DFA Inference using Community Detection

Tommaso Brandirali (tbrandirali@student.tudelft.nl), Annibale Panichella, Mitchell Olsthoorn ({a.panichella, m.j.g.olsthoorn}@tudelft.nl)

1. The Problem:

Automatic generation of high-level system models for the analysis of software behavior.

Deterministic Finite Automata (**DFAs**) are a model of computation commonly used for system modeling.

Current Solutions:

- *Profiling* -> performance impact
- *Tracing* -> performance impact, low-level data
- *Log Interpretation* -> **currently** not scalable

The Goal:

Improving scalability of log interpretation-based system modeling.

How? Using Graph-based algorithms.

2. The Approach:

We assume that the topology of a state graph built from log traces encodes meaningful information about the system behavior.

1. Use log traces to build a naive model of the dataset: the *Prefix Tree.*
2. Use Community Detection algorithms to generate a **clustering** of the Prefix Tree*.
3. **Merge** nodes in the Prefix Tree following the clustering until possible or until desirable results.

   Attempt determinization after each merge.

* We use the inverse of transition probabilities as distance between nodes: if states often occur in sequence, they are close.

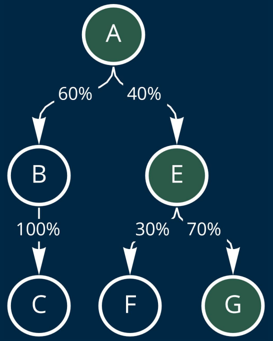

Figure 1: An example Prefix Tree, green nodes represent the matching path for the trace “AEG”

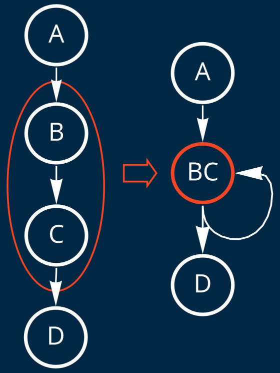

Figure 2: An example of a merge

3. The Results:

Evaluation was performed on a dataset of log traces from the **XRP Ledger Consensus Protocol [1]**, a blockchain-based payment protocol.

- Size reduction was limited to less than 2% by divergence from the clustering due to the determinization.
- Generated models did not generalize to unseen traces.
- Time required for the evaluation of models grows exponentially with each non-deterministic transition.

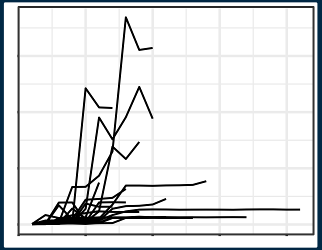

Figure 3: Runtime for evaluation depends heavily on the model’s unique topology, making it highly unpredictable.

Measured runtimes varied by 2 orders of magnitude for models with same parameters.

4. Some Reflections:

During implementation some challenges could not be overcome and require further work:

- Full determinization after merges requires an algorithm which is not yet developed for cyclical graphs.
- The clustering algorithm is not made for DFAs and its implementation must be customized for the purpose.

The sample sizes used in evaluation were too small for conclusive results, *further research is required to fully validate the assumptions of our approach.*

1. Chase, B., & MacBrough, E. (2018). Analysis of the XRP ledger consensus protocol. *arXiv preprint arXiv:1802.07242.*