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. We define $\leq_{\checkmark} \subseteq (\mathbb{N} \cup \{\checkmark\}) \times (\mathbb{N} \cup \{\checkmark\})$ as follows:

- For $i, j \in \mathbb{N}$, we set $i \leq_{\checkmark} j$ iff $i \leq j$.
- For all $i \in \mathbb{N}$, we have $i \leq_{\checkmark} \checkmark$ and $\checkmark \not\leq_{\checkmark} i$.
- $\checkmark \leq_{\checkmark} \checkmark$.

Further, we define an order of "goodness" on parity priorities as $\preceq_{\mathrm{p}} \subseteq \mathbb{N} \times \mathbb{N}$ as $0 \preceq_{\mathrm{p}} 2 \preceq_{\mathrm{p}} 4 \preceq_{\mathrm{p}} \cdots \preceq_{\mathrm{p}} 5 \preceq_{\mathrm{p}} 3 \preceq_{\mathrm{p}} 1$.

Definition 0.1.2. Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, c)$ be a DPA. We define the *delayed simulation automaton* $\mathcal{A}_{\mathrm{de}}(p, q) = (Q_{\mathrm{de}}, \Sigma, (p, q, \gamma(c(p), c(q), \checkmark)), \delta_{\mathrm{de}}, F_{\mathrm{de}})$, 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 $(p, q, \gamma(c(p), c(q), \checkmark))$, where $p, q \in Q$ are parameters given to the automaton, and γ is defined below.
- $\delta_{\mathrm{de}}((p, q, i), a) = (p', q', \gamma(i, c(p'), c(q')))$, where $p' = \delta(p, a)$, $q' = \delta(q, a)$, and γ is the same function as used in the initial state. 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

$$\gamma(i, j, k) = \begin{cases} \checkmark & \text{if } i \text{ is odd and } i \leq_{\checkmark} k \text{ and } j \preceq_{\mathrm{p}} i \\ \checkmark & \text{if } j \text{ is even and } j \leq_{\checkmark} k \text{ and } j \preceq_{\mathrm{p}} i \\ \min_{\leq_{\checkmark}}\{i, j, k\} & \text{else} \end{cases}$$

Definition 0.1.3. 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 $p \equiv_{\mathrm{de}} q$.

$\equiv_{\mathrm{de}}$ is a congruence relation.

Our overall goal is to use $\equiv_{\mathrm{de}}$ to build a quotient automaton of our original DPA. The first step towards this goal is to show that the result is actually a well-defined DPA, by proving that the relation is a congruence.

Lemma 0.1.1. *γ is monotonous in the third component, i.e. if $k \leq_{\checkmark} k'$, then $\gamma(i,j,k) \leq_{\checkmark} \gamma(i,j,k')$ for all $i, j \in \mathbb{N}$.*

Proof. We consider each case in the definition of γ. If i is odd, $i \leq_{\checkmark} k$ and $j \preceq_{\mathrm{p}} i$, then also $i \leq_{\checkmark} k'$ and $\gamma(i,j,k) = \gamma(i,j,k') = \checkmark$.

If j is even, $j \leq_{\checkmark} k$ and $j \preceq_{\mathrm{p}} i$, then also $j \leq_{\checkmark} k'$ and $\gamma(i,j,k) = \gamma(i,j,k') = \checkmark$.

Otherwise, $\gamma(i,j,k) = \min\{i,j,k\}$ and $\gamma(i,j,k') = \min\{i,j,k'\}$. Since $k \leq_{\checkmark} k'$, $\gamma(i,j,k) \leq_{\checkmark} \gamma(i,j,k')$. □

Lemma 0.1.2. *Let $\mathcal{A}$ be a DPA and let $p, q \in Q$, $k \in \mathbb{N} \cup \{\checkmark\}$. If the run of $\mathcal{A}_{\mathrm{de}}$ starting at (p,q,k) on some $\alpha \in \Sigma^\omega$ is accepting, then for all $k \leq_{\checkmark} k'$ also the run of $\mathcal{A}_{\mathrm{de}}$ starting at (p,q,k') on α is accepting.*

Proof. Let ρ be the run starting at (p,q,k) and let ρ' be the run starting at (p,q,k'). Further, let p_i, q_i, k_i, and k'_i be the components of the states of those runs in the i-th step. Via induction we show that $k_i \leq_{\checkmark} k'_i$ for all i. Since k_i is $\checkmark$ infinitely often, the same must be true for k'_i and ρ' is accepting.

For $i = 0$, we have $k_0 = k \leq_{\checkmark} k' = k'_0$. Otherwise, we have $k_{i+1} = \gamma(c(p_{i+1}), c(q_{i+1}), k_i)$ and k'_{i+1} analogously. The rest follows from Lemma 0.1.1. □

Lemma 0.1.3. *Let $\mathcal{A}$ be a DPA and $\rho \in Q_{\mathrm{de}}^\omega$ be a run of $\mathcal{A}_{\mathrm{de}}$ on some word, where the third component is $k \in (\mathbb{N} \cup \{\checkmark\})^\omega$. For all i, $k(i+1) \leq_{\checkmark} k(i)$ or $k(i+1) = \checkmark$.*

Proof. Follows directly from the definition of γ. □

Lemma 0.1.4. *Let $\mathcal{A}$ be a DPA. Then $\leq_{\mathrm{de}}$ is reflexive and transitive.*

Proof. For reflexivitiy, we need to show that $q \leq_{\mathrm{de}} q$ for all states q. This is rather easy to see. For a word $\alpha \in \Sigma^\omega$, the third component of states in the run of $\mathcal{A}_{\mathrm{de}}(q,q)$ on α is always $\checkmark$, as $\gamma(i,i,\checkmark) = \checkmark$.

For transitivity, let $q_1 \leq_{\mathrm{de}} q_2$ and $q_2 \leq_{\mathrm{de}} q_3$. Assume towards a contradiction that $q_1 \not\leq_{\mathrm{de}} q_3$, so there is a word $\alpha \notin L(\mathcal{A}_{\mathrm{de}}(q_1,q_3))$. We consider the three runs ρ_{12}, ρ_{23}, and ρ_{13} of $\mathcal{A}_{\mathrm{de}}(q_1,q_2)$, $\mathcal{A}_{\mathrm{de}}(q_2,q_3)$, anbd $\mathcal{A}_{\mathrm{de}}(q_1,q_3)$ respectively on α. Then ρ_{12} and ρ_{23} are accepting, whereas ρ_{13} is not.

Moreover, we use the notation $q_1(i), q_2(i), q_3(i)$ for the states of the run and $k_{12}(i), k_{23}(i), k_{13}(i)$ for the obligations. More specifically for a run ρ_{ij}, it is true that $\rho_{ij}(n) = (q_i(n), q_j(n), k_{ij}(n))$.

As ρ_{13} is not accepting, k_{13} becomes $\checkmark$ only finitely often. By Lemma 0.1.3, that means k_{13} only grows smaller from some point on and reaches a minimum eventually. Let $n_0 \in \mathbb{N}$ be such a position from which on k_{13} does not change anymore. Let $l_j = \min\{c(q_j(i)) \mid i \geq n_0\}$ be the lowest priority that q_j reaches after n_0.

Below we prove that $l_3 \preceq_p l_1$. If we take this as a fact, we can conclude the proof by seperating cases depending on $k_{13}(n_0)$, the final value of k_{13} that does not change anymore.

If $k_{13}(n_0) = l_3$, let $m \geq n_0$ be a position with $c(q_3(m)) = l_3$. Then

$$k_{13}(m) = l_3 = \gamma(c(q_1(m)), l_3, k_{13}(m-1)).$$

We know that $k_{13}(m-1) \leq k_{13}(m) = l_3$; if $m = n_0$, this follows from Lemma 0.1.3. The definition of γ thus sets $k_{13}(m) = \checkmark$, which is a contradiction to the choice of n_0.

We finish the argumentation by showing $l_3 \preceq_p l_1$ in two different cases.

Case 1: l_2 is even. We claim that l_3 is even and $l_3 \leq l_2$.

First, to show $l_3 \leq l_2$, let $m \geq n_0$ be a position with $c(q_2(m)) = l_2$ and let $n \geq m$ be the minimal position with $k_{23}(n) = \checkmark$. If $m = n$, then $c(q_3(n)) \preceq_p c(q_2(n)) = l_2$ and therefore $c(q_3(n)) \leq l_2$. Otherwise, from m to $n-1$, k_{23} only grows smaller and is at most l_2. As the priority of q_2 never becomes an odd number smaller than l_2, the only way for $k_{23}(m)$ to be $\checkmark$ is that $c(q_3(m))$ is even and $c(q_3(m)) \leq k_{23}(m-1) \leq l_2$.

Second, assume that l_3 is odd and let m be a position with $c(q_3(m)) = l_3$. As l_2 is even, we have $k_{23}(m) \leq l_3 < l_2$. At no future position can $c(q_3)$ both be even and smaller than k_{23}, so k_{23} never becomes $\checkmark$ again. Thus, ρ_{23} is not accepting.

We claim that l_1 is odd or $l_1 \geq l_2$.

Towards a contradiction assume the opposite, so $l_1 < l_2$ and l_1 is even. Let $m \geq n_0$ be a position with $c(q_1(m)) = l_1$. Then $c(q_2(m)) \not\preceq_p c(q_1(m))$ and therefore $k_{12}(m) = l_1$. At no position after m can it happen that the conditions for k_{12} to become $\checkmark$ again are satisfied. Thus, ρ_{12} would not be accepting.

If l_1 is odd and l_3 is even, $l_3 \preceq_p l_1$ follows. For the other case, l_1 and l_3 both being even with $l_3 \leq l_2 \leq l_1$, that also holds.

Case 2: l_2 is odd. We skip the details of this case as it works symmetrically to case 1. In particular, we first show that l_1 is odd and $l_1 \leq l_2$. We continue with l_3 being even or $l_3 \geq l_2$. From these two statements, $l_3 \preceq_p l_1$ again follows. □

Lemma 0.1.5. *Let $\mathcal{A}$ be a DPA. Then $\equiv_{\mathrm{de}}$ is a congruence relation.*

Proof. The three properties that are required for $\equiv_{\mathrm{de}}$ to be a equivalence relation are rather easy to see. Reflexivity and transitivity have been shown for $\leq_{\mathrm{de}}$ already and symmetry follows from the definition. Congruence requires more elaboration.

Let $p \equiv_{\mathrm{de}} q$ be two equivalent states. Let $a \in \Sigma$ and $p' = \delta(p,a)$ and $q' = \delta(q,a)$. We have to show that also $p' \equiv_{\mathrm{de}} q'$. Towards a contradiction, assume that $p' \not\leq_{\mathrm{de}} q'$, so there is a word $\alpha \notin L(\mathcal{A}_{\mathrm{de}}(p',q'))$. Let $(p',q',k) = \delta_{\mathrm{de}}((p,q,\checkmark),a)$. By Lemma 0.1.2, the run of $\mathcal{A}_{\mathrm{de}}$ on α from (p',q',k) cannot be accepting; otherwise, the run of $\mathcal{A}_{\mathrm{de}}$ from $(p',q',\checkmark)$ would be accepting and $\alpha \in L(\mathcal{A}_{\mathrm{de}}(p',q'))$. Hence, $a\alpha \notin L(\mathcal{A}_{\mathrm{de}}(p,q))$, which means that $p \not\equiv_{\mathrm{de}} q$. □

Corollary 0.1.6. *Let $\mathcal{A}$ be a DPA and $\equiv_{\mathrm{de}}$ the corresponding delayed simulation-relation. The quotient automaton $\mathcal{A}/_{\equiv_{\mathrm{de}}}$ is well-defined and deterministic.*

Correctness of the quotient

The quotient automaton itself is used "only" for state space reduction. The main point of delayed simulation is that the priorities of equivalent states can be made equivalent.

Theorem 0.1.7. *Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, c)$ be a DPA. Let $\sim \subseteq Q \times Q$ be a congruence relation such that $p \sim q$ implies $c(p) = c(q)$. Then $L(\mathcal{A}) = L(\mathcal{A}/_\sim)$.*

Proof. Since $\mathcal{A}$ is deterministic and $\sim$ is a congruence relation, $\mathcal{A}/_\sim = (Q_\sim, \Sigma, [q_0]_\sim, \delta_\sim, c_\sim)$ is deterministic as well. Let $\alpha \in \Sigma^\omega$ be a word and let π and ρ be the runs of $\mathcal{A}$ and $\mathcal{A}/_\sim$.

For each $i \in \mathbb{N}$, we have $\rho(i) = [\pi(i)]_\sim$ and $c_\sim(\rho(i)) = c(\pi(i))$. Thus, $\mathrm{Inf}(c(\pi)) = \mathrm{Inf}(c(\rho))$ and π is accepting iff ρ is accepting. □

Lemma 0.1.8. *Let $\mathcal{A}$ be a DPA and let π and ρ be runs of $\mathcal{A}$ on the same word but starting at different states. If $\pi(0) \equiv_{de} \rho(0)$, then $\min Occ(c(\pi)) = \min Occ(c(\rho))$.*

Proof. We do a prove by contradiction. Let $k = \min \mathrm{Occ}(c(\pi))$ and $l = \min \mathrm{Occ}(c(\rho))$. Assume without loss of generality that $k < l$. Let α be the word that is read by the two runs.

If k is even, let σ be the run of $\mathcal{A}_{\mathrm{de}}(\pi(0), \rho(0))$ on α. Let n be a position at which $c(\pi(0)) = k$. We claim that for all $i \geq n$, the third component of $\sigma(i)$ is k.

At $\sigma(n)$, this must be true because $k < l \leq c(\rho(n))$ and thus $c(\rho(n)) \not\preceq_p c(\pi(n))$. At all positions after n, it can never occur that $c(\rho(i))$ is at most k or that $c(\pi(i))$ is odd and smaller than k. The rest follows from the definition of γ.

If k is odd, we can argue similarly on the run of $\mathcal{A}_{\mathrm{de}}(\rho(0), \pi(0))$. As soon as $c(\pi)$ reaches its minimum, the third component of the run will never change again. □

Theorem 0.1.9. *Let $\mathcal{A} = (Q, \Sigma, q_0, \delta, c)$ be a DPA and let $p, q \in Q$ with $p \equiv_{\mathrm{de}} q$ and $c(p) < c(q)$. Define $\mathcal{A}' = (Q, \Sigma, q_0, \delta, c')$ with $c'(s) = \begin{cases} \min c(p) & \text{if } s = q \\ c(s) & \text{else} \end{cases}$. Then $L(\mathcal{A}) = L(\mathcal{A}')$.*

Proof. First, consider the case that $c(p)$ is an even number. The parity of each state is at least as good in $\mathcal{A}'$ as it is in $\mathcal{A}$, so $L(\mathcal{A}) \subseteq L(\mathcal{A}')$. For the other direction, assume there is a $\alpha \in L(\mathcal{A}') \setminus L(\mathcal{A})$, so the respective run $\rho \in Q^\omega$ is accepting in $\mathcal{A}'$ but not in $\mathcal{A}$.

For this to be true, ρ must visit q infinitely often and $c'(q)$ must be the lowest priority that occurs infinitely often; otherwise, the run would have the same acceptance in both automata. Thus, there is a finite word $w \in \Sigma^*$ such that from q, $\mathcal{A}$ reaches again q via w and inbetween only priorities greater than $c'(q)$ are seen.

Now consider the word w^ω and the run π_q of $\mathcal{A}$ on said word starting in q. With the argument above, we know that the minimal priority occuring in $c(\pi)$ is greater than $c'(q)$. If we take the run π_p on w^ω starting at p though, we find that this run sees priority $c(p) = c'(q)$ at the very beginning. This contradicts Lemma 0.1.8, as $p \equiv_{\mathrm{de}} q$. Thus, the described α cannot exist.

If $c(p)$ is an odd number, a very similar argumentation can be applied with the roles of $\mathcal{A}$ and $\mathcal{A}'$ reversed. We omit this repetition. □

Corollary 0.1.10. *For a DPA $\mathcal{A}$, the quotient automaton $\mathcal{A}/_{\equiv_{\mathrm{de}}}$ is a DPA that recognizes the same language.*