Computational Argumentation: Reasoning, Dynamics, and Supporting Explainability

Johannes P. Wallner
Institute of Software Technology, Graz University of Technology, Austria
wallner@ist.tugraz.at

Abstract

This overview accompanies the author's Early Career Track presentation. We survey recent research and research agenda of the author, focusing on contributions in the area of computational argumentation. Contributions span from foundations of static and dynamic forms of argumentative reasoning and approaches to support explainability, e.g., analysis of the computational complexity of argumentative reasoning and algorithmic approaches.

1 Introduction

Computational argumentation provides foundational approaches how to represent and reason with arguments within the broader field of Artificial Intelligence (AI) [Baroni *et al.*, 2018; Gabbay *et al.*, 2021]. Nowadays, computational argumentation finds heterogeneous application areas such as in legal reasoning and medical reasoning [Atkinson *et al.*, 2017], e.g., in the form of supporting fraud detection at the Dutch National Police [Odekerken *et al.*, 2020].

Among the aims of computational argumentation are supporting decision-making, explainability, and automated reasoning. Central to these are formal approaches to argumentative reasoning, which are oftentimes classified into structured [Besnard *et al.*, 2014] and abstract argumentation [Dung, 1995]. The former prescribes formalized workflows [Caminada and Amgoud, 2007], or processes, that specify how to carry out argumentation (Figure 1). Conceptually, from a conflicting and possibly incomplete knowledge base arguments are constructed, which can, e.g., be derivations based on defeasible rules and assumptions. Relations connect arguments together, with an attacking relation being the most prominent one, yet several other notions, e.g., support relations, have been considered as well [Brewka *et al.*, 2014]. Abstracting the content of arguments and looking only at the abstracted arguments and their relations is the topic of abstract argumentation. In this field several argumentation semantics exist [Baroni *et al.*, 2011], which are criteria specifying which (sets of) arguments can be deemed acceptable together. As an example, a basic ingredient of checking whether an argument can be accepted is that of being defended by an admissible set of arguments, which is a conflict-free set of arguments that defends itself against all counter-arguments.

Reflecting the heterogeneity of argumentation, many argumentation formalisms are studied in the field, including general-purpose structured argumentation formalisms such as assumption-based argumentation (ABA) [Bondarenko *et al.*, 1997], ASPIC$^+$ [Modgil and Prakken, 2013], defeasible logic programming (DeLP) [García and Simari, 2004], deductive argumentation [Besnard and Hunter, 2008], Gorgias [Kakas *et al.*, 2019], and Carneades [Gordon *et al.*, 2007]. For abstract argumentation, argumentation frameworks (AFs) [Dung, 1995] are the most prominent formal approach, which were extended in various ways, e.g., allowing general forms of specifying acceptance conditions of arguments in abstract dialectical frameworks (ADFs) [Brewka and Woltran, 2010; Brewka *et al.*, 2018].

All of these approaches, structured and abstract, share a high computational complexity of argumentative reasoning: even checking whether there is an admissible set defending a queried argument is NP-hard, and many other reasoning tasks have higher complexity, e.g., are hard for a class in the polynomial hierarchy [Dvořák and Dunne, 2018].

In this paper I give an overview of our research on the topic of computational argumentation, focusing in particular on the research agenda of furthering various forms of argumentative reasoning both from foundational aspects as well as regarding algorithms and system tools. Naturally, the research outlined in this paper involves several researchers and research groups.

2 Reasoning in Argumentation Formalisms

Towards algorithmic solutions for reasoning in computational argumentation, in particular for more advanced forms of argumentative reasoning, a thorough understanding of the complexity of the tasks to solve is essential. Since even basic reasoning tasks in formal argumentation are NP-hard and several forms of reasoning in this field are hard or complete for a class in the polynomial hierarchy, algorithm design can significantly benefit from understanding the underlying complexity. Based on state-of-the-art search algorithms for NP-complete problems, such as SAT(isfiability) solving [Biere *et al.*, 2021] and answer set programming (ASP) [Gelfond and Lifschitz, 1988; Niemelä, 1999], algorithm design can be tailored to usage of, e.g., SAT solvers, as required from the complexity of the task to solve. For instance, problems complete for Σ_2^P are,

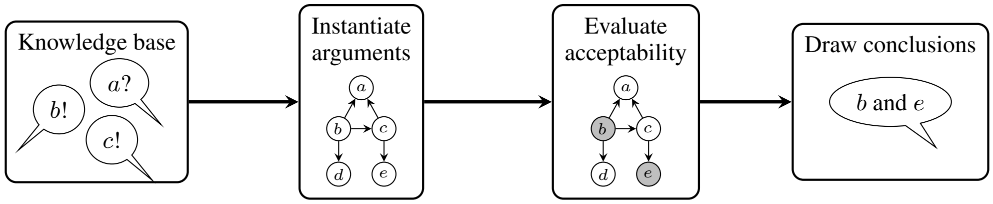

Figure 1: The argumentation process [Caminada and Amgoud, 2007].

presumably, not poly-time reducable to one SAT call, yet can be solved, e.g., via interaction of several SAT sub procedures.

We contributed to understanding the complexity of various reasoning tasks in computational argumentation. For ADFs that allow for flexible specification of acceptance conditions for arguments, with applications in legal reasoning [Al-Abdulkarim *et al.*, 2019], we showed fundamental complexity results for general ADFs [Strass and Wallner, 2015] and for restricted ADFs [Linsbichler *et al.*, 2022]. Complexity of reasoning in general ADFs goes up to the third level of the polynomial hierarchy, and remains NP (or coNP) hard even for computationally simple argumentation semantics, such as the grounded semantics. In brief, these results imply that the complexity of reasoning in ADFs is "one level higher" in the polynomial hierarchy than reasoning in AFs, for corresponding reasoning tasks. AFs feature the fundamental notion of directed attacks, while the additional expressivity of general ADFs comes at a higher computational cost.

To overcome the complexity barrier, we investigated several restrictions to ADFs. In so-called bipolar ADFs [Brewka and Woltran, 2010], acceptance conditions of each argument in an ADF are restricted, intuitively representing only certain forms of attacks or supports between arguments. If it is known which arguments are attacking or supporting, the complexity of bipolar ADFs drops to the level of AFs, thus leading to a fragment of ADFs with additional representational expressivity [Strass, 2015], yet without higher complexity.

Going into other directions, we considered restrictions to the underlying graph structure of ADFs, by considering acyclic, bipartite, and symmetric graph structures, as well as certain distance notions to these graph structures (e.g., removal of some arguments suffices to have an acyclic ADF). While bipartiteness and symmetry does not appear to support milder forms of reasoning, acyclicity does indeed lead to milder complexity: many reasoning tasks are decidable in polynomial time. Yet even with low distances to acyclicity reasoning is not poly-time decidable.

Algorithm design for ADFs can in particular be utilized by bipolarity and further semantical restrictions that allow only a bounded number of interpretations satisfying criteria of a chosen semantics. Using state-of-the-art SAT solving, algorithms for general ADFs can make use of these complexity results to restrict usage of SAT solvers, if possible.

In the area of structured argumentation investigations of algorithmic solutions and complexity analysis face an intrinsic barrier: arguments are not explicitly given, but need to be constructed. For instance, in Figure 2 the atoms a, b, and c may be assumed and use derivations by rules $x_1, \ldots, x_n \rightarrow h$, denoted by edges in the figure, e.g., x can be derived from b in this example. Solid edges and nodes denote "strict components" that cannot be attacked, while dashed components denote defeasible reasoning that may be invalidated by conflicts. Each derivation is then an argument, e.g., arguments A_1 to A_7 in the figure.

In general structured argumentation, the number of arguments is not bounded and can be infinite [Lehtonen *et al.*, 2023]. While in many forms of structured argumentation it is direct to restrict construction to a finite set of arguments, an exponential number of arguments can still be generated [Strass *et al.*, 2019]. Many instantiation procedures give rise to exponentially many arguments. We showed that, under certain complexity assumptions from compilation theory, an exponential number appears unavoidable in ABA [Lehtonen *et al.*, 2024]. When imposing certain restrictions, e.g., allowing no assumptions to be derivable for ABA frameworks, a polynomially sized "core" can be constructed which suffices to perform reasoning [Lehtonen *et al.*, 2023].

Going into a different direction, we also investigated complexity and algorithms when working on a compact representation of arguments, without explication (construction) of arguments. For the case of ABA such a compact representation was developed earlier [Cyras *et al.*, 2018], while for ASPIC$^+$ a novel characterization or representation was required to be developed. These representations involve, in brief terms, specifying which part of a given knowledge base corresponds to, e.g., an admissible set of arguments. Then computation can focus on "selecting" parts of the knowledge base and checking whether a correspondence holds.

Together with earlier results, compact representations allowed to analyze complexity of argumentative reasoning in ABA and ASPIC$^+$, without requiring construction of (potentially exponentially many) arguments. The results indicate that complexity in many cases is the same as for AFs, i.e., complexity results give here a clear indication that exponentially many arguments are not needed to be constructed for reasoning, although at the potential disadvantage of not having explicit arguments to show to a user. For the case of

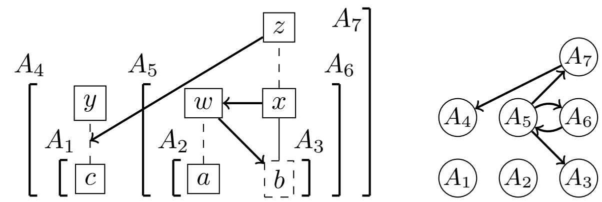

Figure 2: Example rule base in ASPIC$^+$.

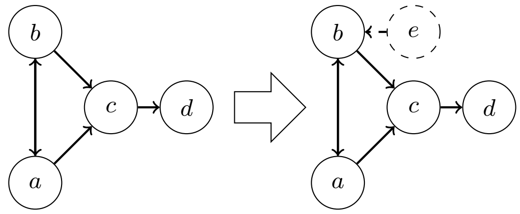

Figure 3: Enforcing d to be accepted under grounded semantics.

preferential argumentative reasoning, which is seen as key to computational argumentation, complexity rises again under certain chosen types of preferences [Lehtonen *et al.*, 2021; Lehtonen *et al.*, 2022].

Algorithms we developed for both ABA and ASPIC$^+$ rely on SAT and ASP solving [Lehtonen *et al.*, 2021; Lehtonen *et al.*, 2020; Lehtonen *et al.*, 2023]. In experiments, algorithms operating on compact representations of arguments were shown to be the current state of the art in terms of runtime performance. However, having explicit arguments turned out to be beneficial in cases, in particular those with certain language features that lead to more involved checks in compact representations, while on explicated arguments are more direct to compute [Lehtonen *et al.*, 2024].

For the area of solving reasoning tasks on different argumentation formalisms, we think more research is needed. We believe that in particular (i) preferential reasoning, (ii) optimized argument construction procedures, and (iii) extending language features that can be solved are promising directions for future research.

3 Dynamics in Argumentative Reasoning

Argumentative reasoning is, naturally, not always static, in the sense that all arguments (or their contents) are known before-hand. What is sometimes called dynamics in computational argumentation refers to situations of dynamically evolving argumentation scenarios. Many such forms are studied in the literature [Doutre and Mailly, 2018].

We contributed to the field of dynamics in computational argumentation by advancing understanding of various forms of dynamic reasoning. One fundamental dynamic operation is called enforcement [Baumann *et al.*, 2021], which aims to find arguments to expand (or remove) in a current situation such that a desired argument can be found acceptable. That is, enforcement aims to "enforce" acceptance of a queried argument, even though the current situation does not warrant acceptance. For instance, in Figure 3, the AF on the left does not include argument d in the grounded extension, which is among the most cautious semantics of AFs. Here unattacked arguments are acceptable, as well as all arguments iteratively defended by the unattacked arguments. By expanding with argument e (right in the figure), an unattacked argument defends a, and, in turn, d is defended. The enforcement problem can be phrased as an optimization problem: the changes (e.g., additions of arguments) may be required to be minimal. An application is then which arguments to pose in a dialogue in order to advance acceptance of certain arguments.

We showed complexity in several settings of enforcement of arguments in abstract and structured argumentation [Wallner *et al.*, 2017] and developed algorithms for finding enforcements. The optimization nature of enforcement in particular suggests to use solvers for NP hard problems including optimization, such as Maximum SAT (MaxSAT) solvers [Bacchus *et al.*, 2021] and ASP solvers. Complexity of enforcement on AFs (usually) inherits complexity of static reasoning, as one has to first check whether an argument is already accepted without modifications. Nevertheless, this form of dynamic reasoning oftentimes leads to higher complexity, e.g., the poly-time grounded semantics leads to NP-complete problems in the enforcement setting.

Approaches that include learning in computational argumentation recently gained more and more traction in the research community [Proietti and Toni, 2023]. We considered the AF Synthesis problem [Niskanen *et al.*, 2019] that provides an automated way of constructing attacks between arguments. The AF Synthesis problem can be seen as an inverse task than "normal" reasoning: based on semantical examples what are, e.g., admissible sets, which attacks have to be present? In the AF Synthesis problem, the task is to synthesize, learn, or generate an AF based on such semantical examples. We allow for noisy and incomplete examples, in the sense that relative trust of examples can be specified and the resulting AF should be of optimal cost (violation of an example incurs its weight as a cost). The AF Synthesis problem is connected to so-called realizability in formal argumentation [Dunne *et al.*, 2015] which studies existence questions like "does an AF exist having certain semantical properties?" Similar to enforcement, MaxSAT solvers can be utilized to solve the AF Synthesis problem.

In the application by the Dutch National Police [Odekerken *et al.*, 2020] a different kind of dynamic scenario is important: stability of an argumentative decision. For instance, if the current state of argumentation draws a certain conclusion, can the conclusion change if further arguments are added? That is, can one expand with further arguments and enforce another outcome? In case the answer is negative, the current outcome is seen as stable. We supported the research on the computational problem of stability, which is phrased in the ASPIC$^+$ framework [Odekerken *et al.*, 2023]. Here, as well, as for static forms of reasoning, compact representations were key to both complexity analysis and algorithms, with the current state of the art relying on ASP solving.

In case a current setting is not stable, i.e., a different outcome is still possible, the computational problem of relevance aims to find open issues that can lead to a stable conclusion. That is, which propositions are relevant in changing an outcome in the sense that after their addition the result is stable. For the grounded semantics, stability and relevance raise the computational complexity by one and two steps, respectively, compared to static forms of reasoning.

Summarizing, computational approaches to dynamic forms of argumentation oftentimes are more involved than more static forms of reasoning. Tasks like enforcement can be seen as basic operations that are part of more complex settings, such as in incomplete scenarios. More advanced language features, e.g., using preferential information, requires more research in the dynamic setting, as well, in our opinion.

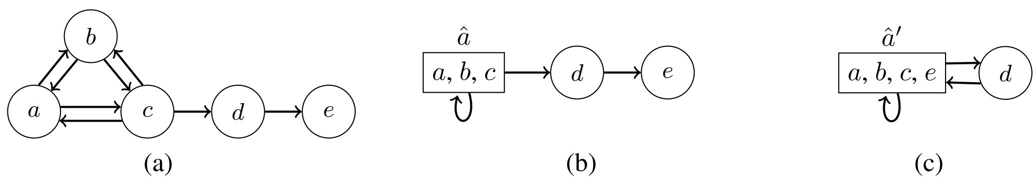

Figure 4: An example AF (a), a faithful clustered AF (b), and a clustered AF with a spurious admissible set (c).

4 Supporting Argumentative Explainability

Computational argumentation lends itself naturally to forms supporting explainability, e.g., an admissible set of arguments contains (in the structure of arguments) a clear indication why a certain conclusion can be (argumentatively) defended. More generally, argumentation in AI is seen as a promising approach to explainability in AI (XAI) [Cyras *et al.*, 2021; Vassiliades *et al.*, 2021].

Nevertheless, arguments, while individually direct to interpret, may face barriers, nevertheless. For instance, as briefly hinted above, the number of arguments may be high (e.g., exponential-sized w.r.t. a given rule-base), the arguments themselves might be complex derivations, and argumentation semantics might not always give clear indications of a result. For instance, checking whether an admissible set (in an AF) exists containing a queried argument is NP-complete. Thus, checking if no such set exists is coNP-complete, a complexity class whose witnesses for being a positive result evades polynomial-size, under certain complexity assumptions. Thus, a witness for an argument being in no admissible set is not expected to be small in general.

We investigated how to cope with high number of arguments, approaches to support "no" answers, and supporting explainability in general.

Strongly accepting or rejecting sub frameworks F', of a given AF F, are parts of the AF such that any AF "in-between" F and F' has the same outcome [Saribatur *et al.*, 2020; Ulbricht and Wallner, 2021]. Say, there is no admissible set containing a queried argument in AF F, then a sub framework F' of F is strongly rejecting the same argument (under the same semantics) if each F'' which contains all parts of F' and may expand up to F finds the same answer. In this way, strongly accepting or rejecting sub frameworks can find small parts of the given framework that suffice to draw a certain conclusion. This behavior is connected to both stability (see above) and incomplete AFs [Baumeister *et al.*, 2018], with a main difference is that with strongly accepting or rejecting sub frameworks an aim is to find small (minimal, optimal) sub frameworks. Complexity of reasoning under these strongly accepting or rejecting sub frameworks, both in abstract and structured argumentation [Ulbricht and Wallner, 2022], is more involved than static forms of reasoning.

While strongly accepting or rejecting sub frameworks may focus reasoning on a part of a given AF (or structured rule base), they are restricted to parts of the given input. In contrast, forms of abstraction, as investigated, e.g., in model checking [Clarke *et al.*, 2003], allow for abstracting parts of the input, while aiming to being faithful to original argumentative outcomes. We considered forms of existential abstraction both on AFs [Saribatur and Wallner, 2021] and in ABA [Apostolakis *et al.*, 2024], by clustering parts of the input. For instance, in Figure 4, in a given AF (a) the left-most three arguments are clustered into one "clustered argument" (Figure 4(b)). By adapting classical argumentation semantics to the clustered setting, in the AF in the middle, the same argumentative conclusions can be drawn, i.e., this clustering is faithful, while the clustering on the right leads to an incorrect conclusion. The latter is deemed spurious, due to faulty reasoning introduced in the abstraction. Abstraction is here "existential" in the sense that attacks in the clustering are present if there exists an attack in the AF from or to the clusters.

An advantage of such clusterings is that they can significantly reduce the set of arguments, but instead of removal, arguments are abstracted into clusters. For instance, in the faithful clustering (Figure 4(b)) argument e can be defended against the attack originating form d, by using the cluster. But which argument is responsible for defense was abstracted away. Abstraction via clustering can also be used interactively: a user may choose more or less abstraction (i.e., "zooming" in or out).

While many research works are strengthening the potential, we believe that more research is needed to fully realize the explainability potential of computational argumentation. In addition to the aforementioned form of abstraction, general simplification procedures are, we believe, useful for users. For instance, users may interactively gain insights into why certain (argumentative) conclusions hold. For this, users might need simplifications in order to digest main components of argumentative reasoning.

5 Conclusions

We gave an overview of the author's contributions to computational argumentation, going from foundational complexity questions to algorithmic solutions. Static forms, in which the current state does not change, and dynamic forms, ranging from finding usable arguments to learning arguments, were studied.

As summarized in the above sections, future research is, in our opinion, needed to further potential of computation argumentation. Among the promising directions are support for more advanced language features in the formal languages, both conceptually and from an algorithmic perspective. Moreover, support for explainability, in its many forms, is a useful research direction. We think in particular interactions with users is a direction with potential for advancing applications.

Acknowledgments

The author is supported by the Austrian Science Fund (FWF) P35632. The author thanks all co-authors for their insightful and continuous cooperations.

References

[Al-Abdulkarim *et al.*, 2019] Latifa Al-Abdulkarim, Katie Atkinson, Trevor J. M. Bench-Capon, Stuart Whittle, Rob Williams, and Catriona Wolfenden. Noise induced hearing loss: Building an application using the ANGELIC methodology. *Argument Comput.*, 10(1):5–22, 2019.

[Apostolakis *et al.*, 2024] Iosif Apostolakis, Zeynep G. Saribatur, and Johannes P. Wallner. Abstracting assumptions in structured argumentation. In *Proc. AAMAS*, pages 2132–2134. ACM, 2024.

[Atkinson *et al.*, 2017] Katie Atkinson, Pietro Baroni, Massimiliano Giacomin, Anthony Hunter, Henry Prakken, Chris Reed, Guillermo Ricardo Simari, Matthias Thimm, and Serena Villata. Towards artificial argumentation. *AI Mag.*, 38(3):25–36, 2017.

[Bacchus *et al.*, 2021] Fahiem Bacchus, Matti Järvisalo, and Ruben Martins. Maximum satisfiabiliy. In Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors, *Handbook of Satisfiability - Second Edition*, volume 336 of *FAIA*, pages 929–991. IOS Press, 2021.

[Baroni *et al.*, 2011] Pietro Baroni, Martin Caminada, and Massimiliano Giacomin. An introduction to argumentation semantics. *Knowl. Eng. Rev.*, 26(4):365–410, 2011.

[Baroni *et al.*, 2018] Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors. *Handbook of Formal Argumentation*. College Publications, 2018.

[Baumann *et al.*, 2021] Ringo Baumann, Sylvie Doutre, Jean-Guy Mailly, and Johannes P. Wallner. Enforcement in formal argumentation. In Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors, *Handbook of Formal Argumentation*, volume 2, chapter 8, pages 445–510. College Publications, 2021.

[Baumeister *et al.*, 2018] Dorothea Baumeister, Daniel Neugebauer, Jörg Rothe, and Hilmar Schadrack. Verification in incomplete argumentation frameworks. *Artif. Intell.*, 264:1–26, 2018.

[Besnard and Hunter, 2008] Philippe Besnard and Anthony Hunter. *Elements of Argumentation*. MIT Press, 2008.

[Besnard *et al.*, 2014] Philippe Besnard, Alejandro Javier García, Anthony Hunter, Sanjay Modgil, Henry Prakken, Guillermo Ricardo Simari, and Francesca Toni. Introduction to structured argumentation. *Argument Comput.*, 5(1):1–4, 2014.

[Biere *et al.*, 2021] Armin Biere, Marijn Heule, Hans van Maaren, and Toby Walsh, editors. *Handbook of Satisfiability - Second Edition*, volume 336 of *FAIA*. IOS Press, 2021.

[Bondarenko *et al.*, 1997] Andrei Bondarenko, Phan Minh Dung, Robert A. Kowalski, and Francesca Toni. An abstract, argumentation-theoretic approach to default reasoning. *Artif. Intell.*, 93:63–101, 1997.

[Brewka and Woltran, 2010] Gerhard Brewka and Stefan Woltran. Abstract dialectical frameworks. In *Proc. KR*, pages 102–111. AAAI Press, 2010.

[Brewka *et al.*, 2014] Gerhard Brewka, Sylwia Polberg, and Stefan Woltran. Generalizations of dung frameworks and their role in formal argumentation. *IEEE Intell. Syst.*, 29(1):30–38, 2014.

[Brewka *et al.*, 2018] Gerhard Brewka, Stefan Ellmauthaler, Hannes Strass, Johannes P. Wallner, and Stefan Woltran. Abstract dialectical frameworks. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 5, pages 237–285. College Publications, 2018.

[Caminada and Amgoud, 2007] Martin Caminada and Leila Amgoud. On the evaluation of argumentation formalisms. *Artif. Intell.*, 171(5-6):286–310, 2007.

[Clarke *et al.*, 2003] Edmund M. Clarke, Orna Grumberg, Somesh Jha, Yuan Lu, and Helmut Veith. Counterexample-guided abstraction refinement for symbolic model checking. *J. ACM*, 50(5):752–794, 2003.

[Cyras *et al.*, 2018] Kristijonas Cyras, Xiuyi Fan, Claudia Schulz, and Francesca Toni. Assumption-based argumentation: Disputes, explanations, preferences. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 7, pages 365–408. College Publications, 2018.

[Cyras *et al.*, 2021] Kristijonas Cyras, Antonio Rago, Emanuele Albini, Pietro Baroni, and Francesca Toni. Argumentative XAI: A survey. In *Proc. IJCAI*, pages 4392–4399. ijcai.org, 2021.

[Doutre and Mailly, 2018] Sylvie Doutre and Jean-Guy Mailly. Constraints and changes: A survey of abstract argumentation dynamics. *Argument Comput.*, 9(3):223–248, 2018.

[Dung, 1995] Phan Minh Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artif. Intell.*, 77(2):321–358, 1995.

[Dunne *et al.*, 2015] Paul E. Dunne, Wolfgang Dvořák, Thomas Linsbichler, and Stefan Woltran. Characteristics of multiple viewpoints in abstract argumentation. *Artif. Intell.*, 228:153–178, 2015.

[Dvořák and Dunne, 2018] Wolfgang Dvořák and Paul E. Dunne. Computational problems in formal argumentation and their complexity. In Pietro Baroni, Dov Gabbay, Massimiliano Giacomin, and Leendert van der Torre, editors, *Handbook of Formal Argumentation*, chapter 13, pages 631–688. College Publications, 2018.

[Gabbay *et al.*, 2021] Dov Gabbay, Massimiliano Giacomin, Guillermo R. Simari, and Matthias Thimm, editors. *Handbook of Formal Argumentation*, volume 2. College Publications, 2021.

[García and Simari, 2004] Alejandro Javier García and Guillermo Ricardo Simari. Defeasible logic programming: An argumentative approach. *Theory Pract. Log. Program.*, 4(1-2):95–138, 2004.

[Gelfond and Lifschitz, 1988] Michael Gelfond and Vladimir Lifschitz. The stable model semantics for logic programming. In *Proc. ICLP/SLP*, pages 1070–1080. MIT Press, 1988.

[Gordon *et al.*, 2007] Thomas F. Gordon, Henry Prakken, and Douglas Walton. The Carneades model of argument and burden of proof. *Artif. Intell.*, 171(10-15):875–896, 2007.

[Kakas *et al.*, 2019] Antonis C. Kakas, Pavlos Moraitis, and Nikolaos I. Spanoudakis. *GORGIAS*: Applying argumentation. *Argument Comput.*, 10(1):55–81, 2019.

[Lehtonen *et al.*, 2020] Tuomo Lehtonen, Johannes P. Wallner, and Matti Järvisalo. An answer set programming approach to argumentative reasoning in the ASPIC+ framework. In *Proc. KR*, pages 636–646. IJCAI, 2020.

[Lehtonen *et al.*, 2021] Tuomo Lehtonen, Johannes P. Wallner, and Matti Järvisalo. Declarative algorithms and complexity results for assumption-based argumentation. *J. Artif. Intell. Res.*, 71:265–318, 2021.

[Lehtonen *et al.*, 2022] Tuomo Lehtonen, Johannes P. Wallner, and Matti Järvisalo. Computing stable conclusions under the weakest-link principle in the ASPIC+ argumentation formalism. In *Proc. KR*, pages 215–225. IJCAI, 2022.

[Lehtonen *et al.*, 2023] Tuomo Lehtonen, Anna Rapberger, Markus Ulbricht, and Johannes P. Wallner. Argumentation frameworks induced by assumption-based argumentation: Relating size and complexity. In *Proc. KR*, pages 440–450. IJCAI, 2023.

[Lehtonen *et al.*, 2024] Tuomo Lehtonen, Anna Rapberger, Francesca Toni, Markus Ulbricht, and Johannes Peter Wallner. Instantiations and computational aspects of non-flat assumption-based argumentation. *CoRR*, abs/2404.11431, 2024.

[Linsbichler *et al.*, 2022] Thomas Linsbichler, Marco Maratea, Andreas Niskanen, Johannes P. Wallner, and Stefan Woltran. Advanced algorithms for abstract dialectical frameworks based on complexity analysis of subclasses and SAT solving. *Artif. Intell.*, 307:103697, 2022.

[Modgil and Prakken, 2013] Sanjay Modgil and Henry Prakken. A general account of argumentation with preferences. *Artif. Intell.*, 195:361–397, 2013.

[Niemelä, 1999] Ilkka Niemelä. Logic programs with stable model semantics as a constraint programming paradigm. *Ann. Math. Artif. Intell.*, 25(3-4):241–273, 1999.

[Niskanen *et al.*, 2019] Andreas Niskanen, Johannes P. Wallner, and Matti Järvisalo. Synthesizing argumentation frameworks from examples. *J. Artif. Intell. Res.*, 66:503–554, 2019.

[Odekerken *et al.*, 2020] Daphne Odekerken, AnneMarie Borg, and Floris Bex. Estimating stability for efficient argument-based inquiry. In *Proc. COMMA*, volume 326 of *FAIA*, pages 307–318. IOS Press, 2020.

[Odekerken *et al.*, 2023] Daphne Odekerken, Tuomo Lehtonen, AnneMarie Borg, Johannes P. Wallner, and Matti Järvisalo. Argumentative reasoning in ASPIC+ under incomplete information. In *Proc. KR*, pages 531–541. IJCAI, 2023.

[Proietti and Toni, 2023] Maurizio Proietti and Francesca Toni. A roadmap for neuro-argumentative learning. In *Proc. of the 17th International Workshop on Neural-Symbolic Learning and Reasoning*, volume 3432 of *CEUR Workshop Proceedings*, pages 1–8. CEUR-WS.org, 2023.

[Saribatur and Wallner, 2021] Zeynep G. Saribatur and Johannes P. Wallner. Existential abstraction on argumentation frameworks via clustering. In *Proc. KR*, pages 549–559. IJCAI, 2021.

[Saribatur *et al.*, 2020] Zeynep G. Saribatur, Johannes P. Wallner, and Stefan Woltran. Explaining non-acceptability in abstract argumentation. In *Proc. ECAI*, volume 325 of *FAIA*, pages 881–888. IOS Press, 2020.

[Strass and Wallner, 2015] Hannes Strass and Johannes P. Wallner. Analyzing the computational complexity of abstract dialectical frameworks via approximation fixpoint theory. *Artif. Intell.*, 226:34–74, 2015.

[Strass *et al.*, 2019] Hannes Strass, Adam Wyner, and Martin Diller. *EMIL*: Extracting meaning from inconsistent language: Towards argumentation using a controlled natural language interface. *Int. J. Approx. Reason.*, 112:55–84, 2019.

[Strass, 2015] Hannes Strass. Expressiveness of two-valued semantics for abstract dialectical frameworks. *J. Artif. Intell. Res.*, 54:193–231, 2015.

[Ulbricht and Wallner, 2021] Markus Ulbricht and Johannes P. Wallner. Strong explanations in abstract argumentation. In *Proc. AAAI*, pages 6496–6504. AAAI Press, 2021.

[Ulbricht and Wallner, 2022] Markus Ulbricht and Johannes P. Wallner. Strongly accepting subframeworks: Connecting abstract and structured argumentation. In *Proc. COMMA*, volume 353 of *FAIA*, pages 320–331. IOS Press, 2022.

[Vassiliades *et al.*, 2021] Alexandros Vassiliades, Nick Bassiliades, and Theodore Patkos. Argumentation and explainable artificial intelligence: a survey. *Knowl. Eng. Rev.*, 36:e5, 2021.

[Wallner *et al.*, 2017] Johannes P. Wallner, Andreas Niskanen, and Matti Järvisalo. Complexity results and algorithms for extension enforcement in abstract argumentation. *J. Artif. Intell. Res.*, 60:1–40, 2017.