

Theory and Practice of SMT Solving - Seminar report

Luís Felipe Ramos Ferreira

December 23, 2024

The paper presents a novel tool called Z3-Noodler, which is a fork of the Z3 SMT theorem prover where the main string solver is substituted by Noodler, a string theory solver based on the *stabilization-based procedure* for solving string formulae. Since the procedure depends on the usage of nondeterministic finite automata (NFA), the authors used a C++ library called Mata for handling operations over automata.

Since the solver was developed over Z3, several tools inside its environment were used, such as the SMT-LIB parser, formula preprocessing, linear integer arithmetic solver, SAT solver, and also the whole DPLL(\mathcal{T}) architecture. In regards of the formula rewriting step, some modifications were made in order to benefit the decision procedure used in Noodler. Some rewritings made by default were not beneficial for the new solver, so they were disabled. For example, rules that state the membership of a string term to a regular language are efficiently handled by Noodler, but they were initially preprocessed and changed by the default Z3 solver.

In general, Noodler interacts with Z3 as follows. First, it receives a satisfying boolean assignment from Z3's SAT solver and then removes useless assignments for dealing with a simpler formula. Noodler then converts the conjunction of string literals to a LIA constraint and feeds it back to the solver as a theory lemma. Noodler also implements, as aforementioned, a decision procedure based on NFAs called *stabilization-based procedure*.

Regarding the concepts around string theory handled by the solver, Noodler uses different ideas for each step. Noodler applies *Axiom Saturation*, which saturates the input formula before the SAT solver starts generating assignments. This makes the solver avoid checking SAT assignments that are clearly false. Another saturation, now regarding string predicates, is to replace some formulas with equivalent ones that are more suitable for the approach, such as length/regular constraints. One example is the replacement of $\neg \text{contains}(s, "abc")$ by $s \notin \Sigma^*abc\Sigma^*$.

The main decision procedure used by Noodler is the *stabilization-based procedure*.

During preprocessing, Noodler decreases the number of equations by applying conversion to regular constraints, propagation of epsilons (empty word), under-

approximation rules, etc. Another thing Noodler does is to check for trivial unsatisfiable parts of the formula for an early termination. Noolder current supports a lot of string functions and predicates such as `replace`, `contains`, `substr`, `at`, `indexOf`, etc. The predicates `replace_all` and `to/from_int` are not supported. In general, the decision procedure implemented in Z3-Noodler can handle the fragment of string theory of chain-free fragment with unbounded disequations and regular constraints, and quadratic equations.

The results presented in the paper show that Z3-Noolder is capable of competing against other string solvers in the state of the art.