Propagating Configuration Decisions with Modal Implication Graphs

Sebastian Krieter[1,2], Thomas Thüm[3], Sandro Schulze[1], Reimar Schröter[1], Gunter Saake[1]

Abstract: This work was originally published as "Propagating Configuration Decisions with Modal Implication Graphs" at the 40th International Conference on Software Engineering 2018 [Kr18].

Keywords: Feature Modeling, Configuration, Software Product Lines, Configurable Systems

Extended Abstract

Modern software systems often comprise thousands of interdependent configuration options, also known as features. Due to the large number of features and their arbitrarily complex interdependencies, configuring and analyzing these highly-configurable systems poses real challenges to developers and users. To properly configure such systems, all dependencies between features must be considered, which requires a non-trivial configuration process that can become tedious and error-prone. Hence, for highly-configurable systems tool support becomes necessary to help users in their decision-making process.

Usually, configuring a system is an iterative and interactive process, in which users decide for one feature at a time, whether it is selected or deselected in the current configuration. An effective mechanism for supporting users during this configuration process is *decision propagation*, which calculates the implications of a user's decision immediately. For instance, the selection of a feature can lead to the mandatory selection or deselection of other features. By applying decision propagation, users are directly informed about the consequences of their decisions. Furthermore, users are not able to introduce conflicting decisions to the configuration.

In our paper [Kr18], we propose a data structure, the *modal implication graph (MIG)*, to improve the performance of current concepts for decision propagation. MIGs represent the feature dependencies of a system and provide efficient access to them via graph traversal algorithms. MIGs are an extensions of regular implication graphs, which can be used to represent binary relations between features (e.g., requires and excludes). In a MIG, we call

[1] University of Magdeburg, Germany
[2] Harz University of Applied Sciences Wernigerode, Germany
[3] TU Braunschweig, Germany

doi:10.18420/se2019-20

these binary relations *strong* edges. We introduce a new type of connection, called *weak* edges that indicate a non-binary relationship between three or more features. We show an example of a MIG in Fig. 1 on the right. On the left, we depict a corresponding feature diagram, which is commonly used to express dependencies between features.

In the worst case, a decision propagation algorithm has to examine every non-configured feature and determine whether its selection or deselection is implied by the user's most recent decision. We use MIGs to reduce the number of features that have to be examined during this process. By traversing through the graph, we can divide all non-configured features into three distinct groups, features that are not, strongly-, or weakly-connected to the current user decision. This increases the overall performance, as only the weakly-connected features must be inspected further. In our evaluation with 120 real-world systems from different domains, we confirm a significant increase in efficiency, when using a MIG for decision propagation.

Though, originally, decision propagation was proposed for an interactive configuration process, it can be applied to other domains as well, which in turn makes our approach also applicable to these domains. For instance, decision propagation can be used in calculating t-wise configuration samples for testing or benchmarking purposes [Al16]. Furthermore, MIGs can support other processes in addition to decision propagation. Due to the efficient access to dependencies between certain features, it can speed up other important analyses such as finding atomic sets, false-optional features, and redundant constraints.

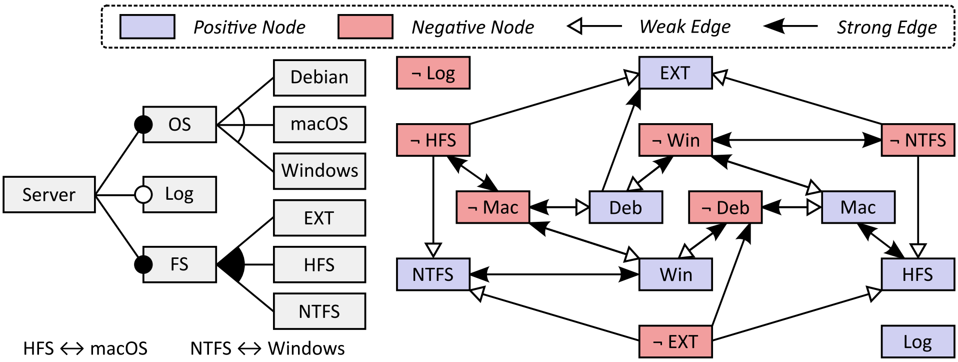

Fig. 1: Feature model (left) and its representation as modal implication graph (right).

References

[Al16] Al-Hajjaji, Mustafa; Krieter, Sebastian; Thüm, Thomas; Lochau, Malte; Saake, Gunter: IncLing: Efficient Product-Line Testing Using Incremental Pairwise Sampling. In: ACM SIGPLAN Notices. volume 52. ACM, pp. 144–155, 2016.

[Kr18] Krieter, Sebastian; Thüm, Thomas; Schulze, Sandro; Schröter, Reimar; Saake, Gunter: Propagating Configuration Decisions with Modal Implication Graphs. In: Proceedings of the 40th International Conference on Software Engineering. ICSE '18. ACM, pp. 898–909, 2018.