

From Contexts to Values: Context-Dependent Defeat in Abstract Argumentation

Albert Sadowski^{1,*}, Jarosław A. Chudziak^{1,†}

¹*Faculty of Electronics and Information Technology, Warsaw University of Technology, Poland*

Abstract

In value-based argumentation, an audience's ordering of values decides which attacks succeed as defeats. In many settings the deciding factor is not the audience but the circumstances: the same attack may succeed at one procedural stage, or under one regulation, and fail at another. Context-dependent argumentation frameworks (CDAFs), a model we recently introduced, capture this directly: one attack relation and a defeat function that switches each attack on or off per context, so every context induces an ordinary Dung framework. This raises a reduction question: can one value assignment with per-context orderings reproduce the defeat function, collapsing the CDAF into a VAF? We present a polynomial-time decision procedure for this question and map the harder neighbouring problems, with upper bounds from NP to Σ_3^P . We also present a validated reference implementation and a measurement: representability is rare and falls fast with the number of contexts.

Keywords

abstract argumentation, value-based argumentation, computational argumentation

1. Introduction

In abstract argumentation, a Dung framework fixes a set of arguments and an attack relation, and acceptance is computed from that fixed structure [1]. Value-based argumentation frameworks (VAFs) added an influential source of variation: arguments promote values, an audience orders the values, and the ordering decides which attacks succeed as defeats [2]. One structure, many audiences, many outcomes.

In many settings the variation does not come from who evaluates, but from the circumstances under which the evaluation happens. A safety argument may defeat a cost argument during an incident and fail to defeat it during routine planning. The arguments and the attack stay the same; what changes is whether the attack succeeds. The circumstance is external to the arguments, and existing formalisms do not take it as a parameter. In recent work we introduced context-dependent argumentation frameworks (CDAFs) to model it directly [3]: a defeat function δ marks each attack as operative or suppressed in each context, so a CDAF is a finite family of Dung frameworks over one shared attack structure.

The family view raises three questions. First, is contextual variation genuinely new? VAFs already vary defeat through audiences, so a CDAF might be a VAF in disguise: one value assignment and one ordering per context reproducing δ exactly. When does this reduction hold, and how fast can it be decided? Second, when δ cannot be reproduced, can a VAF at least reproduce the extensions of every induced framework, and at what cost? Third, from a systems perspective: can the decision procedure be implemented and validated, and how common is reducibility?

Context appears in argumentation in several roles: as modules that constrain one another [4], or as agents' backgrounds moving toward consensus [5]. Closer to our question, other extensions vary which attacks succeed, but internally or epistemically: extended frameworks argue about preferences inside the framework [6], and incomplete frameworks induce a family of completions read as possible worlds [7]. Property-based preferences derive an argument ordering from properties attached to arguments [8];

SAFA'26: Sixth International Workshop on Systems and Algorithms for Formal Argumentation, September, 2026, Barcelona, Spain

*Corresponding author.

† These authors contributed equally.

✉ albert.sadowski.stud@pw.edu.pl (A. Sadowski); jaroslaw.chudziak@pw.edu.pl (J. A. Chudziak)

id 0009-0001-4524-5488 (A. Sadowski); 0000-0003-4534-8652 (J. A. Chudziak)



© 2026 Copyright for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

a value assignment is the special case of one property per argument, and our value-consistent partition fixes the granularity at which such a derived ordering can separate arguments. A CDAF's variation is external: contexts are a parameter, and the family is evaluated as a whole.

Section 3 gives a structural criterion, a decision procedure running in $O(n^2k + nmk)$ time for n arguments, m attacks, and k contexts, and a trace on an instance built to exercise every phase. Section 4 maps the realisability landscape, where the upper bounds run from NP to Σ_3^P with no matching lower bounds. Section 5 presents a validated reference implementation and a measurement. Section 6 lists open problems.

2. Preliminaries

An *argumentation framework* (AF) is a pair $F = \langle \mathcal{A}, R \rangle$ with $\mathcal{A}$ a finite set of arguments and $R \subseteq \mathcal{A} \times \mathcal{A}$ an attack relation [1]. A semantics σ assigns to F a set $\sigma(F)$ of *extensions*, subsets of $\mathcal{A}$. We use the grounded, complete, preferred, and stable semantics with their standard definitions, and refer to Baroni et al. [9]. Throughout, $n = |\mathcal{A}|$ and $m = |R|$.

A *value-based argumentation framework* (VAF) extends an AF with a finite set of values V , an assignment $val: \mathcal{A} \rightarrow V$, and a set of audiences [2], each audience p being a strict total order $\succ_p$ on V . An attack $(a, b) \in R$ succeeds as a *defeat* for p iff $val(b) \not\succ_p val(a)$, so it fails only if the attacked argument carries a strictly preferred value. Audience p thus induces the defeat relation $R_p = \{(a, b) \in R : val(b) \not\succ_p val(a)\}$.

Definition 1 ([3]). A context-dependent argumentation framework (CDAF) is a tuple $D = \langle \mathcal{A}, R, \mathcal{C}, \delta \rangle$, where $\mathcal{A}$ is a finite set of arguments, $R \subseteq \mathcal{A} \times \mathcal{A}$ is an attack relation, $\mathcal{C}$ is a non-empty finite set of contexts, and $\delta: \mathcal{C} \times R \rightarrow \{0, 1\}$ is the defeat function.

Each context $c \in \mathcal{C}$ induces the defeat relation $R_c = \{(a, b) \in R : \delta(c, (a, b)) = 1\}$ and the Dung framework $AF_c = \langle \mathcal{A}, R_c \rangle$; a suppressed attack is deleted, not reversed (see Section 3). Attacks in R are *latent* conflicts; attacks in R_c are the *operative* defeats in c . We write $k = |\mathcal{C}|$.

Cross-context acceptance classifies arguments across the family. Under a semantics σ , an argument is *universally accepted* if it belongs to every σ -extension of every AF_c , *indefensible* if it belongs to none, and *context-contingent* otherwise.

3. Representability and Its Algorithm

VAF-REPRESENTABILITY asks whether audiences suffice to explain a CDAF's variation. Given $D = \langle \mathcal{A}, R, \mathcal{C}, \delta \rangle$, are there a value set V , an assignment $val: \mathcal{A} \rightarrow V$, and a map φ from $\mathcal{C}$ to strict total orders on V with $\delta(c, (a, b)) = 1$ iff $val(b) \not\succ_{\varphi(c)} val(a)$, for every $c \in \mathcal{C}$ and $(a, b) \in R$? The VAF must reproduce every entry of δ ; matching only extensions is the weaker realisability problem of Section 4.

3.1. The criterion and the algorithm

The criterion is a partition condition. For a partition $\mathcal{P}$ of $\mathcal{A}$, write $[a]$ for the class of argument a .

Definition 2 (value-consistent partition). A partition $\mathcal{P}$ of $\mathcal{A}$ is value-consistent for D if the following hold.

- (I) Intra-class unanimity: for all $(a, b) \in R$ with $[a] = [b]$ and all $c \in \mathcal{C}$: $\delta(c, (a, b)) = 1$.
- (U) Cross-class uniformity: for all $(a_1, b_1), (a_2, b_2) \in R$ with $a_1, a_2 \in [\alpha]$, $b_1, b_2 \in [\beta]$, and $[\alpha] \neq [\beta]$: $\delta(c, (a_1, b_1)) = \delta(c, (a_2, b_2))$ for all $c \in \mathcal{C}$.
- (C) Cross-class complementarity: for all $(a, b), (b', a') \in R$ with $a, a' \in [\alpha]$, $b, b' \in [\beta]$, and $[\alpha] \neq [\beta]$: $\delta(c, (a, b)) + \delta(c, (b', a')) = 1$ for all $c \in \mathcal{C}$.

- (A) Cross-class acyclicity: for each $c \in \mathcal{C}$ the relation $\triangleright_c$ on $\mathcal{P}$ is acyclic, where for distinct classes $[\alpha] \triangleright_c [\beta]$ iff some attack from $[\alpha]$ to $[\beta]$ succeeds in c , or some attack from $[\beta]$ to $[\alpha]$ fails in c .

Theorem 1. A CDAF is VAF-representable iff it admits a value-consistent partition.

The conditions unfold the arithmetic of strict total orders. Equal values never block a defeat, so intra-class attacks must always succeed, which is (I). An audience compares values rather than individual attacks, so parallel cross-class attacks share one outcome, which is (U), and opposite cross-class attacks have exactly one winner, which is (C). A successful attack from $[\alpha]$ to $[\beta]$ and a failed attack from $[\beta]$ to $[\alpha]$ both witness that $[\alpha]$'s value is strictly preferred in c ; $\triangleright_c$ collects these forced preferences, and by Szpilrajn's theorem [10] they extend to a strict total order iff they contain no cycle, which is (A). Conversely, take $V = \mathcal{P}$, $val(a) = [a]$, and any $\succ_c$ extending $\triangleright_c$. Under (U) and (C) the two clauses never conflict, so $\triangleright_c$ is well defined and antisymmetric, and three cases reproduce δ : an intra-class attack has equal values, so it defeats, and (I) gives $\delta = 1$; a succeeding cross-class attack gives $[a] \triangleright_c [b]$, hence $val(b) \not\succ_c val(a)$; a failing one gives $[b] \triangleright_c [a]$, hence $val(b) \succ_c val(a)$.

The search for a partition is polynomial because all merges are forced: Algorithm 1 starts from singleton classes and only ever merges.

Theorem 2. VAF-representability is decidable in $O(n^2k + nmk)$ time.

The correctness argument rests on two invariants. First, at every stage the current partition refines every value-consistent partition, because Phase 1 merges are forced by (C) and Phase 3 merges by (A): a $\triangleright_c$ cycle over distinct classes can only be repaired by making them coincide. Second, any partition reaching Phase 4 is value-consistent, and this is checkable rather than asserted: between two distinct

Algorithm 1 Deciding VAF-representability of a CDAF $\langle \mathcal{A}, R, \mathcal{C}, \delta \rangle$

```

1:  $\mathcal{P} \leftarrow \{\{a\} : a \in \mathcal{A}\}$   $\triangleright$  union-find over  $\mathcal{A}$ 
   Phase 1: co-equality merges
2: for all mutual pairs  $(a, b), (b, a) \in R$  do
3:   if  $\delta(c, (a, b)) = 1$  and  $\delta(c, (b, a)) = 1$  for every  $c \in \mathcal{C}$  then
4:     merge the classes of  $a$  and  $b$ 
5:   end if
6: end for
7: repeat
   Phase 2: intra-class check (the only failure point)
8:   for all  $(a, b) \in R$  with  $[a] = [b]$  do
9:     if  $\delta(c, (a, b)) = 0$  for some  $c \in \mathcal{C}$  then
10:      return not representable
11:    end if
12:   end for
   Phase 3: per-context cycle merges
13:    $merged \leftarrow \text{false}$ 
14:   for all  $c \in \mathcal{C}$  do
15:     build  $\triangleright_c$  on the classes of  $\mathcal{P}$ ; record its non-trivial SCCs
16:   end for
17:   for all recorded non-trivial SCCs  $S$  do
18:     merge the classes in  $S$ ;  $merged \leftarrow \text{true}$ 
19:   end for
20: until  $merged = \text{false}$ 
   Phase 4: verification
21: check (U), (C), (A) for  $\mathcal{P}$ 
22: return  $\mathcal{P}$   $\triangleright$  the finest value-consistent partition

```

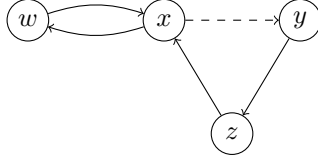


Figure 1: The worked instance. Left: the attack relation R ; the dashed attack (x, y) is the only attack suppressed in some context. Right: the defeat function δ .

classes, a violation of (U) or (C) in a context c always produces a two-cycle in $\triangleright_c$. For (U), the succeeding parallel attack gives $[\alpha] \triangleright_c [\beta]$ and the failing one gives $[\beta] \triangleright_c [\alpha]$; for (C), both succeeding and both failing each give edges in both directions. Phase 3 merges every such cycle, so Phase 4 never fails; Phase 1 is an optimisation, its merges being (C) violations a Phase 3 pass would find anyway. The algorithm therefore rejects exactly the non-representable instances and otherwise returns the unique finest value-consistent partition. Phase 3 tests each $\triangleright_c$ separately, never their union: condition (A) is imposed per context, and edges from different contexts may form a union cycle while every single $\triangleright_c$ stays acyclic.

For the running time: each round costs $O(mk)$ to run Phase 2 and build the relations $\triangleright_c$. Each $\triangleright_c$ carries at most m edges, one per attack, so its strongly connected components cost $O(n + m)$, or $O((n + m)k)$ per round. Every merging round loses at least one class, so there are at most $n - 1$ of them, plus one final round.

Blocked attacks: deletion or inversion. We read a suppressed attack as deleted. The companion paper [3] reads it as inverted, in the style of the critical-attack repair of [11]. Deletion is the reading under which the VAF question has content: an audience's defeats always lie inside R , whereas inversion can produce a defeat outside R , and does so exactly when a one-directional attack is blocked. Representability under inversion therefore requires (S): $\delta(c, (a, b)) = 1$ whenever $(a, b) \in R$ and $(b, a) \notin R$. Given (S) the two readings agree except on mutual pairs blocked in both directions, where inversion supplies both edges and deletion neither. Flipping exactly those pairs to active gives a defeat function δ^* , computable in $O(mk)$, and a CDAF is representable under inversion iff (S) holds and $\langle \mathcal{A}, R, \mathcal{C}, \delta^* \rangle$ is representable under deletion. Theorem 1 and Algorithm 1 therefore transfer after one linear-time pass, with Theorem 2 unchanged. Condition (S) is itself a separation: under inversion, VAF-representability forces all blocking onto mutual attacks.

3.2. A worked trace

We trace Algorithm 1 on an instance built so that one run exercises every phase: a Phase 1 merge, a Phase 3 cycle merge, and a Phase 2 rejection.

Let $\mathcal{A} = \{w, x, y, z\}$ and $\mathcal{C} = \{c_1, c_2\}$, with

$$R = \{(w, x), (x, w), (x, y), (y, z), (z, x)\}.$$

The graph is a mutual pair $\{w, x\}$ plus a directed cycle $x \rightarrow y \rightarrow z \rightarrow x$; c_1 activates every attack and c_2 suppresses only (x, y) , as Figure 1 shows.

1. *Phase 1.* The only mutual pair is $\{w, x\}$. Both (w, x) and (x, w) succeed in both contexts, so w and x merge. Now $\mathcal{P} = \{\{w, x\}, \{y\}, \{z\}\}$.
2. *Phase 2.* The intra-class attacks are (w, x) and (x, w) , and both succeed in every context. Pass.
3. *Phase 3, building the relations.* On the classes $[wx], [y], [z]$. In c_1 every attack succeeds, so the first clause yields $[wx] \triangleright_{c_1} [y]$, $[y] \triangleright_{c_1} [z]$, and $[z] \triangleright_{c_1} [wx]$: a cycle, one non-trivial strongly connected component. In c_2 the attack (x, y) fails, so the second clause yields $[y] \triangleright_{c_2} [wx]$; with $[y] \triangleright_{c_2} [z]$

Table 1

Decision problems for a CDAF with n arguments, m attacks, and k contexts. Cross-context bounds follow from single-framework classifications [12] with a factor k for iterating over contexts; the corresponding audience-dependent questions for VAFs are classified by [13].

Problem	Semantics	Bound	Status
Cross-context acceptance (all classes)	grounded	in P	closed
Universal acceptance	preferred	Π_2^P -complete	closed
Indefensibility	preferred	coNP-complete	closed
Audience inducibility of $R' \subseteq R$	(defeat relation)	$O(n + m)$	closed
VAF-representability	(defeat function)	$O(n^2k + nmk)$	closed
Single-context realisability	grounded	in NP	lower bound open
Single-context realisability	preferred	in Σ_3^P	lower bound open
Family realisability	any	decidable	criterion open

and $[z] \triangleright_{c_2} [wx]$ from the first clause, $\triangleright_{c_2}$ is acyclic. So c_2 forces preferences but no merge; the cycle lives in $\triangleright_{c_1}$ alone.

4. *Phase 3, merging.* The c_1 component merges all three classes: $\mathcal{P} = \{\{w, x, y, z\}\}$. Back to Phase 2.
5. *Phase 2, second pass.* Every attack is now intra-class. The check reaches (x, y) and finds $\delta(c_2, (x, y)) = 0$. The algorithm returns *not representable*, at its only failure point.

The rejection tracks a semantic impossibility that can be stated without the algorithm. Suppose a value-consistent partition existed. In c_1 all three attacks of the cycle $x \rightarrow y \rightarrow z \rightarrow x$ succeed. If x, y, z did not all share one class, the distinct classes among them would carry a $\triangleright_{c_1}$ cycle, violating (A). So $[x] = [y] = [z]$. But then (x, y) is an intra-class attack, and $\delta(c_2, (x, y)) = 0$ violates (I).

Neither context is pathological alone. Each is representable in isolation: c_1 by one value for every argument, c_2 by $val(w) = val(x) = v$ with y and z given fresh values ordered $val(y) \succ val(z) \succ v$. Non-representability is a coupling phenomenon: it appears only when one value assignment must serve both contexts.

4. The Complexity Landscape

Table 1 collects the decision problems that a CDAF generates and what is known about them.

Single-audience inducibility. The basic building block, reused by the implementation of Section 5, is a linear-time test.

Proposition 1 (mixed-graph inducibility test). *Given $\langle \mathcal{A}, R \rangle$ and $R' \subseteq R$, deciding whether $R' = R_p$ for some audience p of some VAF over $\langle \mathcal{A}, R \rangle$ takes $O(n + m)$ time.*

Build a mixed graph H on the arguments: a weak edge $b \rightarrow a$ for each $(a, b) \in R'$, and a strict edge $a \rightarrow b$ for each $(a, b) \in R \setminus R'$. Read an edge $u \rightarrow v$ as “the rank of u is at most the rank of v ”, strictly so for strict edges. The constraint system is satisfiable iff H has no directed cycle through a strict edge: contract the strongly connected components and reject iff some strict edge lies inside one.

Single-context realisability. Realisability weakens representability: the VAF only has to reproduce the extensions. For a context c and semantics σ , the question is whether some audience p satisfies $\sigma(\langle \mathcal{A}, R_p \rangle) = \sigma(AF_c)$. Guess and check gives the upper bounds of Table 1, neither known to be tight. The natural criterion for the grounded case fails: acyclicity of the latent attacks inside the target extension is necessary but not sufficient, and the smallest counterexamples, found by exhaustive enumeration up to four arguments, have three arguments and four attacks. Grounded and preferred

realisability also come apart: some instances are realisable under grounded but not under preferred semantics.

The strategic variant is the subject of [3]: choosing which perspectives are active selects which attacks succeed, and deciding whether some choice makes a target argument credulously accepted is NP-complete under preferred and stable semantics.

5. A Reference Implementation and a Measurement

We implemented Algorithm 1 and the inducibility test of Proposition 1 as a Python script.¹ The tool reads a small extension of the apx format used in ICCMA [16, 17, 18, 19]: a context block declares the contexts and lists, per context, the suppressed attacks. The worked instance of Section 3.2 reads:

```
arg(w). arg(x). arg(y). arg(z).
att(w,x). att(x,w). att(x,y). att(y,z). att(z,x).
context(c1).
context(c2). off(c2,x,y).
```

Every attack is operative unless switched off, so a plain apx file is a CDAF with one context, and projecting to one context yields a plain apx instance on which any ICCMA solver runs unchanged; cross-context queries are thin orchestration over k solver calls.

Correctness is machine-checked against the definitions. One brute-force oracle enumerates all partitions of $\mathcal{A}$ and checks the four conditions of Definition 2 literally; a second decides inducibility by trying every ranking. The implementation agrees with the oracles on all 547,795 instances with three arguments and up to three contexts, on 180,000 uniformly sampled instances with four and five arguments, and on all 532,180 inducibility pairs with up to four arguments. A third oracle decides representability directly from the defeat rule, trying every value assignment and every per-context strict total order; it agrees on all 547,795 three-argument instances, and since it does not use the criterion, that agreement checks Theorem 1 itself, not only the algorithm. The returned partition is checked value-consistent and finest on every representable instance, and the worked trace is asserted event by event.

On VAF-generated instances, representable by construction and so necessarily accepted, the reference code handles 10^3 arguments, 10^5 attacks, and 20 contexts in five seconds; a mask-based variant in the release handles 10^4 arguments, 2×10^6 attacks, and 20 contexts in under three minutes.

5.1. How common is representability under uniform sampling?

The same machinery yields a measurement: the fraction of small CDAFs that is VAF-representable. Table 2 gives it for labeled instances without self-attacks. The generator is uniform over that space: each ordered pair of distinct arguments independently takes one of $1 + 2^k$ equiprobable states, either absent from R or present with one of the 2^k activation patterns over the contexts. So R and δ are drawn in one step, not independently, and every attack is equally likely to carry any pattern. This is a null model, not a model of applications.

Representability collapses in both directions, from 55 percent at $n = 3, k = 1$ to under one instance in ten thousand at $n = 5, k = 2$. Beyond $n = 5$ it falls under the detection limit of uniform sampling, so larger instances would add empty bounds, not measurements. Two readings. First, audience structure is a strong constraint: generic context-dependence is almost never reducible to value orderings. The scope of that claim is the null model above; structured instances from applications, such as attack graphs extracted from text by argument-mining pipelines [20], are not covered by these numbers. Second, instances satisfying three of the four conditions of Definition 2 are too rare to sample and must

¹We release the implementation, the validation suite, and the enumeration data at <https://github.com/albsadowski/cdaf>, archived at <https://doi.org/10.5281/zenodo.21320967>. Small complete instance spaces support conjecture testing, as in the realisability and signature programme [14, 15]; the grounded counterexample of Section 4 came from exactly such an enumeration.

Table 2

Percentage of VAF-representable CDAFs among labeled instances without self-attacks, by arguments n and contexts k . The cells with $n = 3, k \leq 3$ are exact (exhaustive enumeration); the others are estimates from 2×10^5 uniform samples per cell, all 95% confidence intervals below 0.2 percentage points. Cells with no representable sample are bounded above by 0.002 percent.

	$k = 1$	$k = 2$	$k = 3$	$k = 4$
$n = 3$	54.9	10.5	1.51	0.18
$n = 4$	20.3	0.52	0.009	< 0.002
$n = 5$	4.1	0.003	< 0.002	< 0.002

be constructed; the enumeration data locates them. Realisability labels, by contrast, are not free at generation time: they need extension-set comparisons, at the second level of the polynomial hierarchy for preferred semantics, which current competition tracks do not exercise.

6. Open Problems

The main structural question is family-level realisability, where one value assignment couples all contexts: a criterion in the style of Theorem 1, or any upper bound better than brute force. For the grounded single-context case, acyclicity of latent attacks inside the target extension is necessary but not sufficient, and a characterisation would settle whether the NP bound can be lowered; more broadly, the realisability upper bounds have no matching lower bounds. Whether the perspective-labeled fragment of [3], which derives δ from perspective activation and priorities, admits faster algorithms is unexplored. A separate question is where contexts come from: they are external parameters here, whereas dialogue supplies them endogenously, a protocol stage or a shift in the burden of proof being a natural generator of δ . Whether dialogue protocols characterise the CDAF families that arise this way is open. Finally, cross-context queries materialise k frameworks and run a solver per context; declarative encodings working on δ directly, in the spirit of the ASPIC+ encodings of Lehtonen et al. [21], would avoid that and are a natural next step for this community.

7. Conclusion

A CDAF is one attack structure under many contexts. VAF-representability is solved by a polynomial-time partition-refinement algorithm, validated against the definitions on over half a million small instances, while the surrounding realisability problems have upper bounds from NP to Σ_3^P and no matching lower bounds. The measurement shows where the algorithmic interest lies: under uniform sampling representability is thin and thins with every added context, so cross-context reasoning cannot in general be delegated to VAF machinery. The open problems of Section 6 mark where new criteria and encodings are needed; the released code and data are a starting point.

Declaration on Generative AI

During the preparation of this work, the authors used Claude (Anthropic) in order to: draft content, check grammar and spelling, and assist with writing the reference implementation. After using this tool, the authors reviewed and edited the content as needed and take full responsibility for the publication’s content.

References

- [1] 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. doi:10.1016/0004-3702(94)00041-X.
- [2] T. J. M. Bench-Capon, Persuasion in practical argument using value-based argumentation frameworks, *Journal of Logic and Computation* 13 (2003) 429–448. doi:10.1093/logcom/13.3.429.
- [3] A. Sadowski, J. A. Chudziak, Choosing the lens: Strategic perspective activation in context-dependent argumentation, 2026. URL: <https://arxiv.org/abs/2605.31581>. arXiv:2605.31581, accepted at LAMAS&SR 2026.
- [4] G. Brewka, T. Eiter, Argumentation context systems: A framework for abstract group argumentation, in: *Proc. 10th Int. Conf. on Logic Programming and Nonmonotonic Reasoning (LPNMR)*, Springer, 2009, pp. 44–57. doi:10.1007/978-3-642-04238-6_7.
- [5] Z. Yu, S. Ju, W. Chen, Context-based argumentation frameworks and multi-agent consensus building, *Journal of Logic and Computation* 34 (2023) 199–228. doi:10.1093/logcom/exad063.
- [6] S. Modgil, Reasoning about preferences in argumentation frameworks, *Artificial Intelligence* 173 (2009) 901–934. doi:10.1016/j.artint.2009.02.001.
- [7] D. Baumeister, M. Järvisalo, D. Neugebauer, A. Niskanen, J. Rothe, Acceptance in incomplete argumentation frameworks, *Artificial Intelligence* 295 (2021) 103470. doi:10.1016/j.artint.2021.103470.
- [8] R. Booth, S. Kaci, T. Rienstra, Property-based preferences in abstract argumentation, in: P. Perny, M. Pirlot, A. Tsoukiàs (Eds.), *Algorithmic Decision Theory*, Springer Berlin Heidelberg, Berlin, Heidelberg, 2013, pp. 86–100.
- [9] P. Baroni, M. W. A. Caminada, M. Giacomin, Abstract argumentation frameworks and their semantics, in: L. van der Torre, S. Villata, D. M. Gabbay (Eds.), *Handbook of Formal Argumentation*, College Publications, 2018, pp. 159–236.
- [10] E. Szpilrajn, Sur l’extension de l’ordre partiel, *Fundamenta Mathematicae* 16 (1930) 386–389. doi:10.4064/fm-16-1-386-389.
- [11] L. Amgoud, S. Vesic, Rich preference-based argumentation frameworks, *International Journal of Approximate Reasoning* 55 (2014) 585–606. doi:10.1016/j.ijar.2013.10.010.
- [12] W. Dvořák, P. E. Dunne, Computational problems in formal argumentation and their complexity, in: P. Baroni, D. M. Gabbay, M. Giacomin, L. van der Torre (Eds.), *Handbook of Formal Argumentation*, College Publications, 2018, pp. 631–687.
- [13] T. J. M. Bench-Capon, S. Doutre, P. E. Dunne, Audiences in argumentation frameworks, *Artificial Intelligence* 171 (2007) 42–71. doi:10.1016/j.artint.2006.10.013.
- [14] P. E. Dunne, W. Dvořák, T. Linsbichler, S. Woltran, Characteristics of multiple viewpoints in abstract argumentation, *Artificial Intelligence* 228 (2015) 153–178. doi:10.1016/j.artint.2015.07.006.
- [15] T. Linsbichler, J. Pührer, H. Strass, A uniform account of realizability in abstract argumentation, in: *Proc. 22nd European Conf. on Artificial Intelligence (ECAI)*, IOS Press, 2016, pp. 252–260. doi:10.3233/978-1-61499-672-9-252.
- [16] M. Thimm, S. Villata, The first international competition on computational models of argumentation: Results and analysis, *Artificial Intelligence* 252 (2017) 267–294. doi:10.1016/j.artint.2017.08.006.
- [17] S. A. Gaggl, T. Linsbichler, M. Maratea, S. Woltran, Design and results of the second international competition on computational models of argumentation, *Artificial Intelligence* 279 (2020) 103193. doi:10.1016/j.artint.2019.103193.
- [18] S. Bistarelli, L. Kotthoff, J.-M. Lagniez, E. Lonca, J.-G. Mailly, J. Rossit, F. Santini, C. Taticchi, The third and fourth international competitions on computational models of argumentation: Design, results and analysis, *Argument and Computation* 16 (2024) 236 – 299. doi:10.3233/aac-230013.
- [19] M. Järvisalo, T. Lehtonen, A. Niskanen, ICCMA 2023: 5th international competition on computational models of argumentation, *Artificial Intelligence* 342 (2025) 104311. doi:10.1016/j.

artint.2025.104311.

- [20] J. Bąba, J. A. Chudziak, From argument components to graphs: A multi-agent debate with confidence gating for argument relations, 2026. URL: <https://arxiv.org/abs/2606.16047>. arXiv:2606.16047.
- [21] T. Lehtonen, D. Odekerken, J. P. Wallner, M. Järvisalo, Complexity results and algorithms for preferential argumentative reasoning in ASPIC+, in: Proc. 21st Int. Conf. on Principles of Knowledge Representation and Reasoning (KR), 2024. doi:10.24963/kr.2024/49.