Logic-based automated multi-issue bilateral negotiation in Peer-to-Peer E-marketplaces

Azzurra Ragone[1], Tommaso Di Noia[1], Eugenio Di Sciascio[1], Francesco M. Donini[2]

[1] SisInfLab, Politecnico di Bari, Bari, Italy
{a.ragone,t.dinoia,disciascio}@poliba.it
[2] Università della Tuscia and SisInfLab, Viterbo, Italy
donini@unitus.it

Abstract. We present a novel logic-based framework to automate multi-issue bilateral negotiation in e-commerce settings. The approach exploits logic as communication language among agents, and optimization techniques in order to find Pareto-efficient agreements. We introduce $\mathcal{P}(\mathcal{N})$, a propositional logic extended with concrete domains, which allows one to model relations among issues (both numerical and non-numerical ones) via logical entailment, differently from well-known approaches that describe issues as uncorrelated. Through $\mathcal{P}(\mathcal{N})$ it is possible to represent buyer's request, seller's supply and their respective preferences as formulas endowed with a formal semantics, *e.g.*, "*if I spend more than 30000 € for a sedan then I want more than a two-years warranty and a GPS system included*". We mix logic and utility theory in order to express preferences in a qualitative and quantitative way.
We illustrate the theoretical framework, the logical language, the one-shot negotiation protocol we adopt, and show we are able to compute Pareto-efficient outcomes, using a mediator to solve an optimization problem. We prove the computational adequacy of our method by studying the complexity of the problem of finding Pareto-efficient solutions in our setting.

1 Introduction

Negotiation crosses the boundaries of several disciplines, including economics, computer science, decision support theory, organizational theory. A number of definitions have been proposed in the literature referring to negotiation, none of them completely exhaustive or covering the boundless negotiation arena in the various domains. We refer here to the one by Parsons et al. [29], that defines negotiation as "the process by which a group of agents communicate with one other to try and come to a mutually acceptable agreement on some matter." Bilateral negotiation finds applications in different scenarios, each one with its own peculiarities and issues. Among others, *resource and task allocation* problems (*e.g.*, scheduling, logistics, bandwith usage), *online auctions* (eBay probably being the most renowned example), *e-marketplaces*. In this paper we obviously do not claim to propose a negotiation mechanism suitable for all the above cited scenarios, as each scenario has its own key dimensions, more or less crucial, depending on the scenario itself: a mechanism having some good characteristics for some domain, can be, indeed, completely unsuitable for another one. We focus on mechanisms to automate negotiation in peer-to-peer (P2P) e-marketplaces [38], where products (cars, houses, Personal Computers, etc.) or services (travel booking, wedding service, etc.) can be, at the same time, provided by suppliers or searched by potential customers. In the rest of the paper we refer, by way of example, without loss of generality, to an *automotive e-marketplace*.

In the framework we study, the negotiation problem can be characterized by the following key dimensions:

– **Multiple issues.** Differently from e-marketplaces dealing with undifferentiated products (commodities as oil, concrete, etc.) or stocks, where only price, time or quantity have to be taken into account, in a P2P e-marketplace also other features have to be considered during the negotiation process. As a matter of fact, when a potential buyer browses an e-marketplace, she may look for a good (*e.g.*, a car) fulfilling her needs and/or wishes, so that not only the price is relevant, but also other features, such as *e.g.*, warranty or delivery time, as well as look, model, comfort, and so on. Usually, issues are described as uncorrelated terms, without considering any underlying semantics. Moreover, issues may not be established in advance (before the negotiation starts) as it is usually assumed in many problems of resource and task allocation (see Section 8 for an extensive discussion). In our approach, we model a *multi-issue bilateral negotiation* problem where issues can be in some way interrelated (*e.g.*, *"I would like a station wagon endowed with shadow rear windows if its color is black"*).

– **Language.** Agents need to communicate with each other in order to express not only what they are searching for/offering, but also to express preferences on issues or bundles thereof. A language has to be defined to express product/service descriptions, as well as preferences on issues characterizing the product/service itself, *e.g.*, the buyer can state: (1) *"I can spend up to 25000 € for a passenger car only if there is a navigator pack included"*; while the seller can provide (2) a *"Sedan with a price not less than 23000 with a GPS system included"*. How can we determine if there is a negotiation space between them? In order to handle preferences involving numerical features and non-numerical ones, we define (Section 2), the logic $\mathcal{P}(\mathcal{N})$ as communication language, a Propositional Logic endowed with Concrete Domains . So the above preferences can be expressed as (1) $\mathtt{PassengerCar} \wedge (\mathtt{price} \leq 25000) \Rightarrow \mathtt{NavigatorPack}$ and (2) $\mathtt{Sedan} \wedge (\mathtt{price} \geq 23000) \wedge \mathtt{GPS\_system}$, where relations among issues are made explicit in a **logical Theory** (*i.e.*, an **ontology**), where we can express that a *Sedan is a type of Passenger car* ($\mathtt{Sedan} \Rightarrow \mathtt{PassengerCar}$) or that a *Suv is not a Sedan* ($\mathtt{SUV} \Rightarrow \neg\mathtt{Sedan}$) or still, find the meaning of *Navigator pack* ($\mathtt{NavigatorPack} \Leftrightarrow \mathtt{SatelliteAlarm} \wedge \mathtt{GPS\_system}$).

– **Utilities.** Once agents have expressed their own preferences, it is important to compare them, weighting each agent's preferences. This can be done ranking preferences in an ordinal or cardinal way. Furthermore we are interested not only in *local preferences* (over logical formulas), but also in *global preferences* (over entire agreements). Hence, we compare and evaluate different kinds of agreement to find the most suitable agreements for both agents. To this end we define a utility function taking into account how much preferences are satisfied in the final agreement, their relative relevance and how much (for numerical features) each preference is satisfied, see Sections 3 and 4.

– **Information.** Another important dimension of the negotiation problem is: *what I know* about my opponent. An agent can know everything (*preferences* and *worth* of them), as well as only preferences (but not their worth) or nothing about its opponent. The less he knows the more it is difficult to model a negotiation mechanism (Kraus [20] made an extensive discussion on this point). We adopt a *centralized approach*[3] where agents reveal their preferences and worth of them only to a trusted mediator. As pointed out by Raiffa et al. [35, p.311], usually, bargainers may not want to disclose their preferences or utilities to the other party, but they can be more willing to reveal this information to a trusted—automated—mediator, helping negotiating parties to achieve efficient and equitable outcomes.
– **Agreement.** Usually, the main target of an agent is to reach a satisfiable agreement in a reasonable amount of communication rounds. Furthermore, knowing if such an agreement is also *Pareto-efficient* is a matter that can not be left out. A *protocol* and *strategies* have to be defined in order to ensure that the reached agreement is Pareto-efficient. Besides, it is fundamental to assess *how hard* it is to find Pareto-efficient agreements and check whether a given agreement is Pareto-efficient. In our framework, we propose a *one-shot* protocol with the intervention of a *mediator* with a proactive behavior: it collects agent's preferences and proposes to each participant a *fair* Pareto-efficient agreement (see Section 5.2). For what concerns strategy, the players reveal their preferences to the mediator and then, once it has computed a solution, they can accept or refuse the agreement proposed to them; they refuse if they think possible to reach a better agreement looking for another partner, or for a different set of bidding rules. Notice that here we do not consider the influence of the *outside options* in the negotiation strategy [25].
– **Mediator.** As hinted above, we adopt a centralized approach with a mediator acting as a nonbinding arbitrator [35]. Although we admit approaches to negotiation involving a mediation entity are often controversial (Section 8), we stress here its importance in our p2p e-marketplace framework. Its roles are as follows: it collects and stores advertisements (requests and offers), it allows agents to keep preferences and their worth as private information, it solves the optimization problem (Section 5), which allows to determine fair Pareto-efficient agreements, and finally notifies the bidders, paving the way to the actual transaction. Furthermore, the actual deployment of our approach in a real-world system can be made economically feasible if the mediator charges a fee for each successful transaction, *i.e.*, for each proposed agreement accepted by both parties.

Observe that in our framework it is possible to model *positive* and *negative* preferences (*"I would like a car either black or gray, but not red"*), as well as *conditional preferences* (*"I would like leather seats if the car is black"*) involving both numerical features and non-numerical ones (*"If you want a car with GPS system you have to wait at least one month"*) or only numerical ones (*"I accept to pay more than 25000€ only if there is more than a two-years warranty"*).

[3] We refer to Section 8 for a thorough discussion on centralized/distributed approaches.

Besides we model *quantitative* preferences; thanks to the weight assigned to each preference it is possible to determine a relative importance among them, rather than only a total order between them[4].

Summarizing, the main contributions of this paper include: a framework to automate multi-issue bilateral negotiation; the logical language $\mathcal{P}(\mathcal{N})$, able to handle both numerical features and non-numerical ones, as well as to represent existing relations between issues and preferences as formulas. The rationale for this proposal is somehow obvious: using an ontology $\mathcal{T}$ it is possible to catch an inconsistency among preferences (agents cannot agree on A and B at the same time if in $\mathcal{T}$ A is defined as disjoint from B). On the other hand, the use of a formal ontology allows one to discover that an agent preference is satisfied (implied) by a preference of its opponent, even if this is not explicitly modeled at the syntactical level. For instance, if I am searching for a Passenger car and my opponent offers a Sedan, my preference is surely satisfied by the offer because of the axioms in the ontology ($\texttt{Sedan} \Rightarrow \texttt{PassengerCar}$). Knowledge representation techniques are hence mixed up with utility theory in order to express preferences both in a qualitative and quantitative way. We then propose a one-shot protocol with the intervention of a mediator able to compute Pareto-efficient agreements solving an optimization problem.

We also prove the computational adequacy of our method by studying the complexity of the problem of finding Pareto-efficient solutions.

2 Representation of issues

We divide issues involved in a negotiation in two categories. Some issues may express properties that are true or false, like, *e.g.*, in an automotive domain, `ItalianMaker`, or `AlarmSystem`. We represent them as propositional atoms $A_1, A_2, \ldots$ from a finite set $\mathcal{A}$. Other issues involve numerical features like `deliverytime`, or `price` represented as variables $f_1, f_2, \ldots$, each one with its specific domain $D_{f_1}, D_{f_2}, \ldots$, such as $[0, 90]$ (days) for `deliverytime`, or $[1000, 30000]$ (euros), for `price`. The variables representing numerical features are always constrained by comparing them to some constant, like $\texttt{price} < 20000$, or $\texttt{deliverytime} \geq 30$, and such constraints can be combined into complex propositional requirements—also involving propositional issues—*e.g.*, $\texttt{ItalianMaker} \wedge (\texttt{price} \leq 25000) \wedge (\texttt{deliverytime} < 30)$ (representing a car made in Italy, costing no more than 25000 euros, delivered in less than 30 days), or $\texttt{AlarmSystem} \Rightarrow (\texttt{deliverytime} > 30)$ (expressing the seller's requirement*"if you want an alarm system mounted you'll have to wait more than one month"*). We now give precise definitions for the above intuitions, borrowing from a previous formalization of so-called *concrete domains* [2] from Knowledge Representation languages.

Definition 1 (Concrete Domains, [2]). *A* concrete domain D *consists of a finite set* $\Delta_c(D)$ *of numerical values, and a set of predicates* $C(D)$ *expressing numerical constraints on* D*.*

[4] Notice that the whole approach holds also if the user does not specify a weight for each preference, but only a global order on preferences. However, in that case, the relative importance among preferences is missed.

For our numerical features, predicates are always the binary operators $C(D) = \{\geq, \leq, >, <, =, \neq\}$, whose second argument is a constant in $\Delta_c(D)$[5]. We note that in some scenarios other concrete domains could be possible, *e.g.*, colors as RGB vectors in an agricultural market, when looking for or selling fruits.

Once we have defined a concrete domain and constraints, we can formally extend propositional logic in order to handle numerical features. We call this language $\mathcal{P}(\mathcal{N})$.

Definition 2 (The language $\mathcal{P}(\mathcal{N})$). *Let $\mathcal{A}$ be a set of propositional atoms, and F a set of pairs $\langle f, D_f \rangle$ each made of a feature name and an associated concrete domain D_f, and let k be a value in D_f. Then the following formulas are in $\mathcal{P}(\mathcal{N})$:*

1. *every atom $A \in \mathcal{A}$ is a formula in $\mathcal{P}(\mathcal{N})$*
2. *if $\langle f, D_f \rangle \in F$, $k \in D_f$, and $c \in \{\geq, \leq, >, <, =, \neq\}$ then (fck) is a formula in $\mathcal{P}(\mathcal{N})$*
3. *if ψ and φ are formulas in $\mathcal{P}(\mathcal{N})$ then $\neg\psi$, $\psi \wedge \varphi$ are formulas in $\mathcal{P}(\mathcal{N})$. We also use $\psi \vee \varphi$ as an abbreviation for $\neg(\neg\psi \wedge \neg\varphi)$, $\psi \Rightarrow \varphi$ as an abbreviation for $\neg\psi \vee \varphi$, and $\psi \Leftrightarrow \varphi$ as an abbreviation for $(\psi \Rightarrow \varphi) \wedge (\varphi \Rightarrow \psi)$.*

We call $\mathcal{L}_{\mathcal{A},F}$ the set of formulas in $\mathcal{P}(\mathcal{N})$ built using $\mathcal{A}$ and F.

In order to define a formal semantics of $\mathcal{P}(\mathcal{N})$ formulas, we consider interpretation functions $\mathcal{I}$ that map propositional atoms into $\{\mathsf{true}, \mathsf{false}\}$, feature names into values in their domain, and assign propositional values to numerical constraints and composite formulas according to the intended semantics.

Definition 3 (Interpretation and models). *An interpretation $\mathcal{I}$ for $\mathcal{P}(\mathcal{N})$ is a function (denoted as a superscript $\cdot^{\mathcal{I}}$ on its argument) that maps each atom in $\mathcal{A}$ into a truth value $A^{\mathcal{I}} \in \{\mathsf{true}, \mathsf{false}\}$, each feature name f into a value $f^{\mathcal{I}} \in D_f$, and assigns truth values to formulas as follows:*

- *$(fck)^{\mathcal{I}} = \mathsf{true}$ iff $f^{\mathcal{I}}ck$ is true in D_f, $(fck)^{\mathcal{I}} = \mathsf{false}$ otherwise*
- *$(\neg\psi)^{\mathcal{I}} = \mathsf{true}$ iff $\psi^{\mathcal{I}} = \mathsf{false}$, $(\psi \wedge \varphi)^{\mathcal{I}} = \mathsf{true}$ iff both $\psi^{\mathcal{I}} = \mathsf{true}$ and $\varphi^{\mathcal{I}} = \mathsf{true}$, according to truth tables for propositional connectives.*

Given a formula φ in $\mathcal{P}(\mathcal{N})$, we denote with $\mathcal{I} \models \varphi$ the fact that $\mathcal{I}$ assigns true to φ. If $\mathcal{I} \models \varphi$ we say $\mathcal{I}$ is a model *for φ, and $\mathcal{I}$ is a model for a set of formulas when it is a model for each formula.*

Clearly, an interpretation $\mathcal{I}$ is completely defined by the values it assigns to propositional atoms and numerical features. Let $\mathcal{A} = \{\texttt{Sedan}, \texttt{GPL}\}$ be a set of propositional atoms, $D_{\texttt{price}} = \{0, \ldots, 60000\}$ and $D_{\texttt{year\_warranty}} = \{0, 1, \ldots, 5\}$ be two concrete domains for the features `price`, `year_warranty`, respectively. A model $\mathcal{I}$ for both formulas:

$$\left\{ \begin{array}{l} \texttt{Sedan} \wedge (\texttt{GPL} \Rightarrow (\texttt{year\_warranty} \geq 1)), \\ (\texttt{price} \leq 5000) \end{array} \right\}$$

[5] So, strictly speaking, $C(D)$ would be a set of unary predicates with an infix notation, *e.g.*, $x > 5$ is in fact a predicate $P_{>5}(x)$ which is `true` for all values of D_x greater than 5 and `false` otherwise; however, this distinction is not necessary in our formalization.

is $\texttt{Sedan}^{\mathcal{I}} = \text{true}$, $\texttt{GPL}^{\mathcal{I}} = \text{false}$, $\texttt{year\_warranty}^{\mathcal{I}} = 0$, $\texttt{price}^{\mathcal{I}} = 4500$. Given a set of formulas $\mathcal{T}$ in $\mathcal{P}(\mathcal{N})$ (representing an ontology), we denote by $\mathcal{I} \models \mathcal{T}$ the fact that $\mathcal{I}$ is a *model* for $\mathcal{T}$. An ontology is *satisfiable* if it has a model. $\mathcal{T}$ logically implies a formula φ, denoted by $\mathcal{T} \models \varphi$ iff φ is true in all models of $\mathcal{T}$. We denote by $\mathcal{M}_{\mathcal{T}} = \{\mathcal{I}_1, \ldots, \mathcal{I}_n\}$, the set of all models for $\mathcal{T}$, and omit the subscript when no confusion arises.

The following remarks are in order for the concrete domains of our e-marketplace-oriented scenarios:

1. domains are *discrete*, with a *uniform* discretization step ϵ. For example, if the seller states he cannot deliver a car before one month, he is saying that the delivery time will be at least in one month and one day ($\texttt{deliverytime} \geq 32$), where $\epsilon = 1$ (in days).
2. domains are *finite*; we denote with $\max(D_f)$ and $\min(D_f)$ the maximum and minimum values of each domain D_f.
3. even for the same feature name, concrete domains are *marketplace dependent*. For , let us consider $\texttt{price}$ in two different marketplace scenarios: pizzas and cars. For the former one, the discretization step ϵ is the €-cent: the price is usually something like 4.50 or 6.00 €. On the other hand, specifying the price of a car we usually have 10500 or 15000 €; then the discretization step in this case can be fixed as 100 €.

The above Point 1 and the propositional composition of numerical constraints imply that the operators $\{\geq, \leq, >, <, =, \neq\}$ can be reduced to $\{\geq, \leq\}$.

Definition 4 (successor/predecessor). *Given two contiguous elements k_i and k_{i+1} in a concrete domain D_f we denote by:*

- $s : D_f \rightarrow D_f$ *the successor function:* $s(k_i) = k_{i+1} = k_i + \epsilon$
- $p : D_f \rightarrow D_f$ *the predecessor function:* $p(k_{i+1}) = k_i = k_{i+1} - \epsilon$

Clearly, $\max(D_f)$ has no successor and $\min(D_f)$ has no predecessor. Based on the above introduced notions, we can reduce $C_m(D_f)$ to $\{\leq, \geq\}$ using the following transformations:

$$f = k \longrightarrow (f \leq k) \wedge (f \geq k) \tag{1}$$

$$f \neq k \longrightarrow (f < k) \vee (f > k) \tag{2}$$

$$f > k \longrightarrow f \geq (k + \epsilon) \longrightarrow f \geq s(k) \tag{3}$$

$$f < k \longrightarrow f \leq (k - \epsilon) \longrightarrow f \leq p(k) \tag{4}$$

3 Multi-issue Bilateral Negotiation in $\mathcal{P}(\mathcal{N})$

According to Ragone et al. [32], we use logic formulas in $\mathcal{P}(\mathcal{N})$ to model the buyer's demand and the seller's supply. Relations among issues, both propositional and numerical, are represented by a set $\mathcal{T}$ –for Theory (*i.e.*, ontology)—of $\mathcal{P}(\mathcal{N})$ formulas.

In a typical bilateral negotiation scenario, statements within both the buyer's request and the seller's offer can be split into *strict requirements* and *preferences*. Strict requirements represent what the buyer and the seller want to be necessarily satisfied in order to accept the final agreement—in our framework we call strict requirements *demand/supply*. *Preferences* are statements involving issues they are willing to negotiate on: preferences may not be satisfied in the final agreement; obviously, the more preferences are fulfilled in the final agreement, the more an agent will be satisfied.

Example 1 *Suppose to have a buyer's request like "I would like a sedan with leather seats. Preferably I would like to pay less than 12000 € furthermore I'm willing to pay up to 15000 € if warranty is greater or equal than 3 years. I don't want to pay more than 17000 € and I don't want a car with a warranty less than 2 years". In this example we identify:*
demand*: I want a sedan with leather seats. I don't want to pay more than 17000 €. I don't want a car with a warranty less than 2 years*
preferences*: Preferably I would like to pay less than 12000 , furthermore I'm willing to pay up to 15000 € if warranty is greater or equal than 3 years.*

Definition 5 (Demand, Supply, Agreement). *Given an ontology $\mathcal{T}$ represented as a set of formulas in $\mathcal{P}(\mathcal{N})$ representing the knowledge on a marketplace domain*

- *a buyer's* demand *is a formula β (for Buyer) in $\mathcal{P}(\mathcal{N})$ such that $\mathcal{T} \cup \{\beta\}$ is satisfiable.*
- *a seller's* supply *is a formula σ (for Seller) in $\mathcal{P}(\mathcal{N})$ such that $\mathcal{T} \cup \{\sigma\}$ is satisfiable.*
- *$\mathcal{I}$ is a* possible deal *between β and σ iff $\mathcal{I} \models \mathcal{T} \cup \{\sigma, \beta\}$, that is, $\mathcal{I}$ is a model for $\mathcal{T}$, σ, and β. We also call $\mathcal{I}$ an* agreement.

The seller and the buyer model in σ and β the minimal requirements they accept for the negotiation. If seller and buyer have set strict attributes that are in conflict with each other, that is $\mathcal{M}_{\mathcal{T} \cup \{\sigma,\beta\}} = \emptyset$, the negotiation ends immediately because it is impossible to reach an agreement. If the participants are willing to avoid the *conflict deal* [36], and continue the negotiation, it will be necessary they revise their strict requirements.

In the negotiation process both the buyer and the seller express some preferences on attributes, or their combination. The utility function is usually defined based on these preferences. We start defining buyer's and seller's preferences and their associated utilities: u_β for the buyer, and u_σ for the seller.

Definition 6 (Preferences). *The buyer's* negotiation preferences $\mathcal{B} \doteq \{\beta_1, \ldots, \beta_k\}$ *are a set of formulas in $\mathcal{P}(\mathcal{N})$, each of them representing the subject of a buyer's preference, and a utility function $u_\beta : \mathcal{B} \rightarrow \mathbf{Q}^+$ assigning a utility to each formula, such that $\sum_i u_\beta(\beta_i) = 1$.*
Analogously, the seller's negotiation preferences $\mathcal{S} \doteq \{\sigma_1, \ldots, \sigma_h\}$ *are a set of formulas in $\mathcal{P}(\mathcal{N})$, each of them representing the subject of a seller's preference, and a utility function $u_\sigma : \mathcal{S} \rightarrow \mathbf{Q}^+$ assigning a utility to each formula, such that $\sum_j u_\sigma(\sigma_j) = 1$.*

Buyer's request in Example 1 is then formalized as:

$$\begin{aligned}
\beta &= \texttt{Sedan} \wedge \texttt{Leather_seats} \wedge (\texttt{price} \leq 17000) \wedge \\
&\quad (\texttt{year_warranty} \geq 2) \\
\beta_1 &= (\texttt{price} \leq 12000) \\
\beta_2 &= (\texttt{year_warranty} \geq 3) \wedge (\texttt{price} \leq 15000)
\end{aligned}$$

Both agents' utilities are normalized to 1 to eliminate outliers, and make them comparable. Since we assumed that utilities are additive, the *preference utility* is just a sum of the utilities of preferences satisfied in the agreement.

Definition 7 (Preference Utilities). *Let $\mathcal{B}$ and $\mathcal{S}$ be respectively the buyer's and seller's preferences, and $\mathcal{M}_{\mathcal{T}\cup\{\sigma,\beta\}}$ be their agreements set. The* preference utility *of an agreement $\mathcal{I} \in \mathcal{M}_{\mathcal{T}\cup\{\sigma,\beta\}}$ for a buyer and a seller, respectively, are defined as:*

$$u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) \doteq \Sigma\{u_\beta(\beta_i) \mid \mathcal{I} \models \beta_i\}$$
$$u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I}) \doteq \Sigma\{u_\sigma(\sigma_j) \mid \mathcal{I} \models \sigma_j\}$$

where $\Sigma\{\ldots\}$ stands for the sum of all elements in the set.

Notice that if one agent *e.g.*, the buyer, does not specify soft preferences, but only strict requirements, this is modeled as $\beta_1 = \textsf{true}$ and $u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) = 1$, which reflects the fact that an agent accepts whatever agreement not in conflict with its strict requirements. From the formulas related to Example 1, we note that while considering numerical features, it is still possible to express strict requirements and preferences on them. Expressing a strict requirement on numerical features is equivalent to setting a *reservation value* [35] on them. In Example 1 the buyer expresses two reservation values, one on price *"more than 17000 €"* and the other on warranty *"less than 2 years"*.

Both buyer and seller have their own reservation values on each feature involved in the negotiation process. It is the maximum (or minimum) value in the range of possible feature values to reach an agreement, *e.g.*, the maximum price the buyer wants to pay for a car or the minimum warranty required, as well as, from the seller's perspective the minimum price he will accept to sell the car or the minimum delivery time. Usually, each participant knows its own reservation value and ignores the opponent's one. Referring to price and the two corresponding reservation values $r_{\beta,\texttt{price}}$ and $r_{\sigma,\texttt{price}}$ for the buyer and the seller respectively, if the buyer expresses $\texttt{price} \leq r_{\beta,\texttt{price}}$ and the seller $\texttt{price} \geq r_{\sigma,\texttt{price}}$, in case $r_{\sigma,\texttt{price}} \leq r_{\beta,\texttt{price}}$ we have $[r_{\sigma,\texttt{price}}, r_{\beta,\texttt{price}}]$ as a **Z**one **O**f **P**ossible **A**greement — $ZOPA(\texttt{price})$, otherwise no agreement is possible [35]. More formally, given an agreement $\mathcal{I}$ and a feature f, $f^{\mathcal{I}} \in ZOPA(f)$ must hold.

Keeping the price example, let us suppose that the maximum price the buyer is willing to pay is 15000, while the seller minimum allowable price is 10000, then we can set the two reservation values: $r_{\beta,\texttt{price}}$ = 15000 and $r_{\sigma,\texttt{price}}$ = 10000, so the *agreement price* will be in the interval $ZOPA(\texttt{price}) = [10000, 15000]$.

Obviously, the reservation value is considered as private information and will not be revealed to the other party, but will be taken into account by the mediator when the agreement will be computed. Since setting a reservation value on a numerical feature is equivalent to set a strict requirement, then, once the buyer and the seller express their strict requirements, reservation values constraints have to be added to them (see Example 1).

In order to formally define a Multi-issue Bilateral Negotiation problem in $\mathcal{P}(\mathcal{N})$, the only other elements we still need to introduce are the *disagreement thresholds*, also called disagreement payoffs, t_β, t_σ. They are the minimum utility that each agent requires to pursue a deal. Minimum utilities may incorporate an agent's attitude toward concluding the transaction, but also overhead costs involved in the transaction itself, *e.g.*, fixed taxes.

Definition 8 (MBN-$\mathcal{P}(\mathcal{N})$). *Given a $\mathcal{P}(\mathcal{N})$ set of axioms $\mathcal{T}$, a demand β and a set of buyer's preferences $\mathcal{B}$ with utility function $u_{\beta,\mathcal{P}(\mathcal{N})}$ and a disagreement threshold t_β, a supply σ and a set of seller's preferences $\mathcal{S}$ with utility function $u_{\sigma,\mathcal{P}(\mathcal{N})}$ and a disagreement threshold t_σ, a **M**ulti-issue **B**ilateral **N**egotiation problem (MBN) is finding a model $\mathcal{I}$ (agreement) such that all the following conditions hold:*

$$\mathcal{I} \models \mathcal{T} \cup \{\sigma, \beta\} \tag{5}$$

$$u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) \geq t_\beta \tag{6}$$

$$u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I}) \geq t_\sigma \tag{7}$$

Observe that not every agreement $\mathcal{I}$ is a solution of an MBN, if either $u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I}) < t_\sigma$ or $u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) < t_\beta$. Such an agreement represents a deal which, although satisfying strict requirements, is not worth the transaction effort. Also notice that, since reservation values on numerical features are modeled in β and σ as strict requirements, for each feature f, the condition $f^\mathcal{I} \in ZOPA(f)$ always holds by condition (5).

4 Utilities for Numerical Features

Buyer's/seller's preferences are used to evaluate how good is a possible agreement and to select the best one. Obviously, also preferences on numerical features have to be considered, in order to evaluate agreements and how good an agreement is w.r.t. another one. Let us explain the idea considering the demand and buyer's preferences in Example 1.

Example 1. Referring to β, β_1 and β_2 in Example 1 let us suppose the following offer [6]:

$$\sigma = \texttt{Sedan} \wedge (\texttt{price} \geq 15000) \wedge (\texttt{year\_warranty} \leq 5)$$

Three possible agreements between the buyer and the seller are, among others:

$$\begin{aligned}
\mathcal{I}_1 : \{&\texttt{Sedan}^{\mathcal{I}_1} = \text{true}, \texttt{Leather_seats}^{\mathcal{I}_1} = \text{true},\\
&\texttt{price}^{\mathcal{I}_1} = 17000, \texttt{year_warranty}^{\mathcal{I}_1} = 3\}\\
\mathcal{I}_2 : \{&\texttt{Sedan}^{\mathcal{I}_2} = \text{true}, \texttt{Leather_seats}^{\mathcal{I}_2} = \text{true},\\
&\texttt{price}^{\mathcal{I}_2} = 16000, \texttt{year_warranty}^{\mathcal{I}_2} = 4\}\\
\mathcal{I}_3 : \{&\texttt{Sedan}^{\mathcal{I}_3} = \text{true}, \texttt{Leather_seats}^{\mathcal{I}_3} = \text{true},\\
&\texttt{price}^{\mathcal{I}_3} = 15000, \texttt{year_warranty}^{\mathcal{I}_3} = 5\}
\end{aligned}$$

With respect to the above set of agreements, we can say that $\mathcal{I}_1$ is the most convenient from the seller's point of view (and the most disadvantageous for the buyer) whilst $\mathcal{I}_3$ represents the worst agreement from the seller's perspective (and the best one for the buyer). In fact, if the agreement was $\mathcal{I}_1$ then the seller would get 17000 € providing a

[6] For illustrative purpose, in this example we consider an offer where only strict requirements are explicitly stated. Of course, in the most general case also the seller can express his preferences.

warranty of just 3 years rather than getting 15000 € selling a car with 5 years warranty in case the agreement was $\mathcal{I}_3$. In the above set of agreements, $\mathcal{I}_2$ seems to be the best compromise between buyer and seller.

The above example highlights the need for utility functions taking into account the value of each numerical feature involved in the negotiation process. Of course, for each feature two utility functions are needed; one for the buyer — $u_{\beta,f}$, the other for the seller — $u_{\sigma,f}$. These functions have to satisfy at least the basic properties enumerated below. For the sake of conciseness, we write u_f when the same property holds both for $u_{\beta,f}$ and $u_{\sigma,f}$. :

1. u_f is normalized to $[0,1]$ to make agents' preferences comparable and to avoid that agents could manipulate their preference revelation in order to get a better deal stating a bigger (or smaller) number of preferences. Given the pair $\langle f, D_f \rangle$, it must be defined over the domain D_f.
2. From Example 1 we note the buyer is more satisfied as the price decreases whilst the seller is less satisfied. Hence, u_f has to be monotonic and whenever $u_{\beta,f}$ increases then $u_{\sigma,f}$ decreases and vice versa.
3. There is no utility for the buyer if the agreed value on price is greater or equal than its reservation value $r_{\beta,\texttt{price}}$ =17000 and there is no utility for the seller if the price is less than or equal to $r_{\sigma,\texttt{price}}$ =15000. Since concrete domains are finite, for the buyer the best possible price is $\min(D_{\texttt{price}})$ whilst for the seller is $\max(D_{\texttt{price}})$. The contrary holds if we refer to year warranty.

Definition 9 (Feature Utilities). *Let $\langle f, D_f \rangle$ be a pair made of a feature name f and a concrete domain D_f and r_f be a reservation value for f. A* **feature utility function** *$u_f : D_f \rightarrow [0,1]$ is a monotonic function such that*
– if u_f monotonically increases then (see Figure 1)

$$\begin{cases} u_f(v) = 0, v \in [\min(D_f), r_f] \\ u_f(\max(D_f)) = 1 \end{cases} \tag{8}$$

– if u_f monotonically decreases then

$$\begin{cases} u_f(v) = 0, v \in [r_f, \max(D_f)] \\ u_f(\min(D_f)) = 1 \end{cases} \tag{9}$$

Given a buyer and a seller, if $u_{\beta,f}$ increases then $u_{\sigma,f}$ decreases and vice versa.

The simplest utility functions are the two linear functions:

$$u_f(v) = \begin{cases} 1 - \frac{v - \min(D_f)}{r_f - \min(D_f)} \text{ , } v \in [\min(D_f), r_f) \\ \\ \qquad\qquad\qquad 0 \text{ , } v \in [r_f, \max(D_f)] \end{cases} \tag{10}$$

if it monotonically decreases and

$$u_f(v) = \begin{cases} 1 - \frac{v - \max(D_f)}{r_f - \max(D_f)} \text{ , } v \in [r_f, \max(D_f)) \\ \\ \qquad\qquad\qquad 0 \text{ , } v \in [\min(D_f), r_f] \end{cases} \tag{11}$$

if it monotonically increases (see Figure 1).

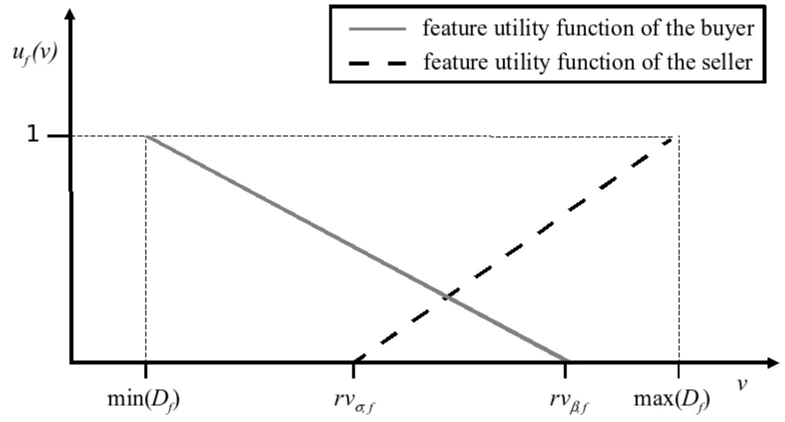

Fig. 1. Linear utility functions

5 Computing Pareto agreements in $\mathcal{P}(\mathcal{N})$

Among all possible agreements that we can compute, given an ontology $\mathcal{T}$ as constraint, we are interested in agreements that are Pareto-efficient. Among them, we are interested on the ones either maximizing the sum of utilities—maximum welfare—or maximizing their product—Nash-bargaining solution [27]. We now outline how an actual solution can be found solving an optimization problem.

5.1 Objective functions

Here we define functions to be maximized to find a solution to an optimization problem.

First of all, we introduce a set of fresh propositional atoms w.r.t. to the ones in $\mathcal{A}$ (see Definition 2) and a new set of equivalence axioms to $\mathcal{T}$. Let:

- $\{B_1, \ldots, B_k, S_1, \ldots, S_h\}$ be $k + h$ new propositional atoms different from the ones in $\mathcal{A}$;
- $\mathcal{T}' = \mathcal{T} \cup \{B_i \Leftrightarrow \beta_i | i = 1, \ldots, k\} \cup \{S_j \Leftrightarrow \sigma_j | j = 1, \ldots, h\}$.

Now, it is easy to see that:

- The new ontology $\mathcal{T}'$ is equivalent to $\mathcal{T}$ w.r.t. implication of formulas in $\mathcal{L}_{\mathcal{A},F}$ (the language we used to model $\mathcal{T}$,β,σ and agents' preferences β_i and σ_j). That is, given a formula $\psi \in \mathcal{L}_{\mathcal{A},F}$, if $\mathcal{T}' \models \psi$ then $\mathcal{T} \models \psi$. Hence, we can use $\mathcal{T}'$ being sure that implication relations will be exactly the same we would have using $\mathcal{T}$.
- given an agreement $\mathcal{I}$, we have $\mathcal{I} \models \beta_i$ iff $\mathcal{I} \models B_i$ and $\mathcal{I} \models \sigma_i$ iff $\mathcal{I} \models S_i$. That is, if B_i is true in a model $\mathcal{I}$ then the corresponding formula β_i is evaluated true w.r.t. $\mathcal{I}$ (and similarly for S_i and σ_i). In order to check if a preference is evaluated true in an agreement we can just look at the truthfulness value of the corresponding propositional atom.

In order to formulate functions to be maximized involving preferences expressed as formulas in $\mathcal{P}(\mathcal{N})$, let $\{b_1, \ldots, b_k\}$ binary variables one-one with $\{B_1, \ldots, B_k\}$ and similarly $\{s_1, \ldots, s_h\}$ for $\{S_1, \ldots, S_h\}$. The functions representing respectively buyer's and seller's utility over preferences can hence be defined as:

$$u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) = \sum_{i=1}^{k} b_i u_\beta(\beta_i) \tag{12}$$

$$u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I}) = \sum_{j=1}^{h} s_j u_\sigma(\sigma_j) \tag{13}$$

where, given an agreement $\mathcal{I}$, $b_i = 1$ if $B_i^{\mathcal{I}} = \mathsf{true}$, $b_i = 0$ if $B_i^{\mathcal{I}} = \mathsf{false}$ and similarly s_j for S_j.

As highlighted in Section 4, also utilities over numerical features have to be taken into account while finding the best solution for both the buyer and the seller. Hence, for each feature f involved in the negotiation process we have a ***feature utility function*** for the buyer $u_{\beta,f}$ and one for the seller $u_{\sigma,f}$. For instance, if we consider `price` and the linear function in equations (10) and (11) we likely will have:

$$u_{\beta,\mathtt{price}}(v) = \begin{cases} 1 - \frac{v - \max(D_{\mathtt{price}})}{r_{\beta,\mathtt{price}} - \max(D_{\mathtt{price}})} \\ 0 \end{cases}$$

$$u_{\sigma,\mathtt{price}}(v) = \begin{cases} 1 - \frac{v - \min(D_{\mathtt{price}})}{r_{\sigma,\mathtt{price}} - \min(D_{\mathtt{price}})} \\ 0 \end{cases}$$

5.2 The Optimization Problem

In order to find a Pareto agreement between the traders, we have to define **global** utility functions that take into account all their objective functions (both their preference utility functions and their feature utility functions).

$$u_\beta^G(\mathcal{I}) = g_\beta(u_{\beta,\mathcal{P}(\mathcal{N})}, u_{\beta,f_1}, \ldots, u_{\beta,f_n})$$
$$u_\sigma^G(\mathcal{I}) = g_\sigma(u_{\sigma,\mathcal{P}(\mathcal{N})}, u_{\sigma,f_1}, \ldots, u_{\sigma,f_n})$$

where n is the cardinality of F, *i.e.*, the number of concrete features involved in the negotiation process.

Definition 10. *Given a MBN-$\mathcal{P}(\mathcal{N})$, we define* MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ *as the problem of finding an agreement $\mathcal{I}$ for which $u_\sigma^G(\mathcal{I}) + u_\beta^G(\mathcal{I})$ is maximal and* MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ *the problem of finding an agreement $\mathcal{I}$ for which $u_\sigma^G(\mathcal{I}) \cdot u_\beta^G(\mathcal{I})$ is maximal.*

Clearly, every solution for MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ is also a Pareto agreement, but not vice versa [14].

In addition to the set of functions to maximize, in our setting, we have three different sets of constraints:

1. the (modified) ontology $\mathcal{T}'$ — see the beginning of Section 5
2. strict requirements β and σ, including reservation values over numerical features
3. conditions (6) and (7) of an MBN on disagreement thresholds t_β and t_σ — see the definition of MBN-$\mathcal{P}(\mathcal{N})$ at the end of Section 3

Notice that the constraints involving disagreements thresholds are already linear ones. In order to model as linear constraints also the ones described in points 1 and 2 of the above enumeration, we proceed as follows.

Clause reduction. Obtain a set of clauses $\mathcal{T}''$ s.t. each clause contains only one single numerical constraint and $\mathcal{T}''$ is satisfiable iff $\mathcal{T}' \cup \{\sigma, \beta\}$ does. In order to have such clauses, if after using standard transformations in clausal form [23] you find a clause with two numerical constraints $\chi : A \vee \ldots (f_i c_i k_i) \vee (f_j c_j k_j)$ pick up a new propositional atom $\overline{A}$ and replace χ with the set of two clauses[7]

$$\left\{ \begin{array}{l} \chi_1 : \overline{A} \vee A \vee \ldots \vee (f_i c_i k_i), \\ \chi_2 : \neg\overline{A} \vee A \vee \ldots \vee (f_j c_j k_j) \end{array} \right\}$$

As a final step, for each clause, replace $\neg(f \leq k)$ with $(f \geq s(k))$ and $\neg(f \geq k)$ with $(f \leq p(k))$ (see (3) and 4).

Example 2. Suppose to have the clause

$$\begin{array}{l} \chi : \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee \\ \quad (\texttt{year\_warranty} \geq 3) \vee \neg(\texttt{price} \geq 20500) \end{array}$$

First of all split the clause in the following two

$$\begin{array}{l} \chi_1 : \overline{A} \vee \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee \\ \quad (\texttt{year\_warranty} \geq 3) \\ \chi_2 : \neg\overline{A} \vee \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee \\ \quad \neg(\texttt{price} \geq 20500) \end{array}$$

then change the second one in

$$\begin{array}{l} \chi_2 : \neg\overline{A} \vee \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee \\ \quad (\texttt{price} \leq 20000) \end{array}$$

Notice that here we consider $\epsilon = 500$ for the concrete domain $D_{\texttt{price}}$.

The last step is needed in order to have a straight mapping from logical clauses to linear inequalities as shown in the following.

[7] It is well known that such a transformation preserves logical entailment [37].

Encoding clauses into linear inequalities. We use a modified version of well-known encoding of clauses into linear inequalities (*e.g.*, [28, p.314]) so that every solution of the inequalities identifies a model of $\mathcal{T}''$. If we identify true with values in $[1 \ldots \infty)$ and false with values in $[0 \ldots 1)$ each clause can be rewritten in a corresponding inequality.

- map each propositional atom A occurring in a clause χ with a binary variable a. If A occurs negated in χ then substitute $\neg A$ with $(1-a)$, otherwise substitute A with a.
- replace $(f \leq k)$ with $\frac{1}{\max(D_f)-k}(\max(D_f) - f)$ and $(f \geq k)$ with $\frac{1}{k} f$.[8]

After this rewriting it is easy to see that, considering $\vee$—logical *or*—as classical addition, in order to have a clause true the evaluation of the corresponding expression must be a value grater or equal to 1.

Example 3. If we consider $\max(D_{\texttt{price}}) = 60000$, continuing Example 2 we have from χ_1 and χ_2 the following inequalities, respectively:

$$\overline{a} + i + (1-a) + \frac{1}{3}\texttt{year\_warranty} \geq 1$$
$$(1-\overline{a}) + i + (1-a) + \frac{1}{60000-20000}(60000 - \texttt{price}) \geq 1$$

where $\overline{a}$, i, a are binary variables representing propositional terms $\overline{A}$, `ItalianMaker` and `AirConditioning`.

Looking at the example below, it should be clear the reason why only one numerical constraint is admitted in a clause.

Example 4. Let us transform the following clause without splitting it in the two corresponding ones

$$\overline{\chi} : \texttt{ItalianMaker} \vee (\texttt{year\_warranty} \geq 3) \vee (\texttt{price} \leq 20000)$$

the corresponding inequality is then

$$i + \frac{1}{3}\texttt{year\_warranty} + \frac{1}{60000-20000}(60000 - \texttt{price}) \geq 1$$

The interpretation $\{\texttt{year\_warranty} = 2, \texttt{price} = 40000\}$ is not a model for $\overline{\chi}$ while the inequality is satisfied.

Notice that the above encoding of the constraints involved in the negotiation process as well as the definition of MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ are not depending from how u_β^G and u_σ^G are formalized.

If both u_β^G and u_σ^G are a linear combination of the preference and feature utility functions, ***i.e.***, a weigthed sum of $u_{\mathcal{P}(\mathcal{N})}$, u_{f_1}, ..., u_{f_n} normalized to 1, then MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ remains linear while MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ is quadratic[17].

[8] If in the previous step we had not replaced $\neg(f \leq k)$ and $\neg(f \geq k)$ with their equivalent positive forms, then we would have had also to transform $\neg(f \leq k)$ in $\frac{1}{s(k)} f$ and $\neg(f \geq k)$ in $\frac{1}{\max(D_f)-p(k)}(\max(D_f) - f)$

5.3 Computational issues

The preceding sections proved that both MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ could be solved by general mathematical programming methods, namely, MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ could be solved by Integer Linear Programming (ILP) [28] and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ by quadratic programming [17]. Since both methods are adequate to solve NP-complete Optimization (NPO) Problems [1], we can conclude that both problems are contained in NPO.

Theorem 1. *Given an instance M of either* MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ *or* MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$*, finding an optimal solution for M is a problem in* NPO.

Proof. The above Sections 5.1 and 5.2 proved that all constraints can be translated into linear inequalities involving variables ranging over either integer or rational numbers. In the case of MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$, the objective function is linear too, and the optimization problem is known to belong to the class NPO [28]. In the case of MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$, the objective function is quadratic; but also in this case, the optimization of a quadratic function over linear constraints is known to be in NPO [17]. □

We recall that when a problem is in NPO, one can ask whether there exists an algorithm approximating the optimum within some guaranteed bound. In the theory of Computational Complexity, this question amounts to ask whether one or both problems are NPO-complete, or instead whether they belong to the smaller class APX of problems that can be approximated within constant bounds. For instance, it is known that MINIMUM VERTEX COVER can be easily approximated within a constant bound of 2 [1]. The question is not merely theoretical, since if either problem admitted some approximation algorithm, using general mathematical programming for solving them might be an overshoot. However, the theorem below proves that this is not the case, by giving an L-reduction [1] of MAX-WEIGHTED-SAT to both problems.

Definition 11. MAX-WEIGHTED-SAT *is the following* NPO*-complete problem: given set of atoms $\mathcal{A}$, a propositional formula $\varphi \in \mathcal{L}_{\mathcal{A}}$ and a weight function $w : \mathcal{A} \rightarrow \mathbf{N}$, find a truth assignment satisfying φ such that the sum of the weights of true variables is maximum.*

It is known that MAX-WEIGHTED-SAT is NPO-complete [1], hence not approximable within any constant bound. We now prove NPO-hardness of both MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$—that is, we prove that tailored algorithms yielding solutions approximated within a guaranteed bound from the optimum are unlikely to exist—by reducing MAX-WEIGHTED-SAT to both problems.

Theorem 2. MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ *and* MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ *are* NPO*-complete problems, even if $\mathcal{T}$ is in 3CNF and both $\mathcal{B}$ and $\mathcal{S}$ are sets of positive literals.*

Proof. Let $W = \langle \varphi, w \rangle$ be an instance of MAX-WEIGHTED-SAT, with φ in 3CNF, and let $\mathcal{A} = \{A_1, \ldots, A_n\}$.

First, we prove the claim for MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$. Define an instance M_+

of MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ as follows. Let $\mathcal{T} = \varphi$, $\mathcal{B} = \mathcal{S} = \mathcal{A}$. Moreover, let $S = \sum_{i=1}^{n} w(A_i)$ and let $u_\beta(A_i) = u_\sigma(A_i) = \frac{1}{S} w(A_i)$ for $i = 1, \ldots, n$. Finally, let $t_\beta = t_\sigma = 0$. Clearly, every solution for W is also a solution for M_+, and for every model $\mathcal{I}$, the value of the objective function $u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) + u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I})$ is proportional with a factor $\frac{2}{S}$ to the one for W. Hence, the above is an L-reduction with $\alpha = \frac{1}{2}S$ and $\beta = \frac{1}{2}S$.

Similarly, define an instance $M_\times$ of MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ as follows. Let $\mathcal{T} = \varphi$, $\mathcal{B} = \mathcal{A}$, $\mathcal{S} = \{A_0\}$ where A_0 is a new atom not in $\mathcal{A}$. Moreover, let $S = \sum_{i=1}^{n} w(A_i)$ and let $u_\beta(A_i) = \frac{1}{S} w(A_i)$ for $i = 1, \ldots, n$ and $u_\sigma(A_0) = 1$. Finally, let $t_\beta = t_\sigma = 0$. Also in this case, every solution for W satisfies also $\mathcal{T}$, and for every model $\mathcal{I}$ the value of the objective function $u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) \cdot u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I})$ of $M_\times$ is proportional to W's one. Hence, also the above reduction is an L-reduction with $\alpha = \beta = S$.

On the other hand, let S be the smallest integer such that both $u_\beta(A_i) \cdot S$, $u_\sigma(A_i) \cdot S$ are integers for every $i = 1 \ldots n$. Also in this case, a solution for MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ can be transformed into a solution for W by multiplying it by S. $\square$

The proof of the above theorem highlights the fact that a source of complexity for both MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$ comes from the inherent conflicts between the preferences of a single agent. In conclusion, linear and quadratic programming are computationally adequate for solving MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ and MAX-PROD-MBN-$\mathcal{P}(\mathcal{N})$.

6 The bargaining process

Summing up, the negotiation process covers the following steps:

Preliminary Phase. The buyer defines strict β and preferences $\mathcal{B}$ with corresponding utilities $u_\beta(\beta_i)$, as well as the threshold t_β, and similarly the seller σ, $\mathcal{S}$, $u_\sigma(\sigma_j)$ and t_σ. Here we are not interested in how to compute t_β,t_σ, and the weight of each preference; we assume they are determined in advance by means of either direct assignment methods (Ordering, Simple Assessing or Ratio Comparison) or pairwise comparison methods (like AHP and Geometric Mean) [31]. Both agents inform the mediator about these specifications and the ontology $\mathcal{T}$ they refer to. Notice that for each feature involved in the negotiation process, both in β and σ their respective reservation values are set either in the form $f \leq r_f$ or in the form $f \geq r_f$.

Keeping the example referring to the automotive e-marketplace, buyer and seller specify respectively their strict requirements β and σ, thresholds t_β and t_σ, preferences and the worth thereof as in the following:

β = `Sedan` $\wedge$ (`price` ≤ 30000) $\wedge$ (`km_warranty` ≥ 120000) $\wedge$ (`year_warranty` ≥ 4)
β_1 = `GPS_system` $\wedge$ `AlarmSystem`
β_2 = `ExternalColorBlack` $\Rightarrow$ `Leather_seats`
β_3 = (`km_warranty` ≥ 140000)
$u_\beta(\beta_1)$ = 0.5
$u_\beta(\beta_2)$ = 0.2
$u_\beta(\beta_3)$ = 0.3

t_β =0.2

σ = $\texttt{Sedan} \wedge (\texttt{price} \geq 20000) \wedge (\texttt{km\_warranty} \leq 160000) \wedge (\texttt{year\_warranty} \leq 6)$
σ_1 = $\texttt{GPS\_system} \Rightarrow (\texttt{price} \geq 28000)$
σ_2 = $(\texttt{km\_warranty} \leq 150000) \vee (\texttt{year\_warranty} \leq 5)$
σ_3 = $\texttt{ExternalColorGray}$
σ_4 = $\texttt{NavigatorPack}$
$u_\sigma(\sigma_1)$ = 0.2
$u_\sigma(\sigma_2)$ = 0.4
$u_\sigma(\sigma_3)$ = 0.2
$u_\sigma(\sigma_4)$ = 0.2
t_σ =0.2

Let $\mathcal{T}$ be the ontology in $\mathcal{P}(\mathcal{N})$, which the participants refer to:

$$\mathcal{T} = \begin{cases} \texttt{ExternalColorBlack} \Rightarrow \neg\texttt{ExternalColorGray} \\ \texttt{SatelliteAlarm} \Rightarrow \texttt{AlarmSystem} \\ \texttt{NavigatorPack} \Leftrightarrow \texttt{SatelliteAlarm} \wedge \texttt{GPS\_system} \end{cases}$$

Negotiation-Core phase. For each $\beta_i \in \mathcal{B}$ the mediator picks up a new propositional atom B_i and adds the axiom $B_1 \Leftrightarrow \beta_i$ to $\mathcal{T}$, similarly for $\mathcal{S}$. Then, it transforms all the constraints modeled in β, σ and (just extended) $\mathcal{T}$ in the corresponding linear inequalities following the procedures illustrated in Section 5.2. Given the preference utility functions $u_{\beta,\mathcal{P}(\mathcal{N})}(\mathcal{I}) = \sum_{i=1}^{k} b_i u_\beta(\beta_i)$ and $u_{\sigma,\mathcal{P}(\mathcal{N})}(\mathcal{I}) = \sum_{j=1}^{h} s_j u_\sigma(\sigma_j)$, the mediator adds to this set of constraints the ones involving disagreement thresholds $u_{\beta,\mathcal{P}(\mathcal{N})} \geq t_\beta$ and $u_{\sigma,\mathcal{P}(\mathcal{N})} \geq t_\sigma$.

With respect to the above set of constraints, the mediator solves an optimization problem maximizing the sum (or the product) of global utilities for both buyer $u_\beta^G(\mathcal{I})$ and seller $u_\sigma^G(\mathcal{I})$. The returned solution to the optimization problem is the agreement proposed to the buyer and the seller. Notice that this solution is a Pareto optimal one, furthermore the solution proposed by the mediator is also a *fair* solution, if among all the Pareto-optimal solutions we take the one maximizing the product of utilities of both the buyer and the seller (see Section 5.2).

With reference to the previous example the mediator proposes the following agreement to the players. We omit for the sake of conciseness propositional atoms interpreted as false, then the final agreement is:

$$\begin{aligned} \mathcal{I} : \{&\texttt{Sedan}^\mathcal{I} = \text{true}, \texttt{ExternalColorGray}^\mathcal{I} = \text{true}, \\ &\texttt{SatelliteAlarm}^\mathcal{I} = \text{true}, \texttt{GPS\_system}^\mathcal{I} = \text{true}, \\ &\texttt{NavigatorPack}^\mathcal{I} = \text{true}, \texttt{AlarmSystem}^\mathcal{I} = \text{true}, \\ &\texttt{price}^\mathcal{I} = 28000, \texttt{km\_warranty}^\mathcal{I} = 160000, \texttt{year\_warranty}^\mathcal{I} = 5\} \end{aligned}$$

From this point on, it is a *take-it-or-leave-it* offer, as the participants can either accept or reject the proposed agreement [18].

7 Discussion

In this section we briefly analyze the properties characterizing the negotiation mechanism here proposed.

Individual rationality. Individual rationality means that no agent will get a worse payoff by participating in the mechanism as compared to not participating, *i.e.*, an agent does not ever lose by participating. This property is assured by the fact that each agent, among his preferences, expresses a threshold, representing the disagreement payoff (see Section 3). This threshold is one of the constraints of the optimization problem, which automatically rules out all the agreements that are below that threshold, and that the agent would have regretted if accepted. In other words, it is not possible to have a final agreement with a utility for one of the participant lower than his threshold, so that for the agent would be better not participating at all.

Efficiency. Our mechanism is efficient, as the mediator computes a solution which is Pareto-efficient (see Section 5.2). Besides, among all the possible Pareto-efficient solutions, it chooses either the one maximizing the sum of the utilities (*welfare maximization*), or the one maximizing the product of the utilities, also known as the *fair* solution or the Nash bargaining solution [27] [9]. Another definition of efficiency is the *ex-post* efficiency, meaning that an agent will not change his strategy even after he observed the result of the negotiation. That is he will not change his preferences after observing the preferences expressed by the other agent. We do not address this issue in this paper as the mediator does not disclose the agent's preferences to the other party, at the end of the negotiation the players can see only the proposed final agreement.

Budget balance. A mechanism is budget balanced when it does not make either a profit or loss, meaning that the amount of money collected and distributed from and to the agents is equal. Our mechanism is budget balanced as the price paid by the buyer is exactly the same of the price received by the seller. That is there is not any third party that can either to subside or exploit the buyer and the seller [26]. In our framework the mediator has more the role of a nonbinding arbitrator than a broker [35]. Even if we do not exclude that the traders have to pay some fee if the transaction ends successfully.

Incentive compatibility. A mechanism is incentive compatible if each participant can maximize his (expected) utility by reporting his true preferences, given that the other participants do the same [26], *i.e.*, the truth-telling strategy is the best response strategy, given that also the other agents report truthfully. Our mechanism is not incentive compatible, but this is not a surprising result, as Myerson and Satterthwaite [26] proved that in a bilateral trading with incomplete information [10] it is impossible to design a Bayes-Nash incentive-compatible mechanism that is simultaneously efficient, budget balanced and individually rational. Therefore it is only possible to design an incentive-compatible mechanism that achieves any two of these three properties. They proved the general impossibility to have such a mechanism without outside subsidies, *i.e.*, without relaxing the budget balanced property.

Furthermore we point out that in our mechanism the absence of incentive compatibility only affect the revelation of the reservation value on numerical features, *e.g.*, price, warranty, etc. Indeed an agent cannot manipulate the mechanism expressing a

[9] Observe that there can be many fair-Pareto-efficient solutions, or welfare solutions. Nevertheless, as they are perfectly equivalent, the mediator will simply randomly choose one of them and propose that to the participants.

[10] With the term "incomplete information" we indicate that each agent ignores both preferences and the worth thereof of the opponent.

bigger, or smaller, number of preferences, as agent's utility is normalized to 1, in order to both rule out such a possibility and make utilities expressed by traders comparable (see Def. 6 in Section 3). Besides, even if an agent expresses a *fake* preference this fact could only lower its utility as, normalizing utilities to 1, the value of the "true" preferences will be lowered.

Moreover the threshold is only used to check if an agreement can be accepted or not by the agents. Therefore if agents express a bigger disagreement threshold they can only rule out some agreements which instead were acceptable. In this way they would end up with a disagreement payoff while they could have reached a satisfactory agreement by acting truthfully. Similarly if they express a smaller threshold they would only add the possibility to end up with an agreement which is not Pareto-efficient (w.r.t. their true evaluation).

Besides, there is no incentive for agents to lie on preferences which do not involve numerical features, because they could only end up with a car different from their true expectations, *e.g.*, a car with a diesel engine instead that a gas one. For what concern numerical features the argument is analogous to the one of the thresholds. Let's take the price example. If the buyer reveals a value lower than her true reservation value, while the seller reveals one higher than his true value, then the buyer's value might be lower than the seller's one, when, in fact, their true value were compatible. Hence, also in this case the player could end up with a disagreement payoff, when being truthfully they could reach a better payoff. Therefore even if the mechanism is not incentive compatible w.r.t. the numerical features the agents are motivated to act in a truthful way, because of the risk to end up with a disagreement payoff when instead a better (efficient) agreement exists.

8 Related Work and conlusion

Automated bilateral negotiation has been widely investigated, both in artificial intelligence and in microeconomics research communities, so this section is necessarily far from complete. Attempting a coarse subdivision, we may note that, in classic game theory, the bargaining problem has been modeled either as *cooperative* or *non-cooperative* games [16]. In the first approach, the aim is finding a solution given a set of possible outcomes, so given a set of axioms and a coalition, one determines how to split the surplus among the participants. Instead, in non-cooperative games there are a well-defined set of rules and strategies. In such games it is possible to define an *equilibrium* strategy, which ensures the rational outcomes of a game: no player could benefit by unilaterally deviating from her strategy, given that the other players follow their own strategies [21].

AI-oriented research has usually focused on automated negotiation among agents and on designing high-level protocols for agent interaction [22]. Agents can play different roles: act on behalf of a buyer or seller, but also play the role of a mediator or facilitator. Depending on the presence of a mediator we can distinguish between *centralized* and *distributed* approaches. In the former, agents elicit their preferences and then a mediator, or some central entity, selects the most suitable deal based on them. In the latter, agents negotiate through various negotiation steps reaching the final deal by means of intermediate deals, without any external help [8]. Distributed approaches do not allow the presence of a mediator because—as stated by Kraus [20, p.25]—agents cannot agree on any entity, so they do not want to disclose their preferences to a third party, that, missing any relevant information, could not help agents. In dynamic systems a predefined conflict resolution cannot be allowed, so the presence of a mediator is discouraged. On the other hand the presence of a mediator can be extremely useful in designing negotiation mechanisms and in practical important commerce settings. According to MacKie-Mason and Wellman [24], negotiation mechanisms often involve the presence of a mediator [11], which collects information from bargainers and exploits them in order to propose an efficient negotiation outcome. The presence of a trusted third party can help parties to reach a Pareto-efficient agreement. As pointed out by Raiffa et al. [35, p.311], usually bargainers may not want to disclose their preferences or utilities to the other party, yet they can be more willing to reveal these information to a trusted—automated—mediator, helping negotiating parties to achieve efficient and equitable outcomes. The presence of a mediator and the one-shot protocol is an incentive for the two parties to reveal the true preferences, because they can trust in the mediator and they have a single possibility to reach the agreement with that counterpart.

Several approaches adopt a mediator [12, 19, 15]. In the paper by Fatima et al. [12] an extended alternating-offers protocol is presented, with the presence of a mediator, which improves the utility of both agents. No inter-dependent issues are taken into account. Klein et al. [19] propose a mediated-negotiation approach for complex contracts, where inter-dependency among issues is investigated. The agreement is a vector of issues, having value 0 or 1 depending on the presence or absence of a given contract clauses. Only binary dependencies between issues are considered: the agent's utility is computed through an influence matrix, where each cell represents the utility of a given pair of issues. However in this approach no semantic relations among issues are investigated.

A large number of negotiation mechanisms have been proposed and studied in the literature; it is possible to distinguish, among other, game-theoretic ones [20, 36], heuristic-based approaches [12, 10] and logic-based approaches. Although game-theoretic and heuristic-based approaches are highly suitable for a wide range of applications, they have some limitations and disadvantages. Often in game-theoretic approaches it is assumed that agents have a complete knowledge about the space of possible outcomes, as well as unbounded computational resources [34]. On the other hand, heuristic-based approaches use empirical evaluations in order to find an agreement, which can be suboptimal, as they do not explore the entire space of possible outcomes. With respect to our approach, the main drawback is that, usually, the issues to negotiate on are fixed in advance and are known by both agents, as in multi-dimensional auctions [30]. Hence agents are not allowed to exchange any additional information during the negotiation process [34].

[11] The most well known –and running– example of mediator is eBay site, where a mediator receives and validates bids, as well as presenting the current highest bid and finally determining the auction winner [24]. Observe also that eBay retains private information of traders, such as selling reservation value.

In the following we give a brief overview of logic-based approaches to automated negotiation, comparing our approach with existing ones and highlighting relevant differences.

8.1 Logic-based approaches

There is a huge amount of literature focused on argumentation-based negotiation [34, 29, 11, 3].In these approaches an agent can accept/reject/critique a proposal of its opponent, so agents can argue about their beliefs, given their desires and so pursue their intentions [29]. With respect to our framework, these approaches require a richer communication language (*e.g.*, modal logic) in order to exchange information and a specific negotiation protocol to constrain the use of the language. While we use a one-shot protocol with the presence of a mediator, which ensures the termination after only one round, in argumentation-based frameworks, usually, agent interactions go back and forth for multiple rounds, without the intervention of a third party. Moreover agents have to be able not only to evaluate opponent proposals or possible agreements, but also generate a critique or a counter-proposal, given the opponent's one. With references to BDI approaches proposed by Parsons et al. [29], *Desires* and *Intentions* match in our framework with Preferences, and *Beliefs* are implicit in each agent: the agent enters the e-marketplace because he believes there will be another agent having what he is searching for.

Several recent logic-based approaches to negotiation are based on propositional logic. Bouveret et al. [6] use Weighted Propositional Formulas (WPF) to express agents preferences in the allocation of indivisible goods, but no common knowledge (as our ontology) is present. The use of an ontology allows *e.g.*, to catch inconsistencies between demand and supply or find out if an agent preference is implied by a preference of its opponent, which is fundamental to model an e-marketplace. Chevaleyre et al. [7] classify utility functions expressed through WPF according to the properties of the utility function (sub/super-additive, monotone, etc.). We used the most expressive functions according to that classification, namely, weights over unrestricted propositional formulas.

Zhang and Zhang [42] adopt a kind of propositional knowledge base arbitration to choose a fair negotiation outcome. However, *common knowledge* is considered as just more entrenched preferences, that could be even dropped in some deals. Instead, the logical constraints in our ontology $\mathcal{T}$ must *always* be enforced in the negotiation outcomes. Finally we devised a *protocol* which the agents should adhere to while negotiating; in contrast, Zhang and Zhang [42] adopt a game-theoretic approach, presenting no protocol at all, since communication between agents is not considered.

We borrow from Wooldridge and Parsons [40] the definition of agreement as a model for a set of formulas from both agents. However, Wooldridge and Parsons [40] only study multiple-rounds protocols and the approach leaves the burden to reach an agreement to the agents themselves, although they can follow a protocol. The approach does not take preferences into account, so that it is not possible to guarantee the reached agreement is Pareto-efficient. Our approach, instead, aims at giving an *automated* support to negotiating agents to reach, in one shot, Pareto agreements. The work presented here builds on the work by Ragone et al. [33], where a basic propositional logic framework endowed of a logical theory was proposed. Afterward Ragone et al. [32] extend the approach also discussing complexity issues. In this paper we further extended the framework, introducing the extended logic $\mathcal{P}(\mathcal{N})$, thus handling numerical features, and showed we are able to compute Pareto-efficient agreements, by solving an optimization problem and adopting a one-shot negotiation protocol.

8.2 Constraint Satisfaction Problems (CSP)

Our framework shares some similarities with approaches related to *Distributed* CSPs and *Soft* CSPs. For what concerns the former, while in Distributed CSPs constraints are distributed among agents and each agent controls its own set of variables [41], in our approach variables (issues) do not belong to any agent and there is a mediator, who rather than simply finding a legal assignment (an assignment to variables that does not violate any constraints), computes assignments which are Pareto-efficients. A further extension of CSPs considering also preferences among solutions is Soft CSPs: preferences are expressed as soft constraints and a solution has to satisfy all hard constraints and as much as possible of soft constraints (preferences) [4]. Depending on the approach, the most important ones (*hierarchical* CSP[39]) can be satisfied, or the number of violated constraints (*Partial* CSP[13]) can be minimized or some satisfaction level (*semiring-based* CSP[4]) can be maximized. Our approach is more similar to the semiring-based one, however in such an approach only a partial order between preferences can be modeled and no conditional preference can be expressed, even if some attempts have been done by Domshlak et al. [9] to mix hard and soft constraints with CP-nets [5], which express qualitative preferences (like conditional ones) over the values of a single property of the outcomes. Moreover, in this approach the translation of conditional preference statements into soft constraints requires some approximations in order to improve the computational efficiency of reasoning about this statements.

8.3 Conclusion

We are aware that there is no universal approach to automate negotiation fitting every scenario, but rather several frameworks suitable for different scenarios, depending on the assumptions made about the domains and agents involved in the interaction. Here, we have proposed a logic-based framework to automate multi-issue bilateral negotiation in P2P e-marketplaces, where agents communicate using the logic $\mathcal{P}(\mathcal{N})$, able to handle both numerical features and non numerical ones. Modeling issues in a $\mathcal{P}(\mathcal{N})$ ontology it is possible to catch inconsistency between preferences and then reach consistent agreements, as well as to discover implicit relations (such as implication) among preferences, which do not immediately appear at the syntactic level. The logic has been mixed to utility theory in order to model preferences both qualitative and quantitative. Exploiting a mediator it is possible to overcome the problem of incomplete information about opponent's preferences. We adopted a one-shot protocol, using a mediator to solve an optimization problem that ensures the Pareto-efficiency of the outcomes. We have also investigated the complexity of the problem of finding Pareto-efficient solution, proving that MAX-SUM-MBN-$\mathcal{P}(\mathcal{N})$ is a NPO-complete problem.

In the near future we plan to extend the approach using more expressive logics, namely, Description Logics, to increase the expressiveness of supply/demand descriptions. We are also investigating other negotiation protocols, without the presence of a mediator, allowing to reach an agreement in a reasonable amount of communication rounds. The use of aggregate operators to be used in order to express both strict requirements and preferences is also under investigation.

Acknowledgments

We acknowledge partial support of EU-FP-6-IST-STREP-026896 "TOWL" project and Apulia Region Strategic Project Ps_092 Distributed Production as Innovative System DIPIS. We wish to thank Stefano Bistarelli, Yevgeniy Vorobeychik, and Michael Wellman for fruitful suggestions and discussions.

References

1. G. Ausiello, P. Crescenzi, V. Kann, G. Gambosi, and A. M. Spaccamela. *Complexity and Approximation: Combinatorial Optimization Problems and Their Approximability Properties*. Springer, 2003.
2. F. Baader and P. Hanschke. A schema for integrating concrete domains into concept languages. In *proc. of IJCAI-91*, pages 452–457, 1991.
3. Jamal Bentahar, Bernard Moulin, John-Jules Ch.Meyer, and Brahim Chaib-draa. A modal semantics for an argumentation-based pragmatics for agent communication. In P. Moraitis I. Rahwan and C. Reed, editors, *Argumentation in Multi-Agent Systems, LNAI*, volume 3366, pages 44–63. Springer, 2005.
4. S. Bistarelli, U. Montanari, and F. Rossi. Semiring-based constraint satisfaction and optimization. *JACM: Journal of the ACM*, 44(2):201–236, 1997.
5. C. Boutilier, R. I. Brafman, C. Domshlak, H. H. Hoos, and D. Poole. Cp-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. *Journal of Articial Intelligence Research*, 21:135–191, 2004.
6. S. Bouveret, M. Lemaitre, H. Fargier, and J. Lang. Allocation of indivisible goods: a general model and some complexity results. In *Proc. of AAMAS '05*, pages 1309–1310, 2005.
7. Y. Chevaleyre, U. Endriss, and J. Lang. Expressive power of weighted propositional formulas for cardinal preference modeling. In *Proc. of KR 2006*, pages 145–152, 2006.
8. Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Negotiating over small bundles of resources. In *Proc. of AAMAS '05*, pages 296–302, 2005.
9. C. Domshlak, S. Prestwich, F. Rossi, K. B. Venable, and T. Walsh. Hard and soft constraints for reasoning about qualitative conditional preferences. *Journal of Heuristics (Special Issue: Preferences and Soft Constraints)*, 12(4-5):263–285, 2006.
10. Peyman Faratin, Carles Sierra, and Nicholas R. Jennings. Using similarity criteria to make issue trade-offs in automated negotiations. *Artificial Intelligence*, 142(2):205–237, 2002.
11. Francesca Toni Fariba Sadri and Paolo Torroni. Dialogues for negotiation: agent varieties and dialogue sequences. In John-Jules Meyer and Milind Tambe, editors, *Intelligent Agent VIII, LNAI 2333*, pages 405–421. Springer-Verlag, 2002.
12. S. Fatima, M. Wooldridge, and N.R. Jennings. Optimal agendas for multi-issue negotiation. In *Proc. of AAMAS'03*, pages 129–136, 2003.
13. Eugene C. Freuder and Richard J. Wallace. Partial constraint satisfaction. *Artificial Intelligence*, 58:21–70, 1992.

14. D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1983.
15. N. Gatti and F. Amigoni. A decentralized bargaining protocol on dependent continuous multi-issue for approximate pareto optimal outcomes. In *Proc. of AAMAS'05*, pages 1213–1214, 2005.
16. E. H. Gerding, D. D. B. van Bragt, and J. A. La Poutre. Scientific approaches and techniques for negotiation: a game theoretic and artificial intelligence perspective. Technical report, SEN-R0005, CWI, 2000.
17. F. Hillier and G. Lieberman. *Introduction to Operations Research*. McGraw-Hill, 2005.
18. N.R. Jennings, P. Faratin, A.R. Lomuscio, S. Parsons, M.J. Wooldridge, and C. Sierra. Automated negotiation: prospects, methods and challenges. *Int. J. of Group Decision and Negotiation*, 10(2):199 – 215, 2001.
19. M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. In *Proc. of AAMAS'02*, pages 753–757, 2002.
20. S. Kraus. *Strategic Negotiation in Multiagent Environments*. The MIT Press, 2001.
21. C. Li, J. Giampapa, and K. Sycara. A review of research literature on bilateral negotiations. Technical report, Carnegie Mellon University, 2003.
22. A. R. Lomuscio, M. Wooldridge, and N. R. Jennings. A classification scheme for negotiation in electronic commerce. *Int Journal of Group Decision and Negotiation*, 12 (1):31–56, 2003.
23. D.W. Loveland. *Automated theorem proving: A logical basis*. North-Holland, 1978.
24. J.K. MacKie-Mason and M.P. Wellman. Automated markets and trading agents. In *Handbook of Computational Economics*. North-Holland, 2006.
25. A. Muthoo. On the strategic role of outside options in bilateral bargaining. *Operations Research*, 43(2):292–297, 1995.
26. Myerson, Roger B. and Satterthwaite, Mark A. Efficient mechanisms for bilateral trading. *Journal of Economic Theory*, 29(2):265–281, April 1983.
27. J. F. Nash The Bargaining Problem. *Econometrica*, 18(2):155–162, 1950.
28. C.H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Prentice-Hall, Inc., 1982.
29. Simon Parsons, Carles Sierra, and Nick Jennings. Agents that reason and negotiate by arguing. *Journal of Logic and Computation*, 8(3):261–292, 1998.
30. Peter R. Wurman and Michael P. Wellman and William E. Walsh. A parametrization of the auction design space. *Games and Economic Behavior*, 35(1-2):304–338, 2001.
31. J.C. Pomerol and S. Barba-Romero. *Multicriterion Decision Making in Management*. Kluwer Series in Operation Research. Kluwer Academic, 2000.
32. A. Ragone, T. Di Noia, E. Di Sciascio, and F.M. Donini. A logic-based framework to compute pareto agreements in one-shot bilateral negotiation. In *Proc. of ECAI'06*, pages 230–234, 2006.
33. A. Ragone, T. Di Noia, E. Di Sciascio, and F.M. Donini. Propositional- logic approach to one-shot multi issue bilateral negotiation. *ACM SIGecom Exchanges*, 5(5):11–21, 2006.
34. Iyad Rahwan, Sarvapali D. Ramchurn, Nicholas R. Jennings, Peter Mcburney, and Simon Parsons. Argumentation-based negotiation. *The Knowledge Engineering Review*, 18(4):343–375.
35. H. Raiffa, J. Richardson, and D. Metcalfe. *Negotiation Analysis - The Science and Art of Collaborative Decision Making*. The Belknap Press of Harvard University Press, 2002.
36. J.S. Rosenschein and G. Zlotkin. *Rules of Encounter*. MIT Press, 1994.
37. S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education-Prentice Hall, 2003.
38. M.P. Wellman. Online marketplaces. In *Practical Handbook of Internet Computing*. CRC Press, 2004.
39. Molly Wilson and Alan Borning. Hierarchical constraint logic programming. *The Journal of Logic Programming*, 16(3 & 4):277–317, August 1993.

40. M. Wooldridge and S. Parsons. Languages for negotiation. In *Proc of ECAI '04*, pages 393–400, 2000.
41. Makoto Yokoo and Katsutoshi Hirayama. Algorithms for distributed constraint satisfaction: A review. *Autonomous Agents and Multi-Agent Systems*, 3(2):185–207, 2000.
42. Dongmo Zhang and Yan Zhang. A computational model of logic-based negotiation. In *Proc. of the AAAI 06*, pages 728–733, 2006.