Automated Negotiation League (ANL) 2025: An overview

March 20, 2025

This document provides an overview of the Automated Negotiating Agents Competition (ANAC) Automated Negotiation League (ANL) 2025. If you consider to participate, read here the challenge of this year and the practicalities for participation. For more information, see the documentation website.

1 Overview

The Automated Negotiating Agent Competition (ANAC) is an international tournament that has been running since 2010 to bring together researchers from the negotiation community. In the Automated Negotiation League (ANL), participants explore the strategies and difficulties in creating efficient agents whose primary purpose is to negotiate with other agent's strategies. Every year, the league presents a different challenge for the participating agents. This year's challenge is:

Design and build a negotiation agent for sequential multi-deal negotiation. The agent encounters multiple opponents in sequence and is rewarded for the specific combination of the deals made in each negotiation.

In previous years, ANL focused on different complex negotiation aspects, such as learning from the negotiation history or multilateral agreements, often in the context between two negotiators. This year, we extend to a one-to-many negotiation, specifically sequential bilateral negotiations between a center agent and multiple opponents, one after the other. The center agent is rewarded for the combination of all the agreements that it made with the opponents, taking into account goals such as achieving a target number of products or adhering to budget constraints. Who can build the best strategies to deal with these complex interactions?

ANL 2025 takes place at the IJCAI conference in Montreal, Canada. Winners will be rewarded with funding to join the conference.

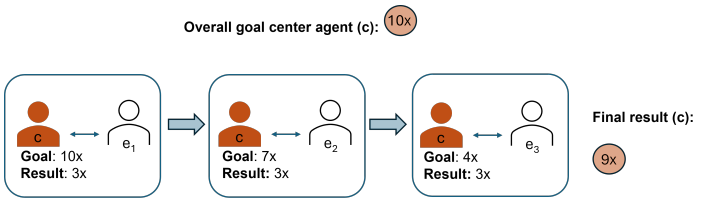

Figure 1: Example sequential negotiation.

2 Example scenario

Suppose Claire wants to plan two evenings out of three days (Mon, Tue, Wed) to eat with a different friend. Her two friends Ace and Bo live far apart while she lives in the middle, so she meets them separately. She first calls friend Ace and then calls friend Bo to set a day. In a phone call, she and her friend take turns in proposing a day, until one of them accepts or until the phone call takes too long and is ended. Because she wants to be a reliable friend, she doesn't cancel or change an agreement after it was made. She tries to maximize her utility ranging from 0 to 1. We summarize her preferences

above in the table. Note that she only cares on what day she meets, not with whom. Therefore we shorten the outcomes (None, Mon) and (Mon, None) to (Mon).

Monday	Tuesday	Wednesday	Utility
	V		0.7
V			0.5
		V	0.4
V	V		0.3
	V	V	0.3
V		V	0.9

She could call friend A, and propose Tuesday, since Tuesday alone is better than Monday or Wednesday ($0.7 > 0.5 > 0.4$). However, then she cannot combine that deal with an appointment with friend B. The best option of all is one meeting on Monday, one on Wednesday (utility is 0.9). However, she does not know in advance if her friends are willing to meet exactly then. So, she should think ahead about the options to find the best deal possible, and balance the risks. Taking the risk could go wrong. If she first tries for a meeting on Monday with friend A and she succeeds, but the negotiation with friend B fails, she ends up with a utility of only 0.5 (see the picture below).

3 Motivation

Consider a collector agent tasked with acquiring 10 paintings, and negotiating with several art suppliers. They take turns in proposing a bid with a painting and associated price. These negotiations are interconnected; the success (or failure) in early negotiations impact the strategies in subsequent negotiation. Think of a budget constraint, forcing the agent to look ahead: it shouldn't spend all the money in the first negotiation! Or the task of a total desired quantity of 10, so the agent cannot buy 6 from the first seller and 7 from the second, as that together exceeds 10.

These challenges that arise from acquiring multiple deals will be tackled in this year's competition. Negotiations with many opponents that allow for multiple deals are difficult to oversee: each negotiation influences other negotiations as well. To simplify the challenge, the negotiations in ANL 2025 will take place in sequence, one after the other, instead of simultaneously, focusing purely on a multi-deal approach. The challenge is to strategize ahead of time across all sequential negotiations to align the outcomes with each opponent and maximize the reward. The strategies build in this competition can be a contribution in solving these complex negotiations with multiple deals in negotiations that take place at the same time as seen in e.g. procurement and supply chain management.

4 Practicalities

When? The deadline for submitting your agent is June 1, 2025, 23.59 GMT.

What? Participants submit their agent source code and (optional) academic report to the submission portal. Specifics on the content can be read in the next sections.

Why? It is fun! And you can help the negotiation community further in research. Also, the winners may join the IJCAI conference in Montreal, Canada (16th to 22nd August, 2025) and possibly give a brief presentation there.

Where? The ANL will use the platform NegMAS. NegMAS supports submissions in Python. You can find instructions on how to start your ANL agent in the next sections, see further tutorials on designing your agent for ANL here, and if necessary check the general NegMAS documentation here.

What next? The next step would be to register on the submission portal. Then you can read the rest of this document and continue to the tutorials on the ANL documentation website.

Further details

So far you have read everything you need to decide whether to join the competition! The rest of this document contains further information on the used protocol and designed scenarios, and closes off with some practical information on how to submit and what rules to follow.

5 Detailed setup

In the ANL, all submitted agents will participate in a one-to-many negotiation tournament. At the start of each negotiation, one agent will receive the role *center agent*, the others will receive the role *edge agents*. The center agent will negotiate against multiple opponents (the edge agents) one after the other, in what we call *subnegotiations*.

Note that a submitted agent needs to handle both the role of center agent and the role of edge agent.

5.1 Protocol

In ANL, each one-to-one subnegotiation follows the Alternating Offers Protocol [1]. Here, the starting agent makes an opening offer, which is followed by the other agent performing one of these three actions:

- Accept the offer of the opponent agent.
- Make a counteroffer (thus rejecting and overriding the previous offer).
- Walk away (i.e. ending the negotiation without any agreement)

This process is repeated in a turn-taking fashion until reaching an agreement or passing the deadline. The deadline that we use in the competition is variable, it could range from 10 to 1000. After the deadline is reached, the center agent continues to the next one-to-one negotiation with another opponent.

5.2 Multi-deal scenarios

In a multi-deal negotiation, one center agent tries to reach agreements with one or more opponents. They can negotiate about anything: arranging a meet up, buying products, a salary negotiation etc.. All possible outcomes together is called *outcome space*, that consists of one or multiple *issues* (e.g. the day of the week, or the meeting place). The agents all have different preferences about what the outcome of the negotiation. The preferences of the agent are modeled using a *utility function*, mapping each outcome of the negotiation to a value ranging from 0 to 1. This *preference profile* of your own agent is fully known, but the preference profile of your opponent is private. The outcome space and preference profiles together is called a scenario.

To give an intuition about different types of multi-deal scenarios, we introduce a few example scenarios. We provide these on the ANL documentation website, which you can use to test your agent. *Important to note that the aim is to design a general negotiation agent, and thus the agent should be able to deal with all types of scenarios.*

Dinner scenario

You have already seen the first simple example called the dinner scenario in Section 2. Claire (the center agent) negotiates with multiple of her friends (the edge agents) in sequence.

There are 6 different outcomes: (Mo), (Tu), (We), (Mo, Tu), (Mo, We), (Tu, We). The center agent has a preference for one outcome over the other, modeled as a utility value) associated with each outcome.

The edge agents negotiate about a smaller outcome space; for them the only possible outcomes are (Mo), (Tu), (We). They also have utility associated with each outcome. For example, edge agent 1 (friend Ace) has the following preferences:

Monday	Tuesday	Wednesday	Utility
V			0
	V		0.16
		V	1

Figure 2: Preferences of edge agent 1 (friend Ace).

Target Quantity

When doing groceries or buying products on a market, a typical goal is a list of goods to buy. For example, consider a buyer agent that negotiates with four sellers. The buyer has a target: Buy a quantity of 10 products. The sellers all have a preference to sell 4 products, but they can sell up to 5 products.

We can model this scenario with the following preference profiles. The center agent has a peak preference (1) when buying 10 products; buying more or less gradually decreases the utility till 0.

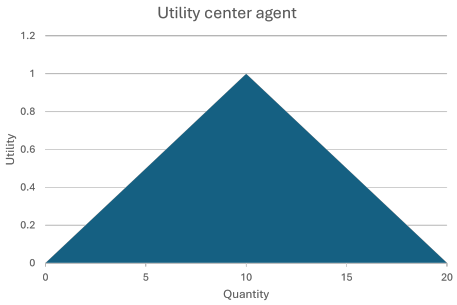

Figure 3: Preferences of the buyer (center agent).

The edge agents show a similar peak preference, centered around a peak of 4 products. An example of such preference profile is portrayed below.

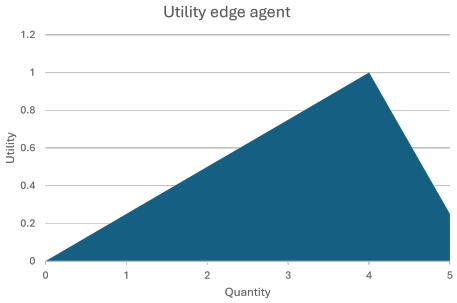

Figure 4: Preferences of the seller (edge agent).

The difficulty as a buyer agent is the large number of possibilities that can lead to the perfect outcome. Should one try for 5 and 5? Or 4 and 4 and 2? The optimal strategy depends on the opponents as well. The seller agent's challenge is to find a balance between stubbornness and concession. If their negotiation is the last negotiation of the center agent, the seller agent is in a stronger position than when their negotiation is the first.

Job hunt

A job hunter often has multiple interviews in a row with potential employers. The job hunter has a favorable position: if one negotiation fails, he can divert to the next one. Therefore, he can be risk-taking in trying to negotiate on the best job conditions.

In this scenario, a job hunter (center agent) negotiates about two issues: the number of office days and the salary. In the end, he receives the maximum of all the deals that he gathered. His preferences within a subnegotiation (called *side utility function*) are modeled using a *linear additive function*, i.e. they will be linear in the number of issues. For example, with the two issues in this example, the utility function can be computed by a *weighted sum* of the values associated with each of these issues. So, let $bid = \langle i_1, i_2 \rangle$ be a particular bid, for example (2, 150), with '2' the number of working days, and 150 the salary, as percentage of minimum wage. Then the utility $u(bid) = u(i_1, i_2)$ (given weights w_1, w_2) can be calculated by:

$$u(i_1, i_2) = w_1 \cdot u(i_1) + w_2 \cdot u(i_2).$$

Whereas the previous two example scenarios modeled the center utility function as mapping between the outcomes (e.g. quantity 5) and the utility (e.g. utility 0.5), is the center utility function in this scenario modeled as a maximum function of the utilities achieved in the individual subnegotiations. Therefore, the center utility function is *maximum* of the *side utilities*.

Further examples

The examples that we introduced do not cover all properties that a multi-deal negotiation could show. For example, all subnegotiations negotiate over the same outcome space: Claire negotiates about the same three days of the week with both friends, and the job hunter negotiates about the same job characteristics with all employers. This is not necessarily the case in the scenarios that your agent will be tested on in the competition tournament.

Furthermore, in the example scenarios, a deal with edge agent 1 has the same value as a deal with edge agent 2; i.e. the scenarios are *opponent independent*. However, this can be different in other cases, for example when buying products from preferred suppliers, so keep in mind that *opponent dependency* can appear in the test scenarios as well.

Your agent is a *general negotiation agent* and should not be tailored to the example scenarios.

6 Evaluation

When the submission period is closed, a final tournament with all agents is run. The tournament settings will cover a range of different test scenarios, inspired by the scenarios introduced in Section 5.2, with a variety of parameters such as the maximum number of rounds and the number of edge agents. To balance out the effect of different parameters on the performance of the agents, we use a sampling method.

Since this year's challenge focuses on multi-deal negotiation, we proportionally weigh the results that agents receive as center agent and as edge agents. Let μ_c the average utility gained from negotiations with the assigned role of center agent, and let μ_e be the average utility as edge agent. Then the final score A will be:

$$A = \frac{\mu_c + \mu_e}{2}.$$

7 Rules of participation

- Agents need to follow the AOP protocol in a one-to-one negotiation.
- Opponents are encountered multiple times, but it is strictly **not** allowed to save any information in memory or on HDD between different sequential negotiations (e.g. no change of global or class-level variables). (Note: a center agent can use local variables for information from one subnegotiation to the next, but not from one multi-deal negotiation to the next).
- Violating the spirit of fair play, e.g. exploiting bugs in the code, will result in disqualification. The ANAC board will be the judge in these matters.
- The competition rules allow multiple entries from a single institution but require each agent to be developed independently.
- No participant can be a co-author of more than three agents.

- The source code of agents must be submitted. This code will be included in the ANL package for future use after the competition is finished.

8 Submissions

Participants submit their agent source code and academic report (optionally) to this submission portal. The deadline for submitting your agent is **June 1, 2025, 23.59 GMT**.

Academic report

Each participant is asked to prepare a 2-4 page report describing the design of their agent according to academic standards. The best teams can give a brief presentation describing their agent depending on the available slots at IJCAI 2025.

As guidelines, the report should address the following aspects:

- Coordination: How the center agent deal with multiple deals.
- Bidding Strategy: How the agent generates bids each turn.
- Acceptance Strategy: How the agent accepts or rejects a bid.

9 Getting started

This year, the ANL will use the platform NegMAS, with a package called ANL2025. NegMAS supports submissions in Python.

- **Tutorials:** You can follow tutorials how to use the package ANL2025 on the ANL documentation website. Follow the steps there to **install** ANL and to **design your first negotiator**.
- **ANL Template:** If you installed ANL and NegMAS successfully, you can download the ANL template for building an agent. The skeleton contains a number of files, including a READ ME and the files for the tutorials. In the README, you can find how to run your first tournament using the ANL template.
- **Negmas documentation:** There is a general documentation available for the platform NegMAS here.

Good luck!