Team automata : a formal approach to the modeling of collaboration between system components

Beek, M.H. ter

Author: Beek, Maurice H. ter
Title: Team automata : a formal approach to the modeling of collaboration between system components
Issue Date: 2003-12-10

3. Automata

The basic concept underlying team automata is an *automaton*. An automaton captures the idea of a system with *states* (configurations, possibly an infinite number of them), together with *actions* the executions of which lead to (nondeterministic) state changes. In addition some of the states may be designated as *initial states* from which the automaton may start its executions. Also final or accepting states may be distinguished, which can be used to define when an execution of the automaton is considered successful. A particular automaton model is the well-known *finite (state) automaton*. Such an automaton has a finite set of states, with initial states and final states, as well as a finite set of actions. Finite automata are among the most basic models in many branches of computer science.

In this thesis automata are used as structures defining a state space that is traversed by executing actions. They come into play when designing and analyzing complex systems with a potentially infinite number of configurations due to, e.g., unbounded data structures such as counters.

We begin this chapter by defining precisely the type of automata we shall use in the sequel, thus laying the foundation on which we shall build our team automata framework. Subsequently we review some notions from automata theory.

3.1 Automata, Computations, and Behavior

Definition 3.1.1. *An* automaton *is a construct* $\mathcal{A} = (Q, \Sigma, \delta, I)$*, where*

Q *is the set of* states *of* $\mathcal{A}$*, which may be infinite,*
Σ *is the set of* actions *of* $\mathcal{A}$ *such that* $\Sigma \cap Q = \varnothing$*,*
$\delta \subseteq Q \times \Sigma \times Q$ *is the set of* labeled transitions *of* $\mathcal{A}$*, and*
$I \subseteq Q$ *is the set of* initial states *of* $\mathcal{A}$*.* □

In the figures, the states of an automaton are drawn as circles and labeled transitions appear as labeled arcs between states. Wavy arcs are used to indicate the initial states. See, e.g., Figure 3.1.

Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let $a \in \Sigma$. Then the set of *a-transitions* (of $\mathcal{A}$) is denoted by δ_a and is defined as $\delta_a = \{(q, q') \mid (q, a, q') \in \delta\}$. An a-transition $(q, q) \in \delta_a$ is called a *loop* (on a). We refer to $\mathcal{A}$ as the *trivial automaton* if $\mathcal{A} = (\varnothing, \varnothing, \varnothing, \varnothing)$. Instead of labeled transition we often simply say transition. Finally, a transition $(q, q') \in \delta_a$ is called an *outgoing transition* of q and an *incoming transition* of q'.

Executing an action in a certain state leads to a change of state as described by the labeled transitions. The consecutive execution of a sequence of actions from an initial state defines a *computation*.

Definition 3.1.2. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

(1) a finite computation *of $\mathcal{A}$ is a finite sequence $\alpha = q_0 a_1 q_1 a_2 q_2 \cdots a_n q_n$, where $n \geq 0$, $q_i \in Q$ for $0 \leq i \leq n$, and $a_j \in \Sigma$ for $1 \leq j \leq n$ are such that $q_0 \in I$ and $(q_i, a_{i+1}, q_{i+1}) \in \delta$ for all $0 \leq i < n$;*
if $n = 0$ and hence $\alpha = q_0 \in I$, then α is a trivial *computation;*
by $\mathbf{C}_{\mathcal{A}}$ we denote the set of all finite computations *of $\mathcal{A}$,*

(2) an infinite computation *of $\mathcal{A}$ is an infinite sequence $\alpha = q_0 a_1 q_1 a_2 q_2 \cdots$, where $q_i \in Q$ for all $i \geq 0$ and $a_j \in \Sigma$ for all $j \geq 1$ are such that $q_0 \in I$ and $(q_i, a_{i+1}, q_{i+1}) \in \delta$ for all $i \geq 0$;*
by $\mathbf{C}^{\omega}_{\mathcal{A}}$ we denote the set of all infinite computations *of $\mathcal{A}$, and*

(3) the set of all computations *of $\mathcal{A}$ is denoted by $\mathbf{C}^{\infty}_{\mathcal{A}}$ and is defined as $\mathbf{C}^{\infty}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}} \cup \mathbf{C}^{\omega}_{\mathcal{A}}$.* □

Thus for a given automaton $\mathcal{A} = (Q, \Sigma, \delta, I)$, its finite computations form a finitary language $\mathbf{C}_{\mathcal{A}} \subseteq I(\Sigma Q)^*$ while its infinite computations form an infinitary language $\mathbf{C}^{\omega}_{\mathcal{A}} \subseteq I(\Sigma Q)^{\omega}$. Observe that $\mathbf{C}_{\mathcal{A}} = \varnothing$ if and only if $I = \varnothing$. Moreover, $\mathbf{C}^{\omega}_{\mathcal{A}}$ may be empty, even when $\mathbf{C}_{\mathcal{A}}$ is infinite (cf. Example 3.1.12).

The infinite computations of $\mathcal{A}$ can be expressed in terms of finite computations, viz. as limits of length-increasing sequences of finite computations.

Lemma 3.1.3. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Let $\alpha \in \mathbf{C}^{\infty}_{\mathcal{A}}$. Then*

$\alpha \in \mathbf{C}^{\omega}_{\mathcal{A}}$ if and only if there exist $\alpha_1 \leq \alpha_2 \leq \cdots \in \mathbf{C}_{\mathcal{A}}$ such that for all $n \geq 1$, $\alpha_n \neq \alpha_{n+1}$ and $\alpha = \lim_{n \to \infty} \alpha_n$.

Proof. (If) Trivial.

(Only if) Obvious from the observation $\text{pref}(\alpha) \cap I(\Sigma Q)^* \subseteq \mathbf{C}_{\mathcal{A}}$. □

Both finite and infinite computations are thus sequences of which every prefix of odd length is a finite computation.

Theorem 3.1.4. *Let $\mathcal{A}$ be an automaton. Then*

$\alpha \in \mathbf{C}_{\mathcal{A}}^{\infty}$ *if and only if for all* $n \geq 1$ *there exist* $\alpha_1 \leq \alpha_2 \leq \cdots \in \mathbf{C}_{\mathcal{A}}$ *such that* $\alpha = \lim_{n\to\infty} \alpha_n$. □

In fact, the infinite computations of an automaton are determined by its set of finite computations.

Lemma 3.1.5. *Let $\mathcal{A}$ and $\mathcal{A}'$ be two automata. Then*

if $\mathbf{C}_{\mathcal{A}} \subseteq \mathbf{C}_{\mathcal{A}'}$, *then* $\mathbf{C}_{\mathcal{A}}^{\omega} \subseteq \mathbf{C}_{\mathcal{A}'}^{\omega}$.

Proof. Let $\alpha \in \mathbf{C}_{\mathcal{A}}^{\omega}$. Hence by Lemma 3.1.3, $\alpha = \lim_{n\to\infty} \alpha_n$ for computations $\alpha_n \in \mathbf{C}_{\mathcal{A}}$ such that $\alpha_n \leq \alpha_{n+1}$ and $\alpha_n \neq \alpha_{n+1}$, for all $n \geq 1$. Since $\mathbf{C}_{\mathcal{A}} \subseteq \mathbf{C}_{\mathcal{A}'}$, again applying Lemma 3.1.3 (now in the other direction) yields that $\alpha \in \mathbf{C}_{\mathcal{A}'}^{\omega}$. □

Theorem 3.1.6. *Let $\mathcal{A}$ and $\mathcal{A}'$ be two automata. Then*

$\mathbf{C}_{\mathcal{A}}^{\infty} = \mathbf{C}_{\mathcal{A}'}^{\infty}$ *if and only if* $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}'}$. □

Given a computation of an automaton one may choose to focus on certain actions while filtering away other information. In this way, behavioral records are made of computations.

Definition 3.1.7. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ be an alphabet disjoint from Q. Then*

(1) $v \in \Theta^{\infty}$ *is a* Θ-record *of $\mathcal{A}$ if $v = pres_{\Theta}(\alpha)$ for some $\alpha \in \mathbf{C}_{\mathcal{A}}^{\infty}$,*

(2) the Θ-behavior *of $\mathcal{A}$ is denoted by $\mathbf{B}_{\mathcal{A}}^{\Theta,\infty}$ and is defined as $\mathbf{B}_{\mathcal{A}}^{\Theta,\infty} = pres_{\Theta}(\mathbf{C}_{\mathcal{A}}^{\infty})$,*

(3) the finitary Θ-behavior *of $\mathcal{A}$ is denoted by $\mathbf{B}_{\mathcal{A}}^{\Theta}$ and is defined as $\mathbf{B}_{\mathcal{A}}^{\Theta} = \mathbf{B}_{\mathcal{A}}^{\Theta,\infty} \cap \Theta^*$, and*

(4) the infinitary Θ-behavior *of $\mathcal{A}$ is denoted by $\mathbf{B}_{\mathcal{A}}^{\Theta,\omega}$ and is defined as $\mathbf{B}_{\mathcal{A}}^{\Theta,\omega} = \mathbf{B}_{\mathcal{A}}^{\Theta,\infty} \cap \Theta^{\omega}$.* □

If Σ is the full set of actions of automaton $\mathcal{A}$, then a Σ-record is also simply called a *record* and the (finitary or infinitary) Σ-behavior of $\mathcal{A}$ is also referred to as the (finitary or infinitary) behavior of $\mathcal{A}$, respectively.

W_1:

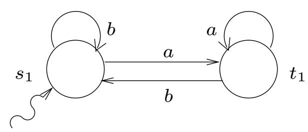

Fig. 3.1. Automaton W_1.

Example 3.1.8. Let $W_1 = (\{s_1, t_1\}, \{a, b\}, \delta_1, \{s_1\})$, where $\delta_1 = \{(s_1, b, s_1), (s_1, a, t_1), (t_1, a, t_1), (t_1, b, s_1)\}$, be an automaton modeling a **w**heel (of a car). It is depicted in Figure 3.1.

The state s_1 indicates that the wheel stands **s**till, while the state t_1 indicates that the wheel **t**urns. The result of **a**ccelerating, modeled by action a, makes the wheel turn. The result of **b**raking, modeled by action b causes the wheel to stand still. Initially the wheel stands still, as indicated by the initial state s_1.

An example of a finite computation of W_1 is $\alpha = s_1 a t_1 b s_1 \in \mathbf{C}_{W_1}$, modeling accelerating and subsequently braking. The record of this computation is $\text{pres}_\Sigma(\alpha) = ab$, which is thus an element of the finitary behavior of W_1: $ab \in \mathbf{B}^{\Sigma}_{W_1}$. An example of an infinite computation of W_1 is $s_1 a t_1 b s_1 b s_1 \cdots \in \mathbf{C}^{\omega}_{W_1}$, which thus leads to an example of an infinitary behavior $ab^\omega \in \mathbf{B}^{\Sigma,\omega}_{W_1}$. □

It is immediate that finite computations define finite records. In fact, all finite Θ-records can be obtained from finite computations. On the other hand, infinite computations may give rise to finite Θ-records even though infinite Θ-records can only be obtained from infinite computations.

Lemma 3.1.9. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ be an alphabet disjoint from Q. Then*

(1) $\mathbf{B}^{\Theta}_{\mathcal{A}} = \mathit{pres}_\Theta(\mathbf{C}_{\mathcal{A}})$ *and*

(2) $\mathbf{B}^{\Theta,\omega}_{\mathcal{A}} = \mathit{pres}_\Theta(\mathbf{C}^{\omega}_{\mathcal{A}}) \cap \Theta^\omega$.

Proof. (1) ($\supseteq$) Immediate.

($\subseteq$) Let $v \in \Theta^*$ and $\alpha \in \mathbf{C}^{\infty}_{\mathcal{A}}$ be such that $\text{pres}_\Theta(\alpha) = v$. Let $\alpha_1 \leq \alpha_2 \leq \cdots \in \mathbf{C}_{\mathcal{A}}$ be such that $\alpha = \lim_{n\to\infty} \alpha_n$. Since pres_Θ is a homomorphism we have $\text{pres}_\Theta(\alpha_1) \leq \text{pres}_\Theta(\alpha_2) \leq \cdots$. By definition $\lim_{n\to\infty} \text{pres}_\Theta(\alpha_n) = \text{pres}_\Theta(\alpha) = v \in \Theta^*$, from which it follows that there exists an $m \geq 1$ such that $\text{pres}_\Theta(\alpha_m) = \text{pres}_\Theta(\alpha_{m+k})$ for all $k \geq 0$. Hence $\text{pres}_\Theta(\alpha) = \text{pres}_\Theta(\alpha_m) \in \text{pres}_\Theta(\mathbf{C}_{\mathcal{A}})$.

(2) ($\supseteq$) Immediate, by Definition 3.1.7(2,4).

($\subseteq$) Let $\alpha \in \mathbf{B}_{\mathcal{A}}^{\Theta,\omega}$. Then Definition 3.1.7(2,4) implies $\alpha \in \mathrm{pres}_{\Theta}(\mathbf{C}_{\mathcal{A}}^{\infty}) \cap \Theta^{\omega}$. Hence either $\alpha \in \mathrm{pres}_{\Theta}(\mathbf{C}_{\mathcal{A}}^{\omega}) \cap \Theta^{\omega}$ or $\alpha \in \mathrm{pres}_{\Theta}(\mathbf{C}_{\mathcal{A}}) \cap \Theta^{\omega} = \varnothing$. □

The finite computations thus determine the finitary behavior of an automaton. By Theorem 3.1.6, moreover, they also determine its infinitary behavior and thus the full behavior.

Theorem 3.1.10. *Let $\mathcal{A}$ and $\mathcal{A}'$ be two automata and let Θ be an alphabet disjoint from their sets of states. Then*

if $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}'}$, then $\mathbf{B}_{\mathcal{A}}^{\Theta} = \mathbf{B}_{\mathcal{A}'}^{\Theta}$ and $\mathbf{B}_{\mathcal{A}}^{\Theta,\omega} = \mathbf{B}_{\mathcal{A}'}^{\Theta,\omega}$. □

Corollary 3.1.11. *Let $\mathcal{A}$ and $\mathcal{A}'$ be two automata and let Θ be an alphabet disjoint from their sets of states. Then*

if $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}'}$, then $\mathbf{B}_{\mathcal{A}}^{\Theta,\infty} = \mathbf{B}_{\mathcal{A}'}^{\Theta,\infty}$. □

Unlike the situation for computations as formulated in Lemma 3.1.5 and Theorem 3.1.6, the finitary behavior of an automaton does not determine its infinitary behavior. The loss of information due to the omission of states prohibits combining "matching" finite records into an infinite record.

Example 3.1.12. Consider the two automata $\mathcal{A} = (Q, \{a\}, \delta, \{q\})$ and $\mathcal{A}' = (Q', \{a\}, \delta', \{q'\})$, where $Q = \{q, q_{11}, q_{21}, q_{22}, q_{31}, q_{32}, q_{33}, \dots\}$, $Q' = \{q', q_1, q_2, q_3, \dots\}$, and δ and δ' are as depicted in Figure 3.2.

It is easy to see that $\mathbf{C}_{\mathcal{A}}^{\omega} = \varnothing$, even though $\mathbf{C}_{\mathcal{A}} = \{q, qaq_{11}, qaq_{21}aq_{22}, \dots\}$ is infinite. We furthermore see that $\mathbf{B}_{\mathcal{A}}^{\{a\}} = \mathbf{B}_{\mathcal{A}'}^{\{a\}} = \{\lambda, a, aa, aaa, \dots\}$, whereas $a^{\omega} \in \mathbf{B}_{\mathcal{A}'}^{\{a\},\infty} \setminus \mathbf{B}_{\mathcal{A}}^{\{a\},\infty}$. In fact, $\mathbf{B}_{\mathcal{A}}^{\Sigma,\omega} = \varnothing$. □

By considering automata with a possibly infinite set of states we have chosen a computationally very powerful model. Any given Turing machine $\mathcal{M}$ can be unfolded into an automaton $\mathcal{A}$ that has the same behavior: $\mathcal{A}$ has all possible configurations of $\mathcal{M}$ as its set of states and a transition from a state C to C' with label p whenever $\mathcal{M}$ can move from configuration C to configuration C' by executing instruction p.

A direct consequence is that many problems or questions concerning automata that are decidable for finite automata are now undecidable, e.g., there exists no effective procedure for deciding for a given automaton whether or not a given state can be reached by a computation that starts from the initial state. If this problem would be decidable, then an effective decision procedure for the halting problem for Turing machines would exist, which is known to be undecidable.

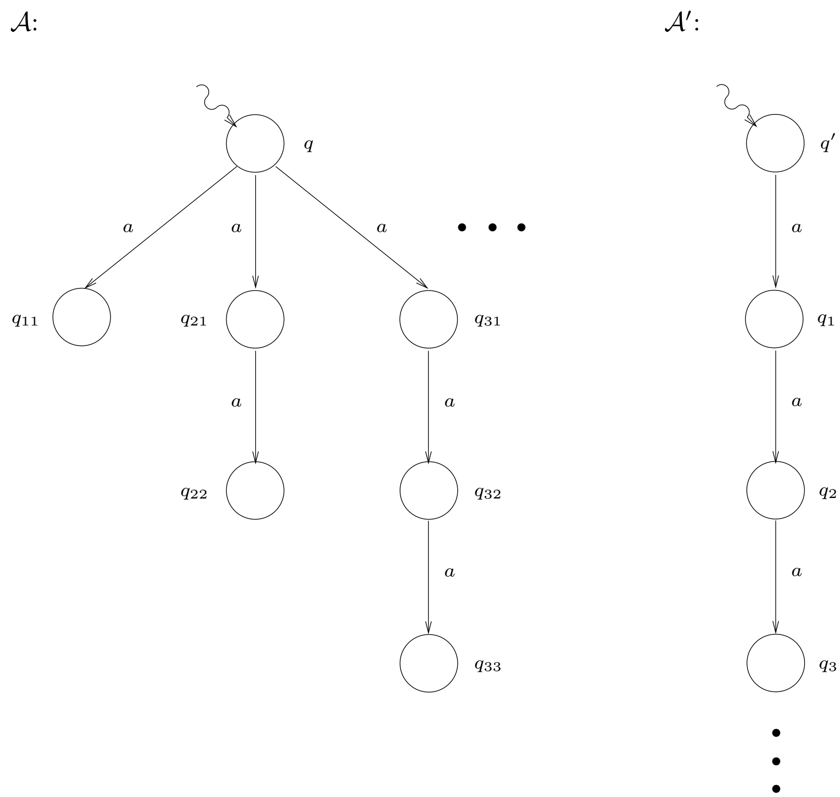

Fig. 3.2. Automata $\mathcal{A}$ and $\mathcal{A}'$.

3.2 Properties of Automata

In this section we discuss some basic notions for automata. In three subsections we consider reduced versions of automata, the enabling of actions in automata, and deterministic automata.

3.2.1 Reduced Versions

An automaton may have states, actions, or transitions that are "superfluous" in the sense that they do not occur in any computation of the automaton. Thus for the description and investigation of the dynamic — behavioral — properties of an automaton these elements are often not relevant and may be ignored.

In this subsection we introduce and relate to each other various reduced versions of an automaton. A reduced version of an automaton has less states, actions, or transitions than, but the same set of computations as, the original automaton.

We begin by identifying those elements of an automaton that are crucial for its set of computations and behavior, and which thus cannot be omitted from an automaton without affecting its set of computations and behavior.

Definition 3.2.1. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

(1) a state $q \in Q$ is reachable *(in $\mathcal{A}$) if there exists a computation $\alpha \in \mathbf{C}^{\infty}_{\mathcal{A}}$ such that $\alpha = \beta q \gamma$ for some $\beta \in (Q\Sigma)^*$ and $\gamma \in (\Sigma Q)^{\infty}$,*

(2) an action $a \in \Sigma$ is active *(in $\mathcal{A}$) if there exists a computation $\alpha \in \mathbf{C}^{\infty}_{\mathcal{A}}$ such that $\alpha = \beta a \gamma$ for some $\beta \in I(\Sigma Q)^*$ and $\gamma \in Q(\Sigma Q)^{\infty}$, and*

(3) a transition $(q, a, q') \in \delta$ is useful *(in $\mathcal{A}$) if there exists a computation $\alpha \in \mathbf{C}^{\infty}_{\mathcal{A}}$ such that $\alpha = \beta q a q' \gamma$ for some $\beta \in (Q\Sigma)^*$ and $\gamma \in (\Sigma Q)^{\infty}$.* □

By Definition 3.1.7, an action can occur in a (Θ-)record of an automaton if and only if it occurs in a computation of that automaton (and belongs to Θ). It thus suffices to focus on computations only and there is no need for an additional definition for actions occurring in the (Θ-)behavior of an automaton.

Every occurrence of a state in a computation marks the end of a finite computation (cf. the proof of Lemma 3.1.3). Thus a state is reachable if and only if it can be reached as a result of a finite computation. Recall that the initial states are always reachable by a trivial computation. Moreover, as an immediate consequence of their definitions, it follows that reachability of states, activity of actions, and usefulness of transitions can be established by following the paths laid out by the labeled transitions starting from initial states. However, one should keep in mind that — since no a priori constraints are imposed on the state space, the alphabet, and the set of transitions of an automaton — this is in general not an effective procedure.

Lemma 3.2.2. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

(1) a state $q \in Q$ is reachable in $\mathcal{A}$ if and only if there exists a finite computation $\alpha \in \mathbf{C}_{\mathcal{A}}$ such that $\alpha = \beta q$ for some $\beta \in (Q\Sigma)^*$,

(2) a transition $(q, a, q') \in \delta$ is useful in $\mathcal{A}$ if and only if q is reachable in $\mathcal{A}$,

(3) an action $a \in \Sigma$ is active in $\mathcal{A}$ if and only if there exists a useful transition $(q, a, q') \in \delta$, and

(4) if $(q, a, q') \in \delta$ is useful in $\mathcal{A}$, then q' is reachable in $\mathcal{A}$ and a is active in $\mathcal{A}$. □

Definition 3.2.3. *Let $\mathcal{A}$ be an automaton. Then*

(1) its set of reachable states is denoted by $Q_{\mathcal{A},S}$,

(2) its set of active actions is denoted by $\Sigma_{\mathcal{A},A}$, and

(3) its set of useful transitions is denoted by $\delta_{\mathcal{A},T}$. □

Whenever $\mathcal{A}$ is clear from the context, then we often simply use Q_S, Σ_A, and δ_T rather than $Q_{\mathcal{A},S}$, $\Sigma_{\mathcal{A},A}$, and $\delta_{\mathcal{A},T}$.

An immediate consequence of these definitions is the fact that the set of computations of an arbitrary automaton contains the set $\mathbf{C}_{\mathcal{A}}$ of computations of a given automaton $\mathcal{A}$, if and only if $Q_{\mathcal{A},S}$ is contained in its set of reachable states, $\Sigma_{\mathcal{A},A}$ is contained in its set of active actions, $\delta_{\mathcal{A},T}$ is contained in its set of useful transitions, and the initial states of $\mathcal{A}$ are among its initial states.

Lemma 3.2.4. *Let $\mathcal{A}$ and $\mathcal{A}'$ be two automata with sets of initial states $I_{\mathcal{A}}$ and $I_{\mathcal{A}'}$, respectively. Then*

> $\mathbf{C}_{\mathcal{A}} \subseteq \mathbf{C}_{\mathcal{A}'}$ *if and only if* $Q_{\mathcal{A},S} \subseteq Q_{\mathcal{A}',S}$, $\Sigma_{\mathcal{A},A} \subseteq \Sigma_{\mathcal{A}',A}$, $\delta_{\mathcal{A},T} \subseteq \delta_{\mathcal{A}',T}$, *and* $I_{\mathcal{A}} \subseteq I_{\mathcal{A}'}$. □

The reduced versions of automata we are about to define will again be automata. Since they are the result of omitting — and not of adding — certain elements, any reduced version of an automaton will always be *contained in* the original automaton in the following sense.

Definition 3.2.5. *Let $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ be two automata. Then*

> $\mathcal{A}_1$ *is* contained in $\mathcal{A}_2$, *denoted by* $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$, *if* $Q_1 \subseteq Q_2$, $\Sigma_1 \subseteq \Sigma_2$, $\delta_1 \subseteq \delta_2$, *and* $I_1 \subseteq I_2$. □

The containment relation $\sqsubseteq$ is reflexive and transitive and hence a partial order on automata. Although it would be natural to say that $\mathcal{A}_1$ is a "subautomaton" of $\mathcal{A}_2$ whenever $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$ holds, we refrain from doing so. The reason being that this might lead to confusion with the notion of subautomaton that we will introduce later in the context of synchronized automata.

Containment of one automaton in another implies that the first automaton has no other (initial) states, actions, or transitions than those already present in the second automaton. Consequently, it will also have no other computations.

Lemma 3.2.6. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two automata. Then* *if* $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$*, then* $\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$. □

Note that by Lemma 3.1.5, $\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$ implies $\mathbf{C}^{\omega}_{\mathcal{A}_1} \subseteq \mathbf{C}^{\omega}_{\mathcal{A}_2}$ and it thus suffices to refer to finite computations only.

Since an automaton may have states, actions, and transitions that never occur in its computations, this statement cannot be reversed unless the condition of containment is weakened by relating to initial states and useful transitions only.

Lemma 3.2.7. *Let* $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ *and* $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ *be two automata. Then*

$\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$ *if and only if* $I_1 \subseteq I_2$ *and* $\delta_{\mathcal{A}_1,T} \subseteq \delta_2$. □

A reduced version $\mathcal{A}'$ of an automaton $\mathcal{A}$ lacks certain elements of $\mathcal{A}$, but should still define the same set of computations. Hence we require that $\mathcal{A}'$ is an automaton. Furthermore, from here on we will focus on finite computations. This is sufficient because according to Theorem 3.1.6 and Corollary 3.1.11, equality of the sets of finite computations of $\mathcal{A}$ and $\mathcal{A}'$ guarantees that also the sets of all computations of $\mathcal{A}$ and $\mathcal{A}'$ will be the same, as well as their Θ-behavior (for every set of actions Θ).

We distinguish three different criteria that can be used to reduce an automaton. We define separately reductions based on states, on actions, and on transitions, and subsequently we combine them. Action reductions and transition reductions are both described relative to a given set Θ of actions, similar to the definitions of the Θ-records and Θ-behavior of an automaton.

We begin by introducing the Θ-*action-reduced version of* an automaton $\mathcal{A}$, which is defined by omitting from the set of actions of $\mathcal{A}$ those actions from Θ that are not active in $\mathcal{A}$. Thus also the transitions of $\mathcal{A}$ which are labeled with an action from Θ that is not active in $\mathcal{A}$, will be omitted.

Definition 3.2.8. *Let* $\mathcal{A} = (Q, \Sigma, \delta, I)$ *be an automaton and let* Θ *be an alphabet disjoint from* Q*. Then*

(1) the Θ-action-reduced version of $\mathcal{A}$ *is the automaton denoted by* $\mathcal{A}^{\Theta}_{A}$ *and is defined as* $\mathcal{A}^{\Theta}_{A} = (Q, \Sigma^{\Theta}_{\mathcal{A},A}, \delta^{\Theta}_{\mathcal{A},A}, I)$*, where*

$$\Sigma^{\Theta}_{\mathcal{A},A} = \{a \in \Sigma \mid a \in \Theta \Rightarrow a \in \Sigma_{\mathcal{A},A}\} \text{ and}$$

$$\delta^{\Theta}_{\mathcal{A},A} = \delta \cap (Q \times \Sigma^{\Theta}_{\mathcal{A},A} \times Q), \text{ and}$$

(2) $\mathcal{A}$ *is* Θ*-action reduced if* $\mathcal{A} = \mathcal{A}^{\Theta}_{A}$. □

Whenever the automaton $\mathcal{A}$ is clear from the context, then we may simply write Σ_A^Θ and δ_A^Θ rather than $\Sigma_{\mathcal{A},A}^\Theta$ and $\delta_{\mathcal{A},A}^\Theta$, respectively.

Note that $\Sigma_A^\varnothing = \Sigma$ and $\Sigma_A^\Sigma = \Sigma_A$. In general, $\Sigma_A^\Theta = (\Sigma \setminus \Theta) \cup (\Sigma_A \cap \Theta)$. Observe furthermore that in δ_A^Θ there may still be transitions labeled with a symbol from Θ which are not useful in $\mathcal{A}$. We have $\delta_A^\Theta = \{(q,a,q') \in \delta \mid a \in \Theta \Rightarrow a \in \Sigma_A\}$. Hence $\delta_A^\varnothing = \delta$ and $\delta_A^\Sigma \supseteq \delta_T$. Consequently $\mathcal{A}_A^\varnothing = \mathcal{A}$, which shows that action reduction relative to $\varnothing$ does not affect the automaton.

Next we define the *Θ-transition-reduced version of* an automaton $\mathcal{A}$. Transitions that are labeled with an action from Θ are retained only if they are useful, while all other transitions remain.

Definition 3.2.9. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ be an alphabet disjoint from Q. Then*

(1) the Θ-transition-reduced version of $\mathcal{A}$ *is the automaton denoted by $\mathcal{A}_T^\Theta$ and is defined as $\mathcal{A}_T^\Theta = (Q, \Sigma, \delta_{\mathcal{A},T}^\Theta, I)$, where*

$$\delta_{\mathcal{A},T}^\Theta = \{(q,a,q') \in \delta \mid a \in \Theta \Rightarrow (q,a,q') \in \delta_{\mathcal{A},T}\}, \text{ and}$$

(2) $\mathcal{A}$ is Θ-transition reduced if $\mathcal{A} = \mathcal{A}_T^\Theta$. □

Whenever the automaton $\mathcal{A}$ is clear from the context, then we may simply write δ_T^Θ rather than $\delta_{\mathcal{A},T}^\Theta$.

Note that $\delta_T^\varnothing = \delta$ and thus $\mathcal{A}_T^\varnothing = \mathcal{A}$. Hence transition reduction relative to $\varnothing$ does not affect the automaton. Moreover, $\delta_T^\Sigma = \delta_T$ and — in general — $\delta_T^\Theta = (\delta \setminus (Q \times \Theta \times Q)) \cup (\delta_T \cap (Q \times \Theta \times Q))$. In fact, $\delta_T \subseteq \delta_T^\Theta \subseteq \delta_A^\Theta$. In the following example we show that both of these inclusions can be proper.

Example 3.2.10. Let $\mathcal{A} = (\{p,q\}, \{a,b\}, \delta, \{p\})$, with $\delta = \{(p,a,p), (q,a,q), (q,b,p)\}$, be an automaton. It is depicted in Figure 3.3(a).

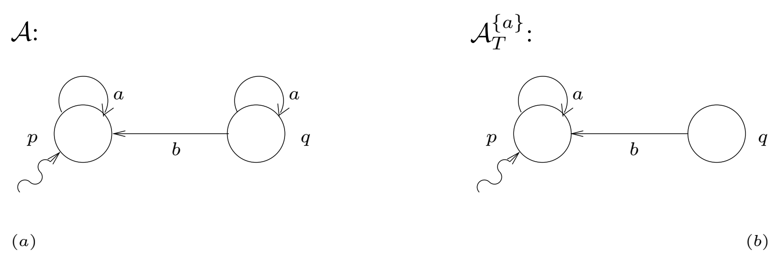

Fig. 3.3. Automata $\mathcal{A}$ and $\mathcal{A}_T^{\{a\}}$.

It is easy to see that $\delta_T = \{(p,a,p)\}$, i.e. $\mathcal{A}$ has only one useful transition. This implies that $\Sigma_A = \{a\}$ and thus $\delta_A^{\{a\}} = \delta$, i.e. $\mathcal{A}$ is $\{a\}$-action reduced: $\mathcal{A}_A^{\{a\}} = \mathcal{A}$. It also implies that the $\{a\}$-transition-reduced version of $\mathcal{A}$ is $\mathcal{A}_T^{\{a\}} = (\{p,q\},\{a,b\},\delta_T^{\{a\}},\{p\})$, with $\delta_T^{\{a\}} = \{(p,a,p),(q,b,p)\}$, as depicted in Figure 3.3(b). Consequently, $\delta_T \subsetneq \delta_T^{\{a\}} \subsetneq \delta_A^{\{a\}}$. □

Lemma 3.2.11. *Let $\mathcal{A} = (Q,\Sigma,\delta,I)$ be an automaton and let Θ be an alphabet disjoint from Q. Let $\mathcal{A}_A^\Theta = (Q,\Sigma_A^\Theta,\delta_A^\Theta,I)$ and let $\mathcal{A}_T^\Theta = (Q,\Sigma,\delta_T^\Theta,I)$. Then*

(1) $\delta_T = \delta_T^\Theta \setminus \{(q,a,q') \in \delta \mid a \notin \Theta$ and $(q,a,q') \notin \delta_T\}$ and

(2) $\delta_T^\Theta = \delta_A^\Theta \setminus \{(q,a,q') \in \delta \mid a \in \Theta$ and $(q,a,q') \notin \delta_T\}$.

Proof. (1) ($\subseteq$) Immediate because δ_T consists only of useful transitions.

($\supseteq$) This follows from the observation that all transitions $(q,a,q') \in \delta_T^\Theta$, with $a \in \Theta$, are useful in $\mathcal{A}$.

(2) ($\subseteq$) Let $(q,a,q') \in \delta_T^\Theta$. Thus $(q,a,q') \in \delta$.

If $a \notin \Theta$, then $a \in \Sigma_A^\Theta$ and so $(q,a,q') \in \delta_A^\Theta$.

If $a \in \Theta$, then $(q,a,q') \in \delta_T$.

Hence $(q,a,q') \in \delta_A^\Theta \setminus \{(q,a,q') \in \delta \mid a \in \Theta$ and $(q,a,q') \notin \delta_T\}$.

($\supseteq$) Let $(q,a,q') \in \delta_A^\Theta$ be such that $a \in \Theta$ implies $(q,a,q') \in \delta_T$. Then by Definition 3.2.9(1), $(q,a,q') \in \delta_T^\Theta$. □

It is immediate from the definitions that for every automaton $\mathcal{A}$ and for every set of actions Θ, both the Θ-action-reduced version $\mathcal{A}_A^\Theta$ of $\mathcal{A}$ and its Θ-transition-reduced version $\mathcal{A}_T^\Theta$ are contained in $\mathcal{A}$. Consequently, $\mathbf{C}_{\mathcal{A}_A^\Theta} \subseteq \mathbf{C}_\mathcal{A}$ and $\mathbf{C}_{\mathcal{A}_T^\Theta} \subseteq \mathbf{C}_\mathcal{A}$ always hold due to Lemma 3.2.6. In addition, Lemma 3.2.11 implies that the transition relations of both $\mathcal{A}_A^\Theta$ and $\mathcal{A}_T^\Theta$ contain δ_T. Since $\mathcal{A}_A^\Theta$ and $\mathcal{A}_T^\Theta$ have the same initial states as $\mathcal{A}$, it follows from Lemma 3.2.7 that $\mathbf{C}_\mathcal{A} \subseteq \mathbf{C}_{\mathcal{A}_A^\Theta}$ and $\mathbf{C}_\mathcal{A} \subseteq \mathbf{C}_{\mathcal{A}_T^\Theta}$.

We conclude that Definitions 3.2.8 and 3.2.9 thus satisfy the requirement that the computations of an automaton are not affected by the reduction.

Theorem 3.2.12. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

$$\mathbf{C}_\mathcal{A} = \mathbf{C}_{\mathcal{A}_A^\Theta} = \mathbf{C}_{\mathcal{A}_T^\Theta}.$$ □

An immediate consequence of this theorem is that an automaton, its Θ-action-reduced version, and its Θ-transition-reduced version, all three have the same reachable states, active actions, and useful transitions.

Corollary 3.2.13. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

(1) $Q_{\mathcal{A},S} = Q_{\mathcal{A}^{\Theta}_{A},S} = Q_{\mathcal{A}^{\Theta}_{T},S}$,

(2) $\Sigma_{\mathcal{A},A} = \Sigma_{\mathcal{A}^{\Theta}_{A},A} = \Sigma_{\mathcal{A}^{\Theta}_{T},A}$, *and*

(3) $\delta_{\mathcal{A},T} = \delta_{\mathcal{A}^{\Theta}_{A},T} = \delta_{\mathcal{A}^{\Theta}_{T},T}$. □

In Definitions 3.2.8 and 3.2.9, the reduced versions of an automaton are defined relative to some given alphabet Θ. From both definitions it is however immediately clear that actions which do belong to Θ but not to the alphabet of the automaton, are not even considered.

Lemma 3.2.14. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

(1) $\mathcal{A}^{\Theta}_{A} = \mathcal{A}^{\Theta}_{T} = \mathcal{A}$ *whenever* $\Theta \cap \Sigma = \varnothing$,

(2) $\mathcal{A}^{\Theta}_{A} = \mathcal{A}^{\Theta \cap \Sigma}_{A}$, *and*

(3) $\mathcal{A}^{\Theta}_{T} = \mathcal{A}^{\Theta \cap \Sigma}_{T}$. □

In addition, both in Definition 3.2.8 and in Definition 3.2.9 the role of each action is assessed on an individual basis, and reduction relative to any action is independent of the role of other actions.

Example 3.2.15. (Example 3.2.10 continued) Let $\mathcal{A}^2$ be the automaton obtained from $\mathcal{A}$ by adding the transition (p, c, p) to its transition relation. Then $\Sigma_{\mathcal{A}^2,A} = \{a, c\}$ are the active actions of $\mathcal{A}^2$. Hence $\mathcal{A}^2$ is $\{a\}$-action reduced, $\{c\}$-action reduced, and $\{a, c\}$-action reduced. Since b is not active in $\mathcal{A}^2$ it follows that $\mathcal{A}^2$ is neither $\{b\}$-action reduced, nor $\{a, b\}$-action reduced, nor $\{b, c\}$-action reduced.

The useful transitions of $\mathcal{A}^2$ are $\delta_{\mathcal{A}^2,T} = \{(p, a, p), (p, c, p)\}$. Hence $\mathcal{A}^2$ is not $\{a\}$-transition reduced as (q, a, q) is not useful in $\mathcal{A}^2$. Since also (q, b, p) is not useful in $\mathcal{A}^2$, it follows that $\mathcal{A}^2$ is neither $\{b\}$-transition reduced nor $\{a, b\}$-transition reduced. Because the only c-transition *is* useful in $\mathcal{A}^2$, we do have that $\mathcal{A}^2$ is $\{c\}$-transition reduced. However, $\mathcal{A}^2$ is neither $\{a, c\}$-transition reduced nor $\{b, c\}$-transition reduced. □

Consequently, as formally stated in the next lemma, the order in which actions are considered is irrelevant and has no effect on the resulting reduced version.

Lemma 3.2.16. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton, let Θ be an alphabet disjoint from Q, and let $\Theta_1, \Theta_2 \subseteq \Theta$ be such that $\Theta = \Theta_1 \cup \Theta_2$. Then*

(1) $(\mathcal{A}_A^{\Theta_1})_A^{\Theta_2} = \mathcal{A}_A^{\Theta}$ *and*

(2) $(\mathcal{A}_T^{\Theta_1})_T^{\Theta_2} = \mathcal{A}_T^{\Theta}$.

Proof. (1) Let $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_A^{\Theta_1}, \delta_A^{\Theta_1}, I)$, $(\mathcal{A}_A^{\Theta_1})_A^{\Theta_2} = (Q, (\Sigma_A^{\Theta_1})_A^{\Theta_2}, (\delta_A^{\Theta_1})_A^{\Theta_2}, I)$, and $\mathcal{A}_A^{\Theta_1 \cup \Theta_2} = \mathcal{A}_A^{\Theta} = (Q, \Sigma_A^{\Theta}, \delta_A^{\Theta}, I)$. First we prove that $(\Sigma_A^{\Theta_1})_A^{\Theta_2} = \Sigma_A^{\Theta}$.

Let $a \in (\Sigma_A^{\Theta_1})_A^{\Theta_2}$. Then $a \in \Sigma_A^{\Theta_1}$, which implies that $a \in \Sigma$.
If $a \notin \Theta$, then $a \in \Sigma_A^{\Theta}$ by definition.
If $a \in \Theta_1$, then $a \in \Sigma_{\mathcal{A},A}$ because $a \in \Sigma_A^{\Theta_1}$, and hence $a \in \Sigma_A^{\Theta}$.
If $a \in \Theta_2$, then $a \in \Sigma_{\mathcal{A}_A^{\Theta_1},A}$ because $a \in (\Sigma_A^{\Theta_1})_A^{\Theta_2}$. By Corollary 3.2.13 it follows that $a \in \Sigma_{\mathcal{A},A}$ and hence $a \in \Sigma_A^{\Theta}$.

Now assume that $a \in \Sigma_A^{\Theta}$. Then $a \in \Sigma$.
If $a \notin \Theta$, then by definition $a \in \Sigma_A^{\Theta_1}$ and $a \in (\Sigma_A^{\Theta_1})_A^{\Theta_2}$.
If $a \in \Theta$, then $a \in \Sigma_{\mathcal{A},A}$ because $a \in \Sigma_A^{\Theta}$ and by Corollary 3.2.13 also $a \in \Sigma_{\mathcal{A}_A^{\Theta_1},A}$. Hence $a \in \Sigma_A^{\Theta_1}$ and $a \in (\Sigma_A^{\Theta_1})_A^{\Theta_2}$.

Having established $(\Sigma_A^{\Theta_1})_A^{\Theta_2} = \Sigma_A^{\Theta}$ we immediately obtain that $(\delta_A^{\Theta_1})_A^{\Theta_2} = \delta_A^{\Theta_1} \cap (Q \times (\Sigma_A^{\Theta_1})_A^{\Theta_2} \times Q) = (\delta \cap (Q \times \Sigma_A^{\Theta_1} \times Q)) \cap (Q \times \Sigma_A^{\Theta} \times Q)$. Since $\Sigma_A^{\Theta} \subseteq \Sigma_A^{\Theta_1}$ this yields $(\delta_A^{\Theta_1})_A^{\Theta_2} = \delta \cap (Q \times \Sigma_A^{\Theta} \times Q) = \delta_A^{\Theta}$.

(2) Let $\mathcal{A}_T^{\Theta_1} = (Q, \Sigma, \delta_T^{\Theta_1}, I)$, let $(\mathcal{A}_T^{\Theta_1})_T^{\Theta_2} = (Q, \Sigma, (\delta_T^{\Theta_1})_T^{\Theta_2}, I)$, and let $\mathcal{A}_T^{\Theta_1 \cup \Theta_2} = \mathcal{A}_T^{\Theta} = (Q, \Sigma, \delta_T^{\Theta}, I)$. We prove that $(\delta_T^{\Theta_1})_T^{\Theta_2} = \delta_T^{\Theta}$.

Let $(q, a, q') \in (\delta_T^{\Theta_1})_T^{\Theta_2}$. Then $(q, a, q') \in \delta_T^{\Theta_1}$, which implies $(q, a, q') \in \delta$.
If $a \notin \Theta$, then $(q, a, q') \in \delta_T^{\Theta}$ by definition.
If $a \in \Theta_1$, then $(q, a, q') \in \delta_{\mathcal{A},T}$ because $(q, a, q') \in \delta_T^{\Theta_1}$, and hence $(q, a, q') \in \delta_T^{\Theta}$.
If $a \in \Theta_2$, then $(q, a, q') \in \delta_{\mathcal{A}_T^{\Theta_1},T}$ because $(q, a, q') \in (\delta_T^{\Theta_1})_T^{\Theta_2}$. By Corollary 3.2.13 it follows that $(q, a, q') \in \delta_{\mathcal{A},T}$ and hence $(q, a, q') \in \delta_T^{\Theta}$.

Now assume that $(q, a, q') \in \delta_T^{\Theta}$. Thus $(q, a, q') \in \delta$.
If $a \notin \Theta$, then by definition $(q, a, q') \in \delta_T^{\Theta_1}$ and $(q, a, q') \in (\delta_T^{\Theta_1})_T^{\Theta_2}$.
If $a \in \Theta$, then $(q, a, q') \in \delta_{\mathcal{A},T}$ because $(q, a, q') \in \delta_T^{\Theta}$. Thus by Corollary 3.2.13 we have $(q, a, q') \in \delta_{\mathcal{A}_T^{\Theta_1},T}$. Hence $(q, a, q') \in \delta_T^{\Theta_1}$ and $(q, a, q') \in (\delta_T^{\Theta_1})_T^{\Theta_2}$. □

An immediate consequence of this lemma is that the Θ-action-reduced and the Θ-transition-reduced versions of an automaton are indeed Θ-action-reduced and Θ-transition-reduced automata, respectively.

Theorem 3.2.17. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

(1) $\mathcal{A}_A^{\Theta}$ *is* Θ*-action reduced and*

(2) $\mathcal{A}_T^{\Theta}$ *is* Θ*-transition reduced.*

Proof. $\mathcal{A}_A^{\Theta} = (\mathcal{A}_A^{\Theta})_A^{\Theta}$ and $\mathcal{A}_T^{\Theta} = (\mathcal{A}_T^{\Theta})_T^{\Theta}$ follow directly from Lemma 3.2.16. □

A more general consequence is that reduction relative to more actions has a cumulative effect, but only for those actions that have not yet been considered there is an effect.

Lemma 3.2.18. *Let* $\mathcal{A} = (Q, \Sigma, \delta, I)$ *be an automaton and let* Θ_1, Θ_2 *be alphabets disjoint from* Q *and such that* $(\Theta_1 \cap \Sigma) \subseteq \Theta_2$*. Then*

(1) (i) $(\mathcal{A}_A^{\Theta_2})_A^{\Theta_1} = \mathcal{A}_A^{\Theta_2}$*,*

(ii) $\mathcal{A}_A^{\Theta_2} \sqsubseteq \mathcal{A}_A^{\Theta_1}$*, and*

(iii) if $\mathcal{A} = \mathcal{A}_A^{\Theta_2}$*, then* $\mathcal{A} = \mathcal{A}_A^{\Theta_1}$*, and*

(2) (i) $(\mathcal{A}_T^{\Theta_2})_T^{\Theta_1} = \mathcal{A}_T^{\Theta_2}$*,*

(ii) $\mathcal{A}_T^{\Theta_2} \sqsubseteq \mathcal{A}_T^{\Theta_1}$*, and*

(iii) if $\mathcal{A} = \mathcal{A}_T^{\Theta_2}$*, then* $\mathcal{A} = \mathcal{A}_T^{\Theta_1}$*.*

Proof. (1) (i) Let Σ' be the alphabet of $\mathcal{A}_A^{\Theta_2}$. Thus $\Sigma' \subseteq \Sigma$ and hence $\Theta_1 \cap \Sigma' \subseteq \Theta_1 \cap \Sigma \subseteq \Theta_2$. From Lemma 3.2.14(2) we know that $(\mathcal{A}_A^{\Theta_2})_A^{\Theta_1} = (\mathcal{A}_A^{\Theta_2})_A^{\Theta_1 \cap \Sigma'}$. Combining these facts with Lemma 3.2.16(1) yields $(\mathcal{A}_A^{\Theta_2})_A^{\Theta_1} = (\mathcal{A}_A^{\Theta_2})_A^{\Theta_1 \cap \Sigma'} = \mathcal{A}_A^{\Theta_2 \cup (\Theta_1 \cap \Sigma')} = \mathcal{A}_A^{\Theta_2}$.

(ii) Lemma 3.2.16(1) implies that $(\mathcal{A}_A^{\Theta_2})_A^{\Theta_1} = (\mathcal{A}_A^{\Theta_1})_A^{\Theta_2}$. Thus, by the above, $\mathcal{A}_A^{\Theta_2} = (\mathcal{A}_A^{\Theta_1})_A^{\Theta_2}$. Since reduction always yields an automaton contained in the original one, we now have $\mathcal{A}_A^{\Theta_2} = (\mathcal{A}_A^{\Theta_1})_A^{\Theta_2} \sqsubseteq \mathcal{A}_A^{\Theta_1}$.

(iii) Let $\mathcal{A} = \mathcal{A}_A^{\Theta_2}$. Then using (i) above we conclude that $\mathcal{A} = \mathcal{A}_A^{\Theta_2} = (\mathcal{A}_A^{\Theta_2})_A^{\Theta_1} = \mathcal{A}_A^{\Theta_1}$.

(2) (i) First we note that Σ is the alphabet of $\mathcal{A}_T^{\Theta_2}$. By Lemmata 3.2.13(3) and 3.2.16(2) we have $(\mathcal{A}_T^{\Theta_2})_T^{\Theta_1} = (\mathcal{A}_T^{\Theta_2})_T^{\Theta_1 \cap \Sigma} = \mathcal{A}_T^{\Theta_2 \cup (\Theta_1 \cap \Sigma)} = \mathcal{A}_T^{\Theta_2}$.

(ii) Lemma 3.2.16(1) implies that $(\mathcal{A}_T^{\Theta_2})_T^{\Theta_1} = (\mathcal{A}_T^{\Theta_1})_T^{\Theta_2}$. Then, by the above, $\mathcal{A}_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_1})_T^{\Theta_2}$. Since the transition reductions always yield an automaton contained in the original one, we now have $\mathcal{A}_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_1})_T^{\Theta_2} \sqsubseteq \mathcal{A}_T^{\Theta_1}$.

(iii) Let $\mathcal{A} = \mathcal{A}_T^{\Theta_2}$. Then from (2) (i) we conclude that $\mathcal{A} = \mathcal{A}_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_2})_T^{\Theta_1} = \mathcal{A}_T^{\Theta_2}$. □

Since all actions of an automaton $\mathcal{A}$ with alphabet Σ have been considered, a further reduction with respect to actions of $\mathcal{A}_A^{\Sigma}$ or a further reduction with respect to transitions of $\mathcal{A}_T^{\Sigma}$ thus has no additional effect.

Theorem 3.2.19. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ be an alphabet disjoint from Q. Then*

(1) $\mathcal{A}_A^\Sigma \sqsubseteq \mathcal{A}_A^\Theta$ *and*

(2) $\mathcal{A}_T^\Sigma \sqsubseteq \mathcal{A}_T^\Theta$. □

From Lemma 3.2.6 it follows that whenever an automaton $\mathcal{A}_1$ is contained in an automaton $\mathcal{A}_2$, then all elements which are superfluous in $\mathcal{A}_2$ will certainly be superfluous in $\mathcal{A}_1$. This implies that action reduction and transition reduction are monotonous operations with respect to containment ($\sqsubseteq$).

Lemma 3.2.20. *Let $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ be two automata such that $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$ and let Θ be an alphabet disjoint from $Q_1 \cup Q_2$. Then*

(1) $(\mathcal{A}_1)_A^\Theta \sqsubseteq (\mathcal{A}_2)_A^\Theta$ *and*

(2) $(\mathcal{A}_1)_T^\Theta \sqsubseteq (\mathcal{A}_2)_T^\Theta$.

Proof. (1) Let $(\mathcal{A}_1)_A^\Theta = (Q_1, (\Sigma_1)_A^\Theta, (\delta_1)_A^\Theta, I_1)$ and let $(\mathcal{A}_2)_A^\Theta = (Q_2, (\Sigma_2)_A^\Theta, (\delta_2)_A^\Theta, I_2)$. Since $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$ we know that $Q_1 \subseteq Q_2$ and $I_1 \subseteq I_2$. By Lemma 3.2.6, $\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$ and thus every action that is active in $\mathcal{A}_1$ is also active in $\mathcal{A}_2$. Hence $(\Sigma_1)_A^\Theta \subseteq (\Sigma_2)_A^\Theta$. This in turn implies that $(\delta_1)_A^\Theta \subseteq (\delta_2)_A^\Theta$ because the transition relation of $\mathcal{A}_1$ is contained in that of $\mathcal{A}_2$. We conclude that $(\mathcal{A}_1)_A^\Theta \sqsubseteq (\mathcal{A}_2)_A^\Theta$.

(2) Let $(\mathcal{A}_1)_T^\Theta = (Q_1, \Sigma_1, (\delta_1)_T^\Theta, I_1)$ and let $(\mathcal{A}_2)_T^\Theta = (Q_2, \Sigma_2, (\delta_2)_T^\Theta, I_2)$. Since $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$ we know that $Q_1 \subseteq Q_2$, $\Sigma_1 \subseteq \Sigma_2$, and $I_1 \subseteq I_2$. From the fact that $\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$ by Lemma 3.2.6, we deduce that every transition that is useful in $\mathcal{A}_1$ is useful also in $\mathcal{A}_2$. Hence $(\delta_1)_T^\Theta \subseteq (\delta_2)_T^\Theta$ and we conclude that $(\mathcal{A}_1)_T^\Theta \sqsubseteq (\mathcal{A}_2)_T^\Theta$. □

Given an alphabet Θ, an automaton $\mathcal{A}$ may contain many automata that are Θ-action reduced or Θ-transition reduced. We can now show that among these $\mathcal{A}_A^\Theta$ and $\mathcal{A}_T^\Theta$, respectively, are the largest (with respect to containment).

Lemma 3.2.21. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Let $\mathcal{A}' \sqsubseteq \mathcal{A}$. Then*

(1) if $\mathcal{A}'$ is Θ-action reduced, then $\mathcal{A}' \sqsubseteq \mathcal{A}_A^\Theta$, and

(2) if $\mathcal{A}'$ is Θ-transition reduced, then $\mathcal{A}' \sqsubseteq \mathcal{A}_T^\Theta$.

Proof. Since $\mathcal{A}' \sqsubseteq \mathcal{A}$, Lemma 3.2.20 implies $(\mathcal{A}')_A^\Theta \sqsubseteq \mathcal{A}_A^\Theta$ and $(\mathcal{A}')_T^\Theta \sqsubseteq \mathcal{A}_T^\Theta$. Hence if $\mathcal{A}' = (\mathcal{A}')_A^\Theta$, then $\mathcal{A}' \sqsubseteq \mathcal{A}_A^\Theta$, and if $\mathcal{A}' = (\mathcal{A}')_T^\Theta$, then $\mathcal{A}' \sqsubseteq \mathcal{A}_T^\Theta$. □

Theorem 3.2.22. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

(1) $\mathcal{A}_A^\Theta$ is the largest Θ-action-reduced automaton contained in $\mathcal{A}$ and

(2) $\mathcal{A}_T^\Theta$ is the largest Θ-transition-reduced automaton contained in $\mathcal{A}$.

Proof. Immediate from Theorem 3.2.17 and Lemma 3.2.21. □

For a given automaton $\mathcal{A}$ and an alphabet Θ, the difference between $\mathcal{A}$ and $\mathcal{A}_A^\Theta$ and between $\mathcal{A}$ and $\mathcal{A}_T^\Theta$ is thus minimal. Nevertheless, by definition, the remaining actions of Θ in $\mathcal{A}_A^\Theta$ are active in both $\mathcal{A}$ and $\mathcal{A}_A^\Theta$, and the remaining transitions in $\mathcal{A}_T^\Theta$ with a label from Θ are useful in both $\mathcal{A}$ and $\mathcal{A}_T^\Theta$. Hence, a further reduction of $\mathcal{A}_A^\Theta$ or $\mathcal{A}_T^\Theta$ that will not affect the computations is only feasible when other elements are considered. We already observed in Theorem 3.2.19 that in case all actions of $\mathcal{A}$ have been involved in action reduction (yielding $\mathcal{A}_A^\Sigma$) or transition reduction (yielding $\mathcal{A}_T^\Sigma$), further action reduction or transition reduction, respectively, will have no additional effect.

From Definitions 3.2.8 and 3.2.9 and the observations immediately following these definitions we know that given an automaton $\mathcal{A} = (Q, \Sigma, \delta, I)$ we have $\mathcal{A}_A^\Sigma = (Q, \Sigma_{\mathcal{A},A}, \delta_A^\Sigma, I)$ and $\mathcal{A}_T^\Sigma = (Q, \Sigma, \delta_{\mathcal{A},T}, I)$, with $\Sigma_{\mathcal{A},A} \subseteq \Sigma$ and $\delta_{\mathcal{A},T} \subseteq \delta_A^\Sigma$. Hence $\mathcal{A}_A^\Sigma$ and $\mathcal{A}_T^\Sigma$ are in general incomparable. We now consider the effect of combining action and transition reductions.

Lemma 3.2.23. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ_1, Θ_2 be alphabets disjoint from Q. Then*

$$(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_2})_A^{\Theta_1}.$$

Proof. Let $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_A^{\Theta_1}, \delta_A^{\Theta_1}, I)$ and $\mathcal{A}_T^{\Theta_2} = (Q, \Sigma, \delta_T^{\Theta_2}, I)$. Then $(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}$ $= (Q, \Sigma_A^{\Theta_1}, \delta_2, I)$ with $\delta_2 = \{(q, a, q') \in \delta_A^{\Theta_1} \mid a \in \Theta_2 \Rightarrow (q, a, q') \in \delta_{\mathcal{A}_A^{\Theta_1},T}\}$. By Corollary 3.2.13(3), $(q, a, q') \in \delta_{\mathcal{A}_A^{\Theta_1},T}$ if and only if $(q, a, q') \in \delta_{\mathcal{A},T}$. Hence $\delta_2 = \{(q, a, q') \in \delta_A^{\Theta_1} \mid a \in \Theta_2 \Rightarrow (q, a, q') \in \delta_{\mathcal{A},T}\} = \delta_A^{\Theta_1} \cap \delta_T^{\Theta_2} = \delta_T^{\Theta_2} \cap (\delta \cap (Q \times \Sigma_A^{\Theta_1} \times Q))$. Since $\delta_T^{\Theta_2} \subseteq \delta$, we have $\delta_2 = \delta_T^{\Theta_2} \cap (Q \times \Sigma_A^{\Theta_1} \times Q)$.

Next consider $(\mathcal{A}_T^{\Theta_2})_A^{\Theta_1} = (Q, \Sigma_1, \delta_1, I)$, with $\Sigma_1 = \{a \in \Sigma \mid a \in \Theta_1 \Rightarrow a \in \Sigma_{\mathcal{A}_T^{\Theta_2},A}\}$ and $\delta_1 = \delta_T^{\Theta_2} \cap (Q \times \Sigma_1 \times Q)$. By Corollary 3.2.13(2), $a \in \Sigma_{\mathcal{A}_T^{\Theta_2},A}$ if and only if $a \in \Sigma_{\mathcal{A},A}$. Thus $\Sigma_1 = \{a \in \Sigma \mid a \in \Theta_1 \Rightarrow a \in \Sigma_{\mathcal{A},A}\} = \Sigma_A^{\Theta_1}$. Hence $\delta_1 = \delta_T^{\Theta_2} \cap (Q \times \Sigma_A^{\Theta_1} \times Q) = \delta_2$. We thus conclude that $(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_2})_A^{\Theta_1}$. □

By this lemma, the order in which action and transition reductions are applied is irrelevant. Together with Lemma 3.2.16 this implies that for every automaton $\mathcal{A}$, any finite succession of action reductions and transition reductions (relative to certain sets of actions) yields an automaton of the form $(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_2})_A^{\Theta_1}$.

Example 3.2.24. (Example 3.2.10 continued) We consider $\mathcal{A}$, as depicted in Figure 3.3(a). Since b is not active in $\mathcal{A}$, the $\{b\}$-action-reduced version of $\mathcal{A}$ is $\mathcal{A}_A^{\{b\}} = (\{p,q\},\{a\},\{(p,a,p),(q,a,q)\},\{p\})$. Because (q,a,q) is not useful in $\mathcal{A}_A^{\{b\}}$, the $\{a\}$-transition-reduced version of $\mathcal{A}_A^{\{b\}}$ is $(\mathcal{A}_A^{\{b\}})_T^{\{a\}} = (\{p,q\},\{a\},\{(p,a,p)\},\{p\})$.

Now we consider the $\{a\}$-transition-reduced version $\mathcal{A}_T^{\{a\}}$ of $\mathcal{A}$, as depicted in Figure 3.3(b). Since b is not active in $\mathcal{A}_T^{\{a\}}$, the $\{b\}$-action-reduced version of $\mathcal{A}_T^{\{a\}}$ is $(\mathcal{A}_T^{\{a\}})_A^{\{b\}} = (\mathcal{A}_A^{\{b\}})_T^{\{a\}}$. □

Theorem 3.2.25. *Let $\mathcal{A}$ be an automaton and let Θ_1, Θ_2 be alphabets disjoint from its set of states. Then*

(1) $(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}$ is the largest automaton contained in $\mathcal{A}$ that is both Θ_1-action reduced and Θ_2-transition reduced, and

(2) $\mathbf{C}_{(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}} = \mathbf{C}_\mathcal{A}$.

Proof. (1) By Lemma 3.2.23, $(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2} = (\mathcal{A}_T^{\Theta_2})_A^{\Theta_1}$. Using Lemma 3.2.16 it is easy to see that $(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}$ is both Θ_1-action reduced and Θ_2-transition reduced. Now let $\mathcal{A}_1$ be an automaton contained in $\mathcal{A}$. Then, by Lemma 3.2.20, $(\mathcal{A}_1)_A^{\Theta_1} \sqsubseteq \mathcal{A}_A^{\Theta_1}$ and thus $((\mathcal{A}_1)_A^{\Theta_1})_T^{\Theta_2} \sqsubseteq (\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}$. If $\mathcal{A}_1$ is Θ_1-action reduced and Θ_2-transition reduced, then $\mathcal{A}_1 = (\mathcal{A}_1)_A^{\Theta_1}$ and $\mathcal{A}_1 = (\mathcal{A}_1)_T^{\Theta_2}$. In that case we have $\mathcal{A}_1 = (\mathcal{A}_1)_A^{\Theta_1} = ((\mathcal{A}_1)_A^{\Theta_1})_T^{\Theta_2} \sqsubseteq (\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}$.

(2) From Theorem 3.2.12 directly follows $\mathbf{C}_{(\mathcal{A}_A^{\Theta_1})_T^{\Theta_2}} = \mathbf{C}_{\mathcal{A}_A^{\Theta_1}} = \mathbf{C}_\mathcal{A}$. □

In particular we now have that given an automaton $\mathcal{A} = (Q,\Sigma,\delta,I)$, the two automata $(\mathcal{A}_A^{\Sigma})_T^{\Sigma}$ and $(\mathcal{A}_T^{\Sigma})_A^{\Sigma}$ are the same. In fact, the definitions together with Theorem 3.2.12 and Corollary 3.2.13 imply that $(\mathcal{A}_A^{\Sigma})_T^{\Sigma} = (Q,\Sigma_{\mathcal{A},A},\delta_{\mathcal{A},T},I) = (\mathcal{A}_T^{\Sigma})_A^{\Sigma}$ and this automaton has neither superfluous actions nor superfluous transitions.

Theorem 3.2.26. *Let $\mathcal{A} = (Q,\Sigma,\delta,I)$ be an automaton. Then*

(1) $\mathcal{A}_T^{\Sigma}$ is the least automaton with set of states Q and alphabet Σ such that $\mathbf{C}_{\mathcal{A}_T^{\Sigma}} = \mathbf{C}_\mathcal{A}$, and

(2) $(\mathcal{A}_A^{\Sigma})_T^{\Sigma}$ is the least automaton with set of states Q such that $\mathbf{C}_{(\mathcal{A}_A^{\Sigma})_T^{\Sigma}} = \mathbf{C}_\mathcal{A}$.

Proof. By Theorem 3.2.12, $\mathbf{C}_{\mathcal{A}_T^\Sigma} = \mathbf{C}_\mathcal{A} = \mathbf{C}_{\mathcal{A}_A^\Sigma} = \mathbf{C}_{(\mathcal{A}_A^\Sigma)_T^\Sigma}$. As observed before, $\mathcal{A}_T^\Sigma = (Q, \Sigma, \delta_{\mathcal{A},T}, I)$ and $(\mathcal{A}_A^\Sigma)_T^\Sigma = (Q, \Sigma_{\mathcal{A},A}, \delta_{\mathcal{A},T}, I)$. Now assume that $\mathcal{A}' = (Q, \Sigma', \delta', I')$ is an automaton such that $\mathbf{C}_{\mathcal{A}'} = \mathbf{C}_\mathcal{A}$. Thus $I' = I$, $\delta_{\mathcal{A}',T} = \delta_{\mathcal{A},T}$, and $\Sigma_{\mathcal{A}',A} = \Sigma_{\mathcal{A},A}$. Since $\delta_{\mathcal{A}',T} \subseteq \delta'$ and $\Sigma_{\mathcal{A}',A} \subseteq \Sigma'$ we have $(\mathcal{A}_A^\Sigma)_T^\Sigma \sqsubseteq \mathcal{A}'$, and if $\Sigma' = \Sigma$, then we have $\mathcal{A}_T^\Sigma \sqsubseteq \mathcal{A}'$. □

Finally, we consider (additional) reductions with respect to states.

The state-reduced version of an automaton is defined by omitting the non-reachable states from its specification. Consequently, the outgoing and incoming transitions of these states are no longer proper transitions and thus disappear as well.

Definition 3.2.27. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

(1) the state-reduced version of $\mathcal{A}$ *is the automaton denoted by $\mathcal{A}_S$ and is defined as $\mathcal{A}_S = (Q_S, \Sigma, \delta_T, I)$, and*

(2) $\mathcal{A}$ is state reduced if $\mathcal{A} = \mathcal{A}_S$. □

Note that $\delta_T = \{(q, a, q') \in \delta \mid q, q' \in Q_S\}$ by Lemma 3.2.2. Exactly those transitions that are outgoing or incoming transitions of a non-reachable state of $\mathcal{A}$ have thus been omitted. Hence $\delta_T = \delta \cap (Q_S \times \Sigma \times Q_S)$ and, since $I \subseteq Q_S$, $\mathcal{A}_S$ is well defined. Now Lemma 3.2.7 immediately implies that $\mathbf{C}_\mathcal{A} \subseteq \mathbf{C}_{\mathcal{A}_S}$. Furthermore, since $\mathcal{A}_S \sqsubseteq \mathcal{A}$ we know from Lemma 3.2.6 that $\mathbf{C}_{\mathcal{A}_S} \subseteq \mathbf{C}_\mathcal{A}$.

Theorem 3.2.28. *Let $\mathcal{A}$ be an automaton. Then*

$$\mathbf{C}_\mathcal{A} = \mathbf{C}_{\mathcal{A}_S}.$$ □

Example 3.2.29. (Example 3.2.10 continued) Consider the automaton $\mathcal{A}$ depicted in Figure 3.3(a). We have seen that $\delta_T = \{(p, a, p)\}$. This implies that $Q_S = \{p\}$. Hence the state-reduced version of $\mathcal{A}$ is $\mathcal{A}_S = (\{p\}, \{a, b\}, \{(p, a, p)\}, \{p\})$ and thus $\mathbf{C}_\mathcal{A} = \mathbf{C}_{\mathcal{A}_S} = \{p, pap, papap, \ldots\}$. □

Using the notion of a state-reduced version we can now reformulate Lemmata 3.2.6 and 3.2.7.

Lemma 3.2.30. *Let $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ be two automata such that $\Sigma_1 \subseteq \Sigma_2$. Then*

$$\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2} \textit{ if and only if } (\mathcal{A}_1)_S \sqsubseteq (\mathcal{A}_2)_S.$$

Proof. (Only if) Let $\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$. Then by Lemma 3.2.7, $I_1 \subseteq I_2$ and $\delta_{\mathcal{A}_1,T} \subseteq \delta_2$. In fact, $\delta_{\mathcal{A}_1,T} \subseteq \delta_{\mathcal{A}_2,T}$ holds because all transitions in $\delta_{\mathcal{A}_1,T}$ are used in the computations of $\mathcal{A}_2$. From $\delta_{\mathcal{A}_1,T} \subseteq \delta_{\mathcal{A}_2,T}$ and Lemma 3.2.2 now follows that we also have $Q_{\mathcal{A}_1,S} \subseteq Q_{\mathcal{A}_2,S}$. Together with the fact that $\Sigma_1 \subseteq \Sigma_2$ this proves that $(\mathcal{A}_1)_S \sqsubseteq (\mathcal{A}_2)_S$.

(If) Let $(\mathcal{A}_1)_S \sqsubseteq (\mathcal{A}_2)_S$. Then $\mathbf{C}_{\mathcal{A}_1} = \mathbf{C}_{(\mathcal{A}_1)_S} \subseteq \mathbf{C}_{(\mathcal{A}_2)_S} = \mathbf{C}_{\mathcal{A}_2}$ by Lemma 3.2.6 and Theorem 3.2.28. □

As a consequence we obtain that also state reduction is a monotonous operation with respect to containment ($\sqsubseteq$).

Lemma 3.2.31. *Let $\mathcal{A}_1$ and $\mathcal{A}_2$ be two automata such that $\mathcal{A}_1 \sqsubseteq \mathcal{A}_2$. Then*

$(\mathcal{A}_1)_S \sqsubseteq (\mathcal{A}_2)_S$.

Proof. By Lemma 3.2.6, $\mathbf{C}_{\mathcal{A}_1} \subseteq \mathbf{C}_{\mathcal{A}_2}$, and since the alphabet of $\mathcal{A}_1$ is contained in that of $\mathcal{A}_2$, Lemma 3.2.30 implies that $(\mathcal{A}_1)_S \sqsubseteq (\mathcal{A}_2)_S$. □

Another consequence of Lemma 3.2.30 is that once an automaton has been reduced with respect to its states, no further state reduction is possible.

Theorem 3.2.32. *Let $\mathcal{A}$ be an automaton. Then*

$\mathcal{A}_S$ is state reduced.

Proof. By definition, $\mathcal{A}$ and $\mathcal{A}_S$ have the same alphabet. By Theorem 3.2.28, $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}_S}$. Since $\mathcal{A}$ and $\mathcal{A}_S$ have the same alphabet we can now apply Lemma 3.2.30 twice and thus obtain $\mathcal{A} = (\mathcal{A}_S)_S$. Consequently, $\mathcal{A}_S$ is state reduced. □

A state-reduced version of an automaton has neither superfluous states nor superfluous transitions.

Theorem 3.2.33. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

$\mathcal{A}_S$ is the least automaton with alphabet Σ such that $\mathbf{C}_{\mathcal{A}_S} = \mathbf{C}_{\mathcal{A}}$.

Proof. By definition, $\mathcal{A}_S$ and $\mathcal{A}$ have the same alphabet. By Theorem 3.2.28, $\mathbf{C}_{\mathcal{A}_S} = \mathbf{C}_{\mathcal{A}}$. Now assume that $\mathcal{A}'$ is an automaton with alphabet Σ and such that $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}'}$. Then by applying Lemma 3.2.30 twice we have $\mathcal{A}_S = (\mathcal{A}')_S \sqsubseteq \mathcal{A}'$. □

Though an automaton $\mathcal{A}$ may still contain many automata that are state reduced, we now show that among these $\mathcal{A}_S$ is the largest (with respect to containment).

Lemma 3.2.34. *Let $\mathcal{A}$ be an automaton and let $\mathcal{A}' \sqsubseteq \mathcal{A}$. Then*

if $\mathcal{A}'$ is state reduced, then $\mathcal{A}' \sqsubseteq \mathcal{A}_S$.

Proof. If $\mathcal{A}' = (\mathcal{A}')_S$, then by Lemma 3.2.31, $\mathcal{A}' = (\mathcal{A}')_S \sqsubseteq \mathcal{A}_S$. □

The difference between $\mathcal{A}$ and $\mathcal{A}_S$ is thus minimal.

Theorem 3.2.35. *Let $\mathcal{A}$ be an automaton. Then*

$\mathcal{A}_S$ is the largest state-reduced automaton contained in $\mathcal{A}$.

Proof. Immediate from Theorem 3.2.32 and Lemma 3.2.34. □

A further reduction can only be achieved through the actions and transitions. We thus combine state reductions with action reductions and transition reductions.

Lemma 3.2.36. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ be an alphabet disjoint from Q. Then*

(1) $(\mathcal{A}^{\Theta}_{A})_S = (\mathcal{A}_S)^{\Theta}_{A}$ and

(2) $(\mathcal{A}^{\Theta}_{T})_S = (\mathcal{A}_S)^{\Theta}_{T} = \mathcal{A}_S$.

Proof. (1) Let $\mathcal{A}^{\Theta}_{A} = (Q, \Sigma^{\Theta}_{A}, \delta^{\Theta}_{A}, I)$. By Corollary 3.2.13, $Q_{\mathcal{A}^{\Theta}_{A},S} = Q_{\mathcal{A},S}$ and $\delta_{\mathcal{A}^{\Theta}_{A},T} = \delta_{\mathcal{A},T}$. Hence $(\mathcal{A}^{\Theta}_{A})_S = (Q_{\mathcal{A},S}, \Sigma^{\Theta}_{A}, \delta_{\mathcal{A},T}, I)$.

Next we consider $(\mathcal{A}_S)^{\Theta}_{A} = (Q', \Sigma', \delta', I')$. By Definitions 3.2.8 and 3.2.27, $I' = I$ and $Q' = Q_{\mathcal{A},S}$. Furthermore, $\Sigma' = \{a \in \Sigma \mid a \in \Theta \Rightarrow a \in \Sigma_{\mathcal{A}_S,A}\}$. Since $\mathbf{C}_{\mathcal{A}_S} = \mathbf{C}_{\mathcal{A}}$ by Theorem 3.2.28, we have $\Sigma' = \{a \in \Sigma \mid a \in \Theta \Rightarrow \Sigma_{\mathcal{A},A}\} = \Sigma^{\Theta}_{A}$. Finally, $\delta' = \delta_{\mathcal{A},T} \cap (Q \times \Sigma^{\Theta}_{A} \times Q) = \delta_{\mathcal{A},T}$. Hence $(\mathcal{A}^{\Theta}_{A})_S = (\mathcal{A}_S)^{\Theta}_{A}$.

(2) Both $\mathcal{A}$ and $\mathcal{A}^{\Theta}_{T}$ have alphabet Σ. By Theorem 3.2.12, $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{\mathcal{A}^{\Theta}_{T}}$ and thus applying Lemma 3.2.30 twice yields $\mathcal{A}_S = (\mathcal{A}^{\Theta}_{T})_S$. Also $\mathcal{A}$ and $(\mathcal{A}_S)^{\Theta}_{T}$ have the same alphabet. Since $\mathbf{C}_{\mathcal{A}} = \mathbf{C}_{(\mathcal{A}_S)^{\Theta}_{T}}$ by Theorems 3.2.12 and 3.2.28, applying Lemma 3.2.30 twice yields $\mathcal{A}_S = ((\mathcal{A}_S)^{\Theta}_{T})_S$. Thus $\mathcal{A}_S = ((\mathcal{A}_S)^{\Theta}_{T})_S \sqsubseteq (\mathcal{A}_S)^{\Theta}_{T} \sqsubseteq \mathcal{A}_S$ and hence it must be the case that $\mathcal{A}_S = (\mathcal{A}_S)^{\Theta}_{T}$. □

Transition reduction in the context of state reduction thus has no effect. All transitions that are not useful will disappear by the state reduction.

Theorem 3.2.37. *Let $\mathcal{A}$ be a state-reduced automaton and let Θ be an alphabet disjoint from its set of states. Then*

$\mathcal{A}$ is Θ-transition reduced.

Proof. Since $\mathcal{A}$ is state reduced we have $\mathcal{A} = \mathcal{A}_S$. Then Lemma 3.2.36(2) implies $\mathcal{A}_T^{\Theta} = (\mathcal{A}_S)_T^{\Theta} = \mathcal{A}_S = \mathcal{A}$ and hence $\mathcal{A}$ is Θ-transition reduced. □

Example 3.2.38. (Example 3.2.29 continued) By definition every transition of $\mathcal{A}_S$ is useful. Hence $\mathcal{A}_S$ trivially is Θ-transition reduced for any set of actions Θ. □

Lemmata 3.2.16, 3.2.23, and 3.2.36 now imply that for every automaton $\mathcal{A}$, any finite succession of action reductions and state reductions (at least one) has the same effect as one state reduction and one action reduction (relative to some alphabet Θ) and yields an automaton $(\mathcal{A}_A^{\Theta})_S = (\mathcal{A}_S)_A^{\Theta}$.

Example 3.2.39. (Examples 3.2.24 and 3.2.29 continued) Consider the state-reduced version $\mathcal{A}_S$ of $\mathcal{A}$. Since $\Sigma_{\mathcal{A}_S,A} = \{a\}$, the $\{b\}$-action-reduced version of $\mathcal{A}_S$ is $(\mathcal{A}_S)_A^{\{b\}} = (\{p\}, \{a\}, \{(p, a, p)\}, \{p\})$.

Now consider the $\{b\}$-action-reduced version $\mathcal{A}_A^{\{b\}}$ of $\mathcal{A}$. We have seen that its only useful transition is (p, a, p), which implies that q is not reachable and thus $(\mathcal{A}_A^{\{b\}})_S = (\mathcal{A}_S)_A^{\{b\}}$. □

Theorem 3.2.40. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

> $(\mathcal{A}_A^{\Theta})_S$ *is the largest automaton contained in $\mathcal{A}$ that is both state reduced and Θ-action reduced.*

Proof. By Lemma 3.2.36(1) and Theorems 3.2.17(1) and 3.2.32, $(\mathcal{A}_A^{\Theta})_S = (\mathcal{A}_S)_A^{\Theta}$ is Θ-action reduced and state reduced.

Now let $\mathcal{A}_1 \sqsubseteq \mathcal{A}$. Then by Lemma 3.2.20(1), $(\mathcal{A}_1)_A^{\Theta} \sqsubseteq \mathcal{A}_A^{\Theta}$, and by Lemma 3.2.31, $((\mathcal{A}_1)_A^{\Theta})_S \sqsubseteq (\mathcal{A}_A^{\Theta})_S$. If $\mathcal{A}_1$ is Θ-action reduced, then by definition $(\mathcal{A}_1)_A^{\Theta} = \mathcal{A}_1$. If — in addition — it is state reduced, then $\mathcal{A}_1 = (\mathcal{A}_1)_S = ((\mathcal{A}_1)_A^{\Theta})_S \sqsubseteq (\mathcal{A}_A^{\Theta})_S$. □

Summarizing, an automaton may have superfluous states, actions, or transitions, which can be omitted without affecting its operational potential (as represented by its set of finite computations). We have considered reductions with respect to each of these elements separately, and in combination. It has been shown that transition reduction is implied by state reduction, whereas the other combinations of reductions are stronger than each reduction separately. Consequently, once an automaton has been reduced with respect to states and actions, then it cannot be reduced any further without losing computations.

In correspondence to the notions of Θ-records and Θ-behavior of an automaton, both action reduction and transition reduction have been investigated relative to an alphabet. In case no special actions are distinguished and every element of the alphabet of an automaton is considered, then we drop in the sequel — as before — the reference to the alphabet if this cannot lead to confusion.

The above implies that for an automaton $\mathcal{A} = (Q, \Sigma, \delta, I)$ we now have $\mathcal{A}_A = \mathcal{A}_A^{\Sigma}$ as its *action-reduced version*, and we have $\mathcal{A}_T = \mathcal{A}_T^{\Sigma}$ as its *transition-reduced version*. Furthermore, we will refer to $\mathcal{A}_R = (\mathcal{A}_A)_S = (\mathcal{A}_S)_A$ as the *reduced version of* $\mathcal{A}$. Note that the definitions of $\mathcal{A}_S$ and $(\mathcal{A}_S)_A^{\Sigma}$, together with Theorem 3.2.28 and Corollary 3.2.13, imply that the automaton $\mathcal{A}_R$ is specified as $\mathcal{A}_R = (Q_S, \Sigma_A, \delta_T, I)$. Hence $\mathcal{A}_R$ has no superfluous elements at all.

Theorems 3.2.37 and 3.2.40 imply that $\mathcal{A}_R$ is the largest automaton contained in $\mathcal{A}$ that is state reduced, action reduced, and transition reduced, and has the same computations as $\mathcal{A}$. We now show that $\mathcal{A}_R$ is the only such automaton.

Theorem 3.2.41. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

> *$\mathcal{A}_R$ is the unique automaton contained in $\mathcal{A}$ that is state reduced, action reduced, and transition reduced, and such that $\mathbf{C}_{\mathcal{A}_R} = \mathbf{C}_{\mathcal{A}}$.*

Proof. Let $\mathcal{A}' = (Q', \Sigma', \delta', I')$ be an action-reduced, transition-reduced, and state-reduced automaton such that $\mathcal{A}' \sqsubseteq \mathcal{A}$. From Theorems 3.2.37 and 3.2.40 we know that $\mathcal{A}' \sqsubseteq \mathcal{A}_R$.

Now assume that $\mathbf{C}_{\mathcal{A}'} = \mathbf{C}_{\mathcal{A}}$. Then $Q_{\mathcal{A}',S} = Q_{\mathcal{A},S}$, $\Sigma_{\mathcal{A}',A} = \Sigma_{\mathcal{A},A}$, $\delta_{\mathcal{A}',T} = \delta_{\mathcal{A},T}$, and $I' = I$. Since $Q_{\mathcal{A}',S} \subseteq Q'$, $\Sigma_{\mathcal{A}',A} \subseteq \Sigma'$, and $\delta_{\mathcal{A}',T} \subseteq \delta'$, we have $\mathcal{A}_R = (Q_{\mathcal{A},S}, \Sigma_{\mathcal{A},A}, \delta_{\mathcal{A},T}, I) \sqsubseteq \mathcal{A}'$. We thus conclude that $\mathcal{A}' = \mathcal{A}_R$. □

3.2.2 Enabling

For an arbitrary automaton and a given action, it is in general not the case that this action can always (i.e. at any give state) be executed by the automaton. For certain types of systems (such as, e.g., reactive systems) it may however be crucial that specific actions (in reaction to stimuli from the environment) can always be executed. Thus when such a system is modeled as an automaton, the transition relation should contain a transition for each of these actions at every (reachable) state.

In this subsection, we define *enabledness* of actions as a local (state dependent) property of the transition relation and then lift it to the level of the automaton. This contrasts with our approach in the previous subsection in which the role of states, actions, and transitions was assessed on basis of their occurrence in computations.

Definition 3.2.42. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

(1) an action $a \in \Sigma$ is enabled (in $\mathcal{A}$) at *a state $q \in Q$, denoted by $a \ en_{\mathcal{A}} \ q$, if $(q, a, q') \in \delta$ for some $q' \in Q$.*

Let Θ be an alphabet disjoint from Q. Then

*(2) $\mathcal{A}$ is Θ-*enabling *if for all $a \in \Theta$ and for all $q \in Q$, $a \in \Sigma \Rightarrow a \ en_{\mathcal{A}} \ q$.* □

Note that, as in previous definitions, also the property of enabling is defined with respect to a separately specified arbitrary set of actions Θ. Similar to those previous notions, whether or not an automaton is Θ-enabling is solely determined by those elements of Θ that are actions of $\mathcal{A}$. To be precise, $\mathcal{A}$ is always $\varnothing$-enabling. Furthermore, $\mathcal{A}$ is Θ-enabling if and only if it is $\Theta \cap \Sigma$-enabling, where Σ is the set of actions of $\mathcal{A}$.

Example 3.2.43. (Example 3.2.10 continued) It is easy to see that $\mathcal{A}$ is $\{a\}$-enabling but not $\{b\}$-enabling. Hence $\mathcal{A}$ is neither $\{a, b\}$-enabling. However, $\mathcal{A}$ is $\{d\}$-enabling, for all $d \notin \Sigma$, and thus also $\{a, d\}$-enabling. □

The deletion of states and/or transitions from an automaton does not affect its enabling of given actions, provided relevant transitions are preserved.

Lemma 3.2.44. *Let $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ be two automata and let Θ_1, Θ_2 be two alphabets disjoint from $Q_1 \cup Q_2$. Let $Q_2 \subseteq Q_1$, $\Theta_2 \cap \Sigma_2 \subseteq \Theta_1 \cap \Sigma_1$, and $\delta_2 \supseteq \delta_1 \cap (Q_2 \times (\Theta_2 \cap \Sigma_2) \times Q_1)$. Then*

if $\mathcal{A}_1$ is Θ_1-enabling, then $\mathcal{A}_2$ is Θ_2-enabling.

Proof. Let $\mathcal{A}_1$ be Θ_1-enabling. Now let $a \in \Theta_2$ and let $q \in Q_2$. If $a \in \Sigma_2$, then $a \in \Theta_1 \cap \Sigma_1$. Since $q \in Q_1$, it then follows that there exists a $q' \in Q$ such that $(q, a, q') \in \delta_1$. Thus $(q, a, q') \in \delta_2$ and we have $a \ \mathrm{en}_{\mathcal{A}_2} \ q$. □

Corollary 3.2.45. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ_1, Θ_2 be two alphabets disjoint from Q and such that $(\Theta_2 \cap \Sigma) \subseteq \Theta_1$. Then*

if $\mathcal{A}$ is Θ_1-enabling, then $\mathcal{A}$ is Θ_2-enabling. □

From the computational and the behavioral point of view, enabledness of actions is especially relevant at the reachable states of an automaton. Recall that for a given automaton $\mathcal{A} = (Q, \Sigma, \delta, I)$ we denote by Q_S its set of reachable states. We have defined $\mathcal{A}_S = (Q_S, \Sigma, \delta_T, I)$ as the state-reduced version of $\mathcal{A}$, where $\delta_T = \delta \cap (Q_S \times \Sigma \times Q_S) = \delta \cap (Q_S \times \Sigma \times Q)$ consists of the useful transitions of $\mathcal{A}$. Thus, as another immediate consequence of Lemma 3.2.44, we have that the state-reduced version of $\mathcal{A}$ is Θ-enabling whenever $\mathcal{A}$ is.

Theorem 3.2.46. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

if $\mathcal{A}$ is Θ-enabling, then $\mathcal{A}_S$ is Θ-enabling. □

The converse clearly does not hold, since actions which are enabled at reachable states of an automaton $\mathcal{A}$ are not necessarily enabled at every non-reachable state of $\mathcal{A}$. The fact that the state-reduced version of $\mathcal{A}$ may have less states than $\mathcal{A}$ thus causes a lack of information concerning outgoing transitions of non-reachable states.

The situation is different when $\mathcal{A}$ is reduced by removing only its non-useful transitions with a label from an alphabet Θ_1, but no states whatsoever, as is done in order to obtain its Θ_1-transition-reduced version $\mathcal{A}_T^{\Theta_1}$. In that case the enabledness of actions in $\mathcal{A}_T^{\Theta_1}$ *can* thus be used to decide their enabledness in $\mathcal{A}$. In fact, since $\mathcal{A}_T^{\Theta_1}$ may have less transitions than $\mathcal{A}$, but it may never have less states than $\mathcal{A}$, Lemma 3.2.44 immediately yields the following result.

Lemma 3.2.47. *Let $\mathcal{A}$ be an automaton and let Θ, Θ_1 be two alphabets disjoint from its set of states. Then*

if $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling, then $\mathcal{A}$ is Θ-enabling. □

Furthermore, all transitions of $\mathcal{A}_T^{\Theta_1}$ with a label from Θ_1 are by definition useful in $\mathcal{A}_T^{\Theta_1}$. Hence if there exists an $a \in \Sigma \cap \Theta_1$ which is enabled at every state of $\mathcal{A}_T^{\Theta_1}$, then all states of $\mathcal{A}_T^{\Theta_1}$ are reachable.

Lemma 3.2.48. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ, Θ_1 be two alphabets disjoint from Q and such that $\Theta \cap \Theta_1 \cap \Sigma \neq \varnothing$. Then*

if $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling, then $Q = Q_{\mathcal{A},S}$.

Proof. Let $\mathcal{A}_T^{\Theta_1} = (Q, \Sigma, \delta_{\mathcal{A},T}^{\Theta_1}, I)$ be Θ-enabling. Since $Q_{\mathcal{A},S} \subseteq Q$ always holds, we only have to prove the converse inclusion $Q \subseteq Q_{\mathcal{A},S}$. Let $q \in Q$. Consider $a \in \Theta \cap \Theta_1 \cap \Sigma$. Then the assumption that $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling implies there exists a $q' \in Q$ such that $(q, a, q') \in \delta_{\mathcal{A},T}^{\Theta_1}$. Since $a \in \Theta_1$, the definition of $\delta_{\mathcal{A},T}^{\Theta_1}$ implies that $(q, a, q') \in \delta_{\mathcal{A},T}$. Consequently, $q \in Q_{\mathcal{A},S}$. □

We have thus established that $\mathcal{A}$ is Θ-enabling whenever $\mathcal{A}_T^{\Theta_1}$ is. Conversely, $\mathcal{A}_T^{\Theta_1}$ obviously is Θ-enabling whenever $\mathcal{A}$ is and no action from Θ is included in both Θ_1 and the set of actions of $\mathcal{A}$. If the latter part of this condition is not met, then the Θ-enabling of $\mathcal{A}$ nevertheless does imply that $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling if $\mathcal{A}$ is Θ_1-transition reduced.

Theorem 3.2.49. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ, Θ_1 be two alphabets disjoint from Q. Then*

> $\mathcal{A}_T^{\Theta_1}$ *is Θ-enabling if and only if $\mathcal{A}$ is Θ-enabling and $\mathcal{A} = \mathcal{A}_S = \mathcal{A}_T^{\Theta_1}$ whenever $\Theta \cap \Theta_1 \cap \Sigma \neq \varnothing$.*

Proof. (Only if) By Lemma 3.2.47, $\mathcal{A}$ is Θ-enabling if $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling. Assume that $\Theta \cap \Theta_1 \cap \Sigma \neq \varnothing$. Then from Lemma 3.2.48 we know that the fact that $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling implies that $Q = Q_{\mathcal{A},S}$. Consequently, $\delta = \delta \cap (Q_{\mathcal{A},S} \times \Sigma \times Q_{\mathcal{A},S})$ and so $\delta = \delta_{\mathcal{A},T}$. Thus we have $\mathcal{A} = \mathcal{A}_S$. Finally, by definition $\delta_{\mathcal{A},T} \subseteq \delta_{\mathcal{A},T}^{\Theta_1} \subseteq \delta$. Hence $\delta_{\mathcal{A},T} = \delta_{\mathcal{A},T}^{\Theta_1} = \delta$, which implies that $\mathcal{A} = \mathcal{A}_T^{\Theta_1}$.

(If) If $\mathcal{A}$ is Θ-enabling and $\mathcal{A} = \mathcal{A}_T^{\Theta_1}$, then it trivially follows that $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling. Thus we assume that $\mathcal{A}$ is Θ-enabling and that $\Theta \cap \Theta_1 \cap \Sigma = \varnothing$. Let $\mathcal{A}_T^{\Theta_1} = (Q, \Sigma, \delta_{\mathcal{A},T}^{\Theta_1}, I)$. By definition $\delta_{\mathcal{A},T}^{\Theta_1} \supseteq \delta \setminus (Q \times \Theta_1 \times Q) = \delta \setminus (Q \times (\Theta_1 \cap \Sigma) \times Q)$. Since $\Theta \cap (\Theta_1 \cap \Sigma) = \varnothing$, it follows that $\delta_{\mathcal{A},T}^{\Theta_1} \supseteq \delta \cap (Q \times \Theta \times Q) = \delta \cap (Q \times (\Theta \cap \Sigma) \times Q)$. Consequently, we can apply Lemma 3.2.44 and conclude that $\mathcal{A}_T^{\Theta_1}$ is Θ-enabling. □

Corollary 3.2.50. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

> $\mathcal{A}_T^{\Theta}$ *is Θ-enabling if and only if $\mathcal{A}$ is Θ-enabling and $\mathcal{A} = \mathcal{A}_T^{\Theta}$.* □

Let us now focus on the interplay between active actions and enabled actions. Recall that whenever an action is active, then there exists at least one reachable state where it is enabled. Given an automaton we can thus delete the non-active actions from its alphabet and the transitions these actions are involved in from its transition relation, without effecting the enabling of this automaton.

Lemma 3.2.51. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ, Θ_1 be two alphabets disjoint from Q. Then*

> *if $\mathcal{A}$ is Θ-enabling, then $\mathcal{A}_A^{\Theta_1}$ is Θ-enabling.*

Proof. Let $\mathcal{A}$ be Θ-enabling. By definition $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_{\mathcal{A},A}^{\Theta_1}, \delta_{\mathcal{A},A}^{\Theta_1}, I)$, with $\Sigma_{\mathcal{A},A}^{\Theta_1} \subseteq \Sigma$ and $\delta_{\mathcal{A},A}^{\Theta_1} = \delta \cap (Q \times \Sigma_{\mathcal{A},A}^{\Theta_1} \times Q)$. Thus $\Theta \cap \Sigma_{\mathcal{A},A}^{\Theta_1} \subseteq \Theta \cap \Sigma$. Furthermore, $\delta_{\mathcal{A},A}^{\Theta_1} \supseteq \delta \cap (Q \times (\Theta \cap \Sigma_{\mathcal{A},A}^{\Theta_1}) \times Q)$. Consequently we can apply Lemma 3.2.44 and conclude that $\mathcal{A}_A^{\Theta_1}$ is Θ-enabling. □

The converse in general does not hold, even though $\mathcal{A}$ contains all transitions of $\mathcal{A}_A^{\Theta_1}$. The reason is that $\mathcal{A}$ may contain more actions than $\mathcal{A}_A^{\Theta_1}$ does. Thus whenever $\mathcal{A}_A^{\Theta_1}$ is Θ-enabling also $\mathcal{A}$ will be Θ-enabling, provided Θ contains no action of Θ_1 that is a non-active action of $\mathcal{A}$. Hence we require all actions from $\Theta_1 \cap \Theta$ that appear also in the set of actions of $\mathcal{A}$, to be active.

Lemma 3.2.52. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ, Θ_1 be two alphabets disjoint from Q and such that $\Theta \cap \Theta_1 \cap \Sigma \subseteq \Sigma_{\mathcal{A},A}$. Then*

if $\mathcal{A}_A^{\Theta_1}$ is Θ-enabling, then $\mathcal{A}$ is Θ-enabling.

Proof. Let $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_{\mathcal{A},A}^{\Theta_1}, \delta_{\mathcal{A},A}^{\Theta_1}, I)$ be Θ-enabling. By definition $\delta_{\mathcal{A},A}^{\Theta_1} \subseteq \delta$ and hence — once we have established that $\Theta \cap \Sigma \subseteq \Theta \cap \Sigma_{\mathcal{A},A}^{\Theta_1}$ — we can apply Lemma 3.2.44 and conclude that $\mathcal{A}$ is Θ-enabling.

Assume that $\Theta \cap \Theta_1 \cap \Sigma \subseteq \Sigma_{\mathcal{A},A}$. Now let $a \in \Theta \cap \Sigma$ and recall that $\Sigma_{\mathcal{A},A}^{\Theta_1} = (\Sigma \setminus \Theta_1) \cup (\Sigma_{\mathcal{A},A} \cap \Theta_1)$.
If $a \notin \Theta_1$, then $a \in (\Sigma \setminus \Theta_1) \subseteq \Sigma_{\mathcal{A},A}^{\Theta_1}$.
If $a \in \Theta_1$, then $a \in \Sigma_{\mathcal{A},A}$ by our assumption and thus $a \in \Sigma_{\mathcal{A},A}^{\Theta_1}$.

Hence in both cases $a \in \Theta \cap \Sigma_{\mathcal{A},A}^{\Theta_1}$ and we are done. □

From Lemma 3.2.2(3) we know that an action $a \in \Sigma$ of an automaton $\mathcal{A} = (Q, \Sigma, \delta, I)$ is active if and only if there exists a useful transition $(q, a, q') \in \delta$. This means that $\Sigma_A = \varnothing$ whenever $Q_S = \varnothing$. If $Q_S \neq \varnothing$, however, and $\mathcal{A}$ is Θ-enabling, for some set of actions Θ, then every action in $\Theta \cap \Sigma$ is active in $\mathcal{A}$. This is due to the fact that a nonempty set of reachable states implies that all actions $\Theta \cap \Sigma$ are enabled in every initial state of $\mathcal{A}$, all of whose outgoing transitions are by definition useful.

Lemma 3.2.53. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton such that $Q_S \neq \varnothing$ and let Θ be an alphabet disjoint from Q. Then*

if $\mathcal{A}$ is Θ-enabling, then $\Theta \cap \Sigma \subseteq \Sigma_A$ and $\mathcal{A} = \mathcal{A}_A^{\Theta}$.

Proof. Let $\mathcal{A}$ be Θ-enabling and let $a \in \Theta \cap \Sigma$. Since $I = \varnothing$ implies that $Q_S = \varnothing$, it must be the case that $I \neq \varnothing$. Now let $q \in I$. Then there exists a $q' \in Q$ such that $(q, a, q') \in \delta$. Since $q \in I \subseteq Q_S$ is reachable in $\mathcal{A}$ this implies that a is active in $\mathcal{A}$, and thus $a \in \Sigma_A$. Hence $\Theta \cap \Sigma \subseteq \Sigma_A$.

Now let $\mathcal{A}_A^{\Theta} = (Q, \Sigma_{\mathcal{A},A}^{\Theta}, \delta_{\mathcal{A},A}^{\Theta}, I)$. Then $\Sigma_{\mathcal{A},A}^{\Theta} = (\Sigma \setminus \Theta) \cup (\Sigma_A \cap \Theta) = (\Sigma \setminus \Theta) \cup (\Sigma \cap \Theta) = \Sigma$ because $\Theta \cap \Sigma = \Theta \cap \Sigma_A$ by the above and $\Sigma_A \subseteq \Sigma$. By definition $\delta_{\mathcal{A},A}^{\Theta} = \delta \cap (Q \times \Sigma_{\mathcal{A},A}^{\Theta} \times Q)$. Hence $\delta_{\mathcal{A},A}^{\Theta} = \delta \cap (Q \times \Sigma \times Q) = \delta$. Consequently, $\mathcal{A}_A^{\Theta} = \mathcal{A}$. □

This lemma, together with Lemmata 3.2.51 and 3.2.52, directly implies the following theorem.

Theorem 3.2.54. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton such that $Q_S \neq \varnothing$ and let Θ, Θ_1 be two alphabets disjoint from Q. Then*

$\mathcal{A}$ *is Θ-enabling if and only if $\mathcal{A}_A^{\Theta_1}$ is Θ-enabling and $\Theta \cap \Theta_1 \cap \Sigma \subseteq \Sigma_{\mathcal{A},A}$.* □

Corollary 3.2.55. *Let $\mathcal{A}$ be an automaton and let Θ be an alphabet disjoint from its set of states. Then*

$\mathcal{A}$ *is Θ-enabling if and only if $\mathcal{A}_A^{\Theta}$ is Θ-enabling and $\mathcal{A} = \mathcal{A}_A^{\Theta}$.* □

In this subsection we have thus presented various conditions under which enabling is preserved from one (reduced) automaton to another. We have considered separately the state-reduced, action-reduced, and transition-reduced versions of automata. We now conclude with a result that incorporates also the reduced version of an automaton. It is obtained as a direct consequence of combining Theorem 3.2.46 with Corollary 3.2.55.

Theorem 3.2.56. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton. Then*

if $\mathcal{A}$ is Σ-enabling, then $\mathcal{A}_S = \mathcal{A}_R$. □

3.2.3 Determinism

For an arbitrary automaton and a given action, it is in general not the case that for each of its states there is at most one possible way to execute this action. For certain types of systems (such as, e.g., transformational systems) it may however be crucial that the outcome of the execution of one of its actions is uniquely determined by the state the automaton is in. Thus when such a system is modeled as an automaton, the transition relation should contain at most one transition for each combination of such an action and a state of the automaton.

In a *deterministic* automaton, there is no choice as to what state the automaton ends up in after the execution of a sequence of actions. As was the case for enabling, the definition of determinism of an automaton is based on a local (state dependent) property of the transition relation.

Definition 3.2.57. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ be an alphabet disjoint from Q. Then*

$\mathcal{A}$ *is* Θ-deterministic *if I contains at most one element and for all $a \in \Theta$ and for all $q \in Q$, $\{q' \in Q \mid (q, a, q') \in \delta\}$ contains at most one element.* □

Note the duality between enabling and determinism: given that a is an action of the automaton, then this automaton is $\{a\}$-enabling if each of its states has *at least* one outgoing a-transition, while it is $\{a\}$-deterministic if each of its states has *at most* one outgoing a-transition.

As in previous definitions, also the property of determinism is defined with respect to a separately specified arbitrary set of actions Θ. Similar to those previous notions, whether or not an automaton is Θ-deterministic is solely determined by those elements of Θ that are actions of $\mathcal{A}$. More precisely, if we assume that $\mathcal{A}$ contains at most one initial state, then $\mathcal{A}$ is always $\varnothing$-deterministic and — moreover — $\mathcal{A}$ is Θ-deterministic if and only if it is $\Theta \cap \Sigma$-deterministic, where Σ is the set of actions of $\mathcal{A}$.

Example 3.2.58. (Example 3.2.10 continued) Let $\mathcal{A}'$ be the automaton obtained from automaton $\mathcal{A}$ of Example 3.2.10 — depicted in Figure 3.3(a) — by replacing transition (q, a, q) with (q, b, q). Then $\mathcal{A}'$ is $\{a\}$-deterministic but not $\{b\}$-deterministic. Hence $\mathcal{A}'$ is neither $\{a, b\}$-deterministic. However, $\mathcal{A}'$ is $\{d\}$-deterministic, for all $d \notin \Sigma$, and thus $\{a, d\}$-deterministic as well. □

The deletion of states and/or transitions from an automaton does not affect its determinism of given actions.

Lemma 3.2.59. *Let $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ be two automata and let Θ_1, Θ_2 be two alphabets disjoint from $Q_1 \cup Q_2$. Let $\Theta_2 \cap \Sigma_2 \subseteq \Theta_1$, let $\delta_2 \cap (Q_2 \times \Theta_2 \times Q_2) \subseteq \delta_1$, and let I_2 contain at most one element. Then*

if $\mathcal{A}_1$ is Θ_1-deterministic, then $\mathcal{A}_2$ is Θ_2-deterministic.

Proof. Let $\mathcal{A}_1$ be Θ_1-deterministic. Now let $a \in \Theta_2$ and let $p \in Q_2$. Suppose that there exist $q, q' \in Q_2$ such that both $(p, a, q) \in \delta_2$ and $(p, a, q') \in \delta_2$. This implies that $a \in \Theta_2 \cap \Sigma_2$ and that both $(p, a, q) \in \delta_1$ and $(p, a, q') \in \delta_1$. Since $\Theta_2 \cap \Sigma_2 \subseteq \Theta_1$ and $\mathcal{A}_1$ is Θ_1-deterministic it follows that it must be the case that $q = q'$. Together with the fact that I_2 contains at most one element this implies that $\mathcal{A}_2$ is Θ_2-deterministic. □

This lemma has several immediate consequences.

Corollary 3.2.60. *Let $\mathcal{A} = (Q, \Sigma, \delta, I)$ be an automaton and let Θ_1, Θ_2 be two alphabets disjoint from Q and such that $(\Theta_2 \cap \Sigma) \subseteq \Theta_1$. Then*

if $\mathcal{A}$ is Θ_1-deterministic, then $\mathcal{A}$ is Θ_2-deterministic. □

Corollary 3.2.61. *Let $\mathcal{A}_1 = (Q_1, \Sigma_1, \delta_1, I_1)$ and $\mathcal{A}_2 = (Q_2, \Sigma_2, \delta_2, I_2)$ be two automata such that $\mathcal{A}_2 \sqsubseteq \mathcal{A}_1$ and let Θ_1, Θ_2 be two alphabets disjoint from $Q_1 \cup Q_2$ and such that $(\Theta_2 \cap \Sigma_2) \subseteq \Theta_1$. Then* *if* $\mathcal{A}_1$ *is* Θ_1*-deterministic, then* $\mathcal{A}_2$ *is* Θ_2*-deterministic.* □

Corollary 3.2.62. *Let* $\mathcal{A} = (Q, \Sigma, \delta, I)$ *and* $\mathcal{A}' = (Q, \Sigma', \delta, I)$ *be two automata such that* $\Sigma \subseteq \Sigma'$ *and let* Θ *be an alphabet disjoint from* Q*. Then*

if $\mathcal{A}$ *is* Θ*-deterministic, then* $\mathcal{A}'$ *is* Θ*-deterministic.* □

From the computational and the behavioral viewpoint also determinism is most relevant at the reachable states of an automaton. We thus finish this subsection with an overview of the influence that the determinism of one type of reduced automaton has on the determinism of another type of reduced automaton.

Theorem 3.2.63. *Let* $\mathcal{A} = (Q, \Sigma, \delta, I)$ *be an automaton and let* Θ, Θ_1 *be two alphabets disjoint from* Q*. Then*

(1) if $\mathcal{A}$ *is* Θ*-deterministic, then so is* $\mathcal{A}_A^{\Theta_1}$*,*

(2) if $\mathcal{A}_A^{\Theta_1}$ *is* Θ*-deterministic, then so is* $\mathcal{A}_T^{\Theta_1}$*, and*

(3) if $\mathcal{A}_T^{\Theta_1}$ *is* Θ*-deterministic, then so is* $\mathcal{A}_S$*.*

Proof. (1) This follows directly from Corollary 3.2.61 since $\mathcal{A}_A^{\Theta_1}$ is a reduced version of $\mathcal{A}$ and thus $\mathcal{A}_A^{\Theta_1} \sqsubseteq \mathcal{A}$.

(2) Let $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_{\mathcal{A},A}^{\Theta_1}, \delta_{\mathcal{A},A}^{\Theta_1}, I)$ be Θ-deterministic. As by definition $\Sigma_{\mathcal{A},A}^{\Theta_1} \subseteq \Sigma$, Corollary 3.2.62 implies that also the automaton $\mathcal{A}' = (Q, \Sigma, \delta_{\mathcal{A},A}^{\Theta_1}, I)$ is Θ-deterministic. Now consider $\mathcal{A}_T^{\Theta_1} = (Q, \Sigma, \delta_{\mathcal{A},T}^{\Theta_1}, I)$. By definition $\delta_{\mathcal{A},T}^{\Theta_1} \subseteq \delta_{\mathcal{A},A}^{\Theta_1}$ and thus $\mathcal{A}_T^{\Theta_1} \sqsubseteq \mathcal{A}'$. Corollary 3.2.61 subsequently implies that also $\mathcal{A}_T^{\Theta_1}$ is Θ-deterministic.

(3) From Lemma 3.2.36(2) we know that $\mathcal{A}_S = (\mathcal{A}_T^{\Theta_1})_S$. Analogous to (1) the result now follows from the fact that $(\mathcal{A}_T^{\Theta_1})_S \sqsubseteq \mathcal{A}_T^{\Theta_1}$. □

In certain cases Θ-determinism is thus preserved from one automaton to another, for a set Θ of actions. The proof of this theorem however is heavily based on the containment of one automaton in another. In case the reverse of such a containment does not hold, often some characteristics crucial for preserving Θ-determinism from one automaton to another, are lacking. When formulating the reverses of the statements of this theorem, we thus settle for a demonstration of the preservation of determinism from one automaton to another for only a subset of Θ.

Theorem 3.2.64. *Let* $\mathcal{A} = (Q, \Sigma, \delta, I)$ *be an automaton and let* Θ, Θ_1 *be two alphabets disjoint from* Q*. Then*

(1) if $\mathcal{A}_S$ *is* Θ*-deterministic, then* $\mathcal{A}_T^{\Theta_1}$ *is* $(\Theta \cap \Theta_1)$*-deterministic,*

(2) if $\mathcal{A}_T^{\Theta_1}$ *is* Θ*-deterministic, then* $\mathcal{A}_A^{\Theta_1}$ *is* $(\Theta \setminus \Theta_1)$*-deterministic, and*

(3) if $\mathcal{A}_A^{\Theta_1}$ *is* Θ*-deterministic, then* $\mathcal{A}$ *is* $(\Theta \setminus (\Theta_1 \setminus \Sigma_{\mathcal{A},A}))$*-deterministic.*

Proof. (1) Let $\mathcal{A}_S = (Q_{\mathcal{A},S}, \Sigma_{\mathcal{A},A}, \delta_{\mathcal{A},T}, I)$ be Θ-deterministic. Now consider $\mathcal{A}_T^{\Theta_1} = (Q, \Sigma, \delta_{\mathcal{A},T}^{\Theta_1}, I)$. Since $(\Theta \cap \Theta_1) \cap \Sigma \subseteq \Theta$ and $\delta_{\mathcal{A},T}^{\Theta_1} \cap (Q \times (\Theta \cap \Theta_1) \times Q) \subseteq \delta_{\mathcal{A},T}$ it follows from Lemma 3.2.59 that $\mathcal{A}_T^{\Theta_1}$ is $(\Theta \cap \Theta_1)$-deterministic.

(2) Let $\mathcal{A}_T^{\Theta_1} = (Q, \Sigma, \delta_{\mathcal{A},T}^{\Theta_1}, I)$ be Θ-deterministic. Now consider $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_{\mathcal{A},A}^{\Theta_1}, \delta_{\mathcal{A},A}^{\Theta_1}, I)$. Since $(\Theta \setminus \Theta_1) \cap \Sigma_{\mathcal{A},A}^{\Theta_1} \subseteq \Theta$ and $\delta_{\mathcal{A},A}^{\Theta_1} \cap (Q \times (\Theta \setminus \Theta_1) \times Q) \subseteq \delta \cap (Q \times (\Sigma \setminus \Theta_1) \times Q) \subseteq \delta_{\mathcal{A},T}^{\Theta_1}$ it follows from Lemma 3.2.59 that $\mathcal{A}_A^{\Theta_1}$ is $(\Theta \setminus \Theta_1)$-deterministic.

(3) Let $\mathcal{A}_A^{\Theta_1} = (Q, \Sigma_{\mathcal{A},A}^{\Theta_1}, \delta_{\mathcal{A},A}^{\Theta_1}, I)$ be Θ-deterministic. Clearly $(\Theta \setminus (\Theta_1 \setminus \Sigma_{\mathcal{A},A})) \cap \Sigma \subseteq \Theta$. Moreover, since $\Theta \setminus (\Theta_1 \setminus \Sigma_{\mathcal{A},A})) = (\Theta \setminus \Theta_1) \cup (\Theta \cap (\Sigma_{\mathcal{A},A} \cap \Theta_1))$ it follows that $\delta \cap (Q \times (\Theta \setminus (\Theta_1 \setminus \Sigma_{\mathcal{A},A})) \times Q) \subseteq (\delta \cap (Q \times (\Sigma \setminus \Theta_1) \times Q)) \cup (\delta \cap (Q \times (\Sigma_{\mathcal{A},A} \cap \Theta_1) \times Q)) = \delta_{\mathcal{A},A}^{\Theta_1}$. Hence by Lemma 3.2.59 it follows that $\mathcal{A}$ is $(\Theta \setminus (\Theta_1 \setminus \Sigma_{\mathcal{A},A}))$-deterministic. □