Proceedings, Fifteenth International Conference on
Principles of Knowledge Representation and Reasoning (KR 2016)

Abstract Argumentation for Case-Based Reasoning

Kristijonas Čyras
Imperial College London
London, UK

Ken Satoh
National Institute of Informatics
Tokyo, Japan

Francesca Toni
Imperial College London
London, UK

Abstract

We investigate case-based reasoning (CBR) problems where cases are represented by abstract factors and (positive or negative) outcomes, and an outcome for a new case, represented by abstract factors, needs to be established. To this end, we employ abstract argumentation (AA) and propose a novel methodology for CBR, called AA-CBR. The argumentative formulation naturally allows to characterise the computation of an outcome as a dialogical process between a proponent and an opponent, and can also be used to extract explanations for why an outcome for a new case is (not) computed.

Introduction

Over the last twenty years, formal argumentation (as overviewed in (Rahwan and Simari 2009)) has gained importance within AI, as a generic framework to support various types of reasoning, including forms of case-based reasoning (CBR) (see e.g. (Bench-Capon and Modgil 2009; Gordon, Prakken, and Walton 2007; Gordon and Walton 2009; Prakken et al. 2015; Athakravi et al. 2015)). CBR itself is extensively used in AI in support of several applications (see e.g. (Richter and Weber 2013) for an overview). At a high-level, in CBR a reasoner in need to assess a *new case* recollects *past cases* and employs the ones most similar to the new case to give the assessment.

Here we propose a new argumentation-based approach to CBR, using argumentation to provide: 1) a method for computing outcomes for new cases, given the past cases and a *default* outcome; and 2) *explanations* for computed outcomes, in the form of dialogical exchanges between a proponent, in favour of the default outcome, and an opponent, against the default outcome. Our method exploits the power of argumentation for resolving conflicts amongst most similar past cases with conflicting outcomes and for justifying outcomes.

In our approach, like elsewhere in the literature (see e.g. (Horty and Bench-Capon 2012; Athakravi et al. 2015)), past cases are represented as sets of *factors* together with an outcome (factors are also known as features or attribute-value pairs (Sørmo, Cassens, and Aamodt 2005)). A common measure in this literature for finding the most similar past cases is setwise comparison of factors involved: past cases that have the largest subset of factors in common with the new case may be deemed most relevant for determining an outcome. However, criteria for similarity and relevance may diverge from one approach to another (see e.g. (Sørmo, Cassens, and Aamodt 2005; Richter and Weber 2013)). Moreover, having the most similar cases need not suffice to make a decision, because past cases may, and usually will, have conflicting outcomes.

The following example, used throughout the paper, illustrates CBR and the challenges that it poses.

Example 1. Alice has bought a chair from an online retailer, but wants to return it and get a refund. The retailer has a system, where a customer can claim for a refund by providing factual information about the situation. In Alice's case: she does not like the chair (factor A); she has used the chair (factor B); the chair shows no signs of wear and tear (C); Alice had the chair for more than 30 days (D). So an outcome for Alice's case $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ needs to be established. By default, the retailer will provide no refund $(-)$ when no factors are present. The retailer has a case base *CB* containing previous cases together with outcomes, e.g. consisting of: a case $(\{\mathsf{A}\}, +)$ with the outcome 'refund' $(+)$ if the customer does not like the chair; $(\{\mathsf{A}, \mathsf{B}\}, -)$ sustaining no refund if in addition the customer has used the chair; and $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$ when in further addition the chair is in a good condition. The outcome of the new (Alice's) case depends on the past cases most similar to the new case: since $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$ is the only such case, Alice should get refunded $(+)$.

But what if the case base contained $(\{\mathsf{A}, \mathsf{D}\}, -)$? Then there would be two nearest cases, $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$ and $(\{\mathsf{A}, \mathsf{D}\}, -)$. Would Alice be entitled to a refund?

We propose a method that uses abstract argumentation (AA) (Dung 1995) to provide a (negative) answer to the above question and to explain this recommendation dialectically as follows: the retailer needs to defend the default outcome $(-)$; Alice's claim rests on the factor A; so the retailer has to counter-argue the case $(\{\mathsf{A}\}, +)$ and may invoke past cases $((\{\mathsf{A}, \mathsf{B}\}, -)$ or $(\{\mathsf{A}, \mathsf{D}\}, -))$; while C favours Alice in the presence of both A and B (via $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$), it is ineffective when D occurs; so the retailer possesses a reasonable objection $(\{\mathsf{A}, \mathsf{D}\}, -)$ against satisfying Alice's claim, whence Alice should not be refunded.

Background

We recap the AA notions we use, adapted from (Dung 1995).

An *AA framework* is a pair $(Args, \rightsquigarrow)$, where *Args* is a set whose elements are called *arguments*, and $\rightsquigarrow$ is a binary *attack* relation on *Args*. For arguments $\mathsf{a}, \mathsf{b} \in Args$, if $\mathsf{a} \rightsquigarrow \mathsf{b}$, then we say that a *attacks* b. For sets of arguments $E, E' \subseteq Args$ and an argument $\mathsf{b} \in Args$, we say that:

- E *attacks* b, denoted $E \rightsquigarrow \mathsf{b}$, if $\exists \mathsf{a} \in E$ with $\mathsf{a} \rightsquigarrow \mathsf{b}$;
- E *attacks* E', denoted $E \rightsquigarrow E'$, if $\exists \mathsf{b} \in E'$ with $E \rightsquigarrow \mathsf{b}$;
- E is *conflict-free* if $E \not\rightsquigarrow E$;
- E *defends* $\mathsf{a} \in Args$ if for all $\mathsf{b} \rightsquigarrow \mathsf{a}$ it holds that $E \rightsquigarrow \mathsf{b}$;
- E is *admissible* if $E \not\rightsquigarrow E$ and E defends all $\mathsf{a} \in E$.

A conflict-free *extension* $E \subseteq Args$ is *grounded* if E is admissible, contains every argument it defends, and is $\subseteq$-maximal such. The grounded extension of any $(Args, \rightsquigarrow)$ always exists, is unique, and can be constructed inductively as $\mathcal{G} = \bigcup_{i \geqslant 0} G_i$, where G_0 is the set of unattacked arguments, and $\forall i \geqslant 0$, G_{i+1} is the set of arguments that G_i defends.

We will define explanations in terms of *dispute trees* (Dung, Kowalski, and Toni 2006; Dung, Mancarella, and Toni 2007), where a *dispute tree* for $\mathsf{a} \in Args$ (in $(Args, \rightsquigarrow)$) is a (possibly infinite) tree $\mathcal{T}$ such that:

1. every node of $\mathcal{T}$ is of the form $[\mathtt{L}:\mathsf{x}]$, with $\mathtt{L} \in \{\mathtt{P}, \mathtt{O}\}$ and $\mathsf{x} \in Args$: the node is *labelled* by argument x and assigned the *status* of either *proponent (*P*)* or *opponent (*O*)*;
2. the root of $\mathcal{T}$ is a P node labelled by a;
3. for every P node n, labelled by some $\mathsf{b} \in Args$, and for every $\mathsf{c} \in Args$ such that $\mathsf{c} \rightsquigarrow \mathsf{b}$, there exists a child of n, which is an O node labelled by c;
4. for every O node n, labelled by some $\mathsf{b} \in Args$, there exists at most[1] one child of n which is a P node labelled by some $\mathsf{c} \in Args$ such that $\mathsf{c} \rightsquigarrow \mathsf{b}$;
5. there are no other nodes in $\mathcal{T}$ except those given by 1-4.

The set of all arguments labelling P nodes in $\mathcal{T}$ is called the *defence set* of $\mathcal{T}$, denoted by $\mathcal{D}(\mathcal{T})$. A dispute tree $\mathcal{T}$ is an *admissible dispute tree* iff (i) every O node in $\mathcal{T}$ has a child, and (ii) no argument in $\mathcal{T}$ labels both P an O nodes. The defence set $\mathcal{D}(\mathcal{T})$ of an admissible dispute tree $\mathcal{T}$ is admissible; and if $\mathsf{a} \in E$, where E is admissible, then there exists an admissible dispute tree $\mathcal{T}$ for a with $\mathcal{D}(\mathcal{T}) \subseteq E$ being admissible (Dung, Mancarella, and Toni 2007).

AA-CBR

We consider a CBR setting with cases represented as sets of factors together with an outcome stemming from those factors. We assume a fixed but otherwise arbitrary (possibly infinite) set $\mathbb{F}$ whose elements are referred to as *factors*. We also assume a binary distribution $\{+, -\}$ of case *outcomes*.

Definition 2. A **case** is a pair (X, o) with a set of factors $X \subseteq \mathbb{F}$ and **outcome** $o \in \{+, -\}$. A **case base** is a finite set $CB \subseteq \wp(\mathbb{F}) \times \{+, -\}$ of cases such that:

for $(X, o_X), (Y, o_Y) \in CB$, if $X = Y$, then $o_X = o_Y$.

A **new case** is a set $N \subseteq \mathbb{F}$.

[1]The original definition of dispute tree requires that there exists *exactly* one child. As in (Fan and Toni 2015), we incorporate this requirement into the definition of *admissible* dispute tree instead.

We insist that all case bases are finite, cf. (Athakravi et al. 2015). The restriction that no two cases have the same set of factors but a different outcome, amounts to consistency (see e.g. (Horty and Bench-Capon 2012)). However, two cases may have different outcomes even though the set of factors of one is a subset of the set of factors of the other.

Example 3. In (the first part of) Example 1, we have a case base $CB = \{(\{\mathsf{A}\}, +), (\{\mathsf{A}, \mathsf{B}\}, -), (\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)\}$, and the new (Alice's) case is $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$.

Case bases can be used to determine the outcome of a new case, often by using the outcome of most similar past cases (Richter and Weber 2013), which in our setting can be understood as the *nearest* cases:

Definition 4. For a case base *CB* and a new case N, a past case $(X, o_X) \in CB$ is **nearest** to N if $X \subseteq N$, and there is no $(Y, o_Y) \in CB$ such that $Y \subseteq N$ and $X \subsetneq Y$.

Namely, (X, o_X) is nearest to N iff $X \subseteq N$ is $\subseteq$-maximal in the case base. In Example 3, the case $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$ in *CB* is nearest to the new case $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$. Based on this single nearest case, $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ can be assigned the outcome $+$.

In general, there can be distinct nearest past cases with different outcomes, as illustrated next.

Example 5. At the end of Example 1, the additional case yields a case base $CB' = CB \cup \{(\{\mathsf{A}, \mathsf{D}\}, -)\}$, where *CB* is as in Example 3. We see then that both $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$ and $(\{\mathsf{A}, \mathsf{D}\}, -)$ are nearest to the new case $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$.

Thus, it is not immediate to determine an outcome based solely on the nearest cases, and a more sophisticated method is required. In what follows, we define one such method.

From now on, unless stated otherwise, we assume a fixed, yet otherwise arbitrary case base *CB*, and a new case N. We assume that the user of our CBR system has a *default outcome* d in mind: in Example 3, it is $-$. We map case bases, default outcomes, and new cases into AA frameworks thus:

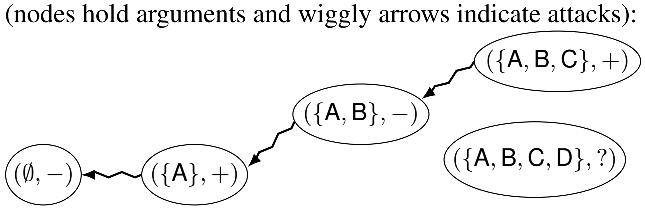
Definition 6. The **AA framework corresponding to *CB*, a default outcome** $d \in \{+, -\}$ **and** N is $(Args, \rightsquigarrow)$ satisfying the following conditions:

- $Args = CB \cup \{(N, ?)\} \cup \{(\emptyset, d)\}$;
- for $(X, o_X), (Y, o_Y) \in CB \cup \{(\emptyset, d)\}$, it holds that $(X, o_X) \rightsquigarrow (Y, o_Y)$ iff
  (1) $o_X \neq o_Y$, and (different outcomes)
  (2) $Y \subsetneq X$, and (specificity)
  (3) $\nexists (Z, o_X) \in CB$ with $Y \subsetneq Z \subsetneq X$; (concision)
- for $(Y, o_Y) \in CB$, $(N, ?) \rightsquigarrow (Y, o_Y)$ holds iff $Y \nsubseteq N$.

We refer to $(N, ?)$ as the **new case argument** and to $(\emptyset, d)$ as the **default case**.

So, we see cases as arguments. The new case argument cannot be attacked, and attacks the past cases whose factors are not contained in the new case, thus discarding 'irrelevant' factors from influencing the outcome. Attacks between past cases occur only if they have different outcomes (1), and are governed by specificity, determined by factor set inclusion: the attacking case has to be more specific than the attacked one (2); at the same the definition of attack accounts for concision, by forcing the attacker to be as close as possible to the attackee (3). Observe that the concision condition mimics a widely used *minimality* requirement on the attacking argument (see e.g. (Besnard and Hunter 2014; García and Simari 2014)). Specificity is used in some argumentation formalisms (e.g. (García and Simari 2014)) too.

Example 7. The AA framework $(Args, \rightsquigarrow)$ corresponding to the case base *CB*, the default outcome $-$ and the new case $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ in Example 3 can be represented graphically (nodes hold arguments and wiggly arrows indicate attacks):



The new case argument, $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}, ?)$, does not attack any argument as the new case contains factors of every case in *CB*. $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +)$ does not attack the default case, because $(\{\mathsf{A}\}, +)$ is more concise and attacks $(\emptyset, -)$.

If, say, $(\{\mathsf{A}, \mathsf{D}, \mathsf{E}\}, +)$ were in *CB*, then we would have $(\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}, ?) \rightsquigarrow (\{\mathsf{A}, \mathsf{D}, \mathsf{E}\}, +)$, since E is 'irrelevant'.

In the remainder of this paper, unless specified otherwise, we assume $(Args, \rightsquigarrow)$ to be the AA framework corresponding to the given case base *CB*, default outcome d and new case N. We will use $\overline{d}$, called the *complement* of d, to refer to $+$ if $d = -$, and to $-$ if $d = +$.

As case bases are finite, so will be their corresponding AA frameworks. From now on, $\mathcal{G}$ will denote the grounded extension of $(Args, \rightsquigarrow)$. This is guaranteed to always exist and be unique (see Background). Further, observe that the new case argument $(N, ?)$, being by definition unattacked in $(Args, \rightsquigarrow)$, always belongs to $\mathcal{G}$ (and therefore $\mathcal{G} \neq \emptyset$).

To determine the outcome of a new case we will use $\mathcal{G}$, by inspecting whether the default case $(\emptyset, d)$ is successfully defended by, and thus contained in, $\mathcal{G}$:

Definition 8. The **AA outcome** of the new case N is

- the default outcome d, if $(\emptyset, d) \in \mathcal{G}$;
- $\overline{d}$, otherwise, if $(\emptyset, d) \notin \mathcal{G}$.

Equating the outcome of the new case with the default case being in the grounded extension amounts to sceptically justifying the default outcome: in Example 3, the retailer has to successfully counter-argue every argument against $(\emptyset, -)$; otherwise, Alice has a valid claim for a refund.

Example 9. $(Args, \rightsquigarrow)$ depicted in Example 7 has $\mathcal{G} = \{(\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}, ?), (\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +), (\{\mathsf{A}\}, +)\}$ such that $(\emptyset, -) \notin \mathcal{G}$, so the AA outcome of $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ is $+$: Alice is refunded, as explained in the Introduction.

Consider now $CB' = CB \cup \{(\{\mathsf{A}, \mathsf{D}\}, -)\}$ from Example 5. In the corresponding $(Args', \rightsquigarrow')$, we find an additional attack $\{(\{\mathsf{A}, \mathsf{D}\}, -)\} \rightsquigarrow \{(\{\mathsf{A}\}, +)\}$, whence $\mathcal{G}' = \{(\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}, ?), (\{\mathsf{A}, \mathsf{B}, \mathsf{C}\}, +), (\{\mathsf{A}, \mathsf{D}\}, -), (\emptyset, -)\}$, so that the AA outcome of $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ is $-$.

We observe some basic properties of AA-CBR. The first result relates AA outcome to the notion of nearest case:

Lemma 1. *$\mathcal{G}$ contains all the nearest past cases to N.*

Our next result says that the unique past case nearest to N determines the AA outcome of N, no matter what the default outcome d is.

Proposition 2. *If there is a unique nearest case (X, o) to N, then for any $d \in \{+, -\}$, the AA outcome of N is o.*

Explanations of AA Outcomes

The notion of AA outcome allows to determine algorithmically whether a new case N should be assigned the default outcome (d) or not ($\overline{d}$), by determining whether or not (respectively) the default case belongs to the grounded extension of the corresponding AA framework. In this section we show how to determine an *explanation* for the computed AA outcome of N, exploiting the argumentative reinterpretation afforded by the corresponding AA framework. The notion of explanation is deemed crucial for CBR in many settings, but is inherently hard to define formally (see e.g. (Sørmo, Cassens, and Aamodt 2005)). In AA-CBR, however, explanations can be obtained naturally, as we aim to show in what follows.

A common form of explanation in CBR amounts to displaying the most similar cases. In addition, *transparency*, in not trying to "hide conflicting evidence" (Sørmo, Cassens, and Aamodt 2005, p. 134), is identified as desirable. By Lemma 1, the grounded extension provides a transparent explanation for an outcome, in that it contains all past cases nearest to the new case, be they of agreeing or diverging outcomes. However, simply presenting the nearest case(s) as explanation does not "help the user to understand how the symptoms connect with the solution" (Sørmo, Cassens, and Aamodt 2005, p. 128). The argumentative nature of AA-CBR naturally lends itself to a method of explanation based not only on the nearest cases, but on a dialectical exchange of past cases too. In particular, dispute trees can serve as (dialectical) explanations of why the AA outcome of a new case is the default outcome:

Definition 10. An **explanation for why the AA outcome of N is** d is any admissible dispute tree for $(\emptyset, d)$.

In Example 9, with case base CB', the admissible dispute tree $\mathcal{T}'$ (depicted below) is an explanation for why the AA outcome of $\{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ is $-$.

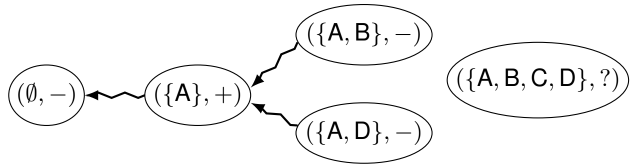
$$\mathcal{T}' \qquad \begin{array}{c} [\mathtt{P}:(\emptyset, -)] \\ | \\ [\mathtt{O}:(\{\mathsf{A}\}, +)] \\ | \\ [\mathtt{P}:(\{\mathsf{A}, \mathsf{D}\}, -)] \end{array}$$

Alice ($\mathtt{P}$) thus knows that the case $(\{\mathsf{A}, \mathsf{D}\}, -)$ has to be attacked in order to weaken the retailer's position.

Proposition 3. *If the AA outcome of the new case N is the default outcome d, then there is an explanation $\mathcal{T}$ for why the AA outcome of N is d (i.e. $\mathcal{T}$ is an admissible dispute tree for $(\emptyset, d)$), which is moreover such that the defence set $\mathcal{D}(\mathcal{T})$ is admissible and $\mathcal{D}(\mathcal{T}) \subseteq \mathcal{G}$.*

Note that there need not be a unique explanation, as shown in the following modification of Example 9.

Example 11. $CB = \{(\{\mathsf{A}\}, +), (\{\mathsf{A}, \mathsf{B}\}, -), (\{\mathsf{A}, \mathsf{D}\}, -)\}$, $d = -$, and $N = \{\mathsf{A}, \mathsf{B}, \mathsf{C}, \mathsf{D}\}$ yield $(Args, \rightsquigarrow)$ below.



$\mathcal{G} = \{(\{\mathsf{A},\mathsf{B},\mathsf{C},\mathsf{D}\},?), (\{\mathsf{A},\mathsf{D}\},-), (\{\mathsf{A},\mathsf{B}\},-), (\emptyset,-)\}$, so the AA outcome of N is $-$. There are two admissible dispute trees for $(\emptyset,-)$, namely

- $\mathcal{T}$: $[\mathtt{P}:(\emptyset,-)]$ — $[\mathtt{O}:(\{\mathsf{A}\},+)]$ — $[\mathtt{P}:(\{\mathsf{A},\mathsf{B}\},-)]$ and
- $\mathcal{T}'$: $[\mathtt{P}:(\emptyset,-)]$ — $[\mathtt{O}:(\{\mathsf{A}\},+)]$ — $[\mathtt{P}:(\{\mathsf{A},\mathsf{D}\},-)]$,

with defence sets $\mathcal{D}(\mathcal{T}) = \{(\{\mathsf{A},\mathsf{B}\},-),(\emptyset,-)\} \subseteq \mathcal{G}$ and $\mathcal{D}(\mathcal{T}') = \{(\{\mathsf{A},\mathsf{D}\},-),(\emptyset,-)\} \subseteq \mathcal{G}$, both admissible. Each tree is an explanation for why the proponent is justified in obtaining the default outcome $-$. Also, each indicates to the opponent where to focus to counter the proponent's claim.

Admissible dispute trees cannot themselves serve as explanations when the outcome is $\overline{d}$, as no such tree exists in this case. Instead, we can use *maximal dispute trees*:

Definition 12. A dispute tree $\mathcal{T}$ for some $\mathsf{a} \in Args$ is a **maximal dispute tree** iff for all opponent nodes $[\mathtt{O}:\mathsf{x}]$ which are leaves in $\mathcal{T}$ there is no $\mathsf{y} \in Args$ such that $\mathsf{y} \leadsto \mathsf{x}$.

Thus, in a maximal dispute tree, no opponent leaf node is 'attackable'. In $(Args', \leadsto')$ from Example 9 there are two maximal dispute trees, $\mathcal{T}^{\mathtt{O}}$: $[\mathtt{P}:(\emptyset,-)]$ — $[\mathtt{O}:(\{\mathsf{A}\},+)]$ — $[\mathtt{P}:(\{\mathsf{A},\mathsf{B}\},-)]$ — $[\mathtt{O}:(\{\mathsf{A},\mathsf{B},\mathsf{C}\},+)]$ and $\mathcal{T}^{\mathtt{P}}$: $[\mathtt{P}:(\emptyset,-)]$ — $[\mathtt{O}:(\{\mathsf{A}\},+)]$ — $[\mathtt{P}:(\{\mathsf{A},\mathsf{D}\},-)]$, for $(\emptyset,-)$.

Lemma 4. *An admissible dispute tree $\mathcal{T}$ for some $\mathsf{a} \in Args$ is a maximal dispute tree for $\mathsf{a} \in Args$.*

The converse does not hold: in Example 9, $\mathcal{T}^{\mathtt{P}}$ (as above) is admissible, $\mathcal{T}^{\mathtt{O}}$ (as above) is not.

Definition 13. An **explanation for why the AA outcome for N is $\overline{d}$** is any maximal dispute tree for $(\emptyset, d)$.

In (the first part of) Example 7, the AA outcome of $\{\mathsf{A},\mathsf{B},\mathsf{C},\mathsf{D}\}$ is $\overline{d} = +$, and there is a unique maximal dispute tree $\mathcal{T}^{\mathtt{O}}$ for $(\emptyset,-)$, as above.

A maximal dispute tree is an explanation of why the default outcome $\overline{d}$ was obtained in that it gives a dialectical justification of $\overline{d}$, including those opponent's arguments that are sufficient to establish $\overline{d}$. Then, an explanation indicates to the proponent which arguments s/he should counter.

Proposition 5. *If the AA outcome of the new case N is $\overline{d}$, then there is an explanation $\mathcal{T}$ for why the AA outcome of N is $\overline{d}$, and moreover $\mathcal{T}$ is such that $\mathcal{D}(\mathcal{T}) \nsubseteq \mathcal{G}$.*

To sum up, the argumentative reading in AA-CBR naturally yields an explanation mechanism for CBR, via off-the-shelf methods involving dispute trees.

To account for issues with explanations in settings similar to ours, (McSherry 2004) proposed criteria based on counting the ratio (or probability) of how often a factor appears in a case with the outcome $d/\overline{d}$. Instead, we use dispute trees to provide explanations without quantifying the appearance of factors, but will investigate such a possibility in the future.

Several works define methods for determining explanations for the (non-)acceptability of arguments in argumentation, e.g. (García et al. 2013; Fan and Toni 2015; Schulz and Toni 2016). These works use trees as the underlying mechanism for computing explanations, but not in a CBR setting. Other work in argumentation, e.g. (Cerutti, Tintarev, and Oren 2014), investigate the usefulness of explanation in argumentation with users. Similar explorations for our approach are also left for the future.

Acknowledgments. This work was partially supported by JSPS KAKENHI Grant Numbers 26280091.

References

Athakravi, D.; Satoh, K.; Law, M.; Broda, K.; and Russo, A. 2015. Automated Inference of Rules with Exception from Past Legal Cases Using ASP. In *LPNMR*.

Bench-Capon, T., and Modgil, S. 2009. Case Law in Extended Argumentation Frameworks. In *ICAIL*, 118–127.

Besnard, P., and Hunter, A. 2014. Constructing Argument Graphs with Deductive Arguments: A Tutorial. *Argument Comput.* 5(1):5–30.

Cerutti, F.; Tintarev, N.; and Oren, N. 2014. Formal Arguments, Preferences, and Natural Language Interfaces to Humans: an Empirical Evaluation. In *ECAI*, 207–212.

Dung, P. M.; Kowalski, R.; and Toni, F. 2006. Dialectic Proof Procedures for Assumption-Based, Admissible Argumentation. *Artif. Intell.* 170(2):114–159.

Dung, P. M.; Mancarella, P.; and Toni, F. 2007. Computing Ideal Sceptical Argumentation. *Artif. Intell.* 171(10-15):642–674.

Dung, P. M. 1995. On the Acceptability of Arguments and its Fundamental Role in Nonmonotonic Reasoning, Logic Programming and n-person Games. *Artif. Intell.* 77:321–357.

Fan, X., and Toni, F. 2015. On Computing Explanations in Abstract Argumentation. In *AAAI*, 1496–1502.

García, A., and Simari, G. 2014. Defeasible Logic Programming: DeLP-servers, Contextual Queries, and Explanations for Answers. *Argument Comput.* 5(1):63–88.

García, A.; Chesñevar, C.; Rotstein, N.; and Simari, G. 2013. Formalizing Dialectical Explanation Support for Argument-Based Reasoning in Knowledge-Based Systems. *Expert Syst. Appl.* 40:3233–3247.

Gordon, T., and Walton, D. 2009. Legal Reasoning with Argumentation Schemes. In *ICAIL*, 137–146.

Gordon, T.; Prakken, H.; and Walton, D. 2007. The Carneades Model of Argument and Burden of Proof. *Artif. Intell.* 171(10-15):875–896.

Horty, J., and Bench-Capon, T. 2012. A Factor-Based Definition of Precedential Constraint. *Artif. Intell. Law* 20(2):181–214.

McSherry, D. 2004. Explaining the Pros and Cons of Conclusions in CBR. In *ECCBR*, 317–330.

Prakken, H.; Wyner, A.; Atkinson, K.; and Bench-Capon, T. 2015. A Formalization of Argumentation Schemes for Legal Case-Based Reasoning in ASPIC+. *J. Log. Comput.* 25(5):1141–1166.

Rahwan, I., and Simari, G. 2009. *Argumentation in Artificial Intelligence*. Springer.

Richter, M., and Weber, R. 2013. *Case-Based Reasoning*. Springer.

Schulz, C., and Toni, F. 2016. Justifying Answer Sets Using Argumentation. *Theory Pract. Log. Program.* 16(1):59–110.

Sørmo, F.; Cassens, J.; and Aamodt, A. 2005. Explanation in Case-Based Reasoning–Perspectives and Goals. *Artif. Intell. Rev.* 24(2):109–143.