Artifact for Certified Decision Procedures for Width-Independent Bitvector Predicates in Interactive Theorem Provers

A Introduction

This artifact contains the decision procedures introduced by the coresponding paper, it aims to demonstrate that they are correct and proved correct using the Lean proof assistant, and to demonstrate they performance.

A.1 Correctness

To demonstrate the correctness, we use the decision procedures to prove lemmas and check that Lean accepts the proof, and that they do not rely on unexpected axioms such as `sorryAx`.

A.2 Performance

To demonstrate their performance, we test the decision procedures on a set of tests, one coming from LLVM and one coming from the obfuscation litterature. The artifact contains the inputs of these benchmarks as well as scripts to run then and generate the plots in the paper.

B Hardware Dependencies

Podman or Docker are necessary to run our artifact (we tested it with the former). The container image has all dependencies needed to compile our framework with Lean4.

The artifact requires a somewhat beefy machine to run: the largest evaluation contains 10000 problems, and 2500 of the problems exercise our k-induction solver which uses a SAT solver, which can require a large amount of RAM.

We ran our evaluation on a machine with 32 cores and 125 GB of RAM, though 32 GB should suffice.

C Getting Started

Access the docker image from 10.5281/zenodo.11519739.

```
$ docker load -i oopsla25-width-indep.tar
$ docker run -it oopsla25-width-indep
# | This clears the build cache,
# | fetches the maths library from the build cache,
# | and builds our framework.
$ cd /code/lean-mlir && lake clean && lake exe cache get && lake build
# Run a smaller version of the experiments, and check that the output is as expected.
$ /code/lean-mlir/artifacts/oopsla25-width-indep/test_experiments.sh
$ cat /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot-data.tex
$ cat /code/lean-mlirSSA/Experimental/Bits/Fast/Dataset2/dataset2-cactus-plot-data.tex
# (Optional) View the cactus plots from within the container.
$ timg -p iterm2 /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot.jpeg
$ timg -p iterm2
↪  /code/lean-mlir/SSA/Experimental/Bits/Fast/Dataset2/automata-automata-circuit-cactus-plot.jpeg
```

Author's Contact Information:

D Step by Step Instructions

As for the previous section, load the image and build the project.

```
$ docker load -i oopsla25-width-indep.tar
$ docker run -it oopsla25-width-indep
# | This clears the build cache,
# | fetches the maths library from the build cache,
# | and builds our framework.
$ cd /code/lean-mlir && lake clean && lake exe cache get && lake build
```

D.1 Checking That Our Decision Procedures Are Verified

Check that the theorems that are automatically proved by our decision procedures do not rely on incomplete proofs, which is materialized as the sorryAx axiom in Lean.

This relies on the `guard_msgs` command of Lean, which enforces that the comment contains exactly the output of a Lean command, which is used here with the `print axioms` command which lists all the axioms used (transitively) by a theorem.

```
# | This allows to check that the key theorems of our decision procedures are
# | guarded, and that they do not contain `sorry`s.
$ rg -g "SSA/Experimental/Bits/**/*.lean" "#guard_msgs in #print axioms" -C7 | grep "sorry"
# | This allows to manually inspect that the key theorems of our paper are sorry-free.
$ rg -g "SSA/Experimental/Bits/**/*.lean" "#guard_msgs in #print axioms" -C7
```

D.1.1 Automata Decision Procedure. The next shows that the key lemma of the automata based decision procedure relies on no axioms except for missing hashMap lemmas in Lean:

```
AutoStructs/FormulaToAuto.lean
1944-/--
1945-info: 'Formula.denote_of_isUniversal' depends on axioms: [hashMap_missing,
1946- propext,
1947- Classical.choice,
1948- Lean.ofReduceBool,
1949- Quot.sound]
1950--/
1951:#guard_msgs in #print axioms Formula.denote_of_isUniversal
```

Furthermore, the tactic, when used against an example, also produces no extra axioms beyond the standard lean axioms, plus `hashMap_missing`:

```
Fast/Tests.lean
58-theorem check_axioms_presburger (w : Nat) (a b : BitVec w) : a + b = b + a := by
59-  bv_automata_gen (config := {backend := .automata} )
60-
61-/--
62-info: 'check_axioms_presburger' depends on axioms: [hashMap_missing,
63- propext,
64- Classical.choice,
65- Lean.ofReduceBool,
66- Lean.trustCompiler,
67- Quot.sound]
```

```
68--/
69:#guard_msgs in #print axioms check_axioms_presburger
```

D.1.2 K-induction decision procedure. The example below show that the key lemma, as well as the use of the tactic on a an example theorem do not use any axioms beyond the standard Lean inbuilt axioms of propositional extensionality, classical choice principles, and soundness of quotient types:

```
Fast/ReflectVerif.lean
2561-    simp
2562-
2563-/--
2564-info: 'ReflectVerif.BvDecide.KInductionCircuits.Predicate.
↪  denote_of_verifyCircuit_mkSafetyCircuit_of_verifyCircuit_mkIndHypCycleBreaking' depends on axioms:
↪  [propext,
2565- Classical.choice,
2566- Quot.sound]
2567--/
2568:#guard_msgs in #print axioms
↪  Predicate.denote_of_verifyCircuit_mkSafetyCircuit_of_verifyCircuit_mkIndHypCycleBreaking
...
Fast/Tests.lean
49-set_option trace.Bits.FastVerif true in
50-theorem check_axioms_cadical (w : Nat) (a b : BitVec w) : a + b = b + a := by
51-  bv_automata_gen (config := {backend := .circuit_cadical_verified} )
52-
53-/--
54-info: 'check_axioms_cadical' depends on axioms: [propext, Classical.choice, Lean.ofReduceBool,
↪  Quot.sound]
55--/
56:#guard_msgs in #print axioms check_axioms_cadical
```

D.1.3 MBA Decision Procedure. The example below show that the key lemma, as well as the use of the tactic on a an example theorem do not use any axioms beyond the standard Lean inbuilt axioms of propositional extensionality, classical choice principles, and soundness of quotient types:

```
Fast/MBA.lean
636-info: 'MBA.Eqn.forall_width_reflect_zero_of_width_one_denote_zero' depends on axioms: [propext,
↪  Classical.choice, Quot.sound]
637--/
638:#guard_msgs in #print axioms Eqn.forall_width_reflect_zero_of_width_one_denote_zero
639-
640-@[simp]
--
985-info: 'MBA.Examples.check_axioms_mba' depends on axioms: [propext, Classical.choice, Quot.sound]
986--/
987:#guard_msgs in #print axioms check_axioms_mba
988-
989-end Examples
```

E Full Scale Run Of Our Experiments

Run the two large experiments and reproduce the data and graphs from the paper:

```
# Run experiments, and check that the output is as expected.
# (Parallelism can be tweaked by changing the -j options by editing the script.)
$ /code/lean-mlir/artifacts/oopsla25-width-indep/run_experiments.sh
$ cat /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot-data.tex
$ cat /code/lean-mlirSSA/Experimental/Bits/Fast/Dataset2/dataset2-cactus-plot-data.tex
# (Optional) View the cactus plots from within the container.
$ timg -p iterm2 /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot.jpeg
$ timg -p iterm2
↪  /code/lean-mlir/SSA/Experimental/Bits/Fast/Dataset2/automata-automata-circuit-cactus-plot.jpeg
```

The tex files contain data correponding to Figure 7 of the paper, and the two graphs correspond to figures (a) and (b) page 18 of the paper as follows.

E.0.1 Verifying the Results for LLVM rewrites (`bv-evaluation`). Please run:

```
$ cat /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot-data.tex
```

This should produce the following output, where the timings might change depending on the machine that is run on. The absolute number of problems solved should be equal, while the total time and geomean time are machine-dependent.

TODO: we should explain the mapping between the names of the macros and the paper names.

```
# £ cat /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot-data.tex
\newcommand{\InstCombineNormCircuitVerifiedNumSolved}{2486} # exact match
\newcommand{\InstCombineNormPresburgerNumSolved}{2126} # exact match
\newcommand{\InstCombineBvDecideNumSolved}{7781} # exact match
\newcommand{\InstCombineNormCircuitVerifiedTotalTime}{...}
\newcommand{\InstCombineNormPresburgerTotalTime}{...}
\newcommand{\InstCombineBvDecideTotalTime}{...}
\newcommand{\InstCombineNormCircuitVerifiedGeoMean}{42ms} # machine-dependent
\newcommand{\InstCombineNormPresburgerGeoMean}{25ms} # machine-dependent
\newcommand{\InstCombineBvDecideGeoMean}{66ms} # machine-dependent
\newcommand{\InstCombineTotalProblems}{7855} # exact match
```

To view the plot data, either use `docker cp` to copy the file out of the container, or use the installed `timg` tool to view the plot data within the container. Use the `timg -p` option to choose the protocol to view images. This will work on terminal emulators that support the sixel protocol.

```
$ timg -p iterm2 /code/lean-mlir/bv-evaluation/automata-automata-circuit-cactus-plot.jpeg
```

E.0.2 Verifying the Results for MBA (`SSA/Experimental/Bits/Fast/Dataset2`). Please run:

```
$ cat /code/lean-mlir/SSA/Experimental/Bits/Fast/Dataset2/dataset2-cactus-plot-data.tex
```

This should produce the following output, where the timings might change depending on the machine that is run on. The absolute number of problems solved should be equal, while the total time and geomean time are machine-dependent.

```
# £ cat /code/lean-mlir/SSA/Experimental/Bits/Fast/Dataset2/dataset2-cactus-plot-data.tex
\newcommand{\InstCombineNormCircuitVerifiedNumSolved}{2486} # exact match
\newcommand{\InstCombineNormPresburgerNumSolved}{2126} # exact match
\newcommand{\InstCombineBvDecideNumSolved}{7781} # exact match
\newcommand{\InstCombineNormCircuitVerifiedTotalTime}{...}
\newcommand{\InstCombineNormPresburgerTotalTime}{...}
\newcommand{\InstCombineBvDecideTotalTime}{...}
\newcommand{\InstCombineNormCircuitVerifiedGeoMean}{42ms} # machine-dependent
\newcommand{\InstCombineNormPresburgerGeoMean}{25ms} # machine-dependent
\newcommand{\InstCombineBvDecideGeoMean}{66ms} # machine-dependent
\newcommand{\InstCombineTotalProblems}{7855} # exact match
```

The plots are located at:

```
$ timg -p iterm2
↪  /code/lean-mlir/SSA/Experimental/Bits/Fast/Dataset2/automata-automata-circuit-cactus-plot.jpeg
```