Evaluating Multiparty Multilateral Turn-Taking Negotiation Protocols

Thesis by

David W.J. Festen

In Partial Fulfillment of the Requirements
for the Degree of
Master of Science

Delft University of Technology
Delft, Netherlands
2018
(Defended 15^{th} of March, 2018)

Abstract

In order to be accepted by negotiating parties, negotiation protocols need to be fair and unbiased. We extend the bilateral turn-taking negotiation protocol to multiparty multilateral turn-taking peer negotiation protocols maintaining its fairness properties. We define the design space of such protocols and capture the design decisions that go into creating them. We use this design space to introduce two novel protocols adhering to the fairness properties: Stacked Alternating Offer Protocol (SAOP) and Alternating Multi-offer Consensus Protocol (AMCP). A method for measuring speed and social welfare properties of multiparty multilateral turn-taking peer negotiation protocols is introduced and used to evaluate SAOP and AMCP using automated negotiation agents from the Automated Negotiation Agent Competition (ANAC). We show that against our expectations, the agents prefer to use the more complex AMCP protocol. Both protocols are usable in automated, human, and hybrid negotiations. The design space enables more informed design decisions when creating protocols and lays the foundation for further studies in finding desirable protocol properties that in turn improve the quality of the negotiation outcome.

Contents

List of Figures

List of Tables

Chapter 1

Introduction & Related work

Negotiating is an important aspect of professional and personal life [32]. Sometimes we conduct explicit negotiations; when buying a car or in case of a career interview for example. We resolve most of daily life negotiations without even thinking of them as such, e.g. when going out for drinks or booking a holiday.

Negotiations is a multidisciplinary field of interest. It is actively being researched in artificial intelligence (AI) [19] as well as other disciplines such as economics [5, 30, 31], game theory [29,30], and psychology [35]. AI focusses on the automation of negotiations; using automated agents to negotiate on people's behalves such as in e-commerce systems [23,25]. This can improve the outcome and reduce the time needed for negotions [25]. It can also be used to improve the person's negotation skills [17].

Most studies focus on bilateral negotiations, which are negotiations between two parties. In bilateral negotations the main challenges are opponent modeling, bidding, and acceptance strategies [3, 10], whilst the protocol primarily used in such negotiations is the Alternating Offer Protocol (AOP) [24, 36]. In contrast, this thesis focuses on designing and evaluating multiparty multilateral turn-taking peer negotiation protocols, which are protocols used by three or more peers to negotiate on a joint agreement. This type of negotiation is a generalized form of bilateral negotiation. Evaluating such protocols becomes more complex: The protocol now has more dimensions and some concepts have more degrees of freedom than in bilateral negotiators; e.g. information exchange between some of the parties.

The terms of multiparty and multilateral are used interchangeably by the community, for this thesis we distinguish them as follows: If there are more than two participants engaging in the negotiation, it is considered a multiparty negotiation. Multiparty negotiations can have different types of engagement or laterality. The engagement can be a *one-to-one*, *one-to-many*, *many-to-one*, or *many-to-many* negotiation. We define *one-to-one* engagements as bilateral negotations. A negotation can be both one-on-one (or bilateral) as well as multiparty simultaneously. Such a multiparty bilateral negotiation is typically a marketplace setting, in this setting, there are multiple buyers and sellers which conduct bilateral negotiations among themselves, i.e. they only interact in pairs. Examples of a marketplace type negotiations can be found in [37, 38]. We define *one-to-many* negotiations as auction negotiations. In contrast to the many-to-many negotiations, in this setting not all negotiators have the same role. The same is true for *many-to-one* negotiations, which we define as mediated negotiations. In this case, multiple parties try to come to an agreement by interacting with some centralized entity. Lastly, we define *many-to-many* negotiations, or peer negotiations. In a peer negotiation setting all parties have similar roles in the negotiation and try to come to a *joint* agreement. An example of peer negotiations is a group of friend negotiating on their holiday. While all but the one-to-one case can be considered multilateral negotiations, in this rest of thesis we will refer to the many-to-many negotiation setting, or peer negotiation setting when talking about multilateral negotiation.

In this thesis we address two unresolved issues in the field of negotiation theory. Firstly, not many protocols support multilateral multiparty negotiations. Secondly we want to introduce a method for evaluating such protocols. We decide to tackle both in a single thesis because of the interaction between the two: By designing protocols with certain properties in mind, we gain a deeper understanding of the implicit trade-offs. We can then create an evaluation method that captures the properties we are most interested in.

In this thesis we address the issue of evaluating multiparty multilateral turn-taking peer negotiation protocols. We do so by formulating the design space for this subset of protocols, introducing an method of evaluation and finally using that method to evaluate two novel protocols. We first continue by exploring and discussing some related protocols to get a general sense on the work already done on this topic. See appendix 7 for the related work search strategy. After discussing the related protocols, we formulate the research questions to be answered in this thesis.

The protocols proposed for multiparty multilateral turn-taking negotiations in the multiagent community mostly use a mediator [2, 8, 13, 15, 16, 22]. In contrast, this thesis proposes protocols and an evaluation method for non-mediated multilateral negotiations. A non-mediated multilateral negotiation protocol, the monotonoic consession protocol was introduced by Endriss [9]. 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 knowledge of the other agent's preferences, expect for the egocentric concession step in which the agent is expected to make a bid that is worse for itself. This is not a property that we desire in our protocols.

A generalization of the alternating offers protocol; a sequential-offer protocol was used by

Zheng et al. in [40]. Although they show that agents using the protocol are incentivized to converge to a solution, which is typically a desired property of a protocol, they fail to explain why they make certain protocol design decisions. For example, they support only time-based deadlines, and have no walk-away option. For the reader, it is not clear why these decisions were made. Although this protocol could be used as a multiparty multilateral negotiation protocol, we use a slightly different approach in this thesis. The core of the work of Zhang et al., is a negotiation strategy that applies a sequential projection method for multilateral negotiations. In that sense it cannot be compared to the research presented in this thesis in which multilateral negotiation protocols are proposed and evaluated.

De Jonge and Sierra recently introduced a new multilateral protocol inspired by human negotiations, called the Unstructured Communication Protocol (UCP) [7]. Unlike the negotiation protocols discussed above, this protocol does not structure the negotiation process. i.e., 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. Compared to the protocols proposed in this thesis, their protocol is more flexible. For example, in UCP an agent can remain silent and wait for the other agents. However, this flexibility comes at 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, negotiating in an agent environment is different: the agents lack information that humans that are physically present in the same negotiation room would normally have, such as body language, tone of voice, and 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. In this thesis, on the other hand, we focus on protocol evaluation for automated negotiations not human negotiators.

As mentioned earlier, this thesis focuses on designing and evaluating multiparty multilateral turn-taking peer negotiation. We introduce two novel protocols based on the Alternating Offer Protocol for bilateral negotiation (AOP). We introduce the extended protocols keeping the fairness properties of AOP. We define three key fairness properties that we will focus on while extending AOP to the multilateral domain:

1) Each party can propose their own offers.

2) Each party gets the same information about other offers.

3) Each party can reject an offer.

To address the issues raised in discussing the related work we answer the following research question:

Research question. In what ways can the bilateral turn-taking negotiation protocol be extended to a multiparty multilateral turn-taking peer negotiation protocol such that its fairness properties are maintained?

Sub-question 1. What is the formal design space for multilateral multiparty turn-taking peer negation protocols?

Sub-question 2. Can we design multilateral multiparty turn-taking peer negotiation protocols that maintain the fairness properties of the bilateral turn-taking negotiation protocol?

Sub-question 3. How do we evaluate the efficiency and social welfare of multilateral multiparty turn-taking peer negation protocols?

The main goal of this thesis is to find protocols that adhere to the fairness properties defined earlier. The rest of this thesis has the following structure: Chapter 2 contains some background information on terminology which is commonly used throughout this thesis, with a focus on some optimal negotiation solutions. We introduce a formal design space for multiparty multilateral turn-taking peer negotiation protocols in Chapter 3 and using this design space, in Chapter 4 we propose two novel protocols: SAOP and AMCP. We evaluate the two protocols on three social welfare metrics using automated negotiation agents in Chapter 5. Finally we draw a conclusion in Chapter 6 and discuss some miscellaneous findings and future work in Chapter 7. Based on this work, we published both the SAOP and AMCP protocols at the AAMAS conference of 2015 as part of the ANAC2015 competition [1].

Chapter 2

Background

In this chapter we explain some terminology that is commonly used in this thesis and in the field of automated negotiation. This chapters is split into two parts: The former part consists of computer science concepts such as agents and algorithms, while the later part describes mathematical concepts of negotiations such as Pareto optimality and Nash equilibrium. If these concepts are not foreign to you as a reader, you can skip the respective sections without missing out on information that is specific to this thesis.

2.1 Negotiation agents, domains and protocols

Autonomous agents are commonly used constructs in the field of computer science and especially in artificial intelligence [12]. An agent is an entity that perceives and possibly manipulates the environment in which it exists. Its manifestations range from actual robots with preceptors and actuators to software agents. Even human beings can be considered agents. A negotiation agent is an agent that can perceive a negotiation and can participate in the negotiation in some way. Two types of agents are involved in this thesis: software programs and human beings. Although both humans and software programs can be considered negotiation agents, we only refer to the software programs as *agents* and call to human agents just *humans* in order to distinguish more precisely. Whenever we refer to both humans and agents, we use the term *actor* or *party*.

A **negotiation domain** represents the issues of a negotiation. When there is a finite amount of issues with a discrete and finite amount of possible values, the negotiation domain can be mathematically described as a finite collection of all possible answers. These constraints are hard to keep up for even the simplest negotiations however: When negotiating about monetary values, a non-discrete issue already occuers as it may range from zero to infinity. Strictly speaking this is not true of course: The range of monetary values can be – and for most negotiations will be – discretized to cents or even to larger values when dealing with large monetary values. Domain modeling is a subject on its own, and out of the scope of this thesis. We use a discrete, finite issue, finite discrete values domain for our experiments. This is explained in more depth in the relevant chapters. Practically, it means that the domain used in this thesis has a finite set of possibilities and *can* be described as a finite collection of possibilities.

A **negotiation protocol** is a set of rules that describes the rules of engagement between agents on domains. A protocol can be thought of as a recipe or algorithm which describes the transition function between negotiation states. This can be – As is the focus of this thesis – a formal protocol, or an informal unwritten form of interaction. We give a formal framework with definitions of negotiation protocols in Chapter 3.

2.2 Pareto optimality, individual and socially optimal bargaining solutions

This thesis' experiments make heavy use of metrics based on Pareto optimal offers. A Pareto optimal offer is an offer that lays on the Pareto frontier $\mathbf{P}$, which consists of all offers that are not strictly dominated by any other offer. An offer is strictly dominated by another offer if that other offer is better for at least one party, while simultaniously not being worse for any other party involved. The Pareto frontier can easily be found by process of elimination: Consider all possible offers and for each offer check if it is strictly dominated by any other offer. If that is the case, remove it. What remains is the Pareto frontier. Finding such a frontier is a NP-Hard problem [26].

Negotiating or bargaining is considered a non-cooperative game [36]; there is something to gain for each party involved, but parties are uncertain of feasible solution vectors. Because of this, a negotiation might not come to an agreement. Such a situation could also occur if the party's Best Alternative to Negotiated Agreement (BATNA) [11] is higher than the offer currently on the table. This BATNA is the worst utility a party can get, for most negotiations this is simply zero utility for all parties involved.

The Nash equilibrium [28] is a competitive solution and thus requires no cooperation. For each party, given there is no cooperation by the other parties, what is the best utility one can get? This equilibrium is usually illustrated by the prisoners dilemma . In this case the Nash equilibrium strategy is the *minimax* strategy: Choose the maximum utility for yourself given that the opponent will choose the minimum for you.

Example: Prisoner's dilemma

Looking at the prisoner's dilemma as described in Table 2.1 from one of the prisoner's perspective, you have to choose between *betray* or *keep silent*. When choosing *keep silent*, the opponent may choose to either *betray* or also *keep silent* with respective utilities 0 and -1 for them. Using the *minmax* strategy you can assume that the opponent will choose the former which means you end up with a utility of -4. Going back and choosing the *betray* action yourself, the opponent gets to choose between *betray* with utility -2 or *keep silent* with utility -4. The opponent will choose the former in this case, giving you a utility of -2. This means that while there are 4 possible outcomes, only one is feasible in a competitive game: The opponent will always *betray* in a competitive game. In order to maximize your utility, this means you have to *betray* as well, resulting in utilities of $(-2, -2)$.

	Betray	Keep silent
Betray	$(-2, -2)$	$(0, -4)$
Keep silent	$(-4, 0)$	$(-1, -1)$

Table 2.1. The prisoners dilemma; A crime has been committed in a prison. Two prisoners are offered a choice to betray one other. They have no way of communicating. If they both remain silent, each prisoner has to serve 1 year of extra jail time. If both betray one other, they both get 2 years of extra time. If only one betrays the other, but the other remains silent, that other gets 3 years of extra time. According to the Nash equilibrium both prisoners should betray one another.

BATNA is also known as the *outside option*, the *point of disagreement*, the *threat point*, the *competitive solution* or the *reservation value* in other work. This concept is known by so many names for a good reason: The starting position has a huge impact on the negotiation. We will see its influence on solutions that we will discuss next. We consider the Nash bargaining Solution (NS), the Kalai-Smorodinsky bargaining Solution (KSS) and the Egalitarian social welfare bargaining Solution (ES), all of which are Pareto optimal solutions. The associated points of agreement are called the Nash point(s), the Kalai-Smorodinsky point, and the egalitarian point respectively. We will now discuss each of the three solutions in detail.

The **Nash bargaining solution** is a bargaining solution proposed by John Nash that maximizes the product of utility gain. Given an arbitrary amount of parties, their utilities at the agreement point $\mathbf{u}$ and their utilities at the disagreement point $\mathbf{u}_d$, the Nash solution tries to maximize the function $f(\mathbf{u}) = \prod(\mathbf{u} - \mathbf{u}_d)$. Note that when only two negotiation parties are involved, this is the same as maximizing the area under the rectangle bounded by the agreement point and the disagreement point, as can been seen in Figure 2.1. Depending on the shape of the Pareto frontier between 1 and all Pareto points can be Nash bargaining solutions and thus multiple Nash point, which are points adhering to the Nash bargaining solution, can exists. In order to find the Nash point(s) consider each point on the Pareto frontier, subtract the utilities that each party would get when not coming to an agreement. This result is called the utility gain. Multiply the utility gains of each party for the specific other and find the maximum of these multiplications. This multiplication is referred to as the Nash product. Please note that the Nash bargaining solution is a cooperative solution to multi-issue negotiation, and should not be confused with the earlier explained Nash equilibrium, which is a different concept describing a competitive strategy and not a Pareto optimal cooperative agreement.

The **Kalai-Smorodinsky bargaining solution**, also known as the utilitarian social welfare

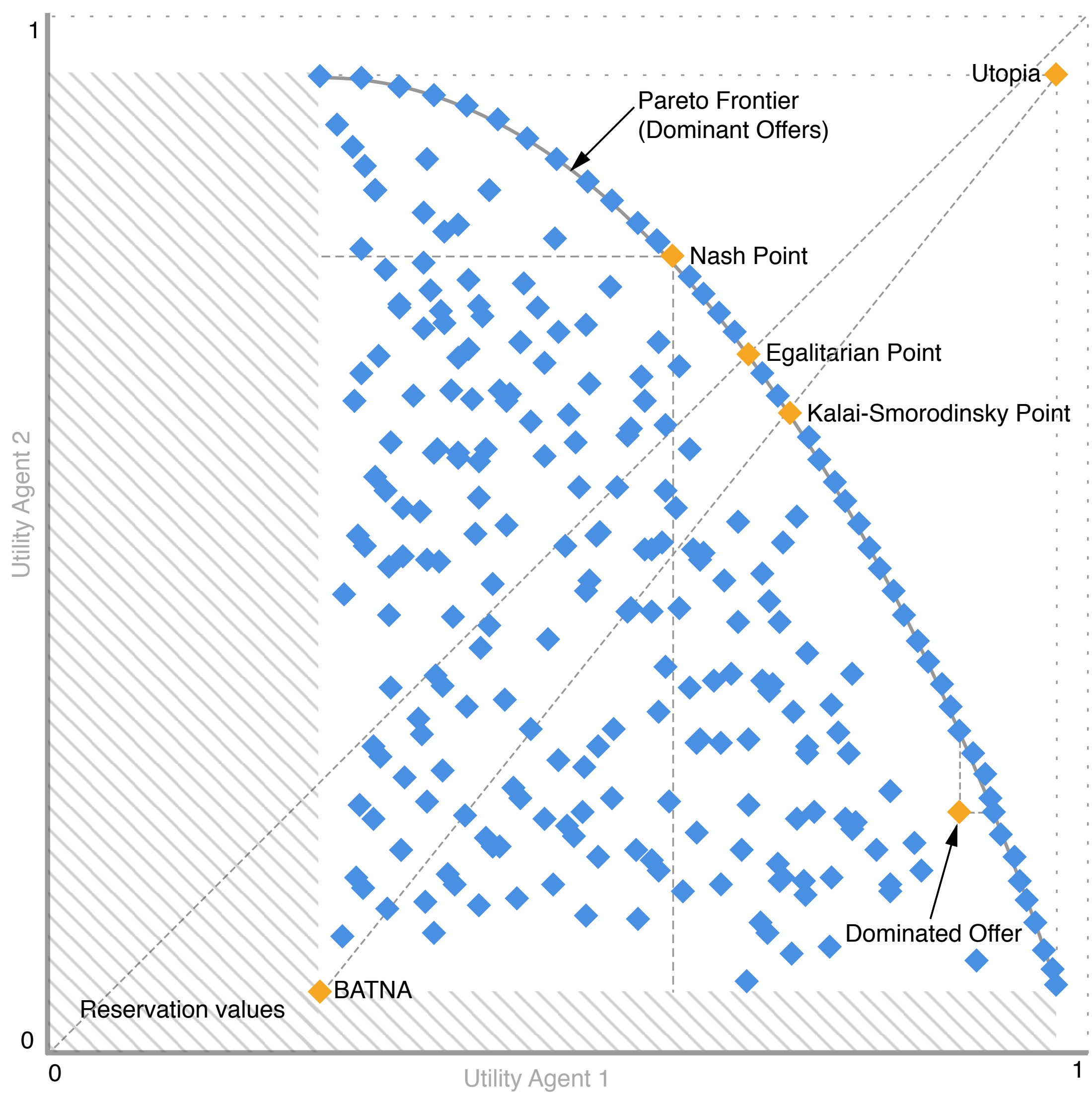

Figure 2.1. In this graphical overview the utilities of two agents are plotted. We see the offers corresponding to the Nash solution, Kalai-Smorodinsky solution and egalitarian solution.

bargaining solution, tries to maintain the ratio of maximum gains [21]. Intuitively, it tries to find the maximum sum of utility gains. For two players, this point can be found by calculating the ratio of maximum utility gain each party can get and comparing this to the ratio of each Pareto efficient bid. Suppose party 1 has a maximum utility gain of m_1 and party 2 that of m_2, then the ratio of equal gain is $r_{eq} = \frac{m_1}{m_2}$. For each point on the pareto frontier $p \in \mathbf{P}$ then calculate the ratio $r_p = \frac{p_1}{p_2}$ and compare to take the point that has the ratio which is the closest to r_{eq}. This point is the Kalai-Smorodinsky point. The philosophy behind the Kalai-Smorodinsky bargaining solution, is that of fair division [27], which state that when there is extra utility to be divided, no one should be worse off. This solution does that while maintaining bargaining positions; when a party has a better starting position, the Kalai-Smorodinsky bargaining solution tends to favor that party.

The **egalitarian social welfare solution** or the Rawls point (after the philosical theory of Rawls [33]) also known as the Kalai point (not to be confused with the Kalai-Smorodinsky point), is a lesser known bargaining solution. It was proposed by Kalai two year after the Kalai-Smorodinsky bargaining solution [20]. Intuitively it tries to maximize the utility of the party with the lowest utility. The egalitarian bargaining solution can be found by choosing the point on the Pareto frontier with the highest minimum utility. The philosophy behind this entails that in a cooperative environment, the utility of the individual that is the worst off, is the utility of a group. The group should therefore try to maximize its utility. The advantage of this bargaining solution, is that optimizing the minimum utility scales well into higher dimensions.

Each of the bargaining solutions previously discussed are based on a set of axioms. In this background chapter, we highlight the differences between the solutions. The Nash bargaining solution does not adhere to resource monotonicity [39]. **Resource monotonicity** ensures that when there is a utility gain compared to the base situation, no party should be worse off. With the Nash solution, a larger product could be found were some parties might strictly be worse off. The Kalai-Smorodinsky bargaining solution adheres to resource monotonicty, but drops the Independence of Irrelevant Alternatives (IIA) axiom that the Nash bargaining solution adheres to [21]. **Independence of Irrelevant Alternatives** states that when a new solution is added to the domain, and this solution is not the agreement, it is irrelevant and should not influence the negotiation outcome. By substituting IIA for resource monotonicity, the Kalai-Smorodinsky solution ensures fair division. Finally the egalitarian social welfare solution adheres to both the resource monoticity axiom as well as IIA, but drops the scale invariance axiom that both the Nash solution and the Kalai-Smorodinsky solution adhere to [20]. **Scale invariance** states that if party p_1 can get a utility gain of $c \cdot u_1$ and player p_2 can get a utility gain of $c \cdot u_2$, the solution should remain the same for any positive real number c. It should be pointed out that in this thesis, we normalized all utilities to be between 0 and 1, and any computation consequently do not suffer from any scale invariance issues.

Chapter 3

Defining the Design Space

While we have always been exercising negotiations, and there are multiple formal protocols for doing so, most of these protocols arose naturally. In contrast, in this chapter we introduce a formal design space for the subset of multiparty multilateral turn-taking peer negotiation protocols, answering Sub-question 1 of the research questions. As phrased in our main research question, we aim at extending the bilateral AOP protocol pertaining it's fairness properties. The design space defined in this chapter helps us formalize the design decisions that we have to make for those extended protocols. We use the design space to aid in the creation of the novel protocols introduced in Chapter 4. Looking at a negotiation from an optimization theoretical point of view, the goal is to optimize the negotiation outcome, of which – as we have already seen in Chapter 2 – there can be many. From this point of view, the negotiation is a Multi-Objective Optimization (MOP) optimization problem. The protocol in conjunction with the negotiating agents can be considered the algorithm used to solve this optimization problem, while the domain is the problem we are trying to solve. The protocol dictates the actions an agent can perform and perceptions an agent receives. Hence, defining a protocol that has desirable properties is quintessential for a optimal negotiation solution.

3.1 Protocol Properties

While exploring the the design space described as a formal definition in Section 3.3 and its related protocols in Chapter 4, we found properties of a negotiation protocol. These properties are a product of a protocol design and inherit to the protocol. One way of obtaining favorable properties for a protocol is by iterating over its design decisions and evaluating the designed protocol. In this section we will explain the properties *convergence*, *fairness*, *welfare bias*, *speed*, *complexity* and *environment* in more depth.

One of the properties that is sought after for a protocol is *convergence*. Does the protocol promote converging to a solution? This promoting of convergence can be strictly enforced by not allowing repeated offers and not allowing infinite domains for example, or it can more subtly nudge the parties towards convergence by enforcing a deadline or by discounting parties' utilities as the negotiation continues.

Two related properties are *fairness* and *bias*. Does the protocol favor a certain party or strategy? Typically, we prefer an unbiased protocol, but it may come at a cost of protocol complexity or speed, so this is a trade-off situation. In addition to the protocol being convergent, it is also interesting to see whether the convergence is fair. This comes back to the fairness properties as discussed in Chapter 1: Is each party allowed to place its own offers? Does each party get the same information? is each party free to reject an offer?

Not to be confused to the fairness property is the *welfare* or *social welfare* property. Does the protocol promote convergence towards a specific optimal solution point such as the Nash point, Kalai-Smorodinsky point or egalitarian welfare solution point? These solutions all have a different view on what welfare is and are discussed in more depth in Chapter 2.

The properties *speed* and *complexity* are usual candidates for trade-offs. When a protocol is complex, it will be harder for agent designers to create agents for it. For human negotiators, it becomes harder to create a mental model of the negotiation, possibly negatively impacting the other properties. Speed is related to this as the protocol becomes more complex, it can involve more steps in a negotiation, which in turn might impact the speed of the algorithm. Here we also see the trade-off more explicitly: When we design a protocol that converges quickly but is complex and slow, the actors take less actions due to this slowness and complexity. It could ultimately converge less fast than a simpler, less strict protocol.

Finally we discuss the property *negotiation environment*. Does the protocol support automated negotiations, human to human negotiations? It could also support both or even intermix between human and automated negotiators. This property is related to complexity. Theoretically speaking, humans could execute a computer algorithm and computers could try to emulate a human negotiator. In practice, a protocol can be complex for just humans or automated agents, and not the other. Note that we included the negotiation environment in properties and not design decisions, as it is not an explicit decision that can be made when designing a protocol, but rather a property that is dependent on choices made when designing a protocol. An overview of the properties just discussed can be found in Table 3.1. We will now formulate the design dimensions of the protocols that we want to evaluate.

Properties	Modes
Convergence	Enforced, promoted, no convergence, etc.
Fairness	Each party proposes own offers, gets same information, can reject offers
Bias	Favors a party or strategy
Welfare	Promotes Nash point, Kalai-Smorodisnky point, egalitarian welfare point
Speed	Number of interactions
Complexity	Number of steps in the protocol
Environment	Only automated agents, human agents, both, or mixed?

Table 3.1. A list of negotiation properties we found that can impact the negotiation.

3.2 Design Decisions

We aim at providing a sound reasoning for the chosen protocols and weight to pros and cons. In order to do that, we provide a formal definition to which multiparty multilateral turn-taking protocols must adhere. By using such a formal definition, we can compare and reason about certain protocol design decisions across multiple protocols in a qualitative manner. For this definition, we need to define the dimensions of negotiation protocols, so that we can capture them in a formal definition of what a negotiation protocol entails. Each of these dimensions can be considered a design decision: With a few exceptions, most of these decisions are discrete – even binary – in nature. The set of all design decisions defines the protocol. The definitions as proposed in Section 3.3, only support a subset of all possible negotiation protocols. Dimensions that we found are out of scope for multiparty multilateral turn-taking peer negotiation protocols, were excluded. In order to accomplish that, some of the design decisions had to be fixed, meaning we do not support differentiating by that dimension. We now highlight the dimensions, and capture every design decisions that we made for the framework.

The first two dimensions that we discuss are *laterality* and *number of participants*. As mentioned in the introduction, Chapter 1, these two dimensions are different. The protocol can support different types of laterality. Most commonly, only bilateral and multilateral are distinguished, meaning that respectively only two and multiple participants can interact at a time. For the number of participants supported by a protocol, a similar distinction is typically made. Negotiation protocols that only support two participants are called biparty negotiation protocols, while protocols supporting more than two participants are called multilateral negotiation protocols.

Another dimension to consider is concurrency. Some protocols allow for agents to negotiate in parallel. In such protocols you have negotiation pools consisting of two or more agents negotiating concurrently. If the protocol allows it, agents can even be in multiple pools at the same time. We restrict ourselves to turn-taking protocols. These protocols impose rules on which agent can react and when it can react. We impose this restriction so that it becomes more workable to reason about the process of negotiation and because it will become more feasible to evaluate the outcome.

The negotiating parties in a protocol can be on equal footing, such as peer negotiations or asymmetric such as buyer and seller negotiations. For this framework we choose to restrict ourselves to equal footed, or symmetric negotiations among peers. This decision makes it more feasible to evaluate the protocols, because it allows us to use the agents designed for the ANAC competition [4] – which are peer negotiation agents – as evaluating agents. More on the evaluation and the used agents can be found in Chapter 5. Peer negotiations are symmetric – every agent has the same role – which also makes it more feasible to reason about.

When observing negotiations among human negotiators, we see that they tend to use partial offers, i.e. they do not offer a solution to all of the issues within a domain, but perform an issue-by-issue negotiation, leaving the remaining issues open for discussion in subsequent offers [34]. Othepacr type of offers include issues with price ranges for monetary issues or offers including some issues that might be indifferent to the negotiating party. For this thesis, we excluded such fuzzy offers and concentrated on full offers or *package deals* in which a party is only allowed to place an offer that includes all the issues of the negotiation domain. Each party is forced to offer a valid value for each issue of the offer, ranges or indifference is thus disallowed. Note that agents can simulate fuzzyness by proposing random values for the same issue on each offer. By excluding these fuzzy offers, reasoning about the protocols produced by the framework becomes less complex. It also becomes harder to find good agents to use in evaluation when imposing fuzzy offers as one of an agent's capabilities.

Information sharing is a more implicit dimension. This framework does not make any assumptions about the amount of information shared. Some protocols give complete information; such protocols give all parties all information – even including each opponent's preference profile – while other protocols do not allow this.In case of protocols containing partial information, agents will encounter a more complex task performing opponent modeling. Protocols that use partial information should define how information sharing takes place. These protocols could have communication actions available, however that does not necessarily have to be the case, as agents could also be restricted to communicating implicitly via the offers that placed. Examples will be provided by the protocols presented in Chapter 4.

We come up with a framework consisting of two parts. Firstly, the dimensions just dis-

Dimension	Modes	Free/Fixed
Number of participants	biparty, multiparty	free*
Laterality	bilateral, multilateral	free*
Concurrency	**turn-taking**, concurrent	fixed
Roles	**symmetric** (peers), specialized (buyers/sellers)	fixed
Information sharing	complete/perfect/partial information	free
Interactions	offers, walk-away, communicate, etc.	free
Bids	**full offers**, partial, indifferent, ranges, etc.	fixed
Continuation	continue after initial agreement?	free

Table 3.2. A list of negotiation dimensions we found that need to be taken into account when creating a negotiation protocol. In the formal definition given in this chapter, we fixed the dimensions *Concurrency*, *Roles* and *Bids* to *turn-taking*, *symmetric* and *full offers* respectively.
* While we fixed the design to multiparty and multilateral, the biparty and bilateral cases can be considered special versions with $N = 2$ and are therefore included in the multiparty and multilateral cases.

cussed and summarized in Table 3.2. These can be considered a non-exhaustive list of design decisions that have to be made when designing a protocol. Secondly based on these dimensions, we design a formal definition for the subset of turn-taking multiparty multilateral peer negotiation protocols.

3.3 Formal Definitions

In this section we describe a protocol mathematically. This section is based on work that we published in [1], but the notations and concepts are slightly different. The main change in notation is that all functions and predicates are now denoted as function mappings, i.e. they have the form $\mathcal{A}^n \rightarrow \mathcal{B}^m$ where $n \geq 1$, $m \geq 1$ which represent the function mapping from domain $\mathcal{A}$ of dimension n to domain $\mathcal{B}$ of dimension m. For predicates specifically this function mapping reduces to $\mathcal{A}^n \rightarrow \mathbb{B}$, i.e the mapping from some domain $\mathcal{A}$ of dimension n to a 1-dimensional boolean value. Conceptually we made some changes with respect to the published work as well: while Definition 9 of [1] includes agents in the protocol description, corresponding Definition 3.10 of this thesis does not. We want a clear separation of concerns when it comes to protocols and agents: Protocols dictate how agents interact with a domain. In this way we can focus on protocols design without considering the agents. Later, in Chapter 5 where we evaluate the protocol, we consider a complete negotiation system consisting of agents, protocols and negotiation domain, but do not dictate which agents are used. Another key difference with the published work, is that the function $agrBids$ now maps to a set of agreed upon offers. While the protocols that are evaluated in this thesis always come to a single agreement, it is possible to create protocols that give multiple feasible agreements. This section continues with some common notations and concept, followed by formal definitions and explanations about that definitions. These definitions eventually lead to Definition 3.10, which is the protocol description itself. They also serve as a basis on which the protocol descriptions can be build, as is shown in Chapter 4.

We now explain some common concepts and constructs used in the definitions that follow: Let $\mathbf{Agents}$ denote the set of agents and let $\mathbf{Bids}$ denote the set of possible bids in the negotiation domain. Each agent can perform actions that are in the set $\mathbf{Actions} \subseteq \mathbf{Bids} \cup \{accept, reject, end\}$; either place a bid, $accept$ a bid, $reject$ a bid, or end the negotiation by walking away. In this section we frequently use sets, tuples and sequences. For any such a construct $\mathbf{T}$, let $\mathbf{T}_i$ and $\mathbf{T}_j$ denote respectively the i^{th} and the j^{th} element. and let $|\mathbf{T}|$ denote the length.

Rounds are used to structure the negotiation process. The structure of rounds may differ per protocol. Some protocols allow for different actions in different rounds, while other are homogeneous. We define the construct $Round$ as follows:

Definition 3.1. Rounds

- Let $\mathbf{Round}$ be the set $\{1, 2, ..., R\}$ where R denotes the total number of rounds.

In this definition we define turn sequence $\mathbf{TurnSeq}$ as well as two functions: $getSeq$ and $prev$. The $\mathbf{TurnSeq}$ defines the order in which the agent can take an action and is bound to a specific round. The function $getSeq$ gives a $\mathbf{TurnSeq}$ given a specific round. Lastly $prev$ resolves to the round prior to the given round. This is a convenience method which we use in later definitions.

Definition 3.2. Turn taking

- The protocol assigns turns to a subset of negotiating agents. This assignment is defined in a turn taking sequence $\mathbf{TurnSeq} \subseteq \mathbf{Agents}$. Let $\mathbf{TurnSeq}$ be a specific ordering of the $\mathbf{Agents}$ set such that:
  - Each agent in $\mathbf{TurnSeq}$ is also in $\mathbf{Agents}$:
    $\forall a_1 \exists a_2 (a_1 \in \mathbf{TurnSeq}, a_2 \in \mathbf{Agents} \rightarrow a_1 = a_2)$
  - Each agent is in the sequence is unique:
    $\forall i \forall j (i \neq j \rightarrow \mathbf{TurnSeq}_i \neq \mathbf{TurnSeq}_j)$
- The function $getSeq : \mathbf{Round} \rightarrow \mathbf{TurnSeq}$ assigns a turn-taking sequence per round. Its specification depends on the protocol.
- The function $prev : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbf{Round} \times \mathbb{N}^+$ defines the previous turn in the negoti-

ation, that can be in this round or a previous round, specified by:

$$prev(r,t) = \begin{cases} \langle r, t-1\rangle, & 1 < t \leq |\mathbf{Agents}| \\ \langle r-1, |\mathbf{Agents}|\rangle, & t = 1 \wedge r > 1 \\ \text{undefined}, & \text{otherwise} \end{cases}$$

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

$$\forall x \in \textbf{Round} \times \mathbb{N}^+ :$$
$$prev^0(x) = x$$
$$prev^{n+1}(x) = prev^n(prev(x))$$

These conditions promote fairness of 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 SAOP protocol, which we discuss in Chapter 4. However, Definition 3.2 allows more freedom which we will need for the AMCP protocol, discussed in the same chapter as SAOP.

Although the actions might differ over protocols, we introduce notions that are general to all negotiation protocols. The functions $action$ and $allowedAction$ specify what actions agents take and what actions they are allowed to take.

Definition 3.3. Allowed actions

- The function $allowedActions : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathcal{P}(\mathbf{Actions})$ determines the allowed actions per turn t at a given round r. The function specification varies over protocols.

Definition 3.4. Actions

- The function $action : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbf{Actions}$ denotes what action was taken given turn t and round r.

Although protocols do not dictate which actions agents take during the negotiation, the function $action$ is defined here, as the type of action taken by the agents do have an effect on the procedure as specified in Definitions 3.6, and 3.7.

Definition 3.5. Deadline

- The predicate $deadline : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ 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.

Two commonly used specifications for deadlines are round-based ($deadline_{round} \leftrightarrow R > R_{deadline}$) and time-based ($deadline_{time} \leftrightarrow T > T_{deadline}$) deadlines. For time-based deadlines, the time T needs to be kept track of. This can be done in a low-tech fashion such as with a stopwatch, or – especially for automated negotiation – using a computer's clock. At the ANAC 2015 competition for example, the time was taken from the system time of the computer running the tournament, and $T_{deadline}$ was set to 3 minutes. A round-based deadline needs to keep track of the number rounds R, which this framework already provides in the form of the $\mathbf{Round}$ set.

Definition 3.6. Agent ending the negotiation.
The predicate $endNego : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ denotes whether or not an agent has ended the negotiation. Its value is determined at the end of the current turn.

$$\forall r \forall t (r \in \mathbf{Round}, t \in \mathbb{N}^+, endNego(r,t) \leftrightarrow action(getSeq(r)_t, r) = end)$$

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. This is why Definition 3.8 determines whether a negotiation continues. We will first define what an agreement is in definition 3.7.

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.

Definition 3.7. Agreement.

- The function $agrBids : \mathbf{Bids} \times \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathcal{P}(\mathbf{Bids})$ denotes the bids that were agreed upon given all proposed offers and the current round and turn. The specification varies over protocols.
- The predicate $agr : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ denotes whether or not an agreement is reached. This predicate is only included for convenience and is based on the $agrBids$ function. It's exact specification is as follows: $\forall r \forall t (agr(r,t) \leftrightarrow agrBids(\mathbf{Bids}, r, t) \neq \emptyset)$.

While for most protocols, including the two that we define in Chapter 4, $agrBids$ resolves to either the empty set $\emptyset$ or a single element set $\{b\}$ where b is the agreed upon offer. There are protocols that end with multiple agreements. In such protocols the $agrBids$ function will resolve to a set of agreements. Note that for some protocols, this function needs access to the complete bid history, which can be accessed using $prev$ function calls.

Definition 3.8. Continuation.

- The predicate $cont : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ denotes whether the negotiation continues after the current turn. Its value is determined at the end of the current turn.

- The default implementation of the predicate is defined by following relation:

$$\forall r \forall t(r \in \mathbf{Round}, t \in \mathbb{N}^{+} : \textit{cont}(r,t) \leftrightarrow \neg deadline(r,t) \wedge \neg \textit{endNego}(r,t) \wedge \neg agr(r,t))$$

Protocols might divert from this default implementation. For example a negotiation might not end on an agreement on when an agent walks away. Another example is a protocol that we left out of this thesis called Alternating Multiple Offers Majority Protocol (AMMP). This protocol will reach an agreement when the majority agrees, but will continue until either consensus or a deadline has been reached.

Definition 3.9. Outcome of the negotiation.
The function $outcome : \mathbf{Round} \times \mathbb{N}^{+} \rightarrow \mathcal{P}(\mathbf{Bids}) \cup \{fail\}$ determines the negotiation outcome at the end of the current turn.

$$\textit{outcome}(r,t) = \begin{cases} \textit{undefined}, & \textit{cont}(r,t) \\ fail, & \neg\textit{cont}(r,t) \wedge \neg\textit{agr}(r,t) \\ agrBid(\mathbf{Bids}, r, t) & \neg\textit{cont}(r,t) \wedge \textit{agr}(r,t) \end{cases}$$

Definition 3.10. Turn-taking Negotiation protocol.

- A multiparty multilateral turn-taking peer negotiation protocol P is a tuple $\langle turnSeq, allowedActions, deadline, endNego, agrBid, cont \rangle$.

i.e. the identity of such a protocol P is defined exclusively by those definitions. A negotiation exists of such a protocol, agents and a domain. The domain is used to look up valid values for $\mathbf{Bids}$ and the agents work in conjunction with the protocol: The protocol dictates, using $turnSeq$, which agent gets to act. The agent will have to choose an action provided by the protocol's $allowedActions$ function. The protocol will be aware of each agent's response and uses $deadline$, $endNego$, $agrBid$ and $cont$ to determine the current state of the negotiation.

The above definitions form the core of the subset of multiparty multilateral turn-taking peer negotiation protocols. Now that we have a definition of such protocols, we look at ways to extend the well-known alternating offers protocol for bilateral negotiations to this multiparty multilateral case. There are different ways to do so. The next sections introduce two ways of extending this protocol: Stacked Alternating Offers Protocol (Section 4.1) and Alternating Multiple Offers Consensus Protocol (Section 4.2). Both protocols are specified by providing the detailed descriptions of those predicates and functions that are protocol dependent.

Chapter 4

Designing Protocols

In this chapter we put the formal definitions and protocol dimensions as proposed in Chapter 3 to use. Our aim is to create state-of-the-art protocols for multiparty multilateral turn-taking peer negotiations that adhere to the fairness properties as introduced in Chapter 1. By creating such protocols we answer Sub-question 2 of the proposed research question, bringing us a step closer to answer the main research question. To this end we introduce two novel protocols: Stacked Alternating Offer Protocol (SAOP) and Alternating Multi-offer Concencus Protocol (AMCP), which we published at the AAMAS conference [1]. Both are based on the well known alternating offer protocol for bilateral negotiations [36] and adapted for multilateral negotiation. The former protocol is a straightforward adaptation of the bilateral protocol, while the latter uses a voting system. In contrast to the related work discussed in Chapter 1, we focus on the following key protocol properties while designing our protocols:

1) The protocol should be fair.
    a) Each party can propose their own offers.
    b) Each party gets the same information about other offers.
    c) Each party can reject an offer.
2) It should be easy to design agents for the proposed protocol.

It is worth nothing that most mediated protocols discussed in Chapter 1 adhere to the points 1b and 1c, but by design they can not adhere to point 1a. Both SAOP and AMCP are mediator-less peer negotiation protocols and can therefore adhere to 1a.

It stands to reason that protocols should be fair, so all parties involved are comfortable using it. The implications of a protocol that is not universally accepted are huge, because if one or more parties refuse to negotiate using the protocol, a new protocol has to be negotiated, which – without a formal protocol – can become quite problematic. The protocol should also minimize the amount of interaction needed as we evaluate them on time-constrained domains. When negotiating using a protocol with a low rate of convergence on these type of domains, less of the space is explored during the negotiation, which we expect results in outcomes with lower utility. We will see how big this impact is in the experiments in Chapter 5, as we evaluate the two protocols which have different rates of convergence. We expect that when a protocol becomes more involved, there is less room for creating good performing agents, hence we want our protocols to be easy to understand and easy to design automated agents for. Note that the properties that we proposed can conflict and have to be balanced with each other when conflicts arise. Such trade-offs are discussed at the end of each proposed protocol. It should be noted that AMCP is also abbreviated as AMOP in related work [1], we choose to lay emphasis on the consensus part, thus the C in AMCP.

4.1 SAOP

The Stacked Alternating Offer Protocol (SAOP) is a generalization of the alternating offer protocol used in bilateral negotiation to multilateral multiparty negotiation. As it is a generalization, some extra design decisions must be made. When dealing with a bilateral negotiation, "alternating" is rather self-explanatory: The rounds alternate between the two parties. In SAOP, rounds are allotted in a round-robin fashion, i.e. the protocol chooses a random starting point and allots rounds clock-wise from there. Each party is allowed one of the following actions: $bid \in \mathbf{Bid}$, $accept$ or end (walk away), with the exception of the first party in the first round, which cannot use the $accept$ action. When using an element of the $\mathbf{Bid}$ action set, the party places a specific counter-offer and automatically rejects the previous offer. If the party decides to walk away (i.e. uses the end action), the negotiation ends with no agreement. Such a situation could occur if the party's BATNA is higher than the offer currently on the table. The party has the option to accept the offer on the table by using the $accept$ action. The negotiation ends when either of the following conditions holds:

1) All parties accept the offer currently on the table. The proposing agent is assumed to accept its own offer.

2) Any agent walks away by doing an end action.

When a party uses the bid action to place a counter-offer, all $accept$ actions are void and parties get to choose one of the three actions again. In this protocol all actions executed are considered common knowledge: Each party has knowledge of the these exact actions and also knows which party executed the action.

In SAOP the same turn taking sequence is used in all rounds.

Definition 4.1. Turn taking (Definition 3.2 for SAOP).
Let s denote that sequence, thus for SAOP the set of turn-taking sequences is $\mathbf{TurnSeq} = \{s\}$. Each agent takes exactly one turn in each round.

The rules for turn taking are those specified in Definition 3.2, i.e., each agent gets exactly one turn per round, as specified by s.

The general definitions dictate that an implementation of $allowedActions$ should be given. As mentioned earlier, there is an edge-case on the first turn of the first round: no accept action can be chosen, because there is no offer to accept that point. Apart from that case, all actions are allowed. The formal specification of $allowedActions$: $\mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathcal{P}(\mathbf{Actions})$ is as follows:

Definition 4.2. Allowed actions (Definition 3.3 for SAOP).

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

Predicate $deadline : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ 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.

Definition 4.3. Deadline (Definition 3.5 for SAOP).
This protocol uses $deadline_{time}$ as defined in Definition 3.5 for this protocol.

The predicate *agr* : $\mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ denotes whether or not an agreement is reached. The function $agrBids : \mathbf{Bids} \times \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathcal{P}(\mathbf{Bids})$ denotes the bid that was agreed on. Their values are determined at the end of turn. The predicate agr follows its default specification as defined in Definition 3.7. The function $agrBids$ is defined as follows:

Definition 4.4. Agreement (Definition 3.7 for SAOP).

$$agrBids(bids,r,t) = \begin{cases} \{action(prev^{|\mathbf{Agents}|}(r,t))\} & \text{if } 0 \leq i < |\mathbf{Agents}| - 1, \\ & \quad action(prev^i(r,t)) = accept \\ \emptyset & \text{otherwise} \end{cases}$$

Informally, we have an agreement iff $|\mathbf{Agent}| - 1$ turns previously, a party made an offer that was subsequently accepted by all the other parties. The agent that made the offer, in the SAOP protocol, is assumed to find its own bid acceptable. In $agrBids$ the offer that was made $|\mathbf{Agent}|-1$ turns ago, is set to be the agreement in the current round and turn.

> **Example: SOAP**
>
> Assume that there are three negotiating negotiation parties, a_1, a_2 and a_3. Party a_1 starts the negotiation with a bid b_1. Party a_2 can accept this bid, make a counter offer or walk way. Lets assume that party a_2 decides to make a counter bid b_2. Assume that parties 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 agreement is bid b_2.

This is a relatively straight-forward generalization of the alternating offer protocol for bilateral negotiation. It is therefore easy to adapt agents written for that protocol to SAOP. Also, SAOP can be considered a fast protocol in terms of rounds, as there is only a single pass per round, which reduces the interaction between parties. When making a counter-offer, a party implicitly starts a new round of negotiation. Also, by allowing every party to counter an offer, we promote a higher rate of convergence. Note that the actual rate of convergence is ultimately depended on the parties negotiating. By forcing the first agent to place an offer and by forcing each subsequent party to immediately reason about this offer, there is an inequality when it comes to information and initiative. The first party will have more initiative and can therefore steer the negotiation in a direction favorable for the initiator. The last party has more information and could use that to its advantage. These biases in information and initiative are generally unwanted, but were accepted as trade-offs for making a simple protocol. The next protocol we discuss is AMCP which tries to avoid this information bias.

4.2 AMCP

The alternating multi-offer consensus protocol (AMCP) works by a different mechanism. This protocol consists of two distinct phases: offering and voting. In the first round each party must do a bid action; *all* parties place an offer on the table. Then, for each offer, a round is started in which parties can vote to accept or reject that offer. Parties are allowed to accept multiple offers.

When there is consensus – each party accepts the same offer – that offer is considered an agreement and the negotiation ends. There is no walk away option, the negotiation either continues until there is consensus or a deadline is reached. In this protocol actions are also considered common knowledge. Offers are not anonymous; parties get an offer from each other party before the voting phase starts.

AMCP is a protocol with heterogeneous rounds: The protocol's rounds consist of repeating sequence of a single offering round, followed by $|\mathbf{Agents}|$ voting rounds. This is implemented by the $allowedActions$ functions, as we will see in Definition 4.6.

Rounds (Definition 3.1 for AMCP).
The concept of $\mathbf{Round}$ is not changed. i.e. $\mathbf{Round} = \{1, 2, ..., R\}$ where R denotes the total number of rounds still holds. We do however, assign special meaning to certain rounds. All rounds are voting rounds, with the exception of every $(|\mathbf{Agents}| + 1)^{\text{th}}$ round, which is an offer round, starting with round 1. The different allowed actions in these rounds are specified in Definition 4.6, but first we define the turn taking sequence.

Definition 4.5. Turn taking (Definition 3.2 for AMCP).
In AMCP the same turn taking sequence is used for all rounds. Let s denote that sequence, i.e., $TurnSeq = \{s\}$. Each agent takes exactly one action in each round.

Definition 4.6. Allowed actions (Definition 3.3 for AMCP).
We define the set of possible actions as $\mathbf{Actions} = \mathbf{Bid} \cup \{accept, reject\}$. The detailed specification of $allowedAction : \mathbf{Round} \times \mathbb{N}^+ \to \mathbf{Actions}$ is as follows:

$$allowedAct(r,t) = \begin{cases} \mathbf{Bid}, & \text{if } \textit{cont}(r,t) \wedge r \text{ mod } (|\mathbf{Agents}| + 1) = 1 \\ \{accept, reject\} & \text{if } \textit{cont}(r,t) \wedge r \text{ mod } (|\mathbf{Agents}| + 1) \neq 1 \\ \emptyset, & \text{otherwise.} \end{cases}$$

The negotiation starts with a bidding phase during which all parties does a bid action to give their offer in turn specified by the turn sequence. The bidding round is followed by a voting round for each bid on the table. This means that all parties first vote on the first bid that was put on the table in this round, then all vote for the second bid and so on. During each voting round, parties take their turn according to turn taking sequence as defined by the turn taking rules. During the voting rounds, parties can only use the $accept$ or $reject$ actions to respectively accept or reject offers on the table. When considering a single round of offering and the related voting rounds, in which round 1 is the offer round, the votes in round $1 + i$, refer to the i^{th} offer in the offer round. This is specified indirectly by Definition 3.7.

This protocol uses a time-based deadline, just like the SAOP protocol.

Definition 4.7. Deadline (Definition 3.5 for AMCP).
This protocol uses $deadline_{time}$ as defined in Definition 3.5 for this protocol.

The predicate $agr : \mathbf{Round} \times \mathbb{N}^+ \to \mathbb{B}$ denotes whether or not an agreement has been reached, it follows the default definition provided in Definition 3.7. The predicate $agrBids : \mathbf{Bids} \times \mathbf{Round} \times \mathbb{N}^+ \to \mathcal{P}(\mathbf{Bids})$ denotes the offer that was agreed on. Their values are determined at the end of turn in voting phases. It's specification is as follows:

Definition 4.8. Agreement (Definition 3.7 for AMCP).
Given some round r we define $r_{bid} = r - t_{bid}$ as the previous offer round and $t_{bid} = r \bmod (|\mathbf{Agents}| + 1) - 1$ as the turn of that round in which the offer was placed.

$$agrBids(bids, r, t) = \begin{cases} \{action(r_{bid}, t_{bid})\} & \text{if } t = |\mathbf{Agents}|, 0 \leq |\mathbf{Actions}| - 1, \\ & action(prev^i(r,t)) = accept \\ \emptyset & \text{otherwise} \end{cases}$$

In this formulas, let r and t denote the current round and turn, and r_{bid} and t_{bid} denote the corresponding bid round and turn. When an agreement is found in a specific round r and turn t, we search back to corresponding offer using r_{bid} and t_{bid}.

The predicate $cont : \mathbf{Round} \times \mathbb{N}^+ \rightarrow \mathbb{B}$ denotes whether the negotiation continues after the current turn. Its value is determined at the end of the current turn. In comparison to the general continuation definition, we removed the end predicate, because in this protocol it is not possible to walk away from the negotiation.

Definition 4.9. Continuation (Definition 3.8 for AMCP).

$$\forall r \forall t (r \in \mathbf{Round}, t \in \mathbb{N}^+, cont(r,t) \leftrightarrow \neg deadline(r,t) \wedge \neg agr(r,t))$$

Example: AMCP

Assume that there are three negotiating negotiation parties, a_1, a_2 and a_3. In the first round party a_1 starts the negotiation with a bid b_1. Party a_2, and a_3 place there own bids; b_2 and b_3 respectively. In the second round – starting with a_1 – each players votes for bid b_1, by using either the $accept$ or the $reject$ action. Let assume that a_1 and a_2 accept this offer, and a_3 rejects it. Since this protocol needs consensus, the offer b_1 is rejected. The third round is used to decide whether b_2 is acceptable. In this example a_1 rejects the offer and a_2 and a_3 accept it. Note that a_2 and a_3 still need to cast their votes, this is because the only way to communicate in this protocol is by performing actions. In the fourth and final round of this negotiation the parties decide whether b_3 is acceptable or not. All parties accept this offer, i.e. a_1, a_2, and a_3 use the $accept$ action. b_3 is now a consensus and the agreement of this negotiation.

The AMCP protocol is an alternating offers protocol in which the emphasis lays on equal opportunities for all players with respect to bidding. This in contrast to the SAOP protocol discussed earlier in Section 4.1. The essential difference is that parties do not overwrite offers and therefore a party can take all offers into account before placing its votes. This is a major difference from an information theoretical point of view.

In answer to Sub-question 2 we introduced two novel protocols: SAOP and AMCP. These protocols adhere to the fairness properties that we posed earlier in this thesis. In the experiments in the next chapter – Chapter 5 – we evaluate the performance of these protocols in terms of social welfare.

Chapter 5

Evaluating Protocols

In this chapter the evaluation of the AMCP and SAOP protocols we defined in Chapter 4 is discussed. A series of experiments were conducted to compare SAOP to AMCP using automated negotiation agents. A problem we found with performing the experiments, is that executing a full tournament in which every party negotiates against every other party including itself in all positions, comes down to n^m negotiations, where n is the number of parties and m the laterality. In our case $n = m = 3$, so we need $3^3 = 27$ negotiations for a single full tournament. In order to get significant results this needs to be repeated several times as well. We choose to evaluate the protocols using automated negotiations as this is a more resource efficient way of evaluating. The alternative, evaluating protocols using human negotiators, is more resource intensive and can be harder to reproduce. When performing experiments with humans in general, they also tend to have a learning effect, i.e. get better as they learn how the experiment works. This can pose an extra problem when ideally we want each participant to perform 27 negotiations. Because of this we use automated agent evaluation in the GENIUS negotiation environment [**?**]. We choose to use the GENIUS environment, because included in it are agents and domains that we can use to run the evaluation.

In Chapter 1 we discussed the research question and related sub-questions. Support for Sub-question 1 was given in Chapter 3 and for Sub-question 2 in Chapter 4. In this chapter we answer Sub-question 3: *How do we evaluate the efficiency and social welfare for multilateral multiparty turn-taking peer negotiation protocols?* The rest of this chapter is divided into four sections: In Section 5.1, we explain the system under test: We define which agents, protocols and domain we use for the experiment. Next the metrics used to evaluate the system are presented in Section 5.2. Using these definitions the system is evaluated in Section 5.3 and finally some conclusions are drawn in Section 5.4.

5.1 Setup

In order to perform an evaluation of the AMCP and SAOP protocols, we need to test a full system, i.e. a protocol in combination with agents and a domain. For this experiment we use state-of-the-art agents to ensure that the protocols are being pushed to their limits and we use a domain that is non-linear and non-enumerable so that the agents can not sort the list of offers, as we hypothesize that the agents have to rely more on the information provided by the protocols in this case, which makes it easier to detect any differences. With this in mind we consider the winning agents of the ANAC2015 competition [**?**] for the automated negotiation experiment. Agents from this competition were chosen, because the SAOP protocol that we created is used in the competition. Also, we consider the winners of this competition a fair representation of the state-of-the-art in negotiation agents strategies. The experiments are performed on the non-linear non-enumerable *Railway* domain, which is included with GENIUS. The Railway domain is a discrete domain consisting 8 issues with 4 to 16 discrete values totaling 655.360 unique offers with interaction effects between to offers. It is also a non-linear domain: iterating over all discrete values of a single issues while keeping the rest of the issues constant will give a non-linear change in utility. This domain is selected so that the agents can not sort the list of offers, as we hypothesize that the agents have to rely more on the information provided by the protocols in this case, which makes it easier to detect any differences. The railway domain is designed to be a challenging, competitive domain.

In ANAC2015, there are two categories in which an automated agent can win: Best individual utility and best Nash product. In the first category the top 3 consists of: *Atlas3*, *ParsAgent* and *RandomDance*, while in the second category the first place again goes to *Atlas3*, followed by *Mercury* and *JonnyBlack*. Since all agents are created with the SAOP protocol in mind, we adapt their source code to be able to make use of the AMCP protocol, which requires slightly different responses. For this purpose we use *Atlas3*, *ParsAgent*, and *RandomDance* since these can be adapted for AMCP most straightforward.

5.2 Evaluation Metrics

In Sub-question 3 we pose the question of evaluating protocols in automated agent peer negotiation. In this section we come up with evaluation metrics for measuring efficiency and social welfare of protocols. We evaluate efficiency using *negotiation speed* and *consensus percentage*. We also compare the negotiation outcome to the Nash bargaining solution, the Kalai-Smorodinsky bargaining solution and the egalitarian social welfare solution. Next we describe each of the metrics for evaluating efficiency and welfare in detail and motivate why it is included. We will also form a hypothesis of each evaluation metric and protocol combination.

We define two metrics for negotiation speed: *proposed offers per second* and *percentage of agreements*. The rationale behind using offers per second as metric for efficiency is that with each offer proposed, a point in the negotiation space is examined. The more points can be explored in a certain amount of time, the faster the protocol will converge to an agreement. We hypothesize that SAOP will be faster in this sense than AMCP as it can explore more options with the same number of interactions. We expect AMCP to have a lower amount of offers per negotiation The consensus percentage metric is used as a second metric to evaluate efficiency. Ideally, we want all negotiations to resolve in consensus. Because the BATNA is zero – i.e. agents get zero utility when not finding consensus – and we use state-of-the-art agents, we expect near 100% of the negotiations end in consensus. We hypothesize that SAOP will perform slightly better in this regard, because of the previous hypothesis that it can perform more offers per second. Because of this, it has more opportunities to concede and find consensus in that way.

Using the utility of the *negotiation outcome* itself as a metric to measure efficiency and fairness of a protocol can prove problematic because it is a n-tuple of utilities, where n is the number of agents. The individual utilities themselves give little information about the efficiency and fairness of the negotiation. Unless one outcome strictly dominates another, in which case it is better for at least one party, not much sensible information with regards to efficiency can be distilled. If we want to measure protocol welfare through negotiation outcome we must therefore explore metrics that measure the outcome as a whole. We turn to Pareto optimality to find such metrics. Pareto optimal solutions are solutions in which the agreement as a whole can not improve without at least one party getting a worse utility. The set of all Pareto optimal points is called the Pareto frontier. We consider three special negotiation solutions on this Pareto frontier to find our metrics; namely the Nash points, which are agreement points that adhere to the Nash bargaining solution [28], the Kalai-Smorodinsky point which is the agreement point of the Kalai-Smorodinsky bargaining solution [21] or utilitarian welfare solution and the egalitarian point, which is the agreement point of the social welfare solution [20]. As each solution is on the Pareto frontier, these offers can be considered efficient and, depending on personal philosophical preferences, also fair. Intuitively, the Nash equilibrium maximizes the product of normalized utilities while the Kalai-Smorodinsky solution maximizes the utility gain of the normalized utilities. The egalitarian solutions maximizes the minimum of utilities. Background information on these theoretical bargaining solutions can be found in Chapter 2.

In order to compare the experimental results to these ideal points, we measure the distance of the solution to the optimal solution. We could use the Euclidean distance in utility space between the *n*-dimensional solution point and the theoretical optimal solution. This approach however, can be problematic especially in multiparty negotiations on non-linear domains were two solutions with nearly identical properties can be far away when comparing distance between utility vectors of the solutions, but close in relevant outcome. The relevant value to compare by differs per metric, for example for the Nash bargaining solution we compare the Nash products. Thus, we will define three derived distance metrics in utility space that take into account the type of solution that the specific bargaining solution is trying to achieve. As an illustration, in Figure 5.1 we explain the differences between calculating the distance using utility vectors and Nash product for the Nash bargaining solution for a hypothetical two-dimensional negotiation space.

Definition 5.1. d_{Nash}
$d_{Nash}(\mathbf{u}) = \prod_i \mathbf{ns}_i - \prod_i \mathbf{u}_i$ where $\mathbf{ns}$ is the utility tuple of the Nash bargaining solution and $\mathbf{u}$ is the utility tuple of the actual solution.

i.e. d_{Nash} is the difference in product between the given solution and the Nash bargaining solution.

Definition 5.2. $d_{Kalai-Smorodinsky}$
$d_{Kalai-Smorodinsky}(\mathbf{u}) = \sum_i \mathbf{kss}_i - \sum \mathbf{u}_i$ where $\mathbf{kss}$ is the utility tuple of the Kalai-Smorodinsky bargaining solution and $\mathbf{u}$ is the utility tuple of the actual solution.

i.e. $d_{Kalai-Smorodinsky}$ is the difference in sum of utility for the given solution and the Kalai-Smorodinsky solution.

Definition 5.3. $d_{egalitarian}$
$d_{egalitarian}(\mathbf{u}) = \min_i \mathbf{es}_i - \min_i \mathbf{u}_i$ where $\mathbf{es}$ is the utility tuple of the egalitarian social welfare solution point and $\mathbf{u}$ is the utility tuple of the actual solution.

i.e. $d_{egalitarian}$ is the distance between the poorest utility in the given solution and the poorest utility in the egalitarian social welfare point.

These three distance metrics and the bargaining solutions they are based on all have their own strengths and weaknesses. As explained in Chapter 2, they are based on different schools of thought and make different trade-offs. It is worth noting that, while d_{Nash} and $d_{Kalai_Smorodinksy}$ are scale invariant,,the egalitarian metric $d_{egalitarian}$ is not. For our experiments the utility scales are fixed between 0 and 1 for each party, which means that $d_{egalitarian}$ only measures utility gains relative to the maximum gain for that specific party. This in turn means that we ignore that some parties might have a better starting position. For our particular choice of domain the starting positions and the maximum utility gain is equal for each party, but this is not a condition that generally holds when using this evaluation method on arbitrary domains. We included all three metrics in our evaluation method so that we can evaluate the protocols from different points of view.

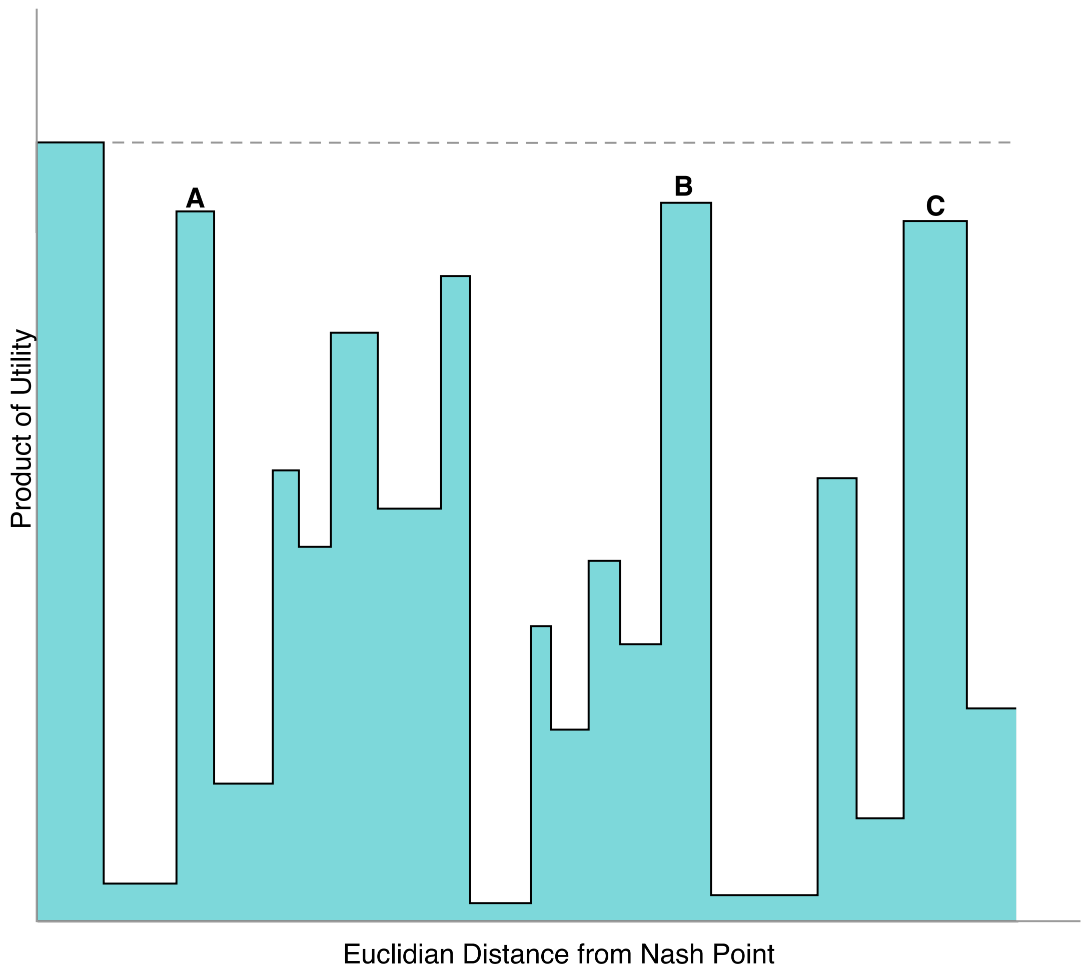

Figure 5.1. In a multiparty negotiation on a non-linear domain, close-to-optimal solutions could be far away from the optimal solution in a euclidean distance sense (i.e. when calculating the distance between solutions using the difference between the utilities of the solution vectors). For example, while solutions **A**, **B** and **C** of this hypothetical negotiation domain are all close to the Nash point when considering their product of utilities, they are relatively far away when considering euclidean distances between their respective utility vectors and the Nash point.

We expect negotiations using the SAOP protocol to obtain a higher utilitarian welfare and thus to have a lower value for $d_{Kalai-Smorodinsky}$, because the protocol promotes conceding. We also expect using the AMCP protocol to obtain a higher egalitarian welfare and therefore a lower value for $d_{egalitarian}$, because parties have more information before committing to an offer.

5.3 Analysis

In order to see the evaluation metrics in action we now perform an evaluation using SAOP and AMCP as protocols described in Chapter 4. The evaluation consists of a a series of negotiations using the *Railway* domain that comes with GENIUS which provides three utility profiles. We use *Atlas3*, *ParsAgent*, and *RandomDance* as agents for this experiment. A full tournament is run, where each party plays against each other party as well as itself in with each profile. i.e. given three parties *a*, *b* and *c* this generates negotiations *aaa*, *aab*, ..., *ccc*, where each of the three positions signifies one of the negotiation profiles. Such a tournament with three agents consists of $3^3 = 27$ negotiations. In total we will perform 37 tournaments per protocol which amounts to a total of 999 negotiations, so we have a sample size of $N = 999$ for both protocols. The deadline for each negotiation is set to 3 minutes and if no agreement is reached within that time, all parties involved in the given negotiation get zero utility, i.e. the BATNA is zero. We now continue by evaluating each of the three distance metrics described earlier followed by an evaluation of speed in number rounds per second and the percentage of agreements that the agents find.

In the experiments we check whether there is any difference between the d_{Nash} (Definition 5.1), $d_{Kalai-Smorodinsky}$ (Definition 5.2) and $d_{egalitarian}$ (Definition 5.3) for SAOP and AMCP. Each triplet of negotiators is compared between the protocols using three distance metrics in a paired-sample t-test. After running the experiments we firstly plot the mean values of the three distance metrics to get an intuitive feeling of the results. These mean values can be found in Figure 5.2. One observation is that the SAOP and AMCP bars, respectively the first and second bar in the figure are close together, which is an indication that if we find a significant difference, it might be a small result. Another observation is that the black bars are quite small for all negotiations. These bars represent the 95% confidence interval, which means we are quite sure of the results. The actual statistical tests will be performed in the next paragraph. We hypothesized that the AMCP will only perform better on the $d_{egalitarian}$ metric, not all three. We will now perform some formal statistical tests to give some formal support for these intuitive observations.

We perform a 3 t-tests; one for each metric. Each t-tests is a paired sample tests in which we compare the results for the first protocol, SAOP, of a certain group with the results

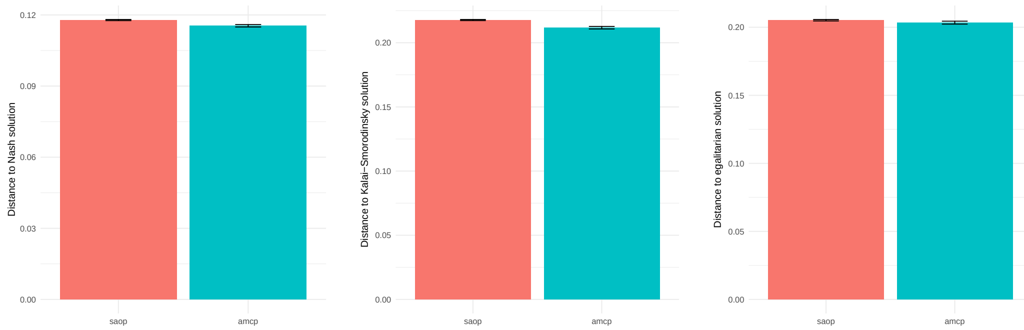

Figure 5.2. Here we see the d_{Nash}, $d_{Kalai-Smorodinsky}$, and $d_{egalitarian}$ plotted respectively. Please note that the y-scales are different, but since the metrics are different as well, that is irrelevant. Lower vales are closer to each respective metric. The error bars in the plot represent the 95% confidence intervals of the values. In each of the three sub-plots, the left bar represents the distance to the given metric for SAOP and the right bar represents the distance to that metric for AMCP. This plot gives a general idea about the performance of the two protocols and helps with putting the t-tests in context.

of the second protocol, AMCP, for the same group. We define the *null*-hypotheses, H_0: *There is no difference between SAOP and AMCP* for each of the metric and the alternative hypotheses, H_1: *There is a difference between SAOP and AMCP* for each metric. For a statistical test to be significant, we should see a p-value below a certain threshold. A commonly used value for threshold α in economics and social studies is $\alpha = 0.05$, meaning that if our model has a p-value below this threshold, we are 95% confident that our findings are not due to noise. In this experiment, we use this value for α because the field of negotiation is interconnected with those disciplines and it will be less involving to compare results with later studies. As we can see in Table 5.1, all p-values for the experiments are below threshold α, so we can reject H_0 for these observations and adopt H_1: There is a statistical significant difference between SAOP and AMCP for automated negotiations on all three metrics. As was expected, we find small differences indeed: The means of the paired differences for SAOP and AMCP on the metrics d_{Nash}, $d_{Kalai-Smorodinsky}$ and $d_{egalitarian}$ are respectively $3.3 \cdot 10^{-3}$, $3.0 \cdot 10^{-3}$ and $2.5 \cdot 10^{-3}$. An overview of these values, as well as the absolute mean values for distances to each metric can be found in table 5.2.

Metric	*t*-value	df	p-value
d_{Nash}	29.722	998	$\ll 0.001$
$d_{Kalai-Smorodinsky}$	32.009	998	$\ll 0.001$
$d_{egalitarian}$	18.795	998	$\ll 0.001$

Table 5.1. In this table we summarized the results of the 3 individual paired sample t-tests; one for each distance metric. All tests were performed on the same set of automated negotiation results. We see that the tests have 998 degrees of freedom because our sample size was $N = 999$.

Metric	mean SAOP	mean AMCP	mean difference	average pair-wise difference
d_{Nash}	0.1179	0.1154	$2.4 \cdot 10^{-3}$	$3.3 \cdot 10^{-3}$
$d_{Kalai-Smorodinsky}$	0.2177	0.2117	$6.0 \cdot 10^{-3}$	$3.0 \cdot 10^{-3}$
$d_{egalitarian}$	0.2052	0.2034	$1.8 \cdot 10^{-3}$	$2.5 \cdot 10^{-3}$

Table 5.2. The means and differences for SAOP and AMCP on the thee distance metrics: d_{Nash}, $d_{Kalai-Smorodinsky}$, $d_{egalitarian}$. Included in this table are two types of differences: In the 4^{th} column the difference between the means of each protocol is shown, while in the 4^{th} column we noted the difference between the paired samples (pair-wise difference). i.e. for each distance metric we calculated the average: $\frac{\sum_i |\text{SAOP}_i - \text{AMCP}_i|}{N}$ where SAOP$_i$ and AMCP$_i$ denote the same i^{th} configuration for SAOP and AMCP respectively and N denotes the sample size.

As mentioned earlier in this chapter, we define speed as number of offers proposed per second. We only keep track of this for the automated negotiation as it is easier to keep track of the number of negotiations they did. The simulation environment GENIUS keeps track of the elapsed time and rounds for each negotiation. Unfortunately, we can not rely on GENIUS' built-in round metric, because it only book keeps rounds. We edited the agents that we used for negotiating to include a ticker that fires every time that a new offer is proposed. the resulting rounds per second for each protocol are plotted in Figure 5.3. For SAOP the mean number of offers per second is 141, while AMCP has a mean of 43.1 offers per second. While the negotiators had 180 seconds of negotiation time for each negotiation, the mean time until agreement was 149.62 seconds for SAOP and 147.29 seconds for AMCP.

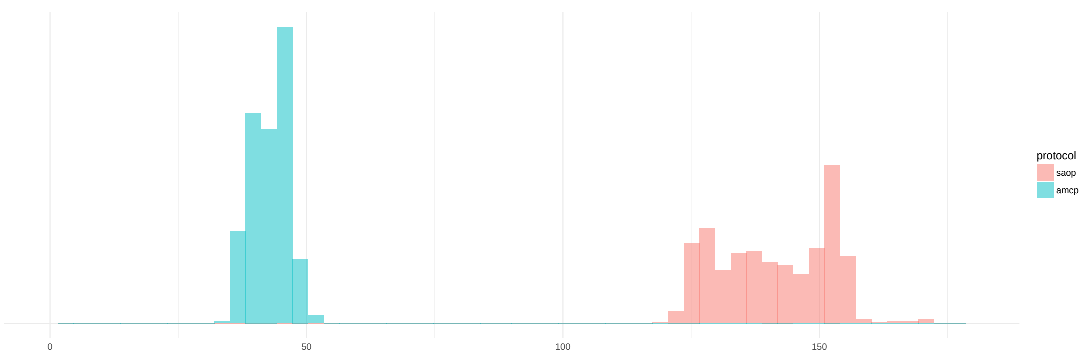

Figure 5.3. Histogram of the speed of proposing offers in offers per second. Bin width is 60. On the left is AMCP with it's center around 43.1 offers per second. On the right is SAOP with around 141 offers per second. We see that there is much less spread within the AMCP protocol than there is in the SAOP protocol. This is because best and worst-case scenario's are equal in AMCP and differ by a factor equal to the number of participating agents in the SAOP protocol. Both graph's seem to have a similar distribution with two modi. This lower modus possibly stems from the fact that when the ATLAS3 agent plays against itself, it finds consensus in less time than the deadline. This makes the start up phase in which no offers are proposed more significant and subsequently lowers the number of proposed offers per second.

We found that agents using AMCP need less offers to come to an agreement than SAOP. In figure 5.4 the log_{10} of the number of rounds per negotiation is plotted. We see a second, smaller modus a bit to the right of the main modus for both protocols. Looking at the data we found that this second modus stems from the *Atlas3* agent: When playing against itself, it proposes significantly more offers than when playing against other agents. AMCP needs an average of 762.2 offers to come to an agreement, while SAOP needs 2006 offers on average. Because of the data skew, it is also interesting to compare the medians: 48 offers for AMCP and 125 offers for SAOP.

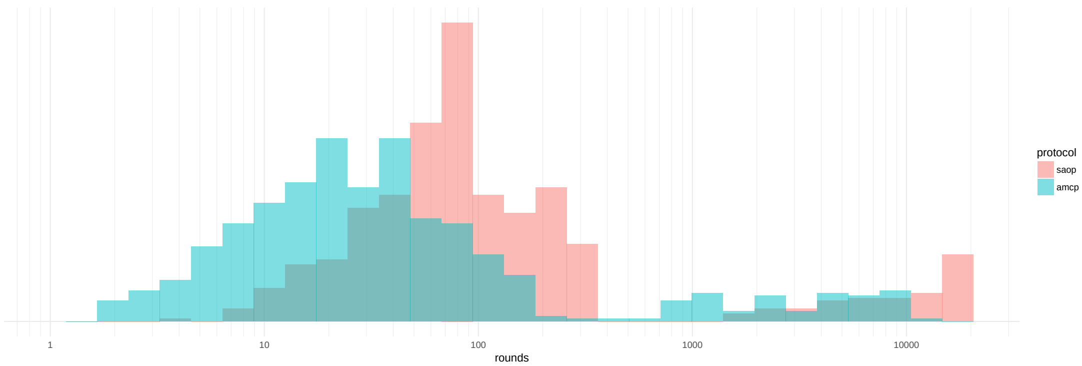

Figure 5.4. Histogram of number of offers until consensus using bin width of 30. The protocol to the left is AMCP and to the right SAOP. We see that both protocols have two modi. The smaller modus to the right stems from the *Atlas3* agent playing against itself. In this case it proposes significantly more offers. The x-axes is in log_{10} scale.

Lastly we looked at the number of times an agent came to an agreement. We expected this to be near 100%, but found a perfect 100% score for both the SAOP and AMCP protocol. While this is a good metric to measure efficiency, we can not use it to discriminate between SAOP and AMCP so we will discard it for this evaluation. The perfect score with respect to percentage of agreements can be contributed to the fact that we set a BATNA of 0 and used the winning ANAC agents to run the negotiations. These agents behave rational and rather have some utility by agreeing on a sub-optimal solution than get 0 utility.

In this section we saw the statistical analysis and experimental results. In the next section we will link them to the hypotheses proposed in Section 5.3 and draw some conclusions.

5.4 Conclusion

The chosen evaluation metrics give some insight in the efficiency and welfare of the protocols. Looking at the negotiation outcome's welfare from the three distance metrics chosen gives us a general idea of the protocol's performance from multiple viewpoints. We expected the metric *number of offers proposed per second* to give more insight into the efficiency of the protocols – especially since the protocols under test have comparable mean times until agreement – but since the welfare of both protocols is similar we expect that the negotiation speed itself is not a good measure for efficiency: It measures the quantity of offers, not the quality. It can be a useful metric when combined with a measurement of proposed offer quality.

Given the intuitive and statistical analysis, we conclude that the chosen ANAC agents prefer to use the AMCP protocol over the SAOP protocol because it strictly dominates its results. We also conclude that the advantage given by using the AMCP protocol over SAOP only grants a slightly better negotiation outcome for automated negotations using these specific agents. We expect that results might be the other way around for different agent combinations. Because of the small differences, we conclude that other factors might be more decisive in choosing which protocol is more fair or efficient.

Earlier, in this chapter we saw that SAOP has a mean speed of 141 offers per second, while AMCP has a mean speed of 43.1 proposed offers per second. This means that SAOP proposes around 3.27 times as many offers as AMCP in our given setup. This difference is expected, as we can also deduce this from the protocols themselves: In the most optimistic case, SAOP will propose an offer on every round if not accepting. In the case were parties only accept at the end, this means that we get $rn - (n-1) = O(rn)$ offers on the table, where r is the number of rounds, and n the number of parties. In the most pessimistic case, all but two parties accept the offer each round and we get $2r - (n-1) = \Omega(r)$ offers on the table. In the case of AMCP, most optimistic and pessimistic cases are equal, because each party needs to vote on each offer. Only every $n+1$ rounds is a new round of offers, so we get $\frac{rn}{n+1} = \Theta(r)$ every time. When we compare best case scenarios, they differ by a factor n, which is the number of agents. We use $n = 3$ for all tests, so we expect around a factor three difference, which we get. We can conclude that SAOP is around three times faster than AMCP with respect to proposed number of protocols per minute. It should be noted however that this metric says nothing about the percentage of repeated offers and quality of proposed offers.

When looking at the number of offers until consensus, we see that both the mean and median are in favor of AMCP. when combining this result with the previously reported speed, it appears that agents using SAOP find their consensus by proposing more offers of lower quality and agents using AMCP find their consensus by proposing less offers of a higher quality.

Chapter 6

Conclusion

In this thesis we introduced two novel multiparty multilateral turn-taking peer negotiation protocols that are extensions of the bilteral alternating offer protocol and maintain its fairness properties, we also proposed a method for evaluating such protocols. Firstly, we identified related studies and concluded that although multiple multilateral protocols have been proposed over the years, to our knowledge no extension of the bilateral alternating offer protocol to multiparty multilateral turn-taking peer negotiation protocols have been proposed. We defined a design space and used it as a basis to create two novel protocols: Stacked Alternating Offer Protocol (SAOP) and Alternating Multi-offer Consensus Protocol (AMCP). A method for evaluating protocols is introduced. Experimental negotiations with automated ANAC agents were conducted using the new evaluation method to compare the efficiency and welfare of the two proposed protocols using both qualitative and quantitative analyses methods. Using this work, we can now answer the research questions that were posed at the start of this thesis:

In this thesis we introduced two novel multiparty multilateral turn-taking peer negotiation protocols and evaluated their efficiency and fairness. F

Research question. In what ways can the bilateral turn-taking negotiation protocol be extended to a multiparty multilateral turn-taking peer negotiation protocol such that its fairness properties are maintained?

Sub-question 1. What is the formal design space for multilateral multiparty turn-taking peer negation protocols?

Sub-question 2. Can we design multilateral multiparty turn-taking peer negotiation protocols that maintain the fairness properties of the bilateral turn-taking negotiation protocol?

Sub-question 3. How do we evaluate the efficiency and social welfare of multilateral multiparty turn-taking peer negation protocols?

In answer to Sub-question 1 we defined the design space in Chapter 3. This design space only defines a subset of protocols, namely the multiparty multilateral turn-taking negotiation protocols. The concrete answer to this question is: The design space for this specific subset of negotiation protocols is defined by the tuple P as described in Definition 3.10 in Chapter 3.

In Chapter 4, using the design space, we defined two novel protocols: SAOP and AMCP adhering to the fairness properties that we identified, thus answering Sub-question 2. Creating these protocols also allowed us to evaluate the design space while we were working on it. Using the definitions of the design space we were able to make an informed decision about the trade-offs between different aspects of the protocol, such as the allowed actions (Definition 3.3 in Chapter 4) in each round, which is the main difference between the SAOP and AMCP protocol.

Sub-question 3 encompasses the brunt of the research. While formulating an answer to this question we found that the question had to be dissected into smaller parts. We will next draw partial conclusions for each of terms speed and negotiation outcome respectively, followed by a general statement about the efficiency and fairness of the protocols as a whole as well as the usability of the metrics themselves.

In terms of speed, in our experiments SAOP outperformed AMCP by a factor 3.27. We deduced that theoretically SAOP is able to propose a factor n more offers per second than AMCP, where n is the number of agents. By using SAOP, the automated agents were able to propose an average of 141 offers per second, while by using AMCP only 43.1 offers per second could be proposed. These numbers do not take into account repeated offers, which both protocols allow. It also does not take into account the quality of the offers proposed. When comparing solely on quantity of offers, SAOP is clearly better.

For the negotiation outcome, we defined three distance metrics: d_{Nash}, $d_{Kalai-Smorodinsky}$, and $d_{egalitarian}$ (Definitions 5.1, 5.2 and 5.3 respectively). For agents we found that AMCP outperforms SAOP on all three metrics. It should be noted that agents performed good on both metrics and there was little margin for improvement. The improvement in mean distance of AMCP over SAOP is $3.3 \cdot 10^{-3}$ for d_{Nash}, $3.0 \cdot 10^{-3}$ for $d_{Kalai-Smorodinsky}$ and $2.5 \cdot 10^{-3}$ for $d_{egalitarian}$.

Given the SAOP and AMCP protocol and the given the agents used to evaluate, we conclude that AMCP is the most fair protocol. The ANAC agents that are used in the experimental negotiations give better answers on all the given metrics. Only in terms of offers per second does SAOP outperform AMCP, but since AMCP gives a similar or even better outcome, it becomes apparent that the quality of offers matters more than the quantity.

In answer to Sub-question 3, we consider the proposed metrics for speed and negotiation outcome. The metric speed in proposed offers per second does not directly measuring efficiency. The protocol with the lower speed was evaluated as being better in terms of negotiation outcome. We expect that this stems from the fact that the offers per second metric does not take into account repeated offers and thus does not give a good metric for searched bid space. We also conclude that using three distance metrics: d_{Nash}, $d_{Kalai-Smorodinsky}$, and $d_{egalitarian}$ as defined in 5.1, 5.2 and 5.3 respectively, gives a good comparison on fairness. Using all three metrics give different moral viewpoints on the negotiation, which give a good insight in how fair the results are.

In answer to the main question of this thesis, we conclude that it is possible to extend the bilateral turn-taking negotiation protocol to suitable multiparty multilateral peer negotiation protocols that are maintain fairness properties:

1) Each party can propose their own offers

2) Each party gets the same information about other offers

3) Each party can reject an offer

We've shown two protocols, SAOP and AMCP that are examples of such an extension. Using the evaluation methods described earlier, we can identify difference of performance on three social welfare metrics. We conclude that given the SAOP and AMCP protocol, the AMCP protocol is more efficient in terms of promoting social welfare: It dominates SAOP on the three defined social welfare metrics.

Chapter 7

Discussion & Future Work

In this chapter, we discuss findings that we established in this thesis. The findings also give suggestions and handles for future work, which will be mentioned where appropriate.

One issue with AMCP is that ideally we simulate a closed envelope protocol. In such a protocol every negotiating party first writes an offer on a pieces of paper and puts them in envelopes. Then followed by one round in which each party receives all offers and can vote. Finally the information is shared, optionally even anonymously. In the AMCP implementation we have one round of voting per offer and all parties receive all information on votes in between voting rounds. This will bias the information sharing which in turn might negatively influence the social welfare of the agreement

Another issue with our experiments involving AMCP, is that we ran the experiments using agents that are not created specifically for the protocol. The ANAC2015 agents that we used, are created with only the SAOP protocol in mind and will therefore not take full advantage of the infromation sharing properties of the AMCP protocol.

One argument for choosing SAOP over AMCP can be the protocol complexity. AMCP is a more elaborate protocol. Although we did not quantify it, this might make it harder for agent designers to utilize the protocol to the fullest. A simpler protocol appeared to be easier as far as reasoning and creation of strategies is concerned.

At the time of the experiments, we were unable to directly compare this protocol to other protocols because we could not find any. Most related protocols have a mediated strategy or are based on multiple bilateral negotiation, e.g. such as marketplace protocols. It would not be to interesting to compare our protocols to the bilateral alternating offer protocol (AOP), because especially SAOP and AOP behave identical for negotiations between two parties. In recent work, Caillere et al. propose a multiparty multilateral protocol [6], which could be compared to SAOP and AMCP in future work. Analyzing that protocol can also be an interesting case study to see how the evaluation method holds up.

Related to the previous point, in Chapter 3 we define a formal design space. The design space proposed only supports a subset of protocols; namely multiparty multilateral turn-taking peer protocols. It could be generalized even further. Two forms of negotiation protocols that would be interesting to include are mediator-based protocols, and buyers and sellers protocols. As mentioned in the introduction in Chapter 1, we decided to exclude mediator based protocols because it is already being extensively researched. Including it in future work could be a bridge between this work and the work done on mediator-based negotiation. The design space as proposed in this paper assumes that the negotiations are always peer to peer, which is a symmetric negotiation. In future work, we could generalize the design space further by allowing negotiators in specific asymmetric roles, such as buyer/seller or mediator roles.

One experiment that we were unable to complete due to resource constraints, is figure out the relationship between domain and protocol. We expect that there is a form of interaction between the two. If we define sort of *degree of conflict* metric for each domain based on the shape of its bid space, we can then compare this to the protocol and actor to check for interaction effects. We expect that there is a positive correlation between cooperative domains, human actors and the $d_{egalitarian}$ metric.

We limited the scope of the design space proposed in Chapter 3 due to resource constraints. One of the design dimensions we did not include is the possibility to differ on type of offers. The currently proposed design space only allows for *packages deals* while including partial offers would be an interesting area to explorer, bringing the negotiations more inline with how humans negotiate. One way to include partial offers in the proposed design space is by defining *bid* elements as sets of issues. We can then allow partial offers by allowing subsets of issues as valid offers.

All of the options for extending the design space discussed in previous paragraphs serve to make the design space encompass more types of negotiation protocols. This is also drills down into a related issue with defining such a design space: It is tailored specifically to this types of negotiations and it will take effort to increase the types of negotiations it can encompass.

We used the design space for a specific purpose, namely creating and later evaluating multiparty multilateral turn-taking peer negotiation protocols. It can also be used to improve the properties of a protocol as described in Table 3.1. The design space is only a first step towards this goal. In future work, we should link the design space explicitly to the protocol properties. We can then use the created framework to pick a protocol with favorable properties for the negotiation we are conducting.

We performed some preliminary evaluations using human negotiators which we initially wanted to include in this thesis as well. Eventually we decided against it, as it is a lot of work to provide a large enough sample size due to the fact that we evaluated three-way negotiations in this thesis. Also when designing an experiment for protocol evaluation using human negotiators, it has to be designed in such a way that any learning effects are mitigated, something we failed to do in our preliminary experiment. In future work, it can be interesting to include a well designed human evaluation so that we can see what the difference are between human and automated negotiators in protocol preference and to see which protocols can be used for both types of negotiator.

We end on a somewhat philosophical note. As we have seen in this paper, the used protocol has an influence on the outcome of the negotiation. This might lead to meta negotiations about which protocol to use. We did not find a clear solution to this "chicken and egg" type of problem. In every day life we see different solutions to this problem. In schools and universities there are examination regulations to which the organization commits itself for multiple years. In Europe, municipalities participate in public European procurement; they have to consider every offer that adheres to a set of formalized requirements.

Acknowledgments

Firstly, I would like to express my gratitude to dr. de Weerdt, my daily supervisor for this thesis project. His guidance and support helped me keep on track and he motivated me to keep going. Also our weekly meetings and his thought provoking questions really helped in improving the quality of this work.

Secondly, I would like to thank dr. Aydogan for supervising and supporting me during the first part of this thesis project and for introducing me to this field with her enthusiasm, caring and encouragement.

My thanks goes out to prof. dr. Jonker for her insightfulness and for the fun academic discussions on the subject of this thesis and of negotiation theory in general. She helped me increase my knowledge of the field of negotiation theory. Her immense domain knowledge is a source of inspiration for anyone.

I also thank Dr. Hindrik for investing his time and thoughts in reading this work as member of the thesis committee.

I thank Anita Hoogmoed for helping me with the paperwork, planning and for helping me out all the times I forgot my key. Ruud de Jong I thank for providing a computer to run the GENIUS simulations on and for providing good coffee.

Off course I thank my family: Father, mother and sister for supporting me spiritually during this thesis and generally in life, my uncle Hans Festen for our weekly sessions on statistics, and all my friends at S.M.N. Kurk for their moral support.

Last but not least, I thank Eleni for all her years of support, in good and bad times, giving me the strength to pursue my dreams and making life in general more colorful.

Related work search strategy

In the introduction, Chapter 1, we introduced some related work to this thesis. The brunt of related work has been found in part by collabration and inquiring with colleges in the field, specifically Prof. Jonkers and Dr. de Weerdt of Delft University of Technology, Dr. Aydogan of Istanbul Ozyegin University and, Dr. Fujita of Tokyo University of Agriculture and Technology. Part of the recent work has been found by attending the 2015 conference and reading the post-proceedings of the AAMAS conference. We also searched for related work using the search engines of Google Scholar [14] and IEEE Xplore [18]. We used the following search terms on those websites: *aamas*, *anac*, *anac2010*, *anac2011*, *anac2012*, *anac2013*, *anac2014*, *anac2015*, *multiparty protocol*, *multilateral protocol*, *many-to-many protocol*, *protocol design*, *saop*, *amcp*, *ammp*, *amop*, *acan*, *negotiation protocol*. For each of these search terms we looked for papers published after 2013 with possibly related titles. For the possibly related work, we then reviewed the abstracts and decided if it would be viable to include in the introduction. The initial search for related work using the search engines was executed in September, 2016, and the search was repeated in August, 2017. All the work included in the introduction was fetched either in September, 2016 or by inquiring with colegues as discussed above. We did find new related work in the second search in August, 2017, namely the work by Caillere et al. [6]. A remark about this finding was included in Chapter 7.

Bibliography

[1] Reyhan AydoÄ§an, David Festen, Koen V. Hindriks, and Catholijn M. Jonker. Alternating offers protocols for multilateral negotiation. In *Studies in Computational Intelligence*, volume 674, pages 153–167. Springer, Cham, 2017.

[2] Reyhan AydoÄ§an, Koen V. Hindriks, and Catholijn M. Jonker. Multilateral mediated negotiation protocols with feedback. In *Studies in Computational Intelligence*, volume 535, pages 43–59. 2014.

[3] Tim Baarslag. *What to Bid and When to Stop*. Number september 2014. 2014.

[4] Tim Baarslag, Reyhan AydoÄ§an, Koen V. Hindriks, Katsuhide Fujita, Takayuki Ito, and Catholijn M. Jonker. The Automated Negotiating Agents Competition, 2010–2015. *AI Magazine*, 36(4):2010–2014, 2015.

[5] Nicholas Bayne and Stephen Woolcock. *The new economic diplomacy : decision-making and negotiation in international economic relations*. 2007.

[6] Romain Caillere, Souhila Arib, Samir Aknine, and Chantal Berdier. A Multiagent Multilateral Negotiation Protocol for Joint Decision-Making. *Recent Advances in Agent-based Complex Automated Negotiation*, 638:71–88, 2016.

[7] Dave de Jonge and Carles Sierra. NB3: a multilateral negotiation algorithm for large, non-linear agreement spaces with limited time. *Autonomous Agents and Multi-Agent Systems*, 29(5):896–942, sep 2015.

[8] Enrique De La Hoz, Miguel A. Lopez-Carmona, Mark Klein, and Ivan Marsa-Maestre. Consensus policy based multi-agent negotiation. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, volume 7047 LNAI, pages 159–173, 2011.

[9] Ulle Endriss. Monotonic concession protocols for multilateral negotiation. *Proceedings of the fifth international joint conference on Autonomous agents and multiagent systems - AAMAS '06*, page 392, 2006.

[10] Shaheen Fatima, Sarit Kraus, and Michael J. Wooldridge. *Principles of automated negotiation*. Cambridge University Press New York, NY, USA ©2014, 2014.

[11] Roger Fisher, William Ury, and Bruce Patton. *Getting to Yes:Negotiating Agreement Without Giving In*. 1983.

[12] Stan Franklin and Art Graesser. Is It an agent, or just a program?: A taxonomy for autonomous agents. In Jörg P Müller, Michael J Wooldridge, and Nicholas R Jennings, editors, *Intelligent Agents III Agent Theories, Architectures, and Languages: ECAI'96 Workshop (ATAL) Budapest, Hungary, August 12–13, 1996 Proceedings*, pages 21–35. Springer Berlin Heidelberg, Berlin, Heidelberg, 1997.

[13] Katsuhide Fujita, Takayuki Ito, and Mark Klein. Preliminary Result on Secure Protocols for Multiple Issue Negotiation Problems. In *Intelligent Agents and Multi-Agent Systems*, volume 3, pages 161–172. Springer Berlin Heidelberg, Berlin, Heidelberg, 2008.

[14] Google. Google Scholar <scholar.google.com>.

[15] Hiromitsu Hattori, Mark Klein, and Takayuki Ito. A Multi-Phase Protocol for Negotiation with Interdependent Issues. In *2007 IEEE/WIC/ACM International Conference on Intelligent Agent Technology (IAT'07)*, pages 153–159. IEEE, nov 2007.

[16] Miniar Hemaissia, Amal El Fallah Seghrouchni, Christophe Labreuche, and Juliette Mattioli. A multilateral multi-issue negotiation protocol. In *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems - AAMAS '07*, page 1, New York, New York, USA, 2007. ACM Press.

[17] Koen V. Hindriks and Catholijn M. Jonker. Creating human-machine synergy in negotiation support systems. In *Proceedings of the 1st International Working Conference on Human Factors and Computational Models in Negotiation - HuCom '08*, pages 47–54, New York, New York, USA, 2009. ACM Press.

[18] IEEE. IEEE Xplore <ieeexplore.ieee.org>.

[19] Nicholas R. Jennings, Peyman Faratin, Alessio R. Lomuscio, S. Parsons, Michael J. Wooldridge, and Carles Sierra. Automated Negotiation: Prospects, Methods and Challenges. *Group Decision and Negotiation*, 10(2):199–215, mar 2001.

[20] Ehud Kalai. Proportional solutions to bargaining situations: interpersonal utility comparisons. *Econometrica: Journal of the Econometric Society*, pages 1623–1630, 1977.

[21] Ehud Kalai and Meir Smorodinsky. Other Solutions to Nash's Bargaining Problem. *Econometrica*, 43(3):513–518, 1975.

[22] Mark Klein, Peyman Faratin, Hiroki Sayama, and Y. Bar-Yam. Protocols for negotiating complex contracts. *IEEE Intelligent Systems*, 18(6):32–38, nov 2003.

[23] Ryszard Kowalczyk, Mihaela Ulieru, and Rainer Unland. Integrating Mobile and Intelligent Agents in Advanced E-commerce: A Survey. pages 295–313. 2003.

[24] Cuihong Li, Joseph Giampapa, and Katia Sycara-Cyranski. A review of research literature on bilateral negotiations. 2003.

[25] Alessio R. Lomuscio, Michael J. Wooldridge, and Nicholas R. Jennings. A Classification Scheme for Negotiation in Electronic Commerce. *Group Decision and Negotiation*, 12(1):31–56, 2003.

[26] R.T. Marler and J.S. Arora. Survey of multi-objective optimization methods for engineering. *Structural and Multidisciplinary Optimization*, 26(6):369–395, apr 2004.

[27] Herve Moulin. *Fair Division and Collective Welfare*, volume 1 of *MIT Press Books*. The MIT Press, jul 2004.

[28] John Nash. Non-Cooperative Games. *The Annals of Mathematics*, 54(2):286, sep 1951.

[29] Martin J. Osborne. A Course in Game Theory. *Computers & Mathematics with Applications*, 29(3):115, 1995.

[30] Martin J. Osborne and Ariel Rubinstein. *Bargaining and Markets*. 1990.

[31] Howard Raiffa. The Art & Science of Negotiation. *Harvard Law Review*, 96(4):969, feb 1983.

[32] Howard Raiffa, John Richardson, and David Metcalfe. *Negotiation Analysis: The Science and Art of Collaborative Decision Making*. 2002.

[33] John Rawls. *A theory of justice*. Harvard university press, 2009.

[34] Avi Rosenfeld, Inon Zuckerman, Erel Segal-Halevi, Osnat Drein, and Sarit Kraus. NegoChat: A Chat-based Negotiation Agent. In *Proceedings of the 2014 International Conference on Autonomous Agents and Multi-agent Systems*, AAMAS '14, pages 525–532, Richland, SC, 2014. International Foundation for Autonomous Agents and Multiagent Systems.

[35] Jeffrey Z. Rubin and Bert R. Brown. *The Social Psychology of Bargaining and Negotiation*. 1975.

[36] Ariel Rubinstein. Perfect Equilibrium in a Bargaining Model. *Econometrica*, 50(1):97, jan 1982.

[37] Melvin F. Shakun. Multi-bilateral Multi-issue E-negotiation in E-commerce with a Tit-for-Tat Computer Agent. *Group Decision and Negotiation*, 14(5):383–392, sep 2005.

[38] T N Wong and Fang Fang. A multi-agent protocol for multilateral negotiations in supply chain management. *International Journal of Production Research*, 48(1):271–299, 2010.

[39] H.P. Young. Monotonic solutions of cooperative games. *International Journal of Game Theory*, 14(2):65–72, jun 1985.

[40] Ronghuo Zheng, Nilanjan Chakraborty, Tinglong Dai, and Katia Sycara. Multiagent negotiation on multiple issues with incomplete information. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems*, pages 1279–1280, 2013.