Reasoning Efficiently with Ontologies and Rules in the Presence of Inconsistencies (Extended Abstract)⋆

Tobias Kaminski, Matthias Knorr, and João Leite

NOVA LINCS, Departamento de Informática, Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa

In this paper, we address the problem of dealing with inconsistent knowledge bases consisting of ontologies and non-monotonic rules, following a paraconsistent reasoning approach with a focus on efficiency.

Description Logics (*DLs*) and *Logic Programs* (*LPs*) provide different strengths when used for *Knowledge Representation and Reasoning*. While DLs employ the *Open World Assumption* and are suited for defining ontologies, LPs adopt the *Closed World Assumption* and are able to express non-monotonic rules with exceptions and preference orders. Combining features of both formalisms has been actively pursued over the last few years, resulting in different proposals with different levels of integration and complexity: while some extend DLs with rules [18, 25], others follow a hybrid combination of ontologies with non-monotonic rules, either providing a modular approach where rules and ontologies use their own semantics, and allowing limited interaction between them [10], or defining a unifying framework for both components [29, 24]. Equipped with semantics that are faithful to their constitutive parts, these proposals allow for the specification of so-called *hybrid knowledge bases* (*hybrid KBs*) either from scratch, benefiting from the added expressivity, or by combining existing ontologies and rule bases.

The complex interactions between the ontology component and the rule component of these hybrid KBs – even more so when they result from combining existing ontologies and rule bases developed independently – can easily lead to contradictions, which, under classical semantics, trivialize standard reasoning and prevent us from drawing any meaningful conclusions, ultimately rendering these hybrid KBs useless.

Example 1. Consider the following simplified (ground) hybrid KB $\mathcal{K}_G$ for assessing the risk of goods at a port.

$$HasCertifiedSender \sqsubseteq \neg IsMonitored \tag{1}$$

$$\mathbf{K} IsMonitored(g) \leftarrow \mathbf{K} risk(g). \tag{2}$$

$$\mathbf{K} risk(g) \leftarrow \mathbf{not}\, isLabelled(g). \tag{3}$$

$$\mathbf{K} isLabelled(g) \leftarrow \mathbf{not}\, risk(g). \tag{4}$$

$$\mathbf{K} resolvedRisk(g) \leftarrow \mathbf{K} IsMonitored(g). \tag{5}$$

$$\mathbf{K} HasCertifiedSender(g) \leftarrow \tag{6}$$

$$\mathbf{K} risk(g) \leftarrow \tag{7}$$

⋆ This is an extended abstract of [22]. Partially supported by Fundação para a Ciência e a Tecnologia under project PTDC/EIA-CCO/121823/2010 and strategic project PEst/UID/CEC/04516/2013. M. Knorr was also supported by grant SFRH/BPD/86970/2012.

Rules (3) and (4) state that good g is either a risk (r) or it is labeled (iL). Any risk is monitored (IM) (2), thus a resolved risk (rR) (5). As g has a certified sender (HCS) (6), it can be proven by means of axiom (1) that it is not monitored. Thus, g can be derived to be monitored and not monitored at the same time if it is considered to be a risk (7), i.e., the hybrid KB is inconsistent, which trivializes standard reasoning.

One way to deal with this problem is to employ some method based on belief revision (e.g. [26, 30, 35, 37, 9] for LPs, [14, 7, 23] for DLs, and [38, 36] for hybrid KBs) to regain consistency so that standard reasoning services can be used, or some method based on repairing (e.g. [5] for LPs, [17] for DLs, and [12, 11] for dl-programs [10]) where hypothetical belief revision is employed for consistent query answering, without actually changing the KB. However, this is not always feasible e.g. because, we may not have permission to change the KB – as for instance in [1] where the KB encodes laws and norms – or because the usual high complexity of belief revision and repairing methods simply renders their application prohibitive. When these methods are not possible or not feasible, paraconsistent reasoning services, typically based on some many-valued logic, offer an alternative by being able to draw meaningful conclusions in the presence of contradictions.

Paraconsistent reasoning has been extensively studied in both base formalisms of hybrid KBs. For DLs, most work [31, 39, 27, 41, 28] focuses on four-valued semantics varying which classical rules of inferences they satisfy. Among them, [27, 28] is most general as it covers $\mathcal{SROIQ}$, the DL behind OWL 2, considers tractable subclasses and truth value removals, and permits re-using classical reasoners. Three-valued semantics for DLs [40] and measuring the degree of inconsistency in *DL-Lite* [42] have also been considered. For LPs, the comprehensive survey [8] discusses e.g. a four-valued semantics without default negation [6], a four-, six-, and nine-valued semantics [34] for answer sets [16], and a seven- [33] and nine-valued [3] well-founded semantics [15]. More recently, a very general framework for arbitrary bilattices of truth values [2] and paraconsistent Datalog [4] have been considered. At the same time, paraconsistent reasoning is still a rather unexplored field in the context of hybrid KBs. Notable exceptions are [20, 19, 13], yet their computation is not tractable in general even if reasoning in the DL component is.

In this paper, we investigate efficient paraconsistent semantics for hybrid KBs. We adopt the base framework of [29] because of its generality and tight integration between the ontology and the rules – cf. [29] for a thorough argument in its favor – under the semantics of [24] because of its computational properties. We extend this semantics with additional truth values to evaluate contradictory pieces of knowledge, following two common views on how to deal with contradictory knowledge bases.

According to one view, contradictions are dealt with locally, in a minimally intrusive way, such that a new truth value is introduced to model inconsistencies, but non-contradictory knowledge only derivable from the inconsistent part of a KB is still considered to be true in the classical sense. This view is adopted in paraconsistent semantics for DLs, e.g. [28], LPs, e.g. [33, 34], and hybrid KBs [20, 13]. Since two different kinds of inconsistencies are identified in the three-valued semantics of [24], two further truth values are introduced when following this first approach in extending the work of [24], resulting in a five-valued semantics. Namely, we extend the set of truth values *true* (**t**), *false* (**f**), and *undefined* (**u**) used in [24] by the truth value **b** for *both*, which is assigned whenever an atom is considered *true* and *false* at the same time, and the truth value **uf** for *undefined false*, which is used whenever an atom would be considered simultaneously *undefined* and *false*.

The alternative view is to distinguish truth which depends on the inconsistent part of a KB from truth which is derivable without involving any contradictory knowledge. This view, commonly referred to as *Suspicious Reasoning*, is adopted in paraconsistent semantics for LPs, e.g. [3, 33, 34] and hybrid KBs [19]. In order to extend the approach of [24] in a way that allows for paraconsistency in combination with Suspicious Reasoning, a sixth truth value *suspiciously true* (**st**) is introduced in addition to those already occurring in the five-valued semantics. This truth value is assigned to atoms only derivable by involving a contradiction in the program. At the same time, the truth value **uf** is replaced by the slightly different truth value *classically false* (**cf**), with the aim to also capture "propagation" on derived classical falsity.

As a result, we obtain solutions following both views through the definition of a five-valued and a six-valued paraconsistent semantics for hybrid KBs, the latter implementing Suspicious Reasoning. This requires the integration of quite different concepts and assumptions w.r.t. paraconsistency developed independently for each of the two base formalisms, e.g. Suspicious Reasoning has not been considered in DLs, while LP semantics may sometimes be defined procedurally. In spite of these obstacles, we can show that both of the resulting semantics enjoy a number of desirable properties.

- Firstly, both semantics are sound w.r.t. the three-valued semantics for consistent hybrid KBs by [24]. In fact, the so-called 5- and 6-models corresponding to models in [24] coincide in this case, so consistent hybrid KBs establish a link between our two semantics.
- Secondly, the semantics assigned to a hybrid KB of which the program component is empty is limited, in both cases, to only three truth values (**t**, **f**, and **b**), which arguably leads to a stronger consequence relation than in common four-valued paraconsistent DL semantics [32]. Still, we can show that, in this case, both semantics coincide with the well-known paraconsistent DL semantics $\mathcal{ALC}4$ by [28] if we omit the truth value **u** (referred to as "removal of gaps"). Moreover, we show that the six-valued semantics is faithful w.r.t. the paraconsistent semantics for extended logic programs $WFSX_p$ [3] when classical negation is only applied to unary atoms. Consequently, properties shown for these paraconsistent semantics for the two base formalisms directly carry over to our approach, e.g. it implements the *Coherence Principle*, which states that classical negation implies default negation.
- Thirdly, we present a sound and complete fixpoint algorithm, which extends the alternating fixpoint construction defined for the three-valued approach in [24]. The algorithm preserves the efficiency of the previous approach in that it is tractable whenever consequences in the DL used for formalizing the ontology component can be computed in polynomial time.

Finally, our approach and results can benefit existing implementations for hybrid knowledge bases. In fact, the comparison between our two fixpoint computations and that in [24] suggest an adaptation of the implementation of the latter, the Protégé plug-in *NoHR* [21], to also consider paraconsistent reasoning based on our semantics.

References

1. Alberti, M., Gomes, A.S., Gonçalves, R., Leite, J., Slota, M.: Normative systems represented as hybrid knowledge bases. In: Procs. of CLIMA XII. Springer (2011)
2. Alcântara, J., Damásio, C.V., Pereira, L.M.: An encompassing framework for paraconsistent logic programs. J. Applied Logic 3(1), 67–95 (2005)
3. Alferes, J.J., Damásio, C.V., Pereira, L.M.: A logic programming system for nonmonotonic reasoning. J. Autom. Reasoning 14(1), 93–147 (1995)
4. de Amo, S., Pais, M.S.: A paraconsistent logic programming approach for querying inconsistent databases. Int. J. Approx. Reasoning 46(2), 366–386 (2007)
5. Arenas, M., Bertossi, L.E., Chomicki, J.: Consistent query answers in inconsistent databases. In: Procs of PODS. ACM Press (1999)
6. Blair, H.A., Subrahmanian, V.S.: Paraconsistent logic programming. Theor. Comput. Sci. 68(2), 135–154 (1989)
7. Calvanese, D., Kharlamov, E., Nutt, W., Zheleznyakov, D.: Evolution of DL-Lite knowledge bases. In: Procs. of ISWC. Springer (2010)
8. Damásio, C.V., Pereira, L.M.: A survey of paraconsistent semantics for logic programs. In: Reasoning with Actual and Potential Contradictions. Springer (1998)
9. Delgrande, J., Schaub, T., Tompits, H., Woltran, S.: A model-theoretic approach to belief change in answer set programming. ACM TOCL 14(2), 14 (2013)
10. Eiter, T., Ianni, G., Lukasiewicz, T., Schindlauer, R., Tompits, H.: Combining answer set programming with description logics for the semantic web. Artif. Intell. 172(12-13), 1495–1539 (2008)
11. Eiter, T., Fink, M., Stepanova, D.: Computing repairs for inconsistent dl-programs over *EL* ontologies. In: Procs. of JELIA. Springer (2014)
12. Eiter, T., Fink, M., Stepanova, D.: Towards practical deletion repair of inconsistent dl-programs. In: Procs. of ECAI. IOS Press (2014)
13. Fink, M.: Paraconsistent hybrid theories. In: Procs of KR. AAAI Press (2012)
14. Flouris, G., Manakanatas, D., Kondylakis, H., Plexousakis, D., Antoniou, G.: Ontology change: classification and survey. Knowledge Eng. Review 23(2), 117–152 (2008)
15. Gelder, A.V., Ross, K.A., Schlipf, J.S.: The well-founded semantics for general logic programs. J. ACM 38(3), 620–650 (1991)
16. Gelfond, M., Lifschitz, V.: Classical negation in logic programs and disjunctive databases. New Generation Comput. 9(3/4), 365–386 (1991)
17. Haase, P., van Harmelen, F., Huang, Z., Stuckenschmidt, H., Sure, Y.: A framework for handling inconsistency in changing ontologies. In: Procs. of ISWC. Springer (2005)
18. Horrocks, I., Patel-Schneider, P.: A proposal for an OWL rules language. In: Procs of WWW. ACM (2004)
19. Huang, S., Hao, J., Luo, D.: Incoherency problems in a combination of description logics and rules. J. Applied Mathematics 2014 (2014)
20. Huang, S., Li, Q., Hitzler, P.: Paraconsistent semantics for hybrid MKNF knowledge bases. In: Procs of RR. Springer (2011)
21. Ivanov, V., Knorr, M., Leite, J.: A query tool for $\mathcal{EL}$ with non-monotonic rules. In: Procs. of ISWC. Springer (2013)
22. Kaminski, T., Knorr, M., Leite, J.: Efficient paraconsistent reasoning with ontologies and rules. In: Procs. of IJCAI. AAAI press (2015), to appear
23. Kharlamov, E., Zheleznyakov, D., Calvanese, D.: Capturing model-based ontology evolution at the instance level: The case of dl-lite. Journal of Computer and System Sciences 79(6), 835–872 (2013)

24. Knorr, M., Alferes, J.J., Hitzler, P.: Local closed world reasoning with description logics under the well-founded semantics. Artif. Intell. 175(9-10), 1528–1554 (2011)
25. Krötzsch, M., Maier, F., Krisnadhi, A., Hitzler, P.: A better uncle for OWL: nominal schemas for integrating rules and ontologies. In: Procs. of WWW. ACM (2011)
26. Leite, J.: Evolving Knowledge Bases. IOS Press (2003)
27. Ma, Y., Hitzler, P., Lin, Z.: Algorithms for paraconsistent reasoning with OWL. In: Procs. of ESWC. Springer (2007)
28. Maier, F., Ma, Y., Hitzler, P.: Paraconsistent OWL and related logics. Semantic Web 4(4), 395–427 (2013)
29. Motik, B., Rosati, R.: Reconciling description logics and rules. J. ACM 57(5) (2010)
30. Osorio, M., Cuevas, V.: Updates in answer set programming: An approach based on basic structural properties. TPLP 7(4), 451–479 (2007)
31. Patel-Schneider, P.: A four-valued semantics for terminological logics. Artif. Intell. 38(3), 319–351 (1989)
32. Priest, G.: The logic of paradox. Journal of Philosophical logic 8(1), 219–241 (1979)
33. Sakama, C.: Extended well-founded semantics for paraconsistent logic programs. In: Procs. of FGCS. IOS Press (1992)
34. Sakama, C., Inoue, K.: Paraconsistent stable semantics for extended disjunctive programs. J. Log. Comput. 5(3), 265–285 (1995)
35. Slota, M., Leite, J.: Robust equivalence models for semantic updates of answer-set programs. In: Procs. of KR. AAAI Press (2012)
36. Slota, M., Leite, J.: A unifying perspective on knowledge updates. In: Procs. of JELIA. Springer (2012)
37. Slota, M., Leite, J.: The rise and fall of semantic rule updates based on se-models. TPLP 14(6), 869–907 (2014)
38. Slota, M., Leite, J., Swift, T.: Splitting and updating hybrid knowledge bases. TPLP 11(4-5), 801–819 (2011)
39. Straccia, U.: A sequent calculus for reasoning in four-valued description logics. In: Procs. of TABLEAUX. Springer (1997)
40. Zhang, X., Lin, Z., Wang, K.: Towards a paradoxical description logic for the semantic web. In: Procs. of FoIKS. Springer (2010)
41. Zhang, X., Xiao, G., Lin, Z.: A tableau algorithm for handling inconsistency in OWL. In: Procs. of ESWC. Springer (2009)
42. Zhou, L., Huang, H., Qi, G., Ma, Y., Huang, Z., Qu, Y.: Paraconsistent query answering over DL-Lite ontologies. Web Intelligence and Agent Systems 10(1), 19–31 (2012)