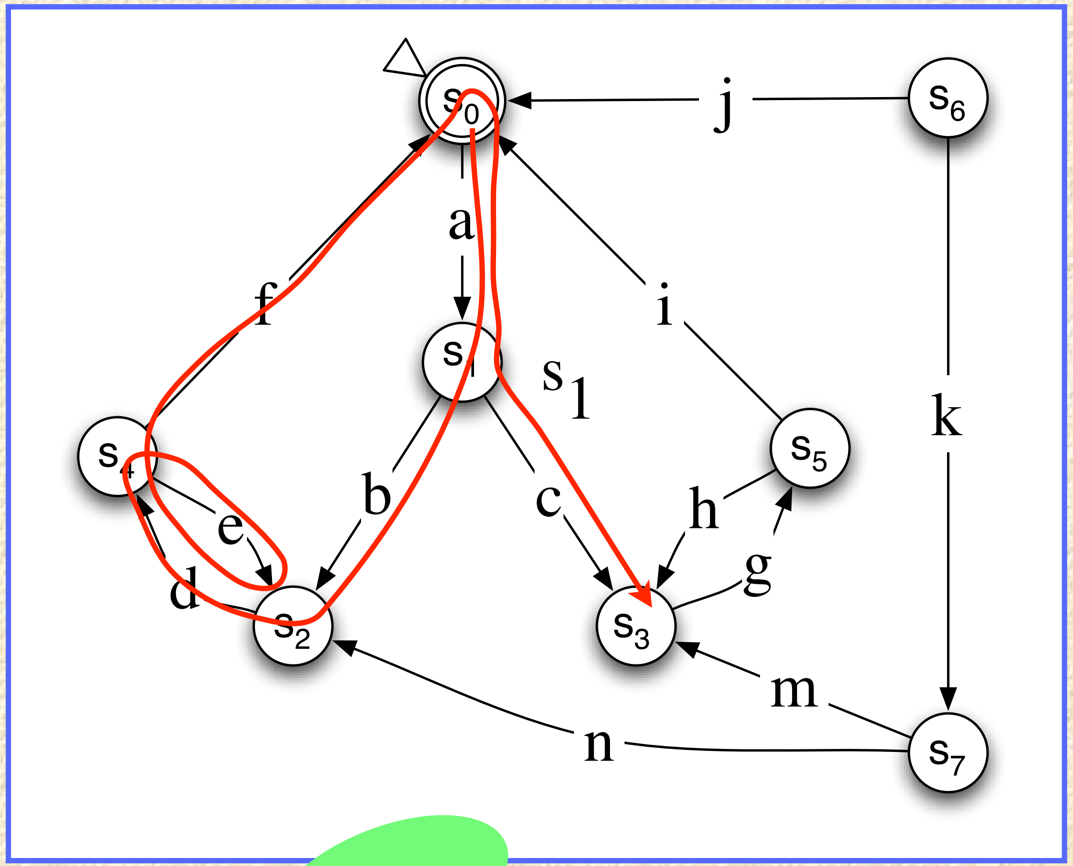
Kapitel 1

1 Transitionssysteme und Verifikation

zentrale Begriffe des Kapitels

$abdedfac \in AS(TS)$

$acgiabdededf \in L(TS)$



Aktions-Sequenzen

s0

- $AS(TS, S_1) := \{w \in A^* \mid \exists s_1 \in S_1, s_1' \in S : s_1 \xrightarrow{w} s_1'\}$ *als* Menge der von S_1 aus in TS möglichen Aktionsfolgen *und*

- $AS(TS) := AS(TS, S^0)$ *als* Aktionsfolgenmenge von TS *definiert.*

Finale Aktions-Sequenzen

- $FS(TS) := \{w \in A^* \mid \exists s_1 \in S_0, \boxed{s_1' \in S^F} : s_1 \xrightarrow{w} s_1'\}$ *ist die* terminale Aktionsfolgenmenge von TS. *Diese Menge heißt oft auch* akzeptable Aktionsfolgenmenge *oder* akzeptierte Sprache *und wird als* $L(TS)$ *bezeichnet.*

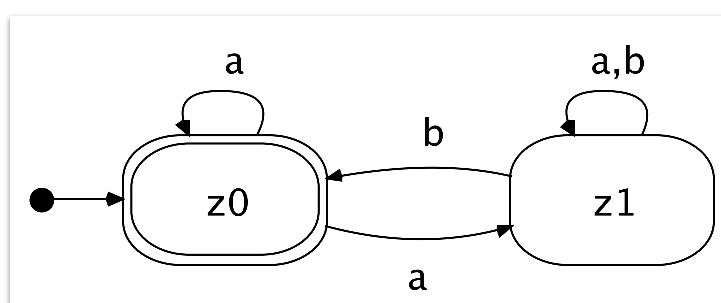
mögliche (terminale) Aktionsfolgen: Sprache $E_A(TS)$

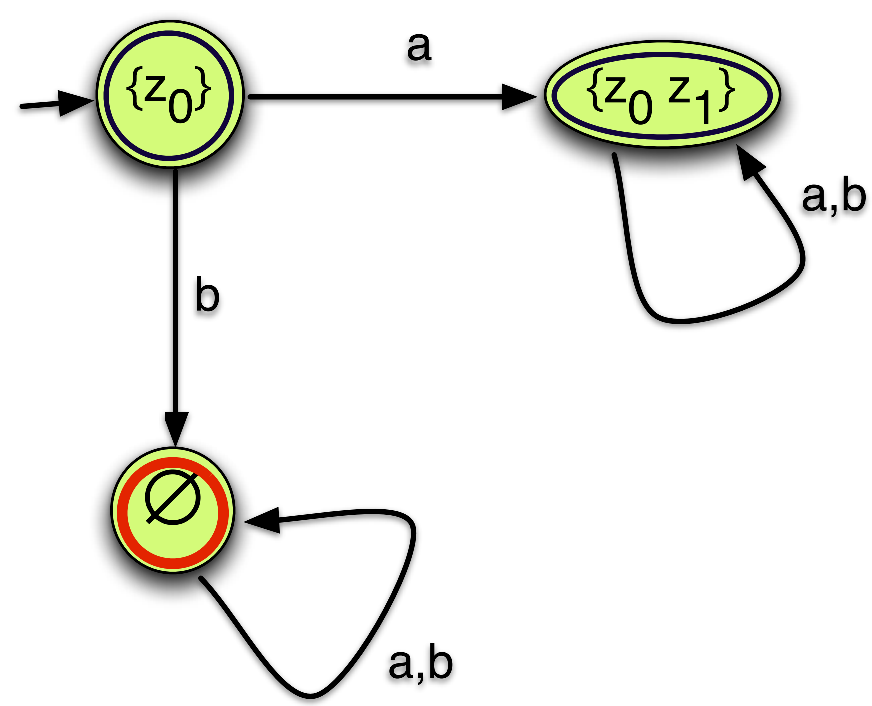
Präsenzaufgabe 1.1:

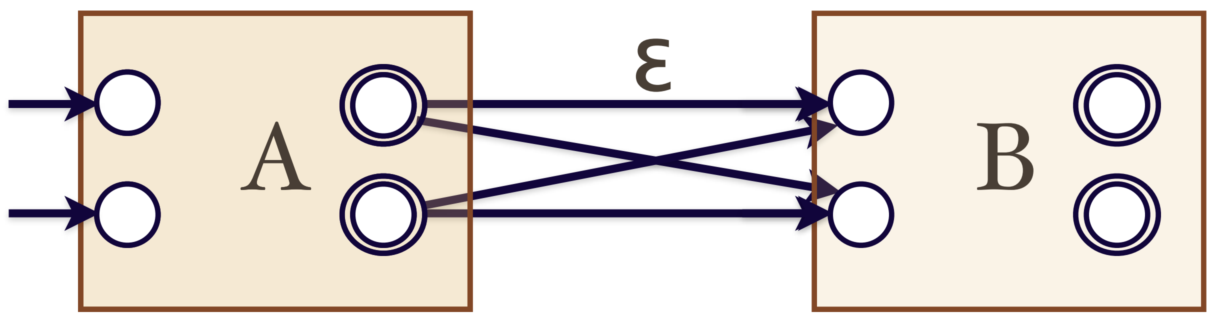
1. Wir wissen aus FGI-1, dass es zu jedem NFA A einen DFA B mit $L(A) = L(B)$ gibt. Kann man B aus A berechnen? Wenn ja, wie?

   Lösung: Potenzautomatenkonstruktion.

3. Konstruiere den Potenzautomaten für folgenden NFA:



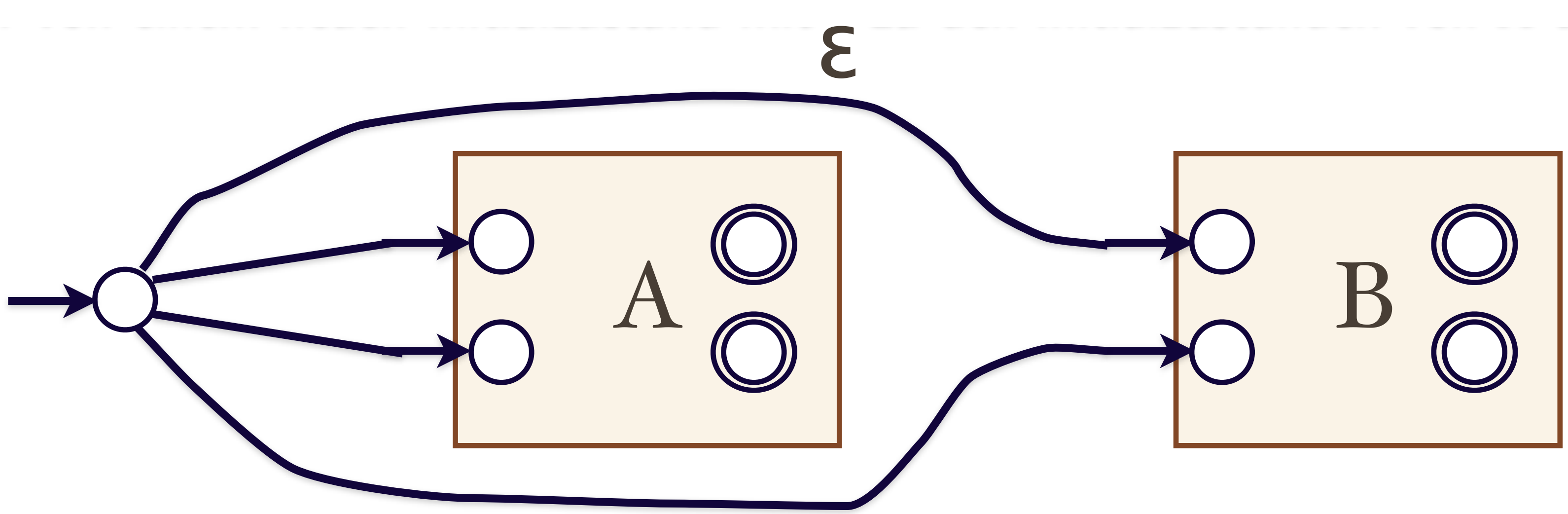




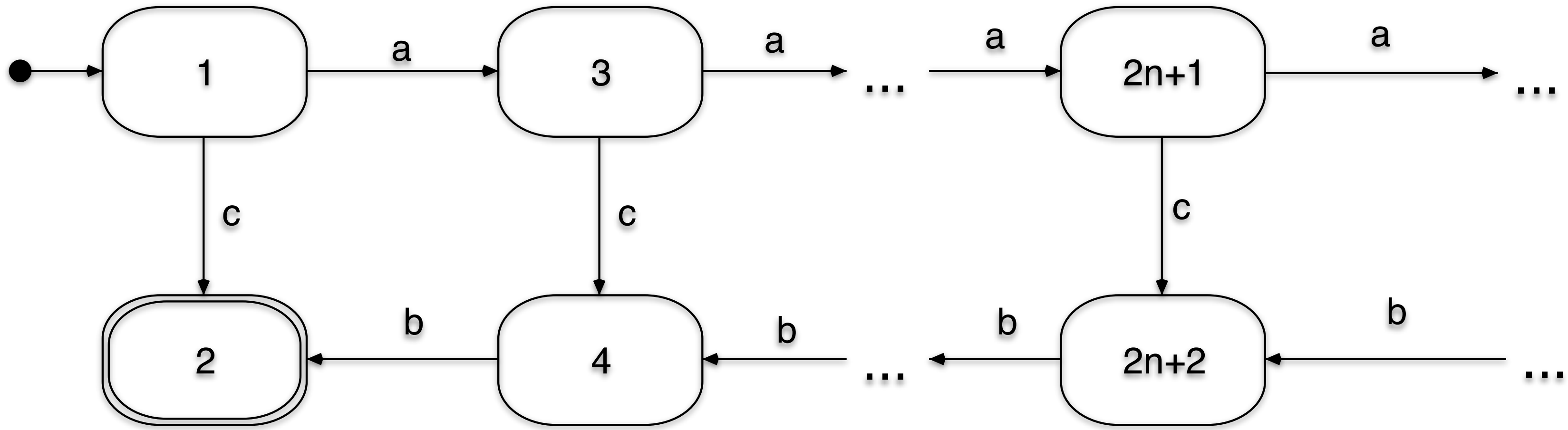
Präsenzaufgabe 1.2: Seien A und B zwei NFA.

1. Gib eine Konstruktionsvorschrift für einen NFA $C_{A \circ B}$ an, für den $L(C_{A \circ B}) = L(A) \circ L(B)$ gilt. (Hierbei ist $A \circ B$ das Komplexprodukt, das durch elementweise Konkatenation entsteht.)

   Lösung: O.B.d.A. seien A und B DFA mit disjunkten Zustandsmengen. $C_{A,B}$ entsteht, indem wir von jedem Endzustand von A mit ϵ zu dem Initialzustand von B verbinden. Die Endzustände sind die von von B, der Initialzustand ist der von A.



Präsenzaufgabe 2.1: Betrachte das unendliche Transitionssystem TS:



1. Bestimme $L(TS)$!

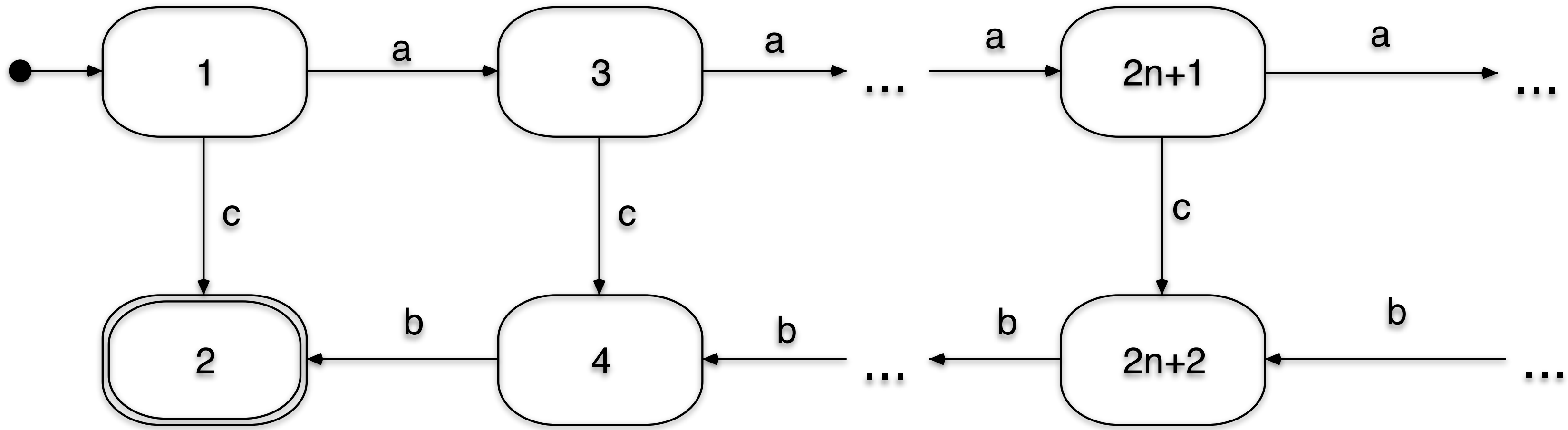
   Lösung: Es ist $L(TS) = \{a^n cb^n \mid n \in \mathbb{N}\}$

2. Ist $L(TS)$ regulär?

3. Kann es ein endliches TS geben, dass $L(TS)$ akzeptiert?

   Lösung: Da $L(TS)$ nicht regulär ist, folgt aus Satz 1.17 a), dass es kein endliches TS geben kann, das die Sprache akzeptiert.

Präsenzaufgabe 2.1: Betrachte das unendliche Transitionssystem TS:



1. Bestimme $L(TS)$!

   Lösung: Es ist $L(TS) = \{a^n c b^n \mid n \in \mathbb{N}\}$

4. Wir definieren die Etikettenfunktion $E_A : A \to \Sigma \cup \{\epsilon\}$ durch $E_A(a) = E_A(b) = E_A(c) = d$. Bestimme $E_A(TS)$!

   Lösung: Es ist $E_A(TS) = \{d^{2n+1} \mid n \in \mathbb{N}\}$.

5. Ist $E_A(TS)$ für obiges E_A regulär? Steht dies im Widerspruch zu vorherigen Resultaten?

   Lösung: $E_A(TS)$ ist eine reguläre Menge, sie kann durch den regulären Ausdruck $(dd)^*d$ beschrieben werden.

Sprachen unendlicher Folgen/Wörter

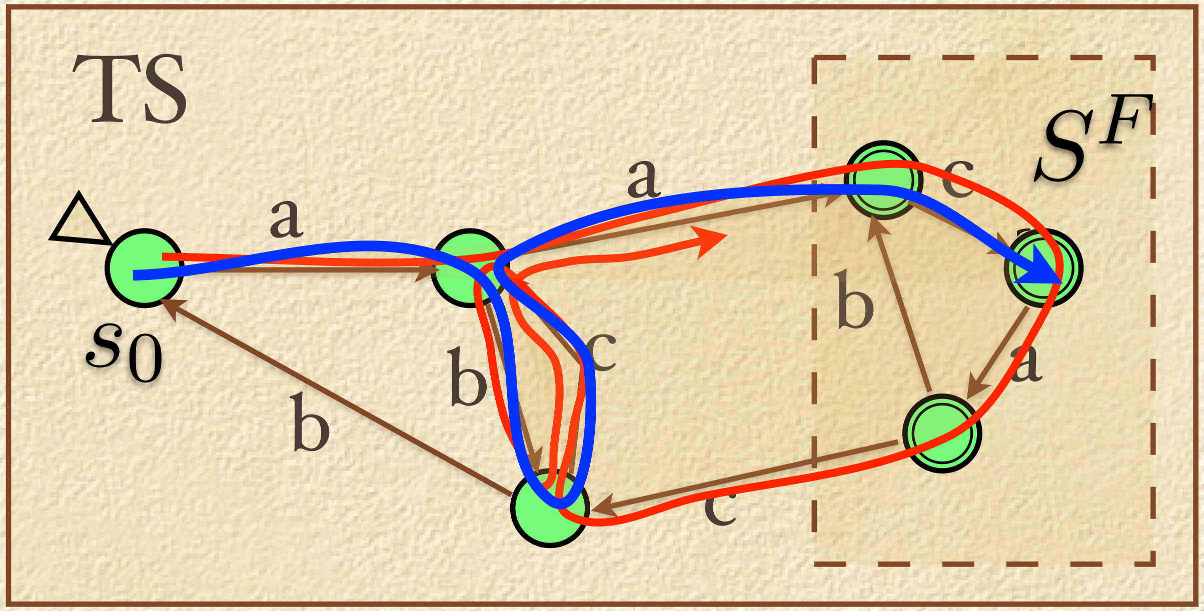
sei A^ω die Menge der unendlichen Folgen von Zeichen aus A

$w \in A^\omega$ wird als $w = a_0 a_1 a_2 \cdots$ dargestellt.

$$L(TS)$$

$$\mathrm{infinite}(w) := \{a \in A | a \textit{ kommt in } w \textit{ unendlich oft vor}\}$$

$$L^\omega(TS) = ?$$

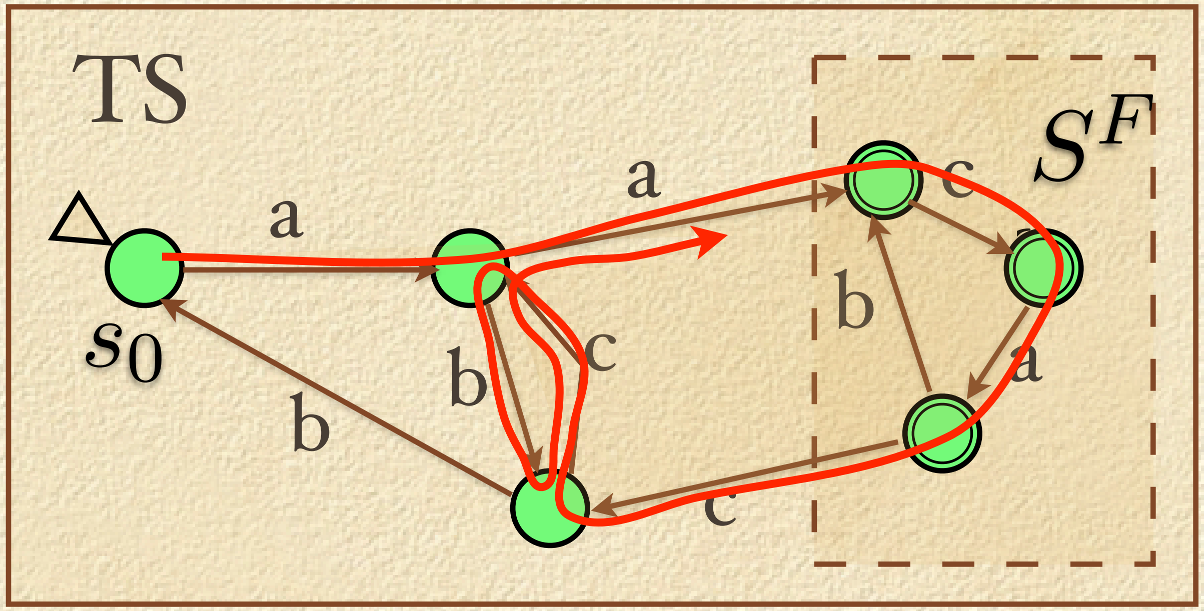


$w = a_0 a_1 a_2 \cdots$ Pfad $s_0 \xrightarrow{a_0} s_1 \xrightarrow{a_1} s_2 \xrightarrow{a_2} s_3 \cdots$

$\text{infinite}(s_0 s_1 s_2 \cdots) \cap S^F \neq \emptyset \Leftrightarrow w$ akzeptabel.

die von TS akzeptierte ω-Sprache

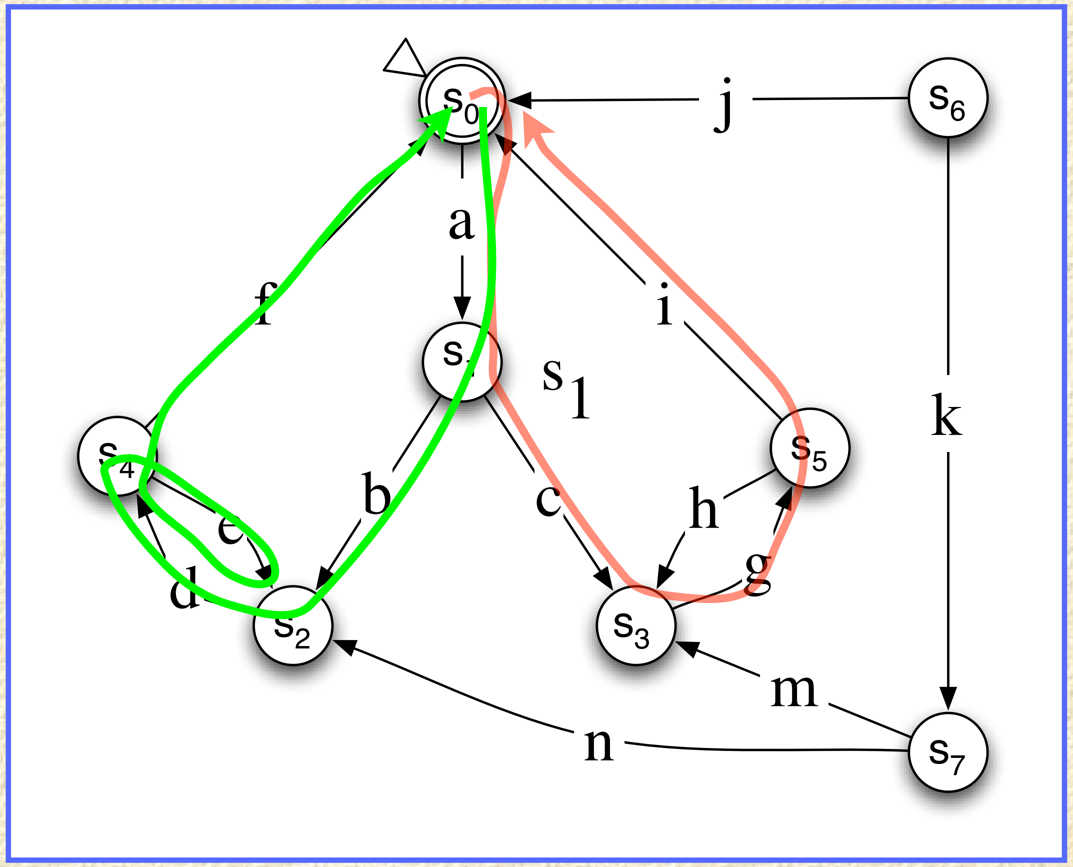
$$L^\omega(TS) = ?$$

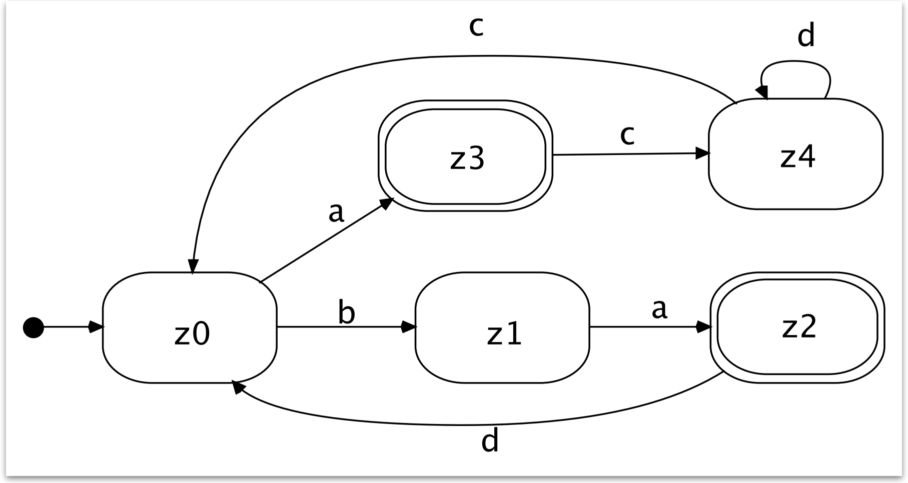


$$abdededf(acgi)^{\omega}$$

$$\in L^{\omega}(TS)$$

$$abdededef(acgi)^{\omega}$$





1. Bestimme $L(TS)$!

   Lösung: $L(TS) = (acd^*c + bad)^*(a + ba)$

2. Bestimme $L^\omega(TS)$!

   Lösung: $L^\omega(TS) = (acd^*c + bad)^\omega$

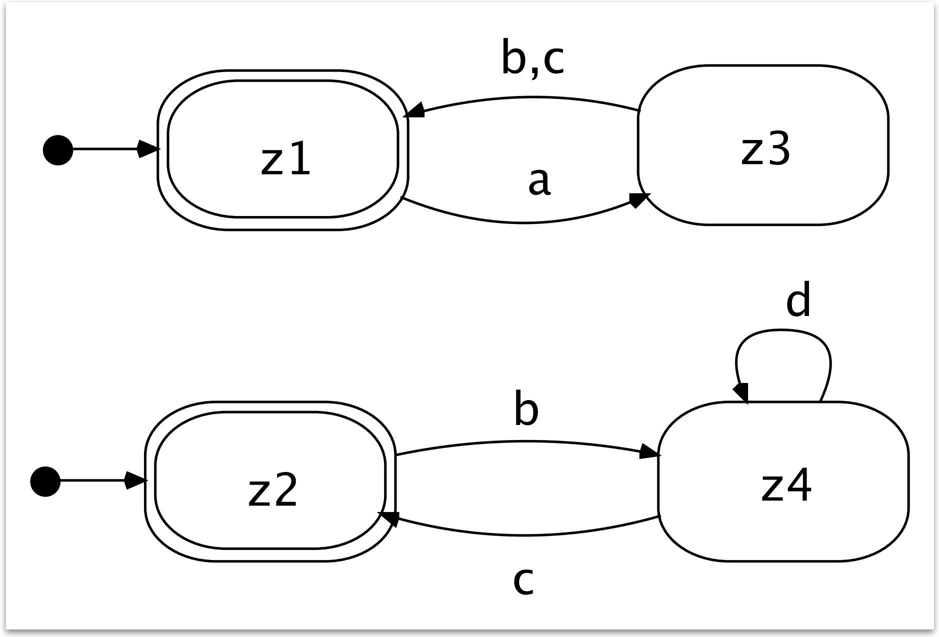
3. Angenommen z_2 sei nicht mehr Endzustand und sei TS' das resultierende TS. Bestimme dann die resultierende Sprache $L^\omega(TS')$!

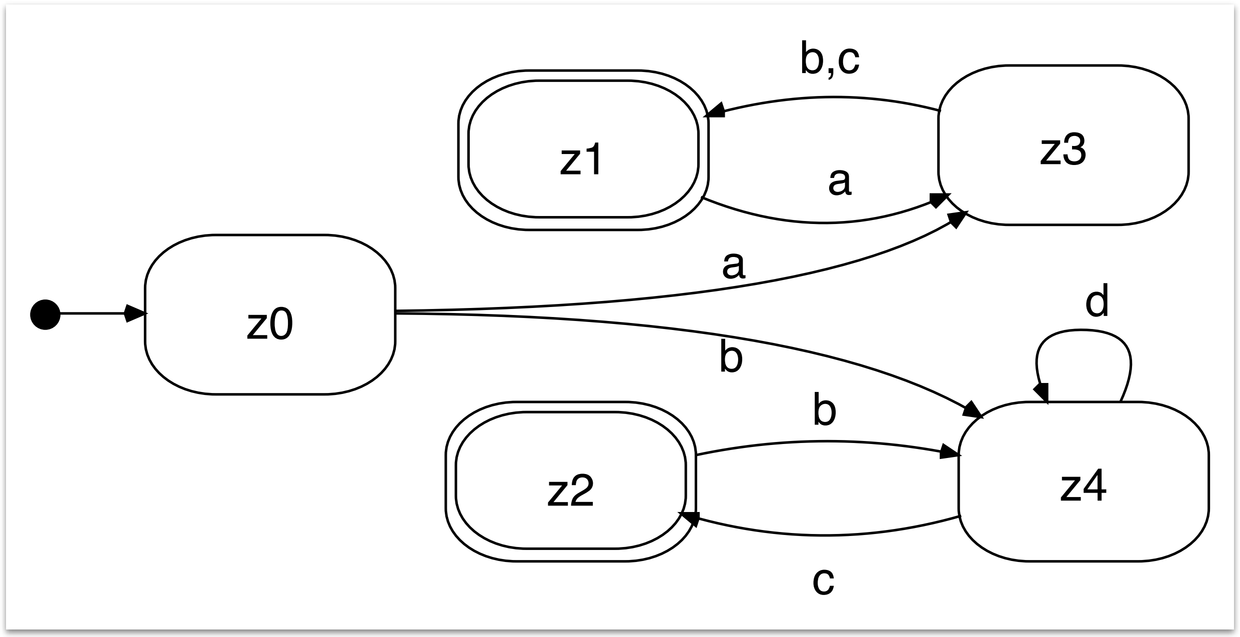
   Lösung: $L^\omega(TS') = ((bad)^* \cdot acd^*c)^\omega$

   (Die obere Schleife muss unendlich oft auftreten, um z_3 unendlich oft zu besuchen. Die untere Schleife $z_0z_1z_2$ kann sowohl endlich als auch unendlich oft auftreten. Eine unendliche Wiederholung der unteren Schleife wird aber nur akzeptiert, wenn auch die obere Schleife mit Endzustand z_3 unendlich oft dazwischen auftritt.)

4. Gib ein TS mit $L^\omega(TS) = (a(b+c))^\omega + (bd^*c)^\omega$ an!

Lösung: Zwei Alternativen:





Die rechte Lösung besitzt nur einen Startzustand und ist daher etwas komplizierter als die linke.

zurück zu Sprachen endlicher Folgen/Wörter

$Zwei \quad Transitionssysteme$

$$TS_1 \quad = \quad (S_1, A, tr_1, S_1^0)$$

$$TS_1 \quad = \quad (S_1, A, tr_1, S_1^0)$$

„folgenäquivalent“, falls

$$TS_1 \sim_{AS} TS_2 :\Leftrightarrow AS(TS_1) = AS(TS_2)$$

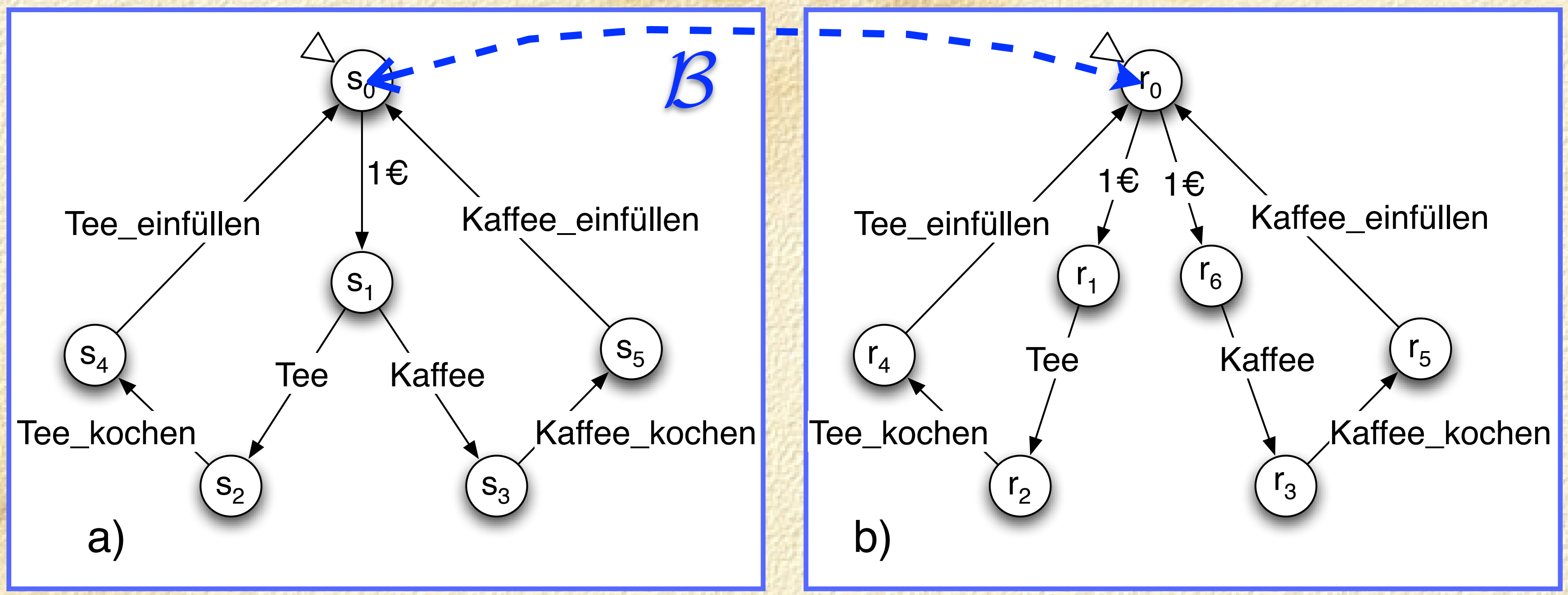


Abbildung 1.4: Zwei folgenäquivalente Transitionssysteme

Definition 1.7 *Gegeben seien für $i \in \{1,2\}$ zwei Transitionssysteme $TS_i = (S_i, A, tr_i, S_i^0, S_i^F)$ mit Endzuständen (und der gleichen Aktionsmenge A) . Eine (aktionsbasierte)* Bisimulation *für (TS_1, TS_2) ist eine binäre Relation $\mathcal{B} \subseteq S_1 \times S_2$, für die folgendes gilt:*

$$\begin{aligned} a)\ & \forall s_0 \in S_1^0\ \exists r_0 \in S_2^0 : (s_0, r_0) \in \mathcal{B} \\ & \forall r_0 \in S_2^0\ \exists s_0 \in S_1^0 : (s_0, r_0) \in \mathcal{B} \end{aligned}$$

$$c)\ \forall (s,r) \in \mathcal{B} : s \in S_1^F \Leftrightarrow r \in S_2^F$$

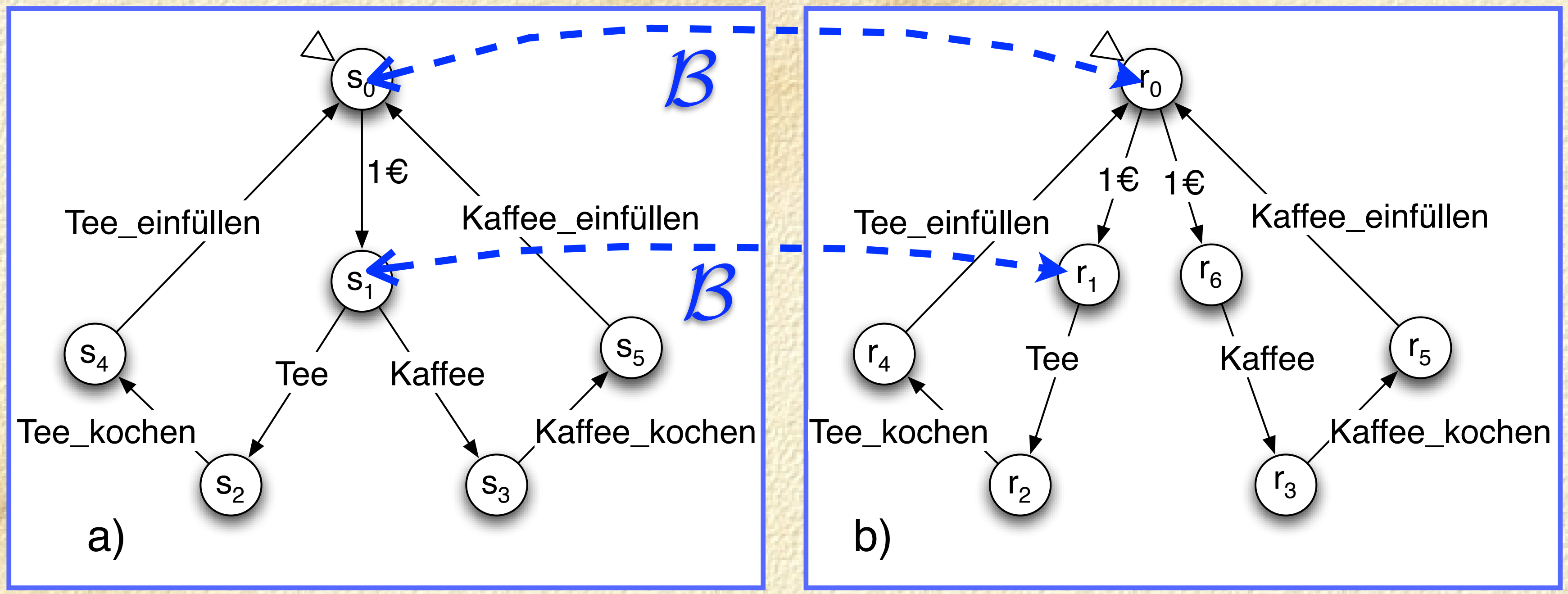


Abbildung 1.4: Zwei folgenäquivalente Transitionssysteme

Definition 1.7 *Gegeben seien für $i \in \{1,2\}$ zwei Transitionssysteme $TS_i = (S_i, A, tr_i, S_i^0, S_i^F)$ mit Endzuständen (und der gleichen Aktionsmenge A)*[5]*. Eine (aktionsbasierte)* $\boxed{\text{Bisimulation}}$ *für (TS_1, TS_2) ist eine binäre Relation $\mathcal{B} \subseteq S_1 \times S_2$, für die folgendes gilt:*

b) Für alle $(s_1, r_1) \in \mathcal{B}$ gilt:

$$s_1 \xrightarrow{a}_1 s_2 \Rightarrow \exists r_2 \in S_2 : r_1 \xrightarrow{a}_2 r_2 \wedge (s_2, r_2) \in \mathcal{B}$$

$$r_1 \xrightarrow{a}_2 r_2 \Rightarrow \exists s_2 \in S_1 : s_1 \xrightarrow{a}_1 s_2 \wedge (s_2, r_2) \in \mathcal{B}$$

$$\begin{array}{ccc} s_1 & \xrightarrow{a}_1 & s_2 \\ \vdots & & \vdots \\ r_1 & \xrightarrow{a}_2 & r_2 \end{array} \quad !$$

TS_1 und TS_2 heißen bisimilar *(in Zeichen $TS_1 \underline{\leftrightarrow} TS_2$) falls eine solche Bisimulationsrelation $\mathcal{B}$ existiert. Zustände mit $(s, r) \in \mathcal{B}$ heißen* bisimilar *(in Zeichen $s \underline{\leftrightarrow} r$).*

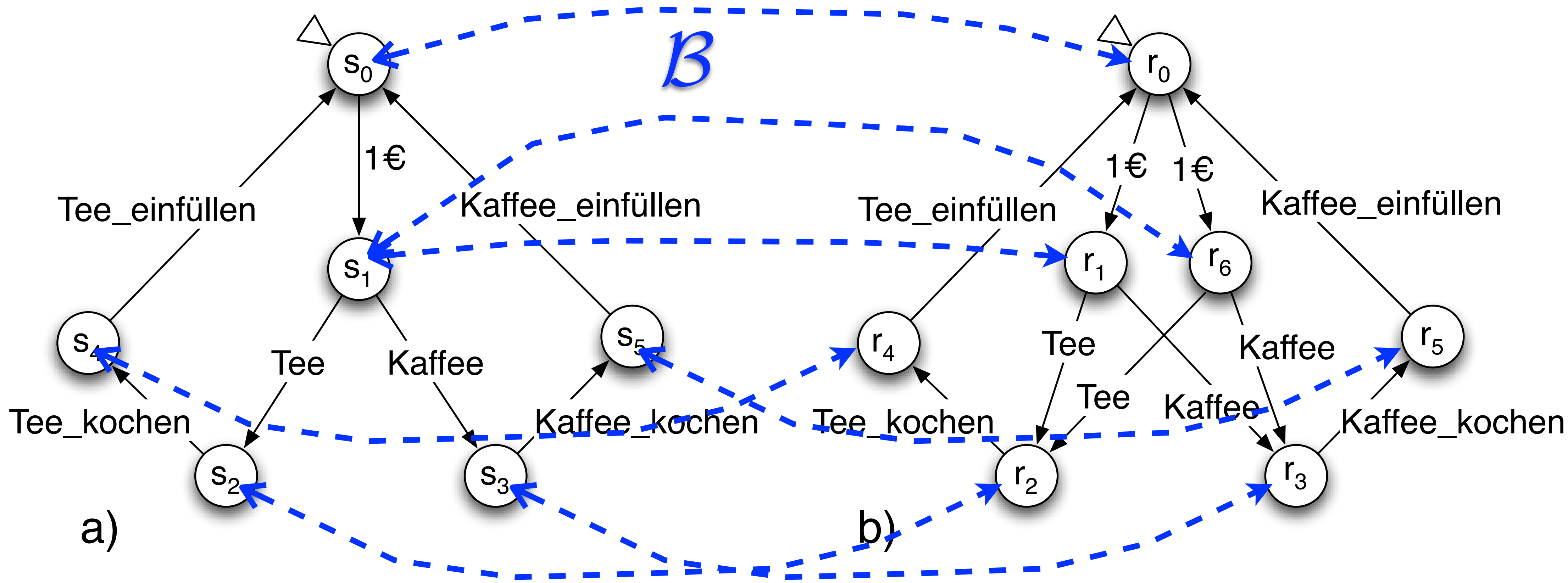
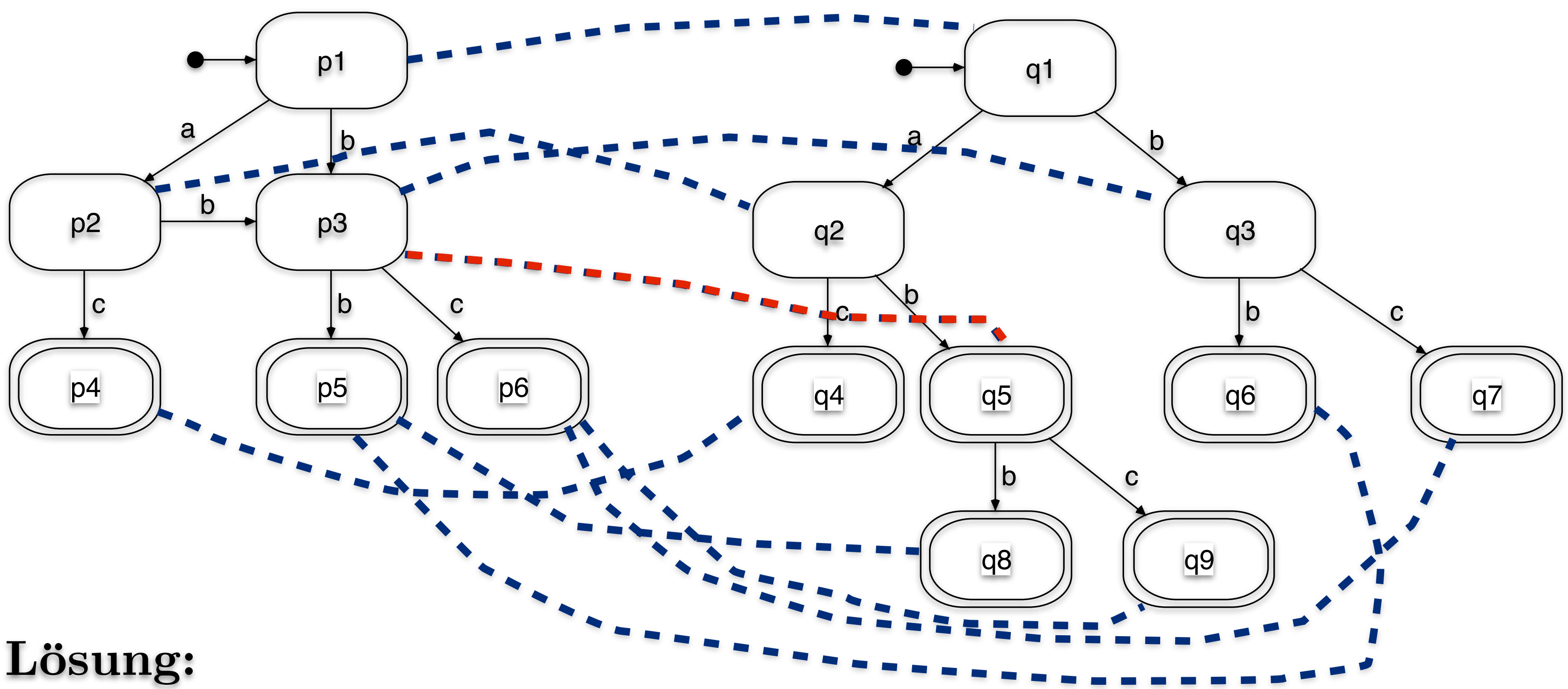


Abbildung 1.5: Zwei bisimilare Transitionssysteme

Präsenzaufgabe 2.2: Prüfen Sie, ob die folgenden Transitionssysteme bisimilar sind. Geben Sie die Bisimulationsrelation explizit an.



1.

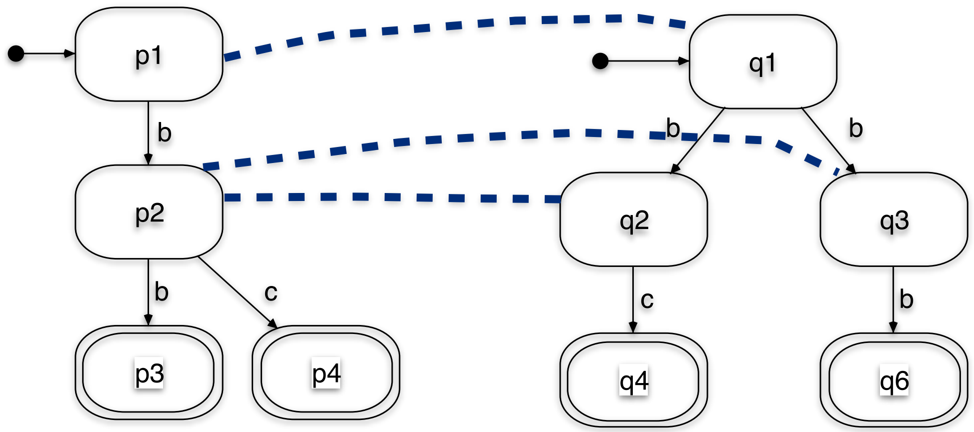
Lösung:

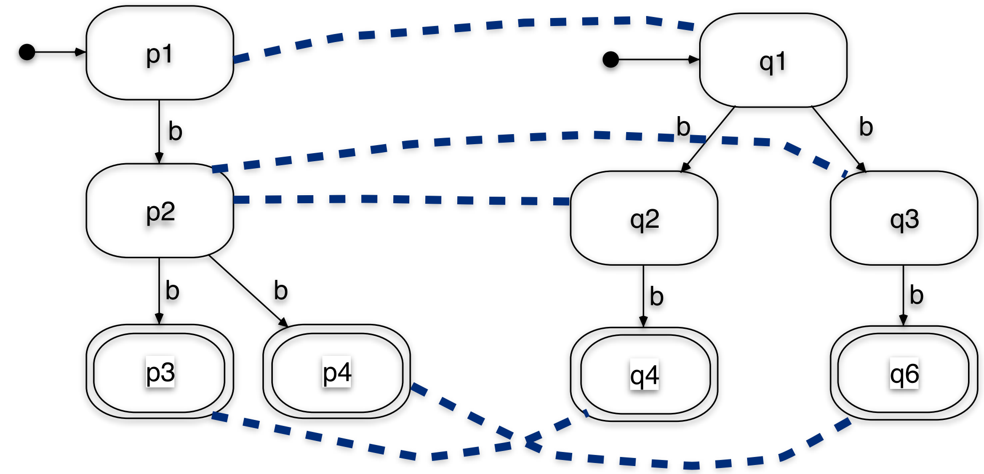
1. Es bietet sich die folgende Relation an:

$$\{(p_1, q_1), (p_2, q_2), (p_3, q_3), (p_4, q_4), (p_3, q_5), (p_5, q_8), (p_6, q_9), (p_5, q_6), (p_6, q_7)\}$$

   Einziger Nachteil: Das Paar (p_3, q_5), denn nur einer ist Endzustand. Diese Relation eignet sich also nicht als Bisimulation. Es ist aber zu begründen, dass keine einzige Bisimulationsrelation existiert.

   Anderer Ansatz: Die beiden Transitionssysteme sind nicht akzeptanzäquivalent (links gibt es die terminale Aktionsfolge ab, rechts nicht). Gemäß Satz 1.8 können nicht akzeptanzäquivalente TS auch nicht bisimilar sein.

2. 

3. 

2. Dies ist der Klassiker für nicht bisimilare TS. Die Begründung läuft über die Eigenschaften aus Def. 1.7:

   - Aus Bedingung a) folgt, dass das Paar (p_1, q_1) in $\mathcal{B}$ enthalten sein muss (zu jedem Startzustand ist ein Partner erforderlich, der ebenfalls Startzustand ist).
   - Wenn $(p_1, q_1) \in \mathcal{B}$, dann muss gemäß Bedingung b) auch $(p_2, q_2) \in \mathcal{B}$ und $(p_2, q_3) \in \mathcal{B}$ gelten.
   - Beide Paare verletzen jeweils Bedingung b), denn in p_2 ist Aktion b möglich, zu welcher q_2 keine Entsprechung hat. Ebenso ist in p_2 die Aktion c möglich, welche in q_3 keine Entsprechung hat.

3. Die beiden TS sind trotz der strukturellen Ähnlichkeit zu Teil 2 bisimilar, da die jeweils 2. Aktion gleich ist. $\mathcal{B} = \{(p_1, q_1), (p_2.q_2), (p_2, q_3), (p_3, q_4), (p_3, q_5), (p_4, q_4), (p_4, q_5)\}$

Satz 1.8 *Wenn zwei Transitionssysteme* $TS_i = (S_i, A, tr_i, S_i^0, S_i^F)$ *mit Endzuständen (und der gleichen*[7] *Aktionsmenge* A *und* $i \in \{1,2\}$*)* $\boxed{\textit{bisimilar}}$ *sind, dann sind sie auch* $\boxed{\textit{akzeptanzäquivalent}}$*, aber nicht umgekehrt,*
d.h.: $\boxed{TS_1 \leftrightarrows TS_2} \Rightarrow \boxed{TS_1 \sim_L TS_2}$*).*

zum Beweis: vollständige Induktion über die Wortlänge

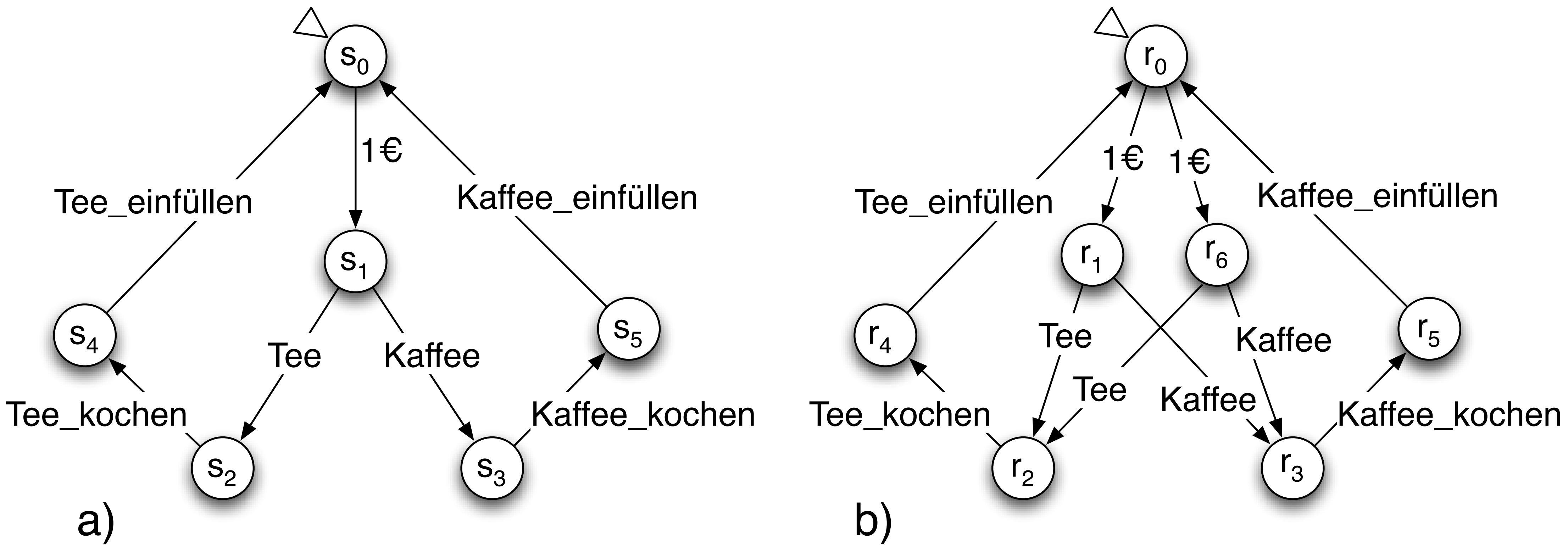


Abbildung 1.5: Zwei bisimilare Transitionssysteme

Satz 1.8 *Wenn zwei Transitionssysteme* $TS_i = (S_i, A, tr_i, S_i^0, S_i^F)$ *mit Endzuständen (und der gleichen*[7] *Aktionsmenge* A *und* $i \in \{1, 2\}$*)* $\boxed{\textit{bisimilar}}$ *sind, dann sind sie auch* $\boxed{\textit{akzeptanzäquivalent}}$*, aber nicht umgekehrt,*

d.h.: $\boxed{TS_1 \underline{\leftrightarrow} TS_2} \Rightarrow \boxed{TS_1 \sim_L TS_2).}$

$\Leftarrow$?

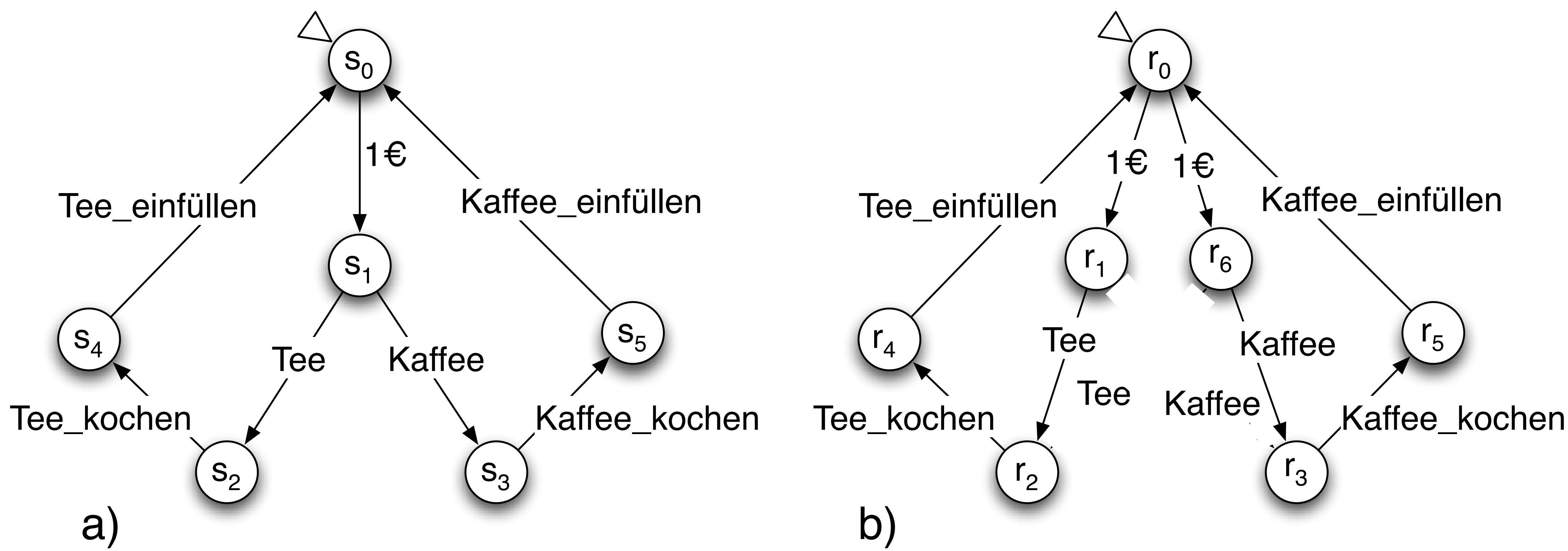
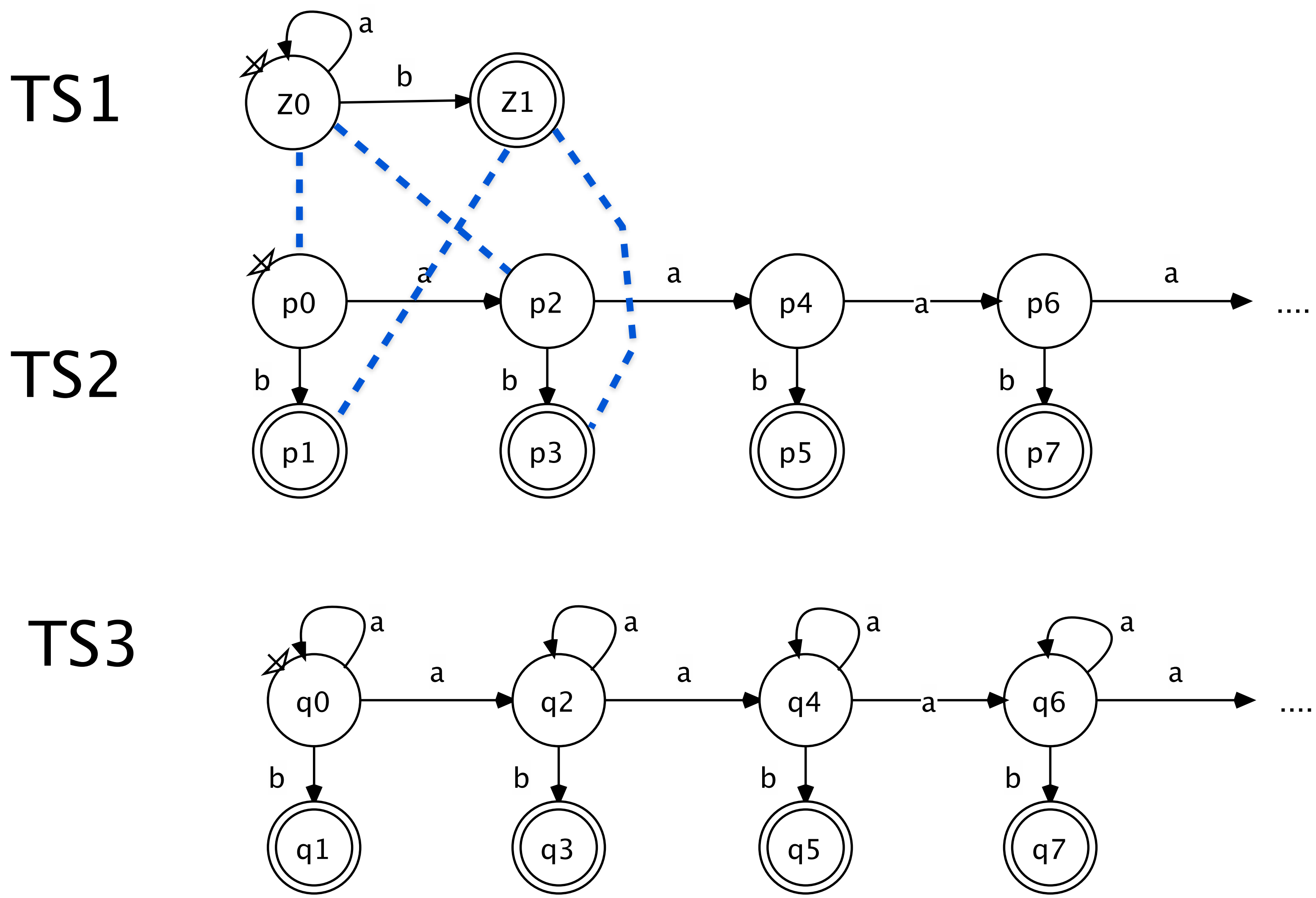


Abbildung 1.5: Zwei nicht bisimilare Transitionssysteme

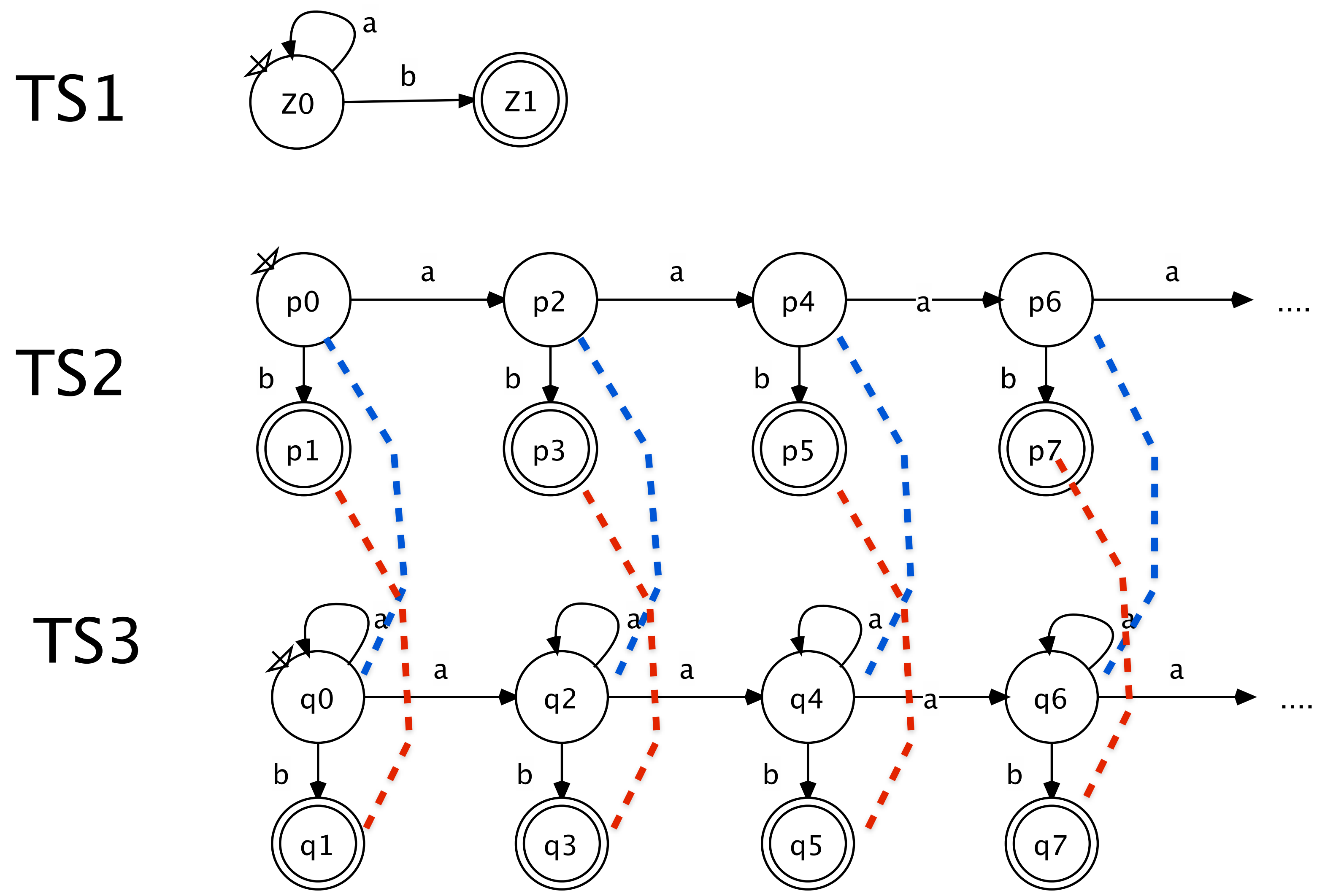
Übungsaufgabe 2.4: Prüfen Sie für alle Zweierkombination der folgenden drei Transitionssysteme, ob diese bisimilar sind. Geben Sie die Bisimulationsrelation explizit an und weisen Sie nach, dass die relationierten Zustände die Definition der Bisimulation erfüllen.

(Sie können sich Arbeit sparen, wenn sie beachten, dass folgende Symmetrie gilt: $TS_1 \underline{\leftrightarrow} TS_2$ impliziert $TS_2 \underline{\leftrightarrow} TS_1$.)



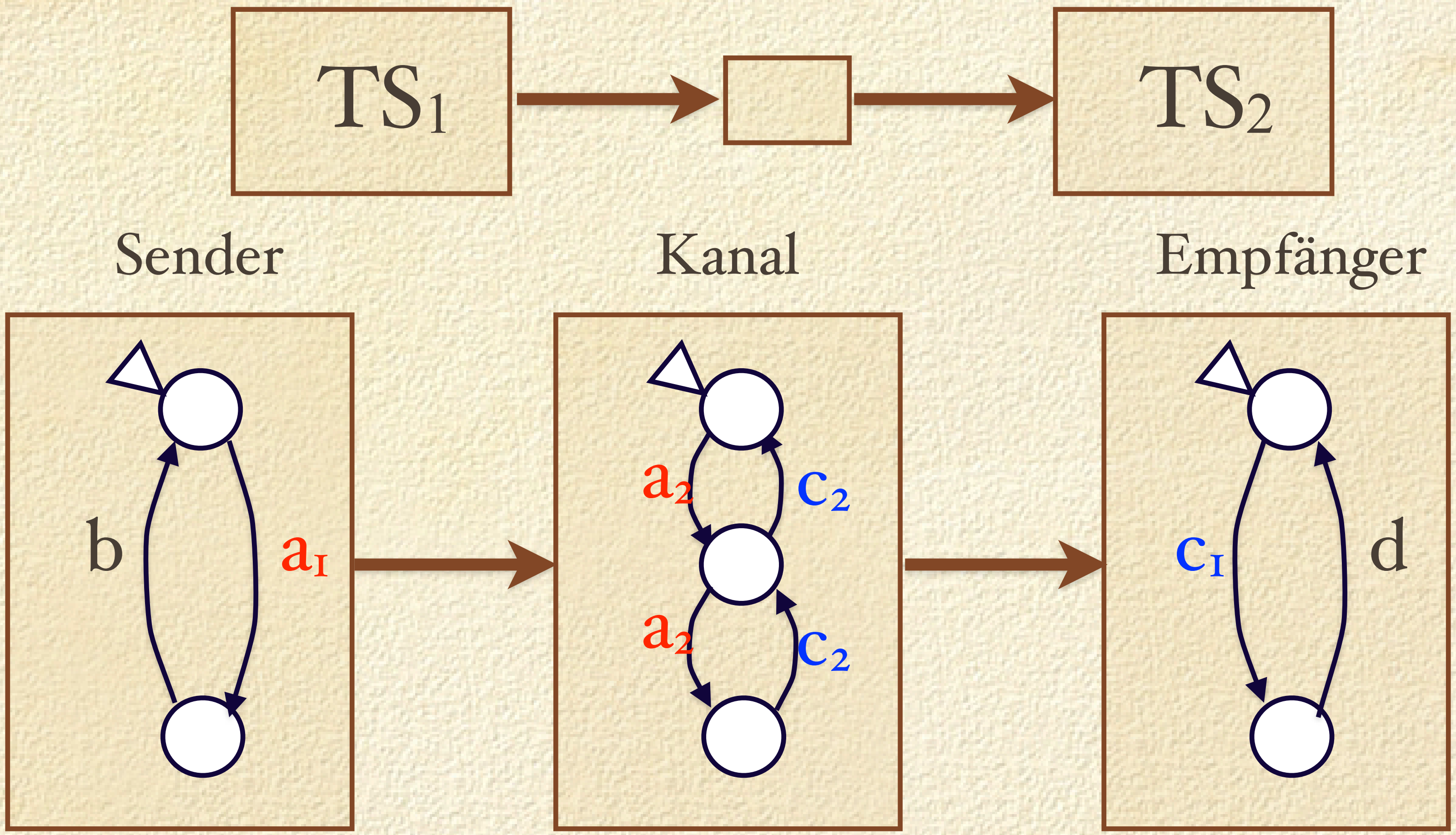
Übungsaufgabe 2.4: Prüfen Sie für alle Zweierkombination der folgenden drei Transitionssysteme, ob diese bisimilar sind. Geben Sie die Bisimulationsrelation explizit an und weisen Sie nach, dass die relationierten Zustände die Definition der Bisimulation erfüllen.

(Sie können sich Arbeit sparen, wenn sie beachten, dass folgende Symmetrie gilt: $TS_1 \underline{\leftrightarrow} TS_2$ impliziert $TS_2 \underline{\leftrightarrow} TS_1$.)



1.2 Produkte von Transitionssystemen

c) *Nachrichten-Synchronisation*

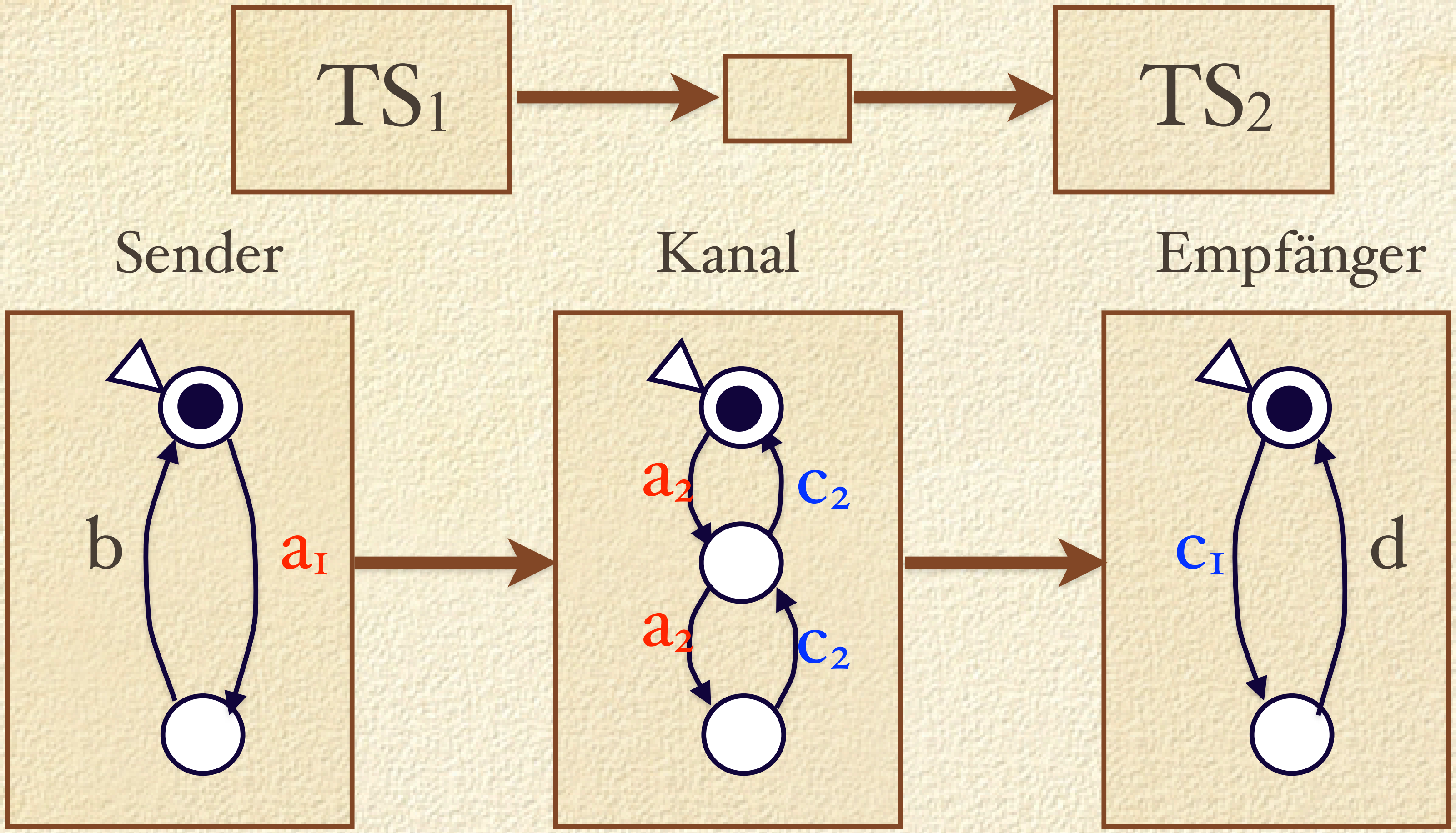


$$Sync = \{(a_1,a_2),(c_1,c_2)\}$$

$$\gamma(a_1,a_2) = a, \;\; \gamma(c_1,c_2) = c$$

1.2 Produkte von Transitionssystemen

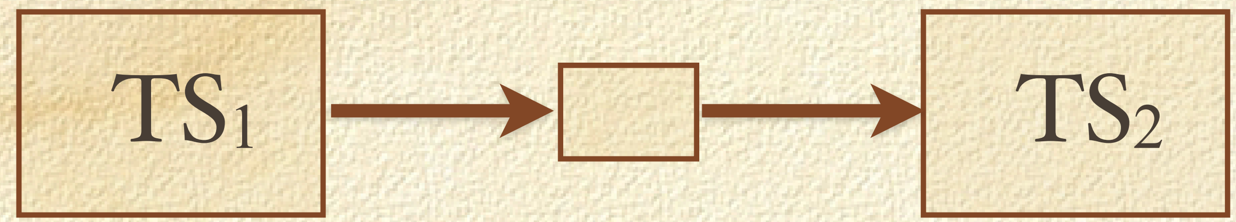
c) *Nachrichten-Synchronisation*

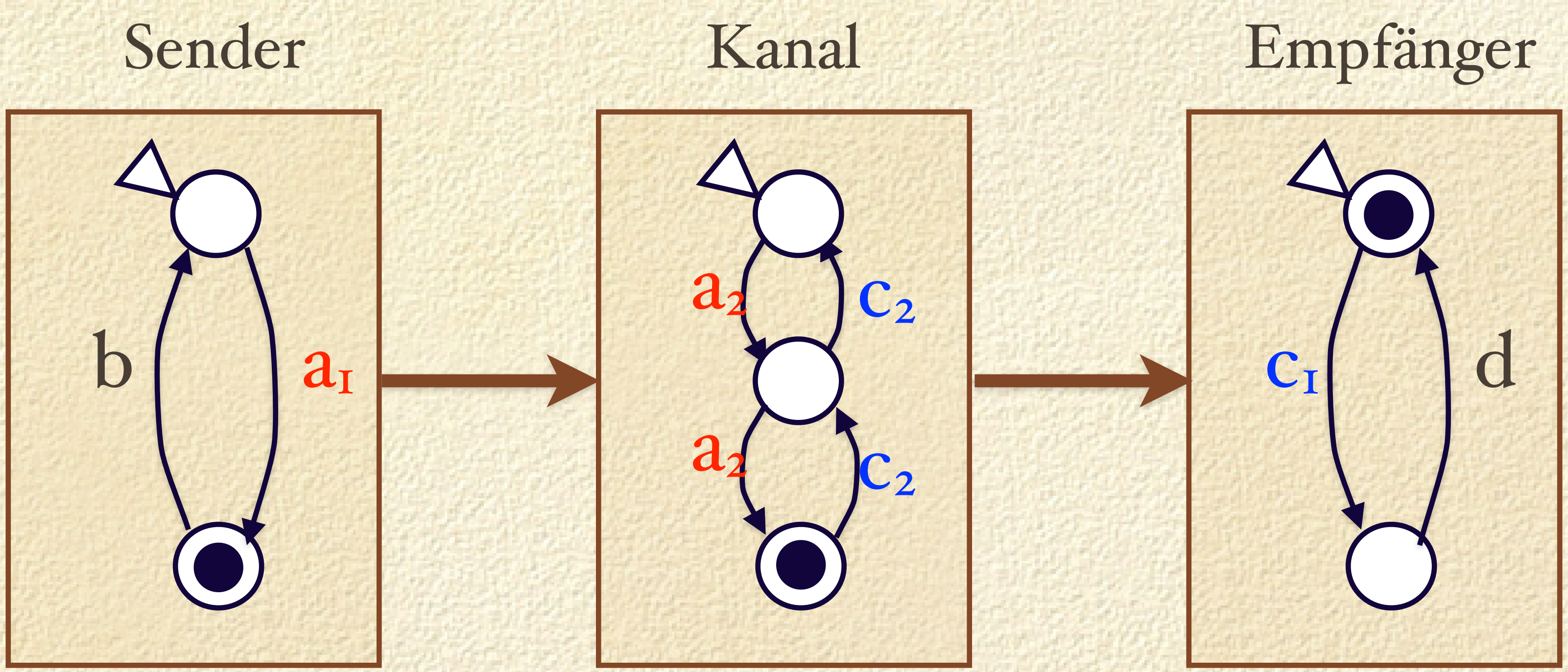


$$Sync = \{(a_1,a_2),(c_1,c_2)\}$$

$$\gamma(a_1,a_2) = a,\ \ \gamma(c_1,c_2) = c\}$$

c) *Nachrichten-Synchronisation*





$$Sync = \{(a_1,a_2),(c_1,c_2)\}$$

$$\gamma(a_1,a_2) = a,\ \ \gamma(c_1,c_2) = c\}$$

$$TS_1 \xleftrightarrow[\gamma]{Sync \subseteq A_1 \times A_2} TS_2$$

Definition 1.9 *Gegeben seien zwei Transitionssysteme* $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ $(i \in \{1,2\})$ *(mit nicht notwendig gleichen Aktionsmengen* A_i*). Das* Produkttransitionssystem *von* TS_1 *und* TS_2 *wird als Transitionssystem*

$$TS_1 \otimes_\gamma TS_2 = (\boxed{S_1 \times S_2}, \boxed{A_1 \cup A_2 \cup \cancel{Sync}}, tr_3, S_1^0 \times S_2^0, S_1^F \times S_2^F, \gamma)$$

definiert, wobei

$$A_3$$

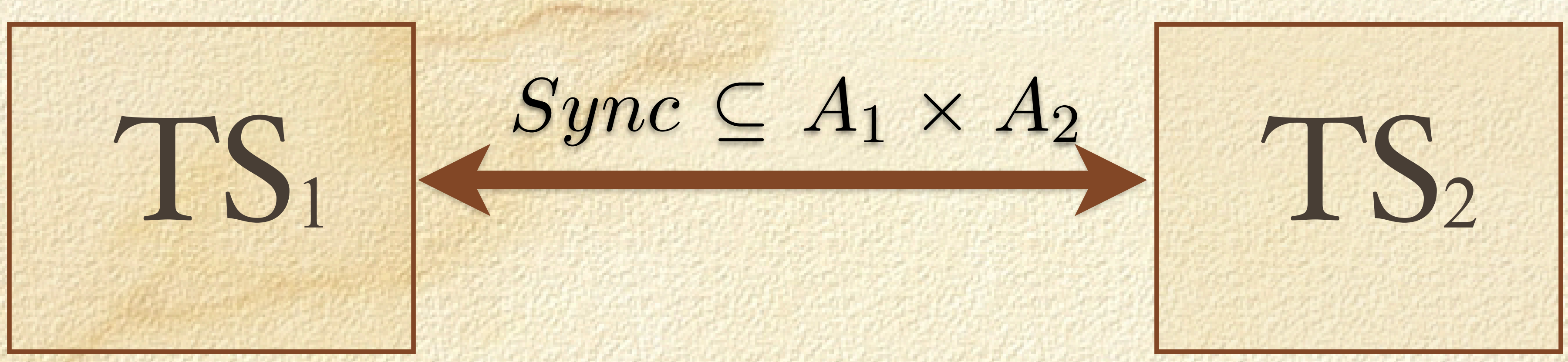
a) $Sync \subseteq A_1 \times A_2$ *eine* Synchronisations-Relation,

b) $\gamma : Sync \to A_3$ *eine Abbildung von* $Sync$ *in ein* Kommunikationsalphabet[7] A_3 *ist und*

Die Kommunikationsabbildung γ *ist optional.*

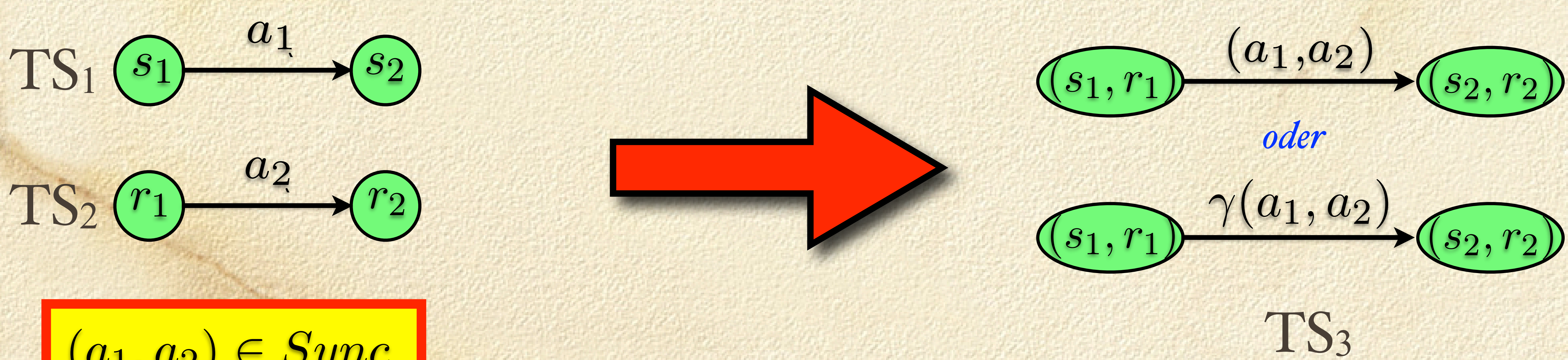
Transitionen

durch Regeln

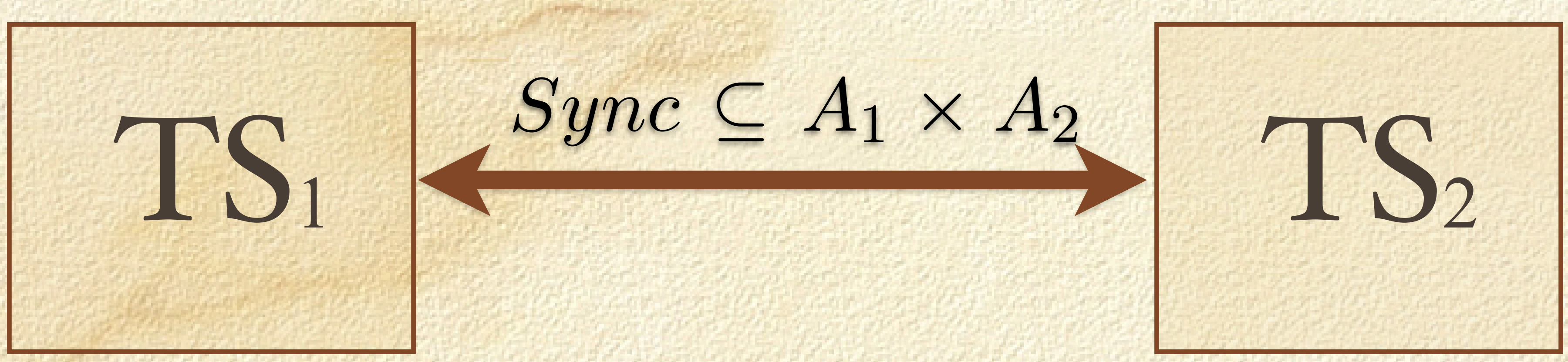


Regel $Sy3$: Kommunikation

$$\frac{s_1 \xrightarrow{a_1}_1 s_2, \quad r_1 \xrightarrow{a_2}_2 r_2, \quad (a_1, a_2) \in Sync}{(s_1, r_1) \xrightarrow{(a_1,a_2)}_3 (s_2, r_2)}$$

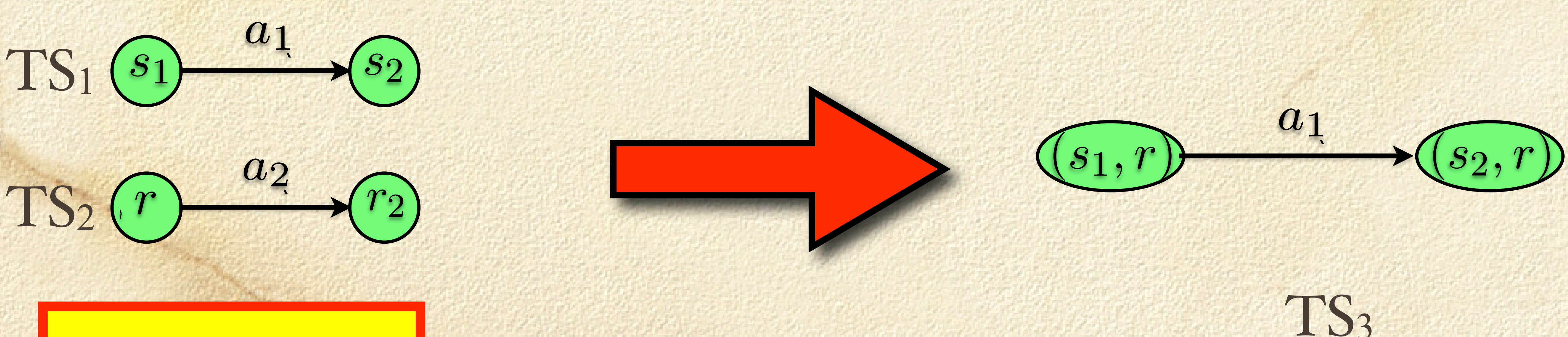


$\boxed{(a_1, a_2) \in Sync}$

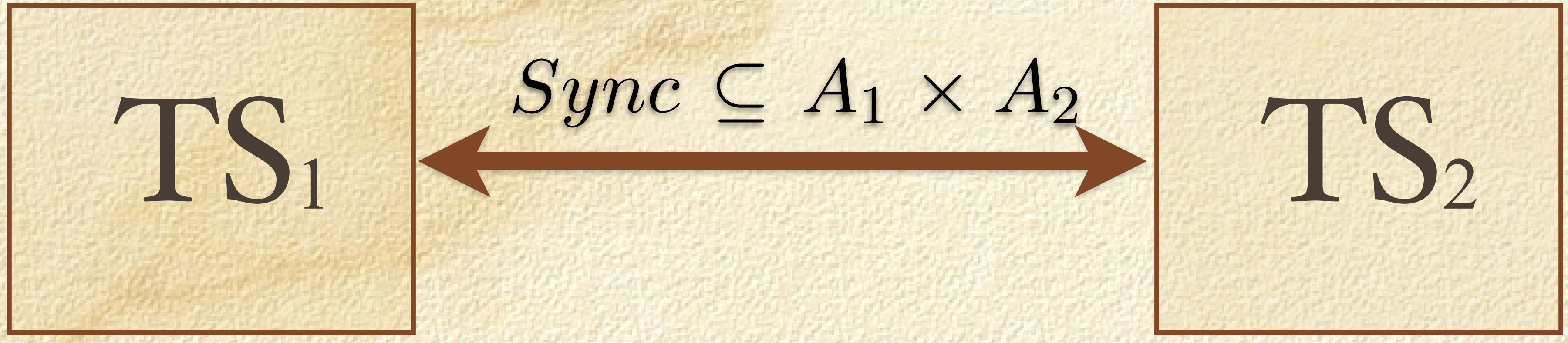


Regel $Sy1$*:* Erste Komponente

$$\frac{s_1 \xrightarrow{a_1}_1 s_2, \quad a_1 \notin pr_1(Sync), \quad r \in S_2}{(s_1, r) \xrightarrow{a_1}_3 (s_2, r)}$$

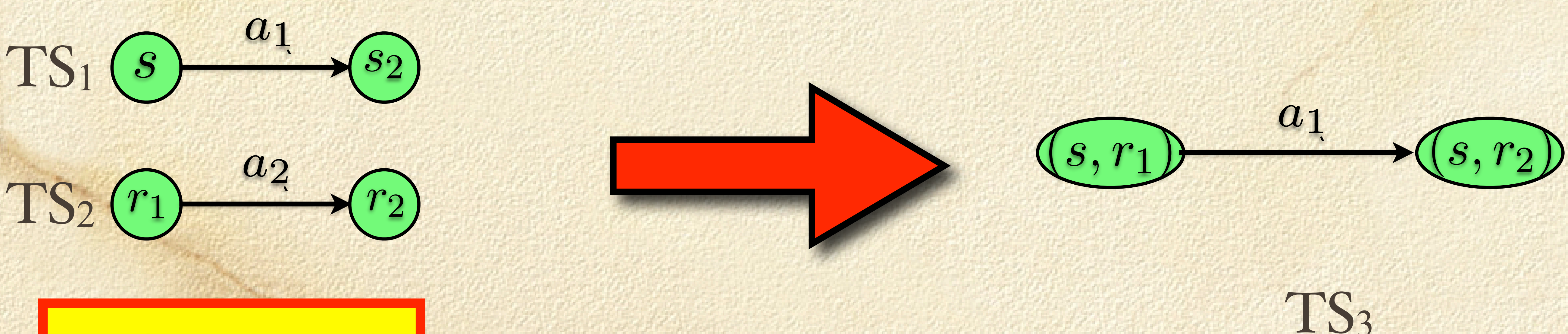


$\boxed{a_1 \notin pr_1(Sync)}$



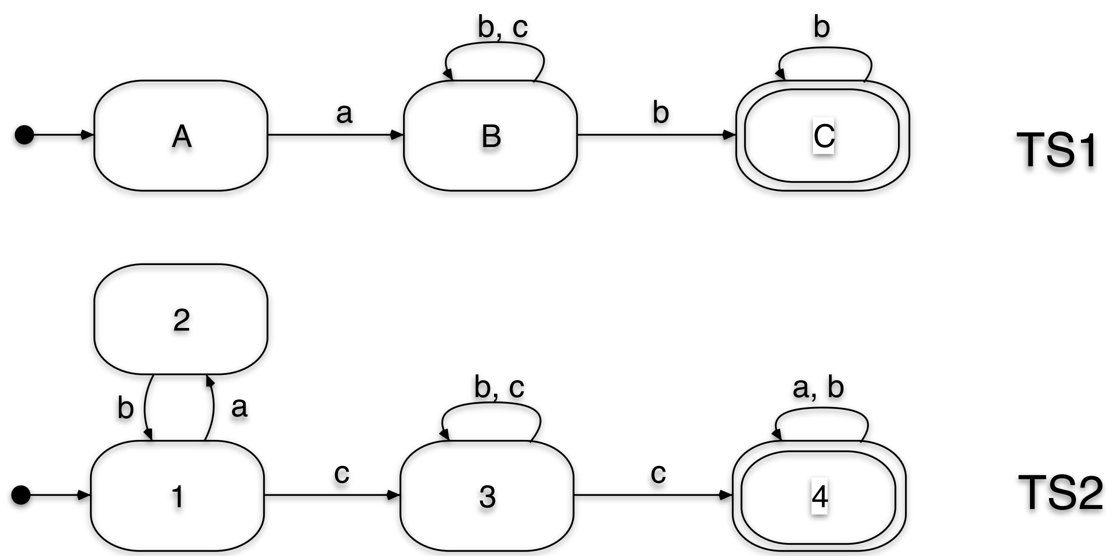
Regel $Sy2$: Zweite Komponente

$$\frac{r_1 \xrightarrow{a_2}_2 r_2,\ a_2 \notin pr_2(Sync), \quad s \in S_1}{(s, r_1) \xrightarrow{a_2}_3 (s, r_2)}$$



$a_2 \notin pr_2(Sync)$

Präsenzaufgabe 1.2: Produkte

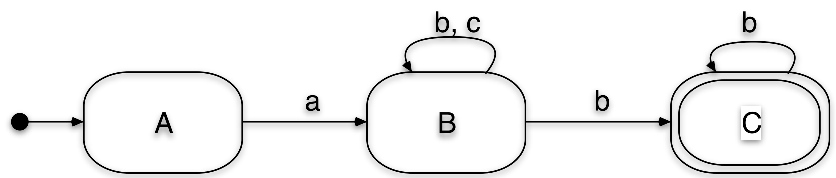


1. Bestimme $L(TS_1)$ und $L(TS_2)$.
2. Konstruiere den Produktautomaten $TS_1 \otimes TS_2$ im Sinne von Satz 1.18, d.h. für $Sync = id_A = \{(x,x) \mid x \in A\}$.
3. Bestimme $L_3 = L(TS_1 \otimes TS_2)$
4. Vergleiche L_3 mit $L(TS_1) \cap L(TS_2)$.

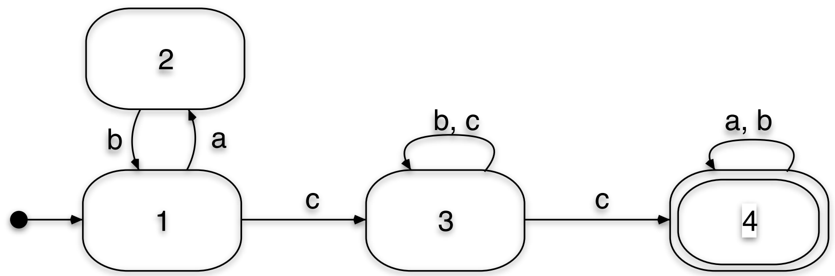
Lösung: Es ist $L(TS_1) = a(b+c)^*b^+$ und $L(TS_2) = (ab)^*c(b+c)^*c(a+b)^*$.

Das folgende TS stellt den Produktautomaten $TS_1 \otimes TS_2$ dar, wobei wir γ mit $\gamma(x,x) = x$ genutzt haben.

Präsenzaufgabe 1.2: Produkte



TS1

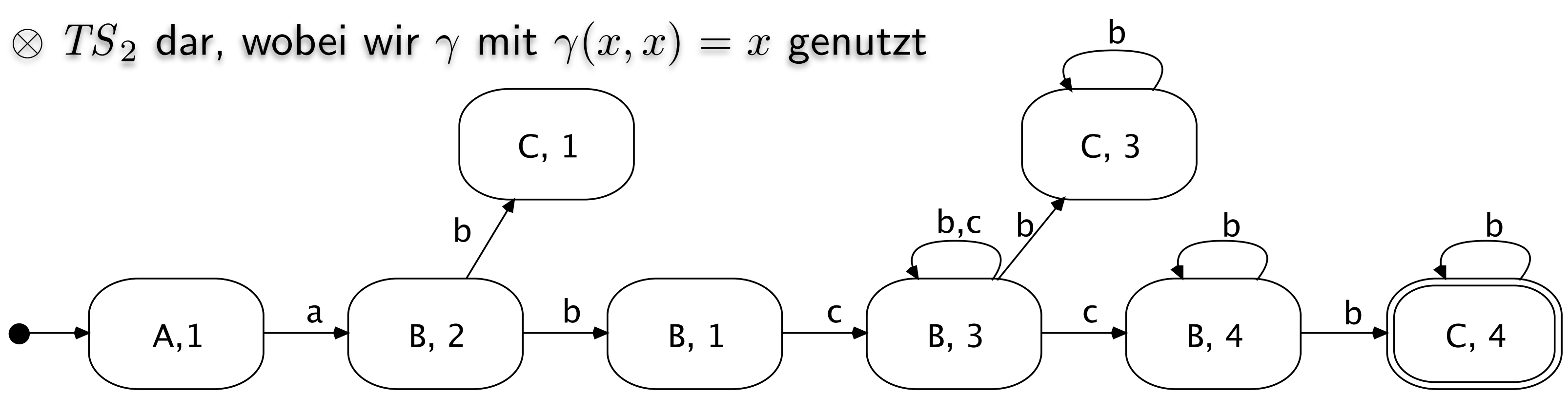


TS2

1. Bestimme $L(TS_1)$ und $L(TS_2)$.
2. Konstruiere den Produktautomaten $TS_1 \otimes TS_2$ im Sinne von Satz 1.18, d.h. für $Sync = id_A = \{(x, x) \mid x \in A\}$.
3. Bestimme $L_3 = L(TS_1 \otimes TS_2)$
4. Vergleiche L_3 mit $L(TS_1) \cap L(TS_2)$.

Lösung: Es ist $L(TS_1) = a(b + c)^* b^+$ und $L(TS_2) = (ab)^* c(b + c)^* c(a + b)^*$.

Das folgende TS stellt den Produktautomaten $TS_1 \otimes TS_2$ dar, wobei wir γ mit $\gamma(x, x) = x$ genutzt haben.



Es ist $L_3 = abc(b + c)^* cb^+$.

Im Falle der Synchronisationmenge $Sync = \{(x, x) \mid x \in A\}$ gilt stets $L(TS_1 \otimes_{Sync} TS_2) = L(TS_1) \cap L(TS_2)$.

$$\Sigma = \{a, b\}$$

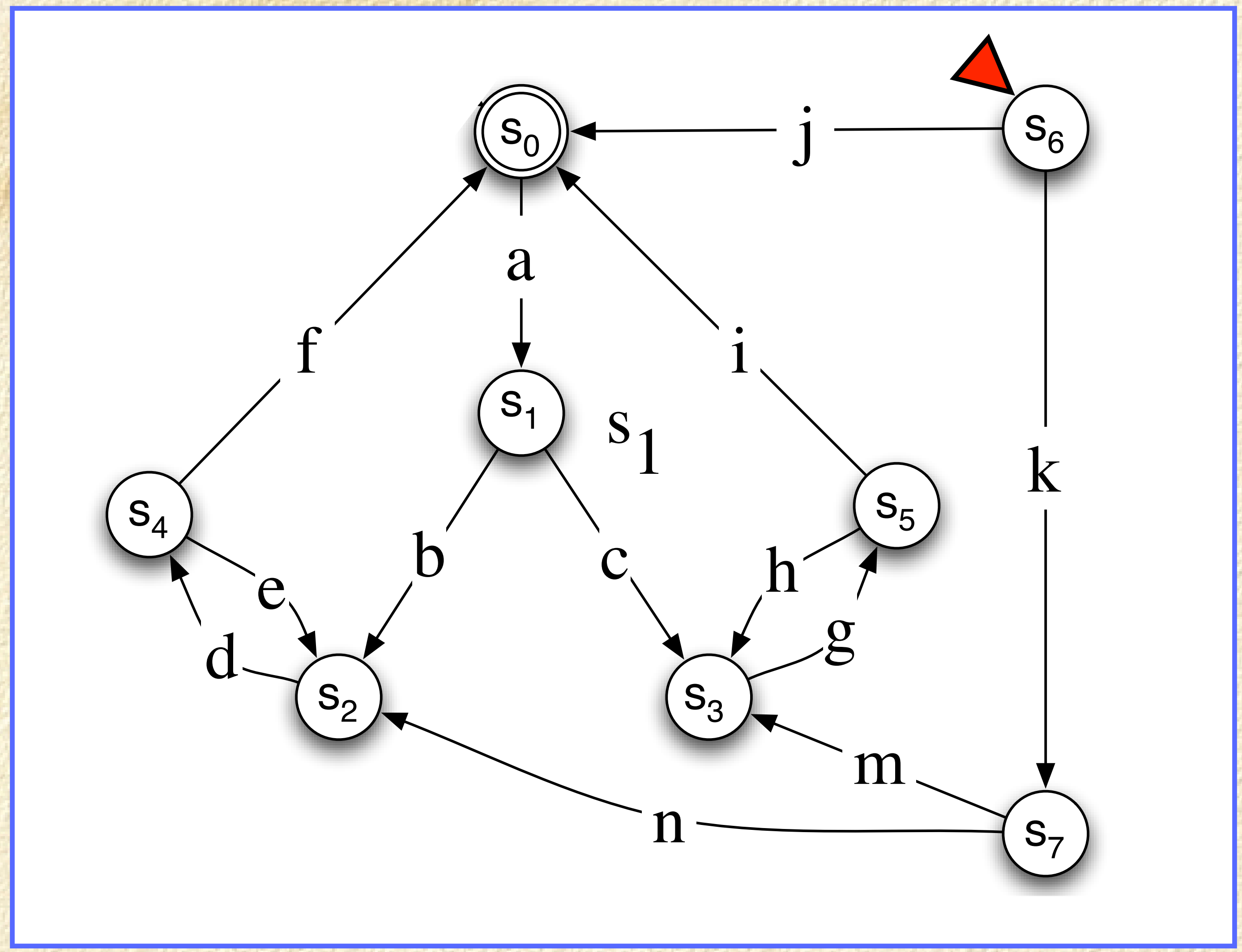
$$\{a + b\}\Sigma^* + \{a + b\}\{a + b\}\Sigma^*$$

$$\{a + b\}(\Sigma^* + \{a + b\}\Sigma^*)$$

$$\Sigma^+$$

Menge aller Wörter über Σ*, die mindestens ein Zeichen enthalten.*

ω-reguläre Ausdrücke



$$L^\omega(TS) = knd(ed)^* fC^\omega + kmg(hg)^* iC^\omega + jC^\omega$$

$$L(TS) = \underbrace{(a(bd(ed)^* f + cg(hg)^* i))^*}_{= C}.$$

$$L^\omega(TS) = knd(ed)^*fC^\omega + kmg(hg)^*iC^\omega$$

Definition 1.15

$+\, j\, C^\omega$

$$\frac{w = a_1 a_2 \cdots a_n \in \Sigma^* \qquad u = b_1 b_2 \cdots \in \Sigma^\omega}{w \cdot u := a_1 a_2 \cdots a_n b_1 b_2 \cdots \in \Sigma^\omega}$$

$$W \cdot U := \{w \cdot u | w \in W, u \in U\} \subseteq \Sigma^\omega$$

$W \subseteq \Sigma^* \quad U \subseteq \Sigma^\omega$

$$W^\omega := \{w_1 \cdot w_2 \cdot w_3 \cdots | w_i \in W \setminus \{\epsilon\}, i \geq 1\}$$

ω-Abschluss

$$L^\omega(TS) = knd(ed)^* fC^\omega + kmg(hg)^* iC^\omega$$

$+ j\,C^\omega$

$$G = A_1 \cdot B_1^\omega + \cdots + A_n \cdot B_n^\omega$$

$$M_G := M_{A_1} \cdot M_{B_1}^\omega \cup \cdots \cup M_{A_n} \cdot M_{B_n}^\omega .$$

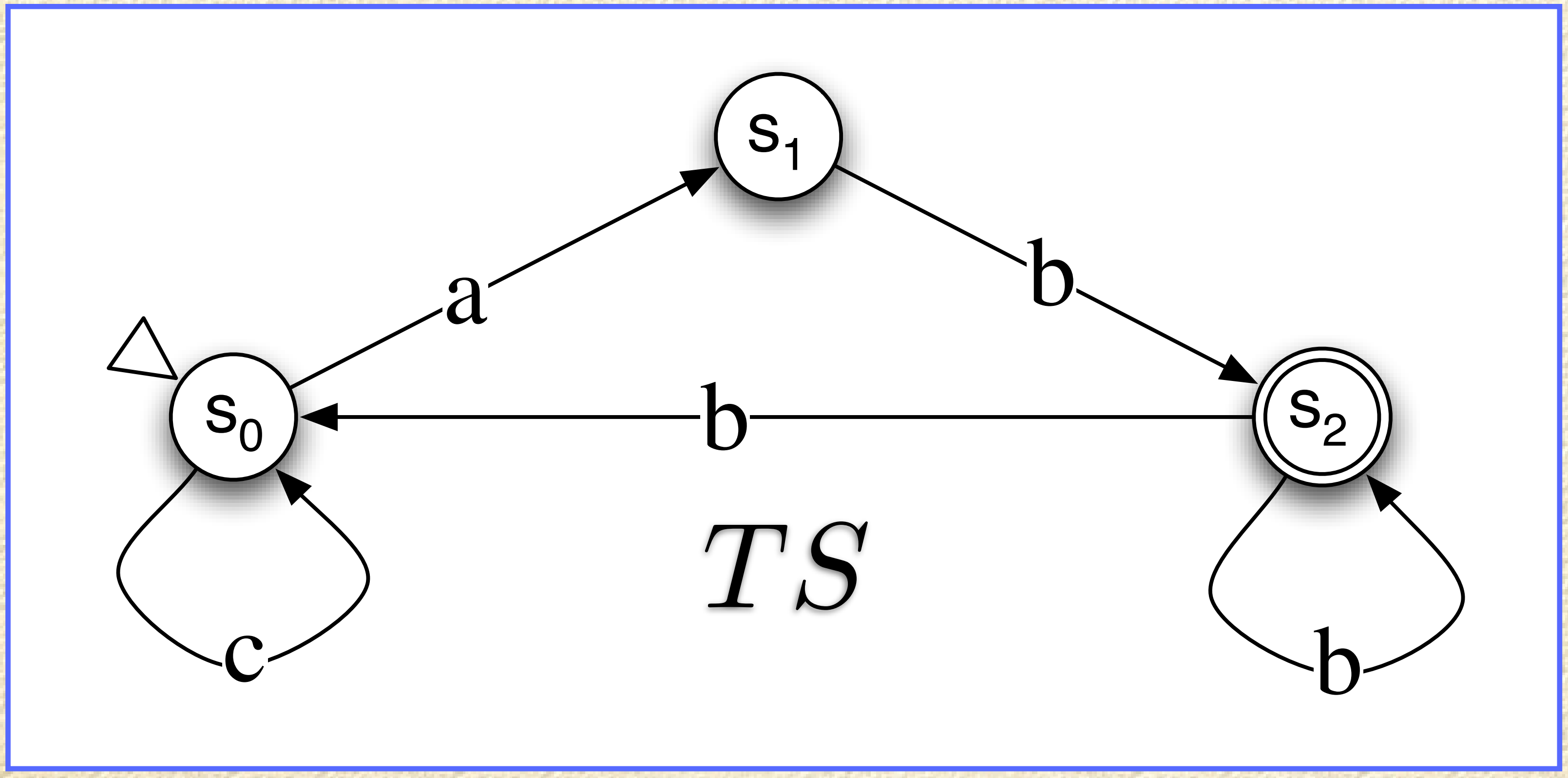
ω-reguläre Menge

ω - *rationale*

Satz 1.17 *a) Die durch endliche Transitionssysteme* $TS = (S, \Sigma, tr, S^0, S^F)$ *akzeptierten Sprachen* $L(TS)$ *sind genau die regulären Sprachen über* Σ.

Satz von Kleene

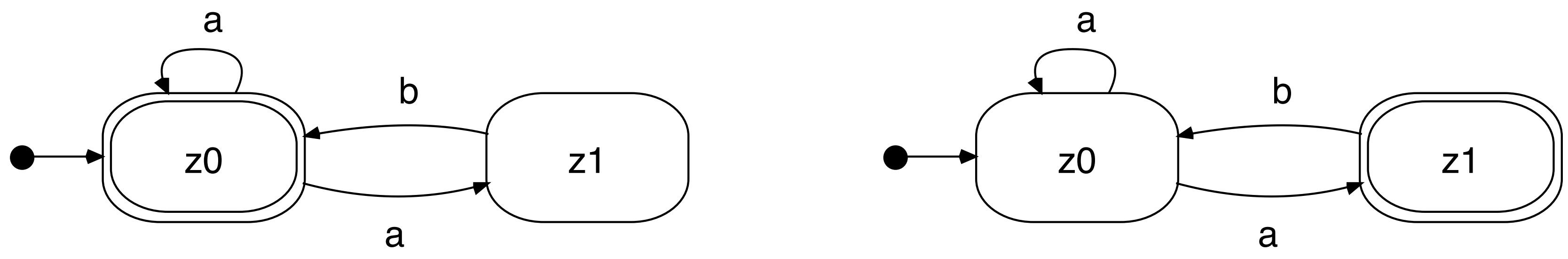
b) Die durch endliche Transitionssysteme $TS = (S, \Sigma, tr, S^0, S^F)$ *akzeptierten* ω*-Sprachen* $L^\omega(TS)$ *sind genau die* ω*-regulären Sprachen über* Σ.



$$L^\omega(TS) = c^*ab(b^+ + bc^*ab)^\omega$$

4. Gib zwei endliche TS für $i = 1, 2$ mit $TS_i = (S, A, tr, S^0, S_i^F)$ und $|S_i^F| = 1$ an, die sich nur in der Wahl ihres Endzustandes unterscheiden und verschiedene ω-Sprachen besitzen.

Lösung: Das Hinzufügen einer a-Schleife zum Beispiel aus der Aufgabenstellung reicht aus:

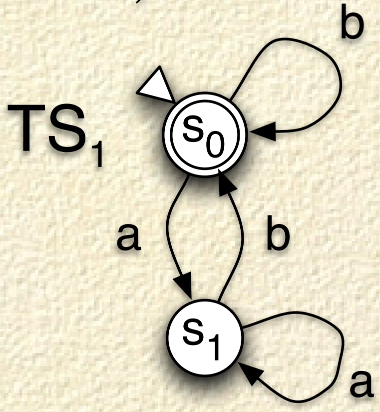


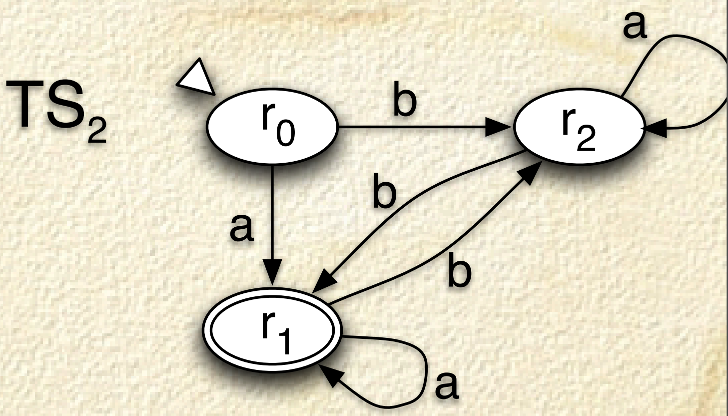
Hier ist $L^\omega(TS_1) = (a + ab)^\omega$ und $L^\omega(TS_2) = a^+(ba^+)^\omega$. Die erste Sprache enthält das Wort a^ω, die zweite nicht.

„Produkt - Automat“

Produkt-Transitionssystem für Durchschnitt

Satz 1.18 *Gegeben seien für $i \in \{1,2\}$ zwei Transitionssysteme $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ mit Endzuständen. Dann können Transitionssyteme TS_3 und TS_4 effektiv konstruiert werden, die den Durchschnitt der akzeptierten Sprachen akzeptieren:*



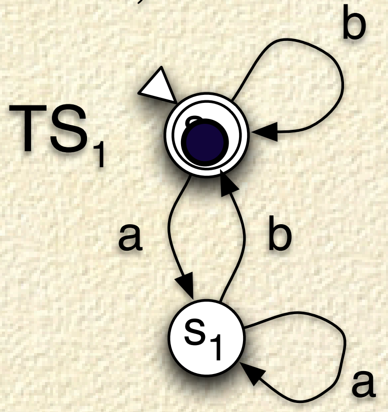


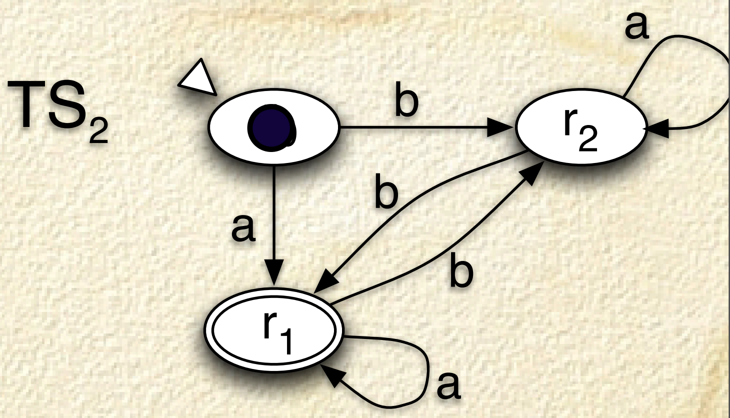
a) $L(TS_3) = L(TS_1) \cap L(TS_2)$

a) Wir definieren TS_3 als das Transitionssystem $TS_1 \otimes_{Sync} TS_2$ mit $Sync = \{(a,a) | a \in A_1 \cap A_2\}$ und $\gamma(a,a) = a$ für $a \in A_1 \cap A_2$ und streichen alle Transitionen $(s_1, r_1) \xrightarrow{a} (s_2, r_2)$ mit $a \notin A_1 \cap A_2$, d.h. bei der Produktdefinition von Definition 1.9 auf Seite 10 kommt nur die Regel $Sy3$ zur Anwendung.

Produkttransitionssystem für Durchschnittbildung

Satz 1.18 *Gegeben seien für $i \in \{1,2\}$ zwei Transitionssysteme $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ mit Endzuständen. Dann können Transitionssyteme TS_3 und TS_4 effektiv konstruiert werden, die den Durchschnitt der akzeptierten Sprachen akzeptieren:*





a) $L(TS_3) = L(TS_1) \cap L(TS_2)$

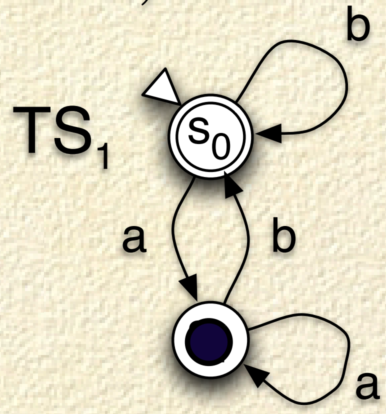
$w \in L(TS_1) \cap L(TS_2)$ $\qquad$ $w = a_1 a_2 \cdots a_n$

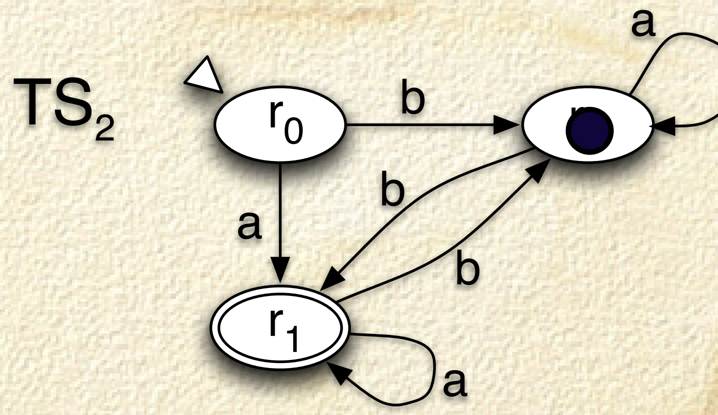
aba

Produkttransitionssystem für Durchschnittbildung

Satz 1.18 *Gegeben seien für* $i \in \{1,2\}$ *zwei Transitionssysteme* $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ *mit Endzuständen. Dann können Transitionssyteme* TS_3 *und* TS_4 *effektiv konstruiert werden, die den Durchschnitt der akzeptierten Sprachen akzeptieren:*

a) $L(TS_3) = L(TS_1) \cap L(TS_2)$





$$s_0 \xrightarrow{a_1}_1 s_1 \xrightarrow{a_2}_1 s_2 \cdots s_{n-1} \xrightarrow{a_n}_1 s_n$$

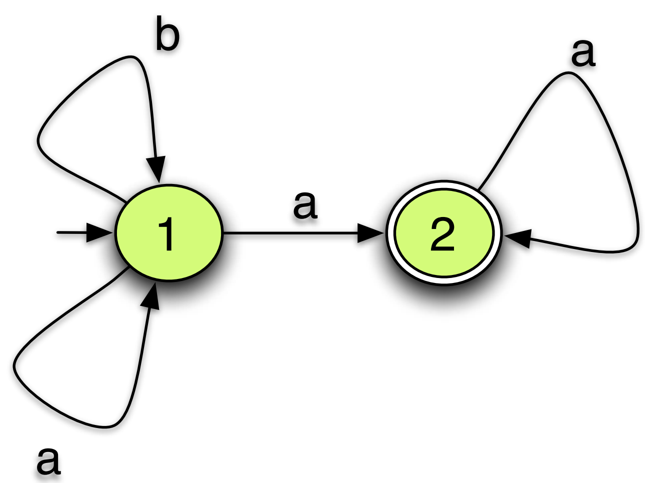
$$r_0 \xrightarrow{a_1}_2 r_1 \xrightarrow{a_2}_2 r_2 \cdots r_{n-1} \xrightarrow{a_n}_2 r_n$$

$$(s_0, r_0) \xrightarrow{a_1}_3 (s_1, r_1) \xrightarrow{a_2}_3 (s_2, r_2) \cdots (s_{n-1}, r_{n-1}) \xrightarrow{a_n}_3 (s_n r_n)$$

Präsenzaufgabe 4.1:

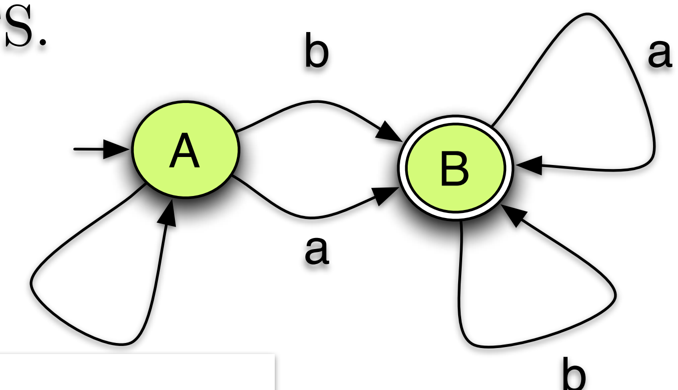
1. Konstruiere ein TS, dass folgende ω-Sprache akzeptiert:

$$(a+b)^* a^\omega$$

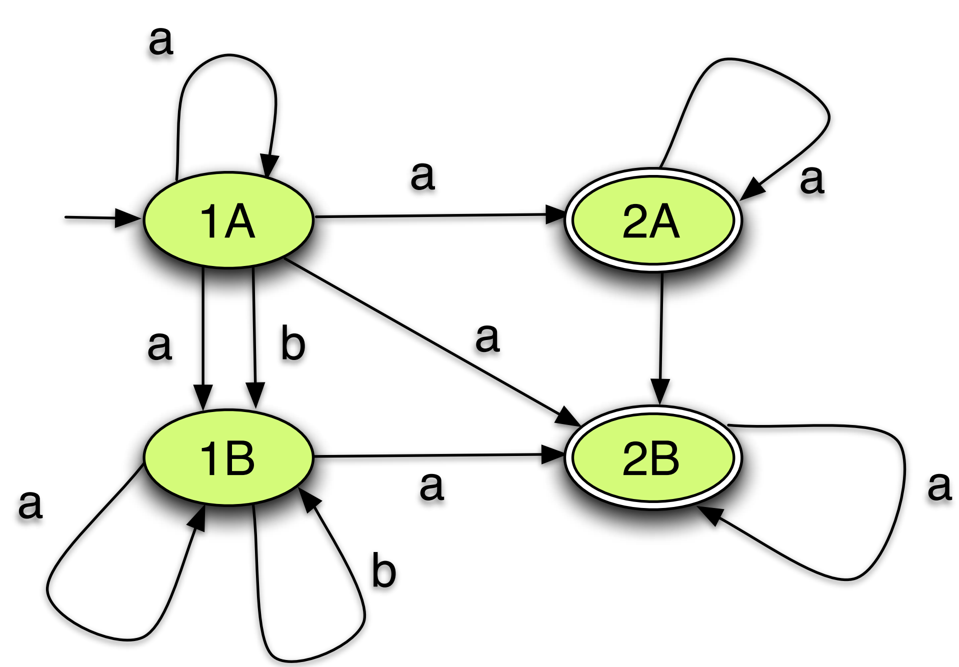


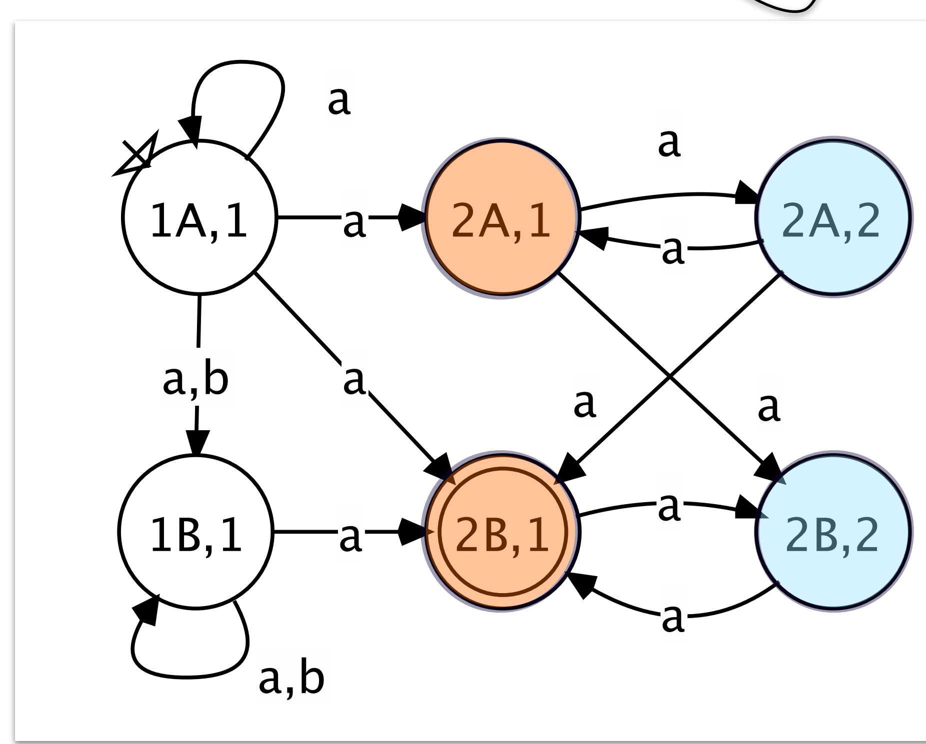
2. Konstruiere ein TS, dass folgende ω-Sprache akzeptiert:

$$a^*(a+b)^\omega$$



3. Konstruiere das Produkt TS_4 aus Satz 1.18 für diese beiden TS.





Produkttransitionssystem für Durchschnittbildung

Satz 1.18 *Gegeben seien für $i \in \{1,2\}$ zwei Transitionssysteme $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ mit Endzuständen. Dann können Transitionssyteme TS_3 und TS_4 effektiv konstruiert werden, die den Durchschnitt der akzeptierten Sprachen akzeptieren:*

a) $L(TS_3) = L(TS_1) \cap L(TS_2)$

b) $L^\omega(TS_4) = L^\omega(TS_1) \cap L^\omega(TS_2)$.

$w \in L(TS_1) \cap L(TS_2)$ $\quad w = a_1 a_2 \cdots a_n$

$$s_0 \xrightarrow{a_1}_1 s_1 \xrightarrow{a_2}_1 s_2 \cdots s_{n-1} \xrightarrow{a_n}_1 s_n \longrightarrow_1$$

$$r_0 \xrightarrow{a_1}_2 r_1 \xrightarrow{a_2}_2 r_2 \cdots r_{n-1} \xrightarrow{a_n}_2 r_n \longrightarrow_2$$

$$(s_0, r_0) \xrightarrow{a_1}_3 (s_1, r_1) \xrightarrow{a_2}_3 (s_2, r_2) \cdots (s_{n-1}, r_{n-1}) \xrightarrow{a_n}_3$$

$$(s_n r_n)$$

$$\longrightarrow_3$$

Produkttransitionssystem für Durchschnittbildung

Satz 1.18 *Gegeben seien für $i \in \{1, 2\}$ zwei Transitionssysteme $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ mit Endzuständen. Dann können Transitionssyteme TS_3 und TS_4 effektiv konstruiert werden, die den Durchschnitt der akzeptierten Sprachen akzeptieren:*

a) $L(TS_3) = L(TS_1) \cap L(TS_2)$ *b)* $L^\omega(TS_4) = L^\omega(TS_1) \cap L^\omega(TS_2)$.

$w \in L(TS_1) \cap L(TS_2)$ $\qquad w = a_1 a_2 \cdots a_n$

$$s_0 \xrightarrow{a_1}_1 s_1 \xrightarrow{a_2}_1 s_2 \cdots s_{n-1} \xrightarrow{a_n}_1 s_n \longrightarrow_1$$

$$r_0 \xrightarrow{a_1}_2 r_1 \xrightarrow{a_2}_2 r_2 \cdots r_{n-1} \xrightarrow{a_n}_2 r_n \longrightarrow_2$$

$$(s_0, r_0) \xrightarrow{a_1}_3 (s_1, r_1) \xrightarrow{a_2}_3 (s_2, r_2) \cdots (s_{n-1}, r_{n-1}) \xrightarrow{a_n}_3 (s_n r_n) \longrightarrow_3$$

Produkttransitionssystem für Durchschnittbildung

Satz 1.18 *Gegeben seien für $i \in \{1,2\}$ zwei Transitionssysteme $TS_i = (S_i, A_i, tr_i, S_i^0, S_i^F)$ mit Endzuständen. Dann können Transitionssyteme TS_3 und TS_4 effektiv konstruiert werden, die den Durchschnitt der akzeptierten Sprachen akzeptieren:*

a) $L(TS_3) = L(TS_1) \cap L(TS_2)$ *b)* $L^\omega(TS_4) = L^\omega(TS_1) \cap L^\omega(TS_2)$.

$w \in L(TS_1) \cap L(TS_2)$ $\quad w = a_1 a_2 \cdots a_n$

$$\triangle s_0 \xrightarrow{a_1}_1 s_1 \xrightarrow{a_2}_1 s_2 \cdots s_{n-1} \xrightarrow{a_n}_1 s_n \longrightarrow_1$$

$$\triangle r_0 \xrightarrow{a_1}_2 r_1 \xrightarrow{a_2}_2 r_2 \cdots r_{n-1} \xrightarrow{a_n}_2 r_n \longrightarrow_2$$

$$\triangle (s_0, r_0) \xrightarrow{a_1}_3 (s_1, r_1) \xrightarrow{a_2}_3 (s_2, r_2) \cdots (s_{n-1}, r_{n-1}) \xrightarrow{a_n}_3$$

$$(s_n r_n)$$

$$\longrightarrow_3$$

Formal definieren wir $TS_4 = (S_4, A_1 \cap A_2, tr_4, S_4^0, S_4^F)$ durch

a) $S_4 = \{(s, r, q) | (s, r) \in S_3, q \in \{1, 2\}\}$

b) $(s, r, q) \xrightarrow{a}_4 (s', r', q') :\Longleftrightarrow (s, r) \xrightarrow{a}_3 (s', r') \ \wedge \ q' := \begin{cases} 2 & \text{falls } q = 1 \wedge s \in S_q^F \\ 1 & \text{falls } q = 2 \wedge r \in S_q^F \\ q & \text{sonst} \end{cases}$

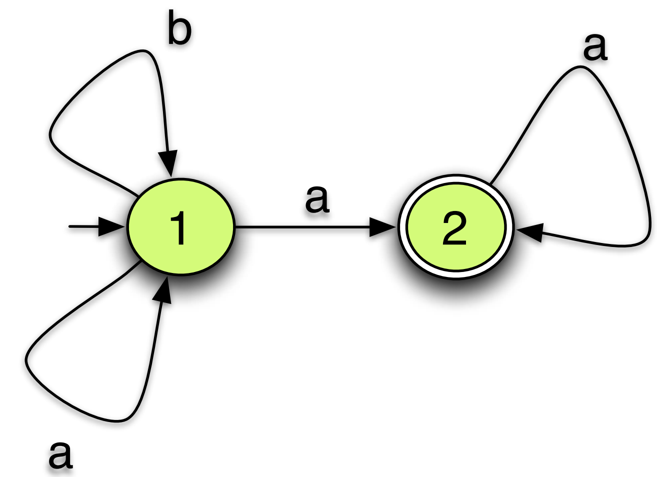
c) $S_4^0 := \{(s, r, 1) | (s, r) \in S_3^0 = S_1^0 \times S_2^0\}$

d) $S_4^F := \{(s, r, 1) | s \in S_1^F\}$

Präsenzaufgabe 4.1:

1. Konstruiere ein TS, dass folgende ω-Sprache akzeptiert:

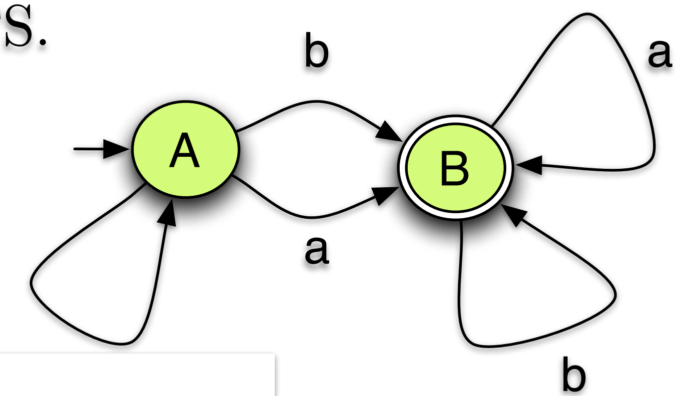
$$(a+b)^*a^\omega$$

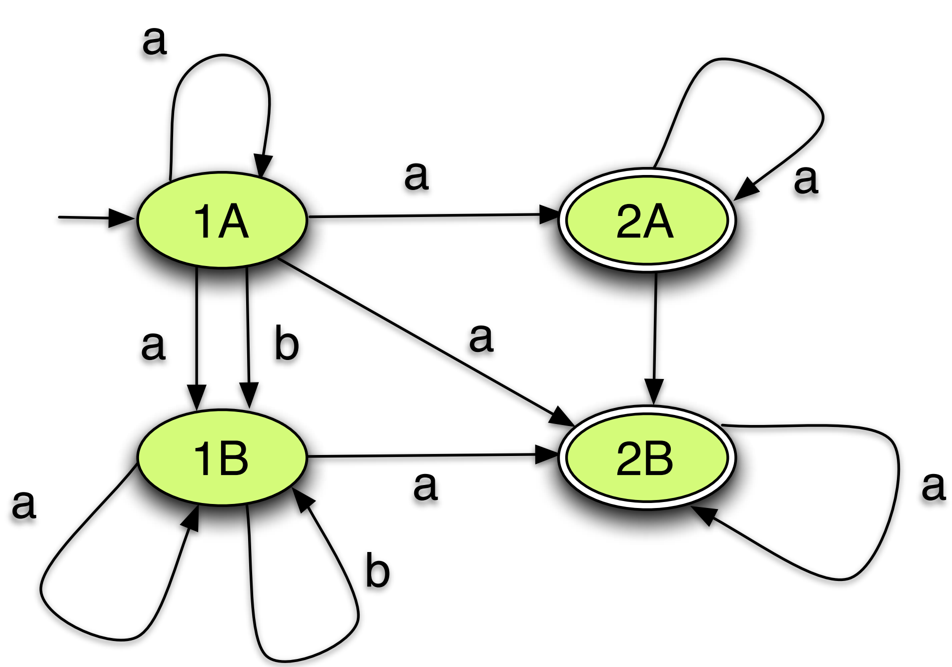


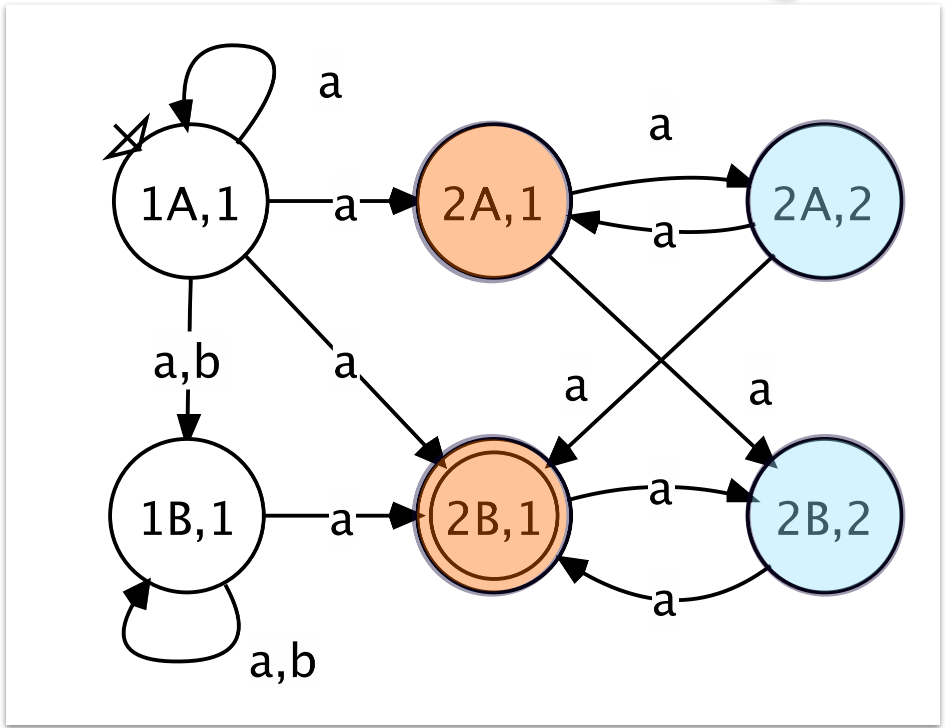
2. Konstruiere ein TS, dass folgende ω-Sprache akzeptiert:

$$a^*(a+b)^\omega$$

3. Konstruiere das Produkt TS_4 aus Satz 1.18 für diese beiden TS.







Model-Checking

Verifikation eines Systems

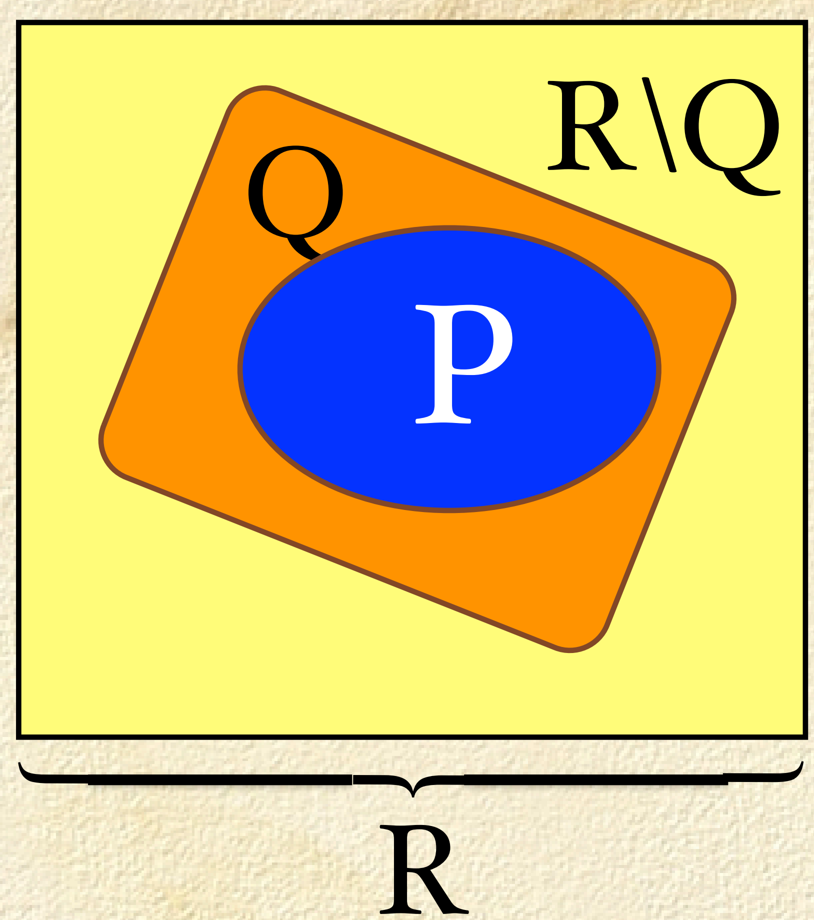
System-Verhalten → *System-Spezifikation*

$$L(TS_{sys}) \subseteq L(TS_{spec})$$

$$L^{\omega}(TS_{sys}) \subseteq L^{\omega}(TS_{spec}).$$

(temporal-)logische Formel f_{spec}

$$L(TS_{sys}) \;\subseteq\; L(TS_{spec})$$



$$P \subseteq Q \;\Leftrightarrow\; P \cap (R \backslash Q) = \emptyset$$

$$L(TS_{sys}) \cap (A^* \backslash L(TS_{spec})) = \emptyset$$

$$L^\omega(TS_{sys}) \cap (A^\omega \backslash L^\omega(TS_{spec})) = \emptyset$$

$$\underbrace{f_{spec}}_{\neg f_{spec}}$$

Klausurtermine (ohne Gewähr)

- 1. Klausur: 01.03.2011, 9.30 - 11.30, Audi 1 (Achtung! Zeit geändert!)
- 2. Klausur: 01.04.2011, 9.30 - 11.30, ESA B (Achtung! Zeit geändert!)

Zur Vorbereitung auf die Klausuren werden Repetitorien angeboten.