Argumentation-Based Defeasible Reasoning For Existential Rules

(Extended Abstract)

Abdelraouf Hecham[*]
University Of Montpellier
hecham@lirmm.fr

Madalina Croitoru
University Of Montpellier
croitoru@lirmm.fr

Pierre Bisquert
IATE, INRA Montpellier
pierre.bisquert@inra.fr

ABSTRACT

Logic based argumentation allows for defeasible reasoning over monotonic logics. In this paper, we introduce **DEFT**, a tool implementing *argumentative defeasible reasoning over existential rules*. We explain how DEFT overcomes derivation loss and discuss DEFT's empirical behavior.

1. RESEARCH CONTEXT

We are interested in argumentation based defeasible reasoning [9, 16, 11] for *existential rules*, a first order logic subset employed for sharing, reuse and reasoning over large databases on the Semantic Web (SW) [7] and we focus on the tractable fragments of existential rules, $Datalog^{\pm}$ [7]. Defeasible $Datalog^{\pm}$ was proposed in [14, 8] but it *does not account for the possibility of loss* of deduced information that can *render the reasoning process non deterministic.* We improve upon the state of the art by proposing a new hypergraph-based algorithm that will prevent derivation loss.

The contribution of the paper is to provide the *first argumentation based tool* for defeasible $Datalog^{\pm}$ in the literature: DEFT (Defeasible Datalog$^{\pm}$ Tool) and compare its behavior with respect to argumentation tools in the literature relevant for defeasible reasoning with ambiguity propagation: ASPIC+ [15] with its grounded semantics [12] and DeLP [11] with its dialectical trees.

2. CHASE DERIVATION LOSS

Defeasible $Datalog^{\pm}$ extends $Datalog^{\pm}$ to include defeasible facts and defeasible rules. $Datalog^{\pm}$ refers to the decidable fragment of existential rules [7] that extends the Datalog language with *existential variables* in the rule head. Defeasible reasoning in this context is based on dialectical trees, which, in turn, are based on the classical notion of *derivation* (i.e. the successive application of a set of rules on a set of facts). *The rule application mechanism is called the chase.* The chase procedure is equipped with a restriction test, called derivation reducer, for detecting when the

---

[*]Corresponding Author.

rule application becomes redundant. In the literature several chases are studied [2]. Here we only restrict ourselves to the derivation reducer used for the Restricted chase [10]. Using a derivation reducer may induce a loss of rule applications depending on the order in which rules are applied. While the order of applications does not impact the final model of the saturated knowledge base (and thus entailment in $Datalog^{\pm}$), it does **affect the set of extracted derivations**. This is important since dialectical-tree-based defeasible reasoning relies on **the set of extracted derivations**.

To allow for lossless derivation extraction we use an adapted combinatorial structure called Graph of Atom Dependency (GAD). This structure can be seen as an improvement over the chase graph [7] that allows *keeping track of all the generated atoms during the chase procedure*. In the GAD the nodes correspond to facts and the labelled directed hyperedges to the rule applications. The intuition behind the use of the GAD is that, for a given GAD and a given atom, there is a *one-to-one mapping, up to derivation equivalence*, between the set of hyperpaths to an atom f and the set of derivations to f. The problem of obtaining all minimal derivations of f can thus be transformed into the problem of generating all minimal hyperpaths of f in the GAD. For every hyperpath of the GAD we can construct a derivation, and for every derivation there exists a hyperpath. This ensures soundness and completeness of all minimal derivations extraction.

Let us present the chase-based implementation of defeasible $Datalog^{\pm}$ called "DEFT". DEFT uses the GAD structure and relies on the $Datalog^{\pm}$ dedicated inference engine called "GRAAL" [3] that accepts a wide variety of formats (OWL2, RuleML and the $Datalog^{\pm}$ format DLGP [3]). We run the restricted chase using the GRAAL framework in order to create the GAD.

3. RESULTS AND DISCUSSION

By using the GAD structure, DEFT is the first sound and complete $Datalog^{\pm}$ defeasible reasoning tool in the literature. The $Datalog^{\pm}$ features impose a dedicated tool due to two main aspects. First, $Datalog^{\pm}$ allows for existential rules, which neither ASPIC+ nor DeLP can handle since they cannot express existential variables. This can easily be checked by trying to answer the query $q(a, a)$ with the rule $p(X) \rightarrow q(X, Y)$ and the fact $p(a)$ using online tools like ASPIC argumentation engine demo[1] or DeLPclient[2]. The

---

[1] http://aspic.cossac.org/ArgumentationSystem/
[2] http://lidia.cs.uns.edu.ar/delp_client/

query is entailed if the system does not support existential rules (as the Y variable is mapped to all known constants). Second, since $Datalog^\pm$ allows for weak negation but not classic negation in the body of rules, DEFT can simply use a fast chase mechanism, whereas existing argumentation-based reasoning tools rely on resolution-based inference mechanisms [6] since they account for classic negation, which induces a **large computational overhead**.

We conducted an empirical evaluation of DEFT in order to measure its performance w.r.t to DeLP and ASPIC+ [3]. The experiments are built upon a pre-established defeasible reasoning benchmark proposed in [13]. The benchmark we consider is composed of 5 parameterized knowledge bases (also known as theories): **Chain Theory** tests performance when faced with a simple chain of rules; **Circle Theory** tests infinite loops (cycles), **Levels and Trees Theories** test a large number of arguments with small derivations and **Teams Theory** tests performance w.r.t. a sizeable number of conflicts. The results are shown in the table below.

Table 1: Execution time in seconds

	Theory	Size	DEFT	ASPIC+	DeLP
chain(n)	$n = 100$	201	**0.02**	0.16	0.97
	$n = 400$	801	**0.04**	23.57	31.01
	$n = 800$	1601	**0.08**	∞	254.96
	$n = 2000$	4001	**0.28**	∞	T.O.
circle(n)	$n = 100$	201	**0.02**	∞	0.99
	$n = 400$	801	**0.04**	∞	49.88
	$n = 800$	1601	**0.06**	∞	T.O.
	$n = 2000$	4001	**0.25**	∞	T.O.
levels(n)	$n = 25$	261	0.02	0.02	169.94
	$n = 4000$	40011	**1.09**	23.22	T.O.
	$n = 8000$	80011	**2.08**	115.68	T.O.
	$n = 11000$	110011	**2.83**	T.O.	T.O.
teams(n)	$n = 2$	232	0.09	**0.02**	6.93
	$n = 3$	808	0.36	**0.13**	T.O.
	$n = 4$	3196	**3.08**	10.50	T.O.
	$n = 5$	15016	**25.99**	133.73	T.O.
trees(n, k)	$n = 3, k = 5$	311	0.10	0.10	227.38
	$n = 4, k = 5$	1561	**0.68**	10.52	T.O.
	$n = 5, k = 5$	7811	**2.14**	109.36	T.O.
	$n = 6, k = 5$	39061	T.O.	T.O.	T.O.

The salient points of this evaluation are two-fold. First, DEFT is the **only tool able to reason with the class of existential rules guaranteed to stop in forward chaining** (namely, Finite Expansion Set [2]). Second, DEFT **out-performs existing argumentation-based defeasible reasoning tool** for general logical fragments with only weak negation.

In future work we plan to investigate the semantic relation between defeasible $Datalog^\pm$ and the floating conclusions

from the work of [1]. We also plan to use DEFT in human reasoning akin frameworks for irrationality [5, 4].

REFERENCES

[1] A. Arioua and M. Croitoru. A dialectical proof theory for universal acceptance in coherent logic-based argumentation frameworks. In *ECAI*, volume 285, pages 55–63. IOS Press, 2016.

[2] J.-F. Baget, F. Garreau, M.-L. Mugnier, and S. Rocher. Extending acyclicity notions for existential rules. In *ECAI*, pages 39–44, 2014.

[3] J.-F. Baget, M. Leclère, M.-L. Mugnier, S. Rocher, and C. Sipieter. Graal: A toolkit for query answering with existential rules. In *International Symposium on Rules and Rule Markup Languages for the Semantic Web*, pages 328–344. Springer, 2015.

[4] P. Bisquert, M. Croitoru, F. D. de Saint-Cyr, and A. Hecham. Formalizing cognitive acceptance of arguments: Durum wheat selection interdisciplinary study. *Mind and Machines (to appear)*, 2016.

[5] P. Bisquert, M. Croitoru, F. D. de Saint-Cyr, and A. Hecham. Substantive irrationality in cognitive systems. In *Proceedings of ECAI 2016*, 2016.

[6] D. Bryant and P. Krause. A review of current defeasible reasoning implementations. *The Knowledge Engineering Review*, 23(03):227–260, 2008.

[7] A. Calì, G. Gottlob, and T. Lukasiewicz. A general datalog-based framework for tractable query answering over ontologies. *Web Semantics: Science, Services and Agents on the World Wide Web*, 14:57–83, 2012.

[8] C. Deagustini, M. Martinez, M. Falappa, and G. Simari. On the influence of incoherence in inconsistency-tolerant semantics for datalog±. 2015.

[9] P. M. Dung. On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. *Artificial intelligence*, 77(2):321–357, 1995.

[10] R. Fagin, P. G. Kolaitis, R. J. Miller, and L. Popa. Data exchange: semantics and query answering. *Theoretical Computer Science*, 336(1):89–124, 2005.

[11] A. J. García and G. R. Simari. Defeasible logic programming: An argumentative approach. *Theory and practice of logic programming*, 4:95–138, 2004.

[12] G. Governatori, M. J. Maher, G. Antoniou, and D. Billington. Argumentation semantics for defeasible logic. *Journal of Logic and Computation*, 14(5):675–702, 2004.

[13] M. J. Maher, A. Rock, G. Antoniou, D. Billington, and T. Miller. Efficient defeasible reasoning systems. *International Journal on Artificial Intelligence Tools*, 10(04):483–501, 2001.

[14] M. V. Martinez, C. A. D. Deagustini, M. A. Falappa, and G. R. Simari. Inconsistency-tolerant reasoning in datalog± ontologies via an argumentative semantics. In *IBERAMIA 2014*, pages 15–27. Springer, 2014.

[15] H. Prakken. An abstract framework for argumentation with structured arguments. *Argument and Computation*, 1(2):93–124, 2010.

[16] H. Prakken and G. Sartor. Argument-based extended logic programming with defeasible priorities. *Journal of applied non-classical logics*, 7(1-2):25–75, 1997.

---

[3] We used DeLP implementation in Tweety1.7 libraries and an author provided implementation for ASPIC+.