

Towards a Fuzzy Logic for Automated Multi-issue Negotiation

Azzurra Ragone¹, Umberto Straccia², Tommaso Di Noia¹, Eugenio Di Sciascio¹,
and Francesco M. Donini³

¹ SisInfLab, Politecnico di Bari, Bari, Italy
{a.ragone,t.dinoia,disciascio}@poliba.it

² ISTI-CNR, Pisa, Italy
straccia@isti.cnr.it

³ Università della Tuscia, Viterbo, Italy
donini@unitus.it

Abstract. We present a novel logic-based approach to automate multi-issue bilateral negotiation in e-marketplaces. In such frameworks issues to negotiate on can be multiple, interrelated, and may not be fixed in advance. We use logic to model relations among issues and to allow agents express their preferences on them. In particular, we introduce the logic $\mathcal{P}(\mathcal{N})$, a fuzzy propositional logic extended with concrete domains in order to handle numerical, as well as non numerical features, and to deal with vagueness in buyer/seller preferences. Hence, agents can express preferences as *e.g.*, *I am searching for a passenger car costing about 25000€ yet if the car has a GPS system and more than two-year warranty I can spend up to 28000€.*

We illustrate the theoretical framework, the logical language, the protocol we adopt and show that using a mediator with a proactive behavior we can compute Pareto-efficient agreements.

1 Introduction

Parsons et al. [16] define 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.” Several negotiation mechanisms have been proposed in literature to model different scenarios, as each scenario has its own peculiarities and issues. In this paper we refer to mechanisms to automate multi-issue bilateral negotiation in peer-to-peer (P2P) e-marketplaces [25], 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 who are endowed of peer opportunities as they enter the marketplace. Automated negotiation mechanisms in such e-marketplaces need to represent, in a machine understandable way, the product characteristics, the request/offer descriptions and the preferences of the users entering the marketplace. In fact, differently from e-marketplaces dealing with undifferentiated products (*e.g.*, oil, commodities) in *e.g.*, an automotive e-marketplace price cannot be the only issue to negotiate on, but also other features as warranty, delivery time, as well as model, color,

optionals, have to be taken into account. Moreover such issues may not be established in advanced, as it is a common assumption in many other negotiation scenarios (task and resource allocation, auctions). Therefore there is a need for Knowledge Representation languages able to model relations among issues and to allow agents share a common protocol during the negotiation.

We propose here a *fuzzy propositional logic* endowed with *concrete domains* to model relations among issues and as a communication language between agents. We may represent facts such as that a *Ferrari is an Italian car* ($\text{Ferrari} \Rightarrow \text{ItalianMaker}$), or that a *Sedan is a type of Passenger Car* ($\text{Sedan} \Rightarrow \text{PassengerCar}$), or the fact that a car cannot have at the same time a *Diesel and a GAS engine* ($\text{Diesel} \Rightarrow \neg \text{Gasoline}$). Such kind of relations can be expressed in a Theory (from now on an Ontology) \mathcal{T} . Furthermore, we may represent *preferences*, such as e.g., a seller can state that “*If you want an embedded alarm system you’ll have to wait more than one month*” ($\text{AlarmSystem} \Rightarrow \text{deliverytime} \geq 30$), as well as a buyer can state that “*I would like a passenger car with an alarm system if it costs more than 25000€*” ($\text{PassengerCar} \wedge (\text{price} \geq 25000 \Rightarrow \text{AlarmSystem})$). In our proposal, concrete domains allow to deal with numerical features, which are mixed, in preferences, with non numerical ones.

We note that in the negotiation scenario we model a buyer request, as well as a seller supply, can be split into two parts: one involving issues that have to be necessarily satisfied in order to accept a final agreement, which we call *hard constraints*, and another one involving issues buyer and seller are willing to negotiate on, we call these *soft constraints*. Among *soft constraints* there can be also *fuzzy constraints*, which are preferences involving numerical features. *Fuzzy constraints* are represented in our approach using fuzzy membership functions, see Section 3, therefore while a simple *soft constraint* can or cannot be satisfied, a *fuzzy constraints* can also be satisfied to a “certain degree”. For example, a buyer can state, among soft constraints, that *if a GPS system is mounted on the car she can spend up to 25000 for a sedan*; if the price in the proposed agreement is equal to 25500 we should not simply say that the preference is not satisfied at all, but rather that is satisfied to a certain degree, as will be better described later on (see Section 4).

We note that in our framework it will be possible to model *positive* and *negative* preferences (I would like a car 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-year warranty).

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. Obviously, 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.

The rest of the paper is structured as follows: next section discusses the assumptions we make and the negotiation mechanism we adopt. In Section 3 we illustrate the modeling of issues through our logical language and then we define the multi-issue

negotiation problem and how to compute Pareto agreements. In Section 6 the whole negotiation process is highlighted with the aid of a simple example. Related Work and discussion close the paper.

2 Negotiation Mechanism

We start outlining the scenario and assumptions that characterize the proposed negotiation mechanism. Following [22] we define the *Space of possible deals*, the *Negotiation Protocol* and the *Negotiation Strategy*. The *Space of possible deals* is the set of all possible agreements, in our framework we define an agreement as a model for the theory and the set of *hard constraints* (see Section 4). Furthermore we are not only interested in a *feasible* agreement, but in agreements which are *Pareto efficient*¹. In order to ensure that agents reached agreements which are Pareto efficient a *protocol* and *strategies*, suitable for such a protocol, have to be defined. We adopt a *one-shot* protocol with the intervention of a *mediator* with a proactive behavior. Differently from the classical Single-shot bargaining [20], where one player proposes a deal and the other player may only accept or refuse it [2], in our framework we hypothesize the presence of an electronic mediator, that may automatically explore the negotiation space and discover Pareto-efficient agreements to be proposed to both parties. As pointed out in [21, p.311], usually bargainers are obviously reluctant to disclose their true preferences or utilities to the other party, but they are 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, as they can trust the fairness of the mediator and they have a single possibility to reach the agreement with that counterpart. Thanks to the presence of a mediator we can model a negotiation with *incomplete information*, where agents do not know anything about their counterparts, neither preferences nor worth of them. For what concerns *strategy*, the bargainers reveal their preferences to the mediator and then, once it has computed a solution, they can accept or refuse the agreement proposed to them; if one of them or both refuse the agreement proposed by the mediator the negotiation ends with a *conflict deal*. Bargainers may refuse if they think possible to reach a better agreement looking for another partner or for a different set of bidding rules.

3 Representation of Issues

We divide issues involved in a negotiation in two categories. Some issues may simply express properties that are true or false, like *e.g.*, in an automotive domain, *Sedan*, *DriverInsurance*. We represent such issues as propositional atoms A_1, A_2, \dots from

¹ An agreement is Pareto-efficient if there is no other agreement that will make at least one participant better off without making at least one other participant worse off. If a negotiation outcome is not Pareto-efficient, then there is another outcome that will make at least one participant happier while keeping everyone else at least as happy [9].

a finite set \mathcal{A} . While we represent issues involving numerical features as variables f_1, f_2, \dots , each one taking values in its specific domain D_{f_1}, D_{f_2}, \dots , such as $[0, 96]$ (months) for `month_warranty`, or $[1,000, 50,000]$ (euros), for `price`.

The variables representing numerical features are either involved in *hard constraints* or *soft constraints*. In hard constraints, the variables are always constrained by comparing them to some constant, like `price < 20,000`, or `month_warranty ≥ 60`, and such constraints can be combined into complex propositional requirements – also involving propositional issues – e.g., `Sedan ∧ (price ≤ 25,000) ∧ (deliverytime < 30)` (representing a sedan, costing no more than 25,000 euros, delivered in less than 30 days), or `AlarmSystem ∧ (price > 26,000)` (expressing the seller’s requirement “if you want an alarm system mounted you’ll have to spend more than 26,000 euros”). Vice-versa when numerical features are involved in *soft constraints*, also called *fuzzy constraints*, the variables representing numerical features are constrained by so-called fuzzy membership functions, as shown in Figure 1. For instance, `price ls(18000, 22000)` dictates that given a price it returns the degree of truth to which the constraint is satisfied. Essentially, `price ls(18000, 22000)` states that if the price is no higher than 18000 then the constraint is definitely satisfied, while if the price is higher than 22000 then the constraint is definitely not satisfied. In between 18000 and 22000, we use linear interpolation, given a price, to evaluate the satisfaction degree of the constraint.

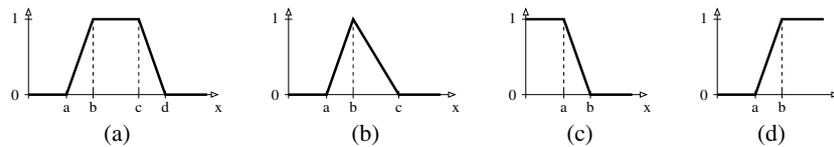


Fig. 1. (a) Trapezoidal function $trz(a, b, c, d)$, (b) triangular function $tri(a, b, c)$, (c) left shoulder function $ls(a, b)$, and (d) right shoulder function $rs(a, b)$

We now give precise definitions for the above intuitions, borrowing from a previous formalization of so-called *concrete domains* [1] from Knowledge Representation languages.

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

For our numerical features, predicates will always be of the form $\geq k, \leq k, = k$ (k is a value in D) if *hard constraints*, e.g., `price ≤ 26000`, or of the form $ls(a, b), rs(a, b), tri(a, b, c), trz(a, b, c, d)$ if *soft constraints*, e.g., `deliverytime ls(30, 40)`. The former predicate expresses a strict constraint (*hard*), *my budget is such that I cannot spend more than 26000 euros*, the latter one expresses a *soft constraint* since *I would prefer not to wait more than one month for a car, but I can be disposal to wait until 40 days, furthermore in such a case I will be less satisfied*.

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 $c \in \{\geq, \leq, =\}$ then $(f \ c \ k)$ is a formula in $\mathcal{P}(\mathcal{N})$
3. if $\langle f, D_f \rangle \in F$ and c is of the form $ls(a, b)$, $rs(a, b)$, $tri(a, b, c)$, $trz(a, b, c, d)$ then $(f \ c)$ is a formula in $\mathcal{P}(\mathcal{N})$
4. if ψ and φ are formulas in $\mathcal{P}(\mathcal{N})$ and $n \in [0, 1]$ then so are $\neg\psi$, $\psi \wedge \varphi$, $\psi \vee \varphi$, $\psi \Rightarrow \varphi$ and $n \cdot \psi$. We use $\psi \Leftrightarrow \varphi$ in place of $(\psi \Rightarrow \varphi) \wedge (\varphi \Rightarrow \psi)$.

In order to define a formal semantics of $\mathcal{P}(\mathcal{N})$ formulas, we consider interpretation functions \mathcal{I} that map propositional atoms into $[0, 1]$, feature names into values in their domain, and define the truth of composite formulas as follows. Let $\neg: [0, 1] \rightarrow [0, 1]$, $\wedge: [0, 1] \times [0, 1] \rightarrow [0, 1]$, $\vee: [0, 1] \times [0, 1] \rightarrow [0, 1]$ and $\Rightarrow: [0, 1] \times [0, 1] \rightarrow [0, 1]$ be functions to be used to interpret negation, conjunction (a t-norm), disjunction (s-norm) and implication, respectively [8]. Note that we use the the same symbol, e.g., \neg for both to identify the negation of a formula as well as the negation of a truth value. The meaning will always be determined by the signature of the symbol. The choice of them is not arbitrary, but is restricted, as usual, by the conditions described in Figure 2. Some specific choices instead are described in Figure 3, while Figure 4 highlights some salient properties of them. It is important to note that we can never enforce that a choice of the interpretation of the connectors satisfies all properties of Figure 4, because then the logic will collapse to classical boolean propositional logic.

Axiom Name	T-norm	S-norm
Tautology / Contradiction	$a \wedge 0 = 0$	$a \vee 1 = 1$
Identity	$a \wedge 1 = a$	$a \vee 0 = a$
Commutativity	$a \wedge b = b \wedge a$	$a \vee b = b \vee a$
Associativity	$(a \wedge b) \wedge c = a \wedge (b \wedge c)$	$(a \vee b) \vee c = a \vee (b \vee c)$
Monotonicity	if $b \leq c$, then $a \wedge b \leq a \wedge c$	if $b \leq c$, then $a \vee b \leq a \vee c$

Axiom Name	Implication Function	Negation Function
Tautology / Contradiction	$0 \rightarrow b = 1$ $a \rightarrow 1 = 1$	$\neg 0 = 1, \neg 1 = 0$
Antitonicity	if $a \leq b$, then $a \rightarrow c \geq b \rightarrow c$	if $a \leq b$, then $\neg a \geq \neg b$
Monotonicity	if $b \leq c$, then $a \rightarrow b \leq a \rightarrow c$	

Usually,

$$a \rightarrow b = \sup\{c: a \wedge c \leq b\}$$

is used and is called *r-implication* and depends on the t-norm only.

Fig. 2. Conditions on norms

Definition 3 (Interpretation and models). An interpretation \mathcal{I} for $\mathcal{P}(\mathcal{N})$ is a function (denoted as a superscript \mathcal{I} on its argument) that maps each atom in \mathcal{A} into a truth value $A^{\mathcal{I}} \in [0, 1]$, each feature name f into a value $f^{\mathcal{I}} \in D_f$, and assigns truth values in $[0, 1]$ to formulas as follows:

	Lukasiewicz Logic	Gödel Logic	Product Logic	Zadeh
$\neg x$	$1 - x$	if $x = 0$ then 1 else 0	if $x = 0$ then 1 else 0	$1 - x$
$x \wedge y$	$\max(x + y - 1, 0)$	$\min(x, y)$	$x \cdot y$	$\min(x, y)$
$x \vee y$	$\min(x + y, 1)$	$\max(x, y)$	$x + y - x \cdot y$	$\max(x, y)$
$x \Rightarrow y$	if $x \leq y$ then 1 else $1 - x + y$	if $x \leq y$ then 1 else y	if $x \leq y$ then 1 else y/x	$\max(1 - x, y)$

Fig. 3. Typical norms

Property	Lukasiewicz Logic	Gödel Logic	Product Logic	Zadeh Logic
$x \wedge \neg x = 0$	•	•	•	
$x \vee \neg x = 1$	•			
$x \wedge x = x$		•		•
$x \vee x = x$		•		•
$\neg \neg x = x$	•			•
$x \rightarrow y = \neg x \vee y$	•			•
$\neg(x \rightarrow y) = x \wedge \neg y$	•			•
$\neg(x \wedge y) = \neg x \vee \neg y$	•	•	•	•
$\neg(x \vee y) = \neg x \wedge \neg y$	•	•	•	•

Fig. 4. Some properties of norms

- for hard constraints, $(f \ c \ k)^{\mathcal{I}} = 1$ iff the relation $f^{\mathcal{I}} \ c \ k$ is true in D_f , $(f \ c \ k)^{\mathcal{I}} = 0$ otherwise
- for soft constraints, $(f \ c)^{\mathcal{I}} = c(f^{\mathcal{I}})$, i.e., the result of evaluating the fuzzy membership function c on the value $f^{\mathcal{I}}$
- $(\neg\psi)^{\mathcal{I}} = \neg\psi^{\mathcal{I}}$, $(\psi \wedge \varphi)^{\mathcal{I}} = \psi^{\mathcal{I}} \wedge \varphi^{\mathcal{I}}$, $(\psi \vee \varphi)^{\mathcal{I}} = \psi^{\mathcal{I}} \vee \varphi^{\mathcal{I}}$, $(n \cdot \psi)^{\mathcal{I}} = n \cdot \psi^{\mathcal{I}}$ and $(\psi \Rightarrow \varphi)^{\mathcal{I}} = \psi^{\mathcal{I}} \Rightarrow \varphi^{\mathcal{I}}$.

Given a formula φ in $\mathcal{P}(\mathcal{N})$, we denote with $\mathcal{I} \models \varphi$ the fact that \mathcal{I} assigns 1 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.

Example 1. Let $\mathcal{A} = \{\text{PassengerCar}, \text{Diesel}\}$ be a set of propositional atoms, $D_{\text{price}} = \{0, \dots, 60000\}$ and $D_{\text{year_warranty}} = \{0, 1, \dots, 5\}$ be two concrete domains for the features price, year_warranty, respectively. A model \mathcal{I} for both formulas:

$$\left\{ \begin{array}{l} \text{PassengerCar} \wedge (\text{Diesel} \Rightarrow (\text{year_warranty} \geq 1)), \\ (\text{price} \leq 5,000) \end{array} \right\}$$

is $\text{PassengerCar}^{\mathcal{I}} = 1$, $\text{Diesel}^{\mathcal{I}} = 0$, $\text{year_warranty}^{\mathcal{I}} = 0$, $\text{price}^{\mathcal{I}} = 4,500$.

Definition 4 (Łukasiewicz Logic as intended semantics). For the sake of our purpose, for the remainder of the paper we will use Łukasiewicz Logic as the specific interpretation of the connectives.

The reason for this choice is due to the nice logical and computational properties of Łukasiewicz Logic. Furthermore, note that $x \wedge_G y = \min(x, y)$ and $x \vee_G y = \max(x, y)$ can also be defined in it by means of $x \wedge (x \rightarrow y)$ and $\neg(\neg x \wedge_G \neg y)$, respectively.

Given a set of formulas \mathcal{T} in $\mathcal{P}(\mathcal{N})$ (representing an ontology), we denote with $\mathcal{I} \models \mathcal{T}$ 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 $\varphi^{\mathcal{I}} = 1$ in all models of \mathcal{T} . We denote with $\mathcal{M}_{\mathcal{T}}$, the set of all models for \mathcal{T} , and omit the subscript when no confusion arises. We also denote with

$$|\phi|_{\mathcal{T}} = \inf_{\mathcal{I} \models \mathcal{T}} \phi^{\mathcal{I}},$$

the lower bound of ϕ 's truth degree over all models \mathcal{T} , while with

$$\langle \phi \rangle_{\mathcal{T}} = \sup_{\mathcal{I} \models \mathcal{T}} \phi^{\mathcal{I}},$$

we denote the maximal truth degree of ϕ over all models \mathcal{T} .

Example 2. Consider \mathcal{T} as the set of formulae

$$\left\{ \begin{array}{l} \text{PassengerCar} \wedge (\text{Diesel} \Rightarrow (\text{year_warranty} \geq 1)), \\ (\text{price} \leq 5,000), \end{array} \right\}$$

Then, $|\text{price rs}(3000, 6000)|_{\mathcal{T}} = 0$, while $\langle \text{price rs}(3000, 6000) \rangle_{\mathcal{T}} = 1/3$.

4 Multi Issue Bilateral Negotiation in $\mathcal{P}(\mathcal{N})$

Following [18], 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 – of $\mathcal{P}(\mathcal{N})$ formulas.

As we have stated before, in a typical bilateral negotiation scenario, issues within both the buyer's request and the seller's offer can be split into *hard constraints* and *soft constraints*. In the rest of the paper we call *hard constraints*, issues that have to be necessarily satisfied in the final agreement, *demand/supply*. *Soft constraints*, denoting issues they are willing to negotiate on, *preferences*.

Example 3. Suppose to have a buyer's request like: "I am searching for a Passenger Car provided with Diesel engine. I need the car as soon as possible, and I can not wait more than one month. Preferably I would like to pay less than 22,000 € furthermore I am willing to pay up to 24,000 € if warranty is greater than 160000 km. (I won't pay more than 27,000 €)". In this example it is possible to distinguish between *hard constraints* (**demand**) and *soft constraints* (**preferences**).

demand: I want a Passenger Car provided with Diesel feeding. I can not wait more than one month. I won't pay more than 27,000 € .

preferences: I would like to pay less than 22,000 € furthermore I am willing to pay up to 24,000 € if warranty is greater than 160000 km.

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. On the other hand, if seller and buyer have set *hard constraints* that are in conflict with each other, that is $\mathcal{T} \cup \{\sigma, \beta\}$ has no models, then the negotiation ends immediately because, it is impossible to reach an agreement. If the participants are willing to avoid the *conflict deal* [22], and continue the negotiation, it will be necessary they revise their *hard constraints*.

In the negotiation process both the buyer and the seller express some preferences on attributes, or their combination in terms of weighted formulae. While there may be many different ways to define preferences, for the sake of our work we define:

Definition 6 (Preferences). *The buyer's negotiation preference \mathcal{B} is a formula of the form $n_1 \cdot \beta_1 \vee \dots \vee n_k \cdot \beta_k$, where each β_i represents the subject of a buyer's preference, and n_i is the utility associated to it. We assume that $\sum_i n_i = 1$. Analogously, the seller's negotiation preference \mathcal{S} is a formula of the form $m_1 \cdot \sigma_1 \vee \dots \vee m_h \cdot \sigma_h$, where each σ_i represents the subject of a seller's preference, and m_i is the utility associated to it. We assume that $\sum_i m_i = 1$.*

Note that a formula of the form $n_1 \cdot \beta_1 \vee \dots \vee n_k \cdot \beta_k$ is under Łukasiewicz logic the weighted sum of the degree of truth of the β_i .

For instance, the Buyer's request in Example 3 is formalized as:

$$\begin{aligned} \beta &= \text{PassengerCar} \wedge \text{Diesel} \wedge (\text{price} \leq 27,000) \wedge \\ &\quad (\text{deliverytime} \leq 30) \\ \beta_1 &= (\text{price}, \text{ls}(22000, 25000)) \\ \beta_2 &= (\text{km_warranty}, \text{rs}(140000, 160000)) \Rightarrow (\text{price}, \text{ls}(24000, 27000)) \end{aligned}$$

As usual, both agents' utilities are normalized to 1 to eliminate outliers, and make them comparable. Since we assumed that utilities are additive, the utility function, that we call *preference utility*, is just a weighted 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 preference, and $\mathcal{M}_{\mathcal{T} \cup \{\alpha, \beta\}}$ be their agreements set. The preference utility of an agreement $\mathcal{I} \in \mathcal{M}_{\mathcal{T} \cup \{\alpha, \beta\}}$ for a buyer and a seller, respectively, are defined as:*

$$\begin{aligned} u_{\beta, \mathcal{P}(\mathcal{N})}(\mathcal{I}) &\doteq \mathcal{B}^{\mathcal{I}} \\ u_{\sigma, \mathcal{P}(\mathcal{N})}(\mathcal{I}) &\doteq \mathcal{S}^{\mathcal{I}}. \end{aligned}$$

Where $\mathcal{B}^{\mathcal{I}}$, as well as $\mathcal{S}^{\mathcal{I}}$, is a weighted sum of the degree of truth of the β_i under Łukasiewicz logic.

Notice that if one agent *e.g.*, the buyer, does not specify *soft constraints*, but only *hard constraints*, it is as $\beta_1 = \top$ and $\mathcal{B}^{\mathcal{I}} = 1$, which reflects the fact that an agent accepts whatever agreement not in conflict with its *hard constraints*.

From the formulas related to Example 3, we note that while considering numerical features, it is still possible to express hard and soft constraints on them. A *hard constraint* expresses on a numerical feature is surely the **reservation value** [21]. In Example 3 the buyer expresses two reservation values, one on price “*more than 27,000 €*” and the other on delivery time “*less than 1 month*”.

Both buyer and seller can express a reservation values on numerical 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 $r_{\beta, \text{price}}$ and $r_{\sigma, \text{price}}$ for the buyer and the seller respectively, if the buyer expresses $\text{price} \leq r_{\beta, \text{price}}$ and the seller $\text{price} \geq r_{\sigma, \text{price}}$, in case $r_{\sigma, \text{price}} \leq r_{\beta, \text{price}}$ we have $[r_{\sigma, \text{price}}, r_{\beta, \text{price}}]$ as a **Zone Of Possible Agreement** — $ZOPA(\text{price})$, otherwise no agreement is possible [21]. 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 25,000, while the seller minimum allowable price is 20,000, then we can set the two reservation values: $r_{\beta, \text{price}} = 25,000$ and $r_{\sigma, \text{price}} = 20,000$, so the *agreement price* will be in the interval $ZOPA(\text{price}) = [20000, 25000]$.

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 3).

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 $\mathcal{B}^{\mathcal{I}}$ and a disagreement threshold t_{β} , a supply σ and a set of seller’s preferences \mathcal{S} with utility function $\mathcal{S}^{\mathcal{I}}$ and a disagreement threshold t_{σ} , a **Multi-issue Bilateral Negotiation problem (MBN)** is finding a model \mathcal{I} (agreement) such that all the following conditions hold:

$$\mathcal{I} \models \mathcal{T} \cup \{\sigma, \beta\} \quad (1)$$

$$\mathcal{B}^{\mathcal{I}} \geq t_{\beta} \quad (2)$$

$$\mathcal{S}^{\mathcal{I}} \geq t_{\sigma} \quad (3)$$

Note that not every agreement \mathcal{I} is a solution of an MBN, if either $\mathcal{B}^{\mathcal{I}} < t_{\sigma}$ or $\mathcal{S}^{\mathcal{I}} < t_{\beta}$. 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 (1).

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

Among all possible agreements that we can compute, given a theory \mathcal{T} as constraint, we are interested in agreements that are Pareto-efficient and *fair* for both the participants, in order to make them equally, and as much as possible, satisfied. Formally, let ψ, φ be two formulae, and let $*$ be a connective interpreted as the product t-norm (see Figure 2). Then a *Pareto agreement* is defined as follows. Let \mathcal{T} be an ontology, let β be the buyer's demand, let σ be the seller's supply, let \mathcal{B} and \mathcal{S} be respectively the buyer's and seller's preferences. Let r be a rational in $[0, 1]$ and let us assume that we admit formulae of the form $r \Rightarrow \psi$ and $\psi \Rightarrow r$. We define $\mathcal{I} \models r \Rightarrow \psi$ iff $\psi^{\mathcal{I}} \geq r$ (the truth degree of ψ is equal or greater than r), while $\mathcal{I} \models \psi \Rightarrow r$ iff $\psi^{\mathcal{I}} \leq r$ (the truth degree of ψ is equal or less than r). Furthermore, let $\bar{\mathcal{T}}$ be the ontology

$$\bar{\mathcal{T}} = \mathcal{T} \cup \{\beta, \sigma\} \cup \{\text{buy} \Leftrightarrow \mathcal{B}, \text{sell} \Leftrightarrow \mathcal{S}, t_{\beta} \Rightarrow \mathcal{B}, t_{\sigma} \Rightarrow \mathcal{S}\}.$$

Then a *Pareto agreement* is an interpretation $\bar{\mathcal{I}}$ such that

$$\bar{\mathcal{I}} = \arg \max_{\mathcal{I} \models \bar{\mathcal{T}}} (\text{buy})^{\mathcal{I}} * (\text{sell})^{\mathcal{I}}.$$

It is not difficult to see that Pareto agreements can be also characterized as the set of all models \mathcal{I} of $\bar{\mathcal{T}}$, such that

$$(\text{buy})^{\mathcal{I}} * (\text{sell})^{\mathcal{I}} = \langle \text{buy} * \text{sell} \rangle_{\bar{\mathcal{T}}}.$$

The value $\langle \text{buy} * \text{sell} \rangle_{\bar{\mathcal{T}}}$ is called the *Pareto agreement value*. It is easily verified that while the Pareto agreement value is unique, there may be many different Pareto agreements (*i.e.*, interpretations) with the same Pareto agreement value.

Computing a Pareto agreement is in fact easy, using Quadratic Mixed Integer Linear Programming. We start with replacing any formula ψ in $\bar{\mathcal{T}}$ with the formula $1 \Rightarrow \psi$ (ψ is true to degree 1). Now, it is not difficult to see that we can recursively associate to any formula $1 \Rightarrow \psi \in \bar{\mathcal{T}}$ a set of linear in-equations $Eq(1 \Rightarrow \psi)$, by assigning to any propositional letter p a variable x_p (see, *e.g.*, [8,14], see also Figure 5).

Then, we solve the Quadratic Mixed Integer Linear Programming problem²

$$\begin{aligned} \max x_{\text{buy}} \cdot x_{\text{sell}} \\ \bigcup_{1 \Rightarrow \psi \in \bar{\mathcal{T}}} Eq(1 \Rightarrow \psi) \end{aligned}$$

Any assignment to the variables in the optimal solution corresponds to a Pareto agreement.

² Note that the fuzzy membership functions in Figure 1 are combination of linear functions and, thus, can be mapped into a set of linear in-equations as well, which we do not report here, – see [10,23].

$$\begin{array}{ll}
\varphi \vee \psi & \mapsto \neg(\neg\varphi \wedge \neg\psi) \\
r \rightarrow p & \mapsto x_p \geq r, x_p \in [0, 1] \\
p \rightarrow r & \mapsto x_p \leq r, x_p \in [0, 1] \\
r \rightarrow \neg\varphi & \mapsto \varphi \rightarrow (1 - r) \\
\neg\varphi \rightarrow r & \mapsto (1 - r) \rightarrow \varphi \\
r \rightarrow (\varphi \wedge \psi) & \mapsto x_1 \rightarrow \varphi, x_2 \rightarrow \psi, y \leq 1 - r, x_i \leq 1 - y, x_1 + x_2 = r + 1 - y, \\
& \quad x_i \in [0, 1], y \in \{0, 1\} \\
(\varphi \wedge \psi) \rightarrow r & \mapsto x_1 \rightarrow \neg\varphi, x_2 \rightarrow \neg\psi, x_1 + x_2 = 1 - r, x_i \in [0, 1] \\
r \rightarrow (\varphi \rightarrow \psi) & \mapsto \varphi \rightarrow x_1, x_2 \rightarrow \psi, r + x_1 - x_2 = 1, x_i \in [0, 1] \\
(\varphi \rightarrow \psi) \rightarrow r & \mapsto x_1 \rightarrow \varphi, \psi \rightarrow x_2, y - r \leq 0, y + x_1 \leq 1, y \leq x_2, y + r + x_1 - x_2 = 1, \\
& \quad x_i \in [0, 1], y \in \{0, 1\} \\
r \rightarrow n\varphi & \mapsto r/n \rightarrow \varphi \\
n\varphi \rightarrow r & \mapsto \varphi \rightarrow r/n
\end{array}$$

Fig. 5. Transformation rules, where φ, ψ are formulae, p is a propositional letter and r, n are rationals in $[0, 1]$

6 The Bargaining Process

Summing up, the negotiation process covers the following steps:

Pre-negotiation Phase. The buyer defines *hard constraints* β and preferences (*soft constraints*) \mathcal{B} with corresponding weights for each preference n_1, n_2, \dots, n_k , as well as the threshold t_β , and similarly the seller σ, \mathcal{S}, m_h and t_σ . Here we are not interested in how to compute t_β, t_σ, n_i and m_i ; 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) [17]. After the previous elements have been set, both agents inform the mediator about these specifications and the theory \mathcal{T} they refer to. Notice that for numerical features 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 $f \geq r_f$.

Negotiation-Core phase. Once the mediator have collected the sets of *hard* and *soft constraints*, the theory \mathcal{T} they refer to, the weights n_i and m_i and the thresholds t_β, t_σ from the bargainers, it exploits such an information in order to compute Pareto agreements (see Section 5). With respect to the set of constraints represented by the theory \mathcal{T} , the *hard constraints* β, σ and the thresholds t_β, t_σ the mediator solves an optimization problem, trying to maximizing the utility of both buyer and seller, *i.e.*, trying to maximizing the number of satisfied preferences of both players in the final agreement. The returned solution to the optimization problem is the agreement proposed to the buyer and the seller. The solution proposed by the mediator is not only a Pareto-optimal one, as it is also a *fair* solution [21]. In fact, among all the Pareto-optimal solutions we take the one maximizing the product of utilities of the players.

From this point on, it is a *take-it-or-leave-it* offer: the bargainers can either accept or reject the proposed agreement [9]. If both players accept then an agreement is reached, otherwise, the negotiation ends in a *conflict deal*.

Let us present a tiny example in order to better clarify the approach. Given the toy ontology \mathcal{T} ,

$$\mathcal{T} = \begin{cases} \text{Sedan} \Rightarrow \text{PassengerCar} \\ \text{ExternalColorBlack} \Rightarrow \neg \text{ExternalColorGray} \\ \text{SatelliteAlarm} \Rightarrow \text{AlarmSystem} \\ \text{InsurancePlus} \Leftrightarrow \text{DriverInsurance} \wedge \text{TheftInsurance} \\ \text{NavigatorPack} \Leftrightarrow \text{SatelliteAlarm} \wedge \text{GPS_system} \end{cases}$$

The buyer and the seller specify their *hard* and *soft constraints*. For each numerical feature involved in *soft constraints* we associate a fuzzy function. If the bargainer has stated a reservation value on that feature, it will be used in the definition of the fuzzy function, otherwise a default value will be used.

$$\begin{aligned} \beta &= \text{PassengerCar} \wedge \text{price} \leq 26000 \\ \beta_1 &= (\text{price}, \text{rs}(23000, 26000)) \wedge \text{AlarmSystem} \\ \beta_2 &= \text{DriverInsurance} \wedge (\text{TheftInsurance} \vee \text{FireInsurance}) \\ \beta_3 &= \text{AirConditioning} \wedge (\text{ExternalColorBlack} \vee \text{ExternalColorGray}) \\ \beta_4 &= (\text{price}, \text{ls}(22000, 24000)) \vee (\text{km_warranty}, \text{rs}(140000, 160000)) \\ \mathcal{B} &= 0.1 \cdot \beta_1 \vee 0.2 \cdot \beta_2 \vee 0.3 \cdot \beta_3 \vee 0.4 \cdot \beta_4 \\ t_\beta &= 0.7 \end{aligned}$$

$$\begin{aligned} \sigma &= \text{Sedan} \wedge \text{price} \geq 24000 \\ \sigma_1 &= \text{NavigatorPack} \wedge (\text{price}, \text{rs}(24000, 26000)) \\ \sigma_2 &= \text{InsurancePlus} \\ \sigma_3 &= (\text{km_warranty}, \text{ls}(150000, 170000)) \\ \sigma_4 &= \text{ExternalColorBlack} \wedge \text{AirConditioning} \\ \mathcal{S} &= 0.3 \cdot \sigma_1 \vee 0.1 \cdot \sigma_2 \vee 0.4 \cdot \sigma_3 \vee 0.2 \cdot \sigma_4 \\ t_\sigma &= 0.6 \end{aligned}$$

Let

$$\bar{\mathcal{T}} = \mathcal{T} \cup \{\beta, \sigma\} \cup \{\text{buy} \Leftrightarrow \mathcal{B}, \text{sell} \Leftrightarrow \mathcal{S}, t_\beta \Rightarrow \mathcal{B}, t_\sigma \Rightarrow \mathcal{S}\}$$

Then, by definition, an interpretation $\bar{\mathcal{I}}$ such that

$$\bar{\mathcal{I}} = \arg \max_{\mathcal{I} \models \bar{\mathcal{T}}} (\text{buy})^{\mathcal{I}} * (\text{sell})^{\mathcal{I}} .$$

is a *Pareto agreement* which is equivalent to solve

$$\begin{aligned} \max x_{\text{buy}} \cdot x_{\text{sell}} \\ \bigcup_{1 \Rightarrow \psi \in \bar{\mathcal{T}}} E_{1 \Rightarrow \psi} . \end{aligned}$$

It turns out that an optimal $\bar{\mathcal{I}}$ is such that

$$\max x_{\text{buy}} \cdot x_{\text{sell}} = 0.933 \cdot 0.7 = 0.651 ,$$

that is, $0.651 = \text{buy}^{\bar{\mathcal{I}}} * \text{sell}^{\bar{\mathcal{I}}} = 0.933 \cdot 0.7$. Furthermore, all $\beta_i^{\bar{\mathcal{I}}} = 1$ except $\beta_1^{\bar{\mathcal{I}}} = 0.333$ and $\sigma_i^{\bar{\mathcal{I}}} = 1$, except $\sigma_1^{\bar{\mathcal{I}}} = 0.0$.

In particular, the final agreement is:

$\text{Sedan}^{\bar{x}} = 1.0,$
 $\text{PassengerCar}^{\bar{x}} = 1.0,$
 $\text{InsurancePlus}^{\bar{x}} = 1.0,$
 $\text{AlarmSystem}^{\bar{x}} = 1.0,$
 $\text{DriverInsurance}^{\bar{x}} = 1.0,$
 $\text{AirConditioning}^{\bar{x}} = 1.0,$
 $\text{NavigatorPack}^{\bar{x}} = 1.0,$
 $(\text{km_warranty}, \text{ls}(150000, 170000))^{\bar{x}} = 0.5, \text{ i.e., } \text{km_warranty}^{\bar{x}} = 160000,$
 $(\text{price}, \text{ls}(23000, 26000))^{\bar{x}} = 0.33, \text{ i.e., } \text{price}^{\bar{x}} = 24000,$
 $\text{TheftInsurance}^{\bar{x}} = 1.0,$
 $\text{FireInsurance}^{\bar{x}} = 1.0,$
 $\text{ExternalColorBlack}^{\bar{x}} = 1.0,$
 $\text{ExternalColorGray}^{\bar{x}} = 0.0.$

Notice that $\beta_1^{\bar{x}} = 0.333$, as the preference is not *fully* satisfied — the price is equal to 24000. Furthermore, thanks to its fuzzy representation, it is possible to say that it is satisfied with a *certain degree*.

7 Related Work and Discussion

Automated bilateral negotiation has been widely investigated, both in artificial intelligence and in microeconomics research communities, so this section is necessarily far from complete.

AI-oriented research has usually focused on automated negotiation among agents, and on designing high-level protocols for agent interaction [13]. 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 [5]. Distributed approaches do not allow the presence of a mediator because – as stated in [12, 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. As stated in [15], negotiation mechanisms often involve the presence of a mediator³, which collects information from bargainers and exploits them in order to propose an efficient negotiation outcome. Various recent proposals adopt a mediator, including [6,11,7]. In

³ 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 [15].

[6] 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. In [11] a mediated-negotiation approach is proposed 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.

Several recent logic-based approaches to negotiation are based on propositional logic. In [3], Weighted Propositional Formulas (WPF) are used 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. Utility functions expressed through WPF are classified in [4] 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.

The work presented in [27] adopts 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, and we introduce a fuzzy propositional logic with concrete domains. Finally we devised a *protocol* which the agents should adhere to while negotiating; in contrast, in [27] a game-theoretic approach is taken, presenting no protocol at all, since communication between agents is not considered.

We borrow from [26] the definition of agreement as a model for a set of formulas from both agents. However, in [26] only multiple-rounds protocols are studied, 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 [19], where a basic propositional logic framework endowed of a logical theory was proposed. In [18] the approach was extended and generalized and complexity issues were discussed. In this paper we further extended the framework, introducing the extended logic $\mathcal{P}(\mathcal{N})$, thus effectively handling numerical features involved in *fuzzy constraints*, and showed we are able to compute Pareto-efficient agreements, by solving an optimization problem and adopting a one-shot negotiation protocol. 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 $\mathcal{P}(\mathcal{N})$, which allows 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 that do not immediately appear at the syntactic level. Moreover, thanks to fuzzy representation it has been possible to model *fuzzy constraints* on numerical features. Exploiting a mediator the proposed approach allows to deal with 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.

In the near future we plan to extend the approach using more expressive logics, namely, Fuzzy Description Logics [24], 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.

References

1. Baader, F., Hanschke, P.: A schema for integrating concrete domains into concept languages. In: Proc. of IJCAI 1991, pp. 452–457 (1991)
2. Binmore, K.: Fun and Games. A Text on Game Theory. D.C. Heath and Company (1992)
3. Bouveret, S., Lemaitre, M., Fargier, H., Lang, J.: Allocation of indivisible goods: A general model and some complexity results. In: Proc. of AAMAS 2005, pp. 1309–1310 (2005)
4. Chevaleyre, Y., Endriss, U., Lang, J.: Expressive power of weighted propositional formulas for cardinal preference modeling. In: Proc. of KR 2006, pp. 145–152 (2006)
5. Chevaleyre, Y., Endriss, U., Lang, J., Maudet, N.: Negotiating over small bundles of resources. In: Proc. of AAMAS 2005, pp. 296–302 (2005)
6. Fatima, S., Wooldridge, M., Jennings, N.R.: Optimal agendas for multi-issue negotiation. In: Proc. of AAMAS 2003, pp. 129–136 (2003)
7. Gatti, N., Amigoni, F.: A decentralized bargaining protocol on dependent continuous multi-issue for approximate pareto optimal outcomes. In: Proc. of AAMAS 2005, pp. 1213–1214 (2005)
8. Hájek, P.: Metamathematics of Fuzzy Logic. Kluwer, Dordrecht (1998)
9. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Wooldridge, M.J., Sierra, C.: Automated negotiation: Prospects, methods and challenges. Int. J. of Group Decision and Negotiation 10(2), 199–215 (2001)
10. Jeroslow, R.G.: Logic-based Decision Support. Mixed Integer Model Formulation. Elsevier, Amsterdam (1989)
11. Klein, M., Faratin, P., Sayama, H., Bar-Yam, Y.: Negotiating complex contracts. In: Falcone, R., Barber, S., Korba, L., Singh, M.P. (eds.) AAMAS 2002. LNCS (LNAI), vol. 2631, pp. 753–757. Springer, Heidelberg (2003)
12. Kraus, S.: Strategic Negotiation in Multiagent Environments. The MIT Press, Cambridge (2001)
13. Lomuscio, A.R., Wooldridge, M., Jennings, N.R.: A classification scheme for negotiation in electronic commerce. Int. Journal of Group Decision and Negotiation 12(1), 31–56 (2003)
14. Lukasiewicz, T., Straccia, U.: Tutorial: Managing uncertainty and vagueness in semantic web languages. In: AAAI 2007. Twenty-Second Conference on Artificial Intelligence (2007)
15. MacKie-Mason, J.K., Wellman, M.P.: Automated markets and trading agents. In: Handbook of Computational Economics. North-Holland, Amsterdam (2006)
16. Parsons, S., Sierra, C., Jennings, N.: Agents that reason and negotiate by arguing. Journal of Logic and Computation 8(3), 261–292 (1998)

17. Pomerol, J.C., Barba-Romero, S.: Multicriterion Decision Making in Management. In: Kluwer Series in Operation Research. Kluwer Academic, Dordrecht (2000)
18. Ragone, A., Di Noia, T., Di Sciascio, E., Donini, F.M.: A logic-based framework to compute pareto agreements in one-shot bilateral negotiation. In: Proc. of ECAI 2006, pp. 230–234 (2006)
19. Ragone, A., Di Noia, T., Di Sciascio, E., Donini, F.M.: Propositional-logic approach to one-shot multi issue bilateral negotiation. ACM SIGecom Exchanges 5(5), 11–21 (2006)
20. Raiffa, H.: The Art and Science of Negotiation. Harvard University Press, Cambridge (1982)
21. Raiffa, H., Richardson, J., Metcalfe, D.: Negotiation Analysis - The Science and Art of Collaborative Decision Making. The Belknap Press of Harvard University Press, Cambridge (2002)
22. Rosenschein, J.S., Zlotkin, G.: Rules of Encounter. MIT Press, Cambridge (1994)
23. Straccia, U.: Description logics with fuzzy concrete domains. In: Bachus, F., Jaakkola, T. (eds.) UAI 2005. 21st Conference on Uncertainty in Artificial Intelligence, pp. 559–567. AUA Press, Edinburgh, Scotland (2005)
24. Straccia, U.: A fuzzy description logic for the semantic web. In: Sanchez, E. (ed.) Fuzzy Logic and the Semantic Web, Capturing Intelligence, ch. 4, pp. 73–90. Elsevier, Amsterdam (2006)
25. Wellman, M.P.: Online marketplaces. In: Practical Handbook of Internet Computing. CRC Press, Boca Raton (2004)
26. Wooldridge, M., Parsons, S.: Languages for negotiation. In: Proc. of ECAI 2004, pp. 393–400 (2000)
27. Zhang, D., Zhang, Y.: A computational model of logic-based negotiation. In: Proc. of the AAAI 2006, pp. 728–733 (2006)