FORMALIZING DIALECTICAL REASONING FOR COMPROMISE-BASED JUSTIFICATION

Hiroyuki Kido, Katsumi Nitta
Interdisciplinary Graduate School of Science and Engineering, Tokyo Institute of Technology, Tokyo, Japan

Masahito Kurihara
Graduate School of Information and Science, Hokkaido University, Sapporo, Japan

Daisuke Katagami
Faculty of Engineering, Tokyo Polytechnic University, Tokyo, Japan

Keywords: Argument-based reasoning, Compromise, Dialectical thought, Deliberation.

Abstract: Chinese traditional philosophy regards dialectics as a style of reasoning that focuses on contradictions and how to resolve them, transcend them or find the truth in both. Compromise is considered to be one possible way to resolve conflicts dialectically. In this paper, we formalize dialectical reasoning as a way for deriving compromise. Both the definition of the notion of compromise and the algorithm for dialectical reasoning are proposed on an abstract complete lattice. We prove that the dialectical reasoning is sound and complete with respect to the compromise. We propose the concrete algorithm for dialectical reasoning characterized by definite clausal language and generalized subsumption. The algorithm is proved to be sound with respect to the compromise. Furthermore, we expand an argumentation system to handle compromise arguments, and illustrate that an agent bringing up a compromise argument realizes a compromise based justification towards argument-based deliberation.

1 INTRODUCTION

Argumentation in artificial intelligence, often called computational dialectics, is rooted in Aristotle's idea of evaluating argumentation in a dialogue model (Hamblin, 1970). In contrast, there exist various definitions of dialectics in history (Rescher, 2007) and it leads to various interpretations in various areas today. For instance, psychologist Nisbett interprets dialectics as a style of reasoning intended to find a middle way (Nisbett, 2003), some logicians as formal logic disrespecting the law of noncontradiction (Carnielli et al., 2007), and some computer scientists as the study of systems mediating discussions and arguments between agents, artificial and human (Gordon, 1995). In particular, Nisbett pointed out that there is a style of reasoning in Eastern thought, traceable to the ancient Chinese, which has been called dialectical, meaning that it focuses on contradictions and how to resolve them or transcend them or find the truth in both. His experiments showed that, compared with Westerners, Easterners have a greater preference for compromise solutions and for holistic arguments, and they are more willing to endorse both of two apparently contradictory arguments. Moreover, he contrasted a logical approach and a dialectical approach for conflicting propositions, and pointed out that the former one would seem to require rejecting one of the propositions in favor of the other in order to avoid possible contradiction, and the latter one would favor finding some truth in both, in a search for the Middle Way. We think that the perspective opens up a new horizon for argumentation in artificial intelligence, especially in argument-based deliberation and negotiation, because argumentation is a prominent way for conflict detection, social decision making and consensus building, and the latter two cannot be achieved without such kinds of thought. However, there is little work on computational argumentation directed to dialectical conflict resolution we mentioned above.

In this paper, we formalize dialectical reasoning as a way for deriving a compromise. This is based on the knowledge that a compromise is a possible way for realizing dialectical conflict resolution, and our

Kido H., Nitta K., Kurihara M. and Katagami D..
FORMALIZING DIALECTICAL REASONING FOR COMPROMISE-BASED JUSTIFICATION.
DOI: 10.5220/0003181903550363
In *Proceedings of the 3rd International Conference on Agents and Artificial Intelligence* (ICAART-2011), pages 355-363
ISBN: 978-989-8425-40-9

idea that compromise mechanisms should be calculated through reasoning mechanism. We give a formal definition of compromise and an algorithm for dialectical reasoning both on an abstract complete lattice. We show that the algorithm is sound and complete with respect to the compromise. We propose a concrete and sound algorithm for dialectical reasoning characterized by definite clausal language and generalized subsumption on the assumption that arguments are constructed from knowledge bases. We expand the argumentation system (Prakken, 1997) to handle compromise arguments and illustrate that compromise arguments realize compromise-based justification towards argumentation for deliberation.

This paper is organized as follows. Section 2 gives a motivational example of dialectical thought and Section 3 shows preliminaries. Section 4 defines a notion of compromise and Section 5 proposes both abstract and concrete algorithms for dialectical reasoning. Section 6 expands an argumentation system and illustrates the effectiveness of compromise arguments. Section 7 discusses the related works, and Section 8 describes the conclusion and future works.

2 MOTIVATIONAL EXAMPLE

Let us consider the simple deliberative dialogue by which agents A and B try to decide which camera to buy. They are assumed to have their individual knowledge bases from which they make arguments. The worst situation for them is assumed that they cannot buy any camera.

A : I want to buy 'a' because it is a compact and light camera. (A_1 =[$compact(a)$, $light(a)$, $camera(a)$, $\forall x.compact(x) \wedge light(x) \wedge camera(x) \rightarrow buy(x)$, $buy(a)$].)

B : We cannot buy 'a' because it is out of stock. ($B_1 = [\neg inStock(a), \forall x.\neg inStock(x) \rightarrow \neg buy(x), \neg buy(a)]$.)

B : I want to buy 'b' because it is high-resolution camera with a long battery life. ($B_2 = [resolution(b, high), battery(b, long), camera(b), \forall x.resolution(x, high) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x), buy(b)]$.)

A : We cannot buy 'b' because it is beyond our budget. ($A_2 = [overBudget(b), \forall x.overBudget(x) \rightarrow \neg buy(x), \neg buy(b)]$.)

Each A_1, A_2, B_1, and B_2 is the argument that is a formal description of the informal statement preceding the argument. Both B_1 and A_1, and A_2 and B_2 defeat each other due to their logical inconsistency. Further, A_1 and B_2 defeat each other due to the existence of alternatives. In this situation, neither A_1 nor B_2 is justified by argumentation semantics, in other words, neither A_1 nor B_2 is nonmonotonic consequence. This leads to the evaluation that both the options $buy(a)$ and $buy(b)$ are unacceptable as a choice of A and B and it is hard to prioritize these options within the scope of logic. Of course, outside the scope of logic, decision theory and game theory allow agents to prioritize these options in different ways. However, it is the case that they have to choose from the given options under the situation that neither A nor B has any other alternative options. On the other hand, in real life, we often work out new options by giving up some of our concerns and try to break such a stalemate peacefully. Let us consider the following argument.

A : How about 'c' because it is user-friendly camera with a long battery life. ($A_3 = [userFriendly(c), battery(c, long), camera(c), userFriendly(x) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x), buy(c)]$.)

The option $buy(c)$ concluded by A_3 reflects each agent's concerns partly, not completely, under the background knowledge $compact(x) \wedge light(x) \rightarrow userFriendly(x)$. For A, A_3 reflects user-friendliness derived from the knowledge and the attributes that A's initial option $buy(a)$ has. For B, A_3 reflects height of resolution that B's initial option $buy(b)$ has. Arguments are constructed from knowledge bases using various kinds of reasoning. From the viewpoint of logic, the question we are interested in here is: What type of reasoning is needed to make the argument with the compromise? Obviously, induction and abduction do not address the compromise because they aim to make up for lacks of knowledge so as to explain all of given examples. In the above dialogue, one possible inductive hypothesis is a general rule $\forall x.camera(x) \rightarrow buy(x)$ when, for example, $buy(a)$ and $buy(b)$ are assumed to be examples. The rule r : $\forall x.userFriendly(x) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x)$ that allows A to make A_3 is not a hypothesis of these reasoning, and therefore, the derivation of the rule is outside of the scope of these reasoning. Deduction neither address the compromise because it aims to derive all necessary conclusions of given theory. In the above dialogue, deduction only derives the cameras satisfying each agent's concerns completely, not partly. Therefore, deriving $buy(c)$ is outside the scope of deductive reasoning. We think agents need another type of reasoning that finds a middle ground among agents' concerns such as r. We think that the studies of such reasoning open up a new horizon for argumentation in artificial intelligence especially in argument-based deliberation or negotiation.

3 PRELIMINARIES

A complete lattice is a 2-tuple of a set and a binary relation of the set. Both of them are abstract in the sense that the internal structures are unspecified.

Definition 1 (Complete Lattice)**.** *Let $<\mathcal{L},\succeq>$ be a quasi-ordered set. If for every $S \subseteq \mathcal{L}$, a least upper bound of S and a greatest lower bound of S exist, then $<\mathcal{L},\succeq>$ is called a complete lattice.*

The abstract argumentation framework (Dung, 1995), denoted by *AF*, gives a general framework for nonmonotonic logics. The framework allows us to define various semantical notions of argumentation extensions that intended to capture various types of nonmonotonic consequences. The basic formal notions, with some terminological changes, are as follows.

Definition 2. *(Dung, 1995) The abstract argumentation framework is defined as a pair $AF =< AR, defeat >$ where AR is a set of arguments, and defeat is a binary relation on AR, i.e. $defeat \subseteq AR \times AR$.*

- *A set S of arguments is said to be conflict-free if there are no arguments A, B in S such that A defeats B.*
- *An argument $A \in AR$ is acceptable with respect to a set S of arguments iff for each argument $B \in AR$: if B defeats A then B is defeated by an argument in S.*
- *A conflict-free set of arguments S is admissible iff each argument in S is acceptable with respect to S.*
- *A preferred extension of an argumentation framework AF is a maximal (with respect to set inclusion) admissible set of AF.*

An argument is justified with respect to *AF* if it is in every preferred extension of *AF*, and is defensible with respect to *AF* if it is in some but not all preferred extensions of *AF* (Prakken and Sartor, 1997).

The argumentation system (Prakken, 1997) uses Reiter's default logic (Reiter, 1980) for defining internal structures of arguments and defeat relations in *AF*. The language consists of a first-order language $\mathcal{L}_0$ and a set of defeasible rules Δ defined below. Informally, $\mathcal{L}_0$ is assumed to be divided into two subsets; one is the set $\mathcal{F}_c$ of contingent facts and the set $\mathcal{F}_n$ of necessary facts. A default theory is a set $\mathcal{F}_c \cup \mathcal{F}_n \cup \Delta$ where $\mathcal{F}_n \cup \mathcal{F}_c$ is consistent. We extract some necessary definitions of the argumentation system (Prakken, 1997).

Definition 3. *(Prakken, 1997)*

- *Let $\varphi_1,\dots,\varphi_n,\psi \in \mathcal{L}_0$. A defeasible rule is an expression of the form*

  $$\varphi_1 \wedge \cdots \wedge \varphi_j \wedge \sim \varphi_k \wedge \cdots \wedge \sim \varphi_n \Rightarrow \psi$$

  $\varphi_1 \wedge \cdots \wedge \varphi_j$ is called the antecedent, $\sim \varphi_k \wedge \cdots \wedge \sim \varphi_n$ is called the justification and ψ is called the consequent of the rule. For any expression $\sim \varphi_i$ in the justification of a defeasible rule, $\neg\varphi_i$, classical negation of φ_i, is an assumption of the rule. And an assumption of an argument is an assumption of any rule in the argument.
- *Let $\varphi_1,\dots,\varphi_n,\psi \in \mathcal{L}_0$. Defeasible modus ponens, denoted by DMP, is an inference rule of the form*

$$\frac{\varphi_1 \wedge \cdots \wedge \varphi_j \wedge \sim \varphi_k \wedge \cdots \wedge \sim \varphi_n \Rightarrow \psi \quad \varphi_1 \wedge \cdots \wedge \varphi_j}{\psi}$$

- *Let Γ be a default theory. An argument based on Γ is a sequence of distinct first-order formulae and/or ground instances of defaults $[\varphi_1,\dots,\varphi_n]$ such that for all φ_i:*
  - *$\varphi_i \in \Gamma$; or*
  - *There exists an inference rule $\psi_1,\dots,\psi_m/\varphi_i \in \mathcal{R}$ such that $\psi_1,\dots,\psi_m \in \{\varphi_1,\dots,\varphi_{i-1}\}$.*

  For argument A, φ is a conclusion of A, denoted by $\varphi \in CONC(A)$, if φ is a first-order formula in A. φ is an assumption of A, denoted by $\varphi \in ASS(A)$, if φ is an assumption of a default in A.
- *Let A_1 and A_2 be two arguments.*
  - *A_1 rebuts A_2 iff $CONC(A_1) \cup CONC(A_2) \cup \mathcal{F}_n \vdash \perp$ and A_2 is defeasible and A_1 is strict.*
  - *A_1 undercuts A_2 iff $CONC(A_1) \cup \mathcal{F}_n \vdash \neg\phi$ and $\phi \in ASS(A_2)$.*

In the definition of defeasible rule, special symbol $\sim$, called weak negation, is introduced in order to represent unprovable propositions. It makes that the language of the system has the full expressiveness of default logic. In *DMP*, assumptions in defeasible rules are ignored, and the ignorance can be correctly disabled by undercutting. An argument is a deduction incorporating default reasoning using defeasible rules and $DMP \in \mathcal{R}$. $\mathcal{R}$ is assumed to consist of all valid first-order inference rules plus *DMP*. We say that an argument is strict if there exist only valid first-order inference rules in the argument. Otherwise, the argument is defeasible. Symbols $\perp$ and $\vdash$ represent a logical contradiction and a logical consequence relation, respectively. Rebutting caused by priorities of defeasible rules is excluded from original definition.

Definition 2 takes no account of the aspect of proof theory that gives a way to determine whether an individual argument is justified or defensible. We extract some necessary definitions of the proof theory for argumentation system, called the dialectical proof theory (Prakken, 1999).

Definition 4. *(Prakken, 1999)*

- *A dialogue is a finite nonempty sequence of moves* $moves_i = (Player_i, Arg_i)$ $(i > 0)$*, such that*
  1. $Player_i = P$ *iff i is odd; and* $Player_i = O$ *iff i is even;*
  2. *If* $Player_i = Player_j = P$ *and* $i \neq j$*, then* $Arg_i \neq Arg_j$*;*
  3. *If* $Player_i = P (i > 1)$*, then* Arg_i *strictly defeats* Arg_{i-1}*;*
  4. *If* $Player_i = O$*, then* Arg_i *defeats* Arg_{i-1}*.*
- *A dialogue tree is a tree of dialogues such that if* $Player_i = P$ *then the children of* $move_i$ *are all defeaters of* $Args_i$
- *A player wins a dialogue if the other player cannot move. And a player wins a dialogue tree if it wins all branches of the tree.*
- *An argument A is a provably justified argument if there is a dialogue tree with A as its root, and won by the proponent.*

The condition 1 in the definition of a dialogue requires that the proponent begins and then the players take turns. Condition 2 prevents the proponent from repeating moves and condition 3 and 4 are burdens of proof for P and O. In the definition of a dialogue tree, all defeaters for every arguments of P are considered. The idea of the definition of win is that if P's last argument is undefeated, it reinstates all previous arguments of P that occur in the same branch of a tree, in particular the root or the tree. It is proved that arguments are justified iff the arguments are provably justified (Prakken, 1999).

4 SEMANTICS FOR REASONING

In this section, we give a declarative definition of reasoning. Figure 1 shows the classification of our conflict handling modes into 5 groups (Thomas, 1992). The vertical and the horizontal axes represent the strength of assertiveness and that of cooperativeness, respectively. Competition, collaboration, avoiding, and accommodation are explained as assertive and uncooperative mode, assertive and cooperative mode, unassertive and uncooperative mode, and unassertive and cooperative mode, respectively. Compromise is a mode taking the middle attitude among the four. In terms of importance in computational argumentation, we focus on the notion of compromise and give more concrete interpretation that compromise is a statement satisfying each agent's statement at least partly, not completely. Our idea here is that we formally define the notion of compromise using a complete lattice. We impose two conditions, incompleteness and relevance, as fundamental requirements to be compromise. The incompleteness requires that compromise must not satisfy each agent's statement completely and the relevance requires that compromise must satisfy each agent's statement at least partly. Further, we impose additional two conditions, collaborativeness and simplicity, in order to capture our intuitions about compromise. The collaborativeness requires that compromise must retain a common ground that all agents commonly have in advance. The simplicity requires that compromise does not include any redundant statements.

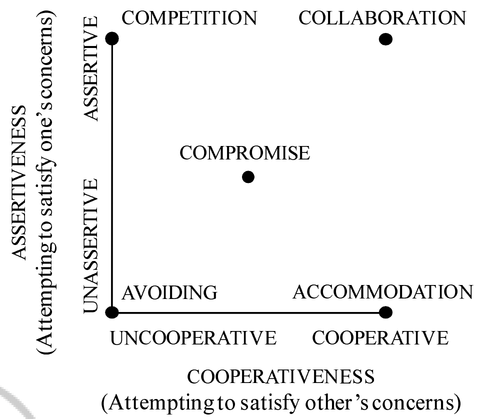

Figure 1: Two-dimensional taxonomy of conflict handling modes (Thomas, 1992).

Definition 5 (Compromise)**.** *Let* $< \mathcal{L}, \succeq >$ *be a complete lattice and* $X_1, \ldots, X_n, Y$ *be elements of* $\mathcal{L}$ *satisfying* $\inf\{X_i \mid 1 \leq i \leq n\} \nsim \bot$*.* Y *is a compromise among* $X_1, \ldots, X_{n-1}$*, and* X_n *iff*

1. *incompleteness:* $\forall X_i (Y \not\succeq X_i)$*; and*
2. *relevance:* $\forall X_i (\inf\{X_i, Y\} \nsim \bot)$*; and*
3. *collaborativeness:* $Y \succeq \inf\{X_j \mid 1 \leq j \leq n\}$*; and*
4. *simplicity:* $Y \sim \sup\{\inf\{X_i, Y\} \mid 1 \leq i \leq n\}$*.*

Definition 5 says that there is no compromise among $X_i (1 \leq i \leq n)$ if their common lower element is only bottom. Intuitively, the condition states that there is no common ground among $X_i (1 \leq i \leq n)$. The incompleteness states that Y is not upper than X_i, for all X_i. The relevance states that there exists a common nonbottom greatest lower bound of $\{X_i, Y\}$, for all X_i. The collaborativeness states that the common lower element of all X_i is also lower than Y and the simplicity states that any lower element of Y is common lower element of Y and X_i.

Example 1 (compromise based on entailment)**.** *Let* $\mathcal{L}$ *be propositional language that has all well-formed formulae composed of alphabets A and B, and* $\models$ *be a satisfiability relation on* $\mathcal{L}$*.* $< \mathcal{L}, \models >$ *is a complete lattice shown in Figure 2.* $\top$ *and* $\bot$ *denote*

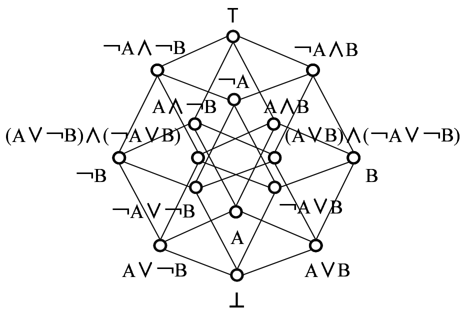

Figure 2: Complete lattice $< \mathcal{L}, \models >$.

false and true, respectively, in accordance with common usage in inductive logic programming. Definition 5 is characterized as follows: 1. $\forall X_i(Y \not\models X_i)$, *2.* $\forall X_i(\not\models X_i \vee Y)$, *3.* $Y \models X_1 \vee \cdots \vee X_n$, *and 4.* $Y \equiv (X_1 \vee Y) \wedge \cdots \wedge (X_n \vee Y)$. $X \equiv Y$ *denotes* $X \models Y$ *and* $Y \models X$. *Each of* $A \vee B, (A \vee B) \wedge (\neg A \vee \neg B)$, *and B is a compromise between* $\neg A \wedge B$ *and A.*

5 DIALECTICAL REASONING

5.1 Abstract Dialectical Reasoning

In this section, we give a procedural definition of dialectical reasoning that derives compromise. As with the definition of compromise, the algorithm is formalized on an abstract complete lattice $< \mathcal{L}, \succeq >$. The algorithm describes common procedures with which each concrete dialectical reasoning complies.

The inputs of Algorithm 1 are $X_1, \ldots, X_n \in \mathcal{L}$. Algorithm 1 calculates a set of lower elements of each $X_i (1 \leq i \leq n)$ at line 4. For every $Y \in \mathcal{Y}_i$, if Y satisfies the conditions given at line 6 then Y is collected in $\mathcal{W}_i$ at line 7. At line 12, the least upper bound of $\{W_i \mid 1 \leq i \leq n\}$ is calculated and collected in $\mathcal{Z}$. If the least upper bound satisfies the condition at line 16 then it is eliminated from $\mathcal{Z}$. In summary, the overall reasoning process has two primal phases: deriving lower elements Y_i of given $X_i (1 \leq i \leq n)$ and the common upper element Z of the lower elements $Y_i (1 \leq i \leq n)$. Note that we cannot detail the procedures at lines 4, 6, 12 and 16 in Algorithm 1 anymore because the complete lattice $< \mathcal{L}, \succeq >$ in Algorithm 1 is abstract. One of the algorithms with the concrete procedures for dialectical reasoning will be given in Algorithm 2. In Algorithm 1, we assume that these derivation and the comparison are computable, i.e., there exist algorithms that can return the right answers to the procedures at lines 4, 6, 12 and 16.

Algorithm 1: Dialectical Reasoning on $< \mathcal{L}, \succeq >$.

Require: $\inf\{X_i \mid 1 \leq i \leq n\} \not\sim \perp$
1: $\mathcal{Z} := \emptyset$
2: **for** $i := 1$ to n **do**
3: $\quad \mathcal{W}_i := \emptyset$
4: $\quad$ compute $\mathcal{Y}_i = \{Y \mid Y \preceq X_i\}$
5: $\quad$ **for all** $Y \in \mathcal{Y}_i$ **do**
6: $\quad\quad$ **if** $Y \not\sim \perp$ and $Y \succeq \inf\{X_j \mid 1 \leq j \leq n\}$ **then**
7: $\quad\quad\quad \mathcal{W}_i = \mathcal{W}_i \cup \{Y\}$
8: $\quad\quad$ **end if**
9: $\quad$ **end for**
10: **end for**
11: **for all** $(W_1, \ldots, W_n) \in \mathcal{W}_1 \times \ldots \times \mathcal{W}_n$ **do**
12: $\quad$ compute $Z \sim \sup\{W_i \mid 1 \leq i \leq n\}$
13: $\quad \mathcal{Z} = \mathcal{Z} \cup \{Z\}$
14: **end for**
15: **for** $i := 1$ to n **do**
16: $\quad$ **if** $Z \succeq X_i$ **then**
17: $\quad\quad \mathcal{Z} = \mathcal{Z} \setminus \{Z\}$
18: $\quad$ **end if**
19: **end for**
20: **return** $\mathcal{Z}$

Theorem 1 (Soundness and Completeness)**.** *Let* $< \mathcal{L}, \succeq >$ *be a complete lattice and* $X_1, \ldots, X_n, Z \in \mathcal{L}$ *that satisfy* $\inf\{X_i \mid 1 \leq i \leq n\} \not\sim \perp$. *Z is a compromise among* $X_1, \ldots, X_{n-1}$, *and* X_n *iff Z is an element of the output of Algorithm 1 where the inputs are* $X_1, \ldots, X_{n-1}$, *and* X_n.

Proof. (Soundness) If Z is an output of Algorithm 1, Z obviously satisfies the condition of incompleteness due to the condition at line 16 in Algorithm 1. Further, if Z is an output of Algorithm 1, there exists at least one $Y_i \preceq X_i$ for all X_i such that $Y_i \not\sim \perp, Y_i \succeq \inf\{X_j \mid 1 \leq j \leq n\}$, and $Z \sim \sup\{Y_i \mid 1 \leq i \leq n\}$. Now we let $W_i \sim Y_i$. Since $Z \sim \sup\{W_i \mid 1 \leq i \leq n\}$ and $X_i \succeq W_i$, $\inf\{X_i, Z\} \succeq W_i$. Since $\inf\{X_i, Z\} \succeq W_i$ and $W_i \not\sim \perp$, $\inf\{X_i, Z\} \not\sim \perp$ for all X_i. Therefore, relevance holds. Since $\inf\{X_i, Z\} \succeq W_i$ and $W_i \succeq \inf\{X_j \mid 1 \leq j \leq n\}$, $\inf\{X_i, Z\} \succeq \inf\{X_j \mid 1 \leq j \leq n\}$ for all X_i. Therefore, collaborativeness holds. Since $Z \succeq \inf\{X_i, Z\}$ for all X_i, $Z \succeq \sup\{\inf\{X_i, Z\} \mid 1 \leq i \leq n\}$. On the other hand, since $Z \sim \sup\{W_i \mid 1 \leq i \leq n\}$ and $\inf\{X_i, Z\} \succeq W_i$ for all X_i, $Z \preceq \sup\{\inf\{X_i, Z\} \mid 1 \leq i \leq n\}$. Hence, $Z \sim \sup\{\inf\{X_i, Z\} \mid 1 \leq i \leq n\}$, and therefore, simplicity holds. (Completeness) It is sufficient to show that, for all compromises Z among $X_1, \ldots, X_{n-1}$, and X_n, (1) $Z \not\succeq X_i$ for all X_i, and (2) there exists at least one $Y_i \preceq X_i$ for all X_i such that $Y_i \not\sim \perp$, $Y_i \succeq \inf\{X_j \mid 1 \leq j \leq n\}$, and $Z \sim \sup\{Y_i \mid 1 \leq i \leq n\}$. The condition of incompleteness directly satisfies condition (1). Now we let $Y_i \sim \inf\{X_i, Z\}$. Then, $Y_i \not\sim \perp, Y_i \succeq \inf\{X_j \mid 1 \leq j \leq n\}, X_i \succeq Y_i$, and $Z \sim \sup\{Y_i \mid 1 \leq i \leq n\}$ are satisfied, and therefore, condition (2) is satisfied. □

5.2 Concrete Dialectical Reasoning

We give a concrete and sound algorithm for dialectical reasoning characterized by definite clausal language and generalized subsumption. Generalized subsumption is a quasi-order on definite clauses with background knowledge expressed in a definite program, i.e., a finite set of definite clauses. It is approximation of relative entailment, i.e., logical entailment with background knowledge, and it can be reduced to ordinary subsumption with empty background knowledge (Nienhuys-Cheng and de Wolf, 1997). The readers are referred to (Buntine, 1988) for the definition. Let $\mathcal{L}_1$ be a definite clausal language, such that $\mathcal{L}_1$ has finite constants, finite predicate symbols and no function symbols, $\mathcal{D} \subseteq \mathcal{L}_1$ be a set of definite clauses that have the same literal in their head, and $\geq_{\mathcal{B}}$ be generalized subsumption with respect to $\mathcal{B}$ where $\mathcal{B} \subseteq \mathcal{L}_1$ is a definite program. Algorithm 2 is the algorithm for dialectical reasoning on complete lattice $< \mathcal{D}, \geq_{\mathcal{B}} >$.

Algorithm 2: Dialectical Reasoning on $< \mathcal{D}, \geq_{\mathcal{B}} >$.

Require: $\mathcal{B} \cup \{X_i\}$ are satisfiable for all X_i and $\inf\{X_i \mid 1 \leq i \leq n\} \not\approx \perp$.

1: $\mathcal{Z} := \emptyset$
2: **for** $i := 1$ to n **do**
3: $\quad \mathcal{W}_i := \emptyset$
4: $\quad$ compute $\mathcal{Y}_i \subseteq \{Y \mid Y$ is a tautology, or there exists SLD-derivation of Y' with X_i as a top clause and members of $\mathcal{B}$ as input clauses and $Y \in \rho_L^m(Y')\}$.
5: $\quad$ **for all** $Y \in \mathcal{Y}_i$ **do**
6: $\quad\quad$ **if** Y is a definite clause, $Y^+\alpha \notin L$ and $Y \sqsupseteq \{(X_1 \vee \cdots \vee X_n)^+\theta\} \cup \overline{M}$ **then**
7: $\quad\quad\quad \mathcal{W}_i := \mathcal{W}_i \cup \{Y\}$
8: $\quad\quad$ **end if**
9: $\quad$ **end for**
10: **end for**
11: **for all** $(W_1, \ldots, W_n) \in \mathcal{W}_1 \times \ldots \times \mathcal{W}_n$ **do**
12: $\quad$ compute LGS Z of $\{\{W_i^+\sigma_i\} \cup \overline{N_i} \mid 1 \leq i \leq n\}$
13: $\quad \mathcal{Z} := \mathcal{Z} \cup \{Z\}$
14: **end for**
15: **for** $i := 1$ to n **do**
16: $\quad$ **if** $Z \sqsupseteq \{X_i^+\phi_i\} \cup \overline{O}_i$ **then**
17: $\quad\quad \mathcal{Z} := \mathcal{Z} \setminus \{Z\}$
18: $\quad$ **end if**
19: **end for**
20: **return** $\mathcal{Z}$

The inputs of Algorithm 2 are $\mathcal{B}, \mathcal{D}$, $X_1, \ldots, X_n \in \mathcal{D}$, and iteration number m for refinement operator ρ_L for subsumption. ρ_L denotes a downward refinement operator for $< \mathcal{C}, \sqsupseteq >$ where $\mathcal{C}$ is a set of clauses. The readers are referred to (Nienhuys-Cheng and de Wolf, 1997) for the definition of ρ_L. In summary, ρ_L is a function whose input is a clause and output is a set of clauses which the input clause subsumes. The derivation is achieved by substituting functions, constants or variables, or adding new literals for the input clause. ρ_L^m in Algorithm 2 means that ρ_L is iterated m-time where an element of the output of ρ_L^i is returned to the input of ρ_L^{i+1}. Clause X subsumes clause Y, denoted by $X \sqsupseteq Y$ in Algorithm 2, if there exists a substitution θ such that $X\theta \subseteq Y$. For definite clause X, X^+ denotes a head of X and X^- denotes a set of the literals in the body of X. $\overline{S}$ denotes a set of negations of formulae in set S. α, θ, σ_i, and ϕ_i denote Skolem substitution for Y with respect to $\mathcal{B}$, Skolem substitution for $X_1 \vee \cdots \vee X_n$ with respect to $\mathcal{B} \cup \{Y\}$, Skolem substitution for W_i with respect to $\mathcal{B} \cup \{W_i \mid 1 \leq j \leq n\}$, and Skolem substitution for X_i with respect to $\mathcal{B} \cup \{X_j \mid 1 \leq j \leq n\}$, respectively. L, M, N_i, and O_i are the least Herbrand models of $\mathcal{B} \cup Y^-\alpha, \mathcal{B} \cup (X_1 \vee \cdots \vee X_n)^-\theta, \mathcal{B} \cup W_i^-\sigma_i$, and $\mathcal{B} \cup X_i^-\phi_i$, respectively. For simplicity, we do not explicitly describe the procedures for calculating the Skolem substitutions and the least Herbrand models. Under the restriction for $\mathcal{L}_1$ we impose, there exists a finite least Herbrand model on $\mathcal{L}_1$ and it is computable by fixed operator T_P (Emden and Kowalski, 1976). We assume the results of the operator.

The computation at line 4 in Algorithm 2 is equivalent to the computation of Y from X_i and $\mathcal{B}$, such that $X_i \geq_{\mathcal{B}} Y$. This is based on the proposition that, for all $X, Y \in \mathcal{D}$, $X \geq_{\mathcal{B}} Y$ iff there exists an SLD-deduction of Y, with X as top clause and members of $\mathcal{B}$ as input clauses (Nienhuys-Cheng and de Wolf, 1997). In the algorithm, SLD-deduction is split into SLD-derivation and subsumption, and refinement operator ρ_L for $< \mathcal{L}_1, \sqsupseteq >$ calculates subsumption. ρ_L is computable because of its locally finiteness (Nienhuys-Cheng and de Wolf, 1997). At line 6, the algorithm evaluates whether Y is a definite clause, $Y \not\approx_{\mathcal{B}} \perp$ and $Y \geq_{\mathcal{B}} \inf\{X_i \mid 1 \leq i \leq n\}$. For decidability, generalized subsumption is translated to decidable ordinary subsumption based on the proposition that if the least Herbrand model of $\mathcal{B} \cup Y^-\sigma$ is finite, then $X \geq_{\mathcal{B}} Y$ iff $X \sqsupseteq \{Y^+\sigma\} \cup \overline{M}$, for all $X, Y \in \mathcal{D}$ (Nienhuys-Cheng and de Wolf, 1997). At line 12, the algorithm computes the least generalization, under generalized subsumption, of $\{W_i \mid 1 \leq i \leq n\}$ by alternatively computing the least generalization, under subsumption, denoted by LGS, of $\{\{W_i^+\sigma_i\} \cup \overline{M_i} \mid 1 \leq i \leq n\}$. At line 16, the algorithm evaluates $Z \not\geq_{\mathcal{B}} X_i$ for all X_i by translating generalized subsumption to ordinary subsumption. Therefore, the following proposition holds.

Proposition 1. *Dialectical reasoning on $< \mathcal{D}, \geq_{\mathcal{B}} >$ is sound with respect to compromise on $< \mathcal{D}, \geq_{\mathcal{B}} >$.*

Example 2 (dialectical reasoning on $< \mathcal{D}, \geq_{\mathcal{B}} >$)**.** *Consider the following two clauses and the background knowledge.*

- $X_1 = compact(x) \wedge light(x) \wedge camera(x) \rightarrow buy(x)$
- $X_2 = resolution(x, high) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x)$
- $\mathcal{B} = \{compact(x) \wedge light(x) \rightarrow user\text{-}Friendly(x)\}$

Following each $Y_i (i = 1, 2)$ is SLD-deducible with X_i as top clause because X_i subsumes Y_i.

- $Y_1 = battery(x, long) \wedge compact(x) \wedge light(x) \wedge camera(x) \rightarrow buy(x)$
- $Y_2 = user\text{-}Friendly(x) \wedge resolution(x, high) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x)$

ρ_L derives Y_1 from X_1 by adding literals $\neg battery(y, z)$ and substituting $\{y/x, z/long\}$ into X_1. ρ_L derives Y_2 from X_2 by adding literal $\neg userFriendly(y)$ and substituting $\{y/x\}$ into X_2. For simplicity, these literals are arbitrary chosen to satisfy the condition at line 6 in Algorithm 2. Following L_1, L_2, and M are the least Herbrand models of $\mathcal{B} \cup Y_1^- \{x/a\}$, $\mathcal{B} \cup Y_2^- \{x/b\}$, and $\mathcal{B} \cup (X_1 \vee \cdots \vee X_n)^- \{x/c\}$, respectively.

- $L_1 = \{battery(a, long), compact(a), light(a), camera(a), user\text{-}Friendly(a)\}$
- $L_2 = \{user\text{-}Friendly(b), resolution(b, high), battery(b, long), camera(b)\}$
- $M = \{compact(c), light(c), user\text{-}Friendly(c), camera(c), resolution(c, high), battery(c, long)\}$

We let $\sigma_1 = \{x/d\}$ and $\sigma_2 = \{x/e\}$. Then, the least Herbrand models N_i of $\mathcal{B} \cup Y^- \sigma_i$ are as follows.

- $N_1 = \{battery(d, long), compact(d), light(d), camera(d), user\text{-}Friendly(d)\}$
- $N_2 = \{user\text{-}Friendly(e), resolution(e, high), battery(e, long), camera(e)\}$

Then, following Z is the least upper bound, under subsumption, of $\{\{Y_i^+ \sigma_i\} \cup \overline{N_i} \mid 1 \leq i \leq n\}$, and therefore, the least upper bound, under generalized subsumption, of $\{Y_1, Y_2\}$.

- $Z = user\text{-}Friendly(x) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x)$

By similar evaluation, Z turns out to be a consequence of the dialectical reasoning.

In this paper, we focus on defining a concrete and sound algorithm for dialectical reasoning, and we do not address the problem with the search space reduction using various biases. We assume the results of Example 2 in the next section.

6 COMPROMISE-BASED JUSTIFICATION

6.1 Handling Compromise Arguments

In contrast to reasoning about what to believe, i.e., theoretical reasoning, reasoning about what to do, i.e., practical reasoning, is closely-linked to agents' goals or desires because it depends not only on their beliefs, but also on their goals. We assume that agents have their common goal G described by a first-order formula with zero or more free variables. Obviously, compromise should be effective only in practical reasoning. In order to handle compromise arguments in the argumentation system, we distinguish practical arguments from theoretical arguments based on whether the arguments satisfy a agents' common goal or not. An argument is called practical if it satisfies agents' common goal, and otherwise it is called theoretical.

Definition 6 (Practical Argument)**.** *Let G be a goal and A be an argument. A is a practical argument for G if there exists a ground substitution α such that $CONC(A) \cup \mathcal{F}_n \models G\alpha$.*

A ground substitution is a mapping from a finite set of variables to terms without variables. We expand the notion of compromise into practical arguments. A compromise argument is a practical argument that uses dialectical reasoning.

Definition 7 (Compromise Argument)**.** *Let G be a goal and $A, B, \cdots, M$ and N be distinct practical arguments for G. A is a compromise argument among $B, \cdots, M$ and N if $a \in CONC(A)$ is a compromise among $b \in CONC(B), ..., m \in CONC(M)$ and $n \in CONC(N)$.*

The rebutting and the undercutting shown in Definition 3 can be viewed as theoretical in the sense that the grounds of conflicts are logical contradiction. On the other hand, practical arguments conflict each other due to the existence of alternatives. Such practical conflict occurs when they satisfy a same goal in different ways. We define the notion of defeat based on the three perspectives: the rebutting, the undercutting and the existence of alternatives.

Definition 8 (Defeat)**.** *Let G be a goal and A and B be distinct arguments. A defeats B iff*

- *A rebuts B and B does not rebut A; or*
- *A undercuts B; or*
- *Both A and B are practical arguments for G such that they have distinct substitutions α and β for G, respectively, and B is not a compromise among arguments, one of which is A.*

We say that A strictly defeats B iff A defeats B and B does not defeat A.

6.2 An Illustrative Example

We detail the motivational example about camera decision in Section 2. Both agents A and B are assumed to have their common goal $G \equiv buy(x) \wedge camera(x)$ and the following individual knowledge bases, denoted by $\Gamma_A = \mathcal{F}_n^A \cup \mathcal{F}_c^A \cup \Delta^A$ and $\Gamma_B = \mathcal{F}_n^B \cup \mathcal{F}_c^B \cup \Delta^B$, respectively.

- $\mathcal{F}_n^A = \{camera(a), camera(c), overBudget(b), \forall x.compact(x) \wedge light(x) \wedge camera(x) \rightarrow buy(x) (= X_1), \forall x.overBudget(x) \rightarrow \neg buy(x), takeShoot(b, 200)\}$
- $\mathcal{F}_c^A = \{compact(a), userFriendly(c)(= f_1), battery(c, long), \forall x.compact(x) \wedge light(x) \rightarrow userFriendly(x)(= r_1), \forall x \forall y.takeShoot(x,y) \wedge > (300, y) \rightarrow \neg battery(x, long)(= r_2)\}$
- $\Delta^A = \{\sim \neg light(x) \Rightarrow light(x)(= d_1)\}$
- $\mathcal{F}_n^B = \{camera(b), \neg inStock(a), \forall x.resolution(x, high) \wedge battery(x, long) \wedge camera(x) \rightarrow buy(x)(= X_2), \forall x.\neg inStock(x) \rightarrow \neg buy(x), price(b, \$200), \forall x \forall y.price(x,y) \wedge \geq (\$300, y) \rightarrow withinBudget(x)(= r_3)\}$
- $\mathcal{F}_c^B = \{resolution(b, high)(= f_2)\}$
- $\Delta^B = \{\sim \neg battery(x, long) \Rightarrow battery(x, long) (= d_2)\}$

Further, only deductive reasoning, *DMP*, and dialectical reasoning on $< \mathcal{D}, \geq_{\mathcal{B}} >$ can be used for constructing arguments. Agents construct arguments only from their own individual knowledge bases with opponent's arguments previously stated. They advance argumentation by constructing dialogue trees whose roots are practical arguments, which means that they try to justify their own practical arguments for G straightforwardly. For readability, we express arguments using proof trees to visualize the reasoning. Deductive reasoning, *DMP*, and dialectical reasoning are expressed by '—,' '$\cdots$,' and '= =,' respectively. Following A_1,A_2 and A_3, that A_2 strictly defeats A_1 and A_3 defeats A_1, form a dialogue tree.

$$A_1 : \frac{\dfrac{d_1\{x/a\}}{light(a)} \quad compact(a) \quad cam(a) \quad X_1}{buy(a)}$$

$$A_2 : \frac{\neg inStock(a) \quad \neg inStock(x) \rightarrow \neg buy(x)}{\neg buy(a)}$$

$$A_3 : \frac{\dfrac{d_2\{x/b\}}{battery(b, long)} \quad f_2 \quad cam(b) \quad X_2}{buy(b)}$$

$buy(a)$ concluded by A_1 is rated as unacceptable because A cannot win the dialogue tree. Then, in turn, B tries to make a practical argument for G. Following A_3,A_4,A_5 and A_6 with A_1, that A_4 defeats A_3, A_5 strictly defeats A_4, A_6 strictly defeats A_3, and A_1 defeats A_3, form a dialogue tree.

$$A_4 : \frac{\sim withinBudget(b) \Rightarrow \neg buy(b)}{\neg buy(b)}$$

$$A_5 : \frac{price(b, \$200) \quad \geq (\$300, \$200) \quad r_3}{withinBudget(b)}$$

$$A_6 : \frac{takeShoot(b, 200) \quad > (300, 200) \quad r_2}{\neg battery(b, long)}$$

Neither B can win the dialogue tree. Thus, $buy(b)$ is rated as unacceptable. Neither of them can make another practical argument for G using only deductive reasoning and *DMP*. However, A can construct following compromise argument A_7 between A_1 and A_3 using dialectical reasoning in combination with these reasoning. X_1,X_2, and Z in A_7 are same as Example 2.

$$A_7 : \frac{\dfrac{X_1 \quad X_2 \quad r_1}{Z} \quad f_1 \quad battery(c, long) \quad cam(c)}{buy(c)}$$

Neither A and B can make any defeating arguments against A_7. Thus, A_7 forms the dialogue tree by itself and $buy(c)$ is rated as acceptable . Note that the argument concluding $buy(c)$ cannot be constructed from $\Gamma_1 \cup \Gamma_2$ without dialectical reasoning.

7 RELATED WORKS

The prime difference between dialectical reasoning and inductive or abductive reasoning is that whereas the consequence of dialectical reasoning satisfies incompleteness, given in definition 5, inductive and abductive hypotheses satisfy completeness. Moreover, dialectical reasoning differs from deductive reasoning in the sense that the consequences of dialectical reasoning are not necessarily deducible. For instance, in Example 1, Z is not a semantical consequence of $\mathcal{B} \cup \{X_1, X_2\}$, although Z is a compromise between X_1 and X_2. However, if the complete lattice is characterized by satisfiability relation, then any compromise is a semantical consequence of the premises. Nonetheless, dialectical reasoning has significance because neither of other reasoning mechanisms address what compromise is and how agents infer compromise.

In (Amgoud et al., 2008), the authors introduce the notion of concession as an essential element of argument-based negotiation. They define concession as a given offer supported by an argument that has suboptimal state in argumentation. Thus, in contrast to our approach, concession is not realized by reasoning mechanism. Similarly, game theory does not address the rational generation of a new option although it gives the way for rational choices. In (Sawamura et al., 2003), the authors introduce seven dialectical inference rules into dialectical logic DL and weaker dialectical logic DM (Routley and Meyer, 1976) in order to make concession and compromise from an inconsistent theory. The authors, however, do not show an underlying principle of these rules. Further, contrary to the philosophical opinion (Sabre, 1991), the set of the premises of each inference rules is restricted to logical contradiction. In contrast, we give the underlying principle of dialectical reasoning by defining abstract reasoning on a complete lattice. Further, as shown in Example 2, we do not restrict the premises to contradiction.

8 CONCLUSIONS AND FUTURE WORKS

We defined compromise on an abstract complete lattice, and proposed a sound and complete algorithm for dialectical reasoning with respect to compromise. Then, we proposed the concrete algorithm for the dialectical reasoning characterized by definite clausal language and generalized subsumption. The concrete algorithm was proved to be sound with respect to the compromise. We expanded the argumentation system proposed by Prakken (Prakken, 1997) to handle compromise arguments, and illustrated that a compromise argument realizes a compromise-based justification towards argument-based deliberation.

We plan to elaborate more applicable algorithms by incorporating language and search biases into our algorithms. Furthermore, recently, some kinds of practical reasoning are proposed for argument-based reasoning (Bench-Capon and Prakken, 2006). However, there is little work that focuses on the relation between phases of argumentation and reasoning. Especially, compromise should be taken at the final phase of deliberation or negotiation. We will enable agents to use appropriate reasoning depending on the phase of argumentation.

REFERENCES

Amgoud, L., Dimopoulos, Y., and Moraitis, P. (2008). A general framework for argumentation-based negotiation. In *Proc. of The 4th International Workshop on Argumentation in Multi-Agent Systems (ArgMAS 2007)*, pages 1–17.

Bench-Capon, T. J. M. and Prakken, H. (2006). Justifying actions by accruing arguments. In *Proc. of The First International Conference on Computational Models of Argument (COMMA 2006)*, pages 247–258.

Buntine, W. (1988). Generalized subsumption and its applications to induction and redundancy. *Artificial Intelligence*, 36:146–176.

Carnielli, W., Coniglio, M. E., and Marcos, J. (2007). *Logics of Formal Inconsistency*, volume 14, pages 1–93. Springer, handbook of philosophical logic, 2nd edition.

Dung, P. M. (1995). On the acceptability of arguments and its funedamental role in nonmonotonic reasoning, logic programming, and *n*-person games. *Artificial Intelligence*, 77:321–357.

Emden, M. H. V. and Kowalski, R. A. (1976). The semantics of predicate logic as a programming language. *Journal of the Association for Computing Machinery*, 23:733–742.

Gordon, T. F. (1995). *The Pleadings Game – An Artificial Intelligence Model of Procedural Justice*. Kluwer Academic Publishers.

Hamblin, C. L. (1970). *Fallacies*. Methuen.

Nienhuys-Cheng, S.-H. and de Wolf, R. (1997). *Foundation of Inductive Logic Programming*. Springer.

Nisbett, R. E. (2003). *The Geography of Thought: How Asians and Westerners Think Differently ... and Why*. FREE PRESS.

Prakken, H. (1997). *Logical Tools for Modelling Legal Argument: A Study of Defeasible Reasoning in Law*. Kluwer Academic Publishers.

Prakken, H. (1999). Dialectical proof theory for defeasible argumentation with defeasible priorities (preliminary report). In *Proc. of The 4th ModelAge Workshop 'Formal Models of Agents'*, pages 202–215.

Prakken, H. and Sartor, G. (1997). Argument-based extended logic programming with defeasible priorities. *Journal of Applied Non-classical Logics*, 7:25–75.

Reiter, R. (1980). A logic for default reasoning. *Artificial Intelligence*, 13:81–132.

Rescher, N. (2007). *Dialectics – A Classical Approach to Inquiry*. ontos verlag.

Routley, R. and Meyer, R. K. (1976). Dialectical logic, classical logic, and the consistency of the world. *Studies in East European Thought*, 16(1-2):1–25.

Sabre, R. M. (1991). An alternative logical framework for dialectical reasoning in the social and policy sciences. *Theory and Decision*, 30(3):187–211.

Sawamura, H., Yamashita, M., and Umeda, Y. (2003). Applying dialectic agents to argumentation in e-commerce. *Electronic Commerce Research*, 3(3-4):297–313.

Thomas, K. W. (1992). Conflict and conflict management: Reflections and update. *Journal of Organizational Behavior*, 13:265–274.