Definition 0.0.1. Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, c)$ be a deterministic parity automaton. For $w \in \Sigma^* \cup \Sigma^\omega$ and $q \in Q$, we define $\lambda_\mathcal{A}(q, w) \in \mathbb{N}^{1+|w|}$ as follows: Let $q_0 q_1 \cdots \in Q^{1+|w|}$ be the unique run of $\mathcal{A}$ on w. Then $\lambda_\mathcal{A}(q, w)(n) = c(q_n)$.

We call two states $p, q \in Q$ **priority almost-equivalent**, if for all words $\alpha \in \Sigma^\omega$, $\lambda_\mathcal{A}(p, \alpha)$ and $\lambda_\mathcal{A}(q, \alpha)$ differ in only finitely many positions.

We define the **reachability order** $\preceq^\mathcal{A}_\text{reach} \subseteq Q \times Q$ as $p \preceq^\mathcal{A}_\text{reach} q$ iff q is reachable from p. ("p is closer to q_0 than q"). Note that $p \preceq^\mathcal{A}_\text{reach} q$ and $q \preceq^\mathcal{A}_\text{reach} p$ together mean that p and q reside in the same SCC.

Definition 0.0.2. Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, c)$ be a DPA and let $\sim \subseteq Q \times Q$ be a congruence relation on $\mathcal{A}$. We define the **Schewe** automaton $\mathcal{S}$ as follows:

For each state q, let $[q]_\sim = \{p \in Q \mid q \sim p\}$ be its equivalence class of $\sim$ and let ${}^{Q}/_{\sim} = \{[q]_\sim \mid q \in Q\}$ be the set of equivalence classes. For each such class $\mathfrak{c}$ we fix a representative $r_\mathfrak{c} \in \mathfrak{c}$ which is $\preceq^\mathcal{A}_\text{reach}$-maximal in its class, meaning that all states in $\mathfrak{c}$ that are reachable from $r_\mathfrak{c}$ are also in its SCC.

The automaton is then almost the same as the original DPA, with only a few modifications. Namely, $\mathcal{S} = (Q, \Sigma, r_{[q_0]_\sim}, \delta_\mathcal{S}, c)$.

For each transition $\delta_\mathcal{S}(q, a)$, let $\delta(q, a) = p$. If $q \prec^\mathcal{A}_\text{reach} r_{[p]_\sim}$ (i.e. q is not reachable from the representative of $[p]_\sim$), then $\delta_\mathcal{S}(q, a) = r_{[p]_\sim}$. Otherwise, we keep $\delta_\mathcal{S}(q, a) = p$. In other words, every time a transition moves to a different quotient class, it skips to the representative which lies as "deep" inside the automaton as possible.

Lemma 0.0.1. *For a given $\mathcal{A}$ and $\sim$, the Schewe automaton $\mathcal{S}$ can be computed in $\mathcal{O}(|\mathcal{A}|)$.*

Proof. Using e.g. Kosaraju's algorithm **??**, the SCCs of $\mathcal{A}$ can be computed in $\mathcal{O}(|\mathcal{A}|)$. ☐

We focus on a specialized version of the Schewe automaton. Let $\sim$ be the priority equivalence and let $\mathcal{S}$ be the according automaton. We define $\mathcal{S}_m$ as the Moore-minimization of $\mathcal{S}$.

Lemma 0.0.2. *Priority almost-equivalence implies language equivalence.*

Proof. Let $\mathcal{A} = (Q_\mathcal{A}, \Sigma, q_0^\mathcal{A}, \delta_\mathcal{A}, c_\mathcal{A})$ and $\mathcal{B} = (Q_\mathcal{B}, \Sigma, q_0^\mathcal{B}, \delta_\mathcal{B}, c_\mathcal{B})$ be two DPA that are priority almost-equivalent and assume towards a contradiction that they are not language equivalent. Due to symmetry we can assume that there is a $w \in L(\mathcal{A}) \setminus L(\mathcal{B})$.

Consider $\alpha = \lambda_\mathcal{A}(q_0^\mathcal{A}, w)$ and $\beta = \lambda_\mathcal{B}(q_0^\mathcal{B}, w)$, the priority outputs of the automata on w. By choice of w, we know that $a := \max \mathrm{Inf}(\alpha)$ is even and $b := \max \mathrm{Inf}(\beta)$ is odd. Without loss of generality, assume $a > b$. That means a is seen only finitely often in β but infinitely often in a. Hence, α and β differ at infinitely many positions where a occurs in α. That would mean w is a witness that the two automata are not priority almost-equivalent, contradicting our assumption. ☐

Lemma 0.0.3. *Let $\mathcal{A}$ a DPA and $\mathcal{S}_m$ be the specialized Schewe automaton. Then $\mathcal{A}$ and $\mathcal{S}_m$ are priority almost-equivalent.*

Proof. Let $\mathcal{A} = (Q,$ ☐

Lemma 0.0.4. *Let $\mathcal{A}$ a DPA and $\mathcal{S}_m = (Q, \Sigma, q_0, \delta, c)$ be the specialized Schewe automaton. If $p, q \in Q$ are almost priority-equivalent, then they lie in the same SCC.*

Lemma 0.0.5. *There is no DPA almost priority-equivalent to $\mathcal{A}$ that is smaller than $\mathcal{S}_m$.*

Theorem 0.0.6. *For a given DPA $\mathcal{A}$, a minimal almost priority-equivalent automaton can be computed in $\mathcal{O}$.*