Reasoning about Actions with $\mathcal{EL}$ Ontologies in a Temporal Action Theory

Extended Abstract

Laura Giordano[1], Alberto Martelli[2] and Daniele Theseider Dupré[1]

[1]DISIT - Università del Piemonte Orientale, Viale Michel 11, I-15121, Alessandria, Italy

[2]Dipartimento di Informatica, Università degli Studi di Torino, Corso Svizzera 185, I-10149,Torino, Italy

Abstract

In this extended abstract we report about an approach for reasoning about actions with domain descriptions including an $\mathcal{EL}^{\bot}$ ontology in a (rule based) temporal action theory. The action theory is based on a Dynamic Linear Time Temporal Logic, and extensions are defined through temporal answer sets. The work provides conditions under which action consistency can be guaranteed with respect to an $\mathcal{EL}^{\bot}$ ontology, by polynomially encoding an $\mathcal{EL}^{\bot}$ knowledge base into the domain description of the temporal action theory.

Keywords

EL Ontologies, Reasoning about Actions, Temporal Action Logic, Answer Set Programming

1. Introduction

In this extended abstract we report about an approach for reasoning about actions with domain descriptions including an $\mathcal{EL}^{\bot}$ ontology in a temporal action theory. The integration of description logics and action formalisms has gained a lot of interest in the past years [1, 2, 3, 4]. In this paper we explore the combination of a rule based temporal action logic [5] and $\mathcal{EL}^{\bot}$ ontologies [6], with the aim of allowing reasoning about action execution in the presence of the constraints given by an $\mathcal{EL}^{\bot}$ knowledge base.

In this work, as in many formalisms integrating Description Logics (DLs) and action languages [1, 7, 3, 4], we regard inclusions in the KB as state constraints of the action theory, which we expect to be satisfied in the state resulting after action execution. In the literature of reasoning about actions it is well known that causal laws and their interplay with domain constraints are crucial for solving the ramification problem [8, 9, 10, 11, 12, 13]. When domain knowledge includes an ontology the issue has been considered, e.g., in [2] where causal laws are used to ensure the consistency with the TBox of the resulting state, after action execution. For instance, given a TBox containing $\exists Teaches.Course \sqsubseteq Teacher$, and an ABox (i.e., a set of assertions on individuals) containing the assertion $Course(math)$, an action which adds the assertion $Teaches(john, math)$, without also adding $Teacher(john)$, will not give rise to a consistent next state with respect to the knowledge base. The addition of the causal law **caused**

DL 2022: 35th International Workshop on Description Logics, August 7–10, 2022, Haifa, Israel

laura.giordano@uniupo.it (L. Giordano); mrt@unito.it (A. Martelli); dtd@uniupo.it (D. Theseider Dupré)

$Teacher(john)$ **if** $Teaches(john, math) \wedge Course(math)$ would force, for instance, the above TBox inclusion to be satisfied in the resulting state.

The approach proposed by Baader et al. [2] uses causal relationships to deal with the ramification problem in an action formalism based on description logics, and it exploits a semantics of actions and causal laws in the style of Winslett's [14] and McCain and Turner's [8] fixpoint semantics. In our work, we aim at extending this approach to reason about actions with an $\mathcal{EL}^{\bot}$ ontology with *temporal answer sets*.

2. Reasoning about Actions with Temporal Answer Sets

Reasoning about actions with temporal answer sets has been proposed in [15, 5, 16] by defining a temporal logic programming language for reasoning about *complex actions* and *infinite computations*. The proposed approach also deals with the verification of temporal goals as advocated in [17]. This action language, besides the usual LTL operators, allows for general Dynamic Linear Time Temporal Logic (DLTL) formulas [18] to be included in domain descriptions to constrain the space of possible extensions.

For the rule-based fragment of this action language, a notion of Temporal Answer Set for domain descriptions has been developed [15, 5], as a generalization of Gelfond and Lifschitz' notion of Answer Set [19], and a translation of domain descriptions into standard Answer Set Programming (ASP) has been provided, by exploiting *bounded model checking techniques* for the verification of DLTL constraints, extending the approach developed by Helianko and Niemela [20] for bounded LTL model checking with Stable Models. An alternative ASP translation of this temporal action language has been investigated in [15, 21], by proposing an approach to bounded model checking which exploits the Büchi automaton construction while searching for a counterexample, with the aim of achieving completeness. This temporal action logic has been shown to be related to extensions of the $\mathcal{A}$ language [22, 23, 24, 25, 13]. Its LTL fragment also relates to the recent temporal extension of Clingo, *telingo* [26], dealing with finite computations, and with the LTL fragment of Temporal Equilibrium Logic (TEL) [27], as the restriction of TEL to the rule based case, leads to a linear-time temporal ASP [28].

In the temporal action language, a domain description can be defined as a pair $(\Pi, \mathcal{C})$, consisting of a set of laws Π and a set of temporal constraints $\mathcal{C}$. The following action laws describe the deterministic effect of the actions *shoot* and *load* for the Russian Turkey scenario [29], as well as the nondeterministic effect of action *spin*, after which the gun may be loaded or not:

$\Box([shoot]\neg alive \leftarrow loaded)$ $\quad$ $\Box[load]loaded$

$\Box([spin]loaded \leftarrow \ not\ [spin]\neg loaded)$ $\quad$ $\Box([spin]\neg loaded \leftarrow \ not\ [spin]loaded)$

The following precondition law: $\Box([load]\bot \leftarrow loaded)$ specifies that, if the gun is loaded, $load$ is not executable. The program $(\neg in\_sight?; wait)^*; in\_sight?; load; shoot$ describes the behavior of the hunter who waits for a turkey until it appears and, when it is in sight, loads the gun and shoots. Actions $in\_sight?$ and $\neg in\_sight?$ are test actions (executable if the corresponding literal holds [5]). If the constraint $\langle(\neg in\_sight?; wait)^*; in\_sight?; load; shoot\rangle\top$ is included in $\mathcal{C}$ then all the runs of the domain description which do not start with an execution of the given program will be filtered out. For instance, an extension in which in the initial state the turkey is not in sight and the hunter loads the gun and shoots is not allowed. The temporal language is also well suited to describe causal dependencies among fluents as *static* and *dynamic causal laws* similar to the ones in the action languages $\mathcal{K}$ [24] and $\mathcal{C}^+$ [13]. For instance, referring to the teacher example, the following dynamic causal rule $\bigcirc Teacher(x) \leftarrow \bigcirc Teaches(x, y) \wedge Course(y)$, where $\bigcirc$ is the next operator, means that if y is a course, and x is caused to teach y, then x is caused to be a teacher.

3. Extending a Temporal Action Theory with an $\mathcal{EL}^\perp$ Ontology

The work investigates *extended temporal action theories*, which combine the temporal action logic mentioned above with an $\mathcal{EL}^\perp$ ontology. By exploiting a fragment of the materialization calculus by Krötzsch [30], it can be shown that, for $\mathcal{EL}^\perp$ knowledge bases in normal form [31], the consistency of the action theory extensions with the ontology can be assured by adding to the action theory a set of causal laws and state constraints.

More precisely, an extended temporal action theory is a triple $(K, \Pi, \mathcal{C})$, where $K = (\mathcal{T}, \mathcal{A})$ is an $\mathcal{EL}^\perp$ ontology in normal form, and $(\Pi, \mathcal{C})$ a domain description including a fluent literal for each assertion $C(a)$ $r(a, b)$ in the language of K, where a, b represent individual names in K or auxiliary individual names, as those introduced to encode $\mathcal{EL}^\perp$ inference in Datalog [30]. Although classical negation is not allowed in $\mathcal{EL}^\perp$, we use *explicit negation* [32] to allow negative literals of the form $\neg C(a)$ in the action language to allow for deleting an assertion from a state.

An extension of $(K, \Pi, \mathcal{C})$ is defined as an extension of the action theory $(\Pi, \mathcal{C})$ [15] satisfying all axioms of the ontology K. Informally, each state w in the extension is required to correspond to an $\mathcal{EL}^\perp$ interpretation and to satisfy all inclusion axioms in TBox $\mathcal{T}$. Additionally, the initial state must satisfy all assertions in the ABox $\mathcal{A}$. Under the assumption that the domain description is well-defined, we prove that such states represent $\mathcal{EL}^\perp$ interpretations, provided an additional set of causal laws and constraints $\Pi_K = \Pi_{\mathcal{L}(K)} \cup \Pi_{\mathcal{T}} \cup \Pi_{\mathcal{A}}$ is included in the action theory. The laws $\Pi_{\mathcal{L}(K)}$ are intended to guarantee that any state w of an extension respects the semantics of DL concepts occurring in K. Its definition is based on a fragment of the materialization calculus for $\mathcal{EL}^\perp$, which provides a Datalog encoding of the $\mathcal{EL}^\perp$ ontology. The constraints $\Pi_{\mathcal{T}}$ guarantee that each state satisfies the inclusion axioms in $\mathcal{T}$, and the laws $\Pi_{\mathcal{A}}$ that all assertions in $\mathcal{A}$ are satisfied in the initial state.

Overall, this provides a transformation of the extended action theory $(K, \Pi, \mathcal{C})$ into a new DLTL action theory $(\Pi \cup \Pi_K, \mathcal{C})$, by eliminating the ontology while introducing the set of static causal laws and constraints $\Pi_K = \Pi_{\mathcal{L}(K)} \cup \Pi_{\mathcal{T}} \cup \Pi_{\mathcal{A}}$, intended to exclude those extensions which do not satisfy the axioms in K. For instance, going back to the initial example, a state constraint $\Box(\perp \leftarrow (\exists Teaches.Course)(x), not\ Teacher(x))$ can be included in $\Pi_{\mathcal{T}}$ to assure that the inclusion $\exists Teaches.Course \sqsubseteq Teacher$ in K is not violated. However, this does not allow the action theory to repair from inconsistency after action execution.

4. Repairing from Inconsistencies

Considering an initial state in which *cs1* is a course, John is not a teacher and does not teaches any course, an action $assign(cs1, john)$, of assigning course *cs1* to John, would not be executable as it would lead to an inconsistent state in which John teaches a course but is not a teacher. As observed in [2], when this happens, the action specification can be regarded as being underspecified, as it is not able to capture the dependencies among fluents which occur in the TBox. To guarantee that TBox is satisfied in the new state, causal laws are needed which allow the state to be *repaired*. In the specific case, adding causal law $\Box(\mathit{Teacher}(x) \leftarrow \mathit{Teaches}(x, y) \wedge \mathit{Course}(y))$ to Π would suffice to cause $\mathit{Teacher}(x)$ in the resulting state, as an indirect effect of action $\mathit{assign}(\mathit{cs1}, \mathit{john})$. The contrapositives of this causal law may as well be of interest to repair from inconsistencies, although some of them might be unintended.

For $\mathcal{EL}^{\bot}$ knowledge bases in normal form, the set of constraints in $\Pi_{\mathcal{T}}$ can indeed be replaced by a set of repair rules, i.e., a set of causal laws which can be used to recover a consistent state, whenever possible. The work identifies a set of repair rules for each axiom in normal form and sufficient conditions to guarantee that Tbox $\mathcal{T}$ is satisfied by the extensions. The more are the repair causal laws considered, the more is the repair capability and the more are the extensions of the domain description.

5. Conclusions

In this paper we have proposed an approach for reasoning about actions by combining a temporal action logic [5], whose semantics is based on a notion of temporal answer set, and an $\mathcal{EL}^{\bot}$ ontology. It is shown that, for $\mathcal{EL}^{\bot}$ knowledge bases in normal form, the consistency of the action theory extensions with respect to an ontology can be verified by adding to the action theory a set of causal laws and state constraints, by exploiting a fragment of the materialization calculus by Krötzsch [30].

Our semantics for actions, as many of the proposals in the literature, requires that a state provides a complete description of the world and is intended to represent an interpretation of the $\mathcal{EL}^{\bot}$ knowledge base. An alternative approach has been adopted in [33], where a state can provide an incomplete specification of the world. In our approach, an incomplete state could be represented as an epistemic state, to distinguish between what is known to be true (or false) and what is unknown. An epistemic extension of our action logic, based on temporal answer sets, has been developed in [21], and it can potentially be exploited for reasoning about actions with incomplete states also in presence of ontological knowledge. We leave for future work the study of this case, as well as an investigation of ASP approaches for combining temporal reasoning with weighted conditional knowledge bases for lightweight DLs [34].

A preliminary version of the work has been presented in ICLP 2021 workshops [35]; an extended and revised version will appear in [36]. We refer therein for comparisons with related work.

Acknowledgments

Thanks to the anonymous referees for their helpful comments and suggestions. This research is partially supported by INDAM-GNCS Project 2022: LESLIE. It was developed in the context of the European Cooperation in Science & Technology (COST) Action CA17124 Dig4ASP.

References

[1] F. Baader, C. Lutz, M. Milicic, U. Sattler, and F. Wolter. Integrating description logics and action formalisms: First results. In *Proc. AAAI 2005*, pages 572–577, 2005.

[2] F. Baader, M. Lippmann, and H. Liu. Using causal relationships to deal with the ramification problem in action formalisms based on description logics. In *LPAR-17*, pages 82–96, 2010.

[3] Liang Chang, Zhongzhi Shi, Tianlong Gu, and Lingzhong Zhao. A family of dynamic description logics for representing and reasoning about actions. *J. Autom. Reasoning*, 49(1):1–52, 2012.

[4] S. Ahmetaj, D. Calvanese, M. Ortiz, and M. Simkus. Managing change in graph-structured data using description logics. In *Proc AAAI 2014*, pages 966–973, 2014.

[5] L. Giordano, A. Martelli, and D. Theseider Dupré. Reasoning about actions with temporal answer sets. *Theory and Practice of Logic Programming*, 13:201–225, 2013.

[6] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In L.P. Kaelbling and A. Saffiotti, editors, *Proc. IJCAI 2005*, pages 364–369, Edinburgh, Scotland, UK, August 2005.

[7] Franz Baader, Hongkai Liu, and Anees ul Mehdi. Verifying properties of infinite sequences of description logic actions. In *ECAI*, pages 53–58, 2010.

[8] N. McCain and H. Turner. A causal theory of ramifications and qualifications. In *Proc. IJCAI 95*, pages 1978–1984, 1995.

[9] F. Lin. Embracing causality in specifying the indirect effects of actions. In *IJCAI 95, Montréal Québec, Canada, August 20-25 1995, 2 Volumes*, pages 1985–1993, 1995.

[10] M. Thielscher. Ramification and causality. *Artif. Intell.*, 89(1-2):317–364, 1997.

[11] M. Denecker, D. Theseider Dupré, and K. Van Belleghem. An inductive definitions approach to ramifications. *Electronic Transactions on Artificial Intelligence*, 2:25–97, 1998.

[12] L. Giordano, A. Martelli, and C. Schwind. Ramification and causality in a modal action logic. *J. Log. Comput.*, 10(5):625–662, 2000.

[13] E. Giunchiglia, J. Lee, V. Lifschitz, N. McCain, , and H. Turner. Nonmonotonic causal theories. *Artificial Intelligence*, 153(1-2):49–104, 2004.

[14] M. Winslett. Reasoning about action using a possible models approach. In *Proc. AAAI, St. Paul, MN, August 21-26*, pages 89–93, 1988.

[15] L. Giordano, A. Martelli, and D. Theseider Dupré. Achieving completeness in bounded model checking of action theories in ASP. In *KR 2012, Rome, Italy, June 10-14, 2012*, 2012.

[16] L. Giordano, A. Martelli, and D. Theseider Dupré. Reasoning about actions with temporal answer sets. In W. Faber and N. Leone, editors, *25th Italian Conference on Computational Logic, Rende, Italy, July 7-9, 2010*, volume 598 of *CEUR Workshop Proceedings*. CEUR-WS.org, 2010.

[17] L. Giordano, A. Martelli, and C. Schwind. Reasoning about actions in dynamic linear time temporal logic. *The Logic Journal of the IGPL*, 9(2):289–303, 2001.

[18] J.G. Henriksen and P.S. Thiagarajan. Dynamic linear time temporal logic. *Annals of Pure and Applied logic*, 96(1-3):187–207, 1999.

[19] M. Gelfond and V. Lifschitz. The stable model semantics for logic programming. In *Logic Programming, Proc. of the 5th Int. Conf. and Symposium*, pages 1070–1080, 1988.

[20] K. Heljanko and I. Niemelä. Bounded LTL model checking with stable models. *Theory and Practice of Logic Programming*, 3(4-5):519–550, 2003.

[21] L. Giordano, A. Martelli, and D. Theseider Dupré. Achieving completeness in the verification of action theories by bounded model checking in ASP. *J. Log. Comp.*, 25(6):1307–30, 2015.

[22] M. Gelfond and V. Lifschitz. Action languages. *Electron. Trans. Artif. Intell.*, 2:193–210, 1998.

[23] C. Baral and M. Gelfond. Reasoning agents in dynamic domains. In *Logic-Based Artificial Intelligence*, pages 257–279. 2000.

[24] T. Eiter, W. Faber, N. Leone, G. Pfeifer, and A. Polleres. A logic programming approach to knowledge-state planning: Semantics and complexity. *ACM TOCL*, 5(2):206–263, 2004.

[25] J. Babb and J. Lee. Cplus2asp: Computing action language $\mathcal{C}$+ in Answer Set Programming. In *Logic Programming and Nonmonotonic Reasoning, LPNMR 2013*, pages 122–134, 2013.

[26] P Cabalar, R. Kaminski, P. Morkisch, and T. Schaub. telingo = ASP + time. In *LPNMR 2019, Philadelphia, USA*, volume 11481 of *LNCS*, pages 256–269. Springer, 2019.

[27] P. Cabalar and G. Pérez Vega. Temporal equilibrium logic: A first approach. In *Computer Aided Systems Theory - EUROCAST 2007, 11th International Conference on Computer Aided Systems Theory*, volume 4739 of *LNCS*, pages 241–248. Springer, 2007.

[28] F.Aguado, P. Cabalar, M. Diéguez, G. Peréz, T. Schaub, A. Schuhmann, and C. Vidal. Linear-time temporal answer set programming. *TPLP*, pages 1–55, 2021.

[29] E. Sandewall. *Features and Fluents: The Representation of Knowledge About Dynamical Systems.* Oxford University Press, 1994.

[30] M. Krötzsch. Efficient inferencing for OWL EL. In *Proc. JELIA 2010*, pages 234–246, 2010.

[31] F. Baader, S. Brandt, and C. Lutz. Pushing the $\mathcal{EL}$ envelope. In *LTCS-Report LTCS-05-01*, TU Dresden, 2005. Inst. for Theoretical Computer Science.

[32] M. Gelfond. *Handbook of Knowledge Representation, ch. 7, Answer Sets.* Elsevier, 2007.

[33] B. Bagheri Hariri, D. Calvanese, M. Montali, G. De Giacomo, R. De Masellis, and P. Felli. Description logic knowledge and action bases. *J. Artif. Intell. Res.*, 46:651–686, 2013.

[34] L. Giordano and D. Theseider Dupré. An ASP approach for reasoning on neural networks under a finitely many-valued semantics for weighted conditional knowledge bases. 2022. To appear in ICLP 2022, a preliminary version in https://arxiv.org/abs/2202.01123.

[35] L. Giordano, A. Martelli, and D. Theseider Dupré. Reasoning about actions with $\mathcal{EL}$ ontologies and temporal answer sets. In *ICLP 2021 Workshops, CEUR Workshop Proc., vol. 2970*, 2021.

[36] L. Giordano, A. Martelli, and D. Theseider Dupré. Reasoning about actions with $\mathcal{EL}$ ontologies and temporal answer sets for DLTL. 2022. To appear in LPNMR 2022.