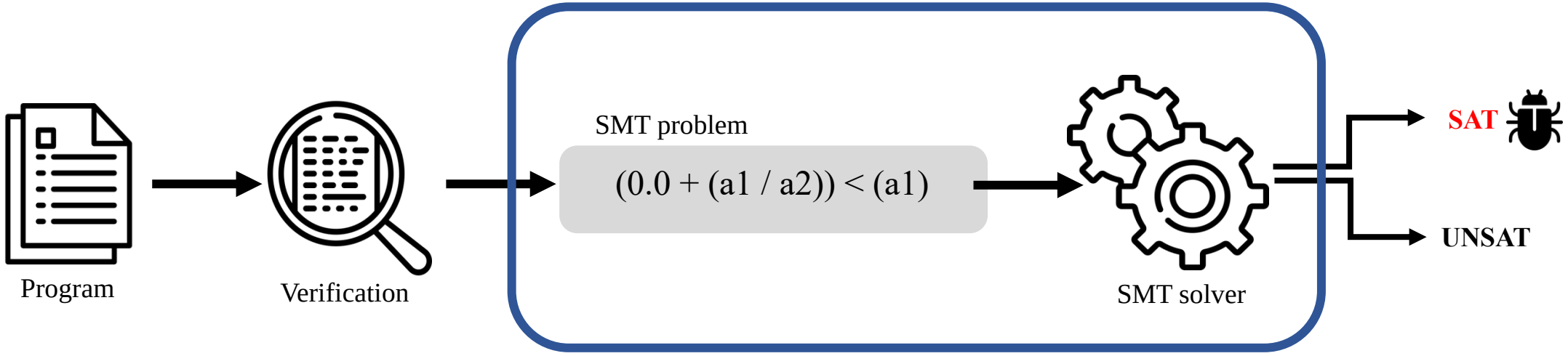
SMT Solver Testing

황도연

SMT Solver란?



- **특정** Logic**에 속하는 논리식이** Satisfiable**한지 판단해주는 소프트웨어**
- **많은 기술이** SMT solver**의 결과에 의존**, SMT Solver**의** Correctness**는 필수적**

문제: SMT Solver가 가진 많은 버그들

버그 사례

Even if formula is satisfiable, z3str3 always returns `unsat` when variable or function is parameter of `re.range`.

```
$ z3 smt.string_solver=z3str3 small.smt2
unsat
$ z3 small.smt2
sat
$ cat small.smt2
(set-logic QF_SLIA)
(declare-fun x () String)
(declare-fun y () String)
(assert (= 1 (str.len y)))
(assert (str.in_re x (re.range "-" y)))
(check-sat)
```

For this instance, z3 `ea365de` gives an invalid model.

```
$ cat delta.smt2
(set-option :smt.string_solver z3str3)
(set-option :model_validate true)
(declare-fun v () String)
(assert (str.<= "B" (str.++ v)))
(check-sat)
$ z3 delta.smt2
sat
(error "line 5 column 10: an invalid model was generated")
```

For this instance, z3 `d5d77df` returns `sat` while cvc5 returns `unsat`.

```
$ cvc5 small.smt2
unsat
$ z3 small.smt2
sat
$ cat small.smt2
(set-logic ALL)
(assert (exists ((x Int)) (and (< 0 x) (> 0 (div 1 (* 2 x))))))
(check-sat)
```

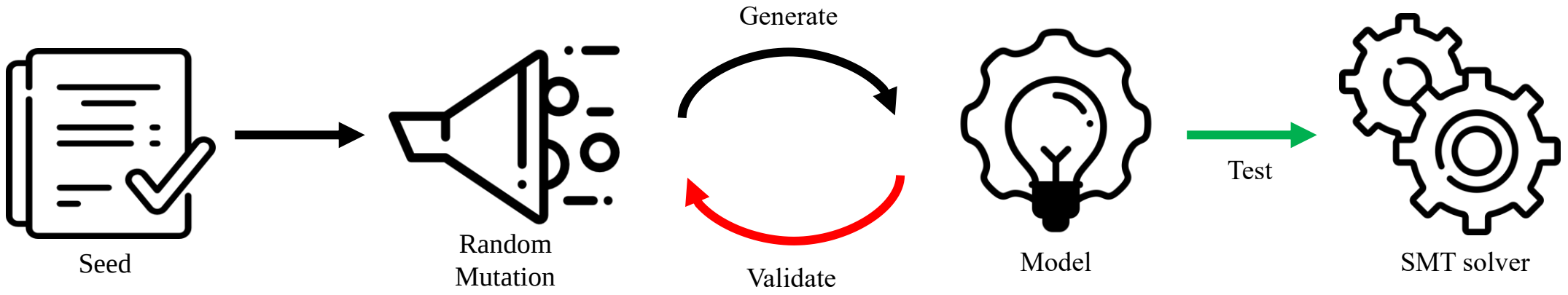
(rewriter.ite_extra_rules=true) Invalid model

```
$z3release model_validate=true rewriter.ite_extra_rules=true bug3.smt2
sat
(error "line 4 column 10: an invalid model was generated")
$z3release model_validate=true bug3.smt2
sat
$cat bug3.smt2
(declare-const a (_ FloatingPoint 11  53))
(declare-const b (_ FloatingPoint 11  53))
(assert (fp.eq (fp.max b a) (fp.fma RTZ b b b)))
(check-sat)
```

이러한 Correctness의 중요성에도 SMT Solver는 너무나 많은 버그를 가지고 있음

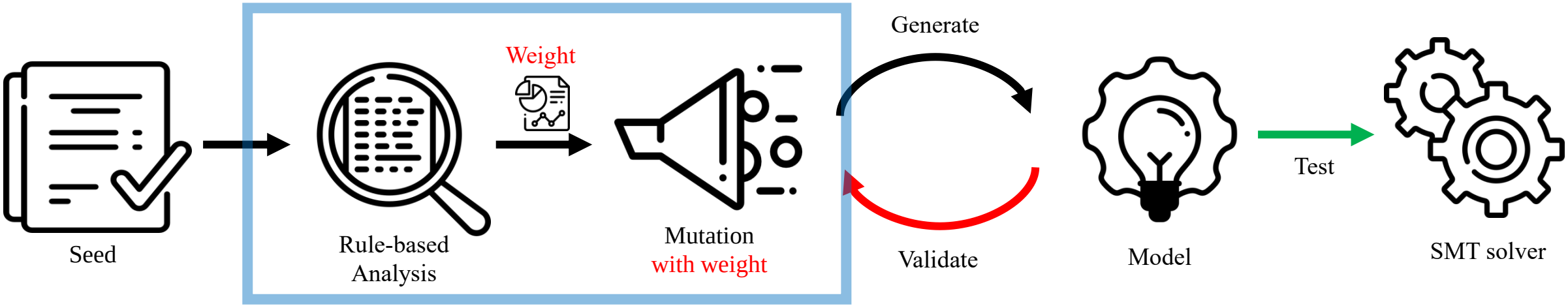
Diver: Oracle-guided testing with unrestricted mutation

- 아이디어: 모델을 Oracle로 사용하여 만들어지는 후보 뮤턴트들을 검증하기



랜덤으로 뮤턴트를 생성하여 형태에 제약이 없으면서도,
생성된 뮤턴트의 Satisfiability를 보장할 수 있도록.

Diver의 동작 흐름 및 규칙 기반 분석의 효용

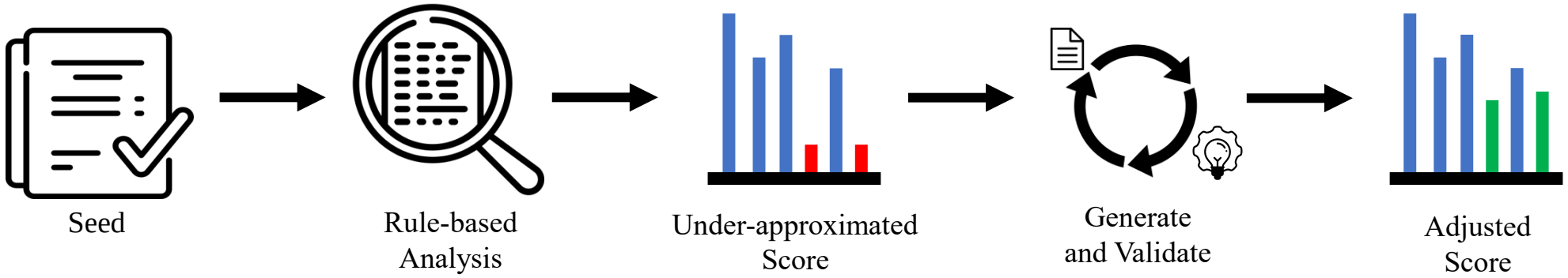


- 규칙 기반 Weight sampling 적용시 동일한 시간에 1.6배 더 많은 뮤턴트를 생성함
- 랜덤 샘플링 대비 버그 재현율 22% 상승

의문: 규칙 기반 Weighted sampling은 항상 버그 검출에 긍정적인 영향만을 주는가?

버그 재현을 위한 접근법: under-approximation 보정

해당 Term들에 대해서만 실제 확률로 보정을 수행



너무 낮은 score(0.001)로 책정된 Term들을 파악

보여야 할 것: 보정한 Score를 Weight로 사용했을 때 버그 검출이 가능한가?

요약

- **목표**: Diver를 확장 및 보완하는 테스팅 방법론 제안
  - 알고리즘 관점에서의 보완: 재현이 안 되는 것을 재현이 되도록
- **문제**: 규칙 기반 샘플링시 일부 term들에 대해 과도한 under-approximation 발생
- **아이디어**: 해당 term에 대해서만 통계 기반 추정을 통해 score 보정