Two-Way Finite Automata: an Alternative Concept

Author: Ing. Dominik Nejedlý Supervisor: prof. RNDr. Alexander Meduna, CSc.

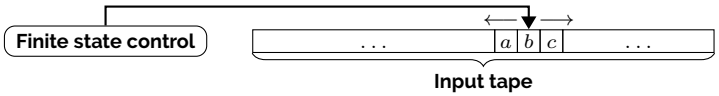
BRNO UNIVERSITY OF TECHNOLOGY
FACULTY OF INFORMATION TECHNOLOGY

Motivation

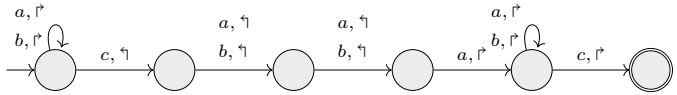
Finite automata, introduced more than eight decades ago in [1], have always fulfilled a crucially important role in computer science both in theory and in practice. Thus, the theory of computation has defined a great variety of these automata in order to provide every computer science area with the version that fits its needs as optimally as possible. Two-way finite automata, independently introduced in [2] and [3], represent significant versions of this kind, which have been constantly and intensively investigated since their introduction from various angles. This thesis continues with this long-time vivid investigation trend by introducing other versions of two-way finite automata, which are **computationally stronger** than their originals. These newly introduced versions characterize the **linear language family**, which properly contains the regular language family defined by classical two-way finite automata.

Classical Two-Way Finite Automata

A **two-way finite automaton** (**2FA**) is a language-recognizing device, which consists of a finite set of *states*, an *input tape*, a *read head*, and a *finite state control*. Essentially, it works just like a standard one-way finite automaton, except that it can freely move its read head either **left** or **right** on its input tape during each computational step, which allows it to re-read any occurrence of any symbol **arbitrarily many times**.



A 2FA accepting the language $\{a, b\}^*\{a\}\{a, b\}^2\{c\}$ can be modeled as can be seen on the following figure.



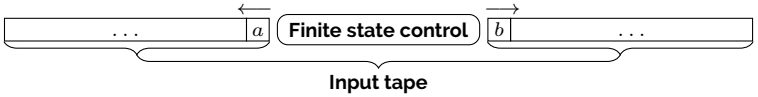
This automaton first scans the whole string, checking that it ends with c and contains only as and bs before it. Then, the automaton shifts its read head three positions back and verifies that a is under it.

Alternative Concept: Input-Erasing Two-Way Finite Automata

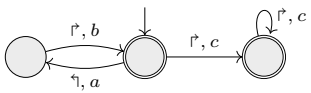
An **input-erasing two-way finite automaton** works, in essence, like a 2FA, except that it

- **erases the input symbols**—once an occurrence of a symbol is read on the input tape, it is erased from it, so the automaton can never re-read it later during its computation—and
- starts its computation at **any position** on the input tape (a 2FA starts processing the tape from the left end).

We define **general** (**IE2GFA**) and **simple** (**IE2SFA**) versions of these automata. IE2GFAs can read multiple consecutive symbols in a single computational step, whereas IE2SFAs can read at most one symbol at a time.



The following IE2SFA accepts the non-regular linear language $\{a^n b^n c^m \mid m, n \geq 0\}$.



This automaton first reads the same number of as to the left and bs to the right, and then it reads an arbitrary number of cs to the right.

Achieved Results

Accepting Power

We proved that

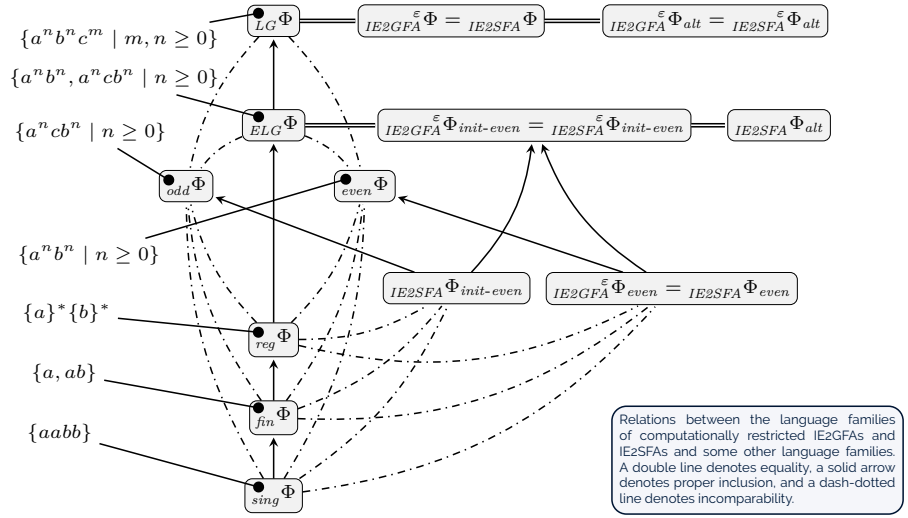
- IE2GFAs are **as strong as linear grammars** (**LGs**) and that
- IE2GFAs and IE2SFAs, along with their ε-free alternatives (versions that read at least one input symbol per computational step), possess the **same accepting power**.

Restrictions

We defined the following restrictions that require the performance of left and right moves in an alternating way and investigated how they affect the computational power of IE2GFAs and IE2SFAs.

- **Alternating computation** (**alt**): Every two consecutive computational steps read symbols in opposite directions.
- **Even computation** (**even**): Alternating computation of even length n such that, for each odd i, $0 \leq i \leq n$, both the ith and $(i+1)$th computational steps read the same number of input symbols.
- **Initialized even computation** (**init-even**): Even computation extended by an initialization computational step.

Achieved results are summarized by the following figure. As can be seen, IE2GFAs and IE2SFAs working under initialized even computation define the same language family as **even linear grammars** (**ELGs**).



Lastly, we investigated the accepting power of IE2GFAs working under the assumption that their input strings or their parts belong to languages from some prescribed language families. We showed that the studied regular-based input-restrictions give rise to **no increase in the power of IE2GFAs** and that some of them can even **decrease their power to that of 2FAs**.

References

[1] Warren S. McCulloch and Walter Pitts. A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133, 1943.
[2] Michael O. Rabin and Dana Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3(2):114–125, 4 1959.
[3] J. C. Shepherdson. The reduction of two-way automata to one-way automata. *IBM Journal of Research and Development*, 3(2):198–200, 1959.