Abstract: Uniform sampling of SAT Solutions for Configurable Systems: Are We There Yet?

Mathieu Acher*, Gilles Perrouin†, Xavier Devroey‡ and Maxime Cordy§
*Univ Rennes, Inria, CNRS, IRISA, Rennes, France. Emails: quentin.plazar@gmail.com, mathieu.acher@irisa.fr
†PReCISE/NaDI, Faculty of Computer Science, University of Namur, Namur, Belgium. Email: gilles.perrouin@unamur.be
‡Delft University of Technology, Delft, The Netherlands. Email: x.d.m.devroey@tudelft.nl
§SnT, University of Luxembourg, Luxembourg, Luxembourg. Email: maxime.cordy@uni.lu

I. Reference

This paper appeared in the proceedings of the 12th IEEE International Conference on Software Testing Verification and Validation, held in X'ian (China), April 22-27, 2019 [1].

II. Introduction

Configurable systems form a vast and heterogeneous class of software systems that encompasses: *Software Product Lines* [2], operating systems kernels, web development frameworks/stacks, e-commerce configurators, code generators, *etc.*. While being very different in their goals and implementations, configurable systems see their behaviour affected by the activation or deactivation of one or more *feature(s)*, i.e. units of variability, configuration *options*. Configurable systems may involve thousands of features with complex dependencies. To model those, we use *feature models* [3] a simple model with a formal semantics. One important issue when testing configurable systems is that the number of variants grows exponentially with the number of features, preventing the exhaustive enumeration of all valid configurations authorized by the feature model in most cases. Therefore, a simple strategy to cope with this issue is to *sample* configurations of interest before testing the corresponding variants. Many sampling strategies exist, from combinatorial interaction testing [4] to distance-based heuristics [5]. Most of these sampling techniques share two important characteristics: *i)* they use various degrees of randomness to explore the configuration space and *ii)* they use SAT solvers to check the validity of the sampled configurations. In this work, we investigate the suitability of random uniform sampling tools UniGen [6] and QuickSampler [7] to test configurable systems.

III. Contributions

1) An empirical assessment of UniGen and QuickSampler on boolean formulas stemming from 128 feature models of various sizes and taken from real-world benchmarks [8], [9]. Some of them exhibit more than 10,000 features.
2) We show that conventional methods for evaluating the uniformity of a sample distribution are not appropriate. Accordingly, we propose an adapted new method of assessing sample distributions based on the frequency of features compared to the ground truth.
3) A presentation of the results showing that while QuickSampler scales on the most significant models, sample distributions deviate from up to 800% from a uniform distribution. Furthermore, UniGen guarantees uniformity to the price of scalability: it cannot process feature model models of more than 1,000 features.
4) A first assessment of the ability of QuickSampler to cover buggy features on JHipster [10], a configurable web development stack. QuickSampler deviations can either over-represent features involved in rare bugs and ignore ones involved in frequent ones.

Gilles Perrouin is a research associate at the FNRS. This research was partially funded by the EU Project STAMP ICT-16-10 No.731529, the FNRS Grant O05518F-RG03, and the ANR-17-CE25-0010-01 VaryVary project. This work was done when Maxime Cordy worked at the University of Namur.

IV. Current and Future Work

We are currently experimenting with improved and new uniform sampling algorithms that show improvement on some formulas but still fail on the largest cases. Future work include the design of new heuristics and the assessment of uniform sampling to improve fault-finding abilities of test strategies that use it.

References

[1] Q. Plazar, M. Acher, G. Perrouin, X. Devroey, and M. Cordy, "Uniform sampling of SAT solutions for configurable systems: Are we there yet?" in *ICST*, 2019, pp. 240–251.
[2] K. Pohl, G. Böckle, and F. J. van der Linden, *Software Product Line Engineering: Foundations, Principles and Techniques*. Springer-Verlag, 2005.
[3] K. Kang, S. Cohen, J. Hess, W. Novak, and S. Peterson, "Feature-Oriented Domain Analysis (FODA)," SEI, Tech. Rep. CMU/SEI-90-TR-21, Nov. 1990.
[4] M. Cohen, M. Dwyer, and Jiangfan Shi, "Constructing Interaction Test Suites for Highly-Configurable Systems in the Presence of Constraints: A Greedy Approach," *IEEE TSE*, vol. 34, no. 5, pp. 633–650, 2008.
[5] C. Henard, M. Papadakis, G. Perrouin, J. Klein, P. Heymans, and Y. L. Traon, "Bypassing the combinatorial explosion: Using similarity to generate and prioritize t-wise test configurations for software product lines," *IEEE Trans. Software Eng.*, 2014.
[6] S. Chakraborty, D. J. Fremont, K. S. Meel, S. A. Seshia, and M. Y. Vardi, "On parallel scalable uniform SAT witness generation," in *TACAS'15*, 2015, pp. 304–319.
[7] R. Dutra, K. Laeufer, J. Bachrach, and K. Sen, "Efficient sampling of SAT solutions for testing," in *ICSE*, 2018, pp. 549–559.
[8] J. H. Liang, V. Ganesh, K. Czarnecki, and V. Raman, "Sat-based analysis of large real-world feature models is easy," in *SPLC '15*, ser. SPLC '15. New York, NY, USA: ACM, 2015, pp. 91–100.
[9] A. Knüppel, T. Thüm, S. Mennicke, J. Meinicke, and I. Schaefer, "Is there a mismatch between real-world feature models and product-line research?" in *ESEC/FSE*, 2017, pp. 291–302.
[10] M. Raible, *The JHipster mini-book*. C4Media, 2015.