ATR: Template-Based Repair for Alloy Specifications

Guolong Zheng[1], ThanhVu Nguyen[5], Simón Gutiérrez Brida[2,4], Germán Regis[2], Nazareno Aguirre[2,4], Marcelo F. Frias[3,4], and Hamid Bagheri[1]

[1] University of Nebraska-Lincoln, USA
[2] Universidad Nacional de Río Cuarto, Argentina
[3] Instituto Tecnológico de Buenos Aires, Argentina
[4] Consejo Nacional de Investigaciones Científicas y Técnicas (CONICET), Argentina
[5] George Mason University, USA

Abstract

Automatic Program Repair (APR) is a practical research topic that studies techniques to automatically repair programs to fix bugs. Most existing APR techniques are designed for imperative programming languages, such as C and Java, and rely on analyzing correct and incorrect executions of programs to identify and repair suspicious statements.

We introduce a new APR approach for software specifications written in the Alloy declarative language, where specifications are not "executed", but rather converted into logical formulas and analyzed using backend constraint solvers, to find specification instances and counterexamples to assertions. We present ATR, a technique that takes as input an Alloy specification with some violated assertion and returns a repaired specification that satisfies the assertion. The key ideas are (i) analyzing the differences between counterexamples that do not satisfy the assertion and instances that do satisfy the assertion to guide the repair and (ii) generating repair candidates from specific templates and pruning the space of repair candidates using the counterexamples and satisfying instances. Experimental results using existing large Alloy benchmarks show that ATR is effective in generating complex repairs. ATR repairs 66.3% of 1974 fault specifications, including specification repairs that cannot be handled by existing Alloy repair techniques. ATR and all benchmarks are open-source and available in the following Github repository: `https://github.com/guolong-zheng/atmprep`.

This work was accepted for publication at *31st ACM SIGSOFT International Symposium on Software Testing and Analysis (ISSTA 2022)* on July 2022.