0.1 Fritz & Wilke

0.1.1 Delayed Simulation Game

In this section we consider delayed simulation games and variants thereof on DPAs. This approach is based on the paper [] which considered the games for alternating parity automata. The DPAs we use are a special case of these APAs and therefore worth examining.

Definition 0.1.1. Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, c)$ be a DPA. We define the *delayed simulation automaton* $\mathcal{A}_{\mathrm{de}}(p, q) = (Q_{\mathrm{de}}, \Sigma, (p, q, \checkmark), \delta_{\mathrm{de}}, F_{\mathrm{de}})$, which is a deterministic Büchi automaton, as follows.

- $Q_{\mathrm{de}} = Q \times Q \times (\mathrm{img}(c) \cup \{\checkmark\})$, i.e. the states are given as triples in which the first two components are states from $\mathcal{A}$ and the third component is either a priority from $\mathcal{A}$ or $\checkmark$.
- The alphabet remains Σ.
- The starting state is a triple $(p, q, \checkmark)$, where $p, q \in Q$ are parameters given to the automaton.
- $\delta_{\mathrm{de}}((p, q, i), a) = (p', q', \gamma(i, c(p'), c(q')))$, where $p' = \delta(p, a)$, $q' = \delta(q, a)$, and γ is defined in detail below. The first two components behave like a regular product automaton.
- $F_{\mathrm{de}} = Q \times Q \times \{\checkmark\}$.

$\gamma : \mathbb{N} \times \mathbb{N} \times (\mathbb{N} \cup \{\checkmark\}) \to \mathbb{N} \cup \{\checkmark\}$ is the update function of the third component and defines the "obligations" as they are called in []. It is defined as

$$\gamma(i, j, \checkmark) = \begin{cases} \checkmark & \text{if } i \text{ and } j \text{ are odd and } i \leq j \\ \checkmark & \text{if } i \text{ and } j \text{ are even and } j \leq i \\ \checkmark & \text{if } i \text{ is odd and } j \text{ is even} \\ \min\{i, j\} & \text{else} \end{cases}$$

$$\gamma(i, j, k) = \begin{cases} \checkmark & \text{if } i \text{ and } j \text{ are odd and } i \leq k \text{ and } i \leq j \\ \checkmark & \text{if } i \text{ and } j \text{ are even and } j \leq k \text{ and } j \leq i \\ \checkmark & \text{if } i \text{ is odd and } j \text{ is even and } (i \leq k \text{ or } j \leq k) \\ \min\{i, j, k\} & \text{else} \end{cases}$$

Definition 0.1.2. Let $\mathcal{A}$ be a DPA and let $\mathcal{A}_{\mathrm{de}}$ be the delayed simulation automaton of $\mathcal{A}$. We say that a state p *de*-simulates a state q if $L(\mathcal{A}_{\mathrm{de}}(p, q)) = \Sigma^\omega$. In that case we write $p \leq_{\mathrm{de}} q$. If also $q \leq_{\mathrm{de}} p$ holds, we write $p \equiv_{\mathrm{de}} q$.