Computational Models of Argument
F. Toni et al. (Eds.)

doi:10.3233/FAIA220141

A Generalized Notion of Consistency with Applications to Formal Argumentation

Pietro BARONI [a,1], Federico CERUTTI [a] and Massimiliano GIACOMIN [a]
[a] *DII, University of Brescia, Italy*
ORCiD ID: Pietro Baroni https://orcid.org/0000-0001-5439-9561, Federico Cerutti https://orcid.org/0000-0003-0755-0358, Massimiliano Giacomin https://orcid.org/0000-0003-4771-4265

Abstract. We propose a generic notion of consistency in an abstract labelling setting, based on two relations: one of intolerance between the labelled elements and one of incompatibility between the labels assigned to them, thus allowing a spectrum of consistency requirements depending on the actual choice of these relations. As a first application to formal argumentation, we show that traditional Dung's semantics can be put in correspondence with different consistency requirements in this context. We consider then the issue of consistency preservation when a labelling is obtained as a synthesis of a set of labellings, as is the case for the traditional notion of argument justification. In this context we provide a general characterization of consistency-preserving synthesis functions and analyze the case of argument justification in this respect.

Keywords. Consistency, Argumentation semantics, Argument justification

1. Introduction

In formal argumentation, the presence of conflicts between arguments is a key aspect that calls for mechanisms able to produce sensible reasoning outcomes. In particular, it is typically required that these outcomes satisfy properties which have intuitively to do with the notion of consistency. For instance, in abstract argumentation semantics [1,2] either extensions or labellings are typically required to satisfy the property of conflict-freeness, while, moving from abstract to structured argumentation, it is desired that the conclusions of arguments regarded as acceptable are not contradictory, as indicated by the properties of direct and indirect consistency in [3]. While consistency appears to permeate the field of formal argumentation as a crucial component, to our knowledge no attempts are available in the literature to provide a general formal treatment of this notion, consistency-related definitions being usually embedded in the context of specific formalisms, without a common reference framework. This appears to be a limitation regarding the possibility of bridging together the consistency notions considered in different formalisms and possibly investigating variations and developments thereof.

To fill this gap, in this paper we introduce a generalized notion of (in)consistency applicable in any context where a labelling approach is adopted. The proposed notion re-

[1]Corresponding Author: Pietro Baroni, DII, University of Brescia, Italy; E-mail: pietro.baroni@unibs.it

lies on two basic elements: an intolerance relation between the labelled elements and an incompatibility relation between the labels, as presented in Section 2. As a first example of the application of the proposed concept, we show in Section 3 that Dung's traditional semantics can be put in correspondence with different consistency requirements, particularly with different incompatibility relations. As a further step, in Section 4, we consider the issue of consistency preservation when a labelling derives from a set of other labellings. We provide some results concerning consistency preservation when a labelling is obtained through a synthesis function and apply these concepts to the case of deriving the argument justification status. The relationships of this work with previous literature and various perspectives of future development are finally discussed in Section 5.

2. Generalizing consistency for labelling-based systems

In a variety of contexts the assessments of entities of various kind are expressed by assigning them a label taken from a predefined set. In many cases these sets of labels have an intuitive underlying order according to some notion of *positivity*. In order to provide a common ground to characterize different assessment labels and to relate and compare them, we first introduce the notion of assessment classes.

Definition 1 *A set of assessment classes is a set C equipped with a total order $\leq$ and including a maximum and a minimum element, which are assumed to be distinct.*

In the following we will abbreviate the term 'set(s) of assessment classes' as sac(s). Intuitively, the order is meant to capture an abstract distinction between different levels of positivity of the assessment, with $c_1 \leq c_2$ meaning that c_2 corresponds to an at least as positive assessment as c_1 (whatever a positive assessment means in a given context). In the following we will mostly use a tripolar sac $C^3 = \{\mathsf{pos}, \mathsf{mid}, \mathsf{neg}\}$ with $\mathsf{neg} \leq \mathsf{mid} \leq \mathsf{pos}$ and the intuitive meaning that pos corresponds to a definitely positive assessment, neg to a definitely negative assessment, and mid to an intermediate situation. The basic idea, expressed by the following definition, is that a sac is used to classify the elements of a set of labels according to their level of positivity. Note that the elements of a sac are called classes because in general more than one label can be mapped to the same class.

Definition 2 *Given a set of assessment classes C, a C-classified set of assessment labels is a set Λ equipped with a total function $C_\Lambda : \Lambda \rightarrow C$. The total preorder induced on Λ by C_Λ will be denoted by $\preceq$ where $\lambda_1 \preceq \lambda_2$ iff $C_\Lambda(\lambda_1) \leq C_\Lambda(\lambda_2)$. As usual, $\lambda_1 \prec \lambda_2$ will denote $\lambda_1 \preceq \lambda_2$ and $\lambda_2 \npreceq \lambda_1$*

We will abbreviate the term 'set(s) of assessment labels' as sal(s) and omit 'C-classified', when C is not ambiguous. Also, to distinguish preorders referring to different sals, given a sal Λ we will denote the relevant preorder as $\preceq_\Lambda$.

The notion of labelling based on a sal is the usual one.

Definition 3 *Given a sal Λ and a set S a Λ-labelling of S is a function $L : S \rightarrow \Lambda$.*

Different sals can be used to express assessments in distinct, but possibly related, evaluation contexts. For instance, in the context of argument acceptance evaluation based on the labelling-based version of Dung's semantics [1,2], the sal $\Lambda^{\mathrm{IOU}} =$

$\{\mathsf{in},\mathsf{out},\mathsf{und}\}$ is used, while in Defeasible Logic Programming (*DeLP*) arguments are marked as D(efeated) or U(ndefeated) corresponding to the use of the sal $\Lambda^{\mathrm{De}} = \{\mathsf{D},\mathsf{U}\}$, and in [4] an approach using the set of four labels $\Lambda^{\mathrm{JV}} = \{+,-,\pm,\emptyset\}$ is proposed. We assume that the sals mentioned above are C^3-classified as follows: $C^3_{\Lambda^{\mathrm{IOU}}} = \{(\mathsf{in},\mathsf{pos}),(\mathsf{out},\mathsf{neg}),(\mathsf{und},\mathsf{mid})\}$; $C^3_{\Lambda^{\mathrm{De}}} = \{(\mathsf{D},\mathsf{neg}),(\mathsf{U},\mathsf{pos})\}$; $C^3_{\Lambda^{\mathrm{JV}}} = \{(-,\mathsf{neg}),(+,\mathsf{pos}),(\pm,\mathsf{mid}),(\emptyset,\mathsf{mid})\}$.

We can now introduce a generalized notion of inconsistency in this formal context. Intuitively, an inconsistency arises when two elements of a set which cannot stand each other are assigned labels which are ‘too positive’ altogether.

This suggests that, in general terms, inconsistency can be understood as arising from two components: an intolerance relation at the level of the assessed elements, indicating who cannot stand whom, and an incompatibility relation at the level of the labels, indicating which pairs of positive assessments correspond to a clash if ascribed to a pair of elements connected by the intolerance relation. In the following we will assume that an incompatibility relation on assessment labels is always induced by an incompatibility relation on assessment classes.

Definition 4 *Given a set S, an intolerance relation on S is a binary relation int* $\subseteq S \times S$*, where* $(s_1,s_2) \in$ *int indicates that* s_1 *is intolerant of* s_2 *and will be denoted as* $s_1 \odot s_2$*, while* $(s_1,s_2) \notin$ *int will be denoted as* $s_1 \ominus s_2$*.*

Note that we do not make any assumption on the intolerance relation, in particular it needs not to be symmetric.

To exemplify, in languages equipped with negation, typically intolerance between language elements coincides with negation (a symmetric relation where each element has exactly one opposite), however more general forms of contrariness have been considered in argumentation contexts, where the corresponding intolerance relation may not be symmetric and allows the existence of multiple contraries for an element [5,6]. At the argument level, the attack relation in Dung’s frameworks can be regarded as an example of intolerance relation.

Definition 5 *Given a sac C, an incompatibility relation on C is a relation inc* $\subseteq C \times C$*, where* $(c_1,c_2) \in$ *inc indicates that* c_1 *is incompatible with* c_2 *and will be denoted as* $c_1 \boxdot c_2$*, while* $(c_1,c_2) \notin$ *inc will be denoted as* $c_1 \boxminus c_2$*. Given a C-classified sal* Λ*, we define the induced incompatibility relation inc*$'$ $\subseteq \Lambda \times \Lambda$ *as follows: for every* $\lambda_1,\lambda_2 \in \Lambda$*,* $(\lambda_1,\lambda_2) \in$ *inc*$'$ *iff* $(C_\Lambda(\lambda_1),C_\Lambda(\lambda_2)) \in$ *inc. With a little abuse of notation we will also denote* $(\lambda_1,\lambda_2) \in$ *inc*$'$ *as* $\lambda_1 \boxdot \lambda_2$*, and analogously for* $\lambda_1 \boxminus \lambda_2$*. Given a label* λ*, we define the set of labels which are compatible with* λ *as* $sc(\lambda) \triangleq \{\lambda' \in \Lambda \mid (\lambda,\lambda') \notin inc'\}$*.*

We remark again that incompatibility refers to the situation where labels are assigned to entities which are linked by intolerance. For intance, in a context where statements are assessed and intolerance between them corresponds to contradiction, two (not necessarily distinct) positive labels expressing belief should be incompatible: they cannot be assigned to two contradictory statements, since you cannot believe both of them.

We can now introduce our generalized notion of inconsistency of a labelling.

Definition 6 *Given a set S equipped with an intolerance relation int, a sac C equipped with an incompatibility relation inc, and a C-classified sal* Λ*, a* Λ*-labelling L of S is int-inc-inconsistent iff* $\exists s_1,s_2 \in S$ *such that* $s_1 \odot s_2$ *and* $L(s_1) \boxdot L(s_2)$*.*

We say that a labelling is int-inc-consistent if it is not int-inc-inconsistent and that a set $\mathscr{L}$ of labellings is int-inc-consistent if every $L \in \mathscr{L}$ is int-inc-consistent.

From the intuition underlying Definition 6, some rather natural properties can be identified for an incompatibility relation on a sac C, based on the idea that inconsistency arises from a sort of 'excess of simultaneous positiveness' in the assessment of some elements linked by intolerance. First, an obvious requirement is that $\max(C) \boxdot \max(C)$: two maximally positive labels cannot be ascribed together to conflicting elements. More generally one can observe that if $c_1 \boxdot c_2$, then for every pair c'_1, c'_2 such that $c_1 \leq c'_1$ and $c_2 \leq c'_2$ it must hold that $c'_1 \boxdot c'_2$, since the simultaneous positiveness expressed by c'_1 and c'_2 is not lesser that the one expressed by c_1 and c_2. We call such an incompatibility relation *monotonic* and take this property for granted in the following.

Note that $\max(C) \boxdot \max(C)$ is a consequence of the monotonicity property if one assumes that inc is not empty. Accordingly we define, for any sac C, the minimal nonempty incompatibility relation as $\underline{\text{inc}}_C = \{(\max(C), \max(C))\}$.

It also follows that, to avoid a degenerate situation where every labelling is inconsistent, it must hold that $\min(C) \boxminus \min(C)$.

Moreover, assuming that the intolerance relation is not empty, for $\max(C)$ to be attainable for every element without necessarily generating inconsistencies it must be the case that the following stronger condition (implying the previous one) holds: $\max(C) \boxminus \min(C)$ and $\min(C) \boxminus \max(C)$ or equivalently $\nexists c \in C$ such that $c \boxdot \min(C)$ or $\min(C) \boxdot c$. Note that this implies that for any C-classified sal Λ, $sc(\lambda) \neq \emptyset$ for any $\lambda \in \Lambda$ under the mild condition that $\exists \lambda \in \Lambda : C_\Lambda(\lambda) = \min(C)$, which we will assume in the following.

The generic definition of inconsistency we have introduced is 'tunable' as its instances can be 'adjusted' varying the incompatibility relation, and possibly also the underlying intolerance relation and C-classification, giving rise to a family of alternative (in)consistency notions. In particular, different argumentation semantics can be put in correspondence with different (in)consistency notions, as discussed next.

3. Consistency properties in argumentation semantics

As well-known, in abstract argumentation an argumentation semantics is a formal specification of a criterion to determine the possible outcomes of a situation of conflict, represented by a binary relation of attack (denoted as $\rightarrow$ in the following), between a set $\mathscr{A}$ of arguments, as expressed by the traditional notion of argumentation framework [1].

Definition 7 *An argumentation framework is a pair $AF = (\mathscr{A}, \rightarrow)$ where $\mathscr{A}$ is a set of arguments and $\rightarrow \subseteq \mathscr{A} \times \mathscr{A}$ is a binary relation of attack between them.*

In the *extension-based* approach to argumentation semantics the conflict outcomes are expressed as sets of arguments called *extensions* and, in this context, a basic consistency notion called *conflict-freeness* has been traditionally considered: a set of arguments is conflict-free if it does not include any pair of arguments α, β such that $(\alpha, \beta) \in \rightarrow$ (also denoted as $\alpha \in \beta^-$). In the *labelling-based* approach to argumentation semantics, the outcomes are expressed as arguments labellings, i.e. as assignments of labels, taken from a given set, to the set of arguments $\mathscr{A}$. Using the set of three labels Λ^{IOU} a correspondence can be drawn between extensions and labellings, while in general the labelling-based approach is more expressive than the extension-based approach. Combining the generalized notion of consistency with three-valued labellings enables to identify correspondences between different notions of consistency and different semantics. In particular, given an abstract argumentation framework, we naturally assume that the intolerance relation coincides with the attack relation, i.e. $\alpha \odot \beta$ iff $\alpha \in \beta^-$, and use the classification $C^3_{\Lambda\text{IOU}}$ introduced above. Then, an analysis of labelling-based semantics in this perspective can be developed, as we do in the following, where we review the definitions of some fundamental labelling-based semantics [2] and analyze their generalized consistency properties.

The simplest semantics notion is the one of conflict-freeness, which is recalled in Definition 8.

Definition 8 *Let L be a labelling of an argumentation framework $AF = (\mathscr{A}, \rightarrow)$. L is* conflict-free *iff for each $\alpha \in \mathscr{A}$ it holds that:*

1. *if* $L(\alpha) = \texttt{in}$ *then* $\nexists \beta \in \alpha^- : L(\beta) = \texttt{in}$
2. *if* $L(\alpha) = \texttt{out}$ *then* $\exists \beta \in \alpha^- : L(\beta) = \texttt{in}$

Item 1 in Definition 8 corresponds exactly to the weakest form of consistency, i.e. to the incompatibility relation $\underline{\text{inc}}_{C^3} = \{(\mathsf{pos}, \mathsf{pos})\}$.

Admissibility of a set of arguments was introduced in [1] with reference to the notion of defense, i.e. the ability of a conflict-free set to defend its members by counterattacking their attackers. The labelling-based counterpart of this idea is given in Definition 9.

Definition 9 *Let L be a labelling of an argumentation framework $AF = (\mathscr{A}, \rightarrow)$. L is* admissible *iff for each $\alpha \in \mathscr{A}$ it holds that:*

1. *if* $L(\alpha) = \texttt{in}$ *then* $\forall \beta \in \alpha^- : L(\beta) = \texttt{out}$
2. *if* $L(\alpha) = \texttt{out}$ *then* $\exists \beta \in \alpha^- : L(\beta) = \texttt{in}$

Item 1 in Definition 9 is a strengthening of item 1 of Definition 8, while item 2 is the same in both Definition 8 and 9. Interestingly, this strengthening corresponds to the choice of a stronger form of consistency: having an attacker labelled und is forbidden for an argument labelled in, while having an attacker labelled in is allowed for an argument labelled und. This coincides with adopting the following asymmetric incompatibility relation $\text{inc}^a_{C^3} = \{(\mathsf{pos}, \mathsf{pos}), (\mathsf{mid}, \mathsf{pos})\}$.

Proposition 1 *The set of admissible labellings coincides with the set of conflict-free labellings which are $\rightarrow$-$\text{inc}^a_{C^3}$-consistent.*

Proof: For a labelling L let us first assume that L is admissible. Then L is conflict-free and by item 1 of Definition 9 $\nexists \alpha, \beta \in \mathscr{A}$ such that $\beta \in \alpha^-$ (i.e. $\beta \odot \alpha$) and $(L(\beta), L(\alpha)) \in \text{inc}^a_{C^3}$ (i.e. $L(\beta) \boxdot L(\alpha)$). Hence L is $\rightarrow$-$\text{inc}^a_{C^3}$-consistent. Let now assume L is conflict-free and $\rightarrow$-$\text{inc}^a_{C^3}$-consistent. To complete the proof we have to show that item 1 of Definition 9 holds: assume by contradiction that $\exists \alpha$ such that $L(\alpha) = \texttt{in}$ and $\exists \beta \in \alpha^- : L(\beta) \neq \texttt{out}$. It follows that $(L(\beta), L(\alpha)) \in \text{inc}^a_{C^3}$ which contradicts the hypotesis that L is $\rightarrow$-$\text{inc}^a_{C^3}$-consistent. □

Completeness of a set of arguments was introduced in [1] and is based on the idea that if an argument is defended by an admissible set of arguments, it should be accepted together with its defenders. The labelling-based counterpart of this idea is given in Definition 10.

Definition 10 *Let L be a labelling of an argumentation framework $AF = (\mathscr{A}, \rightarrow)$. L is* complete *if it is admissible and for each $\alpha \in \mathscr{A}$ it holds that if $L(\alpha) = \mathtt{und}$ then $\nexists \beta \in \alpha^- : L(\beta) = \mathtt{in}$ and $\exists \beta \in \alpha^- : L(\beta) = \mathtt{und}$*

In words a complete labelling is an admissible labelling with the additional requirement that an argument which is labelled und must have an und-labelled attacker and no in-labelled attackers. This amounts to further strengthening the notion of consistency by adopting the incompatibility relation $\mathrm{inc}^c_{C^3} = \{(\mathsf{pos},\mathsf{pos}),(\mathsf{pos},\mathsf{mid}),(\mathsf{mid},\mathsf{pos})\}$ together with enforcing the following reinstatement property.

Definition 11 *A labelling L satisfies the reinstatement property if $\forall \alpha \in \mathscr{A}$ it holds that if $\forall \beta \in \alpha^-$ $L(\beta) = \mathtt{out}$ then $L(\beta) = \mathtt{in}$*

Proposition 2 *The set of complete labellings coincides with the set of admissible labellings which are $\rightarrow$-$\mathrm{inc}^c_{C^3}$-consistent and satisfy the reinstatement property.*

Proof: For a labelling L let us first assume that L is complete, hence admissible. From Proposition 1 we have that $\nexists \alpha, \beta$ such that $\beta \in \alpha^-$ and $(L(\beta), L(\alpha)) \in \{(\mathsf{pos},\mathsf{pos}),(\mathsf{mid},\mathsf{pos})\}$. From Definition 10 we have also that if $L(\alpha) = \mathtt{und}$ then $\nexists \beta \in \alpha^- : L(\beta) = \mathtt{in}$, i.e. $\nexists \alpha, \beta$ such that $\beta \in \alpha^-$ and $(L(\beta), L(\alpha)) \in \{(\mathsf{pos},\mathsf{mid})\}$. It follows that L is $\rightarrow$-$\mathrm{inc}^c_{C^3}$-consistent. Moreover, it is well known that complete labellings satisfy the reinstatement property [2]. Let us now assume L is admissible, $\rightarrow$-$\mathrm{inc}^c_{C^3}$-consistent and satisfies the reinstatement property. Given an argument α such that $L(\alpha) = \mathtt{und}$ it follows (from consistency) that $\nexists \beta \in \alpha^- : L(\beta) = \mathtt{in}$ and (from the reinstatement property) that $\exists \beta \in \alpha^- : L(\beta) \neq \mathtt{out}$, hence $\exists \beta \in \alpha^- : L(\beta) = \mathtt{und}$ and L is a complete labelling. □

Stability of a set of arguments can be characterized in several ways, its key feature being that no room is left for undecidedness (an argument is either accepted or attacked by an accepted argument) as indicated by Definition 12.

Definition 12 *Let L be a labelling of an argumentation framework $AF = (\mathscr{A}, \rightarrow)$. L is* stable *if it is complete and $\nexists \alpha \in \mathscr{A} : L(\alpha) = \mathtt{und}$.*

This constraint can be put in correspondence with the adoption of the strongest notion of consistency, namely with the choice of the incompatibility relation $\overline{\mathrm{inc}}_{C^3} = \{(\mathsf{pos},\mathsf{pos}),(\mathsf{pos},\mathsf{mid}),(\mathsf{mid},\mathsf{pos}),(\mathsf{mid},\mathsf{mid})\}$.

Proposition 3 *The set of stable labellings coincides with the set of complete labellings which are $\rightarrow$-$\overline{\mathrm{inc}}_{C^3}$-consistent.*

Proof: For a labelling L let us first assume that L is stable. It follows that no argument is labelled und hence $\nexists \alpha, \beta$ such that $\beta \in \alpha^-$ and $(L(\beta), L(\alpha)) \in \{(\mathsf{pos},\mathsf{mid}),(\mathsf{mid},\mathsf{pos}),(\mathsf{mid},\mathsf{mid})\}$ and from conflict-freeness we have also that $\nexists \alpha, \beta$ such that $\beta \in \alpha^-$ and $(L(\beta), L(\alpha)) = (\mathsf{pos},\mathsf{pos})$. Therefore L is $\rightarrow$-$\overline{\mathrm{inc}}_{C^3}$-consistent. Assume now L is complete and $\rightarrow$-$\overline{\mathrm{inc}}_{C^3}$-consistent and suppose by contradiction that $\exists \alpha$ such that $L(\alpha) = \mathtt{und}$. It follows that $\alpha^- \neq \emptyset$, otherwise by the reinstatement property it would hold that $L(\alpha) = \mathtt{in}$. For every $\beta \in \alpha^-$ we have that $L(\beta) \notin \{\mathtt{in}, \mathtt{und}\}$ otherwise L would not be

$\rightarrow$-$\overline{\text{inc}}_{C^3}$-consistent. But then $\forall\beta \in \alpha^-$ we get $L(\beta) = \texttt{out}$ which, by the reinstatement property, contradicts $L(\alpha) = \texttt{und}$. □

To summarize, admissible labellings can be characterized in terms of strengthening consistency with respect to conflict-freeness without resorting to the traditional notion of defense, while further strengthenings of consistency, together with the reinstatement property, characterize complete and stable labellings.

In the next section we move beyond the evaluation of acceptability of arguments carried out on the basis of argumentation semantics and consider further evaluations that can be derived from it and raise the issue of preserving consistency across the derivation.

4. Consistency preservation in labelling derivation mechanisms

The outcomes prescribed by an argumentation semantics are typically used as the starting point for the derivation of further evaluations, for instance whether an argument is skeptically justified. It is then interesting to consider the question of whether and how the consistency properties of the original evaluation are preserved in the derived evaluation and of the requirements that can be posed on the derivation mechanism to ensure this preservation.

We focus here on what we call pure synthesis labelling derivation, namely a mechanism where a labelling of a set S is generated from a set of labellings of the same set S. To exemplify, the evaluation of the argument justification status according to a given semantics is derived from the set of the argument extensions/labellings prescribed by the same semantics.

The simplest notion of argument justification, which we will use as running example, is based on three possible states.

Definition 13 *Given a set $\mathscr{L}$ of Λ^{IOU}-labellings of a set of arguments $\mathscr{A}$, an argument $\alpha \in \mathscr{A}$ is:*

- *skeptically justified iff $\forall L \in \mathscr{L}$ $L(\alpha) = \texttt{in}$;*
- *credulously justified iff it is not skeptically justified*[2] *and $\exists L \in \mathscr{L} : L(\alpha) = \texttt{in}$;*
- *not justified iff $\nexists L \in \mathscr{L} : L(\alpha) = \texttt{in}$*

Considering a sal $\Lambda^{\mathrm{AJ}} = \{\mathsf{SkJ}, \mathsf{CrJ}, \mathsf{NoJ}\}$, the evaluation of argument justification can be modelled as the generation of a Λ^{AJ}-labelling from a set of Λ^{IOU}-labellings. Concerning Λ^{AJ} it is intuitive to assume the classification $C^3_{\Lambda^{\mathrm{AJ}}} = \{(\mathsf{SkJ}, \mathsf{pos}), (\mathsf{NoJ}, \mathsf{neg}), (\mathsf{CrJ}, \mathsf{mid})\}$.

At a general level, pure synthesis labelling derivations, like the one of argument justification, can be formalized through a simple synthesis function.

Definition 14 *Given two sets of labels Λ_1 and Λ_2, a simple synthesis function (ssf) from Λ_1 to Λ_2 is a mapping* $\mathsf{syn} : 2^{\Lambda_1} \setminus \{\emptyset\} \rightarrow \Lambda_2$.

The idea is that given a set of Λ_1-labellings of a set S a Λ_2-labelling of S can be derived by applying a ssf to the set of labels relevant to each element of S.

[2] Traditionally credulous justification id regarded as including skeptical justification, we enforce this distinction so that argument justification can be properly modelled as a labelling.

Definition 15 *Let S be a set,* Λ_1 *and* Λ_2 *sets of labels,* syn *a ssf from* Λ_1 *to* Λ_2*, and* $\mathscr{L}_1$ *a set of* Λ_1*-labellings of S. The* Λ_2*-labelling* L_2 *derived from* $\mathscr{L}_1$ *through* syn *is denoted as* $DL^{\mathsf{syn}}_{\mathscr{L}_1}$ *defined, for every* $s \in S$ *as:*

$$DL^{\mathsf{syn}}_{\mathscr{L}_1}(s) = \mathsf{syn}(\{L_1(s) \mid L_1 \in \mathscr{L}_1\})$$

To exemplify, it is easy to see that the argument justification evaluation described above corresponds to the use of a ssf $\mathsf{syn}_{\mathrm{AJ}}$ from Λ^{IOU} to Λ^{AJ} defined, for every $\Lambda \subseteq \Lambda^{\mathrm{IOU}}$ as follows:

- $\mathsf{syn}_{\mathrm{AJ}}(\Lambda) = \mathsf{SkJ}$ if $\Lambda = \{\mathtt{in}\}$;
- $\mathsf{syn}_{\mathrm{AJ}}(\Lambda) = \mathsf{CrJ}$ if $\Lambda \supsetneq \{\mathtt{in}\}$;
- $\mathsf{syn}_{\mathrm{AJ}}(\Lambda) = \mathsf{NoJ}$ otherwise.

Assuming that the labellings used for derivation satisfy some consistency properties, a preservation of these properties in the derived labelling appears to be desirable.

Definition 16 *Let C be a sac equipped with an incompatibility relation inc, and* Λ_1 *and* Λ_2 *be two C-classifed sets of labels. A ssf* syn *from* Λ_1 *to* Λ_2 *is* consistency preserving *iff for any set S equipped with an intolerance relation int and any int-inc-consistent set* $\mathscr{L}_1$ *of* Λ_1*-labellings of S it holds that the labelling* $DL^{\mathsf{syn}}_{\mathscr{L}_1}$ *is int-inc-consistent.*

This in turn raises the issue of analyzing at a general level some properties of the ssf that can ensure consistency preservation.

To start, we introduce a notion of well-behaved ssf which intuitively means that the function is monotonic with respect to some positiveness ordering of sets of labels, introduced in next definition.

Definition 17 *Given a sal* Λ*, and* $\Lambda_1, \Lambda_2 \subseteq \Lambda$*, we say that* Λ_2 *is at least as positive as* Λ_1*, denoted as* $\Lambda_1 \preceq_P \Lambda_2$*, iff* $\forall \lambda \in \Lambda_1\ \exists \lambda' \in \Lambda_2$ *such that* $\lambda \preceq_\Lambda \lambda'$ *and* $\forall \lambda' \in \Lambda_2\ \exists \lambda \in \Lambda_1$ *such that* $\lambda \preceq_\Lambda \lambda'$*.*

The idea of the $\preceq_P$ relation is that every element of Λ_1 can be mapped into an at least as positive element of Λ_2 and at the same time every element of Λ_2 can be mapped into a no more positive element of Λ_1. $\preceq_P$ is reflexive and transitive, i.e. a preorder. To exemplify, $\forall \emptyset \subsetneq \Lambda \subseteq \Lambda^{\mathrm{IOU}}$ it holds that $\Lambda \preceq_P \{\mathtt{in}\}$ and $\{\mathtt{out}\} \preceq_P \Lambda$. Also $\{\mathtt{in}, \mathtt{out}\} \preceq_P \{\mathtt{in}, \mathtt{und}, \mathtt{out}\}$ and $\{\mathtt{in}, \mathtt{und}, \mathtt{out}\} \preceq_P \{\mathtt{in}, \mathtt{out}\}$ while $\{\mathtt{in}, \mathtt{out}\} \not\preceq_P \{\mathtt{und}\}$ and $\{\mathtt{und}\} \not\preceq_P \{\mathtt{in}, \mathtt{out}\}$,

We can now introduce the notion of well-behaved ssf.

Definition 18 *A ssf* syn *is well-behaved iff whenever* $\Lambda_1 \preceq_P \Lambda_2$ $\mathsf{syn}(\Lambda_1) \preceq \mathsf{syn}(\Lambda_2)$.

We then move to consider, given a set of labels Λ_1, whether a set of labels Λ_2 is a compatible dual of Λ_1, meaning that, given an int-inc-consistent set of labellings $\mathscr{L}_1$, if $\Lambda_1 = \{L_1(s) \mid L_1 \in \mathscr{L}_1\}$ for some element s, then it is possible that $\Lambda_2 = \{L_1(s') \mid L_1 \in \mathscr{L}_1\}$ for some s' such that $s \odot s'$.

Definition 19 *Given a sal* Λ*, and* $\Lambda_1 \subseteq \Lambda$*, we say that* $\Lambda_2 \subseteq \Lambda$ *is a compatible dual of* Λ_1*, denoted as* $\Lambda_2 \in CD(\Lambda_1)$*, iff* $\forall \lambda \in \Lambda_1\ \exists \lambda' \in \Lambda_2$ *such that* $\lambda' \in sc(\lambda)$*, and* $\forall \lambda' \in \Lambda_2$ $\exists \lambda \in \Lambda_1$ *such that* $\lambda' \in sc(\lambda)$*.*

Proposition 4 *Let C be a sac equipped with an incompatibility relation inc and Λ be a C-classifed set of labels. For any set S equipped with an intolerance relation int, any int-inc-consistent set $\mathscr{L}_1$ of Λ-labellings of S, and any $s_1, s_2 \in S$ such that $(s_1, s_2) \in int$, it holds that $\mathscr{L}_1^{\downarrow}(s_2) \in CD(\mathscr{L}_1^{\downarrow}(s_1))$ where for any set $\mathscr{L}$ of Λ-labellings of S and any $s \in S$, $\mathscr{L}^{\downarrow}(s) \triangleq \{L(s) \mid L \in \mathscr{L}\}$.*

Proof: Since every $L \in \mathscr{L}_1$ is int-inc-consistent it must be the case that $L(s_2) \in sc(L(s_1))$, hence $\forall \lambda \in \mathscr{L}_1^{\downarrow}(s_1)\ \exists \lambda' \in \mathscr{L}_1^{\downarrow}(s_2)$ such that $\lambda' \in sc(\lambda)$ and also $\forall \lambda' \in \mathscr{L}_1^{\downarrow}(s_2)\ \exists \lambda \in \mathscr{L}_1^{\downarrow}(s_1)$ such that $\lambda' \in sc(\lambda)$, hence $\mathscr{L}_1^{\downarrow}(s_2) \in CD(\mathscr{L}_1^{\downarrow}(s_1))$. □

Towards characterizing well-behaved ssfs which are consistency preserving we focus on the case where the set of labellings to be synthesized is finite, which is common in argumentation semantics and many other kinds of assessments. The case of infinite sets of labellings is left to future work.

To start, we need to consider a compatible dual of a finite set of labels which turns out to be not less positive than any other compatible dual.

Definition 20 *Given a sal Λ, and a finite $\Lambda_1 \subseteq \Lambda$ we define $MCD(\Lambda_1) \triangleq \bigcup_{\lambda \in \Lambda_1} \widehat{MC}(\lambda)$, where $\widehat{MC}(\lambda) \triangleq \{\lambda' \in sc(\lambda) \mid \nexists \lambda'' \in sc(\lambda) : \lambda' \prec \lambda''\}$*

Note that non emptyness of $MCD(\Lambda_1)$ follows from the non emptyness of $sc(\lambda)$ for every λ (Section 2) and from the finiteness of Λ_1 together with the total ordering of C. The following propositions, which assume again Λ_1 finite, provide two interesting properties of $MCD(\Lambda_1)$: it belongs to $CD(\Lambda_1)$ and is maximal with respect to $\preceq_P$.

Proposition 5 $MCD(\Lambda_1) \in CD(\Lambda_1)$.

Proof: From the definition it is immediate to see that for every $\lambda \in \Lambda_1\ \exists \lambda' \in MCD(\Lambda_1)$ such that $\lambda' \in sc(\lambda)$ and that for every $\lambda' \in MCD(\Lambda_1)\ \exists \lambda \in \Lambda_1$ such that $\lambda' \in sc(\lambda)$. □

Proposition 6 $\forall D \in CD(\Lambda_1)$ *it holds that* $D \preceq_P MCD(\Lambda_1)$.

Proof: For any $\lambda' \in D$, from Definition 19 it holds that $\exists \lambda \in \Lambda_1$ such that $\lambda' \in sc(\lambda)$. Then, by Definition 20 $\exists \lambda'' \in MCD(\Lambda_1)$ such that $\lambda'' \in sc(\lambda)$ and $\nexists \lambda''' \in sc(\lambda) : \lambda'' \prec \lambda'''$ which implies that $\lambda' \preceq \lambda''$. Consider now any $\lambda'' \in MCD(\Lambda_1)$: by Definition 20 it holds that $\exists \lambda \in \Lambda_1$ such that $\lambda'' \in sc(\lambda)$. Moreover by Definition 19 $\exists \lambda' \in D$ such that $\lambda' \in sc(\lambda)$. Now by Definition 20 we have again that $\nexists \lambda''' \in sc(\lambda) : \lambda'' \prec \lambda'''$ and hence $\lambda' \preceq \lambda''$. □

On this basis, we can now derive a necessary and sufficient condition for a well-behaved ssf to be consistency preserving for finite sets of labels.

Proposition 7 *A well-behaved ssf* syn *is consistency preserving if and only if for every finite set $\Lambda_1 \subseteq \Lambda$ it holds that* $\mathsf{syn}(MCD(\Lambda_1)) \in sc(\mathsf{syn}(\Lambda_1))$.

Proof: Let syn be a ssf satisfying the hypotheses and assume by contradiction that syn is not consistency preserving. This means that there are two elements $s_1, s_2 \in S$ such that $s_1 \odot s_2$ and a set $\mathscr{L}_1$ of Λ-labellings of S such that $DL_{\mathscr{L}_1}^{\mathsf{syn}}(s_1) \boxdot DL_{\mathscr{L}_1}^{\mathsf{syn}}(s_2)$.

Now $DL^{\mathsf{syn}}_{\mathscr{L}_1}(s_1) = \mathsf{syn}(\mathscr{L}_1^{\downarrow}(s_1))$ and similarly $DL^{\mathsf{syn}}_{\mathscr{L}_1}(s_2) = \mathsf{syn}(\mathscr{L}_1^{\downarrow}(s_2))$. Let $\Lambda_1 = \mathsf{syn}(\mathscr{L}_1^{\downarrow}(s_1))$. From Proposition 4 we have $\mathscr{L}_1^{\downarrow}(s_2) \in CD(\mathscr{L}_1)$ and hence from Proposition 6 $\mathscr{L}_1^{\downarrow}(s_2) \preceq_P MCD(\Lambda_1)$. Since syn is well-behaved $\mathsf{syn}(\mathscr{L}_1^{\downarrow}(s_2) \preceq \mathsf{syn}(MCD(\Lambda_1))$, but this, together with $\mathsf{syn}(MCD(\Lambda_1)) \in sc(\mathsf{syn}(\Lambda_1))$, contradicts $\mathsf{syn}(\mathscr{L}_1^{\downarrow}(s_1)) \boxdot \mathsf{syn}(\mathscr{L}_1^{\downarrow}(s_2))$. As to the other direction of the proof, assume now that syn is consistency preserving. Since by Proposition 5 for every set $\Lambda_1 \subseteq \Lambda$ it holds that $MCD(\Lambda_1) \in CD(\Lambda_1)$, we can identify a consistent set $\mathscr{L}_1$ of Λ-labellings such that $\mathscr{L}_1^{\downarrow}(s_1) = \Lambda_1$ and $\mathscr{L}_1^{\downarrow}(s_2) = MCD(\Lambda_1)$ with $s_1 \odot s_2$. Then by consistency preservation it must also hold that $\mathsf{syn}(MCD(\Lambda_1)) \in sc(\mathsf{syn}(\Lambda_1))$. □

As an example of application of the above concepts, we show that the function $\mathsf{syn}_{\mathrm{AJ}}$ is consistency preserving for the incompatibility relations $\underline{\mathrm{inc}}_{C^3}$, $\mathrm{inc}^a_{C^3}$, and $\mathrm{inc}^c_{C^3}$ while it is not for $\overline{\mathrm{inc}}_{C^3}$.

First we need to show that $\mathsf{syn}_{\mathrm{AJ}}$ is well-behaved.

Proposition 8 *The ssf* syn_{AJ} *is well-behaved.*

Proof: Since the strict order $\prec$ induced on Λ^{AJ} is total, it is sufficient to show that for any two non-empty sets $\Lambda_1, \Lambda_2 \subseteq \Lambda^{\mathrm{IOU}}$ whenever $\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1) \prec \mathsf{syn}_{\mathrm{AJ}}(\Lambda_2)$ it does not hold that $\Lambda_2 \preceq_P \Lambda_1$. First, it is easy to see that $\{\mathtt{in}\} \not\preceq_P \Lambda_1$ for any $\Lambda_1 \subseteq \Lambda^{\mathrm{IOU}}$, with $\Lambda_1 \notin \{\emptyset, \{\mathtt{in}\}\}$ which covers all cases where $\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1) \prec \mathsf{SkJ}$. Then, it is also easy to see that for any non-empty set Λ_1 such that $\mathtt{in} \notin \Lambda_1$ and any set Λ_2 such that $\{\mathtt{in}\} \subsetneq \Lambda_2$, $\Lambda_2 \not\preceq_P \Lambda_1$, covering all cases where $\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1) \prec \mathsf{CrJ}$ and thus completing the proof. □

Proposition 9 *The ssf* syn_{AJ} *is consistency preserving for the incompatibility relations* $\underline{inc}_{C^3}$, $inc^a_{C^3}$, *and* $inc^c_{C^3}$ *while it is not for* $\overline{inc}_{C^3}$.

Proof: We need to show that for every non-empty set $\Lambda_1 \subseteq \Lambda^{\mathrm{IOU}}$ it holds that $\mathsf{syn}_{\mathrm{AJ}}(MCD(\Lambda_1)) \in sc(\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1))$. For $\underline{\mathrm{inc}}_{C^3}$, $\mathrm{inc}^a_{C^3}$, and $\mathrm{inc}^c_{C^3}$ this is illustrated in Table 1, where the first column presents the various possible cases for Λ_1 with the relevant value of $\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1)$ and the following columns (illustrating $\underline{\mathrm{inc}}_{C^3}$, $\mathrm{inc}^a_{C^3}$, and $\mathrm{inc}^c_{C^3}$ respectively) show the corresponding $MCD(\Lambda_1)$ and the relevant value of $\mathsf{syn}_{\mathrm{AJ}}(MCD(\Lambda_1))$. By inspection, it can be checked that, as desired, for every pair $(\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1), \mathsf{syn}_{\mathrm{AJ}}(MCD(\Lambda_1)))$ obtained by taking the first element from a row of the first column, and the second element from any other cell (say the i-th with $i \in \{2,3,4\}$) of the same row it holds that $(\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1), \mathsf{syn}_{\mathrm{AJ}}(MCD(\Lambda_1))) \notin \mathrm{inc}'$ where inc' is the incompatibility relation induced by the inc relation specified at the top of the column from which the second element of the pair was taken. For instance, considering the fifth row, with $\Lambda_1 = \{\mathtt{in}, \mathtt{out}\}$ and $(\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1)) = \mathsf{CrJ}$ and its second cell where (according to $\underline{\mathrm{inc}}_{C^3}$) $MCD(\Lambda_1) = \{\mathtt{in}, \mathtt{und}\}$ we have $(\mathsf{syn}_{\mathrm{AJ}}(MCD(\Lambda_1))) = \mathsf{CrJ}$ and then $(\mathsf{CrJ}, \mathsf{CrJ}) \notin \mathrm{inc}'$ since $(\mathsf{mid}, \mathsf{mid}) \notin \underline{\mathrm{inc}}_{C^3}$.

Concerning $\overline{\mathrm{inc}}_{C^3}$ a counterexample is provided by $\Lambda_1 = \{\mathtt{in}, \mathtt{out}\}$ with $MCD(\Lambda_1) = \{\mathtt{in}, \mathtt{out}\}$ and $\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1) = \mathsf{syn}_{\mathrm{AJ}}(MCD(\Lambda_1)) = \mathsf{CrJ}$ while $(\mathsf{mid}, \mathsf{mid}) \in \overline{\mathrm{inc}}_{C^3}$. □

The fact that $\mathsf{syn}_{\mathrm{AJ}}$ is not consistency preserving according to $\overline{\mathrm{inc}}_{C^3}$ is not surprising, given that $\overline{\mathrm{inc}}_{C^3}$ essentially reflects the fully bipolar nature of stable semantics, while $\mathsf{syn}_{\mathrm{AJ}}$ admits tripolar assessments.

Λ_1 $\mathsf{syn}_{\mathrm{AJ}}(\Lambda_1)$	$\underline{\mathrm{inc}}_{C3}$	inc^{a}_{C3}	inc^{c}_{C3}
{in} SkJ	{und} NoJ	{und} NoJ	{out} NoJ
{out} NoJ	{in} SkJ	{in} SkJ	{in} SkJ
{und} NoJ	{in} SkJ	{und} NoJ	{und} NoJ
{in, out} CrJ	{in, und} CrJ	{in, und} CrJ	{in, out} CrJ
{in, und} CrJ	{in, und} CrJ	{und} NoJ	{und, out} NoJ
{und, out} NoJ	{in} SkJ	{in, und} CrJ	{in, und} CrJ
{in, und, out} CrJ	{in, und} CrJ	{in, und} CrJ	{in, und, out} CrJ

Table 1. Illustration of the proof of Proposition 9.

5. Discussion and conclusion

We have introduced a generalized notion of consistency and provided two initial examples of its possible uses in formal argumentation: revisiting some of Dung's traditional semantics from a perspective of progressive strengthening of consistency requirements and characterizing the consistency preservation of operators which produce assessments as a synthesis of sets of labellings, as is the case for the traditional notion of argument justification.

To our knowledge, providing a generalized form of the notion of consistency has not been previously considered in the formal argumentation literature, while other related and complementary research directions have been pursued. For instance, in [7] the idea of encompassing some inconsistency tolerance, through an *inconsistency budget* in the semantics of weighted argumentation systems is considered. This proposal does not address the issues we consider for traditional argumentation frameworks, while extending our approach to the case of weighted systems appears to be an important direction for future work. In [8] the notion of conflict-tolerant semantics is introduced, which is essentially based on lifting the requirement of conflict-freeness in semantics definition. In the context of our approach, this corresponds to making the intolerance relation empty, while keeping other constraints: again, we consider drawing correspondences between our approach and this proposal as interesting future work. In [9] the problem of measuring inconsistency in (abstract and structured) argumentation formalisms is addressed: this is an orthogonal research direction as we do not aim at quantifying inconsistency in a given setting, but rather at encompassing different notions of inconsistency. Bridging the two directions appears worth investigating.

Extending the analysis beyond tripolar classifications is another important future development. For example, more articulated notions of argument justification have been considered in the literature [10,11,12,13]. Dealing with consistency and its preservation in such a context might require considering different sets of assessment classes and defining a notion of refinement between them.

Addressing the evaluation of argument conclusions and their consistency is a further key step. In particular, it would be interesting to extend the notions presented in this paper to the formalism of multi-labelling systems [14], which can capture a variety of approaches to derive the assessment of conclusions from the assessment of arguments. This will require to tackle several additional aspects, like addressing the connections between intolerance relations involving entities at different levels and dealing with the various possible mechanism for synthesizing the labellings of conclusions after projecting argument labellings on them.

Finally, we suggest that, in the long term, the potential uses of the proposed approach go beyond the formal argumentation field. Consistency is a crucial aspect of most, if not all, reasoning formalisms, typically defined using their structural elements. Exposing the elementary concepts composing the notion of consistency brings, among others, the following advantages. Firstly, it may enable inter-formalism analyses, comparisons, and cross-fertilization. Further, it may provide a basis for developing novel theoretical and practical tools, like, for instance, methods to preserve consistency across different reasoning stages or general-purpose parametric consistency checkers.

References

[1] Dung PM. On the Acceptability of Arguments and Its Fundamental Role in Nonmonotonic Reasoning, Logic Programming, and n-Person Games. Artif Intell. 1995;77(2):321-57.

[2] Baroni P, Caminada M, Giacomin M. An introduction to argumentation semantics. Knowledge Engineering Review. 2011;26(4):365-410.

[3] Caminada M, Amgoud L. On the evaluation of argumentation formalisms. Artif Intell. 2007;171:286 310.

[4] Jakobovits H, Vermeir D. Robust Semantics for Argumentation Frameworks. J of Logic and Computation. 1999;9(2):215-61.

[5] Modgil S, Prakken H. A general account of argumentation with preferences. Artif Intell. 2013;195:361 397.

[6] Baroni P, Giacomin M, Liao B. Dealing with Generic Contrariness in Structured Argumentation. In: Proc. of the 24th Int. Joint Conf. on Artificial Intelligence, IJCAI 2015; 2015. p. 2727-33.

[7] Dunne PE, Hunter A, McBurney P, Parsons S, Wooldridge MJ. Weighted argument systems: Basic definitions, algorithms, and complexity results. Artif Intell. 2011;175(2):457-86.

[8] Arieli O. Conflict-free and conflict-tolerant semantics for constrained argumentation frameworks. J Appl Log. 2015;13(4):582-604.

[9] Hunter A. Measuring Inconsistency in Argument Graphs. CoRR. 2017;abs/1708.02851. Available from: http://arxiv.org/abs/1708.02851.

[10] Caminada MWA, Wu Y. On the Justification Status of Arguments. In: Proc. of the 22nd Benelux Conference on Artificial Intelligence; 2010. .

[11] Wu Y, Caminada MWA. A Labelling-Based Justification Status of Arguments. Studies in Logic. 2010;3(4):12-29.

[12] Dvořák W. On the Complexity of Computing the Justification Status of an Argument. In: Modgil S, Oren N, Toni F, editors. Proc. of the 1st Int. Workshop on Theory and Applications of Formal Argumentation (TAFA 2011). No. 7132 in Lecture Notes in Computer Science. Springer; 2012. p. 32-49.

[13] Baroni P, Giacomin M, Guida G. Towards a Formalization of Skepticism in Extension-based Argumentation Semantics. In: Proc. of the 4th Workshop on Computational Models of Natural Argument (CMNA 2004); 2004. p. 47-52.

[14] Baroni P, Riveret R. Enhancing Statement Evaluation in Argumentation via Multi-labelling Systems. J Artif Intell Res. 2019;66:793-860.