ACAN2016

The 9th International Workshop on Agent-based Complex Automated Negotiations

Singapore, May 9, 2016

Preface

Complex Automated Negotiations have been widely studied and are one of the emerging areas of research in the field of Autonomous Agents and Multi-Agent Systems. The complexity in an automated negotiation depends on several factors. Complex automated negotiation scenarios are concerned with negotiation encounters where we may have for instance, a large number of agents, a large number of issues with strong interdependencies, real time constraints, concurrent and interdependent negotiation, and etc. Many real world negotiation scenarios present one or more of the mentioned elements. Software agents can support the automation of complex negotiations, by negotiating on behalf of their owners and providing adequate strategies to their owners to achieve realistic, win-win agreements. In order to provide solutions to such complex automated negotiation scenarios, research has focused on incorporating different technologies including search, CSP, graphical utility models, Bayesian nets, auctions, utility graphs, optimization and predicting and learning methods. Applications of complex automated negotiations could include e-commerce tools, decision-making support tools, negotiation support tools, collaboration tools, as well as knowledge discovery and agent learning tools.

ACAN2016 will discuss, among others, the following aspects and topics of such complex automated negotiations within the field of Autonomous Agents and Multi-Agent Systems:

- Complex Automated Negotiation Frameworks and Mechanisms
  - Bilateral and Multilateral Negotiation, High dimension Multi-Issue Negotiation, Large Scale Negotiation, Concurrent Negotiation, Multiple Negotiation, Sequential Negotiation, Negotiation under Asymmetric Information, and so on.
- Prediction of Opponent's Behaviours and Strategies in Negotiation
- Simulation Models and Platforms for Complex Negotiations
- Coordination Mechanisms for Complex Negotiations
- Matchmaking and Brokering Mechanisms
- 2-Sided Matching
- Utility and preference representations in negotiation
- Computational Complexity of Multi-Issue Negotiations
- Real-life Aspects of Electronic Negotiations
- Negotiations with Humans
- Negotiations in Social Networks etc.
- Applications for Automated Negotiations
  (e.g. cloud computing, smart grid, electronic commerce etc.)

A considerable number of researchers in various sub-communities of autonomous agents and multi-agent systems are actively collaborating on these and related issues. They are, for instance, being studied in agent negotiations, multi-issue negotiations, auctions, mechanism design, electronic commerce, voting, secure protocols, matchmaking and brokering, argumentation, co-operation mechanisms and distributed optimization. The goal of this workshop is to bring together researchers from these communities to learn about each other's approaches to the complex negotiation problems, encourages the exchange of ideas between the different areas, and potentially fosters long-term research collaborations to accelerate progress towards scaling up to larger and more realistic applications.

ACAN2016 organizers

Organization

Organizers

Prof. Katsuhide Fujita (Tokyo University of Agriculture and Technology, Japan)
Prof. Naoki Fukuta (Shizuoka University, Japan)
Prof. Takayuki Ito (Nagoya Institute of Technology, Japan)
Prof. Minjie Zhang (University of Wollongong, Australia)
Dr. Quan Bai (Auckland University of Technology, New Zealand)
Dr. Fenghui Ren (University of Wollongong, Australia)
Dr. Chao Yu (Dalian University of Technology, China)
Dr. Reyhan Aydogan (Ozyegin University, Turkey)

Program Committee Members

Dr. Rafik Hadfi (Nagoya Institute of Technology, Japan)
Dr. Valentin Robu (Heriot-Watt University, UK)
Prof. Tokuro Matsuo (Advanced Institute of Industrial Technology, Japan)
Prof. Miguel Angel Lopez-Carmona (Universidad de Alcala, Spain)
Prof. Ivan Marsa-Maestre (Universidad de Alcala, Spain)
Prof. Paul Scerri (Carnegie Mellon University, USA)
Prof. Mark Klein (MIT, USA)
Prof. Katia Sycara (Carnegie Mellon University, USA)
Dr. Raz Lin (Bar-Ilan University, Israel)
Prof. Dr. Sarit Kraus (Bar-Ilan University, Israel)
Prof. Dr. Catholjin Jonker (Delft University of Technology, The Netherlands)
Dr. Enrico Gerding (University of Southampton, UK)
Dr. Koen Hindriks (Delft University of Technology, The Netherlands)
Prof. Xudong Luo (Sun Yat-sen University, China)
Dr. Gheorghe Cosmin Silaghi (UBB Cluj, Romania)
Dr. Lotzi Boloni (University Florida, United States)
Dr. Scott Buffett (National Research Council Canada)
Dr. Jiamou Liu (Auckland University of Technology, New Zealand)
Dr. Bo An (Nanyang Technology University, Singapore)
Dr. Dayong Ye (Swinburne University, Australia)
Dr. Susel Fernandez (Nagoya Institute of Technology, Japan)
Dr. Tim Baarslag (University of Southampton, UK)
Prof. Hirofumi Yamaki (Tokyo Denki University, Japan)
Dr. Shaheen S. Fatima (Loughborough University, UK)

Index

Using Stackelberg games to model electric power grid investment in renewable energy settings

Merlinda Andoni
Heriot-Watt University
Edinburgh, UK
ma146@hw.ac.uk

Valentin Robu
Heriot-Watt University
Edinburgh, UK
v.robu@hw.ac.uk

ABSTRACT

High volumes of renewable generation, driven by favourable resource conditions, often in remote regions (such as windy islands) located away from demand centres, coupled with insufficient grid infrastructure, may lead to the utilisation of curtailment schemes. Various strategies, imposed when renewable generation exceeds local aggregate demand, have been studied from a technical scope, however, strategic interaction among different parties was not considered. Our work uses game-theoretic tools to demonstrate how the profitability of existing generators and incentives on future investment vary with different curtailment schemes. Next, we consider the combined effects of curtailment and line access rules to network upgrade decisions. We consider a two-node network, where demand and excess generation are not co-located, and we show that for common access settings, a Stackelberg game emerges between the line and local generation investors. We estimate the generation capacity and profits at equilibrium. Finally, we apply our theoretical results, to a real, UK-based transmission line reinforcement and show that our model can be utilised to set a grid access payment mechanism, that ensures the implementation of both transmission and local generation investments.

Keywords

generation incentives, renewable energy, Stackelberg game, transmission investment

1. INTRODUCTION

Integrating energy generated from renewable sources into existing grids is one of the key challenges for ensuring a sustainable, carbon-free energy future [17]. In recent years, this problem has begun to attract considerable attention from the artificial intelligence and multi-agent community, as part of the computational sustainability agenda [4, 26]. In practical developments, in many countries, the implementation of incentive mechanisms such as feed-in tariffs (FITs) or renewable obligations (ROs) have led fast increases of installed capacity from renewable generation.

A key problem is that locations which are best suited for installing new generation capacity, such as large wind turbines (and where such installations meet least planning resistance), are typically remote locations, such as windy islands, situated far from population/industry centres. Hence, a lot of new renewable generation capacity installed is subject to *curtailment* for substantial periods of the time it is operational. In plain terms, curtailment means that the energy that could have been generated at a particular time, is not generated, as there is no way for the grid to absorb this capacity, either because there is insufficient local demand or insufficient distribution or transmission network capacity to transport it elsewhere.

In practice, other than the level of ROs or FITs, each location's curtailment level (and the curtailment policy applied) is the most significant factor affecting the decision of local investors whether to invest in new renewable generation capacity [2]. While the issue of curtailment has received substantial attention in the power systems literature [8], much of this work takes a strictly technical standpoint, focusing on which curtailment schemes are easier to implement and guarantee operational stability. Yet, it is becoming increasingly clear the curtailment schemes also play a crucial role on the total installed generation capacity, due to their effect on investor decisions.

Clearly the long term solution to this distributed generation problem is to build or reinforce existing transmission and distribution lines between remote locations where capacity (such as wind turbines) can be installed, and areas of high demand. This can be expensive and technically challenging, especially as these lines often have to be deployed in harsh environments, such as at considerable depth on the sea floor. In many countries, such line reinforcements have been traditionally performed - or partially supported - by public investment, usually through the local transmission system or distribution network operators (TSOs/DNOs). This is, however, expensive and increasingly harder to justify (since only a few companies benefit from what is essentially a public investment), and often leads to considerable delays in capacity being installed. Recently, there has been renewed interest in incentivising privately built lines [7], paid by the renewable generators, possibly partially supported by TSOs/DNOs.

This raises the crucial issue of the principles of access (PoA) to be applied to the new transmission lines, as well as the interplay between the line access and the curtailment mechanism. PoA may refer to both the curtailment scheme and the line access rules, however we clarify in context which one is meant.

While lines with completely private access, so called *single merchant access* lines are possible, from a public pol-

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), John Thangarajah, Karl Tuyls, Stacy Marsella, Catholijn Jonker (eds.), May 9–13, 2016, Singapore.*

icy standpoint, it would be highly desirable if new power lines are built under a *common access* principle, especially if they are technically and/or financially supported by public means. This means, line investors may be given a license to build the line under the obligation to allow access from third parties, by setting a payment mechanism per unit of energy transported, the level of which is subject to a cap set by the regulator. The interplay between curtailment mechanism and PoA applied to transmission lines raises potentially complex issues, especially as the line investor, regulator and local renewable generation investors may have different underlying goals. In this paper, we use the tools from game theory to examine these interactions. Specifically, for a particular combination of *common access* PoA and a *fair* curtailment scheme, we show a complex *Stackelberg game* [27] can occur, in which the decision to build a transmission line depends on the equilibrium strategy of local investors to invest in additional generation capacity.

Additionally, we validate our theoretical model using a realistic case study and data of a network reinforcement problem in western Scotland. This scenario, and the issues faced by the local DNO in encouraging the investment in transmission lines, while keeping access to them public, was the initial motivation for this work. Using this principled experiments and data we show that it is possible for Ofgem (the UK system regulator) to encourage private line investors - possibly, with subsidies or public support - to build larger capacity lines under a common access rule, as long as transmission charges, in the equilibrium of the game, are set in such ways that allow sufficient profits.

In summary, our contribution to the state of the art can be stated as follows:

- First, we formalise the effect that 3 commonly-used curtailment rules have on the renewable generation capacity installed at a particular location, using a competitive equilibrium analysis approach. We show that the resulting installed capacity and profitability of different generators can differ widely under different curtailment models.
- Second, we study the grid upgrade as a game between two different parties: the line and local renewable generation investors. We show that a Stackelberg equilibrium emerges, where local renewable investors react to the new line by building additional generation capacity. We derive the amounts generated in equilibrium of this game by different parties.
- Third, we exemplify our analytical results for the case of reinforcement of the Kintyre-Hunterston link, in Western Scotland. The financial parameters used for this are realistic, and in fact much of this information was released by Ofgem and SSE (the local DNO), as part of a public consultation exercise [20].

Finally, we emphasize that, while the numerical application is specific to the UK case, our analysis and equilibrium results are general, and the underlying problem of renewable generation and demand not being co-located occurs in many other places around the globe, facing similar challenges.

The remainder of this paper is organised as follows: Sect. 2 presents the literature review and Sect. 3 elaborates on curtailment strategies. Effects on renewable capacity investment are shown in Sect. 4 and on transmission investment in Sect. 5. Numerical results of our line reinforcement case-study are presented in Sect. 6, while Sect. 7 concludes.

2. RELATED WORK

While a number of both commercial and academic studies [2, 5, 9, 8] have discussed issues around the application of curtailment strategies, these tend to focus on their technical, legal and regulatory implications, rather than their impact on decision-making of investors about generation or transmission expansion.

Strategical behaviour can be simulated by agent-based modeling, such as in [3], where renewable and network upgrade investment are jointly considered or in [13], where the effect of generation capacity on transmission planning is examined. Two alternate market structures for grid upgrades (either by system operators or private investors) are examined in [12] and show they can lead to different optimal results, however curtailment strategies were not considered. Coalition formation is used in [16] to coordinate privately developed grid infrastructure investments, to reduce inefficiencies and transmission losses [16]. The main focus of their work was the group formation and its results in configurations of multiple-location settings, not the effects of transmission access rules, which are the scope of our work.

Several works consider transmission planning or expansion at congested areas of the power network. Joskow and Tirole (2000) analyse a two-node network market behaviour, for settings of players with different market power and allocation of transmission rights [6]. Our work follows a different approach, since we specify our analysis on the transmission access rules and curtailment imposed, rather than analysing the market behaviour in areas of network constraints.

Stackelberg games in transmission upgrade have been used by several works, which considered economic analysis with social welfare [18], Locational Marginal Pricing [22] or highlight the uncertainties of renewable generation [25]. Recent works on the renewable energy domain, use Stackelberg game analysis to describe energy trading of microgrids [1, 10]. Attacker-defender problems in the security domain [15] or efficient test designing [11] are typically modeled as Stackelberg games by the multi-agent community. Finally, Zheng et al. (2015) propose a novel, crowdsourced funding model for renewable energy investments, using a sequential game-theoretic approach. However, transmission issues were not considered [28].

3. CURTAILMENT STRATEGIES

Existing power systems literature discusses a number of criteria for assessing curtailment strategies, such as fairness, transparency, efficiency or reliability [5, 8], and proposes a number of attributes which may be used, including: the technical characteristics of the generators, their size, location, expected response time, and crucially, the time when the generator was installed and started operating.

In this paper we focus our attention to three strategies that are most widely used in current Active Network Management (ANM) schemes in the UK [19, 21, 24]: last-in-first out (LIFO), Pro Rata (or proportional) and Rota-based schemes. In *LIFO*-based curtailment, generators are curtailed based on the inverse order in which they were granted the right to connect to the distribution network, hence giving a clear preference to *early connections*. By contrast, *Pro Rata* shares curtailment equally among installed generators proportionally to the rated capacity or actual power output, at the time of curtailment. *Rota* curtails generators at a rotational basis or a predetermined rota, as specified by the system operator.

The effects and operation of these curtailment mechanisms are best illustrated with an example. Consider a network with no export capability, comprising three wind generators of unequal rated capacities, e.g. $P_{N_1} = 3\ MW$, $P_{N_2} = 5\ MW$ and $P_{N_3} = 2\ MW$, where the subscript denotes the chronological order of their connection to the power grid. Moreover, for example simplicity, let's assume the actual power output at time interval t, equals their rated power i.e. $P_{G,t} = 10\ MW$, while the total demand at time t is $P_{D,t} = 7\ MW$. Given that excess energy cannot be exported, a total of $P_{C,t} = P_{G,t} - P_{D,t} = 3\ MW$ needs to be curtailed. With LIFO, the third generator is completely curtailed, the second is curtailed by $1\ MW$ and the first is not affected. By contrast, with Pro Rata the required curtailment is distributed proportionally among the generators, as $P_{C_i,t} = \frac{P_{C,t}}{P_{G,t}} P_{G_i,t}$. This results in $0.9\ MW$, $1.5\ MW$ and $0.6\ MW$ curtailed from the generator 1, 2 and 3 respectively.

If Pro Rata curtailment is not always desirable, we propose an equivalent Rota-type strategy with the same *fairness* properties, a strategy we call *Fractional Round Robin* (FRR). Technically speaking, Pro Rata curtailment may require special wind turbines with adjustable blades, such that their output can be reduced as needed. These do exist in newer wind farms, but may be more expensive. With FRR, the power curtailed is distributed sequentially on a rotation basis, according to the number of rated capacity units installed, so that larger generators are chosen proportionally more times, in direct relation to their size. This means for instance that, on average, every 10 times a curtailment of $3\ MW$ is needed, the first generator will be curtailed 3 times, the second 5 times and the third 2 times. It can be shown that for a sufficiently long period of time (several years lifetime of the wind turbine), the curtailment rate under FRR converges to the curtailement rate with Pro Rata.

The curtailment schemes applied may affect the viability of existing and future renewable investments. Prior works discuss these implications, such as in [8], where it is shown that LIFO leads to lower capacity factor (CF) for *late* connections, when compared to Pro Rata, and therefore might discourage new investment. A related study observes that the most important factor in the decision-making procedure of a new investor, especially if a LIFO scheme is used, is the CF of the last generator connected [2]. By shutting out newer entrants, a LIFO scheme essentially leads to unexploited network capacity.

We illustrate these effects with a simulation process over the course of one year, which determines the impact of different schemes on the CF, a widely used parameter in electrical engineering. CF is equal to the ratio of the actual energy generated to the maximum energy it could be generated, if operating under nominal conditions. Note that, the CF of a typical wind turbine (even if output is never curtailed) depends on wind conditions at the site's location and has a typical value of about 30% in the UK.

The one-year simulation process performs $8,760$ hourly iterations. Every hour, a curtailment event is decided on, with a probability of curtailment p_c (here, for simplicity we have $p_c = 0.2$, but in practice this parameter would wary on the network congestion conditions of a specific location). For time intervals when a curtailment event occurs, the level

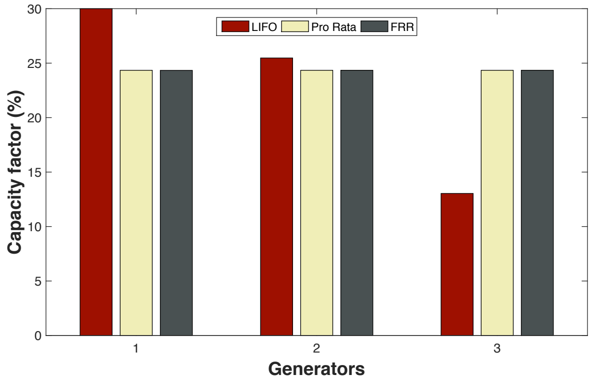


Figure 1: Curtailment mechanisms effects on the CF of wind generators under LIFO, Pro Rata and FRR

of total curtailment is determined probabilistically among four levels: of $P_C = 1.5\ MW$, $3\ MW$, $4.5\ MW$ and lastly $P_C = 6\ MW$, with probabilities of $40\%, 30\%, 20\%$ and 10%, respectively (these numbers follow a realistic distribution, as small curtailment events are much more frequent in practice). Finally, according to the applied PoA, the curtailment is allocated to the three generators.

Results are shown in Figure 1. First, note that under the LIFO scheme, generator 1 is not affected, whereas generators 2 and 3 experience severe CF reduction of 15.10% and 56.53%, respectively. CF is affected equally for all generators at 24.34% with Pro Rata, and similar results are derived for FRR, as expected. Results show, there is a clear market advantage of early connections when LIFO is applied, which is detrimental both for the economic performance of late generators and future investment. With Pro Rata and FRR, CF will continue to decrease if other plants are built, therefore, the commercial and legal clauses of access to the distribution network need to be addressed.

In the next section, we formalise the effects of curtailment to generation capacity investments and determine, given a location with network congestion, an upper level of tolerable curtailment, which enables renewable capacity investment to be profitable.

4. RENEWABLE INVESTMENT IN SINGLE LOCATIONS

Consider n RES generators at a single location of the distribution network with network constraints. Generator i is expected to produce $\mathsf{E}(E_{G_i,t})$ energy units in a time interval of length t, according to the available resource on the site location, without curtailment. Note that, while for a particular time period, such as an hour or a day, the expected generation is uncertain, for the overall lifetime of a RES project, it can be estimated with relatively high certainty from the weather and wind patterns at a particular location. Hence, the term $\mathsf{E}()$ which denotes 'expected', will be omitted from this point on, in order to simplify the notation. The cost of expected generation per unit c_{G_i}, with constant depreciation, is defined as:

$$c_{G_i} = \frac{I_{G,i} + M_{G_i}}{E_{G_i}} \tag{1}$$

where I_{G_i} the cost of building the plant (or initial investment), M_{G_i} the cost of operation and maintenance and E_{G_i} the expected generation, throughout the project lifetime (when referring to the project lifetime, we omit the subscript t).

As this is the case in UK and other countries [14], the energy generated by the RES unit is usually sold at a constant feed-in-tariff price, p_G.

Suppose that, generation curtailment is imposed in the region due to network constraints, where expected curtailed energy units are E_{C_i}. We will define a new parameter, useful for the analysis, which quantifies the curtailment imposed to each generator. We define the curtailment rate of i generator CR_i, as the ratio of expected curtailment to expected generation, over the project lifetime:

$$CR_i = \frac{E_{C_i}}{E_{G_i}} \tag{2}$$

Obviously, $0 \leq CR_i < 1$ and can be directly interpreted to CF reduction, e.g. $CR_i = 5\%$ results in a 5% CF reduction. As shown below, CR_i is a critical parameter which determines the viability of existing and future RES investment.

LEMMA 1. *A generation capacity investment is viable, iff the curtailment rate of i generator CR_i is smaller or equal to a threshold τ_G, $CR_i \leq \tau_G$, where $\tau_G = 1 - c_G/p_G$.*

PROOF. The profit Π of generator i with curtailment, is given by $\Pi_i = (E_{G_i} - E_{C_i}) \cdot p_G - E_{G_i} \cdot c_{G_i} \geq 0$. Dividing the profit function by $E_{G_i} \cdot c_{G_i}$ and then substituting CR_i by Eq.2 gives $CR_i \leq 1 - \frac{c_{G_i}}{p_G}$. We can assume this threshold is location-specific: all generators at this location have approximately the same access to land and required technology. If this is not true, τ_G would depend on the costs c_{G_i} of each generator. □

Hence, the profitability of a renewable development depends on the relation of the generation cost with the FIT price. The smaller the cost ratio, larger amounts of curtailment can be tolerated by the power producers. However, the rules of curtailment remain under the control of the mechanism designer (local DNO or regulator). Hence, a natural question to ask is *which curtailment rule maximises the local generation capacity?* This question is of interest in itself, but it also plays a role in modeling investment decisions of new transmission.

Consider a perfectly competitive setting, in which all generators are price takers, meaning no agent has the market power to influence the price equilibrium. Using assumptions of competitive equilibrium (Cournot) analysis, any investor will be able to install an additional generation unit, as long as its marginal profit exceeds its marginal cost. As discussed above, renewable resources and technology are roughly equal for all investors, so we can assume marginal costs are the same and the decision to invest is taken if the CR_i does not exceed a certain location-dependent threshold τ_G. Given these assumptions, the curtailment rule selected by the system operator, can impact the total generation capacity installed.

LEMMA 2. *In a perfectly competitive equilibrium setting, the local generation capacity installed is maximised under proportional curtailment strategies.*

PROOF. (Sketch) The problem of maximising the generation capacity installed is equivalent to maximising the total energy generated at a single location and it can be formulated as the optimisation problem $\max \left(\sum_{i=1}^{n} E_{G_i} \right)$, subject to a set of n constraints $E_{G_i} \leq \frac{E_{D_i}}{1-\tau_G}, \forall i = 1 \ldots n$, as derived from Lemma 1 and $E_{C_i} = E_{G_i} - E_{D_i}$ (curtailed energy units E_{C_i} equal the energy it could be generated minus the amount required or demand E_{D_i}). Each of the n constraints is satisfied when all curtailment rates CR_i of the generators are equal to each other and to the threshold τ_G, $\frac{E_{C_1}}{E_{G_1}} = ... = \frac{E_{C_i}}{E_{G_i}} = ... = \frac{E_{C_n}}{E_{G_n}} = \tau_G$. Essentially, this condition will be satisfied by proportional or *fair* curtailment strategies, i.e. those mechanisms which divide existing demand, hence curtailment, equally across all participants. □

5. TRANSMISSION INVESTMENT IN MULTIPLE LOCATIONS

In this section, we turn our attention to the central problem of this paper, namely how the applied PoA influences the decision to build or reinforce transmission lines.

In our analysis, we consider two locations: A is a net consumer (demand exceeds supply, e.g. a mainland location with industry or significant population density) and B is a net energy producer (favourable RES conditions, e.g. a remote region rich in wind resource). In practice, there would be some local demand and supply, considered here negligible, and installation of new RES capacity would not be feasible without network upgrade. Location A has a net demand of $E_{D,A}$, equal to local generation minus local demand.

Moreover, we consider two players: a player who is the line investor, who can be merchant-type or a utility company and is building the $A - B$ interconnection (and, possibly, additional renewable generation capacity at B, $E_{G_1,B}$), and a local player, who represents the other renewable generators or investors located at B, $E_{G_2,B}$ (this second player can be thought of as the local community). Note that in Scotland, but also in several other countries such as Denmark, local groups often group together to make land available and invest in local renewable generation projects[1].

The power line cost is estimated over the project lifetime:

$$C_T = I_T + M_T \tag{3}$$

where I_T the cost of building the line (or initial investment) and M_T the cost of operation and maintenance. (Note that, C_T refers to the total costs over the project lifetime, while c_G in Eq. 1 refers to the cost per energy unit). The monetary value of the transmission line is proportional to the energy flowing from B to A, charged under common access rules, with p_T transmission fee per energy unit.

Next, we consider two separate models for the multi-location problem, distinguished by how strategically the local investors respond to the actions of the utility company (i.e. the line player). In the first model (Sect. 5.1), we formalise the decision of the line investor to build the line, but assuming the local players do not react to this line being built, by installing additional capacity themselves. In the second

[1]In Scotland, Community Energy Scotland (CES) is an umbrella organisation representing the interests of such groups.

model, local investors can and do react to this extra line capacity being built. Hence, the line investor has to account for this reaction of the other generator when building the line, leading to a Stackelberg game (Sect. 5.2).

5.1 Implementation in areas with high curtailment

Driven by favourable renewable conditions, we assume several investors have installed considerable volumes of renewable capacity $E_{G_2,B}$ at B, which is required to be curtailed by $E_{C_2,B}$ energy units. Local generators at B act in a perfectly competitive setting or Cournot equilibrium.

A new transmission line will export the renewable energy installed by the line investor $E_{G_1,B}$, but will also take advantage of the curtailed energy by local generators. Similarly to the previous analysis, we ask: *How does the curtailment rate affect the viability of the transmission line investment?*

LEMMA 3. *A transmission capacity investment is viable iff the curtailment rate of local generators $CR_{2,B}$ (before the line is built) is greater or equal to a threshold τ_T*

$$CR_{2,B} \geq \tau_T \tag{4}$$

where

$$\tau_T = \frac{C_T - E_{D,A} \cdot (p_G - c_{G_1})}{c_{G_1} \cdot E_{G_2,B}} \tag{5}$$

PROOF. The line investor has two streams of revenue, the curtailed energy produced by local generators and the energy generated from additional capacity, which lead to a profit function of

$$\Pi_1 = E_{C_2,B} \cdot p_T + E_{G_1,B} \cdot (p_G - c_{G_1}) - C_T \geq 0$$

The latter combined with the capacity of the line $E_{D,A} = E_{G_1,B} + E_{C_2,B}$ and divided by the expected energy from local generators $E_{G_2,B}$, results in the curtailment rate of local producers at B:

$$CR_{2,B} \geq \frac{C_T - E_{D,A} \cdot (p_G - c_{G_1})}{(p_T - p_G + c_{G_1}) \cdot E_{G_2,B}} \tag{6}$$

For local generators, the curtailed energy before the line is built is essentially wasted. Recall that (see Lemma 1), local generator investments are still profitable, as long as $CR_i \leq \tau_G$. This means the line investor can actually impose a large transmission fee, which approaches the FIT price $p_T \to p_G$, in order to maximise his own profits. Considering this, from Eq. 6 we derive to the desired conclusion. □

We assumed here the local investors are not players who have the possibility to react by increasing their own generation once the new line gets built. In practice, they do react, hence the decision of the line investor when building the line, must include an element of *strategic foresight* to include the reaction of local investors. This model is essentially a Stackelberg game, and is examined in the next section.

5.2 Transmission investment as a Stackelberg game

Here, we determine the equilibrium strategies of the line and local investors, who are able to react to the new line being built. For simplicity, we assume there is no renewable capacity installed at location B prior to the construction of the transmission line. The decision of building the power line will elicit a reaction from other investors. However, crucially the line investor has a *first mover* advantage, as only he can build the grid infrastructure, which is expensive and technically challenging and only a limited set of investors (such as DNO-approved or DNOs themselves), have the technical expertise and regulatory approval to carry it out.

The line investor (leader) can assess and evaluate the reaction of other investors to determine his strategy, namely the capacity of the power line and the level of renewable capacity to be installed, aiming to influence the equilibrium price. Other investors (followers) can only act after observing the leader's strategy. In practice, the leader can be thought of as a major utility company developing a wind investment project, while the followers can be thought of as the local community on the island, who do not have the technical/financial capacity to build a line, but may have access to cheaper land, it is easier to get community permission to build turbines etc., hence may have a lower per-unit generation cost. This two-stage process is analysed as a Stackelberg game [27].

Equilibrium is found by backward induction. First of all, the leader estimates the best response of local generators, given its own output and then decides his strategy with profit maximisation criteria. The renewable generation capacity installed by the follower will be a function of the generation capacity installed by the line investor. At a second stage, the follower observes this strategy and decides his generation capacity, according to his best response, i.e. maximising his own profit, as anticipated and predicted by the leader. The solution of this process is the Stackelberg equilibrium of the transmission investment game. The network access arrangements play here a crucial role for the market equilibrium formed. Next, we examine the effects of LIFO and Pro Rata/FRR strategies.

5.2.1 LIFO scheme

LEMMA 4. *The transmission investment game between the line investor and local generators with LIFO curtailment results in the expected generation at Stackelberg equilibrium:*

$$E^*_{G_1,B} = E_{D_A} \tag{7}$$

$$E^*_{G_2,B} = 0 \tag{8}$$

and associated profits

$$\Pi_1{}^* = (p_G - c_{G_1}) \cdot E_{D_A} \tag{9}$$

$$\Pi_2{}^* = 0 \tag{10}$$

PROOF. The capacity of the transmission line is bound by the demand at mainland, therefore total generation capacity at location B, $(E_{G_{1,B}} + E_{G_{2,B}})$ cannot exceed E_{D_A}. Any generation capacity built exceeding the demanded energy, has to be curtailed. Taking this into account, the profit functions of the two players are

$$\begin{aligned}\Pi_1 &= p_T \cdot E_{D_A} + (p_G - p_T - c_{G_1}) \cdot E_{G_{1,B}} - C_T \\ \Pi_2 &= (E_{D_A} - E_{G_{1,B}}) \cdot (p_G - p_T - c_{G_2})\end{aligned}$$

Clearly, under a LIFO scheme the line investor (who acts first) is protected from any curtailment, hence it can build all generation capacity to cover demand E_{D_A} itself and maximise its profits. The local investors would take all curtailment, as they represent "late" connections and have low priority, thus there is no incentive for them to invest in new capacity. □

To conclude, LIFO always protects the line investor, giving it absolute advantage.

5.2.2 Pro Rata or FRR scheme

The main difference from LIFO, is that Pro Rata rules are imposed to all generators, regardless of their order of connection. Therefore, more total capacity $E_{G,B} = E_{G_1,B} + E_{G_2,B}$ than the energy demanded at A can potentially be installed, as long as the curtailment rate or energy curtailed $E_{C,B} = E_{G,B} - E_{D,A}$ allows the investments to be profitable. The curtailment rate at location B is given by

$$CR_B = 1 - \frac{E_{D,A}}{E_{G_1,B} + E_{G_2,B}} \tag{11}$$

Using the curtailment rate from Eq. 11, the general profit functions of the players, which are functions of both players energy outputs, i.e. $\Pi(E_{G_1,B}, E_{G_2,B})$, can be written as:

$$\begin{aligned}\Pi_1 \quad &= \quad \left(\frac{p_G \cdot E_{D,A}}{E_{G_1,B} + E_{G_2,B}} - c_{G_1}\right) \cdot E_{G_1,B} \\ &+ \quad \frac{p_T \cdot E_{D,A}}{E_{G_1,B} + E_{G_2,B}} \cdot E_{G_2,B} - C_T \end{aligned}\tag{12}$$

$$\Pi_2 = \left[\frac{(p_G - p_T) \cdot E_{D,A}}{E_{G_1,B} + E_{G_2,B}} - c_{G_2}\right] \cdot E_{G_2,B} \tag{13}$$

Before stating our main Stackelberg equilibrium result, we need to define the players' best responses.

PROPOSITION 1. *Given the output of the leader $E_{G_1,B}$, the best response of the follower which maximises his profit is*

$$E^*_{G_2,B} = \sqrt{\frac{(p_G - p_T) \cdot E_{D,A} \cdot E_{G_1,B}}{c_{G_2}}} - E_{G_1,B} \tag{14}$$

PROOF. Let the value of $E_{G_2,B}$ which maximises the profit of the follower be ${E_{G_2,B}}^* = \underset{E_{G_2,B}}{argmax}\ \Pi_2$. Setting as zero the partial derivative of Π_2 in Eq. 13, with respect to $E_{G_2,B}$ and rearranging, we get Eq. 14. $\square$

PROPOSITION 2. *Given the output of the follower $E^*_{G_2,B}$, the best (i.e. profit-maximising) response of the leader is:*

$$E^*_{G_1,B} = \frac{(p_G - p_T) \cdot c_{G_2} \cdot E_{D,A}}{4 \cdot {c_{G_1}}^2}$$

PROOF. Let the value of $E_{G_1,B}$ which maximises the profit of the follower be ${E_{G_1,B}}^* = \underset{E_{G_1,B}}{argmax}\ \Pi_1$. Substituting Eq. 14 in Eq. 12 and then setting as zero the partial derivative of Π_1 with respect to $E_{G_1,B}$ gives the stated expression. $\square$

LEMMA 5. *The transmission investment game between the line investor and local generators with Pro Rata, results in expected generation at Stackelberg equilibrium:*

$$E^*_{G_1,B} = \frac{(p_G - p_T) \cdot c_{G_2} \cdot E_{D,A}}{4 \cdot {c_{G_1}}^2} \tag{15}$$

$$E^*_{G_2,B} = \frac{(p_G - p_T) \cdot (2 \cdot c_{G_1} - c_{G_2}) \cdot E_{D,A}}{4 \cdot {c_{G_1}}^2} \tag{16}$$

and associated profits

$${\Pi_1}^* = \frac{(p_G - p_T) \cdot c_{G_2} \cdot E_{D,A}}{4 \cdot c_{G_1}} + p_T \cdot E_{D,A} - C_T \tag{17}$$

$${\Pi_2}^* = \frac{(2 \cdot c_{G_1} - c_{G_2})^2 \cdot (p_G - p_T) \cdot E_{D,A}}{4 \cdot c^2_{G_1}} \tag{18}$$

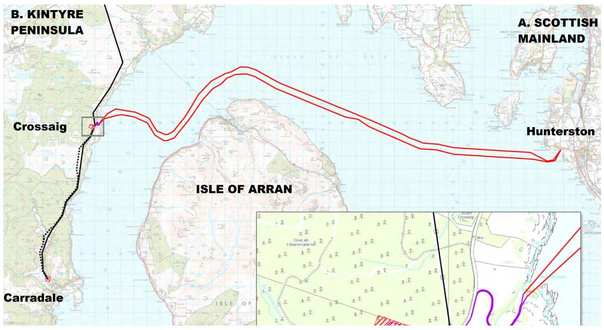


Figure 2: Case-study map: Power line connects mainland (high demand) to Kintyre peninsula (high RES) [20]

PROOF. Replacing Prop. 2 in Eq. 14, the optimum output of local generators $E^*_{G_2,B}$ is found, i.e. Eq. 16. Finally, substituting the energy outputs at equilibrium (Eq. 15 and Eq. 16) in Eq. 12 and Eq. 13, we derive the equilibrium profits ${\Pi_1}^* = max\ \Pi_1$ and ${\Pi_2}^* = max\ \Pi_2$. $\square$

By adding up Eq. 15 and Eq. 16, we see the total generation installed at B depends on the energy demand,the transmission fee and the line investor's generation cost, as:

$${E^*_{G_B}}^* = E^*_{G_1,B} + E^*_{G_2,B} = \frac{(p_G - p_T)E_{D,A}}{2c_{G_1}} \tag{19}$$

Finally note that a curtailment scheme is required if and only if total generation capacity exceeds the net demand at A, $E_{G_1,B} + E_{G_2,B} > E_{D,A}$ (otherwise there is no strategic interaction and no game, as both players can sell all their generated power). This constraint yields the following conditions, which must hold for the setting to actually be game-theoretic (and for our analysis to be relevant):

$$c_{G_2} < p_G - p_T \tag{20}$$

$$c_{G_1} < \frac{p_G - p_T}{2} \tag{21}$$

6. NETWORK UPGRADE CASE STUDY

In this section, we apply the theoretical framework of the Stackelberg model with Pro Rata (c.f. Lemma 5) to the concrete case-study of Kintyre-Hunterston grid reinforcement project, currently under development in the UK.

Grid infrastructure in the Kintyre peninsula was originally designed and built to serve a typical rural area of low demand. Wind energy promotion and its rapid growth through substantial incentives, quickly led to high volumes of renewable investment in the region. RES capacity is expected to reach 454 MW, by the end of 2015, and future connections estimations exceed 793 MW. The necessity for large transmission investment soon became apparent, therefore SSE proceeded in a £230m network upgrade project connecting existing Hunterston substation, partially through a subsea link, to Crossaig, thus creating headroom for additional 150 MW renewable capacity [20], estimated to provide a net lifetime benefit of £520m [23].

Based on these project figures, we consider a simplified two-node network, in which the mainland energy demand

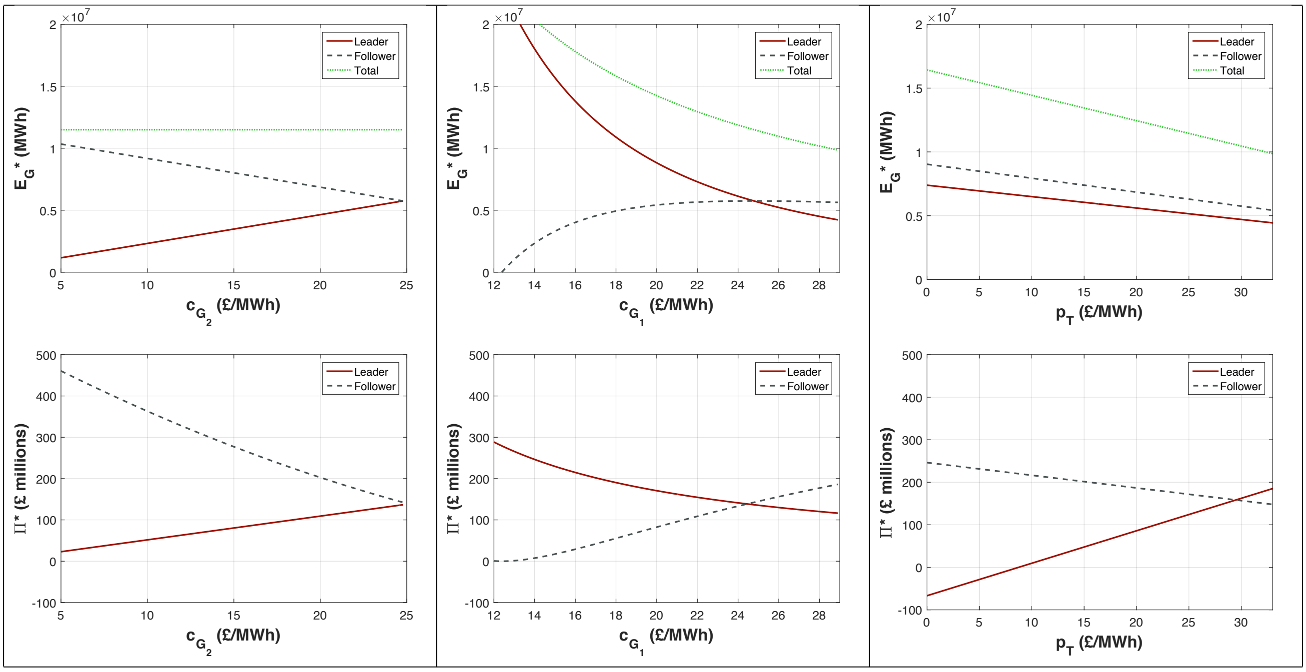


Figure 3: Rows (1) and (2) show generation capacity built and profits at Stackelberg equilibrium, respectively, column (1) shows dependency on generation cost of local generators, (2) on generation cost of line investor and (3) on transmission fee

being met by generation in Kintyre equals the energy transmitted through the power line. With the majority of investment being wind projects, we estimate the total energy demand as $E_{D,A} = 9,855,000\ MWh$ for 25 years project lifetime. As currently valid in UK for medium size wind projects, the FIT price was set to $p_G = £82.60/MWh$ [14].

In Figure 3, we summarise the results of our model, namely the generation capacity built and associated profits at Stackelberg equilibrium, for three scenarios: *Scenario 1* corresponds to the plots of the first column and shows the effect of varying the local investors' generation cost, keeping all other parameters at constant values (set as $c_{G_1} = p_T = 0.3p_G$ and $c_{G_2} = 0 \ldots c_{G_1}$), *Scenario 2* in the second column shows the effect of varying the line investor's generation cost (with settings: $c_{G_2} = p_T = 0.3p_G$ and $c_{G_1} = 0.125p_G \ldots 0.35p_G$) and *Scenario 3* in the third column shows the effect of varying the transmission fee (with other parameters set at $c_{G_1} = 0.3p_G$, $c_{G_2} = 0.9c_{G_1}$ and $p_T = 0 \ldots 0.4p_G$). For each scenario, the range of the "free" parameter (fixing the others) is determined from the constraints in Eq. 20 and Eq. 21.

Given a certain FIT (this parameter is controlled by the regulatory authority), line investment feasibility depends directly on the generation cost c_{G_1} and transmission fee p_T. If the line is built, it sets up a level of total feasible generation investment at B, which combined with Pro Rata access rule, leads to larger volumes of capacity being built than actual demand (see 1st row of Fig. 3 and Eq. 19), as long as the curtailment rate is kept under reasonable levels. Note this total level of generation does *not* depend on the generation costs of local investors, since they cannot act without the existence of the line (see Fig. 3 1st row, 1st column). For all settings, the size of c_{G_2} relative to c_{G_1} determines how exportable level of generation capacity is shared. Cheaper generation has an advantage in all 3 sets of results (c.f. Fig. 3 first row), although as the graphs show, the dependency is not necessarily linear.

Another conclusion is that transmission charges, agreed by the line investor and an independent regulatory authority, has to be set within a specific range. Low values of p_T may lead to transmission investment being aborted, somewhat larger values might theoretically be sufficient to achieve profitability for the line investor, however, hide the risk of "free-riding" from local investors, who pay cheap for the benefits offered by the leader's costly investment. What the result in Fig. 3 (row 3, col. 3) shows is that there exists a range in which p_T can be set such as to assure the line gets built (i.e. when the leader's profits are above 0 – in our case, transmission charges need to be at least $£0.08/kWh$), but also not discourage other local renewable investors.

7. CONCLUSIONS & FUTURE WORK

To our knowledge, this is the first work examining the combined effects of curtailment strategies and transmission access rules on generation capacity investment and network expansion, which is essential aspect for RES integration in practice. Our research focused on the effects of leading curtailment schemes to investment decisions and market behaviour. We model grid reinforcement investment as a two-stage strategic game between the line investor and local generators and determined generation capacities and profits at equilibrium. Based on a UK grid reinforcement project, we proposed a tool to calculate transmission charges, under "common access" rules, which enables the implementation of both transmission and local generation investments.

Future work includes expanding the developed two-location model to more complex settings, in order to model the transmission investment game across multiple locations, such as in the Orkney Islands in NE Scotland. A more detailed model will require the incorporation of energy storage facilities co-located with renewable generation, which are capable of partially deferring curtailment.

Acknowledgments

The authors would like to thank Community Energy Scotland and SSE for all the information provided for this study and the Scottish government for their financial support through a Local Energy Scotland Challenge grant.

REFERENCES

[1] G. E. Asimakopoulou, A. L. Dimeas, and N. D. Hatziargyriou. Leader-follower strategies for energy management of multi-microgrids. *IEEE Transactions on Smart Grid*, 4(4):1909–1916, December 2013.

[2] Baringa Partners and UK Power Networks. Flexible Plug and Play Principles of Access Report. Technical report, December 2012.

[3] L. Baringo and A. J. Conejo. Transmission and wind power investment. *IEEE Transactions on Power Systems*, 27(2):885–893, May 2012.

[4] G. Chalkiadakis, V. Robu, R. Kota, A. Rogers, and N. R. Jennings. Cooperatives of distributed energy resources for efficient virtual power plants. In *10th International conference on Autonomous Agents and Multi-Agent Systems*, pages 787–794, Taipei, May 2011. AAMAS.

[5] R. Currie, B. O'Neill, C. Foote, A. Gooding, R. Ferris, and J. Douglas. Commercial arrangements to facilitate active network management. In *21st International Conference on Electricity Distribution*, Frankfurt, June 2011. CIRED.

[6] P. Joskow and J. Tirole. Transmission rights and market power on electric power networks. *RAND Journal of Economics*, 31(3):450–487, Autumn 2000.

[7] P. Joskow and J. Tirole. Merchant transmission investment. *The Journal of Industrial Economics*, 53(2):233–264, June 2005.

[8] L. Kane and G. Ault. Evaluation of wind power curtailment in active network management schemes. *IEEE Transactions on Power Systems*, 30(2):672–679, 2015.

[9] A. Laguna Estopier, E. Crosthwaite Eyre, S. Georgiopoulos, and C. Marantes. FPP low carbon networks: Commercial solutions for active network management. In *CIRED*, Stockholm, 2013.

[10] J. Lee, J. Guo, J. K. Choi, and M. Zukerman. Distributed energy trading in microgrids: A game theoretic model and its equilibrium analysis. *IEEE Transactions on Industrial Electronics*, 62(6):3524–3533, June 2015.

[11] Y. Li and V. Conitzer. Game-theoretic question selection for tests. In *23rd International Joint Conference on Artificial Intelligence*, pages 254–262, Beijing, August 2013. IJCAI.

[12] L. Maurovich-Horvat, T. K. Boomsma, and A. S. Siddiqui. Transmission and wind investment in a deregulated electricity industry. *IEEE Transactions on Power Systems*, 30(3):1633–1643, May 2015.

[13] A. Motamedi, H. Zareipour, M. O. Buygi, and W. D. Rosehart. A transmission planning framework considering future generation expansions in electricity markets. *IEEE Transactions on Power Systems*, 25(4):1987–1995, November 2010.

[14] Office of Gas and Electricity Markets (Ofgem). Tariff tables, 2015.

[15] P. Paruchuri, J. P. Pearce, J. Marecki, M. Tambe, F. Ordonez, and S. Kraus. Efficient algorithms to solve Bayesian Stackelberg games for security applications. In *23rd Conference on Artificial Intelligence*, pages 1559–1562, Chicago, July 2008. AAAI.

[16] A. Perrault and C. Boutilier. Efficient coordinated power distribution on private infrastructure. In *International conference on Autonomous Agents and Multi-Agent Systems*, pages 805–812, Paris, May 2014. AAMAS.

[17] S. Ramchurn, P. Vytelingum, A. Rogers, and N. R. Jennings. Putting the smarts into the smart grid: A grand challenge for artificial intelligence. *Communications of the ACM*, 55(4):86–97, April 2012.

[18] E. E. Sauma and S. S. Oren. Proactive planning and valuation of transmission investments in restructured electricity markets. *Journal of Regulatory Economics*, 30(3):261–290, September 2006.

[19] Scottish and Southern Energy Power Distribution (SSE). Britains first smart grid, 2012.

[20] Scottish and Southern Energy Power Distribution (SSE). Kintyre-Hunterston, 2015.

[21] Scottish and Southern Energy Power Distribution (SSE). Northern Isles New Energy Solutions (NINES), 2015.

[22] G. B. Shrestha and P. A. J. Fonseka. Congestion-driven transmission expansion in competitive power markets. *IEEE Transactions on Power Systems*, 19(3):1658–1665, August 2004.

[23] Sinclair Knight Merz (SKM). Kintyre-Hunterston 132kV transmission network reinforcement cost benefit analysis. Technical report, January 2013.

[24] UK Power Networks. Innovation Flexible Plug and Play, 2015.

[25] A. H. van der Weijde and B. F. Hobbs. The economics of planning electricity transmission to accommodate renewables: Using two-stage optimisation to evaluate flexibility and the cost of disregarding uncertainty. *Energy Economics*, 34(6):2089–2101, February 2012.

[26] M. Vasirani, R. Kota, R. L. Cavalcante, S. Ossowski, and N. R. Jennings. An agent-based approach to virtual power plants of wind power generators and electric vehicles. *IEEE Transactions on Smart Grid*, 4(3):1909–1916, September 2013.

[27] H. von Stackelberg. *Market structure and equilibrium*. Springer Science & Business Media, 2010.

[28] R. Zheng, Y. Xu, N. Chakraborty, and K. Sycara. A crowdfunding model for green energy investment. In *24th International Joint Conference on Artificial Intelligence*, pages 2669–2675, Buenos Aires, July 2015. IJCAI.

Dynamic Voting for Agent-Mediated Negotiations in Non-linear Utility Spaces

Miguel A. Lopez-Carmona and Ivan Marsa-Maestre
Computer Engineering Department
University of Alcala
28871 Alcala de Henares (Madrid), Spain
miguelangel.lopez@uah.es
ivan.marsa@uah.es

ABSTRACT

Making group decisions or negotiating contracts with interdependent issues and conflicting interests may be challenging because of strategic interactions and incomplete information. Agent-mediated negotiation is a computational tool which can make group decision making much more efficient. Nevertheless, efficient mechanisms for making proposals, expressing and aggregating preferences are needed. Simple voting mechanisms have been successfully used to capture and aggregate the agents' preferences, but their performance decreases in complex negotiation scenarios. In this paper, an agent-mediated negotiation protocol architecture is proposed, which is inspired by general heuristic optimization algorithms for centralized problems. As part of the negotiation protocol architecture, we present and evaluate different mechanisms for proposal generation, aggregation of preferences and voting. The empirical evaluation shows that adaptive voting mechanisms and ordered weighted aggregation operators can efficiently achieve beneficial solutions, outperforming previous approaches. Our architecture and the proposed mechanisms can facilitate the development of efficient group decision making systems in diverse scenarios.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence—*Multiagent Systems, cooperation and coordination*

Keywords

multiagent negotiation, mediated negotation, strategy-proof, cooperation

1. INTRODUCTION

Distributed decision making is a prevalent task in application areas like corporate operational planning, supply chain management, service operations or e-commerce [20, 23, 18, 21]. A lot of decision making problems in these scenarios cannot be optimally solved due to computing power or time constraints. Metaheuristics are higher-level procedures designed to select a heuristic (partial search algorithm) that may provide a sufficiently good solution [4, 3]. However, the competition of the participant entities causes strategic interactions as well as incomplete information, and consequently, centralized optimization may not be feasible.

Agent-mediated negotiation emerges as a computational tool which may improve collective gains, and overcome the challenges of competing negotiators [25, 16, 7]. At the heart of fair and efficient negotiations are the strategies to express and aggregate the participants' preferences, and the procedures to explore efficiently the contract (solution or agreement) space [15, 12, 14].

With the goal of improving collective gains, new mechanisms for searching contracts, aggregating preferences and voting are needed that favor cooperative decision strategies and improve the performance of the exploration of alternatives. We propose a general negotiation protocol architecture which is designed to easily integrate any heuristic search algorithm. The architecture is mainly suited to integrate improvement-based single negotiation text procedures, in which a contract or a set of contracts are iteratively refined looking for Pareto improvements. Our main contribution is on the definition and evaluation of adaptive techniques to implement mechanisms for proposal generation and voting, and on the formalization of the aggregation functions by means of ordered weighted average operators (OWA) [27]. The aim of our work is to perform a comprehensive investigation of the effect on social welfare of using adaptive mechanisms and aggregation operators in heuristic search based negotiation protocols.

The results are encouraging. Our solutions outperform previous static approaches in the same negotiation scenarios and under the same requirements. In addition, the proposed framework can be easily adapted to integrate new heuristic search algorithms and voting systems. Finally, the use of OWA operators allows us to easily meet different social welfare criteria.

After the problem description, in Section 3 we present a negotiation protocol architecture, and in Section 4 an instance of the protocol architecture is devised. In Section 5 the proposal is empirically evaluated, and compared to the most closely-related works in Section 6. The last section summarizes our conclusions and discusses future research.

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

2. PROBLEM DESCRIPTION

A mediated multi-issue negotiation is a dialogue between a set of agents $\mathcal{A} = \{a_1, \ldots, a_A\}$ and a mediator agent who try to find an agreement about a set of issues defined as a finite set of variables $X = \{x_i | i = 1, \ldots, I\}$, where $\mathcal{D}_i$ is the domain of issue x_i. A *contract* is a vector $c = \{x_i^c | i = 1, \ldots, I\}$ defined by the issues' values, and $C \subseteq \mathcal{D}_1 \times \cdots \times \mathcal{D}_I$ is the *contract space*.

To assess the overall outcome of a negotiation, it is useful to compute some measure of social welfare. Thus, an agent a_j owns a *cardinal utility function* $U_j(c) : C \to \mathbb{R}$ and a *vote function* $V_j(c) : C \to \mathbb{R}$. The utility function $U_j(c)$ maps each element of the contract space to a monetary value. Additionally, $V_j(c) : C \to \mathbb{R}$ states the degree to which a_j *supports* c. Note that U_j is a private function stating the reward achieved by the agent, while V_j reflects the information revealed to the mediator agent (i.e., the vote), which will obviously depend on the utility values.

The utility functions are transferable, i.e., agents can transfer any given amount of their utility to other agents by transferring a commodity. By assumming that agents' utilities are transferable, respective utility functions lead to cardinal values that can be used to measure a social welfare function that aggregates individual utilities [24].

The mediator agent is able to perform an aggregation of individual preferences onto single values to obtain a *group support.* Given a set of individual votes:

$$\overline{V}_{c_k} = \{V_1(c_k), \ldots, V_A(c_k)\},$$

the mediator associates a *group support* value $G(\overline{V}_{c_k}) : \overline{V} \to \mathbb{R}$ indicating the degree to which contract c_k is supported by the group of agents.

We assume that the issues are interdependent, i.e., the valuation $U_j(x_i^c)$ can vary depending on the other issues' decisions $x_k | k \neq i$. Furthermore, all preference information is private. To prevent strategic disadvantages, the preferences and the strategy considerations of the agents are private, i.e., they are not known to the other agents. Finally, we assume the agents act rationally, i.e., the aim to achieve the *individually* best outcome [2].

3. A NEGOTIATION PROTOCOL ARCHITECTURE

Our negotiation protocol architecture comprises three functional blocks:

1. **Proposal generation**: Generates a *set of contracts (proposal)* $\{c_1^t, \ldots, c_k^t, \ldots, c_{Z^t}^t\}$ at a given round t, which depends on the used heuristic search algorithm and the **group support function**. The number of contracts (**proposal size**) Z^t may vary over time.

2. **Agents' vote function**: Each agent a_j gives privately its vote $V_j(c_k^t)$.

3. **Group support function**: The mediator captures the agents' votes and computes the group support for each contract $\{G(\overline{V}_{c_1^t}), \ldots, G(\overline{V}_{c_{Z^t}^t})\}$.

This architecture may be subject to different configurations.

Proposal generation.

The mediator has to iteratively explore the contract space with the aim of satisfying the agents' preferences. However, the mediator has only partial information, the agents' votes are subject to endogenous variables. Basically, this is a multiobjective optimization problem with imperfect information, and metaheuristics are higher-level search procedures that may provide sufficiently good solutions in these scenarios [11].

In our architecture, the general strategy that guides the search process is to take the group support values, and then generate new contracts that are expected to optimize the group support —i.e., the social support criterion. Summarizing, the *proposal generation function* implies the *selection and adaptation of a metaheuristic* and the *evaluation of a group support function.*

Restriction of agents' vote function.

It ensures that agents' votes for a set of contracts follow a predefined set of rules. These rules comprise the *voting type* and the *acceptance quota* [1, 14, 17]. For instance, agents could be restricted to accept or reject votes. Based on the voting type, an *acceptance quota q* ensures that the set of agent's votes follow a predefined distribution. For example, we may determine a quota stating the percentage of the contracts that have to be accepted.

Group Support Function.

The group support function:

$$G(\overline{V}_{c_k^t}) = G(\{V_1(c_k^t), \ldots, V_A(c_k^t)\})$$

is used to aggregate the votes for a given contract and reflect a desired mediation imperative. In general, the protocol designer needs to implement mechanisms to compute the goodness of a contract in terms of a social criterion. Our proposal is to use *Ordered Weighted Averaging Operators* (OWA) [27].

An aggregation operator $M : V^A \to G, (V, G \in [0, 1])$ is called an OWA operator of dimension A, if it has an associated weighting vector $W = [w_1 w_2 \ldots w_A]$ such that $w_t \in [0, 1]$ and $\sum_{t=1}^{A} w_t = 1$ and where $M(V_1, \ldots, V_A) = \sum_{t=1}^{A} w_t b_t$, where b_t is the tth largest element of the aggregates $\{V_1, \ldots, V_A\}$. In the OWA aggregation the weights are not directly associated with a particular argument but with the ordered position of the arguments. If ind is an index function such that $ind(t)$ is the index of the tth largest argument, then we can express $M() = \sum_{t=1}^{A} w_t V_{ind(t)}$. The form of the aggregation is dependent upon the associated weighting vector. We have a number of special cases of weighting vectors. For instance, the vector W^* defined such that $w_1 = 1$ and $w_t = 0$ for all $t \neq 1$ gives us the aggregation $Max_i[V_i]$.

Our aim is to define aggregation policies in the form of a linguistic agenda. For example, the mediator should make decisions following mediation rules like *"Most* agents must be satisfied", *"at least* α agents must be satisfied", ... These statements are examples of *quantifier guided aggregations* [26]. Zadeh [28] suggested that any relative linguistic quantifier can be expressed as a fuzzy subset Q of the unit interval $I = [0, 1]$. In this representation for any proportion $y \in I$, $Q(y)$ indicates the degree to which y satisfies the concept expressed by the term Q. Formally, these quantifiers are characterized in the following way: 1) $Q(0) = 0$, 2) $Q(1) = 1$ and 3) $Q(x) \geq Q(y)$ if $x > y$. Examples of this kind of quantifier are *all*, *most*, *many*, *at least* α. An example of quantifiers is *all* which is represented by Q_* where

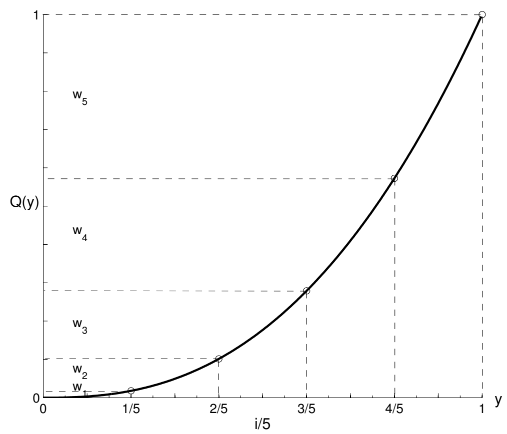


Figure 1: Example of how to obtain the weights from the quantifier for $A = 5$ agents.

$Q_*(1) = 1$ and $Q_*(x) = 0$ for all $x \neq 1$.

Under this approach a group support function is expressed in terms of a linguistic quantifier Q indicating the proportion of agents whose agreement is necessary for a solution to be socially acceptable. The formal procedure used to implement the mediation rule is as follows:

1. Use Q to generate a set of OWA weights, $w_1, \ldots, w_A$.
2. For each contract c_k use these weights to calculate $G(\overline{V}_{c_k})$.

The used procedure for generating the weights from the quantifier (see Figure 1) is to divide the unit interval into n equally spaced intervals and then to compute the length of the mapped intervals using Q:

$$w_t = Q(\frac{t}{A}) - Q(\frac{t-1}{A}) \text{ for } t = 1, \ldots, A\,.$$

Generally, any function $Q : [0,1] \to [0,1]$ such that $Q(x) \geq Q(y)$ for $x \geq y$, $Q(1) = 1$ and $Q(0) = 0$ can be seen to be an appropriate form for generating aggregation policies. For example, we could use a '*most agents satisfied*' quantifier defined by $Q(y) = x^2$. For three agents, the OWA weights would be obtained as follows: $w_1 = Q(1/3) - Q(0) = 1/9$, $w_2 = Q(2/3) - Q(1/3) = 1/3$, $w_3 = Q(1) - Q(2/3) = 8/9$. If, for instance, the support values for a contract c_k^t are $\{1, 0, 1\}$, then the group support value would be $G(c_k^t) = w_1 * 1 + w_2 * 1 + w_3 * 0 = 1/9 + 1/3 + 0 = 0.22$.

4. THE NEGOTIATION PROTOCOL

In this section an instance of the Negotiation Protocol Architecture is proposed.

At first, the mediator proposes an initial random **active contract** c_a^1 and an acceptance quota q^1. In the following, the **active contract** is defined as the last overall accepted contract. The negotiation procedure starts and is repeated until the T-th iteration (negotiation round). In every iteration t, $Z^t - 1$ single mutations to the current active contract c_a^t are computed to generate a proposal of length Z^t, i.e., one issue of the contract is altered. Also, in every iteration, the mediator determines a quota $q^t < Z^t$ stating the number of contracts for which agents have to give a vote. The general idea is to dynamically adapt both q^t and Z^t to the current conditions of the negotiation.

Afterwards, each agent a_j computes its support $V_j() \in [0,1]$ for each contract. The support values (votes) will depend on the utility functions $U_j()$, the agent's strategic behavior and on the vote restrictions (i.e., vote type).

Next step is for the mediator to compute the group support for each c_k^t: $G(\overline{V}_{c_k^t}) = M(V_1(c_k^t), \ldots, V_A(c_k^t))$, where M is an OWA operator. Finally, the contract with the highest group support is selected as the active contract in the next iteration. If there are several contracts, the mediator selects one randomly.

Thereafter, the process starts over and new proposals are generated using the new active contract. After T iterations, the last active contract becomes the **final agreement** of the negotiation.

$c_a^1 \leftarrow \mathit{InitialContract}()$
$Z^1 \leftarrow \mathit{InitialProposalLength}()$
$q^1 \leftarrow \mathit{InitialQuota}()$
for *t=1,...,T* **do** `/* Mediator */`
  $\mathit{Proposal} \leftarrow \{c_a^t\}$
  for *k=1,...,*Z^t **do**
    $c_k^t \leftarrow \mathit{Mutation}(c_a^t)$
    $\mathit{Proposal} \leftarrow \mathit{Proposal} \cup c_k^t$
  end
  forall the $c_k^t \in \mathit{Proposal}$ **do** `/* Agents */`
    $\overline{V}_{c_k^t} \leftarrow \emptyset$
    forall the $j \in \mathcal{A}$ **do**
      $\overline{V}_{c_k^t} \leftarrow V_j(U_j(c_k^t), q^t)$
    end
  end
  $\overline{G} \leftarrow \emptyset$
  forall the $c_k^t \in \mathit{Proposal}$ **do** `/* Mediator */`
    $\overline{G} \leftarrow G(\overline{V}_{c_k^t})$
  end
  $c_a^{t+1} \leftarrow \arg\max_{c^t} \overline{G}$
  $Z^{t+1} \leftarrow \mathit{UpdateProposalLength}(Z^t, \overline{G})$
  $q^{t+1} \leftarrow \mathit{UpdateQuota}(q^t, \overline{G})$
end
$c \leftarrow c_a^T$

Algorithm 1: Mediated Negotiation Protocol

5. EXPERIMENTAL EVALUATION

The hypothesis of this work is that the proposed adaptive mechanisms for voting and aggregation provide an improvement to the optimality of the negotiation processes. To evaluate this, we have performed a set of experiments to compare the results of the basic approaches with the results obtained introducing the adaptive mechanisms.

5.1 Experimental Settings

In each experiment, we conducted 100 negotiations for up to 10 agents and 100 contract issues. In each negotiation, agents try to find a mutually acceptable contract consisting of a vector of 100 boolean-valued issues ($x_i \in \{0,1\}$), corresponding to the presence or absence of a given contract clause. This leads to a contract space size of $2^{100} \approx 10^{30}$ — a number significantly too large to be explored exahustively.

Furthermore, we assumed contract issues to be pairwise interdependent and the utility function to be linearly additive: $U_j(c) = \sum_{p=1}^{100} \sum_{q=p}^{100} H_j(p,q) * x_p * x_q$, where $H_j(p,q)$ is the influence matrix that captures the dependencies between any given pair of issues x_p and x_q. Thus, each cell in the influence matrix takes random values between -100 and 100, which represents the utility increment or decrement caused by the presence of a given pair of issues. We used different random sets of influences matrix for each simulation run, in order to ensure our results were not idiosyncratic to a particular configuration of issue inter-dependencies.

We tested the different negotiation mechanisms for $|A| = \{2, 3, 5, 7, 10\}$ agents, $T = \{100, 1000\}$ negotiation rounds, and an initial population size $Z^1 = 20$. A '*most agents satisfied*' quantifier $Q(y) = x^2$ is used in all the experiments.

We define the following options for the acceptance quota, proposal size and voting type:

- Acceptance quota and proposal size types: {*'fixed', 'dynamic quota', 'dynamic population', 'dynamic', 'decay'*}.
  - **(i)** ***'fixed'*** **(F)**: acceptance quotas $q^{1...T}$ are static, and their values are 15 **(F15)** or 5 **(F5)**.
  - **(ii)** '***dynamic quota***' **(DynQ)**: $q^{t+1} \leftarrow q^t + 10\%$ if the group support for the active contract is below a threshold $G(\overline{V}_{c_a^{t+1}}) < 0.75$, and $q^{t+1} \leftarrow q^t - 10\%$ otherwise.
  - **(iii)** '***dynamic population***' **(DynP)**: $Z^{t+1} \leftarrow Z^t - 10\%$ if $G(\overline{V}_{c_a^{t+1}}) < 0.75$ and $Z^{t+1} \leftarrow Z^t + 10\%$ otherwise.
  - **(iv)** **'*dynamic*' (Dyn)**: the '*dynamic quota*' and '*dynamic population*' options are applied simultaneously.
  - **(v)** **'*decay*' (Dcy)**: quota declines over time, where $q^t = q^1 * (q^T/q^1)^{(t-1)/(T-1)}$, $q^1 = 15$ and $q^T = 1$.

  The rational behind the dynamic (adaptive) mechanisms is to adapt the quota and population size depending on the overall agreement level. Generally, the best scenario is to have a large population, a small quota and a high group support. A large population implies contract diversity, and a small quota means that agents do not loose their voting power on unwanted solutions. In this scenario agents have a high selection power. However, if the group support decreases because of conflicts, our strategy is to reduce the local optimization power of the agents by decreasing the population size or/and by increasing the acceptance quota (see Figure 2).

- Voting mechanisms: {*'binary', 'ranked', 'truthful'*}:
  - **(i)** ***binary*** **(B)**, each agent gives a vote 1 (accept) to the q^t contracts with highest private utility U_j.
  - **(ii)** ***ranked*** **(R)**, each agent votes 1 for the contract with highest utility and $1 - n/q^t$ for the contracts with the n-*th* highest utility.

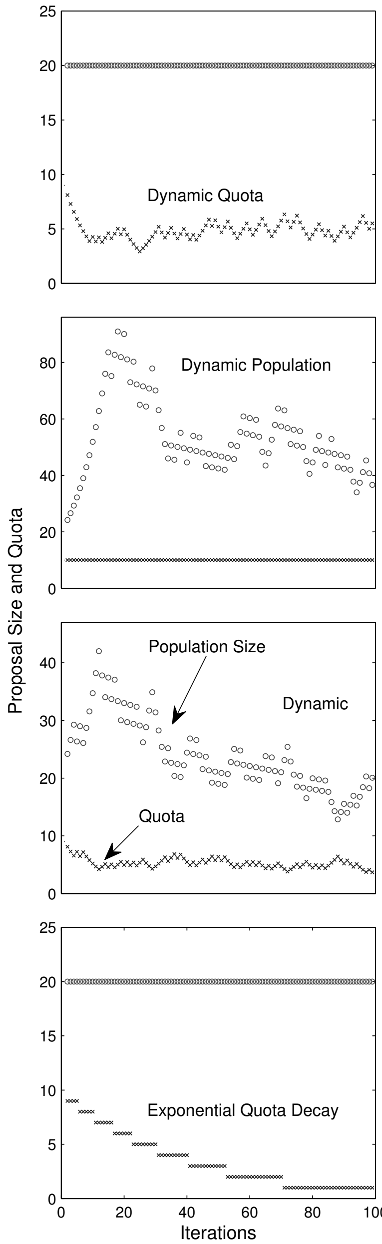


Figure 2: Example of operation of the dynamic mechanisms

– **(iii)** ***truthful*** **(T)**, each agent votes truthfully for the q^t contracts with highest utility. The purpose of this mechanism is to provide a reference in terms of computational efficiency.

5.2 Performance Measures

Pareto efficiency is an important measure in negotiation analysis, which means that no agent can be better off without leaving another agent worse off. Several other concepts can be regarded as desirable [22]. One is the social welfare (SW) optimum ($c^* = \arg\max_c \sum_{j\in\mathcal{A}} U_j(c)$).There are also axiomatic solution approaches. The most famous is the Nash bargaining solution ($c^n = \arg\max_c \prod_{j\in\mathcal{A}} U_j(c)$), which intends to yield a fair agreement whilst maximizing joint efficiency. Finally, the Kalai's Egalitarian solution follows the maximin principle, i.e., the objective is to maximize the utility of the agent worst off ($c^k = \arg\max_c \min\{U_1(c), \ldots, U_A(c)\}$).

In our experimental testbed, the Pareto frontier (i.e., the set of pareto efficient solutions) was approximated using a multi-objective evolutionary algorithm. Because of its complexity, we computed it only for up to 3 agents. The Nash, Kalai and Social welfare optima are approximated using a Genetic optimization algorithm [5]. Regarding the performance, we used the Euclidean distance between a negotiation solution c and the Pareto frontier

$$c^p = \arg\min_{c^{p'}\in P} Euclidean(c, c^{p'}),$$

Nash solution c^n and Kalai's Egalitarian solution c^k. Furthermore, we normalized the results by defining the individually best solutions ($c_j^{best} = \arg\max_c U_j(c)$). Thus, the Pareto (P), Nash (N) and Kalai's Egalitarian (K) performance is computed as a percentage as follows:

$$\{P(c), N(c), K(c)\} = \sqrt{\sum_{j\in\mathcal{A}} \left(\frac{U_j(c) - U_j(\{c^p, c^n, c^k\})}{U_j(c_j^{best})}\right)^2}.$$

Social welfare (SW) is computed as a ratio instead of an Euclidean distance: $SW(c) = \sum_{j\in\mathcal{A}} \frac{U_j(c)}{U_j(c^*)}$ (note that optimal SW solutions may be scattered along the pareto frontier). To test the hypothesis whether a configuration is significantly better, we conducted one-sided Wilcoxon rank sum tests on the results. We used a significance level of 0.05 (i.e., p value is 5%).

5.3 Results

The outcomes close to the Pareto frontier are considered as efficient, the outcomes close to the Nash product maximum as jointly efficient, and the outcomes close to the Egalitarian minimum maximization as socially fair. The results (average values of the 100 negotiations per experiment) for two agents, 100 and $1,000$ iterations are given in Table 1. Overall, the results improve for 1000 iterations. The **fixed quota** and **binary voting (F15+B)** combination has the worst performance, because it leads to too many acceptances.

For **two agents** and the same voting mechanism **(B)**, **(R)** or **(T)**, the use of **dynamic proposal size (DynP)** outperforms any other mechanism. The **decay acceptance quota (Dcy)** significantly worsen the agents' performance. This mechanism seems to introduce conflicting moves in the final negotiation steps. The static approaches **(F5+B, F5+T)** and **dynamic acceptance quota (DynQ+B, DynQ+T)** obtain similar results. However, **ranking voting (DynQ+R)** outperforms **(F5+R)**, obtaining outcomes close to **(DynP+R)**.

Table 1: Pareto, Nash and Egalitarian results for two agents

	Pareto (%)		Nash (%)		Kalai (%)	
	100	1000	100	1000	100	1000
			J=2			
F15+B	24.4035	21.5944	26.3019	23.4750	25.4673	22.5036
F5+B	9.0351	7.0317	11.9636	9.5106	10.6076	8.5551
DynQ+B	9.1773	7.7668	12.1745	10.8191	10.9419	9.5417
DynP+B	7.4651	6.6080	10.1592	9.9701	9.2289	8.4841
Dyn+B	8.6354	7.1051	11.9100	10.3646	10.5099	9.1265
Dcy+B	7.6901	7.2966	11.4719	11.1792	10.7884	10.1315
F5+R	4.7380	3.2419	9.5074	9.2618	8.2201	7.9896
DynQ+R	3.5109	**0.6997**	8.1870	**4.0909**	7.2535	**3.2792**
DynP+R	**2.3618**	**0.6827**	**6.2615**	**3.8571**	**6.2001**	**3.3238**
Dyn+R	4.4346	7.2942	8.5622	10.3618	7.5649	9.3528
Dcy+R	6.7583	4.7328	12.4822	10.0333	11.1682	9.6724
F5+T	3.9241	2.6399	7.6836	7.6325	7.2149	6.6881
DynQ+T	4.0598	3.0200	7.9414	7.5758	7.5362	6.3733
DynP+T	2.4266	1.4781	7.6634	6.8032	6.4387	5.5624
Dyn+T	3.9743	6.6165	7.9182	10.5432	6.8935	8.8880
Dcy+T	5.9099	5.4496	10.6815	10.7014	9.5163	10.3655

The combination of dynamic proposal and dynamic quota **(Dyn)** does not improve the performance when compared with **(DynQ)** or **(DynP)**. We can conclude that **(DynP+R)*** represents the optimal combination. Interestingly, there is no benefit in using truthful voting **(T)**, which is a voting mechanism that introduces the strategical issue in the negotiation process. Overall, the results for **three agents** (see Table 2) are similar to those obtained for two agents.

With many agents the Nash and Egalitarian performance differs depending on the used mechanisms. Figure 3 shows that the results closer to the Nash product maximum correspond to **(DynP+R)** and **(DynQ+R)**. However, the Kalai performance improves with **binary voting**. With five agents and 1000 iterations **(DynQ+B)** and **(DynP+B)** have a mean Kalai distance of $9.9 - 10.1\%$. With seven and ten agents and for 100 iterations the best Kalai performance is obtained with the dynamic mechanisms, but when we jump to 1000 iterations, **(F15+B)** becomes the best choice. Intuitively, scenarios with a higher complexity benefit from mechanisms that enforce cooperation. An accept or reject voting mechanism and a high acceptance quota limit strongly the agents' local optimization power, favoring an egalitarian outcome. Summarizing, the main finding is that with many agents, Nash performance improves with dynamic and ranking mechanisms, while Kalai performance improves with accept and reject votes and a high acceptance quota.

The average social welfare outcomes are presented in Figure 3. The narrow bars show the results for 100 iterations, whereas the wide bars show the results for 1000 iterations. The best performance is obtained with **DynQ+R** and **DynP+R**, which achieve outcomes close to a 99%. Binary voting does not appear as a valid approach when compared to ranking voting mechanisms, exhibiting very poor

Table 2: Pareto, Nash and Egalitarian results for three agents

	Pareto (%)		Nash (%)		Kalai (%)	
	100	1000	100	1000	100	1000
			J=3			
F15+B	14.0849	10.9152	22.3587	19.7686	18.0458	14.9342
F5+B	3.6940	2.9192	11.0031	9.3569	9.0227	7.8539
DynQ+B	4.3858	2.6917	12.4133	9.3138	9.0553	**7.4209**
DynP+B	4.0281	**2.5375**	11.1123	8.3696	**8.2445**	**7.2374**
Dyn+B	4.3157	4.0793	12.4244	11.6642	9.2542	8.8060
Dcy+B	5.4649	3.6120	15.0199	13.2091	11.8977	10.1133
F5+R	**2.6484**	**2.6182**	10.5040	8.8805	9.8143	9.0557
DynQ+R	3.1229	4.2503	8.5926	7.5944	**8.3771**	**7.3297**
DynP+R	**2.8849**	**2.6482**	**7.6881**	**7.1310**	**8.6171**	7.8859
Dyn+R	3.0947	3.2298	10.0558	9.0279	9.1675	7.8255
Dcy+R	4.4417	3.5073	14.9814	13.8523	10.9994	10.6640
F5+T	3.1372	2.8365	10.5081	8.7549	9.2788	7.7896
DynQ+T	3.6475	**2.6759**	11.9229	9.2190	9.6520	**7.1825**
DynP+T	**2.5817**	2.9997	10.5187	8.0636	11.5097	7.8377
Dyn+T	2.8419	2.6693	11.3095	9.3013	9.7742	7.9247
Dcy+T	4.8271	3.8138	14.4667	12.6045	12.0156	10.6212

results with few iterations. Another important result is that as the number of agents increases, the performance gap between 100 and 1000 iterations decreases. Finally, we can see how **Dcy** and **Dyn** exhibit the worst performance. **Dyn** worsen its results for 1000 iterations. This is again an indication of unstable behavior of the combination of dynamic proposal and quota.

6. DISCUSSION AND RELATED WORK

There exist examples of negotiation mechanisms to obtain agreements by using fair direction improvements [6, 10, 19]. These mechanisms are prone to untruthful revelation to bias the direction generated by the mediator. With our voting mechanisms (binary and ranking) and the acceptance quota, agents do not have an incentive to act strategically. In imperfect information scenarios, there is no incentive to reject a contract which is preferred over another, and there is no reason to alter the ranking of preferred contracts. In [14, 9] the concepts of token and limit on strong votes are used. It is not clear whether the token scheme or the limited votes rule is incentive compatible for multilateral negotiations. In [8] a Simulated Annealing based protocol is proposed, in which the agents accept a certain quota over time. However, the incentive compatibility is hard to analyze due to the complexity of strategic interactions.

In more recent research [17] inspired on the seminal work by [13], an approach is taken where the mediator looks for Pareto improvements using a mediated protocol. It is similar to ours in that it uses an acceptance quota function which declines over time (i.e., the Dcy mechanism). Our work defines a more general protocol architecture, which considers a formal framework to aggregate preferences. We extend these works by considering adaptive mechanisms which dinamically modify the acceptance quota and the proposal size during the negotiation process. Moreover, we evaluate different voting mechanisms beyond the accept or reject votes. When compared against an exponential decay quota function and a fixed proposal size, our adaptive mechanisms have exhibited a significantly better behavior.

7. CONCLUSIONS AND FUTURE WORK

In this study we first proposed a negotiation protocol architecture. Its purpose is to facilitate the design of negotiation protocols which are based on heuristic optimization. We use ordered weighted average operators (OWA) as a general framework to aggregate preferences. Then we proposed a negotiation protocol which is inspired by the mutation mechanism of genetic algorithms. The protocol applies a 'most agents satisfied' linguistic quantifier to compute the aggregation of preferences. In addition, we defined adaptive mechanisms for proposal generation and voting. The empirical results show that our proposal outperform significantly previous approaches. The voting mechanism has a strong influence on the negotiation, where ranking voting appears as the best alternative, with the only exception of the egalitarian performance in scenarios with many agents.

The validity of the results is restricted by the generalization issue. Other negotiation scenarios with different non-linear dependencies may potentially lead to different results. Further work includes the enhancement of the adaptive mechanisms that control the dynamic acceptance quota and proposal size, and the development and evaluation of voting mechanisms and heuristics. Our purpose is to facilitate the development of automated group decision making systems in diverse scenarios.

8. ACKNOWLEDGEMENTS

This work has been supported by the Spanish Ministry of Economy and Competitiveness grant: TEC2013-45183-R CIVTRAff.

9. REFERENCES

[1] K. Arrow. *Social choice and individual values*. New Have: Cowles Foundation, 1963.

[2] K. Binmore and N. Vulkan. Applying game theory to automated negotiation. *NETNOMICS*, 1(1):1–9, 1999.

[3] C. Blum and A. Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.*, 35(3):268–308, Sept. 2003.

[4] E. de la Hoz, M. A. López-Carmona, M. Klein, and I. Marsá-Maestre. Hierarchical clustering and linguistic mediation rules for multiagent negotiation. In *International Conference on Autonomous Agents and Multiagent Systems, AAMAS 2012, Valencia, Spain, June 4-8, 2012 (3 Volumes)*, pages 1259–1260, 2012.

[5] K. Deep, K. P. Singh, M. Kansal, and C. Mohan. A real coded genetic algorithm for solving integer and mixed integer optimization problems. *Applied Mathematics and Computation*, 212(2):505 – 518, 2009.

[6] H. Ehtamo, R. P. Hamalainen, P. Heiskanen, J. Teich, M. Verkama, and S. Zionts. Generating pareto solutions in a two-party setting: constraint proposal methods. *Management Science*, 45(12):1697–1709, 1999.

[7] S. Fatima, S. Kraus, and M. Wooldridge. *Principles of Automated Negotiation*. Cambridge University Press, New York, USA, 2014.

[8] A. Fink. Supply chain coordination by means of automated negotiations between autonomous agents. In B. Chaib-draa and J. MÃŒller, editors, *Multiagent*

Table 3: Nash and Egalitarian results for 5, 7 and 10 agents

	Nash (%)		Kalai (%)		Nash (%)		Kalai (%)		Nash (%)		Kalai (%)	
	100	1000	100	1000	100	1000	100	1000	100	1000	100	1000
	J=5				*J=7*				*J=10*			
F15+B	20.7501	16.0241	14.2476	10.8828	21.4515	14.0752	**14.5089**	**12.8833**	30.2213	18.4573	20.2233	**14.9284**
F5+B	14.1760	12.9067	13.0230	12.1873	16.2811	16.3986	15.0867	16.9359	20.1064	18.2436	19.9918	20.3536
DynQ+B	13.8331	12.2061	**10.5852**	**9.9217**	19.7023	12.6858	**14.4189**	13.9974	23.8707	14.9435	**16.7038**	17.7951
DynP+B	14.1666	10.3285	**10.1666**	**10.1477**	19.2505	12.5507	**14.3436**	14.5165	24.5153	14.5992	**16.9826**	17.3286
Dyn+B	15.8782	15.2873	11.6594	12.4055	18.2596	17.5096	15.4124	15.4700	21.6918	18.7086	**16.6089**	17.3667
Dcy+B	18.1394	17.6759	15.4388	15.9162	19.9924	18.5203	17.7636	16.4733	21.5961	19.6789	18.6269	20.6787
F5+R	14.7015	13.9142	13.6427	15.9284	18.0471	15.3337	16.8140	18.6252	18.3558	17.2536	20.0026	22.6709
DynQ+R	**10.9768**	**7.8072**	11.1305	12.2343	**13.7920**	**12.1641**	17.0201	17.3181	**16.3727**	**13.0274**	20.4084	21.6828
DynP+R	**10.5991**	**8.8156**	11.3715	13.0500	**13.2899**	**11.7350**	16.4918	16.5456	**16.1619**	**13.8638**	19.1281	20.5806
Dyn+R	15.2081	12.8314	11.5802	10.6618	16.5554	14.3705	16.2779	15.0290	16.6539	16.2453	18.6770	20.6552
Dcy+R	18.0021	16.0229	15.5532	14.8140	17.3444	21.1181	15.8462	19.3935	20.1378	21.4587	20.3867	21.9018
F5+T	14.1835	13.2478	14.4781	13.2548	17.2183	15.8363	18.0852	18.1382	18.0420	18.6890	20.3222	24.0278
DynQ+T	12.0414	11.2203	13.4227	13.3740	14.3364	15.8606	14.9017	16.8462	17.7174	15.7792	18.2199	20.4979
DynP+T	11.1865	11.7971	11.6252	13.3402	14.3204	13.8120	16.3876	16.0462	17.4119	15.5017	18.3150	20.5665
Dyn+T	13.6669	15.2330	12.5702	13.1354	14.3000	17.0082	14.8762	15.8656	18.1684	17.1185	17.0203	18.1579
Dcy+T	16.9486	16.3095	14.2646	14.4182	20.4441	18.8377	18.8219	19.3305	21.3125	20.4015	19.9724	20.7349

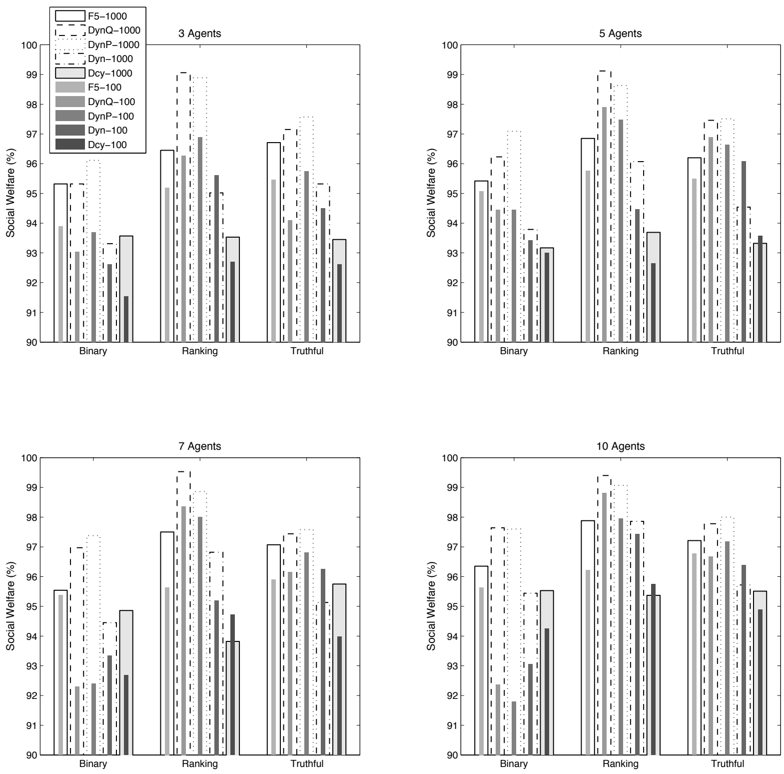


Figure 3: Social welfare: 3, 5, 7 and 10 agents, 100 and 1000 iterations

based Supply Chain Management, volume 28, pages 351–372. Springer Berlin Heidelberg, 2006.

[9] K. Fujita, T. Ito, and M. Klein. Efficient issue-grouping approach for multiple interdependent issues negotiation between exaggerator agents. *Decision Support Systems*, 60(0):10 – 17, 2014. Automated Negotiation Technologies and their Applications.

[10] P. Heiskanen, H. Ehtamo, and R. P. Hamalainen. Constraint proposal method for computing pareto solutions in multi-party negotiations. *European Journal of Operational Research*, 133(1):44–61, 2001.

[11] J. H. Holland. *Adaptation in Natural and Artificial Systems*. MIT Press, Cambridge, MA, USA, 1992.

[12] T. Ito, M. Klein, and H. Hattori. A multi-issue negotiation protocol among agents with nonlinear utility functions. *Journal of Multiagent and Grid Systems*, 4(1):67–83, 2008.

[13] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12(2):111–125, 2003.

[14] M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Protocols for negotiating complex contracts. *IEEE Intelligent Systems*, 18(6):32–38, 2003.

[15] G. Lai, C. Li, and K. Sycara. Efficient multi-attribute negotiation with incomplete information. *Group Decision and Negotiation*, 15(5):511–528, 2006.

[16] G. Lai, C. Li, K. Sycara, and J. Giampapa. Literature review on multiattribute negotiations. Technical Report CMU-RI-TR-04-66, Robotics Institute, Carnegie Mellon University, Pittsburgh, USA, December 2004.

[17] F. Lang and A. Fink. Learning from the metaheuristics: Protocols for automated negotiations. *Group Decision and Negotiation*, 24(2):299–332, 2014.

[18] M. A. Lopez-Carmona, J. R. Velasco, and B. Alarcos. An expressive fuzzy constraint based framework for agent based purchase negotiations. Technical report, University of Alcala, 2006.

[19] I. Marsa-Maestre, M. A. Lopez-Carmona, J. R. Velasco, and B. A. Alcazar. Using expressive dialogues and gradient information to improve trade-offs in bilateral negotiations. In *Proceedings of the 9th International Conference con Electronic Commerce and Web Technologies, ECWEB08 (DEXA)*, pages 1–8, 1-5 September 2008.

[20] T. W. Sandholm. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, chapter 5, pages 201–258. Distributed Rational Decision Making. MIT Press, Cambridge MA, USA, 1999.

[21] C. Schneeweiss. Distributed decision making unified approach. *European Journal of Operational Research*, 150(2):237 – 252, 2003.

[22] J. Sebenius. Negotiation analysis: A characterization and review. *Management Science*, 38(1):18–38, 1992.

[23] M. Subramani and E. Walden. Economic returns fo firms from business-to-business electronic commerce initiatives: An empirical examination. In *Proc. 21st International Conference on Information Systems*, pages 279–295, Brisbane, Australia, 2000.

[24] J. von Neuman and O. Morgenstern. *The Theory of Games and Economic Behaviour*. Princeton University Press, Princeton NJ, USA, 1944.

[25] M. Wooldridge. *An Introduction to Multiagent Systems*. John Wiley & Sons, 2002.

[26] R. Yager. Multi-agent negotiation using linguistically expressed mediation rules. *Group Decision and Negotiation*, 16(1):1–23, January 2007.

[27] R. Yager and J. Kacprzyk. *The Ordered Weighted Averaging Operators: Theory and Applications*. Kluwer, 1997.

[28] L. Zadeh. A computational approach to fuzzy quantifiers in natural languages. *Computing and Mathematics with Applications*, 9:149 – 184, 1983.

A Multi-Criteria Group Based Matching Approach of Buyers and Sellers Through a Broker in Open E-marketplaces

Dien Tuan Le
School of Computing and Information Technology
University of Wollongong
Wollongong, NSW, Australia
dtl844@uowmail.edu.au

Minjie Zhang
School of Computing and Information Technology
University of Wollongong
Wollongong, NSW, Australia
minjie@uow.edu.au

Fenghui Ren
School of Computing and Information Technology
University of Wollongong
Wollongong, NSW, Australia
fren@uow.edu.au

ABSTRACT

Electronic brokers in open e-marketplaces are able to provide services of facilitating and organizing the relationship between buyers and sellers. It is an important decision problem for a broker to achieve the optimal trade matching between buyers and sellers in multi-attribute trading. However, broker's matching theory and mechanisms in multi-attribute trading in e-marketplaces are limited. Thus, broker's matching approach is proposed based on a three party system, i.e., a broker, buyers and sellers to satisfy buyer's requirements and maximize buyer's total satisfaction degree. The contributions of this paper are that (i) a broker models buyer's requirements for attributes with fuzzy information through communications between a broker and buyers; (ii) broker's matching processes are carried to satisfy buyer's requirements and maximize buyer's total satisfaction degree in multi-attribute trading based on a generated objective function; and (iii) broker's strategy is proposed to allocations between buyers and sellers based on buyer's feedbacks from determined matching results. Experimental results demonstrate the good performance of the proposed approach.

Keywords

Matching approach; Buyer's total satisfaction degree; Buyer's requirements; Seller's offers

1. INTRODUCTION

In the recent years, e-marketplaces become more popular in many organizations and gradually replace more and more from the conventional business [6, 12]. Much trading information between buyers and sellers is exchanged in business environments. Specially, matching processes between buyers and sellers become complex and difficult on last twenty years due to the increasing number of buyers and sellers in e-marketplaces. Facing the inconvenience, people are paying more and more attention to electronic brokers. A broker's purposes is to find the best fitting trading target between buyers and sellers in the time as short as possible in e-marketplaces [8, 3].

Research on brokers or intermediaries in e-marketplaces as a third party of the trading processes between buyers and sellers has been a very active direction in recent years. Due to a vast amount of trading information in e-marketplaces, it is extremely difficult for buyers and sellers to distinguish between useful and not useful information to support their decisions. This difficulty has generated opportunities for electronic brokers to enter electronic marketplaces [5]. As the Internet becomes more popular, it is becoming more difficult and more expensive to find out necessary information on organizations and their offers. To solve the situations, a common way of obtaining this information for organizations in e-marketplaces is through intermediaries called brokers or matchmakers [4] or brokerage centres [2]. Khosla et al. [9] proposed an electronic brokerage system based on a human-centered layered architecture to find the most appropriate seller to provide goods or services with the lowest price or best quality for buyers.

The above approaches have focused on studying brokers as the third party in the trading process between buyer's requirements and seller's offers in e-marketplaces. However, there is little theory and few guidelines to help a broker to satisfy buyer's requirements for attributes with fuzzy information and maximize buyer's total satisfaction degree as per seller's offers. Most current brokers in e-marketplaces only provide buyer's or seller's trade information and do not really carry out functions of matching between buyers and sellers. The lack of a comprehensive optimization matching approaches has no solid foundation for improving exchange efficiency and market efficiency under considering buyers or sellers. Therefore, effective approaches should be investigated to find out optimal allocations between buyer's general requirements, which are related to attributes with hard, soft constraints and fuzzy information, and seller's offers in multi-attribute trading.

The aim of this paper is to propose a multi-criteria group based matching approach between buyers and sellers through a broker to maximize buyer's total satisfaction degree as goal. The major contributions of this paper are as follows: (i) a proposed conceptual framework is applicable to help a broker to model buyer's requirements for attributes with fuzzy information; (ii) an objective function and a set of constraints are generated to carry out broker's matching process; and (iii) broker's strategy is proposed to make a

DOI: 10.1145/1235

decision to allocations between buyers and sellers after a broker receives buyer's feedbacks from matching results. Experimental results illustrate the application of the proposed matching approach.

The rest of this paper is organized as follows. Problem description is presented in Section 2. The broker-based proposed matching approach is introduced in Section 3. An experiment is presented in Section 4. Section 5 compares our approach with some related work. Section 6 concludes in this paper and points out our future work.

2. PROBLEM DESCRIPTION

There are three members in the trading process with multi-attribute trading, i.e., buyers, sellers and a broker. The trading process is shown in Figure 1.

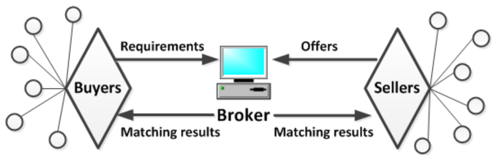


Figure 1: The trading processes through a broker in business environments

A broker is often called a facilitator, who acts as an intermediary between the buyer and the seller in the commodity exchange. In this paper, broker's responsibilities are to match n buyers with m sellers for many different commodities to satisfy buyer's requirements and maximize buyer's total satisfaction degree under multi-attribute trading. Each buyer or seller has a single unit of the commodity with multiple attributes to buy or sell.

From buyer's part, buyers can present their requirements through multiple attributes. In general, some attributes belongs to complete information for buyers. Thus, it is easy for buyers to choose fixed values as their needs. On the other hand, other attributes is vague information for buyers so it is difficult for buyers to estimate the attribute level with exact numerical values. So, product attribute values are incorporated in our proposed approach by taking the fuzzy or linguistic requirements of buyer as inputs. Based on the fuzzy or linguistic requirements of buyers as inputs, a broker will build membership functions to measure buyer's satisfaction degree for attribute type with vague information.

Similarly, from seller's point of view, sellers express their needs through multiple attributes. Due to seller's own products, it is easy for sellers to determine the attribute level with reasonable values. Thus, level of each attribute in seller's offers is provided in details to a broker.

Based on the above analysis, the key problem is how to help a broker to (i) model buyer's behaviors related to various attributes, i.e., attributes with hard, soft constraints and attributes with fuzzy information; (ii) carry out matching processes to maximize buyer's total satisfaction degree; and (iii) to make a decision on allocations between buyers and sellers based on buyer's feedbacks from matching results. Therefore, the proposed matching approach can solve this problem and is presented in Section 3.

3. A BROKER-BASED PROPOSED MATCHING APPROACH

3.1 The conceptual framework of broker's matching process

A trading process between buyers and sellers is conducted through a broker to achieve optimal matching pairs. In this paper, broker's responsibilities are how to maximize buyer's total satisfaction degree as social goals based on buyer's preferences and seller's capacities under multi-attribute trading. The principle of the whole matching process between buyers and sellers through a broker in our approach is presented in Figure 2.

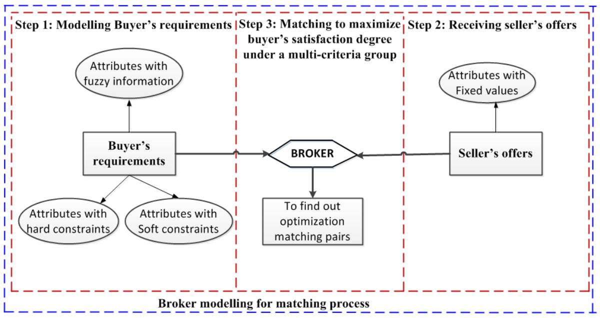


Figure 2: The conceptual framework of broker's matching process in business environments

Step 1: A broker receives buyer's requirements in term of multi attributes. To model buyer's requirements related to attributes with fuzzy information, a broker communicates with buyers to determine buyer's behaviors based on building buyer's fuzzy membership function. In particular, a broker starts the simplified interactive procedure with buyers through asking some questions so that a broker identifies buyer's reference points to build buyer's membership function.

Step 2: Sellers have their product's own requirements. Thus, sellers provide attribute's fixed values to a broker and they would like to find out which buyers satisfy seller's own requirements through a broker. Of course, seller's offers contain the same kinds of attributes in buyer's requirements.

Step 3: After modelling buyer's requirements and receiving seller's offers, a broker carries out matching processes to seek optimal matching paries. Broker's matching process is to maximize buyer's total satisfaction degree under allocations between buyer's requirements and seller's offers.

3.2 Fuzzy representation of product attributes in buyer's requirements

Due to buyer's vague knowledge about some attributes of products in business environments, it is difficult for a buyer to express their preferences with exact numerical values. Thus, a broker employs fuzzy membership functions to express buyer's preferences over attributes of products. The membership functions are not only used to as the equivalents of utility functions over attributes of products but also they can help a broker compare buyer's satisfaction degree with offers of different sellers. In particular, the fuzzy membership functions are defined as follows:

Definition: Let X be a set of objectives. A fuzzy set **A** in X is defined as by its membership function as follows.

$$\mu_{\mathbf{A}} : X \to [0, 1], \tag{1}$$

and $\forall x \in X$ is called the membership degree of x in fuzzy set $\mathbf{A}$.

There are some popular fuzzy numbers to express buyer's behaviors through fuzzy membership functions in business environments shown in Figure 3.

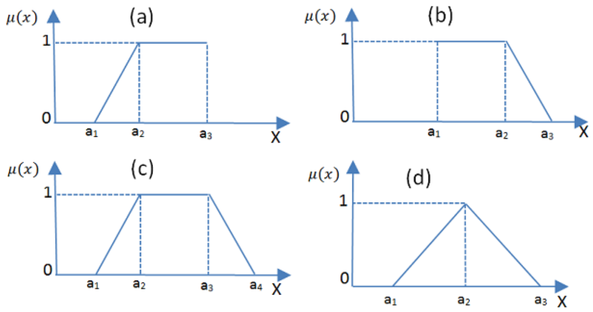


Figure 3: Representation of fuzzy information, (a) left semi-trapezoidal fuzzy number; (b) right semi-trapezoidal fuzzy number; (c) trapezoidal fuzzy number; (d) triangular fuzzy number

In this paper, a broker determines buyer's membership function for each attribute by using the direct rating (point estimation) method [11]. In particular, a broker communicates with buyers to determine buyer's preference point through questions. Broker's questions require buyers to select one point on the reference axis by using numerical scale that best describes this element.

For example, a broker starts the simplified interactive procedure with buyers to build buyer's satisfaction degree as per capacities of hard disk. In particular, a broker requires buyers to answer the three following questions so that a broker identifies buyer's three reference points within the feasible range of hard disk's capacities.

- Question 1:'What is the worst hard disk's capacities?' $\to$ 'everything is the worst if hard disk's capacities are less than or equal to 10G, or more than or equal to 50G'.
- Question 2:'What is the perfect capacities of hard disk that would give you full satisfaction level?' $\to$ 'the perfect capacities of hard disk are between 20G and 40G'.
- Question 3:'What is a medium satisfaction level for you with regard to capacities of hard disk?' $\to$ 'capacities of hard disk are between 10G and 20G, or between 40G and 50G'.

Based on buyer's responses above, buyer's satisfaction function as per hard disk's capacities is presented in Equation (2) and Figure 4.

$$\mu(x) = \begin{cases} 0 & for \quad x \leq 10 \ or \ x \geq 50 \\ \frac{x-10}{10} & for \ 10 < x < 20 \\ 1 & for \quad 20 \leq x \leq 40 \\ \frac{50-x}{10} & for \ 40 < x < 50 \end{cases} \tag{2}$$

3.3 Calculating buyer's satisfaction degree

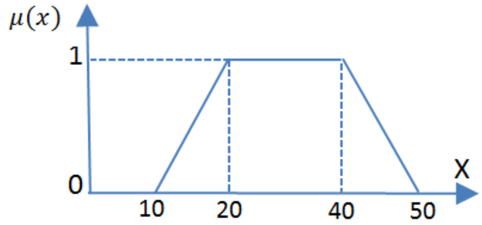


Figure 4: For example, buyer's satisfaction degree as per capacities of hard disk

Let a set of buyer $\mathbf{B} = \{b_1, b_2, \ldots, b_n\}$, a set of sellers $\mathbf{S} = \{s_1, s_2, \ldots, s_m\}$, buyer's requirements and seller's offers are related to multi attributes which can be split into attributes with hard constraints, soft constraints and fuzzy information shown in Figure 2. It is convenient to present the calculation method of buyer's satisfaction degree, group of attributes in buyer's requirements is divided into a set of attributes with hard constraints $\mathbf{H}=\{h_1, h_2, \ldots, h_z\}$ and a set of other attributes $\mathbf{A}=\{a_1, a_2, \ldots, a_k\}$[7].

Seller's offers are provided to a broker and these offers contains the same kinds of attributes requested by buyers. Let m be number of sellers, $j \in [1, \ldots, m]$ is the index of s_j's offer and $Q_{ji'}$ ($i' \in (z+k)$) is the crisp value of the i'^{th} attribute of s_j's offer.

Let n be number of buyers, $i \in [1, \ldots, n]$ is the index of b_i's requirements and $C_{ii'}$ ($i' \in (z+k)$) is the crisp value of the i'^{th} attribute of b_i's requirements. if a_k belongs to a quantitative attribute with fuzzy information, buyer's requirements for a_k is presented the fuzzy behavior of attributes which have their own specific interval ranges of X (referred to subsection 3.2).

Based on notations of buyer's requirements and seller's offers above, the calculation procedure of buyer's satisfaction degree for all attributes is presented as follows:

(i) *for attribute type with fuzzy numbers:* attributes with fuzzy number have different units, and thus normalization is required before performing broker's matching process between buyer's requirements and seller's offers. The normalization helps a broker to determines b_i's satisfaction degree for attribute a_k between b_i and s_j based on b_i's membership function for a_k. Normalization of Q_{jk} in b_i's requirements for attribute a_k, denoted by $\beta_{ij}^{a_k}$, has a value between 0 and 1. Assume that attribute a_k is a triangular fuzzy number, thus depending on value Q_{jk}, which belongs to Figure 5 or Figure 6, $\beta_{ij}^{a_k}$ is calculated as follows.

$$\beta_{ij}^{a_k} = (Q_{jk} - a_1)/(a_2 - a_1) \ for \ a_1 \leq Q_{jk} < a_2 \tag{3}$$

$$\beta_{ij}^{a_k} = (a_3 - Q_{jk})/(a_3 - a_2) \ for \ a_2 \leq Q_{jk} \leq a_3 \tag{4}$$

Similarly, if attribute a_k is trapezoidal fuzzy number, $\beta_{ij}^{a_k}$ is calculated as follows.

$$\beta_{ij}^{a_k} = (Q_{jk} - a_1)/(a_2 - a_1) \ for \ a_1 \leq Q_{jk} < a_2 \tag{5}$$

$$\beta_{ij}^{a_k} = (a_4 - Q_{jk})/(a_4 - a_3) \ for \ a_4 \leq Q_{jk} < a_3 \tag{6}$$

$$\beta_{ij}^{a_k} = 1 \ for \ a_2 \leq Q_{jk} \leq a_3 \tag{7}$$

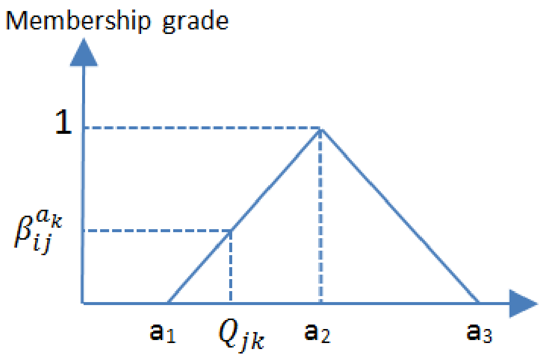


Figure 5: Q_{jk} is between a_1 and a_2

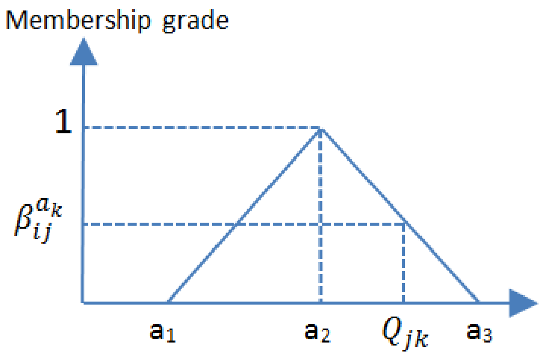


Figure 6: Q_{jk} is between a_2 and a_3

(ii) *for attribute type with hard constraints:*

$$\beta_{ij}^{a_h} = \begin{cases} -1 & \text{if } C_{ih} \neq Q_{jh} \\ 1 & \text{if } C_{ih} = Q_{jh} \end{cases} \tag{8}$$

$\beta_{ij}^{a_h} = -1$ means that a seller s_j does not match with a buyer b_i for attribute a_h and $\beta_{ij}^{a_h} = 1$ means that a seller s_j matches with a buyer b_i for attribute a_h.

(iii) *for attribute type with benefit soft constraints:* if $C_{ik} > Q_{jk}$ then $\beta_{ij}^{a_k} = -1$. It means that a seller s_j does not satisfied a buyer b_i. If $C_{ik} \leq Q_{jk}$, then $\beta_{ij}^{a_k}$ is calculated as follows:

$$\beta_{ij}^{a_k} = (\frac{Q_{jk} - Q_{min-k} + \phi}{Q_{max-k} - Q_{min-k} + \phi})^{t'} \tag{9}$$

where $t' = \frac{C_{ik}}{Q_{min-k}}$, Q_{min-k} is the minimal value of seller in the set of values for the attribute a_k and Q_{max-k} is the maximal value of seller in the set of values for the attribute a_k. A value $t' \in (0, 1]$ helps a broker to carry out comparing buyer's satisfaction degree when t' is used to calculated $\beta_{ij}^{a_k}$. $\phi = \frac{Q_{min-k}}{2}$, ϕ helps a broker to solve some special cases such as only one seller in the e-market or $Q_{max-k} = Q_{min-k}$. $\beta_{ij}^{a_k}$ increases when Q_{jk} increases or C_{ik} decreases.

$\beta_{ij}^{a_k}$ means that a seller s_j matches with a buyer b_i for attribute a_k with buyer's satisfaction degree $(\frac{Q_{jk}-Q_{min-k}+\phi}{Q_{max-k}-Q_{min-k}+\phi})^{t'}$. $\beta_{ij}^{a_k}$ is between 0 and 1. If $\beta_{ij}^{a_k}$ is near 1, it means that b_i is highly satisfied by s_j for attribute a_k.

(iv) *for attribute type with cost soft constraints:* if $C_{ik} < Q_{jk}$ then $\beta_{ij}^{a_k} = -1$. It means that a seller s_j does not satisfy a buyer b_i. If $C_{ik} \geq Q_{jk}$ then $\beta_{ij}^{a_k}$ is calculated as follows:

$$\beta_{ij}^{a_k} = (\frac{Q_{max-k} - Q_{jk} + \phi}{Q_{max-k} - Q_{min-k} + \phi})^{\frac{1}{t'}} \tag{10}$$

$\beta_{ij}^{a_k}$ means that a seller s_j matches with a buyer b_i for attribute a_k with buyer's satisfaction degree $(\frac{Q_{max-k}-Q_{jk}+\phi}{Q_{max-k}-Q_{min-k}+\phi})^{\frac{1}{t'}}$. $\beta_{ij}^{a_k}$ is between 0 and 1. If $\beta_{ij}^{a_k}$ is near 1, it means that b_i is highly satisfied by s_j for attribute a_k.

In summary, a broker considers b_i's satisfaction degree based on s_j's offer under multi-attribute trading. The attributes with hard constraints are necessary conditions in the trading processes and must be satisfied. Thus, attributes with hard constraints do not need its weight. If attributes with hard constraints are not satisfied, then b_i cannot match with s_j. On the other hand, attributes which are not hard constraints are necessary for using the weight because these attributes can be relaxed within the given scope of values. In particular, b_i's total satisfaction degree based on s_j's offers related to the attributes is as follows:

$$\sum_{l=1}^{k} w_l^i \beta_{ij}^{a_l}, \tag{11}$$

where w_l^i is a weight value of attribute a_l for b_i's requirements and $\sum_{l=1}^{k} w_l^i = 1$. In this paper, each b_i expresses the complete weight information for attributes in b_i's requirements.

3.4 Building broker's objective function

Broker's matching process can also be considered as an optimization problem because the goal of matching process is to find a set of optimal pairs, which satisfies buyer's requirements as per seller's offers and maximize buyer's total satisfaction degree. Based on broker's mentioned responsibilities, an objective optimization function and a set of constraints are built as follows.

$$f = \sum_{i=1}^{n} \sum_{j=1}^{m} (\sum_{l=1}^{k} w_l^i \beta_{ij}^{a_l} x_{ij}) \tag{12}$$

$$s.t. \sum_{i=1}^{n} x_{ij} \leq 1, j = 1, 2, \ldots, m \tag{13}$$

$$\sum_{j=1}^{m} x_{ij} \leq 1, i = 1, 2, \ldots, n \tag{14}$$

$$x_{ij} = 1, 0, (i = 1, 2, \ldots, n; j = 1, 2, \ldots, m) \tag{15}$$

$$\sum_{l=1}^{k} w_l^i = 1, (i = 1, 2, \ldots, n; l = 1, 2, \ldots, k) \tag{16}$$

$$\begin{aligned} x_{ij} = 0 \; if \; \beta_{ij}^{a_l} = -1 \; (l = 1, 2, \ldots, k) \\ or \; C_{ig} \neq C_{jg} (g = 1, 2, \ldots, h) \end{aligned} \tag{17}$$

where the objective optimization function in Equation (12) seeks to maximize the weight sum of buyer's satisfaction degree, constraints (13) and (14) are that each buyer (seller) can buy (sell) one unit of the commodity at most. Constraint (15) is assignment variable constraints, if b_i matches with s_j, then $x_{ij} = 1$; otherwise $x_{ij} = 0$. Constraint (16) indicates b_i's attribute weight; and constraint (17) indicates a constraint satisfaction layer to attend broker's matching processes. Furthermore, Equation (12) can be efficiently solved by well-known linear programming methods, such as simplex methods or interior point method [1].

3.5 Broker's strategy for allocations between buyers and sellers

After broker's matching process between buyer's requirements and seller's offers is carried through an objection function, matching pairs between buyers and sellers are found. The matching pairs helps to a broker to maximize buyer's total satisfaction degree as goal but the matching pairs cannot help a broker to evaluate whether buyers accept the determined matching results. Thus, to solve the issue, broker's strategy for making decisions to gain the final matching pairs between buyer's requirements and seller's offers is presented in Algorithm 1 as follows.

Algorithm 1 Broker's algorithm for allocations between buyers and sellers

1: **Input:** a set of buyer $\mathbf{B_1} = \{b_1, b_2, \ldots, b_n\}$, a set of seller $\mathbf{S_1} = \{s_1, s_2, \ldots, s_m\}$, a set of buyer's constraints $\mathbf{C}(\mathbf{B_1})$ and a set of seller's constraints $\mathbf{C}(\mathbf{S_1})$;
2: **Output:** return the final matching pairs between buyers and sellers, which are accepted by buyers;
3: $i \leftarrow 1$
4: $\{\mathbf{M^*_{BS}}\} \leftarrow$ match($\mathbf{B_i}$,$\mathbf{S_i}$,$\mathbf{C}(\mathbf{B_i})$,$\mathbf{C}(\mathbf{S_i})$)
5: $\{\mathbf{B^{ac}_i}, \mathbf{S^{ac}_i}, \mathbf{M^{ac}_{BS}}\} \leftarrow$ check($\mathbf{M^*_{BS}}$)
6: $\mathbf{C}(\mathbf{B_i}) \leftarrow$ update ($\mathbf{C}(\mathbf{B_i})$)
7: **while** ($\neg stopCriterion()$) **do**
8: $\quad i \leftarrow i + 1$
9: $\quad \mathbf{B_i} \leftarrow \mathbf{B_{i-1}} \setminus \{\mathbf{B^{ac}_i}\}$
10: $\quad \mathbf{S_i} \leftarrow \mathbf{S_{i-1}} \setminus \{\mathbf{S^{ac}_i}\}$
11: $\quad \mathbf{C}(\mathbf{B_i}) \leftarrow \mathbf{C}(\mathbf{B_{i-1}} \setminus \mathbf{B^{ac}_i})$
12: $\quad \mathbf{C}(\mathbf{S_i}) \leftarrow \mathbf{C}(\mathbf{S_{i-1}} \setminus \mathbf{S^{ac}_i})$
13: $\quad \{\mathbf{M^*_{BS}}\} \leftarrow$ match($\mathbf{B_i}$,$\mathbf{S_i}$,$\mathbf{C}(\mathbf{B_i})$,$\mathbf{C}(\mathbf{S_i})$)
14: $\quad \{\mathbf{B^{ac}_i}, \mathbf{S^{ac}_i}, \mathbf{M^{ac}_{BS}}\} \leftarrow$ check($\mathbf{M^*_{BS}}$)
15: $\quad \mathbf{M^{ac}_{BS}} \leftarrow$ update($\mathbf{M^{ac}_{BS}}$)
16: $\quad \mathbf{C}(\mathbf{B_i}) \leftarrow$ update ($\mathbf{C}(\mathbf{B_i})$)
17: **end while**
18: return $\mathbf{M^{ac}_{BS}}$

The algorithm shows broker's matching process between buyer's requirements and seller's offers to seek the final matching pairs based on buyer's feedbacks from the determined matching results. Firstly, a broker receives buyer's requirements and seller's offers (Line 1). The output of the algorithm returns the final matching pairs between buyers and sellers (Line 2). Based on buyer's requirements and seller's offers, a broker carries out to match between buyers and sellers by using the objective function in Equation (12) and a set of constraints in Equation (13-17) to achieve matching results (Lines 3-4). Then, a broker sends matching results to buyers to determine whether buyers accept the matching results by using *'check'* function (Line 5). If there exists buyers which do not accept results, a broker will update buyer's requirements through *'update'* function (Line 6). The function *'stopCriterion'* (Line 7) will terminate broker's matching process when (1) all buyers accept matching results or (2) the matching results of previous loop is as same as that of current loop. If the function *'stopCriterion'* returns *'false'*, a broker continues to carry out its matching process. In the stage, before a broker carries out its matching process, a broker has to remove all buyers and sellers which accepts matching results in previous stage (Lines 8-12). After determining matching results again using *'match'* function (Line 13), a broker sends matching results to buyers to determine whether buyers accepts matching results (Line 14). If buyers accept matching results, a broker will update matching results (Line 15). If there exists buyers which do not accept matching results, a broker will update buyer's constraints to carry out broker's next matching process (Line 16). If The function *'stopCriterion'* returns *'true'*, a broker terminates its matching process and returns final matching results (Line 18).

4. EXPERIMENTS

In this section, we present our experimental results and analyse our matching approach's performance. The experiments focus mainly on testing maximizing buyer's total satisfaction degree through matching between buyer's requirements and seller's offers in business environments. The rest of this section is divided into two subsections. Section 4.1 describes the experimental setting that has been applied in the experiments. Section 4.2 shows the experimental results and performance analysis in three different experiments.

4.1 Experimental setting

In the experiments, we generate an artificial data of 10 buyers and 30 sellers related to car's demand in Australia. Each car in buyer's requirements and seller's offers contains five attributes, i.e., *make* (a_1), *model* (a_2), *price* (a_3), *warranty time* (a_4), *delivery time* (a_5). As per buyer's view, *make* attribute (a_1), *model* attribute (a_2) are attributes with hard constraints because their constraints must be satisfied while two attributes with soft constraints are *warranty time* (a_4), *delivery time* (a_5) and an attribute with fuzzy information is *price* (a_3).

Assume that *price* (a_3) is left semi-trapezoidal fuzzy number through communications between a broker and buyers and each buyer accepts broker's matching results based on seller's offered price. If buyer's satisfaction degree, which is determined from buyer's membership function as per seller's offered price, is more than buyer's target satisfaction degree for *price* attribute, buyers accept broker's matching results. Otherwise, buyers send buyer's requirements to a broker so that a broker can seek other sellers to satisfy buyer's requirements. Furthermore, the constraint values of attributes in buyer's requirements and seller's offers, and weights of attributes in buyer's requirements for our simulation experiments were automatically generated because we had no real buyers and sellers with real data. We automatically generated these values based on data information from website of carsales (www.carsales.com.au).

In this experiment, the proposed approach is evaluated under seller's market so the three different experiments include a number of different selected sellers. More specifically, a broker's matching approach is tested in three different experiments presented in Table 1 to maximize buyer's total satisfaction degree under different sellers.

4.2 Experimental results and analysis

In experiment 1, a broker uses the proposed matching to

Table 1: Different experiments

Experiment	Test purpose
1	To maximize buyer's total satisfaction degree with 10 buyers and 10 sellers
2	To maximize buyer's total satisfaction degree with 10 buyers and 20 sellers
3	To maximize buyer's total satisfaction degree with 10 buyers and 30 sellers

maximize buyer's total satisfaction degree through finding out allocations between buyers and sellers under considering that a number of buyers (10 buyers) equals to a number of sellers (10 sellers) in the market. In general principle of markets, when buyer's demand equals to seller's supply, it is difficult for a broker to find out potential sellers to satisfy all buyer's requirements. Furthermore, it is difficult for buyers to obtain their high satisfaction degree because a broker has a fewer opportunity to choose seller's offers to satisfy buyer's requirements. The results of buyer's satisfaction degree in experiment 1 are presented in Figure 7 and the matching results are also presented in Table 2.

Table 2: Optimal matching pairs with the three different scenarios

	Experiment 1	Experiment 2	Experiment 3
1	$B_1 \longleftrightarrow S_1$	$B_1 \longleftrightarrow S_{19}$	$B_1 \longleftrightarrow S_{25}$
2	$B_2 \longleftrightarrow S_{10}$	$B_2 \longleftrightarrow S_{11}$	$B_2 \longleftrightarrow S_{28}$
3	$B_3 \longleftrightarrow S_8$	$B_3 \longleftrightarrow S_{14}$	$B_3 \longleftrightarrow S_{23}$
4	$B_4 \longleftrightarrow S_2$	$B_4 \longleftrightarrow S_{12}$	$B_4 \longleftrightarrow S_{27}$
5	$B_6 \longleftrightarrow S_4$	$B_5 \longleftrightarrow S_{16}$	$B_5 \longleftrightarrow S_{15}$
6	$B_8 \longleftrightarrow S_9$	$B_6 \longleftrightarrow S_{17}$	$B_6 \longleftrightarrow S_{26}$
7	$B_9 \longleftrightarrow S_7$	$B_7 \longleftrightarrow S_8$	$B_7 \longleftrightarrow S_{22}$
8	$B_{10} \longleftrightarrow S_3$	$B_8 \longleftrightarrow S_{15}$	$B_8 \longleftrightarrow S_{21}$
9		$B_9 \longleftrightarrow S_{18}$	$B_9 \longleftrightarrow S_{24}$
10		$B_{10} \longleftrightarrow S_{20}$	$B_{10} \longleftrightarrow S_{18}$
	$f = 0.856$	$f = 0.915$	$f = 0.976$

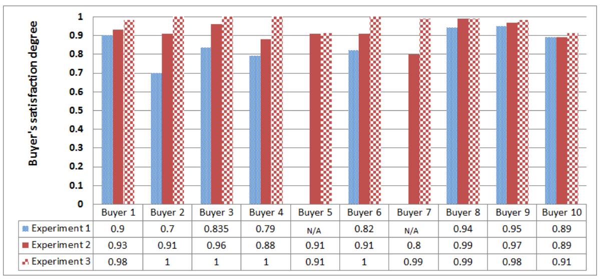


Figure 7: Buyer's satisfaction degree in Experiments 1, 2 and 3

Based on Figure 7 and Table 2, it is clear that there are eight satisfied buyers including B_1, B_2, B_3, B_4, B_6, B_8, B_9 and B_{10} while two remaining buyers do not satisfy. Our proposed approach through a broker helps eight buyers to accept the matching results. However, each buyer's satisfaction degree is not high. In particular, buyer's minimal satisfaction degree is 0.7 and buyer's maximal satisfaction degree is 0.95. Furthermore, eight satisfied buyer's normalized total satisfaction degree in experiment 1 is not high (0.856) because a number of sellers equals to a number of buyers in the market.

Similarly, in experiment 2, a broker considers that a number of sellers (20) is twice as equal as a number of buyers (10). Based on Figure 7 and Table 2, it can be seen that 10 buyers are also satisfied and the matching results are also found for each buyer. More specifically, buyer's minimal satisfaction degree is 0.8 and buyer's maximal satisfaction degree is 0.99. Furthermore, buyer's normalized total satisfaction degree in experiment 2 is relative high (0.915) and is higher than buyer's normalized total satisfaction degree (0.856) in experiment 1 because a broker has many opportunities to select seller's offers which satisfy buyer's requirements and increase buyer's total satisfaction degree.

Finally, a number of sellers (30) is triple as equal as a number of buyers (10) in experiment 3. Based on Figure 7 and Table 2, it is clear that except buyer's satisfied requirements, buyer's normalized total satisfaction degree is very high (0.976) and higher than buyer's normalized total satisfaction degree (0.856) in experiment 1 and buyer's normalized total satisfaction degree (0.915) in experiment 2 because a broker in experiment 3 is more opportunity to select seller's offers which satisfy buyer's requirements than in experiment 1 and 2.

In summary, the proposed approach is perfectly performed under different situations in business environments. In general, if seller's supply is more than buyer's demand, a broker has many opportunities to choose seller's offers to satisfy buyer's requirements and increase each buyer's satisfaction degree as well as buyer's total satisfaction degree.

5. RELATED WORK

There has been a lot of previous work on regarding the indirect interaction between buyer agents and seller agents through intermediaries or broker agents in e-markets. Tsai et al. [13] proposed the establishment of a fuzzy system which acts an intermediary to perform the matching operation between buyer's requests and seller's offers. After the matching system receives buyer's requests and seller's offers, the matching system calculates and selects the highest total evaluation value to offer a buyer. If the buyer accepts the recommended bid offer, the matching process is terminated; otherwise, an adjustment in seller's bid offers is required, and the aftermath negotiations is happened. By comparing with Tsai's work, the advantages of our work are that our proposed approach considered multi-attribute trading including fuzzy attributes and other attributes to carry out the matching process. However, we did not consider the aftermath negotiations in this paper.

Jiang et al. [7] proposed a matching approach based on a bi-objective function to optimize the trade matching in multi-attribute exchanges with incomplete weight information through electronic brokerages (E-brokerages). In particular, the bi-objective optimization function is to maximize the matching degree and trading volume. The difference between Jiang's work and our work is that a broker in our approach uses the point estimation method [11] to communicate with buyers to model buyer's requirements, which are related to attributes with fuzzy information. After modeling buyer's requirements, broker's matching process is carried to satisfy buyer's requirements and maximize buyer's total satisfaction degree while Jiang et al. [7] only considers attributes with crisp values in buyer's requirements.

Li et al. [10] proposed two objective optimization functions to match buyers and sellers in B2B e-marketplace through a matchmaker. The first and second objective optimization function are to maximize the total satisfaction of buyer and seller, respectively. Furthermore, they used priority based multi-objective genetic algorithm to find out optimal matching solutions. By comparing with Li's work, the advantages of our work are that our approach proposed broker's strategy for allocations between buyers and sellers based on buyer's feedbacks from the matching results.

6. CONCLUSION AND FUTURE WORK

This paper proposes the matching approach between buyer's requirements and seller's offers through a broker in multi-attribute trading. The proposed approach is novel because (1) a broker uses the point estimation method to model buyer's requirements for attributes with fuzzy information through communications between a broker and buyers; (2) based on buyer's requirements related to attributes with fuzzy information and seller's offers, broker's objective function and a set of constraints are generated to carry out allocations between buyers and sellers so that buyer's requirements are satisfied and buyer's total satisfaction degree are maximized; and (3) based on buyer's feedbacks from matching results, broker's strategy for allocations between buyers and sellers is proposed to satisfy buyer's requirements. The experimental results demonstrate the good performance for the proposed approach in aspects of satisfying buyer's requirements and buyer's total satisfaction degree.

Future research includes extending the proposed matching approach to solve competition business environments between brokers and we intend to design a decision support system for a broker based on a web-based business environment, in which the proposed matching approach is applied.

7. REFERENCES

[1] R. Fletcher. *Practical methods of optimization.* John Wiley & Sons, 2013.

[2] S. Gamvroulas, D. Polemi, and M. Anagnostou. A secure brokerage network for retail banking services. *Future generation computer systems*, 16(4):423–430, 2000.

[3] G. M. Giaglis, S. Klein, and R. M. O'Keefe. The role of intermediaries in electronic marketplaces: developing a contingency model. *Information Systems Journal*, 12(3):231–246, 2002.

[4] S. H. Ha and S. C. Park. Matching buyers and suppliers: An intelligent dynamic-exchange model. *IEEE Intelligent Systems*, 16(4):28–40, 2001.

[5] J. Hands, M. Bessonov, M. Blinov, A. Patel, and R. Smith. An inclusive and extensible architecture for electronic brokerage. *Decision Support Systems*, 29(4):305–321, 2000.

[6] M. He, N. R. Jennings, and H.-F. Leung. On agent-mediated electronic commerce. *IEEE Transactions on Knowledge and Data Engineering*, 15(4):985–1003, 2003.

[7] Z. Z. Jiang, Z. P. Fan, C. Tan, and Y. Yuan. A matching approach for one-shot multi-attribute exchanges with incomplete weight information in e-brokerage. *International Journal of Innovative Computing, Information and Control*, 7(5):2623–2636, 2011.

[8] Z. Z. Jiang, C. Tan, X. Chen, and Y. Sheng. A multi-objective matching approach for one-shot multi-attribute exchanges under a fuzzy environment. *International Journal of Fuzzy Systems*, 17(1):53–66, 2015.

[9] R. Khosla and S. Kitjongthawonkul. A human-centered agent-based architecture for electronic brokerage. *Soft computing*, 5(6):405–411, 2001.

[10] X. Li and T. Murata. Priority based matchmaking method of buyers and suppliers in b2b e-marketplace using multi-objective optimization. In *Proceedings of the International MultiConference of Engineers and Computer Scientists*, volume 1, 2009.

[11] A. Sancho-Royo and J. L. Verdegay. Methods for the construction of membership functions. *International Journal of Intelligent Systems*, 14(12):1213–1230, 1999.

[12] S. Standing, C. Standing, and P. E. Love. A review of research on e-marketplaces 1997–2008. *Decision Support Systems*, 49(1):41–51, 2010.

[13] K. M. Tsai and F. C. Chou. Developing a fuzzy multi-attribute matching and negotiation mechanism for sealed-bid online reverse auctions. *Journal of theoretical and applied electronic commerce research*, 6(3):85–96, 2011.

A Concession Strategy based on Equilibrium of the Final Phase in a Multi-Lateral Negotiation

Akiyuki MORI
Nagoya Institute of Technology
Aichi, Japan
mori.akiyuki@itolab.ni-tech.ac.jp

Takayuki ITO
Nagoya Institute of Technology
Aichi, Japan
ito.takayuki@nitech.ac.jp

ABSTRACT

Multi-lateral Closed Bargaining Games (MCBG) based on the alternating-offer type of bargaining games is an important research topic in the study of Multi-agent Systems. In particular, choosing an appropriate concession strategy is critical in MCBG. Agents have difficulty in deciding upon a concession strategy because negotiation often involves incomplete information and could be played among more than two agents. In this paper, we propose an adaptive concession strategy derived from a strategic-form game-theoretical analysis of the final phase of a MCBG negotiation. We design an adaptive concession function that uses the expected utility as a parameter. Our agent that incorporates the proposed concession strategy successively won the sixth international Automated Negotiating Agents Competition (ANAC2015). The contributions of this paper are the following: First, we develop a general adaptive concession strategy based on a game theory for MCBG. Second, we demonstrated that our proposed concession strategy can work well on tripartite negotiation. Third, our agent outperformed the state-of-the-art negotiating agents and won the overall ANAC2015 competition.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence - Intelligent Agents

General Terms

Algorithms, Experimentation

Keywords

Game Theory, Multi-agent Simulation, Multi-Lateral Closed Bargaining Game, Automated Negotiating Agent

1. INTRODUCTION

The study of automated negotiations typically includes analysis of the negotiation process in alternating-offer bargaining games between two agents [8, 7, 9]. Recently, **Multi-lateral** Closed Bargaining Games (MCBG) based on the alternating-offer type of bargaining games is an important research topic in the study of Multi-agent Systems. Choosing an appropriate concession strategy is critical in MCBG because the negotiation process and result depend on each agent's concession strategy. However, agents cannot adopt a concession strategy based on the other agent's preference information because they do not know it in MCBG. Also, MCBG has additional conditions based on real-world negotiations, such as a time limit, time discount, and reservation value. Therefore, the agents must consider these conditions and estimate an appropriate concession strategy without access to the other agent's preference information.

We analyze the final phase of MCBG as a strategic form game and derive a agent's expected utility at the equilibrium of the evolutionary stable strategy (evolutionary stable state). The evolutionary stable strategy used in evolutionary game theory is an adaptive strategy and cannot be invaded by other strategies [10]. This strategy is adopted to analyze the hawk-dove game in the study of evolutionary game theory. The hawk-dove game is a typical strategic form game that models the competition for survival between aggressive hawks and moderate doves. The evolutionary stable state of the hawk-dove game distributes the equilibrium state when the game is played repeatedly. It has a mixed Nash equilibrium that we can derive from the pay-off matrix of the game. We define two strategies, an uncompromising hawk-like strategy and a compromising dove-like strategy, in the final phase of MCBG. We derive the expected utility of the evolutionary stable state in the final phase of MCBG. We also design an adaptive concession function that contained the expected utility as a parameter. We further demonstrate that our agent implementing our proposed method outperforms state-of-the-art negotiating agents in MCBG, and won the ANAC2015 competition.

There has been extensive investigation of the concession strategy of bargaining games. Faratin et al. [6] proposed a time dependent concession strategy. This concession strategy determines acceptable utility from the concession function that contains time and tuning parameters. This concession strategy must know the opponent's concession strategy in order to set appropriate tuning parameters. However, the agents cannot determine the opponent's concession strategy because they cannot know the utility of the bid that was offered by the opponent in closed bargaining games. Therefore, a agent cannot set appropriate tuning parameters in closed bargaining games. Consequently, existing concession strategy research for closed bargaining games focuses on the uncertainty of the opponent's preference information and, thus, concession strategies based on machine learning are

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), John Thangarajah, Karl Tuyls, Stacy Marsella, Catholijn Jonker (eds.), May 9–13, 2016, Singapore.*

extensively proposed for closed bargaining games. For example, a concession strategy for bilateral was proposed in [5] based on online prediction of opponent behavior. However, they did not take time discount into account. Therefore, their study only assumes a negotiation setting in which hardline strategies obtain high utility. On the other hand, a concession strategy for bilateral was proposed in [11] based on Gaussian processes of opponent behavior for complex negotiation settings that contain a time discount. This concession strategy predicts an opponent's concession from the distribution of the agent's utility of the opponent's bids in closed bargaining games. However, the effectiveness of this machine-learning-based strategy is unstable in closed bargaining games because the stationarity of the distribution depends on the distribution of a feasible set and the opponent's bidding strategy. Furthermore, this concession strategy is limited to Bilateral Closed alternating-offer Bargaining Games (BCBG). Research on an efficient concession strategy for MCBG is therefore incomplete.

To evaluate our proposed concession strategy, we employed the Sixth International Automated Negotiating Agents Competition (ANAC2015) [3, 4]. The ANAC competitions allow us to evaluate the effectiveness of concession strategies empirically under various negotiation scenarios and conditions. The Stacked Alternating Offers Protocol for Multi-lateral Negotiation (SAOPMN) [2], which is the negotiation protocol for MCBG, was used at ANAC2015. The negotiation setting for ANAC2015 contained many negotiation conditions, such as time limit, time discount, and reservation value.

In this paper, we propose an efficient concession strategy for MCBG. [1] proposed a concession strategy for **bilateral** negotiations based on the expected utility of the final phase. However, in general, there could be more than 3 participants in a negotiation. Thus, we recreate an efficient concession strategy for **multi-lateral** negotiations. Here, we regard more than two opponents as one group in order to analyze the final phase of MCBG.

Our concession strategy can provide beneficial decisions to MCBG agents. In this paper, we:

- develop our concession strategy based on equilibrium of the final phase in MCBG;
- evaluate the effectiveness of our concession strategy using **tripartite** negotiation simulation of automated negotiating agents; and
- demonstrate that our agent, using our concession strategy, outperforms the state-of-the-art negotiating agents and won the competition.

The remainder of this paper is organized as follows: We set forth the negotiation setting of this paper in Section 2, describe our proposed concession strategy for MCBG in Section 3, provide the results of experiments on negotiation simulations with ANAC2015 agents in Section 4, and, finally, present our conclusions and future work in Section 5.

2. THE NEGOTIATION SETTING

We adopt the negotiation setting of ANAC2015 to evaluate our concession strategy.

Negotiation Domain.

A negotiation domain is defined by the issues and their values of the negotiation. The bargaining games in this paper contain multiple issues and agents can access to the perfect information of the negotiation domain. A set of agreement candidates $\mathcal{S}$ is given by:

$$\begin{aligned} \mathcal{S} \;=\; & \{\mathbf{s} = (s_{i_1}, s_{i_2}, \cdots, s_{i_n}) \in \mathbb{N}^n \mid \\ & 0 \leq s_{i_k} \leq v_{i_k}, k = 1, 2, \cdots, n\}, \end{aligned} \tag{1}$$

where $\mathbf{s} \in \mathcal{S}$ is a bid that is an agreement candidate, $\mathcal{I}$ is a set of issues, n is the issue number, and s_{i_k} is the integer value for issue i in the bid $\mathbf{s}$ ($s_{i_k} \in [0, v_{i_k}]$, where v_{i_k} is the maximum integer value of s_{i_k}).

Utility Function.

A utility function is defined by the agent's preference information. We assume a linear additive utility function $U(\cdot)$ of the form:

$$U(\mathbf{s}) = \sum_{k=1}^{n} \beta_{i_k} \cdot u_{i_k}(s_{i_k}), \tag{2}$$

where $\mathbf{s}$ is a bid, $u_{i_k}(\cdot)$ is an evaluation function for issue i_k, and β_{i_k} is the issue's weight.

Negotiation Protocol.

A negotiation protocol is defined as the negotiation process for agents. In this paper, we assume a negotiation as a Stacked Alternating Offers Protocol for Multi-lateral Negotiation (SAOPMN) [2], which is a multi-lateral closed negotiation protocol and adopted in the ANAC2015. In SAOPMN, agents can select one of the following three actions:

Offer **:** proposing a bid to opponent agents.

Accept **:** agreeing to the offered bid. When all agents have selected *Accept*, all agents get the utility, which is defined by its own utility function with a time discount. If a agent wants to reject a bid, the agent makes a counter offer to the opponent agents.

EndNegotiation **:** ending the negotiation without any agreement. In SAOPMN, a negotiation is closed when one agent selects *EndNegotiation*. The agents receive a reservation value with a time discount when the negotiation is closed.

The action order of the players is decided before negotiation begins. For example, consider a SAOPMN negotiation between agents A, B, and C. The action order of the agents is agent $A \to B \to C \to A \to \cdots$ in this example. The negotiation of SAOPMN continues as with the following example. First, agent A offers $\mathbf{s}_1$ to agents B and C. After the action of agent A, agent B can select an action. If agent B selects *Offer*, agent B offers $\mathbf{s}_2$ to agents A and C instead of $\mathbf{s}_1$. If agent B selects *Accept*, agent B declares acceptance of $\mathbf{s}_1$ to agents A and C. If agent B selects *EndNegotiation*, the negotiation is closed. After the action of agent B, agent C can select an action. If agent C selects *Offer*, agent C offers $\mathbf{s}_3$ to agents A and B instead of $\mathbf{s}_1$ or $\mathbf{s}_2$. If agent C selects *Accept*, agent B declares acceptance of $\mathbf{s}_1$ or $\mathbf{s}_2$ to agents A and C. If $\mathbf{s}_1$ is accepted by agent C, in this case, $\mathbf{s}_1$ is the agreement to this negotiation. If agent C selects *EndNegotiation*, the negotiation is closed. After the action of agent C, agent A can select an action again.

Time Pressure.

We assume a time limit for negotiations because many real-life negotiations must be resolved within a finite time. When the negotiating time exceeds the time limit, agents select *EndNegotiation*. Moreover, all agents are aware of the time limit, which is the same for all of them. Furthermore, agents can make offers in real-time.

Time Discount.

We assume a time discount for negotiations because many real-life negotiations require early resolution. In many cases, the utility of an agreement is low if the negotiation time is too long. The discounted utility function $D(\cdot,\cdot)$ is given by:

$$D(u,t) = u \cdot d^{t/t_{\max}}, \quad (3)$$

where u is the undiscounted utility of an agreement, d ($0 < d \leq 1.0$) is the discounting factor, t is the agreement time, and $t_{\max}$ is the deadline. Furthermore, the time discount is applied to the reservation value.

Reservation Value.

We assume a reservation value for negotiations because a agent's utility is guaranteed in some real-life negotiations. For example, if the reservation value of agent A is u_{r}^A, then agent A can get $D(u_{\mathrm{r}}^A, t)$ if the negotiation fails.

3. CONCESSION STRATEGY DESIGN

3.1 Agreement Failure Risk in MCBG

We focus on the final phase of MCBG in order to decide the appropriate concession value. The final phase is a conceptual period when the negotiation might finish after the current action. The agent's risk is special in the final phase when the agent rejects the offered bid. In the final phase, if the agent rejects the offered bid, the agent loses the utility of agreement because agreement failure in the final phase means the negotiation has failed. The negotiation failure loss $L = D(U(\mathbf{s}_c), t_{\mathrm{f}}) - D(u_{\mathrm{r}}, t_{\mathrm{f}})$ where t_{f} is the final phase time, u_{r} is the reservation value, and $\mathbf{s}_c$ is the agreement when the agent compromises for opponents. Agents cannot know t_{f} under the real-time constraints. If $U(\mathbf{s}_c)$ is sufficiently higher than u_{r}, L exhibits a great loss. For example, when $t_{\mathrm{f}} \simeq t_{\max}$, $d = 0.5$, $u_{\mathrm{r}} = 0.2$, and $U(\mathbf{s}_c) = 0.8$, then, in this case, $L \simeq 0.3$. Thus, negotiations in the final phase of MCBG are very important.

On the other hand, in the negotiation period from the beginning before the final phase, the agents' negotiation failure risk is low. The agent's utility of agreement is decreased by the time discount if the agent rejects the offered bid by the opponents. The discount factor per round $\delta = d^{\epsilon/t_{\max}}$ ($0 < \delta \leq 1.0$) , where ϵ is the interval time per round. ϵ is not fixed because the agent with the right to choose an action can select actions at any time within the time limit under the real-time constraints. When ϵ is low, δ is low. For example, when $\epsilon/t_{max} = 0.01$ and $d = 0.5$, then $\delta = 0.99$. Therefore, because ϵ is low, the agent's agreement failure risk is low. We therefore design our concession strategy based on MCBG's final phase.

3.2 The Equilibrium Analysis in the Final Phase

The situation of final phase resembles the chicken game. Agents must compromise in order to avoid negotiation failure. However, a agent can exploit the other agents if they compromise to avoid negotiation failure. On the other hand, the other agents have the same opportunities. If an opponent makes the same selection, the negotiation fails and the agents lose agreement utility. The structure is not so much an ultimatum game as a chicken game because the agents cannot know the exact time of the final offer under real-time constraints. An optimal solution for a general chicken game is undetermined because it has three mixed strategies Nash equilibriums. This is the most difficult step in designing concession strategy.

Consequently, we use the expected utility in an evolutionary stable state to compromise for uncertain opponents. Specifically, we design an appropriate concession function that contains the expected utility in an evolutionary stable state. Although optimal solutions for the chicken game do not exist, an evolutionary stable state does. The evolutionary stable state is used to derive a stable population distribution in the hawk-dove game. The hawk-dove game has the same game structure as the chicken game. In this paper, we assume that a agent can obtain the expected utility of an evolutionary stable state in the final phase to design the concession function.

We assume that agents can select two strategies in the final MCBG phase to analyze the final phase game. We classify the strategies of the final phase into the following two strategies:

Uncompromising Strategy (US): the acceptable utility does not relax until the end of the negotiation.

Compromising Strategy (CS): the acceptable utility relaxes until just before the negotiation's end in order to obtain the agreement utility.

The game is structured so that agents can avoid the negotiation failure when all the agents select CS. Although an agent who selects US can exploit the other agents, the negotiation will fail if more than two agents select US. We analyze the final phase game similarly to a hawk-dove game because the game structure of the final phase and the hawk-dove game are the same. The hawk-dove game models the competition for survival between aggressive hawks and moderate doves as a bilateral strategic form game in evolutionary game theory. If both players are doves, they divide their resources peacefully. However, if either player is a hawk, the hawk player can force the dove out and monopolize the resources. Although the hawk has more power than the dove, if both players are hawks, they will both be injured in a fight. This disadvantage is greater than the disadvantage of sharing resources. In the final phase game, agents who select US are hawks and agents who select CS are doves. However, the hawk-dove game cannot be directly applied to the final phase game in MCBG if there are more than three players. Therefore, we treat opponents as a single group of players. We do not need to consider each opponent's strategy because the negotiation will fail in SAOPMN regardless of strategy if more than two opponents select US. SAOPMN requires a concession strategy when only one opponent selects US or all opponents select CS. Thus, the game model must be appropriately created with the negotiation protocol, SAOPMN.

Table 1 shows the payoff matrix of MCBG's final phase. We evaluate A_{11}, A_{12}, A_{21}, and A_{22} based on our preference and negotiation information. Our evaluated utility,

Table 1: Payoff Matrix of the Final Phase Game

Our agent \ Opponents	US	CS
US	(A_{11}, B_{11})	(A_{12}, B_{12})
CS	(A_{21}, B_{21})	(A_{22}, B_{22})

A_{11}, A_{12}, A_{21}, and A_{22}, are defined by:

$$A_{11} = D_A(u_{\mathrm{r}}^A, t_{\mathrm{f}}) \tag{4a}$$

$$A_{12} = D_A(u_{\max}^A, t_{\mathrm{f}}) \tag{4b}$$

$$A_{21} = D_A(U_A(\mathbf{s}_{\mathrm{c}}^A), t_{\mathrm{f}}) \tag{4c}$$

$$A_{22} = p \cdot A_{12} + (1 - p) \cdot A_{21}, \tag{4d}$$

where $U_A(\cdot)$ is our utility function, $D_A(\cdot,\cdot)$ is our discounted utility function, u_{r}^A is our reservation value, $u_{\max}^A$ is our maximum utility, $\mathbf{s}_{\mathrm{c}}^A$ is the agreement when the agent compromises for the opponents, and p is the probability that our agent will compromise after the opponents' concession when our agent and the opponents select CS.

In formula (4a), if our agent and the opponents select US, the negotiation will fail because the agents failed to compromise before the negotiation ended. Therefore, A_{11} is the reservation value with time discounts. In formulas (4b) and (4c), if CS is selected by either side, but not both, the agent that selects CS is exploited. Therefore, the utility of the agent selecting CS is $U_A(\mathbf{s}_{\mathrm{c}}^A)$ with a time discount and the utility of the agent selecting US is the maximum utility with a time discount. In formula (4d), if both sides select CS, the agent who compromises first is exploited. Therefore, the agents compete in the concession time to exploit their opponent. If they attempt to compromise with each other after their opponent compromises, determining which agent compromised first is difficult. Consequently, we define p to evaluate A_{22} in this paper.

Our agent's expected utility function $F_A(\cdot,\cdot)$ is given by:

$$\begin{aligned} F_A(\mathbf{p}_A,\mathbf{p}_B) &= p_A \cdot p_B \cdot A_{11} + p_A \cdot (1-p_B) \cdot A_{12} \\ &\quad + (1-p_A) \cdot p_B \cdot A_{21} + (1-p_A) \cdot (1-p_B) \cdot A_{22} \\ &= p_A \cdot \{p_B \cdot (A_{11}-A_{21}) + (1-p_B) \cdot (A_{12}-A_{22})\} \\ &\quad + p_B \cdot A_{21} + (1-p_B) \cdot A_{22}, \end{aligned} \tag{5}$$

where $\mathbf{p}_A = (p_A, 1 - p_A)$ is our mixed strategy, p_A is our probability of selecting US, $\mathbf{p}_B = (p_B, 1 - p_B)$ is the opponents' mixed strategy, and p_B is the opponents' probability of selecting US. We derive the expected utility in the evolutionary stable state from formula (5). In this game, we cannot derive our evolutionary stable strategy $\mathbf{p}_A^* = (p_A^*, 1-p_A^*)$ because $\mathbf{p}_A^*$ depends on B_{11}, B_{12}, B_{21}, and B_{22}, which are based on the opponents' unknown preferences and negotiation information. However, we can determine the opponents' evolutionary stable strategy $\mathbf{p}_B^* = (p_B^*, 1 - p_B^*)$ because $\mathbf{p}_B^*$ depends on A_{11}, A_{12}, A_{21}, and A_{22}. Thus, we derive our expected utility function in the evolutionary stable state $F_A(\mathbf{p}_A^*, \mathbf{p}_B^*)$ from $\mathbf{p}_B^*$. In formula (5), if $p_B \cdot (A_{11} - A_{21}) + (1 - p_B) \cdot (A_{12} - A_{22}) = 0$, then $\mathbf{p}_B$ equals the opponents' evolutionary stable strategy $\mathbf{p}_B^*$. We can derive the expected utility in the evolutionary stable state $F_A(\mathbf{p}_A^*, \mathbf{p}_B^*)$ by only $\mathbf{p}_B^*$ without $\mathbf{p}_A$ and $\mathbf{p}_A^*$. Here, $F_A(\mathbf{p}_A^*, \mathbf{p}_B^*)$ is given by:

$$F_A(\mathbf{p}_A^*, \mathbf{p}_B^*) = p_B^* \cdot A_{21} + (1.0 - p_B^*) \cdot A_{22}. \tag{6}$$

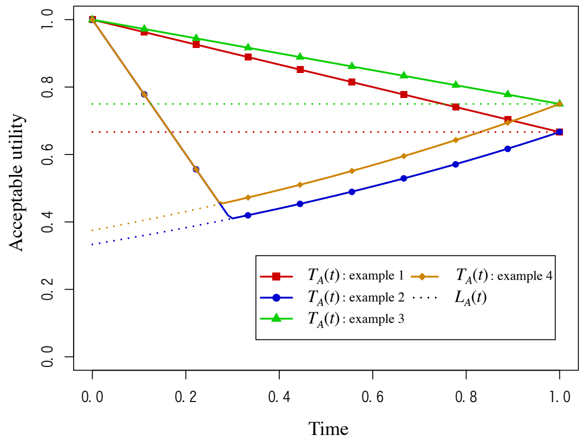


Figure 1: Graph of Concession Function $T_A(t)$

3.3 Designing the Concession Function

We design the concession function for MCBG based on $F_A(\mathbf{p}_A^*, \mathbf{p}_B^*)$. The concession function defines the threshold value of acceptable utility. If the bid utility exceeds concession function value, the agents accept or offer the bid. Our concession function has a lower limit $L_A(t)$, which is our discounted expected utility function in the evolutionary stable state. $L_A(t)$ is given by:

$$L_A(t) = F_A(\mathbf{p}_A^*, \mathbf{p}_B^*)/d_A^t, \tag{7}$$

where t is the current time and d_A is our discounted factor. The value of the concession function decreases from maximum utility to $L_A(t)$. We design a concession function that depends on the time and has a linear sloop. Furthermore, quick agreement is important to reduce the discounted utility when $d < 1.0$. Therefore, our concession function uses multiple concession functions in accordance with the discount factor. Our concession function $T_A(t)$ is given by:

$$T_A(t) = \begin{cases} L_A(t) + (u_{\max}^A - L_A(t)) \cdot (t_{\max} - t)/t_{\max} & (d_A = 1.0) \\ u_{\max}^A \cdot (1.0 - t/\alpha) & (d_A < 1.0 \land u_{\max}^A \cdot (1.0 - t/\alpha) \geq L_A(t)) \\ L_A(t) & (d_A < 1.0 \land u_{\max}^A \cdot (1.0 - t/\alpha) < L_A(t)), \end{cases} \tag{8}$$

where α is the adjusting parameter to compromise. If $0.0 < df < 1.0$, $T_A(t)$ is minimum utility 0 when $t = \alpha$. However, $T_A(t) = L_A(t)$ when $T_A(t) < L_A(t)$. Figure 1 shows the $T_A(t)$ in cases 1, 2, 3, and 4.

example 1 : $d_A = 1.0, u_{\mathrm{r}}^A = 0.00, U_A(\mathbf{s}_{\mathrm{c}}^A) = 0.5$

example 2 : $d_A = 0.5, u_{\mathrm{r}}^A = 0.00, U_A(\mathbf{s}_{\mathrm{c}}^A) = 0.5, \alpha = d_A$

example 3 : $d_A = 1.0, u_{\mathrm{r}}^A = 0.75, U_A(\mathbf{s}_{\mathrm{c}}^A) = 0.5$

example 4 : $d_A = 0.5, u_{\mathrm{r}}^A = 0.75, U_A(\mathbf{s}_{\mathrm{c}}^A) = 0.5, \alpha = d_A$

If $df > 0$ and $RV > 0$ are as in case 4, $T_A(t)$ can be smaller than $RV_D(t)$. In this case, our concession strategy selects *EndNegotiation*.

4. EMPIRICAL EVALUATION

4.1 Experimental Settings

We submitted the agent implementing our proposed concession strategy to ANAC2015. 24 agents were submitted for ANAC2015's qualifying tournament. The agents were divided into four groups for round-robin evaluation in the qualifying round. The negotiation settings of ANAC2015 are same as the negotiation settings of this paper. Table 2 shows the number of issues and negotiating conditions of 10 bargaining problems from ANAC2015. Each bargaining problem has three profiles that define the agent's utility function. Discount factor d and reservation value u_{r} have the same value for all agents. A negotiation agent must compromise appropriately in order to get higher utility because each bargaining problem's conflict level is different. The individual utility and Nash product for the agents were evaluated at ANAC2015. An agent's individual utility is high when that agent reaches a desirable agreement and the agent's Nash product is high when the individual utility of all agents is high. The top two agents from each group moved on to ANAC2015's final round. Table 3 shows the finalists. 13 agents were chosen from the qualifying round. Our agent, agentBuyog, and RandomDance advanced to the final round of both departments. In the final round, the finalists were evaluated in a round-robin tournament by department. The evaluated agents used all preference information five times in all combinations with a three-minute negotiation time limit.

4.2 Setting of Our Agent for ANAC2015

Our agent's settings for ANAC2015 were as follows: Our agent has to estimate $\mathbf{s}_{\mathrm{c}}$ that is the agreement when the agent compromises for opponents when selecting CS that is the acceptable utility relaxes until just before the negotiation's end; our agent assumes the bid that has the highest utility in list L_{c} as $\mathbf{s}_{\mathrm{c}}$; when the bid is accepted by other agents, our agent records the bid to list L_{c}; and, if any bid is not accepted by the other agents, then our agent regards $U(\mathbf{s}_{\mathrm{c}})$ that is utility of $\mathbf{s}_{\mathrm{c}}$ as reservation valueu_{r}.

In the final phase, it is not possible for our agent to derive the evolutionary stable strategy because it depends on a closed opponent preference. Therefore, our agent always selects CS in the final phase. The time of final phase t_{f} is defined by:

$$t_{\mathrm{f}} = 1.0 - h/t_{\max} \cdot (size(L_{\mathrm{c}}) + C'), \tag{9}$$

Table 2: The Number of Issues and Negotiating Conditions of the ANAC2015 Bargaining Problems

Domain	Number of Issues	Number of Bids	d	u_{r}
1	1	5	1.0	0.5
2			1.0	0.5
3	2	25	0.2	0.0
4			1.0	0.5
5	4	320	0.5	0.0
6			0.5	0.0
7	8	3^8	1.0	0.0
8			1.0	0.0
9	16	2^{16}	0.4	0.7
10			0.4	0.7

Table 3: Departmental ANAC2015 Agent Finalists

Agent Name	
Individual Utility	Nash Product
Our Agent	**Our Agent**
agentBuyog	agentBuyog
RandomDance	RandomDance
ParsAgent	Mercury
kawaii	JonnyBlack
PhoenixParty	AgentX
XianFaAgent	CUHKAgent
PokerFace	AgentH

where h is the average time our agent's turn comes around, $size(\cdot)$ is the size of the list, and C' is the error adjustment variable. We set $C' = 1$ for ANAC2015. When our agent selects CS, our agent offers a bid of L_{c} according to the utility. If our agent fails to agree on all bids of L_{c}, our agent accepts the bid that has a greater utility than u_{r}.

Our agent calculates formulas (4a), (4b), and (4c) as $t_f = 1.0$ in ANAC2015; however, $t_f \simeq 1.0$ in ANAC2015 because the average time our agent's turn comes around, h, is very small in negotiation with automated negotiating agents. Therefore, its impact on the evaluation of our proposed strategy is insignificant. Furthermore, we set $\alpha = d_A$, which is the variable of $T_A(t)$, empirically because α must be small when d_A is small. Moreover, we set $p = 0.5$, which is the variable of formula (4d).

4.3 Competition Results

Our agent won both departments of the ANAC2015 competition. Table 4 shows the individual utility and Table 5 shows the Nash product of the final round of the ANAC2015. As shown in Tables 4 and 5, our agent's individual utility and Nash product are the highest of the simulated agents. Therefore, it achieved appropriate concessions in comparison with the state-of-the-art negotiating agents.

Since the simulation log of ANAC2015 has not been published, we carried out additional experiments to analyze our strategy. We simulated a selfplay tournament to evaluate each agent's characteristics. The selfplay tournament is a negotiation played between three agents with the same strategy. In the round-robin tournament, agents using a hardline strategy can exploit agents using a compromise strategy. However, agents using a hardline strategy cannot get the beneficial agreement in the selfplay tournament because the

Table 4: Results of Individual Utility in the Final Round of ANAC2015

Agent Name	Average	SD
Our Agent	**0.481**	**0.162**
ParsAgent	0.471	0.143
RandomDance	0.461	0.198
kawaii	0.460	0.152
agentBuyog	0.459	0.227
PhoenixParty	0.443	0.212
XianFaAgent	0.353	0.216
PokerFace	0.344	0.303

Table 5: Results of Nash Product in the Final Round of ANAC2015

Agent Name	Average	SD
Our Agent	**0.324**	**0.000405**
Mercury	0.322	0.00162
JonnyBlack	0.314	0.00103
AgentX	0.312	0.00139
CUHKAgent	0.309	0.00173
RandomDance	0.295	0.00109
AgentH	0.292	0.00155
agentBuyog	0.282	0.00236

Table 6: Results of Individual Utility and Lost Utility by Time Discount (Selfplay)

Agent Name	Individual Utility		Lost Utility	
	Average	SD	Average	SD
JonnyBlack	0.749	0.200	0.0314	0.0763
Our Agent	**0.716**	**0.170**	**0.0357**	**0.0405**
AgentX	0.691	0.182	0.0261	0.0421
Mercury	0.680	0.224	0.0664	0.0917
AgentH	0.617	0.290	0.161	0.210
CUHKAgent	0.605	0.291	0.0788	0.102
XianFaAgent	0.548	0.382	0.112	0.313
RandomDance	0.545	0.259	0.318	0.415
PhoenixParty	0.530	0.196	0.306	0.402
PokerFace	0.479	0.361	0.291	0.423
ParsAgent	0.418	0.296	0.186	0.315
kawaii	0.382	0.331	0.112	0.313

agents' bids are self-interested. We used the 10 bargaining problems from the ANAC2015 for the selfplay tournament.

Table 6 shows the individual utility and lost utility by time discount of the selfplay tournament. This result shows that our agent can cooperate in order to obtain a beneficial agreement quickly. Our agent's strategy is superior to that of JonnyBlack and AgentX because these agents lost in the ANAC2015 individual department. On the other hand, our agent won in both departments of ANAC2015. It means that JonnyBlack and AgentX are almost cooperative to get a beneficial agreement quickly. These agents only work efficiently when their opponents are also cooperative. As seen in Table 6, in selfplays, our agent's individual utility is the second highest next to JonnyBlack and our agent's lost utility by time discount is the third lowest in the order of JonnyBlack, AgentX, and our agent.

5. CONCLUSIONS AND FUTURE WORK

In this work, we developed a negotiation strategy based on equilibrium of MCBG. We demonstrated our strategy's performance in MCBG containing multiple issues, uncertainty, real-time constraints, a time discount, and a reservation value. Furthermore, we established that our agent implementing our proposed concession strategy outperformed the other state-of-the-art agents at ANAC2015.

In the future, we will improve the slope of the concession strategy. Here, we adopted a linear concession function. However, we need to evaluate more varied concession functions, such as a nonlinear concession function. Moreover, to prove the efficiency of our strategy in MCBG, we will experiment with other negotiation conditions, more than four agents, and various discount factors and reservation values.

REFERENCES

[1] Anonymous. A compromising strategy based on expected utility of evolutionary stable strategy in bilateral closed bargaining problem. In *Proceedings of Agent-based Complex Automated Negotiations (ACAN) 2015*, pages 58–65, 2015.

[2] R. Aydogan, D. Festen, K. V. Hindriks, and C. M. Jonker. Alternating offers protocol for multilateral negotiation. In *Modern Approaches to Agent-based Complex Automated Negotiation*. Springer, To be appeared.

[3] T. Baarslag, R. Aydogan, K. V. Hindriks, K. Fuijita, T. Ito, and C. M. Jonker. The automated negotiating agents competition 2010-2015. *AI Magazine*, 2016.

[4] T. Baarslag, K. Fujita, E. Gerding, K. V. Hindriks, T. Ito, N. R. Jennings, C. M. Jonker, S. Kraus, R. Lin, V. Robu, and C. Williams. The first international automated negotiating agents competition. *Artificial Intelligence Journal (AIJ)*, 2012.

[5] J. Brzostowski and R. Kowalczyk. Adaptive negotiation with on-line prediction of opponent behaviour in agent-based negotiations. In *Proceedings of the IEEE/WIC/ACM international conference on Intelligent Agent Technology*, pages 263–269. IEEE Computer Society, 2006.

[6] P. Faratin, C. Sierra, and N. R. Jennings. Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems*, 24(3):159–182, 1998.

[7] M. J. Osborne and A. Rubinstein. Bargaining and markets, 1990.

[8] A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica: Journal of the Econometric Society*, pages 97–109, 1982.

[9] A. Shaked and J. Sutton. Involuntary unemployment as a perfect equilibrium in a bargaining model. *Econometrica: Journal of the Econometric Society*, pages 1351–1364, 1984.

[10] J. M. Smith. *Evolution and the Theory of Games*. Cambridge University Press, 1982.

[11] C. R. Williams, V. Robu, E. H. Gerding, and N. R. Jennings. Using gaussian processes to optimise concession in complex negotiations against unknown opponents. In *Proceedings of the Twenty-second International Joint Conference on Artificial Intelligence (IJCAI2011)*, volume 1, pages 432–438. AAAI Press, 2011.

Using GDL to Represent Domain Knowledge for Automated Negotiations

Dave de Jonge
Western Sydney University
Sydney, New South Wales, Australia
d.dejonge@westernsydney.edu.au

Dongmo Zhang
Western Sydney University
Sydney, New South Wales, Australia
d.zhang@westernsydney.edu.au

ABSTRACT

Current negotiation algorithms often assume that utility has an explicit representation as a function over the set of possible deals and therefore for any deal its utility value can be calculated quickly. We argue however, that a more realistic model of negotiations would be one in which the negotiator has certain knowledge about the domain and must reason with this knowledge in order to determine the value of a proposal, which is time-consuming. We propose to use Game Description Language to model such negotiation scenarios, because this may enable us to apply existing techniques from General Game Playing to implement negotiating agents for such domains.

Keywords

Automated Negotiation; General Game Playing; Game Description Language

1. INTRODUCTION

Most work on Automated Negotiations focuses purely on the strategy to determine which deals to propose *given* the utility values of the possible deals. Little attention has been given to negotiation settings in which determining the utility value of a deal is itself a hard problem and therefore takes a substantial amount of time.

One often assumes the utility value of any deal is known instantaneously, or can be determined by solving a simple linear equation [1]. In such studies the process of evaluating the proposal is almost completely abstracted away and one either assumes that the negotiation algorithms do not require any domain knowledge or reasoning at all, or that all such knowledge is hardcoded in the algorithm. The utility functions of the agent's opponents on the other hand, are often assumed to be completely unknown.

In this paper however, we argue that in real negotiations it is very important to have domain knowledge, and a good negotiator must be able to reason about this knowledge. One cannot, for example, expect to make profitable deals in the antique business if one does not have extensive knowledge of antique, no matter how good one is at bargaining. Moreover, a good negotiator should also be able to reason about the desires of its opponents. A good car salesman for example would try to find out what type of car would best suit his client's needs, in order to increase the chances of coming to a successful deal.

We therefore propose a new kind of negotiation setting in which the agents do not have an explicit representation of their utility functions but instead are presented with domain knowledge in the form of a logic program. Agents will need to apply logical reasoning in order to determine the value of any proposal.

Another point that is rarely taken into account, is that an agent's utility may not always solely depend on the agreements it makes, but may also depend on decisions taken outside the negotiation thread. For example, suppose that you negotiate with a car salesman to buy a car. If you are single and you live in the city then it may be a very good deal to buy a small car which is easy to park and uses little fuel. However, if one year later you get married and decide to start a family, that deal suddenly is not so good anymore because you would now require a larger family car. Moreover, your utility may not only depend on your own actions, but also on actions of other agents. For example, when you make any kind of business deal, you may need to take into account what strategy your competitors will follow in order to determine your own strategy.

We note that these properties we are addressing here: applying logical reasoning about the domain, and choosing a proper strategy with respect to your opponents' strategies, are also the main issues in the field of *General Game Playing* (GGP). General Game Playing deals with the implementation of agents that can play *any* kind of game. In contrast to specialized Chess- or Go- computers, which can only play one specific game and apply large amounts of knowledge introduced by human experts, a GGP program cannot apply any game-specific heuristics or other kinds of human expert knowledge. A GGP program learns the rules of a game only at run-time, which are provided to it in a language called Game Description Language (GDL).

Therefore, in this paper we propose to use GDL to define negotiation domains and common techniques from GGP to implement negotiating agents. We investigate to what extent GDL is applicable to the field of automated negotiations and compare its advantages and disadvantages. We conclude that describing negotiation domains in GDL is indeed possible, but that some small adaptations may need to be made

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

to GDL to make it more suitable for negotiations.

Another advantage of using GDL for negotiations, is that it would allow us to write protocol-independent agents. Currently, negotiating agents are often implemented for one specific protocol. However, if we could specify negotiation protocols in GDL, then one could write a negotiating agent that is able to negotiate under any protocol as long as the protocol is specified in GDL. So far we have indeed managed to specify the Alternating Offers protocol in GDL.

The rest of this paper is organized as follows. In Section 2 we will give an overview of existing work in Automated Negotiations and General Game Playing. In Section 3 we explain how we can model a negotiation scenario as a game. In Section 4 we give a short description of GDL. Then, in Section 5 we give a short description of a recently introduced language, which can be used by negotiating agents to make proposals to each other. Next, in Section 6 we give more details on how we think GDL can be applied to negotiations. In Section 7 we give some preliminary results that we have so far obtained, and finally, in Section 8, we present our conclusions.

2. RELATED WORK

The earliest work on automated negotiations was mainly focused on highly idealized scenarios in which it was possible to formally prove certain theoretical properties, such as the existence of equilibrium strategies. A seminal paper in this area is the paper by Nash [21] in which he shows that under certain axioms the outcome of a bilateral negotiation is the solution that maximizes the product of the players' utilities. This is known as the *Nash bargaining solution*. Many papers have been written afterwards that generalize or adapt some of these assumptions. A non-linear generalization has been made for example in [9]. Such studies give hard guarantees about the success of their approach, but the downside is that it is difficult to apply those results in real-world settings. A general overview of such game theoretical studies is made in [25].

In later work focus has shifted to more heuristic approaches. Such work focuses more on the implementation of negotiation algorithms for domains where one cannot expect to find any formal equilibrium results, or where such equilibria cannot be determined in a reasonable amount of time. It is usually not possible to give hard guarantees about the success of such algorithms, but they are more suitable to real-world negotiation scenarios. Important examples in this area are [7], and [8]. They propose a strategy that amounts to determining for each time t which utility value should be demanded from the opponent (the *aspiration level*). However, they do not take into account that one first needs to find a deal that indeed yields that aspired utility level. They simply assume that such a deal always exists, and that the negotiator can find it without any effort.

In general, these heuristic approaches still often make many simplifying assumptions; for example that there is only a small set of possible agreements, or that the utility functions are linear additive functions which are explicitly given or which can be calculated without much computational cost. All these assumptions were made for example in the first four editions of the annual Automated Negotiating Agent Competition (ANAC 2010-2013) [1].

Recently, more focus has been given to more realistic negotiation settings in which the number of possible deals is very large so that one needs to apply search algorithms to find good deals to propose, and where utility functions are non-linear, for example in [19, 14, 20]. Although their utility functions are indeed non-linear over the vector space that represents the space of possible deals, the value of any given deal can still be calculated quickly by solving a linear equation. Although in theory any non-linear function can be *modeled* in such a way, in real-world settings utility functions are not always *given* in this way (e.g. there is no known closed-form expression for the utility function over the set of all possible configurations of a Chess game). In order to apply their method one would first need to transform the given expression of a utility function into the expression required by their model, which may easily turn out to be an unfeasible task.

Therefore, the idea of complex utility functions was taken a step further in [4], where the utility functions were not only non-linear, but determining the value of any deal was actually an NP-hard problem.

Another important example of negotiations where determining utility values involves a hard combinatorial problem is the game of Diplomacy. In this game, negotiations are even more complex, because the utility values of the players are not directly defined in terms of the agreements they make, but more indirectly through the moves they make in the game. The players negotiate with one another about which moves they will make and their agreements only influence the utility indirectly. Pioneering work on negotiations in Diplomacy was presented in [23, 17]. New interest in Diplomacy as a test-bed for negotiations has arisen with the development of the DipGame platform [6], which makes the implementation of Diplomacy agents easier for scientific research. Several negotiating agents have been developed on this platform [10, 5, 3].

General Game Playing is a relatively new topic. Although earlier work has been done, it really became an important topic after the introduction of GDL [18] and the organization of the annual AAAI GGP competition since 2005 [12].

Common techniques applied by GGP players are minimax [27] search, alpha-beta pruning [15] and Monte Carlo Tree Search (MCTS) [16]. All these techniques generate a search tree in which each node ν stores a certain state w_ν of the game and a certain player α_ν. The root node represents the initial state, and for each node the edges to its child nodes represent the actions that are legal in the state w_ν, for player α_ν. Each time a new node ν is added to the tree, the algorithm applies the game rules, which are written in GDL, to determine which actions are legal for player α_ν in the state w_ν and, if w_ν is a terminal state, which utility value each player receives.

FluxPlayer [24], the winner of the 2006 AAAI GGP competition applies an iterated deepening depth-first search method with alpha-beta pruning, and uses Fuzzy logic to determine how close a given state is to the goal state. Cadia Player [11], the winner in 2007, 2008, and 2012, is based on MCTS extended with several heuristics to guide the playouts so that they are better informed and hence give more realistic results. Furthermore, also the winner of 2014, Sancho[1], as well as the winner of 2015, Galvanise[2] apply variants of MCTS.

[1]\http://sanchoggp.blogspot.co.uk/2014/05/what-is-sancho.html

[2]https://bitbucket.org/rxe/galvanise_v2

3. NEGOTIATION GAMES

Since GDL is a language to describe games, in this section we explain how a negotiation scenario can be described as a game.

Game theory can be related to negotiations in two possible ways. Firstly, the negotiation protocol itself can be modeled as a game. This approach is for example taken in Nash' famous paper [21] and is the common approach taken when one intends to formally prove certain properties of any negotiation scenario. In this case the moves made by the players consist of making proposals and accepting proposals, and the utility function is directly given as a function over the space of possible outcomes of the negotiation protocol.

However, in this paper we follow a new approach in which we also apply a game theoretical model to definine the utility functions. That is: players receive utility by making certain moves in some game G, and on top of that they are allowed to negotiate about the moves they will make in that game according to some negotiation protocol N. A typical example of such a negotiation scenario is the game of Diplomacy [6]. In this case there are two types of moves: negotiation-moves (i.e. making proposals, accepting proposals or rejecting proposals) and game-moves (the moves defined in the game G). The proposals that players make or accept are proposals about which game-moves they will make. The utilities of the players are only determined by the game-moves. However, since the agreements they make during the negotiations will partially restrict their possible game-moves, the utility values obtained by the players indirectly do depend on the negotiations.

We will first define the notion of a protocol, and then define the concepts of a game and of a negotiation protocol which are two different extensions of the concept of a protocol. Next, we will define a Negotiation Game, which is a combination of a negotiation protocol and a game.

Definition 1. *A **protocol** P is a tuple $\langle Ag, \mathcal{A}, W, w_1, T, L, u\rangle$, where:*

- *Ag is the set of **agents** (or **players**): $Ag = \{\alpha_1, \alpha_2, \ldots \alpha_n\}$*
- *$\mathcal{A}$ is the set of **actions** (or **moves**): $\mathcal{A} = \mathcal{A}_1 \cup \mathcal{A}_2 \cup \ldots \mathcal{A}_n$.*
- *W is a non-empty set of **states**.*
- *$w_1 \in W$ is the **initial** state.*
- *$T \subset W$ is the set of **terminal** states.*
- *$L : Ag \times W \setminus T \to 2^{\mathcal{A}}$ is the **legality** function, which describes for each non-terminal state and each agent which actions are legal.*
- *$u : W \times \mathcal{A}_1 \times \mathcal{A}_2 \cdots \times \mathcal{A}_n \to W$ is the update function that maps each state and action profile to a new state.*

Each player α_i has its own set of actions $\mathcal{A}_i$ with $\mathcal{A}_i \cap \mathcal{A}_j = \emptyset$ if $i \neq j$. For each non-terminal state w, each agent α_i must choose a legal action, that is: some action a such that $a \in L(\alpha_i, w)$. The legality function is defined in terms of individual legality functions: $L(\alpha_i, w) = L_i(w)$ where each L_i is a map from $W \setminus T$ to $\mathcal{A}_i$. A tuple $\vec{a} = (a_1, a_2, \ldots a_n)$ consisting of one action $a_i \in \mathcal{A}_i$ for each player is called an **action-profile**. Given a state w and an action-profile $\vec{a}$ the update function u defines a new state $w' = u(w, \vec{a})$. Therefore, given the initial state w_1 and a sequence of action-profiles $\vec{a}_1, \vec{a}_2 \ldots$ the update function defines a sequence of states $w_1, w_2, \ldots$ where each w_{j+1} equals $u(w_j, \vec{a}_j)$. The protocol finishes once the realized world state w_j is a terminal state $w_j \in T$.

Note that in this model it is assumed that the agents always take their actions simultaneously. This is not really a restriction because any turn-taking protocol can be modeled as special case of a simultaneous-move protocol, by adding a special dummy-move, often called 'noop', to the model which has no effect on the next state. Then, one can define the legality function such that in every state all players except one have only one legal move, which is the 'noop' move. The one player that does have more than one legal move is then called the **active player** of that state.

Definition 2. *A **negotiation protocol** N is a tuple $\langle P, Agr, C\rangle$, where:*

- *P is a protocol.*
- *Agr is a set of possible agreements, known as the **agreement space**.*
- *$C : T \to Agr$ is the **commitment function** that maps every terminal state of the protocol to an agreement.*

The set Agr can be any set that represents the possible deals the agents can make with each another. The interpretation of C is that if the negotiation protocol ends in a terminal state w then $C(w)$ is the agreement that the agents have agreed upon[3]. The set Agr must contain one element c that represents the 'conflict deal' i.e. an element to represent that no deal has been made. So if the agents do not come to any agreement, than the protocol ends in a final state w for which $C(w) = c$.

Example 1. *Let us define the alternating offers protocol [22] using this model. Suppose we have two agents negotiating how to split a pie according to an alternating offers protocol over n rounds. The agents are denoted $Ag = \{\alpha_1, \alpha_2\}$. The possible agreements are the real values between 0 and 1, representing the fraction of the pie assigned to player α_1, so: $Agr = [0, 1] \cup \{c\}$, where c is the 'conflict deal'. The actions of the players are either to propose a division of the pie, or to accept the previous proposal, or to do nothing:*

$$\mathcal{A}_1 = \mathcal{A}_2 = \{propose(x) \mid x \in [0, 1]\} \cup \{accept, noop\}.$$

A state is given as a triple: (r, x, b) where r is the round of the protocol, x is the last proposal made, and b is either 'true' ($\top$) or 'false' ($\bot$) indicating whether x has been accepted or not.

$$W = \{(r, x, b) \mid 0 \leq r \leq n, \quad x \in Agr, \quad b \in \{\top, \bot\}\}$$

The initial state is: $w_1 = (0, c, \bot)$. Terminal states are those states in which either the last round has passed or any of the agents has accepted a proposal:

$$T = \{(r, x, b) \in W \mid r = n \quad \vee \quad b = \top\}$$

In the even rounds player α_1 is the active player and in the odd rounds α_2 is active. In every states all actions are legal for the active player, except that in the initial state it is not allowed to play 'accept' (because no proposal has been made that can be accepted).
If $r = 0$:

$$L(\alpha_1, (r, x, b)) = \mathcal{A}_1 \setminus \{accept\} \quad L(\alpha_2, (r, x, b)) = \{noop\}$$

[3]We could generalize this and allow protocols in which more than one deal can be made. However, we will not do so here for simplicity

If $r > 0$:

$$L(\alpha_i,(r,x,b)) = \mathcal{A}_i \qquad L(\alpha_j,(r,x,b)) = \{noop\}$$

with $i = r \ (mod\ 2) + 1$ and $j \neq i$. The update function is defined as follows:

$$u((r,x,\bot), propose(y), noop) = (r+1, y, \bot)$$

$$u((r,x,\bot), accept, noop) = (r+1, x, \top)$$

And finally, the commitment function returns the proposal that was accepted or, if no proposal was accepted, returns the conflict deal:

$$C(r,x,\top) = x \quad C(n,x,\bot) = c$$

Note that this definition of the alternating offers protocol can be adapted easily to domains other that split-the-pie, simply by replacing Agr by some other agreement space. Everything else remains the same.

Definition 3. *A **game** G is a pair $\langle P, U\rangle$ where P is a protocol and U is a function that assigns a utility value to each terminal state and player. $U : Ag \times T \rightarrow \mathbb{R}^+$.*

The goal of each player α_i is to maximize his or her utility $U(\alpha_i, w_j)$ in the final state w_j of the game.

Example 2. *For example, let G be the prisoner's dilemma. Then we have the following protocol (c stands for 'confess' and d stands for 'deny'):*

- $Ag = \{\alpha_1, \alpha_2\}$
- $\mathcal{A}_1 = \mathcal{A}_2 = \{c, d\}$
- $W = \{w_1, w_{cc}, w_{cd}, w_{dc}, w_{dd}\}$
- $T = \{w_{cc}, w_{cd}, w_{dc}, w_{dd}\}$
- $L(\alpha_1, w_1) = L(\alpha_2, w_1) = \{c, d\}$
- $u(w_1, c, c) = w_{cc}, \quad u(w_1, c, d) = w_{cd},$ $u(w_1, d, c) = w_{dc}, \quad u(w_1, d, d) = w_{dd}$

and the utility function can be defined as:

- $U(\alpha_1, w_{cc}) = U(\alpha_2, w_{cc}) = 2$
- $U(\alpha_1, w_{cd}) = U(\alpha_2, w_{dc}) = 10$
- $U(\alpha_1, w_{dc}) = U(\alpha_2, w_{cd}) = 0$
- $U(\alpha_1, w_{dd}) = U(\alpha_2, w_{dd}) = 8$

Definition 4. *For any game G, a **strategy** σ for player α_i is a map that maps every non-terminal state of that game to a non-empty set of legal moves for that player:*

$$\sigma : W \setminus T \rightarrow 2^{\mathcal{A}_i}$$

*such that for each $w \in W \setminus T$ we have $\sigma(w) \neq \emptyset$ and $\sigma(w) \subseteq L(\alpha_i, w)$. A **complete strategy** is a strategy such that $|\sigma(w)| = 1$ for all $w \in W \setminus T$, and a **partial strategy** is a strategy that is not complete. A tuple $(\sigma_1, \sigma_2 \ldots \sigma_n)$ consisting of one strategy for each player is called a **strategy profile**.*

In the following, we use a superscript G or N to indicate that something is a component of the game G or the negotiation protocol N. For example P^G is the protocol of G, W^N is the set of world states of N, etcetera.

We will next define a negotiation game NG to be a combination of a negotiation protocol N and a game G. The interpretation of NG is that it is a game that consists of two stages: a negotiation stage followed by an action stage. In the action stage the players play the game G, while in the preceding negotiation stage the players negotiate about which moves they will make during the action stage. These agreements are considered binding, therefore, if the players come to an agreement they will have less legal moves during the action stage than they would have in the pure game G.

We say a negotiation protocol N is compatible with a game G if it has the same set of agents as G, and the set of agreements Agr consists purely of strategy profiles for the game G. This means that N is designed for the agents of G to negotiate the strategies they will play in the game G.

Definition 5. *A negotiation protocol N is **compatible** with a game G, if:*

- $Ag^N = Ag^G$.
- *If $x \in Agr^N$ then x is a strategy profile for G.*

The interpretation here, is that if the negotiators agree on some strategy profile $(\sigma_1, \ldots \sigma_n)$ then each player α_i has restricted its legal moves to those in the strategy σ_i. Specifically, if σ_i is a *complete* strategy, then α_i has no more free choice in G, and must play in any state w the unique action in $\sigma_i(w)$. Furthermore, the conflict deal of Agr^N corresponds to the strategy profile in which each player still has its full set of legal actions to choose from: $\sigma_i(w) = L_i(w)$. Indeed, if no agreement is made this means that no agent is restricted by any commitments and may therefore choose any legal action in G.

Definition 6. *Given a game G and a negotiation protocol N compatible with G we define the **negotiation game** NG as a game, such that:*

- *For each player α_i: $\mathcal{A}_i = \mathcal{A}_i^N \cup \mathcal{A}_i^G$*
- *The set of states W is a subset of $W^N \times W^G$. More precisely: $W = W^{nego} \cup W^{act}$ with:*

$$W^{nego} = W^N \times \{w_1^G\}$$

$$W^{act} = T^N \times W^G$$

- *The initial state is defined as: $w_1 = (w_1^N, w_1^G)$.*
- *The terminal states are defined as: $T = T^N \times T^G$.*
- *The update function is defined as:*

$$u((v,z),\vec{a}) = (u^N(v,\vec{a}), z) \quad \text{if } (v,z) \in W^{nego}$$

$$u((v,z),\vec{a}) = (v, u^G(z,\vec{a})) \quad \text{if } (v,z) \in W^{act}$$

- *The legality function is defined as:*

$$L(\alpha_i,(v,z)) = L^N(\alpha_i, v) \quad \text{if } (v,z) \in W^{nego}$$

$$L(\alpha_i,(v,z)) = \sigma_i(z) \quad \text{if } (v,z) \in W^{act}$$

$$\text{where } (\sigma_1, \ldots \sigma_n) = C^N(v)$$

- *The utility function is defined as:*

$$U(v,z) = U^G(z)$$

We have modeled states w of NG as pairs of states $w = (v,z)$ consisting of a protocol-state $v \in W^N$ and a game-state $z \in W^G$. The initial state w_1 of NG is simply the pair of initial states (w_1^N, w_1^G) of N and G.

We have divided the state space W into two subspaces: W^{nego} and W^{act}. Note that the initial state is in W^{nego} and that the update and legality functions are defined such that the realized states are first remain in W^{nego} until a state in W^{act} is reached, after which the realized states always remain in W^{act}. If the current state is in W^{nego} we say that the negotiation game is in the **negotiation stage** and if the state in W^{act} we say that NG is in the **action stage**.

During the negotiation stage the update function u only acts on the protocol-state, and acts on it according to the update function u^N of N, while during the action stage u only acts on the game-state, according to u^G.

Similarly, during the negotiation stage the legality function L is simply the legality function L^N of the negotiation protocol. Therefore, during this stage the agents can make proposals and accept proposals. During the action stage the legality function allows any agent α_i to only make those actions it was committed to during the negotiation stage. Note that indeed, if $(v, z) \in W^{act}$ then v is a terminal state of N, and therefore $C^N(v)$ is some agreement from the agreement space Agr^N of N. Furthermore, since N is compatible with G, we know that $C^N(v)$ is a strategy profile of G. In other words: if during the negotiation stage the agents have agreed to on the strategy profile $C^N(v) = (\sigma_1, \ldots \sigma_n)$ then during the action stage each agent α_i is committed to play according to the strategy σ_i.

EXAMPLE 3. *For example, if G is the Prisoner's Dilemma, and N is the alternating offers protocol compatible with G, then the Negotiating Prisoner's Dilemma NG begins with a negotiation stage in which the prisoners may agree which strategy they will play. The prisoners may propose any strategy-pair (σ_1, σ_2). Since there is only one non-terminal state in the Prisoner's Dilemma, a strategy is defined by its value $\sigma_i(w_1)$ on that non-terminal state w_1. That is, σ_i can be either $\sigma_i(w_1) = \{c\}$ or $\sigma_i(w_1) = \{d\}$ or $\sigma_i(w_1) = \{c, d\}$. If the prisoners are rational, then one of them will propose the strategy profile $(\{d\}, \{d\})$ and the other will accept the proposal. In the action stage they are then both committed to play the action d, which yields a better outcome for both of them then the outcome (c, c) which would be the rational outcome (the Nash-equilibrium) of the pure prisoner's dilemma G.*

In this model, the players of a negotiation game only have one opportunity to negotiate, before they play the game G. However, we could also consider more general models in which, for example, the players have a new opportunity to negotiate before each new turn of the game G. We will not do this however, to keep the discussion simple.

4. GAME DESCRIPTION LANGUAGE

In this section we will give a short introduction to GDL. For more details we refer to [13].

GDL is logical language that was designed to describe games. In principle, it can describe any game G defined according to Definitions 1 and 3. GDL is similar to Datalog [2], but it defines the following relation symbols:[4] *init*, *true*, *next*, *legal*, *goal*, *terminal*, *does*, which have a special meaning related to games.

[4]GDL defines more relations, but these are not relevant for this paper.

In GDL a state w of a game is represented as a set of atomic formulas, which we will here denote as $V(w)$. These atoms are all of the form $true(p)$, where p can be any ground term. For example, in Tic-Tac-Toe the state in which the the center cell contains the marker X and the left upper cell contains the marker O could be represented as:

$$V(w) = \{\ true(cell(2,2,X))\ \ ,\ \ true(cell(1,1,O))\ \}$$

A GDL **rule** is an expression of the following form:

$$s_1 \wedge s_2 \wedge \ldots s_n \rightarrow h$$

where each s_i is a positive or negative literal, and h is a positive literal. The atom h is called the *head* of the rule and the s_i's are called the *subgoals* of the rule. The conjunction of subgoals is called the *body* of the rule. The body of a rule may be the empty conjunction, in which case the rule is also referred to as a *fact*.

A game description is then nothing more then a set of GDL rules. For example, if the game description contains the following rule:

$$true(p) \wedge does(\alpha_i, a) \rightarrow next(q)$$

it means that if the game is in a state w for which $true(p) \in V(w)$ and player α_i plays action a then in the next round the game will be in a state w' for which $true(q) \in V(w')$ holds. Similarly:

$$true(p) \rightarrow terminal$$

means that any state w for which $true(p) \in V(w)$ holds is a terminal state. The fact

$$\rightarrow init(p)$$

means that for the initial state w_1 we have that $true(p) \in V(w_1)$ holds.

$$true(p) \rightarrow legal(\alpha_i, a)$$

means that for any state in which $true(p) \in V(w)$ holds it is legal for player α_i to play the move a.

$$true(p) \rightarrow goal(\alpha_i, 100)$$

means that in any state w for which $true(p) \in V(w)$ holds α_i receives a utility value of 100.

GDL uses what is known as negation-by-failure. This means that a negative literal $\neg p$ is considered true if and only if there is no rule from which one can derive the truth of p.

GDL can only describe games of full information without randomness. However, an extension to GDL exists, called GDL-II [26] which does allow for randomness and private information.

The Game Manager is a server application specially designed for General Game Playing. It allows a user to start a game with a game description written in GDL, and allows game playing agents to connect and start playing. Once connected, the server sends the game description to the players. The players then need to parse the description and determine for themselves which moves they can make, and what the consequences are. Every round, each player is supposed to send a message back with the move it desires to play. If a player fails to send this message, or if it chooses an action that is illegal in the current state, the server will instead pick a random move for that player. Next, the server sends a message back to all players, informing each player which moves have been made by the other players. From this information the players can then compute the next state, and determine which moves they can make in that new state. This continues until a terminal state is reached.

5. A LANGUAGE TO DEFINE STRATEGIES

As explained, the proposals made during the negotiation stage of a negotiation game are in fact strategy profiles. For example, in the negotiating prisoner's dilemma, player α_1 may propose:

$$(\sigma_1, \sigma_2) = (\{d, c\}, \{d\})$$

Here, σ_1 is the partial strategy in which player α_1 has the choice to play either d or c, and σ_2 is the strategy in which α_2 plays d (of course this is a highly unprofitable deal for α_2 so if α_2 is rational he or she will not accept it).

In the case of the Prisoner's Dilemma a strategy can be represented simply as the set of possible actions in the only non-terminal state. However, in some games, such as Diplomacy, the number of possible actions and states can be are extremely large. Therefore, expressing a strategy explicitly as a set of actions for each possible state in such games is infeasible.

Instead, we propose to use Strategic Game Logic: a logical language specifically defined to describe game-strategies [28]. SGL is in fact an extension of GDL. While GDL is used to describe the *rules* of a game, SGL can be used to describe *strategies* of a game. Let us briefly give the ideas here.

A logical formula ϕ in SGL may represent a set of states, or a set of state-action pairs. For example: $\phi = true(p) \wedge true(q)$ would represent the set of states:

$$\{w \in W \mid true(p) \in V(w) \text{ and } true(q) \in W\}$$

while $\phi = true(p) \wedge does(a)$ represents the set of state-action pairs:

$$\{(w, a) \in W \times \{a\} \mid true(p) \in V(w)\}$$

We say that w satisfies ϕ if w is in the set of states represented by ϕ, and we say that (w, a) satisfies ϕ if (w, a) is in the set of state-action pairs represented by ϕ.

Note that a set S of state-action pairs can be seen as a strategy, defined by $a \in \sigma_i(w)$ iff $(w, a) \in S$. Therefore, negotiators may make proposals of the form $propose(\phi_1, \phi_2)$ where ϕ_1 represents a strategy for player α_1 and ϕ_2 represents a strategy for player α_2.

SGL defines a number of new operators on top of GDL. That is, if a is an action and ϕ and φ are formulas, then SGL defines the following expressions: $[a]\phi$, $\lfloor a \rfloor \phi$, and $\wr\varphi\wr\phi$.

Let ϕ be any formula representing a set of states, and a be any action. If w satisfies ϕ, then by definition $u(a, w)$ satisfies $[a]\phi$. That is: $[a]\phi$ represents the set of states that result from the action a being played in a state that is represented by ϕ. Furthermore, w satisfies ϕ if and only if (w, a) satisfies $\lfloor a \rfloor \phi$.

For the last operator we have that $\wr\varphi\wr\phi$ is satisfied only if ϕ is satisfied under a new protocol where the legality relation of the original protocol is replaced by the strategy φ. For example, suppose that we have a state w in which it is legal for player α_i to play either action a or action b. Then the formulas $\phi_1 = legal(\alpha_i, a)$ and $\phi_2 = legal(\alpha_i, b)$ are both satisfied by w. Furthermore, let φ be a strategy in which α_i plays action b in state w. Then we have that w satisfies $\wr\varphi\wr\phi_2$, but not $\wr\varphi\wr\phi_1$.

Using these new operators SGL defines two more operators that allow to combine any two strategies into a new strategy. Firstly, SGL defines *prioritized disjunction*:

$$\phi \circledvee \varphi = \phi \vee (\varphi \wedge \bigwedge_{c \in \mathcal{A}_i} \lfloor c \rfloor \neg \phi)$$

which has the interpretation of *"play strategy ϕ if applicable, otherwise play strategy φ"*. Secondly, SGL defines *prioritized conjunction*:

$$\phi \circledwedge \varphi = \phi \wedge ((\bigvee_{c \in \mathcal{A}_i} \lfloor c \rfloor (\phi \wedge \varphi)) \rightarrow \varphi)$$

Which is a strategy with the interpretation: *"apply both ϕ and φ if both are applicable; otherwise, apply ϕ only"*.

6. APPLYING GDL TO NEGOTIATIONS

In this section we will discuss some technical issues that we encountered when using GDL to describe a negotiation game.

6.1 Enforcement of Commitments

One problem we need to take care of is the question how to enforce that agents obey their agreements. Although in some domains (such as Diplomacy) there simply is no mechanism at all to force agents to obey agreements, in many existing domains one does require agreements to be enforced. For this we can imagine two possible solutions.

The first option would be to explicitly write a game description that defines the negotiation protocol N as well as the game G and include rules that guarantee agreement obedience. That is, it would contain rules of the form "If players α_1 and α_2 make the agreement that α_1 will play action a_1 and α_2 will play action a_2, then those actions will be the only legal actions". For example, if the original game G contains the rule

$$true(p) \rightarrow legal(\alpha_i, a)$$

then the negotiation game NG would instead contain the rules:

$$committed(\alpha_i, a) \wedge a \neq b \rightarrow excluded(\alpha_i, b)$$

$$true(p) \wedge \neg excluded(\alpha_i, a) \rightarrow legal(\alpha_i, a)$$

where the first rule specifies that whenever a player gets committed to an action a_i, then all other actions are excluded. The second rule is an adaptation of the original rule, with the extra premise added that an action a_i can only be legal if it has not been excluded by the commitments.

The second option would be to write a new game server that forbids the players to make moves that are incompatible with their agreements. In that case we see the negotiation protocol N and the game G as two separate games with their own game description. If a player gets committed to action a but tries to play the action b the server will not allow it, even though b is legal according to the rules of the pure game G.

The advantage of the first option is that it is completely compatible with existing GGP standards. A negotiation game NG is just another game that can be described in GDL. Any existing GGP player should therefore be able to participate in such a negotiation game. However, the problem is that one would need to write rules that take all possible commitments into account. After all, a commitment may not simply be a single action, as in this example, but could be a disjunction of actions, or it could be conditional (e.g. if you play a I will play b in the next round, or if you play b then I will play a). It would be very complicated to write rules that are generic enough to cover all possible commitments, especially if we allow the full SGL language to specify agreements. Furthermore, it seems rather redundant to explicitly write rules to enforce commitments, if it is obvious that agreements must be obeyed.

We therefore think it might be more practical to choose for the second option and have a special General Negotiation server that handles rule enforcement. Moreover, it would allow us to re-use existing game specifications and combine them with existing negotiation protocols. There is no need to adapt the rules of the game.

6.2 Very Large Spaces

Currently, many GGP programs are not able to handle domains where the number of possible actions is very large. The reason is that they apply *grounding*: they try to generate a list that explicitly contains all possible actions. Of course, if there are 10^{10} possible proposals that can be made, like in ANAC 2014, this approach will not work.

However, we have managed to implement a GDL specification of a domain like in ANAC 2014, where we avoid this problem with a little trick. In this domain the negotiators propose contracts that consist of 10 properties, and where each property can have 10 different values, so that there are 10^{10} possible contracts.

Instead of mapping each possible proposal to an action, we have defined a proposal to consist of several actions. More precisely: apart from the three standard types of action: 'propose', 'accept' and 'noop' from Example 1, we have added a fourth action called 'setValue'. The setValue action takes two parameters: a property-index and a value. By playing a number of setValue actions the player creates a contract, in which the indicated properties have the indicated values. For example, if you play the following three actions:

$$setValue(1,5), \quad setValue(2,9), \quad setValue(4,2)$$

you create a contract in which the first property has value 5, the second property has value 9 and the fourth property has value 2. All other properties will by default have value 0. Then, after generating the contract the negotiator can propose it by playing the action 'propose'. In this way there are only 103 possible actions, instead of 10^{10}.

6.3 Continuous Time

In GDL it is assumed that games take place over discrete rounds. The duration of each round can be specified in the server. However, in negotiations it is not uncommon to assume that negotiations take place in continuous time. In the ANAC 2014 domain for example, although the agents did take turns, each agent could take as much time as it wanted to make a proposal. In principle, this is not a problem, because we can simple allow agents to make a 'noop' move representing the 'action' of not making an action, and make sure that the other agent only gets the turn when the first agent makes a proposal.

There is however a small technical problem with this, namely that each agent must take care that it indeed makes the 'noop' action before the round finishes. If it does not manage to do this in time (for example because it is doing heavy calculations that take up all its resources) then the server will automatically pick a random action for that agent. Of course, this is undesirable because the server may choose a highly unprofitable proposal which may then be accepted by the opponent.

We think that, in the context of negotiations in continuous time, it would be better to have a server that by default picks the 'noop' move if you fail to play any action within the time limits.

7. RESULTS

We have managed to specify the Negotiating Prisoner's Dilemma of Example 3 in GDL, in which the negotiation stage was modeled as a turn-taking protocol with three rounds in which prisoner 1 is first allowed to make a proposal, next prisoner 2 is allowed to either accept that proposal or make a new proposal, and finally prisoner 1 may accept the last proposal made by prisoner 2.

We have implemented a straightforward minimax algorithm for GGP, and when we let two instances of this algorithm play the Negotiating Prisoner's Dilemma they indeed successfully negotiate and agree to both play 'deny'. This is very interesting, because these algorithms are *not* negotiation algorithms. They are simply game-playing algorithms that may just as well play Tic-Tac-Toe or any other simple game. The reason why they are able to negotiate successfully is that the negotiation scenario was described in GDL.

Moreover, we have implemented a domain similar to ANAC 2014 in which the agents negotiate according to the alternating offers protocol over a space with 10^{10} possible contracts. Since this is a very large domain over many rounds a naive minimax does not work. To be able to handle such domains we need to apply more state-of-the-art GGP techniques. We leave this as future work. Also it would be interesting to see whether any of the top existing GGP players is able to handle this domain.

8. CONCLUSIONS

We conclude that GDL is in essence a good option for the description of general negotiation scenarios, but there are a number of aspects specific to automated negotiations that are not handled well by GDL and the existing GGP server.

Therefore, we think that it is necessary to write a new server application specifically for negotiations. This server will handle rule enforcement and should be able to verify whether any action is compatible with any strategy defined as a formula in SGL.

We have shown that a simple Negotiating Prisoner's Dilemma can be described correctly in GDL, as well as the more complex domain of ANAC 2014.

Furthermore, we have shown that it is indeed possible for a GGP algorithm to successfully negotiate in the Negotiating Prisoner's Dilemma even though it is not designed for negotiations. For the larger ANAC 2014 domain we still need to find out whether existing GGP techniques are able to handle it.

9. ACKNOWLEDGMENTS

This work was sponsored by an Endeavour Research Fellowship awarded by the Australian Government, Department of Education.

10. REFERENCES

[1] T. Baarslag, K. Hindriks, C. M. Jonker, S. Kraus, and R. Lin. The first automated negotiating agents competition (ANAC 2010). In T. Ito, M. Zhang, V. Robu, S. Fatima, and T. Matsuo, editors, *New Trends in Agent-based Complex Automated Negotiations, Series of Studies in Computational Intelligence*. Springer-Verlag, 2010.

[2] S. Ceri, G. Gottlob, and L. Tanca. What you always wanted to know about datalog (and never dared to ask). *IEEE Transactions on Knowledge and Data Engineering*, 1(1):146–166, 1989.

[3] D. de Jonge. *Negotiations over Large Agreement Spaces*. PhD thesis, Universitat Autònoma de Barcelona, 2015.

[4] D. de Jonge and C. 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, 2015.

[5] A. Fabregues. *Facing the Challenge of Automated Negotiations with Humans*. PhD thesis, Universitat Autònoma de Barcelona, 2014.

[6] A. Fabregues and C. Sierra. Dipgame: a challenging negotiation testbed. *Engineering Applications of Artificial Intelligence*, 2011.

[7] P. Faratin, C. Sierra, and N. R. Jennings. Negotiation decision functions for autonomous agents. *Robotics and Autonomous Systems*, 24(3-4):159 – 182, 1998. Multi-Agent Rationality.

[8] P. Faratin, C. Sierra, and N. R. Jennings. Using similarity criteria to make negotiation trade-offs. In *International Conference on Multi-Agent Systems, ICMAS'00*, pages 119–126, 2000.

[9] S. Fatima, M. Wooldridge, and N. R. Jennings. An analysis of feasible solutions for multi-issue negotiation involving nonlinear utility functions. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '09, pages 1041–1048, Richland, SC, 2009. International Foundation for Autonomous Agents and Multiagent Systems.

[10] A. Ferreira, H. Lopes Cardoso, and L. Paulo Reis. Dipblue: A diplomacy agent with strategic and trust reasoning. In *7th International Conference on Agents and Artificial Intelligence (ICAART 2015)*, pages 398–405, 2015.

[11] H. Finnsson. *Simulation-Based General Game Playing*. PhD thesis, School of Computer Science, Reykjavik University, 2012.

[12] M. Genesereth, N. Love, and B. Pell. General game playing: Overview of the aaai competition. *AI Magazine*, 26(2):62–72, 2005.

[13] M. R. Genesereth and M. Thielscher. *General Game Playing*. Synthesis Lectures on Artificial Intelligence and Machine Learning. Morgan & Claypool Publishers, 2014.

[14] T. Ito, M. Klein, and H. Hattori. A multi-issue negotiation protocol among agents with nonlinear utility functions. *Multiagent Grid Syst.*, 4:67–83, January 2008.

[15] D. E. Knuth and R. W. Moore. An analysis of alpha-beta pruning. *Artificial Intelligence*, 6(4):293 – 326, 1975.

[16] L. Kocsis and C. Szepesvári. Bandit based monte-carlo planning. In *Proceedings of the 17th European Conference on Machine Learning*, ECML'06, pages 282–293, Berlin, Heidelberg, 2006. Springer-Verlag.

[17] S. Kraus. Designing and building a negotiating automated agent. *Computational Intelligence*, 11:132–171, 1995.

[18] N. Love, M. Genesereth, and T. Hinrichs. General game playing: Game description language specification. Technical Report LG-2006-01, Stanford University, Stanford, CA, 2006. http://logic.stanford.edu/reports/LG-2006-01.pdf.

[19] I. Marsa-Maestre, M. A. Lopez-Carmona, J. R. Velasco, and E. de la Hoz. Effective bidding and deal identification for negotiations in highly nonlinear scenarios. In *Proceedings of The 8th International Conference on Autonomous Agents and Multiagent Systems - Volume 2*, AAMAS '09, pages 1057–1064, Richland, SC, 2009. International Foundation for Autonomous Agents and Multiagent Systems.

[20] I. Marsa-Maestre, M. A. Lopez-Carmona, J. R. Velasco, T. Ito, M. Klein, and K. Fujita. Balancing utility and deal probability for auction-based negotiations in highly nonlinear utility spaces. In *Proceedings of the 21st International Jont Conference on Artifical Intelligence*, IJCAI'09, pages 214–219, San Francisco, CA, USA, 2009. Morgan Kaufmann Publishers Inc.

[21] J. Nash. The bargaining problem. *"Econometrica"*, "18":155–162, 1950.

[22] J. S. Rosenschein and G. Zlotkin. *Rules of Encounter*. The MIT Press, Cambridge, USA, 1994.

[23] E. E. S. Kraus, D. Lehman. An automated diplomacy player. In D. Levy and D. Beal, editors, *Heuristic Programming in Artificial Intelligence: The 1st Computer Olympia*, pages 134–153. Ellis Horwood Limited, 1989.

[24] S. Schiffel and M. Thielscher. M.: Fluxplayer: A successful general game player. In *In: Proceedings of the AAAI National Conference on Artificial Intelligence*, pages 1191–1196. AAAI Press, 2007.

[25] R. Serrano. bargaining. In S. N. Durlauf and L. E. Blume, editors, *The New Palgrave Dictionary of Economics*. Palgrave Macmillan, Basingstoke, 2008.

[26] M. Thielscher. A general game description language for incomplete information games. In *Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010*, 2010.

[27] J. von Neumann. On the theory of games of strategy. In A. Tucker and R. Luce, editors, *Contributions to the Theory of Games*, pages 13–42. Princeton University Press, 1959.

[28] D. Zhang and M. Thielscher. A logic for reasoning about game strategies. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence (AAAI-15)*, pages 1671–1677, 2015.

A Realistic Scenario for Complex Automated Nonlinear Negotiation: Wi-Fi channel assignment

Enrique de la Hoz,
Jose Manuel Gimenez-Guzman
and Ivan Marsa-Maestre
Computer Engineering Department
University of Alcala
{enrique.delahoz,josem.gimenez,ivan.marsa}
@uah.es

David Orden
Department of Physics and Mathematics
University of Alcala
Alcala de Henares, Spain
david.orden@uah.es

ABSTRACT

Although nonlinear complex automated negotiation techniques have been successfully developed in the last years, there is a lack of realistic scenarios to apply them to. Efforts have been made to generate repositories with a wide variety of nonlinear negotiation settings according to different dimensions, but the generation of these scenarios, although reasonable, was not backed up by realistic application settings. In this paper, we propose a scenario family inspired by a real-world application: Wi-Fi channel assignment. We study this scenario family and we show that nonlinear negotiation techniques are a good match for this problem.

1. INTRODUCTION

Nonlinear negotiation techniques have been studied for some time, and several successful approaches have emerged [6, 3, 8]. However, a recurring problem with these mechanisms has been to have a varied, consistent set of scenarios to evaluate them. There have been successful attempts to sistematize scenario characterization and generation, creating repositories with a wide variety of nonlinear negotiation scenarios classified according to a number of techniques [9]. A remaining limitation of these efforts, however, is that the resulting scenarios are somewhat artificial, that is, they are not backed up or easily matched by real-world problems. To our knowledge, there is no proposal to systematically generate a wide variety of realistic nonlinear negotiation scenarios.

To bridge this gap, in this paper we propose a scenario family inspired by the problem of frequency assignment in Wi-Fi infrastructure networks. In this problem, different internet service providers (ISPs) have to collectively decide how to distribute the channels used by their access points (APs) in order to minimize interference between nodes and thus maximize the utility (i.e. network throughput) for their clients.

The paper is organized as follows. Section 2 presents a model for the problem of Wi-Fi channel assignment, using an interference graph and utilities based on Wi-Fi interferences. Section 3 approaches channel assignment as a negotiation problem, defining the scenario with its interaction protocol and strategies. Section 4 shows our experimental settings, where we have generated a large set of scenario instances for this problem, and conducted an extensive set of experiments. Section 5 discusses our experimental results. The last section summarizes our contributions and sheds light on future lines of research.

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), John Thangarajah, Karl Tuyls, Stacy Marsella, Catholijn Jonker (eds.), May 9–13, 2016, Singapore.*

2. WI-FI CHANNEL ASSIGNMENT

IEEE 802.11 technology, comercially known as WiFi, is a very popular and widespread technology that is able to operate in different frequency bands. At present, the most used standard is IEEE 802.11n, that is able to operate in two frequency bands: 2.4 GHz and 5 GHz, being the 2.4 GHz band the most habitual one, so we focus our attention on that band. Due to the high number of WiFi devices that coexist in that frequency band it is usually congested, and this situation is often worsened by other devices like Bluetooth, ZigBee, microwave ovens, baby monitors or cordless phones. For those reasons, it is of paramount importance that WiFi devices smartly manage the use of the radio spectrum. More concretely, the 2.4 GHz band is divided into 11 partially overlapped channels, so it is important to choose the most advantageous one to minimize interferences.

The scenarios considered in this paper represent WiFi networks deployed in infrastructure mode, which is the most widely deployed WiFi architecture. In this mode, there are two types of devices in the network: access points (APs) and wireless devices (WDs) such as laptops, smartphones... In infrastructure mode, wireless devices are wirelessly connected to a single AP, which is generally a wireless router, and are able to communicate to other devices only through that AP. For that reason, WDs are also called clients. Connectivity among WDs and to the rest of devices of Internet is provided by APs, which are connected to Internet through wired connections, that do not affect to wireless interferences and, therefore, we omit them in our problem formulation.

We model the problem structure using a graph, which is one of the most commonly used tools for modelling the frequency assignment problems, because of the relation of this problem to the graph colouring problem, which has been widely studied by the mathematical community. In particular, to capture the potential interferences between nearby vertices, our model links some node pairs when the distance between them is below an interference radius R (to reflect interferences): AP–AP pairs will be linked provided that the

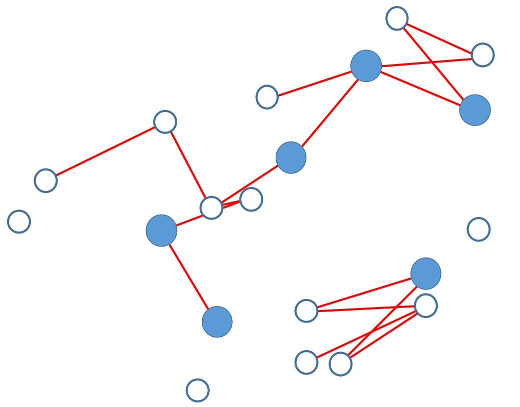


Figure 1: Interference graph model.

distance condition is met, AP–WD pairs only when the device is not associated to that AP, and WD–WD pairs only if both devices are associated to different APs, since the communications among the elements connected to the same AP are coordinated and do not interfere (Figure 1).

To quantify the goodness of the different network colourings, we have used the concept of utility, which is closely related to the perceived throughput and signal to noise ratio ($SINR$), which in turns depends on the interference graph. For a detailed description of how we model interference and how we map it to a utility function depending on the channel assignment and the interference graph, readers are encouraged to check [4]. For the purpose of this paper, it suffices to know that we derive a utility value between 0 and 1 (higher is better) for each wireless access point and client, and that this utility value depends on a propagation model and on the particularities of the underlying interference graph for the scenario.

3. CHANNEL ASSIGNMENT AS A NEGOTIATION PROBLEM

In this work, we propose to tackle the network-structured channel assignment problem in WiFi using automated negotiation techniques. We assume a multiattribute negotiation domain, where for a channel assignment problem with n_{AP} access points, a solution or deal S can be expressed as $S = \{s_i | i \in 1, ..., n_{AP}\}$, where $s_i \in \{1, \ldots, 11\}$ represents the assignation of a WiFi channel to the i-th access point. We also asume there are two network providers (commonly ISPs), so that APs belong to one of the providers. Each provider only has control over the frequency assignment for its own access points. According to this situation, $P = \{p_1, p_2\}$ will be the set of agents that will negotiate the frequency assignment. Each one of these agents will compute its utility for a certain deal or solution according to the model described in the previous section, as the sum of utilities for its associated APs and clients. As we will see in the following sections, the problem settings (high cardinality of the solution space and attribute interdependence) will make the utility functions highly complex, with multiple local minima and maxima.

From the assumption that the negotiation scenarios coming from Wi-Fi channel assignment will be highly nonlinear, and according to the discussion in [6], we have chosen a single text mediation protocol. In its simplest version, the negotiation protocol will be as follows (see Algorithm 1):

1. It starts with a randomly-generated candidate contract (S^c_0). In our case, this means assign a random channel to each access point.
2. In each iteration t, the mediator proposes a contract S^c_t to the rest of agents.
3. Each agent either accepts or rejects the contract proposed by the mediator.
4. The mediator generates a new contract S^c_{t+1} from the previous contracts and from the votes received from the agents and the process moves to step 2.

Algorithm 1: Single-text mediation algorithm

```
Input:
    P = {p_i}: set of providers (for this work, |P| = 2)
    A: set of access points
    T: maximum number of iterations
Output:
    S: final contract, corresponding to a channel assignment for
    each AP
t = 1
S^b = generate_random_contract(A)
while t ≤ T do
    S^c_t = S^b
    V = request_provider_votes(P, S^c_t)
    if reject ∉ V then
        S^b = S^c_t
        notify_mutual_agreement(P, S^c_t)
    end
    S^c_{t+1} = mutate_contract(S^b)
end
t = t + 1
S = S^b
notify_final_agreement(P, S)
```

This process goes on until either a maximum number of iterations is reached or another stop condition is met. The protocol, as defined, is rather generic and must be completed with the definition of the decision mechanism or *strategies* to be used by the negotiating agents and the mediator.

For the mediator, we have implemented a single-text mediation mechanism [6] for the generation of new contracts, which works as follows:

- If at time t all agents have accepted the presented contract S^c_t, this contract will be used as the base contract S^b to generate the next contract S^c_{t+1}. Otherwise, the last mutually accepted contract will be used.
- To generate the next candidate contract S^c_{t+1}, the mediator takes the base contract S_b and mutates one of its issues randomly. In our case of study, this would correspond to choosing a random access point and selecting a new random channel for it.
- After a fixed number of iterations, the mediator advertises the final contract, which will be the last mutually accepted contract.

For the agents, we have considered two different mechanisms to vote about the candidate contracts S^c:

- Hill-climber (HC): In this case, the agent behaves as a greedy utility maximizer. The agent will only accept

Algorithm 2: Hill-climber voting algorithm

```
Input:
    A_p ∈ A: set of access points belonging to the provider
    S_t^c: candidate contract sent by the mediator at time t
    S^b: last mutual agreement notified by the mediator
Output:
    v: vote of this provider (v ∈ {accept, reject})
if S^b = ∅ then
  | U^b = −∞
else
  | U^b = compute_utility(A_p, S^b)
end
U^c = compute_utility(A_p, S_t^c)
if U^c ≥ U^b then
  | v = accept
else
  | v = reject
end
```

a contract when it has at least the same utility for her than the previous mutually accepted contract (see Algorithm 2).

- Annealer (SA): In this case, we use a widespread non-linear optimization technique called *simulated annealing* [6]. When a contract yields a utility loss against the previous mutually accepted contract, there will be a probability for the agent to accept it nonetheless. This probability P_a depends on the utility loss associated to the new contract Δu, and also depends on a parameter known as *annealing temperature* τ, so that $P_a = e^{\frac{-\Delta u}{\tau}}$. Annealing temperature begins at an initial value, and linearly decreases to zero throughout the successive iterations of the protocol (see Algorithm 3).

Algorithm 3: Annealer voting algorithm

```
Input:
    A_p ∈ A: set of access points belonging to the provider
    S_t^c: candidate contract sent by the mediator at time t
    S^b: last mutual agreement notified by the mediator
    τ_0: initial annealing temperature
Output:
    v: vote of this provider (v ∈ {accept, reject})
if S^b = ∅ then
  | U^b = −∞
else
  | U^b = compute_utility(A_p, S^b)
end
U^c = compute_utility(A_p, S_t^c)
ΔU = U^b − U^c
τ = τ_0(1 − t/T)
P_a = e^(−ΔU/τ)
if rand(0,1) ≤ P_a then
  | v = accept
else
  | v = reject
end
```

The choice of these two mechanisms is not arbitrary. *Simulated annealing* techniques have yielded very satisfactory results in negotiation for nonlinear utility spaces [7], and are the basis for several of our previous works [8]. Furthermore, as discussed in [6], the comparison between *hill-climbers* and *annealers* allows to assess whether the scenario under consideration is a highly complex one, since in such scenarios greedy optimizers tend to get stuck in local optima, while the *simulated annealing* optimizer tends to escape from them.

4. EXPERIMENTAL SETTINGS

In this paper, we have made the assumption that both APs and clients are randomly distributed throughout the environment, and that clients associate to the AP which is closer to them. With these assumptions, we have generated scenarios varying the number of APs (15, 50 and 100) and the number of clients per AP (1 and 5). For each of these combinations of parameters we generated 50 different graphs, for a total of 450 scenarios. This allowed us to have a wide range of problem sizes (from tens of nodes to roughly one thousand), and also a wide diversity (due to the randomization of node placement).

In addition to the negotiation techniques under study, presented in the previous section, we have included a comparison with two reference techniques:

- Least Congested Channel Search (LCCS): a sequential baseline inspired by the most commonly used technique for Wi-Fi channel assignment [1]. In this algorithm, APs are activated in sequence, and each of them chooses the channel where it finds the lowest interferences from other active APs and their clients. If there are several least congested channels, it chooses one of them randomly.

- Particle Swarm Optimization (ALPSO): we wanted to have, as a reference, a nonlinear optimizer using complete information. We have chosen a parallel augmented Lagrange multiplier particle swarm optimizer, which solves nonlinear non-smooth constrained problems using an augmented Lagrange multiplier approach to handle constraints [5].

For each of the aforementioned 450 scenarios, we did 20 repetitions with each of the benchmarked techniques, recording the achieved social welfare (sum of utilities for both providers) and the *fairness* of the agreements as defined in [2]. We consider both social welfare and fairness are the key performance indicators in this settings, since network providers would want to maximize the overall satisfaction for their customers, while preventing user disatisfaction due to unfair allocations which could lead to user discontinuing their subscriptions to network services.

5. EXPERIMENTAL RESULTS

In this section, we describe and discuss the results of our experiments. Firstly, we study the performance of the evaluated techniques in the different scenario categories according to the scenario generation parameters (number of APs and number of clients per AP). Figure 2 shows the average utility obtained by each approach for all the graphs in each category. We can see that, for the less complex scenarios, all approaches perform reasonably well, with slightly better results obtained by the particle swarm optimizer (*ALPSO*). As the scenarios grow more complex, we can see the performance of the *LCCS* approach turns worse, which is reasonable since the size of the solution space becomes larger. We can also note significant increasing distance between the performances of the hill climber (*HC*) and the annealer (*SA*) negotiators. This confirms our hypothesis that these scenarios are highly nonlinear [6]. We can also see that, for the more complex scenarios, the *SA* negotiator significantly outperforms the particle swarm optimizer. This is a remarkable

Table 1: Ratio between utility and fairness.

(APs,WDs)	LCCS	HC	SA	ALPSO
(15, 15)	389.8	**871.6**	850.1	858.3
(15, 75)	451.1	752.5	**902.1**	787.9
(15, 150)	639.4	890.5	**1121.7**	941.2
(50, 50)	506.5	734.8	**807.9**	700.0
(50, 250)	921.5	1049.4	**1178.2**	1014.4
(50, 500)	1490.3	1634.1	**1831.5**	1617.7
(100, 100)	639.7	751.2	**836.2**	709.9
(100, 500)	1741.3	1851.8	**1983.9**	1783.0
(100, 1000)	2874.9	3061.2	**3233.3**	2908.1

result, specially taking into account that *SA* reaches the optimum faster than the *ALPSO* optimizer (the SA negotiator is roughly 10 times faster than the complete information optimizer).

As discussed above, the fairness of the outcomes is also an important performance indicator for the Wi-Fi channel assignment scenario. To account for this, in table 1 we show the average of the ratio between the social welfare obtained $SW(S)$ for each agreement solution S, and the fairness $F(S)$ associated with said solution. Note that, as defined in [2], better fairness values are smaller, and therefore high values for this ratio would appear for solutions which combine high social welfares and fair outcomes. We can see that the SA negotiator is the best options in terms of combined utility and fairness, except for the simplest scenarios, where the hill climber achieves the best results. This confirms SA as the mechanism of choice for these scenarios.

Furthermore, we analyze how the relative performance of the benchmarked approaches is influenced by the structure of the scenario graphs. Figure 3 shows the social welfare results of the different approaches with respect to the relative diameter of the underlying interference graph, which is a metric commonly used in graph theory. We have represented the ratio between the average utility achieved by each approach in the 20 runs for a given graph, and the average utility obtained by ALPSO for the same graph (hence the dashed line in the figures corresponds to the ALPSO 1.0 baseline). We can see that the advantage of using SA is higher when the relative diameter of the graph is smaller (that is, for highly connected graphs). This is specially interesting from an application point of view, since a highly connected graph reflects a situation with many interferring sources, which is the usual situation in the real world, and the one where it is more important to optimize channel assignment.

Finally, we study the diversity of the scenarios generated in terms of the different structural and relational scenario metrics we proposed in [9]. Figure 4 shows the distributions of provider utilities for $100,000$ random contracts in some of our scenarios with 15 APs. The colors represent the density of points in each area, creating what we call *utility histograms.* These utility histograms allow us to see the diversity between the scenarios. For instance, some scenarios have the areas of highest density of contracts in middle utility values –see Figures 4 (b) and (k)–, while others have the highest densities in mutually high utility contracts –see Figures 4 (a), (e), and (j)–. Intuitively, it will be easier for an approach to find higher social welfare outcomes in the latter cases than in the former. There are also significant differences in the symmetry of the histograms. Figures (e) and (h), for instance, correspond to scenarios where most contracts yield higher utilities to Provider 2, while Provider 1 has more variance in its potential utility. There are also other patterns in the diagrams which are worth noting, such as the coexistence of "continuous" scenarios (where all the utility values between a minimum and a maximum can be achieved) with "discrete" scenarios (where there are utility gaps in the diagram). Furthermore, we observe that, for a given scenario graph, different mappings of APs to the two providers generate significant variations on the resulting utility histograms. Figures 4 (m), (n), and (o) show this effect for our scenario *15-15-24* -the one also shown in (k)-. We can observe a radical change on the contract densities, the symmetry patterns and the gap patterns in the historgams, which we expect to result in significally different challenges for negotiation. The study of the effect of these observable patterns on the relative performance of the different negotiation approaches, or the correlation between the properties of the underlying graphs modelling the problem and the resulting utility histogram open a promising area of exploration. In particular, the clear influence of the distribution of providers suggests that the connection of APs to providers is worth considering, which could lead to a model based on multilayer networks [10] rather than a flat graph model.

6. DISCUSSION AND CONCLUSIONS

This paper presents a problem family inspired by Wi-Fi channel assignment to be used as a benchmark for nonlinear negotiation approaches. We generate an wide variety of scenarios from this family, and perform an extensive experiment set with several negotiation and reference approaches. Experiments show that nonlinear negotiation approaches significantly outperform the references in both social welfare and fairness. This is a relevant result, since these are usual key performance indicators to apply nonlinear negotiation approaches to real environment. It also shows significant differences between hill-climber negotiators and annealers, which is a clear indicator of the nonlinearity of the scenarios, as discussed in [6].

This works opens a variety of future work lines. We are interested in determining which features of different scenarios may help us determine the more suitable negotiation approach to be used in a specific situation, in a similar way as we did in [9]. We would like to integrate our scenario family with existing benchmarking infrastructures, like GENIUS or Negowiki. Furthermore, it would be interesting to have the scenarios tested against community-created agents in next ANAC competition.

7. ACKNOWLEDGEMENTS

This work has been supported by the Spanish Ministry of Economy and Competitiveness grants TIN2014-61627-EXP, MTM2014-54207, and TEC2013-45183-R and by the University of Alcalá through CCG2015/EXP-053.

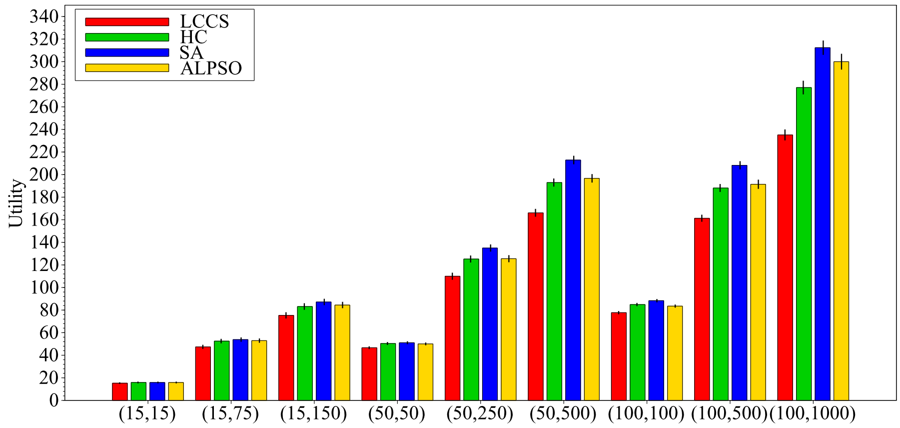


Figure 2: Utility for the different techniques.

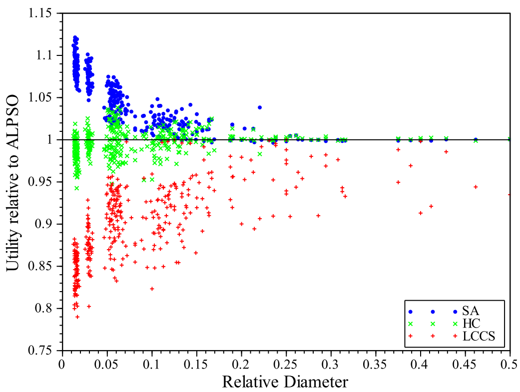


Figure 3: Utility relative to ALPSO as a function of the relative diameter.

REFERENCES

[1] Murali Achanta. Method and apparatus for least congested channel scan for wireless access points. *US Patent App. 10/959,446*. 2006.

[2] Katsuhide Fujita, Takayuki Ito, and Mark Klein. A secure and fair protocol that addresses weaknesses of the Nash bargaining solution in nonlinear negotiation. *Group Decision and Negotiation*, 21(1), 29–47. 2012.

[3] Hattori Hiromitsu, Mark Klein, and Takayuki Ito. Using iterative narrowing to enable multi-party negotiations with multiple interdependent issues. In *Proceedings of 6th international joint conference on Autonomous agents and multiagent systems (AAMAS)*, pages 247:1-247:3, Honolulu (HI), May 2007. ACM.

[4] Enrique de la Hoz, Jose Manuel Gimenez-Guzman, Ivan Marsa-Maestre and David Orden. Automated Negotiation for Resource Assignment in Wireless Surveillance Sensor Networks. *Sensors*, 15(11):29547–29568, 2015.

[5] Peter W. Jansen, and Ruben E. Perez. Constrained Structural Design Optimization via a Parallel Augmented Lagrangian Particle Swarm Optimization Approach. *Computers & Structures*, 89(13-14):1352–1366, July 2011.

[6] Mark Klein, Peyman Faratin, Hiroki Sayama, and Yaneer Bar-Yam. Negotiating complex contracts. *Group Decision and Negotiation*, 12(2):111–125, March 2003.

[7] Fabian Lang, and Andreas Fink. Learning from the metaheuristics: Protocols for automated negotiations. *Group Decision and Negotiation*, 24(2):299–332, March 2015.

[8] Ivan Marsa-Maestre, Miguel A. Lopez-Carmona, Juan R. Velasco, Takayuki Ito, Mark Klein, and Katsuhide Fujita. Balancing utility and deal probability for auction-based negotiations in highly nonlinear utility spaces. In *Proceedings of the 21st international joint conference on Artificial intelligence (IJCAI)*, pages 214–219, Pasadena (CA), July 2009.

[9] Marsa-Maestre, I.; Klein, M.; Jonker, C.M.; Aydoğan, R. From problems to protocols: Towards a negotiation handbook. *Decision Support Systems*, 60:39–54. 2014.

[10] Kivelä, M., Arenas, A., Barthelemy, M., Gleeson, J.P., Moreno, Y., and Porter, M.A. Multilayer networks. *Journal of Complex Networks*, 2(3):203–271, July 2014.

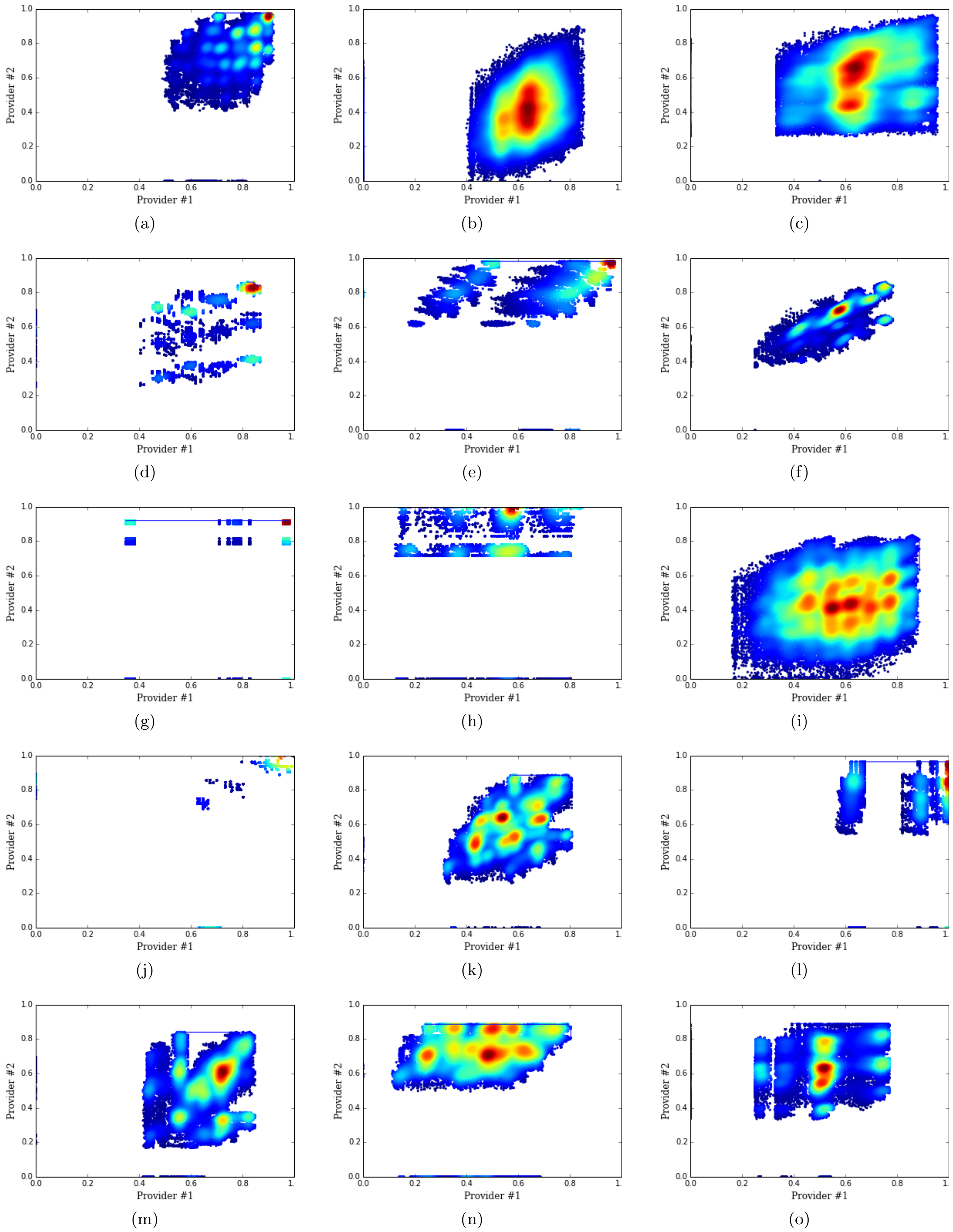


Figure 4: Utility histograms for selected scenarios with 15 APs.

Compromising Strategy using Analytic Hierarchy Process for Multiparty Closed Automated Negotiations

Hiroyuki Shinohara and Katsuhide Fujita
Faculty of Engineering, Tokyo University of Agriculture and Technology
Koganei, Tokyo, Japan
shinohara@katfuji.lab.tuat.ac.jp, katfuji@cc.tuat.ac.jp

ABSTRACT

Multilateral multi-issue closed negotiation is an important class of real-life negotiations. Negotiation problems usually have such constraints as an unknown opponent's utility in real time and time discounting. Recently, the attention in this field has shifted from bilateral to multilateral approaches. In multilateral negotiations, agents must simultaneously consider plural opponents. We propose a negotiation strategy for evaluating opponent bids using the analytic hierarchy process (AHP), which we extend to three-party negotiations by combining the estimated utilities of the opponents. The estimation of opponent utility is decided by counting the opponent bids and the importance allocations. We also propose a compromise strategy that considers evaluation values using AHP and the discount factors. We experimentally evaluated the efficiency of our method by comparing our proposed agent with the top five state-of-the-art agents in the individual utility categories of ANAC 2015. The experimental results demonstrate that our proposed method using AHP obtained higher social welfare than the other state-of-the-art agents in less negotiation time.

Categories and Subject Descriptors

I.2.11 [**Artificial Intelligence**]: Distributed Artificial Intelligence - Multi-agent System

Keywords

Automated Multi-issue Negotiation, Automated Negotiating Agents Competition, Multilateral Negotiation

1. INTRODUCTION

Negotiation is a critical process for forming alliances and reaching trade agreements. Research in the field of negotiation originated in various disciplines including economics, social science, game theory, and artificial intelligence ([9, 17, 18, 20] etc.). If we can develop automated negotiation under realistic settings in a multi-agent field, efficient negotiations and decision-making support will be accomplished.

Motivated by the challenges of bilateral negotiations among automated agents, the automated negotiating agents competition (ANAC) was first organized in 2010 to facilitate research in automated multi-issue closed negotiation [1, 2, 11, 23, 13, 14]. ANAC's setup is a realistic model that includes time discounting, closed negotiations, and alternative offering protocols. By analyzing the ANAC results, the trends of the strategies of automated negotiations and important factors for developing competition have been shown [3]. Other effective automated negotiating agents have also been proposed through competitions [5, 6].

Multiple parties are crucial for achieving automated negotiations in real life. Many real-world negotiation problems assume multiparty situations because negotiations on the web are becoming more common. Although an automated negotiation strategy can be effective for bilateral negotiation, it is not always possible or desirable to apply such a strategy to multiparty negotiations [8]. In other words, designing more efficient automated negotiation strategies against various negotiating opponents in multiparty situations remains an open and interesting problem.

In this study, we propose a negotiation strategy to evaluate opponent bids using the analytic hierarchy process (AHP). Opponent utility functions are estimated using importance allocations and geometric means by counting the offers proposed by opponents. Our agent proposes bids with the highest evaluation value determined by AHP. In addition, we offer proposal, acceptance, and compromising strategies using AHP. Our proposed strategy can adjust the compromising rate by considering the evaluation values using AHP and the discount factor. Our agent has the following three advantages:

1. Accurately estimates opponent utility functions
2. Reaches agreements with high social welfare
3. Adjusts compromising speed based on discount factors

We experimentally evaluate the efficiency of our proposed method by comparing with state-of-the-art tournament negotiation agents. The experimental results demonstrate that our proposed method using AHP obtains higher social welfare than the other state-of-the-art agents in less negotiation time.

The remainder of this paper is organized as follows. First, we describe related works, the negotiation environments, and the stacked alternating offers protocol for multilateral

ACM ISBN 978-1-4503-2138-9.
DOI: 10.1145/1235

negotiation (SAOP). Next we propose a method for estimating opponent utility functions and compromising negotiation strategies using AHP. We present our experimental analysis and finally our conclusions.

2. RELATED WORKS

This paper focuses on research in the area of multi-issue closed negotiation, which is an important class of real-life negotiations. Closed negotiation means that opponents do not reveal their preferences to each other. Negotiating agents designed using a heuristic approach require extensive evaluation, typically through simulations and empirical analysis, since it is usually impossible to predict precisely how the system and the constituent agents will behave in a wide variety of circumstances. Motivated by the challenges of bilateral negotiations between people and automated agents, the automated negotiating agents competition (ANAC) was organized in 2010 [1, 2, 11, 23, 13, 14] to facilitate research in the area of multi-issue closed negotiation.

The following are the competition's declared goals: (1) to encourage the design of practical negotiation agents that can proficiently negotiate against unknown opponents in a variety of circumstances; (2) to provide a benchmark for objectively evaluating different negotiation strategies; (3) to explore different learning and adaptation strategies and opponent models; (4) to collect state-of-the-art negotiating agents and scenarios and make them available to the wider research community. The competition was based on the GENIUS environment: the General Environment for Negotiation with Intelligent multi-purpose Usage Simulation [19].

By analyzing the ANAC results, the stream of the ANAC strategies and the important factors for developing the competition have been shown. Baarslag et al. presented an in-depth analysis and the key insights gained from ANAC 2011 [3]. This paper mainly analyzed different strategies using the classifications of agents with respect to their concession behavior against a set of standard benchmark strategies and empirical game theory (EGT) to investigate their robustness. It also showed that even though the most adaptive negotiation strategies are robust across different opponents, they are not necessarily the ones that win competitions. Furthermore, the EGT analysis highlights the importance of considering metrics.

Chen and Weiss proposed a negotiation approach called OMAC, which learns an opponent's strategy to predict the future utilities of counteroffers by discrete wavelet decomposition and cubic smoothing splines [7]. They also presented a negotiation strategy called EMAR for such environments that relies on a combination of Empirical Mode Decomposition (EMD) and Autoregressive Moving Average (ARMA) [6]. EMAR enables a negotiating agent to acquire an opponent model and to use it to adjust its target utility in real time on the basis of an adaptive concession-making mechanism.

Hao and Leung proposed a negotiation strategy named ABiNeS, which was introduced for negotiations in complex environments [12]. ABiNeS adjusts the time to stop exploiting the negotiating partner and also employs a reinforcement-learning approach to improve the acceptance probability of its proposals.

Williams et al. proposed a novel negotiating agent based on Gaussian processes in multi-issue automated negotiation against unknown opponents [24]. Baarslag et al. focused on the acceptance dilemma; accepting the current offer may be suboptimal, since better offers might still be presented [4].

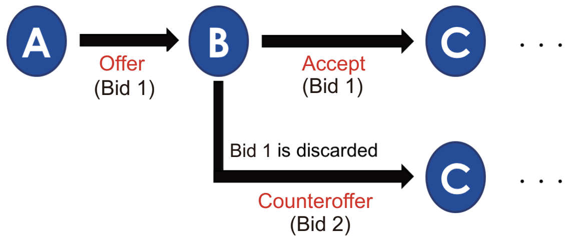


Figure 1: Example of SAOP with Three Agents

Kawaguchi et al. proposed a strategy for compromising the estimated maximum value based on estimated maximum utility [16]. Even though these papers made important contributions for bilateral multi-issue closed negotiation, they failed to deal with multi-times negotiation with learning and reusing the past negotiation sessions. After that, Fujita [10] proposed a compromising strategy by adjusting the speed of reaching agreements using the Conflict Mode and focused on multi-times negotiations.

However, many of these strategies only focused on the bilateral negotiation. In real life, negotiation problems should assume multiparty situations. In this paper, we demonstrate that the novel compromising negotiation strategy based on the analytic hierarchy process can negotiate effectively in multilateral negotiation.

3. MULTILATERAL NEGOTIATIONS PROTOCOL

3.1 Multilateral Negotiation Environments

The Stacked Alternating Offers Protocol for Multilateral Negotiation (SAOP) is a simple extension of a bilateral alternating offers protocol [21]. According to it, all participants get one turn per round; turns are taken clockwise around the table. But our study focuses on negotiations among three parties. Our proposed method and the negotiation situations can be extended to N-lateral negotiations.

We assume that three agents (A_1, A_2, A_3) are involved in the negotiation. First, A_1 makes an offer that is immediately observed by the other agents. Whenever an offer is made, A_2 and A_3 can take the following actions:

- Make a counteroffer: reject and override the previous offer
- Accept the offer
- Walk away: end the negotiation without an agreement

Then $A_{mod(n,3)+1}$ ($n = 1, 2, 3$) selects its next action from *Counteroffer, Accept, or Walk Away.* This process is repeated in clockwise turns until an agreement is reached or a deadline passes. To reach an agreement, all parties must accept the offer. If no agreement is reached by the deadline, the negotiation fails.

Figure 1 shows an example of SAOP with three agents: (1) agent A makes a counteroffer, (2) agent B accepts agent A's offer ($Bid1$) or makes a counteroffer ($Bid2$), and (3) agent C accepts or makes a counteroffer by rejecting the last offer

Table 1: Example of Utility Space

Issue	Issue Weight	Value	Value Evaluation
Issue 1	0.4	Value 1-1	1.00
		Value 1-2	0.35
		Value 1-3	0.55
Issue 2	0.6	Value 2-1	0.80
		Value 2-2	0.10
		Value 2-3	0.40
		Value 2-4	1.00

by agent A. In this negotiation, the agreement failed when agents B or C select a new counteroffer.

The parties negotiate over *issues*, each of which has an associated range of alternatives or *values*. A negotiation outcome consists of a mapping of every issue to a value, and set Ω of all possible outcomes is called the negotiation *domain*. This domain is the common knowledge shared by the negotiating parties and remains fixed during a single negotiation session. All parties have certain preferences prescribed by a *preference profile* over Ω. These preferences can be modeled by utility function U that maps possible outcome $\omega \in \Omega$ to a real number in range $[0, 1]$. In contrast to the domain, a preference profile is private information.

A negotiation lasts a predefined time in seconds (*deadline*). The timeline is normalized, i.e., time $t \in [0, 1]$, where $t = 0$ represents the negotiation's start and $t = 1$ represents its deadline. Apart from a deadline, a scenario may feature discount factors that decrease the utility of the bids under negotiation as time passes. Let δ in $[0, 1]$ be the discount factor. Let t in $[0, 1]$ be the current normalized time, as defined by the timeline. We compute discounted utility U_D^t of outcome ω from undiscounted utility function U as follows:

$$U_D^t(\omega) = U(\omega) \cdot \delta^t.$$

At $t = 1$, the original utility is multiplied by the discount factor. If $\delta = 1$, the utility is not affected by time, and such a scenario is considered undiscounted.

3.2 Weighted-Sum Utility Function

A bid is a set of chosen values $s_1 \dots s_M$ for each M issue (I). Each value is assigned evaluation value $eval(s_i)$ in the utility function, and each issue is assigned normalized weight (w_i, $\sum_{i \in I} w_i = 1$) in the utility function. The utility is the weighted sum of the normalized evaluation values. When three agents negotiate, each agent has its own utility function.

According to issue's element v_j and v_j's evaluation $eval(v_j)$, the utility function is expressed as

$$U(\overrightarrow{s}) = \sum_{j=1}^{M} (w_j \times eval(v_j)). \quad (1)$$

The utility function refers to each agent's preference, which is calculated by the weights of the issues and an evaluation value of the elements for each issue.

Table 1 shows an example of the weighted-summing utility function. Bid $\overrightarrow{s} = (Value1-1, Value2-2)$'s utility, which is $U(\overrightarrow{s}) = 0.4 \times 1.0 + 0.6 \times 0.1 = 0.46$. When the discount factor is considered, the real utility values are reduced as time passes.

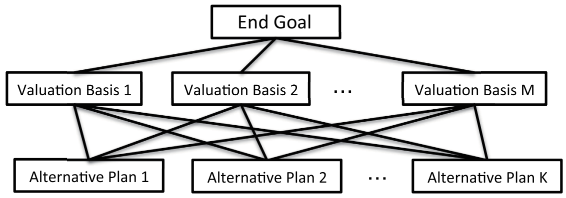


Figure 2: Hierarchical Structure of AHP

4. ANALYTIC HIERARCHY PROCESS

AHP, which is a technique for analyzing decisions through pairwise comparisons, relies on the judgments of experts to derive priority scales [22] as well as on a hierarchical structure. The final goal is set at the top of the hierarchical structure. Alternatives to achieve a goal are set at the bottom of the structure, and the criteria to evaluate the alternatives are set in the middle layer. AHP's hierarchical structure is shown in Figure 2. The weight of each criterion and the evaluation of each alternative by each criterion must be determined to evaluate the usefulness of the alternatives to the final goal. Pairwise comparison is used to determine the weights and the evaluation of alternatives. For the weight of each criterion, the differences between the importance of two criteria are determined by the evaluation method in Table 2.

By performing pairwise comparison among all the evaluation criteria, the pairwise matrix is decided. The x_{m_1,m_2} of a pairwise comparison matrix compares the importance of evaluation criteria i_{m_1} with evaluation criteria i_{m_2}:

$$X = \begin{pmatrix} x_{11} & x_{12} & \dots & x_{1M} \\ x_{21} & x_{22} & \dots & x_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ x_{M1} & x_{M2} & \dots & x_{MM} \end{pmatrix}.$$

After generating the pairwise comparison matrix, the geometric mean of each row is calculated as an approximation of the eigenvalues. The evaluation weight criteria are determined by the geometric mean's normalization, which totals 1. To evaluate the alternative to each evaluation criterion, this process is calculated between the alternative and an evaluation criterion. Then the evaluations of the alternatives are determined by Eq. 2, and the maximum evaluation alternative is considered the most suitable choice:

$$E_{AHP}(\overrightarrow{s}) = \sum_{j=1}^{M} (e_{i_j}^{\overrightarrow{s}} \times w_j). \quad (2)$$

(M: evaluation criteria, $e_{i_j}^{\overrightarrow{s}}$: evaluation of $\overrightarrow{s}$ by i_j, w_j: weight of i_j).

5. NEGOTIATION STRATEGY WITH AHP

5.1 Offering method with AHP

We employ AHP with multi-issue negotiation problems. When AHP corresponds to a multilateral negotiation problem, the following are the mappings between it and the multi-issue negotiation problems:

Table 2: Importance of Pair Comparisons

Importance Value	Meaning
1	Former and latter are equally important
3	Former is slightly more important than the latter
5	Former is more important than the latter
7	Former is much more important than the latter
9	Former is absolutely more important than the latter
Even Number	Complementary Value
Reciprocal Number	Latter is more important than the former

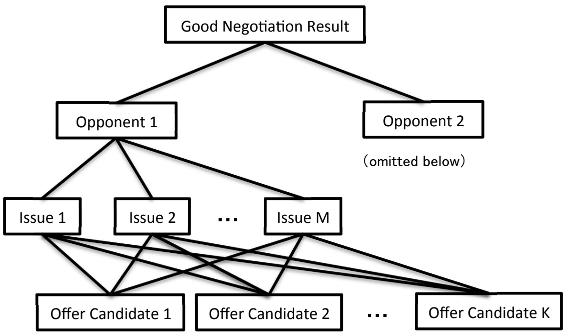


Figure 3: Hierarchical Structure of Multi-issue Negotiations

- Final goal = Improvement of social welfare
- Evaluation criteria = Opponents' utility function
- Alternative = Candidate proposal

On the basis of the above mapping, the hierarchical structure of multi-issue negotiation problems is shown in Figure 3. The evaluation value of the alternative corresponds to the utility of our proposal in multi-issue negotiations. To obtain the proposal's utility, the utility function must determine the weights and the evaluation values of each issue.

We estimate the opponent's utility functions by counting the opponent proposals. Then the estimated opponent utility, which is obtained by the estimated utility function, is used as the evaluation values of AHP. According to SAOP [1], which is an extension of alternating offers [21], an element frequently proposed by an opponent is expected to be an important bid. Therefore, we can estimate the opponent's utility function by counting the occurrences of each element in our proposal. In addition, we propose an efficient offer-making method using AHP evaluation, which uses the estimated opponent's utility. In bilateral negotiation, this counting method is appropriate for predicting opponent utility [15].

A pairwise comparison (importance allocation) and a geometric mean (Section 4) are used to determine the weight of the evaluation criterion and the evaluation value of the alternatives in AHP. At this point, since the negotiation setting supposes that we cannot obtain the opponent information beforehand, the elements, which are proposed many times by opponents, are considered to have high utility for those opponents. The importance is allocated by counting the number of occurrences of the element in an opponent's proposal.

Table 3: Counting Table with Values of Each Issue

Issue	Value	Occurrence
Issue 1	Value 1-1	230
	Value 1-2	130
	Value 1-3	140
Issue 2	Value 2-1	30
	Value 2-2	80
	Value 2-3	60
	Value 2-4	330

Table 4: Importance Allocation Table for Each Element

Range of occurrences	Importance	Applied element
$130 \sim 135$	1	Value1-2
$136 \sim 140$	2	Value1-3
$141 \sim 150$	3	
$151 \sim 160$	4	
$161 \sim 170$	5	
$171 \sim 185$	6	
$186 \sim 200$	7	
$201 \sim 215$	8	
$216 \sim 230$	9	Value1-1

An example of counting 500 proposals is shown in Table 3 for the utility function given in Table 1. The minimum and maximum occurrences of each element are calculated by counting the proposals. The importance is allocated by the multiple parts of the ranges decided by the minimum and maximum count depending on the occurrences. The range of the count is determined by splitting the range [*(the minimum count)*, *(the maximum count)*]. The importance $1 \sim 9$, which is shown in Table 2, is allocated to each split range. The importance of each element is decided by choosing a range that includes the occurrences of each element. For $Issue1$, the importance of each element is allocated, as shown in Table 4. For example, $Value1-2$ is allocated to importance 1, which was proposed 130 times in the negotiation. Similarly, $Value1-3$ is allocated to importance 2. At this point, the way of splitting the range is decided as shown Table 4,5 by the pre-experimental results.

The deviation of the maximum occurrences of each element of each issue is used to calculate each issue's importance. The same process determines the importance of each issue using minimum and maximum occurrences, for example, the importance allocation shown in Table 5 for $Issue1, 2$, which is shown in Table 1.

Table 5: Importance Allocation Table for Each Issue

Range of occurrences	Importance	Applied issue
230 ∼ 235	1	Issue1
236 ∼ 240	2	
241 ∼ 250	3	
251 ∼ 260	4	
261 ∼ 270	5	
271 ∼ 285	6	
286 ∼ 300	7	
301 ∼ 315	8	
316 ∼ 330	9	Issue2

From the above, the predicted utility functions of the opponents are generated by the geometric mean, as an approximation of the eigenvalues, based on the decided importance of each issue and each element of each issue. When the matrix row number is m_1 and the matrix column number is m_2, matrix element $x_{m_1 m_2}$ is represented by importance $I(i_{m1})$ of Issue i_{m1} and importance $I(i_{m2})$ of Issue i_{m2}, given by Eq. 3:

$$x_{m_1 m_2} = I(i_{m_1})/I(i_{m_2}). \tag{3}$$

In addition, the comprehensive evaluation value of the proposal is determined by Eq. 4 using the predictive utility function for each opponent:

$$E_{AHP}(\overrightarrow{s}) = \sum_{a \in Ag} (U^a_{exp}(\overrightarrow{s}) \times w_a), \tag{4}$$

where $U^a_{exp}(\overrightarrow{s})$ is the predicted utility of the proposals of opponent $a(\in Ag)$ and w_a is the weight of opponent a. At this point, w_a is decided as 0.5 by the pre-experimental results.

5.2 Proposal of Bids using AHP

Our agent employs the evaluation values of bids using AHP to generate the bids of opponents. First, it generates multiple bids which have an utility within particular range. Eq. 5 shows the width of its own utilities for generating proposals. $u(t)$ is the utility of generating the proposals in time t. This value is decided using the compromising function:

$$[(1 - 0.05)u(t), (1 + 0.05)u(t)]. \tag{5}$$

Next, our agent evaluates all the alternative bids using Eq. 4 and proposes bids with maximum evaluation values using AHP (Eq. (6)):

$$\vec{s} = \{\vec{s'} | \arg\max_{\vec{s'} \in S} E_{AHP}(\vec{s'})\},$$

where S is the set of all alternative proposals.

5.3 Compromising Strategy using AHP

After that, we assume that opponents use the same estimation method with AHP. When there are three agents A, B and C as negotiation participants, Eq. 6 is the evaluation values using AHP from Agent B. $U_A(\vec{s})$ and $U_C(\vec{s})$ are the evaluation values of the proposal bids ($\vec{s}$) using the utility functions of agents A and C. w_A and w_C are the weights of adjusting the importance of both agents:

$$E^B_{AHP}(\vec{s}) = (U_A(\vec{s}) \times w_A) + (U_C(\vec{s}) \times w_C). \tag{6}$$

Equation 6 is the weighted sum of the utility of opponents. In other words, Eq. 6 means the compromising rates to other opponents when $\vec{s}$ is proposed by Agent B.

We assume that the opponents compromise at a constant rate. In this situation, Eq. 6 is supposed to be the uniform distribution within a certain width. Eq. 7 is the expected maximum evaluation value given by opponent a' in time t when $\mu^{a'}_{AHP}(t)$ and $\sigma^{a'}_{AHP}(t)$ are the average and variance of all the bids using the AHP of opponent a':

$$\begin{aligned} emax^{a'}_{AHP}(t) =& \mu^{a'}_{AHP}(t) \\ &+ (1 - \mu^{a'}_{AHP}(t)) \times \sqrt{12}\sigma^{a'}_{AHP}(t). \end{aligned} \tag{7}$$

This equation is based on a previous work [16]. Equation 8 shows the minimum evaluation value of AHP provided by opponents in the future:

$$emax_{AHP}(t) = \min_{a' \in Op} emax^{a'}_{AHP}(t) \tag{8}$$
$$(Op : Opponents'\ Set).$$

Equation 9 is defined as the compromising function using the minimum AHP evaluation value ($emax_{AHP}(t)$) which expected to opponents in the future:

$$u(t) = 1.0 - mcd \times t^{(\delta^\alpha / emax_{AHP}(t)^\beta)} \tag{9}$$
$$(u(t) :\ utility\ in\ time\ t,\ \delta :\ discount\ factor,$$
$$mcd :\ limitation\ value\ of\ compromising,$$
$$\alpha, \beta :\ parameters\ of\ adjusting\ the\ compromise).$$

Figures 4 and 5 show the changes of the compromise function (Eq. 9) as time (t) passes. As discount factor (δ) becomes small, in other words, the discounting of the utility becomes larger, Eq. 9 compromises rapidly. On the other hand, the compromise speed is slow when $emax_{AHP}(t)$ is small, because the opponent's compromise was not expected. Our agent can effectively compromise with opponents, considering the discount factor and the opponents' strategy by employing this compromising strategy. In addition, our agent can adjust the compromising speed as the discount factor, the opponent cooperation, and some parameters.

5.4 Acceptance Strategy using AHP

Acceptance of the proposal is decided using the following acceptance rate. Acceptance rate ($p(\vec{s})$) of bid ($\vec{s}$) by opponent a' is defined by Eq. 10:

$$p(\vec{s}) = \frac{t^5}{5} + \{E^{a'}_{AHP}(\vec{s}) - emax^{a'}_{AHP}(t)\} + \{U(\vec{s}) - u_t\} \tag{10}$$
$$(t :\ time,\ U(\vec{s}) :\ utility\ of\ \vec{s}\ for\ itself,$$
$$E^{a'}_{AHP}(\vec{s}) :\ evaluation\ value\ of\ \vec{s}\ by\ opponent\ a').$$

Equation 10 is defined by reference to a previous work [16]. The first member considers the negotiation time, the second member considers the difference between the evaluation value and the expected maximum value provided by opponents in the future, and the third member considers the difference between the compromising threshold by itself

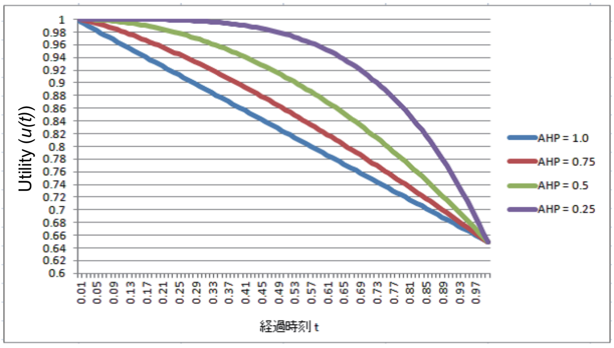


Figure 4: Compromising Function as Time (t) Passes ($\delta = 1.0, mcd = 0.35, \alpha = 1.0, \beta = 1.0$)

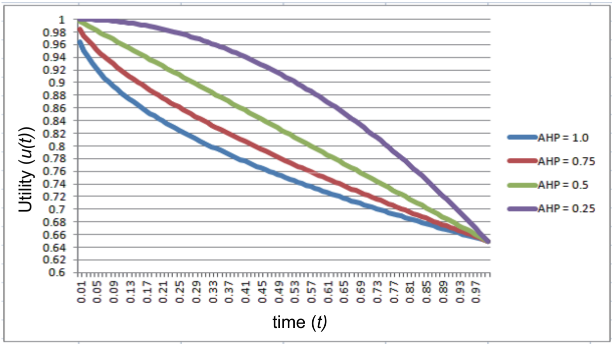


Figure 5: Compromising Function as Time (t) Passes ($\delta = 0.5, mcd = 0.35, \alpha = 1.0, \beta = 1.0$)

and the proposed bids by the opponent. This acceptance probability space decreases smoothly as time passes.

6. EXPERIMENTAL RESULTS

6.1 Experimental Setting

We experimentally evaluated the accuracy of the predicted and real utilities using the GENIUS environment, which is the general environment for negotiation with intelligent multi-purpose usage simulation [19].

In these experiments, we evaluated the proposed strategy (AgentHP) by comparing it with the state-of-the-art agents using the following settings:

- Round-robin competitions
- Number of agents in a negotiation: 3
- Opponents were the top five state-of-the-art agents in the individual utility categories in ANAC 2015, i.e., *Atlas3, ParsAgent, RandomDance, kawaii, and agent-Buyog.*
- Negotiation scenario: party_domain (profiles 1∼6). The *party_domain* domain was used as the negotiation scenario in the experiments. This scenario consists of six issues with three to six elements in each issue. Four patterns of discount factors were used with the *party_domain* domain: $1.0, 0.75$, and 0.5.

Table 6: Average Error Rates of Estimated Utility of Each Opponent

Opponent Agents	Average Error Rates
Atlas3	0.109158433
ParsAgent	0.15052428
RandomDance	0.156942134
kawaii	0.140489011
AgentBuyog	0.153200273

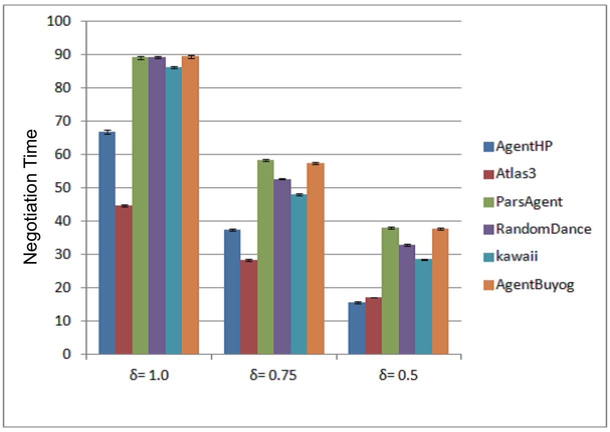


Figure 6: Results of negotiation time of agreements when discount factor is changed

- Deadline: 180 sec.
- Reservation value: 0
- Number of negotiations per tournament: 2400
- Tournament repetitions: 10

AgentHP is our proposed agent with a compromising strategy using AHP. The following are *AgentHP*'s parameters:

- Weights of opponents in Eqs. 4 and 6: $w_a = 0.50$
- Parameters in Eq. 9: $\alpha = 3.0, \beta = 1.0, mcd = 0.35$

These were the best parameters from pre-experiments.

6.2 Experimental Results

The error rates of the estimated utility of each opponent's in the 120 negotiations are shown in Table 6. The error rates are approximately 10 ∼ 15% for each opponent. For *Atlas3*, the error rates of the estimated utility are less than those of the other opponents. It is conceivable that the behavior of *Atlas*3 is suitable for the estimating method. Our agent can also accurately estimate opponent utility.

Figure 6 shows the average and standard deviations of the negotiation times of the agreements when the discount factors were varied. In $\delta = 1.0$, 0.75, *AgentHP*'s negotiation time is about 10 ∼ 20 shorter than the other agents, except for *Atlas3*. In $\delta = 0.5$, *AgentHP*'s negotiation time is the shortest. One reason for its shorter negotiation times is that our proposed method can easily reach agreements by

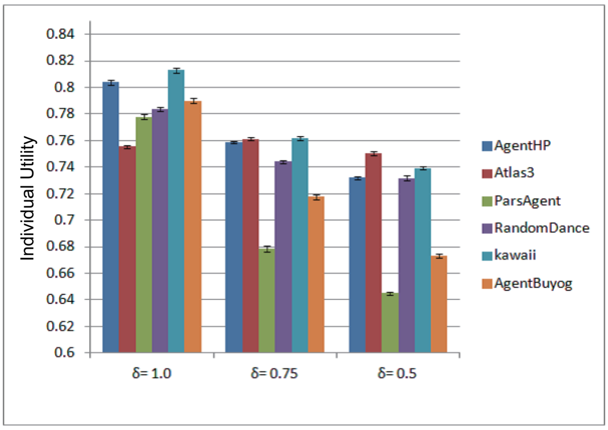


Figure 7: Results of individual utility of agreements when discount factor is changed

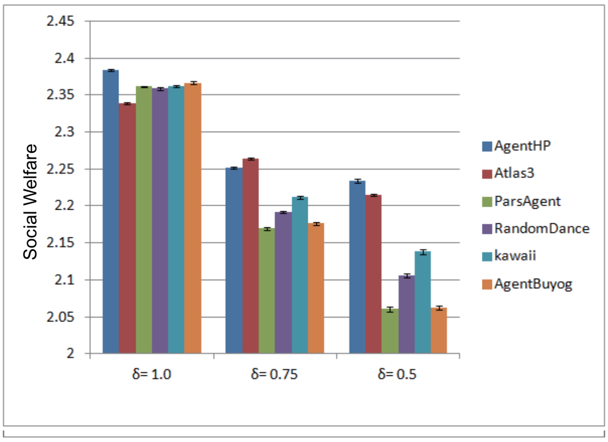


Figure 8: Results of social welfare when discount factor is changed

offering bids which have a maximum sum of the estimated bids of opponents. Another reason is that the compromising function (Eq. 9) tries to reach agreements in less time when the discounting factor becomes large.

Figure 7 shows the average and standard deviations of the individual utilities when the discount factors were varied. *AgentHP* is the second best after *Atlas3* when $\delta = 1.0$. However, our agent loses two positions when discount factor (δ) is small. This is because the compromising function (Eq. 9) rapidly compromises when discount factor (δ) is small.

Figure 8 shows the average and standard deviations of the social welfare values when the discount factors were varied. *AgentHP* has the best scores when $\delta = 1.0,\ 0.5$. In addition, the difference among our agent and the other agents except for *Atlas3* increases when discount factor (δ) is small due to the trade-off between social welfare and individual utility. The difference of the individual utility among our agent and other agents except for *Atlas3* in Figure 7 becomes smaller when discount factor (δ) is small. This is the opposite trend of the social welfare results. In addition, *AgentHP* can make agreements more quickly than the other agents. Our agent can obtain higher social welfare than other agents.

Therefore, our proposed method using AHP obtains greater social welfare than the other state-of-the-art agents. In addition, it can obtain high individual utility in “short” negotiation times.

7. CONCLUSIONS

In this study, we proposed a negotiation strategy to evaluate opponent bids with the analytic hierarchy process (AHP). The estimation of opponent utility is determined by counting opponent bids and the importance allocation. The AHP evaluation is determined by the opponents’ estimated utilities. We also proposed a compromise strategy based on evaluation values using AHP and a discount factor. We experimentally evaluated our method’s efficiency by comparing our proposed agent with the top five state-of-the-art agents in the individual utility categories in ANAC 2015. Our experimental results demonstrated that our proposed method using AHP obtained higher social welfare than other state-of-the-art agents in less negotiation time.

Future work will improve the estimations of opponent utility. To address this problem, our proposed approach must improve the method of importance allocation for each issue and the elements in pairwise comparisons. Another important task is the assessment of opponent strategy on the basis of modeling or machine learning to further enhance the proposed method.

8. ACKNOWLEDGEMENTS

This work was supported by CREST, JST, and JSPS KAKENHI Grant Numbers 15H01703 and 26730116.

9. REFERENCES

[1] R. Aydogan, T. Baarslag, K. Fujita, K. Hindriks, T. Ito, and C. Jonker. The fifth international automated negotiating agents competition(anac2015). http://www.tuat.ac.jp/ katfuji/ANAC2015/, 2015.

[2] R. Aydogan, T. Baarslag, K. Fujita, T. Ito, and C. Jonker. The fifth international automated negotiating agents competition (anac2014). http://www.itolab.nitech.ac.jp/ANAC2014/, 2014.

[3] T. Baarslag, K. Fujita, E. Gerding, K. Hindriks, T. Ito, N. R. Jennings, C. Jonker, S. Kraus, R. Lin, V. Robu, and C. Williams. Evaluating practical negotiating agents: Results and analysis of the 2011 international competition. *Artificial Intelligence Journal (AIJ)*, 198:73–103, 2013.

[4] T. Baarslag and K. Hindriks. Accepting optimally in automated negotiation with incomplete information. In *Proceedings of the 2013 international conference on Autonomous agents and multi-agent systems (AAMAS2013)*, pages 715–722, 2013.

[5] T. Baarslag and K. V. Hindriks. Accepting optimally in automated negotiation with incomplete information. In *Proceedings of the 12th International conference on Autonomous agents and multi-agent systems (AAMAS2013)*, pages 715–722, 2013.

[6] S. Chen, H. B. Ammar, K. Tuyls, and G. Weiss. Conditional restricted boltzmann machines for

negotiations in highly competitive and complex domains. In *Proceedings of the 23th International Joint Conference on Artificial Intelligence (IJCAI2013)*, pages 69–75, 2013.

[7] S. Chen and G. Weiss. An efficient and adaptive approach to negotiation in complex environments. In *Proceedings of the 19th European Conference on Artificial Intelligence (ECAI-2012)*, volume 242, pages 228–233, 2012.

[8] S. Fatima, S. Kraus, and M. Wooldridge. *Principles of Automated Negotiation*. Cambridge University Press, 2014.

[9] S. S. Fatima, M. Wooldridge, and N. R. Jennings. Multi-issue negotiation under time constraints. In *Proceedings of the first international joint conference on Autonomous agents and multiagent systems (AAMAS 2002)*, pages 143–150, New York, NY, USA, 2002.

[10] K. Fujita. Automated strategy adaptation for multi-times bilateral closed negotiations. In *Proceedings of the 13th Inter- national Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*, pages 1509–1510, 2014.

[11] K. Gal, T. Ito, C. Jonker, S. Kraus, K. Hindriks, R. Lin, and T. Baarslag. The forth international automated negotiating agents competition (anac2013). http://www.itolab.nitech.ac.jp/ANAC2013/, 2013.

[12] J. Hao and H.-F. Leung. Abines: An adaptive bilateral negotiating strategy over multiple items. In *2012 IEEE/WIC/ACM International Conferences on Intelligent Agent Technology (IAT-2012),*, volume 2, pages 95–102, 2012.

[13] T. Ito, C. Jonker, S. Kraus, K. Hindriks, K. Fujita, R. Lin, and T. Baarslag. The second international automated negotiating agents competition (anac2011). http://www.anac2011.com/, 2011.

[14] C. Jonker, S. Kraus, K. Hindriks, R. Lin, D. Tykhonov, and T. Baarslag. The first automated negotiating agents competition (anac2010). http://mmi.tudelft.nl/negotiation/tournament, 2010.

[15] S. Kakimoto and K. Fujita. Estimating pareto fronts using issue dependency for bilateral multi-issue closed nonlinear negotiations. In *2014 IEEE 7th International Conference on Service-Oriented Computing and Applications (SOCA)*, pages 289–293, 2014.

[16] S. Kawaguchi, K. Fujita, and T. Ito. Compromising strategy based on estimated maximum utility for automated negotiation agents competition (anac-10). In *24th International Conference on Industrial Engineering and Other Applications of Applied Intelligent Systems (IEA/AIE-2011)*, pages 501–510, 2011.

[17] S. Kraus. *Strategic Negotiation in Multiagent Environments*. Mit Press, October 2001.

[18] S. Kraus, J. Wilkenfeld, and G. Zlotkin. Multiagent negotiation under time constraints. *Artificial Intelligence*, 75(2):297 – 345, 1995.

[19] R. Lin, S. Kraus, T. Baarslag, D. Tykhonov, K. Hindriks, and C. M. Jonker. Genius: An integrated environment for supporting the design of generic automated negotiators. *Computational Intelligence*, 30:48–70, 2012.

[20] M. J. Osborne and A. Rubinstein. *Bargaining and Markets (Economic Theory, Econometrics, and Mathematical Economics)*. Academic Press, April 1990.

[21] A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50(1):97–109, 1982.

[22] T. Saaty. *The Analytic Hierarchy Process: Planning, Priority Setting, Resource Allocation*. Advanced book program. McGraw-Hill, 1980.

[23] C. R. Williams, V. Robu, E. Gerding, N. R. Jennings, T. Ito, C. Jonker, S. Kraus, K. Hindriks, R. Lin, and T. Baarslag. The third automated negotiating agents competition (anac2012). http://anac2012.ecs.soton.ac.uk, 2012.

[24] C. R. Williams, V. Robu, E. H. Gerding, and N. R. Jennings. Using gaussian processes to optimise concession in complex negotiations against unknown opponents. In *Proceedings of the 22nd International Joint Conference on Artificial Intelligence (IJCAI-2011)*, pages 432–438, 2011.

Agent-Based Framework for One-to-many Bi-lateral Negotiation in Online Trading

Tharushi Imalka, Harshani Nisansala, Sudharma Subasinghe, Sujathani Warnakulasuriya, Surangika Ranathunga, Akila Pemasiri
Department of Computer Science and Engineering
University of Moratuwa, Sri Lanka
{tharushi.11, nisansala.11, suba.11, sujathani.11, surangika, akila}@cse.mrt.ac.lk

Upali Kohomban
CodeGen International (Pvt) Ltd.
Sri Lanka
upali@codegen.co.uk

ABSTRACT
Existing Agent-Based Negotiation (ABN) frameworks for bargaining are based on the assumption that private information of agents such as reserve price and deadline is fixed. However, in real world bargaining scenarios, negotiators change their deadline due to their bargaining strategies, personal reasons or market issues. Therefore the assumption of fixed private information limits the usability and performance of the existing ABN approaches. This paper presents a novel global optimization model for offer generation in one-to-many bilateral negotiations that takes dynamism of the deadline into account. To support agents with incomplete information, the model incorporates Bayesian learning. In this approach the agents profess a deadline that is shorter than the actual, thus pressurizing the opponent to get a quick counter offer and finally selects the best opponent to deal with. The model is empirically evaluated and is shown to be effective and robust in a range of one-to-one and one-to-many single issue negotiation scenarios.

Categories and Subject Descriptors
I.2.11 [Distributed artificial intelligence]: Multi-agent systems

Keywords
Automated negotiation; bargaining; offer generation; private information; one-to-many

1. INTRODUCTION
In general terms, negotiation aims at reaching some allocation of resources that is acceptable to all parties [7]. With the advancement in the Internet technologies, various forms of negotiation in online trading have become common day practice for many people. Negotiation in online trading can be divided into three categories; (1) bidding (2) auction and (3) bargaining [14]. However, the inherent complexities of online negotiation make it a time-consuming process. Agent-Based trading aims to provide a solution for this. Agent-Based Negotiation (ABN), whereby the negotiation is automated by software programs, is one of the electronic negotiation approaches that have been identified and practiced [6][14].

When compared with bidding, the bargaining process contains additional steps such as the issuing of proposals and counter proposals between negotiation parties until a mutual agreement is reached [12][16]. Theoretically, this may lead to higher profits via higher revenues or sales quantities. While negotiating, both parties try to maximize their utilities. In real world negotiations, humans use various negotiation strategies such as contending, yielding, and compromising to achieve maximum utilities in a lesser time period [17]. Therefore incorporating real world bargaining strategies in agent-based online negotiations will lead to successful agreements for negotiators. Generally, negotiating parties do not reveal their preferences to other parties [4][18] (i.e. preferences are private). As a result, negotiators do not get complete information of the opponent and they are not able to know which agreements are more efficient [15]. However, in a negotiation scenario, since both parties aim to satisfy their own interests, opponents' preferences are needed to be known to reach a beneficial agreement [5]. Therefore current ABN frameworks use opponent modeling techniques to estimate the opponent's preferences during the negotiation [2][5].

Emerging ABN frameworks that support opponent modeling comprise of two major components [9][18][19]. The first component is the learning component that models the opponent's private information. Bayesian method is the most popular approach among other learning techniques [1][19]. Bayesian learning is used in many researches related to agent-based negotiation for opponent modeling where incomplete information is presented [1]. The second component generates offers during the bargaining process. An offer is generated considering the model of the opponent as estimated by the learning component, in hopes of maximizing self-gain. In order to provide a platform for online bargaining, a number of different agent-based negotiation models have been developed with different negotiation strategies [10][11][18]. Existing ABN frameworks for bargaining are based on the assumption that private information of agents such as reserve price and deadline is fixed [12][19]. However, in the real world complex negotiation scenarios, the reserve price and deadline are not always fixed. Moreover the probability of getting a deal in a negotiation in the existing frameworks can be optimized by adopting real world negotiation strategies into account. Another major drawback in existing ABN frameworks is that agents need to select one opponent among many opponents, to do the deal prior to the bargaining process [12], which is inefficient in a one-to-many bargaining scenario.

In this paper, we present a model that considers the dynamism of deadlines, which is more similar to real world bargaining scenarios. This scenario removes the constraint of fixed private information from existing frameworks while learning and pressurizing the opponent. This approach reduces the time taken for a deal while increasing the probability of getting a deal. Moreover, this strategy generates offers that obtains the best response from the opponent and subsequently achieves a better deal for the agent in a shorter period of time. It employs another novel strategy called acceptance strategy in which an agent selects the best opponent in a one-to-many negotiation scenario.

This work is an extension of the work by Yu et al. [19]. The work by Yu et al. [19] presents an ABN framework for bargaining that uses Bayesian learning to model the opponent under the assumption of fixed private information. As the first step of our ABN framework, the Bayesian learning component presented by Yu et al [19] for opponent modeling is incorporated to model one-to-one bargaining scenario. Their offer generation component (which assumes fixed deadline) is modified to support dynamism of deadlines. Our second step extends this one-to-one model to a one-to-many negotiation that globally optimizes the agent's utility.

Firstly, the negotiating agent estimates the opponent's reserve price and deadline using the opponent modeling component. The estimated opponent model is then used to derive a new deadline for the agent, which will be pretended for the opponent. The generated deadline varies from opponent to opponent. Then the derived deadlines are used in the offer generating component to generate counter offers at each round of negotiation. Seeing these offers with a pretended deadline, which is less than the actual deadline, opponent agents get pressurized and tend to issue counter offers quickly. Our framework selects the opponent with the best offer among all the offers using a novel global optimization strategy. Finally, our agent completes the deal with the selected best opponent. We demonstrate this implementation of dynamism and global optimization leads to quick agreements while pressurizing the opponents. Moreover, we show that our strategy archives maximum time utility values while increasing the probability of getting a successful deal.

The remainder of the paper is organized as follows. Section 2 discusses related work in the area of agent-based bargaining. In Section 3 the approach for offer generation is introduced. Section 4 presents experimental results to demonstrate the effectiveness of the approach. Finally, Section 5 concludes the paper.

2. RELATED WORK

Available ABN frameworks provide different capabilities for bargaining. However, each framework has its own strengths and weaknesses, and various limitations that facilitate various exploitations have been identified. Here we focus only on ABN frameworks that employ learning, as it is the current state of the art. As our model is based on price negotiation, we are considering bi-lateral (i.e. bargaining), single-issue (i.e. price) negotiations here.

Kraus [9] has proposed a negotiation model assuming that the agent's preferences are completely revealed to the opponent. However in real world negotiation practices, people do not know the opponent's preferences, such as reserve price and deadline. Also Kraus's model does not support one-to-many bargaining.

The Intelligent Trading Agency (ITA) proposed by Rahwan et al [15] has assumed that the agents have incomplete information about the preferences and constraints of each other, which is likely to happen in real world bargaining. However, agents in ITA use same negotiation procedure with all opponent agents (i.e. behavior of the agent is fixed throughout the negotiation process, regardless of the opponent it is dealing with) that makes the model inefficient. Moreover, this framework does not provide any techniques for agents to reuse their negotiation experiences or to improve the final outcomes. Nguyen et al [12] has proposed an improved one-to-many bi-lateral negotiation model where various negotiating threads can mutually influence one another to alter the behavior of an agent in another thread. Their bargaining model adopts different negotiation strategies according to the available information of different opponents. However, this model still assumes that private information of agents is fixed.

Several negotiation frameworks have used opponent modeling techniques to address the issue of limited information. Giwak and Sim [3] have introduced a learning based opponent modeling technique in their negotiation framework to deal with incomplete information. Their approach only considers the case that one agent is learning opponent's information using Bayesian learning and deadline information. Therefore, it will be interesting to analyze the case that both agents can learn from each other.

Yu et al [19] proposed an improved one-to-one bi-lateral negotiation model that uses learning to enable self-interested agents to adapt negotiation strategies dynamically according to the received offers. The learning agent has beliefs about the probability distribution of the opponent's negotiation parameters such as reserve price and deadline. Moreover this system learns from the opponents' offers and generates a more beneficial offer. However, its offer generation component assumes that agents have fixed private information. Also it does not provide any one-to-many negotiation strategy. Zafari et al [20] have introduced a bilateral, multi issue negotiation model called DOPPONENT that extracts opponent's preferences using distance-based algorithms. However, finding the proper distance function is difficult [20]. The model proposed by Kolomvatsos et al [8] incorporates a fuzzy logic system for opponent modeling and it does not support one-to-many bargaining. Also fuzzy systems require expert knowledge to define a fuzzy rule base [9]. The Greedy Concession Algorithm (GCA) proposed by Baarslag et al [1] to model sequence of bids for the opponent agent does not consider previous negotiation interactions.

All of the above negotiation frameworks assume that the private information of agents is fixed. However, in real world negotiation scenarios, negotiators change their reserve price and deadline according to their bargaining strategies, due to changes in market prices, opponent offers or personal issues. Therefore it is evident that the assumption of fixed private information in a negotiation model leads to wrong decisions. Also, adopting new strategies for offer generation and use of learning inside the acceptance strategy of a one-to-many scenario lead to significant improvements in the negotiation outcomes [1].

3. NEGOTIATION MODEL FOR BARGAINING

Our model resolves the current state-of-art limitations by considering dynamism of private information and bidding history to generate offers. In particular, the main contribution of this paper is to enhance the offer generation strategy by introducing dynamic deadlines to aggressively pressurize the opponent to obtain the quickest response for a given one-to-many negotiation scenario.

As shown in Figure 1, there are two major components in the one-to-one model: (1) Opponent modeling and (2) Offer generation. In the one-to-many model, the acceptance strategy component is introduced. Opponent modeling component is based on the model proposed by Yu et al [19]. Our model focuses on the offer generating component and acceptance strategy. Here the agents achieve optimal time utilities by; (1) generating better offers while pretending shorter deadlines for the opponent and (2) selecting the best opponent using the acceptance strategy. In our negotiation setting, the agents alternately exchange bids.

3.1 Opponent Modeling

Agents estimate the opponent's reserve price and deadline using this component. This component consists of two parts, a Regression analysis component and a Bayesian learning component. In regression analysis, (1) an agent chooses a set of random reservation points first, based on the belief that this point is the reservation point of the opponent; (2) then the agent conducts regression analysis for all random reservation points; (3) finally the agent compares the fitted offers on each regression line with opponent's historical offers using non-linear correlation.

Subsequently, it chooses the point with the biggest non-linear correlation as the real reservation point. We obtain the estimated reservation point and the deadline of the opponent using this point. Then by using Bayesian learning, the agent's belief on the probability distribution of the real reservation point is dynamically updated at every step of the negotiation.

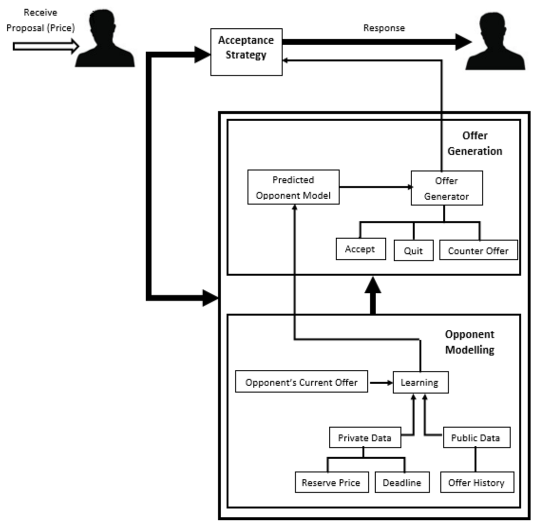


Figure 1. Model architecture diagram

3.2 Offer Generation

As mentioned in section 1, in a real world negotiation scenario, buyers and sellers have dynamic deadlines. Therefore the proposed strategy considers this dynamism of deadlines inside the framework. The bargaining agent in our offer generation component works as follows.

From the buyer's perspective, buyer's expectation is to purchase at a minimum price within his actual deadline negotiating with a seller.

According to our first step of one-to-one model, buyer gives an offer to the opponent pretending that his deadline is shorter. This situation pressurizes the opponent who then tends to issue offers quicker. This increases the buyer's probability of getting a deal. On the other hand, the risk of not having a deal is decreased. However, with the ambition of increasing the chance of making a deal, this model is designed to dynamically change the deadline (pretending closer deadline) of the buyer to aggressively pressure the seller for making a deal fast. If the seller is issuing higher offers, buyer can go to next seller (i.e. one-to-many scenario) pretending a different deadline.

Let b and s represent negotiators, i.e., b for a buyer agent and s for a seller agent.

On the buyer's perspective,

t - Current time

(T_b)act - Actual deadline of the buyer

(T_b)pret - Pretending deadline of the buyer

(RP_b)act - Actual reserve price of the buyer

U_b - Urgency of the buyer to deal; where $U_b = t \,/\, (T_b)$act

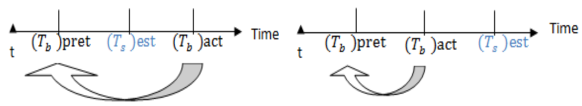
(T_s)est - Estimated deadline of the seller

(RP_s)est - Estimated reserve price of the seller

(R_s)est – Buyer's risk of not getting a deal

To pressurize the seller aggressively, buyer pretends the actual deadline as shown in Figure 2.

According to Figure 2, the pretended deadline of an agent should always lie in between the current time (t) and the opponent's deadline (i.e. (T_s)est). If the agent's actual deadline (i.e. (T_b)act) is already less than the opponent's deadline, then the pretended deadline should lie in between the current time(t) and the agent's actual deadline((T_b)act) as shown in Figure 2:b.



(a)(T_b)act> (T_s)est(b) (T_b)act <(T_s)est

Figure 2. The way of deciding the range of pretended deadline

Buyer's pretended deadline depends on the seller's selling behavior. Therefore to derive a value for the pretended deadline, the buyer needs to establish an estimate of the seller's likelihood of selling an item. If this likelihood of the seller is low then there is a risk of losing the deal for the buyer. As the first step of this opponent modeling, risk value is calculated for the buyer. There, the basic assumption is that the risk is a parameter that is affected only by the deadlines of the buyer and seller, and the urgency of the buyer. Therefore in a particular negotiation round, deadline of the buyer and the estimated deadline of the seller are known as fixed factors. Risk is calculated as a value in between -1 and 1 (i.e. (Rs)est $\in [-1, 1]$).

According to Figure 3,

•If the risk is a negative value, then the risk is low, which is advantageous to the buyer.

•If the risk is a positive value, then the risk is high, which is disadvantageous to the buyer.

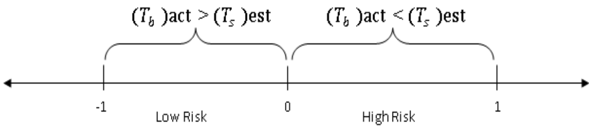


Figure 3. Risk Analysis

When the buyer's deadline is long, there is a chance that the seller may go for another buyer to sell the item that increases the risk value of the buyer. However, if the buyer's deadline is shorter, seller gets pressurized and tends to deal with the same buyer, which reduces the risk of not making a deal. Therefore the risk of dealing with the seller is calculated according to the Equation (1).

$$(R_s)\text{est} = \frac{(T_s)\text{est} - (T_b)\text{act}}{0.5((T_s)\text{est} + (T_b)\text{act}) - \text{t}} \times U_b (1)$$

By substituting the value of U_b to the Equation (1), Equation (2) is obtained,

$$(R_s)\text{est} = \frac{(T_s)\text{est} - (T_b)\text{act}}{0.5((T_s)\text{est} + (T_b)\text{act}) - \text{t}} \times \frac{t}{(T_b)\text{act}} (2)$$

By changing the value of (R_s) in Equation (2), we can generate the predicting deadline for the buyer ((T_b)pret) as shown in Equations (3)-(6).

$$R_s = \frac{(T_s)\text{est} - (T_b)\text{pret}}{0.5((T_s)\text{est} + (T_b)\text{pret}) - \text{t}} \times \frac{t}{(T_b)\text{pret}} (3)$$

$$[0.5(T_s)\text{est} + 0.5(T_b)\text{pret} - \text{t}](T_b)\text{pret}\,(R_s) = [(T_s)\text{est} - (T_b)\text{pret}] \times \text{t} \quad (4)$$

$$[0.5(T_s)\text{est} - (R_s)\text{t}](T_b)\text{pret} + 0.5[(T_b)\text{pret}]^2(R_s) = (T_s)\text{est} \times \text{t} - (T_b)\text{pret} \times \text{t} \quad (5)$$

$$0.5(R_s)[(T_b)\text{pret}]^2 + [(0.5(R_s)(T_s)\text{est}) + (1 - R_s)\text{t}](T_b)\text{pret} - [(T_s)\text{est} \times \text{t}] = 0 \quad (6)$$

This is a quadratic function of (T_b)pret. By solving Equation (6) we can obtain the value of (T_b)prêt as shown in Equation (7).

However, the (T_b)pret> 0. Therefore the model takes the positive values for the (T_b)pret. Then the agent executes the offer generation component using the predicted information (i.e. (T_b)pret).

$$(T_b)\text{pret} = \frac{-[0.5(R_s)(T_s)\text{est} + (1 - R_s)\text{t}]}{R_s} \pm \frac{\sqrt{[0.5(R_s)(T_s)\text{est} + (1 - R_s)\text{t}]^2 + 2(\text{Rs})[(T_s)\text{est} \times \text{t}]}}{R_s} \quad (7)$$

The generated offer in each round is issued to the opponent (i.e. seller) and counter offer of the opponent is obtained. This process is continued until the negotiation is finished.

By using the value (T_b)pret in Equation (7), we can generate the counter offer using (T_b)pret on behalf of the actual deadline of the buyer according to the Equation (8) presented by Yu et al [19].

$$Offer_b(\text{t}) = P_0 + (RP_b - P_0) \times \left(\frac{t - t_0}{(\boldsymbol{T_b})\textbf{pret} - t_0}\right)^{\beta} (8)$$

3.3 Acceptance strategy of One-to-many Scenario

As the next step, our model extends this one-to-one bargaining scenario into a one-to-many bargaining model, which globally optimizes the negotiation gain of the agent.

If we consider the buyer's perspective, where one buyer and multiple sellers are negotiating over a product, the buyer always tries to have a deal within his actual deadline by negotiating with one or many sellers using the pressurizing technique. Moreover, the buyer knows that he has a chance to negotiate with another seller after losing a deal with a current negotiating seller. In each round of negotiation, buyer follows three steps (i.e. global optimization strategy) to select the best seller to complete the deal.

Step 1: Priority to issue counter offer is given to the seller with lowest price offer

Here the buyer has the ability to interact with many sellers to buy a product and therefore he should make a decision on which seller to select to issue the counter offer first. Therefore, the buyer issues counter offers (according to the one-to-one model) for sellers in the increasing order of the offer values given by sellers.

Step 2: Determine the best seller opponent for making a successful deal at the deadline

As the pretended deadline of the buyer is always less than the seller's estimated deadline, we estimate each seller's possible offers at corresponding pretended deadlines of the buyer. The seller who has the lowest estimated offer at pretended deadline is selected as the best seller in each round.

Let's consider a scenario where there is one buyer and 3 sellers negotiating as shown in the Figure 4.

In the current round (i.e. round 4), the three sellers have given the offers in the order of OS1 < OS2 < OS3. Therefore the buyer issues the counter offers in the increasing order of offers as in step 1. Then the buyer infers the minimum offers of seller_1, seller_2 and seller_3 at buyer's pretended deadline using regression analysis described in section 3.1, as MOS3, MOS1,and MOS2 (where MOS3<MOS1,<MOS2) respectively. Therefore according to the fourth round, the buyer has determined that the seller_3 as the best seller.

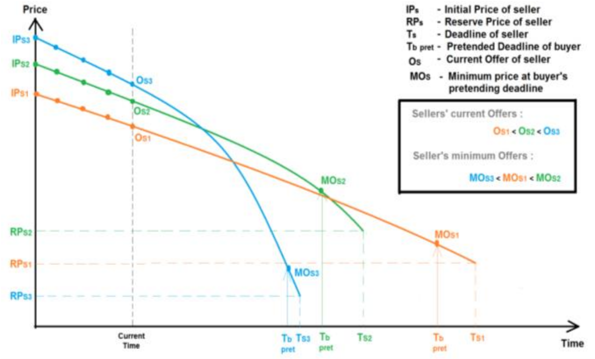


Figure 4. Buyer selects the best seller

Step 3: Make a successful deal with the best seller at the deadline of the negotiation process.

The buyer's pretended deadline for seller_3 is the shortest pretended deadline and it will be reached first. Therefore there is no point of doing further negotiation with other sellers as their minimum offer is higher than that of the seller_3. Finally, the buyer tends to do the successful deal with seller_3 and ends all other negotiation processes. If there is another seller (i.e. seller_4) who has the shortest buyer's pretended deadline, and a minimum offer that is bit higher than the minimum offer of the best seller (i.e. seller_3), then the buyer's pretended deadline for the seller_4 will be reached before that of the best seller (i.e. seller_3). As the minimum offer of the seller_4 is higher than the best seller's minimum offer at the pretended deadline, negotiation with seller_4 is terminated at the buyer's pretended deadline for that seller_4. Buyer proceeds with the other sellers till the pretended deadline for the best seller is reached, as described in step 2.

Accordingly, our model selects the best seller among multiple sellers using the acceptance strategy, which globally optimizes the negotiation outcome. This leads to output optimal gains for the buyer in a given negotiation setting.

Moreover, this model can be used in one seller and multiple buyers negotiation scenarios as well. First, seller prioritizes buyers in the decreasing order of offer values. At the next step, seller infers buyers' maximum offers at seller's pretended deadlines at each round of negotiation and determines the best buyer. Finally, seller completes the deal with the best buyer.

4. EVALUATION

4.1 One-to-one model

The model was evaluated along two directions; (1) checking the probability of getting a successful deal and (2) measuring the time taken for a successful deal. In both directions, the results were taken for the concession strategy proposed by Yu et al. [19] and our pressurizing technique in which the dynamism of the deadline is introduced to the model. In the experiment, a buyer and a seller negotiate over the price range in-between 500 - 800. In order to simplify the comparison process, we set the buyer agent's initial price to 500 and the seller agent's initial price to 800. The buyer's reserve price was randomly selected in-between 650 - 800 and seller's reserve price was randomly selected in-between 500 - 700. Such a setting ensures that the agreement zone between the two agents always exists. The negotiation parameter initialization is shown in Table 1.

We ran 100 episodes to show the generality and robustness of the latest concession strategy and pressurizing techniques. Moreover, we verified the correctness of our model by choosing different values for initial price of buyer and seller.

Table 1. Negotiation parameter initialization

Agent	Initial Price	Reserve Price	Deadline (ms)
Buyer	500	[650,800]	200,000
Seller	800	[500,700]	172,500

1. Probability of getting a successful deal

Figure 5 shows the number of test cases out of 100 test cases used for the evaluation of the new model, which succeeded in reaching a goal using the concession strategy and the pressurizing technique. It is shown that the concession strategy succeeded in only 85 test cases out of 100 test cases where as the pressurizing technique succeeded in 97 test cases out of 100 test cases and reached deals.

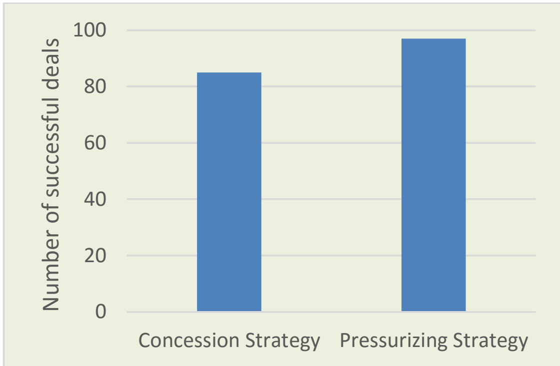


Figure 5. Number of successful deals using the concession strategy vs. pressurizing technique

Therefore the probability of getting a successful deal by using our pressurizing strategy is 97% and that of the concession strategy mentioned in [19] is 85% under this particular price range. Accordingly, using a T-test, we can say with 99.8% confidence that our strategy outperforms the concession strategy proposed by Yu et al [19]. The results were taken with specific values for negotiation settings for both buyer and seller agents as shown in Table 2.

Table 2. Initial offers and private information of buyer and seller

Agent	Initial Price	Reserve Price	Deadline(ms)
Buyer	500	710	200,000
Seller	800	690	172,500

When we evaluated the model for other different price ranges, almost similar results were generated, which verified the robustness of our model.

Figure 6 shows a scenario where the concession strategy has failed to reach a deal for given initial buyer and seller information in Table 2. Buyer and seller agents have not reached a deal even at the deadline. Figure 7 shows that the pressurizing technique has successfully reached a deal for the same initial buyer and seller information in Table 2. According to Figure 7, buyer and seller agents have reached a deal before the deadline. Therefore it is clearly shown that the introduced pressurizing technique has increased the probability of getting a successful deal that finishes the deal successfully.

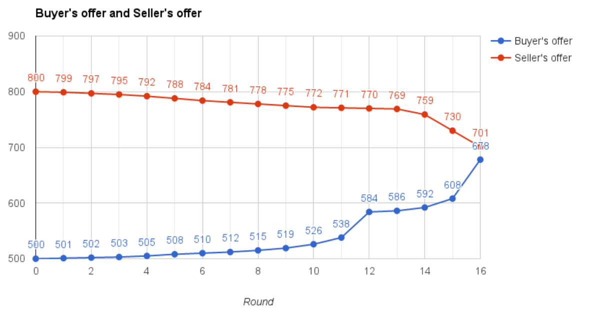


Figure 6. Negotiation results obtained using the concession strategy used by Yu et al. [19].

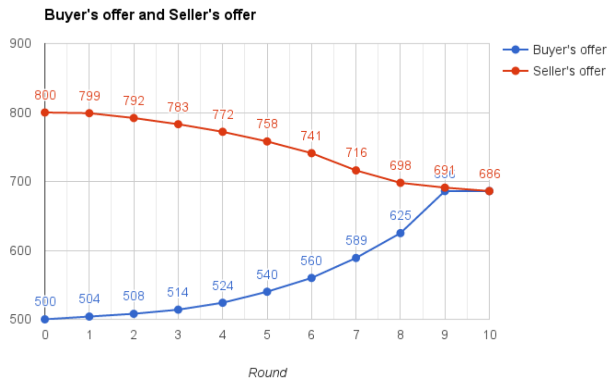


Figure7. Negotiation results obtained using the pressurizing technique

2. Time taken for a successful deal

The advantaged time percentage (atp) within the pressurizing technique is calculated for each test case in the experiment. It is calculated using Equation (9).

$$\text{atp} = \frac{\text{Number of rounds for pressurizing technique x 100\%}}{\text{Number of rounds for latest concession strategy}} \quad (9)$$

The average advantaged time percentage of the pressurizing technique is calculated for all 100 test cases in the experiment. The obtained value for the evaluated test cases was 79.56%. Therefore it is clearly shown that the introduced pressurizing technique has reduced the time taken for a successful deal by 20.44% in average. The results were taken with specific values for the negotiation setting for both buyer and seller agents as in Table1. According to Figure 8, concession strategy has taken 24 rounds to reach a deal for a given buyer and seller for the information in Table 2. Figure 9 shows that the pressurizing technique has taken only 15 rounds to reach a deal for the given buyer and seller information in Table 3. In both these scenarios, buyer and seller have reached a deal, however, within different negotiation times. Therefore it can be seen that the time taken to reach a deal in the pressurizing technique scenario is less than that of the concession strategy by Yu et al [19].

Table 3. Initial offers and private information of buyer and seller

Agent	Initial Price	Reserve Price	Deadline (ms)
Buyer	500	741	200,000
Seller	800	561	172,500

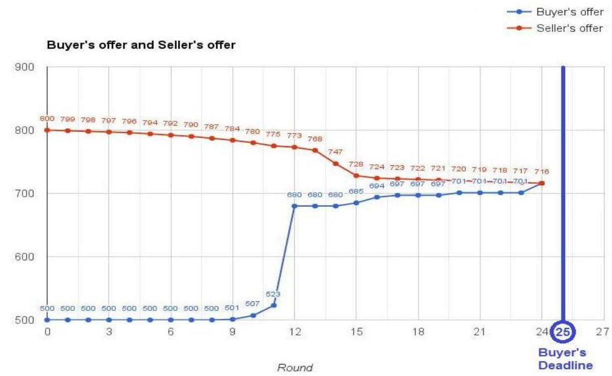


Figure 8. Negotiation results obtained using the concession strategy by Yu et al [19]

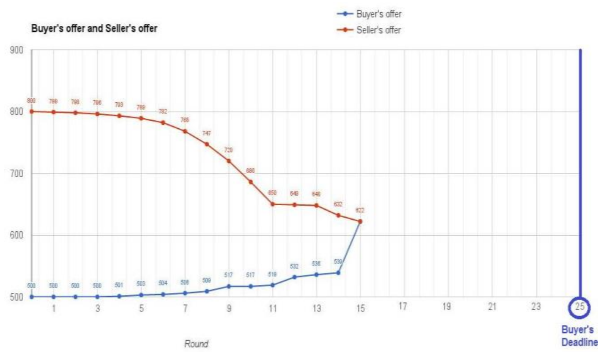


Figure 9. Negotiation results obtained using the pressurizing technique

4.2 One-to-many model

Here we evaluate one-to-many negotiation with aggressively pressurization technique and globally optimizing acceptance strategy. The model evaluations were carried out from both buyer's and seller's perspectives: (1) One buyer many sellers negotiation scenario and (2) One seller many buyers negotiation scenario.

The selected negotiation price range for evaluations is 500 - 800. In order to simplify the one-to-many negotiation processes, one agent with two opponents is selected and in the comparison process, we set the seller agent's initial price to 800 and the buyer agents' initial prices to 500. The buyers' reserve prices were randomly selected in-between 650 - 800 and sellers' reserve prices were randomly selected in-between 500 – 650 for ensuring the agreement zone between the agents exists.

4.2.1 One buyer many sellers negotiation

The negotiation parameter initialization is shown in Table 4. We ran the system 100 episodes for both one-to-one scenario (i.e. selecting one seller at a time) and one-to-many scenario. Figure 10 shows the results of a one-to-one scenario for each seller (for randomly selected values). Buyer makes a successful deal with seller 1 at round 15 with the execution price 638, and makes a successful deal at round 14 and the execution price 642, with seller 2.

Table 4. Negotiation parameter initialization of one buyer and two sellers negotiation scenario

Agent	Initial Price	Reserve Price	Deadline (ms)
Buyer	500	[650,800]= (i.e.704)	200,000
Seller1	800	[500,650]= (i.e. 587)	172,500
Seller2	800	[500,650]= (i.e. 619)	172,500

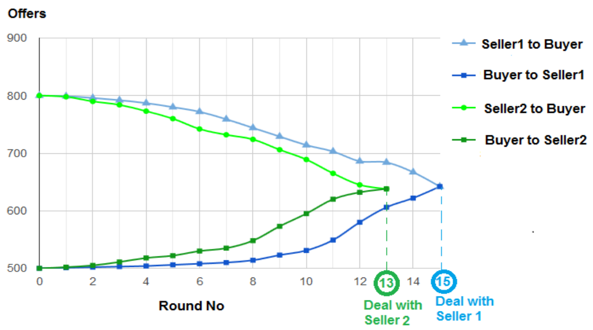


Figure 10. Negotiation results for one-to-one setting

As can be seen in Figure 11, buyer negotiates with both seller 1 and seller 2 and finally completes the deal with seller 1 who gave the lowest execution price, which is 638. Therefore in one-to-many negotiation, buyer acts same as in one-to-one negotiation, while completing the final deal with the most advantageous negotiator among many sellers.

Therefore the one-to-many negotiation system with the acceptance strategy provides the facility to globally optimize the negotiation outcomes for the buyer.

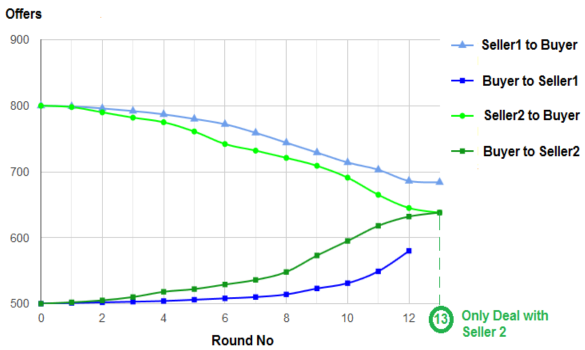


Figure 11. Negotiation results obtained for one buyer two sellers negotiation scenario

4.2.2 One seller many buyers negotiation

The negotiation parameter initialization is shown in Table 5. We execute the system as same as one buyer and multiple sellers scenario.

Table 5. Negotiation parameter initialization of one seller and two buyers negotiation scenario

Agent	Initial Price	Reserve Price	Deadline (ms)
Seller	800	[500,650]= (i.e. 587)	172,500
Buyer1	500	[650,800]= (i.e.685)	200,000
Buyer2	500	[650,800]= (i.e.679)	200,000

Figure 12 shows the results obtained for one-to-one scenario where one seller negotiates with multiple buyers separately.

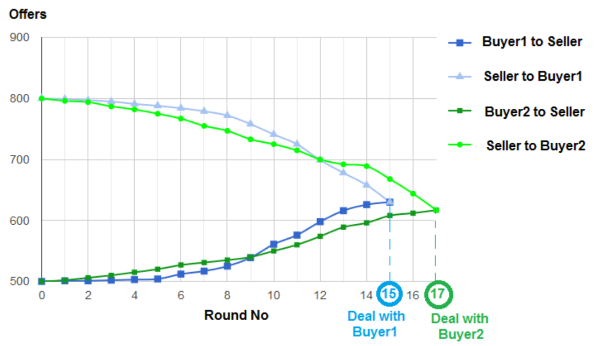


Figure 12. Negotiation results for one-to-one setting

According to Figure 12, seller's deal with buyer 1 is 630 at round 15 and buyer 2 is 617 at round 17.

In our one-to-many model, the seller decides to sell the product to the higher price value (i.e. 630) and accept the buyer 1 and terminate the buyer 2's negotiation process (see Figure 13). Therefore this justifies that our system supports the global optimization from the seller side.

We ran the system for different price ranges and verified the robustness of our system.

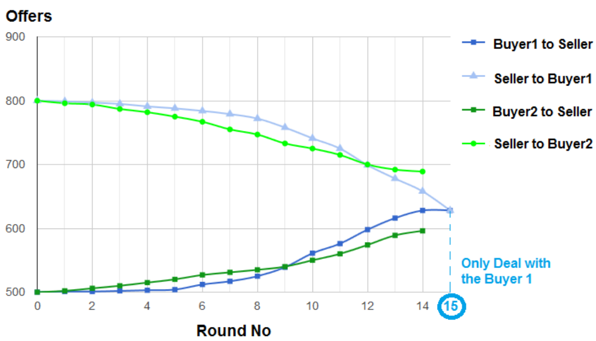


Figure 13.The negotiation results obtained in one seller two buyers negotiation scenario

5. CONCLUSIONS AND FUTURE WORK

Existing agent-based bargaining models assume that private information of agents is fixed, which is not true in most of the real world scenarios. This paper introduced a model for automated single-issue (price), one-to-many bargaining in a market place where private information of agents can be dynamic during the bargaining process. Offer generation is an important part in an agent-based bargaining model as outcomes can be optimized by using a better offer generating strategy. This paper presented a novel strategy to generate the next offer of an agent by introducing dynamism of deadlines (i.e. shorter deadlines) of the negotiator. This leads to quicker and better agreements by pressurizing the opponent agent. The pressurizing technique increases the possibility of reaching a deal and through that reduces the probability of failure of a negotiation. Also the acceptance strategy proposed in the paper, selects the best opponent to deal with while learning. As future work this model is expected to be extended to many-to-many negotiation scenarios where many agents negotiate with many different agents to get a successful deal. Also it is planned to model the agent when reserve price is dynamic. Finally, modeling the scenario for multi-issue negotiation will make the model closer to real world applications.

6. REFERENCES

[1] Baarslag, T., Gerding, E.H. and Aydo, R. Optimal Negotiation Decision Functions in Time-Sensitive Domains.

[2] Chen, S., Ammar, H.B., Tuyls, K. and Weiss, G. Optimizing Complex Automated Negotiation using Sparse Pseudo-input Gaussian processes. in *International conference on Autonomous agents and multi-agent systems*, (2013), International Foundation for Autonomous Agents and Multi-agent Systems, 6–10.

[3] Gwak, J. and Sim, K.M. Bayesian learning based negotiation agents for supporting negotiation with incomplete information. in *International MultiConference of Engineers and Computer Scientists*, (2011) Newswood Limited, 163-168.

[4] Hindriks, K., Jonker, C.M. and Tykhonov, D. The benefits of opponent models in negotiation. in *IEEE/WIC/ACM International Conference on Intelligent Agent Technology*, (Milano, Italy, 2009), IET, 439–444. DOI= http://dl.acm.org/citation.cfm?id=1632556

[5] Hindriks, K. and Tykhonov, D. Opponent Modelling in Automated Multi-Issue Negotiation Using Bayesian Learning. in *7th international joint conference on Autonomous agents and multi-agent systems*, (2008), International Foundation for Autonomous Agents and Multi-agentSystems,307–308. DOI= http://dl.acm.org/citation.cfm?id=1632556

[6] Huang, P.H.P. and Sycara, K. A computational model for online agent negotiation. in *35th Annual Hawaii International Conference on System Sciences*, (2002), IEEE, 438-444.

[7] Iyad Rahwan. 2005. Interest-based Negotiation in Multi-Agent Systems. Master's thesis. Department of Information Systems, University of Melbourne, Australia.

[8] Kolomvatsos, K., Trivizakis, D. and Hadjiefthymiades, S. An adaptive fuzzy logic system for automated negotiations. *Fuzzy Sets and Systems*, 269 (2015), 135–152.

[9] Kraus, S. *Strategic negotiation in multi agent environments*. MIT press, Cambridge, United States of America, 2015.

[10] Lin, F. and Chang, K. A Multiagent Framework for Automated Online Bargaining, *IEEE Intelligent Systems*, 4 (2001), 41–47.

[11] Nguyen, T.D. and Jennings, N.R. A heuristic model for concurrent bi-lateral negotiations in incomplete information settings. in *International Joint Conference on Artificial Intelligence*, (2003), 1467–1469.

[12] Nguyen, T.D.N.T.D. and Jennings, N.R. Concurrent bilateral negotiation in agent systems. in *14th International Workshop on Database and Expert Systems Applications*, (2003), IEEE Intelligent Systems, 3–8.

[13] Nguyen T.D. and Jennings N.R. Coordinating multiple concurrent negotiations. in *3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, (2004), IEEE Computer Society, 1062–1069. DOI= http://dl.acm.org/citation.cfm?id=1018875

[14] Online Trading and Negotiation. May (2000). Retrieved May 15, 2015, from Computer Information System Engineering, San Jose State University: *http://www.engr.sjsu.edu/gaojerry*

[15] Rahwan, I., Kowalczyk, R. and Pham, H.H. Intelligent agents for automated one-to-many e-commerce negotiation. *Australian Computer Science Communication*, 24(1), 197–204. DOI= http://dl.acm.org/citation.cfm?id=563824

[16] Ramchurn, S.D., Jennings, N. R. and Sierra, C. Persuasive negotiation for autonomous agents: A rhetorical approach. *in IJCAI Workshop on Computational Models of Natural Argument*, (2003),AAAI Press, 9-17.

[17] Strategies for negotiating, 2014. Retrieved July 11, 2015, from Queensland Government: https://www.business.qld.gov.au/business/running/managing-business-relationships/negotiating-successfully/negotiating-strategies.

[18] Williams, C.R. 2012. Practical Strategies for Agent-Based Negotiation in Complex Environments. Ph. D. Dissertation. Southampton University.

[19] Yu, C., Ren, F. and Zhang, M. An adaptive bilateral negotiation model based on Bayesian learning. *Complex Automated Negotiations: Theories, Models, and Software Competitions*, (2013), 75–93.

[20] Zafari, F., Nassiri-mofakham, F. and Hamadani, A.Z. DOPPONENT : A Socially Efficient Preference Model of Opponentin Bilateral Multi Issue Negotiations. *Journal of Computing and Security*, 1 (4), 283-292.