Knowledge Forgetting in Propositional μ-calculus

First Author[1,2*], Second Author[2,3†] and Third Author[1,2†]

[1*]Department, Organization, Street, City, 100190, State, Country.
[2]Department, Organization, Street, City, 10587, State, Country.
[3]Department, Organization, Street, City, 610101, State, Country.

*Corresponding author(s). E-mail(s): iauthor@gmail.com;
Contributing authors: iiauthor@gmail.com; iiiauthor@gmail.com;
†These authors contributed equally to this work.

Abstract

The $\boldsymbol{\mu}$-calculus is one of the most important logics describing specifications of transition systems. It has been extensively explored for formal verification in model checking due to its exceptional balance between expressiveness and algorithmic properties. From the perspective of systems/knowledge evolving, one may want to discard some information content in a specification that become irrelevant or unnecessary; one may also need a (weakest) precondition for a system to enjoy some desire properties. This paper is to address these scenarios for $\boldsymbol{\mu}$-calculus in a principle way in terms of knowledge *forgetting*. In particular, it proposes a notion of forgetting by a generalized bisimulation and explores the semantic and logical properties of forgetting, besides some reasoning complexity results. It also shows that the forgetting can be employed to compute the weakest preconditions and to present knowledge update.

Keywords: $\boldsymbol{\mu}$-calculus, Forgetting, Weakest precondition, Knowledge update

1 Introduction

Propositional μ-calculus consists essentially of propositional modal logic with a least fixpoint operator. While it is as expressive as the monadic second-order logic of two successors (S2S) on binary trees [1, 2], it enjoys the small-model property. The exceptional balance between expressiveness and algorithmic properties results in efficient and successful automatic verification (model checking) of livenss, fairness and safety for concurrent systems [3].

From the perspective of systems/knowledge evolving, some information content of a (concurrent) system may become irrelevant due to various reasons, e.g., it might become obsolete by time, or perhaps infeasible due to practical difficulties. It is usually a non-trivial task to keep a system update by discarding or eliminating such information content from the system. To redesign a system from scratch is undesirable when it evolves from another one since it is usually expensive and tedious to design a system meeting given requirements. It is also a challenge to find a (weakest) condition for a system to enjoy some desirable properties (under some restrictions). For instance, when a system $\mathcal{M}$ does not have the property φ, how can one find a (weakest) restriction of $\mathcal{M}$ under which the property φ holds? This is the well-known weakest preconditions [4].

This paper is to address the above scenarios for μ-calculus in a principle way in terms of forgetting, which is deeply rooted in artificial intelligence (AI) and formal logic (with the well-known notion of uniform interpolation). Informally, *knowledge forgetting* is to discard all of the information content over a given signature, or alternatively to extract all of information content over some signature. In this way, a logical approach is at hand to dismiss irrelevant information content without changing the behaviour of the associated system or violating the existing system specification under a given signature. In addition, it also provides a logical way to find a (weakest) precondition (named *weakest sufficient condition* in AI jargon) under a given signature.

Indeed, forgetting has been extensively studied in various logical formal systems to deal with abductive reasoning, reasoning under inconsistency, knowledge updating and epistemic planning, including the classical propositional and first-order logic [5–7], (multi-agent) modal logics [8–12], description logics [13–15] and nonmonotonic logics (answer set programming in particular) [16–21].

To our best, none of existing knowledge forgetting is applicable to μ-calculus. The main contributions of the work are as follows:

- We propose a knowledge forgetting for μ-calculus and prove a presentation theorem to characterize the forgetting. Other properties of forgetting are revealed, including modularity, commutativity, homogeneity and reasoning complexities. When forgetting is involved, various reasoning tasks become harder than without forgetting. These results are mostly applicable to uniform interpolation due to its duality with knowledge forgetting.
- We demonstrate that how the knowledge forgetting can be employed as a flexible notion to compute the weakest sufficient conditions and to represent knowledge update in μ-calculus. In particular, we give a knowledge update operator in terms of forgetting that enjoys the Katsuno and Mendelzon's knowledge update postulates [22].

The rest of the paper is organized as follows. After discussing the related work in the next section, the basic notation and technical preliminaries are introduced in Section 3. The formal definition of forgetting in μ-calculus, its various properties, and the computational complexities are presented in Section **??**. Section **??** shows that the forgetting can be used to compute WSC (SNC) and to present knowledge update. Finally, concluding remarks are given in Section **??**.

To avoid hindering the flow of content, detailed proofs of the technical results are provided in the Appendix.

2 Related work

In this section, we briefly discuss published matter that is technically related to our work.

2.1 The Weakest Precondition

The *weakest precondition*, as an important concept in formal verification, was first proposed by Dijkstra to solve the problem of computing or approximating invariants appearing in the *verification of computer programs and systems* [4], particularly in the "Hoare triple" [23]. Afterwards, it was wildly studied in various fields, especially in refining systems [24], reasoning about assembly language programs [25], formulating verification conditions [26], generating counterexamples [27], and so on.

In the field of *artificial intelligence* (AI), there is a similar concept called the *weakest sufficient condition* (WSC), which was introduced by Lin to generate successor state axioms from causal theories (in planning) [6, 28]. Moreover, the SNC and WSC for proposition q on a restricted subset of the propositional variables under propositional theory T are computed based on the notion of forgetting. Afterwards the SNC and WSC were generalized to first-order logic (FOL) and a direct method based on the *second-order quantifier elimination* (SOQE) technique was proposed to automatically generate the SNC and WSC [29]. In addition, a forgetting-based method is used to compute the SNC and WSC in CTL [12].

2.2 Forgetting

Forgetting was first formally defined in PL and FOL by Lin and Reiter [5, 30]. As a technique for distilling knowledge, it has been explored in various of logic languages and widely used in AI. Except for the WSC (SNC), belief update/revision, and knowledge update talked about in the Introduction, forgetting has been used for conflict solving [31, 32] and knowledge compilation [33]. Informally, forgetting is used to abstract from a knowledge base $\mathcal{T}$ only the part that is relevant to a subset of alphabet $\mathcal{P}$ while not affecting the results of $\mathcal{T}$ on $\mathcal{P}$.

The concept of forgetting can be traced back to the work of Boole on *propositional variable elimination* and the seminal work of Ackermann [34], who recognised that the problem amounts to *the elimination of existential second-order quantifiers*. Moreover, it has been extended to various logic systems, including modal logics [10, 11] and nonmonotonic logics [18, 21].

In PL, forgetting has often been studied under the name 'variable elimination'. Formally, the solution of forgetting a propositional variable p from a PL formula φ is $\varphi[p/\bot] \vee \varphi[p/\top]$ [5], where $\varphi[p/\bot]$ and $\varphi[p/\top]$ denote the formulas obtained from φ by replacing atom p with $\bot$ and $\top$, respectively.

In FOL, the definition of forgetting was defined from the perspective of *strong (or semantic) forgetting* and *weak forgetting* [35]. Although weak forgetting and strong forgetting are not exactly the same, they coincide when the result of strong forgetting exists. We consider semantic forgetting (abbreviated to forgetting) and give its definition in PL and FOL here. Forgetting is considered an instance of the SOQE problem in FOL. In that case, the result of forgetting an n-ary predicate P from a first-order formula φ is $\exists R \varphi[P/R]$ [5], in which R is an n-ary predicate variable and $\varphi[X/Y]$ is a result of replacing every occurrence of X in φ by Y. The task of forgetting in FOL, as a computational problem, is to find a first-order formula that is equivalent to $\exists R \varphi[P/R]$. It is evident that this is an SOQE problem. However, the solution to the SOQE problem is not always expressible in FOL [36],which means that the results of forgetting in FOL are not always expressible in FOL, i.e., forgetting in FOL is *not closed*. Nonetheless, the solution of weak forgetting is always expressible in FOL, although there are cases in which the forgetting solution can be represented only by an infinite set of FOL formulas [35]. See [30] for a recent and comprehensive survey.

In non-classical logics, the knowledge forgetting for S5 modal logic was firstly proposed and was used to represent different forms of knowledge updates [10]. More importantly, four general postulates for knowledge forgetting were revealed to precisely characterize both semantic and logical properties of knowledge forgetting, and the dual notion of *uniform interpolation* [37]. These notions were recently extended to multi-agent modal logics [11] and CTL [12]. The forgetting in description logics (DL) are also explored with the motivation of constructing restricted ontologies by eliminating concept and role symbols from DL-based ontologies [14, 15, 38–40]. In the scenario of non-monotonic reasoning, forgetting in logic programs under answer set semantics has been extensively investigated from the perspective of various forgetting postulates [16–18, 21, 31, 41, 42], see [30, 43] for a comprehensive survey.

One should note that the modal μ-calculus enjoys the *uniform interpolation* property [44]. We will show that the uniform interpolation is indeed the dual notion of our proposed forgetting. Thus, most theoretical results for the forgetting are applicable to uniform interpolation as well, including the four general principles or postulates characterizing this logical forgetting in particular.

3 Preliminaries

In this section, we introduce the technical and notational preliminaries, i.e., the syntax and semantics of μ-calculus, closely related to this paper. Moreover, throughout this paper, we denote by $\overline{V}$ the complement of $V \subseteq B$ on a given set B, i.e., $\overline{V} = B - V$.

3.1 The syntax of μ-calculus

Modal μ-calculus is an extension of modal logic, and we consider the propositional μ-calculus introduced by Kozen [45]. Let $\mathcal{A} = \{p, q, \ldots\}$ be a set of propositional letters (atoms) and $\mathcal{V} = \{X, Y, \ldots\}$ be a set of variables. Then, the formulas of the μ-calculus, called μ-formulas (or formulas), over these sets can be inductively defined in Backus-Naur form:

$$\varphi := p \quad \neg p \quad X \quad \varphi \vee \varphi \quad \varphi \wedge \varphi \quad \text{EX}\varphi \quad \text{AX}\varphi \quad \mu X.\varphi \quad \nu X.\varphi$$

where $p \in \mathcal{A}$ and $X \in \mathcal{V}$. $\top$ and $\bot$ are also μ-calculus formulas, which express 'true' and 'false', respectively.

4 Results

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

5 This is an example for first level head—section head

5.1 This is an example for second level head—subsection head

5.1.1 This is an example for third level head—subsubsection head

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

6 Equations

7 Examples for theorem like environments

For theorem like environments, we require `amsthm` package. There are three types of predefined theorem styles exists—`thmstyleone`, `thmstyletwo` and `thmstylethree`

`thmstyleone`	Numbered, theorem head in bold font and theorem text in italic style
`thmstyletwo`	Numbered, theorem head in roman font and theorem text in italic style
`thmstylethree`	Numbered, theorem head in bold font and theorem text in roman style

For mathematics journals, theorem styles can be included as shown in the following examples:

Theorem 1 (Theorem subhead) *Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text. Example theorem text.*

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

Proposition 2 *Example proposition text. Example proposition text. Example proposition text. Example proposition text. Example proposition text. Example proposition text. Example proposition text. Example proposition text. Example proposition text. Example proposition text.*

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

Example 1 Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem.

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

Remark 1 Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem.

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

Definition 1 (Definition sub head) Example definition text. Example definition text. Example definition text. Example definition text. Example definition text. Example definition text. Example definition text. Example definition text.

Additionally a predefined "proof" environment is available: `\begin{proof} ... \end{proof}`. This prints a "Proof" head in italic font style and the "body text" in roman font style with an open square at the end of each proof environment.

Proof Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. □

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text. Sample body text.

Proof of Theorem 1 Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. Example for proof text. □

For a quote environment, use `\begin{quote}...\end{quote}`

> Quoted text example. Aliquam porttitor quam a lacus. Praesent vel arcu ut tortor cursus volutpat. In vitae pede quis diam bibendum placerat. Fusce elementum convallis neque. Sed dolor orci, scelerisque ac, dapibus nec, ultricies ut, mi. Duis nec dui quis leo sagittis commodo.

Sample body text. Sample body text. Sample body text. Sample body text. Sample body text (refer Figure **??**). Sample body text. Sample body text. Sample body text (refer Table **??**).

8 Methods

9 Discussion

Discussions should be brief and focused. In some disciplines use of Discussion or 'Conclusion' is interchangeable. It is not mandatory to use both. Some journals prefer a section 'Results and Discussion' followed by a section 'Conclusion'. Please refer to Journal-level guidance for any specific requirements.

10 Conclusion

Conclusions may be used to restate your hypothesis or research question, restate your major findings, explain the relevance and the added value of your work, highlight any limitations of your study, describe future directions for research and recommendations.

In some disciplines use of Discussion or 'Conclusion' is interchangeable. It is not mandatory to use both. Please refer to Journal-level guidance for any specific requirements.

Supplementary information. If your article has accompanying supplementary file/s please state so here.

Authors reporting data from electrophoretic gels and blots should supply the full unprocessed scans for key as part of their Supplementary information. This may be requested by the editorial team/s if it is missing.

Please refer to Journal-level guidance for any specific requirements.

Acknowledgments. Acknowledgments are not compulsory. Where included they should be brief. Grant or contribution numbers may be acknowledged.

Please refer to Journal-level guidance for any specific requirements.

Declarations

Some journals require declarations to be submitted in a standardised format. Please check the Instructions for Authors of the journal to which you are submitting to see if you need to complete this section. If yes, your manuscript must contain the following sections under the heading 'Declarations':

- Funding
- Conflict of interest/Competing interests (check journal-specific guidelines for which heading to use)
- Ethics approval
- Consent to participate
- Consent for publication
- Availability of data and materials
- Code availability
- Authors' contributions

If any of the sections are not relevant to your manuscript, please include the heading and write 'Not applicable' for that section.

Editorial Policies for:

Springer journals and proceedings:
https://www.springer.com/gp/editorial-policies

Nature Portfolio journals:
https://www.nature.com/nature-research/editorial-policies

Scientific Reports:
https://www.nature.com/srep/journal-policies/editorial-policies

BMC journals:
https://www.biomedcentral.com/getpublished/editorial-policies

Appendix A Section title of first appendix

An appendix contains supplementary information that is not an essential part of the text itself but which may be helpful in providing a more comprehensive understanding of the research problem or it is information that is too cumbersome to be included in the body of the paper.

References

[1] Emerson, E.A., Jutla, C.S.: Tree automata, μ-calculus and determinacy (extended abstract). In: 32nd Annual Symposium on Foundations of Computer Science, San Juan, Puerto Rico, 1-4 October 1991, pp. 368–377. IEEE Computer Society, ??? (1991). https://doi.org/10.1109/SFCS.1991.185392. https://doi.org/10.1109/SFCS.1991.185392

[2] Niwinski, D.: Fixed points vs. infinite generation. In: Proceedings of the Third Annual Symposium on Logic in Computer Science (LICS '88), Edinburgh, Scotland, UK, July 5-8, 1988, pp. 402–409. IEEE Computer Society, ??? (1988). https://doi.org/10.1109/LICS.1988.5137. https://doi.org/10.1109/LICS.1988.5137

[3] Emerson, E.A.: Model checking and the mu-calculus. In: Immerman, N., Kolaitis, P.G. (eds.) Descriptive Complexity and Finite Models, Proceedings of a DIMACS Workshop 1996, Princeton, New Jersey, USA, January 14-17, 1996. DIMACS Series in Discrete Mathematics and Theoretical Computer Science, vol. 31, pp. 185–214. DIMACS/AMS, ??? (1996). https://doi.org/10.1090/dimacs/031/06. https://doi.org/10.1090/dimacs/031/06

[4] Dijkstra, E.W.: Guarded commands, Nondeterminacy and Formal Derivation of Programs. Commun. ACM **18**(8), 453–457 (1975). https://doi.org/10.1145/360933.360975

[5] Lin, F., Reiter, R.: Forget it! In: In Proceedings of the AAAI Fall Symposium on Relevance, New Orleans, US, pp. 154–159 (1994)

[6] Lin, F.: On strongest necessary and weakest sufficient conditions. Artif. Intell. **128**(1-2), 143–159 (2001). https://doi.org/10.1016/S0004-3702(01)00070-4

[7] Lang, J., Liberatore, P., Marquis, P.: Propositional independence: Formula-variable independence and forgetting. J. Artif. Intell. Res. **18**, 391–443 (2003). https://doi.org/10.1613/jair.1113

[8] Su, K., Lv, G., Zhang, Y.: Reasoning about knowledge by variable forgetting. In: Dubois, D., Welty, C.A., Williams, M. (eds.) Principles of Knowledge Representation and Reasoning: Proceedings of the Ninth

International Conference (KR2004), Whistler, Canada, June 2-5, 2004, pp. 576–586. AAAI Press, ??? (2004). http://www.aaai.org/Library/KR/2004/kr04-060.php

[9] Baral, C., Zhang, Y.: Knowledge updates: Semantics and complexity issues. Artif. Intell. **164**(1-2), 209–243 (2005). https://doi.org/10.1016/j.artint.2005.01.005

[10] Zhang, Y., Zhou, Y.: Knowledge forgetting: Properties and applications. Artif. Intell. **173**(16-17), 1525–1537 (2009). https://doi.org/10.1016/j.artint.2009.07.005

[11] Fang, L., Liu, Y., Van Ditmarsch, H.: Forgetting in multi-agent modal logics. Artificial Intelligence **266**, 51–80 (2019)

[12] Feng, R., Acar, E., Schlobach, S., Wang, Y., Liu, W.: On sufficient and necessary conditions in bounded CTL: A forgetting approach. In: Calvanese, D., Erdem, E., Thielscher, M. (eds.) Proceedings of the 17th International Conference on Principles of Knowledge Representation and Reasoning, KR 2020, Rhodes, Greece, September 12-18, 2020, pp. 361–370 (2020). https://doi.org/10.24963/kr.2020/37. https://doi.org/10.24963/kr.2020/37

[13] Konev, B., Walther, D., Wolter, F.: Forgetting and uniform interpolation in large-scale description logic terminologies. In: IJCAI 2009, Proceedings of the 21st International Joint Conference on Artificial Intelligence, Pasadena, California, USA, pp. 830–835 (2009)

[14] Lutz, C., Wolter, F.: Foundations for uniform interpolation and forgetting in expressive description logics. In: Walsh, T. (ed.) IJCAI 2011, Proceedings of the 22nd International Joint Conference on Artificial Intelligence, Barcelona, Catalonia, Spain, July 16-22, 2011, pp. 989–995. IJCAI/AAAI, ??? (2011). https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-170. https://doi.org/10.5591/978-1-57735-516-8/IJCAI11-170

[15] Zhao, Y., Schmidt, R.A., Wang, Y., Zhang, X., Feng, H.: A practical approach to forgetting in description logics with nominals. In: The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020, pp. 3073–3079 (2020). https://aaai.org/ojs/index.php/AAAI/article/view/5702

[16] Eiter, T., Wang, K.: Semantic forgetting in answer set programming. Artificial Intelligence **172**(14), 1644–1672 (2008). https://doi.org/10.1016/j.artint.2008.05.002

[17] Wang, Y., Wang, K., Zhang, M.: Forgetting for answer set programs revisited. In: Rossi, F. (ed.) IJCAI 2013, Proceedings of the 23rd International Joint Conference on Artificial Intelligence, Beijing, China, August 3-9, 2013, pp. 1162–1168. IJCAI/AAAI, ??? (2013). http://www.aaai.org/ocs/index.php/IJCAI/IJCAI13/paper/view/6807

[18] Wang, Y., Zhang, Y., Zhou, Y., Zhang, M.: Knowledge forgetting in answer set programming. J. Artif. Intell. Res. **50**, 31–70 (2014). https://doi.org/10.1613/jair.4297

[19] Wang, Y., Wang, K., Wang, Z., Zhuang, Z.: Knowledge forgetting in circumscription: A preliminary report. In: Bonet, B., Koenig, S. (eds.) Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA., pp. 1649–1655. AAAI Press, ??? (2015). http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9866

[20] Delgrande, J.P., Wang, K.: A syntax-independent approach to forgetting in disjunctive logic programs. In: Bonet, B., Koenig, S. (eds.) Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA., pp. 1482–1488. AAAI Press, ??? (2015). http://www.aaai.org/ocs/index.php/AAAI/AAAI15/paper/view/9555

[21] Gonçalves, R., Knorr, M., Leite, J., Woltran, S.: On the limits of forgetting in answer set programming. Artif. Intell. **286**, 103307 (2020). https://doi.org/10.1016/j.artint.2020.103307

[22] Katsuno, H., Mendelzon, A.O.: On the difference between updating a knowledge base and revising it. In: Allen, J.F., Fikes, R., Sandewall, E. (eds.) Proceedings of the 2nd International Conference on Principles of Knowledge Representation and Reasoning (KR'91). Cambridge, MA, USA, April 22-25, 1991, pp. 387–394. Morgan Kaufmann, ??? (1991)

[23] Hoare, C.A.R.: An axiomatic basis for computer programming. Commun. ACM **12**(10), 576–580 (1969). https://doi.org/10.1145/363235.363259

[24] Woodcock, J., Morgan, C.: Refinement of state-based concurrent systems. In: Bjørner, D., Hoare, C.A.R., Langmaack, H. (eds.) VDM '90, VDM and Z - Formal Methods in Software Development, Third International Symposium of VDM Europe, Kiel, FRG, April 17-21, 1990, Proceedings. Lecture Notes in Computer Science, vol. 428, pp. 340–351. Springer, ??? (1990). https://doi.org/10.1007/3-540-52513-0_18. https://doi.org/10.1007/3-540-52513-0_18

[25] Legato, W.J.: A weakest precondition model for assembly language programs. April (2002)

[26] Leino, K.R.M.: Efficient weakest preconditions. Inf. Process. Lett. **93**(6), 281–288 (2005). https://doi.org/10.1016/j.ipl.2004.10.015

[27] Dailler, S., Hauzar, D., Marché, C., Moy, Y.: Instrumenting a weakest precondition calculus for counterexample generation. J. Log. Algebraic Methods Program. **99**, 97–113 (2018). https://doi.org/10.1016/j.jlamp.2018.05.003

[28] Lin, F.: Compiling causal theories to successor state axioms and strips-like systems. J. Artif. Intell. Res. **19**, 279–314 (2003). https://doi.org/10.1613/jair.1135

[29] Doherty, P., Lukaszewicz, W., Szalas, A.: Computing strongest necessary and weakest sufficient conditions of first-order formulas. In: Nebel, B. (ed.) Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI 2001, Seattle, Washington, USA, August 4-10, 2001, pp. 145–154. Morgan Kaufmann, ??? (2001)

[30] Eiter, T., Kern-Isberner, G.: A brief survey on forgetting from a knowledge representation and reasoning perspective. Künstliche Intell. **33**(1), 9–33 (2019). https://doi.org/10.1007/s13218-018-0564-6

[31] Zhang, Y., Foo, N.Y.: Solving logic program conflict through strong and weak forgettings. Artif. Intell. **170**(8-9), 739–778 (2006). https://doi.org/10.1016/j.artint.2006.02.002

[32] Lang, J., Marquis, P.: Reasoning under inconsistency: A forgetting-based approach. Artif. Intell. **174**(12-13), 799–823 (2010). https://doi.org/10.1016/j.artint.2010.04.023

[33] Bienvenu, M., Fargier, H., Marquis, P.: Knowledge compilation in the modal logic S5. In: Fox, M., Poole, D. (eds.) Proceedings of the Twenty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2010, Atlanta, Georgia, USA, July 11-15, 2010. AAAI Press, ??? (2010). http://www.aaai.org/ocs/index.php/AAAI/AAAI10/paper/view/1696

[34] Ackermann, W.: Untersuchungen über das eliminationsproblem der mathematischen logik. Mathematische Annalen **110**(1), 390–413 (1935)

[35] Zhang, Y., Zhou, Y.: Forgetting revisited. In: Lin, F., Sattler, U., Truszczynski, M. (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). http://aaai.org/ocs/index.php/KR/KR2010/paper/view/1292

[36] Gabbay, D.M., Schmidt, R.A., Szalas, A.: Second-Order Quantifier Elimination - Foundations, Computational Aspects and Applications. Studies

in logic : Mathematical logic and foundations, vol. 12. College Publications, ??? (2008). http://collegepublications.co.uk/logic/mlf/?00009

[37] Visser, A., *et al.*: Uniform interpolation and layered bisimulation. In: Gödel'96: Logical Foundations of Mathematics, Computer Science and physics—Kurt Gödel's Legacy, Brno, Czech Republic, August 1996, Proceedings, pp. 139–164. Association for Symbolic Logic, ??? (1996)

[38] Wang, Z., Wang, K., Topor, R.W., Pan, J.Z.: Forgetting for knowledge bases in dl-lite. Ann. Math. Artif. Intell. **58**(1-2), 117–151 (2010). https://doi.org/10.1007/s10472-010-9187-9

[39] Konev, B., Ludwig, M., Walther, D., Wolter, F.: The logical difference for the lightweight description logic EL. J. Artif. Intell. Res. **44**, 633–708 (2012). https://doi.org/10.1613/jair.3552

[40] Zhao, Y., Schmidt, R.A.: Role forgetting for alcoqh(universal role)-ontologies using an ackermann-based approach. In: Sierra, C. (ed.) Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI 2017, Melbourne, Australia, August 19-25, 2017, pp. 1354–1361. ijcai.org, ??? (2017). https://doi.org/10.24963/ijcai.2017/188. https://doi.org/10.24963/ijcai.2017/188

[41] Wong, K.-S.: Forgetting in logic programs. PhD thesis, The University of New South Wales (2009)

[42] Delgrande, J.P.: A knowledge level account of forgetting. J. Artif. Intell. Res. **60**, 1165–1213 (2017). https://doi.org/10.1613/jair.5530

[43] Gonçalves, R., Knorr, M., Leite, J.: Forgetting in answer set programming - A survey. CoRR **abs/2107.07016** (2021) https://arxiv.org/abs/2107.07016

[44] D'Agostino, G., Hollenberg, M.: Uniform interpolation, automata and the modal μ-calculus. In: Kracht, M., de Rijke, M., Wansing, H., Zakharyaschev, M. (eds.) Advances in Modal Logic 1, Papers from the First Workshop on "Advances in Modal Logic," Held in Berlin, Germany, 8-10 October 1996, pp. 73–84. CSLI Publications, ??? (1996)

[45] Kozen, D.: Results on the propositional mu-calculus. Theor. Comput. Sci. **27**, 333–354 (1983). https://doi.org/10.1016/0304-3975(82)90125-6

[46] Bonet, B., Koenig, S. (eds.): Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA. AAAI Press, ??? (2015). http://www.aaai.org/Library/AAAI/aaai15contents.php