Good for Games Automata: From Nondeterminism to Alternation

Udi Boker
Interdisciplinary Center (IDC) Herzliya, Israel
udiboker@gmail.com

Karoliina Lehtinen
University of Liverpool, United Kingdom
k.lehtinen@liverpool.ac.uk

Abstract

A word automaton recognizing a language L is good for games (GFG) if its composition with any game whose winning condition is L preserves the game's winner. Deterministic automata are GFG, while nondeterministic automata are generally not. There are various other properties that are used in the literature for defining that a nondeterministic automaton is GFG, including "history deterministic", "compliant with some letter game", "good for trees", and "good for composition with other automata". Yet, it is not formally shown that all of these properties are equivalent.

We clarify the different definitions of GFG automata and prove that they are all indeed equivalent. In the setting of alternating automata, so far only some of the above definitions have been considered. We generalize the other definitions and prove that they all remain equivalent.

We further look into alternating GFG automata, showing that they are as expressive as deterministic automata with the same acceptance conditions and indices. Considering their succinctness, we show that alternating GFG automata over finite words, as well as weak automata over infinite words, are not more succinct than deterministic automata, and that determinizing Büchi and co-Büchi alternating GFG automata involves a $2^{\Theta(n)}$ state blow-up. We leave open the question of whether alternating GFG automata of stronger acceptance conditions allow for doubly-exponential succinctness compared to deterministic automata.

2012 ACM Subject Classification Theory of computation → Logic and verification

Keywords and phrases Good for games, history determinism, alternation

Funding *Udi Boker*: Israel Science Foundation grant 1373/16
Karoliina Lehtinen: EPSRC grant EP/P020909/1 (Solving Parity Games in Theory and Practice)

Acknowledgements We thank Orna Kupferman for suggesting to look into alternating good-for-games automata and for stimulating discussions on the subject, and Thomas Colcombet for introducing to us his work on alternating history-deterministic automata.

1 Introduction

Deterministic automata are more usable than nondeterministic automata in contexts such as synthesis because of their compositional properties. Unfortunately, determinization is complicated and involves an exponential increase in the state space. Nondeterministic automata that are *good for games* (GFG) have been heralded as a potential way to combine the compositionality of deterministic automata with the conciseness of nondeterministic ones. In this article we are interested in the question of whether the benefits of good-for-games automata extend to alternating automata.

The first hurdle of studying good-for-games alternating automata is to settle on definitions. Indeed, for nondeterminism this notion seems to be particularly natural, as it has been invented several times independently under different names: good for games [?], good for trees [?], and history determinism [?].

While Henzinger and Piterman introduced the idea of automata that compose well with games, in their technical development they preferred to use a *letter game* such that one player having a winning strategy in this game implies that the nondeterministic automaton composes well with games [?]. In a similar vein, Kupferman, Safra and Vardi considered already in 1996 a form of nondeterministic automata that resolves its nondeterminism according to the past by looking at tree automata for derived word languages [?]; this notion of *good for trees* was shown to be equivalent to the letter game [?]. Independently, Colcombet introduced history-determinism in the setting of nondeterministic cost automata [?], and later extended it to alternating automata [?]. He showed that history-determinism implies that the automaton is suitable for composition with other alternating automata, a seemingly stronger property than just compositionality with games. Although Colcombet further developed history-determinism for cost automata, here we only consider automata with ω-regular acceptance conditions.

As a result, in the literature there are at least five different definitions that characterize, imply, or are implied by a nondeterministic automaton composing well with games: composition with games, composition with automata, composition with trees, letter games and history determinism. While some implications between them are proved, others are folklore, or missing. Furthermore, these definitions do not all generalize in the same way to alternating automata: compositionality with games and with automata are agnostic to whether the automaton is nondeterministic, universal or alternating, and hence generalize effortlessly to alternating automata; the letter-game and good-for-tree automata on the other hand generalize 'naturally' in a way that treats nondeterminism and universality asymmetrically and hence need be adapted to handle alternation.

In the first part of this article, we give a coherent account of good-for-gameness for alternating automata: we generalize all the existing definitions from nondeterministic to alternating automata, and show them be equivalent. This implies that these are also equivalent for nondeterministic automata. While some of these equivalences were already folklore, at least for nondeterministic automata, others are more surprising: compositionality with one-player games implies compositionality with two-player games and compositionality with automata, despite games being a special case of alternating automata and single-player games being a special case of games. We also show that in the nondeterministic case each definition can be relaxed to an asymmetric requirement: composition with universal automata and composition with universal trees are already equivalent to composition with alternating automata and games.

In the second part of this article, we focus on questions of expressiveness and succinctness. The first examples of GFG automata were built on top of deterministic automata [?], and Colcombet conjectured that history-deterministic alternating automata with ω-regular acceptance conditions are not more concise than deterministic ones [?]. Yet, this has since been shown to be false: already GFG nondeterministic Büchi automata cannot be pruned into deterministic ones [?] and co-Büchi automata can be exponentially more concise than deterministic ones [?]. In general, nondeterministic GFG automata are in between nondeterministic and deterministic automata, having some properties from each [?].

Alternating automata can be doubly exponentially more concise than deterministic automata; whether this is the case for GFG alternating automata is particularly interesting in the wake of quasi-polynomial algorithms for parity games. Indeed, since 2017 when Calude et al. brought down the upper bound for solving parity games from subexponential to quasi-polynomial [?], the automata-theoretical aspects of solving parity games with quasi-polynomial complexity have been studied in more depth [?, ?, ?, ?, ?, ?, ?].

Bojańczyk and Czerwiński [?], and Czerwiński, Daviaud, Fijalkow, Jurdziński, Lazić, and Parys [?] describe the quasi-polynomial algorithms for solving parity games explicitly in terms of deterministic word automata that separate some word languages. A polynomial deterministic *or GFG* safety separating automaton for these languages would imply a polynomial algorithm for parity games. However, it is shown in [?] that the smallest possible such nondeterministic automaton is quasi-polynomial. Since this lower bound only applies for nondeterministic automata, it is interesting to understand whether alternating GFG automata could be more concise.

Expressiveness wise, we show that alternating GFG automata are as expressive as deterministic automata with the same acceptance conditions and indices. The proof extends the technique used in the nondeterministic setting, producing a deterministic automaton from the product of the automaton and the two transducers that model its history determinism.

Regarding succinctness, we first show that GFG automata over finite words, as well as weak automata over infinite words, are not more succinct than deterministic automata. The proof builds on the property that minimal deterministic automata of these types have exactly one state for each Myhill-Nerode equivalence class, and an analysis that GFG automata of these types must also have at least one state for each such class.

We proceed to show that determinizing Büchi and co-Büchi alternating GFG automata involves a $2^{\theta(n)}$ state blow-up. The proof in this case is more involved, going through two main lemmas. The first shows that for alternating GFG Büchi automata, a history-deterministic strategy need not remember the entire history of the transition conditions, and can do with only remembering the prefix of the word read. The second lemma shows that the breakpoint (Miyano-Hayashi) construction, which is used to translate an alternating Büchi automaton into a nondeterministic one, preserves GFGness. We leave open the question of whether alternating GFG automata of stronger acceptance conditions allow for doubly-exponential succinctness compared to deterministic automata.

Due to lack of space, some of the proofs appear in the Appendix.

2 Preliminaries

Words and automata. An *alphabet* Σ is a finite nonempty set of letters, a finite (resp. infinite) *word* $u = u_0 \dots u_k \in \Sigma^*$ (resp. $w = w_0 w_1 \dots \in \Sigma^\omega$) is a finite (resp. infinite) sequence of letters from Σ. A *language* is a set of words, and the empty word is written ϵ.

An *alternating word automaton* is $\mathcal{A} = (\Sigma, Q, \iota, \delta, \alpha)$, where Σ is a finite nonempty alphabet, Q is a finite nonempty set of states, $\iota \in Q$ is an initial state, $\delta : Q \times \Sigma \to \mathsf{B}^+(Q)$ is a transition function and α is an acceptance condition, on which we elaborate below. A *transition condition* is a formula $b \in \mathsf{B}^+(Q)$ in the image of δ. For a state $q \in Q$, we denote by $\mathcal{A}^q$ the automaton that is derived from $\mathcal{A}$ by setting its initial state to q. $\mathcal{A}$ is nondeterministic (resp. universal) if all its transition conditions are disjunctions (resp. conjuctions), and it is deterministic if all its transition conditions are states.

There are various acceptance conditions, defined with respect to the set of states that (a path of) a run of $\mathcal{A}$ visits. Some of the acceptance conditions are defined on top of a labeling of $\mathcal{A}$'s states. In particular, the parity condition is a labeling $\alpha : Q \to \Gamma$, where Γ is a finite set of priorities and a path is accepting if and only if the highest priority seen infinitely often on it is even. A Büchi condition is the special case of the parity condition where $\Gamma = \{1, 2\}$; states of priority 2 are called *accepting* and of priority 1 *rejecting*, and then α can be viewed as the subset of accepting states of Q. Co-Büchi automata are dual, with $\Gamma = \{0, 1\}$. A weak automaton is a Büchi automaton in which every strongly connected component consists of only accepting or only rejecting states.

In Sections 2-5, we handle automata with arbitrary acceptance conditions, and thus consider α to be a mapping from Q to a finite set Γ, on top of which some further acceptance criterion is implicitly considered (as in the parity condition). In Section 6, we focus on weak, Büchi, and co-Büchi automata, and then view α as a subset of Q.

Games. A *finite* Σ*-arena* is a finite $\Sigma \times \{A, E\}$-labeled Kripke structure. An *infinite* Σ*-arena* is an infinite $\Sigma \times \{A, E\}$-labeled tree. Nodes with an A-label are said to belong to Adam; those with an E-label are said to belong to Eve. We represent a Σ-arena as $R = (V, X, V_E, L)$,

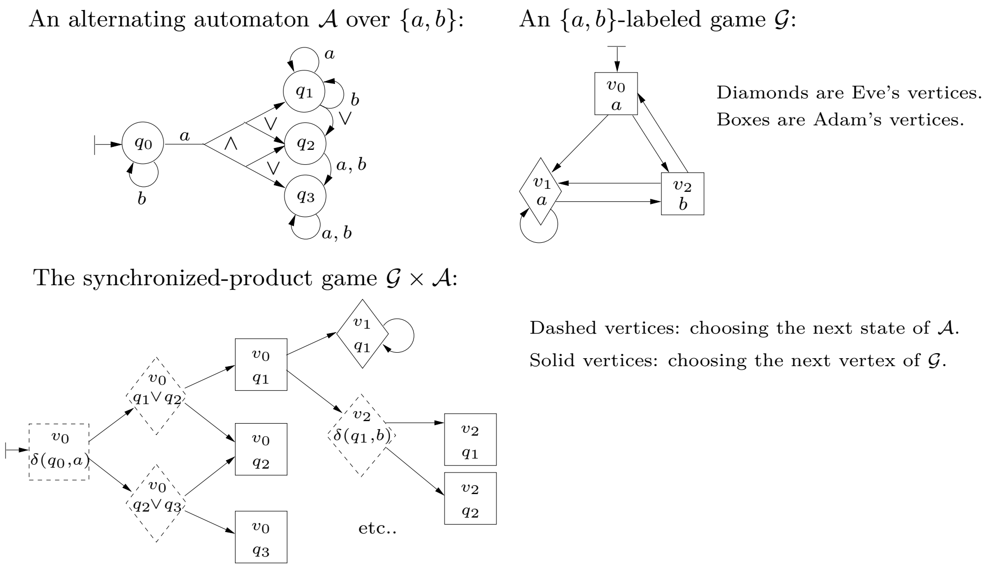

Figure 1 An example of a product between an alternating automaton and a finite-arena game.

where V is its set of nodes, X its transitions, V_E the E-labeled nodes, $V \setminus V_E$ the A-labeled nodes and $L : V \to \Sigma$ its Σ-labeling function. We will assume that all states have a successor.

A play in a R is an infinite path in R. A *game* is a Σ-arena together with a winning condition $W \subseteq \Sigma^\omega$. A play π is said to be winning for Eve in the game if the Σ-labels along π form a word in W. Else π is winning for Adam.

A *strategy* for Eve (Adam, resp.) is a function $\tau : V^* \to V$ that maps a history $v_0 \ldots v_i$, i.e. a finite prefix of a play in R, to a successor of v_i whenever $v_i \in V_E$ ($v_i \notin V_E$). A play $v_0, v_1, \ldots$ agrees with a strategy τ for Eve (Adam) if whenever $v_i \in V_E$ ($v_i \notin V_E$), we have $v_{i+1} = \tau(v_i)$. A strategy for Eve (Adam) is winning if all plays that agree with it are winning for Eve (Adam). We say that a player wins the game if they have a winning strategy.

All the games we consider have ω-regular winning conditions and are therefore determined and the winner has a finite-memory strategy [**?**]. Finite-memory strategies can be modeled by *transducers*. Given alphabets I and O, an (I/O)-*transducer* is a tuple $\mathcal{M} = (I, O, M, \iota, \rho, \chi)$, where M is a finite set of states (memories), $\iota \in M$ is an initial memory, $\rho : M \times I \to M$ is a deterministic transition function, and $\chi : M \to O$ is an output function. The strategy $\mathcal{M} : I^* \to O$ is obtained by following ρ and χ in the expected way: we first extend ρ to words in I^* by setting $\rho(\epsilon) = \iota$ and $\rho(u \cdot a) = \rho(\rho(u), a)$, and then define $\mathcal{M}(u) = \chi(\rho(u))$.

Products.

▶ **Definition 1** (Synchronized product)**.** *The* synchronized product $R \times \mathcal{A}$ *between a Σ-arena $R = (V, X, V_E, L)$ and an alternating automaton $\mathcal{A} = (Q, \Sigma, \iota, \delta, \alpha)$ with mapping $\alpha : Q \to \Gamma$ is a $\Gamma \cup \{\bot\}$-arena of which the states are $V \times \mathsf{B}^+(Q)$ and the successor relation is defined by:*

- *(v, q), for a state q of Q, has successors $(v', \delta(q, L(v')))$ for each successor v' of v in R.*
- *$(v, b \wedge b')$ and $(v, b \vee b')$ have two successors (v, b) and (v, b');*
- *If R is rooted at v then the root of $R \times \mathcal{A}$ is $(v, \delta(\iota, L(v)))$.*

The positions belonging to Eve are (v, b) where either b is a disjunction, or b is a state in Q and $v \in V_E$. The label of (v, b) is $\alpha(b)$ if b is a state of Q, and $\bot$ otherwise.

An example, without labeling, of a synchronized product is given in Figure 1.

▶ **Definition 2** (Automata composition). *Given alternating automata $\mathcal{B} = (\Sigma, Q^{\mathcal{B}}, \iota^{\mathcal{B}}, \delta^{\mathcal{B}}, \beta : Q^{\mathcal{B}} \to \Gamma)$ and $\mathcal{A} = (\Gamma, Q^{\mathcal{A}}, \iota^{\mathcal{A}}, \delta^{\mathcal{A}}, \alpha)$, their* composition $\mathcal{B} \times \mathcal{A}$ *consists of the synchronized product automaton $(\Sigma, Q^{\mathcal{B}} \times Q^{\mathcal{A}}, (\iota^{\mathcal{B}}, \iota^{\mathcal{A}}), \delta, \alpha')$, where $\alpha'(q^B, q^A) = \alpha(q^A)$ and $\delta((q^{\mathcal{B}}, q^{\mathcal{A}}), a)$ consists of $f(\delta^{\mathcal{B}}(q^B, a), q^{\mathcal{A}})$ where:*

- $f(c \vee c', q) = f(c, q) \vee f(c', q)$
- $f(c \wedge c', q) = f(c, q) \wedge f(c', q)$
- $f(q', q) = g(q', \delta^{\mathcal{A}}(q, \beta(q'))$ *where*
- $g(q, c \vee c') = g(q, c) \vee g(q, c')$
- $g(q, c \wedge c') = g(q, c) \wedge g(q, c')$
- $g(q, q') = (q, q')$.

*Note that this stands for first unfolding the transition condition in $\mathcal{B}$ and then the transition condition in $\mathcal{A}$, and it is equivalent to the following substitution, which matches Colcombet's notation [**?**]: $\delta^{\mathcal{B}}(q^{\mathcal{B}}, a)[q \in Q^{\mathcal{B}} \leftarrow \delta^{\mathcal{A}}(q^{\mathcal{A}}, \beta(q))[p \in Q^{\mathcal{A}} \leftarrow (q, p)]]$*

Acceptance of a word by an automaton. We define the acceptance directly in terms of the model-checking (membership) game, which happens to be exactly the product of the automaton with a path-like arena describing the input word. More precisely, $\mathcal{A}$ accepts a word w if and only if Eve wins the *model-checking game* $\mathcal{G}(w, \mathcal{A})$, defined as the product $R_w \times \mathcal{A}$, where the arena R_w consists of an infinite path, of which all positions belong to Eve (although it does not matter), and the label of the i^{th} position is the i^{th} letter of w. We will refer to the positions of R_w by the suffix of w that labels the path starting there. We denote by $\mathcal{G}_\tau(w, \mathcal{A})$ the model-checking game that agrees with a strategy τ of Adam or Eve. The language of an automaton $\mathcal{A}$, denoted by $L(\mathcal{A})$, is the set of words that it accepts (recognizes). Two automata are equivalent if they recognize the same language.

3 Good for Games Automata: Five Definitions

We clarify the five definitions that are used in the literature for stating that an automaton is good for games, while generalizing them form a nondeterministic to an alternating word automaton $\mathcal{A} = (\Sigma, Q, \iota, \delta, \alpha)$.

Good for game composition. The first definition matches the intuition that "$\mathcal{A}$ is good for playing games". It was given in [**?**] for nondeterministic automata and applies as is to alternating automata, by properly defining the synchronized product of a game and an alternating automaton. (See Definition 1 and Figure 1.) We shall prove in Section 4 that Definition 3 is equivalent when speaking of only one-player finite-arena games and two-player finite/infinite-arena games.

▶ **Definition 3** (GFG1: Good for game composition). *$\mathcal{A}$ is* good for game composition *if for every [one-player] game G with a [finite] Σ-labeled arena and a winning condition $L(\mathcal{A})$, Eve has a winning strategy in G if and only if she has a winning strategy in the synchronized-product game $G \times \mathcal{A}$.*

Compliant with the letter games. The first definition is simple to declare, but is not convenient for technical developments. Thus, Henzinger and Piterman defined the "letter-game", our next definition, while independently Colcombet defined history-determinism, which we provide afterwards. The two latter definitions are easily seen to be equivalent and they were shown in [**?**] to imply the game-composition definition. We are not aware of a full proof of the other direction in the literature; we include one in this article.

In the letter-game for nondeterministic automata [**?**], Adam generates a word letter by letter, and Eve resolves the nondeterminism "on the fly", such that the generated run of $\mathcal{A}$ accepts every word in the language. It has not been generalized yet to the alternating setting, and there are various ways in which it can be generalized, as it is not clear who should pick the letters and how to resolve the nondeterminism and universality. It turns out that a generalization that works well is to consider two independent games, one in which Eve resolves the nondeterminism while Adam picks the letters and resolves the universality, and another in which Eve picks the letters.

▶ **Definition 4** (GFG2: Compliant with the letter games). *There are two letter games, Eve's game and Adam's game.*

Eve's game: A configuration is a transition condition $b \in \mathsf{B}^+(Q)$ and a letter $\sigma \in \Sigma^* \cup \epsilon$. (We abuse ϵ to also be an empty letter.) A play begins in $(b_0, \sigma_0) = (\iota, \epsilon)$ and consists of an infinite sequence of configurations $(b_0, \sigma_0)(b_1, \sigma_1)\dots$. In a configuration (b_i, σ_i), the game proceeds to the next configuration (b_{i+1}, σ_{i+1}) as follows.

- *If b_i is a state of Q, Adam picks a letter a from Σ, having $(b_{i+1}, \sigma_{i+1}) = (\delta(b_i, a), a)$.*
- *If b_i is a conjunction $b_i = b' \wedge b''$, Adam chooses between (b', ϵ) and (b', ϵ).*
- *If b_i is a disjunction $b_i = b' \vee b''$, Eve chooses between (b', ϵ) and (b', ϵ).*

In the limit, a play consists of an infinite sequence $\pi = b_0, b_1, \dots$ of transition conditions and an infinite word $w = \sigma_0, \sigma_1, \dots$. Let ρ be the restriction of π to transition conditions that are states of Q. Eve wins the play if either $w \notin L(\mathcal{A})$ or ρ satisfies $\mathcal{A}$'s acceptance condition.

The nondeterminism in $\mathcal{A}$ is compliant with the letter games *if Eve wins this game.*

Adam's game: It is similar to Eve's game, except that Eve chooses the letters instead of Adam, and Adam wins if either $w \in L(\mathcal{A})$ or ρ does not satisfy $\mathcal{A}$'s acceptance condition. The universality in $\mathcal{A}$ is compliant with the letter games *if Adam wins this game.*

$\mathcal{A}$ is compliant with the letter games *if its nondeterminism and universality are compliant with the letter games.*

History determinism. A nondeterministic automaton is history deterministic [?] if there is a strategy to resolve the nondeterminism that only depends on the word read so far, i.e., that is uniform for all possible futures. Colcombet generalized the definition to alternating automata [?], considering a strategy to be a function from a finite sequence of transition conditions and letters to a transition condition, and considering its adequacy in the model-checking game of $\mathcal{A}$ and a word w.

We first define how to use a strategy $\tau : (\Sigma \times \mathsf{B}^+(Q))^* \to \mathsf{B}^+(Q)$ for playing in a model-checking game $\mathcal{G}(w, \mathcal{A})$, as the history domains are different. Recall that in the model-checking game $\mathcal{G}(w, \mathcal{A})$, positions consist of a transition of $\mathcal{A}$ and a suffix of w, so histories have type $(\Sigma^\omega \times \mathsf{B}^+(Q))^*$. From such a history h, let h' be the history obtained by only keeping the first letter of the Σ^ω component of h's elements, that is, the letter at the head of the current suffix. Then, we extend τ to operate over the $(\Sigma^\omega \times \mathsf{B}^+(Q))^*$ domain, by defining $\tau(h) = \tau(h')$.

For conveniency, we often refer to a history in $(\Sigma \times \mathsf{B}^+(Q))^*$, as a pair in $\Sigma^* \times \mathsf{B}^+(Q)^*$.

▶ **Definition 5** (GFG3: History determinism [?]).

- *The* nondeterminism in $\mathcal{A}$ is history-deterministic *if there is a strategy $\tau_E : (\Sigma \times \mathsf{B}^+(Q))^* \to \mathsf{B}^+(Q)$ such that for all $w \in L(\mathcal{A})$, τ_E is a winning strategy for Eve in $\mathcal{G}(w, \mathcal{A})$.*
- *The* universality in $\mathcal{A}$ is history-deterministic *if there is a strategy $\tau_A : (\Sigma \times \mathsf{B}^+(Q))^* \to \mathsf{B}^+(Q)$ such that for all $w \notin L(\mathcal{A})$, τ_A is a winning strategy for Adam in $\mathcal{G}(w, \mathcal{A})$.*
- *$\mathcal{A}$ is* history-deterministic *if its nondeterminism and universality are history deterministic.*

Good for automata composition. The next definition comes from Colcombet's proof that alternating history-deterministic automata behave well with respect to composition with other alternating automata. We shall show in Section 4 that it also implies proper compositionality with tree automata, and that for nondeterministic automata, it is enough to require compositionality with universal, rather than alternating, automata.

▶ **Definition 6** (GFG4: Good for automata composition [?]). *$\mathcal{A}$ is* good for automata composition *if for every alternating word (or tree) automaton $\mathcal{B}$ with Σ-labeled states and acceptance condition $L(\mathcal{A})$, the composed automaton $\mathcal{B} \times \mathcal{A}$ is equal to $\mathcal{B}$.*

Good for trees. The next definition comes from the work in [?, ?] on the power of nondeterminism in tree automata. It states that a nondeterministic word automaton $\mathcal{A}$ is good-for-trees if we can "universally expand" it to run on trees and accept the "universally derived language" $L(\mathcal{A})\triangle$—trees all of whose branches are in the word language of $\mathcal{A}$.

Observe that every universal word automaton for a language L is trivially good for $L\triangle$. Therefore, for universal automata, we suggest that a dual definition is more interesting: its "existential expansion to trees" accepts $L\triangledown$—trees in which there exists a path in L.

For an alternating automaton $\mathcal{A}$, we generalize the good-for-trees notion to require that $\mathcal{A}$ is good for both $L(\mathcal{A})\triangle$ and $L(\mathcal{A})\triangledown$, when expanded universally and existentially, respectively.

We first formally generalize the definition of "expansion to trees" to alternating automata. It follows the standard definition of a run of an alternating automaton on a word, while rather than considering in each step the next letter, it considers the label of a single child or all children of the current tree node: The *universal (resp. existential) expansion* of $\mathcal{A}$ to trees accepts a tree t iff Eve wins the game $t \times \mathcal{A}$, when t is viewed as a game in which all nodes belong to Adam (resp. Eve).

▶ **Definition 7** (GFG5: Good for trees). *$\mathcal{A}$ is* good for trees *if its universal- and existential-expansions to trees recognize the tree languages $L(A)\triangle$ and $L(A)\triangledown$, respectively.*

4 Equivalence of All Definitions

We prove in this section the equivalence of all of the definitions in the alternating setting, as given in Section 3, which implies their equivalence also in the nondeterministic (and universal) setting. We may therefore refer to an automaton as *good-for-games (GFG)* if it satisfies any of these definitions. In some cases, we provide additional equivalences that only apply to the nondeterministic setting.

▶ **Theorem 8.** *An alternating automaton either satisfies all of Definitions 3-7 or none of them.*

Proof.

- Lemma 9: History-determinism = compliance with the letter games. (Def. 4 = Def. 5).
- Lemma 13: Compliance with the letter games ⇒ compositionality with arbitrary games. (Def. 4 ⇒ "strong" Def. 3).
- Lemma 14: Compositionality, even with just one-player finite-arena games ⇒ compliance with the letter games. ("weak" Def. 3 ⇒ Def. 4).
- Lemma 15: Good for trees is = compositionality with one-player games. (Def. 7 = "medium" Def. 3).
- Lemma 16: Compositionality with games = compositionality with automata. (Def. 6 = Def. 3).

◀

We start with the simple equivalence of history determinism and compliance with the letter game. (Observe that the letter-game strategies are of the same type as the strategies that witness history determinism: a function from $(\mathsf{B}^+(Q) \times \Sigma)^*$ to $\mathsf{B}^+(Q)$.)

▶ **Lemma 9.** *Consider an alternating automaton $\mathcal{A} = (\Sigma, Q, \iota, \delta, \alpha)$.*

- *A strategy τ_E for Eve in her letter game is winning if and only if it witnesses the history-determinism of the nondeterminism in $\mathcal{A}$.*

- *A strategy τ_A for Adam in his letter game is winning if and only if it witnesses the history-determinism of the universality in $\mathcal{A}$.*
- *An alternating automaton $\mathcal{A}$ is history-deterministic if and only if it is compliant with the letter games.*

▶ **Corollary 10.** *If $\mathcal{A}$ is history-deterministic, then there are finite-memory strategies τ_E and τ_A to witness it.*

The following two propositions state that "standard manipulations" of alternating automata preserve history determinism. The *dual* of an automaton $\mathcal{A}$, denoted by $\overline{\mathcal{A}}$, is derived from $\mathcal{A}$ by changing every conjunction of a transition condition to a disjunction, and vice versa, and changing the acceptance condition to reject every sequence of states (labelings) that $\mathcal{A}$ accepts, and accept every sequence that $\mathcal{A}$ rejects.

▶ **Proposition 11.** *Consider an alternating automaton $\mathcal{A}$ and its dual $\overline{\mathcal{A}}$. The nondeterminism (resp. universality) of $\mathcal{A}$ is history deterministic iff the universality (resp. nondeterminism) of $\overline{\mathcal{A}}$ is history deterministic.*

▶ **Proposition 12.** *Consider an alternating automaton $\mathcal{A}$, and let $\mathcal{A}'$ be an automaton that is derived from $\mathcal{A}$ by changing some transition conditions to different, but equivalent, boolean formulas. Then the nondeterminism/universality in $\mathcal{A} = (\Sigma, Q, \iota, \delta, \alpha)$ is history-deterministic iff it is history-deterministic in $\mathcal{A}'$.*

The following lemma was shown in [**?**] for nondeterministic automata and can be deduced for alternating automata from Lemma 9 and Colcombet's result [**?**] on the equivalence of history-determinism and being good for composition with automata. We provide here a direct simple proof.

▶ **Lemma 13.** *If an alternating automaton $\mathcal{A}$ is compliant with the letter games then it is good for game-composition.*

If $\mathcal{A}$ is good for infinite games, it is clearly good for finite games, which can be unfolded into infinite games. The following lemma shows that the other direction holds too: compositionality with finite games implies compliance with the letter games, and therefore, from the previous lemma, composition with infinite games.

Note that this correspondence does not extend to the notion of *good for small games* [**?**, **?**]: an automaton can be good for composition with games up to a bounded size, without being good for games.

▶ **Lemma 14.** *If an alternating automaton is good for composition with finite-arena one-player games then it is compliant with the letter games.*

Proof. Consider an alternating automaton $\mathcal{A}$ over the alphabet Σ. We show that if $\mathcal{A}$ is not compliant with the letter games then it is not good for finite-arena one-player games. By Proposition 12, we may assume that the transition conditions in $\mathcal{A}$ are in CNF.

If $\mathcal{A}$ is not compliant with Eve's letter game, then since this game is ω-regular, Adam has some finite-memory winning strategy, modeled by a transducer $\mathcal{M}$. Observe that states of $\mathcal{M}$ output the next move of Adam, namely a letter and a disjunctive clause, and the transitions of $\mathcal{M}$ correspond to moves of Eve.

We translate $\mathcal{M}$ into a one-player Σ-labeled game with winning condition $L(\mathcal{A})$, in which all states belong to Adam: we consider every state/transition of $\mathcal{M}$ as a vertex/edge of $\mathcal{G}$, take the letter output of a state as the labeling of the corresponding vertex, and ignore the other outputs and the transition labelings. (See an example in Figure 2.) We claim that $\mathcal{A}$ is not good for one-player finite-arena games, since i) Adam loses $\mathcal{G}$; and ii) Adam wins $\mathcal{G} \times \mathcal{A}$.

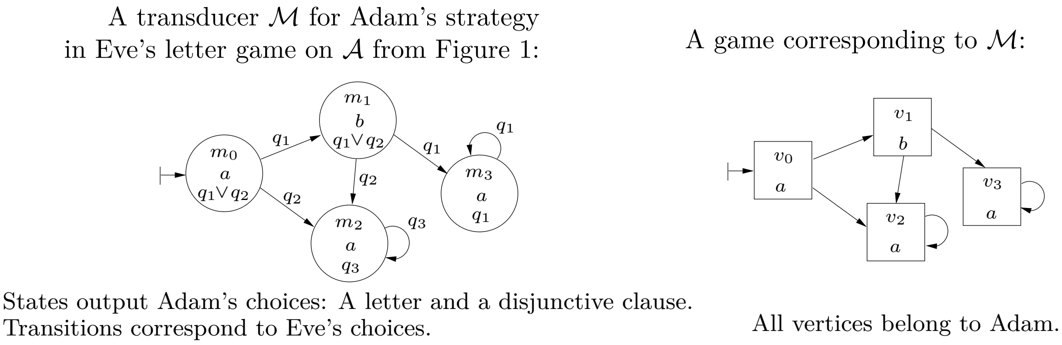

Figure 2 An example of a strategy for Adam in Eve's letter game, and the corresponding game, as used in the proof of Lemma 14.

Indeed, considering claim (i), a play of $\mathcal{G}$ corresponds to a possible path in $\mathcal{M}$, which corresponds to a possible play in Eve's letter game that agrees with $\mathcal{M}$. If there is a play of $\mathcal{G}$ whose labelings are not in $L(\mathcal{A})$, it follows that Eve can win her letter game against $\mathcal{M}$, by forcing a word not in $L(\mathcal{A})$, which contradicts the assumption that $\mathcal{M}$ is a winning strategy.

As for claim (ii), Adam can play in the $\mathcal{G} \times \mathcal{A}$ game according to $\mathcal{M}$: whenever in a vertex (v, b) of $\mathcal{G} \times \mathcal{A}$, where v is a vertex of $\mathcal{G}$ and b a transition condition of $\mathcal{A}$, Adam chooses the next vertex according to the transition in the corresponding state in $\mathcal{M}$. Thus, the generated play in $\mathcal{G} \times \mathcal{A}$ corresponds to a play in Eve's letter game that agrees with $\mathcal{M}$, which Adam is guaranteed to win.

In the case that $\mathcal{A}$ is not compliant with Adam's letter game, we do the dual: Consider the transition conditions in $\mathcal{A}$ to be in DNF, have a winning strategy for Eve, modeled by a transducer $\mathcal{M}$ whose states output a letter and a conjunctive clause and whose transitions correspond to Adam's choices, and translate it to a Σ-labeled one-player game $\mathcal{G}$, in which all vertices belong to Eve. Then, for analogous reasons, Eve loses $\mathcal{G}$, but wins $\mathcal{G} \times \mathcal{A}$. ◀

The equivalence between the 'good for trees' notion and being good for composition with one-player games, follows directly from the generalized definition of 'good for trees' (Definition 7) and the following observation: Every one-player Σ-labeled game is built on top of a Σ-labeled tree (its arena, in case it is infinite, or the expansion of all possible plays, in case of a finite arena), and every Σ-labeled tree can be viewed as a one-player game by assigning ownership of all positions to either Adam or Eve. Clearly, every Σ-labeled tree t belongs to $L(A)\triangle$ iff Eve wins the game on t in which all nodes belong to Adam.

▶ **Lemma 15.** *An alternating automaton $\mathcal{A}$ is good for trees iff it is good for composition with one-player games.*

A finite-arena game can be viewed as an alternating automaton over a singleton alphabet, suggesting that being good for composition with alternating automata implies being good for composition with finite games. This is indeed the case and by Lemmas 13 and 14, it also implies being good for infinite games. It turns out that even though alternating automata over a non-singleton alphabet cannot be just viewed as games, the other direction also holds.

▶ **Lemma 16.** *An alternating automaton $\mathcal{A}$ is good for game-composition if and only if it is good for automata-composition.*

Proof. We start with showing that being good for automata-composition implies being good for game-composition. Given a game over a finite Σ-arena $R = (V, X, V_E, L)$ with initial position ι and winning condition $W \subseteq \Sigma^\omega$, consider the automaton $A_R = (V, \{a\}, \iota, \delta, L)$ over the alphabet $\{a\}$ with acceptance condition W, where $\delta(v, a) = \bigvee\{v' | (v, v') \in X\}$ if $v \in V_E$

and $\delta(v,a) = \bigwedge\{v' | (v,v') \in X\}$ otherwise. A_R accepts the unique word in $\{a\}^\omega$ if and only if Eve has a winning strategy in R from ι, because a strategy in R is exactly a run of A_R over this unique word, and it is winning if and only if the run is accepting.

Then, observing that the synchronized product $R \times \mathcal{A}$ between a finite game and an automaton is the special case of the synchronized product $A_R \times \mathcal{A}$, we conclude that if an automaton is good for automata-composition, then Eve wins $R \times \mathcal{A}$ if and only if $A_R \times \mathcal{A}$ is non-empty, if and only if A_R is non-empty, i.e. if and only if Eve has a winning strategy in R. That is, $\mathcal{A}$ is good for finite game-composition. From Lemmas 13 and 14, $\mathcal{A}$ is also good for composition with infinite games.

For the other direction, assume that $\mathcal{A}$ is good for game composition. We show that $\mathcal{A}$ is also good for automata composition. Consider an alternating automaton $\mathcal{B}$ with acceptance condition $L(\mathcal{A})$. Let $w \in L(\mathcal{B})$ and consider the model-checking game $\mathcal{G}(w,\mathcal{B})$ in which Eve has a winning strategy. Since $\mathcal{A}$ is good for game composition, Eve also has a winning strategy s in $\mathcal{G}(w,\mathcal{B}) \times \mathcal{A}$. We use this strategy to build a strategy s' for Eve in $\mathcal{G}(w,\mathcal{B} \times \mathcal{A})$.

First recall from Def. 2, that the transitions of $\mathcal{B} \times \mathcal{A}$ are of the form $f(c,q) \vee f(c',q)$, $f(c,q) \wedge f(c',q)$, corresponding to choices in $\mathcal{B}$, or of the form $g(q,c) \vee g(q,c')$ or $g(q,c) \wedge g(q,c')$, corresponding to choices in $\mathcal{A}$. At a position $(w, f(c,q) \vee f(c',q))$, Eve plays as s plays at $((w, c \vee c'), q)$; at $(w, g(q,c) \vee g(q,c'))$ Eve plays as s plays at $((w, c \vee c'), q)$. Since the winning condition in both games is determined by the states of $\mathcal{A}$ visited infinitely often, if s is winning, so is s'. Therefore $L(\prod \mathcal{B}\mathcal{A})$ accepts w and $L(\mathcal{B}) \subseteq L(\mathcal{B} \times \mathcal{A})$. In the case $w \notin L(\mathcal{B})$, Adam can similarly copy his strategy from $\mathcal{G}(w,\mathcal{B}) \times \mathcal{A}$ into $\mathcal{G}(w,\mathcal{B} \times \mathcal{A})$.

We conclude that $\mathcal{B} \times \mathcal{A}$ is equal to $\mathcal{B}$ and therefore $\mathcal{A}$ is good for automata composition. ◀

▶ Remark 17. We observe that compositionality with word automata also implies compositionality with (symmetric, unranked) tree automata. A tree automaton is similar to a word automaton, except that its transitions have modalities $\Box q$ and $\Diamond q$ instead of states. Then, the model-checking (or membership) game of a tree and an automaton is a game, as for words, where, in addition, the modalities $\Box q$ and $\Diamond q$ dictate whether the choice of successor in the tree is given to Adam or Eve. Then, if $\mathcal{A}$ composes with games, it must in particular compose with the model-checking game of t and a tree automaton $\mathcal{B}$ with acceptance condition $L(\mathcal{A})$. If Eve (Adam) wins the model-checking game $\mathcal{G}(t,\mathcal{B})$, she (he) also wins $\mathcal{G}(t,\mathcal{B}) \times \mathcal{A}$. Her (his) winning strategy in this game is also a winning strategy in $\mathcal{G}(t,\mathcal{B} \times \mathcal{A})$, so $\mathcal{B} \times \mathcal{A}$ must accept (reject) t. $\mathcal{A}$ therefore composes with tree-automata.

While Theorem 8 obviously holds also for nondeterministic automata, we observe that in the absence of universality, the definitions of Section 3 can be relaxed into asymmetrical ones. For letter games, history determinism, and good-for-trees, it follows directly from the definitions, as only their 'nondeterministic part' applies. For composition with games and automata, we also show that it suffices to compose with universal automata and games.

▶ **Lemma 18.** *A nondeterministic automaton $\mathcal{A}$ is good for automata-composition if and only if it is good for composition with universal automata.*

5 Expressiveness

For some acceptance conditions, such as weak, Büchi, and co-Büchi, alternating automata are more expressive than deterministic ones. For other conditions, such as parity, Rabin, Streett, and Muller, they are not. Yet, also for the latter conditions, once considering the condition's *index*, which is roughly its size, alternating automata are more expressive than deterministic automata with the same acceptance condition and index. (More details on the different acceptance conditions can be found, for example, in [**?**].)

Most acceptance conditions are preserved, together with their index, when taking the product of an automaton $\mathcal{A}$ with an auxiliary memory M. In such a product, the states of the resulting automaton are pairs (q, m) of a state q from $\mathcal{A}$ and a state m from M, while the acceptance condition is defined according to the projection of the states to their $\mathcal{A}$'s component. In particular, the weak, Büchi, co-Büchi, parity, Rabin, and Streett conditions are preserved, together with their index, under memory product, while the very-weak and Muller conditions are not.

For showing that GFG automata are not more expressive than deterministic automata with the same acceptance condition and index, we generalize the proof of [?] from nondeterminism to alternation. The idea is to translate an alternating GFG automaton $\mathcal{A}$ to an equivalent deterministic automaton $\mathcal{D}$ by taking the product of $\mathcal{A}$ with the transducers that model the history deterministic strategies of $\mathcal{A}$.

▶ **Theorem 19.** *Every alternating GFG automaton with acceptance condition that is preserved under memory-product can be translated to a deterministic automaton with the same acceptance condition and index. In particular, this is the case for weak, co-Büchi, Büchi, parity, Rabin, and Streett GFG alternating automata of any index.*

Proof. Consider an alternating GFG automaton $\mathcal{A} = (\Sigma, Q, \iota, \delta, \alpha)$. For conveniency, we may assume by Proposition 12 that $\mathcal{A}$'s transition conditions are in DNF.

By Corollary 10, the history-determinism of $\mathcal{A}$'s universality and nondeterminism is witnessed by finite-memory strategies, modeled by transducers $M_A = (I_A, O_A, M_A, \iota_A, \rho_A, \chi_A)$ and $M_E = (I_E, O_E, M_E, \iota_E, \rho_E, \chi_E)$, respectively. Observe that since transition conditions of $\mathcal{A}$ are in DNF, the strategy $\mathcal{M}_A$ chooses a state of $\mathcal{A}$ for every letter in Σ and clause of states of $\mathcal{A}$, while the transitions in $\mathcal{M}_E$ are made in pairs, first choosing a clause for a letter in Σ and then updating the memory again according to Adam's choice of a state of $\mathcal{A}$.

Formally, we have that the elements of I_A are pairs (a, C), where $a \in \Sigma$ and C is a clause of states in Q, and that elements of O_A are states in Q, while elements of I_E are in $\Sigma \cup Q$ and elements of O_E are either clauses of states in Q or ϵ (when only updating the memory).

Let $\mathcal{D} = (\Sigma, Q', \iota', \delta', \alpha')$ be the deterministic automaton that is the product of $\mathcal{A}$ and $\mathcal{M}_A$, in which the universality is resolved according to $\mathcal{M}_A$ and the nondeterminism according to $\mathcal{M}_E$. That is, $Q' = Q \times M_A \times M_E$, $\iota' = (\iota, \iota_A, \iota_E)$, $\alpha'(q, x, y) = \alpha(q)$, and for every $q \in Q$, $x \in M_A$, $y \in M_E$, and $a \in \Sigma$, we have $\delta'((q, x, y), a) = (q', x', y')$, where $x' = \rho_A(x, (a, \chi_E(\rho_E(y, a))))$, $q' = \chi_A(x')$, and $y' = \rho_E(\rho_E(y, a), q')$.

Observe that $\mathcal{A}$ and $\mathcal{D}$ have the same acceptance condition and the same index, as $\mathcal{A}$'s acceptance condition is preserved under memory-product. We also have that $\mathcal{A}$ and $\mathcal{D}$ are equivalent, since for every word w, the games $\mathcal{G}(w, \mathcal{A})$, $\mathcal{G}_{\mathcal{M}_A, \mathcal{M}_E}(w, \mathcal{A})$, and $\mathcal{G}(w, \mathcal{D})$ have the same winner. ◀

6 Succinctness

Nondeterministic GFG automata over finite words and weak GFG automata over infinite words can be pruned to equivalent deterministic automata [?, ?]. We show that this remains true in the alternating setting. The succinctness of nondeterministic GFG Büchi automata compared to deterministic ones is still an open question, having no lower bound and a quadratic upper bound, whereas nondeterministic GFG co-Büchi automata can be exponentially more succinct than their deterministic counterparts [?] . We show that in the alternating setting, both Büchi and co-Büchi GFG automata are singly-exponential more succinct than deterministic ones. We leave open the question of whether stronger acceptance conditions can allow GFG automata to be doubly-exponential more succinct than deterministic ones.

In this section we focus on specific classes of automata, and for brevity use three letter acronyms in {D, N, A} × {W, B, C} × {W} when referring to them. The first letter stands for the transition mode (deterministic, nondeterministic, alternating); the second for the acceptance-condition (weak, Büchi, co-Büchi); and the third indicates that the automaton runs on words. For example, DBW stands for a deterministic Büchi automaton on words. We also use DFA, NFA, and WFA when referring to automata over finite words.

In the nondeterministic setting, the proof that GFG NFAs and GFG NWWs are not more succinct than DFAs and DWWs, respectively, is based on two properties: i) In a minimal DFA or DWW for a language L, there is exactly one state for every Myhill-Nerode equivalence class of L. (Recall that finite words u and v are in the same class C when for every word w, $uw \in L$ iff $vw \in L$. For a class C, the language $L(C)$ of C is $\{w \mid \exists u \in C$ such that $uw \in L\}$.); and ii) In a nondeterministic GFG automaton $\mathcal{A}$ that has no redundant transitions, for every finite word u and states $q, q' \in \delta(u)$, we have $L(\mathcal{A}^q) = L(\mathcal{A}^{q'})$.

For showing that GFG AFAs and AWWs are not more succinct than DFAs and DWWs, respectively, we provide in the following lemma as a variant of the above second property.

▶ **Lemma 20.** *Consider a GFG alternating automaton $\mathcal{A}$. Then for every class C of the Myhill-Nerode equivalence classes of $L(\mathcal{A})$, there is a state q in $\mathcal{A}$, such that $L(\mathcal{A}^q) = L(C)$.*

Proof. Let τ and η be history-deterministic strategies of $\mathcal{A}$ for Eve and Adam, respectively. For every finite word u, let $C(u)$ be the Myhill-Nerode equivalence class of u, and $q(u)$ be the state that $\mathcal{A}$ reaches when running on u along τ and η. We claim that $L(\mathcal{A}^{q(u)}) = L(C(u))$.

Indeed, if there is a word $w \in L(C(u)) \setminus L(\mathcal{A}^{q(u)})$ then Adam wins the model-checking game $\mathcal{G}_\tau(uw, \mathcal{A})$, by playing according to η until reaching $q(u)$ over u and then playing unrestrictedly for rejecting the w suffix, contradicting the history determinism of τ.

Analogously, if there is a word $w \in L(\mathcal{A}^{q(u)}) \setminus L(C(u))$ then Eve wins the model-checking game $\mathcal{G}_\eta(uw, \mathcal{A})$, by playing according to τ until reaching $q(u)$ over u and then playing unrestrictedly for accepting the w suffix, contradicting the history determinism of η. ◀

The insuccinctness of GFG AFAs and GFG AWWs directly follows.

▶ **Theorem 21.** *For every GFG AFA or GFG AWW $\mathcal{A}$, there is an equivalent DFA or DWW $\mathcal{A}'$, respectively, such that the number of states in $\mathcal{A}'$ is not more than in $\mathcal{A}$.*

As opposed to weak automata, minimal deterministic Büchi and co-Büchi automata do not have the Myhill-Nerode classification, and indeed, it was shown in [**?**] that GFG NCWs can be exponentially more succinct than DCWs.

We show that GFG ACWs are also only singly-exponential more succinct than DCWs. We translate a GFG ACW $\mathcal{A}$ to an equivalent DCW $\mathcal{D}$ in four steps: i) Dualize $\mathcal{A}$ to a GFG ABW $\mathcal{B}$; ii) Translate $\mathcal{B}$ to an equivalent NBW, having an $O(3^n)$ state blow-up [**?**, **?**], and prove that the translation preserves GFGness; iii) Translate $\mathcal{B}$ to an equivalent DBW $\mathcal{C}$, having an additional quadratic state blow-up [**?**]; and iv) Dualize $\mathcal{C}$ to a DCW $\mathcal{D}$.

The main difficulty is, of course, in the second step, showing that the translation of an ABW to an NBW preserves GFGness. For proving it, we first need the following key lemma, stating that in a GFG ABW in which the transition conditions are given in DNF, the history-deterministic strategies can only use the current state and the prefix of the word read so far, ignoring the history of the transition conditions.

▶ **Lemma 22.** *Consider an ABW $\mathcal{A}$ with transition conditions in DNF and history-deterministic nondeterminism. Then Eve has a strategy $\tau : Q \times \Sigma^* \to \mathsf{B}^+(Q)$ (and not only a strategy $(\mathsf{B}^+(Q) \times \Sigma)^* \to \mathsf{B}^+(Q)$), such that for every word $w \in L(\mathcal{A})$, Eve wins $\mathcal{G}_\tau(w, A)$.*

Proof. Let $\xi : (\mathsf{B}^+(Q) \times \Sigma)^* \to \mathsf{B}^+(Q)$ be a 'standard' history-deterministic strategy for Eve. Observe that since the transition conditions of $\mathcal{A}$ are in DNF, ξ's domain is $(Q \times \Sigma)^*$, and the run of $\mathcal{A}$ on w following ξ, namely $\mathcal{G}_\xi(w, \mathcal{A})$, is an infinite tree, in which every node is labeled with a state of $\mathcal{A}$. A history h for ξ is thus a finite sequence of states and a finite word. Let $\mathsf{yearn}(h)$ denote the number of positions in the sequence of states in h from the last visit to α until the end of the sequence. We shall say "a history h for a word u" when h's word component is u, and "h ends with q" when the sequence of states in h ends in q.

We inductively construct from ξ a strategy $\tau : Q \times \Sigma^* \to \mathsf{B}^+(Q)$, by choosing for every history $(q, u) \in Q \times \Sigma^*$ some history h of ξ, as detailed below, and setting $\tau(q, u) = \xi(h)$.

At start, for $u = \epsilon$, we set $\tau(\iota, \epsilon) = \xi(\iota, \epsilon)$. In a step in which every history of ξ for u ends with a different state, we set $\tau(q, u) = \xi(h)$, where h is the single history that ends with q.

The challenge is in a step in which several histories of ξ for u end with the same state, as τ can follow only one of them. We define that $\tau(q, u) = \xi(h)$, where h is a history of ξ for u that ends with q, and $\mathsf{yearn}(h)$ is maximal among the histories of ξ for u that ends in q. Every history of ξ that is not followed by τ is considered "stopped", and in the next iterations of constructing τ, histories of ξ with stopped prefixes will not be considered.

As ξ is a winning strategy for Eve, all paths in $\mathcal{G}_\xi(w, A)$ are accepting. Observe that $\mathcal{G}_\tau(w, A)$ is a tree in which some of the paths are from $\mathcal{G}_\xi(w, A)$ and some are not—whenever a history is stopped in the construction of τ, a new path is created, where its prefix is of the stopped history and the continuation follows the path it was redirected to. We will show that, nevertheless, all paths in $\mathcal{G}_\tau(w, A)$ are accepting.

Assume toward contradiction a path ρ of $\mathcal{G}_\tau(w, A)$ that is not accepting, and let k be its last position in α. The path ρ must have been created by infinitely often redirecting it to different histories of ξ, as otherwise there would have been a rejecting path of ξ. Now, whenever ρ was redirected, it was to a history h, such that $\mathsf{yearn}(h)$ was maximal. Thus, in particular, this history did not visit α since position k. Therefore, by König's lemma, there is a path π of $\mathcal{G}_\xi(w, A)$ that does not visit α after position k, contradicting the assumption that all paths of $\mathcal{G}_\xi(w, A)$ are accepting. ◀

We continue with showing that the translation of an ABW to an NBW preserves GFGness.

▶ **Lemma 23.** *Consider an ABW $\mathcal{A}$ for which the nondeterminism is history deterministic. Then the nondeterminism in the NBW $\mathcal{A}'$ that is derived from $\mathcal{A}$ by the breakpoint (Miyano-Hayashi) is also history deterministic.*

Proof. Consider an alternating GFG automaton $\mathcal{A} = (\Sigma, Q, \iota, \delta, \alpha)$. We write $\overline{\alpha}$ for $Q \setminus \alpha$. By Proposition 12, we may assume that the transition conditions of $\mathcal{A}$ are given in DNF.

The breakpoint construction [?] generates from $\mathcal{A}$ an equivalent NBW $\mathcal{A}'$, by intuitively keeping track of a pair $\langle S, O\rangle$ of sets of states of $\mathcal{A}$, where S is the set of states that are visited at the current step of a run, and O are the states among them that "owe" a visit to α. A state owes a visit to α if there is a path leading to it with no visit to α since the last "breakpoint", which is a state of $\mathcal{A}'$ in which $O = \emptyset$. The accepting states of $\mathcal{A}'$ are the breakpoints.

For providing the formal definition of $\mathcal{A}'$, we first construct from δ and each letter $a \in \Sigma$, a set $\Delta(a)$ of transition functions $\gamma_1, \gamma_2, \ldots \gamma_k$, for some $k \in \mathbb{N}$, such that each of them has only universality and corresponds to a possible resolution of the nondeterminism in every state of $\mathcal{A}$. For example, if $\mathcal{A}$ has states q_1, q_2, and q_3, and its transition function δ for the letter a is $\delta(q_1, a) = q_1 \wedge q_3 \vee q_2; \delta(q_2, a) = q_2 \vee q_3 \vee q_1$; and $\delta(q_3, a) = q_2 \wedge q_3 \vee q_1$, then $\Delta(a)$ is a set of twelve transition functions, where $\gamma_1(q_1, a) = q_1 \wedge q_3; \gamma_1(q_2, a) = q_2$; and $\gamma_1(q_3, a) = q_2 \wedge q_3$, etc.., corresponding to the possible ways of resolving the nondeterminism in each of the states.

For convenience, we shall often consider a conjunctive formula over states as a set of states, for example $q_1 \wedge q_3$ as $\{q_1, q_3\}$. For a set of states $S \subseteq Q$, a letter a, and a transition function $\gamma \in \Delta(a)$, we define $\gamma(S) = \bigcup_{q \in S} \gamma(q, a)$.

Formally, the breakpoint construction [?] generates from $\mathcal{A}$ an equivalent NBW $\mathcal{A}' = (\Sigma, Q', \iota', \delta', \alpha')$ as follows:

- $Q' = \{\langle S, O\rangle \mid O \subseteq S \subseteq Q\}$
- $\iota' = (\{\iota\}, \{\iota\} \cap \overline{\alpha})$
- δ' : For a state $\langle S, O\rangle$ of $\mathcal{B}$ and a letter $a \in \Sigma$:
  - If $O = \emptyset$ then $\delta'(\langle S, O\rangle, a) = \{\langle \hat{S}, \hat{O}\rangle \mid$ exists a transition function $\gamma \in \Delta$, such that $\hat{S} = \gamma(S, a)$ and $\hat{O} = \gamma(S, a) \cap \overline{\alpha}\}$
  - If $O \neq \emptyset$ then $\delta'(\langle S, O\rangle, a) = \{\langle \hat{S}, \hat{O}\rangle \mid$ exists a transition function $\gamma \in \Delta$, such that $\hat{S} = \gamma(S, a)$ and $\hat{O} = \gamma(O, a) \cap \overline{\alpha}\}$
- $\alpha' = \{(S, \emptyset) \mid S \subseteq Q\}$

Observe that the breakpoint construction determinizes the universality of $\mathcal{A}$, while morally keeping its nondeterminism as is. This will allow us to show that Eve can use her history-deterministic strategy for $\mathcal{A}$ also for resolving the nondeterminism in $\mathcal{A}'$.

At this point we need Lemma 22, guaranteeing a strategy $\tau : Q \times \Sigma^* \to \mathsf{B}^+(Q)$ for Eve. At each step, Eve should choose the next state in $\mathcal{A}'$, according to the read prefix u and the current state (S, O) of $\mathcal{A}'$. Observe that τ assigns to every state $q \in S$ a set of states $S' = \tau(q, u)$, following a nondeterministic choice of $\delta(q, u)$; Together, all this choices comprise some transition function $\gamma \in \Delta$. Thus, in resolving the nondeterminism of $\mathcal{A}'$, Eve's strategy τ' is to choose the transition γ that is derived from τ.

Since τ guarantees that all the paths in the τ-run-tree of $\mathcal{A}$ on a word $w \in L(\mathcal{A})$ are accepting, the corresponding τ'-run of $\mathcal{A}'$ on w is accepting, as infinitely often all the $\mathcal{A}$-states within $\mathcal{A}'$'s states visit α. ◀

▶ **Theorem 24.** *The translation of a GFG ABW or GFG ACW $\mathcal{A}$ to an equivalent DBW or DCW, respectively, involves a $2^{\Theta(n)}$ state blow-up.*

7 Conclusions

GFG in alternating automata is the sum of its parts. Through studying the various definitions of good-for-games and their generalizations, a common theme emerged: each definition can be divided into a definition for nondeterminism and a definition for universality, and the conjunction of these suffices to guarantee good-for-gameness. For example, it suffices for an automaton to compose with both universal automata and nondeterministic automata for it to compose with alternating automata, even alternating tree automata. In other words, good-for-games nondeterminism and universality cannot interact pathologically to generate alternating automata not good-for-games, and neither can they ensure good-for-gameness without each being independently good-for-games. This should in particular facilitate checking whether an automaton is good for games, as it can be done separately for universality and nondeterminism.

Between words, trees, games, and automata. Good for games automata allow us to go between word automata, tree automata, and games. In the recent translations from alternating parity word automata into weak automata [?, ?], the key techniques involve adapting methods that use *finite one-player games* to process *infinite* structures that are in some sense between words and trees, and use these to manipulate alternating automata. These translations depend, implicitly or explicitly, on the compositionality that enable the step from asymmetrical one-player games, i.e. trees, to alternating automata. Studying good-for-gameness provides us with new tools to move between words, trees, games, and automata, and better understand how nondeterminism, universality, and alternations interact in this context.

A Additional Proofs

Proof of Lemma 9. For the first direction, we assume that the nondeterminism in $\mathcal{A}$ is history-deterministic, witnessed by a strategy τ_E of Eve. Then Eve wins her letter game by following τ_E, since if Adam plays a word $w \in L(\mathcal{A})$, then the resulting play of the letter game, consisting of a sequence $\pi = b_0, b_1 \dots$ of transition conditions and a word $w = w_0, w_1 \dots$, induces a play in $\mathcal{G}(w, \mathcal{A})$ that agrees with τ_E. Since τ_E witnesses the history-determinism of $\mathcal{A}$, such a play must be winning, that is, π restricted to Q must satisfy $\mathcal{A}$'s acceptance condition.

Symmetrically, if the universality in $\mathcal{A}$ is history-deterministic, witnessed by a strategy τ_A of Adam, it induces a winning strategy for him in his letter-game.

For the converse, assume that Eve wins her letter game with a strategy s. We argue that this strategy also witnesses the history-determinism of the nondeterminism in $\mathcal{A}$, namely that Eve wins $\mathcal{G}_s(w, \mathcal{A})$ for every word $w \in L(\mathcal{A})$.

Indeed, if a play $\pi \in \mathcal{G}_s(w, \mathcal{A})$ does not satisfy the acceptance condition of $\mathcal{A}$ while $w \in L(\mathcal{A})$, then the play in Eve's letter game in which Adam plays w and resolves universality as in π would both agree with s and be winning for Adam, contradicting that s is winning for Eve. The nondeterminism of $\mathcal{A}$ is therefore history-deterministic.

Symmetrically, if Adam wins his letter game with strategy τ_A, the universality in $\mathcal{A}$ is history-deterministic, witnessed by τ_A. Hence, if $\mathcal{A}$ is compliant with the letter games, it is also history deterministic. ◀

Proof of Corollary 10. Since the letter game is a finite ω-regular game, its winner has a finite-memory strategy. ◀

Proof of Proposition 11. For every word w, the model-checking games $\mathcal{G}(w, \mathcal{A})$ and $\mathcal{G}(w, \overline{\mathcal{A}})$ are the same, just switching roles between Adam and Eve. Thus, the history-deterministic strategy for Adam in $\mathcal{A}$ can serve Eve in $\overline{\mathcal{A}}$ and vice versa. ◀

Proof of Proposition 12. It is enough to show that changing any transition condition of an alternating automaton $\mathcal{A}$ to DNF does not influence its history determinism for Eve/Adam.

Assume that the nondeterminism in $\mathcal{A}$ is history-deterministic, witnessed by a strategy τ of Eve. Let $\mathcal{A}'$ be an automaton that is derived from $\mathcal{A}$ by changing any transition condition b of A, for a state q and a letter a, into its DNF form b'. Let $k \in \mathbb{N}$ be the depth of alternation between nondeterminism and universality in b.

We show that Eve has a history-deterministic strategy τ' for $\mathcal{A}'$, by adapting τ. We call *local b-strategy* a way of resolving the nondeterminism in b. First observe that for every local b-strategy s, there is a corresponding local b'-strategy s' that chooses the set of states that Adam can force in k steps over b if Eve follows s; conversely for every local b'-strategy s', there is a corresponding local b-strategy s such that the set of states that Adam can force in k steps over b if Eve follows s is exactly Eve's choice in s'.

The strategy τ' can then be defined by replacing b-local strategies from τ with the corresponding b'-local strategies. More precisely, for every history $(h', u) \in (\mathsf{B}^+(Q))^* \times \Sigma^*$, we have that $\tau'(h'q, u)$ is the b'-local strategy corresponding to $\tau(hq, u)$, where h is the sequence of transition conditions derived from h', by replacing the b' transitions consistent with a b'-local strategy s' with the b transitions consistent with the corresponding b-local strategy s. Since the corresponding local strategies only differ in the paths taken within b and b', but not in the resulting states reached, τ' preserves Eve's victory.

The arguments for the other direction, that is assuming that the nondeterminism in $\mathcal{A}'$ is history-deterministic, and proving that this is also the case for $\mathcal{A}$, are analogous, and so are the arguments for how to adapt a history-deterministic strategy for Adam. ◀

Proof of Lemma 13. Assume $\mathcal{A}$ is compliant with the letter games but that for some Σ arena G, the game on G with winning condition $L(\mathcal{A})$ and the synchronized product $G \times \mathcal{A}$ have different winners. If Eve wins in G, then she can combine her winning strategy τ in G and her winning strategy τ' in her letter-game to win in the synchronized product $G \times \mathcal{A}$: she resolves the choices in G according to τ, thus ensuring that the play in $G \times \mathcal{A}$ follows a path of G labeled with a word accepted by $\mathcal{A}$. Then, she can resolve the nondeterminism in $\mathcal{A}$ according to τ'. Since τ' is winning in the letter game and all plays agreeing with τ follow a word in G that is in $L(A)$, all plays agreeing with the combination of τ and τ' are accepting.

Similarly, if Adam wins in G, his strategy in G and in his letter game combine into a winning strategy in the product $G \times \mathcal{A}$. ◀

Proof of Lemma 18. Since universal automata is a subclass of alternating automata, one direction is immediate and we only need to show that if $\mathcal{A}$ is good for composition with all universal automata, it is good for composition with all automata.

Assume that $\mathcal{A}$ is good for composition with universal automata. We will show that $\mathcal{A}$ composes with any game G with acceptance condition $L(\mathcal{A})$. Assume Eve wins in G. Let G' be the game induced by a positional winning strategy s for Eve in G, seen as a universal automaton on the singleton alphabet. Since $\mathcal{A}$ composes with universal automata, it composes with G', and Eve has a winning strategy s' in $G' \times \mathcal{A}$. Then, Eve's strategy in $G \times \mathcal{A}$ consisting of using s to resolve the branching in G and s' to resolve the nondeterminism in $\mathcal{A}$ is winning. If Adam wins in G, then his winning strategy in $G \times \mathcal{A}$ resolves the branching in G according to a winning strategy. This forces the play to follow a word not in $L(\mathcal{A})$. Eve has no accepting run in $\mathcal{A}$ for such a word and therefore can not win in $G \times \mathcal{A}$ against this strategy. ◀

Proof of Theorem 21. The argument below corresponds to a DWW $\mathcal{A}$, and stands also for a DFA $\mathcal{A}$.

By [?], a minimal DWW for a language $L(\mathcal{A})$ has a single state for every Myhill-Nerode class of $L(\mathcal{A})$. By Lemma 20, $\mathcal{A}$ has at least one state for each such class, from which the claim follows. ◀

Proof of Theorem 24. The lower bound follows from [?], where it is shown that determinization of GFG NCWs is in $2^{\Omega(n)}$. It directly generalizes to GFG ACWs, and by dualization to GFG ABWs: Given a GFG ACW $\mathcal{A}$, we can dualize it to an ABW $\mathcal{B}$, which is also GFG by Proposition 11. Then, we can determinize $\mathcal{B}$ to a DBW $\mathcal{D}$ and dualize the latter to a DCW $\mathcal{C}$ equivalent to $\mathcal{A}$.

The upper bound follows from Lemma 23, getting an $O(3^n)$ state blow-up for translating a GFG ABW to an equivalent GFG NBW, and then another quadratic state blow-up, due to [?], from GFG NBW to DBW. For determinizing a GFG ACW, we have the same result due to dualization and Proposition 11. ◀