Representing and Extracting Support via Complement-based Argumentation Frameworks

Jack Mumford, Katie Atkinson and Trevor Bench-Capon

Department of Computer Science, University of Liverpool, L69 3BX, UK

Abstract

Both support and attack are essential concepts in natural argumentation. As originally introduced, however, abstract argumentation considered only attack. Although there have been attempts to add a support relation to abstract argumentation, these do not fulfil all desiderata. In this paper we show how the various notions of necessary and sufficient support can be captured using only the attack relation, and highlight the problematic nature of the notion of general support. We suggest that leveraging abstract argumentation semantics and the attack relation to represent support, and the consequent expression of argument in a simple graphical architecture, will yield computational benefits.

Keywords

Abstract argumentation, Support, Structured argumentation

1. Introduction

Although abstract argumentation has provided a highly effective way to analyse and evaluate sets of arguments, end users require a more intuitive interface (see [1] for a discussion of the usability of argumentation tools to support e-democracy). To exploit the wealth of formal technical work to enable automated reasoning to be conducted using abstract argumentation, presentation needs to use the concepts of natural argumentation. One such concept is *support*: arguments are seen not only as attacking one another, but also supporting one another. Attempts to capture this notion in abstract argumentation have been made using Bipolar Argumentation Frameworks (BAFs) [2], and using structured argumentation frameworks such as $ASPIC^+$ [3]. These attempts capture several different notions of support, and BAF notions are difficult to relate to the structured notions. Here we show that the concept of support can be subsumed into the attack relation, allowing for simple expression of the reasoning task in a standard abstract argumentation framework (AF) graphical form [4].

We build on ideas raised in [5, 6, 7] but our formalisation differs in that rules are not instantiated at the object level. Thus we have only statements and arguments as nodes in the AF graph, and we explicitly tie the representation of support via the attack relation alone to the formal theory of $ASPIC^+$ and BAF semantics. We take inspiration from the discussion of types

CMNA'22: Workshop on Computational Models of Natural Argument, September 12, 2022, Cardiff

Jack.Mumford@liverpool.ac.uk (J. Mumford); K.M.Atkinson@liverpool.ac.uk (K. Atkinson); tbc@liverpool.ac.uk (T. Bench-Capon)

https://jamumford.github.io (J. Mumford); https://www.csc.liv.ac.uk/~katie/ (K. Atkinson); https://www.csc.liv.ac.uk/~tbc/ (T. Bench-Capon)

of structured and abstract support from [8] and [9] (the interplay between the attack relation and an implicit rather than explicit support relation was mused on in [9]), and the avocation of a theory-based validation of abstract accounts of argumentation [10, 11]. Correspondence between the structured approach adopted by $ASPIC^+$ and abstract BAFs has been shown to be problematic [8]. We propose that the four ways in which support can be expressed in $ASPIC^+$, and the three semantics that exist for BAFs, have a corresponding representation using various argumentation semantics designed for complement-based argumentation frameworks (CoAFs).

2. Modelling support in argumentation

We take our definitions of the relevant theory from [11]. Four types of support are described as relevant in $ASPIC^+$. An argument a is a *proper subargument* of some argument b iff a consists solely of premises pertaining to b and is not equal to b. Arguments a and b *conclusion support* one another iff a and b are independent and have the same conclusion. An argument a *premise-supports* some argument b iff the conclusion of a is a premise of b. An argument a *intermediate-supports* some argument b iff the conclusion of a is not a premise of b but is the conclusion of a proper subargument of b. The last three represent defences to the standard types of attack: rebuttal, undermining and undercutting respectively.

BAFs present the support relation as distinct from the attack relation, such that the intersection of the two is the empty set. Graphically, a BAF is depicted as a digraph in which any edge between any two arguments in a specific direction may be an attack or support but not both. There are four types of attack which are used to define support in BAFs [8]. We denote that a attacks b, as $a \in b^-$ and $b \in a^+$. Argument a *supported attacks* some argument b iff there exists an argument c such that there is sequence of supports from a to c and $c \in b^-$. Argument a *secondary attacks* some argument b iff there exists an argument c such that there is a sequence of supports from c to b and $a \in c^-$. Argument a *extended attacks* some argument b iff there exists an argument c such that there is a sequence of supports from c to a and $c \in b^-$. Argument a *mediated attacks* some argument b iff there exists an argument c such that there is a sequence of supports from b to c and $a \in c^-$.

There are three BAF semantics, which are derived from closure properties under the four types of attack. *General support* [2] semantics is satisfied iff the attack relation is closed under supported and secondary attacks. *Necessary support* [12, 13] semantics is satisfied iff its attack relation is closed under secondary and extended attacks and the support relation is transitive. *Sufficient support* [14, 15] semantics (also known as *deductive support*) is satisfied iff its attack relation is closed under supported and mediated attacks and the support relation is transitive. We do not consider *evidentiary support* [13, 16] in this paper since it presupposes prima-facie arguments and is therefore not as general as the other three semantics.

3. Support via the attack relation

The methodology represents support via the use of an attack relation and the explicit invocation of the complements of statements/arguments in accordance with the type of support intended. Whilst our methodology should apply to asymmetric frameworks, here we assume symmetric attacks by default, to represent the symmetry of conflict and that elements are defeasible by default. We regard asymmetric attacks to represent some abstract preference ranking, although we do not discuss the means by which such preferences are determined (such as via value-based approaches, or argument schemes). We conjecture that the complement-based approach will maintain effective representation regardless of whether any given attack is symmetric or asymmetric, since the presence of conflict is what determines the relevant properties.

The objective is to produce semantics corresponding to the three BAF and the four $ASPIC^+$ interpretations of support using only an attack relation. In this paper, we do not present formal proofs of the relevant semantics, but we do present the foundations for this research in terms of appropriate representations and definitions. Firstly, we present in Definition 1 our specification of CoAFs used throughout this paper. Intuitively, one can see that Dung's AFs present a special case of CoAF in which no complements are expressed in A. Formal semantics for CoAFs are not presented in this paper.

Definition 1. (CoAFs) *A complement-based argumentation framework is a pair* $CoAF = \langle A, R \rangle$. *$A$ is a finite set of nodes representing arguments/statements, where for each node $p_k \in A$ its complement $\overline{p_k}$ may also be in A. R is a binary attack relation on A such that $R \subseteq A \times A$, where for each node p_k in A the set of attackers of p_k is denoted p_k^- and the set of nodes attacked by p_k as p_k^+. Finally, to preserve non-contradiction, for any node p_i that attacks another node p_j, each node in $\overline{p_j}^-$ must also attack and/or be attacked by p_i.*

We now move on to the BAF interpretations, Definitions 2 and 3 offer formal expressions that we conjecture satisfy the closure properties for necessary and sufficient support respectively. In essence, support is indicated by an conflict with complements, where a is necessary support for b if the complement of a is in conflict with b, and a is sufficient support for b if a is in conflict with the complement of b.

Figure 1 offers a simple illustration of the duality of the attack relation that gives rise to these types of support. The figure treats the attack relation as symmetric, which applies when one accepts modus tollens by default and ignores preferential conflict. Of course, in standard argumentation we frequently relax the requirement for modus tollens and permit asymmetric attacks. Whilst we do not wish to limit the discussion to symmetric frameworks, Figure 1 does assist in visualising the relationship between the two types of support. In the figure, a is necessary support for b, and b is sufficient support for a. One can check for closure under each of the four types of attack, with the thick line indicating a mandatory attack that must be added to the attack relation to ensure consistency with the BAF support definitions. In so doing we allow for the appropriate attack type closures, and consistency of complement labellings, such that for each complement pairing if one is labelled IN, then the other is labelled out, and vice versa.

Of course, as already stated, Figure 1 depicts necessary and sufficient support with symmetric attacks. However, Definitions 2 and 3 generalise to asymmetric attack relations, in which contrary statements/arguments may have some preference ordering and modus tollens is abandoned. Figures 2 and 3 unpack Figure 1 to illustrate the two cases for each type of support in accordance with Definitions 2 and 3. In both figures, the left graph represents the graphical architecture where the second condition is trivially satisfied ($b \notin \overline{a}^-$ for necessary support, and $a \notin \overline{b}^-$ for sufficient support), and the right graph represents where the second condition is

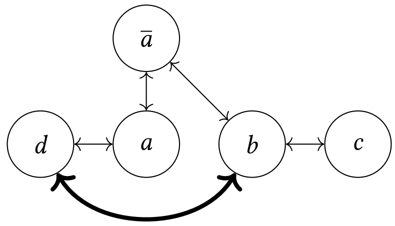

Figure 1: Abstract AF representation, where a is necessary support for b, and where if a defeat relationship between b and d is leveraged, we have closure under secondary and extended attacks, but not under supported and mediated attacks as per the BAF semantics for necessary support. By duality, b is sufficient support for a, and where if a defeat relationship between a and c is leveraged, we have closure under supported and mediated attacks, but not under secondary and extended attacks as per the BAF semantics for sufficient support.

meaningfully satisfied. In the right graphs, the thick line attack is analogous to the thick line attack that is added in Figure 1 in order to maintain rationality. Note that in Figures 2 and 3 the symmetry of attack of the thick line is not mandatory; as long as at least one node attacks the other then the definitions are satisfied and the closure properties upheld.

Definition 2. (Necessary support) *For any $a, b \in A$, for some argumentation framework (A, R), we say that a is necessary support for b iff*

1. $\overline{a} \in b^- \cup b^+$*; and*
2. $\forall c : (c \in a^-), b \in \overline{a}^- \implies b \in c^- \cup c^+$.

Figure 2: Two abstract AF representations where a is necessary support for b, providing closure under secondary and extended attacks. In the right graph, since $b \in \overline{a}^-$ we must have $b \in c^- \cup c^+$.

Definition 3. (Sufficient support) *For any $a, b \in A$, for some argumentation framework (A, R), we say that a is sufficient support for b iff*

1. $a \in \overline{b}^- \cup \overline{b}^+$*; and*
2. $\forall c : (c \in b^-), a \in \overline{b}^- \implies a \in c^- \cup c^+$.

For some argument/statement a to be *necessary* for b, then for every labelling in which $L(b) = \text{IN} \implies L(a) = \text{IN}$. For some argument/statement a to be *sufficient* for b, then for every

Figure 3: Two abstract AF representations where a is sufficient support for b, providing closure under supported and mediated attacks. In the right graph, since $a \in \overline{b}^{-}$ we must have $a \in c^{-} \cup c^{+}$.

labelling in which $L(a) = \text{in} \implies L(b) = \text{in}$. Yet, these specifications are not aligned with an intuitive notion of support as used independently of existing attacks. That is, necessary and sufficient support should be expressible in an attack relation, but not be a context-dependent artefact of argumentation dynamics. Let us consider an illustrative example:

Example 1. (BAF support) *Given statements a* = *itisavehicle* , *b* = *it is a plane, c* = it is not mechanical *and d* = it cannot fly, *we can derive a CoAF as in Figure 1. Intuitively a is necessary support for b, and b is sufficient support for a. Indeed every labelling in which $L(b) = \text{in}$, we have $L(a) = \text{in}$. On the other hand, we can see that in every labelling where $L(d) = \text{in}$, we have $L(c) = \text{in}$. This might imply that d is sufficient support for c, and that c is necessary support for d. But intuitively we can think of counter examples to this relationship. If we were to add a statement e* = it is a bird, *then the attack relation would be adjusted with conflict between e and a, b and c, leaving a possible labelling in which $L(e) = \text{in}$, $L(d) = \text{in}$ and $L(c) = \text{out}$. Conversely, no additional statements or attacks can be added in a manner coherent with CoAF semantics such that the necessary and sufficient relationships between a and b are removed such that $L(b) = \text{in} \not\Longrightarrow L(a) = \text{in}$. Hence we distinguish between artefacts of the labellings resulting from incomplete knowledge representations, and genuine necessary and sufficient support relations which are evoked by specific interactions in the attack relation, which will hold regardless of any growth of the statement/argument set and the accompanying attack relation.*

We suggest that *necessary support* and *sufficient support* are readily compatible with expression under modified abstract argumentation semantics and will adhere to the rationality postulates from [17]. However, we suggest that the modifications required to express *general support* are rationally incoherent, indicating problems with the use of general support in practical reasoning. Whilst not formally proven here, one can intuitively see in Figure 1 how closure under secondary and extended attacks are connected graphically, and how closure under supported and mediated attacks are connected. Trying to separate closure under secondary attacks from closure under extended attacks, and closure under supported attacks from closure under mediated attacks, appears to be highly problematic. Formal proofs will need to be forthcoming; it was suggested in [18] that the attack relation was incapable of expressing this notion of support, but this would require confirmation with explicit use of complements. Nonetheless, the trouble with representing *general support* lends weight to the criticism of the rationality of this type of support that was raised in [11].

4. Extracting arguments from the attack relation

The descriptions of necessary and sufficient support have been framed as applying to frameworks in which the nodes can be either statements or arguments. However, one might find more application when the nodes are statements and the attack relation can be used to express support in the form of argument structure, as in Example 1. If a CoAF consists of statements as nodes, one can express argument structure and support as defined for $ASPIC^+$ in accordance with Definitions 1, 2 and 3 and their graphical representations in Figures 2 and 3.

Recall that there are four types of support available in an $ASPIC^+$ framework, which are illustrated in Figure 4 as a CoAF. Arguments can be extracted from a CoAF by selecting a starting node to act as the claim, and establishing the remaining structure in accordance with nodes providing sufficient support in an iterative manner. In order to represent the structure appropriately, we allow for premises, claims, and collectors to be expressed as nodes. Collectors are nodes that are used to represent rules from premise/s to claim by presenting sufficient support for the claim and receiving necessary support from the premise/s. Collectors can represent defeasible rules from a conjunction or single premise, as well as strict rules from a sufficient conjunction of premises (strict rules from a single premise do not need a collector node). Strict rules from a conjunction of necessary premises require a strict collector node, and require that every necessary supporter not in the conjunction is sufficiently supported by a node in the conjunction (see Examples 2, 3, 4 and 5). Figure 4 is restricted to strict rules in order to more concisely represent the four types of $ASPIC^+$ support, and is illustrated further in Example 2. We will demonstrate use with defeasible rules in Examples 3, 4 and 5 when indicating how the attack relation incorporated the three types of $ASPIC^+$ attack: rebuttal, undercut, and undermine.

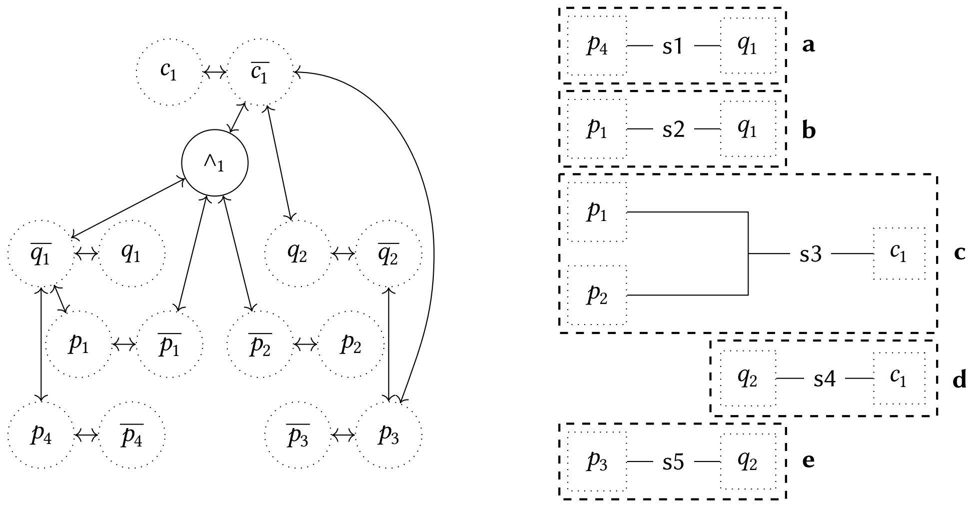

Figure 4: Abstract complement-based AF representation of argument components (left) from which arguments **a, b, c, d** and **e** are extracted (right) to demonstrate the four types of support in $ASPIC^+$. Namely: **b** is a proper subargument of **c**; **c** and **d** conclusion-support one another; **e** premise-supports **d**; and **a** intermediate supports **c**. Note that dotted edges indicates defeasible components.

Example 2. ($ASPIC^+$ support) *Given statements* p_1 = *Josh has four oranges,* p_2 = *Josh has six apples,* p_3 = Josh has ten nectarines, p_4 = Josh has two limes, q_1 = *Josh has citrus fruit,* q_2 = Josh has at least ten stone fruit *and* c_1 = Josh has at least ten fruit, *we can derive a CoAF and extract arguments as in Figure 4. We can provide two examples of argument extraction, for arguments* ***d*** *and* **c**. *When a collector node is not involved then the process is simple: for argument* ***d*** *we begin with* c_1 *as the claim, and since* q_2 *is a sufficient supporter by itself then we have a strict inference rule from* q_2 *to* c_1. *For argument* **c**, *we begin with* c_1 *as the claim, but the sufficient supporter* $\wedge_1$ *is a collector, which means we use necessary supporters of* $\wedge_1$, *the premises* p_1 *and* p_2, *which provide a strict inference rule for* c_1, *since* q_1 *is sufficiently supported by* p_1.

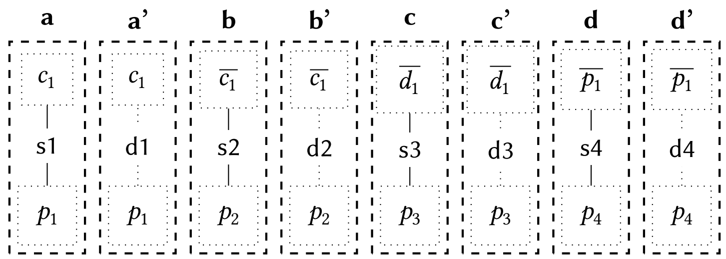

Figure 5: Arguments to be used to illustrate representation of rebutting (Figure 6), undercutting (Figure 7), and undermining (Figure 8) attacks. Each argument has two abstractions: 1) where the inference rule is strict (e.g. a, b, etc); 2) where the rule is defeasible (e.g. a', b', etc).

We will now showcase how the CoAF approach can express the three types of attack defined for $ASPIC^+$: rebutting, undercutting, and undermining. Figure 5 provides the arguments used in Figures 6, 7 and 8 and in Examples 3, 4 and 5.

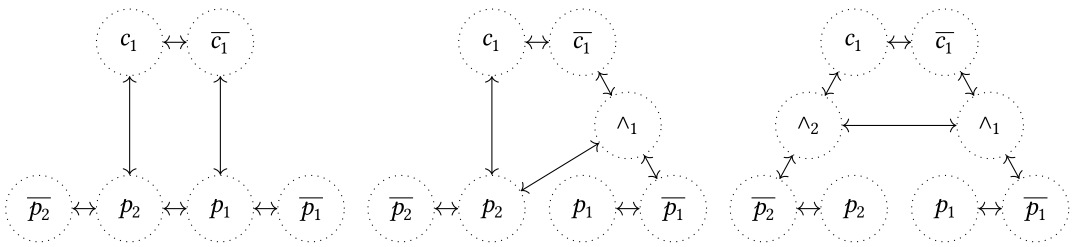

Figure 6: Complement-based AF representation of three variants of **rebutting** attacks using arguments from Figure 5. Namely rebuttal: between arguments **a** and **b** with only strict rules (left); where **a'** contains a defeasible rule and **b** contains a strict rule (middle); and between arguments **a'** and **b'** with only defeasible rules (right).

Example 3. (Rebutting) *Given statements* p_1 = *Murphy is devilishly handsome,* p_2 = *Murphy has missed the date, and* c_1 = *Murphy will have a successful date, we can derive a CoAF and extract arguments in accordance with the top right graph in Figure 6. We regard the rule* $s_2 = p_2 \implies \overline{c_1}$ *as strict and so argument* ***b*** *is straightforward to extract. However the rule* $d_1 = p_1 \implies c_1$ *is regarded as defeasible, which means it must be made explicit in the graph and the collector node* $\wedge_1$ *is added to collect the rule. Argument* ***a'*** *is extracted by beginning with* c_1, *moving to* $\wedge_1$ *as a sufficient* *supporter, and selecting p_1 as a necessary supporter. Since $\wedge_1$ has been marked as defeasible, the inference rule $d_1 = p_1 \implies c_1$ must be defeasible. Thus we have extracted arguments **a'** and **b** which attack one another via rebuttal.*

Figure 7: Complement-based AF representation of two variants of **undercutting** attacks using arguments from Figure 5. Namely: where **c** contains a strict rule ($\wedge_1$ be strict or defeasible) and undercuts **a'** (left); and where **c'** contains a defeasible rule and undercuts **a'** (right).

Example 4. (Undercutting) *Given statements p_1 = The weather forecaster says it will rain tomorrow, p_3 = Weather forecasters are wrong sometimes, and c_1 = It will rain tomorrow, we can derive a CoAF and extract arguments in accordance with the left graph in Figure 7. We regard the rule $p_1 \wedge \overline{p_3} \implies c_1$ as strict and so $\wedge_1$ is a strict collector. Nonetheless we extract argument **a'** by beginning with c_1, moving to $\wedge_1$ as a sufficient supporter, and selecting p_1 as a necessary supporter. Since p_1 is not the sole necessary supporter of $\wedge_1$ and $\overline{p_3}$ is not sufficiently supported by p_1, the inference rule $d_1 = p_1 \implies c_1$ must be defeasible. Argument **c** is extracted by beginning with $\overline{\wedge_1}$ ($\overline{\wedge_1}$ is not represented graphically since it has no effect other than symbolic) and moving to the sufficient supporter p_3 to derive the strict rule $s_3 = p_3 \implies \overline{d_1}$, since p_3 attacks all rules that rely on $\wedge_1$ (which is only d_1 in this case). Thus we have extracted arguments **a'** and **c**, where **c** undercuts **a'**.*

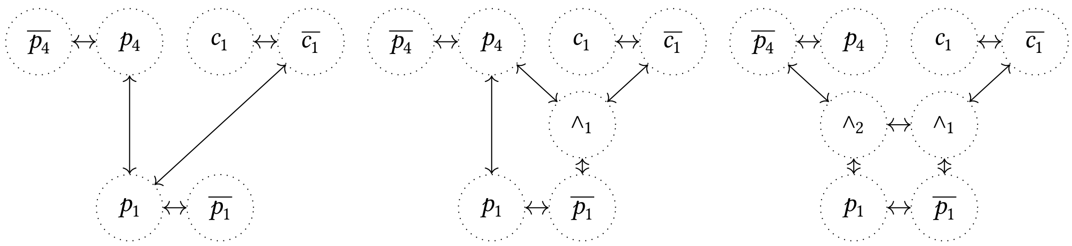

Figure 8: Complement-based AF representation of three variants of **undermining** attacks using arguments from Figure 5. Namely: with only strict rules, where **d** undermines **a** (left); where **d** contains a strict rule and undermines **a'** which contains a defeasible rule (middle); and with only defeasible rules, where **d'** undermines **a'** (right).

Example 5. (Undermining) *Given statements p_1 = A Bordeaux is a vastly superior wine to a Claret, p_4 = Bordeaux and Claret are the same, and c_1 = I shall order a Bordeaux wine, we can derive a CoAF and extract arguments in accordance with the top right graph in Figure 8. We regard the rule $s_4 = p_4 \implies \overline{p_1}$ as strict and so argument **d** is straightforward to extract. Argument **a'** is extracted as in Example 3. Thus we have extracted arguments **a'** and **d**, where **d** undermines **a'**.*

5. Concluding Remarks

We have presented a means of representing support solely via the use of the attack relation. The various types of support for abstract BAF semantics, and $ASPIC^+$ frameworks, have been examined and we proposed definitions for necessary and sufficient support that we conjecture are capable of representing these types of support. We do not, however, capture general support for BAF semantics: we believe that general support as a practical and natural notion of support is problematic and intend to explore this unease further. Several examples were suggested in order to illustrate how one may extract arguments from our CoAFs. This would enable supporting arguments to be used as part of the explanations offered to users. Next steps will be to formally prove that our definitions of necessary and sufficient support fulfill the BAF definitions via closure under the various types of attack. Extending the formal analysis to general support and evidentiary support as CoAFs, would be fruitful research directions.

Being able to incorporate support into the attack relation allows for the calculation of acceptability via an AF and some labelling semantics, which has potential benefits in terms of ease in comparison with, for instance, $ASPIC^+$ which frequently duplicates statements that are expressed in more than one argument, complicating calculation. We also consider that an AF may be easier to integrate with machine learning (ML) models that are commonly graph-based, which would be advantageous for building hybrid ML/argumentation systems.

References

[1] D. R. Cartwright, Digital decision-making: using computational argumentation to support democratic processes, Ph.D. thesis, University of Liverpool, 2011.

[2] C. Cayrol, M.-C. Lagasquie-Schiex, On the acceptability of arguments in bipolar argumentation frameworks, in: European Conference on Symbolic and Quantitative Approaches to Reasoning and Uncertainty, Springer, 2005, pp. 378–389.

[3] S. Modgil, H. Prakken, The aspic+ framework for structured argumentation: a tutorial, Argument & Computation 5 (2014) 31–62.

[4] P. M. Dung, On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games, Artificial intelligence 77 (1995) 321–357.

[5] A. Wyner, T. Bench-Capon, P. Dunne, On the instantiation of knowledge bases in abstract argumentation frameworks, in: International Workshop on Computational Logic in Multi-Agent Systems, Springer, 2013, pp. 34–50.

[6] A. Wyner, T. Bench-Capon, P. Dunne, F. Cerutti, Senses of 'argument'in instantiated argumentation frameworks, Argument & Computation 6 (2015) 50–72.

[7] A. Z. Wyner, T. J. Bench-Capon, K. Atkinson, Three senses of "argument", in: Computable Models of the Law, Springer, 2008, pp. 146–161.

[8] A. Cohen, S. Parsons, E. I. Sklar, P. McBurney, A characterization of types of support between structured arguments and their relationship with support in abstract argumentation, International Journal of Approximate Reasoning 94 (2018) 76–104.

[9] L. Yu, R. Markovich, L. van der Torre, Interpretation of support among arguments, in: Legal Knowledge and Information Systems, 2020.
[10] H. Prakken, On support relations in abstract argumentation as abstractions of inferential relations, in: ECAI 2014, IOS Press, 2014, pp. 735–740.
[11] H. Prakken, On validating theories of abstract argumentation frameworks: the case of bipolar argumentation frameworks., in: CMNA 2020, CEUR Workshop Proceedings, 2020, pp. 21–30.
[12] F. Nouioua, V. Risch, Argumentation frameworks with necessities, in: International Conference on Scalable Uncertainty Management, Springer, 2011, pp. 163–176.
[13] N. Oren, M. Luck, C. Reed, Moving between argumentation frameworks, in: Proceedings of COMMA 2010, IOS Press, 2010.
[14] G. Boella, D. M. Gabbay, L. van der Torre, S. Villata, Support in abstract argumentation, in: Proceedings of COMMA 2010, IOS Press, 2010, pp. 40–51.
[15] C. Cayrol, M.-C. Lagasquie-Schiex, Bipolarity in argumentation graphs: Towards a better understanding, International Journal of Approximate Reasoning 54 (2013) 876–899.
[16] N. Oren, T. J. Norman, Semantics for evidence-based argumentation, in: Proceedings of COMMA 2008, IOS Press, 2008, pp. 276–284.
[17] M. Caminada, L. Amgoud, On the evaluation of argumentation formalisms, Artificial Intelligence 171 (2007) 286–310.
[18] N. Potyka, Generalizing complete semantics to bipolar argumentation frameworks, in: European Conference on Symbolic and Quantitative Approaches with Uncertainty, Springer, 2021, pp. 130–143.