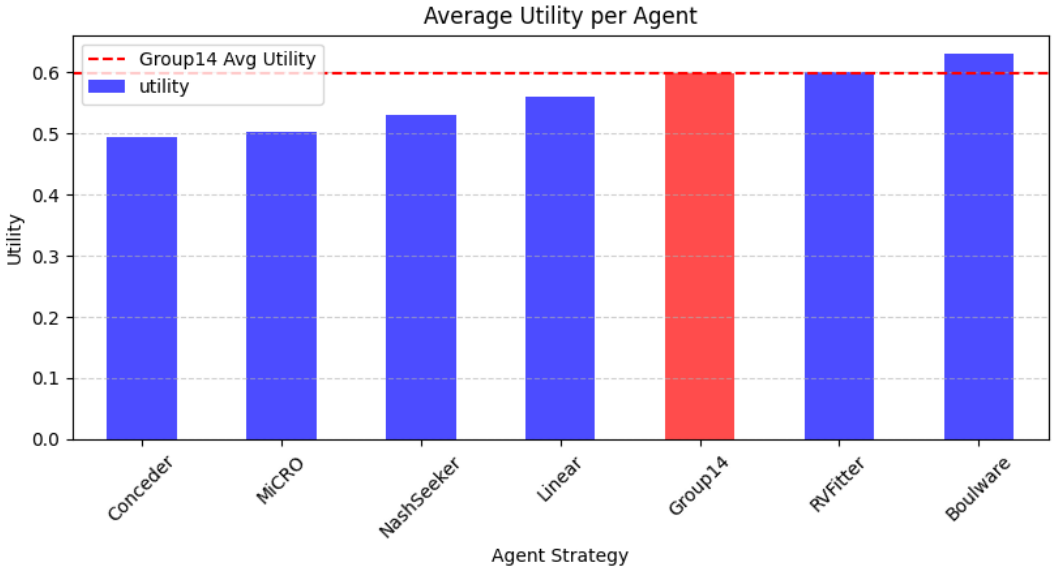
1. Combining ML with Game Theory – refines opponent behavior estimation dynamically, leading to smarter counteroffers.
2. Precomputed Value Iteration – speeds up negotiations by precomputing optimal utilities instead of recalculating on the fly.
3. Opponent-Aware Bidding Strategy – filters out unrealistic offers, increasing acceptance rates and reducing deadlocks.
4. Hybrid Acceptance Strategy – adapts based on reservation value, time, and expected utility changes, making decisions more flexible.

Testing and evaluating

To evaluate Group14, we analyzed the tournament score logs (one of them) and generated several plots to assess its performance.

The tournament consisted of 6 opponents, 10 scenarios, 5 repetitions, and 1000 outcomes, providing a comprehensive evaluation of the agent's effectiveness.

Comparing Utility Across Agents:

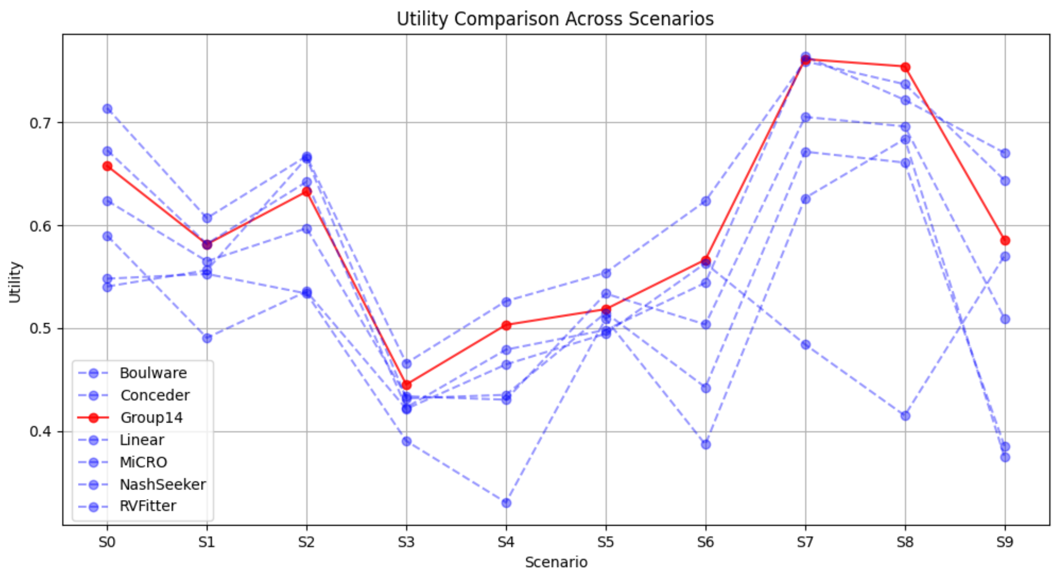


Group14 achieves a higher-than-average utility, performing better than most agents except Boulware and RVFitter.

The red bar highlights our agent, showing it competes well against existing strategies.

Conclusion: Group14's opponent-aware bidding and ML-based modeling contribute to its strong performance.

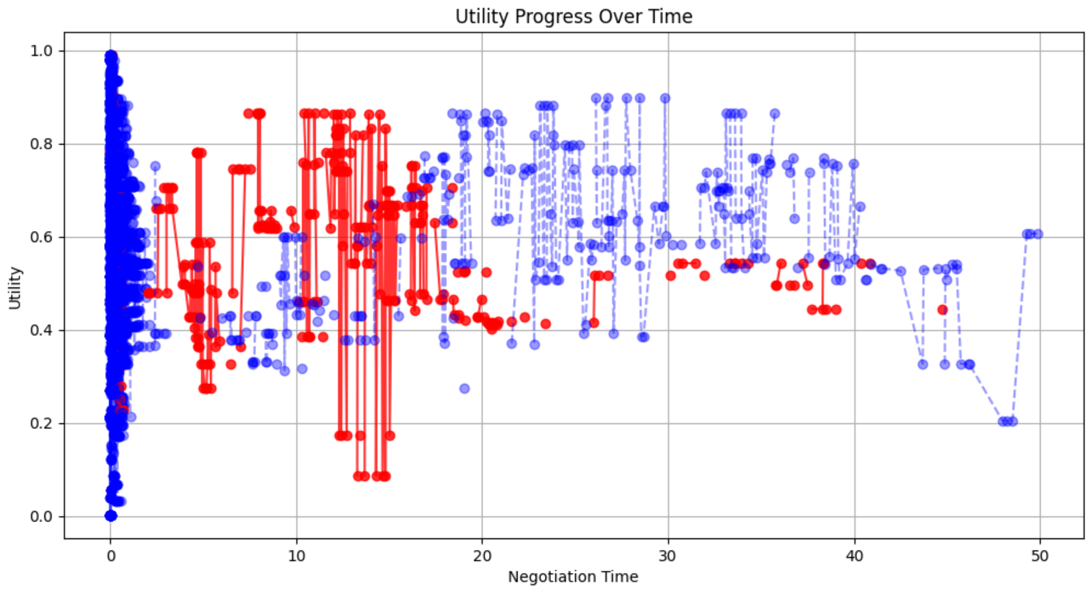
Performance Across Different Scenarios:



Group14 (in red) follows a consistent pattern, performing better than most agents in most scenarios. The agent struggles slightly in some cases (S5 and S9), suggesting specific scenarios may not favor our negotiation style.

Conclusion: The agent is robust across multiple domains, but further tuning is needed in challenging scenarios.

Utility Progress Over Time:



Group14 (red) starts with lower utility offers but gradually improves. Other agents (blue) follow different strategies, with some agents achieving higher gains earlier.

Note: The plot is limited on the x-axis due to some agents exhibiting very slow convergence over time. This adjustment helps to focus on the Group14 and other strong agents' data, providing a clearer view of the negotiation progress and strategy effectiveness within a practical time frame.

Conclusion: Group14's opponent modeling improves over time, allowing better decisions in later rounds.

Strengths of Group14:

- Group14 effectively uses machine learning to estimate the opponent’s reservation value, improving the quality of counteroffers.
- Precomputed utilities via Value Iteration enable faster decision-making, reducing response times and enhancing strategic bidding.
- Group14 demonstrated consistent performance across different scenarios and opponents, achieving high utility in most cases.

Weaknesses of Group14:

- The agent sometimes underperforms in early negotiation phases due to conservative initial offers.
- Performance drops slightly when facing highly unpredictable opponents, suggesting improvements in the learning model.

Testing Setup and Metrics

To evaluate Group14, we conducted a series of tests:

- Multiple Sessions: 5 repetitions across 10 scenarios, involving 6 different opponent strategies, ensuring comprehensive coverage.
- Utility: Average utility achieved in negotiations (highlighted in Figures 1 & 2).

  Motivation - utility reflects the agent's effectiveness in maximizing gains.
- Time to Convergence: Speed of reaching agreements (highlighted in Figure 3).

  Motivation - Time to Convergence assesses efficiency in real-time negotiations.
- Optimality Measures: Nash, Kalai, Pareto optimality and welfare optimality scores for quality assessment, were used in the ML training and prediction stages.

  Motivation - optimality measures provide insight into the fairness and balance of the agreements.

Future Perspectives

To extend Group14's capabilities for real-life negotiations:

- Enable handling multiple conflicting objectives, considering both quantitative and qualitative aspects.
- Incorporate natural language processing (NLP) for more human-like communication, allowing explanations and justifications for offers.
- Integrate emotion detection to better understand human opponents' preferences and adapt strategies accordingly.
- Maintain a history of past negotiations to improve long-term learning and strategy refinement.
- Enhance real-time decision-making with online learning algorithms that adapt during ongoing negotiations.

These extensions would make Group14 more practical for human-assisted negotiations, ensuring more flexible, effective, and personalized outcomes.