Inference Operators for Argumentation Formalisms – The Case of Dung-style Frameworks

Ringo Baumann[1], Leendert van der Torre[2]

[1]Leipzig University, Germany
[2]University of Luxembourg, Luxembourg

Abstract

We study abstract properties of possible inference operators for Dung-style Argumentation Frameworks. In this first attempt, we revisit classical non-monotonic formalisms such as Default Logic and Logic Programming and adapt their core concepts to the realm of Dung's Argumentation Theory. The resulting operators provide initial formal insights and open avenues for future work, such as exploring the full range of existing semantics, acceptance modes or extending the approach to more expressive abstract and structured argumentation formalisms.

Keywords

Inference Operators, Non-monotonic Reasoning, Dung-style Argumentation Frameworks

1. Introduction

After several decades of research, the field of knowledge representation now offers a variety of formalisms that allow for *non-monotonic reasoning* – that is, the ability to retract previously drawn conclusions. To compare different non-monotonic formalisms – or more generally, to clarify what we expect from such formalisms – a very active and fruitful line of research has focused on studying abstract properties of the associated *inference operators*, which map sets of premises to sets of conclusions. One major contribution was made by Dov Gabbay in 1985 [1], who introduced the notion of *cumulativity*, a combination of the principles known as *Cut* and *Cautious Monotony*, which remain hidden in classical monotonic logics. In this paper, we revisit this classical line of research and apply it directly (i.e., without considering an underlying logic [2]) to one of the most influential formalisms in argumentation theory: the so-called *Dung-style Argumentation Frameworks* [3]. The present study shows inherent properties and paves the way for the long-term goal, namely enabling comparisons with different abstract as well as structured argumentation formalisms [4, 5].

2. Inference Operators – Properties and Examples

In classical logic, a formula ϕ is said to be a logical consequence of a set T of formulas, denoted $T \models \phi$, if and only if $\mathsf{Mod}(T) \subseteq \mathsf{Mod}(\phi)$. The classical *consequence operator* [6, 7] then yields the set of all such consequences, i.e.: $\mathsf{Cn}(T) = \{\phi \mid T \models \phi\}$. This operator can be generalized by allowing other kinds of inputs – for instance, different sets of well-formed formulas, atoms only, assumptions, or even arguments (formalized in an appropriate way). Let $\mathcal{F}$ denote the set of suitable inputs, and let C be a function, a so-called *inference operator*, operating on subsets of $\mathcal{F}$, formally defined as: $\mathsf{C} : 2^{\mathcal{F}} \to 2^{\mathcal{F}}, \quad T \mapsto \mathsf{C}(T)$.

Consider the following abstract properties that enable a systematic comparison [8, 9]. **Inclusion:** $T \subseteq \mathsf{C}(T)$ (no information from T is lost), **Idempotence:** $\mathsf{C}(\mathsf{C}(T)) \subseteq \mathsf{C}(T)$ (applying C again yields nothing new), **Monotonicity:** If $S \subseteq T$, then $\mathsf{C}(S) \subseteq \mathsf{C}(T)$ (adding information preserves old conclusions),

NMR'25: 23rd International Workshop on Nonmonotonic Reasoning, November 11–13, 2025, Melbourne, Australia
baumann@informatik.uni-leipzig.de (R. Baumann); leon.vandertorre@uni.lu (L. v. d. Torre)

Cautious Monotony: If $S \subseteq T \subseteq \mathsf{C}(S)$, then $\mathsf{C}(S) \subseteq \mathsf{C}(T)$ (new information already entailed doesn't invalidate old results), **Cut:** If $S \subseteq T \subseteq \mathsf{C}(S)$, then $\mathsf{C}(T) \subseteq \mathsf{C}(S)$ (new information already entailed doesn't give rise for new results), **Cumulativity:** If $S \subseteq T \subseteq \mathsf{C}(S)$, then $\mathsf{C}(T) = \mathsf{C}(S)$ (robustness under intermediate results), **Compactness:** $\mathsf{C}(T) \subseteq \bigcup\{\mathsf{C}(T') \mid T' \subseteq T,\ T' \text{ finite}\}$ (finite subtheories suffice for inference), **Supraclassicality:** $\mathsf{Cn}(T) \subseteq \mathsf{C}(T)$ (all classical consequences are preserved).

Default Logic introduced by Raymond Reiter [10], is a nonmonotonic formalism based on *defaults* $\delta : \frac{A:B_1,\ldots,B_n}{C}$ where A is the prerequisite, $B_1, \ldots, B_n$ are consistency conditions, and C is the conclusion. For instance, the default $\delta_1 : \frac{\text{kid}(x):\text{likesIceCream}(x)}{\text{likesIceCream}(x)}$ expresses the knowledge "Kids usually like ice cream."

A *default theory* is a pair (D, T), where D is a set of defaults and T a set of formulas, e.g., $T = \{\text{kid}(samantha)\}$. A *Reiter extension* E is defined quasi-inductively as $E = \bigcup_{i=0}^{\infty} E_i$ with $E_0 = T$, and $E_{i+1} = \mathsf{Cn}(E_i) \cup \left\{C \mid \frac{A:B_1,\ldots,B_n}{C} \in D,\ A \in E_i,\ \neg B_j \notin E \text{ for all } j\right\}$. Extensions need not always exist; there may be none, exactly one, or arbitrarily many. In our example, we obtain the unique extension $E = \mathsf{Cn}(\{\text{kid}(samantha), \text{likesIceCream}(samantha)\})$.

The subsequent associated (sceptical) inference operator satisfies inclusion, idempotence, cut, and supraclassicality, but generally fails compactness, monotonicity, cumulativity, and cautious monotonicity.

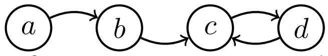
Definition 2.1 (Inference operator – Reiter Extensions). *Let $\mathcal{E}_{(D,T)}$ denote the set of all extensions of a default theory (D, T). The inference operator is defined as:* $\mathsf{C}_D : 2^{\mathcal{F}} \to 2^{\mathcal{F}}, \quad T \mapsto \mathsf{C}_D(T) = \bigcap \mathcal{E}_{(D,T)}$

Logic Programs (LPs) are finite sets of rules of the form: $r : A \leftarrow B_1, \ldots, B_n, \text{not } C_1, \ldots, \text{not } C_m$ where A is the head of r, and $B_1, \ldots, B_n$ (positive atoms) together with not $C_1, \ldots,$ not C_m (default-negated atoms) form the body of r. For example, the rule $r_1 : \mathit{Hunts}(x) \leftarrow \mathit{Owl}(x),\ \text{not } \mathit{InZoo}(x)$ states intuitively that "Owls hunt, unless they live in the zoo." The famous stable model semantics introduced by Michael Gelfond and Vladimir Lifschitz [11] requires the *reduct* of a given LP P w.r.t. a set I defined as: $P^I = \{\, A \leftarrow B_1, \ldots, B_n \mid A \leftarrow B_1, \ldots, B_n, \text{not } C_1, \ldots, \text{not } C_m \in P \text{ with } \{C_1, \ldots, C_m\} \cap I = \emptyset\}$. A set of atoms $I \subseteq \mathcal{A}$ is a *stable model* of P if it is the $\subseteq$-least model of P^I, where a rule $A \leftarrow B_1, \ldots, B_n$ is interpreted as the material implication $B_1 \wedge \ldots \wedge B_n \to A$.

The following associated (sceptical) operator behaves as in the case of default logic; it satisfies inclusion, idempotence, cut, and supraclassicality (under the standard translation to classical logic), but generally fails compactness, monotonicity, cumulativity, and cautious monotonicity.

Definition 2.2 (Inference operator — Stable Models). *Let $\mathrm{SM}(P)$ denote the set of all stable extensions of the logic program P. The inference operator is defined as:* $\mathsf{C}_P : 2^{\mathcal{A}} \to 2^{\mathcal{A}}, \quad I \mapsto \mathsf{C}_P(I) = \bigcap \mathrm{SM}(P \cup I)$

3. Dung-style Argumentation

In 1995, Dung introduced one of the most influential argumentation formalisms which represent arguments and attacks as abstract entities – that is, neither the internal structure of arguments nor the reasons why one argument attacks another are taken into account [3]. Consequently, an *Argumentation Frameworks (AF)* F is simply a directed graph (A, R), where $A \subseteq \mathcal{U}$ is a set of arguments and $R \subseteq A \times A$ a binary relation representing attacks. The main focus lies in resolving conflicts. For the purposes of this paper, we focus exclusively on *stable semantics* (see [12] for other semantics). A set $E \subseteq A$ is a stable extension if (i) E is conflict-free, and (ii) every argument not in E is attacked by some argument in E. Let $stb(F)$ denote the set of all stable extensions of an AF F. As with Logic Programming, stable extensions are not guaranteed to exist; a framework may have none, exactly one, or multiple stable extensions. For instance, the following AF F:  yields two stable extensions, namely $E_1 = \{a, c\}$ and $E_2 = \{a, d\}$. Which inference operator is suitable to adequately capture the dynamics of argumentation? In this initial paper, we draw inspiration from the inference operators already introduced for Default Logic and Logic Programming under stable model semantics.

Default Logic–style Inference Operator In the case of Default Logic, the inference operator is parameterized by a set of defaults D. This means that for each different set D, we obtain a different operator C_D. Each such operator takes as input a set of (non-default) knowledge T, and its output $\mathsf{C}_D(T)$ is defined as the intersection of all Reiter extensions, i.e., $\mathsf{C}_D(T) = \bigcap \mathcal{E}_{(D,T)}$. Transferring this concept to Dung-style Argumentation, we may parameterize an operator by a set of attacks R and take as input a set of arguments A. The output is then defined as the intersection of all stable extensions of the restricted AF $(A, R|_A)$. This operator behaves quite interestingly, as it satisfies idempotence, cautious monotony, cut, and thus cumulativity, but generally fails inclusion, monotonicity, and compactness. We note that the corresponding credulous inference operator – i.e., taking the union of all stable extensions – satisfies the same properties as the sceptical one and, in addition, even satisfies compactness.

Definition 3.1 (Inference operator – Default Logic-style). *Let $R \subseteq \mathcal{U} \times \mathcal{U}$ be an attack relation. The associated (sceptical) inference operator is defined as:* $\mathsf{C}_R : 2^{\mathcal{U}} \to 2^{\mathcal{U}}, \quad A \mapsto \mathsf{C}_R(A) = \bigcap stb\left((A, R|_A)\right)$

Logic Programming–style Inference Operator In the case of logic programs, the inference operator is parameterized by a logic program P. This means that for each different program P, we obtain a different operator C_P. Each such operator takes as input a set of atoms I and outputs $\mathsf{C}_P(I) = \bigcap stb(P \cup I)$, i.e., the intersection of all stable models of $P \cup I$. Transferring this concept in a straightforward way to the realm of Dung's Argumentation Frameworks yields an operator parameterized by an AF $F = (B, R)$, which takes as input a set of arguments A. The output is then defined as the intersection of all stable extensions of the augmented framework $(A \cup B, R)$.

Definition 3.2 (Inference operator – Logic Programming-style). *Let $F = (B, R)$ be an AF. The associated (sceptical) inference operator is defined as:* $\mathsf{C}_F : 2^{\mathcal{U}} \to 2^{\mathcal{U}}, \quad A \mapsto \mathsf{C}_F(A) = \bigcap stb\left((A \cup B, R)\right)$

Although both operator definitions look quite similar, their behaviour differs significantly. Whereas the Default Logic–style inference operator exhibits non-monotonic behaviour (as expected), its Logic Programming–style counterpart satisfies classical monotonicity. This may be surprising at first glance, but becomes intuitive upon closer inspection: adding arguments that introduce no new conflicts does not provide a reason to retract previously accepted information. Note that in the case of the Default Logic–style operator, additional attacks are implicitly introduced due to the presence of a background attack relation. In summary, the inference operator as defined above satisfies idempotence, compactness, monotonicity, and thus also cautious monotony, cut, and cumulativity, but it generally fails inclusion.

4. Conclusions and Future Lines

The present study admits instantiation through different acceptance modes, argumentation semantics, or even solely by considering abstract principles underlying argumentation semantics [13, 14].

One central question is how the presented operators are related. While the Default-style operator exhibits full non-monotonicity, as expected, the LP-style operator can be seen as a *monotonic bridge logic*, as termed by Makinson [8]. We note that identifying monotonic fragments within argumentation has already attracted some interest [15, 16, 17]. Finally, the presented operators do not satisfy inclusion. We believe this is an essential feature of any argumentation operator, as one cannot expect that all arguments they put forward will be accepted or pass unchallenged.

Acknowledgments

This work was supported by the German Research Foundation (DFG, BA 6170/3-1) and by the German Federal Ministry of Education and Research (BMBF, 01/S18026A-F) by funding the competence center for Big Data and AI "ScaDS.AI" Dresden/Leipzig. Leon van der Torre acknowledges the financial support of the Luxembourg National Research Fund (FNR) through the projects: The Epistemology of AI Systems (EAI) (C22/SC/17111440), DJ4ME – A DJ for Machine Ethics: the Dialogue Jiminy (O24/18989918/DJ4ME), Logical Methods for Deontic Explanations (LoDEx) (INTER/DFG/23/17415164/LoDEx) and Symbolic and Explainable Regulatory AI for Finance Innovation (SERAFIN) (C24/19003061/SERAFIN).

Declaration on Generative AI

The authors have not employed any Generative AI tools.

References

[1] D. M. Gabbay, Theoretical Foundations for Non-Monotonic Reasoning in Expert Systems, in: K. R. Apt (Ed.), Logics and Models of Concurrent Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 1985, pp. 439–457.

[2] O. Arieli, A. Borg, J. Heyninck, C. Straßer, Logic-Based Approaches to Formal Argumentation, FLAP 8 (2021) 1793–1898. URL: https://collegepublications.co.uk/ifcolog/?00048.

[3] 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.

[4] G. Brewka, S. Woltran, Abstract Dialectical Frameworks, in: F. Lin, U. Sattler, M. Truszczynski (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Twelfth International Conference, KR 2010, Toronto, Ontario, Canada, May 9-13, 2010, AAAI Press, 2010. URL: http://aaai.org/ocs/index.php/KR/KR2010/paper/view/1294.

[5] A. Bondarenko, F. Toni, R. A. Kowalski, An Assumption-Based Framework for Non-Monotonic Reasoning, in: Logic Programming and Non-monotonic Reasoning, Proceedings of the Second International Workshop, Lisbon, Portugal, June 1993, 1993, pp. 171–189.

[6] A. Tarski, Fundamentale Begriffe der Methodologie der deduktiven Wissenschaften., Monatshefte für Mathematik und Physik 37 (1930) 361–404.

[7] A. Tarski, Uber einige fundamentale Begriffe der Methamathematik, Comptes rendus de la Societe des Sciences et des Lettres de Varsovien Classe XIX2Y (1930).

[8] D. Makinson, How to Go Nonmonotonic, Springer Netherlands, Dordrecht, 2005, pp. 175–278. URL: https://doi.org/10.1007/1-4020-3092-4_3. doi:10.1007/1-4020-3092-4_3.

[9] S. Kraus, D. Lehmann, M. Magidor, Nonmonotonic Reasoning, Preferential Models and Cumulative Logics, Artificial Intelligence 44 (1990) 167–207. URL: https://www.sciencedirect.com/science/article/pii/0004370290901015. doi:https://doi.org/10.1016/0004-3702(90)90101-5.

[10] R. Reiter, A Logic for Default Reasoning, Artificial Intelligence 13 (1980) 81–132. URL: https://www.sciencedirect.com/science/article/pii/0004370280900144. doi:https://doi.org/10.1016/0004-3702(80)90014-4, Special Issue on Non-Monotonic Logic.

[11] M. Gelfond, V. Lifschitz, The Stable Model Semantics for Logic Programming, in: Logic Programming, Proceedings of the Fifth International Conference and Symposium, Seattle, Washington, USA, August 15-19, 1988 (2 Volumes), 1988, pp. 1070–1080.

[12] P. Baroni, M. Caminada, M. Giacomin, Abstract Argumentation Frameworks and Their Semantics, in: Handbook of Formal Argumentation, College Publications, 2018, pp. 159–236.

[13] P. Baroni, M. Giacomin, On Principle-Based Evaluation of Extension-Based Argumentation Semantics, Artificial Intelligence 171 (2007) 675–700. URL: https://doi.org/10.1016/j.artint.2007.04.004. doi:10.1016/J.ARTINT.2007.04.004.

[14] L. van der Torre, S. Vesic, The Principle-Based Approach to Abstract Argumentation Semantics, FLAP 4 (2017). URL: http://www.collegepublications.co.uk/downloads/ifcolog00017.pdf.

[15] R. Baumann, G. Brewka, Expanding Argumentation Frameworks: Enforcing and Monotonicity Results, in: P. Baroni, F. Cerutti, M. Giacomin, G. R. Simari (Eds.), Computational Models of Argument: Proceedings of COMMA 2010, Desenzano del Garda, Italy, September 8-10, 2010, volume 216 of *Frontiers in Artificial Intelligence and Applications*, IOS Press, 2010, pp. 75–86. URL: https://doi.org/10.3233/978-1-60750-619-5-75. doi:10.3233/978-1-60750-619-5-75.

[16] C. Cayrol, F. D. de Saint-Cyr, M. Lagasquie-Schiex, Revision of an Argumentation System, in: G. Brewka, J. Lang (Eds.), Principles of Knowledge Representation and Reasoning: Proceedings of the Eleventh International Conference, KR 2008, Sydney, Australia, September 16-19, 2008, AAAI Press, 2008, pp. 124–134. URL: http://www.aaai.org/Library/KR/2008/kr08-013.php.

[17] R. Booth, S. Kaci, T. Rienstra, L. W. N. van der Torre, Monotonic and Nonmonotonic Inference for Abstract Argumentation, in: C. Boonthum-Denecke, G. M. Youngblood (Eds.), Proceedings of the Twenty-Sixth International Florida Artificial Intelligence Research Society Conference, FLAIRS 2013, St. Pete Beach, Florida, USA, May 22-24, 2013, AAAI Press, 2013. URL: http://www.aaai.org/ocs/index.php/FLAIRS/FLAIRS13/paper/view/5863.