

Arguing about end-of-life of packagings: Preferences to the Rescue

Bruno Yun¹, Pierre Bisquet², Patrice Buche², Madalina Croitoru¹

¹ LIRMM, University of Montpellier, France,

²IATE, INRA, France,

Abstract. Argumentation methods and associated tools permit to analyze arguments against or in favor of a set of alternatives under discussion. The outputs of the argument methods are sets of conflict-free arguments collectively defending each other, called extensions. In case of multiple extensions, it is often difficult to select one out of many alternatives. We present in this paper the implementation of an complementary approach which permits to filter or rank extensions according to the expression of preferences. Methods and tools are illustrated on a real use case in food packagings. The aim is to help the industry choose among different end-of-life possibilities by linking together consumer behavior insights, socio-economic developments and technical properties of packagings. The tool has been used on a real use-case concerning end-of-life possibilities for packagings.

1 Introduction

Communication is a pillar of our society, humans have always been concerned with debating and arguing as it constitutes a great part of our daily interactions. Argumentation dialogues are of important effect on our lives as it is implied in debates and decision-making. It is within such argumentation dialogues that opinions of different stakeholders are confronted against each other and arguments are advanced to support them [8]. One can then extract the several coherent viewpoints from an argumentation framework called extensions (sets of conflict-free arguments collectively defending each others). For a more detailed formalization of arguments other than the one of abstract argumentation, one can take the road of structured argumentation where the construction of arguments is based on a formal language. In this approach, arguments have a specific structure and attacks are defined with respect to this structure.

Within the framework of the European project EcoBioCap [11–13], a Decision Support System (DSS) based on the *ASPIC+ argumentation framework* has been implemented as a java GXT/GWT web application¹. This DSS takes as input a collection of textual arguments in favor or against a set of alternatives under debate. It implements the entire process from argument elicitation to extension computation and it also provides several GUIs for visualization purposes. The process is composed of four steps: formalizing text arguments, processing arguments, computing extensions. Hereinafter,

¹ Accessible online at <http://pfl.grignon.inra.fr/EcoBioCapProduction/> (although the access is restricted)

some user interfaces are displayed showing the obtained result in the case of the viewpoint “end of life” within EcoBioCap. The main interface of the system is illustrated in Figure 1. It is divided into five zones. Zone 1 corresponds to the task bar implementing user accounts management and general functions applied on projects (create, load, close, refresh, export, etc.). Zone 2 lists the text arguments by stakeholders. Zone 3 displays the extracted concepts and rules from the text arguments; they are also listed by stakeholders. Zone 4 displays the graphical representation of the formalized concepts and arguments. Zone 5 is a notification area displaying the computed conflicts and extensions.

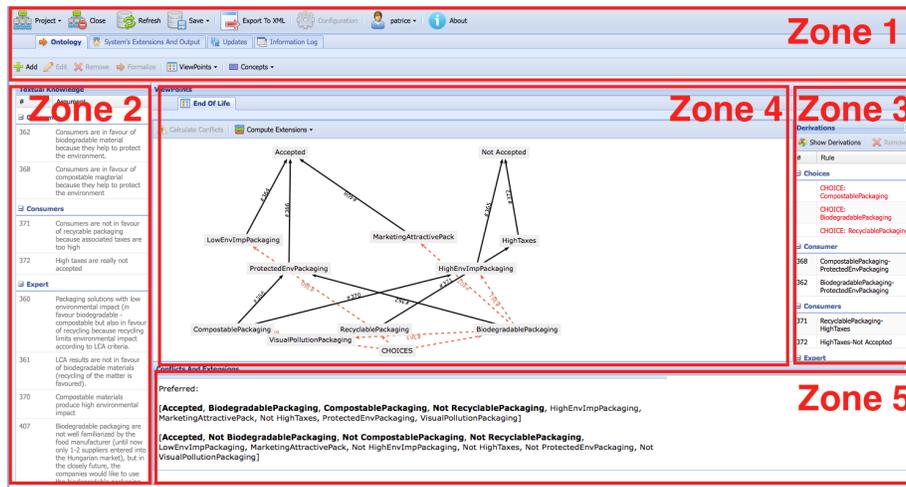


Fig. 1: Main interface of the argumentation system.

It was decided that this decision support system based on argumentation could be used to select the best end-of-life according to possibly conflicting requirements provided by multiples stakeholders. For instance, one can discuss the pros and cons of incineration, anaerobic digestion or landfill for the end-of-life of a packaging. Incineration may produce energy but may hurt the human health by producing dioxin. Packagings that are processed by anaerobic digestion will also be used to produce gases but these packagings may disturb the sorting of recyclable packagings. Likewise, landfill is a good alternative because it is low-cost but it also have long-term effects on grounds.

In this paper, we address a crucial problem for decision-making tools that are using argumentation frameworks, that is the existence of *multiple extensions*. For instance, in Zone 5 of Figure 1, there are two extensions with justifications for each of them: one promoting the use of biodegradable and compostable packagings because they protect the environment but can induce visual pollution and high environment impact, and the other promoting to not use them. Indeed, argumentation frameworks are able to extract several coherent viewpoints from the arguments but in the event that the argumentation system returns more than one extension, it is often difficult to select one out of many

alternatives. Many researchers have studied this problem and came up with various ideas. In [1], the authors suggested to vote on extensions. Another idea introduced in [9, 6] was to use preferences on pieces of information that are used to generate the arguments. These preferences can represent either the importance or the confidence of the information and are usually gathered from experts. We chose to focus on preferences as they are widely studied in the field of argumentation and constitute a simple and comprehensive way to explain decisions to users.

The next section recall the notions needed to comprehend the fundamental components of the web application, i.e. the *ASPIC+ framework* and the *propositional language*.

2 Preliminaries

In this section, we present useful notions: Dung's semantics (Section 2.1), the logical language used in this application (Section 2.2) and the *ASPIC+* argumentation framework (Section 2.3).

2.1 Dung's semantics

Here, we briefly recall the acceptability semantics introduced by Dung [8] and used in the rest of this paper.

Definition 1. *Given an argumentation framework $\mathcal{AS} = (\mathcal{A}, Att)$, where \mathcal{A} is a set of arguments and Att is a binary attack relation between arguments of \mathcal{A} . We say that an argument $a \in \mathcal{A}$ is acceptable w.r.t a set of arguments $\varepsilon \subseteq \mathcal{A}$ iff $\forall b \in \mathcal{A}$ such that $(b, a) \in Att, \exists c \in \varepsilon$ such that $(c, b) \in Att$. Moreover, an extension can follow different semantics:*

- ε is conflict-free iff $\nexists a, b \in \varepsilon$ such that $(a, b) \in Att$.
- ε is admissible iff ε is conflict-free and all arguments of ε are acceptable w.r.t ε .
- ε is preferred iff it is maximal (for set inclusion) and admissible.
- ε is stable iff it is conflict-free and $\forall a \in \mathcal{A} \setminus \varepsilon, \exists b \in \varepsilon$ such that $(b, a) \in Att$.

2.2 The language

Formally, we consider a propositional language and we denote by \mathcal{L} the set of well formed formulas of this language given the usual connectives $\wedge, \vee, \rightarrow, \neg$, the constants \perp, \top and extended with the defeasible inference \Rightarrow . The set of symbols in the language is denoted by \mathcal{V} . A *strict rule* (or strict implication) is a propositional sentence of the form $P \rightarrow Q$ where P and Q are propositions. Strict rules are important because they enable us to infer certain information from a knowledge base. Likewise, a *defeasible rule* (or defeasible implication) is a propositional sentence of the form $P \Rightarrow Q$ where P and Q are propositions. Defeasible rules represent reasonings that are not always true. A *negative constraint* (or simply a constraint) is a strict rule (resp. defeasible rule) of the form $P \wedge Q \rightarrow \perp$ (resp. $P \wedge Q \Rightarrow \perp$) where P and Q are propositions. In order to simplify the notation, we will introduce the function *Incompatible* that takes as input a set of propositions $\{P_1, \dots, P_n\}$ and returns the set of corresponding negative constraints $(P_i \wedge P_j \rightarrow \perp)$ for all pairs of propositions $(P_i, P_j), i \neq j$.

2.3 ASPIC+ argumentation system

The ASPIC+ argumentation framework was proposed as a simple tool for structured argumentation. It is based on a logical language, a set of strict and defeasible rules, a contrariness function and a preference ordering over the defeasible rules.

Definition 2. As expressed in [11–13], an ASPIC+ argumentation system is a tuple $\mathcal{AS} = (\mathcal{L}, cf, \mathcal{R}, \succeq)$ where:

- \mathcal{L} is the logical language of the system.
- cf is a contrariness function which associates to each formula f of \mathcal{L} a set of its incompatible formulas (in $2^{\mathcal{L}}$): in our case, cf corresponds to classical negation \neg .
- $\mathcal{R} = \mathcal{R}_s \cup \mathcal{R}_d$ is the set of strict (\mathcal{R}_s) and defeasible (\mathcal{R}_d) inference rules where $\mathcal{R}_s \cap \mathcal{R}_d = \emptyset$. Please note that for each strict rule $P \rightarrow Q$, the transposed rule $\neg Q \rightarrow \neg P$ is generated to ensure the completeness and the consistency of reasoning.
- \succeq is a preference ordering over defeasible rules, not used in this work.

A knowledge base in an $\mathcal{AS} = (\mathcal{L}, \mathcal{R}, cf, \succeq)$ is $\mathcal{K} \subseteq \mathcal{L}$, which contains the concepts defined in the domain and the alternative choices under discussion.

Argument structure An argument in ASPIC+ can be in two forms. Form 1 represents basic arguments that are deduced from the knowledge base. Arguments in Form 2 are more complex arguments that are constructed from other arguments using strict and defeasible rules.

Definition 3. An ASPIC+ argument A can be of the following forms:

1. $\emptyset \Rightarrow C$ with $C \in \mathcal{K}$, such that $Prem(A) = \{C\}$, $Sub(A) = \{A\}$ and $Conc(A) = C$, with $Prem$ returns premises of A , Sub returns its sub-arguments and $Conc$ returns its conclusion,
2. $A_1, \dots, A_m \rightarrow C$ (resp. $A_1, \dots, A_m \Rightarrow C$), such that there exists a strict (resp. defeasible) rule in \mathcal{R}_s (resp. \mathcal{R}_d) of the form $Conc(A_1), \dots, Conc(A_m) \rightarrow C$ (resp. $Conc(A_1), \dots, Conc(A_m) \Rightarrow c$), with $Prem(A) = Prem(A_1) \cup \dots \cup Prem(A_m)$, $Conc(A) = C$, $Sub(A) = Sub(A_1) \cup \dots \cup Sub(A_m) \cup \{A\}$.

The attack relation The engine only considers the rebutting attack as defined in [10]. This attack relation represents the incompatibility between two arguments with conflicting conclusions.

Definition 4. Argument A rebuts argument B on B' if and only if $Conc(A) \in cf(\phi)$ (where ϕ is an atom in the language) for some $B' \in Sub(B)$ of the form $B'_1, \dots, B'_m \Rightarrow \phi$. Finally, A defeats B if A rebuts B .

Example 1. Let \mathcal{AS} be an ASPIC+ argumentation framework defining the set of rules $\mathcal{R} = \mathcal{R}_s \cup \mathcal{R}_d$.

- $\mathcal{R}_s = \{BP \rightarrow HIP, \neg HIP \rightarrow \neg BP, HIP \rightarrow \neg ACC, ACC \rightarrow \neg HIP\}$
- $\mathcal{R}_d = \{BP \Rightarrow PEV, PEV \Rightarrow ACC\}$

The following structured arguments can be built on the knowledge base $\mathcal{K} = \{BP\}$:

- $A_0 : \emptyset \Rightarrow BP$
- $A_1 : A_0 \rightarrow HIP$
- $A_2 : A_1 \rightarrow \neg ACC$
- $B_1 : A_0 \Rightarrow PEV$
- $B_2 : B_1 \Rightarrow ACC$
- $B_3 : B_2 \rightarrow \neg HIP$
- $B_4 : B_3 \rightarrow \neg BP$

Following the definition of the attack, we have that argument B_4 rebuts argument A_1 on A_0 .

Stakeholder	Argument
Consumer	Consumers are in favor of biodegradable materials because they help to protect the environment .
Consumer	Consumers are in favor of compostable materials because they help to protect the environment.
Consumer	Concerning incineration, consumers express concerns because of dioxin production which has an impact on human health.
Consumer	Consumers are not ready to pay higher prices for biodegradable packagings.
Restaurateur	Restaurants are not in favors of compostable materials because they need heavy procedures to function (designated bin, trained employees and consumers).
Restaurateur	Restaurants have to contact their local composting facility to arrange a pick-up or drop-off procedure.
Expert	LCA results are in favor of recycling.
Expert	LCA results are not in favor of biodegradable materials.
Expert	In France, recyclable materials benefit from eco-tax bonus (Eco-emballage).
Expert	A European directive forbids new landfill centers in the horizon of 2020.
Researcher	Biodegradable materials could encourage people to throw their packagings in nature, causing visual pollution.
Researcher	Plastic materials cause pollution of oceans.
Researcher	The bio-polyesters (compostable) materials as PLA are disturbing PET recycling (non-organic polyesters).
Waste Management	Biodegradable materials may disturb the sorting of recyclable packagings.
Waste Management	In France, landfill is encouraged because it is low-cost. (around 80 euros per ton).
Waste Management	In France, composting is not encouraged because of high treatment cost (around 130 euros per ton).
Waste Management	Incineration (other pack) permits to produce energy.
Waste Management	Anaerobic digestion permits to produce gazes.
Waste Management	Compostable materials permit to produce fertilizers.
Waste Management	Landfill have long-term negative effects on grounds (residues heavy metals).

Fig. 2: Set of arguments obtained during meetings with experts.

3 Use-Case

In this section we will describe the use-case we obtained from several meetings with experts concerning the end-of-life of packagings. The use-case presents the text arguments (see Figure 2) given by several stakeholders (consumers, restaurateurs, etc.) regarding the end-of-life possibilities for packagings (Anaerobic digestion, Incineration, etc.). From these text arguments, we first formalized a set \mathcal{C} of *propositional constants* (also called *concepts*), corresponding to the several important notions of the text arguments (see Figure 3a) and identified a set \mathcal{A} of specific concepts that correspond to the alternative choices under discussion (AE, I, LF, C, R). Then, we formalized the inferences contained in the text arguments as strict rules and sorted concepts between positive and negative by linking them to either *Accepted* or *Not Accepted* using defeasible rules to represent that such a concept is a justification for accepting or rejecting the associated alternative. Moreover, we added negative constraints to represent that each of the end-of-life possibilities are mutually exclusive. This use-case can be represented by a knowledge base $\mathcal{K} = \{AE, I, LF, C, R\}$ in an argumentation framework $\mathcal{AS} = (\mathcal{L}, \mathcal{R}, cf, \geq)$ with $\mathcal{R} = Incompatible(\{AE, I, LF, C, R\}) \cup \mathcal{R}'$, where \mathcal{R}' is the set of rules displayed in Figure 3b.

We inputed this model in the web application presented in [11–13] and after calculation, the argumentation framework used the preferred semantics and produced five extensions. The preferred semantics was introduced in *Dung*'s seminal paper [8] alongside three other argumentation semantics: *grounded*, *stable* and *complete* semantics. We chose this semantics because it captures the intuition of the *stable semantics* and avoids its drawbacks (non-existence of extensions, etc.). Each of those preferred extensions² corresponds to one alternative:

- $\varepsilon_1 = \{\mathbf{AE}, PGS, PEV, HIP, VPL, DPR, LCD\}$
- $\varepsilon_2 = \{\mathbf{I}, DXP, PEN\}$
- $\varepsilon_3 = \{\mathbf{LF}, FNL, LCT, LTE\}$
- $\varepsilon_4 = \{\mathbf{C}, PFZ, PEV, HIP, VPL, DPR, HEP, APP, LCD\}$
- $\varepsilon_5 = \{\mathbf{R}, LCA, ETX\}$.

In [7, 6], the authors introduced the notion of base of an extension to denote the elements of the knowledge base representing the arguments of the extension. We reused this term here to represent the concepts corresponding to the alternatives and appearing in an extension. Please note that arguments of an extension ε that appear in bold are said to belong to the base of that extension (denoted by $Base(\varepsilon)$). At this point, we introduce two new methods for decision-making using preferences: refining the set of extensions using the *globally optimal extension semantics* or using *scores* to rank extensions and extract a ranking.

4 Preferences Module in Argumentation Software

After instantiating an argumentation framework, one can choose to add preferences to refine the output of the framework. Preferences can either occur in the computation of

² Please note that for simplicity purposes, we write that a concept belongs to an extension instead of writing that the argument with this concept as a conclusion is contained in the extension.

Name	Concepts
AE	Uses Anaerobic digestion
I	Uses Incineration
LF	Uses Landfill
C	Uses Compostable
R	Recycling
PEV	Protects the environment
DXP	Produces dioxin
HIP	Has higher prices
HEP	Needs heavy procedures
APP	Needs to arrange pick-up procedures
LCA	LCA results in favor
LCD	LCA results in disfavor.
ETX	Has Eco-tax
FNL	Forbids new landfills
VPL	Induces Visual pollution
LCT	Is Low-cost
DPR	Disturbs plastic recycling
PEN	Produces energy
PGS	Produces gazes
PFZ	Produces fertilizers
LTE	Causes long term effect on grounds.

(a) Concepts and their initials.

Strict rules	Defeasible rules
$AE \rightarrow PEV$	$PEV \Rightarrow Accepted$
$AE \rightarrow HIP$	$LCT \Rightarrow Accepted$
$AE \rightarrow VPL$	$LCA \Rightarrow Accepted$
$AE \rightarrow DPR$	$ETX \Rightarrow Accepted$
$AE \rightarrow LCD$	$HIP \Rightarrow NotAccepted$
$I \rightarrow DXP$	$VPL \Rightarrow NotAccepted$
$LF \rightarrow FNL$	$DPR \Rightarrow NotAccepted$
$LF \rightarrow LCT$	$LCD \Rightarrow NotAccepted$
$LF \rightarrow LTE$	$DXP \Rightarrow NotAccepted$
$C \rightarrow PEV$	$FNL \Rightarrow NotAccepted$
$C \rightarrow HIP$	$LTE \Rightarrow NotAccepted$
$C \rightarrow VPL$	$HEP \Rightarrow NotAccepted$
$C \rightarrow DPR$	$APP \Rightarrow NotAccepted$
$C \rightarrow HEP$	
$C \rightarrow APP$	
$C \rightarrow LCD$	
$C \rightarrow DPR$	
$R \rightarrow LCA$	
$R \rightarrow ETX$	

(b) Rules of the knowledge base.

Fig. 3: Rules and concepts extracted from the text arguments.

extensions or in the refining of the solutions as described in [2]. In the latter, we do not change the computation of extensions and only extract different subsets of extensions (locally optimal, Pareto optimal and globally optimal extensions) from the extensions produced. It was shown that this preference-based argumentation system satisfies rationality postulates [7]. Please note that these methods do not always produce a strict order on extensions.

In [6], preferences are viewed as a relation \geq on facts (not necessarily total) to represent the confidence we have in the pieces of information. However, it appeared that in the area of decision-making, a preference relation on the facts that are induced may be more useful because preferences are often stated on the effects of decisions rather than on the decisions themselves. A preference is a statement of the form: "I am ready to pay higher prices in order to protect the environment" and is formalized as a binary relation on concepts ($LCT < PEV$).

4.1 Refining extensions using semantics

In this section, we introduce a new method for refining a set of extensions E using semantics (locally, Pareto and globally optimal) inspired from [6]. These semantics return subsets of the original set of extensions. We introduce here the three notions.

An extension ε is said to not be locally optimal if we can find another extension ε' such that the concepts of ε are either included in ε' or dominated by elements of ε' (there is at most one concept dominated).

Definition 5. We say that an extension ε of E is locally optimal if and only if $\nexists x \in \varepsilon \setminus \text{Base}(\varepsilon)$ and $y \in \mathcal{C}$ such that $\exists \varepsilon' \in E \setminus \{\varepsilon\}, ((\varepsilon \setminus \text{Base}(\varepsilon)) \setminus \{x\}) \cup \{y\} \subseteq \varepsilon'$ and $x < y$.

An extension ε is said to not be Pareto optimal if we can find another extension ε' such that the concepts of ε are either included in ε' or dominated by elements of ε' (they are dominated by a single concept).

Definition 6. We say that an extension ε of E is Pareto optimal if and only if $\nexists X \subseteq \varepsilon \setminus \text{Base}(\varepsilon)$ and $Y \subseteq \mathcal{C}$ and $X \neq \emptyset$ such that $\exists \varepsilon' \in E \setminus \{\varepsilon\}, (((\varepsilon \setminus \text{Base}(\varepsilon)) \setminus X) \cup Y) \subseteq \varepsilon'$ and $\forall x \in X, x < y$.

An extension ε is said to not be globally optimal if we can find another extension ε' such that the concepts of ε are either included in ε' or dominated by elements of ε' (no restrictions).

Definition 7. We say that an extension ε of E is globally optimal if and only if $\nexists X \subseteq \varepsilon \setminus \text{Base}(\varepsilon)$ and $Y \subseteq \mathcal{C}$ and $X \neq \emptyset$ such that $\exists \varepsilon' \in E \setminus \{\varepsilon\}, (((\varepsilon \setminus \text{Base}(\varepsilon)) \setminus X) \cup Y) \subseteq \varepsilon'$ and $\forall x \in X, \exists y \in Y$ such that $x < y$.

These semantics enable us to obtain a simple refining, i.e. we obtain four subsets of the initial set of extensions. The following example shows the approach.

Example 2. Suppose that $E = \{\varepsilon_1, \varepsilon_2, \varepsilon_3, \varepsilon_4, \varepsilon_5\}$ is the set of extensions returned by the argumentation system as described in Section 3; we add the following preferences:

- Having good LCA results is better than producing dioxin: $DXP < LCA$,
- Beneficing of the EcoTax is preferred to producing energy: $PEN < ETX$.

The module removes the extension ε_2 corresponding to the alternative “Incineration” from the set of globally optimal extensions because the set of concepts of ε_2 (DXP, PEN) is dominated by the concepts of ε_5 . If we further add the preference:

- Producing fertilizers is more important than producing gazes: $PGS < PFZ$.

The module no longer considers the extension ε_1 corresponding to the alternative “anaerobic digestion” as being important because its concepts are included in ε_4 and PGS is dominated by PFZ . The preferences module removes it from the set of locally optimal extensions (and Pareto/globally optimal extensions). Please find the results in Figure 4.

Following this result, we can say that according to the preferences stated, the three more preferred end-of-life possibilities for packagings are “Recycling”, “Landfill” and “Compostable”. Moreover, an ordering can be deduced from these semantics:

$$\{\varepsilon_5, \varepsilon_4, \varepsilon_3\} > \{\varepsilon_1\} > \{\varepsilon_2\}$$

Not locally optimal	Locally optimal	Pareto optimal	Globally optimal
ε_2	ε_5	ε_5	ε_5
	ε_4	ε_4	ε_4
	ε_3	ε_3	ε_3
	ε_1	ε_1	

Fig. 4: Overview of the results after application of the preferences.

Note that while those semantics allow to refine the extensions, they may be unable to output only one extension as it is the case in the previous example. This is of course dependent of the preferences the user has used: the more preferences are used, the more refining is going to happen. Note as well that it is possible to use the preferences differently, namely in a more “quantitative” fashion. We study this new approach in the next section.

4.2 Ranking methods using scores

This new approach using scores is interesting in many ways. First, it is obviously easier and faster to compute than the approach introduced in [6]. Furthermore, an extension can be accurately scored (using the preferences) even if we do not have the entire set of extensions. This can be useful in the event that we do not have enough time to compute all the extensions. In this section, we introduce two scores for ranking extensions.

First scoring: High score means less dominated The first method gives the highest points to the extension that is the least dominated. Namely, the score of an extension ε is $Score_1(\varepsilon) = \sum_{a \in (\mathcal{C} \setminus Base(\varepsilon))} |\{c \in \mathcal{C} | c < a\}|$. It is obvious that with this score, the best extension is the one with the highest score. If we reuse the previous preferences, we get the following scores:

Extension	$Score_1$
ε_1	0
ε_2	0
ε_3	0
ε_4	1
ε_5	2

$\{\varepsilon_5\} > \{\varepsilon_4\} > \{\varepsilon_1, \varepsilon_2, \varepsilon_3\}$

Fig. 5: Scores obtained with $Score_1$ and the associated rank.

Second scoring: High score means more dominated The second method gives the highest points to the extension that is the most dominated. Namely, the score of an extension ε is $Score_2(\varepsilon) = \sum_{a \in (\mathcal{C} \setminus Base(\varepsilon))} |\{c \in \mathcal{C} | a < c\}|$. With this score, the best

extension is the one with the lowest score. If we reuse the previous preferences, we get the following scores:

Extension	$Score_2$
ε_1	1
ε_2	2
ε_3	0
ε_4	0
ε_5	0

$$\{\varepsilon_5, \varepsilon_4, \varepsilon_3\} > \{\varepsilon_1\} > \{\varepsilon_2\}$$

Fig. 6: Scores obtained with $Score_2$ and the associated rank.

We noticed that with the first score, the ranking obtained for the most preferred extensions is more detailed ($\varepsilon_5 > \varepsilon_4$) than the globally optimal semantics (ε_5 and ε_4 are ranked equally). However, it is less accurate for the least preferred extensions ($\varepsilon_1, \varepsilon_2$ and ε_3 have the same score). On the contrary, with the second score, the ranking obtained for the least preferred extensions is as detailed ($\varepsilon_1 > \varepsilon_2$) as the one obtained with globally optimal semantics. However, it is less accurate for the most preferred extensions ($\varepsilon_3, \varepsilon_4$ and ε_5 have the same score).

A research issue is to find a way to combine the two scores in order to produce a more efficient ranking. This can be achieved by using multi-criteria methods. We provide a naive way to combine the two scores, namely $Score_3(\varepsilon) = Score_1(\varepsilon) - Score_2(\varepsilon)$. Using this new score, we get the following results:

Extension	$Score_3$
ε_1	-1
ε_2	-2
ε_3	0
ε_4	1
ε_5	2

$$\{\varepsilon_5\} > \{\varepsilon_4\} > \{\varepsilon_3\} > \{\varepsilon_1\} > \{\varepsilon_2\}$$

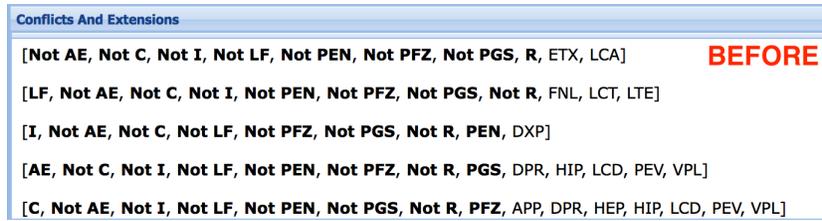
Fig. 7: Scores obtained with the combination of $Score_1$ and $Score_2$ and the associated rank.

In our example, this new score leads to a strict total order, which is arguably more useful in terms of decision-making.

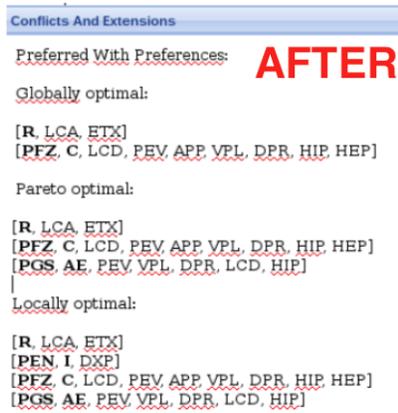
4.3 Implementation

We integrated a simple and intuitive interface in the web application for inputting preferences which enables users to clearly visualize the preferences implied and the possible incoherences (see Figure 8c). The preferences are saved in a database and are specific to a particular argumentation. We also implemented all the preferences methods discussed

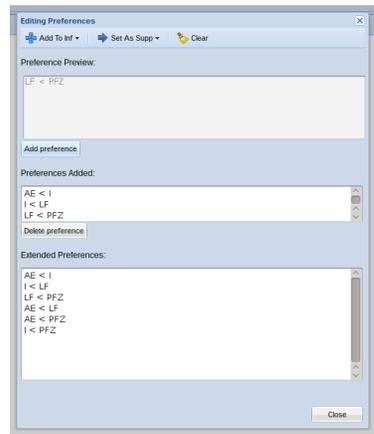
in this paper. The processing of the argumentation framework is hidden to the user and only the different extensions produced are displayed (see Figure 8a). The user can then add preferences and use the refining method introduced in Section 4.1 (see Figure 8b). Although the process has been simplified, more work is required to make it easier to understand for end users that are not expert in argumentation theory.



(a) Extensions outputted by the argumentation framework.



(b) Extensions outputted by the argumentation framework after preferences filtering.



(c) Preference interface in EcobioCap.

Fig. 8: Different interfaces of the web application.

5 Conclusion

In this paper, we described a real life use-case we obtained from several meetings with experts concerning the end-of-life of packagings. We applied an argumentation approach and showed how preferences can have a real impact on the selection of alternatives in a decision-making problem from a real agronomy inspired use case. This new approach is implemented as a web application and a demonstration of the tool can also be provided upon request.

Future work includes the investigation of a natural language processing module that will be able to semi-automatically extract arguments from text files. Another current research avenue includes the investigation of explanatory dialogues with the users that will help better understand the output of our system [5, 4, 3].

6 Acknowledgements

The authors would like to thank the partners of the Pack4Fresh project, for all their help during the argument elicitation phase as well as for their constant feedback.

References

1. L. Amgoud and H. Prade. Explaining qualitative decision under uncertainty by argumentation. In *Proceedings, The Twenty-First National Conference on Artificial Intelligence and the Eighteenth Innovative Applications of Artificial Intelligence Conference, July 16-20, 2006, Boston, Massachusetts, USA*, pages 219–224, 2006.
2. L. Amgoud and S. Vesic. Two roles of preferences in argumentation frameworks. In *Symbolic and Quantitative Approaches to Reasoning with Uncertainty - 11th European Conference, ECSQARU 2011, Belfast, UK, June 29-July 1, 2011. Proceedings*, pages 86–97, 2011.
3. A. Arioua and M. Croitoru. Formalizing Explanatory Dialogues. In *SUM'2015: 9th International Conference on Scalable Uncertainty Management*, Quebec, Canada, Sept. 2015.
4. A. Arioua and M. Croitoru. Dialectical characterization of consistent query explanation with existential rules. In *Proceedings of FLAIRS 2016*, pages 621–625, 2016.
5. A. Arioua and M. Croitoru. A dialectical proof theory for universal acceptance in coherent logic-based argumentation frameworks. In *Proceedings of ECAI 2016*, page To appear, 2016.
6. M. Croitoru, R. Thomopoulos, and S. Vesic. Introducing preference-based argumentation to inconsistent ontological knowledge bases. In *PRIMA 2015: Principles and Practice of Multi-Agent Systems - 18th International Conference, Bertinoro, Italy, October 26-30, 2015, Proceedings*, pages 594–602, 2015.
7. M. Croitoru and S. Vesic. What can argumentation do for inconsistent ontology query answering? In *Scalable Uncertainty Management - 7th International Conference, SUM 2013, Washington, DC, USA, September 16-18, 2013. Proceedings*, pages 15–29, 2013.
8. P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.
9. S. Kaci. *Working with Preferences: Less Is More*. Cognitive Technologies. Springer, 2011.
10. S. Modgil and H. Prakken. The ASPIC⁺ framework for structured argumentation: a tutorial. *Argument & Computation*, 5(1):31–62, 2014.
11. N. Tamani, P. Mosse, M. Croitoru, P. Buche, and V. Guillard. A food packaging use case for argumentation. In *Metadata and Semantics Research - 8th Research Conference, MTSR 2014, Karlsruhe, Germany, November 27-29, 2014. Proceedings*, pages 344–358, 2014.
12. N. Tamani, P. Mosse, M. Croitoru, P. Buche, V. Guillard, C. Guillaume, and N. Gontard. Eco-efficient packaging material selection for fresh produce: Industrial session. In *Graph-Based Representation and Reasoning - 21st International Conference on Conceptual Structures, ICCS 2014, Iași, Romania, July 27-30, 2014, Proceedings*, pages 305–310, 2014.
13. N. Tamani, P. Mosse, M. Croitoru, P. Buche, V. Guillard, C. Guillaume, and N. Gontard. An argumentation system for eco-efficient packaging material selection. *Computers and Electronics in Agriculture*, 113:174 – 192, 2015.