Alternating Offers Protocols for Multilateral Negotiation

Reyhan Aydoğan[1,2] and David Festen[1], Koen V. Hindriks[1], and Catholijn M. Jonker[1]

[1] Interactive Intelligence Group, Delft University of Technology, The Netherlands
[2] Computer Science Department, Özyeğin University, Istanbul,Turkey
{R.Aydogan,D.Festen,K.V.Hindriks,C.M.Jonker}@tudelft.nl

Abstract. This paper presents a general framework for multilateral turn-taking protocols and two fully specified protocols namely Stacked Alternating Offers Protocol (SAOP) and Alternating Multiple Offers Protocol (AMOP). In SAOP, agents can make a bid, accept the most recent bid or walk way (i.e., end the negotiation without an agreement) when it is their turn. AMOP has two different phases: *bidding* and *voting*. The agents make their bid in the bidding phase and vote the underlying bids in the voting phase. Unlike SAOP, AMOP does not support walking away option. In both protocols, negotiation ends when the negotiating agents reach a joint agreement or some deadline criterion applies. The protocols have been evaluated empirically, showing that SAOP outperforms AMOP with the same type of conceder agents in a time-based deadline setting. SAOP was used in the ANAC 2015 competition for automated negotiating agents.

Keywords: Multilateral negotiation, Turn-taking negotiation protocol, Alternating offers protocol

1 INTRODUCTION

Multilateral negotiation is an important form of group decision making [2,6]. In many aspects of life, whether in a personal or a professional context, consensus decisions have to be made (e.g., setting the agenda in a business meeting, and the time, and location of the meeting). The complexity of multilateral negotiation increases with the number of negotiating parties [16], and with the complexity of the negotiation domain (see, e.g., [13]). The more complex the negotiations, the more human negotiators may have difficulty in finding joint agreements and the more they might benefit from the computational power of automated negotiation agents and/or negotiation support tools.

For bilateral negotiation the main challenges are opponent modeling, bidding and acceptance strategies have been extensively studied in the multi-agent community [7]. The brunt of the work is based on the alternating offers protocol to govern the interaction between negotiating agents. According to this protocol, one of the negotiating parties starts the negotiation with an offer. The other party can either accept or reject the given offer. By accepting the offer, the negotiation ends with an agreement. When rejecting the offer the other party can either end the negotiation (walk away), or make a counter offer. This process continues in a turn-taking fashion.

This paper presents a general framework for multilateral turn-taking negotiation protocols, in which fundamental definitions and rules are described formally. Based on this formal framework, we define two negotiation protocols, namely *Stacked Alternating Offers Protocol* (SAOP) and *Alternating Multiple Offers Protocol* (AMOP). In both protocols, negotiating agents can only take their action when it is their turn, the turn taking sequences are defined before the negotiation starts. SAOP allows negotiating agents to evaluate only the most recent bid in their turn and accordingly they can either accept that bid or make a counter offer or walk away. By contrast, in AMOP all agents bid sequentially and then, they vote on all bids iteratively (i.e., either accept or reject). Consequently, agents can see each agent's opinion on their bid. As a result, in AMOP the agents have a better overview of the outcome space (e.g., which bids are acceptable or not acceptable for their opponents). On the other hand, the communication cost is higher in contrast to the stacked alternating offers protocol. SAOP was used in the ANAC 2015 competition for automated negotiating agents that was organized to facilitate the research on multilateral negotiation. AMOP was developed as an alternative in which agents can get more information from their opponents by getting votes from all agents on all bids made.

To see how well the agents perform in each protocol and to judge the fairness of the outcomes, we implemented both protocols in Genius and compared them empirically. The current results show that SAOP outperforms AMOC on the given negotiation scenarios with respect to the social welfare criterion.

The rest of this paper is organized as follows. Section 2 presents the general framework for multilateral turn-taking protocols. The stacked alternating offers protocol and alternating multiple offers protocol are explained in Section 3 and Section 4 respectively. Section 5 explains our experimental setup, metrics and results. Section 6 discusses the related work. Finally, we conclude the paper with directions to future work in Section 7.

2 Formal Framework for Multilateral Turn-taking Protocols

Before presenting two variants of turn-taking protocols for multilateral negotiation, we first introduce a general formal framework for specifying these protocols. The framework consists of a number of general definitions regarding alternating offers protocols for multilateral negotiations. In later sections where we present two turn-taking protocols for multilateral negotiation, those concepts that are protocol dependent will be revisited.

Basic notation The basic notions of a negotiation are the agents that negotiate, the bids that they exchange, and the other actions that they can take during the negotiation. We use Agt to denote a finite set of agent names, Bid to denote a set of bids over the negotiation domain, and $Act \subseteq Bid \cup \{accept, reject, end\}$ to denote a set of possible actions that can be taken during the negotiation where *end* denotes that the agent walks away. In this document tuples and sequences are used frequently. For any tuple or sequence t and any index $i \in \mathbb{N}$, let t_i denote the i^{th} element of tuple t, and similarly, for any tuple, sequence or set t, let $|t|$ denote the number of elements in t.

Definition 1. *Round and Phase.*
Rounds and phases within rounds are used to structure the negotiation process. Although the structure of the phases differs over protocols, the concepts are defined generally as follows:

- *$Round \subseteq \mathbb{N}^+$ is the set of round numbers. Rounds are numbered from 1 onwards, if i is the current round, then the next round is numbered $i+1$.*
- *$Phase \subseteq \mathbb{N}$ is the set of phase identifiers. Phases are numbered from 0 onwards, if i is the current phase, then the next phase is numbered $i+1$. The set Phase can be a singleton. Let ℓ denote the last phase, which is equal to $|Phase| - 1$.*
- *$RPhase = Round \times Phase$, the first argument denotes the round number whereas the second argument denotes the specific phase of that round. This depends on the protocol at hand. In case Phase is $\{\ 0\ \}$, then, for convenience, RPhase is collapsed to Round only.*

Definition 2. *Turn taking.*
Alternating offer protocols assign turns to the negotiating agents. Turns are taken according to a turn-taking sequence.

- *$TurnSeq = Agt^{|Agt|}$ is a sequence of agents, such that*
  - $\forall\ s \in TurnSeq\ \forall\ a \in \mathrm{Agt}, \exists i \in \mathbb{N}^+, i \leq |s|$ *such that* $s_i = a$ *and*
  - $\forall\ s \in TurnSeq\ \forall\ i, j \leq |s| : s_i{=}s_j \rightarrow i{=}j$.
- *The function $rpSeq : RPhase \rightarrow TurnSeq$ assigns a turn-taking sequence per round and phase. Its specification depends on the protocol.*
- *The function $prev : RPhase \times \mathbb{N}^+ \rightarrow RPhase \times \mathbb{N}^+$ defines the previous turn in the negotiation, that can be in this round-phase or a previous round-phase, specified by:*

$$prev(r,t) = \begin{cases} \langle r, t-1 \rangle, & 1 < t \leq |Agt| \\ \langle\langle r_1, r_2 - 1\rangle, |Agt|\rangle, & t = 1 \wedge r_2 > 0 \\ \langle\langle r_1 - 1, \ell\rangle, |Agt|\rangle, & t = 1 \wedge r_2 = 0 \wedge r_1 > 1 \\ \textit{undefined}, & \textit{otherwise} \end{cases} \tag{1}$$

To be able to specify what happened $k \in \mathbb{N}$ turns ago, we recursively define $prev^k : RPhase \times \mathbb{N}^+ \rightarrow RPhase \times \mathbb{N}^+$ as follows:

$$\begin{aligned} \forall x \in RPhase \times \mathbb{N}^+ : \\ prev^0(x) &= x \\ prev^1(x) &= prev(x) \\ prev^{n+1}(x) &= prev^n(prev(x)) \end{aligned}$$

The conditions ensure fairness in protocols in the sense that every agent gets a turn and no agent gets more than one turn in a sequence. In case the same turn-taking sequence is used in all rounds and phases, this sequence is denoted by s. This is true for the protocols SAOP and AMOP of the later sections. However, Definition 2 allows more freedom.

Although the actions might differ over protocols, we introduce notions that are general to all negotiation protocols.

Definition 3. *Actions and allowed actions.*
The functions action *and* allowedAction *specify what actions agents take and what actions they are allowed to take.*

- $action : Agt \times RPhase \rightarrow Act$. *The term $action(a, r)$ denotes what action agent $a \in Agt$ took in round-phase $r \in RPhase$.*
- allowedAct*: $RPhase \times \mathbb{N}^+ \rightarrow \mathcal{P}(Act)$. The function determines the allowed actions per turn t at a given round-phase r. The function specification varies over protocols.*

Although protocols do not specify what actions agents take during the negotiation, the function *action* is defined here, as the type action taken by the agents do have an effect on the procedure as specified in Definitions 5, and 6.

Definition 4. *Deadline.*
$d : RPhase \times \mathbb{N}^+$ is a predicate that denotes whether or not the negotiation deadline has been reached. Its value is determined at the end of the current turn. Its specification depends on the protocol.

Examples of such criteria are round-based ($r > R_{deadline}$), and time-based ($time > T_{deadline}$).

Definition 5. *Agent ending the negotiation.*
The predicate $endP : RPhase \times \mathbb{N}^+$ denotes whether or not an agent has ended the negotiation. Its value is determined at the end of the current turn.

$$\forall r \in RPhase\ , \forall t \in \mathbb{N}^+ : endP(r, t) \leftrightarrow action(rpSeq(r)_t, r) = end \tag{2}$$

Note that, in typical protocols, the negotiation terminates as soon as one of the negotiators walks away, i.e., takes the action *end*. However, there might be protocols in which the other negotiators might continue. In that case Definition 7 that determines whether a negotiation continues will have to be adapted.

Definition 6. *Agreement.*
For use in the next predicates and functions two predicates are introduced to identify when an agreement has been reached and what that agreement is.

- *The predicate agr: RPhase* $\times \mathbb{N}^+$ *denotes whether or not an agreement is reached. Its value is determined at the end of the current turn. The exact specification varies over protocols.*
- *The predicate agrB : Bid* $\times$ RPhase $\times \mathbb{N}^+$ *denotes the bid that was agreed on.*

Definition 7. *Continuation.*
The predicate cont*:* RPhase $\times \mathbb{N}^+$ *denotes whether the negotiation continues after the current turn. Its value is determined at the end of the current turn.*

$$\forall r \in RPhase\ \forall t \in \mathbb{N}^+ : cont(r,t) \leftrightarrow \neg d(r,t) \wedge \neg endP(r,t) \wedge \neg agr(r,t) \tag{3}$$

Definition 8. *Outcome of the negotiation.*
The function outcome: Round $\times \mathbb{N}^+ \rightarrow$ Bid $\cup \{fail\}$ *that determines the negotiation outcome at the end of the current turn.*

$$outcome(r,t) = \begin{cases} undefined, & cont(r,t) \\ fail, & \neg cont(r,t) \wedge \neg agr(r,t) \\ b, & t > 0 \wedge \neg cont(r,t) \wedge \ agrB(b,r,t) \end{cases} \tag{4}$$

Definition 9. *Turn-taking Negotiation protocol.*
A turn-taking negotiation protocol P *is a tuple* $\langle$ *Agt, Act, Rules* $\rangle$ *where Agt denotes the set of agents participating in the negotiation, Act is the set of possible actions the agents can take, and Rules is the set of rules that specify the particulars of the protocol. It contains the following rules, or specializations thereof.*

1. *Turn-taking Rule 1: Each agent gets turns according to the turn taking sequences of the protocol as specified by the definitions for rounds, phases, and turn-taking.*
2. *Turn-taking Rule 2: There is no turn after the negotiation has terminated, according to the Termination Rule.*
3. *Actions Rule 1: The agents can only act in their turn, as specified by the Turn-taking Rules.*
4. *Actions Rule 2: The agents can only perform actions that are allowed at that moment, as specified by the definitions for allowed actions.*
5. *Termination Rule: The negotiation is terminated after round-phase* r *and turn* t *if* $\neg cont(r,t)$*, as defined by the definitions for continuation, agreement, deadline and agent ending the negotiation.*
6. *Outcome Rule: The outcome of a negotiation is determined by the definitions for outcome and agreement.*

The above definitions form the core of a formal framework for multilateral turn-taking negotiation protocols. There are different ways to extend the bilateral alternating offers protocol to the multilateral case. The next sections introduce two variants of this protocol: Stacked Alternating Offers Protocol (Section 3) and Alternating Multiple Offers Protocol (Section 4). Both protocols are specified by providing the detailed descriptions of those predicates and functions that are protocol dependent.

3 Stacked Alternating Offers Protocol (SAOP)

According to this protocol, all of the participants around the table get a turn per round; turns are taken clock-wise around the table, also known as a Round Robin schedule[14]. One of the negotiating parties starts the negotiation with an offer that is observed by all others immediately. Whenever an offer is made, the next party in line can take the following actions:

- Accept the offer
- Make a counter offer (thus rejecting and overriding the previous offer)
- Walk away (thereby ending the negotiation without any agreement)

This process is repeated in a turn-taking clock-wise fashion until reaching a termination condition is met. The termination condition is met, if a unanimous agreement or a deadline is reached, or if one of the negotiating parties ends the negotiation. Formally, the *Stacked Alternating Offer Protocol* is defined by the following definitions. We only provide an instantiated version of those definitions that are protocol dependent, i.e., phases of the negotiation, turn taking, actions and allowed actions, agreement, and the rules of encounter. Note that we only specify what changed in those definitions with respect to Section 2. SAOP can work with any deadline, or no deadline at all.

Definition 10. *Round and Phase (Definition 1 for SAOP).*
The concept of Round is not changed, there is only one phase per round in SAOP, i.e., $Phase = \{0\}$.

Definition 11. *Turn taking (Definition 2 for SAOP).*
In SAOP the same turn taking sequence is used in all rounds. Let s denote that sequence, thus for SAOP the set of turn-taking sequences is $TurnSeq = \{s\}$.

The rules for turn taking are those specified in Definition 2, i.e., each agent gets exactly one turn per round, as specified by s. Note that, since there is only one phase per round, instead of mentioning phases per round, in SAOP only rounds are mentioned.

Definition 12. *Actions and allowed actions (Definition 3 for SAOP).*
The function action *is unchanged. The detailed specification of allowedAction: $RPhase \times \mathbb{N}^+ \to Act$ is as follows:*

$$allowedAct(r,t) = \begin{cases} Bid \cup \{end\}, & \text{if } cont(r,t) \wedge t = 1 \wedge r_1 = 1 \\ Bid \cup \{accept, end\}, & \text{if } cont(r,t) \wedge (t \neq 1 \vee r_1 \neq 1) \\ \emptyset, & otherwise \end{cases}$$

Definition 13. *Deadline (Definition 4 for SAOP).*
Predicate $d : RPhase \times \mathbb{N}^+$ denotes whether or not the negotiation deadline has been reached. Its value is determined at the end of the current turn according to the following.

$$\forall r \in RPhase\ \forall t \in \mathbb{N}^+ : d(r,t) \leftrightarrow currenttime - negostarttime \geq maxnegotime \tag{5}$$

The variables *negostarttime* and *maxnegotime* are set per negotiation. For example in the ANAC 2015 competition, the variables *currenttime* and *negostarttime* were taken from the system time of the computer running the tournament, and *maxnegotime* was set at 3 minutes.

Definition 14. *Agreement (Definition 6 for SAOP).*
The predicate agr: RPhase $\times\mathbb{N}^+$ denotes whether or not an agreement is reached. The predicate agrB : Bid $\times$ RPhase *$\times\mathbb{N}^+$ denotes the bid that was agreed on. Their values are determined at the end of turn. Their specifications are as follows.*

$$\begin{aligned}&\forall r \in RPhase,\ \forall t \in \mathbb{N}^+ : agr(r,t) \leftrightarrow \\ &\qquad action(s_{prev_2^{|Agt|-1}(r,t)}, prev_1^{|Agt|-1}(r,t)) \in Bid \wedge \\ &\qquad\quad \forall 0 \leq i \leq |Agt|-2 : action(s_{prev_2^i(r,t)}, prev_1^i(r,t)) = accept\end{aligned}$$

$$\begin{aligned}&\forall r \in RPhase,\ \forall t \in \mathbb{N}^+ : \\ &\quad agrB(action(s_{prev_2^{|Agt|-1}(r,t)}, prev_1^{|Agt|-1}(r,t)), r, t) \leftrightarrow cont(r,t) \wedge agr(r,t)\end{aligned}$$

Informally, we have an agreement iff $|Agt| - 1$ turns previously, an agent made a bid that was subsequently accepted by all the other agents. The agent that made the bid, in the SAOP protocol, is assumed to find its own bid acceptable. In $agrB$ that bid that was made $|Agt| - 1$ turns ago, is set to be the agreed bid in the current round-phase and turn.

Example Assume that there are three negotiating negotiation parties, a_1, a_2 and a_3. Agent a_1 starts the negotiation with an bid b_1. Agent a_2 can accept this bid, make a counter offer or walk way. Let assume that she decides to make a counter bid (b_2). Assume that agents a_3 and a_1 accept this offer. As they all agree on this bid (i.e. b_2 made by a_2 in the previous round), the negotiation ends, and the outcome is bid b_2.

4 Alternating Multiple Offers Protocol (AMOP)

The AMOP protocol is an alternating offers protocol in which the emphasis is that all players will get the same opportunities with respect to bidding. That is, all agents have a bid from all agents available to them, before they vote on these bids. This implemented in the following way: The AMOP protocol has a bidding phase followed by voting phases. In the bidding phase **all** negotiators put their offer on the table. In the voting phases all participants vote on all of the bids on the negotiation table. If one of the bids on the negotiation table is accepted by all of the parties, then the negotiation ends with this bid. This is an iterative process continuing until reaching an agreement or reaching the deadline. The essential difference with the SAOP protocol, is that the players do not override each other's offers and the agents can take all offers into account before they vote on the proposals. From an information theoretical point of view, this is a major difference. The specification of this protocol asks for detailed specifications of the protocol dependent definitions, i.e., on round-phases, turn taking, actions and allowed actions, agreement, and the rules of encounter. Only the changes are specified.

Definition 15. *Round and Phase (Definition 1 for AMOP).*
The concept of Round is not changed. Protocol AMOP has one bidding phase, followed by $|Agt|$ *voting phases, i.e.,* $Phase = \{0, 1, ..., |Agt|\}$ *where* 0 *denotes the bidding phase while for each* $i \in [1, |Agt|]$*,* i *denotes the voting phase on the bid made in the* i^{th} *turn.*

Definition 16. *Turn taking (Definition 2 for AMOP).*
In AMOP the same turn taking sequence is used at each phase of all rounds. Let s *denote that sequence, i.e.,* $TurnSeq = \{s\}$*.*

Definition 17. *Actions and allowed actions (Definition 3 for AMOP).*
We define the set of possible actions as $Act = Bid \cup \{accept,\ reject\}$*. The function* action *is unchanged. The detailed specification of* $allowedAction\colon RPhase \times \mathbb{N}^+ \to Act$ *is as follows:*

$$allowedAct(r, t) = \begin{cases} Bid, & \text{if } cont(r, t) \wedge r_2 = 0 \\ \{accept,\ reject\}, & \text{if } cont(r, t) \wedge r_2 > 0 \\ \emptyset, & \text{otherwise.} \end{cases}$$

All rounds starts with a bidding phase during which all agents make a bid in turn specified by the turn sequence. The bidding phase is followed by a voting phase for each bid on the table. This means that all agents first vote on the first bid that was put on the table in this round, then all votes for the second bid and so on. During each voting phase, agents take their turn according to turn taking sequence as defined by the turn taking rules. During the voting phases, agents can only accept or reject bids. That the votes in phase i, refer to the i^{th} bid in the bidding phase is specified indirectly by Definition 18.

Definition 18. *Agreement (Definition 6 for AMOP).*
The predicate $agr\colon RPhase \times \mathbb{N}^+$ *denotes whether or not an agreement is reached. The predicate* $agrB : Bid \times \mathrm{RPhase} \times \mathbb{N}^+$ *denotes the bid that was agreed on.*

Their values are determined at the end of turn in voting phases. Their specifications are as follows.

$$\forall r \in RPhase\ \forall t \in \mathbb{N}^+ : agr(r,t) \leftrightarrow$$
$$r_2 > 0 \wedge t = |Agt| \wedge\ action(s_{r_2}, \langle r_1, 0\rangle) \in Bid \wedge \forall 1 \leq i \leq t : action(s_i, r) = accept$$

$$\forall r \in RPhase\ \forall t \in \mathbb{N}^+ : agrB(action(s_{r_2}, \langle r_1, 0\rangle), r, t) \leftrightarrow cont(r,t) \wedge agr(r,t)$$

In other words, we have an agreement at the i^{th} phase of a given round-phase r, iff all agents in that round voted to accept the bid made by the i^{th} agent in the turn taking sequence s.

Definition 19. *Continuation (Definition 7 for AMOP).*
The predicate cont*:* RPhase $\times \mathbb{N}^+$ *denotes whether the negotiation continues after the current turn. Its value is determined at the end of the current turn.*

$$\forall r \in RPhase\ \forall t \in \mathbb{N}^+ : cont(r,t) \leftrightarrow \neg d(r,t) \wedge \neg agr(r,t) \tag{6}$$

Illustration In $Phase = 0$, all players put an offer on the table (b_1 by a_1, b_2 by a_2 etc). Note that there is no restriction on the bids; agents are allowed to make the same bid as others, or the same bid they made before. In the $Phase = 1$, all agents vote for the bid made by a_1, in $Phase = 2$, they all vote for the bid made by a_2 and so on. When all agents accept a bid during a voting phase, negotiation ends with this bid. Suppose that all agents, for example, vote to accept bid b_2, then the negotiation terminates at the end of phase 2 of round 1. If there were more than 2 agents, then this implies that the agents don't vote anymore for bid b_3.

5 Experimental Evaluation

In order to compare the performance of SAOP and AMOP empirically, we incorporated these two protocols into the Genius [15] negotiation platform, that was developed to enable researchers to test and compare their agents in various settings. Genius serves as a platform for the annual Automated Negotiating Agents Competition (ANAC) [3]. Our extension enables Genius to run multilateral negotiations; subsequently, the challenge of the ANAC 2015 competition was chosen to be multilateral negotiation.

A state-of-the-art agent, *Conceder* agent has been adapted for both multilateral protocols. This agent calculates a target utility and makes an arbitrary bid within a margin of 0.05 of this target utility. The target utility is calculated as $targetUtil(t) = 1 - t^{0.5}$ where $0 \geq t \geq 1$, t is the remaining time. This formula is derived from the general form proposed in [8]. In this paper, we adopted the ANAC 2015 setup, where three negotiating agents negotiate to come to an agreement within a three-minute deadline. We generated 10 different negotiation scenarios for three parties. Agent preferences are represented by means of additive utility functions. The size of the negotiation domains ranges from 216 to 2304.

To investigate the impact of the degree of conflict on the performance of the negotiation protocols, the scenarios tested in our experiment are chosen in such a way that half of those scenarios are collaborative and the rest are competitive. In competitive scenarios, there are relatively less outcomes which make everyone happy. We ran each negotiation ten times. Each agent negotiates for each preference profile in different order; that results 600 negotiations in total per each protocol (6 ordering permutations of 3 agents $\times$ 10 scenarios $\times$ 10 times).

We evaluated the protocols in term of the fairness of their negotiation outcome and social welfare. For social welfare, we picked the well known utilitarian social welfare metric [6], which is the sum of the utilities gained by each agent at the end of a negotiation. For fairness, we adopt the product of the utilities gained by each agents [12]. Recall that the Nash solution is the negotiation outcome with the maximum product of the agent utilities. Table 1 shows the average sum and product of the agent utilities with their standard deviation over 60 negotiations per each negotiation scenario. It is worth noting that the first five negotiation scenarios are cooperative and the last five scenarios are competitive. As expected, the negotiations resulted in higher sum and product of utilities when the negotiation scenarios are cooperative.

	Social welfare			Nash Product			Distance to Nash	
	SAOP	AMOP	Δ	SAOP	AMOP	Δ	SAOP	AMOP
Scenario 1	2.74 ± 0.01	2.44 ± 0.06	0.30	0.76 ± 0.01	0.54 ± 0.04	0.22	0.00 ± 0.04	0.23 ± 0.31
Scenario 2	2.36 ± 0.00	2.01 ± 0.06	0.35	0.48 ± 0.00	0.30 ± 0.03	0.18	0.11 ± 0.05	0.33 ± 0.26
Scenario 3	2.60 ± 0.00	2.38 ± 0.05	0.22	0.65 ± 0.00	0.50 ± 0.03	0.15	0.00 ± 0.00	0.18 ± 0.29
Scenario 4	2.74 ± 0.00	2.53 ± 0.06	0.21	0.76 ± 0.00	0.60 ± 0.04	0.16	0.00 ± 0.01	0.17 ± 0.34
Scenario 5	2.89 ± 0.00	2.80 ± 0.03	0.09	0.90 ± 0.00	0.81 ± 0.02	0.09	0.07 ± 0.00	0.12 ± 0.15
Scenario 6	2.20 ± 0.01	1.90 ± 0.05	0.30	0.39 ± 0.01	0.25 ± 0.02	0.14	0.00 ± 0.22	0.27 ± 0.23
Scenario 7	1.73 ± 0.01	1.59 ± 0.04	0.14	0.19 ± 0.00	0.14 ± 0.01	0.05	0.25 ± 0.06	0.38 ± 0.29
Scenario 8	2.19 ± 0.00	2.11 ± 0.02	0.08	0.39 ± 0.00	0.35 ± 0.01	0.04	0.06 ± 0.03	0.17 ± 0.17
Scenario 9	2.03 ± 0.00	1.96 ± 0.03	0.07	0.31 ± 0.00	0.26 ± 0.02	0.05	0.14 ± 0.01	0.25 ± 0.33
Scenario 10	2.06 ± 0.01	2.00 ± 0.03	0.03	0.32 ± 0.00	0.29 ± 0.01	0.06	0.14 ± 0.03	0.26 ± 0.24

Table 1. Social welfare and Nash product for cooperative domains (Scenario 1 to 5) and competitive domains (Scenario 6 to 10). All intervals are 95% confidence intervals. Sample size: $N = 60$.

When we compare the performance of two protocols with time-based conceder agents in terms of social welfare, it is obviously seen that on average SAOP outperformed AMOP in all scenarios. However, the average social welfare difference between two protocols is higher in cooperative negotiation scenarios compared to the competitive scenarios. We have similar results when we look at the average product of agent utilities. The distinction between cooperative and competitive scenarios became more visible for the product of agent utilities since there are a few outcomes that can make everyone happy. The agents gained higher product of utilities when they followed SAOP. Similarly, the negotiation outcomes in SAOP are closer to the Nash solution compared to the outcomes in AMOP. Based on the statistical t-test on both the average sum and product of agent utilities, it can be concluded that the results for SAOP with *Conceder* agent are statistically significantly better than the results for AMOP with *Conceder* agent on the given negotiation scenarios ($p \ll 0.001$).

The potential reasons why the social welfare of the agents are higher in SAOP compared to AMOP although they employ the same *Conceder* strategy stem from the main differences between SAOP and AMOP. One of these is that according to SAOP, the agents evaluate only the most recent bid in their turn whereas in AMOP, they evaluate all bids made by all agents in the current round. Although it sounds more fair to evaluate all bids made by all, the agents do not obtain a more fair outcome in AMOP. This may stem from the fact that AMOP protocol is less time-efficient protocol as it has the extensive voting phases in a round. Because they spend extra time in the voting phase, the estimated target utility in each bidding phase may be relatively lower than those in SAOP. That may be the reason the agents miss out on some good solutions for all parties. That also implies that there are less rounds within the same time period in AMOP compared to SAOP (3000 rounds versus 15000 rounds); therefore, there is less time to explore the outcome space. That is, 9000 offers were made during a negotiation in AMOP while agents made around between 22500 and 45000 offers in total in SAOP. As a future work, we would like test the protocols in a round-based deadline setting to see how their performance would be when they have the same number of rounds.

6 Discussion

The terms of *multiparty* and *multilateral* are used interchangeably in the community. In this work, we distinguish them as follows. If there are more than two participants engaged in the negotiation, it is considered a multiparty negotiation. This engagement can be in different forms such as *one-to-many*, *many-to-many* or *many-to-one* negotiations. For instance, William *et al.* propose a many-to-many concurrent negotiation protocol that allows agents to commit and to decommit their agreement [17]. Wong and Fang introduce the Extended Contract-Net-like multilateral Protocol (ECNPro) [1] for multiparty negotiations between a buyer and multiple sellers, which can be considered as multiple bilateral negotiations. In this work, we define multilateral negotiations as negotiations in which more than two agents negotiate in order to reach a joint agreement; in other words, all the negotiating parties have the same role during the negotiation process (e.g., a group of friends negotiating on their holiday), and these negotiations might or might not be mediated by an independent party that has no personal stake in the outcome of the negotiation.

The protocols proposed for multilateral negotiations in the multiagent community mostly use a mediator [2,5,9,10,11,13]. In contrast, this paper proposes protocols for non-mediated multilateral negotiations. Endriss presents a monotonic concession protocol for non-mediated multilateral negotiations and discusses what a concession means in the context of multilateral negotiation, see [6]. The monotonic concession protocol enforces the agents to make a concession or to stick to their previous offer, while our protocols do not interfere with what to bid, only when to bid. The concession steps suggested in that work require to know the other agent's preferences except for the egocentric concession step in which the agent is expected to make a bid that is worse for itself.

A generalization of the alternating offers protocol, namely, a sequential-offer protocol was used in [18]. Similar to SAOP, the agents make sequential offers in predefined turns or accept the underlying offer according to this protocol. A minor difference is that it does not provide a walk-away option for the agents as SAOP does. The core of the work is a negotiation strategy that applies a sequential projection method for multilateral negotiations. In that sense it cannot be compared to the work presented in this paper, in which two multilateral negotiation protocols are proposed and evaluated.

De Jonge and Sierra recently introduced a new multilateral protocol inspired from human negotiations, called the *Unstructured Communication Protocol* (UCP) [4]. Unlike the negotiation protocols discussed above, this protocol does not structure the negotiation process. That is, any agent may propose an offer at any time and offers can be retracted at any time. Agents can accept a given offer by repeating the same offer. When all agents propose the same offer, this offer is considered an agreement. There are some similarities between their protocol and AMOP such as the agents can see multiple offers on the negotiation table and evaluate them. Compared to AMOP, their protocol is more flexible. For example, in AMOP agents have to bid in the bidding phase and have to vote in the voting phase, whereas agents in UCP can remain silent and wait for the other agents. However, flexibility comes with a price. Designing an agent having the intelligence to deal with the uncertainties in UCP is quite a challenge: how do you decide whether the agent should bid or remain silent? How do you know if another agent is still participating or whether it walked away? What does it mean if some of the agents are silent? Although the protocol is more natural from a human point of view, the situation is different: the agents lack information that humans that are physically present in the same negotiation room would have, such as body language, tone of voice, eye contact. Our point of view is that if we would like to develop a multilateral negotiation protocol in which humans and agents are to engage each other, then we should get the protocol as close as possible to the human way of negotiating, like UCP, while realizing that developing agents that can fully understand and act in such a heterogeneous setting is still a Grand Challenge. If, on the other hand, we are aiming for agents-only negotiations, then deviating from protocols that humans would use is quite alright, which opens the door for mechanism design, game theory and, of course, strategy development for the participating agents. The alternating multilateral negotiation protocols presented in this paper, are motivated by the search for protocols for agents-only negotiations.

7 Conclusion

In this paper, we introduce two extensions of alternating offers protocol for multilateral negotiations, namely *Stacked Alternating Offers Protocol* and *Alternating Multiple Offers Protocol*. We provide formal definitions of these protocols based on a general formalization for turn-taking multilateral negotiation protocols. Furthermore, we compare the performance of these protocols with time-based Conceder agents empirically. Our results show that SAOP performed better than AMOP in terms of social welfare and fairness of the negotiation outcome on the chosen negotiation scenarios. Therefore, we make SAOP public to facilitate the research in multilateral negotiation. In ANAC 2015 the participants developed negotiating agents for three-party negotiation governed by SAOP.

As future work, we are planning to characterize negotiation protocols using properties and show to which extent these are satisfied by SAOP, UCP, AMOP, and other (new) protocols. For instance, it would be interesting to investigate the effect of the agents' ordering in a given turn sequence on the negotiation outcome (e.g., whether or not the first agent starting the negotiation has an advantage over the others). In this work we used one type of agents in our evaluation of both protocols to ensure that we are not, at the same time, comparing negotiation strategies. However, it is still an open question what makes a protocol a good protocol. In future, we plan to perform more systematic evaluations for properties that characterize negotiation protocols, such as the speed with which agreements are reached, more fairness aspects (beyond distance to Nash Product, and ordering effects), scalability, robustness against manipulative agents (e.g., truth-revealing), and communication overhead.

8 Acknowledgement

This work was supported by the ITEA M2MGrids Project, grant number ITEA141011.

References

1. A multi-agent protocol for multilateral negotiations in supply chain management. *International Journal of Production Research*, 48, 2010.
2. R. Aydoğan, K. V. Hindriks, and C. M. Jonker. Multilateral mediated negotiation protocols with feedback. In *Novel Insights in Agent-based Complex Automated Negotiation*, pages 43–59. 2014.
3. T. Baarslag, K. Hindriks, C. M. Jonker, S. Kraus, and R. Lin. The first automated negotiating agents competition (ANAC 2010). In T. Ito, M. Zhang, V. Robu, S. Fatima, and T. Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence*, pages 113–135, Berlin, Heidelberg, 2012. Springer-Verlag.

4. D. de Jonge and C. Sierra. Nb^3: a multilateral negotiation algorithm for large, non-linear agreement spaces with limited time. *Autonomous Agents and Multi-Agent Systems*, 29(5):896–942, 2015.
5. E. de la Hoz, M. Lopez-Carmona, M. Klein, and I. Marsa-Maestre. Consensus policy based multi-agent negotiation. In D. Kinny, J.-j. Hsu, G. Governatori, and A. Ghose, editors, *Agents in Principle, Agents in Practice*, volume 7047 of *Lecture Notes in Computer Science*, pages 159–173. Springer Berlin Heidelberg, 2011.
6. U. Endriss. Monotonic concession protocols for multilateral negotiation. In *Proceedings of the Fifth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 392–399, Japan, 2006.
7. S. Fatima, S. Kraus, and M. Wooldridge. *Principles of Automated Negotiation.* Cambridge University Press, 2014.
8. S. S. Fatima, M. Wooldridge, and N. R. Jennings. Optimal negotiation strategies for agents with incomplete information. In *Revised Papers from the 8th International Workshop on Intelligent Agents VIII*, ATAL '01, pages 377–392, London, UK, UK, 2002. Springer-Verlag.
9. K. Fujita, T. Ito, and M. Klein. Preliminary result on secure protocols for multiple issue negotiation problems. In *Intelligent Agents and Multi-Agent Systems, 11th Pacific Rim International Conference on Multi-Agents, PRIMA 2008, Hanoi, Vietnam, December 15-16, 2008. Proceedings*, pages 161–172, 2008.
10. H. Hattori, M. Klein, and T. Ito. A multi-phase protocol for negotiation with interdependent issues. In *Proceedings of the 2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology, Silicon Valley, CA, USA, November 2-5, 2007*, pages 153–159, 2007.
11. M. Hemaissia, E. Seghrouchni, A., C. Labreuche, and J. Mattioli. A multilateral multi-issue negotiation protocol. In *Proceedings of the Sixth International Joint Conference on Autonomous Agents and Multiagent Systems*, pages 939–946, Hawaii, 2007.
12. H. Kameda, E. Altman, C. Touati, and A. Legrand. Nash equilibrium based fairness. *Mathematical Methods of Operations Research*, 76(1):43–65, 2012.
13. M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Protocols for negotiating complex contracts. *IEEE Intelligent Systems*, 18:32–38, 2003.
14. L. Kleinrock. Analysis of a time shared processor. *Naval Research Logistics Quarterly*, 11(1):59–73, March 1964.
15. R. Lin, S. Kraus, T. Baarslag, D. Tykhonov, K. Hindriks, and C. M. Jonker. Genius: An integrated environment for supporting the design of generic automated negotiators. *Computational Intelligence*, 30(1):48–70, 2014.
16. H. Raiffa. *The art and science of negotiation: How to resolve conflicts and get the best out of bargaining.* Harvard University Press, Cambridge, MA, 1982.
17. C. R. Williams, V. Robu, E. H. Gerding, and N. R. Jennings. Negotiating concurrently with unknown opponents in complex, real-time domains. In *20th European Conference on Artificial Intelligence*, volume 242, pages 834–839, August 2012.
18. R. Zheng, N. Chakraborty, T. Dai, and K. Sycara. Multiagent negotiation on multiple issues with incomplete information: Extended abstract. In *Proceedings of the 2013 International Conference on Autonomous Agents and Multi-agent Systems*, AAMAS '13, pages 1279–1280, 2013.