Combining Boolean Games with the Power of Ontologies for Automated Multi-Attribute Negotiation in the Semantic Web

Thomas Lukasiewicz[1,3] and Azzurra Ragone[2]

[1] Computing Laboratory, University of Oxford
Wolfson Building, Parks Road, Oxford OX1 3QD, UK
thomas.lukasiewicz@comlab.ox.ac.uk

[2] Information Systems Laboratory, Politecnico di Bari
Via E. Orabona 4, 70125 Bari, Italy
a.ragone@poliba.it

Abstract. Recently, multi-attribute negotiation has been extensively studied from a game-theoretic viewpoint. Since normal and extensive form games have the drawback of requiring an explicit representation of utility functions (listing the utility values for all combinations of strategies), logical preference languages have been proposed, which provide a convenient way to compactly specify multi-attribute utility functions. Among these preference languages, there are also Boolean games. In this paper, towards automated multi-attribute negotiation in the Semantic Web, we introduce Boolean description logic games, which are a combination of Boolean games with ontological background knowledge, formulated in expressive description logics. We include and discuss several generalizations, and show how a travel and a service negotiation scenario can be formulated in Boolean description logic games, which shows their practical usefulness.

1 Introduction

During the recent decade, a huge amount of research activities has been centered around the problem of automated negotiation. This is especially due to the development of the World Wide Web, which has provided the means and the commercial necessity for the further development of computational negotiation and bargaining techniques [1].

Another area with an impressive amount of recent research activities is the *Semantic Web* [2,3], which aims at an extension of the current World Wide Web by standards and technologies that help machines to understand the information on the Web so that they can support richer discovery, data integration, navigation, and automation of tasks. The main ideas behind it are to add a machine-readable meaning to Web pages, to use ontologies for a precise definition of shared terms in Web resources, to use knowledge representation technology for automated reasoning from Web resources, and to apply cooperative agent technology for processing the information of the Web.

Only a marginal amount of research activities, however, focuses on the intersection of automated negotiation and the Semantic Web (see Section 6). This is surprising,

[3] Alternative address: Institut für Informationssysteme, Technische Universität Wien, Favoritenstr. 9-11, 1040 Wien, Austria; email: lukasiewicz@kr.tuwien.ac.at.

since representation and reasoning technologies from the Semantic Web may be used to further enhance automated negotiation on the Web, e.g., by providing ontological background knowledge. Moreover, although one important ingredient of the Semantic Web is agent technology, the agents are still largely missing in Semantic Web research to date [4]. This paper is a first step in direction to filling this gap. Towards automated multi-attribute negotiation in the Semantic Web, we introduce Boolean description logic games. The main contributions of this paper are briefly summarized as follows:

- We define Boolean description logic games, which are a combination of n-player Boolean games with description logics. They informally combine n-player Boolean games with ontological background knowledge; in addition, we also introduce strict agent requirements and overlapping agent control assignments.
- We then generalize to Boolean dl-games where each agent has a set of weighted goals, which may be defined over free description logic concepts. We finally propose another generalization, where the agents own roles rather than concepts.
- We provide many examples (from a travel and a service negotiation scenario), which illustrate the introduced concepts related to Boolean description logic games, and which give evidence of the practical usefulness of our approach.

Intuitively, such games aim at a centralized one-step negotiation process, where the agents reveal their preferences to a central mediator, which then calculates one optimal strategy for each agent. Clearly, this is also closely related to service matchmaking and resource retrieval, since the service provider and the service consumer can be both considered as agents having certain service specifications and service preferences, and the result of the negotiation process is then the service where the service specifications are matching optimally the service preferences (see also Example 5.1).

The rest of this paper is organized as follows. In Section 2, we recall the basics of description logics and Boolean games. Section 3 defines Boolean description logic games. In Section 4, we introduce Boolean description logic games with weighted generalized goals. Section 5 generalizes the ontological ownerships. In Section 6, we discuss related work. Section 7 summarizes the main results and gives an outlook on future research.

2 Preliminaries

In this section, we recall the basic concepts of description logics and Boolean games.

2.1 Description Logics

We now recall the description logics $\mathcal{SHIF}(\mathbf{D})$ and $\mathcal{SHOIN}(\mathbf{D})$, which stand behind the web ontology languages OWL Lite and OWL DL [5], respectively. Intuitively, description logics model a domain of interest in terms of concepts and roles, which represent classes of individuals and binary relations between classes of individuals, respectively. A description logic knowledge base encodes especially subset relationships between concepts, subset relationships between roles, the membership of individuals to concepts, and the membership of pairs of individuals to roles.

Syntax. We first describe the syntax of $\mathcal{SHOIN}(\mathbf{D})$. We assume a set of *elementary datatypes* and a set of *data values*. A *datatype* is either an elementary datatype or a set of data values (called *datatype oneOf*). A *datatype theory* $\mathbf{D} = (\Delta^{\mathbf{D}}, \cdot^{\mathbf{D}})$ consists of a *datatype domain* $\Delta^{\mathbf{D}}$ and a mapping $\cdot^{\mathbf{D}}$ that assigns to each elementary datatype a subset of $\Delta^{\mathbf{D}}$ and to each data value an element of $\Delta^{\mathbf{D}}$. The mapping $\cdot^{\mathbf{D}}$ is extended to all datatypes by $\{v_1, \ldots\}^{\mathbf{D}} = \{v_1^{\mathbf{D}}, \ldots\}$. Let $\mathbf{A}$, $\mathbf{R}_A$, $\mathbf{R}_D$, and $\mathbf{I}$ be pairwise disjoint (nonempty) denumerable sets of *atomic concepts*, *abstract roles*, *datatype roles*, and *individuals*, respectively. We denote by $\mathbf{R}_A^-$ the set of inverses R^- of all $R \in \mathbf{R}_A$.

A *role* is an element of $\mathbf{R}_A \cup \mathbf{R}_A^- \cup \mathbf{R}_D$. *Concepts* are inductively defined as follows. Every $\phi \in \mathbf{A}$ is a concept, and if $o_1, \ldots, o_n \in \mathbf{I}$, then $\{o_1, \ldots, o_n\}$ is a concept (called *oneOf*). If ϕ, ϕ_1, and ϕ_2 are concepts and if $R \in \mathbf{R}_A \cup \mathbf{R}_A^-$, then also $(\phi_1 \sqcap \phi_2)$, $(\phi_1 \sqcup \phi_2)$, and $\neg\phi$ are concepts (called *conjunction*, *disjunction*, and *negation*, respectively), as well as $\exists R.\phi$, $\forall R.\phi$, $\geqslant nR$, and $\leqslant nR$ (called *exists*, *value*, *atleast*, and *atmost restriction*, respectively) for an integer $n \geqslant 0$. If D is a datatype and $U \in \mathbf{R}_D$, then $\exists U.D$, $\forall U.D$, $\geqslant nU$, and $\leqslant nU$ are concepts (called *datatype exists*, *value*, *atleast*, and *atmost restriction*, respectively) for an integer $n \geqslant 0$. We write $\exists R$ and $\forall R$ to abbreviate $\exists R.\top$ and $\forall R.\top$, respectively. We write $\top$ and $\bot$ to abbreviate the concepts $\phi \sqcup \neg\phi$ and $\phi \sqcap \neg\phi$, respectively, and we eliminate parentheses as usual.

An *axiom* has one of the following forms: (1) $\phi \sqsubseteq \psi$ (called *concept inclusion axiom*), where ϕ and ψ are concepts; (2) $R \sqsubseteq S$ (called *role inclusion axiom*), where either $R, S \in \mathbf{R}_A$ or $R, S \in \mathbf{R}_D$; (3) $\mathrm{Trans}(R)$ (called *transitivity axiom*), where $R \in \mathbf{R}_A$; (4) $\phi(a)$ (called *concept membership axiom*), where ϕ is a concept and $a \in \mathbf{I}$; (5) $R(a, b)$ (resp., $U(a, v)$) (called *role membership axiom*), where $R \in \mathbf{R}_A$ (resp., $U \in \mathbf{R}_D$) and $a, b \in \mathbf{I}$ (resp., $a \in \mathbf{I}$ and v is a data value); and (6) $a = b$ (resp., $a \neq b$) (*equality* (resp., *inequality*) *axiom*), where $a, b \in \mathbf{I}$. A *knowledge base* L is a finite set of axioms. For decidability, number restrictions in L are restricted to simple abstract roles [6]. Since knowledge bases encode ontologies, we also use *ontology* to denote a knowledge base.

The syntax of $\mathcal{SHIF}(\mathbf{D})$ is as the above syntax of $\mathcal{SHOIN}(\mathbf{D})$, but without the oneOf constructor and with the atleast and atmost constructors limited to 0 and 1.

Example 2.1 (travel ontology). A description logic knowledge base L encoding a travel ontology (adapted from http://protege.cim3.net/file/pub/ontologies/travel/) is given by the axioms in Fig. 1. For example, there are some axioms encoding that bed and breakfast accommodations and hotels are different accommodations, and that a budget accommodation is an accommodation that has one or two stars as a rating.

Semantics. An *interpretation* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ w.r.t. a datatype theory $\mathbf{D} = (\Delta^{\mathbf{D}}, \cdot^{\mathbf{D}})$ consists of a nonempty (*abstract*) *domain* $\Delta^{\mathcal{I}}$ disjoint from $\Delta^{\mathbf{D}}$, and a mapping $\cdot^{\mathcal{I}}$ that assigns to each atomic concept $\phi \in \mathbf{A}$ a subset of $\Delta^{\mathcal{I}}$, to each individual $o \in \mathbf{I}$ an element of $\Delta^{\mathcal{I}}$, to each abstract role $R \in \mathbf{R}_A$ a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathcal{I}}$, and to each datatype role $U \in \mathbf{R}_D$ a subset of $\Delta^{\mathcal{I}} \times \Delta^{\mathbf{D}}$. We extend $\cdot^{\mathcal{I}}$ to all concepts and roles, and we define the *satisfaction* of an axiom F in an interpretation $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$, denoted $\mathcal{I} \models F$, as usual [5]. We say $\mathcal{I}$ *satisfies* the axiom F, or $\mathcal{I}$ is a *model* of F, iff $\mathcal{I} \models F$. We say $\mathcal{I}$ *satisfies* a knowledge base L, or $\mathcal{I}$ is a *model* of L, denoted $\mathcal{I} \models L$, iff $\mathcal{I} \models F$ for all $F \in L$. We say L is *satisfiable* (resp., *unsatisfiable*) iff L has a (resp., no) model. An axiom F is a *logical consequence* of L, denoted $L \models F$, iff each model of L satisfies F.

$$
\begin{aligned}
&\mathit{BedAndBreakfast} \sqsubseteq \mathit{Accomodation};\\
&\mathit{Hotel} \sqsubseteq \mathit{Accomodation};\\
&\mathit{BedAndBreakfast} \sqsubseteq \neg\mathit{Hotel};\\
&\mathit{BudgetAccommodation} \equiv \mathit{Accomodation} \sqcap \exists \mathit{hasRating}.\{\mathit{OneStarRating}, \mathit{TwoStarRating}\};\\
&\mathit{UrbanArea} \sqsubseteq \mathit{Destination};\\
&\mathit{City} \sqsubseteq \mathit{UrbanArea};\\
&\mathit{Capital} \sqsubseteq \mathit{City};\\
&\mathit{RuralArea} \sqsubseteq \mathit{Destination};\\
&\mathit{NationalPark} \sqsubseteq \mathit{RuralArea};\\
&\mathit{RuralArea} \sqsubseteq \neg\mathit{UrbanArea};\\
&\mathit{BudgetHotelDestination} \equiv \exists \mathit{hasAccomodation}\\
&\qquad\qquad\qquad\qquad\qquad \sqcap \forall \mathit{hasAccomodation}.(\mathit{BudgetAccommodation} \sqcap \mathit{Hotel});\\
&\mathit{AccommodationRating} \equiv \{\mathit{OneStarRating}, \mathit{TwoStarRating}, \mathit{ThreeStarRating}\};\\
&\mathit{Sightseeing} \sqsubseteq \mathit{Activity};\\
&\mathit{Hiking} \sqsubseteq \mathit{Sport};\\
&\mathit{Sport} \sqsubseteq \mathit{Activity};\\
&\mathit{ThemePark} \sqsubseteq \mathit{Activity};\\
&\mathit{FamilyDestination} \equiv \exists \mathit{hasDestination} \sqcap \exists \mathit{hasAccomodation} \sqcap \geqslant 3\, \mathit{hasActivity};\\
&\mathit{RelaxDestination} \equiv \exists \mathit{hasDestination}.\mathit{NationalPark} \sqcap \exists \mathit{hasActivity}.\mathit{Sightseeing};\\
&\mathit{hasActivity} \equiv \mathit{isOfferedAt}^{-}.
\end{aligned}
$$

Fig. 1. Travel ontology.

Example 2.2 (travel ontology cont'd). It is not difficult to verify that the description logic knowledge base L of Example 2.1 is satisfiable, and that the two concept inclusion axioms *Capital* $\sqsubseteq$ *UrbanArea* and *Capital* $\sqsubseteq \neg$*RuralArea* are logical consequences of L. Informally, L implies that capitals are urban and not rural areas.

2.2 Boolean Games

We now recall n-player Boolean games from [7], which are a generalization of 2-player Boolean games from [8,9]. Such games are essentially normal form games where propositional logic is used for compactly specifying multi-attribute utility functions. We first give some preparative definitions, and then recall n-player Boolean games, including their ingredients, strategy profiles, and important notions of optimality.

We assume a finite set of propositional variables $V = \{p_1, p_2, \ldots, p_k\}$. We denote by $\mathcal{L}_V$ the set of all propositional formulas (denoted by Greek letters $\psi, \phi, \ldots$) built inductively from V via the Boolean operators $\neg$, $\wedge$, and $\vee$.

An *n-player Boolean game* $G = (N, V, \pi, \Phi)$ consists of

(1) a set of n players $N = \{1, 2, \ldots, n\}$, $n \geqslant 2$,
(2) a finite set of propositional variables V,
(3) a *control assignment* $\pi\colon N \to 2^V$, which associates with every player $i \in N$ a set of variables $\pi(i) \subseteq V$, which she controls, such that $\{\pi(i) \mid i \in N\}$ partitions V, and
(4) a *goal assignment* $\Phi\colon N \to \mathcal{L}_V$, which associates with every player $i \in N$ a propositional formula $\Phi(i) \in \mathcal{L}_V$, denoted the *goal* of i.

Example 2.3 (Boolean game). A two-player Boolean game $G = (N, V, \pi, \Phi)$ is given by:

(1) the set of two players $N = \{1, 2\}$,

	b	$\overline{b}$
$a\,c$	(1,0)	(0,1)
$a\,\overline{c}$	(1,0)	**(1,1)**
$\overline{a}\,c$	(0,0)	(0,1)
$\overline{a}\,\overline{c}$	(0,0)	**(1,0)**

Fig. 2. Normal form of a two-player Boolean game.

(2) the set of propositional variables $V = \{a, b, c\}$,
(3) the control assignment $\pi(1) = \{a, c\}$ and $\pi(2) = \{b\}$, and
(4) the goal assignment $\Phi(1) = (a \wedge b) \vee (\neg c \wedge \neg b)$ and $\Phi(2) = (c \wedge \neg b) \vee (a \wedge \neg b)$.

Informally, we have two players 1 and 2, and three propositional variables a, b, and c. Player 1 (resp., 2) controls the variables a and c (resp., the variable b) and has the goal expressed by the propositional formula $\Phi(1)$ (resp., $\Phi(2)$).

A *strategy* for player $i \in N$ is any truth assignment s_i to the variables in $\pi(i)$. A *strategy profile* $s = (s_1, \ldots, s_n)$ consists of one strategy s_i for every $i \in N$. The *utility* to player $i \in N$ under s, denoted $u_i(s)$, is 1, if s satisfies i's goal $\Phi(i)$, and 0, otherwise.

Towards optimal behavior of the players in an n-player Boolean game, we are especially interested in strategy profiles s, called Nash equilibria, where no agent has the incentive to deviate from its part, once the other agents play their parts. More formally, a strategy profile $s = (s_1, \ldots, s_n)$ is a *Nash equilibrium* iff $u_i(s \lhd s'_i) \leqslant u_i(s)$ for every strategy s'_i of player i and for every player $i \in N$, where $s \lhd s'_i$ is the strategy profile obtained from $s = (s_1, \ldots, s_n)$ by replacing s_i by s'_i.

Another important notion of optimality is Pareto-optimality. Informally, a strategy profile is Pareto-optimal if there exists no other strategy profile that makes one player better off and no player worse off in the utility. More formally, a strategy profile s is *Pareto-optimal* iff there exists no strategy profile s' such that (i) $u_i(s') > u_i(s)$ for some player $i \in N$ and (ii) $u_i(s') \geqslant u_i(s)$ for every player $i \in N$.

Example 2.4 (Boolean game cont'd). Consider again the two-player Boolean game $G = (N, V, \pi, \Phi)$ of Example 2.3. Player 1 has all truth assignments to the variables a and c (that is, $a, c \mapsto \mathbf{true}, \mathbf{true}$, $a, c \mapsto \mathbf{true}, \mathbf{false}$, $a, c \mapsto \mathbf{false}, \mathbf{true}$, and $a, c \mapsto \mathbf{false}, \mathbf{false}$, denoted $a\,c$, $a\,\overline{c}$, $\overline{a}\,c$, and $\overline{a}\,\overline{c}$, respectively) as strategies, while player 2 has all truth assignments to b as strategies (that is, $b \mapsto \mathbf{true}$ and $b \mapsto \mathbf{false}$, denoted b and $\overline{b}$, respectively). Any combination of the strategies of two players is a strategy profile. For example, $(a\,c, b)$ is a strategy profile combining the strategy $a\,c$ of player 1 and the strategy b of player 2.

The *normal form* of this two-player Boolean game, using the above strategy profiles $s = (s_1, s_2)$, which combine all strategies s_1 and s_2 of the players 1 and 2, respectively, is depicted in Fig. 2: for every strategy profile $s = (s_1, s_2)$, the matrix has one entry, which shows the pair of utilities $(u_1(s), u_2(s))$ under s to the two players. The utility $u_i(s)$ is equal to 1, when $\Phi(i)$ is satisfied in s, and 0, otherwise.

It is then not difficult to verify that the strategy profile $(a\,\overline{c}, \overline{b})$ is a (pure) Nash equilibrium of this two-player Boolean game G, which is also Pareto-optimal, while $(\overline{a}\,\overline{c}, \overline{b})$ is also a (pure) Nash equilibrium of G, but not Pareto-optimal.

3 Boolean Description Logic Games

In this section, we combine classical n-player Boolean games with ontologies. The main differences to classical n-player Boolean games are summarized as follows (note that many of the new features are also illustrated in Example 3.1):

– Rather than unrelated propositional variables, the agents now control atomic description logic concepts, which may (abbreviate complex description logic concepts and) be related via a description logic knowledge base. In fact, the assumption that the controlled variables are unrelated in classical n-player Boolean games is quite unrealistic; often the variables are related through some background knowledge.
– Rather than having only preferences, the agents may now also have *strict goals*, which have to be necessarily true in an admissible agreement. This reflects the fact that agents accept no agreement where some strict conditions are not true; such strict conditions are very common in many applications in practice.
– Rather than defining a partition, the control assignment may now be overlapping. In fact, such overlapping control assignments are also more realistic.

We first give some preparative definitions as follows. We use a finite set of atomic concepts $\mathcal{A}$ as set of propositional variables V in n-player Boolean games. We denote by $\mathcal{L}_{\mathcal{A}}$ the set of all concepts (denoted by Greek letters $\psi, \phi, \ldots$) built inductively from $\mathcal{A}$ via the Boolean operators $\neg$, $\sqcap$, and $\sqcup$. An *interpretation* I is a full conjunction of atomic concepts and negated atomic concepts from $\mathcal{A}$. We say I *satisfies* a description logic knowledge base L, denoted $I \models L$, iff $L \cup \{I(o)\}$ is satisfiable, where o is a new individual. We say I *satisfies* a concept ϕ over $\mathcal{A}$ under L, denoted $I \models_L \phi$, iff $L \models I \sqsubseteq \phi$. We say ϕ is *satisfiable* under L iff there exists an interpretation I such that $I \models_L \phi$. We are now ready to define n-agent Boolean description logic games.

Definition 3.1 **(n-agent Boolean description logic games).** An *n-agent Boolean description logic game* (or *n-agent Boolean dl-game*) $G = (L, N, \mathcal{A}, \pi, \Sigma, \Phi)$ consists of

(1) a description logic knowledge base L,
(2) a finite set of n agents $N = \{1, 2, \ldots, n\}$, $n \geqslant 2$,
(3) a finite set of atomic concepts $\mathcal{A}$,
(4) a *control assignment* $\pi\colon N \rightarrow 2^V$, which associates with every agent $i \in N$ a set of atomic concepts $\pi(i) \subseteq \mathcal{A}$, which she controls,
(5) a *strict goal assignment* $\Sigma\colon N \rightarrow \mathcal{L}_{\mathcal{A}}$, which associates with every agent $i \in N$ a concept $\Sigma(i) \in \mathcal{L}_{\mathcal{A}}$ that is satisfiable under L, denoted the *strict goal* of i, and
(6) a *goal assignment* $\Phi\colon N \rightarrow \mathcal{L}_{\mathcal{A}}$, which associates with every agent $i \in N$ a concept $\Phi(i) \in \mathcal{L}_{\mathcal{A}}$ that is satisfiable under L, denoted the *goal* of i.

As for the difference between strict and general goals, the agents necessarily want their strict goals to be satisfied, but they only would like their general goals to be satisfied. The following example illustrates n-agent Boolean dl-games.

Example 3.1 (travel negotiation). A two-agent Boolean dl-game $G = (L, N, \mathcal{A}, \pi, \Sigma, \Phi)$, where the *traveler* (agent 1) negotiates with the *travel agency* (agent 2) on the conditions of a vacation, is given as follows:

(1) L is the travel ontology of Example 2.1, depicted in Fig. 1.
(2) $N = \{1, 2\}$, where agent 1 (resp., 2) is the *traveler* (resp., *travel agent*).
(3) $\mathcal{A}$ consists of the following atomic concepts (that are relevant to the negotiation):

$$\begin{array}{ll}
U & \equiv \exists hasDestination \sqcap \forall hasDestination.UrbanArea; \\
R & \equiv \exists hasDestination \sqcap \forall hasDestination.RuralArea; \\
BHD & \equiv BudgetHotelDestination; \\
BA & \equiv \exists hasAccomodation \sqcap \forall hasAccomodation.BudgetAccommodation; \\
H & \equiv \exists hasAccomodation \sqcap \forall hasAccomodation.Hotel; \\
BB & \equiv \exists hasAccomodation \sqcap \forall hasAccomodation.BedAndBreakfast; \\
NP & \equiv \exists hasDestination \sqcap \forall hasDestination.NationalPark; \\
C & \equiv \exists hasDestination \sqcap \forall hasDestination.Capital.
\end{array}$$

(4) Agents 1 and 2 control the following concepts $\pi(1)$ and $\pi(2)$, respectively:

$$\pi(1) = \{U, R, BHD\};$$
$$\pi(2) = \{BA, H, BB, NP, C\}.$$

Informally, agent 1 decides whether the trip takes place to an urban, rural, or budget hotel destination, while agent 2's offers fix the budget, hotel, or bed and breakfast accommodation, and the destination to a national park or a capital city.
(5) Agents 1 and 2 have the following strict goals $\Sigma(1)$ and $\Sigma(2)$, respectively:

$$\Sigma(1) = (U \sqcup R) \sqcap (H \sqcup BB);$$
$$\Sigma(2) = NP \sqcup C.$$

Informally, agent 1 necessarily wants a destination in an urban or a rural area, e.g., she does not like beach destinations, and she also wants an accommodation for her trip in a hotel or a bed and breakfast, so she is excluding e.g. camping grounds. Whereas agent 2 is trying to sell a destination in a national park or a capital city.
(6) Agents 1 and 2 have the following goals $\Phi(1)$ and $\Phi(2)$, respectively,

$$\Phi(1) = (R \sqcap BB) \sqcup (C \sqcap BHD);$$
$$\Phi(2) = (U \sqcap BB) \sqcup (NP \sqcap BHD).$$

Informally, agent 1 would like a destination in a rural area and an accommodation in a bed and breakfast, or a budget hotel accommodation in a capital city. Whereas agent 2 would like to sell a destination in an urban area and an accommodation in a bed and breakfast, or a budget hotel destination in a national park.

We next define the notions of strategies, strategy profiles, and utility functions. In classical n-agent Boolean games, a strategy for agent i is a truth assignment s_i to all the variables she controls, and the utility functions of the agents depend on their goals built from the variables. In our setting, in contrast, atomic concepts are related to each other through a description logic knowledge base L, and each agent may have some strict requirements, and so some truth assignments to the atomic concepts may be infeasible because of L and the strict requirements. We thus exclude such infeasible strategies. In addition, some combinations I of feasible strategies may result in an infeasible strategy profile due to L and the fact that the control assignment may be overlapping. We model

	$U \sqcap \neg R \sqcap BHD$	$\neg U \sqcap R \sqcap BHD$	$U \sqcap \neg R \sqcap \neg BHD$	$\neg U \sqcap R \sqcap \neg BHD$
$BA \sqcap H \sqcap \neg BB \sqcap NP \sqcap \neg C$	$(-1,-1)$	$(\mathbf{0},\mathbf{1})$	$(-1,-1)$	$(0,0)$
$BA \sqcap \neg H \sqcap BB \sqcap NP \sqcap \neg C$	$(-1,-1)$	$(-1,-1)$	$(-1,-1)$	$(\mathbf{1},\mathbf{0})$
$BA \sqcap H \sqcap \neg BB \sqcap \neg NP \sqcap C$	$(\mathbf{1},\mathbf{0})$	$(-1,-1)$	$(\mathbf{0},\mathbf{0})$	$(-1,-1)$
$BA \sqcap \neg H \sqcap BB \sqcap \neg NP \sqcap C$	$(-1,-1)$	$(-1,-1)$	$(\mathbf{0},\mathbf{1})$	$(-1,-1)$
$\neg BA \sqcap H \sqcap \neg BB \sqcap NP \sqcap \neg C$	$(-1,-1)$	$(-1,-1)$	$(-1,-1)$	$(0,0)$
$\neg BA \sqcap \neg H \sqcap BB \sqcap NP \sqcap \neg C$	$(-1,-1)$	$(-1,-1)$	$(-1,-1)$	$(\mathbf{1},\mathbf{0})$
$\neg BA \sqcap H \sqcap \neg BB \sqcap \neg NP \sqcap C$	$(-1,-1)$	$(-1,-1)$	$(\mathbf{0},\mathbf{0})$	$(-1,-1)$
$\neg BA \sqcap \neg H \sqcap BB \sqcap \neg NP \sqcap C$	$(-1,-1)$	$(-1,-1)$	$(\mathbf{0},\mathbf{1})$	$(-1,-1)$

Fig. 3. Normal form of a two-agent Boolean dl-game.

this, exploiting the utility structure: if I is infeasible due to L or the overlapping control assignment, then the utility to all agents is -1; in contrast, if I is feasible, then the utility to agent i under I is equal to 1, if its goal $\Phi(i)$ is satisfied, and 0, otherwise. Therefore, when the agreement I is unsatisfiable, then the utilities are always negative, that is, always less than the utilities when the agreement I is satisfiable. Hence, the unsatisfiable agreement will never be chosen by the agents.

Definition 3.2 (strategies, strategy profiles, utilities). Let $G = (L, N, \mathcal{A}, \pi, \Sigma, \Phi)$ be an n-agent Boolean dl-game. Then, a *strategy* for agent $i \in N$ is an interpretation I_i for the concepts in $\pi(i)$ that satisfies both (i) L and (ii) $\Sigma(i)$ under L. A *strategy profile* $I = (I_1, I_2, \ldots, I_n)$ consists of one strategy I_i for every agent $i \in N$. We say $I = (I_1, I_2, \ldots, I_n)$ is *consistent* iff (i) there exists an interpretation J for $\mathcal{A}$ such that I_i is the restriction of J to $\pi(i)$, for every agent $i \in N$, and (ii) I satisfies L. The *utility* to agent $i \in N$ under I, denoted $u_i(I)$, is defined as follows:

$$u_i(I) \;=\; \begin{cases} -1 & \text{if } I \text{ is inconsistent, or } I \not\models_L \Sigma(i); \\ 1 & \text{if } I \text{ is consistent, } I \models_L \Sigma(i), \text{ and } I \models_L \Phi(i); \\ 0 & \text{if } I \text{ is consistent, } I \models_L \Sigma(i), \text{ and } I \not\models_L \Phi(i). \end{cases}$$

We illustrate the above ideas with the help of a simple example.

Example 3.2 (travel negotiation cont'd). The sets of all strategies $\mathcal{I}_1$ and $\mathcal{I}_2$ of agents 1 and 2, respectively, in the travel negotiation example are given as follows:

$$\begin{aligned} \mathcal{I}_1 = \{ & BA \sqcap H \sqcap \neg BB \sqcap NP \sqcap \neg C,\ BA \sqcap \neg H \sqcap BB \sqcap NP \sqcap \neg C, \\ & BA \sqcap H \sqcap \neg BB \sqcap \neg NP \sqcap C,\ BA \sqcap \neg H \sqcap BB \sqcap \neg NP \sqcap C, \\ & \neg BA \sqcap H \sqcap \neg BB \sqcap NP \sqcap \neg C,\ \neg BA \sqcap \neg H \sqcap BB \sqcap NP \sqcap \neg C, \\ & \neg BA \sqcap H \sqcap \neg BB \sqcap \neg NP \sqcap C,\ \neg BA \sqcap \neg H \sqcap BB \sqcap \neg NP \sqcap C\}; \\ \mathcal{I}_2 = \{ & U \sqcap \neg R \sqcap BHD,\ \neg U \sqcap R \sqcap BHD,\ U \sqcap \neg R \sqcap \neg BHD,\ \neg U \sqcap R \sqcap \neg BHD\}. \end{aligned}$$

The set of all strategy profiles is $\mathcal{I}_1 \times \mathcal{I}_2$. The utility pairs $(u_1(I), u_2(I))$ for each strategy profile $I = (I_1, I_2)$ are shown in Fig. 3, which actually depicts the normal form of the two-agent Boolean dl-game G. Note that all inconsistent strategy profiles (due to the description logic knowledge base L) are associated with two negative utilities.

We next define (pure) Nash equilibria of n-agent Boolean dl-games. Informally, as in the classical case, they are strategy profiles where no agent has the incentive to deviate from its part once the other agents stick to their parts.

Definition 3.3 (pure Nash equilibria). Let $G=(L,N,\mathcal{A},\pi,\Phi)$ be an n-agent Boolean dl-game with $N=\{1,\ldots,n\}$. Then, a strategy profile $I=(I_1,\ldots,I_n)$ is a *(pure) Nash equilibrium* of G iff $u_i(I \lhd I'_i) \leqslant u_i(I)$ for every strategy I'_i of agent i and for every agent $i \in N$, where $I \lhd I'_i$ is the strategy profile obtained from I by replacing I_i by I'_i.

Another concept of optimality for strategy profiles is the notion of Pareto-optimality. Informally, a strategy profile is Pareto-optimal if there exists no other strategy profile that makes one agent better off and no agent worse off in the utility. Note that, as in the classical case, Nash equilibria are not necessarily Pareto-optimal.

Definition 3.4 (Pareto-optimal strategy profiles). Let $G=(L,N,\mathcal{A},\pi,\Phi)$ be an n-agent Boolean dl-game with $N=\{1,\ldots,n\}$. Then, a strategy profile $I=(I_1,\ldots,I_n)$ is *Pareto-optimal* iff there exists no strategy profile I' such that (i) $u_i(I')>u_i(I)$ for some agent $i\in N$ and (ii) $u_i(I')\geqslant u_i(I)$ for every agent $i\in N$.

We illustrate the notions of Nash equilibria and Pareto-optimality in our example.

Example 3.3 (travel negotiation cont'd). The set of all (pure) Nash equilibria of the two-agent Boolean dl-game G of Example 3.1 are given by the bold entries in Fig. 3. It is not difficult to verify that all except for the $(0,0)$ ones are also Pareto-optimal.

4 Weighted Generalized Goals

In this section, we further extend Boolean dl-games by weighted and generalized goals:

- Instead of one single goal that each agent wants to satisfy, we now assume a set of goals for each agent, where each goal of an agent is associated with a weight. This considers the fact that goals can have different importance, so the best agreement is not necessarily the agreement satisfying the greatest number of goals for each agent. We first define Boolean dl-games with weighted goals, that is, multi-valued preferences. Note that agent utilities are normalized to 1 to make them comparable.
- As another difference to Boolean dl-games, we also do not assume anymore that agent goals are constructed from the controlled atomic concepts.

Definition 4.1 (n-agent Boolean dl-games with weighted goals). An *n-agent Boolean dl-game with weighted goals* $G=(L,N,\mathcal{A},\pi,\Sigma,\Phi)$ consists of

(1) a description logic knowledge base L,
(2) a finite set of n agents $N=\{1,2,\ldots,n\}$, $n\geqslant 2$,
(3) a finite set of atomic concepts $\mathcal{A}$,
(4) a *control assignment* $\pi\colon N \rightarrow 2^V$, which associates with every agent $i\in N$ a set of atomic concepts $\pi(i)\subseteq\mathcal{A}$, which she controls,
(5) a *strict goal assignment* $\Sigma\colon N \rightarrow \mathcal{L}_{\mathcal{A}}$, which associates with every agent $i\in N$ a concept $\Sigma(i)\in\mathcal{L}_{\mathcal{A}}$ that is satisfiable under L, denoted the *strict goal* of i, and
(6) a *weighted goal assignment* Φ, which associates with every agent $i\in N$ a mapping Φ_i from a finite set of concepts $\mathcal{L}_i$ that are satisfiable under L (denoted the *weighted goals* of i) to $\Re^+$ such that $\sum_{\phi\in\mathcal{L}_i}\Phi_i(\phi)=1$.

We give an example of a Boolean dl-game with weighted goals.

Example 4.1 (travel negotiation cont'd). A two-agent Boolean dl-game with weighted goals $G' = (L', N', \mathcal{A}', \pi', \Sigma', \Phi')$ for the travel negotiation example is obtained from the two-agent Boolean dl-game $G = (L, N, \mathcal{A}, \pi, \Sigma, \Phi)$ of Example 3.1 as follows:

(1) $L' = L$.
(2) $N' = N$.
(3) $\mathcal{A}'$ consists of the atomic concepts in $\mathcal{A}$ and the following new ones:

$$\begin{aligned} &\mathit{TP} \equiv \exists \mathit{hasActivity}.\mathit{ThemePark};\\ &\mathit{SS} \equiv \exists \mathit{hasActivity}.\mathit{Sightseeing};\\ &\mathit{HK} \equiv \exists \mathit{hasActivity}.\mathit{Hiking}.\end{aligned}$$

(4) Agents 1 and 2 control the following concepts $\pi(1)$ and $\pi(2)$, respectively:

$$\begin{aligned} &\pi(1) = \{U, R, \mathit{BHD}, \mathit{SS}, \mathit{HK}\};\\ &\pi(2) = \{\mathit{BA}, H, \mathit{BB}, \mathit{NP}, C, \mathit{TP}\}.\end{aligned}$$

More concretely, compared to Example 3.1, the agents now control more variables, namely, *Sightseeing* and *Hiking* for agent 1, and *ThemePark* for agent 2.

(5) Agents 1 and 2 have the following strict goals $\Sigma(1)$ and $\Sigma(2)$, respectively:

$$\begin{aligned} &\Sigma(1) = (U \sqcup R) \sqcap (H \sqcup \mathit{BB}) \sqcap \mathit{BHD};\\ &\Sigma(2) = (\mathit{NP} \sqcup C) \sqcap \geqslant 1\, \mathit{hasActivity}.\end{aligned}$$

More specifically, compared to Example 3.1, the agents 1 and 2 now also require *BudgetHotelDestination* and $\geqslant 1$ *hasActivity*, respectively, in the strict goals. Informally, agent 1 also wants a budget hotel destination, while agent 2 wants to include in the travel package that she is trying to sell at least one activity.

(6) Agents 1 and 2 have the following weighted goals Φ_1 and Φ_2, respectively,

$$\begin{aligned} &\Phi_1(\mathit{FamilyDestination}) &&= 0.3;\\ &\Phi_1(\mathit{RelaxDestination}) &&= 0.3;\\ &\Phi_1(\exists \mathit{hasDestination}.(\mathit{Capital} \sqcup \mathit{RuralArea}) \sqcap \\ &\qquad\qquad \exists \mathit{hasActivity}.(\mathit{Sport} \sqcap \mathit{ThemePark})) &&= 0.4;\\[1ex] &\Phi_2(\exists \mathit{hasDestination}.\mathit{RuralArea} \sqcap \exists \mathit{hasActivity}.\mathit{Sightseeing}) &&= 0.3;\\ &\Phi_2(\mathit{FamilyDestination} \sqcap \exists \mathit{hasActivity}.\mathit{ThemePark}) &&= 0.3;\\ &\Phi_2(\mathit{RelaxDestination} \sqcap \exists \mathit{hasActivity}.\mathit{Hiking}) &&= 0.4.\end{aligned}$$

Informally, agent 1 would like either (a) a family destination, or (b) a relax destination, or (c) a capital or rural destination with sports activities in a theme park, the latter with a slightly higher weight. Whereas agent 2 would like to sell either (a) a destination in a rural area with sightseeing, or (b) a family destination with theme park, or (c) a relax destination with hiking, the latter with slightly higher weight.

The notions of strategies and strategy profiles along with the consistency of strategy profiles are defined in the same way as for Boolean dl-games with binary goals. The following definition extends the notion of utility to weighted goals.

	$BA \sqcap H \sqcap \neg BB \sqcap NP \sqcap \neg C \sqcap TP$	$BA \sqcap H \sqcap \neg BB \sqcap \neg NP \sqcap C \sqcap TP$	$BA \sqcap H \sqcap \neg BB \sqcap NP \sqcap \neg C \sqcap \neg TP$	$BA \sqcap H \sqcap \neg BB \sqcap \neg NP \sqcap C \sqcap \neg TP$
$U \sqcap \neg R \sqcap BHD \sqcap SS \sqcap HK$	$(-1,-1)$	$(\mathbf{0.7},\mathbf{0.3})$	$(-1,-1)$	$(0.4,0)$
$\neg U \sqcap R \sqcap BHD \sqcap SS \sqcap HK$	$(\mathbf{1},\mathbf{1})$	$(-1,-1)$	$(0.7,0.7)$	$(-1,-1)$
$U \sqcap \neg R \sqcap BHD \sqcap SS \sqcap \neg HK$	$(-1,-1)$	$(0.4,0)$	$(-1,-1)$	$(0,0)$
$\neg U \sqcap R \sqcap BHD \sqcap SS \sqcap \neg HK$	$(0.7,0.3)$	$(-1,-1)$	$(0.3,0.3)$	$(-1,-1)$
$U \sqcap \neg R \sqcap BHD \sqcap \neg SS \sqcap HK$	$(-1,-1)$	$(0.4,0)$	$(-1,-1)$	$(\mathbf{0.4},\mathbf{0})$
$\neg U \sqcap R \sqcap BHD \sqcap \neg SS \sqcap HK$	$(0.4,0)$	$(-1,-1)$	$(0.4,0)$	$(-1,-1)$

Fig. 4. Normal form of a two-agent Boolean dl-game with weighted generalized goals.

Definition 4.2 (utilities with weighted goals). Let $G = (L, N, \mathcal{A}, \pi, \Phi, \Sigma)$ be an n-agent Boolean dl-game with weighted goals. Then, the *utility* to agent $i \in N$ under I, denoted $u_i(I)$, is defined as follows:

$$u_i(I) \;=\; \begin{cases} -1 & \text{if } I \text{ is inconsistent, or } I \not\models_L \Sigma(i); \\ \Sigma_{\phi \in \mathcal{L}_i,\, I \models_L \phi}\, \Phi_i(\phi) & \text{if } I \text{ is consistent, } I \models L, \text{ and } I \models_L \Sigma(i). \end{cases}$$

We give an example to illustrate the utilities in the case of weighted goals.

Example 4.2 (travel negotiation cont'd). The normal form representation of the two-agent Boolean dl-game with weighted goals G of Example 4.1 is depicted in Fig. 4. Its only (pure) Nash equilibria are given by the bold entries in Fig. 4. Observe that the Nash equilibrium with utility pair $(1,1)$ is also Pareto-optimal.

5 Controlling Roles

In this section, we present a further generalization of Boolean dl-games where agents control roles instead of concepts. In this case, every strategy is intuitively an instantiation of concepts. We also provide a further application scenario from web service negotiation, along which we sketch this generalization of Boolean dl-games.

Example 5.1 (web service negotiation). Consider a service negotiation scenario, where a service provider (agent 2) and a service requester (agent 1) are negotiating on the conditions of a supply. The description logic knowledge base L is given by the ontology in Fig. 5. We assume the set of two agents $N = \{1,2\}$. The roles $\pi(1)$ and $\pi(2)$ controlled by agents 1 and 2, respectively, are given as follows:

$\pi(1) = \{\mathit{delivery}, \mathit{hasQuality}\}$;
$\pi(2) = \{\mathit{hasType}\}$.

$EU \sqsubseteq WorldWide;$
$US \sqsubseteq WorldWide;$
$Contract1 \sqsubseteq Contract;$
$Contract2 \sqsubseteq Contract;$
$Contract1 \sqsubseteq \neg Contract2;$
$Cash \sqsubseteq PaymentType;$
$Instalments \sqsubseteq PaymentType;$
$HighQualityService \sqsubseteq \exists assistance \sqcap \forall assistance.Onsite \sqcap {=}2\ year\_guarantee;$
$LowQualityService \sqsubseteq \exists assistance \sqcap \forall assistance.Phone \sqcap {=}1\ year\_guarantee;$
$Contract1 \equiv \exists payment \sqcap \forall payment.Instalments \sqcap \exists delivery \sqcap \forall delivery.(US \sqcap EU);$
$Contract2 \equiv \exists payment \sqcap \forall payment.Cash \sqcap \exists delivery \sqcap \forall delivery.WorldWide.$

Fig. 5. Service ontology.

	$C1$	$C2$
$HQ \sqcap WW$	$(-1,-1)$	$(\mathbf{0},\mathbf{1})$
$HQ \sqcap SE$	$(1,0)$	$(\mathbf{0},\mathbf{1})$

Fig. 6. Normal form of a two-agent Boolean dl-game with controlled roles.

Agents 1 and 2 have the following goals $\Phi(1)$ and $\Phi(2)$, respectively (for ease of presentation, we omit strict and weighted goals here):

$$\begin{aligned}\Phi(1) &= \exists payment \sqcap \forall payment.Instalments;\\ \Phi(2) &= (\exists hasQuality \sqcap \forall hasQuality.LowQualityService \sqcap\\ &\quad \exists hasType \sqcap \forall hasType.Contract1) \sqcup\\ &\quad (\exists hasQuality \sqcap \forall hasQuality.HighQualityService \sqcap\\ &\quad \exists hasType \sqcap \forall hasType.Contract2).\end{aligned}$$

The normal form of the two-agent Boolean dl-game is depicted in Fig. 6, where (for the sake of conciseness) we define the following atomic concepts:

$$\begin{aligned}C1 &\equiv \exists hasType \sqcap \forall hasType.Contract1;\\ C2 &\equiv \exists hasType \sqcap \forall hasType.Contract2;\\ HQ &\equiv \exists hasQuality \sqcap \forall hasQuality.HighQualityService;\\ WW &\equiv \exists delivery \sqcap \forall delivery.WorldWide;\\ SE &\equiv \exists delivery \sqcap \forall delivery.(US \sqcap EU).\end{aligned}$$

Notice that in this approach agents do not have to enumerate all the possible combinations of concepts they control, as before, but, as they control roles instead of concepts, it is enough to consider only concepts that they are interested in, such as e.g. for agent 1 *HighQualityService* or for agent 2 only the type of contracts she wants to offer. This approach is surely more compact than the previous one, even if it could be not exhaustive and give more power w.r.t. some attributes to one agent, the one controlling the role indeed can control an entire set of attributes, e.g., thanks to the control on *hasType*, agent 2 is the only one that can choose what type of contract to offer.

6 Related Work

A large number of negotiation mechanisms have been proposed and studied in the literature. It is possible to distinguish, among others, game-theoretic ones [10,11], heuristic-based approaches [12,13] and logic-based approaches. Although pure game-theoretic and heuristic-based approaches are highly suitable for a wide range of applications, they have some limitations and disadvantages. Often in game-theoretic approaches, it is assumed that no relation exists between agent's strategies and that all the combinations of strategies are possible. Moreover, they usually do not model relations about issues, which is, instead, fundamental in multi-attribute negotiation. On the other hand, heuristic-based approaches use empirical evaluations to find an agreement, which can be sub-optimal, as they do not explore the entire space of possible outcomes.

In the following, we give a brief overview of logic-based approaches to automated negotiation, comparing our approach to existing ones and highlighting relevant differences. There is an extensive literature on argumentation-based negotiation [14,15,16]. In these approaches, an agent can accept/reject/critique a proposal of its opponent, so agents can argue about their beliefs, given their desires and so pursue their intentions. With respect to our framework, these approaches require a larger number of communication rounds in order to exchange information, while our approach is a one-shot negotiation, which ensures the termination after only one round; indeed in argumentation-based frameworks, usually, agent interactions go back and forth for multiple rounds.

Several recent logic-based approaches to negotiation are based on propositional logic. Bouveret et al. [17] use weighted propositional formulas (WPFs) to express agent 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 discover inconsistencies between strategies, as well as attributes, or find out if an agent preference is implied by a combination of strategies (an interpretation) which is fundamental to model a multi-attribute negotiation. Chevaleyre et al. [18] classify utility functions expressed through WPFs 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 formulas. Zhang and Zhang [19] adopt a kind of propositional knowledge base arbitration to choose a fair negotiation outcome. However, *common knowledge* is considered as just more entrenched preferences, that could be even dropped in some deals. Instead, the logical constraints in our ontology must *always* be enforced in the negotiation outcomes. Wooldridge and Parsons [20] define an agreement as a model for a set of formulas from both agents. However, Wooldridge and Parsons [20] only study multiple-rounds protocols and the approach leaves the burden to reach an agreement to the agents themselves, although they can follow a protocol. The approach does not take preferences into account, so that it is not possible to compute utility values and check if the reached agreement is Pareto-optimal or a Nash equilibrium. In the work by Ragone et al. [21], a basic propositional logic framework endowed with an ontology was proposed, which is further extended in [22], introducing the extended logic $\mathcal{P}(\mathcal{N})$ (a propositional logic with concrete domains), thus handling numerical features, and showed how to compute Pareto-optimal agreements, by solving an optimization problem and adopting a one-shot negotiation protocol.

For what concerns approaches using more expressive ontology languages, namely, description logics, there is the work by Ragone et al. [23], which although uses a rather inexpressive description logic, $\mathcal{ALEH}(D)$, proposes a semantic-based alternating-offers protocol exploiting non-standard inference services, as concept contraction, and utility theory to find the most suitable agreements. Concept contraction can be useful to provide an explanation of "what is wrong" between request and offer, that is, the reason why agents cannot reach an agreement and *what* has to be given up in order to reach that. Furthermore, differently from our approach, no game-theoretic analysis is provided about Nash equilibria, even if in this framework, agents do not have to reveal their utilities to the opponent. Another work exploits description logics in negotiation scenarios [24], where the more expressive $\mathcal{SHOIN}(\mathbf{D})$ is used to model the logic-based negotiation protocol; a scenario with *fully* incomplete information is studied, where agents do not know anything about the opponent (neither preferences nor utilities). Furthermore, also this framework lacks a game-theoretic analysis about Nash equilibria.

7 Summary and Outlook

Towards automated multi-attribute negotiation in the Semantic Web, we have introduce Boolean description logic games, which combine classical Boolean games with expressive description logics. As further generalizations of classical Boolean games, they also include strict agent requirements and overlapping agent control assignments. We have also considered two generalizations, one with weighted goals, which may be defined over free description logic concepts, and one where the agents own roles rather than concepts. Furthermore, formulations of a travel and a service negotiation scenario have given evidence of the practical usefulness of our approach.

An interesting topic for future research is to more deeply analyze the semantic and the computational properties of Boolean description logic games. In particular, an important issue is the development of algorithms for computing optimal strategy profiles, and the analysis of its computational complexity. Furthermore, it would be interesting to implement a tool for solving Boolean dl-games and testing it on negotiation scenarios. Another topic for future research is a generalization to qualitative conditional preference structures, such as the ones expressed through CP-nets [25]. From a larger perspective, Boolean dl-games aim at a centralized one-step negotiation process, where the agents reveal their preferences to a central mediator, which then calculates one optimal strategy for each agent. In this framework, it is important to study how it is possible to avoid that the agents report untruthful preferences in order to obtain better strategies, which is touching the problem of mechanism design [26].

Acknowledgments. Azzurra Ragone wishes to thank Francesco M. Donini and Tommaso Di Noia for fruitful suggestions and discussions. This work has been partially supported by the Apulia Region funded project PS_092 Distributed Production to Innovative System (DIPIS), by the Austrian Science Fund (FWF) under the project P18146-N04, and by the German Research Foundation (DFG) under the Heisenberg Programme.

References

1. Jennings, N.R., Faratin, P., Lomuscio, A.R., Parsons, S., Wooldridge, M., Sierra, C.: Automated negotiation: Prospects, methods and challenges. Group Decision and Negotiation **10** (2001) 199–215

2. Berners-Lee, T.: Weaving the Web. Harper, San Francisco, CA (1999)
3. Fensel, D., Wahlster, W., Lieberman, H., Hendler, J., eds.: Spinning the Semantic Web: Bringing the World Wide Web to Its Full Potential. MIT Press (2002)
4. Hendler, J.A.: Where are all the intelligent agents? IEEE Intelligent Systems **22**(3) (2007) 2–3
5. Horrocks, I., Patel-Schneider, P.F.: Reducing OWL entailment to description logic satisfiability. In: Proc. ISWC-2003. Volume 2870 of LNCS, Springer (2003) 17–29
6. Horrocks, I., Sattler, U., Tobies, S.: Practical reasoning for expressive description logics. In: Proc. LPAR-1999. Volume 1705 of LNCS, Springer (1999) 161–180
7. Bonzon, E., Lagasquie-Schiex, M.C., Lang, J., Zanuttini, B.: Compact preference representation and Boolean games. Autonomous Agents and Multi-Agent Systems (2008) In press.
8. Harrenstein, P., van der Hoek, W., Meyer, J.J.C., Witteveen, C.: Boolean games. In: Proc. TARK-2001, Morgan Kaufmann (2001) 287–298
9. Harrenstein, P.: Logic in Conflict. PhD thesis, Utrecht University, The Netherlands (2004)
10. Kraus, S.: Strategic Negotiation in Multiagent Environments. MIT Press (2001)
11. Rosenschein, J.S., Zlotkin, G.: Rules of Encounter. MIT Press (1994)
12. Fatima, S., Wooldridge, M., Jennings, N.R.: Optimal agendas for multi-issue negotiation. In: Proc. AAMAS-2003, ACM Press (2003) 129–136
13. Faratin, P., Sierra, C., Jennings, N.R.: Using similarity criteria to make issue trade-offs in automated negotiations. Artificial Intelligence **142**(2) (2002) 205–237
14. Rahwan, I., Ramchurn, S.D., Jennings, N.R., McBurney, P., Parsons, S.: Argumentation-based negotiation. The Knowledge Engineering Review **18**(4) (2003) 343–375
15. Sadri, F., Toni, F., Torroni, P.: Dialogues for negotiation: Agent varieties and dialogue sequences. In: Intelligent Agents VIII. Volume 2333 of LNCS, Springer (2002) 405–421
16. Bentahar, J., Moulin, B., Meyer, J.J.C., Chaib-draa, B.: A modal semantics for an argumentation-based pragmatics for agent communication. In: Argumentation in Multi-Agent Systems. Volume 3366 of LNCS, Springer (2005) 44–63
17. Bouveret, S., Lemaitre, M., Fargier, H., Lang, J.: Allocation of indivisible goods: A general model and some complexity results. In: Proc. AAMAS-2005, ACM Press (2005) 1309–1310
18. Chevaleyre, Y., Endriss, U., Lang, J.: Expressive power of weighted propositional formulas for cardinal preference modeling. In: Proc. KR-2006, AAAI Press (2006) 145–152
19. Zhang, D., Zhang, Y.: A computational model of logic-based negotiation. In: Proc. AAAI-2006, AAAI Press (2006) 728–733
20. Wooldridge, M., Parsons, S.: Languages for negotiation. In: Proc. ECAI-2000, IOS Press (2000) 393–400
21. 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. ECAI-2006, IOS Press (2006) 230–234
22. Ragone, A., Di Noia, T., Di Sciascio, E., Donini, F.M.: Logic-based automated multi-issue bilateral negotiation in peer-to-peer e-marketplaces. Autonomous Agents and Multi-Agent Systems **16**(3) (2008) 249–270
23. Ragone, A., Di Noia, T., Di Sciascio, E., Donini, F.M.: Alternating-offers protocol for multi-issue bilateral negotiation in semantic-enabled marketplaces. In: Proc. ISWC-2007. Volume 4825 of LNCS. Springer (2007) 395–408
24. Ragone, A., Di Noia, T., Di Sciascio, E., Donini, F.M.: Description logics for multi-issue bilateral negotiation with incomplete information. In: Proc. AAAI-2007, AAAI Press (2007) 477–482
25. Boutilier, C., Brafman, R.I., Domshlak, C., Hoos, H.H., Poole, D.: CP-nets: A tool for representing and reasoning with conditional ceteris paribus preference statements. J. Artif. Intell. Res. **21** (2004) 135–191
26. Sandholm, T.: Computing in mechanism design. In: New Palgrave Dictionary of Economics. (2008)