Extending Propositional Logic with Concrete Domains in Multi-issue Bilateral Negotiation

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 , Viterbo, Italy
donini@unitus.it

Abstract. We present a novel approach to knowledge-based automated one-shot multi-issue bilateral negotiation handling, in a homogeneous setting, both numerical features and non-numerical ones. The framework makes possible to formally represent typical situations in real e-marketplaces such as "*if I spend more than 20000 € for a sedan then I want a navigator pack included*" where both numerical (price) and non-numerical (sedan, navigator pack) issues coexist. To this aim we introduce $\mathcal{P}(\mathcal{N})$, a propositional logic extended with concrete domains, which allows to: model relations among issues (both numerical and not numerical ones) via logical entailment, differently from well-known approaches that describe issues as uncorrelated; represent buyer's request, seller's supply and their respective preferences as formulas endowed with a formal semantics. By modeling preferences as formulas it is hence possible to assign a utility value also to a bundle of issues, which is obviously more realistic than the trivial sum of utilities assigned to single elements in the bundle itself. 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 a multi objective optimization problem.

1 Introduction

Bilateral negotiation between agents is a challenging problem, which finds applications in a number of different scenarios, each one with its own peculiarities and issues. In this work we focus on automated negotiation in e-marketplaces [30]. Clearly, in such domains we do not simply deal with undifferentiated products (commodities as oil, cement, etc.) or stocks, where only price, time or quantity have to be taken into account. In fact also other features have to be considered during the negotiation process. When a potential buyer browses an automobile e-marketplace, she looks for a car fulfilling her needs and/or wishes, so not only the price is important, but also warranty or delivery time, as well as look, model, comfort and so on. In such domains it is harder to model not only the negotiation process, but also the request/offer descriptions, as well as finding the best suitable agreement. Recently, there has been a growing interest toward multi-issue negotiation, also motivated by the idea that richer and expressive descriptions of demand and supply can boost e-marketplaces (see *e.g.*, [29] for a reasonable set of motivations) but –to the best of our knowledge– also in recent literature, issues are usually described as uncorrelated terms, without considering any underlying semantics. Notable exceptions are discussed in Section 8. In our approach we use knowledge representation in two ways: (1) exploiting a logic theory to represent relations among issues and (2) assigning utilities to formulas to represent agents having preferences over different bundles of issues. For what concerns the former, we introduce a logical theory that allows to represent, *e.g.*, through logical implication, that a Ferrari is an Italian car ($\texttt{Ferrari} \Rightarrow \texttt{ItalianMaker}$) or that an Italian car is not a German car ($\texttt{ItalianMaker} \Rightarrow \neg\texttt{GermanMaker}$). Furthermore we can express agent preferences over bundle of issues, *e.g.*, the buyer can state she would like to have a car with navigator pack, where the meaning of navigator pack is in the Theory ($\texttt{NavigatorPack} \Leftrightarrow \texttt{SatelliteAlarm} \wedge \texttt{GPS\_system}$). In this case, the utility assigned to a bundle is obviously not necessarily the sum of utilities assigned to single elements in the bundle itself. Moreover issues are often inter-dependent: the selection of one issue depends on the selection made for other issues: in our framework agents can express conditional preferences as *I would like a car with leather seats if its color is black* ($\texttt{ExternalColorBlack} \Rightarrow \texttt{Leather\_seats}$). In this work we introduce an extended propositional logic, $\mathcal{P}(\mathcal{N})$ enriched with concrete domains, which allows –as it is in the real world– to take into account preferences involving both numerical features and not numerical ones, *e.g.*, the seller can state that if you want a car with a GPS system you have to wait at least one month: ($\texttt{GPS\_system} \Rightarrow \texttt{deliverytime} \geq 31$) as well as preferences can involve only numerical ones: *e.g.*, the buyer can state that she can accept to pay more than 25000€ for a sedan only if there is more than a two years warranty ($\texttt{price} > 25000 \Rightarrow \texttt{year\_warranty} > 2$). Contributions of this paper include: the framework for automated multi-issue bilateral negotiation, the logical language to represent existing relations between issues and preferences as formulas, which is able to handle both numerical features and not numerical ones as correlated issues w.r.t. a logical Theory and the one-shot protocol we adopt, which allows to compute Pareto-efficient agreements, exploiting a mediator that solves a multi objective optimization problem. The rest of the paper is structured as follows: next section discusses the scenario and the assumptions we make; then we illustrate the modeling of issues through our logical language and the negotiation mechanism. Section 4 presents the multi-issue bilateral negotiation problem, Section 5 describes the computation of utilities for numerical fetures. Section 6 shows how to compute Pareto-efficient agreement and Section 7 summarizes the bargaining process. Related work and discussion close the paper.

2 Negotiation Scenario

We start introducing the negotiation mechanism and the assumptions characterizing our framework. So, in accordance with [25], we define: the *Space of possible deals*, the *Negotiation Protocol* and the *Negotiation Strategy*. For what concerns the *Space of possible deals*, since we solve a multi objective optimization problem, possible deals are all the solutions of the problem that satisfy the constraints, even if they do not maximize the objective function (the so called *feasible region* [11]). The *Negotiation Protocol* we adopt is a *one-shot* protocol with the presence of a mediator. Differently from the classical *Single-shot* bargaining [23], 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. Such parties may then accept or refuse them. We recall that, basically, two different approaches to automated negotiation exist: *centralized* and *distributed* ones. In the first ones, agents elicit their preferences and then a mediator, or some central entity, selects the most suitable deal based on them. In the latter ones, 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 [14, 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 system 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 [17], negotiation mechanisms often involve the presence of a mediator [3], which collects information from bargainers and exploit them in order to propose an efficient negotiation outcome. In Section 8 some approaches adopting a centralized approach are described. Although the main target of an agent is reaching a satisfying agreement, this alone it is not enough, since knowing if this agreement is also Pareto-efficient is a matter that cannot be left out. It is fundamental to assess *how hard* is to find Pareto-efficient agreements and check whether a given agreement is also Pareto-efficient. The presence of a trusted third party can help the parties to reach a Pareto-efficient agreement. As pointed out in [24, 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 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. Therefore in our framework we propose a one-shot protocol with the intervention of a *mediator* with a proactive behavior: it suggests to each participant a *fair* Pareto-efficient agreement. 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 another shot, 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 [18].

3 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`,

[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 [17].

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 taking values in its specific domain $D_{f_1}, D_{f_2}, \ldots$, such as $[0, 90]$ (days) for deliverytime, or $[1,000, 20{,}000]$ (euros), for price. The variables representing numerical features are always constrained by comparing them to some constant, like $\texttt{price} < 20{,}000$, or $\texttt{deliverytime} \geq 30$, and such constraints can be combined into complex propositional requirements – also involving propositional issues – *e.g.*, $\texttt{ItalianMaker} \wedge (\texttt{price} \leq 25{,}000) \wedge (\texttt{deliverytime} < 30)$ (representing a car made in Italy, costing no more than 25,000 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* [1] from Knowledge Representation languages.

Definition 1 (Concrete Domains, [1]). *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.

For our numerical features, predicates will always be the binary operators $C(D) = \{\geq, \leq, >, <, =, \neq\}$, whose second argument is a constant in $\Delta_c(D)$[4]. 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 $(\psi \Rightarrow \varphi) \wedge (\varphi \Rightarrow \psi)$.*

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:*

[4] 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.

– $(fck)^{\mathcal{I}} =$ *true iff* $f^{\mathcal{I}} ck$ *is true in* D_f, $(fck)^{\mathcal{I}} =$ *false otherwise*
– $(\neg\psi)^{\mathcal{I}} =$ *true iff* $\psi^{\mathcal{I}} =$ *false*, $(\psi \wedge \varphi)^{\mathcal{I}} =$ *true iff both* $\psi^{\mathcal{I}} =$ *true and* $\varphi^{\mathcal{I}} =$ *true, etc., 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.

Example 1. Let $\mathcal{A} = \{\mathtt{Sedan}, \mathtt{GPL}\}$ be a set of propositional atoms, $D_{\mathtt{price}} = \{0, \ldots, 60000\}$ and $D_{\mathtt{year\_warranty}} = \{0, 1, \ldots, 5\}$ be two concrete domains for the features `price`, `year_warranty`, respectively. A model $\mathcal{I}$ for both formulas:

$$\left\{ \begin{array}{l} \mathtt{Sedan} \wedge (\mathtt{GPL} \Rightarrow (\mathtt{year\_warranty} \geq 1)), \\ (\mathtt{price} \leq 5{,}000) \end{array} \right\}$$

is $\mathtt{Sedan}^{\mathcal{I}} =$ true, $\mathtt{GPL}^{\mathcal{I}} =$ false, $\mathtt{year\_warranty}^{\mathcal{I}} = 0$, $\mathtt{price}^{\mathcal{I}} = 4{,}500$.

Given a set of formulas $\mathcal{T}$ in $\mathcal{P}(\mathcal{N})$ (representing an ontology), we denote *model* for $\mathcal{T}$ as $\mathcal{I} \models \mathcal{T}$. 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 with $\mathcal{M}_{\mathcal{T}} = \{\mathcal{I}_1, \ldots, \mathcal{I}_n\}$, 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. 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 ($\mathtt{deliverytime} \geq 32$), 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*. Let us consider `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 10,500 or 15,000 €; 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 only to $\geq, \leq$.

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

– $s : D \rightarrow D$ *the successor function:* $s(k_i) = k_{i+1} = k_i + \epsilon$
– $p : D \rightarrow D$ *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 $C_m(D)$ to $\{\leq, \geq\}$ using the following transformations:

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

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

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

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

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

Following [21], 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.

In a typical bilateral negotiation scenario, the issues 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 denote issues they are willing to negotiate on – this is what we call *preferences*.

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 12,000 € furthermore I'm willing to pay up to 15,000 € if warranty is greater or equal than 3 years. (I don't want to pay more than 17,000 € 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 17,000 €. I don't want a car with a warranty less than 2 years*
preferences*: Preferably I would like to pay less than 12,000 , furthermore I'm willing to pay up to 15,000 € 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. On the other hand, 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* [25], 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 $u_\beta : \mathcal{B} \rightarrow \Re^+$ 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 $u_\sigma : \mathcal{S} \rightarrow \Re^+$ 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:

$$\begin{aligned}\beta &= \texttt{Sedan} \wedge \texttt{Leather\_seats} \wedge (\texttt{price} \leq 17{,}000) \wedge \\ &\quad (\texttt{year\_warranty} \geq 2) \\ \beta_1 &= (\texttt{price} \leq 12{,}000) \\ \beta_2 &= (\texttt{year\_warranty} \geq 3) \wedge (\texttt{price} \leq 15{,}000)\end{aligned}$$

As usual, both agents' utilities are normalized to 1 to eliminate outliers, and make them comparable. Since we assumed that utilities are additive, the *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 $\mathcal{M}_{\mathcal{T}\cup\{\alpha,\beta\}}$ 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:*

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

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, it is as $\beta_1 = \top$ 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. A strict requirement is surely the ***reservation value*** [24]. In Example 1 the buyer expresses two reservation values, one on price *"more than 17,000 €"* 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 [24]. 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 15,000, while the seller minimum allowable price is 10,000, then we can set the two reservation values: $r_{\beta,\texttt{price}}$ = 15,000 and $r_{\sigma,\texttt{price}}$ = 10,000, 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 $u_\sigma(\mathcal{I}) < t_\sigma$ or $u_\beta(\mathcal{I}) < t_\beta$. 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).

5 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. On the other hand, 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 2. Referring to β, β_1 and β_2 in Example 1 let us suppose to have the offer [5]:

$$\sigma = \texttt{Sedan} \wedge (\texttt{price} \geq 15{,}000) \wedge (\texttt{year\_warranty} \leq 5)$$

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

$$\begin{aligned}\mathcal{I}_1 : \{&\texttt{Sedan}^{\mathcal{I}_1} = \text{true}, \texttt{Leather\_seats}^{\mathcal{I}_1} = \text{true},\\ &\texttt{price}^{\mathcal{I}_1} = 17{,}000, \texttt{year\_warranty}^{\mathcal{I}_1} = 3\}\\ \mathcal{I}_2 : \{&\texttt{Sedan}^{\mathcal{I}_2} = \text{true}, \texttt{Leather\_seats}^{\mathcal{I}_2} = \text{true},\end{aligned}$$

[5] 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.

$$\texttt{price}^{\mathcal{I}_2} = 16{,}000, \texttt{year\_warranty}^{\mathcal{I}_2} = 4\}$$
$$\mathcal{I}_3 : \{\texttt{Sedan}^{\mathcal{I}_3} = \mathsf{true}, \texttt{Leather\_seats}^{\mathcal{I}_3} = \mathsf{true},$$
$$\texttt{price}^{\mathcal{I}_3} = 15{,}000, \texttt{year\_warranty}^{\mathcal{I}_3} = 5\}$$

Looking at the values of numerical features, $\mathcal{I}_1$ is the best agreement from the seller's perspective whilst $\mathcal{I}_3$ is the best from the buyer's one. In fact, the buyer the less he pays, the happier he is and the contrary holds for the seller! The contrary is for the warranty: the buyer is happier if he gets a greater year warranty. On the other hand, $\mathcal{I}_2$ is a good compromise between buyer's and seller's requirements.

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. Since u_f is a utility function, it is normalized to $[0 \ldots, 1]$. Given the pair $\langle f, D_f \rangle$, it must be defined over the domain D_f.
2. From Example 2 we note the buyer is happier as the price decreases whilst the seller is sadder. 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 $r_{\beta,\texttt{price}}$ =17,000 and there is no utility for the seller if the price is less than or equal to $r_{\sigma,\texttt{price}}$ =15,000. 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** *$u_f : D_f \to [0 \ldots, 1]$ 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.

Clearly, the simplest utility functions are the two linear functions:

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

if it monotonically decreases and

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

if it monotonically increases (see Figure 1).

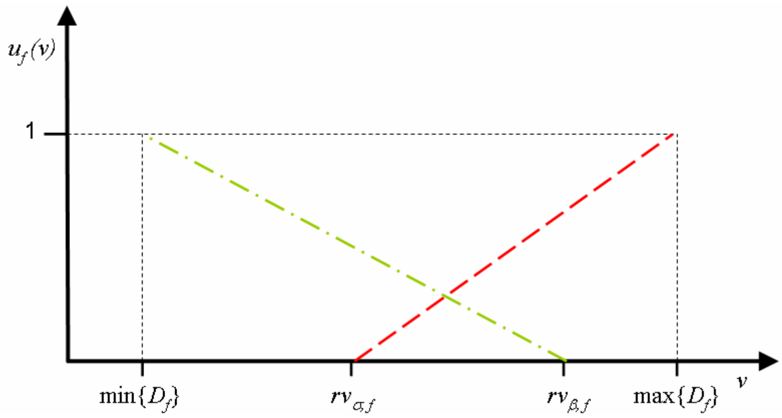

Fig. 1. Linear utility functions

6 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. We now outline how an actual solution can be found solving a multi objective optimization problem.

First of all, let $\{B_1, \dots, B_k, S_1, \dots, S_h\}$ be $k + h$ new propositional atoms, and let $\mathcal{T}' = \mathcal{T} \cup \{B_i \Leftrightarrow \beta_i | i = 1, \dots, k\} \cup \{S_j \Leftrightarrow \sigma_j | j = 1, \dots, h\}$ – that is, every preference in $\mathcal{B} \cup \mathcal{S}$ is equivalent to a new atom in $\mathcal{T}'$.

6.1 Objective functions

Here we define functions to be maximized to find a solution to a multi objective optimization problem. In order to formulate functions to be maximized involving preferences expressed as formulas in $\mathcal{P}(\mathcal{N})$, let $\{b_1, \dots, b_k\}$ the (0,1)-variables one-one with $\{B_1, \dots, B_k\}$ and similarly $\{s_1, \dots, s_h\}$ for $\{S_1, \dots, S_h\}$. The functions representing respectively buyer's and seller's utility over preferences can hence be defined as:

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

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

As highlighted in Section 5, 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_t involved in the negotiation process we have a ***feature utility function*** for the buyer u_{β,f_t} and one for the seller u_{σ,f_t}. For instance, if we consider `price` and the linear function in equations (10) and (11) we likely will have:

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

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

6.2 The Multi Objective Optimization Problem

Given the objective functions to be optimized – the *feature* utility functions and the *preference* utility functions – in order to compute a Pareto agreement we reduce to a multi objective optimization problem (MOP). The functions to be optimized are utility functions both for the buyer and the seller, as we want them equally satisfied.

In addition to the set of functions to maximize (or minimize), in a MOP there are a set of constrained numerical variables. In our setting, we have three different sets of constraints:

1. the (modified) ontology $\mathcal{T}'$ —see the beginning of Section 6
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 4

Notice that the ones involving disagreements thresholds are already linear constraints. In order to model as linear constraints also the ones described in points 1 and 2 of the above enumeration, 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 [16] 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[6]

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

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).

[6] It is well know that such a transformation preserves logical entailment[27].

Example 3. Suppose to have the clause

$$\chi : \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee$$
$$(\texttt{year\_warranty} \geq 3) \vee \neg(\texttt{price} \geq 20{,}500)$$

First of all split the clause in the following two

$$\chi_1 : \overline{A} \vee \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee$$
$$(\texttt{year\_warranty} \geq 3)$$
$$\chi_2 : \neg\overline{A} \vee \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee$$
$$\neg(\texttt{price} \geq 20{,}500)$$

then change the second one in

$$\chi_2 : \neg\overline{A} \vee \texttt{ItalianMaker} \vee \neg\texttt{AirConditioning} \vee$$
$$(\texttt{price} \leq 20{,}000)$$

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

Encoding clauses into linear inequalities Use a modified version of well-known encoding of clauses into linear inequalities (*e.g.*, [19, 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 (0,1)-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 $\frac{1}{k}f$.

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 4. If we consider $\max(D_{\texttt{price}}) = 60{,}000$, continuing Example 3 we have from χ_1 and χ_2 the following inequalities respectively:

$$\overline{a} + i + (1 - a) + \frac{1}{3}\texttt{year\_warranty} \geq 1$$
$$(1 - \overline{a}) + i + (1 - a) + \frac{1}{60{,}000 - 20{,}000}(60{,}000 - \texttt{price}) \geq 1$$

where $\overline{a}$, i, a are (0,1)-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 5. Let us transform the following clause without splitting in the two corresponding ones

$$\overline{\chi} : \texttt{ItalianMaker} \vee (\texttt{year\_warranty} \geq 3) \vee (\texttt{price} \leq 20{,}000)$$

the corresponding inequality is then

$$i + \frac{1}{3}\texttt{year\_warranty} + \frac{1}{60{,}000 - 20{,}000}(60{,}000 - \texttt{price}) \geq 1$$

The interpretation $\{\texttt{year\_warranty} = 2, \texttt{price} = 19{,}500\}$ is not a model for $\overline{\chi}$ while the inequality is satisfied.

7 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,$u_\beta(\beta_i)$ and $u_\sigma(\sigma_j)$; 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) [20]. Both agents inform the mediator about these specifications and the theory $\mathcal{T}$ 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 $f \geq r_f$.

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 6.2 and Section 6.2. Given the preference utility functions $u_{\beta,\mathcal{P}(\mathcal{N})} = \sum_{i=1}^{k} b_i u_\beta(\beta_i)$ and $u_{\sigma,\mathcal{P}(\mathcal{N})} = \sum_{j=1}^{h} s_j u_\sigma(\sigma_j)$, 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 a MOP maximizing the *preference* utility functions $u_{\beta,\mathcal{P}(\mathcal{N})}$, $u_{\sigma,\mathcal{P}(\mathcal{N})}$ and for each feature f involved in the negotiation process also the *feature* utility functions $u_{\beta,f}$ and $u_{\sigma,f}$. The returned solution to the MOP is the agreement proposed to the buyer and the seller. Notice that a solution to a MOP is always Pareto optimal [11], furthermore the solution proposed by the mediator is also a *fair* solution, because among all the Pareto-optimal solutions we take the one maximizing the utilities of both the buyer and the seller (see Sec. 6.1). From this point on, it is a *take-it-or-leave-it* offer, as the participants can either accept or reject the proposed agreement [12]. Let us present a tiny example in order to better clarify the approach. Given the toy ontology in $\mathcal{P}(\mathcal{N})$,

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

the buyer and the seller specify their strict requirements and preferences:

β = $\texttt{Sedan} \wedge (\texttt{price} \leq 30{,}000) \wedge (\texttt{km\_warranty} \geq 120{,}000) \wedge (\texttt{year\_warranty} \geq 4)$
β_1 = $\texttt{GPS\_system} \wedge \texttt{AlarmSystem}$
β_2 = $\texttt{ExternalColorBlack} \Rightarrow \texttt{Leather\_seats}$
β_3 = $(\texttt{km\_warranty} \geq 140{,}000)$
$u_\beta(\beta_1)$ = 0.5

$u_\beta(\beta_2)$ = 0.2
$u_\beta(\beta_3)$ = 0.3
t_β =0.2

σ = $\texttt{Sedan} \wedge (\texttt{price} \geq 20{,}000) \wedge (\texttt{km\_warranty} \leq 160{,}000) \wedge (\texttt{year\_warranty} \leq 6)$
σ_1 = $\texttt{GPS\_system} \Rightarrow (\texttt{price} \geq 28{,}000)$
σ_2 = $(\texttt{km\_warranty} \leq 150{,}000) \vee (\texttt{year\_warranty} \leq 5)$
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

Then the final agreement is:

$$\begin{aligned}\mathcal{I} : \{&\texttt{Sedan}^{\mathcal{I}} = \text{true}, \texttt{ExternalColorGray}^{\mathcal{I}} = \text{true},\\ &\texttt{SatelliteAlarm}^{\mathcal{I}} = \text{true}, \texttt{GPS\_system}^{\mathcal{I}} = \text{true},\\ &\texttt{NavigatorPack}^{\mathcal{I}} = \text{true}, \texttt{AlarmSystem}^{\mathcal{I}} = \text{true},\\ &\texttt{price}^{\mathcal{I}} = 28{,}000, k^{\mathcal{I}} = 160{,}000, \texttt{year\_warranty}^{\mathcal{I}} = 5\}\end{aligned}$$

Here, for the sake of conciseness, we omit propositional atoms interpreted as false.

8 Related Work and discussion

Automated bilateral negotiation among agents has been widely investigated, both in artificial intelligence and in microeconomics research communities, so this section is necessarily far from complete. Several definitions have been proposed in the literature for bilateral negotiation. Rubinstein [26] defined the *Bargaining Problem* as the situation in which "two individuals have before them several possible contractual agreements. Both have interests in reaching agreement but their interests are not entirely identical. What 'will be' the agreed contract, assuming that both parties behave rationally?" In game theory, the bargaining problem has been modeled either as *cooperative* or *non-cooperative* games [10]. AI-oriented research has been more focused on automated negotiation among agents and on designing high-level protocols for agent interaction [15]. Agents can play different roles: act on behalf of buyer or seller, but also play the role of a mediator or facilitator. Approaches exploiting a mediator include among others [8, 13, 9]. In [8] an extended alternating offers protocol was presented, with the presence of a mediator, which improves the utility of both agents. In [13] a mediated-negotiation approach was proposed for complex contracts, where inter dependency among issues is investigated. In [3] the use of propositional logic in multi-issue negotiation was investigated, while in [4] weighted propositional formulas in preference modeling were considered. However, in such papers, no semantic relation among issues is taken into account. In our approach we adopt a logical theory, *i.e.*, an ontology, which allows *e.g.*, to catch inconsistencies between demand and supply or find out a feasible agreement in a bundle, which is fundamental to model an e-marketplace. Self-interested agents negotiating over a set of resources to obtain an optimal allocation of such resources have been studied in [7, 6, 5]. Endriss et al. [7] propose an optimal resource allocation in two different negotiation scenarios: one, with money transfer, determines an allocation with maximal social welfare; the second is a money-free framework, which results in a Pareto outcome. In [5] agents negotiate over small bundles of resources, and a mechanism of resource allocation is investigated, which maximizes the social welfare by means of a sequence of deals involving at most k items each. Both papers [7, 5] extend the framework proposed in [28], which focused on negotiation for (re)allocating tasks among agents. We borrow from [31] the definition of agreement as a model for a set of formulas from both agents. However, in [31] 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 [22], where a basic propositional logic framework endowed of a logical theory was proposed. In [21] 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 handling numerical features, and showed we are able to compute Pareto-efficient agreements, solving a multi objective optimization problem adopting a one-shot negotiation protocol.

References

1. F. Baader and P. Hanschke. A schema for integrating concrete domains into concept languages. In *proc. of IJCAI-91*, pages 452–457, 1991.
2. K. Binmore. *Fun and Games. A Text on Game Theory*. D.C. Heath and Company, 1992.
3. 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.
4. 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.
5. Y. Chevaleyre, U. Endriss, J. Lang, and N. Maudet. Negotiating over small bundles of resources. In *Proc. of AAMAS '05*, pages 296–302, 2005.
6. P. E. Dunne, M. Wooldridge, and M. Laurence. The complexity of contract negotiation. *Artif. Intell.*, 164(1-2):23–46, 2005.
7. U. Endriss, N. Maudet, F. Sadri, and F. Toni. On optimal outcomes of negotiations over resources. In *Proc. of AAMAS '03*, pages 177–184, 2003.
8. S. Fatima, M. Wooldridge, and N.R. Jennings. Optimal agendas for multi-issue negotiation. In *Proc. of AAMAS'03*, pages 129–136, 2003.
9. 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.
10. 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.
11. F. Hillier and G. Lieberman. *Introduction to Operations Research*. McGraw-Hill, 2005.

12. 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.
13. M. Klein, P. Faratin, H. Sayama, and Y. Bar-Yam. Negotiating complex contracts. In *Proc. of AAMAS'02*, pages 753–757, 2002.
14. S. Kraus. *Strategic Negotiation in Multiagent Environments*. The MIT Press, 2001.
15. 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.
16. D.W. Loveland. *Automated theorem proving: A logical basis*. North-Holland, 1978.
17. J.K. MacKie-Mason and M.P. Wellman. Automated markets and trading agents. In *Handbook of Computational Economics*. North-Holland, 2006.
18. A. Muthoo. On the strategic role of outside options in bilateral bargaining. *Operations Research*, 43(2):292–297, 1995.
19. C.H. Papadimitriou and K. Steiglitz. *Combinatorial optimization: algorithms and complexity*. Prentice-Hall, Inc., 1982.
20. J.C. Pomerol and S. Barba-Romero. *Multicriterion Decision Making in Management*. Kluwer Series in Operation Research. Kluwer Academic, 2000.
21. 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.
22. 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.
23. H. Raiffa. *The Art and Science of Negotiation*. Harvard University Press, 1982.
24. 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.
25. J.S. Rosenschein and G. Zlotkin. *Rules of Encounter*. MIT Press, 1994.
26. A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50:97–109, 1982.
27. S. Russell and P. Norvig. *Artificial Intelligence: A Modern Approach*. Pearson Education-Prentice Hall, 2003.
28. T. Sandholm. Contract types for satisficing task allocation: I theoretical results. In *Proceedings of the AAAI Spring Symposium*, 1998.
29. D. Trastour, C. Bartolini, and C. Priest. Semantic Web Support for the Business-to-Business E-Commerce Lifecycle. In *Proc. WWW '02*, pages 89–98, 2002.
30. M.P. Wellman. Online marketplaces. In *Practical Handbook of Internet Computing*. CRC Press, 2004.
31. M. Wooldridge and S. Parsons. Languages for negotiation. In *Proc of ECAI '04*, pages 393–400, 2000.