

Verification-Driven Sound Refactoring for Readability

Jarred Bettencourt & John Broderick

March 19, 2023

1 Research Question(s)

The research questions which our project aims to answer are listed below, ordered in terms of decreasing importance.

1. Can the automated process of altering code by generating mutants, verifying them for correctness, and then choosing the mutant based on a readability heuristic result in code with equivalent correctness but higher readability?
2. Is it feasible (with regards to human and computational factors) to implement our proposed workflow into an industrial or academic workflow? What advantages and disadvantages are introduced with our project included into a typical workflow?
3. What other metrics (maintainability, complexity, etc.) would be useful to try to optimize for rather than readability?
4. Does our proposed process to improve readability work better or worse on certain projects? For instance, does it work better on large or small projects?

2 Key-Idea

The main idea of our project is to perform Verification-Driven Sound Refactoring, but with an emphasis on assessing the readability of the generated mutants, given that the mutants are behaviorally equivalent to the original code. As a group that is interested in open source software, we would like to help ensure that readability of software (especially software with a large sense of scale) remains high, which lowers the bar for new contributors who want to learn a code base. To do this, we will choose a mutation engine, a method to prove equivalence of programs, an adequate test suite, a readability heuristic, and 5-10 open source projects to test our findings with.

3 Evaluation

Our evaluation step consists of multiple steps, which all concern themselves with different aspects of our project.

1. We will first develop our project, which includes picking a suitable mutation program, a suitable test suite, and an adequate method to verify behavior equivalence of the mutant to the original program. All of these operations will be able to be done with one command.
2. Additionally, we will decide on a satisfactory heuristic for gauging readability of a particular code segment.
3. We will then choose around 5-10 reputable open source github projects with varying degrees of characteristics (project size, commit history, number of contributors, etc.)
4. Finally, the project will be run on a subset of the files in the aforementioned projects, and we will determine if introducing our readability heuristic did in fact improve the readability of the files we applied it to.