THE UNIVERSITY *of* EDINBURGH

Multipebble Simulations for Alternating Automata - (Extended Abstract)

Citation for published version:
Clemente, L & Mayr, R 2010, 'Multipebble Simulations for Alternating Automata - (Extended Abstract)'. in P Gastin & F Laroussinie (eds), CONCUR 2010 - CONCURRENCY THEORY. Springer-Verlag GmbH, BERLIN, pp. 297-312, 21st Conference on Concurrency Theory, Paris, 31-3 September.

Published In:
CONCUR 2010 - CONCURRENCY THEORY

Multipebble Simulations for Alternating Automata
(Extended Abstract)

Lorenzo Clemente and Richard Mayr

LFCS. School of Informatics. University of Edinburgh. UK

Abstract. We study generalized simulation relations for alternating Büchi automata (ABA), as well as alternating finite automata. Having multiple pebbles allows the Duplicator to "hedge her bets" and delay decisions in the simulation game, thus yielding a coarser simulation relation. We define (k_1, k_2)-simulations, with k_1/k_2 pebbles on the left/right, respectively. This generalizes previous work on ordinary simulation (i.e., $(1, 1)$-simulation) for nondeterministic Büchi automata (NBA) in [3] and ABA in [4], and $(1, k)$-simulation for NBA in [2].
We consider direct, delayed and fair simulations. In each case, the (k_1, k_2)-simulations induce a complete lattice of simulations where $(1, 1)$- and (n, n)-simulations are the bottom and top element (if the automaton has n states), respectively, and the order is strict. For any fixed k_1, k_2, the (k_1, k_2)-simulation implies (ω-)language inclusion and can be computed in polynomial time. Furthermore, quotienting an ABA w.r.t. $(1, n)$-delayed simulation preserves its language. Finally, multipebble simulations yield new insights into the Miyano-Hayashi construction [10] on ABA.

1 Introduction

We consider simulation relations on (alternating) finite- and infinite word automata: nondeterministic finite automata (NFA), alternating finite automata (AFA), nondeterministic Büchi automata (NBA) and alternating Büchi automata (ABA). Simulation preorder is a notion of semantic comparison of two states, called left state and right state, in automata, where the larger right state can match all moves of the smaller left one in a stepwise way. Simulation preorder implies language inclusion on NFA/AFA/NBA/ABA [3, 4], but not vice-versa. While checking language inclusion is PSPACE-complete for all these classes of automata [7, 11], the simulation relation can be computed in polynomial time [3, 4].

Checking simulation preorder between two states can be presented as a game with two players, Spoiler and Duplicator, where Spoiler tries to prove that the simulation relation does not hold while Duplicator has the opposite objective. In every round of the simulation game, Spoiler chooses a transition from the current left state and Duplicator must choose a transition from the current right state which has the same action label. Duplicator wins iff the game respects the accepting states in the automata, and different requirements for this yield finer or coarser simulation relations. In *direct simulation*, whenever the left state is accepting, the right state must be accepting. In *delayed simulation*, whenever the left state is accepting, the right state must be eventually accepting. In *fair simulation*, if the left state is accepting infinitely often, then the right state must be accepting infinitely often. For finite-word automata, only direct simulation is meaningful, but for Büchi automata delayed and fair simulation yield coarser relations; see [3] for an overview.

These notions have been extended in two directions. Etessami [2] defined a hierarchy of $(1, k)$ multipebble simulations on NBA. Intuitively, the k pebbles on the right side allow Duplicator to “hedge her bets” and thus to delay making decisions. This extra power of Duplicator increases with larger k and yields coarser simulation relations.

A different extension by Wilke and Fritz [4] considered simulations on ABA. In an ABA, a state is either existential or universal. The idea is that Spoiler moves from existential left states and universal right states, and dually for Duplicator.

Our contribution. We consider (k_1, k_2)-simulations on ABA, i.e., with multiple pebbles on both sides: k_1 on the left and k_2 on the right. Intuitively, Duplicator controls pebbles on universal states on the left and existential states on the right (and dually for Spoiler). This generalizes all previous results: the $(1, k)$-simulations on NBA of [2] and the $(1, 1)$-simulations on ABA of [4].

For each acceptance condition (direct, delayed, fair) this yields a lattice-structured hierarchy of (k_1, k_2)-simulations, where $(1, 1)$- and (n, n)-simulations are the bottom and top element if the automaton has n states. Furthermore, the order is strict, i.e., more pebbles make the simulation relation strictly coarser in general. For each fixed $k_1, k_2 \geq 0$, (k_1, k_2)-simulations are computable in polynomial time and they imply language inclusion (over finite or infinite words, depending on the type of simulation).

Quotienting AFA w.r.t. (k_1, k_2)-simulation preserves their language. We also provide a corresponding result for ABA by showing that quotienting ABA w.r.t. $(1, n)$-delayed simulation preserves the ω-language. This is a non-trivial result, since a naïve generalization of the definition of semielective-quotients [4] does not work. We provide the correct notion of semielective-quotients for $(1, n)$-simulations on ABA, and show its correctness. Moreover, unlike for NBA [2], quotienting ABA w.r.t. $(1, k)$ delayed simulation for $1 < k < n$ does *not* preserve their language in general.

Finally, multipebble simulations have close connections to various determinization-like constructions like the subset construction for NFA/AFA and the Miyano-Hayashi construction [10] on ABA. In particular, multipebble simulations yield new insights into the Miyano-Hayashi construction and an alternative correctness proof showing an even stronger property.

2 Preliminaries and Basic Definitions

Automata. An alternating Büchi automaton (ABA) $\mathcal{Q}$ is a tuple $(Q, \Sigma, q_I, \Delta, E, U, F)$, where Q is a finite set of states, Σ is a finite alphabet, q_I is the initial state, $\{E, U\}$ is a partition of Q into *existential* and *universal* states, $\Delta \subseteq Q \times \Sigma \times Q$ is the transition relation and $F \subseteq Q$ is the set of accepting states. We say that a state q is accepting if $q \in F$. We use n to denote the cardinality of Q. A nondeterministic Büchi automaton (NBA) is an ABA with $U = \emptyset$, i.e., where all choices are existential. We say that $\mathcal{Q}$ is *complete* iff $\forall (q, a) \in Q \times \Sigma.\ \exists (q, a, q') \in \Delta$.

An ABA $\mathcal{Q}$ recognizes a language of infinite words $\mathcal{L}^\omega(\mathcal{Q})$. The acceptance condition is best described in a game-theoretic way [5]. Given an input word $w \in \Sigma^\omega$, the *acceptance game* $\mathbb{G}^\omega(\mathcal{Q}, w)$ is played by two players, Pathfinder and Automaton. Existential states are controlled by Automaton, while Pathfinder controls universal states. Automaton wins the game $\mathbb{G}^\omega(\mathcal{Q}, w)$ iff she has a winning strategy s.t., for any

Pathfinder counter-strategy, the resulting computation visits some accepting state in F infinitely often. The language $\mathcal{L}^\omega(\mathcal{Q})$ recognized by $\mathcal{Q}$ is defined as the set of words $w \in \Sigma^\omega$ s.t. Automaton wins $\mathbb{G}^\omega(\mathcal{Q}, w)$. See [4] for a formal definition.

If we view an ABA $\mathcal{Q}$ as an acceptor of *finite* words, then we obtain an alternating finite automaton (AFA). For $w = w_0 \dots w_m \in \Sigma^*$, the finite acceptance game $\mathbb{G}^{\mathrm{fin}}(\mathcal{Q}, w)$ is defined as above for $\mathbb{G}^\omega(\mathcal{Q}, w)$, except that the game stops when the last symbol w_m of w has been read, and Automaton wins if the last state is in F. $\mathcal{L}^{\mathrm{fin}}(\mathcal{Q})$ is defined in the obvious way. An alternating transition system (ATS) $\mathcal{Q}$ is an AFA where all states are accepting, and $\mathcal{T}r(\mathcal{Q}) := \mathcal{L}^{\mathrm{fin}}(\mathcal{Q})$ is its trace language. When we just say "automaton", it can be an ABA, AFA or ATS, depending on the context.

If Q is a set, with 2^Q we denote the set of subsets of Q, and, for any $k \in \mathbb{N}$, with $2^{Q,k}$ we denote the subset of 2^Q consisting of elements of cardinality at most k. When drawing pictures, we represent existential states by q and universal states by $\boxed{q}$.

Multipebble simulations. We define multipebble simulations in a game-theoretic way. The game is played by two players, Spoiler and Duplicator, who play in rounds. The objective of Duplicator is to show that simulation holds, while Spoiler has the complementary objective. We use the metaphor of pebbles for describing the game: We call a pebble existential if it is on an existential state, and universal otherwise; *Left* if it is on the l.h.s. of the simulation relation, and *Right* otherwise. Intuitively, Spoiler controls existential *Left* pebbles and universal *Right* pebbles, while Duplicator controls universal *Left* pebbles and existential *Right* pebbles. The presence of > 1 pebbles in each side is due to the further ability of Duplicator to split pebbles to several successors. Moreover, Duplicator always has the possibility of "taking pebbles away". Since not all available pebbles have to be on the automaton, $k+1$ pebbles are at least as good as k.

Formally, let $\mathcal{Q}$ be an alternating automaton, $\mathbf{q}_0 \in 2^{Q,k_1}$ a k_1-set and $\mathbf{s}_0 \in 2^{Q,k_2}$ a k_2-set. We define the basic (k_1, k_2)-simulation game $\mathbb{G}_{(k_1,k_2)}(\mathbf{q}_0, \mathbf{s}_0)$ as follows. Let Γ^{Sp} and Γ^{Dup} be a set of actions (or transitions) for the two players (to be specified below). In the initial configuration $\langle \mathbf{q}_0, \mathbf{s}_0 \rangle$, *Left* pebbles are on $\mathbf{q}_0$ and *Right* pebbles on $\mathbf{s}_0$. If the current configuration at round i is $\langle \mathbf{q}_i, \mathbf{s}_i \rangle$, then the next configuration $\langle \mathbf{q}_{i+1}, \mathbf{s}_{i+1} \rangle$ is determined as follows:

- Spoiler chooses a transition $(\mathbf{q}_i, \mathbf{s}_i, a_i, \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Sp}}$.
- Duplicator chooses a transition $(\mathbf{q}_i, \mathbf{s}_i, a_i, \mathbf{q}', \mathbf{s}', \mathbf{q}_{i+1}, \mathbf{s}_{i+1}) \in \Gamma^{\mathrm{Dup}}$.

We now define the two transition relations. Let $\mathbf{q}^E := \mathbf{q} \cap E$ be the set of existential states in $\mathbf{q}$, and define $\mathbf{q}^U, \mathbf{s}^E, \mathbf{s}^U$ similarly. Let $P_1 := 2^{Q,k_1} \times 2^{Q,k_2}$ and $P_0 := \Sigma \times 2^{Q,k_1} \times 2^{Q,k_2}$. $\Gamma^{\mathrm{Sp}} \subseteq P_1 \times P_0$ models Spoiler's moves: $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Sp}}$ iff Spoiler chooses a as the next input symbol, and

- $\mathbf{q}'$ is obtained from $\mathbf{q}^E$ by choosing a successor for *each pebble* in $\mathbf{q}^E$. Formally, $\mathbf{q}' = \{ \operatorname{select}(\Delta(q, a)) \mid q \in \mathbf{q}^E \}$, where $\operatorname{select}(\mathbf{r})$ chooses an element in $\mathbf{r}$.
- Similarly, $\mathbf{s}'$ is obtained from $\mathbf{s}^U$ by choosing a successor for each pebble in $\mathbf{s}^U$.

Duplicator's moves are of the form $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}', \mathbf{s}', \mathbf{q}'', \mathbf{s}'') \in \Gamma^{\mathrm{Dup}} \subseteq P_1 \times P_0 \times P_1$:

- $\mathbf{q}''$ is a non-empty k_1-subset of $\mathbf{q}' \cup \Delta(\mathbf{q}^U, a)$, and
- $\mathbf{s}''$ is a non-empty k_2-subset of $\mathbf{s}' \cup \Delta(\mathbf{s}^E, a)$.

Notice that Duplicator is always allowed to "take pebbles away", and to "hedge her bets" by splitting pebbles into different successors. We say that a pebble on state q is *stuck* if q has no a-successor (where a is clear from the context).

We now formally define strategies. A strategy for Spoiler is a function $\delta : P_1^* P_1 \mapsto P_0$ *compatible with* Γ^{Sp}, i.e., for any $(\pi \cdot \langle \mathbf{q}, \mathbf{s} \rangle) \in P_1^* P_1$, $\delta(\pi \cdot \langle \mathbf{q}, \mathbf{s} \rangle) = (a, \mathbf{q}', \mathbf{s}')$ implies $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Sp}}$. Similarly, a strategy for Duplicator is a function $\sigma : P_1^* P_1 \mapsto (P_0 \mapsto P_1)$ *compatible with* Γ^{Dup}, i.e., for any $\pi \in P_1^* P_1$ and $(a, \mathbf{q}', \mathbf{s}') \in P_0$, $\sigma(\pi)(a, \mathbf{q}', \mathbf{s}') = \langle \mathbf{q}'', \mathbf{s}'' \rangle$ implies $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}', \mathbf{s}', \mathbf{q}'', \mathbf{s}'') \in \Gamma^{\mathrm{Dup}}$. A play $\pi = \langle \mathbf{q}_0, \mathbf{s}_0 \rangle \langle \mathbf{q}_1, \mathbf{s}_1 \rangle \cdots \in P_1^* \cup P_1^\omega$ is a finite or infinite sequence of configurations in P_1. For a word $w = a_0 a_1 \cdots \in \Sigma^* \cup \Sigma^\omega$ s.t. $|w| = |\pi| - 1$ (with $|\pi| = \omega = \omega - 1$ if $\pi \in \Sigma^\omega$), we say that a play π is σ*-conform to* w iff, for any $i < |\pi|$, there exists some $(\mathbf{q}_i, \mathbf{s}_i, a_i, \mathbf{q}_i', \mathbf{s}_i') \in \Gamma^{\mathrm{Sp}}$ s.t. $\sigma(\langle \mathbf{q}_0, \mathbf{s}_0 \rangle \ldots \langle \mathbf{q}_i, \mathbf{s}_i \rangle)(a_i, \mathbf{q}_i', \mathbf{s}_i') = \langle \mathbf{q}_{i+1}, \mathbf{s}_{i+1} \rangle$. Intuitively, σ-conform plays are those plays which originate when Duplicator's strategy is fixed to σ; δ-conform plays, for δ a Spoiler's strategy, are defined similarly. Below, both strategies are fixed, and the resulting, unique play is conform to both.

The game can halt prematurely, for pebbles may get stuck. In this case, the winning condition is as follows: If there exists a *Left* pebble which cannot be moved, then Duplicator wins. Dually, if no *Right* pebble can be moved, then Spoiler wins.

Remark 1. Our winning condition differs from the one in [4] when pebbles get stuck. There, the losing player is always the one who got stuck. If we let Duplicator win when Spoiler is stuck on a universal *Right* pebble, we would obtain a simulation which *does not imply* language containment. (Notice that "simulation implies containment" is proved in [4] under the assumption that pebbles do not get stuck.) Furthermore, the condition in [4] is unnecessarily strong when Duplicator is stuck on a universal *Left* pebble, where letting Spoiler win is too conservative. Our definition generalizes the correct winning condition to multiple pebbles, for which we prove "simulation implies containment" without further assumptions.

In all other cases, we have that all *Left* pebbles can be moved and at least one *Right* pebble can be moved, and the two players build an infinite sequence of configurations $\pi = \langle \mathbf{q}_0, \mathbf{s}_0 \rangle \langle \mathbf{q}_1, \mathbf{s}_1 \rangle \cdots \in P_1^\omega$. The winning condition is defined in terms of a predicate $C(\pi)$ on π. Different choices of $C(\pi)$ lead to different notions of simulation.

1. *Ordinary* (k_1, k_2)*-simulation*. The acceptance condition is ignored, and Duplicator wins as long as the game doesn't halt: $C(\pi) :\Longleftrightarrow \texttt{true}$.
2. *Existential direct* (k_1, k_2)*-simulation*. Duplicator wins if, whenever *every* $q \in \mathbf{q}_i$ is accepting, then *some* $s \in \mathbf{s}_i$ is accepting:
$$C(\pi) :\Longleftrightarrow (\forall i.\ \mathbf{q}_i \subseteq F \implies \mathbf{s}_i \cap F \neq \emptyset) \ .$$
3. *Universal direct* (k_1, k_2)*-simulation*. Duplicator wins if, whenever *some* $q \in \mathbf{q}_i$ is accepting, then *every* $s \in \mathbf{s}_i$ is accepting:
$$C(\pi) :\Longleftrightarrow (\forall i.\ \mathbf{q}_i \cap F \neq \emptyset \implies \mathbf{s}_i \subseteq F) \ .$$

As we will see, ordinary simulation is used for ATSs, while existential and universal direct simulation are used for automata over finite and infinite words, respectively.

The winning condition for delayed and fair simulation requires some technical preparation, which consists in the notion of being existentially/universally good since some previous round. Given the current round m, we say that a state $q \in \mathbf{q}_m$ *has seen* a state $\hat{q}$ since some previous round $i \leq m$, written $\mathrm{has\_seen}^i_m(q, \hat{q})$, iff either 1) $q = \hat{q}$, or $i < m$ and there exists $q' \in \mathbf{q}_{m-1}$ s.t. 2.1) $q \in \Delta(q', a_{m-1})$, and 2.2) $\mathrm{has\_seen}^i_{m-1}(q', \hat{q})$. Dually, we write $\mathrm{cant\_avoid}^i_m(q, \hat{q})$ iff either 1) $q = \hat{q}$, or $i < m$ and, for all $q' \in \mathbf{q}_{m-1}$, $q \in \Delta(q', a_{m-1})$ implies $\mathrm{cant\_avoid}^i_{m-1}(q', \hat{q})$. We overload the notation on the set of accepting states, and we write $\mathrm{has\_seen}^i_m(q, F)$ to mean that q has seen some $\hat{q} \in F$; and similarly for $\mathrm{cant\_avoid}^i_m(q, F)$. Finally, we say that $\mathbf{s}_j$ is *existentially good since round* $i \leq j$, written $\mathrm{good}^\exists(\mathbf{s}_j, i)$, if at round j every state in $\mathbf{s}_j$ has seen an accepting state since round i, and j is the least round for which this holds [2]. Similarly, we say that $\mathbf{q}_j$ is *universally good since round* $i \leq j$, written $\mathrm{good}^\forall(\mathbf{s}_j, i)$, if at round j every state in $\mathbf{q}_j$ cannot avoid an accepting state since round i, and j is the least round for which this holds. Formally,

$$
\begin{aligned}
\mathrm{good}^\exists(\mathbf{s}_j, i) \iff \quad & (\forall s \in \mathbf{s}_j.\ \mathrm{has\_seen}^i_j(s, F)) \quad \wedge \\
& \forall j'.\ (\forall s' \in \mathbf{s}_{j'}.\ \mathrm{has\_seen}^i_{j'}(s', F)) \implies j' \geq j \\
\mathrm{good}^\forall(\mathbf{s}_j, i) \iff \quad & (\forall s \in \mathbf{s}_j.\ \mathrm{cant\_avoid}^i_j(s, F)) \quad \wedge \\
& \forall j'.\ (\forall s' \in \mathbf{s}_{j'}.\ \mathrm{cant\_avoid}^i_{j'}(s', F)) \implies j' \geq j
\end{aligned}
$$

We write $\mathrm{good}^\exists(\mathbf{s}_j)$, omitting the second argument, when we just say that $\mathbf{s}_j$ is good since *some* previous round. For a path $\pi = \mathbf{s}_0\mathbf{s}_1 \ldots$, we write $\mathrm{good}^\exists(\pi, \infty)$, with the second argument instantiated to $i = \infty$, to mean that $\mathrm{good}^\exists(\mathbf{s}_j)$ holds for infinitely many j's; and similarly for $\mathrm{good}^\forall(\mathbf{s}_j)$ and $\mathrm{good}^\forall(\pi, \infty)$.

We are now ready to define delayed and fair simulations.

4. *Delayed* (k_1, k_2)*-simulation*. Duplicator wins if, whenever $\mathbf{q}_i$ is universally good, then there exists $j \geq i$ s.t. $\mathbf{s}_j$ is existentially good since round i:

$$C(\pi) :\iff \forall i.\ \mathrm{good}^\forall(\mathbf{q}_i) \implies \exists j \geq i.\ \mathrm{good}^\exists(\mathbf{s}_j, i)\ .$$

5. *Fair* (k_1, k_2)*-simulation*. Duplicator wins if, whenever there are infinitely many i's s.t. $\mathbf{q}_i$ is universally good, then, for any such i, there exists $j \geq i$ s.t. $\mathbf{s}_j$ is existentially good since round i:

$$C(\pi) :\iff \mathrm{good}^\forall(\pi_0, \infty) \implies (\forall i.\ \mathrm{good}^\forall(\mathbf{q}_i) \implies \exists j \geq i.\ \mathrm{good}^\exists(\mathbf{s}_j, i))\ ,$$

where $\pi_0 = \mathbf{q}_0\mathbf{q}_1 \ldots$ is the projection of π onto its first component.

We will denote the previous acceptance conditions with $x \in \{\mathrm{o}, \exists\mathrm{di}, \forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$, and the corresponding game is denoted as $\mathbb{G}^x_{(k_1,k_2)}(\mathbf{q_0}, \mathbf{s_0})$.

Remark 2. Notice that the condition for fair simulation is equivalent to the following simpler one: If $\mathbf{q}_i$ is universally good since some previous round infinitely often, then $\mathbf{s}_i$ is existentially good since some previous round infinitely often: $C'(\pi) :\iff \mathrm{good}^\forall(\pi_0, \infty) \implies \mathrm{good}^\exists(\pi_1, \infty)$, where $\pi_1 = \mathbf{s}_0\mathbf{s}_1 \ldots$ is the projection of π onto its second component.

We are now ready to define the simulation relation $\sqsubseteq^x_{(k_1,k_2)}$, with x as above. We say that a k_2-set $\mathbf{s}$ x-simulates a k_1-set $\mathbf{q}$, written $\mathbf{q} \sqsubseteq^x_{(k_1,k_2)} \mathbf{s}$, if Duplicator has a winning strategy in $\mathbb{G}^x_{(k_1,k_2)}(\mathbf{q}, \mathbf{s})$. We overload the simulation relation $\sqsubseteq^x_{(k_1,k_2)}$ on singletons: $q \sqsubseteq^x_{(k_1,k_2)} s \iff \{q\} \sqsubseteq^x_{(k_1,k_2)} \{s\}$. For two automata $\mathcal{A}$ and $\mathcal{B}$, we write $\mathcal{A} \sqsubseteq^x_{(k_1,k_2)} \mathcal{B}$ for $q_I^{\mathcal{A}} \sqsubseteq^x_{(k_1,k_2)} q_I^{\mathcal{B}}$, where the simulation is actually computed on the disjoint union of $\mathcal{A}$ and $\mathcal{B}$. If $\sqsubseteq^x_{(k_1,k_2)}$ is a simulation, then its transitive closure is defined as $\preceq^x_{(k_1,k_2)}$. Note that, in general, $\sqsubseteq^x_{(k_1,k_2)}$ is not itself a transitive relation.

Multipebble simulations hierarchy. In general, having more pebbles (possibly) gives more power to the Duplicator. This is similar to the $(1,k)$-simulations for NBA studied in [2], but in our context there are two independent directions of "growing power".

Theorem 1. *Let* $x \in \{\mathrm{o}, \exists\mathrm{di}, \forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$ *and* $k_1' \geq k_1$, $k_2' \geq k_2$.

1. *Inclusion:* $\sqsubseteq^x_{(k_1,k_2)} \subseteq \sqsubseteq^x_{(k_1',k_2')}$. *(In particular,* $\preceq^x_{(k_1,k_2)} \subseteq \preceq^x_{(k_1',k_2')}$.*)*
2. *Strictness: If* $k_1' > k_1$ *or* $k_2' > k_2$*, there exists an automaton* $\mathcal{Q}'$ *s.t.* $\sqsubseteq^x_{(k_1,k_2)} \neq \sqsubseteq^x_{(k_1',k_2')}$.

Proof (Sketch). Point 1) follows directly from the definitions, since Duplicator can always take pebbles away. Point 2) is illustrated in Figure 1, which holds for any kind of simulation $x \in \{\mathrm{o}, \exists\mathrm{di}, \forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. □

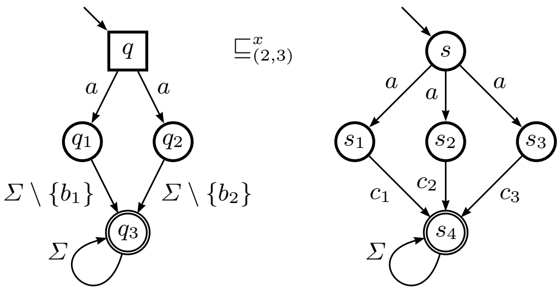


Fig. 1. Example in which $q \sqsubseteq^x_{(2,3)} s$, but $q \not\sqsubseteq^x_{(k_1,k_2)} s$ for any $k_1 \leq 2, k_2 \leq 3$, with $k_1 < 2$ or $k_2 < 3$. The alphabet is $\Sigma' = \{a\} \cup \Sigma$, with $\Sigma = \{b_1, b_2, c_1, c_2, c_3\}$. Note that both automata recognize the same language, both over finite and infinite words: $\mathcal{L}^{\mathrm{fin}}(q) = \mathcal{L}^{\mathrm{fin}}(s) = a(c_1 + c_2 + c_3)\Sigma^*$ and $\mathcal{L}^{\omega}(q) = \mathcal{L}^{\omega}(s) = a(c_1 + c_2 + c_3)\Sigma^{\omega}$.

Theorem 2. *For any* $k_1, k_2 \in \mathbb{N}_{>0}$ *and any automaton* $\mathcal{Q}$,

1. $\sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} \subseteq \sqsubseteq^{\mathrm{o}}_{(k_1,k_2)}$
2. $\sqsubseteq^{\forall\mathrm{di}}_{(k_1,k_2)} \subseteq \sqsubseteq^{\mathrm{de}}_{(k_1,k_2)} \subseteq \sqsubseteq^{\mathrm{f}}_{(k_1,k_2)} \subseteq \sqsubseteq^{\mathrm{o}}_{(k_1,k_2)}$.

Moreover, for each containment, there exists $\mathcal{Q}$ *s.t. the containment is strict.*

Proof. The containments follow directly from the definitions. For the strictness, consider again the example in Figure 1, with the modifications below. If no state on the right is accepting, then no simulation holds except ordinary simulation. If q is accepting, then universal direct simulation does not hold, but delayed simulation does. Finally, if the only accepting state is q, then delayed simulation does not hold, but fair simulation does. Is is easy to generalize this example for any $k_1, k_2 \in \mathbb{N}_{>0}$. □

3 Finite words

Lemma 1. *For any automaton $\mathcal{Q}$ with n states and states $q, s \in Q$:*

1. $q \sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} s$ *implies* $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$*, for any* $k_1, k_2 \in \mathbb{N}_{>0}$*.*
2. $q \sqsubseteq^{\mathrm{o}}_{(k_1,k_2)} s$ *implies* $\mathcal{T}r(q) \subseteq \mathcal{T}r(s)$*, for any* $k_1, k_2 \in \mathbb{N}_{>0}$*.*
3. $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$ *implies* $q \sqsubseteq^{\exists\mathrm{di}}_{(n,n)} s$*, provided that* $\mathcal{Q}$ *is complete.*
4. $\mathcal{T}r(q) \subseteq \mathcal{T}r(s)$ *implies* $q \sqsubseteq^{\mathrm{o}}_{(n,n)} s$*, provided that* $\mathcal{Q}$ *is complete.*

In particular, the last two points above show that existential-direct (resp., ordinary) simulation "reaches" language inclusion (resp., trace inclusion) at (n, n)*.*

Subset constructions. The subset construction is a well-known procedure for determinizing NFAs [7]. It is not difficult to generalize it over *alternating* automata, where it can be used for eliminating existential states, i.e., to perform the *de-existentialization* of the automaton. The idea is the same as in the subset construction, except that, when considering a-successors of a macrostate (for a symbol $a \in \Sigma$), existential and universal states are treated differently. For existential states, we apply the same procedure as in the classic subset construction, by taking always all a-successors. For universal states, each a-successor induces a different transition in the subset automaton. This ensures that macrostates can be interpreted purely disjunctively, and the language of a macrostate equals the union over the language of the states belonging to it. Accordingly, a macrostate is accepting if it contains *some* state which is accepting.

The previous construction can be dualized for de-universalizing finite automata. For an AFA $\mathcal{Q}$, let $\mathcal{S}^{\exists}(\mathcal{Q})$ and $\mathcal{S}^{\forall}(\mathcal{Q})$ be its de-existentialization and de-universalization, respectively. (See Definitions 1 and 2 in Appendix B.1.)

The following lemma formalizes the intuition that multipebble simulations for AFA in fact correspond to $(1, 1)$-simulations over the appropriate subset-constructions.

Lemma 2. *Let* $\mathcal{Q}_1, \mathcal{Q}_2$ *be two AFAs over the same alphabet* Σ*, with* $|Q_1| = n_1$ *and* $|Q_2| = n_2$*. Then, for any* $k_1 \leq n_1$ *and* $k_2 \leq n_2$*,*

$$\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(k_1,n_2)} \mathcal{Q}_2 \iff \mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(k_1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2) \tag{1}$$

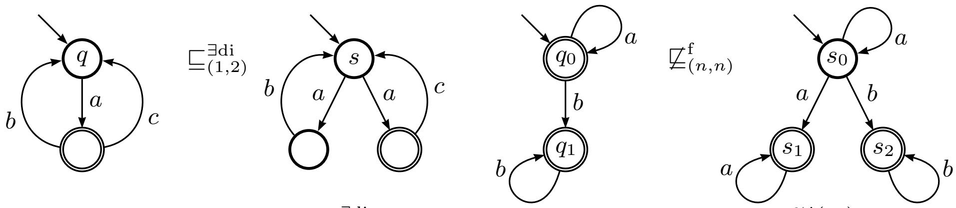
$$\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(n_1,k_2)} \mathcal{Q}_2 \iff \mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,k_2)} \mathcal{Q}_2 \tag{2}$$

$$\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(n_1,n_2)} \mathcal{Q}_2 \iff \mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2) \; . \tag{3}$$

4 Infinite words

Multipebble existential-direct simulation is not suitable for being used for ω-automata, since it does not even imply ω-language inclusion.

Theorem 3. *For any* $k_1, k_2 \in \mathbb{N}_{>0}$*, not both equal to* 1*, there exist an automaton* $\mathcal{Q}$ *and states* $q, s \in \mathcal{Q}$ *s.t.* $q \sqsubseteq^{\exists di}_{(k_1,k_2)} s$ *holds, but* $\mathcal{L}^{\omega}(q) \not\subseteq \mathcal{L}^{\omega}(s)$*.*



(a) An example in which $q \sqsubseteq^{\exists\mathrm{di}}_{(1,2)} s$ holds, but $\mathcal{L}^\omega(q) \not\subseteq \mathcal{L}^\omega(s)$.

(b) An example in which $\mathcal{L}^\omega(q_0) \subseteq \mathcal{L}^\omega(s_0)$ holds, but $q_0 \not\sqsubseteq^{\mathrm{f}}_{(n,n)} s_0$.

Fig. 2. Two examples.

Proof. Consider the example in Figure 2(a). Clearly, $q \sqsubseteq^{\exists\mathrm{di}}_{(1,2)} s$ holds, since Duplicator can split pebbles on the successors of s, and one such pebble is accepting, as required by existential-direct simulation. But $\mathcal{L}^\omega(q) \not\subseteq \mathcal{L}^\omega(s)$: In fact, $(ab)^\omega \in \mathcal{L}^\omega(q) = (a(b+c))^\omega$, but $(ab)^\omega \notin \mathcal{L}^\omega(s) = ((ab)^* ac)^\omega$. $\square$

This motivates the definition of *universal*-direct simulation, which *does imply* ω-*language inclusion*, like the coarser delayed and fair simulations.

Theorem 4. *For* $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$*, automaton* $\mathcal{Q}$*,* $k_1, k_2 \in \mathbb{N}_{>0}$ *and states* $q, s \in \mathcal{Q}$*,*

$$q \sqsubseteq^x_{(k_1,k_2)} s \quad \textit{implies} \quad \mathcal{L}^\omega(q) \subseteq \mathcal{L}^\omega(s) \; .$$

Unlike in the finite word case, ω-language inclusion is not "reached" by the simulations $\{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. See Figure 2(b) and Appendix C.

Theorem 5. *For any* $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$*, there exist an automaton* $\mathcal{Q}$ *and states* $q_0, s_0 \in \mathcal{Q}$ *s.t.* $\mathcal{L}^\omega(q_0) \subseteq \mathcal{L}^\omega(s_0)$*, but* $q_0 \not\sqsubseteq^x_{(n,n)} s_0$.

The Miyano-Hayashi construction The Miyano-Hayashi (MH) construction [10] is a subset-like construction for ABAs which removes universal non-determinism, i.e., it performs the *de-universalization* of ω-automata. The idea is similar to the analogous construction over finite words, with extra bookkeeping needed for recording visits to accepting states, which may occur not simultaneously for different runs. A set of obligations is maintained, encoding the requirement that, independently of how universal non-determinism is resolved, an accepting state has to be eventually reached. There is a tight relationship between these obligations and fair multipebble simulation. For an ABA $\mathcal{Q}$, let $\mathcal{Q}_{\mathrm{nd}}$ be the de-universalized automaton obtained by applying the MH-construction. (See also Definition 3 in Appendix C.1.)

The following lemma says that the MH-construction produces an automaton which is $(n,1)$-fair-simulation equivalent to the original one, and this result is "tight" in the sense that it does not hold for either direct, or delayed simulation.

Lemma 3. *For any ABA* $\mathcal{Q}$*, let* $\mathcal{Q}_{nd}$ *be the NBA obtained according to the Miyano-Hayashi de-universalization procedure applied to* $\mathcal{Q}$*. Then,*

a) $\mathcal{Q} \sqsubseteq^x_{(n,1)} \mathcal{Q}_{nd}$*, for* $x \in \{\mathrm{f}, \forall\mathrm{di}\}$*, and a') ∃ automaton* $\mathcal{Q}^1$ *s.t.* $\mathcal{Q}^1 \not\sqsubseteq^{\mathrm{de}}_{(n,1)} \mathcal{Q}^1_{nd}$*,*
b) $\mathcal{Q}_{nd} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{Q}$*, and b') ∃ automaton* $\mathcal{Q}^2$ *s.t.* $\mathcal{Q}^2_{nd} \not\sqsubseteq^x_{(1,1)} \mathcal{Q}^2$*, for* $x \in \{\mathrm{de}, \forall\mathrm{di}\}$.

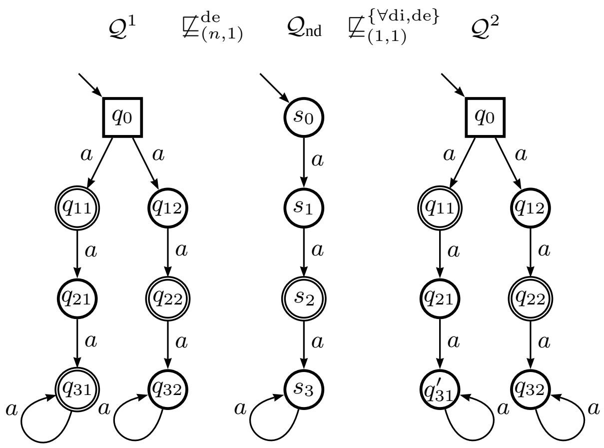


Fig. 3. An example showing automata $\mathcal{Q}^1$ and $\mathcal{Q}^2$ s.t. $\mathcal{Q}^1 \not\sqsubseteq^{\mathrm{de}}_{(n,1)} \mathcal{Q}_{\text{nd}}$ ($n = 2$ suffices), and $\mathcal{Q}_{\text{nd}} \not\sqsubseteq^{x}_{(1,1)} \mathcal{Q}^2$ for $x \in \{\forall\mathrm{di}, \mathrm{de}\}$. The only difference between $\mathcal{Q}^1$ and $\mathcal{Q}^2$ is the state q_{31} being accepting in the former and q'_{31} being non-accepting in the latter. Notice that $\mathcal{Q}^1_{\text{nd}} = \mathcal{Q}^2_{\text{nd}} = \mathcal{Q}_{\text{nd}}$. The states in $\mathcal{Q}_{\text{nd}}$ are: $s_0 = (\{q_0\}, \{q_0\})$, $s_1 = (\{q_{11}, q_{12}\}, \{q_{12}\})$, $s_2 = (\{q_{21}, q_{22}\}, \emptyset)$, $s_3 = (\{q_{31}, q_{32}\}, \{q_{32}\})$.

Since fair simulation implies language inclusion, $\mathcal{Q}$ and $\mathcal{Q}_{\text{nd}}$ have the same language. This constitutes an alternative proof of correctness for the MH-construction.

The MH-construction "preserves" fair simulation in the following sense.

Lemma 4. *Let $\mathcal{Q}, \mathcal{S}$ be two ABAs. Then, $\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S} \iff \mathcal{Q}_{nd} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S}_{nd}$.*

Remark 3. A weaker version of the "only if" direction of Lemma 4 above, namely $\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S} \implies \mathcal{Q}_{\text{nd}} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S}_{\text{nd}}$ (notice the $(1,1)$ in the premise), had already appeared in [4]. The same statement for both direct and delayed simulation is false, unlike as incorrectly claimed in [4]. In fact, it can be shown (with an example similar to Figure 3) that there exist automata $\mathcal{Q}$ and $\mathcal{S}$ s.t. $\mathcal{Q} \sqsubseteq^{x}_{(1,1)} \mathcal{S}$, but $\mathcal{Q}_{\text{nd}} \not\sqsubseteq^{x}_{(1,1)} \mathcal{S}_{\text{nd}}$, with $x \in \{\mathrm{di}, \mathrm{de}\}$. Finally, the "if" direction of Lemma 4 can only be established in the context of multiple pebbles, and it is new.

Transitivity. While most (k_1, k_2)-simulations are not transitive, some limit cases are. By defining a notion of join for $(1,n)$- and $(n,1)$-strategies (see Appendix C.2), we establish that $(1,n)$ and $(n,1)$ simulations are transitive.

Theorem 6. *Let $\mathcal{Q}$ be an ABA with n states, and let $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. Then, $\sqsubseteq^{x}_{(1,n)}$ and $\sqsubseteq^{x}_{(n,1)}$ are transitive.*

Remark 4 (Difficulties for (n,n) transitivity.). We did consider transitivity for (n,n)-simulations on ABA, but found two major issues there. The first issue concerns directly the definition of the join of two (n,n)-strategies, and this holds for any $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$: The so-called "puppeteering technique", currently used for defining the join for $(1,n)$- and $(n,1)$-strategies, requires to maintain several games, and to pipe the output from one game to the input of one or more other games. This creates a notion of dependency between different games. For $(1,n)$ and $(n,1)$, there are no cyclic dependencies, and we were able to define the joint strategy. However, for (n,n)-simulations, there are cyclic dependencies, and it is not clear how the joint strategy should be defined.

The second issue arises from the fact that we further require that the join of two winning strategies is itself a winning strategy. Therefore, the joint strategy needs to carry an invariant which implies the x-winning condition, for $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. While such an invariant for $x = \forall\mathrm{di}$ is straightforward, it is not clear what the correct invariant should be for either delayed or fair simulation.

5 Quotienting

In the following we discuss how multipebble simulation preorders can be used for state-space reduction of alternating automata, i.e., we discuss under which notions of quotient the quotient automaton recognizes the same language as the original one.

Let $\mathcal{Q} = (Q, \Sigma, q_I, \Delta, E, U, F)$ be an alternating automaton, over finite or infinite words. Let $\preceq$ be any binary relation on Q, and let $\approx$ be the induced equivalence, defined as $\approx = \preceq^* \cap (\preceq^*)^{-1}$. $[\cdot] : Q \mapsto [Q]$ is the function that maps each element $q \in Q$ to the equivalence class $[q] \in [Q]$ it belongs to, i.e., $[q] := \{q' \mid q \approx q'\}$. We overload $[P]$ on sets $P \subseteq Q$ by taking the set of equivalence classes.

In all the notions of quotients that will be defined, only the transition relation varies. Thus, we gather the common part under a quotient skeleton. We define the *quotient skeleton* $\mathcal{Q}_\approx = ([Q], \Sigma, [q_I], \Delta_\approx, E', U', F')$ as follows: $E' := [E]$, $U' := [Q] \setminus E' = \{\ [q] \mid [q] \subseteq U\ \}$ and $F' = [F]$. We leave $\Delta_\approx$ unspecified at this time, as it will have different concrete instantiations later. Notice that mixed classes, i.e., classes containing both existential and universal states, are declared existential.

The following definitions are borrowed from [4]. We say that $q' \in \Delta(q, a)$ is a *k-x-minimal a-successor of q* iff there there is no strictly $\sqsubseteq^x_{(1,k)}$-smaller a-successor of q, i.e., for any $q'' \in \Delta(q, a)$, $q'' \sqsubseteq^x_{(1,k)} q'$ implies $q' \sqsubseteq^x_{(1,k)} q''$. Similarly, $q' \in \Delta(q, a)$ is a *k-x-maximal a-successor of q* iff for any $q'' \in \Delta(q, a)$, $q' \sqsubseteq^x_{(1,k)} q''$ implies $q'' \sqsubseteq^x_{(1,k)} q'$. Let $\min^{k,x}_a(q)$/$\max^{k,x}_a(q)$ be the set of minimal/maximal successors.

5.1 Finite words

Let $\preceq$ be *any* preorder which implies language inclusion over finite words, i.e., $q \preceq s \implies \mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$. In particular, one can take $\preceq = (\sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)})^*$, or even $\preceq$ equal to language inclusion itself. As before, let $\approx$ be the equivalence induced by $\preceq$. It is well known that automata over finite words can be quotiented w.r.t. any preorder which implies language equivalence. Here, we show that not all transitions are needed, and that is is sufficient to consider $\preceq$-maximal successors of existential states and $\preceq$-minimal successors of universal states. We define the *minimax* [4] quotient automaton $\mathcal{Q}^{\mathrm{m}}_\approx$ by instantiating the quotient skeleton (see Section 5) with transition relation $\Delta_\approx := \Delta^{\mathrm{m}}_\approx$, where $([q], a, [q']) \in \Delta^{\mathrm{m}}_\approx$ iff either

- $[q] \in E'$ and $\exists\, \widehat{q} \in [q] \cap E, \widehat{q}' \in [q']$ s.t. $(\widehat{q}, a, \widehat{q}') \in \Delta \wedge \widehat{q}' \in \max^{\preceq}_a(\widehat{q})$, or
- $[q] \in U'$ and $\exists\, \widehat{q} \in [q], \widehat{q}' \in [q']$ s.t. $(\widehat{q}, a, \widehat{q}') \in \Delta$ and $\widehat{q}' \in \min^{\preceq}_a(\widehat{q})$.

Notice that transitions from universal states in mixed classes are ignored altogether.

Lemma 5. *Let $\mathcal{Q}$ be any alternating finite automaton, and let $\preceq$ be any preorder which implies finite-language inclusion. Then, for any $q \in Q$, $\mathcal{L}^{\text{fin}}(q) = \mathcal{L}^{\text{fin}}([q]_m)$.*

5.2 Infinite words

Unlike for finite words, it is well known that quotienting ω-automata w.r.t. ω-language-equivalence does not preserve the ω-language. It has even been shown that quotienting w.r.t. $(1,1)$-fair (bi)simulation does not preserve the ω-language either [6, 3]. Therefore, one has to look for finer simulations, like delayed or direct simulation. Notice that multipebble *existential*-direct simulation cannot be used for quotienting, since it does not even imply ω-language inclusion—see Theorem 3.

Theorem 7. *For any $k_1, k_2 \in \mathbb{N}^{>0}$ and $x \in \{\exists\text{di},\ f\}$ there exists an ABA $\mathcal{Q}$ s.t. $\mathcal{L}^\omega(\mathcal{Q}) \neq \mathcal{L}^\omega(\mathcal{Q}_\approx)$, with $\approx := \approx^x_{(k_1,k_2)}$. For $x = \exists\text{di}$, k_1 and k_2 must not be both equal to 1. (Note that $\approx^{\exists\text{di}}_{(1,1)}$-quotienting* does preserve *the ω-language.)*

Thus, in the following we concentrate on *universal*-direct and delayed simulation.

Minimax quotients for universal-direct simulation. In [4] it has been shown that minimax quotients preserve the ω-language (for direct simulation), and that one can consider just maximal/minimal successors of existential/universal states, respectively. Here, we improve this notion, by showing that, when considering multiple-pebbles, it is not needed to consider *every* maximal successor of existential states, but it is safe to discard those maximal successors which are $(1,k)$-simulated by a k-*set* of other maximal successors. This suggests the following definition: For $\widehat{q} \in E$, $a \in \Sigma$ and $k > 0$, we say that $\widehat{\mathbf{q}}'$ is a set of *k-maximal representatives for a-successors of $\widehat{q}$* iff

$$\widehat{\mathbf{q}}' \subseteq \max_a^{k,\forall\text{di}}(\widehat{q}) \wedge \left(\forall q'' \in \left(\max_a^{k,\forall\text{di}}(\widehat{q}) \setminus \widehat{\mathbf{q}}'\right) . \exists \widehat{\mathbf{q}}'' \in 2^{\widehat{\mathbf{q}}',k} .\ q'' \sqsubseteq^{\forall\text{di}}_{(1,k)} \widehat{\mathbf{q}}''\right) \quad (4)$$

Notice that the above definition is non-deterministic, in the sense that there might be different sets of maximal representatives: In this case, one can just take any $\subseteq$-minimal set satisfying Equation 4. In the following, we assume that a set of maximal representatives $\widehat{\mathbf{q}}'$ has been selected for any $\widehat{q} \in E$ and $a \in \Sigma$.

We define the *minimax+* quotient automaton $\mathcal{Q}^{\text{m+}}_\approx$ by instantiating the quotient skeleton (see Section 5) with transition relation $\Delta_\approx := \Delta^{\text{m+}}_\approx$, which differs from $\Delta^{\text{m}}_\approx$ just for existential and mixed classes: $([q], a, [q']) \in \Delta^{\text{m+}}_\approx$ with $[q] \in E'$ iff

– there exist $\widehat{q} \in [q] \cap E$ and $\widehat{q}' \in [q']$ s.t. $(\widehat{q}, a, \widehat{q}') \in \Delta$ and $\widehat{q}' \in \widehat{\mathbf{q}}'$, where $\widehat{\mathbf{q}}'$ is a fixed set of k-maximal representatives for a-successors of $\widehat{q}$, as defined above.

Our definition of minimax+ quotient differs from the one in [4] also w.r.t. the treatment of mixed classes, as discussed in the following remarks.

Remark 5. While in [4] universal states in mixed classes do induce transitions (to minimal elements), in our definition we ignore these transitions altogether. In the setting of $(1,1)$-simulations these two definitions coincide, as they are shown in [4] to yield exactly the same transitions, but this needs not be the case in our setting: In the context of multiple-pebbles, one minimal transition from a universal state q^U might be subsumed by no single transition from some existential state q^E in the same class, but it is always the case that q^E has a set of transitions which *together* subsume the one from q^U (cf. Lemma 15 in Appendix D.3). In this case, we show that one can in fact always discard the transitions from q^U. Thus, in the context of multiple-pebbles, minimax+ quotients result in less transitions than just minimax quotients from [4].

Remark 6. While minimax mixed classes are deterministic when considering $(1,1)$-simulations [4], this is not necessarily true when multiple pebbles are used.

Theorem 8. *$q \approx^{\forall\mathrm{di}}_{(1,n)} [q]_{\mathrm{m}+}$, where the quotient is taken w.r.t. the transitive closure of $\sqsubseteq^{\forall\mathrm{di}}_{(1,k)}$, for any k such that $1 \leq k \leq n$. In particular, $\mathcal{L}^\omega(q) = \mathcal{L}^\omega([q]_{\mathrm{m}+})$.*

Semielective quotients for delayed simulation. It has been shown in [4] that minimax quotients w.r.t $(1,1)$-delayed simulation on ABA do not preserve the ω-language. The reason is that taking just maximal successors of existential states is incorrect for delayed simulation, since a visit to an accepting state might only occur by performing a non-maximal transition. (This is not the case with direct simulation, where if a simulation-smaller state is accepting, then every bigger state is accepting too.) This motivates the definition of *semielective quotients* [4], which are like minimax quotients, with the only difference that *every* transition induced by existential states is considered, not just maximal ones. Except for that, all previous remarks still apply. In particular, in mixed classes in semielective quotients it is necessary to ignore non-minimal transitions from universal states—the quotient automaton would recognize a bigger language otherwise.

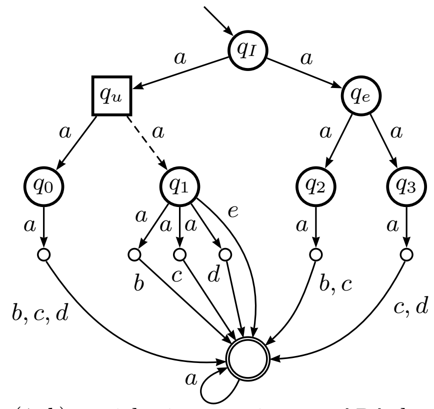
While for the $(1,1)$-simulations on ABA in [4] it is actually possible to ignore transitions from universal states in mixed classes altogether (see Remark 5), in the context of multiple-pebbles this is actually incorrect, as shown in Figure 5, Appendix D.3. The reason is similar as why non-maximal transitions from existential states cannot be discarded: This might prevent accepting states from being visited. We define the *semielective+* quotient automaton $\mathcal{Q}^{\mathrm{se+}}_{\approx}$ by instantiating the quotient skeleton (see Section 5) with $\Delta_\approx := \Delta^{\mathrm{se}+}_\approx$, where

$$([q], a, [q']) \in \Delta^{\mathrm{se}+}_\approx \iff (q, a, q') \in \Delta \text{ and either } q \in E, \text{ or } q \in U \text{ and } q' \in \min^{n,\mathrm{de}}_a(q)$$

Theorem 9. *$q \approx^{\mathrm{de}}_{(1,n)} [q]_{\mathrm{se}+}$, where the quotient is taken w.r.t. $\sqsubseteq^{\mathrm{de}}_{(1,n)}$. In particular, $\mathcal{L}^\omega(q) = \mathcal{L}^\omega([q]_{\mathrm{se}+})$.*

Remark 7. It is surprising that, unlike for NBA [2], quotienting ABA w.r.t. $(1,k)$-de simulations, for $1 < k < n$, does not preserve the language of the automaton in general. The problem is again in the mixed classes, where minimal transitions from universal states can be selected only by looking at the full $(1,n)$-simulation. See the counterexample in Figure 4, where the dashed transition is present in the $(1,k)$-quotient, despite being non-$(1,n)$-minimal.

Remark 8. Semielective multipebble quotients can achieve arbitrarily high compression ratios relative to semielective 1-pebble quotients, (multipebble-)direct minimax quotients and mediated preorder quotients [1] (see Figure 6 in Appendix D.3).



$q_u \approx^{\mathrm{de}}_{(1,2)} q_e$
$q_0 \sqsubseteq^{\mathrm{de}}_{(1,3)} q_1$
$q_0 \not\sqsubseteq^{\mathrm{de}}_{(1,2)} q_1$
$q_1 \not\sqsubseteq^{\mathrm{de}}_{(1,3)} q_0$

Fig. 4. $(1, k)$-semielective+ quotients on ABA do not preserve the ω-language for $1 < k < n$ in general. Let $k = 2$. The only two $(1, k)$/$(1, n)$-equivalent states are q_u and q_e, and in the quotient they form a mixed class. q_1 *is not* a $(1, n)$-minimal a-successor of q_u, but it is a $(1, k)$-minimal successor for $k = 2$. Thus, the only difference between the $(1, n)$- and $(1, k)$-semielective+ quotients is that the dashed transition is (correctly) not included in the former, but (incorrectly) included in the latter. Thus the $(1, k)$-semielective+ quotient automaton would incorrectly accept the word $w = aaea^\omega \notin \mathcal{L}^\omega(q_I) = aaa\{b + c + d\}a^\omega$.

6 Solving Multipebble Simulation Games

In this section we show how to solve the multipebble simulation games previously defined. We encode each simulation game into a 2-player game-graph with an ω-regular winning condition. In the game-graph, Eve will take the rôle of Duplicator, and Adam the one of Spoiler. A game-graph is a tuple $\mathcal{G} = \langle V_\mathrm{E}, V_\mathrm{A}, \rightarrow \rangle$, where nodes in V_E belong to Eve (mimicking Duplicator), and nodes in V_A belong to Adam (mimicking Spoiler). Transitions are represented by elements in $\rightarrow \subseteq (V_\mathrm{E} \times V_\mathrm{A} \cup V_\mathrm{A} \times V_\mathrm{E})$, where we write $p \rightarrow q$ for $(p, q) \in \rightarrow$. Notice that the two players strictly alternate while playing, i.e., the game graph is *bipartite*. We write V for $V_\mathrm{E} \cup V_\mathrm{A}$. We introduce the following monotone operator on 2^{V_A}: For any $\mathbf{x} \subseteq V_\mathrm{A}$, $\mathrm{cpre}(\mathbf{x}) := \{v_0 \in V_\mathrm{A} \mid \forall v_1 \in V_\mathrm{E}.\ (v_0 \rightarrow v_1 \implies \exists v_2 \in \mathbf{x}.\ v_1 \rightarrow v_2)\}$, i.e., $\mathrm{cpre}(\mathbf{x})$ is the set of nodes where Eve can force the game into $\mathbf{x}$.

We define various game-graphs for solving simulations. We express the winning region of Eve as a μ-calculus fixpoint expression over V_A [8], which can then be evaluated using standard fixpoint algorithms. We derive the desired complexity upper bounds using the following fact:

Lemma 6. *Let e be a fixpoint expression over a graph V, with $|V| \in n^{O(k)}$. Then, for any fixed $k \in \mathbb{N}$, evaluating e can be done in time polynomial in n.*

For solving direct and fair simulation, we refer the reader to Appendix E. Here, we consider just delayed simulation, which is the most difficult (and interesting).

The natural starting point for defining $\mathcal{G}^{\mathrm{de}}$ is the definition in [2] of the game-graph for computing $(1, k)$-simulations for NBAs. Unfortunately, the game-graph in [2] is actually incorrect: According to the definition of delayed simulation (cf. Section 2), every new obligation encountered when the left side is accepting at some round should be *independently* satisfied by the right side, which has to be good since *that* round. Now, the algorithm in [2] just tries to satisfy the most recent obligation, which overrides all the previous ones. This is an issue: If the left side is continuously accepting, for example, then the right side might simply have not enough time to satisfy any obligation at all. Therefore, [2] actually computes an *under-approximation* to delayed simulation.

We overcome this difficulty by explictly bookkeeping all pending constraints. This leads to the following definitions. The game-graph for delayed simulation is $\mathcal{G}^{\mathrm{de}} = \langle V_{\mathrm{E}}^{\mathrm{de}}, V_{\mathrm{A}}^{\mathrm{de}}, \rightarrow^{\mathrm{de}} \rangle$, where nodes in $V_{\mathrm{A}}^{\mathrm{de}}$ are of the form $v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good})}$, and nodes in $V_{\mathrm{E}}^{\mathrm{de}}$ of the form $v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good},a,\mathbf{q}',\mathbf{s}')}$, with $\mathbf{q}, \mathbf{q}', \mathbf{s}, \mathbf{s}' \subseteq Q$. $\mathsf{Bad} = \langle \mathbf{b}_1 \supset \cdots \supset \mathbf{b}_{m_1} \rangle$ and $\mathsf{Good} = \langle \mathbf{g}_1 \subset \cdots \subset \mathbf{g}_{m_2} \rangle$ are two *sequences* of sets of states from Q, strictly ordered by set-inclusion, which are used to keep track of multiple obligations.

Intuitively, Bad is used to detect when new constraints should be created, i.e., to detect when every *Left* pebble is universally good since some previous round. At each round, a new set of bad pebbles $\mathbf{b} = \mathbf{q} \setminus F$ is added to Bad. When accepting states are visited by *Left* pebbles, they are discarded from every set $\mathbf{b} \in \mathsf{Bad}$. When some $\mathbf{b}$ becomes eventually empty, this means that, at the current round, all *Left* pebbles are universally good since some previous round: At this point, $\mathbf{b}$ is removed from Bad, and we say that *the red light flashes*.

The sequence Good represents a set of constraints to be eventually satisfied. Each $\mathbf{g} \in \mathsf{Good}$ is a set of good pebbles, which we require to "grow" until it becomes equal to $\mathbf{s}$. When $\mathsf{Good} = \emptyset$, there is no pending constraint. Constraints are added to Good when the red light flashes (see above): In this case, we update Good by adding the new empty constraint $\mathbf{g} = \emptyset$. When accepting states are visited by *Right* pebbles, we upgrade every constraint $\mathbf{g} \in \mathsf{Good}$ by adding accepting states. Completed constraints $\mathbf{g} = \mathbf{s}$ are then removed from Good, and we say that *the green light flashes*.

Lemma 7. $\left|V^{\mathrm{de}}\right| \leq 2 \cdot (n+1)^{2(k_1+k_2)} \cdot \left(1 + (k_1+1)^{k_1+1}\right) \cdot \left(1 + 2(k_2+1)^{k_2+1}\right) \cdot |\Sigma|$.

Transitions in $\mathcal{G}^{\mathrm{de}}$ are defined as follows. For any $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}'', \mathbf{s}'') \in \Gamma^{\mathrm{Sp}}$, we have $v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good})} \rightarrow^{\mathrm{de}} v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good},a,\mathbf{q}'',\mathbf{s}'')}$, and for $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}'', \mathbf{s}'', \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Dup}}$, we have $v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good},a,\mathbf{q}'',\mathbf{s}'')} \rightarrow^{\mathrm{de}} v_{(\mathbf{q}',\mathsf{Bad}',\mathbf{s}',\mathsf{Good}')}$, where $\mathsf{Bad}', \mathsf{Good}'$ are computed according to Algorithm 1 in Appendix E.3.

We have that Eve wins iff every red flash is matched by at least one green flash, and different red flashes are matched by different green ones. This can be checked by verifying that infinitely often either $\mathsf{Good} = \emptyset$ or $\mathbf{s} \in \mathsf{Good}$, i.e., it is not the case that Good contains a constraint that it is not eventually "completed" and discarded. Let $T = \{v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good})} \mid \mathsf{Good} = \emptyset \vee \mathbf{s} \in \mathsf{Good}\}$, and define the initial configuration as

$$v_I = \begin{cases} v_{(\mathbf{q},\{\mathbf{q}\setminus F\},\mathbf{s},\emptyset)} & \text{if } \mathbf{q} \setminus F \neq \emptyset \\ v_{(\mathbf{q},\emptyset,\mathbf{s},\{\mathbf{s}\cap F\})} & \text{otherwise} \end{cases}$$

$\mathbf{q} \sqsubseteq^{\mathrm{de}}_{k_1,k_2} \mathbf{s}$ iff T is visited infinitely often iff $v_I \in W^{\mathrm{de}} = \nu\mathbf{x}\mu\mathbf{y}\,(\mathrm{cpre}(\mathbf{y}) \cup T \cap \mathrm{cpre}(\mathbf{x}))$.

Theorem 10. *For any fixed* $k_1, k_2 \in \mathbb{N}$, $x \in \{\forall\mathrm{di}, \exists\mathrm{di}, \mathrm{de}, \mathrm{f}\}$ *and sets* $\mathbf{q}, \mathbf{s} \subseteq Q$, *deciding whether* $\mathbf{q} \sqsubseteq^{x}_{(k_1,k_2)} \mathbf{s}$ *can be done in polynomial time.*

7 Conclusions and Future Work

Transitivity for (n,n)*-simulations.* As discussed at the end of Section 4, composing (n,n) (winning) strategies is apparently much more difficult than in the $(1,n)$ and $(n,1)$ case. We conjecture that all types of (n,n)-simulations discussed in this paper are transitive, and showing this would conceivably solve the join problem as well.

Quotienting with $(n, 1)$- *and* (n, n)*-simulations.* While we have dealt with $(1, n)$-quotients, we have not considered $(n, 1)$- or (n, n)-quotients. For the latter, one should first solve the associated transitivity problem, and, for both, an appropriate notion of semielective-quotient has to be provided. We have shown that this is already a non-trivial task for $(1, n)$-simulations on ABA.

Future directions. Our work on delayed simulation has shown that several generalizations are possible. In particular, two issues need to be addressed. The first is the complexity of the structure of the game-graph needed for computing delayed simulation. A possible generalization of delayed simulation involving looser "synchronization requirements" between obligations and their satisfaction might result in simpler game-graphs. The second issue concerns Lemmas 3 and 4: We would like to find a weaker delayed-like simulation for which the counterexample shown there does not hold. This would give a better understanding of the MH-construction.

As in [3], it is still open to find a hierarchy of (k_1, k_2)-multipebble simulations converging to ω-language inclusion when $k_1 = k_2 = n$.

Acknowledgment. We thank K. Etessami and C. Fritz for helpful discussions, and an anonymous referee for suggesting the comparison to mediated preorder [1].

References

1. P. A. Abdulla, Y.-F. Chen, L. Holik, and T. Vojnar. Mediating for reduction (on minimizing alternating Büchi automata). In *FSTTCS 2009*, volume 4 of *LIPIcs*, pages 1–12, Dagstuhl, Germany.
2. K. Etessami. A hierarchy of polynomial-time computable simulations for automata. In *CONCUR '02: Proceedings of the 13th International Conference on Concurrency Theory*, pages 131–144, London, UK, 2002. Springer-Verlag.
3. K. Etessami, T. Wilke, and R. A. Schuller. Fair simulation relations, parity games, and state space reduction for Büchi automata. *SIAM J. Comput.*, 34(5):1159–1175, 2005.
4. C. Fritz and T. Wilke. Simulation relations for alternating Büchi automata. *Theor. Comput. Sci.*, 338(1-3):275–314, 2005.
5. Y. Gurevich and L. Harrington. Trees, automata, and games. In *STOC '82: Proceedings of the fourteenth annual ACM symposium on Theory of computing*, pages 60–65, New York, NY, USA, 1982. ACM.
6. T. A. Henzinger and S. Rajamani. Fair bisimulation. In *TACAS '00: Proceedings of the 6th International Conference on Tools and Algorithms for Construction and Analysis of Systems*, pages 299–314. Springer-Verlag, 2000.
7. J. Hopcroft and J. Ullman. *Introduction to Automata Theory, Languages, and Computation*. Addison-Wesley, 1979.
8. D. Kozen. Results on the propositional mu-calculus. *Theor. Comput. Sci.*, 27:333–354, 1983.
9. Z. Manna and A. Pnueli. A hierarchy of temporal properties. Technical Report STAN-CD-87-1186, Dept. of Computer Sciences, Stanford University, Oct. 1987.
10. S. Miyano and T. Hayashi. Alternating finite automata on ω-words. *Theoretical Computer Science*, 32:321–330, 1984.
11. M. Y. Vardi. Alternating automata and program verification. In *Computer Science Today*, volume 1000 of *LNCS*, pages 471–485. Springer-Verlag, 1995.

A Preliminaries (Section 2)

Theorem 1. *Let $x \in \{\mathrm{o}, \exists\mathrm{di}, \forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$ and $k_1' \geq k_1$, $k_2' \geq k_2$.*

1. *Inclusion: $\sqsubseteq^x_{(k_1,k_2)} \subseteq \sqsubseteq^x_{(k_1',k_2')}$. (In particular, $\preceq^x_{(k_1,k_2)} \subseteq \preceq^x_{(k_1',k_2')}$.)*
2. *Strictness: If $k_1' > k_1$ or $k_2' > k_2$, there exists an automaton $\mathcal{Q}'$ s.t. $\sqsubseteq^x_{(k_1,k_2)} \neq \sqsubseteq^x_{(k_1',k_2')}$.*

Proof. 1. This follows directly from the definitions, since having more pebbles can only help Duplicator, who is always allowed to take pebbles away.

2. For showing the strictness of the inclusion, consider the example in Figure 1, for any kind of simulation $x \in \{\mathrm{o}, \exists\mathrm{di}, \forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. This example shows that Duplicator wins by "hedging her bets" on both sides, using 2 pebbles on the left and 3 pebbles on the right. Hence, $q \sqsubseteq^x_{(2,3)} s$ holds.
   To see that the 2 *Left* pebbles are necessary, it sufficies to note that if there were only one *Left* pebble, then Spoiler could choose either b_1 or b_2, and every *Right* pebble would get stuck. But with two *Left* pebbles, Spoiler can no longer play either b_1 or b_2, since one *Left* pebble would get stuck, which would be winning for Duplicator. A similar reasoning for the symbols $\{c_1, c_2, c_3\}$ shows that 3 *Right* pebbles are neccessary and sufficient for Duplicator to win the game.
   It is easy to generalize Figure 1 to more pebbles (k_1, k_2). Moreover, a similar example can be crafted without using the stuckness condition, but using only the acceptance condition.

□

Two following two observations will be useful in later proofs: When pebbles are good infinitely often, then it is the case that they are always good, as stated below.

Lemma 8. *Let $\pi = \mathbf{q}_0\mathbf{q}_1 \ldots$ be any sequence of k-sets. Then,*

$$\Big(\mathrm{good}^{\exists}(\pi, \infty) \implies \forall i \geq 0.\ \mathrm{good}^{\exists}(\mathbf{q}_i)\Big) \wedge \Big(\mathrm{good}^{\forall}(\pi, \infty) \implies \forall i \geq 0.\ \mathrm{good}^{\forall}(\mathbf{q}_i)\Big) .$$

The following is a consequence of König's Lemma:

Lemma 9. *Let $\pi = \mathbf{s}_0\mathbf{s}_1 \ldots$ be an infinite sequence of k-sets. If $\mathrm{good}^{\exists}(\pi, \infty)$, then there exists an infinite path $\pi_{acc} = p_0p_1 \ldots$ s.t. 1) for any i, $p_i \in \mathbf{s}_i$, and 2) for infinitely many i's, $p_i \in F$.*

Proof. We make use of König's Lemma: We build an infinite tree which is finitely branching, hence by König's Lemma there exists an infinite path π_{acc} starting from the root, and we show that this path contains infinitely many accepting states. First, we extract a subsequence $\{\mathbf{s}_j'\}_{j\geq 0}$ from $\{\mathbf{s}_i\}_{i\geq 0}$, as follows: $\mathbf{s}_j' := \mathbf{s}_{i_j}$, where $i_0 = 0$ and, inductively, i_j is the least index $i > i_{j-1}$ s.t. $\mathrm{good}^{\exists}(\mathbf{s}_i, i_{j-1})$. It follows that, for any $j > 0$, $\mathrm{good}^{\exists}(\mathbf{s}_j', i_{j-1})$. For any state $r \in \mathbf{s}_j'$ there is a node $v(r, j)$ at level $j \geq 0$ in the tree. (For example, the root of the tree is $v(s, 0)$, where s is the only state in $\mathbf{s}_0' = \mathbf{s}_0 = \{s\}$.) The parenthood relation between nodes is defined as follows: We have that, for any $r \in \mathbf{s}_{j+1}'$, there exists $r' \in \mathbf{s}_j'$ s.t. there exists a path from r' to r which visits at least one accepting state. In this case, $v(r, j+1)$ is a children of $v(r', j)$. This tree is infinite and finitely branching. Moreover, the infinite path π_{acc}, whose existence is guaranteed by König's Lemma, visits accepting states infinitely often. □

B Section 3

Lemma 1. *For any automaton $\mathcal{Q}$ with n states, and states $q, s \in Q$:*

1. $q \sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} s$ *implies* $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$*, for any* $k_1, k_2 \in \mathbb{N}_{>0}$*.*
2. $q \sqsubseteq^{\mathrm{o}}_{(k_1,k_2)} s$ *implies* $\mathcal{T}r(q) \subseteq \mathcal{T}r(s)$*, for any* $k_1, k_2 \in \mathbb{N}_{>0}$*.*
3. $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$ *implies* $q \sqsubseteq^{\exists\mathrm{di}}_{(n,n)} s$*, provided that* $\mathcal{Q}$ *is complete.*
4. $\mathcal{T}r(q) \subseteq \mathcal{T}r(s)$ *implies* $q \sqsubseteq^{\mathrm{o}}_{(n,n)} s$*, provided that* $\mathcal{Q}$ *is complete.*

In particular, the last two points above show that existential-direct (resp., ordinary) simulation "reaches" language inclusion (resp., trace inclusion) at (n, n)*.*

Proof. Point 2) follows from Point 1), and Point 4) follows from Point 3), since when the set of accepting state is the full set of states, i.e., $F = Q$, ordinary and direct simulation coincide, and the trace language equals the finite language in this case.

For Points 1) and 3), we defer their proof at the end of the next section. □

B.1 Subset constructions

Below, we give a formal definition for the de-existentialization and de-universalization procedures for AFAs.

Definition 1. *Given an AFA* $\mathcal{Q} = (Q, \Sigma, q_I, \Delta, F, E, U)$ *with* $|Q| = n$*, the existential* n*-subset construction yields a purely universal finite automata*

$$\mathcal{S}^{\exists}(\mathcal{Q}) := (Q', \Sigma, \{q_I\}, \Delta', F', E', U') ,$$

where $Q' := 2^Q$*,* $F' := \{P \in Q' \mid P \cap F \neq \emptyset\}$*,* $E' := \emptyset$*,* $U' := Q'$*, and the transition relation* $\Delta' \subseteq 2^Q \times \Sigma \times 2^Q$ *satisfies:* $(P, a, R) \in \Delta'$ *iff there exists a choice function* $\mathrm{select} : P \cap U \times \Sigma \mapsto Q$ *which fixes an element in* $\Delta(p, a)$ *for any universal state* $p \in P \cap U$*, and* $R = \bigcup_{p \in P \cap E} \Delta(p, a) \cup \{\mathrm{select}(p, a) \mid p \in P \cap U\}$*.*

Intuitively, the choice function select resolves the universal choice, and then we take the union over all possible resolutions of the existential choice.

Definition 2. *Given an AFA* $\mathcal{Q} = (Q, \Sigma, q_I, \Delta, F, E, U)$ *with* $|Q| = n$*, the universal* n*-subset construction yields a purely existential finite automata*

$$\mathcal{S}^{\forall}(\mathcal{Q}) := (Q', \Sigma, \{q_I\}, \Delta', F', E', U') ,$$

where $Q' := 2^Q$*,* $F' := \{P \in Q' \mid P \subseteq F\}$*,* $E' := Q'$*,* $U' := \emptyset$*, and* $(P, a, R) \in \Delta'$ *iff there exists a choice function* $\mathrm{select} : P \cap E \times \Sigma \mapsto Q$ *which fixes an element in* $\Delta(p, a)$ *for any existential state* $p \in P \cap E$*, and* $R = \bigcup_{p \in P \cap U} \Delta(p, a) \cup \{\mathrm{select}(p, a) \mid p \in P \cap E\}$*.*

Lemma 2. *Let* $\mathcal{Q}_1, \mathcal{Q}_2$ *be two AFAs over the same alphabet* Σ*, with* $|Q_1| = n_1$ *and* $|Q_2| = n_2$*. Then, for any* $k_1 \leq n_1$ *and* $k_2 \leq n_2$*,*

$$\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(k_1,n_2)} \mathcal{Q}_2 \iff \mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(k_1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2) \tag{S1}$$

$$\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(n_1,k_2)} \mathcal{Q}_2 \iff \mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,k_2)} \mathcal{Q}_2 \tag{S2}$$

$$\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(n_1,n_2)} \mathcal{Q}_2 \iff \mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2) . \tag{S3}$$

Proof. First notice that Equation (S3) follows by subsequent application of (S1) and (S2). For Equation (S1), the idea is that maximally splitting pebbles on existential states in $\mathcal{Q}_2$ is exactly the same as moving the only pebble in $\mathcal{S}^{\exists}(\mathcal{Q}_2)$. More formally, one can show how to mantain the following invariant: $\langle \mathbf{q}_1, \mathbf{q}_2 \rangle$ is the current configuration in the (k_1, n_2)-game on the left iff $\langle \mathbf{q}_1, \{\mathbf{q}_2\} \rangle$ is the current configuration in the $(k_1, 1)$-game on the right. From the invariant, the winning condition is easily verified: If $\mathbf{q}_1 \subseteq F$, then $\mathbf{q}_2 \cap F \neq \emptyset$, which is the same as saying $\mathbf{q}_2 \in F'$, where F' is the set of accepting states in $\mathcal{S}^{\exists}(\mathcal{Q}_2)$. Equation (S2) is proved similarly. □

Proof (of Lemma 1). We first prove Point 1), i.e., $q \sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} s$ implies $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$. Let $\mathcal{Q}_1$ and $\mathcal{Q}_2$ be two disjoint copies of $\mathcal{Q}$, where the initial states are, respectively, q and s. By definition, $q \sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} s$ iff $\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} \mathcal{Q}_2$. Since $\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(k_1,k_2)} \mathcal{Q}_2$ implies $\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(n_1,n_2)} \mathcal{Q}_2$, then by Equation (S3), one has $\mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2)$. But $\sqsubseteq^{\exists\mathrm{di}}_{(1,1)}$ is known to imply language inclusion [4], hence $\mathcal{L}^{\mathrm{fin}}(\mathcal{S}^{\forall}(\mathcal{Q}_1)) \subseteq \mathcal{L}^{\mathrm{fin}}(\mathcal{S}^{\exists}(\mathcal{Q}_2))$. Finally, since the subset constructions are language-preserving (this follows from their correctness), $\mathcal{L}^{\mathrm{fin}}(\mathcal{Q}_1) \subseteq \mathcal{L}^{\mathrm{fin}}(\mathcal{Q}_2)$, implying $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$.

For Point 3), the crucial observation is that $\mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2)$ is equivalent to language inclusion, since $\mathcal{S}^{\forall}(\mathcal{Q}_1)$ is a purely existential automaton and $\mathcal{S}^{\exists}(\mathcal{Q}_2)$ is a purely universal automaton, hence only Spoiler plays. Thus, $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}(s)$ implies $\mathcal{S}^{\forall}(\mathcal{Q}_1) \sqsubseteq^{\exists\mathrm{di}}_{(1,1)} \mathcal{S}^{\exists}(\mathcal{Q}_2)$, and, by Equation (S3), $\mathcal{Q}_1 \sqsubseteq^{\exists\mathrm{di}}_{(n,n)} \mathcal{Q}_2$, i.e., $q \sqsubseteq^{\exists\mathrm{di}}_{(n,n)} s$. □

C Section 4

Theorem 4. *For $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$, automaton $\mathcal{Q}$, $k_1, k_2 \in \mathbb{N}_{>0}$ and states $q, s \in \mathcal{Q}$,*

$$q \sqsubseteq^{x}_{(k_1,k_2)} s \quad \textit{implies} \quad \mathcal{L}^{\omega}(q) \subseteq \mathcal{L}^{\omega}(s) \ .$$

Proof. It suffices to prove the claim for $k_1 = k_2 = n = |Q|$, since, by Theorem 1, (n, n)-simulation contains (k_1, k_2)-simulation. Similarly, by the containment between universal-direct, delayed and fair simulation established in Theorem 2, it is sufficient to consider just fair simulation, which is the coarsest. Let $w = a_0 a_1 \cdots \in \mathcal{L}^{\omega}(q)$ be a word in $\mathcal{L}^{\omega}(q)$. We have to show $w \in \mathcal{L}^{\omega}(s)$. Let $\mathbb{G}^{\mathsf{sim}} = \mathbb{G}^{(n,n)}(q, s)$ be the simulation game between q and s, let $\mathbb{G}^{\mathsf{acc}}_0 = \mathbb{G}^{\omega}(q, w)$ be the acceptance game for w from state q, and let $\mathbb{G}^{\mathsf{acc}}_1 = \mathbb{G}^{\omega}(s, w)$ be the acceptance game for w from state s.

In order to show $w \in \mathcal{L}^{\omega}(s)$, we use the winning strategy of Duplicator in $\mathbb{G}^{\mathsf{sim}}$ and the information in $\mathbb{G}^{\mathsf{acc}}_0$ to witness the existence of a winning strategy for Automaton in $\mathbb{G}^{\mathsf{acc}}_1$. We use the so-called “puppeteering technique” to coordinate the various games. There are two “real players”, the Automaton (A_1) and Pathfinder (P_1) players in $\mathbb{G}^{\mathsf{acc}}_1$, and four “puppet players”, which are controlled by A_1:

– The Duplicator (D) and Spoiler (S) puppets in the simulation game $\mathbb{G}^{\mathsf{sim}}$.
– The Automaton (A_0) and Pathfinder (P_0) puppets in the acceptance game $\mathbb{G}^{\mathsf{acc}}_0$.

The orchestration job of A_1 is complicated by the fact that in the simulation game $\mathbb{G}^{\mathsf{sim}}$ multiple *Left* and *Right* pebbles may be present in any given round. Henceforth, A_1 mantains a family of acceptance games, depending on the current configuration of the simulation game. The flow of information between the various acceptance games and the simulation game is shown below.

$$
\begin{tikzcd}
\mathbb{G}^{\mathsf{acc}}_{0,E} \arrow[dr, "A_0" near start, "S_{\mathit{Left}}" near end] & & \mathbb{G}^{\mathsf{acc}}_{0,U} \\
& \mathbb{G}^{\mathsf{sim}} \arrow[ur, "D_{\mathit{Left}}" near start, "P_0" near end] \arrow[dr, "D_{\mathit{Right}}"' near start, "A_1"' near end] & \\
\mathbb{G}^{\mathsf{acc}}_{1,U} \arrow[ur, "P_1"' near start, "S_{\mathit{Right}}"' near end] & & \mathbb{G}^{\mathsf{acc}}_{1,E}
\end{tikzcd}
$$

The meaning of the picture is the following. Recall that in $\mathbb{G}^{\mathsf{sim}}$, Duplicator is allowed to "hedge her bets", i.e., to split pebbles, on universal *Left* states and on existential *Right* states. Every time D splits *Left* universal pebbles, a new acceptance game in $\mathbb{G}^{\mathsf{acc}}_0$ is spawned, and D's choices in $\mathbb{G}^{\mathsf{sim}}$ are mimicked by P_0 in $\mathbb{G}^{\mathsf{acc}}_0$. Similarly, when D splits *Right* existential pebbles, a new acceptance game in $\mathbb{G}^{\mathsf{acc}}_1$ is spawned, and D's choice in $\mathbb{G}^{\mathsf{sim}}$ is mimicked by A_1 in $\mathbb{G}^{\mathsf{acc}}_1$. This is represented by the r.h.s. of the figure above. Symmetrically, the other side represents S's behaviour, which moves *Left* existential pebbles copying A_0, and *Right* universal pebbles copying P_1.

The correctness is guaranteed by the fact that, since A_0 is playing a winning strategy in every $\mathbb{G}^{\mathsf{acc}}_0$ game, then regardless of universal choices in $\mathbb{G}^{\mathsf{acc}}_0$, the resulting run will visit accepting states infinitely often. Thus, by construction, *Left* pebbles in $\mathbb{G}^{\mathsf{sim}}$ are universally good infinitely often, and, since D is playing a winning strategy for fair simulation, *Right* pebbles are existentially good infinitely often. We won't explicitly define A_1's winning strategy for accepting w in $\mathbb{G}^{\mathsf{acc}}_1$, but, using Lemma 9 (which relies on König's Lemma) we will show that one such strategy does exist: Indeed, since *Right* pebbles are existentially good infinitely often, by Lemma 9, there exists an accepting run for w in $\mathbb{G}^{\mathsf{acc}}_1$, which witnesses the existence of a winning strategy for A_1.

For bookkeeping the state of the various simulation games, we use a *logbook*. Assume that, at round i, the current partial play in $\mathbb{G}^{\mathsf{sim}}$ is

$$\pi_i = \langle \mathbf{q}_0, \mathbf{s}_0 \rangle \dots \langle \mathbf{q}_i, \mathbf{s}_i \rangle \ ,$$

with $\mathbf{q}_0 = \{q\}$ and $\mathbf{s}_0 = \{s\}$, and that the remaining input word to be read is $w_i = a_i a_{i+1} \dots$. Then, a logbook $L_i = (L_i^0, L_i^1)$ for round i is a pair of finite sets of partial plays from $\mathbb{G}^{\mathsf{acc}}_0$ and $\mathbb{G}^{\mathsf{acc}}_1$, respectively, where $\left|L_i^0\right| = j_i^0$, $\left|L_i^1\right| = j_i^1$ and

$$
\begin{aligned}
L_i^0 &= \{\pi_{i,j}^0 := \langle q_{0,j}, w_0 \rangle \dots \langle q_{i,j}, w_i \rangle \mid 1 \leq j \leq j_i^0\}, \text{ and} \\
L_i^1 &= \{\pi_{i,j}^1 := \langle s_{0,j}, w_0 \rangle \dots \langle s_{i,j}, w_i \rangle \mid 1 \leq j \leq j_i^1\}
\end{aligned}
$$

(with $\langle r, w_i \rangle$ we mean that in the language acceptance game the current state is r and the remaining input word is w_i). We say that L_i is *valid* if it further satisfies the following *logbook properties*:

$$\mathbf{q}_i = \{q_{i,1}, \dots, q_{i,j_i^0}\} \tag{LP0}$$

$$\mathbf{s}_i = \{s_{i,1}, \dots, s_{i,j_i^1}\} \tag{LP1}$$

$$\forall \pi_{i,j}^0 \in L_i^0.\ \pi_{i,j}^0 \text{ is a } A_0\text{-conform partial play,} \tag{LP2}$$

i.e., (LP0) and (LP1) say that the logbook is correctly "synchronized" with the simulation game, and (LP2) says that every $\pi^0_{i,j}$ is built applying A_0's winning strategy.

Inductively, A_1 ensures that a valid logbook at round i is updated into a valid logbook in the next round $i+1$. In the first round $i = 0$, the initial $\mathbb{G}^{\mathsf{sim}}$-configuration is $\langle \mathbf{q}_0, \mathbf{s}_0\rangle = \langle \{q\}, \{s\}\rangle$, and the two acceptance games $\mathbb{G}^{\mathsf{acc}}_0$ and $\mathbb{G}^{\mathsf{acc}}_1$ are in $\langle q, w\rangle$ and $\langle s, w\rangle$, respectively. Clearly, $L_0 = (L^0_0, L^1_0)$ with $L^0_0 = \{\langle q, w\rangle\}$ and $L^1_0 = \{\langle s, w\rangle\}$ is a valid logbook.

Assume that L_i is a valid logbook for round i, that the current configuration of the simlulation game is $\langle \mathbf{q}_i, \mathbf{s}_i\rangle$, and that these two sets are partitioned into existential and universal states, in the following way:

$$\mathbf{q}_i = \{q^E_{i,1}, \ldots, q^E_{i,j^0_E}\} \cup \{q^U_{i,1}, \ldots, q^U_{i,j^0_U}\} \text{ and } \mathbf{s}_i = \{s^E_{i,1}, \ldots, s^E_{1,j^1_E}\} \cup \{s^U_{i,1}, \ldots, s^U_{1,j^1_U}\}$$

(Notice that $j^0_E + j^0_U = j^0_i$ and $j^1_E + j^1_U = j^1_i$.) The next input symbol in $\mathbb{G}^{\mathsf{sim}}$ is determined by the remaining input word $w_i = a_i a_{i+1} \ldots$, and it is equal to a_i. I.e., A_1 makes the S puppet choose a_i as the next input symbol in the simulation game $\mathbb{G}^{\mathsf{sim}}$, and the remaining input for the next round is $w_{i+1} = a_{i+1}a_{i+2}\ldots$. For determining the next $\mathbb{G}^{\mathsf{sim}}$-configuration $\langle \mathbf{q}_{i+1}, \mathbf{s}_{i+1}\rangle$, as well as updating the logbook to $L_{i+1} = (L^0_{i+1}, L^1_{i+1})$, A_1 orchestrates the various puppets according to the following steps.

1. The A_0 puppet moves from existential $\mathbb{G}^{\mathsf{acc}}_0$-configurations $\langle q^E_{i,1}, w^i\rangle \ldots \langle q^E_{i,j^0_E}, w^i\rangle$ according to her winning strategy in $\mathbb{G}^{\mathsf{acc}}_0$. Notice that *all* such pebbles in $\mathbf{q}^E_i$ can be moved to some a_i-successor, since 1) $w \in \mathcal{L}^\omega(q)$, and 2) different *Left* states are the result of different past choices at universal *Left* states, hence all states $q^E_{i,j} \in \mathbf{q}^E_i$ accept w_i (in fact, all states in $\mathbf{q}_i$). In this way, an a_i-successor $q^{(E)}_{i+1,j}$ is built for every $q^E_{i,j} \in \mathbf{q}^E_i$. Let $\mathbf{q}^{(E)}_{i+1} = \{q^{(E)}_{i+1,j} \mid 1 \leq j \leq j^0_E\}$ be the set of such a_i-successors. For any $q^E_{i,j} \in \mathbf{q}^E_{i+1}$ and $q^{(E)}_{i+1,j} \in \mathbf{q}^{(E)}_{i+1}$ s.t. $q^{(E)}_{i+1,j} \in \Delta(q^E_{i,j}, a_i)$, let $\pi^0_{i,j}$ be the partial play in L^0_i ending in $\langle q^E_{i,j}, w_i\rangle$ (which, by the induction hypothesis, exists by (LP0)). Then, we add $\pi^0_{i+1,j} = \pi^0_{i,j} \cdot \langle q^{(E)}_{i+1,j}, w_{i+1}\rangle$ to L^0_{i+1}. This establishes (LP0) and (LP2) at round $i+1$ for existential states in $\mathbf{q}_i$. (Notice that states in $\mathbf{q}^{(E)}_{i+1}$ need not be existential, we use the superscript (E) just to record that prececessors were existential.)
2. Similarly, the P_1 puppet chooses a successor for every universal $\mathbb{G}^{\mathsf{acc}}_1$-configurations $\langle s^U_{i,1}, w^i\rangle \ldots \langle s^U_{i,j^1_U}, w^i\rangle$. Let $\mathbf{q}^{(U)}_{i+1} = \{s^{(U)}_{i+1,j} \mid 1 \leq j \leq j^1_U\}$ be the set of such a_i-successors, if any. Notice that, at this point, one or even all such configurations may get stuck, i.e., with no a_i-successor, and, consequently, $\mathbf{q}^{(U)}_{i+1}$ might be empty. We need not worry about this now.
3. Then, the S puppet in the simulation game $\mathbb{G}^{\mathsf{sim}}$ copies the A_0's and P_1's moves above, moving from $\mathbf{q}^E_i$ to $\mathbf{q}^{(E)}_{i+1}$ and from $\mathbf{s}^U_i$ to $\mathbf{s}^{(U)}_{i+1}$.
4. Now, the D puppet moves according to her winning strategy: Pebbles in $\mathbf{q}^U_i$ are moved to $\mathbf{q}^{(U)}_{i+1}$ (possibly being split for D may hedge her bets), and pebbles in $\mathbf{s}^E_i$ are moved to $\mathbf{s}^{(E)}_{i+1}$ (possibly being split or thrown away), and the simulation game goes into state $(\mathbf{q}_{i+1}, \mathbf{s}_{i+1})$, where $\mathbf{q}_{i+1} = \mathbf{q}^{(E)}_{i+1} \cup \mathbf{q}^{(U)}_{i+1}$, and $\mathbf{s}_{i+1} \subseteq \mathbf{s}^{(E)}_{i+1} \cup \mathbf{s}^{(U)}_{i+1}$.

Since D's strategy is winning, at least one *Right* pebble can be moved to some successor. Thus, $\mathbf{s}_{i+1} \neq \emptyset$. D's move above is copied by the puppte P_0 and the player A_1, as specified below.

5. The P_0 puppet copies the D puppet's move from $\mathbf{q}_i^U$ to $\mathbf{q}_{i+1}^{(U)}$: For any $q_{i,j}^U \in \mathbf{q}_i^U$ and $q_{i+1,j}^{(U)} \in \mathbf{q}_{i+1}^{(U)} \cap \Delta(q_{i,j}^U, a_i)$, we add $\pi_{i+1,j}^0 = \pi_{i,j}^0 \cdot \langle q_{i+1,j}^{(U)}, w_{i+1}\rangle$ to L_{i+1}^0, where $\pi_{i,j}^0$ is the partial play ending in $\langle q_{i,j}^U, w_i\rangle$, which exists by (LP0). This establishes (LP0) and (LP2) for universal states in $\mathbf{q}_i$.
6. Finaly, the A_1 player in the family of acceptance games $\mathbb{G}_1^{\mathsf{acc}}$ copies the D puppet's move from $\mathbf{s}_i^E$ to $\mathbf{s}_{i+1}^{(E)}$: For any $s_{i,j}^E \in \mathbf{s}_i^E$ s.t. $s_{i+1,j}^{(E)} \in \mathbf{s}_{i+1}^{(E)} \cap \Delta(s_{i,j}^E, a_i)$, we add $\pi_{i+1,j}^1 = \pi_{i,j}^1 \cdot \langle s_{i+1,j}^{(E)}, w_{i+1}\rangle$ to L_{i+1}^1, where $\pi_{i,j}^1$ is the partial play ending in $\langle s_{i,j}^E, w_i\rangle$, which exists by (LP1). We also update the logbook for universal states in $\mathbf{s}_i$ which were not discarded in the previous step 2.: For any $s_{i,j}^U \in \mathbf{s}_i^U$ s.t. $s_{i+1,j}^{(U)} \in \mathbf{s}_{i+1}^{(U)} \cap \Delta(s_{i,j}^U, a_i)$, we add $\pi_{i+1,j}^1 = \pi_{i,j}^1 \cdot \langle s_{i+1,j}^{(U)}, w_{i+1}\rangle$ to L_{i+1}^1, where $\pi_{i,j}^1$ is the partial play ending in $\langle s_{i,j}^U, w_i\rangle$, which exists by (LP1). This establishes (LP1) for the next round $i+1$. (Notice that, since $\mathbf{s}_{i+1} \neq \emptyset$, then L_{i+1}^1 contains at least one partial play.)

Below we argue about the correctness of this construction. Let $\pi = \langle \mathbf{q}_0, \mathbf{s}_0\rangle\langle \mathbf{q}_1, \mathbf{s}_1\rangle \ldots$ be the resulting infinite play, and let $\pi^0 = \mathbf{q}_0\mathbf{q}_1 \ldots$ and $\pi^1 = \mathbf{s}_0\mathbf{s}_1 \ldots$ be its projections. Since every partial play in L_i^0 is A_0-conform (by (LP2)), and the A_0 puppet is playing according to a winning strategy (which exists, since $w \in \mathcal{L}^\omega(q)$), it follows that, for every $n \geq 0$, there exists $i \geq 0$ s.t. every $\pi_{i,j}^0 \in L_i^0$ has visited at least n accepting states. By (LP0) and since no *Left* pebble is thrown away, we have that $\mathrm{good}^\forall(\pi^0, \infty)$. Since D's strategy is winning, by the winning condition of fair simulation, we have $\mathrm{good}^\exists(\pi^1, \infty)$. By Lemma 9, there exists an infinite accepting path π_{acc}. Thus, A_1 has a strategy s.t., for any P_1's strategy, there exists a accepting path π_{acc} which is conform to both strategies. Thus, $w \in \mathcal{L}^\omega(s)$. □

Theorem 5. *For any $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$, there exist an automaton $\mathcal{Q}$ and states $q_0, s_0 \in \mathcal{Q}$ s.t. $\mathcal{L}^\omega(q_0) \subseteq \mathcal{L}^\omega(s_0)$, but $q_0 \not\sqsubseteq_{(n,n)}^x s_0$.*

Proof. By the inclusion between simulations (see Theorem 2), it is enough to consider fair simulation. Take the example in Figure 2(b). We have that $\mathcal{L}^\omega(q_0) = \mathcal{L}^\omega(s_0) = a^\omega + a^*b^\omega$, hence language inclusion holds between q_0 and s_0, but, as we shall see, no Duplicator strategy is winning in the (n,n)-simulation game $\mathbb{G}_{(n,n)}^{\mathrm{f}}(q_0, s_0)$.

This can be seen as follows. Spoiler chooses the a action and we can assume that Duplicator "hedges her bets" by going in $\mathbf{s}' = \{s_0, s_1\}$. Now, Spoiler keeps looping on q_0 by choosing the a action for an arbitrarily high number of moves. Duplicator can only reply by staying in a subset of $\mathbf{s}'$. Notice that Duplicator has to eventually take the pebble on s_0 away: Indeed, Spoiler's pebble in q_0 is accepting infinitely often, hence Duplicator would lose if the pebble on $s_0 \notin F$ is not taken away. When Duplicator takes the pebble on s_0 away, Spoiler plays the b action and Duplicator loses, his remaining pebble on s_1 being stuck. Similar examples may be conceived in which the acceptance condition, instead of the stuckness condition, is used to show that Duplicator loses. □

C.1 Infinite words: ABAs and the Miyano-Hayashi construction

Definition 3. *Given an ABA $\mathcal{Q} = (Q, \Sigma, q_I, \Delta, F, E, U)$, the Miyano-Hayashi construction [10] yields a de-universalized NBA*

$$\mathcal{Q}_{nd} := (Q', \Sigma, (\{q_I\}, \{q_I\} \setminus F), \Delta', F', E', U') ,$$

where the new set of states $Q' \subseteq 2^Q \times 2^Q$ (called macrostates*) consists of pairs of subsets of Q, the set of accepting macrostates F' satisfies*

$$(P, O) \in F' \iff O = \emptyset,$$

i.e., a macrostate is accepting if no obligation is pending, $E' = Q'$ and $U' = \emptyset$, i.e., $\mathcal{Q}_{nd}$ is a purely non-deterministic automaton, and the the transition relation $\Delta' \subseteq (2^Q \times 2^Q) \times \Sigma \times (2^Q \times 2^Q)$ satisfies: $((P, O), a, (P', O')) \in \Delta'$ iff there exists a choice function $\mathrm{select} : P \cap E \times \Sigma \mapsto Q$ *s.t.* $\forall p \in P \cap E, a \in \Sigma : \ (p, a, \mathrm{select}(p, a)) \in \Delta$ *(i.e.,* $\mathrm{select}(p, a)$ *fixes an element in $\Delta(p, a)$) such that*

$$P' = \bigcup_{p \in P \cap U} \Delta(p, a) \cup \{\mathrm{select}(p, a) \mid p \in P \cap E\} ,$$

and if $O = \emptyset$, then $O' = P' \setminus F$, otherwise,

$$O' = \left(\bigcup_{o \in O \cap U} \Delta(o, a) \cup \{\mathrm{select}(o, a) \mid p \in O \cap E\} \right) \setminus F .$$

Lemma 3. *For any ABA $\mathcal{Q}$, let $\mathcal{Q}_{nd}$ be the NBA obtained according to the Miyano-Hayashi de-universalization procedure applied to $\mathcal{Q}$. Then,*

a) $\mathcal{Q} \sqsubseteq^x_{(n,1)} \mathcal{Q}_{nd}$*, for $x \in \{\mathrm{f}, \forall\mathrm{di}\}$, and*
b) $\mathcal{Q}_{nd} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{Q}$*.*

Moreover, there exist automata $\mathcal{Q}^1$ and $\mathcal{Q}^2$ s.t.

a') $\mathcal{Q}^1 \not\sqsubseteq^{\mathrm{de}}_{(n,1)} \mathcal{Q}^1_{nd}$*, and*
b') $\mathcal{Q}^2_{nd} \not\sqsubseteq^x_{(1,1)} \mathcal{Q}^2$*, for $x \in \{\mathrm{de}, \forall\mathrm{di}\}$.*

Proof. We first prove Point a), i.e., $\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{Q}_{\mathrm{nd}}$. Intuitively, the strategy of Duplicator is to maximally hedge her bets on $\mathcal{Q}$ (i.e., *Left* universal pebbles), and to select successors in $\mathcal{Q}_{\mathrm{nd}}$ (which is a purely existential automaton) by copying Spoiler's moves from *Left* existential pebbles in $\mathcal{Q}$. More formally, there exists a strategy for Duplicator which mantains the following invariant: If at round k the current configuration is $\langle \mathbf{q}_k, (\mathbf{q}'_k, \mathbf{o}_k) \rangle$, then $\mathbf{q}_k = \mathbf{q}'_k$, i.e., Duplicator has a strategy that mimicks exactly the MH-construction.

We now argue that this strategy is winning for Duplicator. If some *Left* pebble gets stuck, then Duplicator wins. Otherwise, by the properties of the MH-construction, it is the case that the *Right* pebble can always be moved; in this case, we argue as follows. Let $\pi = \langle \mathbf{q}_0, (\mathbf{q}_0, \mathbf{o}_0) \rangle \langle \mathbf{q}_1, (\mathbf{q}_1, \mathbf{o}_1) \rangle \ldots$ be the resulting sequence of configurations.

For $x = \forall\mathrm{di}$, assume that $\mathbf{q}_k \subseteq F$. By the definition of the MH-construction $\mathbf{o}_k \subseteq \mathbf{q}_k \cap \overline{F}$, hence $\mathbf{o}_k = \emptyset$, i.e., $(\mathbf{q}_k, \emptyset) \in F'$. For $x = f$, assume that there are infinitely many i's s.t. $\mathbf{q}_i$ is universally good since some previous round j_i. Now consider the sequence of indices $\{k_i\}_{i \geq 0}$ defined as follows: $k_0 = 0$, and, inductively, k_{i+1} is s.t. $\mathbf{q}_{k_{i+1}}$ is good since round k_i. (Notice that this sequence is well-defined and infinite: Since there are infinitely many i's s.t. $\mathbf{q}_i$ is universally good since some previous round j_i, by Lemma 8, this implies that for any i, there exists $j_i \geq i$ s.t. $\mathbf{q}_{j_i}$ is universally good since round i.) We have that, by the definition of the MH-construction, $\mathbf{o}_{k_i} = \emptyset$ for any $i > 0$. Hence, $(\mathbf{q}_{k_i}, \emptyset) \in F'$ for infinitely many k_i's.

We now prove Point b), i.e., $\mathcal{Q}_{\text{nd}} \sqsubseteq^{\text{f}}_{(1,1)} \mathcal{Q}$. We can assume that the *Left* pebble never gets stuck, otherwise Duplicator wins trivially. Here, the strategy for Duplicator is to maintain the following invariant: If at round k the current configuration is $\langle (\mathbf{q}_k, \mathbf{o}_k), q_k \rangle$, then $q_k \in \mathbf{q}_k$, i.e., Duplicator can force the *Right* pebble to be somewhere in $\mathbf{q}_k$. Clearly, the invariant holds for the initial configuration: For $i = 0$, $(\mathbf{q}_0, \mathbf{o}_0) = (\{q_I\}, \{q_I\} \setminus F)$ and $q_0 = q_I$. Inductively, assume that the invariant $q_k \in \mathbf{q}_k$ holds for $k \geq 0$. We show how Dupicator can ensure it in the next round $k+1$. Assume that Spoiler moves the *Left* pebble to $(\mathbf{q}_{k+1}, \mathbf{o}_{k+1})$ and that the next input symbol is a_k. We have two cases to consider:

- If $q_k \in E$, then, by the MH-construction, there exists $q' \in \mathbf{q}_{k+1}$ s.t. $q' \in \Delta(q_k, a_k)$. In this case, Duplicator moves the *Right* pebble from q_k to $q_{k+1} := q'$.
- If $q_k \in U$, then, by the MH-construction, it is the case that $\Delta(q_k, a_k) \subseteq \mathbf{q}_{k+1}$. Hence, Spoiler moves the *Right* universal pebble q_k to any successor $q_{k+1} \in \Delta(q_k, a_k)$. For every Spoiler's move, $q_{k+1} \in \mathbf{q}_{k+1}$.

We now argue that this invariant-preserving strategy is winning for Duplicator. Assume that, for infinitely many i's, $(\mathbf{q}_i, \mathbf{o}_i)$ is accepting, i.e., $\mathbf{o}_i = \emptyset$. Hence, we can build an infinite sequence $\{i_j\}_{j \geq 0}$ of indices s.t. $\mathbf{o}_{i_j} = \emptyset$ for any $j > 0$. From the structure of the MH-construction, it follows that for any $j > 0$, there exists k s.t. $i_{j-1} < k \leq i_j$ and $q_k \in F$. Hence, q_k is accepting for infinitely many indices k's.

Points a') and b') are actually shown in Figure 3, where we give two automata $\mathcal{Q}^1$ and $\mathcal{Q}^2$ which results in the same de-universalized automaton, i.e., $\mathcal{Q}^1_{\text{nd}} = \mathcal{Q}^2_{\text{nd}} =: \mathcal{Q}_{\text{nd}}$. For Point a'), the simulation game $\mathbb{G}^{\text{de}}_{(n,1)}(\mathcal{Q}^1, \mathcal{Q}^1_{\text{nd}})$ results in the (unique) sequence of configurations

$$\begin{aligned}\pi = &\langle \{q_0\}, (\{q_0\}, \{q_0\}) \rangle \langle \{q_{11}, q_{12}\}, (\{q_{11}, q_{12}\}, \{q_{12}\}) \rangle \\ &\langle \{q_{21}, q_{22}\}, (\{q_{21}, q_{22}\}, \emptyset) \rangle \langle \{q_{31}, q_{32}\}, (\{q_{31}, q_{32}\}, \{q_{32}\}) \rangle^\omega ,\end{aligned}$$

but at round $k = 3$, the *Left* pebbles on $\{q_{31}, q_{32}\}$ are universally good since round 2 (since q_{22} and q_{31} are in F), but the *Right* pebble is never accepting for $k \geq 3$. For Point b'), the reasoning is similar, but now $q'_{31} \notin F$. We have that Spoiler can force the game $\mathbb{G}^x_{(1,1)}(\mathcal{Q}^2_{\text{nd}}, \mathcal{Q}^2)$ in the following sequence of configurations:

$$\pi = \langle s_0, q_0 \rangle \langle s_1, q_{11} \rangle \langle s_2, q_{21} \rangle \langle s_3, q'_{31} \rangle^\omega ,$$

s.t., at round $k = 2$, $s_2 \in F'$, but 1) $q_{21} \notin F$ (hence, $\mathcal{Q}^2_{\text{nd}} \not\sqsubseteq^x_{(1,1)} \mathcal{Q}^2$ for $x = \forall\mathrm{di}$), and 2) for no later round $k' \geq 2$, the *Right* pebble is accepting (in fact, it is trapped in $q'_{31} \notin F$), hence $\mathcal{Q}^2_{\text{nd}} \not\sqsubseteq^x_{(1,1)} \mathcal{Q}^2$ for $x = \mathrm{de}$. $\square$

Lemma 4. *Let $\mathcal{Q}, \mathcal{S}$ be two alternating Büchi automata. Then,*

$$\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S} \iff \mathcal{Q}_{nd} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S}_{nd} \; .$$

Proof. We make use of transitivity, which will established later.

"Only if". Assume $\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S}$. Then, by a double application of Lemma 3,

$$\mathcal{Q}_{\mathrm{nd}} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S}_{\mathrm{nd}} \; ,$$

and, since $(1,1)$-simulation is contained in $(n,1)$-simulation (see Theorem 1), by transitivity, we obtain $\mathcal{Q}_{\mathrm{nd}} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S}_{\mathrm{nd}}$. But $\mathcal{Q}_{\mathrm{nd}}$ is a purely existential automaton, hence $(n,1)$-simulation reduces to $(1,1)$-simulation in this case. Thus, $\mathcal{Q}_{\mathrm{nd}} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S}_{\mathrm{nd}}$.

"If". Assume $\mathcal{Q}_{\mathrm{nd}} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S}_{\mathrm{nd}}$. By Lemma 3,

$$\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{Q}_{\mathrm{nd}} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S}_{\mathrm{nd}} \sqsubseteq^{\mathrm{f}}_{(1,1)} \mathcal{S} \; ,$$

thus, by "upgrading" $(1,1)$-simulation to $(n,1)$, and by transitivity, $\mathcal{Q} \sqsubseteq^{\mathrm{f}}_{(n,1)} \mathcal{S}$. □

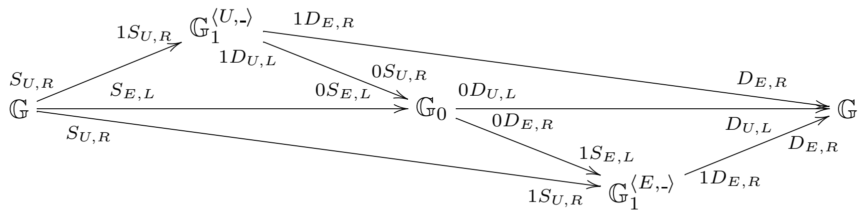
C.2 Transitivity

Theorem 6. *Let $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. Then,*

$$q \sqsubseteq^{x}_{(1,n)} r \wedge r \sqsubseteq^{x}_{(1,n)} s \implies q \sqsubseteq^{x}_{(1,n)} s$$
$$q \sqsubseteq^{x}_{(n,1)} r \wedge r \sqsubseteq^{x}_{(n,1)} s \implies q \sqsubseteq^{x}_{(n,1)} s \; .$$

Proof. Directly from Lemma 10 and Lemma 11 below. □

Definition of the join of two strategies for $(1,n)$ simulations. Let $\mathbb{G}_0 = \mathbb{G}^{(1,n)}(q,r)$ and $\mathbb{G}_1 = \mathbb{G}^{(1,n)}(r,s)$ be the basic simulation games between q and r, and between r and s, respectively. Let σ_0 and σ_1 be two Duplicator's strategies in $\mathbb{G}_0$ and $\mathbb{G}_1$, respectively. We construct a *joint strategy* $\sigma_0 \bowtie \sigma_1$ for Duplicator in the basic simulation game $\mathbb{G} = \mathbb{G}^{(1,n)}(q,s)$. In the definition of the join, we assume that the automaton is *complete*, hence it is always possible to select successors and the simulation game never halts.



We keep track of the current state of the $\mathbb{G}_0$ game and at most n games in $\mathbb{G}_1$. In the picture above, $\mathbb{G}_0$ is shown in the center, where games in $\mathbb{G}_1$ are shown at the top/bottom of the picture, where, for $X \in \{E, U\}$, with $\mathbb{G}_1^{\langle X,\text{-}\rangle}$ we mean those games in $\mathbb{G}_1$ where the (only) *Left* pebble is existential/universal. An arrow of the form $\cdot \xrightarrow{P_{X,Y} \qquad P'_{X,Y'}} \cdot$, means that X-pebbles on side Y moved by player P in the source game induce a move by player P' in the destination game on side Y', where $P, P' \in \{S, D, 0S, 0D, 1S, 1D\}$, $X \in \{E, U\}$ and $Y, Y' \in \{L(\mathit{eft}), R(\mathit{ight})\}$. (Notice that the kind of pebbles X does not change across an arrow.)

The necessary bookkeeping is done by using a *logbook*. At round $k \geq 0$, the current logbook is a triple $L_k = (L_k^0, L_k^1, w_k)$, where $w_k = a_0 a_1 \dots a_{k-1}$ is the input word constructed so far (and we let $w_0 = \varepsilon$ if $k = 0$), $L_k^0 = \pi_k^0$, with $\pi_k^0 = \langle q_0^0, \mathbf{r}_0^0 \rangle \langle q_1^0, \mathbf{r}_1^0 \rangle \dots \langle q_k^0, \mathbf{r}_k^0 \rangle$, is a partial play in $\mathbb{G}_0$ of length $k+1$, which is σ_0-conform to w_k, and $L_k^1 = \{\pi_{k,0}^1, \pi_{k,1}^1, \dots, \pi_{k,l(k)}^1\}$ is a set of partial plays in $\mathbb{G}_1$, which are σ_1-conform to w_k, of length $k+1$, and of the form

$$\pi_{k,i}^1 = \langle r_{0,i}^1, \mathbf{s}_{0,i}^1 \rangle \langle r_{1,i}^1, \mathbf{s}_{1,i}^1 \rangle \cdots \langle r_{k,i}^1, \mathbf{s}_{k,i}^1 \rangle, \text{ for } i \in 0, \dots, l(k) \; .$$

Assume that at round $k \geq 0$ the current partial play in $\mathbb{G}$ is π_k, where

$$\pi_k = \langle q_0, \mathbf{s}_0 \rangle \langle q_1, \mathbf{s}_1 \rangle \dots \langle q_k, \mathbf{s}_k \rangle \; . \tag{*}$$

We say that a logbook L_k is *valid* if it satisfies the *logbook properties* below.

$$q_k^0 = q_k \tag{P0}$$

$$\mathbf{r}_k^0 = \bigcup_{i=0}^{l(k)} \{r_{k,i}^1\} \tag{P1}$$

$$\mathbf{s}_k = \bigcup_{i=0}^{l(k)} \mathbf{s}_{k,i}^1 \tag{P2}$$

Notice that (P1) entails the following property: (P1') For every $r \in \mathbf{r}_k^0$, there exists i s.t. $\pi_{k,i}^1 = \langle r_{0,i}^1, \mathbf{s}_{0,i}^1 \rangle \cdots \langle r_{k,i}^1, \mathbf{s}_{k,i}^1 \rangle$ with $r = r_{k,i}^1$, i.e., r is the r.h.s. of the last configuration of some partial play in L_k^1.

We inductively show how to build a valid logbook and we define the joint strategy $\sigma_0 \bowtie \sigma_1$. The initial configuration in $\mathbb{G}_0$ is $\langle q, r \rangle$, the one in $\mathbb{G}_1$ is $\langle r, s \rangle$, and the one in $\mathbb{G}$ is $\langle q, s \rangle$. Hence, the initial logbook $L_0 := (L_0^0, L_0^1)$, with $L_0^0 = \langle q_0^0, r_0^0 \rangle$ and $L_0^1 = \{\langle r_0^1, s_0^1 \rangle\}$, is clearly valid, where $q_0^0 = q$, $r_0^0 = r_0^1 = r$ and $s_0^1 = s$.

Inductively assume that, at round k, L_k is a valid logbook, and that the current (partial) play in $\mathbb{G}$ is π_k, with $\pi_k = \langle q_0, \mathbf{s}_0 \rangle \langle q_1, \mathbf{s}_1 \rangle \cdots \langle q_k, \mathbf{s}_k \rangle$ (as in (*)). We show how to build a new logbook $L_{k+1} = (L_{k+1}^0, L_{k+1}^1, w_{k+1})$ for the next round, and we prove it valid. Assume that Spoiler moves as follows:

$$(\{q_k\}, \mathbf{s}_k, a_k, (\mathbf{q}_k)', (\mathbf{s}_k)') \in \mathit{\Gamma}_{\mathbb{G}}^{\mathrm{Sp}} \; , \tag{S}$$

i.e., the next input symbol is a_k and Spoiler moves universal-r.h.s. pebbles from $\mathbf{s}_k$ to $(\mathbf{s}_k)'$. Notice that, if q_k is existential, then Spoiler moves the only l.h.s. pebble from $\mathbf{q}_k := \{q_k\}$ to $(\mathbf{q}_k)' := \{(q_k)'\}$, otherwise $(\mathbf{q}_k)' := \emptyset$. Notice that we can already update the next input word to $w_{k+1} := w_k \cdot a_k$, which defines the third component of the next logbook.

For any universal state $r^U_{k,i} \in \mathbf{r}^0_k \cap U$, let $\pi^1_{k,i}$ be the path in L^1_k s.t. $\pi^1_{k,i} = \langle r^1_{0,i}, \mathbf{s}^1_{0,i}\rangle \cdots \langle r^1_{k,i}, \mathbf{s}^1_{k,i}\rangle$ with $r^1_{k,i} = r^U_{k,i}$, which is guaranteed to exist by property (P1'). Spoiler's move (S) above induces the $\mathbb{G}_1$-Spoiler's move from $\langle r^U_{k,i}, \mathbf{s}^1_{k,i}\rangle$ below:

$$(\{r^U_{k,i}\}, \mathbf{s}^1_{k,i}, a_k, \{\}, (\mathbf{s}^1_{k,i})') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_1}\ , \tag{1-S$_i$}$$

i.e., a_k is fixed by move (S) above, and $(\mathbf{s}^1_{k,i})'$ is the subset of $(\mathbf{s}_k)'$ obtained by restricting Spoiler's move (S) to $\mathbf{s}^1_{k,i} \subseteq \mathbf{s}_k$, the containment following by property (P2). We now apply $\mathbb{G}_1$-Duplicator's strategy σ_1, obtaining

$$\sigma_1(\langle r^1_{0,i}, \mathbf{s}^1_{0,i}\rangle \cdots \langle r^U_{k,i}, \mathbf{s}^1_{k,i}\rangle)(a_k, r^U_{k,i}, (\mathbf{s}^1_{k,i})') = \langle (r^U_{k,i})', (\mathbf{s}^1_{k,i})''\rangle\ . \tag{1-D$_i$}$$

By the completeness condition, it is always possible to select some successor $(r^U_{k,i})'$ and $(\mathbf{s}^1_{k,i})'' \neq \emptyset$.

The move (1-D$_i$) above fixes a successor $(r^U_{k,i})'$ for each universal state $r^U_{k,i}$ in $\mathbf{r}^0_k$. We now consider these moves as adversarial in $\mathbb{G}_0$, i.e., they induce a $\mathbb{G}_0$-Spoiler's move

$$(\{q^0_k\}, \mathbf{r}^0_k, a_k, (\mathbf{q}^0_k)', (\mathbf{r}^0_k)') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_0}\ , \tag{0-S}$$

where $(\mathbf{r}^0_k)'$ is the set of elements $(r^U_{k,i})'$ above. We then apply $\mathbb{G}_0$-Duplicator's strategy σ_0, obtaining

$$\sigma_0(\langle q^0_0, \mathbf{r}^0_0\rangle \cdots \langle q^0_k, \mathbf{r}^0_k\rangle)(a_k, (\mathbf{q}^0_k)', (\mathbf{r}^0_k)') = \langle q^0_{k+1}, \mathbf{r}^0_{k+1}\rangle\ . \tag{0-D}$$

Notice that, if q^0_k is existential, then $\{q^0_{k+1}\} = (\mathbf{q}^0_k)'$ as determined in (S), otherwise q^0_{k+1} is determined by (0-D) above. The new configuration of the game $\mathbb{G}_0$ is $\langle q^0_{k+1}, \mathbf{r}^0_{k+1}\rangle$, and, accordingly, the first component L^0_{k+1} of the new logbook is defined as $L^0_{k+1} := \pi^0_k \cdot \langle q^0_{k+1}, \mathbf{r}^0_{k+1}\rangle$. By the completeness condition, it is always possible to select some successor q^0_{k+1} and $\mathbf{r}^0_{k+1} \neq \emptyset$.

The $\mathbb{G}_0$-Duplicator's move (0-D) above fixes a successor $(r^E_{k,i})'$ for any existential $r^E_{k,i} \in \mathbf{r}^0_k \cap E$ By the logbook property (P1'), for any such $r^E_{k,i}$, there exists a path $\pi^1_{k,i}$ in L^1_k s.t. $\pi^1_{k,i} = \langle r^1_{0,i}, \mathbf{s}^1_{0,i}\rangle \cdots \langle r^E_{k,i}, \mathbf{s}^1_{k,i}\rangle$. The $\mathbb{G}_0$-Duplicator's move (0-D) above is interpreted adversarially in $\mathbb{G}_1$:

$$(\{r^E_{k,i}\}, \mathbf{s}^1_{k,i}, a_k, (r^E_{k,i})', (\mathbf{s}^1_{k,i})') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_1}\ . \tag{1-S$'_i$}$$

We then apply $\mathbb{G}_1$-Duplicator's winning strategy σ_1:

$$\sigma_1(\langle r^1_{0,i}, \mathbf{s}^1_{0,i}\rangle \cdots \langle r^E_{k,i}, \mathbf{s}^1_{k,i}\rangle)(a_k, (r^E_{k,i})', (\mathbf{s}^1_{k,i})') = \langle (r^E_{k,i})', (\mathbf{s}^1_{k,i})''\rangle\ . \tag{1-D$'_i$}$$

Once again, the completeness condition entails that it is always possible to select some successor $(r^E_{k,i})'$ and $(\mathbf{s}^1_{k,i})'' \neq \emptyset$.

We are now ready to define L^1_{k+1}. Let $(r^1_{k,i})'$ be any state $(r^U_{k,i})'$ defined in (1-D$_i$), or any state $(r^E_{k,i})'$ defined in (1-D$'_i$), *which is not discarded in* (0-D), i.e., $(r^1_{k,i})' \in \mathbf{r}^0_{k+1}$. (Notice that $\mathbf{r}^0_{k+1} \neq \emptyset$ by the completeness property, as already noticed above. Hence, there exists at least one such $(r^1_{k,i})'$.) We then add $\pi^1_{k,i} \cdot \langle (r^1_{k,i})', (\mathbf{s}^1_{k,i})''\rangle$ to the second component L^1_{k+1} of the new logbook. It is easy to check that, by construction, property (P1) holds at round $k+1$.

Finally, we *define* Duplicator's move in $\mathbb{G}$ as

$$(\{q_k\}, \mathbf{s}_k, a_k, (\mathbf{q}_k)', (\mathbf{s}_k)', \{q_{k+1}\}, \mathbf{s}_{k+1}) \in \Gamma_{\mathbb{G}}^{\mathrm{Dup}} \ , \tag{D}$$

where $q_{k+1} := q^0_{k+1}$ is fixed by move (0-D) if $q_k = q^0_k$ is existential, and it is fixed by (S) if it is universal. (Notice that this establishes property (P0).) Moreover, $\mathbf{s}_{k+1}$ is taken to be the union of all sets $(\mathbf{s}^1_{k,i})''$ constructed in (1-D$_i$) and (1-D$'_i$) above, i.e., $\mathbf{s}_{k+1} := \bigcup_i (\mathbf{s}^1_{k,i})''$, which, in turn, establishes property (P2). Hence,

$$(\sigma_0 \bowtie \sigma_1) \left(\langle q_0, \mathbf{s}_0 \rangle \cdots \langle q_k, \mathbf{s}_k \rangle \right) (a_k, (\mathbf{q}_k)', (\mathbf{s}_k)') := \langle q_{k+1}, \mathbf{s}_{k+1} \rangle \ .$$

The following theorem shows that $\sqsubseteq_{(1,n)}$ is transitive, i.e., it shows that when σ_0 and σ_1 are both winning, then $\sigma_0 \bowtie \sigma_1$ is winning as well.

Lemma 10. *Let* $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. *Then,*

$$q \sqsubseteq^x_{(1,n)} r \wedge r \sqsubseteq^x_{(1,n)} s \implies q \sqsubseteq^x_{(1,n)} s \ .$$

Proof. We refer to the logbook L_k at round k as defined above. We first deal with the case in which the game $\mathbb{G}$ never ends prematurely.

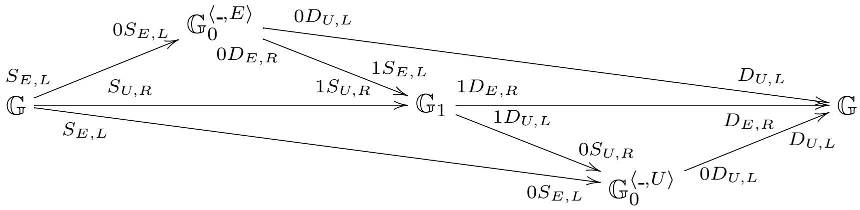
For $x = \forall\mathrm{di}$, we have to show that, whenever q_k is accepting, so is every pebble in $\mathbf{s}_k$. Assume $q_k \in F$. Since σ_0 is a winning strategy, then $\mathbf{r}^0_k \subseteq F$. By the logbook property (P1'), the current configuration of every game in $\mathbb{G}_1$ is of the form $\langle r, \mathbf{s} \rangle$, for some $r \in F$ and $\mathbf{s} \subseteq Q$. But σ_1 is winning, hence every such $\mathbf{s}$ is contained in F. By (P2), $\mathbf{s}_k \subseteq F$.

For $x = \mathrm{de}$, assume that q_k is accepting. Since σ_0 is winning, there exists $j \geq k$ s.t. $\mathrm{good}^{\exists}(\mathbf{r}^0_j, k)$. Thus, for any $r^1_{j,i} \in \mathbf{r}^0_j$, there exists $j(i)$ s.t. $r^1_{j(i),i} \in F$. By (P1), let $\langle r^1_{j(i),i}, \mathbf{s}_{j(i),i} \rangle$ be the configuration at round $j(i)$ of some $\mathbb{G}_1$-game. Since σ_1 is winning, there exists $j(i)' \geq j(i)$ s.t. $\mathrm{good}^{\exists}(\mathbf{s}_{j(i)',i}, j(i))$. Let $j^* = \max_i(j(i)')$: Then, for all i, every state in $\mathbf{s}_{j*,i}$ has seen an accepting state since round k. By (P2), $\mathbf{s}_{j^*} = \bigcup_i \mathbf{s}^1_{j^*,i}$. Thus, every state in $\mathbf{s}_{j^*}$ has seen an accepting state since round k. Therefore, there exists a minimal k^* s.t. $k \leq k^* \leq j^*$ and $\mathrm{good}^{\exists}(\mathbf{s}_{k^*}, k)$.

For $x = \mathrm{f}$, the reasoning is entirely similar to the previous paragraph.

We now deal with the case in which the game $\mathbb{G}$ ends prematurely. If the *Left* pebble on q_k is stuck, then Duplicator wins, and we are done. Otherwise, assume that the *Left* pebble is never stuck. We show that, in this case, the game actually never stops. In fact, since σ_0 is a winning strategy in $\mathbb{G}_0$, then there always exists some *Right* pebble $r_{k,i} \in \mathbf{r}_k$ which can proceed. By (P1)', there exists some configuration $\langle r_{k,i}, \mathbf{s}_{k,i} \rangle$ in $\mathbb{G}_1$ which can go on, and, being σ_1 winning in such a game, then some pebble in $\mathbf{s}_{k,i}$ can be moved, and, therefore, some pebble in $\mathbf{s}_k$ can be moved as well (by (P2)). Thus, $\mathbb{G}$ never stops. □

Definition of the join of two strategies for $(n,1)$ simulations. The definitions for $(n,1)$ simulations are dual to the $(1,n)$ case in the previous section. Let $\mathbb{G}_0 = \mathbb{G}^{(n,1)}(q,r)$ and $\mathbb{G}_1 = \mathbb{G}^{(n,1)}(r,s)$ be two basic simulation games. Let σ_0 and σ_1 be two Duplicator's strategies in $\mathbb{G}_0$ and $\mathbb{G}_1$, respectively. We construct a *joint strategy* $\sigma_0 \bowtie \sigma_1$ for Duplicator in the basic simulation game $\mathbb{G} = \mathbb{G}^{(n,1)}(q,s)$.



We keep track of the current state of the $\mathbb{G}_1$ game and (at most) n games in $\mathbb{G}_0$ using the logbook technique. At round $k \geq 0$, the current logbook is a triple $L_k = (L_k^0, L_k^1, w_k)$, where $w_k = a_0 a_1 \ldots a_{k-1}$ is the input word constructed so far, $L_k^1 = \pi_k^1 = \langle \mathbf{r}_0, s_0^1\rangle\langle \mathbf{r}_1, s_1^1\rangle \ldots \langle \mathbf{r}_k, s_k^1\rangle$ is a partial σ_1-conform to w_k play of length $k+1$ in $\mathbb{G}_1$, and $L_k^0 = \{\pi_{k,0}^0, \pi_{k,1}^0, \ldots, \pi_{k,l(0)}^0\}$ is a set of partial σ_0-conform (w.r.t. w_k) plays of length $k+1$ in $\mathbb{G}_0$, with $\pi_{k,i}^0 = \langle \mathbf{q}_{0,i}^0, r_{0,i}^0\rangle\langle \mathbf{q}_{1,i}^0, r_{1,i}^0\rangle \cdots \langle \mathbf{q}_{k,i}^0, r_{k,i}^0\rangle$, for $i \in \{0, \ldots, l(k)\}$. Every logbook L_k will satisfy an invariant, which consists of the *logbook properties* (P0)–(P2) specified below. Assume that at round $k \geq 0$ the current play in $\mathbb{G}$ is π_k, where $\pi_k = \langle \mathbf{q}_0, s_0\rangle\langle \mathbf{q}_1, s_1\rangle \ldots \langle \mathbf{q}_k, s_k\rangle$. Then, L_k is a valid logbook if

$$\mathbf{q}_k = \bigcup_{i=0}^{l(k)} \mathbf{q}_{k,i}^0 \tag{P0}$$

$$\mathbf{r}_k = \bigcup_{i=0}^{l(k)} \{r_{k,i}^0\} \tag{P1}$$

$$s_k = s_k^1 \tag{P2}$$

We inductively show how to build a valid logbook and how to define the joint strategy $\sigma_0 \bowtie \sigma_1$. The initial configuration in $\mathbb{G}_0$ is $\langle \{q\}, \{r\}\rangle$ (there is only one such game initially), the one in $\mathbb{G}_1$ is $\langle \{r\}, \{s\}\rangle$, and the one in $\mathbb{G}$ is $\langle \{q\}, \{s\}\rangle$. Let $\mathbf{q}_0 = \{q\}$, $r_0 = r$, $\mathbf{r}_0 = \{r\}$, and $s_0 = s$. Hence, the initial logbook $L_0 := (L_0^0, L_0^1, w_0)$, with $L_0^0 = \{\langle \{q_0\}, \{r_0\}\rangle\}$, $L_0^1 = \langle \{r_0\}, \{s_0\}\rangle$, and $w_0 = \varepsilon$, is clearly valid.

Inductively assume that, at round k, the current partial play in $\mathbb{G}$ is π_k, and that $L_k = (L_k^0, L_k^1, w_k)$ is a valid logbook, where L_k^0, L_k^1 and w_k are defined as above. By (P2),

$$\begin{aligned}
\pi_k &= \langle \mathbf{q}_0, s_0\rangle\langle \mathbf{q}_1, s_1\rangle \ldots \langle \mathbf{q}_k, s_k\rangle \\
\pi_{k,i}^0 &= \langle \mathbf{q}_{k,0}^0, r_{0,i}^0\rangle\langle \mathbf{q}_{k,1}^0, r_{1,i}^0\rangle \cdots \langle \mathbf{q}_{k,i}^0, r_{k,i}^0\rangle, \text{ for } i \in \{0, \ldots, l(k)\} \\
\pi_k^1 &= \langle \mathbf{r}_0, s_0\rangle\langle \mathbf{r}_1, s_1\rangle \ldots \langle \mathbf{r}_k, s_k\rangle
\end{aligned}$$

We show how to build a new, valid logbook $L_{k+1} = (L^0_{k+1}, L^1_{k+1}, w_{k+1})$ for the next round. Assume that $\mathbb{G}$-Spoiler moves as follows:

$$(\mathbf{q}_k, \{s_k\}, a_k, (\mathbf{q}_k)', (\mathbf{s}_k)') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}} , \tag{S}$$

i.e., the next input symbol is a_k and Spoiler moves existential-*Left* pebbles from $\mathbf{q}_k$ to $(\mathbf{q}_k)'$. We take $w_{k+1} = w_k \cdot a_k$. Notice that, if s_k is existential, then Spoiler moves the only *Right* pebble from $\mathbf{s}_k := \{s_k\}$ to $(\mathbf{s}_k)' := \{(s_k)'\}$, otherwise $(\mathbf{s}_k)' := \emptyset$.

For any existential state $r^E \in \mathbf{r}^E_k$, let $\pi^0_{k,i}$ be the path in L^0_k ending in $\langle \mathbf{q}^0_{k,i}, r^E\rangle$ (i.e., $r^0_{k,i} = r^E$), which is guaranteed to exist by property (P1). Spoiler's move (S) above induces the $\mathbb{G}_0$-Spoiler's move below

$$(\mathbf{q}^0_{k,i}, \{r^E\}, a_k, (\mathbf{q}^0_{k,i})', \{\}) \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_0} , \tag{0-S$_i$}$$

which is obtained by restricting to $\mathbf{q}^0_{k,i} \subseteq \mathbf{q}_k$ the transiton from $\mathbf{q}_k$ to $(\mathbf{q}_k)'$, the inclusion following from (P0). We apply $\mathbb{G}_0$-Duplicator's strategy σ_0, obtaining

$$\sigma_0(\langle \mathbf{q}^0_{0,i}, r^0_{0,i}\rangle a_0 \cdots a_{k-1} \langle \mathbf{q}^0_{k,i}, r^E\rangle)(a_k, (\mathbf{q}^0_{k,i})', \{\}) = (\mathbf{q}^0_{k+1,i}, r^0_{k+1,i}) . \tag{0-D$_i$}$$

The move (0-D$_i$) above fixes a successor $r^0_{k+1,i}$ for each existential state r^E in $\mathbf{r}^E_k$ (in $\mathbb{G}_0$). We now consider these moves as adversarial in $\mathbb{G}_1$, i.e., they induce a $\mathbb{G}_1$-Spoiler's move

$$(\mathbf{r}_k, \{s_k\}, a_k, (\mathbf{r}_k)', (\mathbf{s}_k)') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_1} , \tag{1-S}$$

where $(\mathbf{r}_k)'$ is the set of elements $r^0_{k+1,i}$ defined above. We then apply $\mathbb{G}_1$-Duplicator's strategy σ_1, obtaining

$$\sigma_1(\langle \mathbf{r}_0, s_0\rangle a_0 \cdots a_{k-1} \langle \mathbf{r}_k, s_k\rangle)(a_k, (\mathbf{r}_k)', (\mathbf{s}_k)') = (\mathbf{r}_{k+1}, \{s^1_{k+1}\}) . \tag{1-D}$$

Notice that, if s_k is universal, then $s^1_{k+1} = (s_k)'$, as determined in (S), otherwise s^1_{k+1} is determined by (1-D) above. The second component L^1_{k+1} of the new logbook is $L^1_{k+1} = \pi^1_k \cdot \langle \mathbf{r}_{k+1}, s^1_{k+1}\rangle$.

Call a state $r \in \mathbf{r}_k$ *useful* iff it has not been discarded by move (1-D), i.e., iff it has some successor in $\mathbf{r}_{k+1}$. By the logbook property (P1), for each useful universal state $r^U \in \mathbf{r}^U_k$, there exists a play $\pi^0_{k,i} \in L^0_k$ s.t. $\pi^0_{k,i} = \langle \mathbf{q}^0_{0,i}, r^0_{0,i}\rangle \cdots \langle \mathbf{q}^0_{k,i}, r^U\rangle$ with $r^0_{k,i} = r^U$. The $\mathbb{G}_1$-Duplicator's move (1-D) is then interpreted adversarially in $\mathbb{G}_0$:

$$(\mathbf{q}^0_{k,i}, \{r^U\}, a_k, (\mathbf{q}^0_{k,i})', \{r^0_{k+1,i}\}) \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_0} , \tag{0-S$'_i$}$$

and we apply $\mathbb{G}_0$-Duplicator's winning strategy σ_0, yielding

$$\sigma_0(\langle \mathbf{q}^0_{0,i}, r^0_{0,i}\rangle a_0 \cdots a_{k-1} \langle \mathbf{q}^0_{k,i}, r^U\rangle)(a_k, (\mathbf{q}^0_{k,i})', r^0_{k+1,i}) = (\mathbf{q}^0_{k+1,i}, r^0_{k+1,i}) . \tag{0-D$'_i$}$$

We now update the first component L^0_k of the logbook. For any useful existential or universal state $r^0_{k,i} \in \mathbf{r}_k$ with corresponding play $\pi^0_{k,i} \in L^0_k$ (as above), let $r^0_{k+1,i}$ be as determined in (0-D$_i$) or (0-S$'_i$), respectively. Then, we add $\pi^0_{k,i} \cdot \langle \mathbf{q}^0_{k+1,i}, r^0_{k+1,i}\rangle$ to L^0_{k+1}. Since every element in $\mathbf{r}_{k+1}$ arises as a successor of some *useful* element in $\mathbf{r}_k$, we have that (P1) holds at round $k + 1$.

Finally, we *define* Duplicator's move in $\mathbb{G}$ as

$$(\sigma_0 \bowtie \sigma_1)\left(\langle \mathbf{q}_0, s_0\rangle a_0 \cdots a_{k-1}\langle \mathbf{q}_k, s_k\rangle\right)(a_k, (\mathbf{q}_k)', (\mathbf{s}_k)') := (\mathbf{q}_{k+1}, s_{k+1})\ . \tag{D}$$

where $\mathbf{q}_{k+1}$ is as union over all sets $\mathbf{q}^0_{k+1,i}$ defined by equations (0-D$_i$) and (0-D$'_i$), and $s_{k+1} := s^1_{k+1}$ is defined according to (S) or (1-D), depending on whether s_k was universal or existential, respectively. Notice that, by definition of $\mathbf{q}_{k+1}$ and s_{k+1}, properties (P0) and (P2) hold for the new logbook. This completes the description of the joint strategy $\sigma_0 \bowtie \sigma_1$.

The following theorem shows that $\sqsubseteq^x_{(n,1)}$ is transitive, i.e., it shows that when σ_0 and σ_1 are both winning, then $\sigma_0 \bowtie \sigma_1$ is winning as well.

Lemma 11. *Let $x \in \{\forall\mathrm{di}, \mathrm{de}, \mathrm{f}\}$. Then,*

$$q \sqsubseteq^x_{(n,1)} r \wedge r \sqsubseteq^x_{(n,1)} s \implies q \sqsubseteq^x_{(n,1)} s\ .$$

Proof. We refer to the logbook L_k at round k as defined above. For $x = \forall\mathrm{di}$, we have to show that, whenever some pebble in $\mathbf{q}_k$ is accepting, so is s_k. Assume $\mathbf{q}_k \cap F \neq \emptyset$. Then, there exists $q^F \in \mathbf{q}_k \cap F$ and, by (P0), there exists $\mathbf{q}^0_{k,i} \subseteq \mathbf{q}_k$ s.t. $q^F \in \mathbf{q}^0_{k,i}$ and $\langle \mathbf{q}^0_{k,i}, r^0_{k,i}\rangle$ is the current configuration in some $\mathbb{G}_0$ game. Since $\mathbf{q}^0_{k,i} \cap F \neq \emptyset$ and σ_0 is a winning strategy, then $r^0_{k,i} \in F$. Hence, by the logbook property (P1), $\mathbf{r}_k = \bigcup_i \{r^0_{k,i}\} \cap F \neq \emptyset$, where $\langle \mathbf{r}_k, s_k\rangle$ is the current configuration in $\mathbb{G}_1$. Since $\mathbf{r}_k \cap F \neq \emptyset$ and σ_1 is winning, we have $s_k \in F$.

For $x = \mathrm{de}$, assume that at round k every pebble in $\mathbf{q}_k$ is universally good since some previous round, i.e., $\mathrm{good}^\forall(\mathbf{q}_k)$ holds. Let $\langle \mathbf{q}^0_{k,i}, r^0_{k,i}\rangle$ be any configuration in $\mathbb{G}_0$. By (P0), $\mathbf{q}^0_{k,i} \subseteq \mathbf{q}_k$, thus $\mathrm{good}^\forall(\mathbf{q}^0_{k,i})$. Since σ_0 is winning, then, for every i, there exists $k(i)$ s.t. $\mathrm{good}^\exists(\{r^0_{k(i),i}\}, k)$, i.e., $k(i)$ is the least index k' s.t. $r^0_{k',i} \in F$. Let i^* be the index for which $r^0_{k(i^*),i^*} \in F$ is the last pebble being accepting for the first time since round k, i.e., $i^* = \underset{i}{\mathrm{argmax}}(k(i))$. Hence, at round $k(i^*) \geq k$, every pebble in $\mathbf{r}_{k(i^*)}$ has been universally good since round k. Since σ_1 is winning, then there exists $k' \geq k(i^*) \geq k$ s.t. $s_{k'} \in F$. Let $k^* \leq k'$ be the minimal $k'' \in [k, \ldots, k']$ s.t. $s_{k''} \in F$. Therefore, $\mathrm{good}^\exists(\{s_{k^*}\}, k)$, i.e., $\{s_{k^*}\}$ is existentially good since round k.

For $x = \mathrm{f}$, assume that $\mathbf{q}_k$ is universally good since some previous round for infinitely many k's. By reasoning as above for delayed simulation, since σ_0 is winning, then $\mathbf{r}_k$ is universally good since some previous round for infinitely many k's. Finally, being σ_1 winning, we conclude that $s_k \in F$ for infinitely many k's. $\square$

D Section 5

D.1 Finite words

Lemma 5. *Let $\mathcal{Q}$ be any alternating finite automaton, and let $\preceq$ be any preorder which implies finite-language inclusion. Then, $\mathcal{L}^{\mathrm{fin}}(q) = \mathcal{L}^{\mathrm{fin}}([q])$.*

Proof. We proceed by induction on the length of $w \in \Sigma^*$. Let q any state in Q, and let $[q]$ be its equivalence class. Notice that, $\mathcal{L}^{\mathrm{fin}}(q) = \mathcal{L}^{\mathrm{fin}}(q')$ for any $q' \in [q]$.

Assume $w = \varepsilon$. If $w \in \mathcal{L}^{\mathrm{fin}}(q)$ then $q \in F$, hence $[q] \in F'$ and $w \in \mathcal{L}^{\mathrm{fin}}([q])$ as well. Conversely, if $w \in \mathcal{L}^{\mathrm{fin}}([q])$ then $[q] \in F'$, hence there exists $q^F \in [q] \cap F$. But $\mathcal{L}^{\mathrm{fin}}(q^F) = \mathcal{L}^{\mathrm{fin}}(q)$, hence $w \in \mathcal{L}^{\mathrm{fin}}(q)$.

Assume $w = a_0 \dots a_{k-1}$ is a word of length k, and let $w' = a_1 \dots a_{k-1}$. We proceed by case analysis on the type of $[q]$.

- Case 1: $[q] \in E'$. We prove $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}([q])$, distinguishing two subcases.
  Subcase 1.1: $q \in E$. Assume $w \in \mathcal{L}^{\mathrm{fin}}(q)$. Then, there exists $(q, a_0, q') \in \Delta$ s.t. $w' \in \mathcal{L}^{\mathrm{fin}}(q')$, and, w.l.o.g., we may assume that q' is an a_0-maximal successor of q. (If not, then there exists a state $q'' \succeq q'$ which actually is an a_0-maximal successor of q, thus $w' \in \mathcal{L}^{\mathrm{fin}}(q'')$ and then one can proceed from q''.) By induction hypothesis, $w' \in \mathcal{L}^{\mathrm{fin}}([q'])$. Hence, by the definition of quotient, there exists $([q], a_0, [q']) \in \Delta_{\approx}^{\mathrm{m}}$, thus $w \in \mathcal{L}^{\mathrm{fin}}([q])$.
  Subcase 1.2: $q \in U$. Let $q^E \in [q] \cap E$ and, by definition of quotient, $\mathcal{L}^{\mathrm{fin}}(q) = \mathcal{L}^{\mathrm{fin}}(q^E)$, and then one can proceed as above from q^E. Thus, $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}([q])$.
  We now prove $\mathcal{L}^{\mathrm{fin}}([q]) \subseteq \mathcal{L}^{\mathrm{fin}}(q)$. If $w \in \mathcal{L}^{\mathrm{fin}}([q])$, then $([q], a_0, [q']) \in \Delta_{\approx}^{\mathrm{m}}$ s.t. $w' \in \mathcal{L}^{\mathrm{fin}}([q'])$. By the definition of quotient, there exist $\widehat{q}^E \in [q]$ and $\widehat{q}' \in [q']$ s.t. $(\widehat{q}^E, a_0, \widehat{q}') \in \Delta$. By induction hypothesis, $w' \in \mathcal{L}^{\mathrm{fin}}(\widehat{q}')$ (we do not use the maximality of $\widehat{q}'$ here), hence $w \in \mathcal{L}^{\mathrm{fin}}(\widehat{q}^E) = \mathcal{L}^{\mathrm{fin}}(q)$.
- Case 2: $[q] \in U'$. We prove $\mathcal{L}^{\mathrm{fin}}(q) \subseteq \mathcal{L}^{\mathrm{fin}}([q])$. Assume $w \in \mathcal{L}^{\mathrm{fin}}(q)$. Let $[q']$ be *any* element in $[Q]$ s.t. $([q], a_0, [q']) \in \Delta_{\approx}^{\mathrm{m}}$. We have to show that $w' \in \mathcal{L}^{\mathrm{fin}}([q'])$. By the definition of quotient, there exist $\widehat{q} \in [q]$ and $\widehat{q}' \in [q']$ (we do not use the minimality of $\widehat{q}'$ here) s.t. $(\widehat{q}, a_0, \widehat{q}') \in \Delta$. We have that $\mathcal{L}^{\mathrm{fin}}(q) = \mathcal{L}^{\mathrm{fin}}(\widehat{q})$, hence $w \in \mathcal{L}^{\mathrm{fin}}(\widehat{q})$. Since $\widehat{q} \in U$, then every a_0-successor of $\widehat{q}$ accepts w'. In particular, $w' \in \mathcal{L}^{\mathrm{fin}}(\widehat{q}')$, and, by induction hypothesis, $w' \in \mathcal{L}^{\mathrm{fin}}([q'])$, But $[q']$ was arbitrary, thus $w \in \mathcal{L}^{\mathrm{fin}}([q])$.
  We prove $\mathcal{L}^{\mathrm{fin}}([q]) \subseteq \mathcal{L}^{\mathrm{fin}}(q)$. Assume $w \in \mathcal{L}^{\mathrm{fin}}([q])$. Let q' be *any* element in Q s.t. $(q, a_0, q') \in \Delta$, and we have to show $w' \in \mathcal{L}^{\mathrm{fin}}(q')$ for any such q'. In particular, it is sufficient to show $w' \in \mathcal{L}^{\mathrm{fin}}(q')$ for any a_0-minimal q', since $\mathcal{L}^{\mathrm{fin}}(q') \subseteq \mathcal{L}^{\mathrm{fin}}(q'')$ for any $q'' \succeq q'$. Hence, we assume that q' is an a_0-minimal successor of q. Being $[q] \in U'$, we have that $w' \in \mathcal{L}^{\mathrm{fin}}([q'])$ for any a_0-successor $[q']$ of $[q]$. As $(q, a_0, q') \in \Delta$ and by the definition of quotient, there exists $([q], a_0, [q']) \in \Delta_{\approx}^{\mathrm{m}}$. Thus, $w' \in \mathcal{L}^{\mathrm{fin}}([q'])$, and, by induction hypothesis, $w' \in \mathcal{L}^{\mathrm{fin}}(q')$. But q' was arbitrary, thus $w \in \mathcal{L}^{\mathrm{fin}}(q)$. $\square$

D.2 Infinite words: Direct simulation

The two directions in Theorem 8 are proved, resp., by Lemma 12 and Lemma 13 below.

Lemma 12. *If* $q \sqsubseteq^{\forall\mathrm{di}}_{(1,n)} \mathbf{s}$*, then* $[q]_{\mathrm{m}+} \sqsubseteq^{\forall\mathrm{di}}_{(1,n)} \mathbf{s}$*, where the quotient is taken w.r.t.* $\sqsubseteq^{\forall\mathrm{di}}_{(1,k)}$.

Proof. Let $\mathbb{G} = \mathbb{G}^{\forall\mathrm{di}}_{(1,n)}([q], \mathbf{s})$ and, at round i, if the current configuration of $\mathbb{G}$ is $\langle [q_i], \mathbf{s}_i \rangle$, let $\mathbb{G}_i = \mathbb{G}^{\forall\mathrm{di}}_{(1,n)}(q_i, \mathbf{s}_i)$. We maintain the following invariant: At round i, $q_i \sqsubseteq^{\forall\mathrm{di}}_{(1,n)} \mathbf{s}_i$. Notice that the invariant implies the lemma: The crucial observation is that $[q_i] \in F'$ implies $[q_i] \subseteq F$, i.e., if one state in the quotient is acceping, then, by the definition of direct simulation, all states in the quotient are accepting as well, and, in particular, $q_i \in F$. By the invariant and by the definition of $\forall$di-simulation, $q_i \in F$ implies $\mathbf{s}_i \subseteq F$.

Assume the current configuration in $\mathbb{G}$ is $\langle [q_i], \mathbf{s}_i \rangle$, and $q_i \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_i$. Let Spoiler choose the next input symbol a_i. We consider two cases, depending on whether $[q_i]$ is existential or universal.

First case: $[q_i] \in E'$. Let Spoiler choose an a_i-successor $[q_{i+1}]$ of $[q_i]$, i.e., Spoiler chooses transition

$$(\{[q_i]\}, \mathbf{s}_i, a_i, \{[q_{i+1}]\}, \mathbf{s}') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}} \ .$$

By the definition of minimax quotient, there exist $\widehat{q} \in [q_i] \cap E$ and $q' \in [q_{i+1}]$ s.t. $q' \in \mathit{\Delta}(\widehat{q}, a)$. (Note that we do not use the maximality of q' in this proof.) We have $\widehat{q} \sqsubseteq^{\forall \mathrm{di}}_{(1,k)} q_i \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_i$. But $(1,k)$-simulation implies $(1,n)$-simulation (Theorem 1), therefore $\widehat{q} \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} q_i$, and, by transitivity, $\widehat{q} \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_i$. We let $\mathbb{G}(\widehat{q}, \mathbf{s}_i)$-Spoiler choose transition $(\{\widehat{q}\}, \mathbf{s}_i, a_i, \{q'\}, \mathbf{s}') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}(\widehat{q}, \mathbf{s}_i)}$, and then we apply $\mathbb{G}(\widehat{q}, \mathbf{s}_i)$-Duplicator's winning strategy, obtaining transition $(\{\widehat{q}\}, \mathbf{s}_i, a_i, \{q'\}, \mathbf{s}', \{q'\}, \mathbf{s}_{i+1}) \in \mathit{\Gamma}^{\mathrm{Dup}}_{\mathbb{G}(\widehat{q}, \mathbf{s}_i)}$. Clearly, $q' \in [q_{i+1}]$, $q' \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_{i+1}$, and the invariant is preserved.

We define $\mathbb{G}$-Duplicator's response as

$$(\{[q_i]\}, \mathbf{s}_i, a_i, \{[q_{i+1}]\}, \mathbf{s}', \{[q_{i+1}]\}, \mathbf{s}_{i+1}) \in \mathit{\Gamma}^{\mathrm{Dup}}_{\mathbb{G}} \ .$$

Second case: $[q_i] \in U'$. By the definition of quotient, $q_i \in U$. Let Spoiler choose transition

$$(\{[q_i]\}, \mathbf{s}_i, a_i, \{\}, \mathbf{s}') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}} \ .$$

We let $\mathbb{G}_i$-Spoiler choose transition $(\{q_i\}, \mathbf{s}_i, a_i, \{\}, \mathbf{s}') \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}_i}$, and then we apply $\mathbb{G}_i$-Duplicator's winning strategy, obtaining $(\{q_i\}, \mathbf{s}_i, a_i, \{\}, \mathbf{s}', \{q_{i+1}\}, \mathbf{s}_{i+1}) \in \mathit{\Gamma}^{\mathrm{Dup}}_{\mathbb{G}_i}$. The crucial point is that we can assume w.l.o.g. that q_{i+1} is a n-$\forall$di-minimal a_i-successor of q_i. In particular, it is also k-$\forall$di-minimal. This implies that there exists a a_i-transition in the quotient automaton from $[q_i]$ to $[q_{i+1}]$. Thus, $\mathbb{G}$-Duplicator's response is defined as

$$(\{[q_i]\}, \mathbf{s}_i, a_i, \{\}, \mathbf{s}', \{[q_{i+1}]\}, \mathbf{s}_{i+1}) \in \mathit{\Gamma}^{\mathrm{Dup}}_{\mathbb{G}} \ .$$

Clearly, $q_{i+1} \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_{i+1}$, and the invariant is preserved also in this case. □

Lemma 13. *If $q \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}$, then $q \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} [\mathbf{s}]_{\mathrm{m}+}$, where the quotient is taken w.r.t. $\sqsubseteq^{\forall \mathrm{di}}_{(1,k)}$.*

Proof. Let $\mathbb{G} = \mathbb{G}(q, [\mathbf{s}])$ and, at round i, if the current configuration of $\mathbb{G}$ is $\langle q_i, [\mathbf{s}_i] \rangle$, let $\mathbb{G}_i = \mathbb{G}(q_i, \mathbf{s}_i)$. We maintain the following invariant: $q_i \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_i$. The invariant implies the lemma: if $q_i \in F$, then, by the definition of $\forall$di-simulation, $\mathbf{s}_i \subseteq F$, thence, by the definition of quotient, $[\mathbf{s}_i] \subseteq F'$.

Assume the current configuration in $\mathbb{G}$ is $\langle q_i, [\mathbf{s}_i] \rangle$, and $q_i \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_i$. Let Spoiler choose the next input symbol a_i and a transition

$$(\{q_i\}, [\mathbf{s}_i], a_i, \mathbf{q}', [\mathbf{s}']) \in \mathit{\Gamma}^{\mathrm{Sp}}_{\mathbb{G}} \ ,$$

where $[\mathbf{s}']$ is obtained by fixing a successor $[s']$ for any $[s^U] \in [\mathbf{s}_i] \cap U'$. (Notice that, if $q_i \in E$, then $\mathbf{q}' = \{q_{i+1}\}$ is just a singleton, for some $q_{i+1} \in \mathit{\Delta}(q_i, a)$, otherwise, when

$q_i \in U$, we have that $\mathbf{q}' = \{\}$ as *Left* universal pebbles are under Duplicator's control.) By the definition of minimax quotient, $([s^U], a_i, [s']) \in \Delta_{\mathrm{m}+}$ implies that there exist $\widehat{s}^U \in [s^U]$ and $\widehat{s}' \in [s']$ s.t. $\widehat{s}' \in \Delta(\widehat{s}^U, a_i)$. (*) Let $\mathbf{s}'$ be the set of states $\widehat{s}'$ obtained above. (We do not use the minimality of $\widehat{s}'$ in this proof.) For any mixed class $[s^U_{\mathrm{mix}}] \in [\mathbf{s}_i]$, for which its representative $s^U_{\mathrm{mix}} \in \mathbf{s}_i$ is universal, let $s^E_{\mathrm{mix}} \in [s^U_{\mathrm{mix}}] \cap E$ be a $(1,k)$-$\forall di$-equivalent existential representative, for which, in particular, $s^U_{\mathrm{mix}} \sqsubseteq^{\forall \mathrm{di}}_{(1,k)} s^E_{\mathrm{mix}}$. Let $\mathbf{p}_i$ be equal to $\mathbf{s}_i$, but where each s^U_{mix} is replaced by s^E_{mix}. By the invariant, $q_i \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_i$ and, by the definition of $\mathbf{p}_i$ and by transitivity, $q_i \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{p}_i$. The game then proceeds by using $\mathbf{p}_i$ in place of $\mathbf{s}_i$. Notice that universal states in $\mathbf{p}_i$ are exactly those universal states in $\mathbf{s}_i$ which belong to a purely universal quotient. We let $\mathbb{G}(q_i, \mathbf{p}_i)$-Spoiler choose transition $(\{q_i\}, \mathbf{p}_i, a_i, \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Sp}}_{\mathbb{G}(q_i,\mathbf{p}_i)}$, where $\mathbf{s}'$ is obtained from $\mathbf{p}_i \cap U \subseteq \mathbf{s}_i \cap U$ by fixing successors as prescribed in (*) above. We then apply $\mathbb{G}(q_i, \mathbf{p}_i)$-Duplicator's winning strategy, yielding transition $(\{q_i\}, \mathbf{p}_i, a_i, \mathbf{q}', \mathbf{s}', \{q_{i+1}\}, \mathbf{p}_{i+1}) \in \Gamma^{\mathrm{Dup}}_{\mathbb{G}(q_i,\mathbf{p}_i)}$. It might be the case that some transition $(p^E, a_i, p') \in \Delta$ induced above, for $p^E \in \mathbf{p}_i \cap E$ and $p' \in \mathbf{p}_{i+1}$, (if any) does not induce a transition in the quotient, i.e., there exists no corresponding transition $([p^E], a_i, [p']) \in \Delta_{\mathrm{m}+}$. This happens when, in definition of minimax+ quotient, p' is not selected as a k-maximal representative for a_i-successors of p^E. If this is the case, then, by the definition of minimax+ quotient, there exists some k-maximal a_i-successor $\mathbf{s}'' \subseteq \max^{k,\forall \mathrm{di}}_{a_i}(p^E)$ s.t. $p' \sqsubseteq^{\forall \mathrm{di}}_{(1,k)} \mathbf{s}''$ and, for all $s'' \in \mathbf{s}''$, $([p^E], a_i, [s'']) \in \Delta_{\mathrm{m}+}$. We define $\mathbf{s}_{i+1}$ as $\mathbf{p}_{i+1}$, where elements p' are replaced by $\mathbf{s}''$, as specified above. Then, $\mathbb{G}$-Duplicator's response is defined as

$$(\{q_i\}, [\mathbf{s}_i], a_i, \mathbf{q}', [\mathbf{s}'], \{q_{i+1}\}, [\mathbf{s}_{i+1}]) \in \Gamma^{\mathrm{Dup}}_{\mathbb{G}} .$$

Since $q_{i+1} \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{p}_{i+1}$, and, by the definition of $\mathbf{s}_{i+1}$ and transitivity, $q_{i+1} \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \mathbf{s}_{i+1}$, the invariant is preserved. ☐

Theorem 8. *$q \approx^{\forall \mathrm{di}}_{(1,n)} [q]_{\mathrm{m}+}$, where the quotient is taken w.r.t. $\sqsubseteq^{\forall \mathrm{di}}_{(1,k)}$. In particular, $\mathcal{L}^\omega(q) = \mathcal{L}^\omega([q]_{\mathrm{m}+})$.*

Proof. Since $q \sqsubseteq^{\forall \mathrm{di}}_{(1,n)} \{q\}$ trivially holds, the theorem follows from previous Lemma 12 and 13. ☐

D.3 Infinite words: Delayed simulation

Lemma 14. *Let $q, s \in U$. If $q \approx^x_{(1,n)} s$, then, for any $q' \in \min^{n,x}_a(q)$, there exists $s' \in \min^{n,x}_a(s)$ s.t. $q' \approx^x_{(1,n)} s'$.*

Proof. We actually prove the following richer statement.

Claim. Let $q, s \in U$ s.t. $q \approx^x_{(1,n)} s$. Then, for any $q' \in \min^{n,x}_a(q)$, 1) there exists $s' \in \Delta(s,a)$ s.t. $s' \sqsubseteq^x_{(1,n)} q'$, and, for any $s'' \in \Delta(s,a)$, $s'' \sqsubseteq^x_{(1,n)} q'$ implies both 2.1) $s'' \in \min^{n,x}_a(s)$, and 2.2) $q' \approx^x_{(1,n)} s''$.

Let $q \approx^x_{(1,n)} s$, and let $q' \in \min^{n,x}_a(q)$. Point 1) follows from the definition of simulation, i.e., there exists $s' \in \Delta(s,a)$ s.t. $s' \sqsubseteq^x_{(1,n)} q'$.

We now show Points 2.1) and 2.2), i.e., we show that any such s' is in fact an x-minimal a-successor of s. Let $s'' \in \Delta(s,a)$ be any other a-successor of s s.t. $s'' \sqsubseteq^x_{(1,n)} s'$. We have to show $s' \sqsubseteq^x_{(1,n)} s''$ as well. Since $q \sqsubseteq^x_{(1,n)} s$, from the definition of simulation, there exists $q'' \in \Delta(q,a)$ s.t. $q'' \sqsubseteq^x_{(1,n)} s''$. Hence, we have the following chain of inclusions: $q'' \sqsubseteq^x_{(1,n)} s'' \sqsubseteq^x_{(1,n)} s' \sqsubseteq^x_{(1,n)} q'$. By the transitivity of $\sqsubseteq^x_{(1,n)}$ established in Theorem 6, we have $q'' \sqsubseteq^x_{(1,n)} q'$, and, by the minimality of q', $q' \sqsubseteq^x_{(1,n)} q''$. By transitivity, all states in $\{q'', s'', s', q'\}$ are x-simulation equivalent. In particular, $s' \sqsubseteq^x_{(1,n)} s''$, which establishes Point 2.1), and $q' \sqsubseteq^x_{(1,n)} s'$, which establishes Point 2.2). $\square$

Lemma 15. *Let $s \in U$ and $q \in E$. If $q \approx^x_{(1,n)} s$, then there exists $\mathbf{q}' \subseteq \Delta(q,a)$ s.t., for any $s' \in \min^{n,x}_a(s)$, $s' \sqsubseteq^x_{(1,n)} \mathbf{q}'$.*

Proof. Let $s \in U$, $q \in E$, and $q \approx^x_{(1,n)} s$. From $s \sqsubseteq^x_{(1,n)} q$ and by the definition of simulation, there exists $\mathbf{q}' \subseteq \Delta(q,a)$ and $s'' \in \Delta(s,a)$ s.t. $s'' \sqsubseteq^x_{(1,n)} \mathbf{q}'$.

Let s' be any element in $\min^{n,x}_a(s)$. From $q \sqsubseteq^x_{(1,n)} s$ and by the definition of simulation, it follows that, for any $q' \in \mathbf{q}'$, we have $q' \sqsubseteq^x_{(1,n)} s'$. Since $s'' \sqsubseteq^x_{(1,n)} \mathbf{q}'$, and any element in $\mathbf{q}'$ is simulated by s', we obtain, by transitivity (Theorem 6), $s'' \sqsubseteq^x_{(1,n)} s'$, and, by the minimality of s', $s' \sqsubseteq^x_{(1,n)} s'' \sqsubseteq^x_{(1,n)} \mathbf{q}'$. By transitivity, $s' \sqsubseteq^x_{(1,n)} \mathbf{q}'$. $\square$

Lemma 16. *If $q \sqsubseteq^{\mathrm{de}}_{(1,n)} \mathbf{s}$, then $[q]_{\mathrm{se}+} \sqsubseteq^{\mathrm{de}}_{(1,n)} \mathbf{s}$, where the quotient is taken w.r.t. $\sqsubseteq^{\mathrm{de}}_{(1,n)}$.*

Proof. In the following, we simply write $\sqsubseteq$ instead of $\sqsubseteq^{\mathrm{de}}_{(1,n)}$. Then, when we write $q \sqsubseteq_\sigma \mathbf{s}$, we mean that σ is Duplicator's winning strategy in $\mathbb{G}(q, \mathbf{s})$, i.e., the one witnessing $q \sqsubseteq \mathbf{s}$. In the proof, we need the following definitions: For any Duplicator's strategy $\sigma : PP_1 \mapsto (P_0 \mapsto P_1)$ and for any $\pi \in P$, we define a new Duplicator's strategy σ^π in the following way: For any $\pi' \in PP_1$, $\sigma^\pi(\pi') := \sigma(\pi \cdot \pi')$. Given any Duplicator's strategy σ, we say that a sequence $\pi^1 = \mathbf{s}_k \mathbf{s}_{k+1} \dots$ is *σ-right-conform starting at p_k* iff there exist sequences $\pi^0 = p_k p_{k+1} \dots$ and $w = a_k a_{k+1} \dots$ s.t. $\pi = \langle p_k, \mathbf{s}_k\rangle \langle p_{k+1}, \mathbf{s}_{k+1}\rangle \dots$ is σ-conform w.r.t. w. We will use the following fact:

Claim. Assume $\pi^1 = \mathbf{s}_k \mathbf{s}_{k+1} \dots$ is σ-right-conform starting at p_k. If $p_k \in F$ and σ is a winning strategy, then there exists $i \geq k$ s.t. $\mathrm{good}^\exists(\mathbf{s}_i, k)$.

We are now ready for proving the lemma. Let $\mathbb{G} = \mathbb{G}([q], \mathbf{s})$ and, at round k, if the current configuration of $\mathbb{G}$ is $\langle [q_k], \mathbf{s}_k\rangle$, let $\mathbb{G}_k = \mathbb{G}(q_k, \mathbf{s}_k)$. We build a sequence of winning strategies $\sigma_0, \sigma_1, \dots$, s.t., at round k, σ_k is a winning strategy in $\mathbb{G}_k$, i.e., $q_k \sqsubseteq_{\sigma_k} \mathbf{s}_k$. Then, we define a strategy σ for Duplicator in $\mathbb{G}$, which, at round k, is defined in terms of σ_k. Finally, we prove that σ is winning.

Assume the current configuration in $\mathbb{G}$ is $\langle [q_k], \mathbf{s}_k\rangle$, and that σ_k is a winning strategy in $\mathbb{G}_k$ s.t. $q_k \sqsubseteq_{\sigma_k} \mathbf{s}_k$. Let Spoiler choose the next input symbol a_k. We consider two cases, depending on whether $[q_k]$ is existential or universal.

First case: $[q_k] \in E'$. Let Spoiler choose an a_k-successor $[q_{k+1}]$ of $[q_k]$, i.e., Spoiler chooses transition

$$(\{[q_k]\}, \mathbf{s}_k, a_k, \{[q_{k+1}]\}, \mathbf{s}') \in \Gamma^{\mathrm{Sp}}_{\mathbb{G}} .$$

By the definition of semielective quotient, there exist $\widehat{q} \in [q_k]$ and $q' \in [q_{k+1}]$ s.t. $q' \in \Delta(\widehat{q}, a)$. If $[q_k] \cap F \neq \emptyset$, then let q^F be any accepting state in $[q_k]$, otherwise let q^F be just $\widehat{q}$. We distinguish two subcases, depending on whether $\widehat{q}$ is in E or in U.

– First subcase: $\widehat{q} \in E$. We have that

$$\widehat{q} \sqsubseteq_{\widehat{\sigma}} q^F \sqsubseteq_{\sigma^F} q_k \sqsubseteq_{\sigma_k} \mathbf{s}_k \ .$$

We let $\mathbb{G}(\widehat{q}, \mathbf{s}_k)$-Spoiler choose transition $(\{\widehat{q}\}, \mathbf{s}_k, a_k, \{q'\}, \mathbf{s}') \in \Gamma^{\mathrm{Sp}}_{\mathbb{G}(\widehat{q}, \mathbf{s}_k)}$. Let $\bar{\sigma} = \widehat{\sigma} \bowtie \sigma^F \bowtie \sigma_k$, and let $\mathbf{s}_{k+1}$ be the result of $\mathbb{G}(\widehat{q}, \mathbf{s}_k)$-Duplicator playing according to $\bar{\sigma}$, i.e., $\bar{\sigma}(\{\widehat{q}\}, \mathbf{s}_k)(a_k, \{q'\}, \mathbf{s}') = (\{q'\}, \mathbf{s}_{k+1})$. Clearly, $q' \sqsubseteq_{\bar{\sigma}^\pi} \mathbf{s}_{k+1}$, with $\pi = \langle \widehat{q}, \mathbf{s}_k \rangle$. But $q' \in [q_{k+1}]$, thus $q_{k+1} \sqsubseteq_{\sigma'} q' \sqsubseteq_{\bar{\sigma}^\pi} \mathbf{s}_{k+1}$ for some σ'. By transitivity, $q_{k+1} \sqsubseteq_{\sigma' \bowtie \bar{\sigma}^\pi} \mathbf{s}_{k+1}$. We let $\sigma_{k+1} := \sigma' \bowtie \bar{\sigma}^\pi$.

– Second subcase: $\widehat{q} \in U$. By the definition of semielective quotient, $q' \in \min_a^{n,\mathrm{de}}(\widehat{q})$. (Notice that, although $\widehat{q}$ is a universal state in this case, it is still Spoiler who has to choose a successor q' of $\widehat{q}$, since $[q_k]$ is an existential state in the quotient automaton). Since $[q_k]$ is a mixed class, there exists $q^E \in [q_i] \cap E$ s.t.

$$\widehat{q} \sqsubseteq_{\widehat{\sigma}} q^E \sqsubseteq_{\sigma^E} q^F \sqsubseteq_{\sigma^F} q_k \sqsubseteq_{\sigma_k} \mathbf{s}_k \ .$$

Since $\widehat{q} \sqsubseteq q^E$, by the minimality of q' and Point 2) of Lemma 15, there exists $\mathbf{q}' \subseteq \Delta(q^E, a)$ s.t. $q' \sqsubseteq \mathbf{q}'$. Hence, w.l.o.g. $\widehat{\sigma}$ can be taken s.t. $\widehat{\sigma}(\{\widehat{q}\}, \{q^E\})(a_k, \{\}, \{\}) = (\{q'\}, \mathbf{q}')$, where q' is fixed by $\mathbb{G}([q], \mathbf{s})$-Spoiler, and not under the control of $\mathbb{G}(\widehat{q}, q^E)$-Duplicator.
Let $\bar{\sigma} := \widehat{\sigma} \bowtie \sigma^E \bowtie \sigma^F \bowtie \sigma_k$. Similarly to the previous point, $\mathbb{G}(\widehat{q}, \mathbf{s}_k)$-Spoiler chooses a transition $(\{\widehat{q}\}, \mathbf{s}_k, a_k, \{\}, \mathbf{s}') \in \Gamma^{\mathrm{Sp}}$. We let $\mathbb{G}(\widehat{q}, \mathbf{s}_k)$-Duplicator answer with $\bar{\sigma}(\{\widehat{q}\}, \mathbf{s}_k)(a_k, \{\}, \mathbf{s}') = (\{q'\}, \mathbf{s}_{k+1})$, where q' is the a a_k-successor fixed by $\mathbb{G}([q], \mathbf{s})$-Spoiler above. As before, $q_{k+1} \sqsubseteq_{\sigma'} q' \sqsubseteq_{\bar{\sigma}^\pi} \mathbf{s}_{k+1}$, where $\pi = \langle \widehat{q}, \mathbf{s}_k \rangle$. Hence, by transitivity, $q_{k+1} \sqsubseteq_{\sigma' \bowtie \bar{\sigma}} \mathbf{s}_{k+1}$. We let $\sigma_{k+1} := \sigma' \bowtie \bar{\sigma}^\pi$.

In both cases, $q_{k+1} \sqsubseteq_{\sigma_{k+1}} \mathbf{s}_{k+1}$. We define $\mathbb{G}$-Duplicator's winning strategy σ as

$$\sigma(\pi_k \langle \{[q_k]\}, \mathbf{s}_k \rangle)(a_k, \{[q_{k+1}]\}, \mathbf{s}') = (\{[q_{k+1}]\}, \mathbf{s}_{k+1}) \ .$$

Second case: $[q_k] \in U'$. By the definition of quotient, $q_k \in U$. Let Spoiler choose transition

$$(\{[q_k]\}, \mathbf{s}_k, a_k, \{\}, \mathbf{s}') \in \Gamma^{\mathrm{Sp}}_{\mathbb{G}} \ .$$

If $[q_k] \cap F \neq \emptyset$, let $q^F \in U$ be any accepting state in $[q_k]$, otherwise let q^F be just q_k. Then, we have

$$q^F \sqsubseteq_{\sigma^F} q_k \sqsubseteq_{\sigma_k} \mathbf{s}_k \ .$$

Let $\bar{\sigma} = \sigma^F \bowtie \sigma_k$. Let $\mathbb{G}(q^F, \mathbf{s}_k)$-Spoiler choose transition $(\{q^F\}, \mathbf{s}_k, a_k, \{\}, \mathbf{s}') \in \Gamma^{\mathrm{Sp}}_{\mathbb{G}(q^F, \mathbf{s}_k)}$, and let $\mathbb{G}(q^F, \mathbf{s}_k)$-Duplicator choose transition $(\{q^F\}, \mathbf{s}_k, a_k, \{\}, \mathbf{s}', \{q_{k+1}\}, \mathbf{s}_{k+1}) \in \Gamma^{\mathrm{Dup}}_{\mathbb{G}(q^F, \mathbf{s}_k)}$ according to $\bar{\sigma}$. We let $\sigma_{k+1} := \bar{\sigma}^\pi$, with $\pi = \langle \{q^F\}, \mathbf{s}_k \rangle$. The crucial point is that we can assume w.l.o.g. that q_{k+1} is a de-minimal a_k-successor of q^F. This implies that there exists a a_k-transition in the quotient automaton from $[q_k]$ to $[q_{k+1}]$. Thus, $\mathbb{G}$-Duplicator's response is defined as

$$\sigma(\pi_k\langle\{[q_k]\}, \mathbf{s}_k\rangle)(a_k, \{\}, \mathbf{s}') = (\{[q_{k+1}]\}, \mathbf{s}_{k+1}) .$$

This concludes the description of the second case.

We now argue about the correctness of the construction above, showing that Duplicator's strategy is winning in $\mathbb{G}$. If the *Left* pebble in $\mathbb{G}$ gets stuck, then Duplicator wins, and we are done. Otherwise, assume the *Left* pebble never gets stuck. By construction, since we are taking joins of winning strategies, it follows that some *Right* pebble can always be moved, and the game does not halt prematurely. Thus, an infinite path $\pi = \langle[q_0], \mathbf{s}_0\rangle\langle[q_1], \mathbf{s}_1\rangle\ldots$ results, where $q_0 = q$ and $\mathbf{s}_0 = \mathbf{s}$. Assume $[q_k] \in F'$, for some k. There exists $q^F \in [q_k]$ s.t. $q^F \in F$ and, in any of the cases above, there exists a winning strategy σ^F s.t. $q^F \sqsubseteq_{\sigma^F} q_k \sqsubseteq_{\sigma_k} \mathbf{s}_k$. Let $\widetilde{\sigma} := \sigma^F \bowtie \sigma_k$ be a winning strategy in $\mathbb{G}(q^F, \mathbf{s}_k)$. By construction, the sequence $\pi^1 = \mathbf{s}_k\mathbf{s}_{k+1}\ldots$ is $\widetilde{\sigma}$-right-conform starting at q^F. By the above claim, there exists $i \geq k$ s.t. $\mathrm{good}^{\exists}(\mathbf{s}_i, k)$. ☐

Corollary 1. $[q]_{\mathrm{se}+} \sqsubseteq^{\mathrm{de}}_{(1,n)} q$.

The lemma below implies $q \sqsubseteq^{\mathrm{de}}_{(1,1)} [q]_{\mathrm{se}+}$. Notice that we actually prove the much stronger claim that $[q]_{\mathrm{se}+}$ *di*-simulates q.

Lemma 17. *For any* $q \in Q$, $q \sqsubseteq^{\mathrm{di}}_{(1,1)} [q]_{\mathrm{se}+}$.

Proof. We maintain the following invariant: If $(s_k, [q_k])$ is the current configuration in $\mathbb{G}^{\mathrm{di}}_{(1,1)}(q, [q])$, then $s_k \in [q_k]$. Clearly, the invariant implies that the winning condition for direct simulation is satisfied: If $s_k \in F$, then $[q_k] \in F'$.

The initial configuration is $(s_0, [q_0])$ with $s_0 = q$, and $[q_0] = [q]$, and the invariant clearly holds since $s_0 \in [q_0]$.

Inductively, assume the current configuration is $(s_k, [q_k])$ and the invariant $s_k \in [q_k]$ holds. We distinguish three different cases.

- Case 1: $s_k \in E$. Then $[q_k] \in E'$. Assume Spoiler chooses transition

  $$(\{s_k\}, \{[q_k]\}, a_k, \{s_{k+1}\}, \{\}) \in \Gamma^{\mathrm{Sp}} .$$

  From $(s_k, a_k, s_{k+1}) \in \Delta$, the invariant $s_k \in [q_k]$ and by the definition of semielective quotient, there exists a transition $([q_k], a_k, [s_{k+1}]) \in \Delta^{\mathrm{se}+}_{\approx}$. Thus, Duplicator can select transition

  $$(\{s_k\}, \{[q_k]\}, a_k, \{s_{k+1}\}, \{\}, \{s_{k+1}\}, \{[s_{k+1}]\}) \in \Gamma^{\mathrm{Dup}} .$$

  Clearly $s_{k+1} \in [s_{k+1}]$, and the invariant is preserved.
- Case 2: $s_k \in U$ and $[q_k] \in E'$. In this case, Spoiler only chooses a_k:

  $$(\{s_k\}, \{[q_k]\}, a_k, \{\}, \{\}) \in \Gamma^{\mathrm{Sp}} .$$

If s_k has no a_k-successor, then Duplicator wins. Otherwise, let $s_{k+1} \in \min_{a_k}^{n,\mathrm{de}}(s_k)$ be a de-minimal a_k-successor of s_k. By the definition of semielective quotient and by the minimality of s_{k+1}, there exists a transition $([q_k], a_k, [s_{k+1}]) \in \Delta_{\approx}^{\mathrm{se}+}$, thus

$$(\{s_k\}, \{[q_k]\}, a_k, \{\}, \{\}, \{s_{k+1}\}, \{[s_{k+1}]\}) \in \Gamma^{\mathrm{Dup}} \ .$$

Clearly $s_{k+1} \in [s_{k+1}]$, and the invariant is preserved.

– Case 3: $s_k \in U$ and $[q_k] \in U'$. In this case, we use the minimality of successors of universal states in universal classes. Assume Spoiler chooses transition

$$(\{s_k\}, \{[q_k]\}, a_k, \{\}, \{[q_{k+1}]\}) \in \Gamma^{\mathrm{Sp}} \ .$$

From the definition of quotient, there exists a transition $(q_k, a_k, q_{k+1}) \in \Delta$ s.t. $q_{k+1} \in \min_{a_k}^{n,\mathrm{de}}(q_k)$. From the invariant $s_k \in [q_k]$, we have $s_k \approx_{(1,n)}^{\mathrm{de}} q_k$. By Lemma 14, there exists $s_{k+1} \in \min_{a_k}^{n,\mathrm{de}}(s_k)$ s.t. $s_{k+1} \approx_{(1,n)}^{\mathrm{de}} q_{k+1}$. Therefore, Duplicator can select transition

$$(\{s_k\}, \{[q_k]\}, a_k, \{\}, \{[q_{k+1}]\}, \{s_{k+1}\}, \{[q_{k+1}]\}) \in \Gamma^{\mathrm{Dup}}$$

s.t. $s_{k+1} \in [q_{k+1}]$, thus preserving the invariant. □

Remark 9. Lemma 17 above is even true when quotienting w.r.t. fair simulation, or even ordinary simulation. Notice that requiring minimal transitions from universal states in mixed semielective-classes not only is required for correctness (see Section 5.2), but it also makes the proof much easier.

Theorem 9. *$q \approx_{(1,n)}^{\mathrm{de}} [q]_{\mathrm{se}+}$, where the quotient is taken w.r.t. $\sqsubseteq_{(1,n)}^{\mathrm{de}}$. In particular, $\mathcal{L}^\omega(q) = \mathcal{L}^\omega([q]_{\mathrm{se}+})$.*

Proof. Directly from Corollary 1 and Lemma 17, and from the fact that simulation implies language inclusion (Theorem 4). □

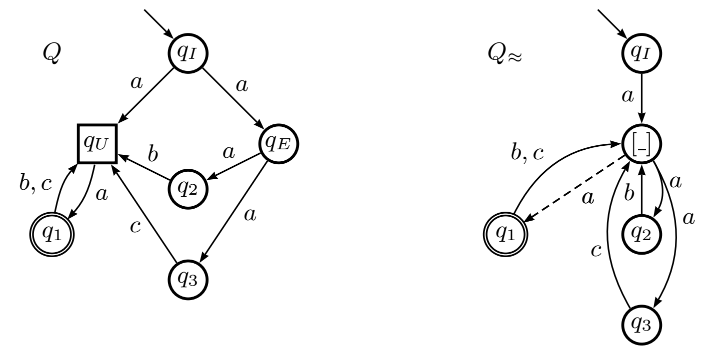


Fig. 5. An example showing that (minimal) transitions from universal states in mixed classes are needed in semielective quotients. The only two $(1, n)$-simulation equivalent states in Q are q_U and q_E. (In fact, $n = 2$ suffices.) The resulting mixed class in $Q_\approx$ is $[\_] = \{q_U, q_E\}$. The dashed a-transition on the right (due to q_U) is needed and cannot be discarded: Indeed, $\mathcal{L}^\omega(Q) \neq \emptyset$, while removing the dashed transition from $Q_\approx$ would make $\mathcal{L}^\omega(Q_\approx) = \emptyset$.

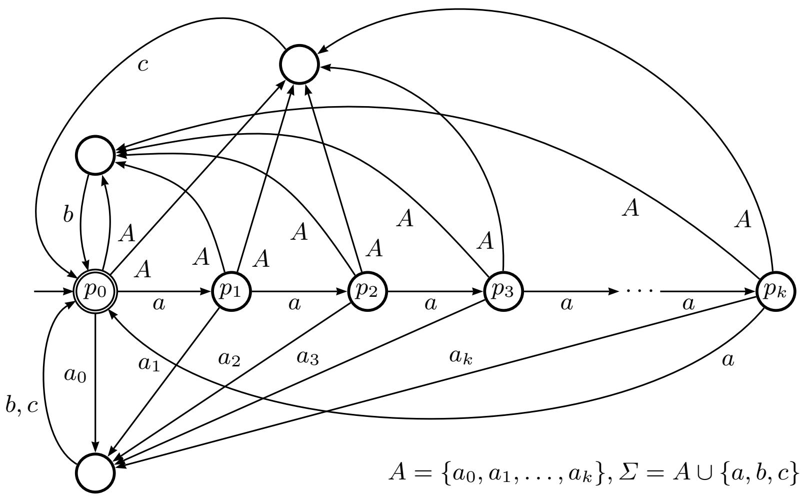


Fig. 6. An example showing that multipebble-semielective quotients can achieve arbitrarily high compression ratios. The NBA above has $k + 4$ states, and the p_i's are $(1, 1)$-delayed simulation incomparable: Thus, the $(1, 1)$-semielective quotient has $k + 4$ states. However, the p_i's are all $(1, n)$-delayed simulation equivalent (and $n = 2$ suffices), therefore the $(1, n)$-semielective quotient has only 4 states. Moreover, the p_i's are incomparable also w.r.t. $(1, n)$-*universal direct* simulation, which shows that semielective quotients can achieve arbitrarily high compression ratios relative to minimax quotients. Finally, notice that direct backward simulation does not help either: In fact, any two p_i, p_j, with $i \neq j$, are backward-simulation incomparable, as there is just one way of backward reaching the unique initial state p_0. (Remember that backward simulations should be compatible with the initial states, at least.) Therefore, quotienting methods which employ backward simulations, like *mediated preorder* [1], do not result in a smaller automaton.

E Section 6

We give upper-bounds on the size of game-graphs necessary for computing multipebble simulations. When considering the size of the those game-graphs, we will make use of the following counting function:

$$\mathtt{sub}_n(k) = \sum_{i=0}^{k} \binom{n}{i} ,$$

which counts the number of subsets of size $\leq k$ of a given set of size n, and we will approximate its value from above by using the following rough upper bound

$$\mathtt{sub}_n(k) \leq (n+1)^k .$$

Intuitively, the bound above may be seen as follows: Instead of counting sets of size $\leq k$, one counts *ordered* sets; each ordered set can be represented as a k-string over an alphabet of size $n+1$, where we use an extra end-of-string symbol. We also give a formal calculation.

$$\begin{aligned}
\sum_{i=0}^{k} \binom{n}{i} &= \sum_{i=0}^{k} \left(\frac{n}{i}\right)\left(\frac{n-1}{i-1}\right)\cdots\left(\frac{n-i+1}{1}\right) \\
&\leq \sum_{i=0}^{k} \left(\frac{n}{i}\right)\left(\frac{n}{i-1}\right)\cdots\left(\frac{n}{1}\right) \\
&\leq \sum_{i=0}^{k} n^i \cdot \left(\frac{k}{i}\right)\left(\frac{k-1}{i-1}\right)\cdots\left(\frac{k-i+1}{1}\right) \\
&= \sum_{i=0}^{k} \binom{k}{i} \cdot n^i = (n+1)^k .
\end{aligned}$$

E.1 Solving existential and universal direct simulation

For computing the winner for direct simulation, we construct a 2-player game $\mathcal{G}^{\mathrm{di}} = \langle V_{\mathrm{E}}^{\mathrm{di}}, V_{\mathrm{A}}^{\mathrm{di}}, \rightarrow^{\mathrm{di}} \rangle$, where Eve has a safety objective. The game-graph is the same for both existential and universal direct simulation, but the safety objective is different. Nodes in $V_{\mathrm{A}}^{\mathrm{di}}$ take the form $v_{(\mathbf{q},\mathbf{s})}$, while nodes in $V_{\mathrm{E}}^{\mathrm{di}}$ take the form $v_{(\mathbf{q},\mathbf{s},a,\mathbf{q}',\mathbf{s}')}$, with $\mathbf{q}, \mathbf{q}' \in 2^{Q,k_1}$ and $\mathbf{s}, \mathbf{s}' \in 2^{Q,k_2}$, and $a \in \Sigma$.

Lemma 18. $\left|V^{\mathrm{di}}\right| \leq 2 \cdot (n+1)^{2(k_1+k_2)} \cdot |\Sigma|$.

Proof. $\left|V_{\mathrm{A}}^{\mathrm{di}}\right| = \mathtt{sub}_n(k_1) \cdot \mathtt{sub}_n(k_2)$, and $\left|V_{\mathrm{E}}^{\mathrm{di}}\right| \leq [\mathtt{sub}_n(k_1)]^2 \cdot [\mathtt{sub}_n(k_2)]^2 \cdot |\Sigma|$. Hence, $\left|V_{\mathrm{A}}^{\mathrm{di}} \cup V_{\mathrm{E}}^{\mathrm{di}}\right| \leq 2 \cdot [\mathtt{sub}_n(k_1)]^2 \cdot [\mathtt{sub}_n(k_2)]^2 \cdot |\Sigma| \leq 2 \cdot (n+1)^{2(k_1+k_2)} \cdot |\Sigma|$. □

Transitions in $V_{\mathrm{A}}^{\mathrm{di}} \times V_{\mathrm{E}}^{\mathrm{di}}$ model choices of Spoiler: For any $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Sp}}$, there is a transition for Adam $v_{(\mathbf{q},\mathbf{s})} \rightarrow v_{(\mathbf{q},\mathbf{s},a,\mathbf{q}',\mathbf{s}')}$. Similarly, for any Duplicator's move $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}', \mathbf{s}', \mathbf{q}'', \mathbf{s}'') \in \Gamma^{\mathrm{Dup}}$, there exists a transition for Eve $v_{(\mathbf{q},\mathbf{s},a,\mathbf{q}',\mathbf{s}')} \to v_{(\mathbf{q}'',\mathbf{s}'')}$ in $V^{\mathrm{di}}_{\mathrm{E}} \times V^{\mathrm{di}}_{\mathrm{A}}$.

The winning criterion for existential direct simulation induces a set $T^{\exists}$ of *safe vertices* $T^{\exists} = \{v_{(\mathbf{q},\mathbf{s})} \in V^{\mathrm{di}}_{\mathrm{A}} \mid \mathbf{q} \subseteq F \text{ implies } \mathbf{s} \cap F \neq \emptyset\}$. Similarly, the safe set for universal direct simulation is $T^{\forall} = \{v_{(\mathbf{q},\mathbf{s})} \in V^{\mathrm{di}}_{\mathrm{A}} \mid \mathbf{q} \cap F \neq \emptyset \text{ implies } \mathbf{s} \subseteq F\}$. For $x \in \{\exists, \forall\}$, we have that $\mathbf{q} \sqsubseteq^{x\mathrm{di}}_{(k_1,k_2)} \mathbf{s}$ iff Eve can ensure never leaving T^x when starting from $v_{(\mathbf{q},\mathbf{s})}$. This can be verified by checking whether $v_{(\mathbf{q},\mathbf{s})} \in W^{x\mathrm{di}}$, where

$$W^{x\mathrm{di}} = \nu \mathbf{y}.\, T^x \cap \mathrm{cpre}(\mathbf{y}) \ .$$

E.2 Solving fair simulation

The game-graph for fair simulation is similiar to the previous one for direct simulation, but with the difference that we need extra bookkeeping for recording whether each pebble has visited an accepting state or not. We let $\mathcal{G}^{\mathrm{f}} = \langle V^{\mathrm{f}}_{\mathrm{E}}, V^{\mathrm{f}}_{\mathrm{A}}, \to^{\mathrm{f}} \rangle$, where Adam's nodes in $V^{\mathrm{f}}_{\mathrm{A}}$ are of the form $v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good})}$ and Eve's nodes in $V^{\mathrm{f}}_{\mathrm{E}}$ are of the form $v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good},a,\mathbf{q}',\mathbf{s}')}$, where $\mathbf{q}, \mathbf{q}', \mathbf{s}, \mathbf{s}' \subseteq Q$. The sets $\mathsf{bad} \subseteq \mathbf{q}$ and $\mathsf{good} \subseteq \mathbf{s}$ record the current "badness/goodness" of states in $\mathbf{q}$ and $\mathbf{s}$, respectively, and they are used to detect events like "being good since some previous round": Specifically, the event $\mathsf{bad} = \emptyset$ is used to detect when $\mathrm{good}^{\forall}(\mathbf{q})$, and similarly for $\mathsf{good} = \mathbf{s}$ and $\mathrm{good}^{\exists}(\mathbf{s})$.

Lemma 19. $\left|V^{\mathrm{f}}\right| \leq 2 \cdot (n+1)^{2(k_1+k_2)} \cdot 2^{k_1+k_2} \cdot |\Sigma|$.

Proof. $\left|V^{\mathrm{f}}_{\mathrm{A}}\right| = \mathtt{sub}_n(k_1) \cdot \mathtt{sub}_n(k_2) \cdot 2^{k_1+k_2}$, and $\left|V^{\mathrm{f}}_{\mathrm{E}}\right| \leq [\mathtt{sub}_n(k_1) \cdot \mathtt{sub}_n(k_2)]^2 \cdot 2^{k_1+k_2} \cdot |\Sigma|$. Hence, $\left|V^{\mathrm{f}}_{\mathrm{A}} \cup V^{\mathrm{f}}_{\mathrm{E}}\right| \leq 2 \cdot [\mathtt{sub}_n(k_1) \cdot \mathtt{sub}_n(k_2)]^2 \cdot 2^{k_1+k_2} \cdot |\Sigma| \leq 2 \cdot (n+1)^{2(k_1+k_2)} \cdot 2^{k_1+k_2} \cdot |\Sigma|$. ☐

For any $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}'', \mathbf{s}'') \in \Gamma^{\mathrm{Sp}}$, $v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good})} \to v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good},a,\mathbf{q}'',\mathbf{s}'')}$, and, for any $(\mathbf{q}, \mathbf{s}, a, \mathbf{q}'', \mathbf{s}'', \mathbf{q}', \mathbf{s}') \in \Gamma^{\mathrm{Dup}}$, $v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good},a,\mathbf{q}'',\mathbf{s}'')} \to v_{(\mathbf{q}',\mathsf{bad}',\mathbf{s}',\mathsf{good}')}$, where

$$\mathsf{good}' = \begin{cases} \mathbf{s}' \cap F & \text{if } \mathsf{good} = \mathbf{s} \\ \{s \in \mathbf{s}' \mid s \in F \vee s \in \Delta(\mathsf{good}, a)\} & \text{otherwise} \end{cases}$$
$$\mathsf{bad}' = \begin{cases} \mathbf{q}' \setminus F & \text{if } \mathsf{bad} = \emptyset \\ \{q \in \mathbf{q}' \mid q \notin F \wedge q \in \Delta(\mathsf{bad}, a)\} & \text{otherwise} \end{cases}$$

We notice the striking similarity of the update rule for bad pebbles and the updating rule for the second component in the MH-construction (Section 4). Intuitively, states in bad' are those states in $\mathbf{q}'$ which are not accepting and with some bad predecessor. Similarly, states in good' are those states in $\mathbf{s}'$ which are either accepting, or with some good predecessor. The correctness follows from the following simple fact:

Claim. Let $\pi = v_0 v_1 \ldots$ be an infinite sequence of vertices, with $v_i = v_{(\mathbf{q}_i,\mathsf{bad}_i,\mathbf{s}_i,\mathsf{good}_i)}$ and s.t. $v_i \to v_{i+1}$. Let $\pi_1 = \mathbf{q}_0 \mathbf{q}_1 \ldots$ and $\pi_3 = \mathbf{s}_0 \mathbf{s}_1 \ldots$ be the projections of π to the first and third component, respectively. Then, $\mathrm{good}^{\forall}(\pi_1, \infty)$ iff $\mathsf{bad}_i = \emptyset$ for infinitely many i's, and $\mathrm{good}^{\exists}(\pi_3, \infty)$ iff $\mathsf{good}_i = \mathbf{s}_i$ for infinitely many i's.

Let T_1 be the set of states of the form $v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good})}$ with $\mathsf{bad} = \emptyset$, and let T_2 be the set of states of the form $v_{(\mathbf{q},\mathsf{bad},\mathbf{s},\mathsf{good})}$ with $\mathsf{good} = \mathbf{s}$. The winning criterion for fair simulation is translated in the following 1-pair Street condition (also known as a *reactivity condition* [9]): If T_1 is visited infinitely often, then T_2 is visited infinitely often. Therefore, winning nodes for Eve are those in

$$W^{\mathrm{f}} = \nu\mathbf{x} \,.\, \mu\mathbf{y} \,.\, \nu\mathbf{z} \,.\, \big(T_2 \cap \mathrm{cpre}(\mathbf{x}) \cup T_1 \cap \mathrm{cpre}(\mathbf{y}) \cup \overline{T}_1 \cap \mathrm{cpre}(\mathbf{z})\big) \,.$$

E.3 Solving delayed simulation

We recall the definition of the game-graph for computing delayed simuation: $\mathcal{G}^{\mathrm{de}} = \langle V_{\mathrm{E}}^{\mathrm{de}}, V_{\mathrm{A}}^{\mathrm{de}}, \to^{\mathrm{de}} \rangle$, where

$$V_{\mathrm{A}}^{\mathrm{de}} = \{v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good})} \mid \mathbf{q}, \mathbf{s} \subseteq Q\}$$
$$V_{\mathrm{E}}^{\mathrm{de}} = \{v_{(\mathbf{q},\mathsf{Bad},\mathbf{s},\mathsf{Good},a,\mathbf{q}',\mathbf{s}')} \mid \mathbf{q}, \mathbf{q}', \mathbf{s}, \mathbf{s}' \subseteq Q\}$$

and $\mathsf{Bad}, \mathsf{Good}$ are two sequences of sets of states from Q, strictly ordered by set-inclusion. More precisely, $\mathsf{Bad} = \langle \mathbf{b}_1, \ldots, \mathbf{b}_{m_1} \rangle$ with $0 \le m_1 \le k_1$, satisfies, for any $i \in \{1, \ldots, m_1\}$,

$$\mathbf{b}_i \subseteq \mathbf{q} \tag{B1}$$
$$\mathbf{b}_{i+1} \subset \mathbf{b}_i, \text{ when } i < m_1 \tag{B2}$$
$$\mathbf{b}_{m_1} \neq \emptyset \,, \tag{B3}$$

and $\mathsf{Good} = \langle \mathbf{g}_1, \ldots, \mathbf{g}_{m_2} \rangle$ with $0 \le m_2 \le k_2$, satisfies, for any $i \in \{1, \ldots, m_2\}$,

$$\mathbf{g}_i \subseteq \mathbf{s} \tag{G1}$$
$$\mathbf{g}_i \subset \mathbf{g}_{i+1}, \text{ when } i < m_2 \tag{G2}$$

We also denote with Bad the set $\{\mathbf{b}_1, \ldots, \mathbf{b}_{m_1}\}$, and similarly for Good.

The following lemma states that Algorithm 1 preserves the definition of $\mathsf{Bad}, \mathsf{Good}$:

Lemma 20. *If* Bad *and* Good *satisfy properties (B1)-(B3) and (G1)-(G2), resp., then the same holds for the sets* Bad' *and* Good' *as computed by Algorithm 1.*

Proof. Properties (B1) and (G1) are preserved by how $\mathbf{b}'$ and $\mathbf{g}'$ are constructed, on lines 4 and 15, respectively. Similarly, the strictess of the order, i.e., (B2) and (G2), is preserved by removing duplicate elements (lines 6 and 17). Finally, property (B3) follows by the check at line 8, which enforces that empty elements are removed from Bad', if any (line 9). □

Lemma 7. $\left|V^{\mathrm{de}}\right| \le 2 \cdot (n+1)^{2(k_1+k_2)} \cdot \big(1 + (k_1+1)^{k_1+1}\big) \cdot \big(1 + 2(k_2+1)^{k_2+1}\big) \cdot |\Sigma|$.

Proof. We first count the number of pairs $(\mathbf{s}, \mathsf{Good})$. Assume $|\mathbf{g}_{m_2}| = h \le k_2$ (notice that $m_2 \le h$). We consider two cases, depending on whether $\mathbf{g}_1 \neq \emptyset$ or not. First case: $\mathbf{g}_1 \neq \emptyset$. Then, we can represent the strictly increasing sequence $\mathbf{g}_1 \subset \mathbf{g}_2 \subset \cdots \subset \mathbf{g}_{m_2}$ by the sequence $\{\mathbf{d}_i\}_{1 \le i \le m_2}$ of *non-empty* differences, defined as $\mathbf{d}_1 = \mathbf{g}_1$ and

Algorithm 1: *Updating the sequences* Good *and* Bad

Input: The sequences $\mathsf{Good} = \langle \mathbf{g}_1 \subset \cdots \subset \mathbf{g}_{m_2} \rangle$ and $\mathsf{Bad} = \langle \mathbf{b}_1 \supset \cdots \supset \mathbf{b}_{m_1} \rangle$ to be updated, the current input symbol $a \in \Sigma$ and the next configuration $\langle \mathbf{q}', \mathbf{s}' \rangle$.

Output: The updated sequences Good' and Bad'

Add $\mathbf{q}$ to Bad, i.e.,

$$\mathsf{Bad} = \langle \mathbf{q} \supseteq \mathbf{b}_1 \supset \cdots \supset \mathbf{b}_{m_1} \rangle$$

$\mathsf{Bad}' = \langle \rangle$;

foreach $(\mathbf{b} \in \mathsf{Bad})$ **do**

$$\mathbf{b}' = \{ \, s \in \mathbf{q}' \mid s \notin F \wedge s \in \Delta(\mathbf{b}, a) \, \}$$

  Add $\mathbf{b}'$ to Bad';

Remove duplicate elements from Bad';

Assume $\mathsf{Bad}' = \langle \mathbf{b}'_1 \supset \cdots \supset \mathbf{b}'_{m'_1} \rangle$;

if $(\mathbf{b}'_{m'_1} = \emptyset)$ **then**

  Remove $\mathbf{b}'_{m'_1}$ from Bad';

  Add $\emptyset$ to the front of Good, i.e.,

$$\mathsf{Good} = \langle \emptyset \subseteq \mathbf{g}_1 \subset \cdots \subset \mathbf{g}_{m_2} \rangle$$

if $(\mathbf{g}_{m_2} = \mathbf{s})$ **then**

  Remove $\mathbf{g}_{m_2}$ from Good;

$\mathsf{Good}' = \langle \rangle$;

foreach $(\mathbf{g} \in \mathsf{Good})$ **do**

$$\mathbf{g}' = \{ \, s \in \mathbf{s}' \mid s \in F \vee s \in \Delta(\mathbf{g}, a) \, \}$$

  Add $\mathbf{g}'$ to Good';

Remove duplicate elements from Good';

return Good' *and* Bad';

$\mathbf{d}_{i+1} = \mathbf{g}_{i+1} \setminus \mathbf{g}_i$ for $i > 1$. (We have that $\mathbf{g}_i = \bigcup_{j=1}^{i} \mathbf{d}_i$, so no information is lost.) Notice that, by definition, $D = \{\mathbf{d}_1, \ldots, \mathbf{d}_{m_2}\}$ is a partition of $\mathbf{g}_{m_2} = \{g_1, \ldots, g_h\}$, hence D may be represented by a *surjective function* f from $\mathbf{g}_{m_2}$ to D s.t. $f(g_i) = \mathbf{d}_j$ iff $g_i \in \mathbf{d}_j$. Let $\left\{ {h \atop m_2} \right\}$ be the Stirling number of the second kind. Then, the number of sequences $\emptyset \neq \mathbf{g}_1 \subset \mathbf{g}_1 \subset \cdots \subset \mathbf{g}_{m_2}$ is

$$\mathtt{seq}_h(m_2) := \left\{ {h \atop m_2} \right\} \cdot m_2! \, .$$

Second case: $\mathbf{g}_1 = \emptyset$. Thus $\mathbf{g}_2 \neq \emptyset$, hence the number of sequences $\emptyset = \mathbf{g}_1 \subset \mathbf{g}_2 \subset \cdots \subset \mathbf{g}_{m_2}$ is just as before, but with one less element in the sequence, i.e., $\mathtt{seq}_h(m_2 - 1)$. Hence, the number of pairs $(\mathbf{s}, \mathsf{Good})$ is

$$f_2(n, k_2) = \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + \sum_{h=1}^{i} \binom{i}{h} \sum_{m_2=1}^{h} \left(\mathtt{seq}_h(m_2) + \mathtt{seq}_h(m_2 - 1) \right) \right) \, .$$

I.e., we sum over all sizes i for sets $\mathbf{s}$, and either Good is empty, or Good is non-empty. In this second case, we sum over all possibilities for the size $h \in \{1, \ldots, i\}$ of the largest $\mathbf{g}_{m_2}$ (by (G1), every $\mathbf{g} \in \mathsf{Good}$ is $\subseteq \mathbf{s}$), and over all possibilities for the number of elements m_2 in Good: For each such combination of indices, we have $\mathtt{seq}_h(m_2) + \mathtt{seq}_h(m_2 - 1)$ sequences. We now proceed to derive a bound on $f_2(n, k_2)$. As $\mathtt{seq}_h(m_2)$ represents the number of surjective functions from a set of size h to a set of size m_2, clearly $\mathtt{seq}_h(m_2) \leq m_2^h$, just considering all such functions. Then,

$$
\begin{aligned}
f_2(n,k_2) &\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + \sum_{h=1}^{i} \binom{i}{h} \sum_{m_2=1}^{h} 2 \cdot m_2^h \right) \\
&\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + \sum_{h=1}^{i} \binom{i}{h} \sum_{m_2=1}^{h} 2 \cdot h^h \right) \\
&\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + 2 \cdot \sum_{h=1}^{i} \binom{i}{h} h^{h+1} \right) \\
&\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + 2 \cdot i \cdot \sum_{h=1}^{i} \binom{i}{h} i^{h} \right) \\
&\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + 2 \cdot i \cdot (i+1)^i \right) \\
&\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + 2 \cdot (i+1)^{i+1} \right) \\
&\leq \sum_{i=1}^{k_2} \binom{n}{i} \left(1 + 2 \cdot (k_2+1)^{k_2+1} \right) \\
&\leq (n+1)^{k_2} \cdot \left(1 + 2 \cdot (k_2+1)^{k_2+1} \right) \ .
\end{aligned}
$$

We now count the number of pairs $(\mathbf{q}, \mathsf{Bad})$. Assume $|\mathbf{b}_1| = h \leq k_1$ (notice that $m_1 \leq h$). By an argument similar to the one in the previous paragraph, we have that the number of non-empty sequences $\mathbf{b}_1 \supset \mathbf{b}_2 \supset \cdots \supset \mathbf{b}_{m_1} \neq \emptyset$ is $\mathtt{seq}_h(m_1)$. Hence, the number of pairs $(\mathbf{q}, \mathsf{Bad})$ is

$$
f_1(n,k_1) = \sum_{i=1}^{k_1} \binom{n}{i} \left(1 + \sum_{h=1}^{i} \binom{i}{h} \sum_{m_1=1}^{h} \mathtt{seq}_h(m_1) \right) \ ,
$$

which, with a similar calculation to the above, can be shown to be bounded by

$$
f_1(n,k_1) \leq (n+1)^{k_1} \cdot \left(1 + (k_1+1)^{k_1+1}\right) \ .
$$

Finally, $\left|V_{\mathrm{A}}^{\mathrm{de}}\right| \leq f_1(n,k_1) \cdot f_2(n,k_2)$, and $\left|V_{\mathrm{E}}^{\mathrm{de}}\right| \leq f_1(n,k_1) \cdot f_2(n,k_2) \cdot (n+1)^{k_1+k_2} \cdot |\Sigma|$, thus $\left|V_{\mathrm{A}}^{\mathrm{de}} \cup V_{\mathrm{E}}^{\mathrm{de}}\right| \leq 2 \cdot f_1(n,k_1) \cdot f_2(n,k_2) \cdot (n+1)^{k_1+k_2} \cdot |\Sigma| \leq 2 \cdot (n+1)^{2(k_1+k_2)} \cdot \left(1 + (k_1+1)^{k_1+1}\right) \cdot \left(1 + 2 \cdot (k_2+1)^{k_2+1}\right) \cdot |\Sigma|$. $\square$