Theory of Mind in the Game of Diplomacy

Hallgrímur Thorsteinsson (s240410) & Hróbjartur Höskuldsson (s230374)

The Technical University of Denmark

September 15, 2025

Abstract

The game of Diplomacy presents a rich "mixed-motive" environment in which agents must balance cooperation and competition under hidden communication. Building on DeepMind's simulation-based negotiation using Sampled Best Response (SBR), Sampled Team Aware Value Estimate (STAVE), Restriction Simulation Sampling (RSS), and Mutually Beneficial Deal Sampling (MBDS) we investigate how higher-order Theory of Mind (ToM–k) and relationship modeling can enhance agent behavior. We first formalize first- and second-order ToM, allowing agents to simulate opponents' fallback values via BATNA updates and to refine deal proposals using modified Nash Bargaining Scores. We then introduce a simple relationship score, updated each round via STAVE based utility differences, and integrate it into a relationship aware state-value function. Our theoretical analysis shows that these minimal extensions yield complex social behaviors. Our results suggest a interesting path toward defining a general Diplomacy agents that shows cooperative behavior without being explicitly defined to do so.

1 Introduction

The game of Diplomacy is a stragic, negotiation board game that provides a ideal environment for studying complex decision making, trust between agents and social reasoning. Diplomacy is a "mixed intention" game space where agents can both compete and cooperate. Players simultaneously negotiate, make alliances, and potentially deceive one another to achieve their objectives.

Recent advancements in artificial intelligence have demonstrated remarkable success in tactical, fully observable games, such as chess or checkers. However, Diplomacy introduces additional layers of complexity due to hidden information and the necessity of effective communication.

Two significant breakthroughs have elevated the AI-field of Diplomacy agents: Meta's Cicero [1], which combines strategic reasoning with advanced language models, and DeepMind's negotiation-focused approach [3], which relies extensively on simulation based methods without incorporating large language models. In this paper, we focus on DeepMind's contributions, as Cicero's reliance on language models introduces additional complexities and uncertainness beyond the core framework we aim to analyze.

We will provide an overview of DeepMind's current methodologies, including protocols and algorithms such as Sampled Best Response (SBR), Sampled Value Rollout (SVR), and Sampled Team Aware Value Estimate (STAVE), alongside detailed negotiation mechanisms like Restriction Simulation Sampling (RSS) and Mutually Beneficial Deal Sampling (MBDS). Afterwards we aim to extend these frameworks with higher-order Theory of Mind, and demonstrate the benefits of higher order reasoning by introducing relationship aware agents.

2 Background

In this section we will briefly explain the history and rules for the board game *Diplomacy* and why it is considered a milestone in AI research. We will then discuss the most relevant section in DeepMind's Diplomacy paper [3] as well as provide a brief introduction to Theory of Mind (ToM). In later sections, we will combine the two and by doing so, we will build upon the theories provided in this section.

2.1 The Game of Diplomacy

Diplomacy is a strategic boardgame. Where seven players battle it out on a map of Europe over the control of "supply centers". The game ends when one player controls the majority of centers or all the remaining players on the board agree to a draw.

The game consists of multiple rounds where players negotiate in private conversations and try to strike a deal, lie, help or influence each other. At the end of each round all players choose one move for each unit that are performed simultaneously.

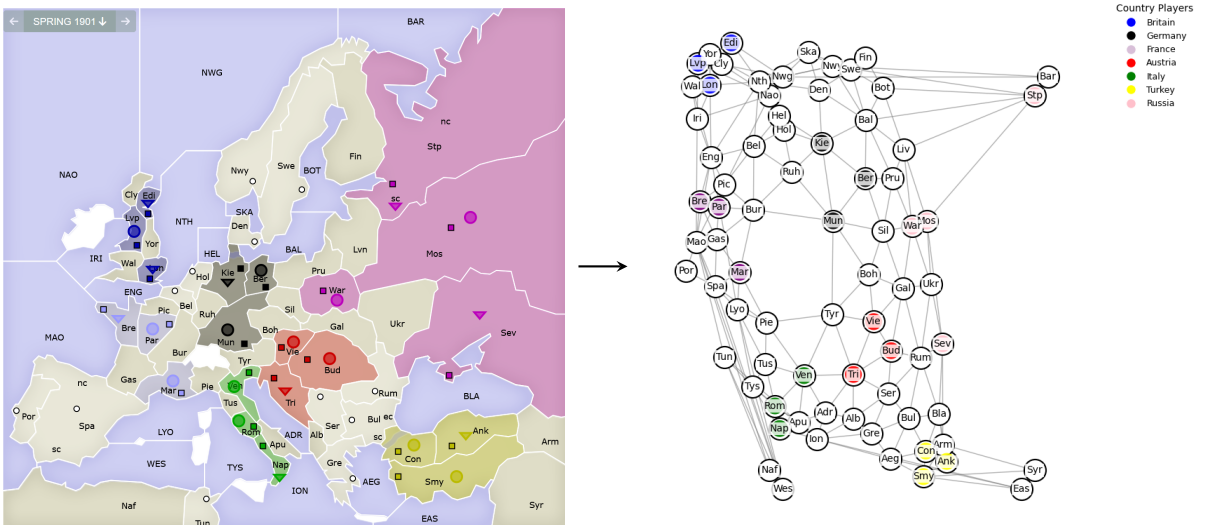


Figure 1: Board of Diplomacy and its graphical counterpart

Agents that outperform humans have been made in the so called "No-press" version of Diplomacy where communication is taken out of the game leaving only the tactical play. This makes the game setting similar to chess or checkers where all moves from each players are fully observable. However in this paper we look at the standard version of Diplomacy with hidden communications between players.

Diplomacy is a "mixed-motive" game setting and is a therefore a interesting game to work with because each player has a mix of cooperative and competitive motives. The game also has a simple ruleset, all players move simultaneously, all units are equally strong, and there is no random elements. All of these reasons make Diplomacy an ideal testbed for testing trust and negotiation agents.

2.1.1 The board of Diplomacy

To bring Diplomacy's map into our framework, we first converted the standard board (Figure 1) into an undirected, bidirectional graph $G = (V, E)$. Each province (land or sea/coastal area) becomes a node $v \in V$, and an edge $(u, v) \in E$ exists whenever a unit in province u can legally move to province v in a single turn. For this full graph, we computed the degree $\deg(v)$ of each node v, capturing how many immediate neighbors it has.

In order to keep our analysis to a reasonable complexity, we compressed the graph into a smaller version. The resulting simplified graph (Figure 2) keeps the essential connectivity of the original board while abstracting away coastal/land rules. We will use this graph going forward in the project.

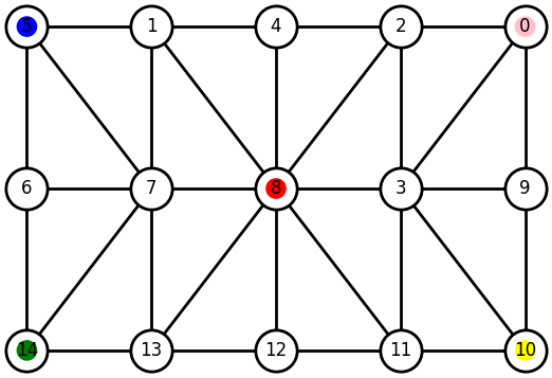


Figure 2: Simplified graph of a Diplomacy like game

2.1.2 Basic actions in Diplomacy

Diplomacy orders come in three basic types: *Hold*, *Move*, and *Support*.

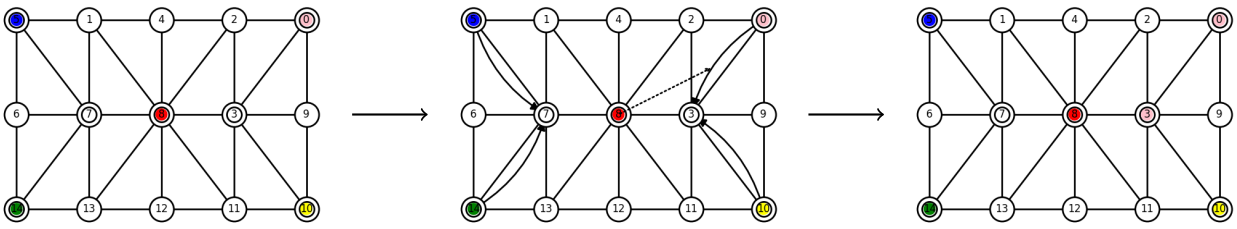
- A *Hold* order keeps a unit in its current province.
- A *Move* order directs a unit to an adjacent province.
- A *Support* order allows one unit to bolster another unit's move or hold, increasing its strength.

In Figure 3 the main moves of the game are showed.

(a) **Starting position:** Each unit occupies its home province. Here, Red, Pink, Blue, Yellow and Green each have one unit ready to receive orders.

(b) **Support phase:** Red issues a *Support* order to assist Pink's army moving into node 3. Meanwhile, Green and Blue both go to node 7 resulting in a "bounce" where no one gains the point. By supporting Pink, Red contributes an extra combat point, making Pink's move succeed.

(c) **Resulting move:** With support from Red, Pink's move succeeds and overpowers Yellow, which moved in with only one army. Now Pink occupies node 3 and builds another army at its home supply center.



(a) Starting position (b) Red supports Pink (c) Pink gains an army

Figure 3: Progression of moves showing, move, and support orders.

2.2 Negotiation agents and their protocols

The negotiation framework [3] distinguishes three classes of negotiating agents:

- **Baseline Negotiators.** Assume agreements are binding and always honor any contract they reach. In the absence of a deal, they select actions via Sampled Best Response (SBR) over their full move set; once a contract $D = (R_i, R_j)$ is agreed, they restrict SBR to R_i and still assume the partner respects R_j.
- **Deviators.** Behave like Baseline Negotiators during negotiation but may break their commitments in the action phase:
  - *Simple Deviators* ignore all contracts when choosing actions, effectively "forgetting" any deals they signed.
  - *Conditional Deviators* consider whether breaking a contract is immediately profitable—under the assumption that their partner will uphold its end—and only deviate when this yields higher expected value.
- **Defensive Agents.** Punish deviations by negatively responding to broken promises while otherwise acting as Baseline Negotiators:
  - *Binary Negotiators* cease all communication with any peer who has deviated.
  - *Sanctioning Agents* actively choose actions that both improve their own win probability and reduce the deviator's win probability.

In this paper, we focus exclusively on the honest **Baseline Negotiator**, which cannot lie or deviate from its agreements. We belive that this will serve as a good foundation enabling us to add more complicated agents in the future.

2.2.1 Pretrained No-Press Diplomacy model

The decision making components of the Diplomacy agents rely on simulations and these simulations depend on a pretrained No-Press Diplomacy model. This model was trained through imitation learning on a large dataset of human Diplomacy games. Then, the models are further refined through reinforcement learning. Once trained, these networks are frozen and used throughout all agent evaluations. This network provides policies $\pi_i : SxA_i -> [0, 1]$ that maps any possible game state to probability distribution of an action a. It also provides $V : S$, a value function that returns the expected win probability for a given state. The policy network is used to sample plausible actions during forward simulation while the value network is used to estimate outcomes during these simulations

2.2.2 SBR, SVR, and STAVE: Simulation-Based Evaluation Methods

A key component of Diplomacy agents is their ability to evaluate actions and deals using forward simulation. This is accomplished through three interconnected tools: Sampled Best Response (SBR), Sampled Value Rollout (SVR), and Sampled Team Aware Value Estimate (STAVE).

SBR is used when an agent needs to decide which action to take. Given a belief over what the other agents might do, the agent samples a set of possible actions and simulates the resulting games to see which option performs best.

SVR and STAVE are both used during negotiation to estimate wether to continue negotiation or or not by evaluating how good a certain move is. SVR is used during Restriction simulation sampling (RSS) protocol. Rather than choosing an action, SVR assumes that all agents are following fixed policies and estimates how well a given move would perform. STAVE is used during the he Mutually Beneficial Deal Sampling (MBDS) protocol and tells agents how good a deal would be if it were accepted.

2.2.3 Protocols

The paper introduces two protocols, which dictate how the agent communicate throughout the game. The Mutual Proposal protocol allows agents to make simple peace agreements, and The Propose-Choose protocol allows agents to make joint action move agreements. In addition, there are two algorithms provided that these protocols use: Restriction Simulation Sampling algorithm (RSS) and Mutually Beneficial Deal Sampling (MBDS), respectively. Both protocols have a proposal phase and an action phase. During the proposal phase, agents can enter in agreements by proposing to restrict their move sets. During the action phase, agent p_i either select a move from legal move set $A_i(s)$, or a move from the unrestricted $R_i \subseteq A_i(s)$. The key difference between these protocols is in how the respective move sets are restricted. We will therefore go through the proposal phase in both protocols separately in the next subsections.

During the action phase:

- If no contract is agreed, the agent selects an action using Sampled Best Response (SBR) over its full action set A_i:

$$a_i^* = \text{SBR}(s, A_i) \tag{1}$$

- If contract $D = (R_i, R_j)$ was agreed, the agent selects an action using SBR restricted to R_i:

$$a_i^* = \text{SBR}(s, R_i) \tag{2}$$

We will briefly discuss these two algorithms in the following sections. However, the rest of this paper will exclusively focus on agents using the MBDS algorithm. The reason for this is that the simplicity of the RSS algorithm does not allow for additional nuance in agent behavior.

2.2.4 Restriction Simulation Sampling

The Mutual Proposal protocol uses Restriction simulation sampling (RSS) and is intended exclusively for the Baseline Negotiator. During the negotiation phase, the agent p_i considers proposing a contract $D = (R_i, R_j)$ to another agent p_j. Agents within this protocol are considerably simpler since they only consider Peace agreements. The agent estimates two expected values using simulation value estimation (SVE):

- Estimated value without agreement:

$$\hat{V}_B^i(s) \tag{3}$$

- Estimated value with agreement:

$$\hat{V}_{B'}^i(s) \tag{4}$$

The agent proposes the contract D if:

$$\hat{V}_{B'}^i(s) > \hat{V}_B^i(s) \tag{5}$$

In the case where a peace agreement takes place, the agents in the agreement restrict their move set R_i by exuding all moves where units from p_i attempts to move into a province occupied by p_j, or take one of its research stations.

2.2.5 Mutually Beneficial Deal Sampling

The MBDS algorithm governs how all agents in Diplomacy under the Propose-Choose protocol generate, evaluate, and agree upon bilateral contracts. MBDS uses the Nash Bargaining Score to measure the quality of a deal for both players. This means that all agents under the propose choose protocol are inherently trying to form contracts that help both agents. The algorithm can be separated into three phases; (A) an internal bargaining loop to refine fallback expectations; (B) a proposal phase where agents offer contracts to others; and, (C) a choose phase where agents select deals from those proposed to them.

The following pseudo-code outlines the different phases of the MBDS algorithm. This serves as the backbone of the negotiation process used throughout the remainder of this paper. The upcoming chapters will build on this baseline, extending it with additional reasoning layers and relationship modeling mechanisms.

A. Internal Dynamic Bargaining Simulation (BATNA update)

1. Initialize fallback utilities for each agent using STAVE without any contract restrictions (i.e., assuming no deals are made).
2. Repeat for a fixed number of iterations:
   (a) For every pair of agents (p_i, p_j):
      i. Sample a set of candidate contracts $D = (R_i, R_j)$ from the MBDS generator.
      ii. For each candidate contract:
         - Estimate utilities d_i and d_j using STAVE.
         - Compare each utility to the current fallback (d_i^0, d_j^0).
         - Compute the Nash Bargaining Score.
      iii. Keep the highest-scoring deal that satisfies $d_i > d_i^0$ and $d_j > d_j^0$.
   (b) Update fallback utilities using a damping factor κ to ensure smooth convergence.

B. Proposal Phase

1. For each agent p_i:
   (a) For each partner $p_j \neq p_i$:
      i. Use the updated fallback values from phase A.
      ii. Sample a new batch of candidate contracts $D = (R_i, R_j)$.
      iii. Evaluate each contract using STAVE.
      iv. Compute the Nash Bargaining Score for each candidate.
      v. Select the contract with the highest score that improves utility for both agents.
      vi. Propose this contract to agent p_j.

C. Choose Phase

1. Each agent p_i receives all contracts proposed to them by other agents.
2. For each received contract:
   - Recompute utilities using STAVE and compare them to updated fallback utilities.
   - Mark a contract as acceptable if both agents benefit from it.
3. Among the acceptable contracts, select the one with the highest Nash Bargaining Score.
4. If two agents select the same contract, it is finalized as an agreement.
5. If both agents mark each other's proposals as acceptable but rank them differently, a contract is selected at random from the agreed pair.

During the negotiation phase, for each candidate contract $D = (R_i, R_j)$ between agent p_i and agent p_j, the agent estimates the utility d_i and d_j as well as the no-deal utility d_i^0 and d_j^0 The contract is evaluated using the Nash Bargaining Score:

$$\text{NBS}_i(D) = (d_i - d_i^0)(d_j - d_j^0) \tag{6}$$

The agent proposes the contract that maximizes this score, provided that $d_i > d_i^0$ and $d_j > d_j^0$. In case an agreement takes place, the restricted move set R contains only those moves that were agreed upon. The agreements always specify each units move, so the length of this move set will be the same length as units available. An agent can only make a single agreement at a time.

2.3 Theory of Mind

The concept of *Theory of Mind* (ToM) originates from developmental psychology, where it describes an individual's ability to reason about the mental states of others such as beliefs and intensions. First introduced by Premack and Woodruff (1978) [4].

This paper primarily uses the formalization of ToM presented in the thesis by Harmen de Weerd (2015) [2], which provides a framework for modeling agents with varying levels of ToM. In this framework, a *zero-order* agent acts without reasoning about the mental states of others. A *first-order* agent can model what another agent believes or desires, allowing for strategic reasoning such as deception or cooperation. *Second-order* agents and higher, can recursively reason about what others believe about their own beliefs, enabling deeper anticipatory behavior.

Included in the thesis is the study on how agents with different order ToM behave in mixed-motive negotiations in the game Colored Trails [5]. Like Diplomacy, Colored Trails is a mixed-motive game where agents negotiate over actions. The Colored Trails experiments show that agents using ToM1 and ToM2 reasoning outperform simpler agents and help avoid negotiation failure. Inspired by this work, we apply a similar ToM-based negotiation framework in our Diplomacy setting.

3 Theory of Mind in Diplomacy

In this section we will analyse how ToM will effect agents in diplomacy. First we will look at the Baseline negotiators and define how they can be modified with ToM. After that we will show an example where different orders of ToM will behave in the same scenario. Lastly we will introduce our own agent modified further to track relationships between itself and other agents.

3.1 Baseline Negotiators

Each round, agents propose and evaluate contracts of the form $D = (R_i, R_j)$, where R_i and R_j are the restricted move sets for players p_i and p_j respectively. These restricted sets fully specify the moves for each player's units during the next game phase. Additionally, all agreements are final and there is no option to deviate.

3.1.1 First-Order Theory of Mind (ToM1)

The baseline negotiator agents implement a first-order theory of mind by reasoning about how the opponent will evaluate a proposed contract. These agents assume that their opponent is a *zero-order* agent, one that does not simulate others, and acts purely based on evaluating contracts using their own expected win probabilities. As described in the MBDS algorithm, the agent proposes the deal $D_i = (R_i, R_j)$ that maximizes the $\text{NBS}_i(D)$ score. This is how the agents work in DeepMinds's paper.

3.1.2 Second-Order Theory of Mind (ToM2)

We now extend this to second-order theory of mind agents. A ToM2 agent assumes that its negotiation with a ToM_1 *agent*. That is, an agent who uses the Nash Bargaining Score to select deals, assuming they are negotiating with an ToM_0 agent. The simplest approach would be to simulate what the opposing agent p_j would propose to the higher order p_i. After all, we know that agents only send the best deal according to NBS to each other agent. This could be advantageous to the agent since it could simulte what contract proposal it would receive before sending its own proposal. However, this naive approach is too simple for an agent to get any real benefit out of and we will show that here. For this we have to look into the MBDS algorithm in more detail. Each agent p_i selects one contract to propose to every other agent p_j. This process begins by sampling a set of candidate actions C_i from its own policy and estimating a corresponding candidate set C_j for the opponent. The Cartesian product of these sets forms the candidate contract space:

$$C_i \times C_j = \{(a_i, a_j) \mid a_i \in C_i,\ a_j \in C_j\} \tag{7}$$

Each candidate contract (a_i, a_j) is evaluated using the expected values from STAVE for both agents:

$$\hat{Q}_i(a_i, a_j) = \mathbb{E}_{a_{-\{i,j\}}} \left[V_i(T(s, (a_i, a_j, a_{-\{i,j\}}))) \right] \tag{8}$$

$$\hat{Q}_j(a_i, a_j) = \mathbb{E}_{a_{-\{i,j\}}} \left[V_j(T(s, (a_i, a_j, a_{-\{i,j\}}))) \right] \tag{9}$$

Its important to note here that since both agents use the same value function, and there is no information unknown to both agents at this point. Wether these values are computed by agent p_i or p_j, they are likely to end up with the same ranking. The only variability is the Monte-Carlo simulation performed during calculation. To decide which actions to prioritize, agent p_i computes a score for each of its own candidate actions a_i, combining its own value with an estimate of p_j's value, rescaled using their no-deal baseline values q_i and q_j:

$$\text{Score}_i(a_i) = \hat{Q}_i(a_i, a_j) + \beta \cdot \frac{q_i}{q_j} \cdot \hat{Q}_j(a_j, a_i) \tag{10}$$

The scaling factor $\frac{q_i}{q_j}$ ensures that both utilities are evaluated on a comparable scale. The parameter $\beta \in (0, 1)$ controls how much emphasis agent i places on the partner's benefit when evaluating a deal. A low β results in self interested proposals, while a high β favors cooperative behavior. After ranking actions according to this score, each agent selects the top K actions for itself and for its partner, forming a filtered set of candidate contracts. Each of these is then scored using the Nash Bargaining Score:

$$\text{NBS}_i(a_i, a_j) = \max(0, \hat{Q}_i(a_i, a_j) - q_i) \cdot \max(0, \hat{Q}_j(a_i, a_j) - q_j) \tag{11}$$

The contract with the highest NBS is selected as the proposal from agent i to agent j. We can see that with a suffiently high K and when $\beta = 1$, the resulting proposed contracts would always be the same between all agents. In that specific case, a higher order could not result in any benefits to the agents. The paper unfortunately does not mention if these hyperparameter are defined across the board, or are agent specific. For simplicity, we will continue with the assumption that they are fixed, shared among all agents and that $\beta \neq 1$, which allows agents to create a set of proposals that are biased towards their own utility. This case would result in asymmetric proposal rankings and agent p_i could for example be able to know what deal they could propose that would maximize their utility and minimize the opponents utility while still being an acceptable deal for agent p_j. This however, would be a naive solution since there are factors that agent p_i is not considering. Mainly the fact that we are not considering the choose phase.

Once an agent reaches the choose phase, it considers all proposals where the NBS is positive. From the remaining deals, it will always select the deal that yields the most utility for itself, given that it is better than the no-deal baseline. Therefore an agent that simulates just the propose phase would be unlikely to benefit unless it also simulates which other proposals, agent p_j is receiving.

To address the limitations of simulating only the propose phase, we will now consider a second-order agent that additionally simulates the BATNA update process. As described in the MBDS algorithm 4, the BATNA (Best Alternative To a Negotiated Agreement) update iteratively refines the no-deal baseline scores q_i and q_j, by simulating what contracts agents would likely form with others in the absence of a deal between p_i and p_j. A higher-order ToM agent that incorporates this BATNA simulation can anticipate not only which contracts are good enough to be accepted but what baseline the contracts have to beat to stop the trading partner from looking at alternative options. Concretely, if agent p_i wishes to propose a contract $D = (a_i, a_j)$ to agent p_j, it would now estimate the opponent's BATNA score q_j not as a fixed value, but as the result of p_j's own simulation of deals with other agents. We note here that we are under the assumption that the agent knows the parameters K and β.

Lets now consider a ToM2 agents. We will introduce a higher order theory of mind equation for ranking deals. Instead of changing how the NBS is used to select agreements, we will instead change how the top K contracts are ranked. For this agent, the new score system is defined as

$$\text{Score}_i^{(2)}(a_i) = H^{(2)}(a_i, a_j) \cdot \left(\hat{Q}_i(a_i, a_j) + \beta \cdot \frac{q_i}{q_j} \cdot \hat{Q}_j(a_i, a_j) \right) \tag{12}$$

where

$$H^{(2)}(a_i, a_j) = \begin{cases} 1 & \text{if } \hat{Q}_i(a_i, a_j) > q_j \\ 0 & \text{otherwise} \end{cases} \tag{13}$$

This introduces a new binary function binary function $H(Q_j, q_j)$ that will *eliminate* all potential contracts that are lower than the fallback values for the opposing agents. More formally, if the opposing agent p_j can expect to get q_j from another deal, and this value is higher than the proposed utility from Q_j, then we will not consider it, since it will be ignored. This allows the second-order agent to rank higher-quality deals, in the sense that the deal that is proposed will not be ignored. This will result in a higher likelihood of no proposal being sent, but increase the likelihood of sent deals to be chosen. Furthermore, since a ToM2 agents assumes other agents are ToM1 agents, we do not need to simulate the choose phase since it assumes its proposed deal will beat other deals. The agent can make this assumption since during the BATNA simulation, we have found a fallback value q_j that is a upper ceiling of the quality of proposals coming from other agents.

We can now extend score for ToM2 agents to k order agents. We do this by first extending the BATNA algorithm to allow for recursive fallback values to be computed. With only a slight change we introduce the BATNA algorithm for higher order agents.

A. Internal Dynamic Bargaining Simulation (BATNA update for ToM-k agent)

1. Initialize fallback utilities for each agent using STAVE without any contract restrictions (i.e., assuming no deals are made).
2. Repeat for a fixed number of iterations:
   (a) For every pair of agents (p_i, p_j):
      i. Sample a set of candidate contracts $D = (R_i, R_j)$ from the MBDS generator.
      ii. For each candidate contract:
         - Estimate utilities d_i and d_j using STAVE.
         - Compare each utility to the current fallback (d_i^0, d_j^0).
         - Simulate p_i as a ToM-k agent and p_j as a ToM-$(k-1)$ agent by recursively computing d_j^0 assuming p_j negotiates with all other agents as ToM-$(k-2)$.
         - Compute the Nash Bargaining Score.
      iii. Keep the highest-scoring deal that satisfies $d_i > d_i^0$ and $d_j > d_j^0$.
   (b) Update fallback utilities using a damping factor κ to ensure smooth convergence.

This iterative BATNA simulations allows agents to update their fallback values by simulating the deals they are able to get from other agents. This allows us to extend the *Score* formula from before to become:

$$\text{Score}_i^{(k)}(a_i, a_j) = H^{(k)}(a_i, a_j) \cdot \left(\hat{Q}_i(a_i, a_j) + \beta \cdot \frac{q_i}{q_j^{(k-1)}} \cdot \hat{Q}_j(a_i, a_j) \right) \tag{14}$$

$$H^{(k)}(a_i, a_j) = \begin{cases} 1 & \text{if } \hat{Q}_j(a_i, a_j) > q_j^{(k-1)} \\ 0 & \text{otherwise} \end{cases} \tag{15}$$

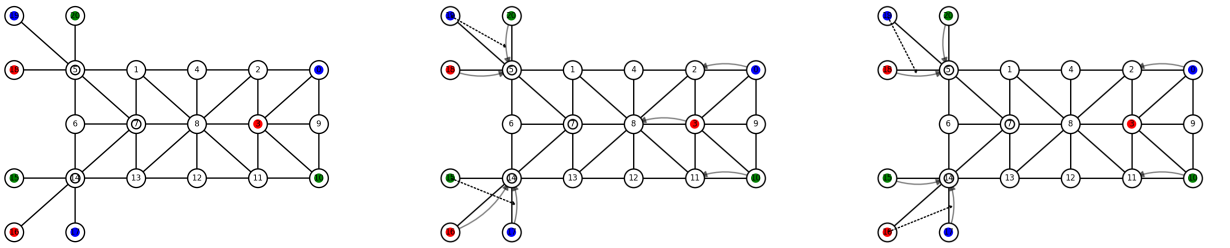
Here, $q_j^{(k-1)}$ represents the fallback values of the opposing agent p_j which is assumed to be an order below agent p_i. This value is calculated through the iterative BATNA simulation.

We can now discuss what pattern will emerge with increased k agents. Lets assume an agent p_i with $k = 2$. This agent will propose the deal that maximizes its utility while being over agents p_j fallback value. For the rest of the agents, p_i will assign this deal as its fallback. Lets now assume that p_j is not a ToM1 agent but a ToM3 agent. Agent p_j will now propose a deal that maximizes its utility but is better than the fallback for agent p_i. This means that when k is increased, the deals proposed will get closer and closer to their own fallback value (the deals are likelier to be accepted but are getting worse for themselves). This means that having higher k is not always advantageous. If other agents are considerably lower order, then the agent is making worse deals then it has to be. This happens because the agent does not know, nor is predicting, what the order k is for other agents.

In Figure 4 we can see an example of how a second order ToM agent can navigate a tricky situation using its higher order. The starting position 4a shows a pivotal point in the game where each agent can make a

fair deal of helping each other claim a supply center $\{5, 14\}$. The agent p_{red} is in an advantageous position since it is closer to the last supply center $\{7\}$.

Lets now go through the possible scenarios as agent p_{red} which is a 2nd order ToM agent playing against baseline 1st order ToM agents. Any agent that is not able to secure a deal to secure one of the corner supplies will loose a considerable amount of utility. The agent's best deal would be to propose to both agents p_{blue} and p_{green} to claim one supply center each and then move closer to the middle. However, since p_{red} is able to simulate the deals that agents p_{blue} and p_{green} receive, it knows that this move will be below their fallback value, which would be for p_{blue} and p_{green} to make their own agreement since p_{red} is in a better position. This scenario is shown in scenario 4b where the greediness of 1st order results in p_{red} being left out of a deal. Since it simulated this, it removes such deals from their ranking, leaving only deals that are slightly worse, but are deemed acceptable by the other agents. This results in scenario 4c where p_{red} purposefully does not move closer to the center, but is able to form a deal with p_{blue}. The higher order agent p_{red} was able to find the minimal acceptable deal possible that maximizes its utility.



(a) Starting position available points are at node 14, 5 and 7

(b) Blue and green make a deal and red is left out.

(c) Red and Blue cooperate since Red does not advancer to the center

Figure 4: (a) Starting position before anyone moves, (b) Scenario when Red is ToM-1. Red moves up to node 8 prompting Green and Blue to work together helping each other gain the corners. (c) Red being ToM-2 agent knows that in order to make a deal he can't move to 8.

3.2 Relationship Based Baseline Negotitor agents

In the previous section we demonstrated how the agents can achieve advantages from a higher order ToM by strategically evaluating which deals to propose. However, despite these advantages, higher-order ToM agents are still not fully utilizing their potential capabilities. Agents can currently only reason about how other perceive them within a single round which results in simply not that many opportunities to do so. They lack an explicit mechanism for tracking evolving interpersonal relationships based on historical interactions. In this section we will formulate a mechanism that allows higher order agents to benefits further from their reasoning. Moreover, we will show how such a mechanism can generate long term friendships and betrayal without explicit instructions.

For an agent p_i we define relationships U_i such that the relationship value $u_{i \to j} \in U_i$. At the start of the game, the relationships between players are initialized at 1 and can dynamically change throughout the game. Calling this a relationship value is thematically appropriate for a setting like Diplomacy which is based on building and destroying relationships.

The update rule for this value has to reflect the actions of agents on the board. There are multiple ways to implement this and for this exploration we will go for a simple approach, that uses the procedures and estimations given in DeepMind's paper. Given agents p_i and p_j, the update rule for $u_{i \to j}$ after a round is defined as

$$U^t_{i \to j} = U^{t-1}_{i \to j} + \gamma \cdot \left(\hat{Q}_i(s^{t-1}, (a_i, a_j)) - V_i(s^{t-1}) \right) \tag{16}$$

Here, $U^{t-1}_{i \to j}$ represents the relationship score at the start of the round that just concluded. The STAVE estimate $\hat{Q}_i(s^{t-1}, (a_i, a_j))$ represents the estimated utility gain or loss of agent p_i given the moves that p_j

played. This procedure is used in the proposal phase to determine what deals to propose, and is now being used to determine how damaging a move is to you. To get the true change of state utility we then subtract the value state at the start of the round. This will result in $U^t_{i\to j}$ increasing if p_j played a move that would increase p_i utility, and decrease otherwise.

The second mechanism needed is an updated state value estimation. This allows agents to weigh the strengths and weaknesses of their relationships when evaluating the board. We introduce a new *Relationship-aware State-Value Function*:

$$V_i^U(s^t) = V_i(s^t) + \frac{1}{n-1}\sum_{j\neq i} U^{(t)}_{i\to j} V_j(s^t) \tag{17}$$

With this introduction, agents will not only consider the state of the board but the accumulative relationship between the rest of the players, where the sum is taken over all other agents $j \in \{1,\dots,n\} \setminus \{i\}$. This also directly changes how agents evaluates moves and deals. The expected utility of a move that makes a lot of enemies is lower than it was before, and moves that increases relationships will be preferred. The effect of the relationships is scaled for each agent $U^{(t)}_{i\to j} V_j(s^t)$. This allows agents to put more weight on relationships with players in better positions. If a loosing agent with close to zero expected win rate is our main enemy it will not effect our utility as much as if they were in the winning position.

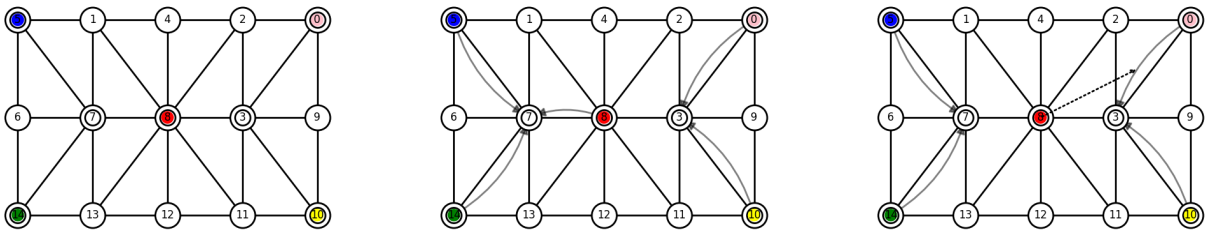
Remarks: In Equation 16, the parameter γ controls the sensitivity of the relationship update and is critical to how agents will behave within this relationship protocol. The coefficient γ is currently treated as a fixed scaling factor, but in practice, it may be better represented as a non-linear function. Using a static γ risks producing unstable behavior, for example, exponential growth or decay in relationship values which could distort agent decision making. Determining the correct form of γ would require further experimentation with the underlying state value models, which is beyond the scope of this paper. Nonetheless, it provides a principled starting point for integrating relational dynamics into agent behavior.

3.2.1 First-Order Theory of Mind (ToM1)

We will now examine a scenario that can occur using baseline negotiators using the relationship aware state value function defined in eq. 17. An agent p_i that is relationship aware will constantly evaluate the board state s_i differently than a baseline agent. It will consider it self to be in a better or worse position depending on the relationship with the other agents. This will change how it selects its best move, how it proposes deals, and how it selects deals.

When agent p_i is ranking its best move, it will favor moves that increase its utility $V_i(s)$ as well as opposing agents utility $V_j(s)$ if their relationship $u_{i\to j}$ is sufficiently high. On the other hand, it will favor moves that increase $V_i(s)$ and decrease $V_j(s)$ if $u_{i\to j}$ is sufficiently low. The strength of relationship $u_{i\to j}$ is determined by $V_j(s)$, meaning that agents with higher expected win rate will inherently have more influence on the board. This ranking will in turn have an effect on the propose and choose phase. An agent p_i will propose better deals to agents with higher $u_{i\to j}$ and be less likely to send any to deals at all to agents with negative $u_{i\to j}$. The decreased likelihood of proposing deals to agent with negative $u_{i\to j}$ happens because moves that reduce the win rate of the opposing agents are unlikely to have a positive Nash Bargaining score. During the course of the game, the relationship between two agents can become so high or low that the agent will start considering moves that a baseline agent would never consider. An example of such a move is demonstrated in Figure: 5. The starting position 5a contains a deadlock scenario, where all agents will fight for the two remaining supply centers $\{7, 3\}$. In this scenario, getting these centers is vital, and will increase the expected win rate of the agent dramatically, while decreasing all others. The chosen move will therefore be shown in 5b where the agents in the corner will bounce each other out, and the red agent in the middle will chose one side at random. For a normal baseline agent, this scenario will repeat itself indefinitely. However, relationship based agents will solve the otherwise deadlock. Lets consider this scenario from the perspective of the center agent p_{red}. After the first round the relationships $u_{red\to blue}$ and $u_{red\to blue}$ will decrease because the joint actions $\hat{Q}_i^{red,\{blue,green\}}(s^{t-1}) < 0$. In other words, they tried to harm red's win rate by blocking their move. The opposite happened for agents pink and yellow, which did not block red, and kept space 3 empty. This occurs even though the state space remains the same, since eq.16 considers what move was made regardless if it was successful or not. During the next round, red will try to foster the relationship between pink and yellow and make a deal with one of them to repeat the same move again.

This will repeat itself until we reach some round t where red instead decides to support one of its allies to claim the center spot $\{3\}$. This is shown in scenario 5c. This happens when $u_{red \to pink}$ becomes high enough such that when red is evaluating the move where it supports pink, it evaluates $V_i^U(s^t) > q_{red}$, where q_{red} is the fallback value for red when no deal is made. This allows red to sacrifice its own utility for the benefit of its ally. The benefit for red in this scenario is that the relationship value $u_{pink \to red}$ will increase drastically. This will make pink favor moves that increase red's utility. This however, does not guarantee that pink will do so. If a move is available that will increase its win rate high enough it will choose that move even if decreases red's utility. In others words, there is no guarantee that the favor will be reciprocated.



(a) Starting position available points are located at node 7 and 3

(b) Stalemate move everyone tries to secure a point for them self.

(c) Red supports the move to 3 securing the point for Pink

Figure 5: Example of benefits of relationship aware agents. (a) Starting position before anyone moves, (b) If the agents would not have any U score this position would repeat indefinitely, each agents tries to secure a point but bounces back because there is no majority. The U score for red in regards to pink and yellow increases. (c) After repeating enough rounds, the U score of Red and Pink is so high that Red decides to help Pink get a point.

3.2.2 Second-Order Theory of Mind (ToM2)

We have so far discussed how the baseline agent can be transformed into a relationship aware agent. With only first order of mind, the baseline agent can form dynamic relationships with other agents. In section 3.1.2, we formalized how the baseline agent can be changed to become a second order theory of mind agent.

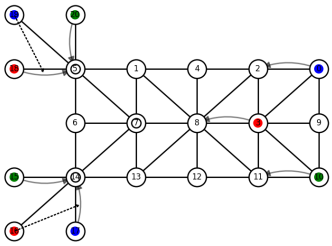
We now consider how these two ideas, relationship aware agents and higher order reasoning, can be combined. Specifically, we propose a second order ToM agent that not only anticipates its opponent's fallback values through recursive BATNA simulations, but also adjusts its evaluations and proposals based on its evolving interpersonal relationships.

The mechanisms for generating proposals and updating fallback values remain the same as those used by the baseline agent, following Internal Dynamic Bargaining Simulation (BATNA update for ToM-k agent) algorithm described in Section. 3.1.2. However, with the introduction of relationship aware value functions using U, some interesting new dynamics begin to emerge.

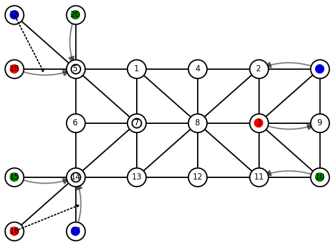
Returning our attention to example in Figure: 4 We explained how the baseline agent p_{red} can utilize its higher ToM to recognize that it might have to not advance their third unit towards the center to be able to secure a deal with p_{blue}. For the relationship aware agent, there is a new dynamic that the agent considers during its simulation:

1. $u_{blue \to red}$: How the opposing agent p_{blue} evaluates their relationship
2. $u_{blue \to green}$: How the opposing agent p_{blue} evaluates the relationship between the other agent

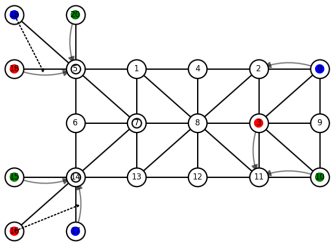
This will change what move p_{red} will suggest to be able to secure a deal and receive one of the available supply centers. The different moves are shown in 6. If the relationship $u_{blue \to red}$ is high enough (red has helped blue in the past), then p_{blue} will accept p_{red} best move, which is to move closer to the center. This results in scenario 6a. If $u_{blue \to red}$ is not high enough, or if $u_{blue \to green}$ is high enough for p_{blue} to favor deals to p_{green} then p_{red} might have to deliberately reduce its win rate by moving away from the center. This results in scenario 6b. If $u_{blue \to green}$ is very low, then p_{blue} will favor deals that reduces the win rate



(a) Red is able to play its best move and make a deal with blue.



(b) Red makes a deal with blue and moves back.



(c) Red is able to make a deal with blue by blocking green.

Figure 6: Three possible scenarios where red can use its higher order theory of mind to make a deal. (a) Blue and red have high relationship, and red is able to rely on that and use its best move, (b) Blue and green have high relationship, and the only way red is able to make a deal is by going back to 9. (c) Blue and green have low relationship and red is able to make a deal by blocking green (blue's enemy).

of p_{green}. This results in scenario 6c where p_{red} is able to make a deal with p_{blue} by blocking the path of p_{green}.

4 Discussions

After analysis of the U_i we now hypothesise that more complicated behavior like betrayal and sanctioning could emerge naturally from the relationship score. Take for example the stalemate scenario in Figure 5 initial mutual blocks keep all players in deadlock, but repeated rounds cause Red's relationships $U_{\text{red}\to\text{pink}}$ and $U_{\text{red}\to\text{yellow}}$ to diverge, ultimately enabling a cooperative breakthrough.

Our baseline analysis shows signs that even Baseline Negotiator gain substantially from just ToM-k. When k is too high however, the agents starts overestimating the quality of deals made by other agents. Perhaps there is a need for a mechanism that dynamically lowers k.

The introduction of a relationship score shows sign of general agents that can perform complex negotiation without the need to construct specific Deviators or Defensive Agents. By updating a single relationship variable per opponent we believe that agents can exhibit nuanced behaviors such as alliances, betrayals, and sanctions.

Across fully simulated scenarios, agents will fall into patterns that mirror human "long-term friendship" (sustained mutual support) or "backstab" (sudden switches when Nash scores change). We define this as agent alignment, and believe this is what agents should work to uphold with their allies. It is not enough to only uphold a high relationship score with your ally, you also never want to give them the opportunity to betray you.

4.1 Future Development

While our theoretical framework demonstrates how higher-order Theory of Mind and relationship dynamics can enhance agent behavior in Diplomacy, several aspects remain to be looked deeper into. First, we plan to implement, agent-based simulations to empirically verify our hypothesis. Relationship-aware ToM–k agents against standard Baseline Negotiators might measure metrics such as deal-acceptance rates, alliance longevity, and win probabilities. These experiments will test relationship variable and may uncover aspects not captured in our static analysis.

Also, we want to explore the relationship update rate γ, and proposal-set size K—to identify regions that yield compelling strategic behaviors (e.g., stable alliences, timely betrayals, or sanctioning). Techniques like grid search or Bayesian optimization might help us tuning these variables to design agents with tailored social strategies.

Finally, recognizing that real Diplomacy often involves multi-party treaties, we aim to extend our MBDS framework and ToM–k reasoning to support n-agent deals. This will allow us to study how higher-order

reasoning scales in more complex negotiation settings and whether new forms of strategic complexity emerge.

5 Conclusions

In this paper, we have examined DeepMind's negotiation [3] framework for the game of Diplomacy and proposed extensions based on higher-order Theory of Mind and modelling relationships . Starting from honest Baseline Negotiators, which always honor agreements, we formalized how first- and second-order ToM agents can anticipate opponent behavior with BATNA simulation. We then introduced a rudementary relationship score that updates after each round based on STAVE utility estimates, and showed how incorporating this score into the state-value function yields richer, more human-like behaviors—such and might introduce alliance formation, betrayal, and sanctioning—even without explicit Deviator or Defensive Agent protocols.

Our theoretical analysis suggests that even minimal extensions a single relationship variable per opponent and a higher-order reasoning can produce complex strategic patterns, including sustained cooperation and sudden betrayal. These mirror observed elements in human play and point unified method for modeling both trust and mistrust in multi-agent negotiations. By focusing on the honest Baseline Negotiator as our core building block, we establish a clear foundation for future work on more sophisticated agents and multi-party treaties.

Overall, our contributions lay the groundwork for empirically validating these ideas through simulation, tuning behavioral hyperparameters, and ultimately integrating human opponents. We believe this framework offers a promising path toward developing Diplomacy agents that not only achieve strong performance but also exhibit the social reasoning of humans.

References

[1] Anton Bakhtin, Noam Brown, Emily Dinan, Gabriele Farina, Colin Flaherty, Daniel Fried, Andrew Goff, Jonathan Gray, Hengyuan Hu, Athul Paul Jacob, et al. Human-level play in the game of diplomacy by combining language models with strategic reasoning. *Science*, 2022.

[2] Harmen de Weerd, Rineke Verbrugge, and Bart Verheij. Agent-based models for higher-order theory of mind. *Advances in Intelligent Systems and Computing*, 229:213–232, 01 2014.

[3] János Kramár, Tom Eccles, Ian Gemp, Andrea Tacchetti, Kevin R McKee, Mateusz Malinowski, Thore Graepel, and Yoram Bachrach. Negotiation and honesty in artificial intelligence methods for the board game of diplomacy. *Nature Communications*, 2022.

[4] David Premack and Guy Woodruff. Does a chimpanzee have a theory of mind. *Behavioral and Brain Sciences*, 1:515 – 526, 12 1978.

[5] Harmen Weerd, Rineke Verbrugge, and Bart Verheij. Negotiating with other minds: the role of recursive theory of mind in negotiation with incomplete information. *Autonomous Agents and Multi-Agent Systems*, 31(2):250–287, March 2017.