Assumption-Based Argumentation Equipped with Preferences and Constraints

Toshiko Wakaki(✉)

Shibaura Institute of Technology, 307 Fukasaku, Minuma-ku,
Saitama-city, Saitama 337–8570, Japan
twakaki@sic.shibaura-it.ac.jp

Abstract. Čyras and Toni claimed that assumption-based argumentation equipped with preferences (p_ABA) cannot solve two examples presented by them since the given preferences don't work in their p_ABAs whose underlying ABAs have a unique extension, and hence they proposed ABA$^+$. However in p_ABAs encoded by them, we found that they mistook hypotheses contained in their example for assumptions, while Čyras ignored some constrains contained in another example. Hence against their claim, first this paper shows that p_ABAs in which we expressed the respective knowledge correctly give us solutions of them without any difficulties. Second we present the technique to represent hypotheses in ABA as well as a method to incorporate some kind of constraints in p_ABA. Finally we show a famous non-monotonic reasoning example with preferences that ABA$^+$ leads to incorrect results.

1 Introduction

Assumption-Based Argumentation (ABA) [1,9] is a general-purpose argumentation framework whose arguments are structured. It does not have a mechanism to deal with the given explicit preferences though explicit preferences are often required to resolve conflicts between arguments in human argumentation.

Recently to overcome difficulties of the existing approaches that map the explicit preferences into ABA, we proposed an assumption-based argumentation framework equipped with preferences (p_ABA) [22,23], which incorporates *explicit preferences over sentences* into ABA. As discussed in [23], our approach introducing preferences over sentences in the framework is inspired by *prioritized circumscription* [15,16], namely, the most well-established formalization for commonsense reasoning with preferences that enables us to represent various preferences by means of priorities over its minimized predicates. In regard to the semantics of p_ABA, we provide a method to 'lift' a sentence ordering given in p_ABA to the argument ordering. Accordingly we can freely give any semantics to the proposed p_ABA based on either an argument ordering or a sentence ordering. W.r.t. other frameworks (e.g. $ASPIC^+$ [17,18]), the lift from preferences to argument orderings is also performed. However based on their argument orderings, the altered argumentation framework with a modified successful *attacks* (i.e. *defeat*) is constructed, to which Dung's argument-based

S. Moral et al. (Eds.): SUM 2017, LNAI 10564, pp. 178–193, 2017.
DOI: 10.1007/978-3-319-67582-4_13

semantics is applied. This denotes that the extension of the altered argumentation framework is not always an extension of the initial argumentation framework without preferences but a modified one, which is in conflict with our philosophy based on the idea to treat preferences in prioritized circumscription [15,16]. Now recall that, when every model is a Herbrand model, a model of prioritized circumscription expressed by $Circum(T; P^1 > \cdots > P^k; Z)$ is a minimal one among Herbrand models of the first order theory T w.r.t. the model (i.e. structure) ordering $\leq^{P^1 > \cdots > P^k; Z}$ lifted from the given predicate ordering $P^1 > \cdots > P^k$. Thus in a similar way, we presented a method to lift a sentence ordering (resp. an argument ordering) to the extension ordering ($\sqsubseteq_{ex}$). Thanks to the extension ordering, the semantics of p_ABA is given by $\mathcal{P}$-argument extensions along with $\mathcal{P}$-assumption extensions which are maximal ones w.r.t. the extension ordering $\sqsubseteq_{ex}$ among extensions of its underlying ABA. Thus in a special case of $Circum(T; P^1 > \cdots > P^k; Z)$ where T has a unique model, $Circum(T; P^1 > \cdots > P^k; Z)$ has a unique model which coincides with the model of T since the unique model of T is always minimal w.r.t. $\leq^{P^1 > \cdots > P^k; Z}$ regardless of the given $P^1 > \cdots > P^k$. This means that when T has a unique model, any of other interpretations of T which are inevitably inconsistent is never selected as a model of $Circum(T; P^1 > \cdots > P^k; Z)$ by taking account of priorities. Hence inheriting the same property, in a special case of p_ABA such that its underlying ABA (which satisfies rationality postulates [10]) has a unique extension, p_ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$ has a unique $\mathcal{P}$ extension which coincides with the extension of its underlying ABA since the unique argument extension of such ABA is maximal w.r.t. $\sqsubseteq_{ex}$ regardless of the given $\preceq$.

As for this property, Čyras and Toni claimed that a p_ABA framework cannot solve two examples (i.e. [5, Example 1], [4, Example 1]) since the given preferences don't work in p_ABAs encoded by them whose underlying ABAs have a unique extension, and hence they proposed ABA$^+$ [5]. However in their p_ABAs, we found that they mistook hypotheses contained in [5, Example 1] for assumptions, while Čyras ignored some constrains contained in another one [4, Example 1]. Hence against their claim, first we show that p_ABAs in which we expressed the respective knowledge correctly give us solutions of them without any difficulties, and the given preferences work well in the p_ABAs where the underlying ABAs have the multiple extensions. Second we present the technique to represent hypotheses in ABA as well as a method to incorporate some kind of constraints in ABA and p_ABA. Finally, and perhaps most importantly, we show a famous non-monotonic reasoning example involving the use of preferences that ABA$^+$ leads to incorrect results, while p_ABA avoids this problem.

This paper is organized as follows. Section 2 gives preliminaries. Section 3 presents how to represent hypotheses in ABA. Section 4 presents an ABA equipped with preferences and constrains. Section 5 discusses related work. Section 6 concludes this paper while showing a counterexample to ABA$^+$.

2 Preliminaries

Definition 1 (ABA). *An assumption-based argumentation framework (an ABA framework, or an ABA, for short) [1,9,13] is a tuple* $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$*, where*

- $(\mathcal{L}, \mathcal{R})$ *is a deductive system, with* $\mathcal{L}$ *a language consisting of countably many sentences and* $\mathcal{R}$ *a set of inference rules of the form* $b_0 \leftarrow b_1, \ldots, b_m (m \geq 0)$*, where* b_0 *(resp.* $b_1, \ldots, b_m$*) is called the head (resp. the body) of the rule.*
- $\mathcal{A} \subseteq \mathcal{L}$*, is a (non-empty) set, referred to as assumptions.*
- $\mathcal{C}$ *is a total mapping from* $\mathcal{A}$ *into* $2^{\mathcal{L}} \setminus \{\emptyset\}$*, where each* $c \in \mathcal{C}(\alpha)$ *is a contrary of* $\alpha \in \mathcal{A}$*.*

We enforce that ABA frameworks are flat, namely assumptions do not occur as the heads of rules. For a special case such that each assumption has the unique contrary sentence (i.e. $|\mathcal{C}(\alpha)|=1$ *for* $\forall \alpha \in \mathcal{A}$*), an ABA framework is usually defined as* $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \bar{\ } \rangle$*, where a total mapping* $\bar{\ }$ *from* $\mathcal{A}$ *into* $\mathcal{L}$ *is used.*

In ABA, an *argument for (the claim)* $c \in \mathcal{L}$ *supported by* $K \subseteq \mathcal{A}$ ($K \vdash c$ in short) is a (finite) tree with nodes labelled by sentences in $\mathcal{L}$ or by τ, and *attacks* against arguments are directed at the assumptions in their supports as follows.

- An argument $K \vdash c$ *attacks* an assumption α iff $c \in \mathcal{C}(\alpha)$.
- An argument $K_1 \vdash c_1$ *attacks* an argument $K_2 \vdash c_2$ iff $c_1 \in \mathcal{C}(\alpha)$ for $\exists \alpha \in K_2$.

Corresponding to ABA $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$, the abstract argumentation framework $AF_{\mathcal{F}} = (AR, attacks)$ is constructed based on *arguments* and *attacks* addressed above, and all argumentation semantics [7] can be applied to $AF_{\mathcal{F}}$. For a set $\mathcal{A}rgs$ of arguments, let $\mathcal{A}rgs^{+} = \{A |$ there exists an argument in $\mathcal{A}rgs$ that *attacks* $A\}$. $\mathcal{A}rgs$ is *conflict-free* iff $\mathcal{A}rgs \cap \mathcal{A}rgs^{+} = \emptyset$. $\mathcal{A}rgs$ *defends* an argument A iff each argument that attacks A is attacked by an argument in $\mathcal{A}rgs$.

Definition 2 [2,7,9,13]**.** *Let* $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ *be an ABA framework, and* AR *the associated set of arguments. Then* $\mathcal{A}rgs \subseteq AR$ *is: admissible iff* $\mathcal{A}rgs$ *is conflict-free and defends all its elements; a complete argument extension iff* $\mathcal{A}rgs$ *is admissible and contains all arguments it defends; a preferred (resp. grounded) argument extension iff it is a (subset-)maximal (resp. (subset-)minimal) complete argument extension; a stable argument extension iff it is conflict-free and* $\mathcal{A}rgs \cup \mathcal{A}rgs^{+} = AR$*; an ideal argument extension iff it is a (subset-)maximal complete argument extension that is contained in each preferred argument extension.*

The various ABA semantics [1] is also described in terms of sets of assumptions.

- A set of assumptions $\mathcal{A}sms$ *attacks* an assumption α iff $\mathcal{A}sms$ enables the construction of an argument for the claim $\exists c \in \mathcal{C}(\alpha)$.
- A set of assumptions $\mathcal{A}sms_1$ *attacks* a set of assumptions $\mathcal{A}sms_2$ iff $\mathcal{A}sms_1$ *attacks* some assumption $\alpha \in \mathcal{A}sms_2$.

For a set of assumptions $\mathcal{A}sms$, let $\mathcal{A}sms^{+} = \{\alpha \in \mathcal{A} | \mathcal{A}sms$ attacks $\alpha\}$. $\mathcal{A}sms$ is *conflict-free* iff $\mathcal{A}sms \cap \mathcal{A}sms^{+} = \emptyset$. $\mathcal{A}sms$ *defends* an assumption α iff each set of assumptions that attacks α is attacked by $\mathcal{A}sms$. Assumption extensions are defined like argument extensions as follows.

Definition 3 [2,9,13]**.** *Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ be an ABA framework. Then $\mathcal{A}sms$ is: admissible iff $\mathcal{A}sms$ is conflict-free and defends all its elements; a complete assumption extension iff $\mathcal{A}sms$ is admissible and contains all assumptions it defends; a preferred (resp. grounded) assumption extension iff it is a (subset-) maximal (resp. (subset-)minimal) complete assumption extension; a stable assumption extension iff it is is conflict-free and $\mathcal{A}sms \cup \mathcal{A}sms^+ = \mathcal{A}$; an ideal assumption extension iff it is a (subset-)maximal complete assumption extension that is contained in each preferred assumption extension.*

Let $Sname \in \{complete, preferred, grounded, stable, ideal\}$. It is shown that there is a one-to-one correspondence between assumption extensions and argument extensions of a given ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ under the $Sname$ semantics as follows.

Theorem 1 [2,22,23]**.** *Let $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ be an ABA framework, AR be the set of all arguments that can be constructed using this ABA framework, and $\texttt{Asms2Args}: 2^{\mathcal{A}} \rightarrow 2^{AR}$ and $\texttt{Args2Asms}: 2^{AR} \rightarrow 2^{\mathcal{A}}$ be functions such that,*

$$\texttt{Asms2Args}(\mathcal{A}sms) = \{K \vdash c \in AR \mid K \subseteq \mathcal{A}sms\},$$
$$\texttt{Args2Asms}(\mathcal{A}rgs) = \{\alpha \in \mathcal{A} \mid \alpha \in K \text{ for an argument } K \vdash c \in \mathcal{A}rgs\}.$$

Then if $\mathcal{A}sms \subseteq \mathcal{A}$ is a Sname assumption extension, then $\texttt{Asms2Args}(\mathcal{A}sms)$ is a Sname argument extension, and if $\mathcal{A}rgs \subseteq AR$ is a Sname argument extension, then $\texttt{Args2Asms}(\mathcal{A}rgs)$ is a Sname assumption extension.

Proof. In [23], proofs for $Sname \in \{complete, preferred, grounded, stable\}$ are given. For $Sname = ideal$, it is also easily proved like [2,8]. □

For notational convenience, let $claim(Ag)$ stand for the claim c of an argument Ag such that $K \vdash c$, and $\texttt{Concs}(E) = \{c \in \mathcal{L} \mid K \vdash c \in E\}$ for an extension E.

Definition 4 (ABA equipped with preferences [22,23]*). An assumption-based argumentation framework equipped with preferences (a p_ABA framework, or p_ABA for short) is a tuple $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$, where*
- *$\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ is an ABA framework,*
- *$\preceq \subseteq \mathcal{L} \times \mathcal{L}$ is a sentence ordering called a priority relation, which is a preorder, that is, reflexive and transitive. As usual, $c' \prec c$ iff $c' \preceq c$ and $c \not\preceq c'$. For any sentences $c, c' \in \mathcal{L}$, $c' \preceq c$ (resp. $c' \prec c$) means that c is at least as preferred as c' (resp. c is strictly preferred to c').*

For a special case such that each assumption has the unique contrary sentence (i.e. $|\mathcal{C}(\alpha)|=1$ for $\forall\alpha \in \mathcal{A}$), a p_ABA framework may be represented as $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \bar{\ }, \preceq \rangle$ instead of $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$, where a total mapping $\bar{\ }$ from $\mathcal{A}$ into $\mathcal{L}$ is used instead of a total mapping $\mathcal{C}$ from $\mathcal{A}$ into $2^{\mathcal{L}} \setminus \{\emptyset\}$ like ABA.

For p_ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$, let $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ be the associated ABA and $AF_{\mathcal{F}} = (AR, attacks)$. Then $\leq \subseteq AR \times AR$ is the argument ordering over AR constructed from $\preceq$ as follows.

For any arguments $Ag_1, Ag_2 \in AR$ such that $K_1 \vdash c_1$ and $K_2 \vdash c_2$,

$$Ag_1 \leq Ag_2 \quad \text{iff} \quad c_1 \preceq c_2 \qquad \text{for } c_i = claim(Ag_i)\ (1 \leq i \leq 2)$$

Definition 5 (Preference relations $\sqsubseteq_{ex}$). *Given p_ABA* $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$, *let* $\mathcal{E}$ *be the set of Sname argument extensions of the AA framework* $AF_{\mathcal{F}} = (AR, attacks)$ *corresponding to the ABA framework* $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ *under Sname semantics and* $f : 2^{AR} \times 2^{AR} \rightarrow 2^{AR}$ *be the function s.t.* $f(U, V) = \{X | \; claim(X) = claim(Y) \text{ for } X \in U, \; Y \in V\}$. *Then* $\sqsubseteq_{ex}$ *over* $\mathcal{E}$ *(i.e.* $\sqsubseteq_{ex} \subseteq \mathcal{E} \times \mathcal{E}$*) is defined as follows [22,23]. For any Sname argument extensions,* E_1, E_2 *and* E_3 *from* $\mathcal{E}$,

1. $E_1 \sqsubseteq_{ex} E_1$,
2. $E_1 \sqsubseteq_{ex} E_2$ *if for some argument* $Ag_2 \in E_2 \setminus \Delta_2$,
   (i) there is an argument $Ag_1 \in E_1 \setminus \Delta_1$ *s.t.* $claim(Ag_1) \preceq claim(Ag_2)$ *and,*
   (ii) there is no argument $Ag_3 \in E_1 \setminus \Delta_1$ *s.t.* $claim(Ag_2) \prec claim(Ag_3)$,
   where $\Delta_1 = f(E_1, E_2)$ *and* $\Delta_2 = f(E_2, E_1)$,
3. *if* $E_1 \sqsubseteq_{ex} E_2$ *and* $E_2 \sqsubseteq_{ex} E_3$, *then* $E_1 \sqsubseteq_{ex} E_3$;

$\sqsubseteq_{ex}$ *is a preorder. We write* $E_1 \sqsubset_{ex} E_2$ *if* $E_1 \sqsubseteq_{ex} E_2$ *and* $E_2 \not\sqsubseteq_{ex} E_1$ *as usual.*

The preference relation $\sqsubseteq_{ex}$ can be also defined by using the argument ordering $\leq$ in a way that $claim(Ag_1) \preceq claim(Ag_2)$ and $claim(Ag_2) \prec claim(Ag_3)$ is replaced with $Ag_1 \leq Ag_2$ and $Ag_2 < Ag_3$ in item no. 2 of Definition 5 [22,23].

Let $Sname \in \{complete, preferred, grounded, stable, ideal\}$. The semantics of p_ABA is given by $Sname$ $\mathcal{P}$ extensions which are the maximal ones w.r.t. $\sqsubseteq_{ex}$ among $Sname$ extensions as follows.

Definition 6 ($\mathcal{P}$-extensions [22,23]). *Given a p_ABA framework* $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$, *let* $\mathcal{E}$ *be the set of Sname argument extensions of* $AF_{\mathcal{F}} = (AR, attacks)$ *corresponding to the ABA framework* $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C} \rangle$ *under Sname semantics. Then a Sname argument extension* $E \in \mathcal{E}$ *is called a Sname* $\mathcal{P}$*-argument extension of the p_ABA framework if* $E \sqsubseteq_{ex} E'$ *implies* $E' \sqsubseteq_{ex} E$ *(with respect to* $\preceq$*) for any* $E' \in \mathcal{E}$. *In other words,* E *is a Sname* $\mathcal{P}$*-argument extension of a p_ABA iff there is no Sname argument extension* $E' \in \mathcal{E}$ *such that* $E \sqsubset_{ex} E'$. *For a Sname* $\mathcal{P}$*-argument extension* E, `Args2Asms`(E) *is called a Sname* $\mathcal{P}$*-assumption extension. Both a Sname* $\mathcal{P}$*-argument extension and a* $\mathcal{P}$*-assumption extension may be called a Sname* $\mathcal{P}$ *extension for short.*

3 Representing Hypotheses in ABA

In logic programming, NAF literals are used to perform non-monotonic and default reasoning, while hypotheses (i.e. abducibles, abducible facts) are used to perform abductive reasoning or hypothetical reasoning. In [14,20], it is shown that an abductive logic program (or an abductive program) can be transformed into a logic program without abducibles, where for each abducible a, a new atom a' is introduced representing the complement of a and a new pair of rules:

$$a \leftarrow not \; a', \qquad a' \leftarrow not \; a$$

is added to the program. Such transformation for hypotheses was also used to compute abductive argumentation [21].

In ABA, hypotheses are different from assumptions as abducible literals are different from NAF literals in logic programming. Hence when hypotheses (or abducible facts) are contained in the knowledge, each one, say a, can be also expressed by a new pair of rules in ABA as follows:

$$a \leftarrow \delta, \qquad a' \leftarrow \delta'$$

where a' is a newly introduced sentence representing the complement of a, while δ, δ' are newly introduced assumptions such that $\overline{\delta} = a'$, $\overline{\delta'} = a$.

Example 1. Consider the example shown in [5, Example 1] as follows:
"Zed wants to go out and two of his friends, Alice and Bob, are available. Best, Zed would take them both, but as far as he knows, Bob does not like Alice, although she does not have anything against Bob. If Zed offers to both of them at the same time, Bob may be in the awkward position to refuse Alice's company. Offering separately, Alice is up for all three going, while Bob insists on cutting Alice out. Zed may opt for the latter option. However, had Zed a preference between the two, - say Alice were a better friend of his - then he would go out with her." □

In what follows, a (resp. b) denotes that Alice (resp. Bob) might go out with Zed, while $\neg a$ (resp. $\neg b$) denotes the negation of a (resp. b). Since Alice (resp. Bob) might go out with Zed or might not, a and b are not assumptions but hypotheses. Hence the situation about them is expressed by the following rules:

$$a \leftarrow \alpha, \qquad \neg a \leftarrow \alpha', \qquad b \leftarrow \beta, \qquad \neg b \leftarrow \beta',$$

where $\mathcal{A} = \{\alpha, \alpha', \beta, \beta'\}$, $\overline{\alpha} = \neg a$, $\overline{\alpha'} = a$, $\overline{\beta} = \neg b$ and $\overline{\beta'} = b$. The opted option such that Bob insists on cutting Alice out is expressed by $\neg a \leftarrow b$. Preferences such that best, Zed would take them both, but he prefers Alice to Bob are expressed by:

$$\{\neg a, \neg b\} \preceq \{b\} \preceq \{a\} \preceq \{a, b\} \tag{1}$$

According to [23, Definition 25], preferences between conjunctive knowledge shown above can be encoded in p_ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \preceq \rangle$ by introducing new rules:

$$c_1 \leftarrow a, b, \qquad\qquad c_2 \leftarrow \neg a, \neg b,$$

along with preferences:

$$c_2 \preceq b \preceq a \preceq c_1 \qquad (a, b, c_1, c_2 \in \mathcal{L} \setminus \mathcal{A}) \tag{2}$$

Then based on p_ABA consisting of $\mathcal{R}$= $\{\neg a \leftarrow b,\ a \leftarrow \alpha,\ \neg a \leftarrow \alpha',\ b \leftarrow \beta,\ \neg b \leftarrow \beta',\ c_1 \leftarrow a, b,\ c_2 \leftarrow \neg a, \neg b\}$, $\mathcal{A} = \{\alpha, \alpha', \beta, \beta'\}$, $\overline{\alpha} = \neg a$, $\overline{\alpha'} = a$, $\overline{\beta} = \neg b$, $\overline{\beta'} = b$ and $c_2 \preceq b \preceq a \preceq c_1$, arguments and *attacks* are constructed as follows:

- $A : \{\alpha\} \vdash a$
- $A' : \{\alpha'\} \vdash \neg a$
- $B : \{\beta\} \vdash b$
- $B' : \{\beta'\} \vdash \neg b$
- $X : \{\beta\} \vdash \neg a$
- $C_1 : \{\alpha, \beta\} \vdash c_1$
- $C_2 : \{\alpha', \beta'\} \vdash c_2$
- $\xi : \{\alpha\} \vdash \alpha$
- $\xi' : \{\alpha'\} \vdash \alpha'$
- $\eta : \{\beta\} \vdash \beta$
- $\eta' : \{\beta'\} \vdash \beta'$

$attacks = \{(A, A'), (A, \xi'), (A, C_2), (A', A), (A', \xi), (A', C_1), (X, A), (X, \xi), (X, C_1), (B, B'), (B, \eta'), (B, C_2), (B', B), (B', \eta), (B', X), (B', C_1)\}$.

The underlying ABA has the preferred (resp. stable) argument extensions E_i $(1 \leq i \leq 3)$ as follows:

$$\begin{aligned} E_1 &= \{A', B, X, \xi', \eta\}, &&\text{with } \texttt{Concs}(E_1) = \{\neg a, b, \alpha', \beta\}, \\ E_2 &= \{A, B', \xi, \eta'\}, &&\text{with } \texttt{Concs}(E_2) = \{a, \neg b, \alpha, \beta'\}, \\ E_3 &= \{A', B', C_2, \xi', \eta'\}, &&\text{with } \texttt{Concs}(E_3) = \{\neg a, \neg b, c_2, \alpha', \beta'\}. \end{aligned}$$

Due to $E_3 \sqsubseteq E_1 \sqsubseteq E_2$ derived from (2), E_2 is the unique preferred (resp. stable) $\mathcal{P}$-argument extension in the p_ABA. Hence against Čyras and Toni's claim [5], E_2 gives us the solution that Zed would go out with Alice and without Bob.

Remark: In [5], Čyras and Toni expressed this example by p_ABA consisting of $\mathcal{R} = \{\overline{\alpha} \leftarrow \beta\}$, $\mathcal{A} = \{\alpha, \beta\}$, $\beta \preceq \alpha$. Then they claimed that regarding arguments $\texttt{A}$ for Alice and $\texttt{B}$ for Bob, $\texttt{A}$ for Alice cannot be obtained from the extension of p_ABA encoded by them since its underlying ABA has the unique extension $\{\texttt{B}\}$, where $attacks = \{(\texttt{B}, \texttt{A})\}$ and $\texttt{B} < \texttt{A}$. Therefore the reason why they could not obtain the solution based on p_ABA is that they mistook the hypothetical knowledge a, b for assumptions α, β; and it is not due to the property of p_ABA.

4 Assumption-Based Argumentation Equipped with Preferences and Constraints

4.1 Problematic Knowledge Representation

As addressed in introduction, Čyras [4] claimed that p_ABA cannot solve the example called "Cakes" presented by him, whose scenario is shown as follows.

Example 2 (*Cakes* [4, Example 1]). There are three pieces of cakes on a table: a piece of Almond cake, a Brownie, and a piece of Cheesecake. You want to get as many cakes as possible, and the following are the rules of the game.

1. You can take cakes from the table in two 'rounds':
   (a) In the first round you can take at most two cakes;
   (b) In the second round you can take at most one cake.
2. If you take Almond cake and Cheesecake in the first round, Brownie will not be available in the second round. (Nothing is known about other possible combinations.)
3. Finally, very importantly, suppose that you prefer Brownie over Almond cake. (No other preferences.)

Which pair(s) of cakes would you choose in the first round? □

The solution of *Cakes* is that "either a pair of Brownie cake and Cheesecake or a pair of Almond and Brownie cakes is chosen in the first round".

In [4], Čyras expressed the knowledge of *Cakes* in p_ABA consisting of inference rules $\mathcal{R}= \{\overline{\mathtt{b}} \leftarrow \mathtt{a}, \mathtt{c}\}$, assumptions $\mathcal{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ and preference $\mathtt{a} < \mathtt{b}$, and claimed that p_ABA cannot obtain its solution since the given preference doesn't work in his p_ABA whose underlying ABA has a unique extension. On the other hand, he expressed the knowledge in $ASPIC^+$ consisting of the strict rules $\mathcal{R}_s = \{\mathtt{a}, \mathtt{c} \rightarrow \neg\mathtt{b},\ \mathtt{a}, \mathtt{b} \rightarrow \neg\mathtt{c},\ \mathtt{b}, \mathtt{c} \rightarrow \neg\mathtt{a}\}$, premises $\mathcal{K}_p = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$ and preference $\mathtt{a} < \mathtt{b}$, and concluded that the $ASPIC^+$ cannot obtain the solution since three extensions exist under the Elitist comparison (resp. the Democratic comparison). Now recall the Prakken and Modgil's result [17,18] that ABA is a special case of $ASPIC^+$ with only strict inference rules $\mathcal{R}_s$, premises $\mathcal{K}_p$ [17] (or assumptions $\mathcal{K}_a$ [18]) and no preferences. However for *Cakes*, there is no correspondence between his $ASPIC^+$ except preferences (i.e. $\mathcal{R}_s$) and the underlying ABA of his p_ABA (i.e. $\mathcal{R}$). This indicates that Čyras' knowledge representation for *Cakes* is problematic. Thereby for *Cakes*, let us construct the p_ABA from his $ASPIC^+$ according to Prakken and Modgil's result. Then we obtain p_ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ \ }, \preceq \rangle$, where $\mathcal{R} = \{\neg\mathtt{b} \leftarrow \mathtt{a}, \mathtt{c},\ \neg\mathtt{c} \leftarrow \mathtt{a}, \mathtt{b},\ \neg\mathtt{a} \leftarrow \mathtt{b}, \mathtt{c}\}$, $\mathcal{A} = \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}$, $\overline{\mathtt{a}} = \neg\mathtt{a}$, $\overline{\mathtt{b}} = \neg\mathtt{b}$, $\overline{\mathtt{c}} = \neg\mathtt{c}$ and $\mathtt{a} \preceq \mathtt{b}$. We can construct arguments $\mathtt{A}\colon \{\mathtt{a}\} \vdash \mathtt{a}$, $\mathtt{B}\colon \{\mathtt{b}\} \vdash \mathtt{b}$, $\mathtt{C}\colon \{\mathtt{c}\} \vdash \mathtt{c}$, $\mathtt{A}'\colon \{\mathtt{b}, \mathtt{c}\} \vdash \neg\mathtt{a}$, $\mathtt{B}'\colon \{\mathtt{a}, \mathtt{c}\} \vdash \neg\mathtt{b}$, $\mathtt{C}'\colon \{\mathtt{a}, \mathtt{b}\} \vdash \neg\mathtt{c}$, and obtain $attacks = \{(\mathtt{C}', \mathtt{C}), (\mathtt{C}', \mathtt{A}'), (\mathtt{C}', \mathtt{B}'), (\mathtt{B}', \mathtt{B}), (\mathtt{B}', \mathtt{A}'), (\mathtt{B}', \mathtt{C}'), (\mathtt{A}', \mathtt{A}), (\mathtt{A}', \mathtt{B}'), (\mathtt{A}', \mathtt{C}')\}$. Its associated ABA has three extensions: $\mathtt{E}_1 = \{\mathtt{C}, \mathtt{A}, \mathtt{B}'\}$, $\mathtt{E}_2 = \{\mathtt{B}, \mathtt{C}, \mathtt{A}'\}$, $\mathtt{E}_3 = \{\mathtt{A}, \mathtt{B}, \mathtt{C}'\}$. Since $\mathtt{E}_1 \sqsubseteq_{ex} \mathtt{E}_2$ is derived due to $\mathtt{a} \preceq \mathtt{b}$, both $\mathtt{E}_2$ and $\mathtt{E}_3$ (resp. $\mathtt{Args2Asms}(\mathtt{E}_2) = \{\mathtt{b}, \mathtt{c}\}$, $\mathtt{Args2Asms}(\mathtt{E}_3) = \{\mathtt{a}, \mathtt{b}\}$) is obtained as the preferred and stable $\mathcal{P}$-argument extensions (resp. $\mathcal{P}$-assumption extensions). This means that the solution is obtained based on the p_ABA reconstructed from his $ASPIC^+$.

However it should be noted that the constraints no. 2 and no. 1 (b) in *Cakes* are not expressed in $\mathcal{R}_s$ of his $ASPIC^+$, while constraints no. 1 (a) and no. 1 (b) are not expressed in $\mathcal{R}$ of his p_ABA. In the following, we show that the solution of *Cakes* can be obtained from each of three different p_ABAs respectively where the knowledge of *Cakes* is expressed in three different ways.

4.2 Solving Cakes Example Based on the Semantics of P_ABA

p_ABA with the contrary function $\mathcal{C}$ gives us the solution of *Cakes* as follows.

Example 3 (Cont. Example 2). Suppose that a, b, c stand for a piece of Almond cake, a piece of Brownie, and a piece of Cheesecake respectively, while a_i (resp. b_i, c_i) $(1 \leq i \leq 2)$ stands for the solution of the problem such that Almond cake (resp. Brownie cake, Cheesecake) is taken at i-th round, where $\mathcal{A} = \{a_1, b_1, c_1, a_2, b_2, c_2\}$. Moreover the symbol t_x1 (resp. t_x2) denotes the operation such that $x \in \{a, b, c\}$ is taken in the first (resp. second) round, and the symbol t_xy1 denotes the operation such that both $x \in \{a, b, c\}$ and $y \in \{a, b, c\}$ where $x \neq y$ are taken in the first round according to the rules of the game. Then the cake example is modeled in p_ABA $\mathcal{F}_{\mathtt{pABA}} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$, where

- $\mathcal{R}= \{t\_ab1 \leftarrow a_1, b_1,\quad t\_bc1 \leftarrow b_1, c_1,\quad t\_ca1 \leftarrow c_1, a_1,\quad t\_a1 \leftarrow a_1,$
  $t\_b1 \leftarrow b_1,\quad t\_c1 \leftarrow c_1,\quad t\_a2 \leftarrow a_2,\quad t\_b2 \leftarrow b_2,\quad t\_c2 \leftarrow c_2\}$

– $\mathcal{A} = \{a_1, b_1, c_1, a_2, b_2, c_2\}$,
– $\mathcal{C}(a_1) = \{t\_a2, t\_bc1\}$, $\mathcal{C}(b_1) = \{t\_b2, t\_ca1\}$, $\mathcal{C}(c_1) = \{t\_c2, t\_ab1\}$, $\mathcal{C}(a_2) = \{t\_a1, t\_b2, t\_c2\}$, $\mathcal{C}(b_2) = \{t\_b1, t\_a2, t\_c2, t\_ca1\}$, $\mathcal{C}(c_2) = \{t\_c1, t\_a2, t\_b2\}$ and $a_i \preceq b_j$, $a_i \preceq a_i$, $b_j \preceq b_j$ $(1 \leq i, j \leq 2)$.

15 arguments are constructed in $\mathcal{F}_{\texttt{pABA}}$ as follows.
- $A_1 : \{a_1\} \vdash t\_a1$ • $A_2 : \{a_2\} \vdash t\_a2$ • $B_1 : \{b_1\} \vdash t\_b1$
- $B_2 : \{b_2\} \vdash t\_b2$ • $C_1 : \{c_1\} \vdash t\_c1$ • $C_2 : \{c_2\} \vdash t\_c2$
- $AB : \{a_1, b_1\} \vdash t\_ab1$ • $BC : \{b_1, c_1\} \vdash t\_bc1$ • $CA : \{c_1, a_1\} \vdash t\_ca1$
- $\alpha_1 : \{a_1\} \vdash a_1$ • $\alpha_2 : \{a_2\} \vdash a_2$ • $\beta_1 : \{b_1\} \vdash b_1$ • $\beta_2 : \{b_2\} \vdash b_2$
- $\gamma_1 : \{c_1\} \vdash c_1$ • $\gamma_2 : \{c_2\} \vdash c_2$

Then the associated ABA of $\mathcal{F}_{\texttt{pABA}}$ has preferred and stable extensions as follows:

$E_1 = \{A_1, C_1, \alpha_1, \gamma_1, CA\}$, with $\texttt{Args2Asms}(E_1) = \{a_1, c_1\}$
$E_2 = \{A_2, B_1, C_1, \alpha_2, \beta_1, \gamma_1, BC\}$, with $\texttt{Args2Asms}(E_2) = \{a_2, b_1, c_1\}$
$E_3 = \{A_1, B_1, C_2, \alpha_1, \beta_1, \gamma_2, AB\}$, with $\texttt{Args2Asms}(E_3) = \{a_1, b_1, c_2\}$

Since $E_1 \sqsubseteq_{ex} E_2$ is derived due to $\alpha_i \leq \beta_j$ or $a_i \preceq b_j$ $(1 \leq i, j \leq 2)$, both E_2 and E_3 (resp. $\{a_2, b_1, c_1\}$ and $\{a_1, b_1, c_2\}$) are obtained as preferred and stable $\mathcal{P}$-argument extensions (resp. $\mathcal{P}$-assumption extensions) in $\mathcal{F}_{\texttt{pABA}}$. Hence E_2 and E_3 give us solution of *Cakes* that either a pair of Brownie cake and Cheesecake or a pair of Almond and Brownie cakes is chosen in the first round.

4.3 Assumption-Based Argumentation Equipped with Preferences and Constraints

In this subsection, we present a general method to express some kind of constraints in ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$ as well as p_ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{}, \preceq \rangle$. Moreover we show that the solution of *Cakes* is also obtained by applying the method to p_ABA.

Definition 7 (Constraints). *Given an ABA framework $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$, a rule without head of the form:*

$$\leftarrow a_1, \ldots, a_m \qquad \text{(or equivalently } \leftarrow \{a_1, \ldots, a_m\})$$

is called a constraint, where $a_i \in \mathcal{A}$ $(1 \leq i \leq m)$.

In general, let $b_i \in \mathcal{L}$ such that there exists an argument $B_i \vdash b_i$ where $B_i \neq \emptyset$. Then $\leftarrow b_1, \ldots, b_m$, or equivalently $\leftarrow \bigcup_{i=1}^{m} \{b_i\}$ stands for a set of the constraints $\leftarrow \bigcup_{i=1}^{m} B_i$ obtained by replacing $\{b_i\}$ with $B_i \subseteq \mathcal{A}$ in every possible way.

Satisfaction of constraints is defined as follows.

Definition 8 (Satisfaction).

– *A set of assumptions $\mathcal{A}sms \subseteq \mathcal{A}$ satisfies a constraint $\leftarrow a_1, \ldots, a_m$ iff $\{a_1, \ldots, a_m\} \not\subseteq \mathcal{A}sms$ holds.*
– *A set of assumptions $\mathcal{A}sms \subseteq \mathcal{A}$ satisfies a set of constraints $\mathtt{C}$ iff $\{a_1, \ldots, a_m\} \not\subseteq \mathcal{A}sms$ holds for $\forall \leftarrow a_1, \ldots, a_m \in \mathtt{C}$.*

Definition 9 (ABA equipped with constraints). *Given an ABA framework $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$ and a set of constraints $\mathtt{C}$, an ABA framework $\mathcal{F}_{\mathtt{C}}$ equipped with constraints is defined as*

$$\mathcal{F}_{\mathtt{C}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, \overline{} \rangle, \text{ where}$$

$\mathcal{R}_{\mathtt{C}} = \{\neg a_i \leftarrow a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_m | \leftarrow a_1, \ldots, a_m \in \mathtt{C},\ \bar{a_i} = \neg a_i, 1 \leq i \leq m\}$.

Constrains defined in Definition 7 help users in expressing knowledge. And furthermore they are useful to eliminate undesirable *Sname* extensions in ABA just as integrity constraints are used to eliminate undesirable answer sets in answer set programming [19,20]. (Details are omitted due to limitations of space.)

The following properties hold for $\mathcal{F}_c$.

Theorem 2. *$\mathcal{A}sms \subseteq \mathcal{A}$ is conflict-free in $\mathcal{F}_{\mathtt{C}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, \overline{} \rangle$ if and only if $\mathcal{A}sms$ is conflict-free in $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$ and satisfies a set of constraints $\mathtt{C}$, namely $\{a_1, \ldots, a_m\} \not\subseteq \mathcal{A}sms$ for $\forall \leftarrow a_1, \ldots, a_m \in \mathtt{C}$.*

Proof. See Appendix. □

Theorem 3. *A conflict-free set $\mathcal{A}sms$ in $\mathcal{F}_{\mathtt{C}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, \overline{} \rangle$ satisfies a set of constraints $\mathtt{C}$.*

Proof. See Appendix. □

Proposition 1. *In $\mathcal{F}_{\mathtt{C}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, \overline{} \rangle$, every Sname assumption extension $\mathcal{A}sms \subseteq \mathcal{A}$ satisfies a set of constraints $\mathtt{C}$.*

Proof. This is obviously proved based on Theorem 3. □

Example 4 (Cont. Example 2). Let us express the knowledge of *Cakes* except preferences (i.e. the game rule no. 3) in ABA. Suppose that a_i (resp. b_i, c_i) ($1 \leq i \leq 2$) stands for the solution of the problem such that Almond cake (resp. Brownie cake, Cheesecake) is taken at i-th round. Let $\mathcal{A} = \{a_1, b_1, c_1, a_2, b_2, c_2\}$ and $\bar{a_i} = \neg a_i$, $\bar{b_i} = \neg b_i$, $\bar{c_i} = \neg c_i$, where $\neg a_i$ (resp. $\neg b_i$, $\neg c_i$) denotes the negation of a_i (resp. b_i, c_i). Then

- the game rule no. 1 (a) is expressed by four constraints as follows:
  $\leftarrow a_1, b_1, c_1 \quad \leftarrow a_1, a_2 \quad \leftarrow b_1, b_2 \quad \leftarrow c_1, c_2$
- The game rule no. 1 (b) is expressed by three constraints as follows:
  $\leftarrow a_2, b_2 \quad \leftarrow b_2, c_2 \quad \leftarrow c_2, a_2$
- The game rule no. 2 is expressed by the rule as follows:
  $\neg b_2 \leftarrow c_1, a_1$

Cakes except preferences is modeled in ABA $\mathcal{F}_{\mathtt{C}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, ^{-} \rangle$, where

– $\mathcal{R} = \{\neg b_2 \leftarrow c_1, a_1\}$
– $\mathcal{R}_{\mathtt{C}} = \{\neg c_1 \leftarrow a_1, b_1, \quad \neg a_1 \leftarrow b_1, c_1, \quad \neg b_1 \leftarrow c_1, a_1, \quad \neg a_2 \leftarrow a_1,$
$\neg a_1 \leftarrow a_2, \quad \neg b_2 \leftarrow b_1, \quad \neg b_1 \leftarrow b_2, \quad \neg c_2 \leftarrow c_1, \quad \neg c_1 \leftarrow c_2,$
$\neg b_2 \leftarrow a_2, \quad \neg a_2 \leftarrow b_2, \quad \neg c_2 \leftarrow b_2, \quad \neg b_2 \leftarrow c_2, \quad \neg a_2 \leftarrow c_2, \quad \neg c_2 \leftarrow a_2\}$
– $\mathcal{A} = \{a_1, b_1, c_1, a_2, b_2, c_2\}$, and $\bar{a_i} = \neg a_i$, $\bar{b_i} = \neg b_i$, $\bar{c_i} = \neg c_i$ $(i = 1, 2)$.

22 arguments are constructed in $\mathcal{F}_{\mathtt{C}}$ as follows.

- $A_1 : \{a_1\} \vdash \neg a_2$ • $A_2 : \{a_2\} \vdash \neg a_1$ • $B_1 : \{b_1\} \vdash \neg b_2$
- $B_2 : \{b_2\} \vdash \neg b_1$ • $C_1 : \{c_1\} \vdash \neg c_2$ • $C_2 : \{c_2\} \vdash \neg c_1$
- $AB : \{a_1, b_1\} \vdash \neg c_1$ • $BC : \{b_1, c_1\} \vdash \neg a_1$ • $CA : \{c_1, a_1\} \vdash \neg b_1$
- $\alpha_1 : \{a_1\} \vdash a_1$ • $\alpha_2 : \{a_2\} \vdash a_2$ • $\beta_1 : \{b_1\} \vdash b_1$ • $\beta_2 : \{b_2\} \vdash b_2$
- $\gamma_1 : \{c_1\} \vdash c_1$ • $\gamma_2 : \{c_2\} \vdash c_2$ • $CA2 : \{c_1, a_1\} \vdash \neg b_2$
- $P_1 : \{a_2\} \vdash \neg b_2$ • $P_2 : \{a_2\} \vdash \neg c_2$ • $Q_1 : \{b_2\} \vdash \neg a_2$
- $Q_2 : \{b_2\} \vdash \neg c_2$ • $R_1 : \{c_2\} \vdash \neg b_2$ • $R_2 : \{c_2\} \vdash \neg a_2$

Thus $\mathcal{F}_{\mathtt{C}}$ has three preferred and stable argument extensions E_i (resp. assumption extensions $asms_i = \mathtt{Args2Asms}(E_i)$ satisfying constraints) $(1 \le i \le 3)$ as follows:

$$E_1 = \{A_1, C_1, \alpha_1, \gamma_1, CA, CA2\}, \quad \text{with } asms_1 = \{a_1, c_1\}$$
$$E_2 = \{A_2, B_1, C_1, \alpha_2, \beta_1, \gamma_1, BC, P_1, P_2\}, \quad \text{with } asms_2 = \{a_2, b_1, c_1\}$$
$$E_3 = \{A_1, B_1, C_2, \alpha_1, \beta_1, \gamma_2, AB, R_1, R_2\}, \quad \text{with } asms_3 = \{a_1, b_1, c_2\}$$

Note that E_1 (resp. E_2, E_3) as well as $asms_1$ (resp. $asms_2$, $asms_3$) denote that a pair of Almond and Cheesecake cakes (resp. a pair of Brownie cake and Cheesecake, a pair of Almond and Brownie cakes) is chosen in the first round.

Definition 10 (ABA equipped with preferences and constraints). *Given an ABA framework $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, ^{-} \rangle$, a set of constraints* C *and a sentence ordering $\preceq \subseteq \mathcal{L} \times \mathcal{L}$, an ABA framework $\mathcal{F}_{\mathtt{PC}}$ equipped with preferences $\preceq$ and constraints* C *is defined as*

$$\mathcal{F}_{\mathtt{PC}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, ^{-}, \preceq \rangle, \text{ where}$$

$\mathcal{R}_{\mathtt{C}} = \{\neg a_i \leftarrow a_1, \ldots, a_{i-1}, a_{i+1}, \ldots, a_m | \leftarrow a_1, \ldots, a_m \in \mathtt{C}, \ \bar{a_i} = \neg a_i, 1 \le i \le m\}$.

Example 5 (Cont. Example 4). By incorporating the preferences given in *Cakes*, i.e. $a_i \preceq b_j$ $(1 \le i, j \le 2)$ into the ABA $\mathcal{F}_{\mathtt{C}}$ shown in Example 4, we obtain p_ABA $\mathcal{F}_{\mathtt{PC}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, ^{-}, \preceq \rangle$, where $\mathcal{F}_{\mathtt{C}} = \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\mathtt{C}}, \mathcal{A}, ^{-} \rangle$ has three preferred (resp. stable) argument extensions E_1, E_2, E_3 as shown in Example 4.

Since $E_1 \sqsubseteq_{ex} E_2$ is derived due to $\alpha_i \le \beta_j$ or $a_i \preceq b_j$ $(1 \le i, j \le 2)$, both E_2 and E_3 (resp. $\{a_2, b_1, c_1\}$ and $\{a_1, b_1, c_2\}$) are obtained as preferred and stable $\mathcal{P}$-argument extensions (resp. $\mathcal{P}$-assumption extensions) in $\mathcal{F}_{\mathtt{PC}}$. Accordingly we again obtain the solution of *Cakes*.

4.4 Prioritized Logic Programming As Argumentation Equipped with Preferences

In [23], we showed that p_ABA can capture Sakama and Inoue's preferred answer sets of a prioritized logic program (PLP) [19]. Hence we show that the PLP expressing the knowledge of *Cakes* as well as the p_ABA instantiated with the PLP enable us to obtain its solution based on the respective semantics as follows.

Example 6 (Cont. Example 2). Let a_i (resp. b_i, c_i) $(1 \le i \le 2)$ be a propositional atom which means that Almond cake (resp. Brownie cake, Cheesecake) is taken at i-th round. Then

- The game rule no. 1 (a) is expressed by rules of a normal logic program as follows:
  $b_1 \leftarrow not\ a_1,\quad c_1 \leftarrow not\ a_1,\quad c_1 \leftarrow not\ b_1,\quad a_1 \leftarrow not\ b_1,\quad a_1 \leftarrow not\ c_1,$
  $b_1 \leftarrow not\ c_1$
- The game rules no. 1 (b) and no. 2 are expressed by rules as follows:
  $a_2 \leftarrow not\ a_1,\quad b_2 \leftarrow not\ a_1,\ not\ c_1,\quad c_2 \leftarrow not\ c_1$
- The game rule no. 3 is expressed by $a_i \preceq b_j$ $(1 \le i, j \le 2)$.

These lead to PLP (P, Φ) as follows:

$$P = \{b_1 \leftarrow not\ a_1,\ \ c_1 \leftarrow not\ a_1,\ \ c_1 \leftarrow not\ b_1,\ \ a_1 \leftarrow not\ b_1,\ \ a_1 \leftarrow not\ c_1,$$
$$b_1 \leftarrow not\ c_1,\ \ a_2 \leftarrow not\ a_1,\ \ b_2 \leftarrow not\ a_1, not\ c_1,\ \ c_2 \leftarrow not\ c_1\}$$
$$\Phi = \{(a_i, b_j) | 1 \le i, j \le 2)\}.$$

P has three answer sets (i.e. stable models) S_i $(1 \le i \le 3)$ as follows:

$$S_1 = \{a_1, c_1\},\quad S_2 = \{b_1, c_1, a_2\},\quad S_3 = \{a_1, b_1, c_2\}.$$

$S_1 \sqsubseteq_{as} S_2$ is derived due to Φ^* [19]. Hence S_2 and S_3 corresponding to $asms_2$ and $asms_3$ in Example 5 are obtained as preferred answer sets of the PLP (P,Φ).

On the other hand, according to [23, Corollary 2] (i.e. [22, Theorem 2]), we can construct the p_ABA $\mathcal{F}_{\texttt{PLP}} = \langle \mathcal{L}_P, P, \mathcal{A}, \overline{\ \ }, \Phi^* \rangle$ instantiated with this PLP, where $\mathcal{A} = HB_{not} = \{not\ p \mid p \in HB_P\}$ for $HB_P = \{a_1, b_1, c_1, a_2, b_2, c_2\}$, $\mathcal{L}_P = HB_P \cup HB_{not}$, $\overline{not\ p} = p$ for $p \in HB_P$ and Φ^* is the reflexive and transitive closure of $\Phi = \{(a_i, b_j)|1 \le i, j \le 2)\}$. Then as indicated by [23, Corollary 2], we can obtain two stable $\mathcal{P}$-argument extensions $\texttt{E}_2$ and $\texttt{E}_3$ of $\mathcal{F}_{\texttt{PLP}}$ with

$$\texttt{Concs}(\texttt{E}_2) = \{b_1, c_1, a_2, not\ a_1, not\ b_2, not\ c_2\},$$
$$\texttt{Concs}(\texttt{E}_3) = \{a_1, b_1, c_2, not\ c_1, not\ a_2, not\ b_2\}$$

corresponding to S_2 and S_3. As a result, we again obtain the solution of *Cakes*.

5 Related Work

Čyras and Toni [5] proposed an ABA$^+$ framework: $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ \ }, \leqslant \rangle$, where $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{\ \ } \rangle$ is an ABA framework and $\leqslant$ is a preorder on $\mathcal{A}$. They newly introduced $<$-*attacks* $\subseteq \mathcal{P}(\mathcal{A}) \times \mathcal{P}(\mathcal{A})$ consisting of two types depending on $\leqslant$. Its semantics is given by a $<$-*Sname* extension $E \subseteq \mathcal{A}$ as defined by replacing the notion of *attacks* with <-*attacks* in standard ABA. Compared ABA^+ with p_ABA, the form of ABA^+ is a special case of p_ABA $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \mathcal{C}, \preceq \rangle$ as far as its underlying ABA is flat because $\leqslant \subseteq \mathcal{A} \times \mathcal{A}$ is a subset of $\preceq \subseteq \mathcal{L} \times \mathcal{L}$ and $^-$ is a special case of $\mathcal{C}$ s.t. $|\mathcal{C}(\alpha)|=1$ for $\forall \alpha \in \mathcal{A}$. Hence none of preferences over hypotheses (e.g. (1), (2)), preferences over goals $\mathtt{G} \subseteq \mathcal{L} \setminus \mathcal{A}$ which are often required in decision-making and practical reasoning, and preferences on (defeasible) rules for epistemic reasoning can be expressed in ABA^+. In contrast, p_ABA has a mechanism to represent and reason with all of these preferences in its framework [22,23]. Therefore p_ABA has far much more expressive power than ABA^+.

Prakken proposed $ASPIC^+$ for structured argumentation with preferences [17,18]. Comparison between $ASPIC^+$ and p_ABA is discussed in detail in [23].

Dung [11,12] proposed a new approach of structured argumentation with priorities for $ASPIC^+$-type argumentation formalisms. A novel attack relation (assignment) called *regular* [12] (resp. *normal* [11]) which takes account of priorities over defeasible rules is defined without constructing argument orderings.

Coste-Marquis et al. proposed constrained argumentation frameworks [3] where constraints on admissible arguments in abstract argumentation are considered. Instead in our approach, constraints on assumptions expressed by rules without head can be treated in ABA and p_ABA as shown in Subsect. 4.3.

6 Discussion and Conclusion

Čyras and Toni claimed that p_ABA cannot solve two examples (i.e. [5, Example 1], [4, Example 1]) since the given preferences do not work in their p_As whose underlying ABAs have a unique extension, and proposed ABA^+. Against their claim, it is shown in Sects. 3 and 4 that p_ABAs in which we encoded the respective knowledge give us solutions of them without any difficulties. In conclusion, they could not obtain the solutions of these examples not due to the property of p_ABA but due to their incorrect knowledge encodings in p_ABA.

In what follows, we show that the semantics of ABA^+ has a serious problem as to treating preferences. As addressed in Example 1, Čyras and Toni presented ABA^+ consisting of $\mathcal{R} = \{\overline{\alpha} \leftarrow \beta\}$, $\mathcal{A} = \{\alpha, \beta\}$, $\beta \leqslant \alpha$. Based on its semantics, $\{\alpha\}$ is selected as a unique <-complete extension due to $\beta \leqslant \alpha$ though $\langle \mathcal{L}, \mathcal{R}, \mathcal{A}, ^- \rangle$ has a unique extension $\{\beta\}$ [5]. Now consider a real world problem as follows.

Example 7. Usually the famous legal principle: "innocent until proven guilty" is applied to the suspect under no evidence.

It is expressed by ABA $\mathcal{F}$ consisting of $\mathcal{R} = \{innocent \leftarrow not\ guilty\}$, $\mathcal{A} = \{not\ innocent, not\ guilty\}$, $\overline{not\ innocent} = innocent$, $\overline{not\ guilty} = guilty$.

Hereupon suppose that someone prefers "not innocent" to "not guilty" (since he prefers "guilty" to "innocent") though there is no evidence proving the suspect is guilty. Obviously under this situation, the human legal reasoning result in court is "innocent" regardless of any preference for innocence or guilt because there is no evidence of "guilty". On the other hand, given the preference s.t.

"*not guilty* $\leqslant$ *not innocent*" along with $\mathcal{F}$, ABA$^+$ has $\{not\ innocent\}$ as its unique extension, whereas p_ABA has the unique $\mathcal{P}$-argument extension E with $\texttt{Concs}(E) = \{innocent, not\ guilty\}$ regardless of preferences since the underlying ABA has the unique extension E. Thus p_ABA yields "innocent", while ABA$^+$ yields "not innocent". Hence thanks to the property discussed in the introduction, p_ABA gives us the human legal reasoning result, i.e. "innocent, whereas ABA$^+$ cannot perform such typical non-monotonic reasoning with preferences.

In contrast, according to the correspondence between $ASPIC^+$ and ABA [17,18], this ABA$^+$ can be faithfully mapped to $ASPIC^+$ consisting of $\mathcal{R}_s = \{\beta \rightarrow \neg\alpha\}$, $\mathcal{K}_p = \{\alpha, \beta\}$ (or $\mathcal{K}_a = \{\alpha, \beta\}$), $\overline{\alpha} = \neg\alpha$ and $\beta \leq^{'} \alpha$. As for the case $\mathcal{K}_p = \{\alpha, \beta\}$, $defeat = \emptyset$ is derived. Then the mapped $ASPIC^+$ has a unique complete extension E^+ with $\texttt{concs}(E^+) = \{\alpha, \beta, \neg\alpha\}$, which is not directly consistent [17,18]. Similarly this ABA$^+$ may be also mapped to Dung's rule-based system [11,12], say $\mathcal{R}_{\text{dung}}$, which consists of $d_0\colon \Rightarrow \beta \quad d_1\colon \Rightarrow \alpha\ r : \beta \rightarrow \neg\alpha$ and $d_0 \prec d_1$. Surprisingly when we replace the symbol β (resp. α) with a (resp. b), $\mathcal{R}_{\text{dung}}$ coincides with the rule-based system shown in [12, Example 7] in which no regular attack relation assignment exists as discussed by Dung [12].

We are the first to show *prioritized logic programming as argumentation equipped with preferences* (cf. Subsect. 4.4) [23] as Dung showed logic programming as argumentation [7]. Nevertheless our future work is to explore the other types of the semantics for p_ABA so that it can capture the other types of prioritized logic programming such as Brewka and Eiter's preferred answer sets, Delgrande, Schaub and Tompits' preferred answer sets and so on [6].

Acknowledgments. This work was supported by KAKENHI (Grant-in-Aid for Scientific Research(S)17H06103).

Appendix

Proof of Theorem 2 ($\Longleftarrow$). Let $\mathcal{A}sms$ be conflict-free in $\mathcal{F}$ and satisfies $\{a_1, \ldots, a_m\} \not\subseteq \mathcal{A}sms$ for $\forall \leftarrow a_1, \ldots, a_m \in \texttt{C}$. Since $\mathcal{A}sms$ is conflict-free in $\mathcal{F}$, $\forall\alpha \in \mathcal{A}sms$ is not attacked by arguments constructed by using only rules from $\mathcal{R}$ in $\mathcal{F}_{\texttt{C}}$. Now suppose that $\mathcal{A}sms$ is not conflict-free in $\mathcal{F}_{\texttt{C}}$. Then for some $\{a_1, \ldots, a_{k-1}, a_{k+1}, \ldots, a_m\} \subseteq \mathcal{A}sms$, there exists an argument $\{a_1, \ldots, a_{k-1}, a_{k+1}, \ \ldots, a_m\} \vdash \neg a_k$ constructed by the rule from $\mathcal{R}_{\mathcal{C}}$ that attacks $\mathcal{A}sms$, which denotes that a_k $(1 \leq k \leq m)$ is in $\mathcal{A}sms$. Thus $\{a_1, \ldots, a_{k-1}, a_k, a_{k+1}, \ldots, a_m\} \subseteq \mathcal{A}sms$ is derived. This contradicts that $\mathcal{A}sms$ satisfies $\{a_1, \ldots, a_m\} \not\subseteq \mathcal{A}sms$ for $\forall \leftarrow a_1, \ldots, a_m \in \texttt{C}$. Thus it is derived that $\mathcal{A}sms$ is conflict-free in $\mathcal{F}_{\texttt{C}}$.

($\Longrightarrow$) Let $\mathcal{A}sms$ be conflict-free in $\mathcal{F}_{\texttt{C}}= \langle \mathcal{L}, \mathcal{R} \cup \mathcal{R}_{\texttt{C}}, \mathcal{A}, \overline{} \rangle$. Then $\mathcal{A}sms$ is also conflict-free in $\mathcal{F} = \langle \mathcal{L}, \mathcal{R}, \mathcal{A}, \overline{} \rangle$ due to $\mathcal{R} \subseteq (\mathcal{R} \cup \mathcal{R}_{\texttt{C}})$. Now suppose that for this $\mathcal{A}sms$ which is conflict-free in $\mathcal{F}$, there exists some constraint $\exists \leftarrow a_1, \ldots, a_m \in \texttt{C}$ which satisfies $\{a_1, \ldots, a_m\} \subseteq \mathcal{A}sms$. Then in $\mathcal{F}_{\texttt{C}}$, there exists some argument $\{a_1, \ldots, a_{k-1},\ a_{k+1}, \ldots, a_m\} \vdash \neg a_k$ built from $\mathcal{R}_{\texttt{C}}$ that attacks $\mathcal{A}sms$. This contradicts that $\mathcal{A}sms$ is conflict-free in $\mathcal{F}_{\texttt{C}}$. Hence it holds $\{a_1, \ldots, a_m\} \not\subseteq \mathcal{A}sms$ for $\forall \leftarrow a_1, \ldots, a_m \in \texttt{C}$ w.r.t. $\mathcal{A}sms$ which is the conflict-free in $\mathcal{F}$. $\square$

Proof of Theorem 3. Suppose that in $\mathcal{F}_{\mathtt{C}}$, there is some conflict-free set $\mathcal{A}sms \subseteq \mathcal{A}$ which does not satisfy some constraint in $\mathtt{C}$, that is, $\{a_1, \ldots, a_m\} \subseteq \mathcal{A}sms$ holds for $\exists \leftarrow a_1, \ldots, a_m \in \mathtt{C}$. Then using rules from $\mathcal{R}_{\mathtt{C}}$, it is possible to construct the argument $\{a_1, \ldots, a_{k-1}, a_{k+1}, \ldots, a_m\} \vdash \neg a_k$ that attacks $a_k \in \mathcal{A}sms$ $(1 \leq k \leq m)$. This contradicts that $\mathcal{A}sms$ is conflict-free. □

References

1. Bondarenko, A., Dung, P.M., Kowalski, R.A., Toni, F.: An abstract, argumentation-theoretic approach to default reasoning. Artif. Intell. **93**, 63–101 (1997)
2. Caminada, M., Sá, S., Alcântara, J., Dvořák, W.: On the difference between assumption-based argumentation and abstract argumentation. In: BNAIC 2013, pp. 25–32 (2013). IFCoLog J. Logic Appl. **2**(1), 15–34 (2015)
3. Coste-Marquis, S., Devred, C., Marquis, P.: Constrained argumentation frameworks. In: Proceedings of KR 2006, pp. 112–122 (2006)
4. Čyras, K.: Argumentation-based reasoning with preferences. In: Proceedings of PAAMS 2016, pp. 199–210 (2016)
5. Čyras, K., Toni, F.: ABA$^+$: assumption-based argumentation with preferences. In: Proceedings of KR 2016, pp. 553–556 (2016)
6. Delgrande, J.P., Schaub, T., Tompits, H., Wang, K.: A classification and survey of preference handling approaches in nonmonotonic reasoning. J. Comput. Intell. **20**(2), 308–334 (2004). Wiley
7. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artif. Intell. **77**, 321–357 (1995)
8. Dung, P.M., Mancarella, P., Toni, F.: Computing ideal sceptical argumentation. Artif. Intell. **171**(10–15), 642–674 (2007)
9. Dung, P.M., Kowalski, R.A., Toni, F.: Assumption-based argumentation. In: Simari G., Rahwan I. (eds.) Argumentation in Artificial Intelligence, pp. 199–218, Springer, Boston (2009). doi:10.1007/978-0-387-98197-0_10
10. Dung, P.M., Thang, P.M.: Closure and consistency rationalities in logic-based argumentation. In: Balduccini, M., Son, T.C. (eds.) Logic Programming, Knowledge Representation, and Nonmonotonic Reasoning. LNCS, vol. 6565, pp. 33–43. Springer, Heidelberg (2011). doi:10.1007/978-3-642-20832-4_3
11. Dung, P.M.: An axiomatic analysis of structured argumentation with priorities. Artif. Intell. **231**, 107–150 (2016)
12. Dung, P.M.: A Canonical semantics for structured argumentation with priorities. In: Proceedings of COMMA-2016, pp. 263–274 (2016)
13. Fan, X., Toni, F.: A general framework for sound assumption-based argumentation dialogues. Artif. Intell. **216**, 20–54 (2014)
14. Kakas, A.C., Kowalski, R.A., Toni, F.: Abductive logic programming. J. Logic Comput. **2**(6), 719–770 (1992)
15. Lifschitz, V.: Computing circumscription. In: Proceedings of IJCAI 1985, pp. 121–127 (1985)
16. McCarthy, J.: Applications of circumscription to formalizing commonsense knowledge. Artif. Intell. **28**, 89–116 (1986)
17. Modgil, S.J., Prakken, H.: The *ASPIC*+ framework for structured argumentation: a tutorial. Argum. Comput. **5**, 31–62 (2014)

18. Prakken, H.: An abstract framework for argumentation with structured arguments. Argum. Comput. **1**, 93–124 (2010)
19. Sakama, C., Inoue, K.: Prioritized logic programming and its application to commonsense reasoning. Artif. Intell. **123**, 185–222 (2000)
20. Sakama, C., Inoue, K.: An abductive framework for computing knowledge base updates. TPLP **3**(6), 671–713 (2003)
21. Wakaki, T., Nitta, K., Sawamura, H.: Computing abductive argumentation in answer set programming. In: McBurney, P., Rahwan, I., Parsons, S., Maudet, N. (eds.) ArgMAS 2009. LNCS, vol. 6057, pp. 195–215. Springer, Heidelberg (2010). doi:10.1007/978-3-642-12805-9_12
22. Wakaki, T.: Assumption-based argumentation equipped with preferences. In: Dam, H.K., Pitt, J., Xu, Y., Governatori, G., Ito, T. (eds.) PRIMA 2014. LNCS, vol. 8861, pp. 116–132. Springer, Cham (2014). doi:10.1007/978-3-319-13191-7_10
23. Wakaki, T.: Assumption-based argumentation equipped with preferences and its application to decision-making, practical reasoning, and epistemic reasoning. J. Comput. Intell,. doi:10.1111/coin.12111, Accepted 17 October 2016. Published early view article (online version) on 20 March 2017. Wiley (2017)