On the Benefits of Knowledge Compilation for Feature-Model Analyses (Extended Abstract)

Chico Sundermann
University of Ulm
Germany

Elias Kuiter
University of Magdeburg
Germany

Tobias Heß
University of Ulm
Germany

Heiko Raab
University of Ulm
Germany

Sebastian Krieter
Paderborn University
Germany

Thomas Thüm
Paderborn University
Germany

ABSTRACT

In practice, feature models often have up-to thousands of features and, thus, are typically infeasible to analyze manually. Hence, many different analyses have been proposed which often rely on multiple computationally complex computations, such as solving SAT or #SAT problems. Knowledge compilation is the process of compiling the original input into a computationally more advantageous format, such as BDDs. The upfront compilation effort is subsequently amortized by faster computations on the target format. In our work, we examine the benefits of applying knowledge compilation for feature-model analysis with three major contributions. First, we classify existing feature-model analyses regarding the underlying computational problems. Second, we perform a survey to collect available knowledge-compilation languages and inspect their capabilities with respect to the identified feature-model analyses. Third, we gather and empirically evaluate available knowledge compilers on industrial feature models.

CCS CONCEPTS

• **Software and its engineering → Software product lines**; • **Computing methodologies → Knowledge representation and reasoning**.

ACM Reference Format:
Chico Sundermann, Elias Kuiter, Tobias Heß, Heiko Raab, Sebastian Krieter, and Thomas Thüm. 2024. On the Benefits of Knowledge Compilation for Feature-Model Analyses (Extended Abstract). In *28th ACM International Systems and Software Product Line Conference (SPLC '24), September 2–6, 2024, Dommeldange, Luxembourg*. ACM, New York, NY, USA, 1 page. https://doi.org/10.1145/3646548.3676540

EXTENDED ABSTRACT

Feature-model analyses often rely on solving numerous computationally complex computations, such as solving SAT [2] or #SAT [6] problems. This multitude of complex computations on the same feature model motivates the usage of *knowledge compilation* [3]. Here, the feature model is translated to another format (e.g., BDD or d-DNNF [3]) in an offline phase. Then, the computations which were computationally demanding on the original format can be performed faster which amortizes the initial effort of compiling.

Even though various example analyses motivate the usage of knowledge compilation for feature models, the application in practice and research is still limited. First, there has not been a systematic study of underlying computations required for feature-model analyses. Second, while some knowledge compilation languages have been applied in the context of feature model analyses [1, 4], full capabilities for feature-model analyses and other knowledge-compilation languages have not been considered. Third, the scalability of many available knowledge compilers has not been examined but is essential for the applicability of respective languages.

We inspect the benefits of applying knowledge compilation for feature-model analyses with three major contributions:
Potential for Feature-Model Analyses We classify feature-model analyses from the literature w.r.t. their underlying computational problem and provide an estimate of the number of required queries.
Capabilities of Knowledge-Compilation Languages We perform a systematic literature survey to collect available knowledge compilation target languages. For each, we examine which analysis can be performed in polynomial time with that language.
Scalability of Knowledge Compilers With our survey, we also collect tooling for knowledge compilation and evaluate their scalability for industrial feature models.

Original Paper. The original work [5] was published[1] at the *Annals of Mathematics and Artificial Intelligence*-journal in 2023.

REFERENCES

[1] Mathieu Acher, Patrick Heymans, Philippe Collet, Clément Quinton, Philippe Lahire, and Philippe Merle. 2012. Feature Model Differences. In *CAiSE*. Springer, 629–645.
[2] David Benavides, Sergio Segura, and Antonio Ruiz-Cortés. 2010. Automated Analysis of Feature Models 20 Years Later: A Literature Review. *Information Systems* 35, 6 (2010), 615–708.
[3] Adnan Darwiche and Pierre Marquis. 2002. A Knowledge Compilation Map. *JAIR* 17, 1 (2002), 229–264.
[4] Shubham Sharma, Rahul Gupta, Subhajit Roy, and Kuldeep S Meel. 2018. Knowledge Compilation Meets Uniform Sampling. In *Proc. Int'l Conf. on Logic for Programming, Artificial Intelligence, and Reasoning*. EasyChair, 620–636.
[5] Chico Sundermann, Elias Kuiter, Tobias Heß, Heiko Raab, Sebastian Krieter, and Thomas Thüm. 2023. On the Benefits of Knowledge Compilation for Feature-Model Analyses. *AMAI* (2023).
[6] Chico Sundermann, Michael Nieke, Paul Maximilian Bittner, Tobias Heß, Thomas Thüm, and Ina Schaefer. 2021. Applications of #SAT Solvers on Feature Models. In *VaMoS*. ACM, Article 12, 10 pages.

[1] https://link.springer.com/article/10.1007/s10472-023-09906-6

SPLC '24, September 2–6, 2024, Dommeldange, Luxembourg

ACM ISBN 979-8-4007-0593-9/24/09.
https://doi.org/10.1145/3646548.3676540