Excel@FIT 2026
http://excel.fit.vutbr.cz

Advancing the Complementation of Elevator Automata

Ondrej Alexaj*

Abstract
Complementation of nondeterministic ω-automata is a key step in automata-based verification, but it is expensive in practice, especially for transition-based Emerson-Lei automata (TELAs). We extend an SCC-based modular approach from Büchi automata to TELAs of the practically relevant *elevator* class, using dedicated constructions for different structural types of strongly connected components. We also introduce *elevatorization*, which transforms general TELAs into elevator ones. In experiments, the resulting construction produces significantly smaller automata than the tool Spot.

*xalexa09@stud.fit.vut.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

Formal verification provides mathematically rigorous guarantees about system behavior and is indispensable in safety-critical domains. Automata-theoretic model checking represents both the system and the specification as ω-automata and reduces verification to language-theoretic decision problems. In this workflow, complementation is a bottleneck: language inclusion $L(S) \subseteq L(A_\varphi)$ is commonly reduced to emptiness of $L(S) \cap \overline{L(A_\varphi)}$, which requires complementing the specification automaton.

This work addresses the complementation of transition-based Emerson-Lei automata (TELAs), an ω-automata model whose acceptance is a Boolean formula over atoms $Inf(c)$ and $Fin(c)$ evaluated on colors occurring infinitely (or finitely) often. TELAs subsume standard acceptance conditions and can be more succinct than classical automata, but this also makes algorithmic operations such as determinization and complementation more challenging [1].

For general nondeterministic ω-automata, complementation can be done via determinization followed by dualizing acceptance, but determinization is expensive for rich acceptance conditions[1]. Büchi complementation has many dedicated constructions, and recent work has started addressing TELA complementation more directly [2], yet the general case remains costly with the best upper bound $\mathcal{O}\left(n^{2^k}(0.76\,nk)^{n2^k}\right)$ established in [2] and lower bound 2^{2^n} [1].

Our complementation procedure builds on a modular approach [3]. Accepting behavior of an ω-automaton is witnessed inside accepting strongly connected components (SCCs), so we partition accepting SCCs by structural type and complement them via specialized *partial algorithms* that run in parallel in a top-level construction. This approach is particularly effective for *elevator automata*, where every SCC is either inherently weak (every cycle is accepting) or deterministic, eliminating the hard case of nondeterministic accepting SCCs. Namely, we extend modular complementation from Büchi to TELAs by designing new partial algorithms for deterministic accepting components under Emerson-Lei acceptance.

2. Background: ω-automata and Emerson-Lei acceptance

We consider transition-based ω-automata running on infinite words Σ^ω. Acceptance is evaluated on transitions: colors are assigned to transitions, and a run satisfies an atomic predicate $Inf(c)$ if color c appears infinitely often along the run, and $Fin(c)$ if it appears only finitely often.

A transition-based Emerson-Lei automaton (TELA) generalizes classical acceptance by allowing an acceptance formula built from $Inf(c)$ and $Fin(c)$ atoms using conjunction and disjunction.

3. Complementation and the modular viewpoint

Complementation of a nondeterministic ω-automaton cannot be achieved by simply negating the acceptance condition: a word is accepted if *there exists* an accepting run, so the complement must ensure that *no* run satisfies the original acceptance. Determinism makes complementation easy, but obtaining determinism (fully or partially) can be expensive.

The modular approach introduced in [3] avoids complementing the entire automaton at once. It decomposes the accepting part of the automaton into SCCs and groups them into *partition blocks* of the same structural type. A key advantage is that each SCC type can be handled by a dedicated partial algorithm.

4. Elevator automata

An automaton is an *elevator* if each SCC is either inherently weak or deterministic. This restriction removes nondeterministic accepting SCCs, which are the main source of complexity in SCC-based complementation. Elevator automata form a class complete for ω-regular languages, and they occur frequently in practice, e.g., in automata produced from translations of temporal logic formulas.

For Büchi elevators, modular complementation yields a worst-case $\mathcal{O}(4^n)$ blow-up [3]. The main contribution in this work is to lift the modular framework from Büchi acceptance to the Emerson-Lei acceptance.

5. Complementing elevator TELAs

Inherently weak accepting components can be complemented essentially independently of the exact Emerson-Lei formula: since every cycle inside such an SCC is accepting, a word belongs to the complement whenever all runs that enter the SCC eventually leave it. This can be tracked by sampling currently active runs and repeatedly checking that the sampled set drains to empty.

The more involved case is deterministic accepting components. Here, acceptance depends on infinite-vs-finite occurrence of multiple colors combined by Boolean structure. The decisive observation is that within a deterministic SCC, each run has a unique continuation; hence, after a run enters a deterministic SCC, we can verify violations *per run*.

Our constructions separate runs into: (i) a stabilizing set whose long-run "violation witness" has not yet been fixed, and (ii) committed sets whose membership encodes which acceptance atom (or subformula) the run will violate. The algorithm alternates between waiting (postponing commitment) and checking (all runs committed), and uses breakpoint mechanisms to enforce recurring obligations stemming from $Fin(\cdot)$ violations.

6. Elevatorization: extending applicability beyond elevators

Not every practically obtained TELA is an elevator: nondeterministic accepting SCCs can still occur. To nevertheless benefit from elevator-based complementation, we introduce *elevatorization* to transform each nondeterministic accepting component into a non-accepting core and attach deterministic components, obtaining an equivalent *elevator* automaton. The construction adapts approach from limit-determinization construction [4].

7. Conclusions

Our complementation procedure for transition-based Emerson-Lei automata builds on a simple idea: instead of tackling the whole automaton at once, we exploit its SCC structure and handle the relevant cases separately.

In practice, this pays off. The construction, especially with our optimizations that reduce unnecessary nondeterminism, tends to produce significantly smaller automata than the state-of-the-art tool. While the existing tool is often faster, it frequently generates much larger state spaces, whereas our approach keeps the blow-up under better control. This makes it convenient for applications such as language inclusion, where the size of the complement can be the limiting factor.

References

[1] Udi Boker. Why these automata types? In Gilles Barthe, Geoff Sutcliffe, and Margus Veanes, editors, *LPAR-22. 22nd International Conference on Logic for Programming, Artificial Intelligence and Reasoning*, volume 57 of *EPiC Series in Computing*, pages 143–163. EasyChair, 2018.

[2] Vojtěch Havlena, Ondřej Lengál, and Barbora Šmahlíková. Complementation of emerson-lei automata. In Parosh Aziz Abdulla and Delia Kesner, editors, *Foundations of Software Science and Computation Structures*, pages 88–110, Cham, 2025. Springer Nature Switzerland.

[3] Vojtěch Havlena, Ondřej Lengál, Yong Li, Barbora Šmahlíková, and Andrea Turrini. Modular mix-and-match complementation of büchi automata. In

Sriram Sankaranarayanan and Natasha Sharygina, editors, *Tools and Algorithms for the Construction and Analysis of Systems*, pages 249–270, Cham, 2023. Springer Nature Switzerland.

[4] Tobias John, Simon Jantsch, Christel Baier, and Sascha Klüppelholz. From emerson-lei automata to deterministic, limit-deterministic or good-for-mdp automata. *Innov. Syst. Softw. Eng.*, 18(3):385–403, September 2022.