DL-based alternating-offers protocol for automated 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 logic-based bilateral negotiation in e-commerce systems. We use Description Logics to describe both requests/offers submitted by buyers and sellers, and relations among issues as axioms in a TBox. Moreover, exploiting concept contraction in DLs, we are able to handle *conflicting* information both in goods and services descriptions. We ground the approach in a P2P e-marketplace framework, and introduce a *logic-based* alternating-offers protocol. In such a protocol we exploit both knowledge representation tools and utility theory to find the most suitable agreements.

1 Introduction

We study automated bilateral negotiation in peer-to-peer (P2P) e-marketplaces, where buyers and sellers may want to submit articulate advertisements to find best available counterparts, and price is obviously not the single issue to negotiate on. In such frameworks, using a logic formalism it is possible to recognize that an advertisement for a *Notebook equipped with a Linux operating system* actually fulfills a buyer's request for a *PC having a Unix operating system*. Or, conversely, that a buyer's request for a *Notebook with Wi-Fi adapter* is in conflicting with a seller's supply for a *Notebook with Wired Adapter*. To manage automated bilateral negotiation in such a framework we introduce a novel *logic-based* alternating-offers protocol. The protocol merges both Description Logics formalism and reasoning services, and utility theory, to find the most suitable agreements. To this aim it takes into account existing logical relations between issues in requests and offers and related utilities of agents, expressed through logical formulas.

The roadmap to the remainder of this paper is as follows: next section presents an outline of the whole approach. Then we move on the DL we adopt and related inference services. We show the modeling of advertisements and then the protocol is presented and discussed. Related work and discussion close the paper.

2 Negotiation Scenario

In order to outline the negotiation mechanism we define: the *negotiation protocol*, the *negotiation strategy*[6], the *utility function* of the agents [7]. The assumptions characterizing our negotiation mechanism are:

one-to-many: the negotiation is a one-to-many negotiation, since the buyer's agent will negotiate simultaneously with others *m* different agents – each of them representing a seller, whose offer has been previously stored in the system.

rationality: agents are *rationals*, they behave according to their preferences and try to maximize their utilities [7, p.19] doing in each step the minimum possible concession, *i.e.*, the concession involving the minimum utility loss, see protocol Section 5.

incomplete information: each agent knows its utility function and ignores the opponent disagreement thresholds and utility functions.

conflict deal: disagreement is better than an agreement iff the agent's utility over such an agreement is smaller than disagreement thresholds[3] set by the agent before negotiation starts. Therefore when the agent's utility deriving from accepting an agreement (or going on with the negotiation) and opting out it is the same, it will prefer not to opt out [7]. The protocol we propose is inspired by Rubinstein's alternating-offers one [9]. In that setting an agent starts making an offer to its opponent, who can either accept, make a counter-offer or exit the negotiation. If a counter-offer is made, the negotiation goes on until one of the agent accepts an offer or exits the negotiation. In some cases there is a negotiation *deadline*; if the deadline is reached before one agent has accepted an offer, the negotiation ends in a conflict deal. Our protocol anyway is quite different from that of Rubinstein; actually we consider *multi-issue negotiation*: buyer and seller do not negotiate on a single item or on a single bundle of items, but on many issues, which are related with each other through an ontology; such issues may also characterize a more complex item (*e.g.*, in the computer domain a notebook equipped with Wi-Fi adapter and DVD recorder). Differently from many alternating-offers protocols we do not consider a time deadline.

The protocol is sorted out by a finite set of steps[4]: the negotiation always terminates because either the agreement has been reached or because one agent opts out. The agent who moves first is selected randomly for each negotiation. At each step the agent who moves has two choices: *concede* or *opt out*, while the other one *stands still*. Agents are forced to concede until a *logical compatibility* is reached between the initial request and the initial supply, *i.e.*, until the inconsistency sources are eliminated in both the demand and the supply. At each step, amongst all the allowed concessions that satisfy the concession criteria enforced by the protocol, the agent should choose the concession that gives the highest utility to himself (and then the concession less decreasing its utility): the *minimal concession*. Therefore a concession should be *minimal* w.r.t. the utility loss paid by the agent who makes the concession [4]. The negotiation ends either if a logical compatibility is reached (*the negotiation succeeds*) or if one agent opts out (*the negotiation ends in a conflict deal*). For what concerns **strategy**, the main target of the agent is to reach the compatibility, because only through compatibility it is possible to reach an agreement. If it is its turn to move, an agent can choose to concede or opt

[3] "*disagreement thresholds*, also called disagreement payoffs, or reservation values, [. . .] are the minimum utility that each agent requires to pursue a deal "[8].

[4] In the following, for the sake of clarity, we always describe an interaction between only two opposite agents; although notice that multiple negotiations can be performed at the same time, among *one* agent and *many* candidate partners.

out: if the utility of the agent at that step is smaller than its disagreement threshold, then the agent opts out and the negotiation ends immediately. Otherwise, it will do a concession. We define an agent's utility function over all possible outcomes [7]:

$$u^p : \{A \cup \{Opt\}\} \rightarrow \Re$$

where $p \in \{\beta, \sigma\}$ —β and σ stand for buyer and seller respectively— A is the set of all possible agreements, *Opt* stands for Opt out.

3 Description Logics for negotiation

Here we refer to $\mathcal{AL}(D)$. Besides concepts and roles, $\mathcal{AL}(D)$ allows one to express quantitative properties on objects such as year of building, length, weight and many others by means of *concrete domains*. For the scope of the framework we propose in this paper, it is sufficient to introduce only unary predicates $=_x (\cdot)$ and $>_x (\cdot)$ where $x \in D$ [5]. Without loss of generality we assume that concrete domains we deal with are admissible [1]. In order to model the domain knowledge and represents relationships among elements, an ontology $\mathcal{O}$ is used in the form:

$$CN_1 \sqsubseteq CN_2 \qquad\qquad CN_1 \sqsubseteq \neg CN_2$$

Formulas representing demands D and supplies S, are expressed as generic formulas $\exists R \sqcap \forall R.C$, so an example advertisement can be formalized as in the following formula:

$\texttt{PC} \sqcap \neg\texttt{Notebook} \sqcap (\texttt{ram} \geq 1024) \sqcap (\texttt{hdd} \leq 160) \sqcap \exists\texttt{hasOS} \sqcap \forall\texttt{hasOS}.\texttt{linux} \sqcap \exists\texttt{monitor} \sqcap \forall\texttt{monitor}.(\texttt{LCDmonitor} \sqcap (\texttt{inch} \geq 17))$

Notice that for what concerns numerical properties, also range expressions are allowed in the form $(f \geq n) \sqcap (f \leq n)$. Even though subsumption and satisfiability are very useful reasoning tasks for matchmaking in e-commerce scenarios [3], there are typical problems related to negotiation that need non-standard reasoning services. For instance, suppose you have the buyer's agent β with her demand represented by the concept D and the seller's agent σ with his supply represented by S. In case $S \sqcap D \sqsubseteq_{\mathcal{O}} \bot$ holds, how to suggest to β what in D is in conflict with S and conversely to σ what in S is conflict with D? The above question is very common, among others, in negotiation scenarios where you need to know "what is wrong" between D and S and negotiate on it. In order to give an answer to the previous question and provide explanations, concept contraction[3] can be exploited.

Concept Contraction . Given two concepts C_1 and C_2 and an ontology $\mathcal{O}$, where both $C_1 \sqcap C_2 \sqsubseteq_{\mathcal{O}} \bot$ holds, find two concepts K (for Keep) and G (for Give up) such that both $C_1 \equiv K \sqcap G$ and $K \sqcap C_2 \not\sqsubseteq_{\mathcal{O}} \bot$.

In other words K represents a contraction of C_1 which is satisfiable with C_2, whilst G represents the reason why C_1 and C_2 are not compatible with each other. With concept contraction, conflicting information both in β's request w.r.t. σ'supply can be computed

[5] Hereafter, for the sake of clarity we will use an infix notation instead of a prefix one to deal with predicates over concrete domains *e.g.*, $(f \leq n) = \neg >_n (f)$.

and vice versa. Actually, for concept contraction minimality criteria have to be introduced. Following the Principle of Informational Economy [5], for G we have to give up as little information as possible. In [2, 3] some minimality criteria were introduced and analyzed. In particular, if the adopted DL admits a normal form with conjunctions of concepts as $\mathcal{AL}(D)$, $G_\exists$ *minimal* irreducible solutions can be defined.

Let C_1 and C_2 be two concepts such that $C_1 \sqcap C_2 \sqsubseteq_{\mathcal{O}} \bot$. For the corresponding Concept Contraction problem $\mathcal{Q}$, we say the solution $\langle G_{irr}, K_{irr} \rangle$ problem is G-*irreducible* if the following conditions hold:

1. $G_{irr} = \bigsqcap_{i=1...n} G_i$ where G_i is in the form $\exists R$ iff $C_2 \sqsubseteq \forall R.\bot$; [$G_\exists$ minimal condition]
2. $K \sqcap G_{\bar{i}} \sqcap C_2 \sqsubseteq_{\mathcal{O}} \bot$, for any $G_{\bar{i}}$, $\bar{i} = 1 \ldots n$;
3. if $\langle \overline{G}, \overline{K} \rangle$ is another solution to $\mathcal{Q}$ satisfying condition 1, then $\overline{G} \sqsubseteq G_{irr}$.

4 Dealing with Incomplete Information

Information about supply/demand descriptions can be, in our setting, incomplete. This may happen not only because some information may be unavailable, but also because some details have been considered irrelevant by either the seller or the buyer when they submitted their advertisements. Some user may find tiresome to specify a lot of characteristics related *e.g.*, to the brand or more technical characteristics of the product the user can be unaware of. The most common approach to this problem is avoiding incompleteness by forcing the user to fill long and tedious forms. There are several ways to deal with incomplete information and the choice may influence a negotiation.

Under an ***open-world assumption*** we have two possible choices. First, we can keep incomplete information as *missing* information: we do not know *e.g.*, if the buyer is not interested in a particular characteristic or he simply has forgotten to specify it. In this case the system has to contact to buyer/seller to further refine her/his description. Asking the users to refine their descriptions before the negotiation process starts it seems quite unrealistic, because of the amount of descriptions that can be stored in the system itself. It appears more feasible to leave this phase after the negotiation process has been performed with the counterparts in the e-marketplace, and only a small amount of supplies/demands have been retrieved. For instance, the ones with the highest utility product [9].

Once buyer and seller have refined their descriptions it is possible to start a new negotiation (the so-called *post-negotiation* phase) where only the *updated* information is negotiated.

On the other way, still in the open-world assumption setting, a second possible choice can be to assume incomplete information as an *any-would-fit* assertion (don't care), so the system should cope with this incompleteness as is. Therefore also this information will be presented in the final agreement.

5 A logic-based alternating-offers protocol

In this Section we show how to use DLs and a non-standard reasoning service, namely Concept Contraction, to model an alternating-offer protocol taking into account the semantics of request and offers as well as the domain knowledge modeled within an ontology.

For the sake of clarity and without loss of generality, from now on we consider that the agent entering the marketplace is the buyer β and her potential partners are the sellers' agents σ.

First of all, the buyer's demand D is normalized considering the equivalence $\forall R.(C \sqcap D) \equiv \forall R.C \sqcap \forall R.D$ as a rewrite rule from left to right.

After the normalization D is then a conjunction of elements in the form

$$D = \prod_k \exists R_k \sqcap \prod_i C_i \tag{1}$$

where $C_i \in \{CN, \neg CN, p(f), \neg p(f), \forall R.C\}$. As an example consider the concept in Section 3. After the normalization it is then rewritten as $\texttt{PC} \sqcap \neg\texttt{Notebook} \sqcap (\texttt{ram} \geq 1024) \sqcap (\texttt{hdd} \leq 160) \sqcap \exists\texttt{hasOS} \sqcap \forall\texttt{hasOS.linux} \sqcap \exists\texttt{monitor} \sqcap \forall\texttt{monitor.LCDmonitor} \sqcap \forall\texttt{monitor.}(\texttt{inch} \geq 17)$

In the normalized form $\exists R_k$ and C_i represent issues on which the user is willing to negotiate on. The buyer is able to express her utilities on single issues or on bundles of them. For instance, w.r.t. the previous request the buyer may set utility values on a single issue $\texttt{PC}$as well as on the whole formula $(\texttt{ram} \geq 1024) \sqcap \forall\texttt{monitor.LCDmonitor}$ (bundle of issue). We indicate these concepts with P_k — for **P**references.

Now, for each P_k the buyer β expresses a utility value $u^\beta(P_k)$ such that $\sum_i u^\beta(P_k) = 1$. As usual, both agents' utilities are normalized to 1 to eliminate outliers, and make them comparable. Since we assumed that utilities are additive, the global utility is just a sum of the utilities related to preferences.

$$u^\beta = \sum_k u^\beta(P_k)$$

After single item's utilities have been elicited, β set the **disagreement threshold** t_β (see Section 2). The same for the seller.

5.1 The Protocol

Summing up, before the real negotiation starts (step 0) we have a demand D and a supply S such that

$$D = \prod_k \exists R_k \sqcap \prod_i C_i \qquad\qquad S = \prod_l \exists R_l \sqcap \prod_j C_j$$

Based on C_i and C_j, the buyer and seller respectively, formulate their preferences P_k (for the buyer) and P_h (for the seller) and for each of them set a utility value such that:

$$\sum_k u^\beta(P_k) = 1 \qquad\qquad \sum_h u^\sigma(P_h) = 1$$

Finally, both for β and σ we have the corresponding **disagreement thresholds** and utility functions t_β, u^β and t_σ, u^σ.

If $D \sqcap S \sqsubseteq_{\mathcal{O}} \bot$ then the demand and the supply are in conflict with each other and β and σ need to negotiate on conflicting information if they want to reach an agreement. The negotiation will follow the alternating offers protocol as described in Section 2. At each step, either β or σ gives up a portion of its conflicting information choosing the item with the minimum utility. At the beginning, both β and σ need to know what are the conflicting information. Notice that both agents β and σ know D and S, but they have no information neither on counterpart utilities nor preferences. Both β and σ solve two Concept Contraction problems, computing a $G_{\exists}$ *minimal irreducible* solution, and rewrite D and S as:

$$D = G_0^{\beta} \sqcap K_0^{\beta} \qquad\qquad S = G_0^{\sigma} \sqcap K_0^{\sigma}$$

In the above rewriting G_0^{β} and G_0^{σ} represent respectively the reason why D is in conflict with S and the reason why S is in conflict with D. At a first glance it would seem β needs only $\langle G_0^{\beta}, K_0^{\beta} \rangle$ and σ needs $\langle G_0^{\sigma}, K_0^{\sigma} \rangle$. We will see later that β needs also information on σ in order to check its fairness during negotiation steps.

Since we compute *G-irreducible* solutions we can normalize G_0^{β} and G_0^{σ}, following the same procedure for D and S, as:

$$G_0^{\beta} = G_{(0,1)}^{\beta} \ldots \sqcap G_{(0,n)}^{\beta} = \prod_{i=1}^{n} G_{(0,i)}^{\beta} \quad G_0^{\sigma} = G_{(0,1)}^{\sigma} \ldots \sqcap G_{(0,m)}^{\sigma} = \prod_{j=1}^{m} G_{(0,j)}^{\sigma}$$

In the previous formulas, indexes $(0, i)$ and $(0, j)$ represent the i-th and j-th conjunctive element in G^{β} and G^{σ}at round 0.

Due to the logic adopted for D, S and $\mathcal{O}$ we have that: ***for each*** $G_{(0,i)}^{\beta}$ ***there always exists a*** C_i ***in the normalized version of*** D*— as represented in equation (1) —* ***such that*** $G_{(0,i)}^{\beta} = C_i$. The same relation holds between each $G_{(0,j)}^{\sigma}$ and C_j in the normalized form of S. Hence, some of P_k and P_h can be partially rewritten in terms of $G_{(0,i)}^{\beta}$ and $G_{(0,j)}^{\sigma}$ respectively. Since the information in G_0^{β} and G_0^{σ} are the reason why an agreement is not possible, then either β or σ will start conceding one of $G_{(0,i)}^{\beta}$ or $G_{(0,j)}^{\sigma}$ reducing their global utility of $u(G_{(0,i)}^{\beta})$ or $u(G_{(0,j)}^{\sigma})$ respectively.

Suppose β starts the negotiation and gives up $G_{(0,2)}^{\beta} = C_5$ with $P_3 \sqsubseteq_{\mathcal{O}} G_{(0,2)}^{\beta}$. Then it reformulates its request as $D_1 = \prod_k \exists R_k \sqcap \prod_{i=1..4,6..} C_i$ and sends it to σ. Notice that since $P_3 \sqsubseteq_{\mathcal{O}} G_{(0,2)}^{\beta}$, the global utility of β decreases to $u_1^{\beta} = \sum_{k=1..2,4..} u(P_k)$.

Now, σ is able to validate if β really changed its request to reach an agreement and did not lie. To do so, σ computes $\langle G_1^{\beta}, K_1^{\beta} \rangle$ solving a concept contraction problem w.r.t. the new demand D_1 and checks if $G_0^{\beta} \sqsubseteq_{\mathcal{O}} G_1^{\beta}$. In case of positive answer, then σ knows that β did not lie and it continues the negotiation process. Otherwise it may decide to leave the negotiation (conflict deal) or ask β to reformulate its counteroffer.

If the negotiation continues, σ computes its conflicting information w.r.t. to D_1 and rewrites S as $S = G_1^{\sigma} \sqcap K_1^{\sigma}$ where $G_1^{\sigma} = \prod_{j=1}^{m} G_{(1,j)}^{\sigma}$: Again, for each $G_{(1,j)}$ there exists a C_j in the normalized version of S. Hence, if σ decides to concede $G_{(1,j)}$, its global utility decreases proportionally to the utility of P_h to which $G_{(1,j)}$ belongs to.

Similarly to σ in step 0, β computes $\langle G_1^\sigma, K_1^\beta \rangle$ and checks if $G_0^\sigma \sqsubseteq_{\mathcal{O}} G_1^\sigma$ in order to check if σ lied.

The process ends when one of the following two conditions holds:

1. the global utility of an agent is lower than its **disagreement threshold**. In this case the negotiation terminates with a conflict deal.

2. there is nothing more to negotiate on and the global utility of each agent is greater than its disagreement threshold. In this case the negotiation terminates with an agreement. **The agreement** A **is computed** simply as $A = D_{last} \sqcap S_{last}$, where D_{last} and S_{last} are the request and the offer in the last step.

5.2 Minimum Concession

Since users can express an utility value also on bundles, whenever they concede an issue as the **minimum concession** (in term of minimum global utility decrease), the set of all the bundles in which the issue is present has to be taken into account. They choose based on the utility of the whole set.

For instance, consider the buyer set as preferences the following ones:

$$P_1 = \forall R.CN_1 \qquad u^\beta(P_1) = 0.1$$
$$P_2 = (f \leq 200) \qquad u^\beta(P_2) = 0.4$$
$$P_3 = \forall R.CN_1 \sqcap \forall R.CN_2 \qquad u^\beta(P_3) = 0.5$$

and at the n-th step the conflicting information is:

$$G_n^\beta = \forall R.CN_1 \sqcap (f \leq 200)$$

Hence, β can concede whether $\forall R.CN_1$ or $(f \leq 200)$. If it concedes $\forall R.CN_1$ then its global utility decreases of $u^\beta(P_1) + u^\beta(P_3) = 0.6$, while conceding $(f \leq 200)$ its utility decreases of only $u^\beta(P_2) = 0.4$. In this case the **minimum concession** is $(f \leq 200)$.

5.3 The Algorithm

Here we define the behavior of agents during a generic n-th round of the negotiation process. For the sake of conciseness, we present only the algorithm related to β's behavior. The behavior of σ is dual w.r.t. to the one of β.

1-4 If there is nothing in conflict between the old D_{n-1} and just-arrived S_n, then there nothing more to negotiate on and the agreement is reached and computed. Notice that while computing the final agreement we use the "any-would-fit" approach to deal with *incomplete information* (see Section 4).

5-11 If β discovers that σ lied on its concession, then β decides to exit the negotiation and terminates with a conflict deal. If we want β ask σ to concede again it is straightforward to change the protocol to deal with such a behavior.

13-15 If after the minimum concession, the utility of β is less than its **disagreement threshold**, then the negotiation ends with a conflict deal.

if $D_{n-1} \sqcap S_n \not\sqsubseteq_{\mathcal{O}} \perp$ **then**
    agreement A reached;
    return $A = D_{n-1} \sqcap S_n$;
end
if $n > 0$ **then**
    compute $\langle G_n^\sigma, K_n^\sigma \rangle$ from D_{n-1} and S_n;
    if $G_{n-1}^\sigma \not\sqsubseteq_{\mathcal{O}} G_n^\sigma$ **then**
        σ lied;
        conflict deal: **exit**;
    end
end
compute minimum concession $G_{(n-1,i)}^\beta$;
if $u_{n-1}^\beta < t^\beta$ **then**
    conflict deal: **exit**;
end
formulate D_n deleting $G_{(n-1,i)}^\beta$ from D_{n-1};
send D_n to σ ;

Algorithm 1: The behavior of β at step n

6 Conclusion

We have motivated and illustrated a logic-based approach to bilateral negotiation in P2P e-marketplaces, and proposed a *DL-based* alternating-offers protocol exploiting Description Logics and related inference services and utility theory to find the most suitable agreements. Work is ongoing on various directions, namely: extending the DL adopted, finding a ”cheap” way to ensure that the reached agreement is Pareto-efficient, and carry out large scale experiments with real advertisements.

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. S. Colucci, T. Di Noia, E. Di Sciascio, F. Donini, and M. Mongiello. Concept Abduction and Contraction in Description Logics. In *Proc. of DL'03*, volume 81 of *CEUR Workshop Proceedings*, September 2003.
3. S. Colucci, T. Di Noia, E. Di Sciascio, F. Donini, and M. Mongiello. Concept Abduction and Contraction for Semantic-based Discovery of Matches and Negotiation Spaces in an E-Marketplace. *Electronic Commerce Research and Applications*, 4(4):345–361, 2005.
4. U. Endriss. Monotonic concession protocols for multilateral negotiation. In *AAMAS*. ACM, 2006.
5. P. Gärdenfors. *Knowledge in Flux: Modeling the Dynamics of Epistemic States*. Bradford Books, MIT Press, Cambridge, MA, 1988.
6. J.S. Rosenschein and G. Zlotkin. *Rules of Encounter*. MIT Press,, 1994.
7. S. Kraus. *Strategic Negotiation in Multiagent Environments*. The MIT Press.
8. A. Ragone, T. Di Noia, E. Di Sciascio, and F. Donini. A logic-based framework to compute pareto agreements in one-shot bilateral negotiation. In *Proc. of ECAI'06*, pages 230–234, 2006.
9. A. Rubinstein. Perfect equilibrium in a bargaining model. *Econometrica*, 50:97–109, 1982.