Tight Bounds for Complementing Parity Automata

Sven Schewe and Thomas Varghese
Department of Computer Science, University of Liverpool

Abstract

We follow a connection between tight determinisation and complementation and establish a complementation procedure from transition-labelled parity automata to transition-labelled nondeterministic Büchi automata. We prove it to be tight up to an $O(n)$ factor, where n is the size of the nondeterministic parity automaton. This factor does not depend on the number of priorities.

1 Introduction

The precise complexity of complementing ω-automata is an intriguing problem for two reasons: first, the quest for optimal algorithms is a much researched problem [1, 17, 12, 24, 15, 11, 25, 10, 8, 7, 6, 13, 27, 29], and second, complementation is a valuable tool in formal verification (c.f., [9]), in particular when studying language inclusion problems of ω-regular languages. Complementation is also useful to check the correctness of translation techniques [27, 26]. The GOAL tool [26], for example, provides such a test suite and incorporates recent algorithms [15, 25, 8, 13] for Büchi complementation.

While devising optimal complementation algorithms for nondeterministic finite automata is simple—nondeterministic *finite* automata can be determinised using a simple subset construction, and deterministic finite automata can be complemented by complementing the set of final states [14, 17]—devising optimal complementation algorithms for nondeterministic ω-automata is hard, because simple subset constructions are not sufficient to determinise or complement them [11, 10].

Given the hardness and importance of the problem, the complementation of ω-automata enjoyed much attention. The initial focus was on the complementation of Büchi automata with state-based acceptance [1, 12, 24, 11, 15, 10, 25, 8, 7, 6, 27, 29, 18, 26], and it resulted in a continuous improvement of its upper and lower bounds.

The first complementation algorithm dates back to the introduction of Büchi automata in 1962. In his seminal paper "On a decision method in restricted second order arithmetic" [1], Büchi develops a doubly exponential complementation procedure. While Büchi's result shows that nondeterministic Büchi automata (and thus ω-regular expressions) are closed under complementation, complementing an automaton with n states may, when using Büchi's complementation procedure, result in an automaton with $2^{2^{O(n)}}$ states, while an $\Omega(2^n)$ lower bound [17] is inherited from finite automata.

In the late 80s, these bounds have been improved in a first sequence of results, starting with establishing an EXPTIME upper bound [12, 24], which matches the EXPTIME lower bound [17] inherited from finite automata. However, the early EXPTIME complementation techniques produce automata with up to $2^{O(n^2)}$ states [12, 24]; hence, these upper bounds were still exponential in the lower bounds.

This situation changed in 1988, when Safra introduced his famous determinisation procedure for nondeterministic Büchi automata [15], resulting in an $n^{O(n)}$ bound for Büchi complementation, while Michel [11] established a seemingly matching $\Omega(n!)$ lower bound in the same year. Together, these results imply that Büchi complementation is in $n^{\theta(n)}$, leaving again the impression of a tight bound.

As pointed out by Vardi [27], this impression is misleading, because the $O()$ notation hides an $n^{\theta(n)}$ gap between both bounds. This gap has been narrowed down in 2001 to $2^{\theta(n)}$ by the introduction of an alternative complementation technique that builds on level rankings and a cut-point construction [8]. The complexity of the plain method is approximately $(6n)^n$ [8], leaving a $(6e)^n$ gap to Michel's lower bound [11].

Subseqently, tight level rankings [6, 29] have been exploited by Friedgut, Kupferman, and Vardi [6] to improve the upper complexity bound to $O\big((0.96n)^n\big)$, and by Yan [29] to improve the lower complexity bound to $\Omega\big((0.76n)^n\big)$. Schewe [18] has provided a matching upper bound, showing tightness up to an $O(n^2)$ factor.

In recent works, more succinct acceptance mechanism have been studied, where the most important ones are parity and generalised Büchi automata, as they occur naturally in the translation of μ-calculi and LTL specifications, respectively. In [22], we gave tight bounds for the determinisation and complementation of generalised Büchi automata. For Rabin, Streett, and parity automata, there has been much progress [4, 3, 2], in particular establishing an $n^{\theta(n)}$ bound for parity complementation with state-based acceptance, which has been a great improvement and pushed tightness of parity comple- mentation to the level known from Büchi complementation since the late 80s [15, 11].

Contribution. In this paper, we establish tight bounds for the complementation of parity automata with transition-based acceptance. A generalisation of the ranking-based complementation procedures quoted above to transition-based acceptance is straight forward, and the Safra-style determinisation procedures from the literature [15, 16, 13, 19, 22] have a natural representation with an acceptance condition on transitions. Their translation to state-based acceptance is by multiplying the acceptance from the last transition to the state space.

A similar observation can be made for other automata transformations, like the removal of ε-transitions from translations of μ-calculi [28, 20] and the treatment of asynchronous systems [21], where the state-space grows by multiplication with the acceptance information (e.g., maximal priority on a finite sequence of transitions), while it cannot grow in case of transition-based acceptance. Similarly, tools like SPOT [5] offer more concise automata with transition-based acceptance mechanism as a translation from LTL. Using state-based acceptance in the automaton that we want to complement would also complicate the presentation of the complementation procedure. But first and foremost, using transition-based acceptance provides cleaner results.

This is the case because in state-based acceptance, the role of the states is overloaded. In finite automata over infinite structures, each state represents the class of tails of the word that can be accepted from this state. In state-based acceptance, they have to account for the acceptance mechanism itself, too, while they are relieved from this burden in transition-based acceptance. In complementation techniques based on rankings, this results in a situation where states with certain properties, such as final states for Büchi automata, can only occur with some ranks, but not with all.

As transition-based acceptance separates these concerns, the presentation becomes cleaner. The natural downside is that we lose the $n^{O(n)}$ bound [3] for parity complementation, as the number of priorities in a parity automaton with transition-based acceptance can grow arbitrarily. But in return, we do get a clean and simple complementation procedure based on a data structure we call flattened nested history trees (FNHTs), which is inspired by a generalisation of history trees [19] to multiple levels, one for each even priority ≥ 2.

In [22], we showed a connection between optimal determinisation and complementation for generalised Büchi automata, where we exploit the nondeterministic power of a Büchi automaton to devise a tight complementation procedure. In this paper, we follow this connection between tight determinisation [23] and complementation to devise a tight complementation construction from parity to nondeterministic Büchi automata.

We show that any procedure that complements full parity automata with states Q and maximal priority π has at least $|\mathsf{fnht}(Q,\pi)|/2$ states, where $\mathsf{fnht}(Q,\pi)$ is the set of FNHTs for a given set Q of states and maximal priority π of the parity automaton that is to be complemented. Our complementation construction uses a marker in addition for its acceptance mechanism. Essentially, it is used to mark some position of interest in an FNHT. It accounts for the $O(n)$ gap between the upper and lower bound. We show that, for $\pi \geq 2$ (and hence for Büchi automata upwards) the number of states of our complementation construction is bounded by $4n+1$ times the lower bound.

2 Preliminaries

We denote the non-negative integers by $\omega = \{0, 1, 2, 3, ...\}$. For a finite alphabet Σ, an infinite *word* α is an infinite sequence $\alpha_0\alpha_1\alpha_2 \cdots$ of letters from Σ. We sometimes interpret ω-words as functions $\alpha : i \mapsto \alpha_i$, and use Σ^ω to denote the ω-words over Σ.

ω-automata are finite automata that are interpreted over infinite words and recognise ω-regular languages $L \subseteq \Sigma^\omega$. Nondeterministic parity automata are quintuples $\mathcal{P} = (Q, \Sigma, I, T, \mathsf{pri} : T \to \Pi)$, where Q is a finite set of states with a non-empty subset $I \subseteq Q$ of initial states, Σ is a finite alphabet, $T \subseteq Q \times \Sigma \times Q$ is a transition relation that maps states and input letters to sets of successor states, and pri is a priority function that maps transitions to a finite set $\Pi \subset \omega$ of non-negative integers.

A *run* ρ of a nondeterministic parity automaton $\mathcal{P}$ on an input word α is an infinite sequence $\rho : \omega \to Q$ of states of $\mathcal{P}$, also denoted $\rho = q_0q_1q_2 \cdots \in Q^\omega$, such that the first symbol of ρ is an initial state $q_0 \in I$ and, for all $i \in \omega$, $(q_i, \alpha_i, q_{i+1}) \in T$ is a valid transition. For a run ρ on a word α, we denote with $\overline{\rho} : i \mapsto \big(\rho(i), \alpha(i), \rho(i + 1)\big)$ the transitions of ρ. Let $\mathsf{infin}(\rho) = \{q \in Q \mid \forall i \in \omega\ \exists j > i \text{ such that } \rho(j) = q\}$ denote the set of all states that occur infinitely often during the run ρ. Likewise, let $\mathsf{infin}(\overline{\rho}) = \{t \in T \mid \forall i \in \omega\ \exists j > i \text{ such that } \overline{\rho}(j) = t\}$ denote the set of all transitions that are taken infinitely many times in $\overline{\rho}$. Acceptance of a run is defined through the priority function pri. A run ρ of a parity automaton is *accepting* if $\limsup_{n\to\infty} \mathsf{pri}\big(\overline{\rho}(n)\big)$ is even, that is, if the highest priority that occurs infinitely often in the transitions of ρ is even. A word α is accepted by a parity automaton $\mathcal{P}$ iff it has an accepting run, and its language $\mathcal{L}(\mathcal{P})$ is the set of words it accepts.

Parity automata with $\Pi \subseteq \{1, 2\}$ are called *Büchi* automata. Büchi automata are denoted $\mathcal{B} = (Q, \Sigma, I, T, F)$, where $F \subseteq T$ are called the final or accepting transitions. A run is accepting if it contains infinitely many accepting transitions. $\mathcal{B}$ is thus a rendering of the parity automaton, where $\mathsf{pri} : t \mapsto 2$ if $t \in F$ and $\mathsf{pri} : t \mapsto 1$ if $t \notin F$.

We assume w.l.o.g. that the set Π of priorities satisfies that $\min \Pi \in \{0, 1\}$. If this is not the case, we can simply change pri accordingly to $\mathsf{pri}' : t \mapsto \mathsf{pri}(t) - 2$ several times until this constraint is satisfied. We likewise assume that Π has no holes, that is, $\Pi = \{i \in \omega \mid \max \Pi \geq i \geq \min \Pi\}$. If there is a hole $h \notin \Pi$ with $\max \Pi > h > \min \Pi$, we can change pri to $\mathsf{pri}' : t \mapsto \mathsf{pri}(t)$ if $\mathsf{pri}(t) < h$ and $\mathsf{pri}' : t \mapsto \mathsf{pri}(t) - 2$ if $\mathsf{pri}(t) > h$. Obviously, these changes do not affect the acceptance of any run, and applying finitely many of these changes brings Π into this normal form.

The different priorities have a natural order $\succcurlyeq$, where $i \succ j$ if i is even and j is odd; i is even and $i > j$; or j is odd and $i < j$. For a non-empty set $\Pi' \subseteq \Pi$ of priorities, $\mathsf{opt}\Pi' = \{i \in \Pi' \mid \forall j \in \Pi'.\ i \succcurlyeq j\}$ denotes the optimal priority for acceptance.

The complexity of a parity automaton $\mathcal{P} = (Q, \Sigma, I, T, \mathsf{pri} : T \to \Pi)$ is measured by its size $n = |Q|$ and its set of priorities Π. For a given size n and set of priorities Π, there is an automaton that recognises a hardest language. This automaton is referred to as the full automaton $\mathcal{P}_n^\Pi = (Q, \Sigma, I, T, \mathsf{pri} : T \to \Pi)$, with $|Q| = n$, $I = Q$, $\Sigma = Q \times Q \to 2^\Pi$, $T = \{q, \sigma, q') \mid \sigma(q, q') \neq \emptyset$, and $\mathsf{pri}(q, \sigma, q') = \mathsf{opt}\sigma(q, q')$.

Note that partial functions from $Q \times Q$ to Π would work as well as the alphabet. The larger alphabet is chosen for technical convenience in the proofs. Any other language recognised by a nondeterministic parity automaton $\mathcal{P}$ with n states and priorities Π can essentially be obtained by a language restriction via alphabet restriction from $\mathcal{P}_n^\Pi$.

3 Complementing parity automata

The construction described in this section draws from two main sources of inspiration. One source is the introduction of efficient techniques for the determinisation of parity automata in [23]. The nested history trees used there have been our inspiration for the *flattened nested history trees* that form the core data structure in the complementation from Subsection 3.2 and are the backbone of the lower bound proof from Subsection 3.4. The second source of inspiration is the connection [22] between the efficient determinisation based on history trees [19] for Büchi automata and generalised Büchi automata [22] and their level ranking based complementation [8, 6, 18, 22].

The intuition for the complementation is to use the nondeterministic power of a Büchi automaton to reduce the size of the data stored for determinisation. As usual, this nondeterministic power is intuitively used to guess a point in time, where all nodes of the nested history trees from parity determinisation [23], which are eventually always stable, are henceforth stable. Alongside, the set of stable nodes can be guessed.

Like in the construction for Büchi automata, the structure can then be flattened, preserving the 'nicking order', the order in which older nodes and descendants take preference in taking states of the nondeterministic parity automaton that is determinised. The complement automaton runs in two phases: a first phase before this guessed point in time, and a second phase after this point, where the run starts in such a flattened tree.

In the first subsection, we introduce *flattened nested history trees* as our main data structure. While we take inspiration from nested history trees [23], the construction is self-contained. In the second subsection, we show that Büchi automata recognising the complement language of the full nondeterministic parity automaton $\mathcal{P}_n^\Pi$ need to be large by showing disjointness properties of accepting runs for a large class of words, one for each full flattened nested history tree introduced in Subsection 3.1. The definition of this language is also instructive in how the data structure is exploited.

We extend our data structure by markers, resulting in *marked flattened trees*, which are then used as the main part of the state space of the natural complementation construction introduced in Subsection 3.2. We show correctness of our complementation construction in Subsection 3.3 and tightness up to an $O(n)$ factor in Subsection 3.4.

Note that all our constructions assume $\max \Pi \geq 2$, and therefore do not cover the less expressive CoBüchi automata.

3.1 Flattened nested history trees & marked flattened trees

Flattened nested history trees (FNHTs) are the main data structure used in our complementation algorithm. For a given parity automaton $\mathcal{P} = (Q, \Sigma, I, T, \mathsf{pri} : T \to \Pi)$, an FNHT over the set Q of states, maximal priority $\pi_m = \max \Pi$ and maximal even priority $\pi_e = \mathsf{opt}\Pi$, is a tuple $(\mathcal{T}, l_s : \mathcal{T} \to 2^Q, l_l : \mathcal{T} \to 2\mathbb{N}, l_p : \mathcal{T} \to 2^Q, l_r : \mathcal{T} \to 2^Q)$, where $\mathcal{T}$(an ordered, labelled tree) is a non-empty, finite, and prefix closed subset of finite sequences of natural numbers and a special symbol $\mathfrak{s}$ (for *stepchild*), $\omega \cup \{\mathfrak{s}\}$, that satisfies the constraints given below. We call a node $v\mathfrak{s} \in \mathcal{T}$ a *stepchild* of v, and refer to all other nodes vc with $c \in \omega$ as the *natural children* of v. $\mathsf{nc}(v) = \{vc \mid c \in \omega$ and $vc \in \mathcal{T}\}$ is the set of natural children of v. The root is a stepchild.

The **constraints** an FNHT quintuple has to satisfy are as follows:

- Stepchildren have only natural children, and natural children only stepchildren.
- Only natural children and, when the highest priority π is odd, the root may be leafs.
- $\mathcal{T}$ is order closed: for all $c, c' \in \omega$ with $c < c'$, $vc' \in \mathcal{T}$ implies $vc \in \mathcal{T}$.
- For all $v \in \mathcal{T}$, $l_s(v) \neq \emptyset$.
- If v is a stepchild, then $l_p(v) = \emptyset$.
- If v is a stepchild, then $l_s(v) = l_r(v) \cup \bigcup_{v' \in \mathsf{nc}(v)} l_s(v')$.

  The sets $l_s(v')$ and $l_s(v'')$ are disjoint for all $v', v'' \in \mathsf{nc}(v)$ with $v' \neq v''$, and $l_r(v)$ is disjoint with $\bigcup_{v' \in \mathsf{nc}(v)} l_s(v')$.
- If v is a natural child, then $l_p(v) \neq \emptyset$, $l_s(v) = l_p(v) \cup l_r(v)$, and $l_p(v) \cap l_r(v) = \emptyset$.
- If a natural child v is not a leaf, then $l_s(v\mathfrak{s}) = l_p(v)$.
- $l_l(\varepsilon) = \pi_e$ and, for all $v \in \mathcal{T}$, $l_l(v) \geq 2$.

- If $v\mathfrak{s} \in \mathcal{T}$, then $l_l(v\mathfrak{s}) = l_l(v) - 2$, and if $vc \in \mathcal{T}$ for $c \in \omega$, then $l_l(vc) = l_l(v)$.

The elements in $l_s(v)$ are called the states, $l_p(v)$ the pure states, and $l_r(v)$ the recurrent states of a node v, and $l_l(v)$ is called its level. Note that the level follows a simple pattern: the root is labelled with the maximal even priority, $l_l(\varepsilon) = \pi_e$, the level of natural children is the same as the level of their parents, and the level of a stepchild $v\mathfrak{s}$ of a node v is two less than the level of v. For a given maximal even priority π_e, the level is therefore redundant information that can be reconstructed from the node and π_i. For a given set Q and maximal priority π, $\mathsf{fnht}(Q,\pi)$ denotes the flattened nested history trees over Q. An FNHT is called *full* if the states $l_s(\varepsilon) = Q$ of the root is the full set Q.

To include an acceptance mechanism, we enrich FNHTs to *marked flattened tress* (MFTs), which additionally contain a *marker* v_m and a marking set Q_m, such that

- either $v_m = (\overline{v}, r)$ with $\overline{v} \in \mathcal{T}$ is used to mark that we follow a breakpoint construction on the recurrent states, in this case $l_r(\overline{v}) \supseteq Q_m \neq \emptyset$,
- or $v_m = (\overline{v}, p)$ such that $\overline{v}$ is a leaf in $\mathcal{T}$ is used to mark that we follow a breakpoint construction on the pure states of a leaf $\overline{v}$, in this case $l_p(\overline{v}) \supseteq Q_m \neq \emptyset$.

The marker is used to mark a property to be checked. For markers $v_m = (\overline{v}, r)$, the property is that a particular node would not spawn stable children in a nested history tree [23]. As usual in Safra like constructions, this is checked with a breakpoint, where a breakpoint is reached when all children of a node spawned prior to the last breakpoint die. For markers $v_m = (\overline{v}, p)$, the property is that all runs that are henceforth trapped in the pure nodes of v must eventually encounter a priority $l_l(v) - 1$. This priority is then dominating, and implies rejection as an odd priority. We check these properties round robin for all nodes in $\mathcal{T}$, skipping over nodes, where the respective sets $l_r(\overline{v})$ or $l_p(\overline{v})$ are empty, as the breakpoint there is trivially reached immediately.

For a given FNHT $(\mathcal{T}, l_s, l_l, l_p, l_r)$, $\mathsf{next}(v_m)$ is a mapping from a marker v_m to a marker/marking set pair $(\overline{v}, r), l_r(\overline{v})$ or $(\overline{v}, p), l_p(\overline{v})$. The new marker is the first marker after v_m in some round robin order such that the set $l_r(\overline{v})$ or $l_p(\overline{v})$, resp., is non-empty.

If $(\mathcal{T}, l_s, l_l, l_p, l_r)$ is an FNHT and v_m and Q_m satisfy the constraints for markers and marking sets from above, then $(\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m)$ is a marked flattened tree. For a given set Q and priorities Π with maximal priority $\pi = \max \Pi$, $\mathsf{mft}(Q,\pi)$ denotes the marked flattened trees over Q. A marking is called *full* if either $v_m = (\overline{v}, r)$ and $Q_m = l_r(\overline{v})$, or $v_m = (\overline{v}, p)$ and $Q_m = l_p(\overline{v})$.

3.2 Construction

For a given nondeterministic parity automaton $\mathcal{P} = (Q, \Sigma, I, T, \mathsf{pri} : T \to \Pi)$ with maximal even priority $\pi_e > 1$, we construct a nondeterministic Büchi automaton $\mathcal{C} = (Q', \Sigma, \{I\}, T', F)$ that recognises the complement language of $\mathcal{P}$ as follows. First we set $Q' = Q_1 \cup Q_2$ with $Q_1 = 2^Q$ and $Q_2 = \mathsf{mft}(Q,\pi)$, and $T' = T_1 \cup T_t \cup T_2$, where

- $T_1 \subseteq Q_1 \times \Sigma \times Q_1$ are transitions in an initial part Q_1 of the states of $\mathcal{C}$,
- $T_t \subseteq Q_1 \times \Sigma \times Q_2$ are transfer transitions that can be taken only once in a run, and
- $T_2 \subseteq Q_2 \times \Sigma \times Q_2$, are transitions in a final part Q_2 of the states of $\mathcal{C}$,

where T_1 and T_2 are deterministic. We first define a transition function δ for the subset construction and functions δ_i for all priorities $i \in \Pi$, and then the sets T_1, T_t, and T_2.

- $\delta : (S, \sigma) \mapsto \{q \in Q \mid \exists s \in S.\ (s, \sigma, q) \in T\}$,
- $\delta_i : (S, \sigma) \mapsto \big\{q \in Q \mid \exists s \in S.\ (s, \sigma, q) \in T \text{ and } \mathsf{pri}\big((s, \sigma, q)\big) \succcurlyeq i\big\}$,
- $T_1 = \big\{(S, \sigma, S') \in Q_1 \times \Sigma \times Q_1 \mid S' = \delta(S, \sigma)\big\}$,
  where only transitions $(\emptyset, \sigma, \emptyset)$ are accepting.

- $T_t = \big\{\big(S, \sigma, (\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m)\big) \in Q_1 \times \Sigma \times Q_2 \mid l_s(\varepsilon) = \delta(S, \sigma)\big\}$ and we have that$(\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m)$ is a marked flattened tree.
- $T_2 = \big\{\big((\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m), \sigma, s\big) \in Q_2 \times \Sigma \times Q_2 \mid$
  - if v is a stepchild, then $l''_s(v) = \delta_{l_l(v)+1}\big(l_s(v), \sigma\big)$
  - if v is a natural child, then $l''_s(v) = \delta_{l_l(v)-1}\big(l_s(v), \sigma\big)$
  - if v is a natural child, then $l''_r(v) = \delta_{l_l(v)-1}\big(l_r(v), \sigma\big) \cup \delta_{l_l(v)}\big(l_s(v), \sigma\big)$,
  - starting at the root, we then define inductively:
    - $l'_s(\varepsilon) = l''_s(\varepsilon)$,
    - if vc is a natural child, then $l'_s(vc) = \big(l''_s(vc) \cap l'_s(v)\big) \smallsetminus \bigcup_{c'<c} l''_s(vc')$, $l'_r(vc) = l''_r(vc) \cap l'_s(vc)$, and $l'_p(vc) = l'_s(vc) \smallsetminus l'_r(vc)$, and
    - if $v\mathfrak{s}$ is a stepchild, then $l'_s(v\mathfrak{s}) = l'_p(v)$.
  - if one exists, we extend the functions to obtain the unique FNHT $(\mathcal{T}, l'_s, l_l, l'_p, l'_r)$ (otherwise $\mathcal{C}$ blocks)
  - if $v_m = (\overline{v}, r)$ then $Q'_m = \delta_{l_l(\overline{v})-1}(Q_m, \sigma) \cap l'_r(\overline{v})$, and
    if $v_m = (\overline{v}, p)$ then $Q'_m = \delta_{l_l(\overline{v})-3}(Q_m, \sigma) \cap l'_p(\overline{v})$,
  - if $Q'_m = \emptyset$, then the transition is accepting and $s = \big(\mathcal{T}, l'_s, l_l, l'_p, l'_r; \mathsf{next}(v_m)\big)$,
  - if $Q'_m \neq \emptyset$, then the transition is not accepting and $s = (\mathcal{T}, l'_s, l_l, l'_p, l'_r; v_m, Q'_m)$.

3.3 Correctness

To show that $\mathcal{L}(\mathcal{C})$ is the complement of $\mathcal{L}(\mathcal{P})$, we first show that a word accepted by $\mathcal{C}$ is rejected by $\mathcal{P}$ and then, vice versa, that a word accepted by $\mathcal{P}$ is rejected by $\mathcal{C}$.

Lemma 3.1 *If $\mathcal{C}$ has an accepting run on α, then $\mathcal{P}$ rejects α.*

Proof. Let $\rho = S_0 S_1 \dots$ be an accepting run of $\mathcal{C}$ on α that stays in Q_1. Thus, there is an $i \in \omega$ such that, for all $j \geq i$, $S_j = \emptyset$. But if we consider any run $\rho' = q_0 q_1 q_2 \dots$ of $\mathcal{P}$ on α, then it is easy to show by induction that $q_k \in S_k$ holds for all $k \in \omega$, which contradicts $S_i = \emptyset$; that is, in this case $\mathcal{P}$ has no run on α.

Let us now assume that $\rho = S_0 S_1 \dots S_i s_{i+1} s_{i+2} \dots$ is an accepting run of $\mathcal{C}$ on α, where $(S_i, \alpha_i, s_{i+1}) \in T_t$ is the transfer transition taken. (Recall that runs of $\mathcal{C}$ must either stay in Q_1 or contain exactly one transfer transition.)

Let us assume for contradiction that $\mathcal{P}$ has an accepting run $\rho' = q_0 q_1 q_2 \dots$ with even dominating priority $e = \limsup_{j \to \infty} \mathsf{pri}\big((q_j, \alpha_j, q_{j+1})\big)$. Let, for all $j > i$, $s_j = (\mathcal{T}, l^j_s, l_l, l^j_p, l^j_r; v^j_m, Q^j_m)$ and $S_j = l^j_s(\varepsilon)$. It is again easy to show by induction that $q_j \in S_j$ for all $j \in \omega$. Let now $v_j \in \mathcal{T}$ be the longest node with $l^j_l(v_j) \geq e$ and $q_j \in l^j_s(v_j)$. Note that such a node exists, as $q_j \in S_j = l^j_s(\varepsilon)$ holds. We now distinguish the two cases that the v_j do and do not stabilise eventually.

First case. Assume that there are an $i' > i$ and a $v \in \mathcal{T}$ such that, for all $j \geq i'$, $v_j = v$. We choose i' big enough that $\mathsf{pri}(q_{j-1}, \alpha_{j-1}, q_j) \succ e + 1$ holds for all $j \geq i'$.

If v is a stepchild, then $q_j \in l^j_r(v)$ for all $j \geq i'$. Using the assumption that ρ is accepting, there is an $i'' > i'$ such that $(s_{i''-1}, \alpha_{i''-1}, s_{i''})$ is accepting, and $v^{i''}_m = (v, r)$. (Note that $q_{i''} \in l^{i''}_r(v)$ implies $l^{i''}_r(v) \neq \emptyset$.) But then we have $q_{i''} \in Q^{i''}_m = l^{i''}_r(v)$, and an inductive argument provides $(s_j, \alpha_j, s_{j+1}) \notin F$ and $q_j \in Q^j_m$ for all $j \geq i''$. This contradicts that ρ is accepting.

If v is a natural child, then we distinguish three cases. The first one is that there is a $j' \geq i'$ such that $q_{j'} \in l^{j'}_r(v)$. Then we can show by induction that $q_j \in l^j_r(v)$ for all $j \geq j'$ and follow the same argument as for stepchildren, using $i'' > j'$.

The second is that $q_j \in l^j_p(v)$ holds for all $j \geq i'$. There are now again a few sub-cases that each lead to contradiction. The first is that $l_l(v) = e$. But in this case, we can choose a $j > i'$ with

$\mathsf{pri}\big((q_j,\alpha_j,q_{j+1})\big) = e$ and get $q_{j+1} \in l_r^{j+1}(v)$ (contradiction). The second is that $l_l(v) > e$ and v is not a leaf. But in that case, $l_l(v\mathfrak{s}) \geq e$ holds and $q_j \in l_p^j(v)$ implies $q_j \in l_p^j(v\mathfrak{s})$, which contradicts the maximality of v. Finally, if $l_l(v) > e$ and v is a leaf of $\mathcal{T}$, we get a similar argument as for stepchildren: Using the assumption that ρ is accepting, there is an $i'' > i'$ such that $(s_{i''-1}, \alpha_{i''-1}, s_{i''})$ is accepting, and $v_m^{i''} = (v,p)$. (Note that $q_{i''} \in l_p^{i''}(v)$ implies $l_p^{i''}(v) \neq \emptyset$.) But then we have $q_{i''} \in Q_m^{i''} = l_p^{i''}(v)$, and an inductive argument provides $(s_j,\alpha_j,s_{j+1}) \notin F$ and $q_j \in Q_m^j$ for all $j \geq i''$. This contradicts that ρ is accepting.

Second case. Assume that the v_j do not stabilise. Let v be the longest sequence such that v is an initial sequence of almost all v_j, and let $i' > i$ be an index such that v is an initial sequence of v_j for all $j \geq i'$. Note that q_j is in $l_s(v_j')$ for all ancestors v_j' of v_j.

First, we assume for contradiction that there is a $j > i'$ with $\mathsf{pri}\big((q_j,\alpha_j,q_{j+1})\big) = e' \succ l_l(v)$ (note that the 'better than' relation implies that $e' > l_l(v)$ is even). Then we select a maximal ancestor v' of v with $l_l(v') = e'$; note that such an ancestor is a natural child, as a stepchild has only natural children, and all of them have the same level.

As v' is an ancestor of v_j and v_{j+1}, $q_j \in l_s^j(v')$ and $q_{j+1} \in l_s^{j+1}(v')$ hold, and by the transition rules thus imply $q_{j+1} \in l_r^{j+1}(v')$, which contradicts $q_{j+1} \in l_s^{j+1}(v_{j+1})$. (Note that $l_l(v') > l_l(v) \geq l_l(v_{j+1})$ holds.)

Second, we show that $\mathsf{pri}\big((q_j,\alpha_j,q_{j+1})\big) \preccurlyeq l_l(v)+1$ holds infinitely many times. For this, we first note that the non-stability of the sequence of v_j-s implies that at least one of the following three events happen for infinitely many $j > i'$.

1. v is a stepchild, $q_j \in l_s^j(vc)$ for some child vc of v, but, for all children vc' of v, $q_{j+1} \notin l_s^{j+1}(vc')$,
2. v is a stepchild, $q_j \in l_s^j(vc)$ for some child vc of v, and $q_{j+1} \in l_s^{j+1}(vc')$ for some older sibling vc' of vc, that is, for $c' > c$, or
3. v is a natural child, $q_j \notin l_s^j(v\mathfrak{s})$, but $q_{j+1} \in l_s^{j+1}(v\mathfrak{s})$.

Note that this is just the counter position to "v_j stabilises or v is not maximal". In all three cases, the definition of T_2 requires that $\mathsf{pri}\big((q_j,\alpha_j,q_{j+1})\big) \preccurlyeq l_l(v)+1$.

As the first observation implies that there may only be finitely many transitions with even priority $> l_l(v)$ and the second observation implies that there are infinitely many transitions in ρ' with odd priority $> l_l(v)$, they together imply that $\limsup_{j\to\infty} \mathsf{pri}\big((q_j,\alpha_j,q_{j+1})\big)$ is odd, which leads to the final contradiction. □

Lemma 3.2 *If $\mathcal{P}$ has an accepting run on α, then $\mathcal{C}$ rejects α.*

Proof. Let $\rho = q_0q_1q_2\ldots$ be an accepting run of $\mathcal{P}$ on α with even dominating priority $e = \limsup_{j\to\infty} \mathsf{pri}\big((q_j,\alpha_j,q_{j+1})\big)$.

Let us first assume for contradiction that $\mathcal{C}$ has an accepting run $\rho' = S_0S_1\ldots$ which is entirely in Q_1. It is then easy to show by induction that $q_i \in S_i$ holds for all $i \in \omega$, such that no transition of (S_i,α_i,S_{i+1}) is accepting.

Let us now assume for contradiction that $\mathcal{C}$ has an accepting run $\rho' = S_0S_1\ldots S_is_{i+1}s_{i+2}\ldots$, where $(S_i,\alpha_i,s_{i+1}) \in T_t$ is the transfer transition taken. (Recall that runs of $\mathcal{C}$ must either stay in Q_1 or contain exactly one transfer transition.) Let further $s_j = \big(\mathcal{T}, l_s^j, l_l, l_p^j, l_r^j; v_m^j, Q_m^j\big)$ and $S_j = l_s^j(\varepsilon)$ for all $j > i$.

It is easy to show by induction that, for all $j \in \omega$, $q_j \in S_j$ holds. We choose an $i_\varepsilon > i$ such that, for all $k \geq i_\varepsilon$, $\mathsf{pri}\big((q_{k-1},\alpha_{k-1},q_k)\big) \leq e$ holds.

Let us now look at the nodes $v \in \mathcal{T}$, such that $q_j \in l_s^j(v)$, where $j \geq i_\varepsilon$.

Construction basis. We have already shown $q_j \in S_j = l_s^j(\varepsilon)$ for all $j > i$, and thus in particular for all $j \geq i_\varepsilon$.

Construction step. If, for some **stepchild** $v \in \mathcal{T}$ with $l_l(v) \geq e$ and some $i_v \geq i_\varepsilon$, it holds for all $j \geq i_v$ that $q_j \in l_s^j(v)$, then the following holds for all $j \geq i_v$: if $v' \in \mathsf{nc}(v)$ is a natural child of v and $q_j \in l_s^j(v')$, then either $q_{j+1} \in l_s^{j+1}(v')$, or there is a younger sibling v'' of v' in $\mathcal{T}$ such that $q_{j+1} \in l_s^{j+1}(v'')$.

As transitions to younger siblings can only occur finitely often without intermediate transitions to older siblings, we have one of the following two cases:

1. for all $j \geq i_v$, $q_j \in l_s^j(v)$, but for every natural child v' of v, $q_j \notin l_s^j(v')$, or

2. there is a natural child v' of v and an index $i_{v'} \geq i_v$ such that, for all $j \geq i_{v'}$, $q_j \in l_s^j(v')$.

As v is a stepchild, the first case implies that $q_j \in l_r^j(v)$ for all $j \geq i_v$. However, using the assumption that ρ' is accepting, there is an $i'_v > i_v$ such that $(s_{i'_v-1}, \alpha_{i'_v-1}, s_{i'_v})$ is accepting, and $v_m^{i'_v} = (v, r)$, as the marker is circulating in a round robin fashion. (Note that $q_{i'_v} \in l_r^{i'_v}(v)$ implies $l_r^{i'_v}(v) \neq \emptyset$.) But then we have $q_{i'_v} \in Q_m^{i'_v} = l_r^{i'_v}(v)$, and an inductive argument provides $(s_j, \alpha_j, s_{j+1}) \notin F$ and $q_j \in Q_m^j$ for all $j \geq i'_v$.

In the second case, we continue with v' and the index $i_{v'}$.

If, for some **natural child** $v \in \mathcal{T}$ with $l_l(v) > e$ and some $i_v \geq i_\varepsilon$, it holds for all $j \geq i_v$ that $q_j \in l_s^j(v)$, then one of the following holds.

1. There is an $i'_v \geq i_v$ such that $q_{i'_v} \in l_r^{i'_v}(v)$.

2. For all $j \geq i_v$, $q_j \in l_p^j(v)$.

In the first case, it is easy to show by induction that $q_j \in l_r^j(v)$ holds for all $j \geq i_{v'}$. We can then again use the assumption that ρ' is accepting. Consequently, there is an $i''_v > i'_v$ such that $(s_{i''_v-1}, \alpha_{i''_v-1}, s_{i''_v})$ is accepting, and $v_m^{i''_v} = (v, r)$, as the marker is circulating in a round robin fashion. (Note that $q_{i''_v} \in l_r^{i''_v}(v)$ implies $l_r^{i''_v}(v) \neq \emptyset$.) But then we have again $q_{i''_v} \in Q_m^{i''_v} = l_r^{i''_v}(v)$, and an inductive argument provides $(s_j, \alpha_j, s_{j+1}) \notin F$ and $q_j \in Q_m^j$ for all $j \geq i''_v$.

In the second case, if v is not a leaf, then it holds for all $j \geq i_{v\mathfrak{s}} = i_v$ that $q_j \in l_s^j(v\mathfrak{s})$, and we can continue with $v\mathfrak{s}$. If v is a leaf, we again use the assumption that ρ' is accepting. Consequently, there is an $i'_v > i_v$ such that $(s_{i'_v-1}, \alpha_{i'_v-1}, s_{i'_v})$ is accepting, and $v_m^{i'_v} = (v, p)$, as the marker is circulating in a round robin fashion. (Note that v is a leaf and that $q_{i'_v} \in l_p^{i'_v}(v)$ implies $l_p^{i'_v}(v) \neq \emptyset$.) But then we have $q_{i'_v} \in Q_m^{i'_v} = l_p^{i'_v}(v)$, and an inductive argument provides $(s_j, \alpha_j, s_{j+1}) \notin F$ and $q_j \in Q_m^j$ for all $j \geq i'_v$.

If, for some **natural child** $v \in \mathcal{T}$ with $l_l(v) = e$ and some $i_v \geq i_\varepsilon$, it holds for all $j \geq i_v$ that $q_j \in l_s^j(v)$, then there is, by the definition of e, a $j > i_v$ with $\mathsf{pri}(q_{j-1}, \alpha_{j_1}, q_j) = e$. But then $q_{j-1} \in l_s^{j-1}(v)$ and $q_j \in l_s^j(v)$ imply $q_j \in l_r^j(v)$. It is then easy to establish by induction that $q_{j'} \in l_r^{j'}(v)$ for all $j' \geq j$. We can then again use the assumption that ρ' is accepting. Consequently, there is a $j' > j$ such that $(s_{j'-1}, \alpha_{j'-1}, s_{j'})$ is accepting, and $v_m^{j'} = (v, r)$, as the marker is circulating in a round robin fashion. (Note that $q_{j'} \in l_r^{j'}(v)$ implies $l_r^{j'}(v) \neq \emptyset$.) But then we have again $q_{j'} \in Q_m^{j'} = l_r^{j'}(v)$, and an inductive argument provides $(s_k, \alpha_k, s_{k+1}) \notin F$ and $q_k \in Q_m^k$ for all $k \geq j'$.

Contradiction. As the level is reduced by two every second step, one of the arguments that contradict the assumption that ρ' is accepting is reached in at most π_e steps. □

Corollary 3.3 *$\mathcal{C}$ recognises the complement language of $\mathcal{P}$.* □

3.4 Lower bound and tightness

In order to establish a lower bound, we use a sub-language of the full automaton $\mathcal{P}_n^\Pi$, and show that an automaton that recognises it must have at least as many states as there are full FNHTs in $\mathsf{fnht}(Q,\pi)$ for $n=|Q|$ and $\pi=\max\Pi$.

To show this, we define two letters for each full FNHT $t=(\mathcal{T},l_s,l_l,l_p,l_r)\in\mathsf{fnht}(Q,\pi)$. $\beta_t : Q\times Q\to 2^\Pi$ is the letter where:

- if v is a stepchild and $p,q\in l_s(v)$, then $l_l(v){+}1\in\beta_t(p,q)$ (provided $l_l(v){+}1\in\Pi$),
- if v is a stepchild, $p\in l_r(v)$, and $q\in l_s(vc)$ for some $c\in\omega$, then $l_l(v)\in\beta_t(p,q)$,
- if v is a stepchild, $c,c'\in\omega$, $c<c'$, $vc'\in\mathcal{T}$, $p\in l_s(vc')$, and $q\in l_s(vc)$, then $l_l(v)\in\beta_t(p,q)$,
- if v is a natural child, $p\in l_p(v)$, and $q\in l_r(v)$ then $l_l(v)\in\beta_t(p,q)$.
- if v is a natural child and $p,q\in l_r(v)$, then $l_l(v)-1\in\beta_t(p,q)$, and
- if v is a natural child and $p,q\in l_p(v)$, then $l_l(v)-1\in\beta_t(p,q)$.

$\gamma_t : Q\times Q\to 2^\Pi$ is the letter where $i\in\gamma_t(p,q)$ if $i\in\beta_t(p,q)$ and additionally:

- if v is a natural child, $l_l(v)-2\in\Pi$, and $p,q\in l_r(v)$, then $l_l(v)-2\in\gamma_t(p,q)$,
- if v is a stepchild and $p,q\in l_r(v)$, then $l_l(v)\in\gamma_t(p,q)$, and
- if v is a natural child, $l_l(v)-2\in\Pi$, and $p,q\in l_p(v)$, then $l_l(v)-2\in\gamma_t(p,q)$.

For a high integer $h>|\mathsf{fnht}(Q,\pi)|$, we now define the ω-word $\alpha^t=(\beta_t{\gamma_t}^{h-1})^\omega$, which consists of infinitely many sequences of length h that start with a letter β_t and continue with $h-1$ repetitions of the letter γ_t, for each full FNHT $t\in\mathsf{fnht}(Q,\pi)$.

We first observe that α^t is rejected by $\mathcal{P}_n^\Pi$.

Lemma 3.4 $\alpha^t\notin\mathcal{L}(\mathcal{P}_n^\Pi)$.

Proof. By Lemma 3.3, it suffices to show that the complement automaton $\mathcal{C}$ of $\mathcal{P}_n^\Pi$, as defined in Section 3.2 accepts α^t. The language is constructed such that $\mathcal{C}$ has a run $\rho=Q(t;v_m^1,Q_m^1)(t;v_m^2,Q_m^2)(t;v_m^3,Q_m^3)\ldots$, such that the transition $\big((t;v_m^i,Q_m^i),\alpha_i^t,(t;v_m^{i+1},Q_m^{i+1})\big)$ is accepting for $i>0$ if $i\mod h=0$. □

Let $\mathcal{B}$ be some automaton with states S that recognises the complement language of $\mathcal{P}_n^\Pi$. We now fix an accepting run $\rho_t=s_0s_1s_2\ldots$ for each word α^t and define the set A_t of states in an 'accepting cycle' as $A_t=\big\{s\in S\mid\exists i,j,k\in\omega$ with $1\le j<k\le h$ such that $s=s_{ih+j}=s_{ih+k}\big\}$ holds, and define the interesting states $I_t=A_t\cap\,\mathsf{infin}(\rho_t)$.

Lemma 3.5 *For t $\neq t'$, I_t and $I_{t'}$ are disjoint ($I_t\cap I_{t'}=\emptyset$).*

Proof idea. The proof idea is to assume that a state $s\in I_t\cap I_{t'}$, and use it to construct a word from α_t and $\alpha_{t'}$ and an accepting run of $\mathcal{B}$ on the resulting word from ρ_t and $\rho_{t'}$, and then show that it is also accepted by $\mathcal{P}_n^\Pi$.

Proof. Let us assume for contradiction that $s\in I_t\cap I_{t'}$ for $t=(\mathcal{T},l_s,l_l,l_p,l_r)\neq t'=(\mathcal{T}',l_s',l_l',l_p',l_r')$.

Noting that we can change the role of t and t', we fix two positions i and i' in the run ρ_t of α_t such that $s=s_i=s_{i'}$, and there is a $j\in\omega$ such that $jh<i<i'\le j(h+1)$, and two positions j and j' in $\rho_{t'}=s_0's_1's_2'\ldots$ such that $j<j'$, $s=s_j'=s_{j'}'$ and there is a $k\in\omega$ with $j\le k<j'$ such that $(s_k',\alpha_k^{t'},s_{k+1}')$ is an accepting transition of $\mathcal{B}$. Note that the definition of I_t provides the first and the definition of $I_{t'}$ the latter.

For the finite words $\beta_1 = \alpha_0^t\alpha_1^t \dots \alpha_{i-1}^t$, $\gamma_1 = s_0s_1 \dots s_{i-1}$, $\beta_2 = {\gamma_t}^{i'-1}$, $\gamma_2 = s_is_{i+1} \dots s_{i'-1}$, $\beta_3 = \alpha_j^{t'}\alpha_{j+1}^{t'} \dots \alpha_{j'-1}^{t'}$, and $\gamma_3 = s_js_{j+1} \dots s_{j'-1}$, $\rho_t^{t'} = \gamma_1(\gamma_2\gamma_3)^\omega$ is an accepting run of the input word $\alpha_t^{t'} = \beta_1(\beta_2\beta_3)^\omega = \alpha_0\alpha_1\alpha_2 \dots$.

We now show that $\alpha_t^{t'}$ or $\alpha_{t'}^t$ is accepted by $\mathcal{P}_n^\Pi$.

We start with the degenerated case that $\mathcal{T} = \{\varepsilon\}$ is the FNHT where the root is a leaf, and thus $\pi = \max\Pi$ odd. (The case $\mathcal{T}' = \{\varepsilon\}$ is similar.) We select a $q \in l_s'(0)$, and consider the run $\rho = q^\omega$ of $\mathcal{P}_n^\Pi$ on $\alpha_t^{t'}$. By construction, $\mathsf{pri}(q, \alpha_k, q) \leq \mathsf{opt}\Pi = l_l(\varepsilon)$ holds for all $k \geq i$. Moreover, $\alpha_k = \gamma_t$ holds for infinitely many $k \in \omega$. (In particular, it holds if $k \geq i$ and $(k-i) \mod (i'-i+j'-j) < i'-i$.) For all of these transitions, $\mathsf{pri}(q, \alpha_k, q) = \mathsf{opt}\Pi = l_l(\varepsilon)$ holds, such that $\limsup_{n\to\infty}\big(\overline{\rho}(i)\big) = \mathsf{opt}\Pi$ is even.

Starting with the level $\lambda = \mathsf{opt}\Pi$ of the root and the whole trees $\mathcal{T}$ and $\mathcal{T}'$, we now run through the following construction.

We **firstly** look at the **case** that there is **some difference in the label of some natural child** $v \in \mathcal{T} \cap \mathcal{T}'$ on the level λ. If there is an oldest child $v \in \mathcal{T} \cap \mathcal{T}'$ with $l_s(v) \neq l_s'(v)$, we assume w.l.o.g. that there is a $q \in l_s(v) \smallsetminus l_s'(v)$. Then there are two sub-cases, first that there is a $q' \in l_s(v) \cap l_s'(v)$, and second that $l_s(v) \cap l_s'(v) = \emptyset$. In the latter case we choose a $q' \in l_s'(v)$. In both sub-cases, the run $\rho = q^{i'}(q'^{j'-j}q^{i'-i})^\omega = q_0q_1q_2 \dots$ of $\mathcal{P}_n^\Pi$ on $\alpha_t^{t'}$ satisfies $\mathsf{pri}(q_k, \alpha_k, q_{k+1}) \succcurlyeq \lambda - 1$ for all $k \in \omega$, and $\mathsf{pri}(q_k, \alpha_k, q_{k+1}) \succcurlyeq \lambda$ when $q_k = q$ and $q_{k+1} = q'$. (Note that in this case $\alpha_k \in \{\beta_{t'}, \gamma_{t'}\}$ holds.)

Otherwise $l_s(v) = l_s'(v)$ holds for all natural children $v \in \mathcal{T} \cap \mathcal{T}'$ on level λ, and there is a $v \in \mathcal{T} \cap \mathcal{T}'$ on level λ such that $l_r(v) \neq l_r'(v)$. We assume w.l.o.g. that there is a $q \in l_r(v) \smallsetminus l_r'(v)$. We choose a $q' \in l_p(v)$. (Note that $q \neq q' \in l_s(v) = l_s'(v)$.) Then the run $\rho = q^{i'}(q'^{j'-j}q^{i'-i})^\omega = q_0q_1q_2 \dots$ of $\mathcal{P}_n^\Pi$ on $\alpha_t^{t'}$ satisfies $\mathsf{pri}(q_k, \alpha_k, q_{k+1}) \succcurlyeq \lambda - 1$ for all $k \in \omega$, and $\mathsf{pri}(q_k, \alpha_k, q_{k+1}) \succcurlyeq \lambda$ when $q_k = q$ and $q_{k+1} = q'$. (Note that in this case $\alpha_k = \gamma_t$ holds.)

We **secondly** look at the **case** where $l_s(v) = l_s'(v)$ and $l_r(v) = l_r'(v)$ holds for all natural children $v \in \mathcal{T} \cap \mathcal{T}$ on level π_e, but there is a natural **child** v on level λ **in the symmetrical difference** of $\mathcal{T}$ and $\mathcal{T}'$. Let us assume w.l.o.g. that $v \in \mathcal{T}'$. Let $q \in l_s'(v)$ and let v be the child of v'. This immediately implies that $q \in l_r(v)$. Thus, the run $\rho = q^\omega$ of $\mathcal{P}_n^\Pi$ on $\alpha_t^{t'}$ satisfies $\mathsf{pri}(q, \alpha_k, q) \succcurlyeq \lambda - 1$ for all $k > i$, and $\mathsf{pri}(q, \alpha_k, q) \succcurlyeq \lambda$ whenever $\alpha_k = \gamma_t$, which happens infinitely often.

We **finally** look at the **case** where the nodes of $\mathcal{T}$ and $\mathcal{T}'$ on level λ are the same, and where $l_s(v) = l_s'(v)$ and $l_r(v) = l_r'(v)$ hold for all nodes v of $\mathcal{T}$ on level λ, but there is a node v on level λ which is a **leaf** in $\mathcal{T}$ but not in $\mathcal{T}'$. (The case "leaf in $\mathcal{T}'$ but not in $\mathcal{T}$" is entirely symmetric.) Thus, $v\mathfrak{s}0$ is a node in $\mathcal{T}'$, and we select a $q \in l_s(v\mathfrak{s}0)$. If we now consider the run $\rho = q^\omega$ of $\mathcal{P}_n^\Pi$ on $\alpha_t^{t'}$, then $\mathsf{pri}(q, \alpha_k, q) \succcurlyeq \lambda - 3$ holds for all $k > i$. At the same time $\mathsf{pri}(q, \alpha_k, q) \succcurlyeq \lambda - 2$ holds whenever $\alpha_k = \gamma_t$, which happens infinitely often.

If neither of these cases holds, then there must be a natural child v on level λ such that $v\mathfrak{s} \in \mathcal{T} \cap \mathcal{T}'$ and $l_s(v\mathfrak{s}) = l_p(v) = l_p'(v) = l_s'(v\mathfrak{s})$, such that t and t' differ on the descendants of v. We then continue the construction by reducing λ to $\lambda - 2$ and intersecting $\mathcal{T}$ and $\mathcal{T}'$ with the descendants of v in t and t', respectively, and restrict the co-domain of the labelling functions of t and t' accordingly. This construction will lead to a difference in at most $0.5 \cdot \mathsf{opt}\Pi$ steps. □

Theorem 3.6 *$\mathcal{B}$ has at least as many states as $\mathsf{fnht}(Q, \max\Pi)$ contains full FNHTs.*

Proof. We prove the claim with a case distinction. The first case is that $I_t \neq \emptyset$ holds for all full FNHT $t \in \mathsf{fnht}(Q, \max\Pi)$. Lemma 3.5 shows that the sets of interesting states are pairwise disjoint for different trees $t \neq t'$, such that, as none of them is empty, $\mathcal{B}$ has at least as many states as $\mathsf{fnht}(Q, \max\Pi)$ contains full FNHTs.

The second case is there is a full FNHT $t \in \mathsf{fnht}(Q, \max\Pi)$ such that $I_t = \emptyset$. By Lemma 3.4, each $\rho_t = s_0s_1s_2 \dots$ is an accepting run. Let now $i \in \omega$ be an index, such that, for all $j \geq i$, $s_j \in \mathsf{infin}(\rho_t)$, and $k \geq i$ an integer with $k \mod h = 0$. $I_t = \emptyset$ implies that $s_{k+j} \neq s_{k+j'}$ for all j, j' with $1 \leq j < j' \leq h$. Then $\mathcal{B}$, and even $\mathsf{infin}(\rho_t)$, has at least $h - 1$ different states, and the claim follows with $h > |\mathsf{fnht}(Q, \max\Pi)|$. □

To show tightness, we proceed in three steps. In a first step, we provide an injection from MFTs with non-full marking to MFTs with full marking.

Next, we argue that the majority of FNHTs is full. Taking into account that there are at most $|Q|$ different markers makes it simple to infer that the states of our complementation construction divided by the lower bound from Theorem 3.6 is in $O(n)$.

Lemma 3.7 *There is an injection from MFTs with non-full marking to MFTs with full marking in* $\mathsf{mft}(Q, \pi)$.

Proof. For non-trivial trees $\mathcal{T} \neq \{\emptyset\}$, we can simply map an MFT $(\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m)$

- for $v_m = (\overline{v}, p)$ to the MFT $\big(\mathcal{T}', l'_s, l'_l, l'_p, l'_r; v_m, l'_p(\overline{v})\big)$ and
- for $v_m = (\overline{v}, r)$ to the MFT $\big(\mathcal{T}', l'_s, l'_l, l'_p, l'_r; v_m, l'_r(\overline{v})\big)$, where

$\mathcal{T}'$ differs from $\mathcal{T}$ only in that it has a fresh node v, which is the youngest sibling of v_m. l'_s, l'_p, l'_r differ from l_s, l_p, l_r only in $\overline{v}$ and v (where v is only in the pre-image of l'_s, l'_l, l'_p, l'_r). We set $l'_s(v) = l'_p(v) = Q_m$ and, consequently, $l'_r(v) = \emptyset$. We also set $l'_s(v) = l_s(v) \smallsetminus Q_m$.

For $v_m = (\overline{v}, p)$, we set $l'_r(\overline{v}) = l_r(\overline{v})$ and $l'_p(\overline{v}) = l_p(\overline{v}) \smallsetminus Q_m$. Note that, by the definition of markers, $\overline{v}$ is a leaf, and $l'_p(\overline{v})$ is non-empty because the marking in $(\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m)$ is not full.

For $v_m = (\overline{v}, r)$, we set $l'_r(\overline{v}) = l_r(\overline{v}) \smallsetminus Q_m$ and $l'_p(\overline{v}) = l_p(\overline{v})$. Note that $l'_r(\overline{v})$ is non-empty because the marking in $(\mathcal{T}, l_s, l_l, l_p, l_r; v_m, Q_m)$ is not full.

It is easy to see that the resulting MFT is well formed in both cases. What remains is the corner case of $\mathcal{T} = \{\varepsilon\}$.

$(\mathcal{T}, l_s, l_l, l_p, l_r; (\varepsilon, r), Q_m)$ and map it to $(\mathcal{T}', l'_s, l'_l, l'_p, l'_r; (\varepsilon, r), Q_m)$ for $\mathcal{T}' = \{\varepsilon, 0\}$ and $l'_s(\varepsilon) = l_s(\varepsilon)$, $l'_s(\varepsilon) = Q_m$, $l'_p(\varepsilon) = l'_p(0) = \emptyset$, and consequently $l'_s(0) = l'_r(0) = l_s(\varepsilon) \smallsetminus Q_m$. (Note that the latter is non-empty because the marking in $(\mathcal{T}, l_s, l_l, l_p, l_r; (\varepsilon, r), Q_m)$ is not full.) This is again a well formed MFT with full marking.

It is easy to see that the resulting function is injective. □

In Lemma 3.7, we have shown that the majority of MFTs have a full marking. Next we will see that the majority of FNHTs is full. (Note that neither mapping is surjective.)

Lemma 3.8 *There is an injection from non-full to full FNHTs in* $\mathsf{fnht}(Q, \pi)$.

Proof. To obtain such an injection, it suffices to map a non-full FNHT $(\mathcal{T}, l'_s, l'_l, l'_p, l'_r)$ to the FNHT $(\mathcal{T}', l'_s, l'_l, l'_p, l'_r)$ where $\mathcal{T}'$ differs from $\mathcal{T}$ only in that it has a fresh youngest child v of the root.

l'_s agrees with l_s on every node of $\mathcal{T}$ except for the root ε, and l'_p, l'_r agree with l_p, l_r on every node of $\mathcal{T}$. We set $l'_s(\varepsilon) = Q$, $l'_s(v) = l'_p(v) = Q \smallsetminus l_s(\varepsilon)$, and $l'_r(v) = \emptyset$.

It is obvious that the resulting FNHT $(\mathcal{T}', l'_s, l'_l, l'_p, l'_r)$ is full and well formed, and it is also obvious that the mapping is injective. □

Theorem 3.9 *The complementation construction from this section is tight up to a factor of* $4n+1$*, where* $n = |Q|$ *is the number of states of the complemented parity automaton.*

Proof. For the number of MFTs, Lemma 3.7 shows that they are at most twice the number of MFTs with full marking. Note that the marker (v_m, p) can only refer to leafs where $l_p(v_m)$ is non-empty and markers (v_m, r) can only refer to nodes where $l_r(v_m)$ is non-empty. It is easy to see that all sets described in this way are pairwise disjoint. This implies that there are at most $|Q|$ such markers. Thus, the number of MFTs with full marking is at most n times the number of FNHTs.

By Lemma 3.8, the number of FNHTs is in turn at most twice as high as the number of all full FNHTs. Thus we have bounded the number of MFTs by $4n$ times the number of full FNHTs used to estimate the lower bound in Theorem 3.6, irrespective of the priorities.

What remains is the trivial observation that the second part of the state-space, the subset construction, is dwarfed by the number of MFTs. Consequently, we can estimate the state-space of the complement automaton divided by the lower bound from Theorem 3.6 by $4n + 1$. □

References

[1] J. R. Büchi. On a decision method in restricted second order arithmetic. *Proc. of the Int. Congress on Logic, Methodology, and Philosophy of Science 1960*, pp. 1–11, 1962.

[2] Y. Cai and T. Zhang. A tight lower bound for streett complementation. In *Proc. of FSTTCS 2011*, volume 13 of *LIPIcs*, pages 339–350, 2011.

[3] Y. Cai and T. Zhang. Tight upper bounds for streett and parity complementation. In *Proc. of CSL 2011*, volume 12 of *LIPIcs*, pages 112–128, 2011.

[4] Y. Cai, T. Zhang, and H. Luo. An improved lower bound for the complementation of rabin automata. In *Proc. of LICS 2009*, 2009.

[5] A. Duret-Lutz. Ltl translation improvements in spot. In *Proc. of VECoS 2011*, pages 72–83. British Computer Society, 2011.

[6] E. Friedgut, O. Kupferman, and M. Y. Vardi. Büchi complementation made tighter. *International Journal of Foundations of Computer Science*, 17(4):851–867, 2006.

[7] S. Gurumurthy, O. Kupferman, F. Somenzi, and M. Y. Vardi. On complementing nondeterministic Büchi automata. In *Proc. of CHARME 2003*, pages 96–110, 2003.

[8] O. Kupferman and M. Y. Vardi. Weak alternating automata are not that weak. *ACM Transactions on Computational Logic*, 2(2):408–429, July 2001.

[9] R. P. Kurshan. *Computer-aided verification of coordinating processes: the automata-theoretic approach*. Princeton University Press, 1994.

[10] C. Löding. Optimal bounds for transformations of ω-automata. In *Proc. of FSTTCS 1999*, volume 1738 of *Lecture Notes in Computer Science*, pages 110–121. Springer-Verlag, 1999.

[11] M. Michel. Complementation is more difficult with automata on infinite words. Technical report, CNET, Paris (Manuscript), 1988.

[12] J.-P. Pécuchet. On the complementation of Büchi automata. *TCS*, 47(3):95–98, 1986.

[13] N. Piterman. From nondeterministic Büchi and Streett automata to deterministic parity automata. *Journal of Logical Methods in Computer Science*, 3(3:5), 2007.

[14] M. O. Rabin and D. S. Scott. Finite automata and their decision problems. *IBM Journal of Research and Development*, 3:115–125, 1959.

[15] S. Safra. On the complexity of ω-automata. In *Proc. of FOCS 1988*, pages 319–327, 1988.

[16] S. Safra. Exponential determinization for omega-automata with strong-fairness acceptance condition (extended abstract). In *Proc. of STOC 1992*, pages 275–282, 1992.

[17] W. J. Sakoda and M. Sipser. Non-determinism and the size of two-way automata. In *Proc. of STOC 1978*, pages 274–286. ACM Press, 1978.

[18] S. Schewe. Büchi complementation made tight. In *Proc. of STACS 2009*, volume 3 of *LIPIcs*, pages 661–672, 2009.

[19] S. Schewe. Tighter bounds for the determinisation of Büchi automata. In *Proc. of FoSSaCS 2009*, volume 5504 of *LNCS*, pages 167–181. Springer-Verlag, 2009.

[20] S. Schewe and B. Finkbeiner. Satisfiability and finite model property for the alternating-time μ-calculus. In *Proc. of CSL 2006*, volume 4207 of *LNCS*, pages 591–605, 2006.

[21] S. Schewe and B. Finkbeiner. Synthesis of asynchronous systems. In *Proc. of LOPSTR 2006*, volume 4407 of *LNCS*, pages 127–142. Springer-Verlag, 2006.

[22] S. Schewe and T. Varghese. Tight bounds for the determinisation and complementation of generalised Büchi automata. In *Proc. of ATVA 2012*, pages 42–56, 2012.

[23] S. Schewe and T. Varghese. Determinising parity automata. In *Proc. of MFCS 2014*, pages xx–xx, 2014.

[24] A. P. Sistla, M. Y. Vardi, and P. Wolper. The complementation problem for Büchi automata with applications to temporal logic. *Theoretical Computer Science*, 49(3):217–239, 1987.

[25] W. Thomas. Complementation of Büchi automata revisited. In *Jewels are Forever, Contributions on Theoretical Computer Science in Honor of Arto Salomaa*, pages 109–122, 1999.

[26] M.-H. Tsai, Y.-K. Tsay, and Y.-S. Hwang. Goal for games, omega-automata, and logics. volume 8044 of *LNCS*, pages 883–889. Springer-Verlag, 2013.

[27] M. Y. Vardi. The Büchi complementation saga. In *Proc. of STACS 2007*, pages 12–22, 2007.

[28] T. Wilke. Alternating tree automata, parity games, and modal μ-calculus. *Bulletin of the Belgian Mathematical Society*, 8(2), May 2001.

[29] Q. Yan. Lower bounds for complementation of *omega*-automata via the full automata technique. *Journal of Logical Methods in Computer Science*, 4(1:5), 2008.