Why These Automata Types?

Udi Boker[*]

Interdisciplinary Center (IDC) Herzliya, Israel

Abstract

There are various types of automata on infinite words, differing in their acceptance conditions. The most classic ones are weak, Büchi, co-Büchi, parity, Rabin, Streett, and Muller. This is opposed to the case of automata on finite words, in which there is only one standard type. The natural question is why—Why not a single type? Why these particular types? Shall we further look into additional types?

For answering these questions, we clarify the succinctness of the different automata types and the size blowup involved in performing boolean operations on them. To this end, we show that unifying or intersecting deterministic automata of the classic ω-regular-complete types, namely parity, Rabin, Streett, and Muller, involves an exponential size blowup.

We argue that there are good reasons for the classic types, mainly in the case of nondeterministic and alternating automata. They admit good size and complexity bounds with respect to succinctness, boolean operations, and decision procedures, and they are closely connected to various logics.

Yet, we also argue that there is place for additional types, especially in the case of deterministic automata. In particular, generalized-Rabin, which was recently introduced, as well as a disjunction of Streett conditions, which we call hyper-Rabin, where the latter further generalizes the former, are interesting to consider. They may be exponentially more succinct than the classic types, they allow for union and intersection with only a quadratic size blowup, and their nonemptiness can be checked in polynomial time.

1 Introduction

Automata on infinite words were introduced in the 1960s in the course of solving decision problems in logic, and since the 1980s they play a key role in formal verification of reactive systems. Unlike automata on finite words, these automata have various acceptance conditions, the most classic of which are weak, Büchi, co-Büchi, parity, Rabin, Streett, and Muller.

We look into the question of why these automata types: Can we do with a single type? If not, are these particular types the right ones? And do we still need to look for new types?

A simple answer to the question might be "for historical reasons". Yet, though historical events always play a role, they certainly do not give the full answer, and do not explain why these types survived the test of time, while others, such as ∀-automata [42], did not.

A more thorough answer should consider the properties of the different types, as put for example by Kurshan [36]: "The choice of automaton type to use in connection with formal verification is governed by two issues: syntactic suitability and computational complexity. "

Toward answering the question, we look into the following properties of the classic automata types: i) Their succinctness, compared to each other and to an arbitrary automaton type; ii) The size blowup involved in performing boolean operations on them; and iii) The complexity of resolving their decision questions.

[*]This work was supported by the Israel Science Foundation grant 1373/16

Succinctness. Considering automata succinctness, there is a massive literature on translations between the classic types, accumulated along the past 55 years, and continuing to these days. Having "only" seven classic types, where each can be deterministic or nondeterministic, we have 175 possible non-self translations between them, which has become difficult to follow. Moreover, it turns out that there is inconsistency in the literature results concerning the size of automata—Some only consider the number of states, some also take into account the index (namely, the size of the acceptance condition), while ignoring the alphabet size, and some do consider the alphabet size, but ignore the index.

To make an order with all of these results, we maintain a website [5] that provides information and references for each of the possible translations. The high-level tables of the size blowup and of the state blowup involved in the translations are given in Table 1. The size blowup in the table relates to an automaton of size n, and the state blowup to an automaton with n states (and index as large as desired). The capital letters stand for the type names: Weak, Co-Büchi, Büchi, etc. A question mark in the tables stands for an exponential gap between the currently known lower and upper bounds.

As for succinctness with respect to an arbitrary type, we argue in Section 4 that the classic types are "well positioned" with respect to the inevitable tradeoff between succinctness, determinization, and complementation—We observe that for an automaton type T whose complementation only involves a single-exponential size blowup, there must also be at least a single-exponential size blowup in translating arbitrary ω-regular automata into T-automata. In this sense, we show that the classic types, except for Muller, provide a reasonable tradeoff between their succinctness and the size blowup involved in their determinization and complementation, having all of these measures singly exponential.

Boolean operations. There are many works on automata complementation (see [50] for a survey until 2007, after which there are yet many new results), while very few on the positive operations of union and intersection. This is possibly because positive operations are simple on nondeterministic automata, and were less interesting on deterministic automata, as nondeterministic ones are adequate for model checking. However, in recent years there is a vast progress in synthesis and in probabilistic model checking, which require deterministic or almost deterministic automata, such as limit-deterministic [51] or good-for-games automata [26, 8, 10].

We complete the picture of the size blowup involved in boolean operations, as summarized in Table 2. To this end, we provide a new construction for intersecting nondet. Rabin automata with only a quadratic size and state blowup, show that intersecting nondet. Muller automata involves an exponential size blowup, and complementing them a double-exponential blowup. For deterministic automata, we show that unifying or intersecting the classic ω-regular-complete types, namely parity, Rabin, Streett, and Muller, involve an exponential size blowup.

Decision problems. We look into the nonemptiness problem, which allows, in combination with boolean operations, to also solve the other decision problems of equivalence and containment—The language of an automaton $\mathcal{A}$ is contained in the language of an automaton $\mathcal{B}$ iff the intersection of $\mathcal{A}$ with the complement of $\mathcal{B}$ is empty. The complexity of the nonemptiness problem of the classic types is clear in the literature and is summarized in Table 2.

So? Based on these, and other results, we argue that the classic types are interesting and well deserve the attention they get, yet there is a need for additional types, especially in the deterministic setting. There is no inherent reason for having an exponential size blowup in positive boolean operations on deterministic ω-regular-complete automata. These operations are particularly interesting in verification of compound systems, in which setting there may already be some deterministic automata for the individual systems, which are then to be combined.

Indeed, the problem with boolean operations on classic deterministic automata and the current interest in the deterministic setting, may explain the emergence of new, or renewed, automata types in the past five years. Among these are "Emerson-Lei" (EL), which was presented in 1985 [22], and was recently "rediscovered" within the "Hanoi" format [2], "generalized-Rabin" [32], and "generalized-Streett" [3]. The EL condition allows for an arbitrary boolean formula over sets of states that are visited finitely or infinitely often, generalized-Rabin extends the Rabin pairs into lists, and generalized-Streett analogously extends Streett pairs. We analyze these, and some other types, in Section 5, and show that they indeed provide additional benefits.

While positive boolean operations on EL automata are obviously simple, it is known that its nonemptiness check is NP-complete [22] and complementing nondet. EL automata involves a doubly-exponential size blowup [49]. Nevertheless, due to the practical progress in solving the SAT problem, to which the EL condition is naturally related, it may still be interesting to further pursue deterministic EL automata. (See [20, 45].)

We observe that generalized-Rabin is a special case of a disjunction of Streett conditions, which was considered in [22] under the name "canonical form", and which we dub "hyper-Rabin". These types provide an interesting potential, as they may be exponentially more succinct than the classic types, they allow for union and intersection with only a quadratic size blowup, and their nonemptiness check is in PTIME. Indeed, in the deterministic setting there seem to also be practical benefits for generalized-Rabin automata [32, 15, 23], which may possibly be extended to the more general hyper-Rabin condition.

The generalized-Streett condition is analogously a special case of a conjunction of Rabin conditions, which we dub "hyper-Streett". Positive boolean operations on them only involves a quadratic size blowup, yet we show that like EL automata, their nonemptiness check is NP-complete, and complementing their nondeterministic version is doubly-exponential.

Paper structure and main contributions. The paper aims at providing the big picture of ω-regular automata, and along the way provides quite a few new results, the main of which are Theorems 1, 3, 7, 12, and 24.

Table 1 puts an order in the chaos of translations between the different automata types. Section 3 organizes the complexity of boolean operations, as summarized in Table 2, while providing a new algorithm for intersecting Rabin automata, and several new negative results on the exponential size blowup involved in some boolean operations. Section 4 is mostly of a non-technical nature, providing reasons for why the classic types are indeed so. Section 5 analyzes a family of types that are stronger than the classical ones, as summerized in Tables 3 and 4, while providing new results on the size blowup involved in boolean operations on them and in translations between them.

2 Preliminaries

A *word* over a finite alphabet Σ is a sequence $w = w(0) \cdot w(1) \cdots$ of letters in Σ.

A *nondeterministic automaton* is a tuple $\mathcal{A} = \langle \Sigma, Q, \delta, \iota, \alpha \rangle$, where Σ is the input alphabet, Q is a finite set of states, $\delta : Q \times \Sigma \to 2^Q$ is a transition function, $\iota \subseteq Q$ is a set of initial states, and α is an acceptance condition. The first four elements, namely $\langle \Sigma, Q, \delta, \iota \rangle$, are the automaton's *structure*. In the case where $|\iota| = 1$ and for every $q \in Q$ and $\sigma \in \Sigma$, we have $|\delta(q, \sigma)| \leq 1$, we say that $\mathcal{A}$ is *deterministic*.

Size Blowup (From \ To)		Det. W	Det. C	Det. B	Det. P	Det. R	Det. S	Det. M	Nondet. W	Nondet. C	Nondet. B	Nondet. P	Nondet. R	Nondet. S	Nondet. M
Det.	W	.							.						
	C		.							.		$\Theta(n)$			
	B			.	$\Theta(n)$						.				
	P				.			$2^{\Theta(n)}$			$\Theta(n^2)$	.			$2^{\Theta(n)}$
	R		$2^{\Theta(n)}$			.			$O(n^2)$ $\Omega(n)$			$O(n^2)$ $\Omega(n)$	.	$O(n^2)$ $\Omega(n)$	
	S			$2^{\Theta(n)}$		$\Theta(2^{n \log n})$	.		$\Theta(n)$			$2^{\Theta(n)}$		.	
	M		$O(n^2)$ $\Omega(n)$	?		$O(2^{n \log n})$ $2^{\Omega(n)}$		.	$O(n^2)$ $\Omega(n)$		$\Theta(n^3)$		$\Theta(n^2)$		.
Non-Det.	W								.						
	C		$2^{\Theta(n)}$							.		$\Theta(n)$			
	B				$2^{\Theta(n \log n)}$			$*^1$			.				
	P							$2^{2^{\Omega(n)}}$			$\Theta(n^2)$	.			$2^{\Theta(n)}$
	R			$*^2$		$2^{\Theta(n^2 \log n)}$			$2^{\Theta(n)}$			$O(n^2)$ $\Omega(n)$	.	$O(n^2)$ $\Omega(n)$	
	S													.	
	M			$*^3$		$2^{O(n^3 \log n)}$ $2^{\Omega(n \log n)}$					$\Theta(n^3)$		$\Theta(n^2)$		.

State Blowup (From \ To)		Det. W	Det. C	Det. B	Det. P	Det. R	Det. S	Det. M	Nondet. W	Nondet. C	Nondet. B	Nondet. P	Nondet. R	Nondet. S	Nondet. M
Det.	W	.							.						
	C		.							.		$\Theta(n)$			
	B			.	$\Theta(n)$						.				
	P				.						$\Theta(n^2)$	.			
	R		$2^{\Theta(n)}$			.							.		
	S			$2^{\Theta(n)}$		$\Theta(2^{n \log n})$	.		$\Theta(n)$		$2^{\Theta(n)}$		$\Theta(n^2)$	.	
	M							.							.
Non-Det.	W								.						
	C		$2^{\Theta(n)}$							.		$\Theta(n)$			
	B					$2^{\Theta(n \log n)}$					.				
	P										$\Theta(n^2)$	.			
	R			?		$2^{2^{\Theta(n)}}$			$2^{\Theta(n)}$				.		
	S			$*^4$	?	$*^5$	?	$*^4$					$\Theta(n^2)$	.	
	M			?		$2^{2^{\Theta(n)}}$									.

$*^1$: Upper bounds between $2^{2^{O(n)}}$ and $2^{2^{O(n^3 \log n)}}$ $*^2$: Lower bound to DBW: $2^{\Omega(n \log n)}$
$*^3$: Lower bound to DBW: $2^{\Omega(n)}$ $*^4$: $2^{O(n^2 \log n)}$ and $2^{\Omega(n \log n)}$ $*^5$: $2^{\Theta(n^2 \log n)}$

Table 1: Size blowup and state blowup involved in automata translations [5].

A *run* $r = r(0), r(1), \cdots$ of $\mathcal{A}$ on $w = w(0) \cdot w(1) \cdots \in \Sigma^\omega$ is an infinite sequence of states such that $r(0) \in \iota$, and for every $i \geq 0$, we have $r(i+1) \in \delta(r(i), w(i))$.

Acceptance is defined with respect to the set $\mathit{inf}(r)$ of states that the run r visits infinitely often, for which reason these automata are called *ω-regular automata*. Formally, $\mathit{inf}(r) = \{q \in Q \mid \text{ for infinitely many } i \in \mathbb{N}, \text{ we have } r(i) = q\}$. We describe below the most classic types of acceptance conditions. In Section 5, we will describe additional types.

- *Büchi*, where $\alpha \subseteq Q$, and r is accepting iff $\mathit{inf}(r) \cap \alpha \neq \emptyset$. (The states of α are *accepting*.)
- *co-Büchi*, where $\alpha \subseteq Q$, and r is accepting iff $\mathit{inf}(r) \cap \alpha = \emptyset$. (The states of α are *rejecting*.)
- *weak* is a special case of the Büchi condition, where every strongly connected component of the automaton is either contained in α or disjoint to α.
- *parity*, where $\alpha = \{S_1, S_2, \dots, S_{2k}\}$ with $S_1 \subset S_2 \subset \cdots \subset S_{2k} = Q$, and r is accepting if the minimal i for which $\mathit{inf}(r) \cap S_i \neq \emptyset$ is even.
- *Rabin*, where $\alpha = \{\langle B_1, G_1\rangle, \langle B_2, G_2\rangle, \dots, \langle B_k, G_k\rangle\}$, with $B_i, G_i \subseteq Q$ and r is accepting iff for some $i \in [1..k]$, we have $\mathit{inf}(r) \cap B_i = \emptyset$ and $\mathit{inf}(r) \cap G_i \neq \emptyset$.
- *Streett*, where $\alpha = \{\langle B_1, G_1\rangle, \langle B_2, G_2\rangle, \dots, \langle B_k, G_k\rangle\}$, with $B_i, G_i \subseteq Q$ and r is accepting iff for all $i \in [1..k]$, we have $\mathit{inf}(r) \cap B_i = \emptyset$ or $\mathit{inf}(r) \cap G_i \neq \emptyset$.
- *Muller*, where $\alpha = \{\alpha_1, \alpha_2, \dots, \alpha_k\}$, with $\alpha_i \subseteq Q$ and r is accepting iff for some $i \in [1..k]$, we have $\mathit{inf}(r) = \alpha_i$.

Notice that Büchi and co-Büchi are special cases of the parity condition, which is in turn a special case of both the Rabin and Streett conditions. In the latter conditions, we refer to the B_i and G_i sets as the "bad" and "good" sets, respectively.

The number of sets in the parity and Muller acceptance conditions or pairs in the Rabin and Streett acceptance conditions is called the *index* of the automaton. For weak, co-Büchi, and Büchi automata, the index is 1.

The *size* of an automaton is the maximum size of its elements; that is, it is the maximum of the alphabet length, the number of states, the number of transitions, and the index.

An automaton accepts a word if it has an accepting run on it. The language of an automaton $\mathcal{A}$, denoted by $L(\mathcal{A})$, is the set of words that $\mathcal{A}$ accepts. We also say that $\mathcal{A}$ *recognizes* the language $L(\mathcal{A})$. Two automata, $\mathcal{A}$ and $\mathcal{A}'$, are *equivalent* iff $L(\mathcal{A}) = L(\mathcal{A}')$.

The *class* of an automaton characterizes its transition mode (deterministic, nondeterministic, or alternating) and its acceptance condition. We sometimes abbreviate the different classes by three letter acronyms in $\{\text{D, N, A}\} \times \{\text{W, B, C, P, R, S, M}\} \times \{\text{W, T}\}$. The first letter stands for the transition mode; the second for the acceptance-condition (weak, Büchi, co-Büchi, parity, Rabin, Streett, or Muller); and the third indicates whether the automaton runs on Words or on Trees. For example, DBW stands for a deterministic Büchi automaton on words.

Büchi, parity, Rabin, Streett, and Muller automata are *ω-regular complete*, recognizing all ω-regular languages. Weak, co-Büchi, and deterministic Büchi automata, are less expressive.

3 Classic Types – Boolean Operations

In the nondeterministic setting, boolean operations on the classic automata types, except for Muller, roughly involve an asymptotically optimal size blowup: linear for union, quadratic for intersection, and singly exponential for complementation. In the deterministic setting, however, the picture is different, having an exponential size blowup on union or intersection for all of the ω-regular-complete types. We provide below known and new results on the boolean operations, and summarize them in Table 2.

3.1 Nondeterministic Automata

Union is simply achieved by adding up the two automata via an initial nondeterministic choice.

Intersection requires at least a quadratic blowup, already for automata on finite words, and can indeed be done with that size blowup for most classic types. For parity automata the exact blowup is still to be resolved. Intersection of Streett automata is straightforward due to the conjunctive nature of the acceptance condition, while for Rabin automata it is more involved, and given here for the first time. For Muller automata, we prove that it is exponential.

Complementation involves at least an exponential blowup, already for automata on finite words, and for all classic types, except for Muller, it involves a size blowup between $2^{\Theta(n)}$ and $2^{\Theta(n^2 \log n)}$. Yet, the state blowup for Rabin automata (with index exponential in the number of states) might be doubly exponential [14]. For Muller automata, we prove that the size blowup is doubly exponential.

We elaborate below on the intersection operation for each of the types, considering input automata $\mathcal{A}$ and $\mathcal{B}$. For Muller, we also elaborate on complementation.

Weak and Co-Büchi. Intersection is done directly on the product automaton $\mathcal{A} \times \mathcal{B}$, with the definition that a state (a, b) is accepting iff a is accepting in $\mathcal{A}$ and b is accepting in $\mathcal{B}$. Notice that the resulting automaton retains the weak property, when both $\mathcal{A}$ and $\mathcal{B}$ have it, and accepts exactly the words in $L(\mathcal{A}) \cap L(\mathcal{B})$, as the runs of both $\mathcal{A}$ and $\mathcal{B}$ eventually remain in only accepting/rejecting states.

Büchi. For Büchi intersection, the product automaton is not enough, since parallel accepting runs of $\mathcal{A}$ and $\mathcal{B}$ need not visit their accepting states simultaneously. Nevertheless, the standard intersection construction only needs two copies of the product automaton, where a jump from one component to the other is done once visiting states accepting w.r.t. $\mathcal{A}$ (resp. $\mathcal{B}$), thus guaranteeing infinitely many visits in the accepting states of both $\mathcal{A}$ and $\mathcal{B}$.

Parity. A parity automaton can be translated to a Büchi automaton with only a quadratic state and size blowup [17]. Thus, intersection involves up to a quartic state and size blowup (yielding a Büchi automaton). The exact inevitable blowup in direct intersection of parity automata into a parity automaton is yet to be resolved.

Streett. Streett intersection can be done directly on the product automaton, taking advantage of the conjunctive nature of the Streett condition.

Rabin. A Rabin automaton can be translated to a Büchi automaton with only a quadratic size blowup [17], implying intersection construction with up to a quartic size blowup. However, state wise, this approach is inadequate, as there might be an exponential blowup in the translation of Rabin to Büchi automata [7]. We show below that the Rabin condition allows for a direct intersection construction, involving only a quadratic state and size blowup.

The idea is to extend the Büchi-intersection construction, taking advantage of the Rabin condition. Recall that in the construction for intersecting Büchi automata, a jump from one copy of the product automaton into the other is done once reaching a state accepting w.r.t. $\mathcal{A}$ (respectively $\mathcal{B}$). This does not work for the Rabin condition, since a state is no longer accepting w.r.t. $\mathcal{A}$, but may belong to several "good" and "bad" sets of $\mathcal{A}$'s accepting pairs.

We extend the construction by adding a "bridge" that is visited when jumping from the first to the second copy of the product automaton, and another bridge between the second and first copy. A bridge is a limited copy of the product automaton, in which all states can only move to the next copy. Then, for every acceptance pairs $\langle B_1, G_1\rangle$ of $\mathcal{A}$ and $\langle B_2, G_2\rangle$ of $\mathcal{B}$, we define an acceptance pair $\langle B, G\rangle$ that enforces a visit in G_1 when going through the first bridge and a visit in G_2 when in the second bridge. This enforcement is done by having in B all the states of the first bridge that are not in G_1 and all the states of the second bridge that are not in G_2.

An example of intersecting Rabin automata along this construction is illustrated in Figure 1.

Theorem 1. *For every two NRWs $\mathcal{A}_1$ and $\mathcal{A}_2$ with n_1 and n_2 states, m_1 and m_2 transitions, and indices k_1 and k_2, respectively, there is an NRW recognizing $L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$ with $4n_1n_2$ states, $6m_1m_2$ transitions, and index k_1k_2.*

Proof.
Construction. Consider NRWs $\mathcal{A}_1 = \langle \Sigma, Q_1, \iota_1, \delta_1, \alpha_1\rangle$ and $\mathcal{A}_2 = \langle \Sigma, Q_2, \iota_2, \delta_2, \alpha_2\rangle$. We define the NRW $\mathcal{A} = \langle \Sigma, Q, \iota, \delta, \alpha\rangle$, which we claim to recognize $L(\mathcal{A}) \cap L(\mathcal{B})$, as follows.

- $Q = Q_1 \times Q_2 \times [0..3]$. (We shall call each instance of $Q_1 \times Q_2$ a "component", and the odd components we dub "bridges".)
- $\iota = \iota_1 \times \iota_2 \times \{0\}$.
- For every state $\langle q_1, q_2, i\rangle \in Q$ and $\sigma \in \Sigma$, the transition function is defined as follows.
  - If i is even, then $\delta(\langle q_1, q_2, i\rangle, \sigma) = \{\langle \hat{q_1}, \hat{q_2}, \hat{i}\rangle \mid \hat{q_1} \in \delta_1(q_1, \sigma),\ \hat{q_2} \in \delta_2(q_2, \sigma)$, and $\hat{i} \in \{i, i{+}1\}\}$.
  - If i is odd, then $\delta(\langle q_1, q_2, i\rangle, \sigma) = \{\langle \hat{q_1}, \hat{q_2}, (i{+}1) \bmod 4\rangle \mid \hat{q_1} \in \delta_1(q_1, \sigma)$ and $\hat{q_2} \in \delta_2(q_2, \sigma)\}$.
- For every acceptance pairs $\langle B_1, G_1\rangle \in \alpha_1$ and $\langle B_2, G_2\rangle \in \alpha_2$, we have in α the acceptance pair $\langle B, G\rangle$, where B and G are defined as follows.
  - $G = G_1 \times Q_2 \times \{1\}$.
  - B is the union of two sets B' and B''. The first includes all the states that are bad w.r.t. B_1 or B_2. The second handles the transitions through the bridges, adding in component $2i-1$ the states that are not in G_i. Formally, $B = B' \cup B''$, where $B' = B_1 \times Q_2 \times [0..3] \cup Q_1 \times B_2 \times [0..3]$ and $B'' = (Q_1 \setminus G_1) \times Q_2 \times \{1\} \cup Q_1 \times (Q_2 \setminus G_2) \times \{3\}$.

Correctness. Consider a word $w \in L(\mathcal{A}_1) \cap L(\mathcal{A}_2)$. Then there are runs r_1 of $\mathcal{A}_1$ and r_2 of $\mathcal{A}_2$, and acceptance pairs $\langle B_1, G_1\rangle \in \alpha_1$ and $\langle B_2, G_2\rangle \in \alpha_2$, such that r_i visits finitely often B_i and infinitely often G_i, for $i \in \{1, 2\}$. For showing that $w \in L(\mathcal{A})$, we will show that there is a run r of $\mathcal{A}$ that visits finitely often B and infinitely often G, where B and G are defined from $\langle B_1, G_1\rangle$ and $\langle B_2, G_2\rangle$ as described in the construction.

The leftmost projection of r, namely the q_1 element of a $\langle q_1, q_2, i\rangle$ state that r visits, is identical to r_1 and the middle projection is identical to r_2. We will describe the rightmost projection of r, namely the series of components that r traverses along the run. Notice that when r is in a bridge (an odd component) it must move to the next component, and when it is in an even component it has the choice of whether to stay there or move to the next one. We explain next how r behaves in the even components.

Let t be the first position of w after which both r_1 and r_2 no longer visit states in B_1 and B_2, respectively. The run r remains in the component 0 until position t. After position t, when r is in a component $2(i-1)$, for $i \in \{1, 2\}$, it remains there until the next state of r_i is in G_i, in which case it proceeds to the next component.

Observe that r satisfies the acceptance pair $\langle B, G\rangle$: Considering the "good" set G, notice first that r visits all the components infinitely often, since it cannot stay in odd components for two consecutive positions, and cannot stay in a component $2(i-1)$ when $\mathcal{A}_i$ visits G_i (after position t), which happens infinitely often. Then, whenever in the component 1 after position t, it visits a state in $G_1 \times Q_2 \times \{1\}$, which guarantees infinitely many visits in G. Considering the "bad" set $B = B' \cup B''$, the states of B' are visited only finitely often, since the left and middle projections of r are identical to r_1 and r_2, respectively, and they visit B_1 and B_2, respectively, only finitely often. As for B'', its states are visited only finitely often, since the only case in which r visits after position t a state $\langle q_1, q_2, 2i-1\rangle$, for $i \in \{1, 2\}$, is when $q_i \in G_i$.

As for the other direction, consider a word $w \in L(\mathcal{A})$. Then there is a run r of $\mathcal{A}$ that satisfies some acceptance pair $\langle B, G\rangle$ of $\mathcal{A}$. By the construction of $\mathcal{A}$, the pair $\langle B, G\rangle$ corresponds to some pairs $\langle B_1, G_1\rangle \in \alpha_1$ and $\langle B_2, G_2\rangle \in \alpha_2$. We claim that the left (resp. middle) projection of r is a run r_i of $\mathcal{A}_i$ that satisfies $\langle B_i, G_i\rangle$, for $i \in \{1, 2\}$, respectively.

First observe that due to the subset B' of B, the run r visits states whose leftmost projection is in B_1 or middle projection in B_2 only finitely often. Hence, r_1 and r_2 visit finitely often B_1 and B_2, respectively. Next, observe that r must visit infinitely often all components—it visits $G_1 \times Q_2 \times \{1\}$ infinitely often, and going from component 1 back to itself enforces a visit in all components. Now, by the subset B'' of B, the states that r visits infinitely often in component 1 must be in $G_1 \times Q_2 \times \{1\}$ and in component 3 they must be in $Q_1 \times G_2 \times \{3\}$. Hence, r_i visits infinitely often G_i, for $i \in \{1, 2\}$.

□

Muller. A Muller automaton can be translated to a Büchi automaton with only a cubic size blowup [6]. However, when considering the intersection of Muller automata into a Muller automaton, one cannot take advantage of the quadratic intersection of Büchi automata, as the translation back from Büchi to Muller automata might involve an exponential size blowup [48]. Moreover, we show below that due to the very descriptive nature of the Muller condition, intersection might involve an exponential size blowup.

Theorem 2. *Intersection of (deterministic) Muller automata involves an exponential size blowup. In particular, for every $n \geq 1$, there are DMWs $\mathcal{M}'_n$ and $\mathcal{M}''_n$ with up to n states, $2n$ transitions, and index $n+1$ each, over an alphabet of 3 letters, such that every NMW that recognizes $L(\mathcal{M}'_n) \cap L(\mathcal{M}''_n)$ has an index of at least 2^n.*

Proof. Consider the DMWs $\mathcal{D}'_n$ and $\mathcal{D}''_n$ of Figure 2. Observe that both $\mathcal{D}'_n$ and $\mathcal{D}''_n$ accept all infinite runs on their structures, and are thus equivalent to the corresponding weak automaton on their structures, all of whose states are accepting. Thus, the language $L(\mathcal{D}'_n) \cap L(\mathcal{D}''_n)$ is recognized by the weak automaton $\mathcal{D}_n$, whose structure is the product of $\mathcal{D}'_n$ and $\mathcal{D}''_n$'s structures. Now, in [6, proof of Theorem 12], it is shown that an NMW equivalent to $\mathcal{D}_n$ must have an index of at least 2^n. □

Complementation of Muller automata is also very inefficient, involving a doubly exponential size blowup.

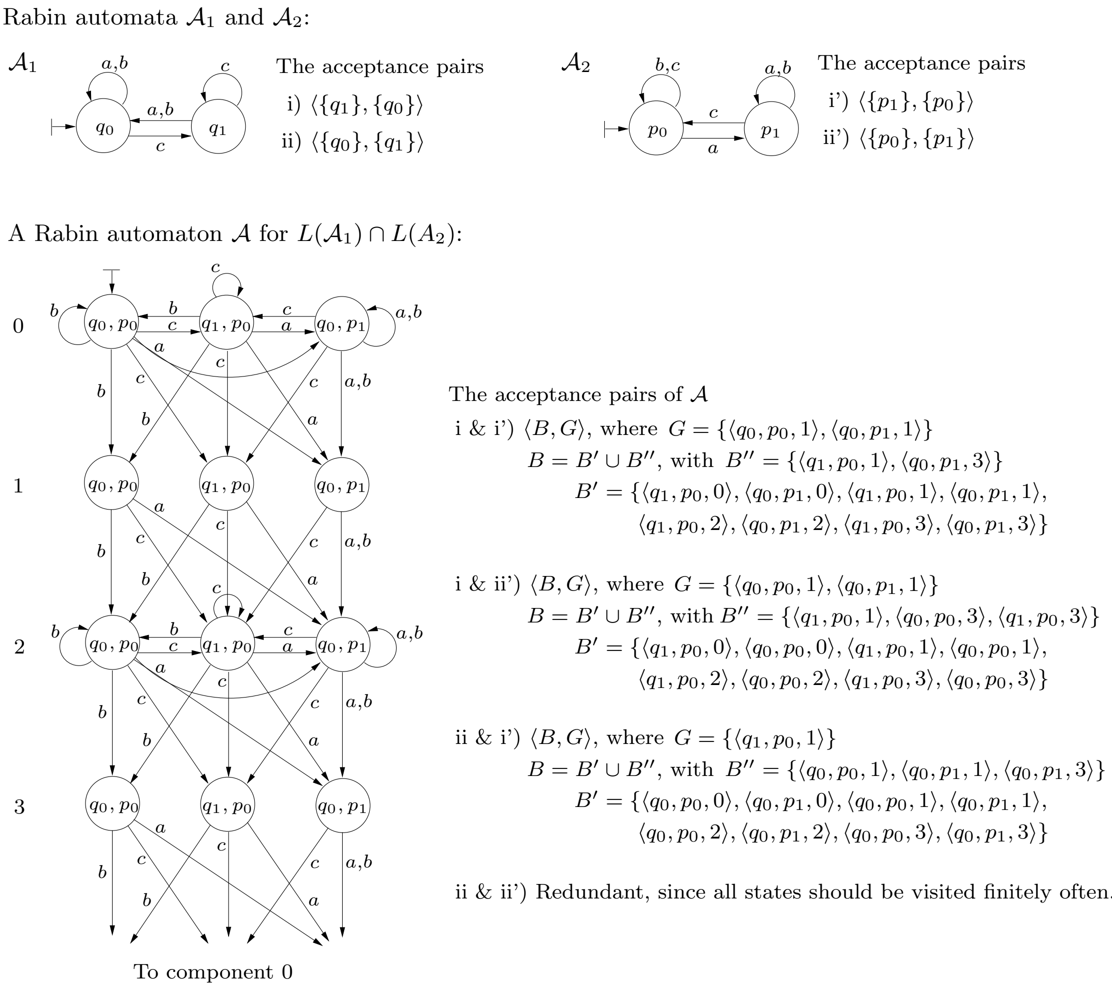


Figure 1: An example of intersecting Rabin automata with a quadratic state and size blowup, as per the construction of Theorem 1.

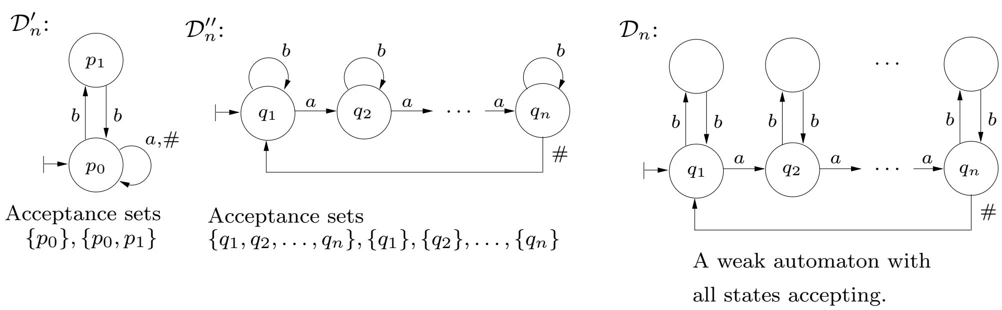


Figure 2: Deterministic Muller automata $\mathcal{D}'_n$ and $\mathcal{D}''_n$ of size in $O(n)$, whose intersection, which is equivalent to $\mathcal{D}_n$, is recognized by a nondeterministic Muller automaton with index of at least 2^n.

Theorem 3. *Complementation of Muller automata into Muller automata involves a doubly-exponential size blowup. In particular, for every $n \geq 1$, there is a language L_n over an alphabet of 4 letters, recognized by an NMW with $O(n)$ states and index in $O(1)$, while an NMW for $\overline{L_n}$ has an index of at least 2^{2^n}.*

Proof. Let $\Sigma = \{a, b, \#, \$\}$. For every $n \geq 1$, consider the language $H_n = \{u_1\#u_1\$u_2\#u_2\$\ldots \mid$ for every $i \in \mathbb{N}, u_i \in \{a,b\}^n\}$ over Σ. That is, H_n consists of infinite words that are sequences of pairs of finite words of length n over $\{a,b\}$, such that the two words in each pair are identical. Let L_n be the complement of H_n. We claim that an NMW for H_n has an index of at least 2^{2^n}, while there is an NMW of size in $O(n)$ recognizing L_n.

An NMW of size in $O(n)$ for L_n is straightforward. There are two options for a word not in H_n: either it violates the pattern of n $\{a,b\}$ letters between a # and a \$, or there are two different letters that are n positions apart and there is a # between them. Both cases can be recognized by a nondeterministic finite automaton with $O(n)$ states, after which there is a forever accepting state.

We continue with analyzing an NMW $\mathcal{M}$ for H_n. Let $U = \{u \mid u \in \{a,b\}^n\}$ be the set of words of length n over $\{a,b\}$. For every subset $T \subseteq U$, let t be the number of words in T, and let $w_T = (u_1\#u_1\$u_2\#u_2\$\ldots u_t\#u_t\$)^\omega$ be an infinite word that repeats exactly the words in T. Notice that for every such T, the word w_T belongs to H_n. Let S_T be an acceptance set of $\mathcal{M}$ via which $\mathcal{M}$ accepts w_T. We show that for every $T \neq T'$, we have $S_T \neq S_{T'}$.

Indeed, for every word $u \in U$, consider the set S_u of states of $\mathcal{M}$, such that $s \in S_u$ iff there is a finite word x ending with u and an infinite word y starting with $\#u$, such that there is an accepting run r of $\mathcal{M}$ on xy, and $\mathcal{M}$ reaches s along r after reading x. We claim that for every $u \neq u' \in U$, we have S_u and $S_{u'}$ are disjoint. Indeed, assume by way of contradiction a state $s \in S_u \cap S_{u'}$. Let r and r' be the runs of $\mathcal{M}$ witnessing that $s \in S_u$ and $s \in S_{u'}$, and reaching s at positions p and p', respectively. Then the run $\hat{r}$ that starts as r until position p and then continues as r' from the latter's position p' is a legal run of $\mathcal{M}$ and it is accepting, since the set of states that it visits infinitely often is the same as that of r'. Yet, it accepts a word not in H_n, having an infix $u\#u'$.

Hence, for every $T \neq T'$ with some $u \in T \setminus T'$, we have S_T contains some state $s \in S_u$, such that $s \notin S_{T'}$. Thus, since there are 2^{2^n} different sets $T \subseteq U$, $\mathcal{M}$ must have an index of at least 2^{2^n}. ☐

3.2 Deterministic Automata

We show below that positive boolean operations might involve an exponential size and state blowup over deterministic parity, Rabin, Streett, and Muller automata.

Rabin and Streett. We start with straightforward positive results on constructions that are natural to each type.

Proposition 4. *Intersection of (deterministic) Streett automata and union of deterministic Rabin automata can be done with quadratic state and size blowup.*

Proof. The NSW $\mathcal{S}$ can be defined over the product of $\mathcal{S}'$ and $\mathcal{S}''$, where the acceptance condition is taken to be the union of the two accepting conditions. When both $\mathcal{S}'$ and $\mathcal{S}''$ are deterministic, so is $\mathcal{S}$.

Formally, let $\mathcal{S}' = (\Sigma, Q', \iota', \delta', \alpha')$ and $\mathcal{S}'' = (\Sigma, Q'', \iota'', \delta'', \alpha'')$. We define $\mathcal{S} = (\Sigma, Q, \iota, \delta, \alpha)$, where

- $Q = Q' \times Q''$
- $\iota = \langle \iota', \iota'' \rangle$
- For every $q' \in Q', q'' \in Q''$, and $a \in \Sigma$, $\delta(\langle q', q'' \rangle, a) = \langle \delta'(q', a), \delta''(q'', a) \rangle$
- $\alpha = \{\langle B' \times Q'', G' \times Q'' \rangle \mid \langle B', G' \rangle \in \alpha'\} \cup \{\langle Q' \times B'', Q' \times G'' \rangle \mid \langle B'', G'' \rangle \in \alpha''\}$

By the duality of deterministic Streett and Rabin automata, the result follows for the union of DRWs $\mathcal{R}'$ and $\mathcal{R}''$: $L(\mathcal{R}'_n) \cap L(\mathcal{R}''_n) = \overline{L(\mathcal{S}'_n) \cup L(\mathcal{S}''_n)}$. □

For showing the exponential size blowup in the intersection of det. Rabin automata and union of det. Streett automata, we leverage a recent lower-bound proof from [1].

Consider the family of languages $\{L_n\}_{n \geq 1}$, where the alphabet of L_n is $\{1, 2, \ldots, n\}$ and a word belongs to it iff the following two conditions are met:

i. A letter i is always followed by a letter j, such that $i-1 \leq j \leq i+1$. For example, $5433245\ldots$ is a bad prefix, since 2 was followed by 4, while $55434322\ldots$ is a good prefix.

ii. The number of letters that appear infinitely often is odd. For example, $23321(22343233)^\omega$ is in L_n, while $1(233)^\omega$ is not.

We provide for every n, det. Streett automata $\mathcal{S}'_n$ and $\mathcal{S}''_n$ of size n, such that $L(\mathcal{S}'_n) \cup L(\mathcal{S}''_n) = L_n$. Intuitively, $\mathcal{S}'_n$ requires both the smallest and biggest letters that appear infinitely often to be odd, while $\mathcal{S}''_n$ requires both of them to be even. This covers, together with the enforcement of condition (i), all cases in which the number of letters that appear infinitely often is odd.

Lemma 5. *For every $n \geq 1$, there are deterministic Streett automata $\mathcal{S}'_n$ and $\mathcal{S}''_n$ of size n, such that $L(\mathcal{S}'_n) \cup L(\mathcal{S}''_n) = L_n$.*

Proof. We define the automata $\mathcal{S}' = (\Sigma, Q, \iota, \delta, \alpha')$ and $\mathcal{S}' = (\Sigma, Q, \iota, \delta, \alpha')$, where $\Sigma = \{1, 2, \ldots, n\}$; $Q = \{q_1, q_2, \ldots, q_n\}$; $\iota = q_n$; and for every $i, j \in [1..n]$, $\delta(q_i, j) = q_j$ if $i-1 \leq j \leq i+1$, and nothing otherwise.

The acceptance condition of $\mathcal{S}'$ ensures that the smallest letter that appears infinitely often is odd, by having the pair $\langle \{q_i\}, \{q_{i-1}\} \rangle$ for every even number i, and it ensures that the biggest letter that appear infinitely often is odd, by having the pair $\langle \{q_i\}, \{q_{i+1}\} \rangle$ for every even number that is smaller than n. In the case that n is even, it also have the pair $\langle \{q_n\}, \emptyset \rangle$.

Analogously, the acceptance condition of $\mathcal{S}''$ ensures that both the smallest and biggest letters that appear infinitely often are even.

As the transition function δ ensures the condition (i) in the definition of L_n, we have $L(\mathcal{S}'_n) \cup L(\mathcal{S}''_n) = L_n$. □

Yet, deterministic Rabin and Streett automata for L_n must have at least $2^{\lfloor \frac{n-1}{2} \rfloor}$ states.

Lemma 6 ([1, Lemma 5.11 and Remark 5.13]). [1] *Every deterministic Rabin and Streett automata recognizing L_n must have at least $2^{\lfloor \frac{n-1}{2} \rfloor}$ states.*

By the above lemmas, we get the exponential blowup in intersecting deterministic Rabin automata and unifying deterministic Streett automata.

[1] The statement of Lemma 5.11 in [1] speaks of a family of languages that is slightly different from the one presented here, yet its proof considers a variety of language families, including the one used here.

Theorem 7. *Union of det. Streett automata and intersection of det. Rabin automata involve an exponential state and size blowup. In particular, for every* $n \geq 1$*, there are DSWs* $\mathcal{S}'_n$ *and* $\mathcal{S}''_n$ *(over the same structure) with* n *states,* $3n-2$ *transitions, and* n *accepting pairs over an alphabet of* n *letters, s.t. every DSW that recognizes* $L(\mathcal{S}'_n) \cup L(\mathcal{S}''_n)$ *has at least* $2^{\lfloor \frac{n-1}{2} \rfloor}$ *states.*

Proof. The case of Streett automata directly follows from Lemmas 5 and 6. As for the intersection of Rabin automata, the result follows from the duality to Streett: Let $\mathcal{R}'_n$ and $\mathcal{R}''_n$ be the DRWs that result from dualizing $\mathcal{S}'$ and $\mathcal{S}''$, respectively. Since $L(\mathcal{S}'_n) \cup L(\mathcal{S}''_n) = \overline{L(\mathcal{R}'_n) \cap L(\mathcal{R}''_n)}$, we have that a DRW that recognizes $L(\mathcal{R}'_n) \cap L(\mathcal{R}''_n)$ must have at least $2^{\lfloor \frac{n-1}{2} \rfloor}$ states. ☐

Parity and generalized parity. Complementation of det. parity automata involves no blowup, however both union and intersection might involve an exponential state and size blowup, as shown below. (This exponential blowup was independently shown by Christof Löding and Haidi Yue, while investigating the memory requirements in two-player infinite games [40].)

We first construct for every n, four deterministic parity automata over the same structure, such that their union and intersection provide L_n.

Lemma 8. *For every* $n \geq 1$*, there are deterministic parity automata* $\mathcal{P}'_n, \mathcal{P}''_n, \mathcal{P}'''_n$*, and* $\mathcal{P}''''_n$ *of size* n *over the same structure, such that* $L(\mathcal{P}'_n) \cap L(\mathcal{P}''_n) \cup L(\mathcal{P}'''_n) \cap L(\mathcal{P}''''_n) = L_n$*.*

Proof. Let $\mathcal{P}' = (\Sigma, Q, \iota, \delta, \alpha')$, $\mathcal{P}'' = (\Sigma, Q, \iota, \delta, \alpha'')$, $\mathcal{P}''' = (\Sigma, Q, \iota, \delta, \alpha''')$, and $\mathcal{P}'''' = (\Sigma, Q, \iota, \delta, \alpha'''')$ be parity automata, where $\Sigma = \{1, 2, \ldots, n\}$; $Q = \{q_1, q_2, \ldots, q_n\}$; $\iota = q_n$; and for every $i, j \in [1..n], \delta(q_i, j) = q_j$ if $i-1 \leq j \leq i+1$, and nothing otherwise. The acceptance conditions color the states with $n+1$ colors, namely they are functions from Q to $\langle 0, 1, 2, \ldots, n\rangle$.

The acceptance condition of $\mathcal{P}'$ ensures that the smallest letter that appears infinitely often is even, by the coloring $\alpha'(q_i) = i$. Analogously, $\mathcal{P}'''$ ensures that the smallest letter that appears infinitely often is odd, by the coloring $\alpha'''(q_i) = i - 1$.

Now, $\mathcal{P}''$ ensures that the biggest letter that appears infinitely often is even via $\alpha''(q_i) = n-i$, in case that n is even, and $\alpha''(q_i) = n+1-i$, in case that n is odd. Analogously, $\mathcal{P}''''$ ensures that the biggest letter that appears infinitely often is odd via $\alpha''''(q_i) = n - i$, in case that n is odd, and $\alpha''''(q_i) = n+1-i$, in case that n is even.

As the transition function δ ensures the condition (i) in the definition of L_n, we have $L(\mathcal{P}'_n) \cap L(\mathcal{P}''_n) \cup L(\mathcal{P}'''_n) \cap L(\mathcal{P}''''_n) = L_n$. ☐

Now, using the above lemma and the negative results on unifying det. Streett automata, we get the negative results on det. parity automata.

Theorem 9. *Union and intersection of deterministic parity automata involve an exponential state and size blowup. In particular, for every* $n \geq 1$*, there are DPWs* $\mathcal{P}'_n$ *and* $\mathcal{P}''_n$ *(over the same structure) with* n *states,* $3n-2$ *transitions, and* n *colors over an alphabet of* n *letters, such that every DPW that recognizes* $L(\mathcal{P}'_n) \cap L(\mathcal{P}''_n)$ *has at least* $2^{\lfloor \frac{n-1}{4} \rfloor}$ *states.*

Proof. For every $n \geq 1$, consider the DPWs $\mathcal{P}'_n, \mathcal{P}''_n, \mathcal{P}'''_n$, and $\mathcal{P}''''_n$ as per Lemma 8. Let $\mathcal{A}$ and $\mathcal{B}$ be DPWs recognizing $L(\mathcal{P}'_n) \cap L(\mathcal{P}''_n)$ and $L(\mathcal{P}'''_n) \cap L(\mathcal{P}''''_n)$, respectively. Observe that $\mathcal{A}$ and $\mathcal{B}$ are also DRWs, and $L(\mathcal{A}) \cup L(\mathcal{B}) = L_n$.

By Proposition 4, the union of DRWs involves at most a quadratic size blowup. Now, by Lemma 6, a DRW that recognizes L_n must have at least $2^{\lfloor \frac{n-1}{2} \rfloor}$ states. Hence, either of $\mathcal{A}$ or $\mathcal{B}$ must be of a size that is at least the square root of $2^{\lfloor \frac{n-1}{2} \rfloor}$, which is $2^{\lfloor \frac{n-1}{4} \rfloor}$. This concludes the claim concerning the intersection of DPWs.

As for the union of DPWs, we get the result from the self duality of parity automata: Assume that $\mathcal{P}'$ and $\mathcal{P}''$ are the DPWs whose intersection involves the exponential size blowup. Let $\mathcal{D}'$ and $\mathcal{D}''$ be the DPWs that result from dualizing $\mathcal{P}'$ and $\mathcal{P}''$, respectively. Since $\overline{L(\mathcal{P}'_n) \cap L(\mathcal{P}''_n)} = L(\mathcal{D}'_n) \cup L(\mathcal{D}''_n)$, we have that a DPW that recognizes $L(\mathcal{D}'_n) \cup L(\mathcal{D}''_n)$ must have at least $2^{\lfloor \frac{n-1}{4} \rfloor}$ states. □

The parity condition is also considered in the literature in a generalized form, called "generalized parity" [16], allowing for either a disjunction or a conjunction of standard parity conditions. Analogously to Proposition 4, unifying disjunctive-generalized-parity automata, as well as intersecting conjunctive-generalized-parity automata, is easy. Yet, the dual operations involve an exponential size blowup.

Corollary 10. *Union of deterministic conjunctive-generalized-parity automata and intersection of disjunctive-generalized-parity automata involve an exponential state and size blowup.*

Proof. For every $n \geq 1$, consider the DPWs $\mathcal{P}'_n, \mathcal{P}''_n, \mathcal{P}'''_n$, and $\mathcal{P}''''_n$ as per Lemma 8. As these four automata are defined over the same structure, the language $L_1 = L(\mathcal{P}'_n) \cap L(\mathcal{P}''_n)$ is recognized by a det. conjunctive-generalized-parity automaton $\mathcal{A}_1$ of size in $O(n)$, having the same structure as that of $\mathcal{P}'_n$ and $\mathcal{P}''_n$ and an acceptance condition that is the conjunction of their conditions. Likewise, the language $L_2 = L(\mathcal{P}'''_n) \cap L(\mathcal{P}''''_n)$ is recognized by an analogous det. conjunctive-generalized-parity automaton $\mathcal{A}_2$ of size in $O(n)$.

Let $\mathcal{B}$ be a det. conjunctive-generalized-parity automaton for the language $L_n = L(\mathcal{A}_1) \cup L(\mathcal{A}_2)$. Observe that the conjunctive-generalized-parity condition is a special case of the Streett condition. Now, by Lemma 6, a DSW that recognizes L_n must have at least $2^{\lfloor \frac{n-1}{2} \rfloor}$ states. Hence, union of det. conjunctive-generalized-parity automata involves an exponential state and size blowup.

By duality, intersection of det. disjunctive-generalized-parity automata also involves an exponential state and size blowup. □

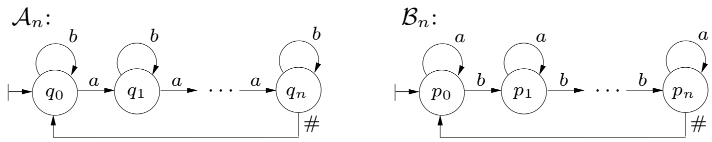
Muller. Union and intersection of det. Muller automata only involves a quadratic *state* blowup, and for automata over the same structure, also only a linear size blowup. Yet, for automata over different structures, both union and intersection involve an exponential *size* blowup.

Proposition 11. *Union and intersection of deterministic Muller automata involves a quadratic state blowup. Union and intersection of (deterministic) Muller automata over the same structure involves no state blowup and linear size blowup.*

Proof. Consider NMWs (or DMWs) $\mathcal{M}'$ and $\mathcal{M}''$ over the same structure. The structure of NMW (resp. DMW) $\mathcal{M}$ that recognizes $L(\mathcal{M}'_n) \cup L(\mathcal{M}''_n)$ (resp. $L(\mathcal{M}'_n) \cap L(\mathcal{M}''_n)$) can be the same as of $\mathcal{M}'$ and $\mathcal{M}''$. In case of union, the set of accepting sets of $\mathcal{M}$ is the union of the accepting sets of $\mathcal{M}'$ and $\mathcal{M}''$, and in the case of intersection, it is their intersection. Observe that the new index is linear in the two original indices.

As a corollary, we have that the *state* blowup in union and intersection of Muller automata over different structures is only quadratic, since we may first enlarge the structure of each of the two automata to be the structure of their product (which might, though, significantly enlarge the corresponding Muller index), and then take their union or intersection. □

Exponential size blowup in intersection is shown in Theorem 2, when the target automaton may even be a nondeterministic Muller automaton. We show below that when the target automaton is a deterministic Muller automaton, union also involves an exponential size blowup.



Both automata have a single Muller acceptance set of all states.

Figure 3: Deterministic Muller automata of size in $O(n)$, whose union is recognized by det. Muller automata of index at least 2^n.

Theorem 12. *Union of deterministic Muller automata involves an exponential size blowup. In particular, for every $n \geq 1$, there are DMWs $\mathcal{A}_n$ and $\mathcal{B}_n$ with $n+1$ states, $2(n+1)$ transitions, and index 1 each, over an alphabet of 3 letters, such that every DMW that recognizes $L(\mathcal{A}_n) \cup L(\mathcal{B}_n)$ has an index of at least 2^n.*

Proof. Consider the DMWs $\mathcal{A}_n$ and $\mathcal{B}_n$ of Figure 3. The language of $\mathcal{A}_n$ is $((b^*a)^n b^* \#)^\omega$ and of $\mathcal{B}_n$ is $((a^*b)^n a^* \#)^\omega$.

Let $\mathcal{M}$ be a DMW for $L(A_n) \cup L(B_n)$. We classify its states according to their relation with the states of $\mathcal{A}_n$ and $\mathcal{B}_n$. Formally, for every $i, j \in [0..n]$, let $S_{i,j}$ be the set of states s of M, for which there exists a finite word u, such that $\mathcal{M}(u) = s$, $\mathcal{A}_n(u) = q_i$, and $\mathcal{B}_n(u) = p_j$.

Observe that a state s of $\mathcal{M}$ cannot belong to both $S_{i,j}$ and $S_{i',j'}$ if $i \neq i'$ or $j \neq j'$, since there is an infinite word y, such that y belongs to $L(A_n^{q_i}) \cup L(B_n^{p_j})$, but not to $L(A_n^{q_{i'}}) \cup L(B_n^{p_{j'}})$.

Now, let U be the set of finite words of length $2n$ over $\{a, b\}$, having exactly n a's and n b's. Notice that for every $u \in U$, the infinite word $w_u = (u\#)^\omega$ belongs to both $L(\mathcal{A}_n)$ and $L(\mathcal{B}_n)$, and in particular also belongs to $L(\mathcal{M})$. Therefore, for every $u \in U$, there is a set S_u of states of $\mathcal{M}$ that is visited infinitely often in the run of $\mathcal{M}$ on w_u, and S_u belongs to the acceptance sets of $\mathcal{M}$.

We claim that for every $u \neq u' \in U$, $S_u \neq S_{u'}$. Indeed, observe first that for every $u \in U$, both $\mathcal{A}_n$ and $\mathcal{B}_n$ return to their initial states in every position of w_u that equals to $0 \bmod (2n{+}1)$, namely after every repetition of $(u\#)$. Hence, for every $t \in [1..2n]$, $u \in U$, and positions p and p' of w_u such that $(p = p' = t) \bmod (2n{+}1)$, $\mathcal{A}_n$ visits the same state, and the same holds for $\mathcal{B}_n$. Since the order of a's and b's is different in u and u', there must be a number $t \in [1..2n]$, such that either $\mathcal{A}_n$ or $\mathcal{B}_n$ visits a different state in positions that equal to $t \bmod (2n{+}1)$ when reading w_u and $w_{u'}$. That is, at these positions, $\mathcal{A}_n$ and $\mathcal{B}_n$ visit the states q_i and p_j, respectively, when reading w_u, and visit states $q_{i'}$ and $p_{j'}$, respectively, when reading $w_{u'}$, with $(i \neq i'$ or $j \neq j')$ and $i + j = i' + j' = t$. Hence, $\mathcal{M}$ visits different sets of states $S_{i,j}$ and $S_{i',j'}$ at positions that equal to $t \bmod (2n{+}1)$. Since $S_{i,j}$ and $S_{i',j'}$ are disjoint, and are also disjoint from every $S_{i'',j''}$ for which $i'' + j'' \neq i + j$, we get that S_u has some state from $S_{i,j}$ that does not belong to $S_{u'}$.

Since there are $\binom{2n}{n}$ different words u in U (recall that $\binom{2n}{n} > 2^n$), there are at least $\binom{2n}{n}$ different Muller sets in the acceptance condition of $\mathcal{M}$, and we are done. □

4 In Favor of The Classic Types

We start with some general observations on the classic types, claiming that they are "well positioned" with respect to the inevitable tradeoff between succinctness, determinization, and complementation. We then briefly specify some good qualities of each type.

[2]This result was independently proved by Christof Löding and Haidi Yue [40].

Operations Size Blowup	On Deterministic Automata			On Nondet. Automata			Nonemptiness Complexity
	Union	Intersect.	Complement.	Union	Intersect.	Complement.	
Weak	Quadratic		No blowup	Linear	Quad.	$2^{\Theta(n)}$ [44] (if possible)	Linear time, NL-complete [19, 52]
Co-Büchi			No blowup [34] (if possible)				
Büchi						$2^{\Theta(n \log n)}$ [43, 48, 13]	
Parity	Exponential Thm. 9[2]		No blowup		Quad. - Quartic		$O(m \log k)$ time, NL-Comp. [22, 29]
Rabin	Quad. Prop. 4	Exp. Thm. 7	Exp. [38]		Quad. Thm. 1, Prop. 4	$2^{\Theta(n^2 \log n)}$ [35, 14, 12]	$O(mk)$ time, NL-Comp. [52]
Streett	Exp. Thm. 7	Quad. Prop. 4					PTIME-comp. [22]
Muller	Exp. Thm. 2,12		Exp. [48]		Exp. Thm. 2	Double-Exp. Thm. 3	NL-comp. [18]

Table 2: The size blowup involved in boolean operations on the classic word automata, and the complexity of checking their nonemptiness. The complexity is w.r.t. the automaton size n, and if specified, w.r.t. m states and index k.

Inevitable: Succinctness + Complementation $\geq$ Double-Exp

There is an inherent tradeoff between the succinctness of an automaton and the size blowup involved in its complementation—It is shown in [49] that there is a family of ω-regular languages $\{L_n\}_{n\geq 1}$, such that for every n, there is an Emerson-Lei automaton of size n for L_n, while every ω-regular automaton for $\overline{L_n}$ has at least 2^{2^n} states. (See Section 5 for details.)

Hence, for an automaton of some type T whose complementation only involves a single-exponential size blowup, there must also be at least a single-exponential size blowup in translating arbitrary ω-regular automata into T-automata. Analogously, if we aim for a single-exponential blowup in determinization, and no blowup in the complementation of deterministic automata, there must be at least a double-exponential size blowup in translating arbitrary automata into deterministic T-automata.

In this sense, the classic types, except for Muller, provide a reasonable tradeoff between their succinctness and the size blowup involved in their determinization and complementation, having all of these measures singly exponential.

Proposition 13. *For every $n \in I\!N$ and nondeterministic ω-regular automaton of size n, there is an equivalent nondeterministic Büchi automaton of size in $2^{O(n)}$ and an equivalent deterministic parity automaton of size in $2^{2^{O(n)}}$.*

Proof. Consider an arbitrary nondet. ω-regular automaton $\mathcal{A}$ of size n. As the acceptance condition of $\mathcal{A}$ is based on the set of states visited infinitely often, there is an equivalent NMW $\mathcal{M}$ over the same structure. Since $\mathcal{A}$ has at most n states, $\mathcal{M}$ has at most 2^n accepting sets. Now, $\mathcal{M}$ can be translated to an equivalent NBW $\mathcal{B}$ with a cubic size blowup [6], namely of size up to $(2^n)^3 \in 2^{O(n)}$.

For an arbitrary nondeterministic automaton $\mathcal{A}$ of size n, there is by the previous paragraph an NBW of size in $2^{O(n)}$, which can be translated to a DPW of size in $2^{2^{O(n)} \log(2^{O(n)})} = 2^{2^{O(n)} O(n)} = 2^{2^{O(n)}}$. □

For the classic types, except for Muller, positive boolean operations on nondet. automata are optimal, as union and intersection are generally done with only linear and quadratic size blowup, respectively (see Table 2), which matches the lower bound for arbitrary automata types. So is the case with the nonemptiness check (see Table 2), which is generally resolved in nondet. logarithmic space.

The classic types have close connection to the Borel hierarchy, to various logics, and to other common formalisms. We list below some distinguishing features of each type.

Weak. The restriction on the structure of the automaton often limits the expressive power, and allows for simple analysis. In particular, the Myhill-Nerode property holds for DWW, while not for the other types, and a language is expressible by a DWW iff it is expressible by both a DBW and a DCW [37, 41, 4].

When considering alternating automata, the weak condition no longer limits the expressive power and AWWs recognize all ω-regular languages [46]. When further restricting the automaton structure to only allow cycles that are self loops ("very weak" automata), we get the exact expressiveness of linear temporal logic (LTL) [39]. Over trees, weak alternating automata are particularly natural [46] and are equivalent to alternation free mu-calculus (AFMC) [47].

Büchi and Co-Büchi. Somewhat surprisingly, the simple Büchi condition, consisting of a single set of states, allows nondet. automata to capture all of the ω-regular languages [11]. Due to its simplicity and natural connection to fairness constraints, it is the most preferred condition in the nondet. setting. Dually, for universal automata, the co-Büchi condition is very useful [25, 9].

Deterministic Büchi automata are less expressive, yet every ω-regular language is equal to a boolean combination of DBWs, namely to a positive boolean combination of DBWs and DCWs.

Generalized-Büchi (GB) automata, in which there are several sets of accepting states, each of which should be visited infinitely often, are useful in the translation of LTL to automata (which involves an exponential size blowup), and can be translated into equivalent Büchi automata with a quadratic size blowup. Deterministic GB automata are as expressive as DBWs.

Parity. The parity condition is naturally related to fixpoint expressions [31, 21], and alternating parity tree automata are expressively equivalent to μ-calculus [21, 28]. This equivalence follows to the hierarchy of parity automata w.r.t. their index and the hierarchy of μ-calculus formulas w.r.t. their alternation depth. Deterministic parity automata are attractive due to their self duality, and the fact that they are ω-regular complete, even though their index is bounded by the number of states.

In the game setting, the parity condition enjoys a special popularity, as both players can do with memoryless strategies [21], and deciding the winner is in $NP \cap coNP$.

Rabin and Streett. The Rabin and Streett conditions are union-closed with respect to rejecting and accepting, respectively, sets of states visited infinitely often. This allows to simplify many aspects of automata and games with the Rabin and Streett conditions. In particular, it allows Rabin (resp. Streett) games to have memoryless strategies for the first (resp. second) player [30], and provides the *typeness* property of deterministic Rabin and Streett automata with deterministic Büchi and co-Büchi automata, respectively [33].

The Streett condition naturally relates to strong fairness, and NSWs are only up to quadratically less succinct than every other classic nondet. automaton (though not than arbitrary nondet. ω-regular automaton). State-wise, on the other hand, every nondet. ω-regular automaton can be translated to an equivalent Rabin automaton with only a quadratic state blowup [7].

Muller. The very descriptive Muller condition is generally impractical, as evident from the its insuccinctness (see Table 1) and the blowup involved in boolean operations on it (see Table 2). Nevertheless, it is very convenient for theoretical purposes, as every nondet. ω-regular automaton admits an equivalent NMW over the same structure, and the Muller condition, though very descriptive, is very simple. (See, e.g., Proposition 13.) Another useful theoretical aspect of the Muller condition is evident by the Wagner hierarchy [53].

5 In Favor of Additional Types

The good qualities of the nondeterministic classic types do not follow to the deterministic setting. In particular, union or intersection of all the classic deterministic ω-regular-complete automata involve an exponential size blowup (see Table 2). This has recently become a practical problem, as det. automata are required in synthesis and in probabilistic model checking, which are rapidly developing. As a result, new, or renewed, acceptance types have emerged, on which positive boolean operations only involve a quadratic size blowup.

We formally define these types and then elaborate on each. Acceptance is defined, as usual, with respect to the set $\mathit{inf}(r)$ of states that the run r visits infinitely often. We also define for a set S of states that $\mathit{Inf}(S)$ holds in a run r if $S \cap \mathit{inf}(r) \neq \emptyset$ and $\mathit{Fin}(S)$ holds otherwise. We describe for each type the form of boolean formula over Fin and Inf that it allows.

- *Emerson-Lei*: An arbitrary boolean formula over Fin and Inf. (A positive boolean formula is enough, as $\neg \mathit{Fin}(S) = \mathit{Inf}(S)$.)
- *Generalized-Rabin*: $\bigvee_{i=1}^{n} \mathit{Fin}(B_i) \wedge \mathit{Inf}(G_{i_1}) \wedge \ldots \wedge \mathit{Inf}(G_{i_{k_i}})$.
- *Generalized-Streett*: $\bigwedge_{i=1}^{n} \mathit{Inf}(G_i) \vee \mathit{Fin}(B_{i_1}) \vee \ldots \vee \mathit{Fin}(B_{i_{k_i}})$.
- *Hyper-Rabin*: $\bigvee_{i=1}^{n} \bigwedge_{j=1}^{m} \mathit{Fin}(B_{i,j}) \vee \mathit{Inf}(G_{i,j})$.
- *Hyper-Streett*: $\bigwedge_{i=1}^{n} \bigvee_{j=1}^{m} \mathit{Fin}(B_{i,j}) \wedge \mathit{Inf}(G_{i,j})$.

Another related type is *circuit* [27], which further shortens Emerson-Lei, by representing the acceptance formula as a boolean circuit.

The index of an automaton is the length of the boolean formula describing its acceptance condition. (For the classic types, it coincides with the standard definition of Section 2.)

The three-letter acronyms of classic automata is extended to four-letter acronyms for the above types. For example, DGRW stands for a deterministic generalized-Rabin word automaton.

Emerson-Lei. The acceptance condition was introduced in 1985 by Emerson and Lei [22], gained some popularity shortly after, was much less popular afterwards, and regained popularity in the past five years. It now appears also as part of the Hanoi Omega-Automata Format (HOAF), which is a new standard for representing automata with some boolean condition over states or transitions [2]

It obviously allows for simple boolean operations. However, complementing as well as determinizing nondet. EL automata involves a doubly-exponential size blowup [49], and their nonemptiness check is NP-complete [22]. Yet, due to the tremendous progress in practically solving the SAT problem, to which the EL condition is naturally related, it may still be interesting to further pursue deterministic EL automata. (See [20, 45])

Hyper-Rabin and generalized-Rabin. The generalized-Rabin condition generalizes both the Rabin and the Muller conditions, and naturally occurs in the translation of various fragments of LTL into automata [32, 15, 23]. The hyper-Rabin condition further generalizes the generalized-Rabin condition, having a disjunction of Streett conditions. It was used under the name "canonical form" in [22]. It allows for union and intersection with only a quadratic size blowup, and the its nonemptiness check is in PTIME.

Proposition 14. *The Muller and Rabin conditions are special cases of the generalized-Rabin condition, which is a special case of the hyper-Rabin condition.*

By Proposition 14, we get lower bounds on translations to classic automata. (See Table 1.)

Proposition 15. *There is an exponential size blowup in the translation of det. generalized-Rabin and hyper-Rabin automata to det. parity, Rabin, Streett, and Muller automata.*

Their positive boolean operations are simple.

Proposition 16. *Union and intersection of (deterministic) generalized-Rabin and hyper-Rabin automata can be done with quadratic size blowup.*

Proof. Consider NGRWs/DGRWs/NHRWs/DHRWs $\mathcal{A}$ and $\mathcal{B}$ with n states, m transitions, and index k.

We construct corresponding NGRWs/DGRWs/NHRWs/DHRWs $\mathcal{C}$ and $\mathcal{D}$ that recognize $L(\mathcal{A}) \cup L(\mathcal{B})$ and $L(\mathcal{A}) \cap L(\mathcal{B})$, respectively, with n^2 states and m^2 transitions, where the index of $\mathcal{C}$ is $2k$ and of $\mathcal{D}$ is k^2.

The structures of $\mathcal{C}$ and $\mathcal{D}$ are achieved by taking the product of the input automata $\mathcal{A}$ and $\mathcal{B}$. For getting $L(\mathcal{C}) = L(\mathcal{A}) \cup L(\mathcal{B})$, the acceptance condition of $\mathcal{C}$ is the union of the conditions of $\mathcal{A}$ and $\mathcal{B}$. For getting $L(\mathcal{D}) = L(\mathcal{A}) \cap L(\mathcal{B})$, for every disjuncts X and Y in $\mathcal{A}$'s and $\mathcal{B}$'s conditions, respectively, we have in $\mathcal{C}$ the disjunct $X \wedge Y$. □

Still, their nonemptiness check is in PTIME.

Proposition 17 ([22, Theorem 4.6])**.** *Nonemptiness check of hyper-Rabin automata is PTIME-complete.*

In the nondet. setting, hyper-Rabin automata are quite similar to standard Streett automata, analogously to the connection between standard Rabin and Büchi automata: The disjunction in the acceptance condition is turned into nondeterminism between copies of the automaton with the weaker condition. Hence, there is only a quadratic size blowup in translating between them.

Proposition 18. *Nondeterministic hyper-Rabin automata can be translated to equivalent nondeterministic Streett automata with a quadratic size blowup.*

As a result, nondet. hyper-Rabin automata can be complemented and determinized with only a single exponential size blowup.

Proposition 19. *Complementation (and determinization) of nondeterministic hyper-Rabin automata can be done with a single-exponential size blowup.*

Proof. Consider an NHRW automaton $\mathcal{A}$ of size n. Then there is an NSW $\mathcal{B}$ of size in $O(n^2)$ equivalent to $\mathcal{A}$. Complementation of $\mathcal{B}$ into an NSW, which is also an NHRW, can be done with a $2^{O(n^2 \log n)}$ size blowup. Hence, the overall size blowup is in $2^{O(n^4 \log n)}$. □

Hyper-Streett and generalized-Streett. The generalized-Streett condition [3] is the dual of generalized-Rabin, and Hyper-Streett is the dual of Hyper-Rabin.[3] These types naturally occur in n-player ω-regular games [24]. Their succinctness and the simplicity of performing boolean operations on them follow from their duality to hyper/generalized-Rabin automata.

Proposition 20. *There is an exponential size blowup in the translation of det. generalized-Streett and hyper-Streett automata to det. parity, Rabin, Streett, and Muller automata.*

Proposition 21. *Union and intersection of (deterministic) generalized-Streett and hyper-Streett automata can be done with quadratic size blowup.*

However, the simplicity of the nonemptiness check does not follow.

Proposition 22. *Nonemptiness check of generalized-Streett and hyper-Streett automata is NP-complete.*

Proof. In [22, Theorem 4.7], it is shown that the nonemptiness problem of EL automata is NP-complete. In the proof, they reduce an instance of the 3-SAT problem to an EL automaton with condition of the form $\wedge_{i=1}^{n}(\mathit{Inf}(S_i) \vee \mathit{Inf}(S'_i))$, which is also definable by generalized-Streett and hyper-Streett conditions of length in $O(n)$. □

Furthermore, while nondet. hyper-Rabin automata can be easily translated to standard Streett automata, nondet. hyper-Streett automata cannot be easily translated to standard Rabin automata, and complementing them involves a doubly-exponential size blowup. It was also recently shown in [24] that their universality problem is ExpSpace-complete.

Theorem 23. *Complementation (and determinization) of nondet. generalized-Streett and hyper-Streett automata of size n into any ω-regular automaton results in an automaton with at least 2^{2^n} states.*

Proof. In [48, Lemma 2.4], it is shown that for every $n > 0$, there exists a language L_n that is recognized by a nondet. EL automaton $\mathcal{A}$ of size n, while any nondet. ω-automaton for the complement of L_n has at least 2^{2^n} states.

The automaton $\mathcal{A}$ uses a condition of the form $\wedge_{i=1}^{n}(\mathit{Inf}((\{q_i\} \leftrightarrow \mathit{Fin}(\{q'_i\})) \wedge (\mathit{Inf}((\{s_i\} \leftrightarrow \mathit{Fin}(\{s'_i\})) \wedge (\mathit{Inf}((\{q_i\} \leftrightarrow \mathit{Inf}((\{s_i\}))$, for states q_i, q'_i, s_i, and s'_i.

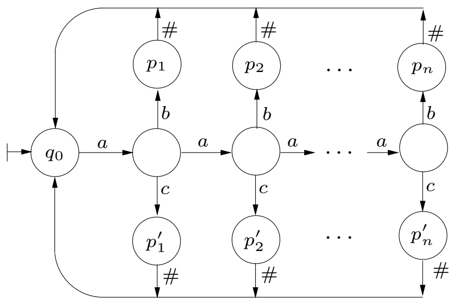
By the structure of $\mathcal{A}$, for every $i \in [1..n]$, at least one of q_i and q'_i, and at least one of s_i and s'_i are visited infinitely often in every accepting run. Hence, over $\mathcal{A}$, the above condition is equal to $\wedge_{i=1}^{n}(\mathit{Fin}(\{q_i\}) \vee \mathit{Fin}(\{q'_i\})) \wedge (\mathit{Fin}(\{s_i\}) \vee \mathit{Fin}(\{s'_i\})) \wedge (\mathit{Fin}(\{q_i\}) \vee \mathit{Inf}((\{s_i\})) \wedge (\mathit{Fin}(\{s_i\}) \vee \mathit{Inf}((\{q_i\}))$, which is definable by generalized-Streett and hyper-Streett conditions of length in $O(n)$. □

Translating between hyper-Rabin and hyper-Streett automata. While the size blowup involved in translating a nondet. hyper-Rabin to an hyper-Streett automaton is quadratic (Proposition 18), in the opposite direction it is exponential, even if starting from a det. generalized-Streett automaton. (A matching single-exp. upper bound follows from Proposition 13.)

Theorem 24. *For every $n \in I\!N$, there is a deterministic generalized-Streett automaton of size n, for which equivalent nondeterministic hyper-Rabin automata are of size at least $2^{n/2}$.*

[3] In [15], they refer to the generalized-Rabin condition as "generalized Rabin pairs" and to the hyper-Streett condition as "generalized Rabin". As we will show, the two conditions have very different properties. In [24], they refer to the hyper-Streett condition as "multi-Rabin".

Deterministic generalized-Streett automaton $\mathcal{D}_n$



The acceptance condition: $\bigwedge_{i=1}^{n}(Fin(\{p_i\}) \vee Fin(\{p'_i\}))$

Figure 4: A deterministic generalized-Streett automaton $\mathcal{D}_n$ of size in $O(n)$, for which equivalent nondeterministic hyper-Rabin automata are of size in $2^{\Omega(n)}$.

Proof. Consider the family $\{\mathcal{D}_n\}_{n\geq 1}$ of DGSWs depicted in Figure 4. Intuitively, we shall describe 2^n different words, such that $\mathcal{D}_n$ accepts each of them, while it rejects each combination of a pair of them. As a result, we will prove that an equivalent NSW will need a unique state for each such word, using the fact that the combination of two Streett-accepting cycles must yield a Streett-accepting cycle. Then, by the result that every NHRW can be translated to an NSW with only a quadratic size blowup, the required lower bound will follow.

For every $i \in [1..n]$, define the finite words $x_i = a^i b\#$ and $x'_i = a^i c\#$. For every $h \subseteq [1..n]$, define the finite word u_h that concatenates for every $i \in h$ the word x_i and for every $j \in [1..n] \setminus h$ the word x'_i. For example, in the case that $n = 4$ and $h = \{2, 3\}$, we have $u_h = ac\#aab\#aaab\#aaaac\#$.

Observe that for every $h \subseteq [1..n]$, the word $w_h = (u_h)^\omega$ is accepted by $\mathcal{D}_n$, since its run on it visits infinitely often exactly one of the states p_i and p'_i, for every $i \in [1..n]$. We shall call the positions of w_h in which $\mathcal{D}_n$ reaches q_0 "big positions". (These are the positions after every full instance of u_h.) Observe also that for every $h \neq h' \subseteq [1..n]$ and every $m, m' \in \mathbb{N} \setminus 0$, the word $(u_h^m u_{h'}^{m'})^\omega$ is rejected by $\mathcal{D}_n$, since its run on it visits infinitely often both p_i and p'_i, for at least one $i \in [1..n]$.

Let $\mathcal{A}$ be an NSW equivalent to $\mathcal{D}_n$, and for every $h \subseteq [1..n]$, let r_h be an accepting run of $\mathcal{A}$ on w_h. We now show that for every $h \neq h' \subseteq [1..n]$, the states that r_h and $r_{h'}$ visit infinitely often in big positions are disjoint, implying that $\mathcal{A}$ has at least 2^n states.

Assume toward contradiction that for some $h \neq h' \subseteq [1..n]$, both r_h and $r_{h'}$ visit the same state s infinitely often in big positions. Let t and t' be big positions of w_h in which r_h visits s, and between which r_h visits exactly the states that it visits infinitely often. Let u be the subword of w_h between positions t and t'. Now, let w be the word that is derived from $w_{h'}$ by adding u in every big position in which $r_{h'}$ visits s.

Consider the run r of $\mathcal{A}$ on w that follows $r_{h'}$, while making extra cycles from s back to itself in every big position that u was added to w. In these extra cycles, r follows the cycle that r_h does between positions t and t'. Notice that the states that r visits infinitely often are the union of the states that r_h and $r_{h'}$ visit infinitely often. Hence, due to the property of the Streett condition that the union of two accepting cycles is accepting, we have that r is accepting. Yet, $\mathcal{D}_n$ rejects w, leading to the required contradicting, and showing that $\mathcal{A}$ has at least 2^n states.

By Proposition 18, every NHRW can be translated to an equivalent NSW with only a quadratic size blowup, implying that an NHRW equivalent to $\mathcal{D}_n$ is of size at least $2^{n/2}$. ☐

<table>
<tr><th colspan="2" rowspan="2">Translations Size Blowup
From / To</th><th colspan="2">Deterministic</th><th colspan="2">Nondeterministic</th></tr>
<tr><th>H-Rabin</th><th>H-Streett</th><th>H-Rabin</th><th>H-Streett</th></tr>
<tr><td rowspan="2">Det.</td><td>Hyper-Rabin</td><td>·</td><td>Exp.
Cor. 25</td><td>·</td><td>$O(n^2)$
Prop. 18</td></tr>
<tr><td>Hyper-Streett</td><td>Exp.
Cor. 25</td><td>·</td><td>Exp.
Thm. 24</td><td>·</td></tr>
<tr><td rowspan="2">Non-Det.</td><td>Hyper-Rabin</td><td colspan="2">Exp.
Prop. 19</td><td>·</td><td>$O(n^2)$
Prop. 18</td></tr>
<tr><td>Hyper-Streett</td><td colspan="2">Double-Exp.
Thm. 23</td><td>Exp.
Thm. 24</td><td>·</td></tr>
</table>

Table 3: The size blowup involved in translations between hyper-Rabin/Streett automata. The translations to and from generalized-Rabin/Streett automata have the same blowup.

<table>
<tr><th rowspan="2">Operations Size Blowup</th><th colspan="3">On Deterministic Automata</th><th colspan="3">On Nondet. Automata</th><th rowspan="2">Nonemptiness Complexity</th></tr>
<tr><th>Union</th><th>Intersect.</th><th>Complement.</th><th>Union</th><th>Intersect.</th><th>Complement.</th></tr>
<tr><td>Hyper-Rabin</td><td colspan="2" rowspan="3">Quadratic
Prop. 16,21</td><td rowspan="2">Exp.
Thm. 24,
Cor. 25</td><td rowspan="3">Linear</td><td rowspan="3">Quad.
Prop. 16,21</td><td>Exp.
Prop. 19</td><td>PTIME-comp.
[22]</td></tr>
<tr><td>Hyper-Streett</td><td rowspan="2">Double-Exp.
Thm. 23</td><td rowspan="2">NP-Complete
Prop. 22</td></tr>
<tr><td>Emerson-Lei</td><td>No blowup</td></tr>
</table>

Table 4: The size blowup involved in boolean operations on the non-classic word automata, and the complexity of checking their nonemptiness. The complexity is w.r.t. the automaton size. Generalized-Rabin/Streett have the same blowup and complexity as hyper-Rabin/Streett.

Complementing deterministic hyper-Rabin and hyper-Streett automata is the same, due to their duality, which is also the same as translating between them. It can be done with a single exponential size blowup, using the quadratic translation of hyper-Rabin automata to NSWs, and the single exponential blowup in determinizing the latter. As a corollary of Theorem 24, this exponential blowup is inevitable.

Corollary 25. *Complementation of deterministic hyper-Rabin and hyper-Streett automata involves a single exponential size blowup.*

6 Conclusions

The paper provides a comprehensive picture of ω-regular automata, analyzing and summarizing the properties of classic and non-classic automata types. Along the way, it completes the data on the size blowup involved in boolean operations, providing quite a few new results.

The need for various automata types on infinite words is clear from the richness of ω-regular languages; Each of the classic types has its own good qualities with respect to simplicity, expressibility, succinctness, complexity of decision problems, and connection to specific logics. Yet, we show that positive boolean operations on deterministic automata of all of the classic ω-regular-complete types involve an exponential size blowup. This is not a must for ω-regularcompleteness, as shown in our analysis of stronger, non-classic, types; Generalized- and hyper-Rabin automata, which may be exponentially more succinct than the classic types, allow for union and intersection with only a quadratic size blowup, and their nonemptiness check is in PTIME. This suggests their usefulness in the setting of deterministic automata, which are essential in the rapidly developing fields of synthesis and probabilistic model-checking.

References

[1] D. Angluin, U. Boker, and D. Fisman. Families of DFAs as acceptors of ω-regular languages. *Logical Methods in Computer Science*, 14(1):1–21, 2018.

[2] T. Babiak, F. Blahoudek, A. Duret-Lutz, J. Klein, J. Křetínský, D. Müller, D. Parker, J. Strejček, and C. S. Păsăreanu. The Hanoi omega-automata format. In *Proc. of CAV*, pages 479–486, 2015.

[3] F. Blahoudek. Translation of an LTL fragment to deterministic Rabin and Streett automata. Master's thesis, Masarykova Univerzita, 2012.

[4] B. Boigelot, S. Jodogne, and P. Wolper. On the use of weak automata for deciding linear arithmetic with integer and real variables. In *Proc. of IJCAR*, pages 611–625, 2001.

[5] U. Boker. Word-automata translations, 2010–2018. URL: http://www.faculty.idc.ac.il/udiboker/automata.

[6] U. Boker. On the (in)succinctness of Muller automata. In *CSL*, pages 12:1–12:16, 2017.

[7] U. Boker. Rabin vs. Streett automata. In *FSTTCS*, pages 17:1–17:15, 2017.

[8] U. Boker, D. Kuperberg, O. Kupferman, and M. Skrzypczak. Nondeterminism in the presence of a diverse or unknown future. In *Proc. of ICALP*, pages 89–100, 2013.

[9] U. Boker and O. Kupferman. Translating to co-Büchi made tight, unified, and useful. *ACM Trans. Comput. Log.*, 13(4):29:1–29:26, 2012.

[10] U. Boker, O. Kupferman, and M. Skrzypczak. How deterministic are good-for-games automata? In *Proceedings of FSTTCS*, pages 18:1–18:14, 2017.

[11] J.R. Büchi. On a decision method in restricted second order arithmetic. In *Proc. Int. Congress on Logic, Method, and Philosophy of Science. 1960*, pages 1–12. Stanford University Press, 1962.

[12] Y. Cai and T. Zhang. A tight lower bound for Streett complementation. In *FSTTCS*, pages 339–350, 2011.

[13] Y. Cai and T. Zhang. Tight upper bounds for Streett and parity complementation. In *CSL*, pages 112–128, 2011.

[14] Y. Cai, T. Zhang, and H. Luo. An improved lower bound for the complementation of Rabin automata. In *LICS*, pages 167–176, 2009.

[15] K. Chatterjee, A. Gaiser, and J. Křetínský. Automata with generalized Rabin pairs for probabilistic model checking and LTL synthesis. In *Proc. of CAV*, pages 559–575, 2013.

[16] K. Chatterjee, T. A. Henzinger, and N. Piterman. Generalized parity games. In *Proc. of FOSSACS*, pages 153–167, 2007.

[17] Y. Choueka. Theories of automata on ω-tapes: A simplified approach. *Journal of Computer and Systems Science*, 8:117–141, 1974.

[18] E. Clarke, I. Browne, and R. Kurshan. A unified approach for showing language containment and equivalence between various types of ω-automata. In *CAAP '90*, pages 103–116, 1990.

[19] E.M. Clarke, E.A. Emerson, and A.P. Sistla. Automatic verification of finite-state concurrent systems using temporal logic specifications. *ACM Transactions on Programming Languagues and Systems*, 8(2):244–263, 1986.

[20] A. Duret-Lutz. *Contributions to LTL and ω-Automata for Model Checking*. Habilitation thesis, Pierre et Marie Curie, 2017.

[21] E.A. Emerson and C. Jutla. Tree automata, μ-calculus and determinacy. In *Proc. 32nd IEEE Symp. on Foundations of Computer Science*, pages 368–377, 1991.

[22] E.A. Emerson and C.-L. Lei. Modalities for model checking: Branching time logic strikes back. *Science of Computer Programming*, 8:275–306, 1987.

[23] J. Esparza, J. Křetínský, and S. Sickert. From LTL to deterministic automata: A Safraless compositional approach. *Formal Methods in System Design*, 49(3):219–271, 12 2016.

[24] E. Filiot, R. Gentilini, and J. F. Raskin. Rational synthesis under imperfect information. In *Proceedings of LICS*, pages 422–431, 2018.

[25] E. Friedgut, O. Kupferman, and M.Y. Vardi. Büchi complementation made tighter. In *Proc. of ATVA*, volume 3299 of *LNCS*, pages 64–78. Springer, 2004.

[26] T.A. Henzinger and N. Piterman. Solving games without determinization. In *Proc. of CSL*, volume 4207 of *LNCS*, pages 394–410. Springer, 2006.

[27] P. Hunter and A. Dawar. Complexity bounds for regular games. In *Proc. of MFCS*, pages 495–506, 2005.

[28] D. Janin and I. Walukiewicz. Automata for the modal μ-calculus and related results. In *Proc. of MFCS*, volume 969 of *LNCS*, pages 552–562. Springer, 1995.

[29] V. King, O. Kupferman, and M.Y. Vardi. On the complexity of parity word automata. In *Proc. of FoSSaCS*, volume 2030 of *LNCS*, pages 276–286. Springer, 2001.

[30] N. Klarlund. Progress measures, immediate determinacy, and a subset construction for tree automata. *Annals of Pure and Applied Logic*, 69(2):243 – 268, 1994.

[31] D. Kozen. Results on the propositional μ-calculus. *Theoretical Computer Science*, 27:333–354, 1983.

[32] J. Kretínský and J. Esparza. Deterministic automata for the (f, g)-fragment of LTL. In *Proc. of CAV*, volume 7358 of *LNCS*, pages 7–22. Springer, 2012.

[33] S.C. Krishnan, A. Puri, and R.K. Brayton. Deterministic ω-automata vis-a-vis deterministic Büchi automata. In *Algorithms and Computations*, volume 834 of *LNCS*, pages 378–386. Springer, 1994.

[34] O. Kupferman, G. Morgenstern, and A. Murano. Typeness for ω-regular automata. In *Proc. of ATVA*, volume 3299 of *LNCS*, pages 324–338. Springer, 2004.

[35] O. Kupferman and M.Y. Vardi. Complementation constructions for nondeterministic automata on infinite words. In *Proc. of TACAS*, volume 3440 of *LNCS*, pages 206–221. Springer, 2005.

[36] R.P. Kurshan. *Computer Aided Verification of Coordinating Processes*. Princeton Univ. Press, 1994.

[37] L.H. Landweber. Decision problems for ω–automata. *Mathematical Systems Theory*, 3:376–384, 1969.

[38] C. Löding. Optimal bounds for the transformation of omega-automata. In *Proceedings of FSTTCS*, volume 1738 of *LNCS*, pages 97–109, 1999.

[39] C. Löding and W. Thomas. Alternating automata and logics over infinite words. In *Theoretical Computer Science*, volume 1872 of *LNCS*, pages 521–535. Springer, 2000.

[40] C. Löding and H. Yue. Memory bounds for winning strategies in infinite games, 2008, unpublished.

[41] O. Maler and L. Staiger. On syntactic congruences for ω-languages. *Theoretical Computer Science*, 183(1):93–112, 1997.

[42] Z. Manna and A. Pnueli. Specification and verification of concurrent programs by $\forall$-automata. In *Proc. of PoPL*, pages 1–12, 1987.

[43] M. Michel. Complementation is more difficult with automata on infinite words. CNET, Paris, 1988.

[44] S. Miyano and T. Hayashi. Alternating finite automata on ω-words. *Theoretical Computer Science*, 32:321–330, 1984.

[45] D. Müller and S. Sickert. LTL to deterministic emerson-lei automata. In *Proceedings of GandALF*

2017, pages 180–194, 2017.

[46] D.E. Muller, A. Saoudi, and P.E. Schupp. Alternating automata, the weak monadic theory of the tree and its complexity. In *Proc. 13th Int. Colloq. on Automata, Languages, and Programming*, volume 226 of *LNCS*, pages 275 – 283. Springer, 1986.

[47] D.E. Muller and P.E. Schupp. Simulating alternating tree automata by nondeterministic automata: New results and new proofs of theorems of Rabin, McNaughton and Safra. *Theoretical Computer Science*, 141:69–107, 1995.

[48] S. Safra. *Complexity of automata on infinite objects*. PhD thesis, Weizmann Institute of Science, 1989.

[49] S. Safra and M.Y. Vardi. On ω-automata and temporal logic. In *Proc. 21st ACM Symp. on Theory of Computing*, pages 127–137, 1989.

[50] M. Vardi. The Büchi complementation saga. In *STACS 2007*, pages 12–22. Springer Berlin Heidelberg, 2007.

[51] M.Y. Vardi. Automatic verification of probabilistic concurrent finite-state programs. In *Proc. 26th IEEE Symp. on Foundations of Computer Science*, pages 327–338, 1985.

[52] M.Y. Vardi and P. Wolper. Reasoning about infinite computations. *Information and Computation*, 115(1):1–37, 1994.

[53] K. W. Wagner. A hierarchy of regular sequence sets. In Jirí Becvár, editor, *MFCS*, volume 32 of *LNCS*, pages 445–449. Springer, 1975.