Mechanized Undecidability of Halting Problems for Reversible Machines

First Seminar Talk

Hizbullah A. A. Jabbar
Advisor: Andrej Dudenhefner
Supervisor: Prof. Gert Smolka

Saarland University

22/07/2021

Outline

- Introduction
- FRACTRAN
- Counter Machines
- Morita's construction
- Future Work

Reversible Machines

- Reversible machines are those whose operations can be logically reversed.
- Machine operations can be described in terms of *configurations*.
  - A counter machine configuration is a pair of its current state and the values of its counters.
  - A Turing machine configuration is a tuple of its current state, tape contents, and position of its head.

Reversible Machines

- Let $P \vdash s \to t$ denotes a machine P going from configuration s to configuration t in one step.
- P is (*extensionally*) reversible iff for all configurations s, t, and u,

  If $P \vdash s \to u$,
  and $P \vdash t \to u$,
  then $s = t$.
- Reversibility is dual to determinism.
- Interests in reversible machines stem from Landauer's Principle (Landauer, 1961).

Goal

The goal of this thesis is to mechanize in Coq the (un)-decidability of the halting problem for:

- Reversible FRACTRAN programs
- Reversible 2-counter machines
- Reversible cellular automata

Outline

FRACTRAN

- A FRACTRAN (Conway, 1987) program is a list of fractions of natural numbers.
- FRACTRAN configurations are also natural numbers.
- Let $[\frac{1}{2}, \frac{4}{3}]$ be a FRACTRAN program and suppose that the initial configuration is 6.
  - The second configuration is 3.
  - The third configuration is 4.
  - The fourth configuration is 2.
  - The final configuration is 1.

$$\textsc{frac-0}\quad \frac{c \cdot s = d \cdot t}{\frac{c}{d}, L \vdash s \to t} \qquad\qquad \textsc{frac-1}\quad \frac{d \nmid c \cdot s \qquad L \vdash s \to t}{\frac{c}{d}, L \vdash s \to t}$$

FRACTRAN

Theorem

Termination of reversible FRACTRAN programs is decidable.

Proof.

Let L be a FRACTRAN program and we proceed by induction on the length of L.

- If L is empty or singleton, termination of L is decidable.
- Otherwise, $L = \frac{a}{b}, \frac{c}{d}, L'$ for some L'.
  - If $b \mid d$ then $\frac{c}{d}$ can be dropped and the claim follows from the inductive hypothesis.
  - Otherwise L is not reversible and we have a contradiction.

□

Outline

Counter Machines

- A counter machine configuration t is a pair of its internal state and the values of its counters.
- There are two instructions: INC x or DEC x j.
- A counter machine P is simply a list of instructions.
- P halts on input s if after executing P starting from s, eventually the program index is outside P.

Counter Machines

- An intuitive way to make a reversible 2-counter machine:
  - Add extra counters to store computation history,
  - "Compress" back to 2 counters.
- Existing "compression" algorithms:
  - Via FRACTRAN (Larchey-Wendling and Forster, 2020; Larchey-Wendling, 2021).
  - *Prime exponentiation* (Morita, 1996) as part of Morita's construction.
- Morita's construction (ibid.) provides a way to convert any k-counter machine into a reversible 2-counter machine.
- Undecidability of the halting problem for reversible 2-counter machine is established via reduction from the halting problem of 2-counter machine.

Outline

Morita's construction

- Morita uses a different counter machine formalization which is graph-like.
- *Quadruples* $\delta = (p, x, i, q)$ where $i = \{Z, P, -, 0, +\}$ instead of instructions.
- Morita proposes a *syntactic* criterion for reversibility using the so-called *range overlap* on quadruples.
- A counter machine is (syntactically/*intensionally* reversible) iff none of its quadruples overlap in range.

Morita's construction

Lemma

Intensional reversibility implies extensional reversibility.

- However, the converse is not true.
- Furthermore, we are not aware of an easy way to construct an equivalent counter machine such that the converse is true.

Morita's construction

Let M be a deterministic k-counter machine with an indegree n.

1. If $n \leq 1$ then we are done, otherwise reduce the indegree of M to 2.
2. Add two extra counters: one to keep track of which quadruple was executed and the other for working.
3. Compress via prime exponentiation: change $(m_1, \ldots, m_k)$ into $(p_1^{m_1} \cdots p_k^{m_k}, 0)$ where $p_1, \ldots, p_k$ are primes.

Morita's construction

We chose to work with the instruction-based counter machine formalization to implement Morita's construction.

- It is deterministic by construction.
- Simpler halting condition.
- Better development support.

Outline

What has been done

- Full mechanization of the decidability of reversible FRACTRAN programs ($\backsim$ 400 LoC).
- Full mechanization of Morita's syntactic reversibility lemma ($\backsim$ 200 LoC).
- Partial mechanization of Morita's construction ($\backsim$ 400 LoC so far).

Future Work

Must-have goals:

- Full mechanization of Morita's construction and undecidability of the halting problem for reversible 2-counter machines.
- Mechanization of cellular automata and undecidability of the halting problem for (reversible) cellular automata.

Nice-to-haves:

- Mechanization of undecidability of the halting problem for other reversible machines e.g. reversible Turing machines.

Thank you for your attention! Questions?