Extending DIVINE with Symbolic Verification using SMT

Henrich Lauko, Vladimír Štill, Petr Ročkai and Jiří Barnat

Masaryk University

April 5, 2019

DIVINE 4

- **Explicit-state** model checker for C/C++
- Based on the **LLVM** toolchain
- Support of control-flow non-determinism – **parallelism**
- Reachability, LTL, assertions, memory safety, deadlocks

Implementing Symbolic Computation

Let program do the symbolic computation.

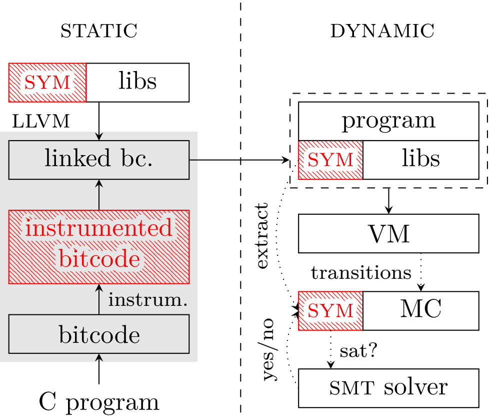
Concrete program:

```
int a = __nondet();
int b = factorial(7);
int c = a + b;
if (a == c) {
    ...
}
```

Symbolic program:

```
sym_int a = __sym_val();
int b = factorial(7);
sym_int c = s_add(a,b);
sym_bool d = s_eq(a,c);
if (nondet_bool()) {
    assume(d);
    ...
}
```

Extending DIVINE with Symbolic Verification



No need to complicate the verification core.

https://divine.fi.muni.cz