

AaNanteLucky: An Agent Submitted to the ANAC 2026 ANL League

Rinon Asanuma
Tokyo University of Agriculture and Technology
asanuma@katfujii.lab.tuat.ac.jp

June 11, 2026

Abstract

This report describes AANANTELUCKY, a negotiation agent for the ANAC 2026 ANL league. The agent is implemented as a BOA-style modular negotiator consisting of an offering policy, an acceptance policy, and opponent-modelling components. The main design goal is to obtain high own utility while avoiding predictable concession behaviour and unsafe agreements. To this end, the agent combines a utility-driven time-dependent offering policy with mild preference concealment, a conservative but adaptive acceptance policy, a robust frequency-based opponent model, and an auxiliary early-frequency model exposed through the negotiator’s private information. The strategy deliberately separates safety, namely avoiding below-reservation agreements, from exploitation, namely using observed opponent behaviour to select mutually promising offers near the deadline.

1 Introduction

Automated bilateral negotiation requires an agent to balance two competing goals. It should protect its own utility, especially early in the negotiation, while also identifying offers that the opponent is likely to accept. In the ANL setting, this problem is complicated by limited time, unknown opponent preferences, and domain-dependent outcome spaces.

AANANTELUCKY was designed as a robust agent rather than as a strategy specialized to a single domain. The agent is implemented using the BOA architecture, in which bidding, opponent modelling, and acceptance are handled by separate components. This modular design makes it possible to tune each decision layer independently. The agent’s core idea is to start from strong self-interested offers, gradually introduce opponent-aware ranking, and use specialized late-stage rules to avoid failed negotiations when the opponent is clearly uncooperative, repetitive, diverse, sparse, or stuck.

2 The Design of AaNanteLucky

The agent extends a BOA-compatible negotiator and initializes three main components:

`ConcealingOfferingPolicy`, `SafeAcceptancePolicy`, and `RobustOpponentModel`. An additional component, `EarlyFrequencyExposedModel`, is registered as `exposed_opponent_model` and exposes a lightweight rank-oriented model through `private_info["opponent_ufun"]`.

The implementation uses normalized utility throughout most decisions. For an offer o , utility is normalized relative to the reservation value and the best known outcome:

$$U_n(o) = \frac{U(o) - U_{res}}{U(o^*) - U_{res}}.$$

This makes thresholds more comparable across domains. Outcomes below the reservation value are filtered out in the offering strategy and rejected in the acceptance strategy. The implementation also includes small helper functions for reservation-value handling, safe utility evaluation, clamping, and stable offer keys. These helpers make the policy more robust to missing utilities, non-hashable offers, and exceptional utility calls.

The offering component first enumerates or samples the outcome space up to a fixed limit and stores all outcomes that are above the reservation value. These outcomes are sorted by own normalized utility. During negotiation, the agent chooses from a utility band whose lower bound decreases over time. Opponent-aware scoring is then applied inside this band, rather than over the whole outcome space. This is important because the agent should not sacrifice too much own utility simply because an offer appears attractive to the opponent.

3 Concealing Bidding Strategy

The bidding strategy is implemented in `ConcealingOfferingPolicy`. Its baseline target utility is time dependent. The target starts very high, decreases moderately during the middle of the negotiation, and becomes more flexible near the deadline. The default target levels are 0.98 early, 0.82 in the middle, 0.52 late, and 0.30 in the final phase.

Preference concealment is achieved in three ways. First, the agent does not always offer the absolute best outcome. Instead, it searches among candidates within a small utility band near the current target. Second, recently sent outcomes are avoided when possible, using a recent-offer window that is enlarged in higher-dimensional domains. Third, the opponent model is used only with a bounded influence that increases after the middle of the negotiation. The maximum partner influence is normally 0.35, but can be raised to at least 0.36 in domains with five or more issues. Thus, early offers are mainly self-interested, while later offers can become more opponent-oriented.

The policy contains special handling for very small one-issue spaces. If only two or three acceptable outcomes exist, the agent initially chooses a deliberately non-best candidate in some cases, which reduces immediate preference revelation. In normal domains, the policy first applies the current target threshold, then narrows the candidate set to outcomes close to the best own utility within that threshold, and finally ranks candidates using own utility, local opponent-model score, distance from the opponent's latest offer, and offer-key stability.

The strategy also contains several late-stage behavioural detectors. For example, if the opponent repeatedly sends weak offers, diverse weak offers, stable medium offers, sparse middle offers, narrow sparse offers, low diverse offers, or late middle offers, the bidding policy switches to specialized candidate selection rules. These rules rank candidates using combinations of own utility, predicted opponent score, product of both scores, minimum of both scores, and distance from the opponent's latest offer. The purpose is to make the final proposals more acceptable without fully revealing or abandoning the agent's own preferences.

4 Acceptance Strategy

The acceptance strategy is implemented in `SafeAcceptancePolicy`. Its first rule is safety: an offer whose utility is at or below the reservation value plus a small margin is rejected. This prevents agreements that are worse than no agreement.

After the safety check, the policy compares the normalized utility of the received offer with a time-dependent threshold. The threshold is strict early in the negotiation, remains conservative in medium-size domains, and decreases near the deadline. The basic threshold levels are 0.90 early, 0.78 in the middle, 0.50 late, and 0.35 in the final phase, with a special final threshold of 0.30 for low-issue domains. The policy also adapts to domain structure. High-dimensional domains use more conservative middle and late floors, while medium-dimensional and broad-medium domains keep higher floors for a longer period.

Several additional acceptance rules handle repeated or strategically relevant offers. High-utility offers may be accepted before the final phase if they are repeated and the recent offer history is sufficiently diverse. Medium-high offers may be accepted late when recent opponent offers have low average utility and the agent's own sent offers show limited diversity, indicating that continued negotiation may not improve the result. In the final moments, the policy can accept very low but still positive normalized-utility offers only under specific pressure conditions, such as diverse weak final pressure or stable low-issue final pressure.

5 Opponent Model

The main opponent model, `RobustOpponentModel`, combines two frequency-based models. It uses a `HardHeaded`-style frequency model as the primary estimate and mixes in a `Smith`-frequency model with weight 0.20. The motivation is to keep the model stable while reducing the risk that a single frequency heuristic dominates all opponent predictions. Both internal models are connected to the same negotiator and receive the same preference-change and partner-proposal events.

The auxiliary `EarlyFrequencyExposedModel` records received offers and evaluates outcomes according to value matches with those offers. For tuple outcomes, it computes issue-wise match scores with recency weighting. In small domains, it additionally weights issues by concentration: if the opponent repeatedly uses the same value for an issue, that issue is treated as more informative. When fewer than three partner offers have been observed, the model adds a small inverse-own-utility prior with weight 0.18. This prevents the exposed model from becoming constant or fully tied in sparse traces.

The offering policy queries the opponent model to score candidate outcomes. These raw scores are normalized within the candidate set before ranking. This local normalization is useful because the model is used to choose among already acceptable outcomes, not to estimate absolute opponent utility.

6 Evaluation

The current implementation is intended to be evaluated along three dimensions: individual utility, agreement rate, and robustness across domains. Because the agent explicitly rejects below-reservation offers, it is expected to avoid unsafe agreements. Because the target utility remains high until the late stages, it is expected to perform well when the opponent is willing to concede. Finally, because the bidding policy includes late-stage compromise rules, it is expected to maintain a reasonable agreement rate against opponents that repeat, diversify, or stagnate in their offers.

A qualitative summary of the expected behaviour is shown in Table 1.

Phase	Main bidding behaviour	Main acceptance behaviour
Early	High own-utility offers with preference concealment	Accept only near-optimal offers
Middle	Utility-band search with limited opponent influence	Conservative, domain-aware thresholding
Late	Pattern-aware candidate ranking	Accept good repeated or medium-high offers
Final	Replay, compromise, or pressure response	Avoid failure only if offer is safely above reserve

Table 1: Qualitative behaviour of AANANTELUCKY across negotiation phases.

The main risk is that many thresholds are hand tuned. This can improve performance on known benchmark patterns but may reduce generality if the test set contains substantially different domains or opponent behaviours. Another risk is computational cost in large domains. The implementation mitigates this by limiting enumeration or sampling to at most 100,000 outcomes and by ranking only bounded candidate prefixes in several decision branches.

7 Lessons and Suggestions

The implementation suggests three lessons. First, a strong reservation-value guard is essential; without it, late-stage compromise can accidentally accept poor agreements. Second, opponent modelling is most useful when it is applied locally within a self-utility band. This avoids the common failure mode in which an agent over-optimizes for a noisy opponent model. Third, final-phase behaviour should not be a single threshold rule. Different opponent patterns, such as repetition, diversity, sparsity, and stagnation, require different forms of compromise.

For future improvement, the hand-tuned thresholds should be replaced or supplemented by validation-based tuning. The agent could also estimate opponent concession speed more explicitly and use that estimate to decide when to shift from self-oriented offers to compromise offers. Finally, the exposed early-frequency model could be integrated more directly with the main opponent model rather than being maintained as an auxiliary component.

Conclusions

AANANTELUCKY is a BOA-style ANL agent designed around conservative self-utility protection, mild preference concealment, robust frequency-based opponent modelling, and adaptive late-stage compromise. Its offering strategy starts from high-utility candidates and gradually incorporates opponent-aware ranking. Its acceptance strategy rejects unsafe offers and uses time-, domain-, and history-dependent thresholds. Its opponent model combines robust frequency-based estimates with an auxiliary early-offer model that remains useful in sparse traces. Overall, the agent aims to achieve high utility without becoming too predictable and to preserve agreement opportunities near the deadline.