



Simplifying Reactive Programs using E-Graphs

Advisor: **Benedikt Maderbacher**

Motivation

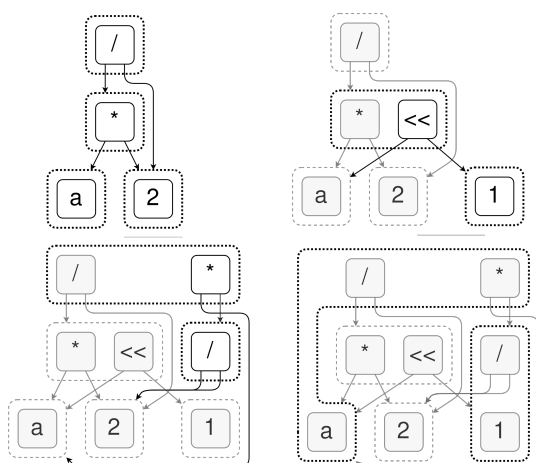
Program synthesis can be used to automatically generate a program from a formal specification. However, the resulting programs are often much more verbose than necessary and thus hard to understand. In his project we want to explore how to simplify these programs. This should result in an equivalent, but smaller and more readable versions of the same program.

One promising technique is to use e-graphs and equality saturation. E-graphs are a data structure to efficiently store multiple, even infinitely many, equivalent programs. Equality saturation starts with an e-graphs representing one program and repeatedly applies rewrite rules to build an e-graph that contains all equivalent programs. At the end an extraction procedure can be used to get the smallest program out of the e-graph.

In this project we want to apply the e-graph and equality saturation library `egg` and apply it to the problem of simplifying reactive programs.

Goals and Tasks

- > Understand e-graphs and `egg`.
- > Implement a simplifier for reactive programs.
- > Evaluate the effectiveness of your simplifier.



Literature

- > [M. Willsey et al.](#)
egg: Fast and extensible equality saturation
Proc. ACM Program. Lang. 2021

Courses & Deliverables

- Introduction to Scientific Working**
Short report on background
Short presentation
- Bachelor Project**
Project code and documentation
- Bachelor's Thesis**
Project code
Thesis
Final presentation

Recommended if you're studying

- CS
- ICE
- SEM

Prerequisites

- > Interest in logic and programming languages.

Advisor Contact

benedikt.maderbacher@tugraz.at