Contents

Multideal options

I use the following terminology:

- N Number of edges.
- $u_c(\omega^1, \ldots, \omega^N)$ The utility of center of outcomes ω^i in negotiation thread i where $\omega^i = \phi$ indicates disagreement.
- $\Omega_i^+ = \Omega_i \bigcup \{\phi\}$ is the outcome space of thread i plus the special value ϕ for disagreement and reserved value is the value of this special outcome $rv = u_i(\phi)$.

Timing Options

Negotiation Thread Stepping options

- One-offer-per-step (Atomic Steps): Every step is one action from one negotiator. In AOP this is a single counter-offer (offer if the first one) without reply
- One-round-per-step: Every step is one action from **every** negotiator. In AOP this is one counter offer from each partner.
- General: This is the general stepping available in Generalized Bargaining Protocols in which one or more negotiators are active at any time-step.

Negotiation Timing Options

- Sequential: Negotiations on different threads are conducted one by one. This is `sequential` in `Mechanism.runall()`
  - This can be made dynamic by making the order depends on what happens in previous negotiation threads.
- Statically Ordered: Negotiations are **stepped** in some predefined order but only one is active at any step. This is `ordered` in `Mechanism.runall()` with `ordering` passed.
- Dynamically Ordered: Negotiations are stepped one at a time but with with an order that is determined dynamically based on the states of negotiations. Not sure how interesting is this category. This is `ordered` in `Mechanism.runall()` with `ordering_fun` passed.
- Concurrent: All negotiations are stepped together. This is `ordered` with `keep_order=False` passed.
- General: This is the general stepping available in Generalized Bargaining Protocols in which one or more negotiations are active at any time-step. This is the timing function in `GBMechanism`.

All of these options are straightforwardly supported in negmas now.

Types of Center Utility Functions

Some important types of center utility functions

1. Global Center Utility Function: (This is what we have now in `anl2025tmp`).
   - The center ufun is only defined for the set of agreements received with no utility attached to each individual agreement (except by assigning some result like disagreement for the rest).
   - $u_c(\omega^1, \ldots, \omega^N) : \Omega_1^+ \times \cdots \times \Omega_N^+ \to \mathfrak{R}$ Any general mapping of outcomes to utility
   - The function u_c may be symmetric in permutations of agreements if the outcome-space is shared. We call the resulting center utility function homogeneous
2. Homogeneous Accumulating Center Utility Function:
   - The center ufun is a function of some function of agreement *issue values*.
   - All negotiation threads share the same outcome space: $\Omega_1^+ = \Omega_2^+ = \cdots = \Omega_N^+ = \Omega^+$.

   - The outcome space is a Cartesian product of issue spaces: $\Omega = V_1 \times \cdots \times V_j \cdots \times V_M$ where M is the number of issues.
   - **Issue** accumulation functions that are symmetric over permutation of parameters: $\nu_j(\{\omega_j^1, \omega_j^2, \ldots, \omega_j^N\}) : V_j^N \rightarrow \Pi^j$ where ω_j^k is the value of issue j of the agreement on thread k and Π^j is the range of ν_j.
   - $u_c(\{\omega^1, \ldots, \omega^N\}) : (\Omega^+)^N \rightarrow \mathfrak{R} = \sum_{j=1}^{M} \nu_j(\{\omega_j^1, \omega_j^2, \ldots, \omega_j^N\})$
   - If the outcome-space is not shared, singleton issues (i.e. issues with a single value) with value 0 can be added to each one to make the outcome space homogeneous.
   - Examples include collecting items from multiple suppliers without considering price.
   - SCMLOneShot is *almost* of this type (price has very little non-zero effect). The only extra complication is that the utility functions of edges are themselves changing during the negotiation (not only the reserved values) because they are centers of other negotiation sets (Many-to-Many).

3. Generalized Homogeneous Accumulating Center Utility Function:

   - The center ufun is a function of some function of agreement *issue values*.
   - All negotiation threads share the same outcome space: $\Omega_1^+ = \Omega_2^+ = \cdots = \Omega_N^+ = \Omega^+$.
   - The outcome space is a Cartesian product of issue spaces: $\Omega = V_1 \times \cdots \times V_j \cdots \times V_M$ where M is the number of issues.
   - **Multi-Issue** accumulation functions that are symmetric over permutation of sides (i.e. swapping all issues from side x with all issues from side y): $\nu_k(\omega^1_{g_1^k}, \ldots, \omega^1_{g_{n_k}^k}, \ldots, \omega^N_{g_1^k}, \ldots, \omega^N_{g_{n_k}^k}) : (V_{g_1^k} \times \cdots \times V_{g_{n_k}^k})^N \rightarrow \Pi^k$ where ω_j^l is the value of issue j of the agreement on thread l and Π^k is the range of ν_k, n_k is the number of issues involved in the evaluation of ν_k, $G^k = \{g_1^k, \ldots, g_{n_k}^k\}$ is the set of issues involved in ν_k.
   - $u_c(\{\omega^1, \ldots, \omega^N\}) : (\Omega^+)^N \rightarrow \mathfrak{R} = \sum_{j=1}^{K} \nu_k(\omega^1_{g_1^k}, \ldots, \omega^1_{g_{n_k}^k}, \ldots, \omega^N_{g_1^k}, \ldots, \omega^N_{g_{n_k}^k})$
   - If the outcome-space is not shared, singleton issues (i.e. issues with a single value) with value 0 can be added to each one to make the outcome space homogeneous.
   - Examples include collecting items from multiple suppliers while considering price.
   - SCMLStd is of this type. The only extra complication is that the utility functions of edges are themselves changing during the negotiation (not only the reserved values) because they are centers of other negotiation sets (Many-to-Many).

4. Side Utility Function Aggregation: $u_c(\omega^1, \ldots, \omega^N) : \mathfrak{R}^N \rightarrow \mathfrak{R} = f(u_1(\omega^1), \ldots, u_N(\omega^N))$ Any general mapping of utilities on each side. $u_i : \Omega_i^+ \rightarrow \mathfrak{R}$ are called *side utility functions*

   - Max is a special case of this where $f = \max$
   - The function f may be symmetric in permutations of agreements if the outcome-space is shared. We call the resulting center utility function homogeneous

5. Generalized Utility Function Aggregation: Any general mapping of utilities on each **group of sides**. $u_s(\omega^{g_1^1}, \ldots, \omega^{g^s_{n_1}}) : \Omega^+_{g_1^s} \times \cdots \times \Omega^+_{g^s_{n_s}} \rightarrow \mathfrak{R}$ are called **multi-side utility functions** where n_s is the number of sides involved with multi-side utility function u_s, $G^s = \{g_1^s, \ldots, g_{n_s}^s\}$ is the set of these sides (threads).

   - $u_c(\omega^1, \ldots, \omega^N) = f(u_1(\ldots), \ldots, u_s(\ldots))$
   - This can model cases where the thread negotiations are divided into groups and utility is defined in terms of what we get in each group instead of each individual negotiation thread.
   - Examples include a company negotiating with two types of suppliers (e.g. to buy engines and wheels for its car production).

All of these options are supported in negmas now.