Baumautomaten: Abschlusseigenschaften und Algorithmen

Edgar Schmidt

Seminarvortrag bei Prof. Dr. Wim Martens

Universität Bayreuth

30. April, 2014

1. Abschlusseigenschaften

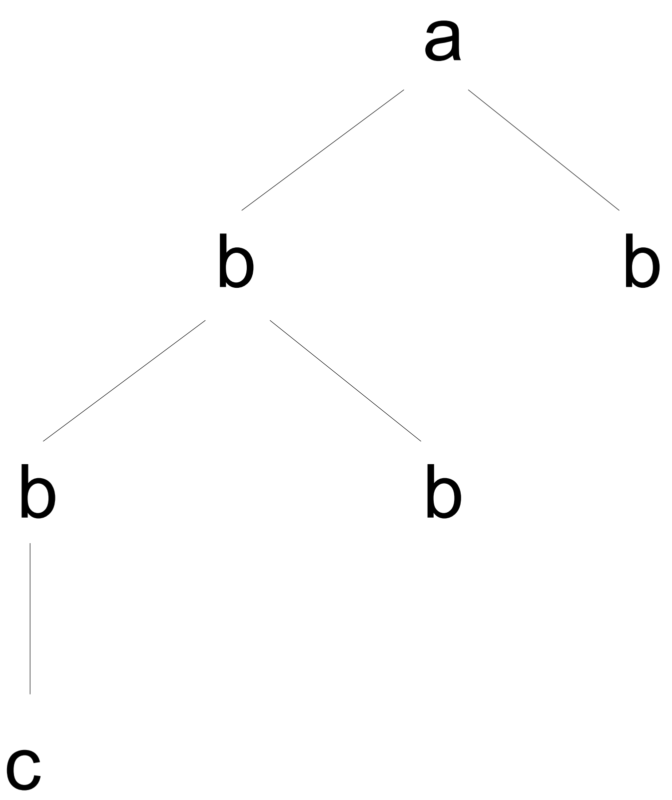
2. Algorithmen

Präposition: Binär verzweigte Bäume

Wir werden im folgenden nur binär beschränkt verzweigte Bäume betrachten:

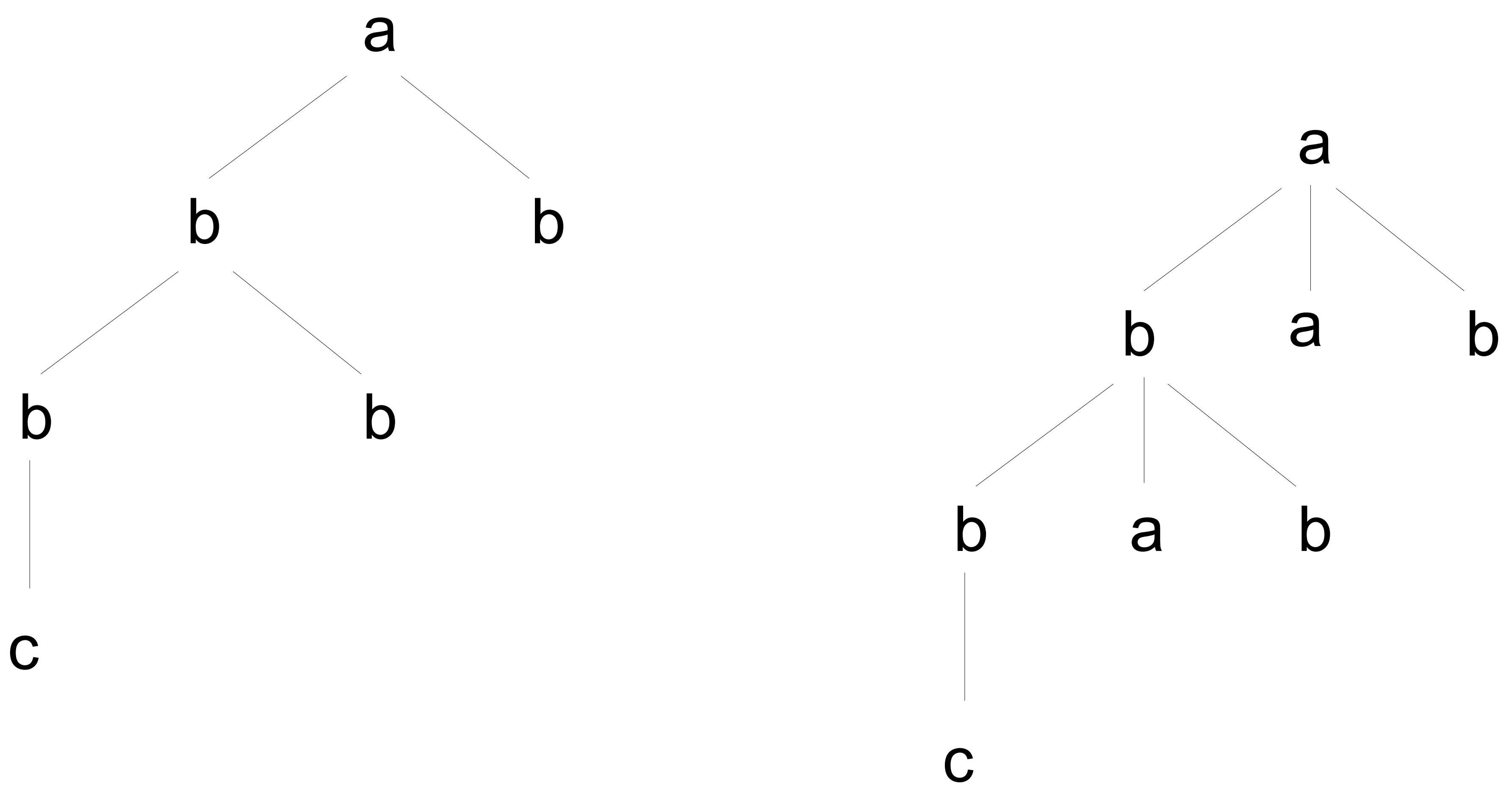
Präposition: Binär verzweigte Bäume

Wir werden im folgenden nur binär beschränkt verzweigte Bäume betrachten:

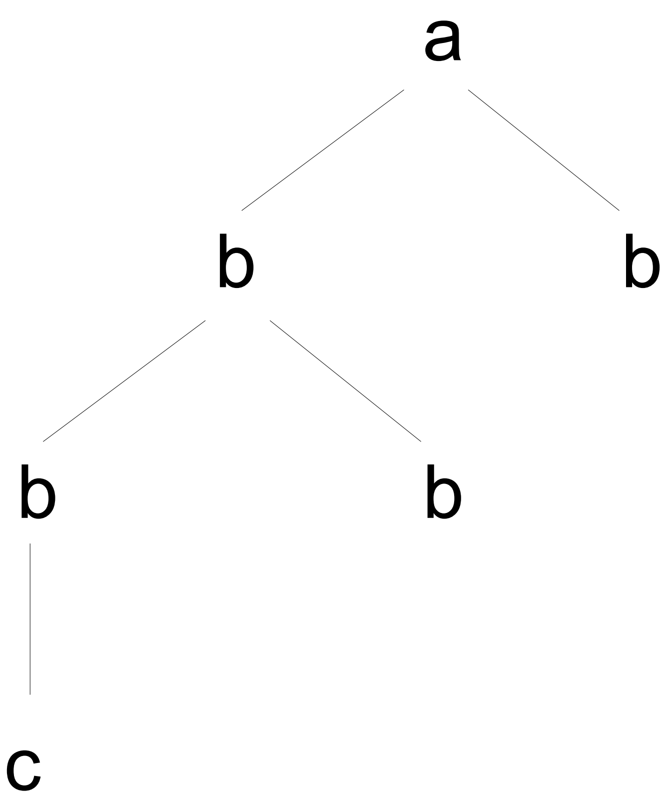


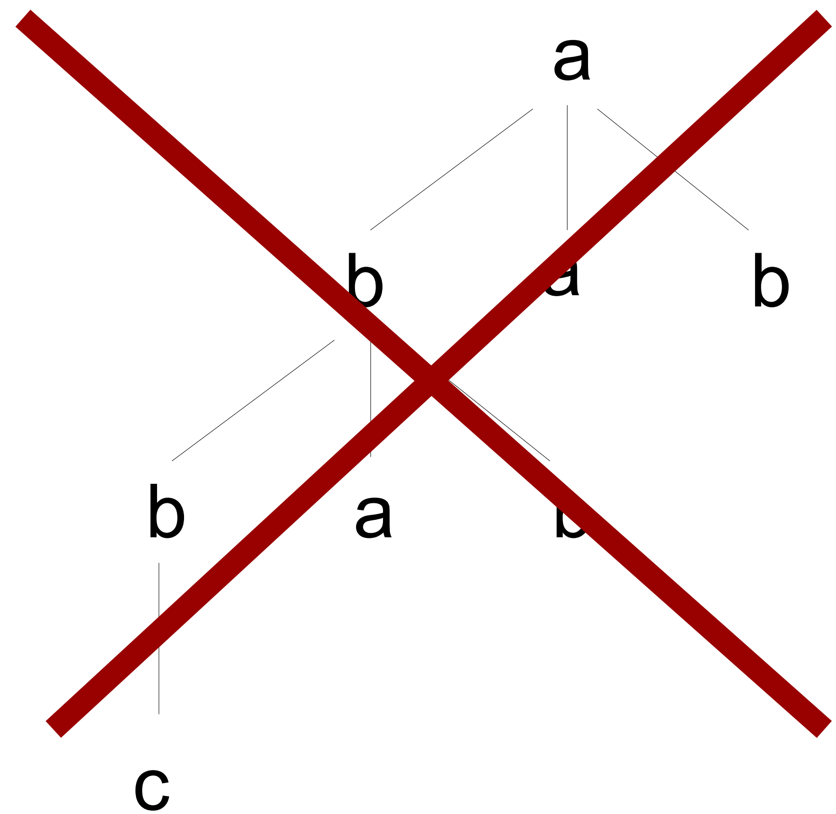
Präposition: Binär verzweigte Bäume

Wir werden im folgenden nur binär beschränkt verzweigte Bäume betrachten:



Wir werden im folgenden nur binär beschränkt verzweigte Bäume betrachten:





Theorem 1:
Die Klasse von regulären Baumsprachen ist abgeschlossen unter:

Theorem 1:
Die Klasse von regulären Baumsprachen ist abgeschlossen unter:

Vereinigung

Theorem 1:
Die Klasse von regulären Baumsprachen ist abgeschlossen unter:

Vereinigung

Komplement

Theorem 1:
Die Klasse von regulären Baumsprachen ist abgeschlossen unter:

Vereinigung

Komplement

Schnitt

Gegeben: Zwei Baumautomaten A_1 und A_2

Gesucht: $L(A) = L(A_1) \cup L(A_2)$

Gegeben: Zwei Baumautomaten A_1 und A_2

Gesucht: $L(A) = L(A_1) \cup L(A_2)$

Algorithmus:

Bildung des Produktautomaten, der akzeptiert, wenn mindestens ein Automat akzeptiert

Abschluss unter Vereinigung

Automat A_1 sei wie folgt definiert:

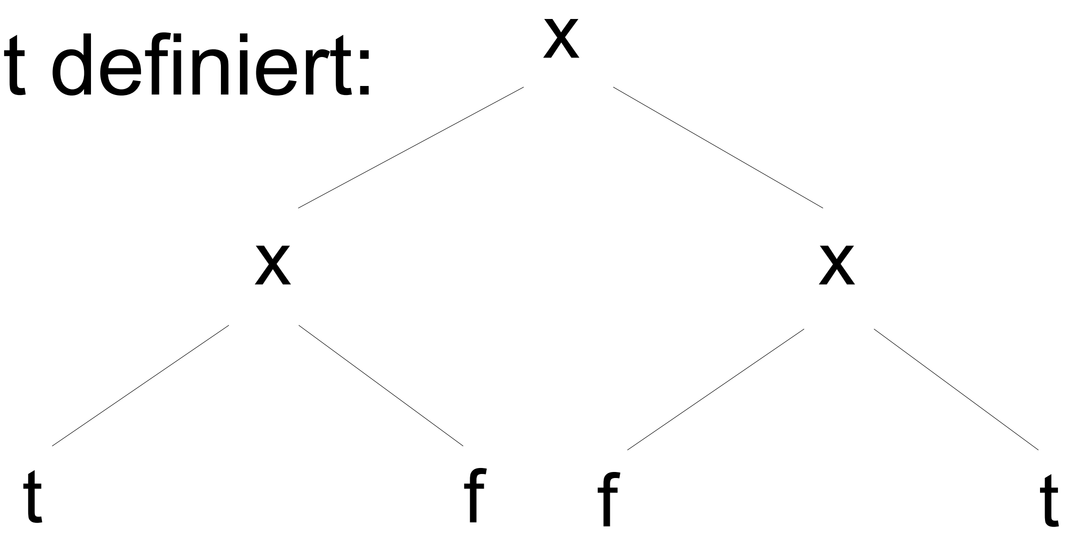
Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t \qquad \varepsilon \xrightarrow{f} q_f$$

$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$

Automat A_1 sei wie folgt definiert:

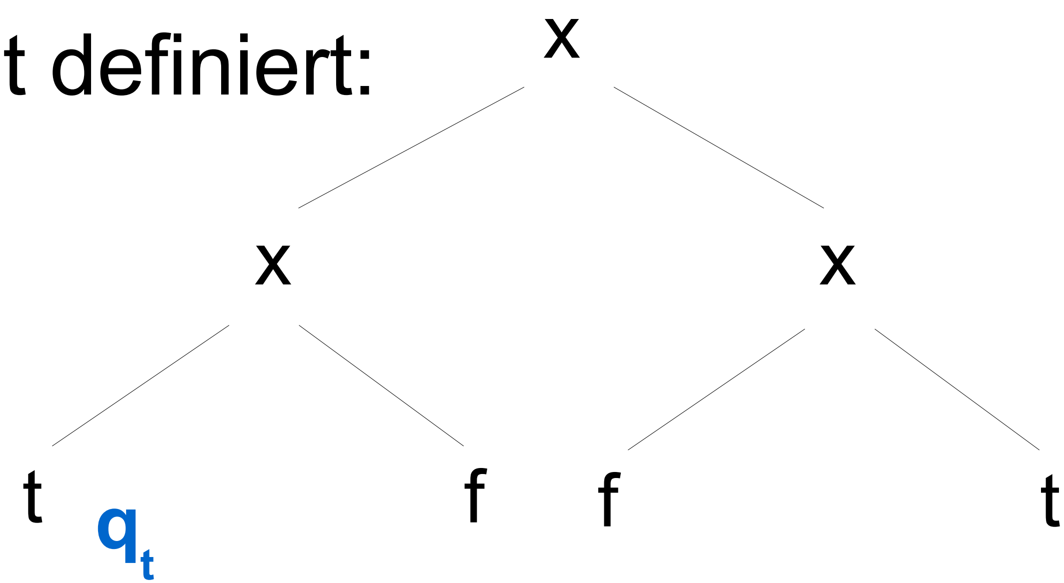
$$\varepsilon \xrightarrow{t} q_t \qquad (q_t,q_t) \xrightarrow{x} q_t$$
$$\varepsilon \xrightarrow{f} q_f \qquad (q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t \qquad \varepsilon \xrightarrow{f} q_f$$

$$(q_t, q_t) \xrightarrow{x} q_t$$
$$(q_t, q_f) \xrightarrow{x} q_t$$
$$(q_f, q_t) \xrightarrow{x} q_t$$
$$(q_f, q_f) \xrightarrow{x} q_f$$



Automat A_1 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t$

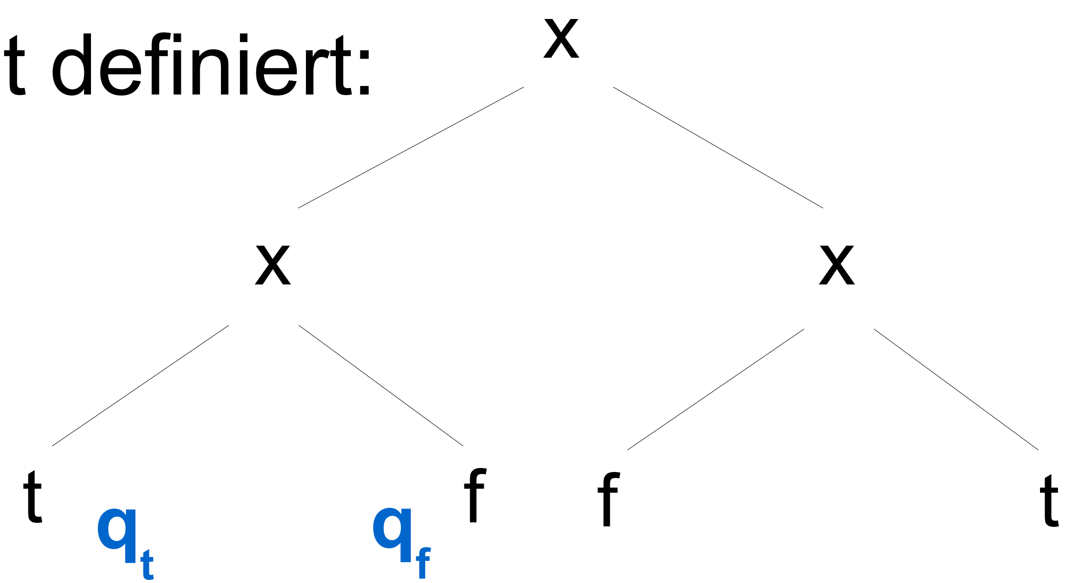
$\varepsilon \xrightarrow{f} q_f$

$(q_t,q_t) \xrightarrow{x} q_t$

$(q_t,q_f) \xrightarrow{x} q_t$

$(q_f,q_t) \xrightarrow{x} q_t$

$(q_f,q_f) \xrightarrow{x} q_f$

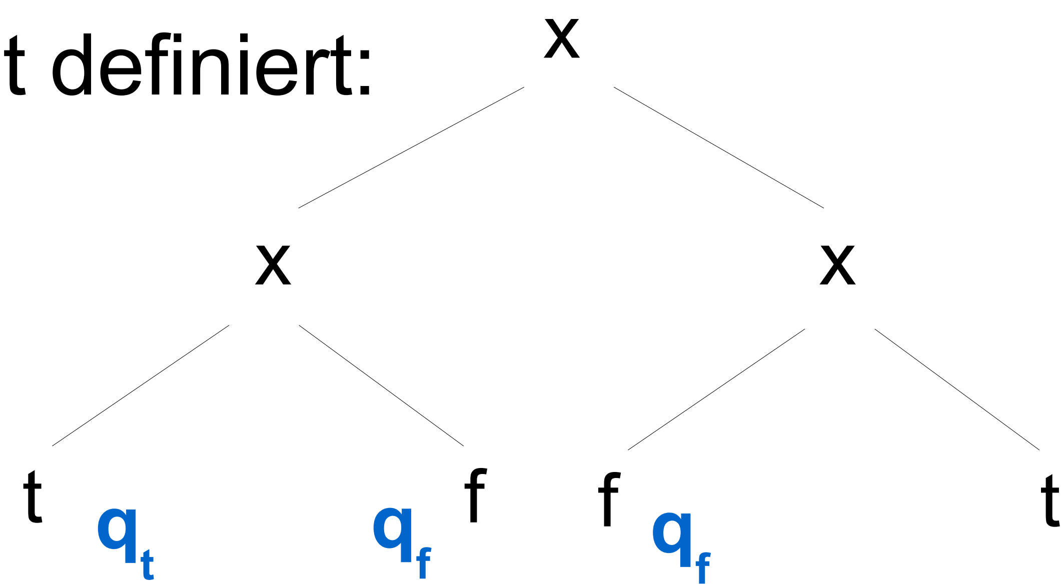


Abschluss unter Vereinigung

Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t$$
$$\varepsilon \xrightarrow{f} q_f$$

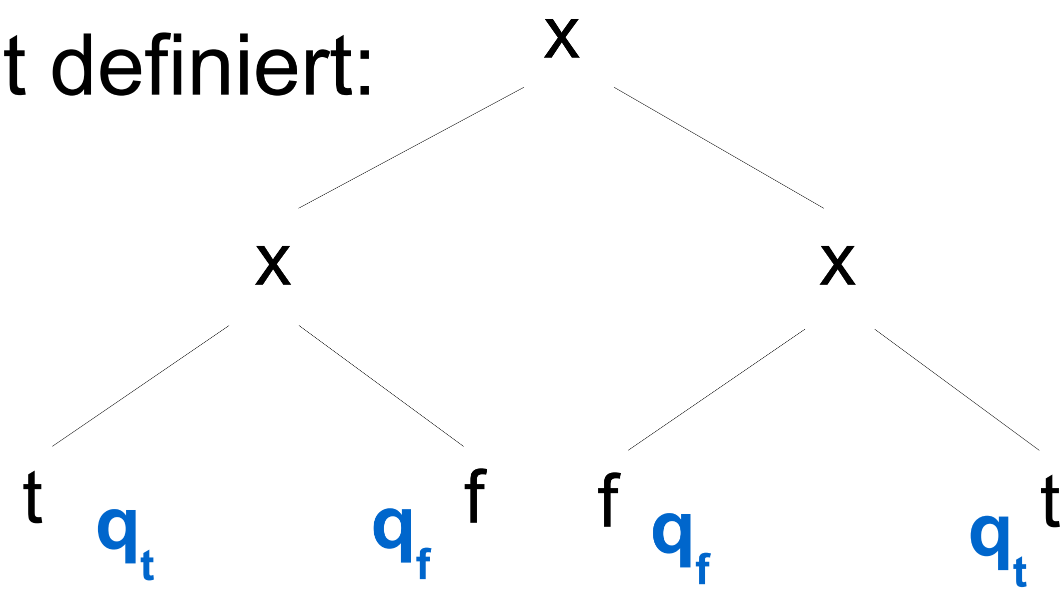
$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t$$
$$\varepsilon \xrightarrow{f} q_f$$

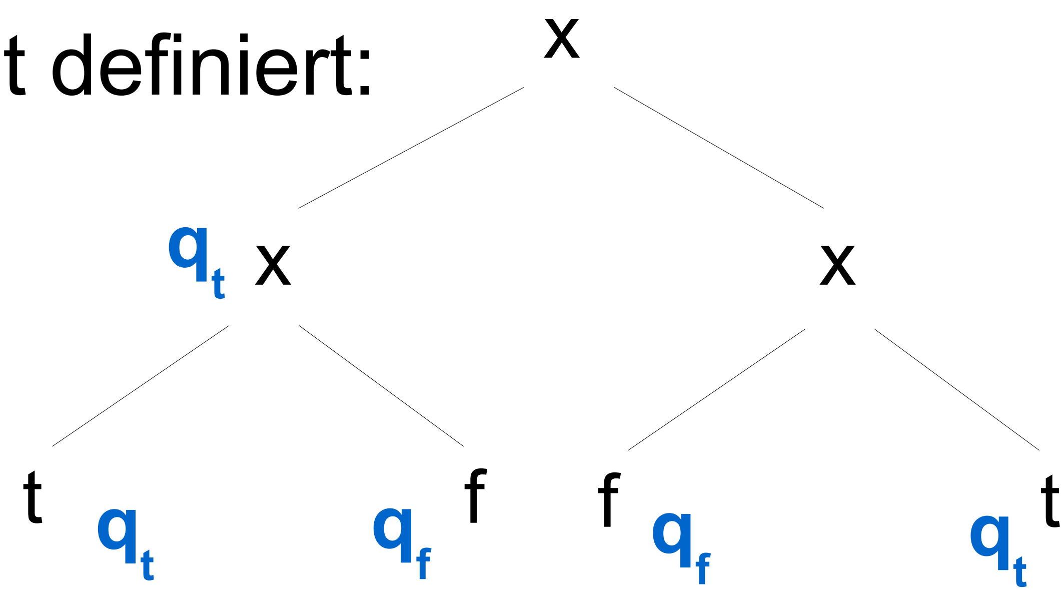
$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t$$
$$\varepsilon \xrightarrow{f} q_f$$

$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$

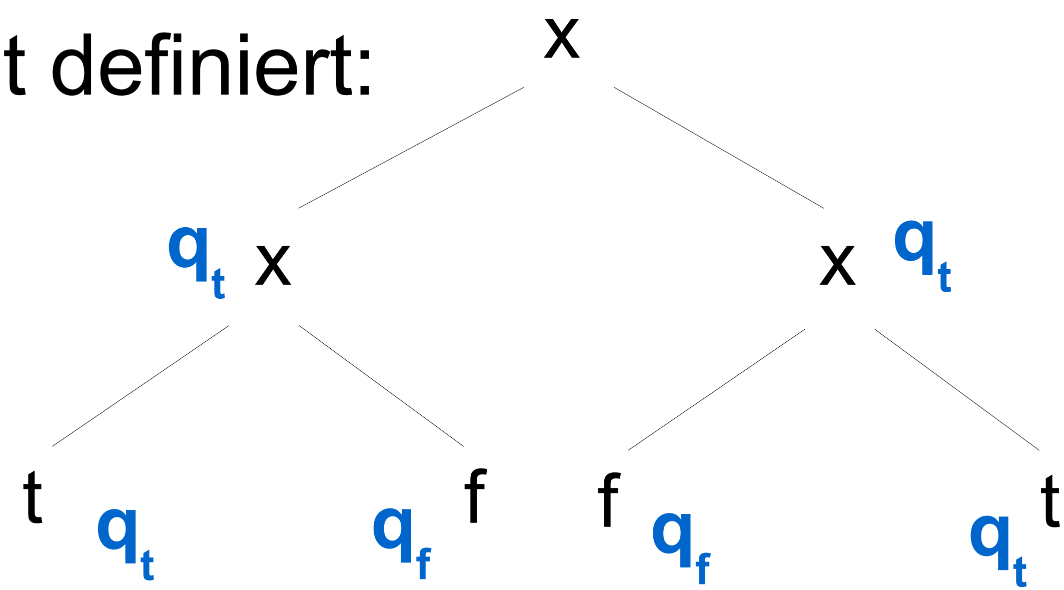


Abschluss unter Vereinigung

Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t$$
$$\varepsilon \xrightarrow{f} q_f$$

$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t$$

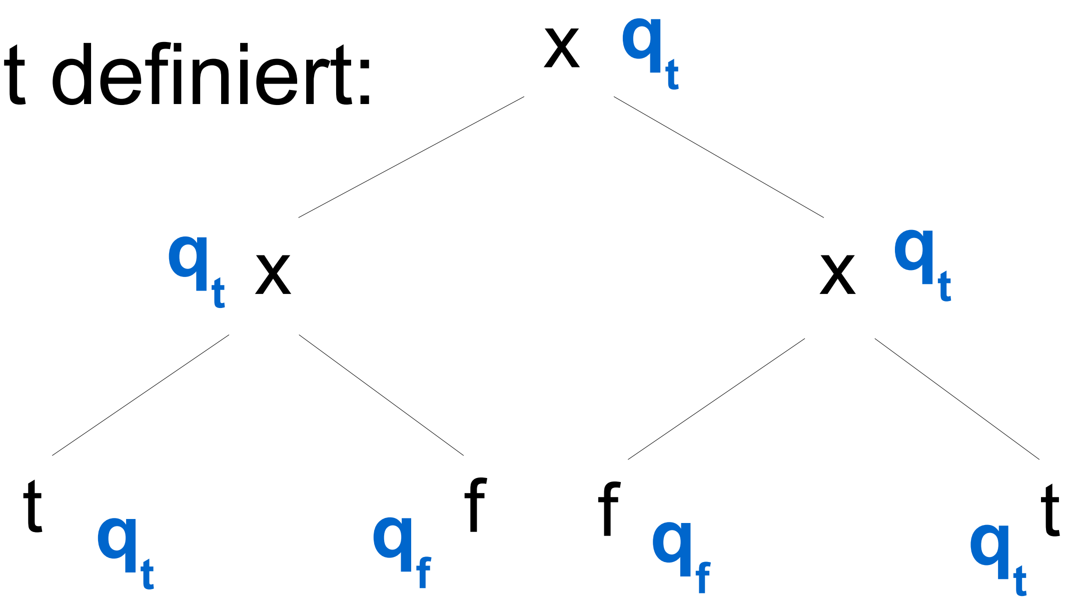
$$\varepsilon \xrightarrow{f} q_f$$

$$(q_t,q_t) \xrightarrow{x} q_t$$

$$(q_t,q_f) \xrightarrow{x} q_t$$

$$(q_f,q_t) \xrightarrow{x} q_t$$

$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_1 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t$

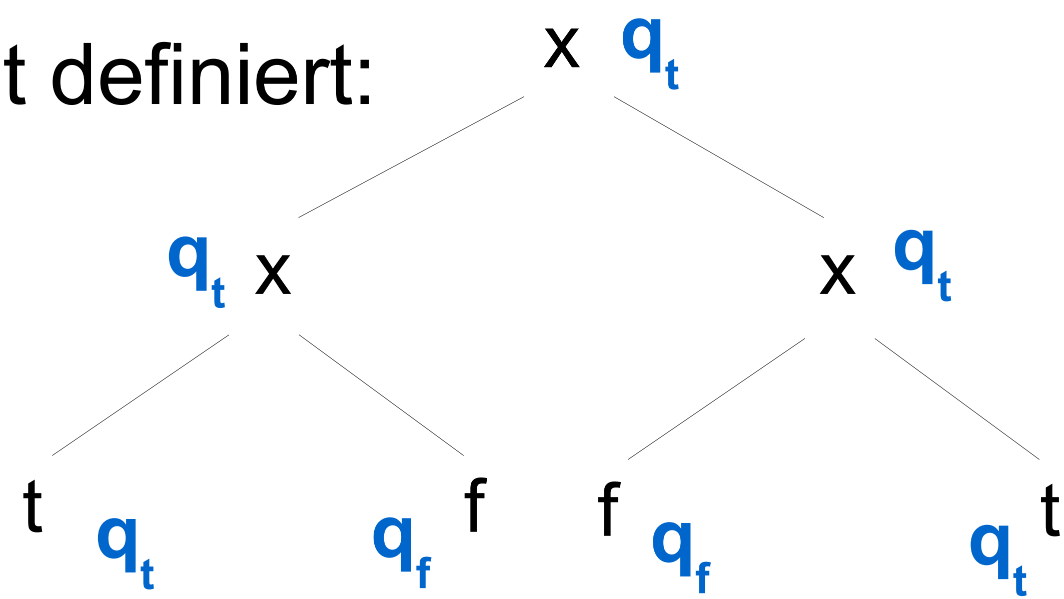
$\varepsilon \xrightarrow{f} q_f$

$(q_t,q_t) \xrightarrow{x} q_t$

$(q_t,q_f) \xrightarrow{x} q_t$

$(q_f,q_t) \xrightarrow{x} q_t$

$(q_f,q_f) \xrightarrow{x} q_f$



Automat A_2 sei wie folgt definiert:

Automat A_1 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t$

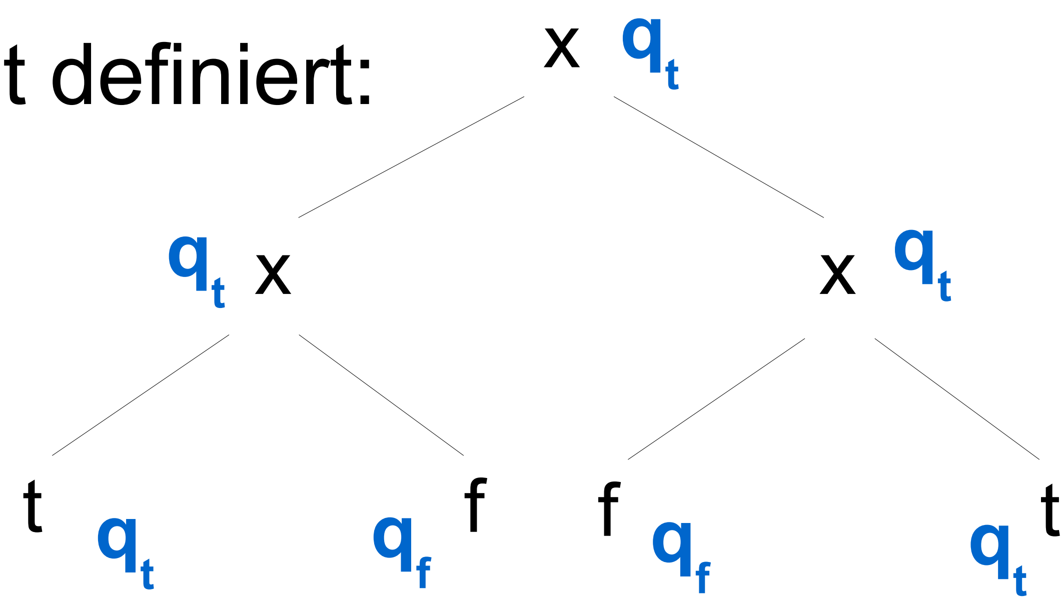
$\varepsilon \xrightarrow{f} q_f$

$(q_t,q_t) \xrightarrow{x} q_t$

$(q_t,q_f) \xrightarrow{x} q_t$

$(q_f,q_t) \xrightarrow{x} q_t$

$(q_f,q_f) \xrightarrow{x} q_f$



Automat A_2 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t'$

$\varepsilon \xrightarrow{f} q_f'$

$(q_t',q_t') \xrightarrow{x} q_t'$

$(q_t',q_f') \xrightarrow{x} q_f'$

$(q_f',q_t') \xrightarrow{x} q_f'$

$(q_f',q_f') \xrightarrow{x} q_f'$

Automat A_1 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t$

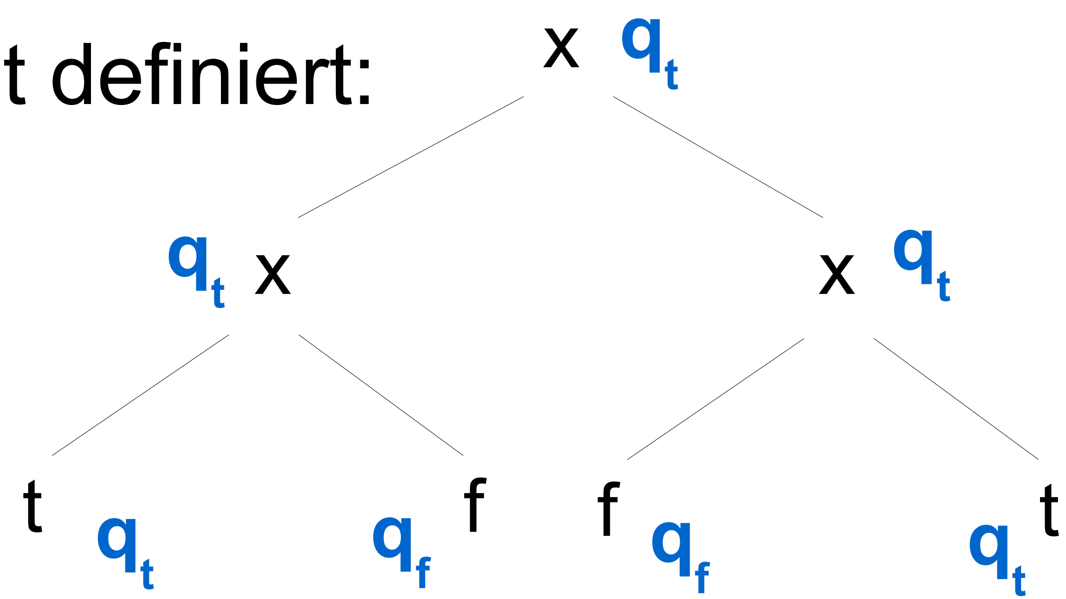
$\varepsilon \xrightarrow{f} q_f$

$(q_t,q_t) \xrightarrow{x} q_t$

$(q_t,q_f) \xrightarrow{x} q_t$

$(q_f,q_t) \xrightarrow{x} q_t$

$(q_f,q_f) \xrightarrow{x} q_f$



Automat A_2 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t'$

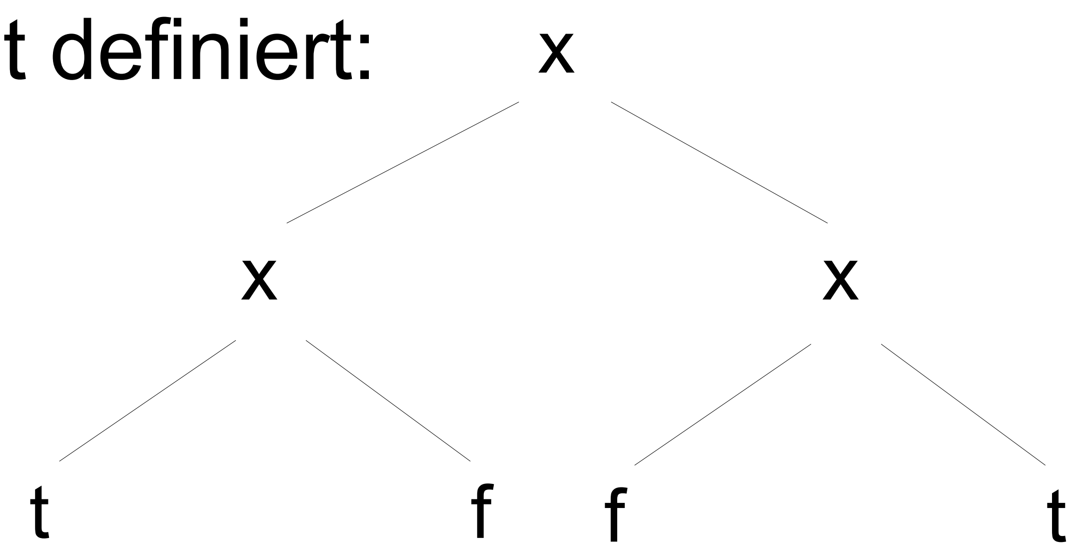
$\varepsilon \xrightarrow{f} q_f'$

$(q_t',q_t') \xrightarrow{x} q_t'$

$(q_t',q_f') \xrightarrow{x} q_f'$

$(q_f',q_t') \xrightarrow{x} q_f'$

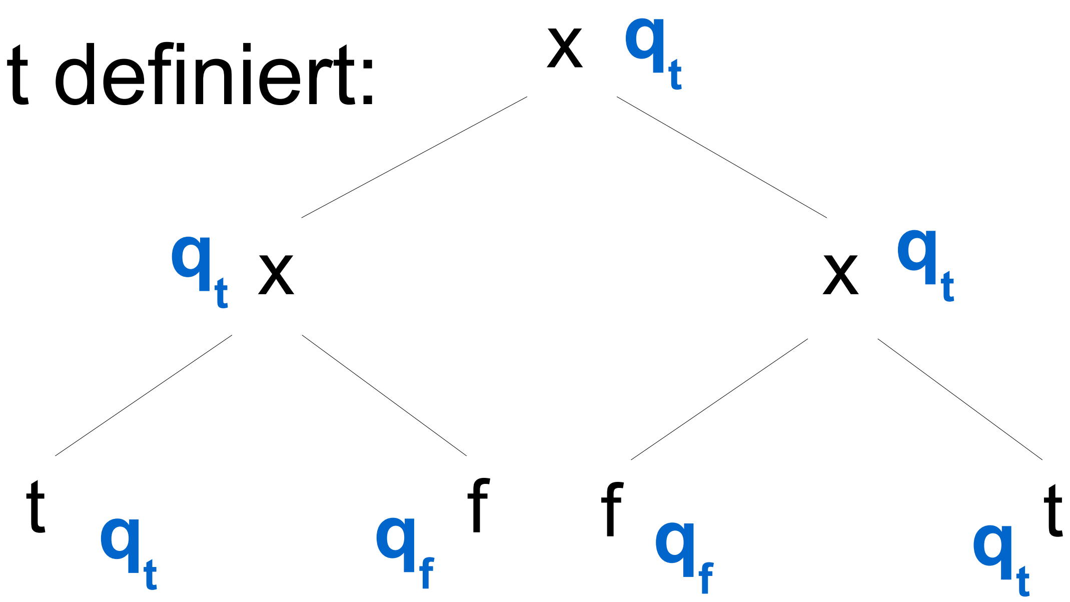
$(q_f',q_f') \xrightarrow{x} q_f'$



Abschluss unter Vereinigung

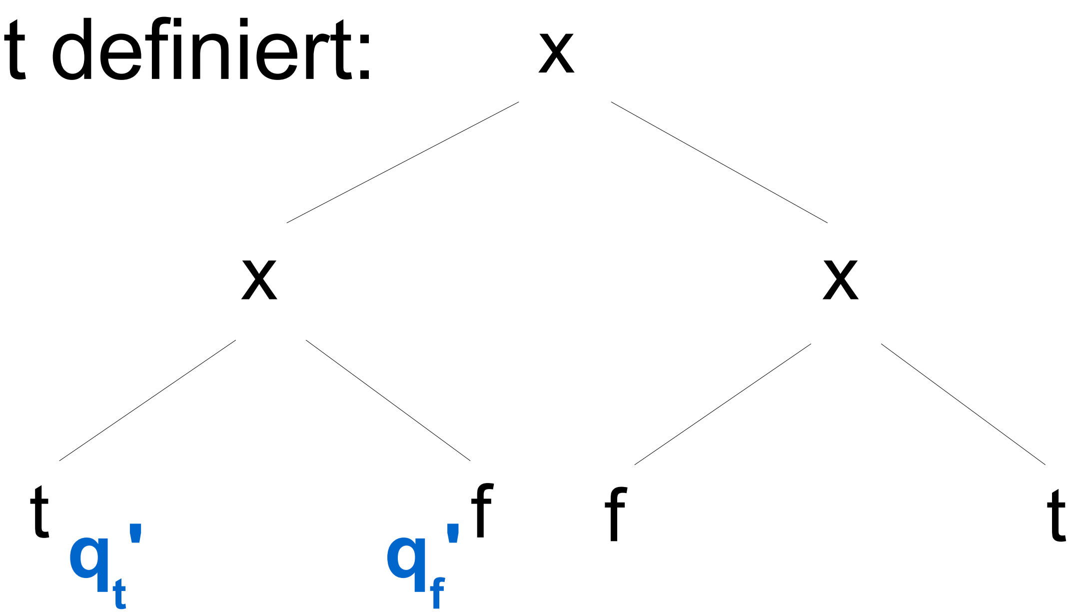
Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} \color{green}{q_t} \qquad (q_t,q_t) \xrightarrow{x} \color{green}{q_t}$$
$$\varepsilon \xrightarrow{f} q_f \qquad (q_t,q_f) \xrightarrow{x} \color{green}{q_t}$$
$$(q_f,q_t) \xrightarrow{x} \color{green}{q_t}$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_2 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} \color{green}{q_t'} \qquad (q_t',q_t') \xrightarrow{x} \color{green}{q_t'}$$
$$\varepsilon \xrightarrow{f} q_f' \qquad (q_t',q_f') \xrightarrow{x} q_f'$$
$$(q_f',q_t') \xrightarrow{x} q_f'$$
$$(q_f',q_f') \xrightarrow{x} q_f'$$



Abschluss unter Vereinigung

Automat A_1 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t$

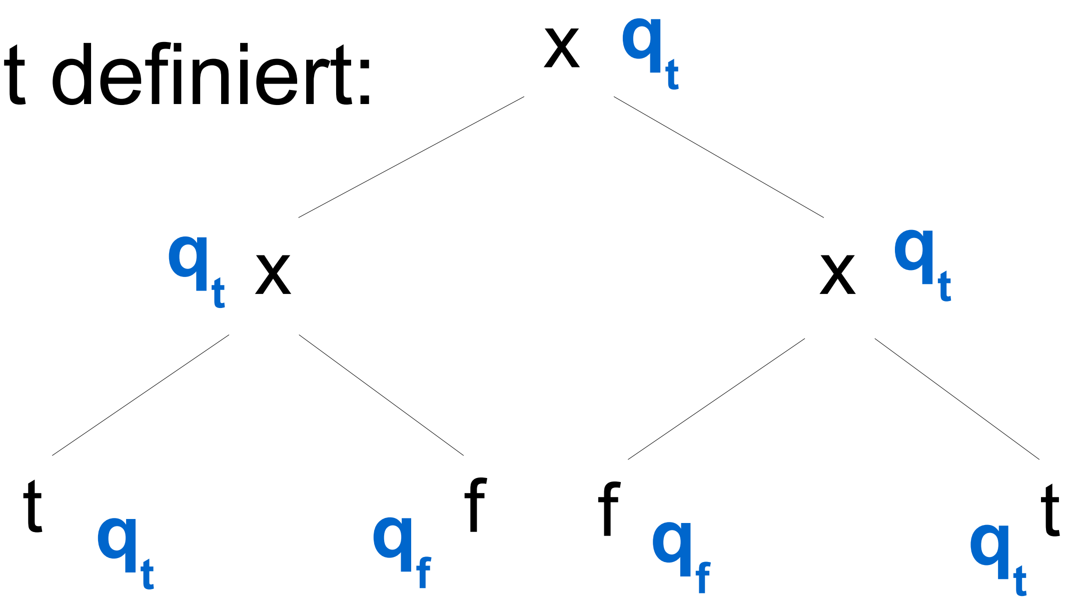
$\varepsilon \xrightarrow{f} q_f$

$(q_t,q_t) \xrightarrow{x} q_t$

$(q_t,q_f) \xrightarrow{x} q_t$

$(q_f,q_t) \xrightarrow{x} q_t$

$(q_f,q_f) \xrightarrow{x} q_f$



Automat A_2 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t'$

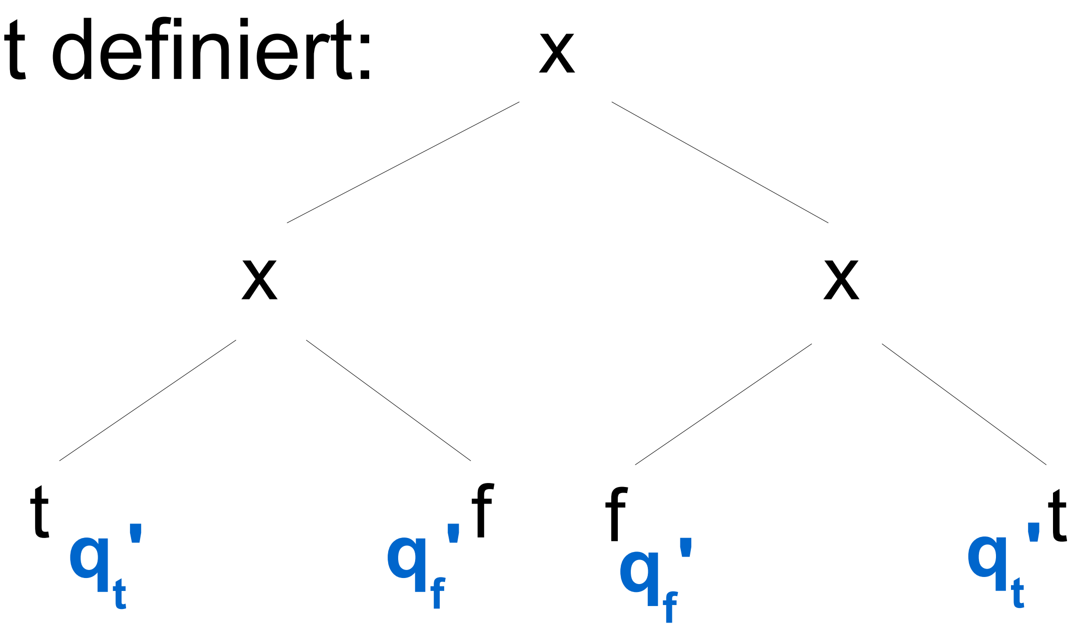
$\varepsilon \xrightarrow{f} q_f'$

$(q_t',q_t') \xrightarrow{x} q_t'$

$(q_t',q_f') \xrightarrow{x} q_f'$

$(q_f',q_t') \xrightarrow{x} q_f'$

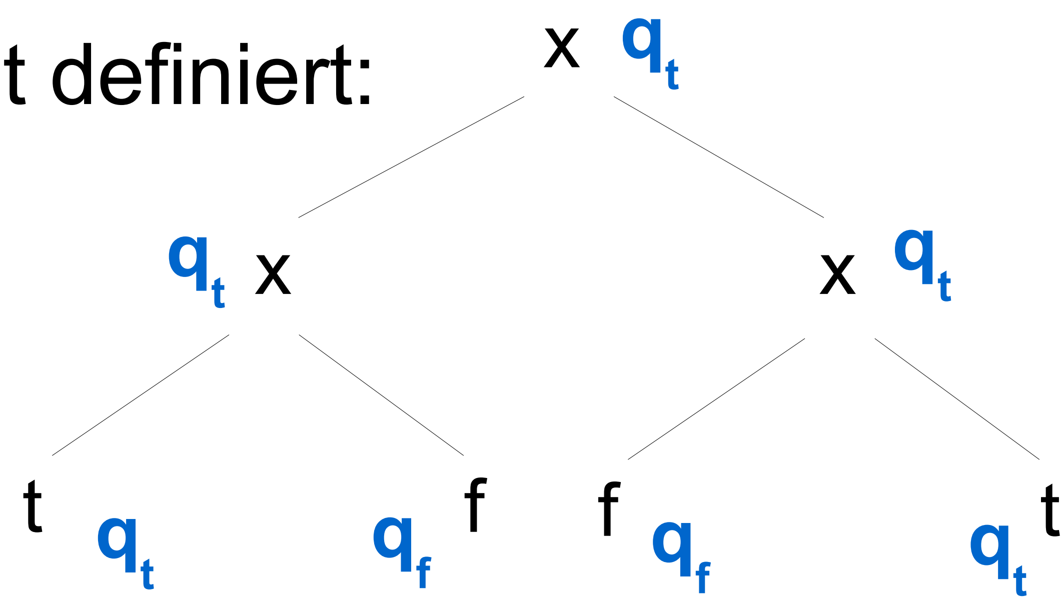
$(q_f',q_f') \xrightarrow{x} q_f'$



Automat A_1 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t \qquad \varepsilon \xrightarrow{f} q_f$$

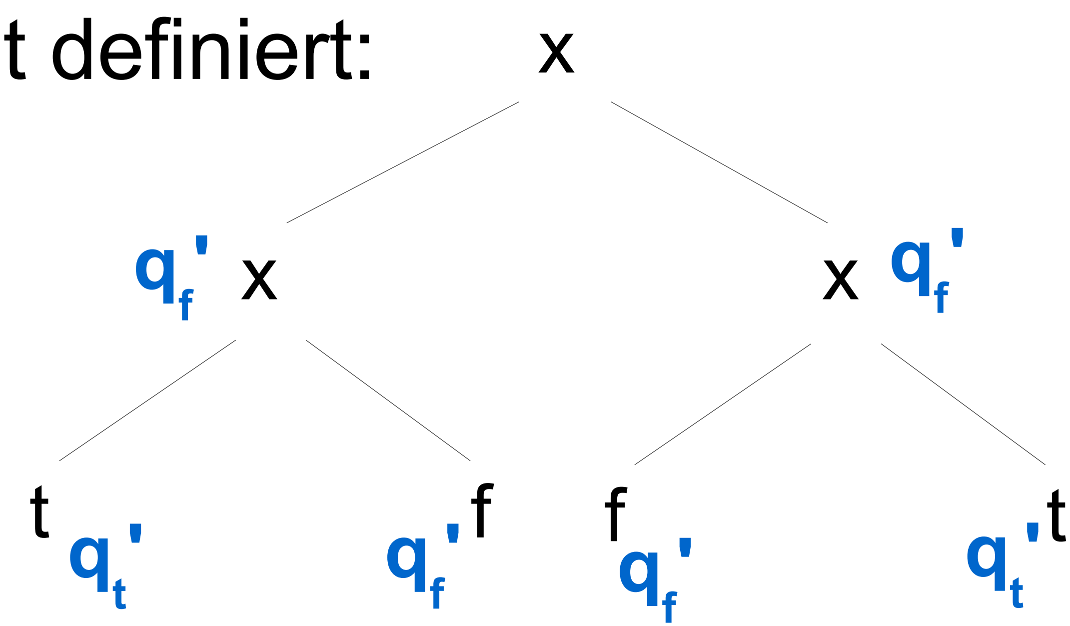
$$(q_t,q_t) \xrightarrow{x} q_t \qquad (q_t,q_f) \xrightarrow{x} q_t \qquad (q_f,q_t) \xrightarrow{x} q_t \qquad (q_f,q_f) \xrightarrow{x} q_f$$



Automat A_2 sei wie folgt definiert:

$$\varepsilon \xrightarrow{t} q_t' \qquad \varepsilon \xrightarrow{f} q_f'$$

$$(q_t',q_t') \xrightarrow{x} q_t' \qquad (q_t',q_f') \xrightarrow{x} q_f' \qquad (q_f',q_t') \xrightarrow{x} q_f' \qquad (q_f',q_f') \xrightarrow{x} q_f'$$



Automat A_1 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t$

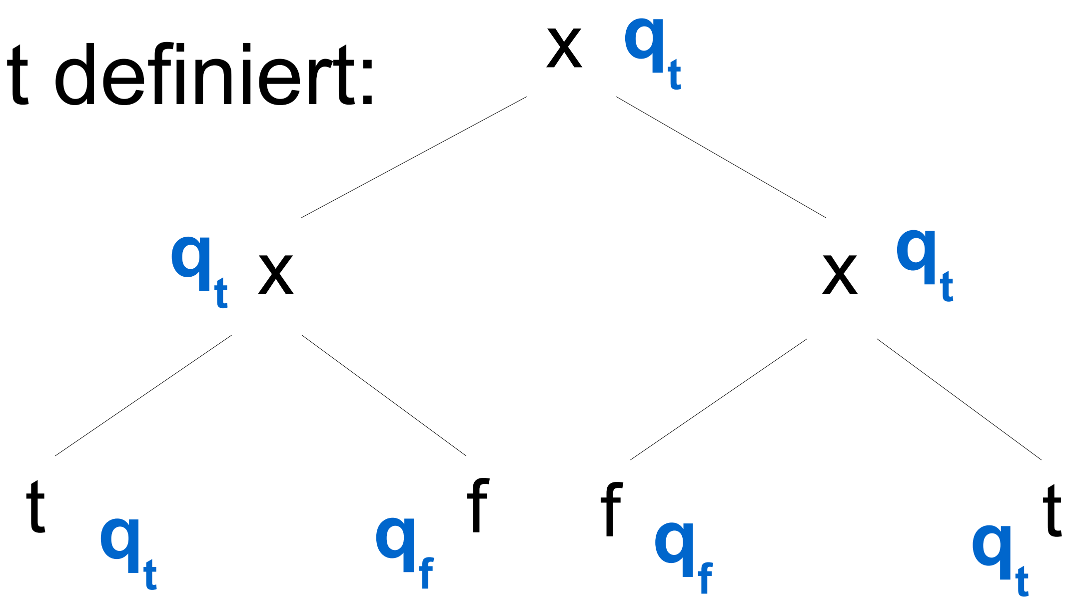
$\varepsilon \xrightarrow{f} q_f$

$(q_t,q_t) \xrightarrow{x} q_t$

$(q_t,q_f) \xrightarrow{x} q_t$

$(q_f,q_t) \xrightarrow{x} q_t$

$(q_f,q_f) \xrightarrow{x} q_f$



Automat A_2 sei wie folgt definiert:

$\varepsilon \xrightarrow{t} q_t'$

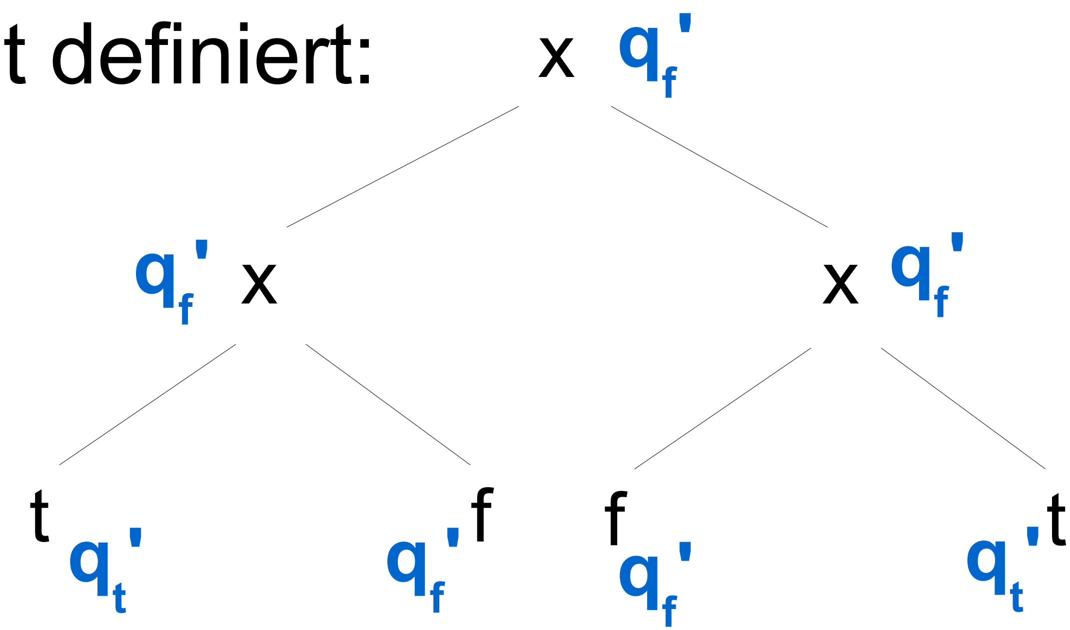
$\varepsilon \xrightarrow{f} q_f'$

$(q_t',q_t') \xrightarrow{x} q_t'$

$(q_t',q_f') \xrightarrow{x} q_f'$

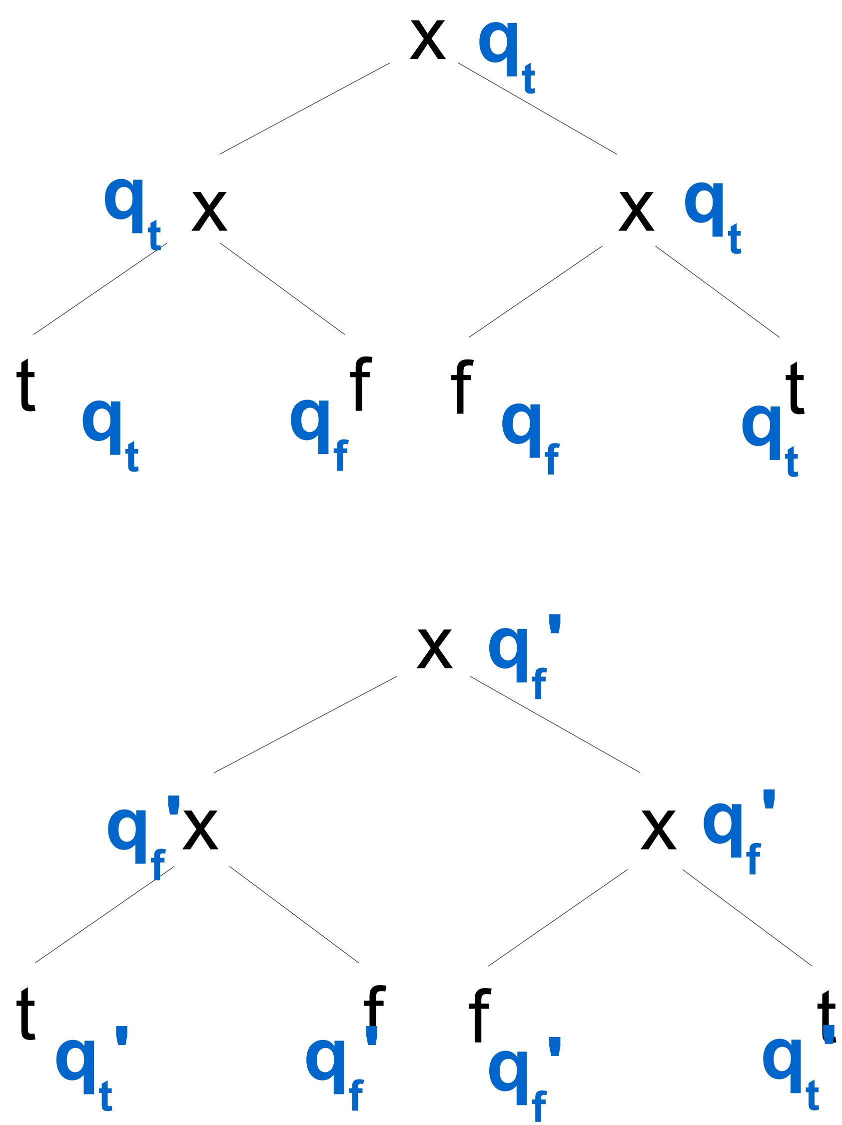
$(q_f',q_t') \xrightarrow{x} q_f'$

$(q_f',q_f') \xrightarrow{x} q_f'$

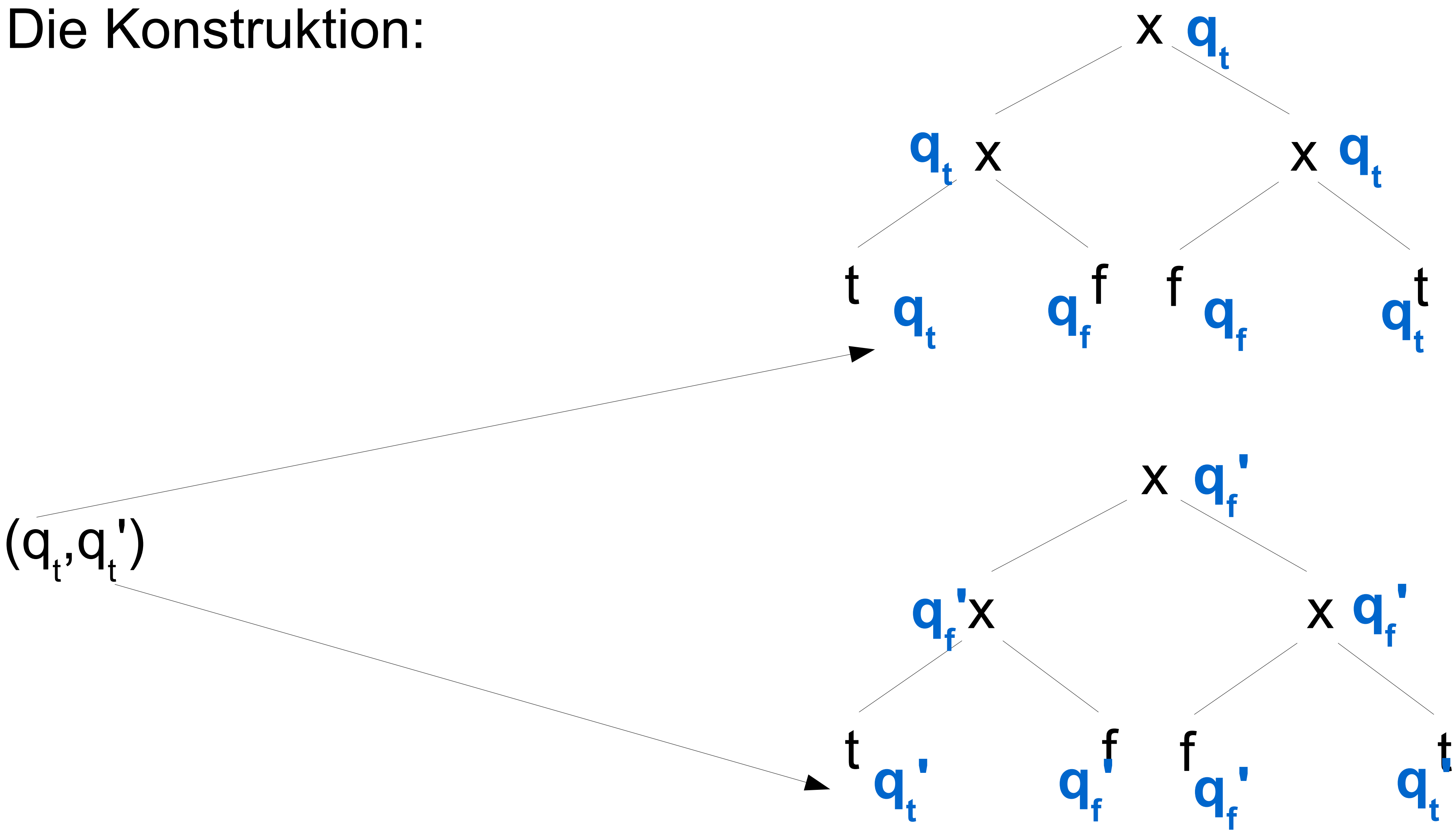


Die Konstruktion:

Die Konstruktion:

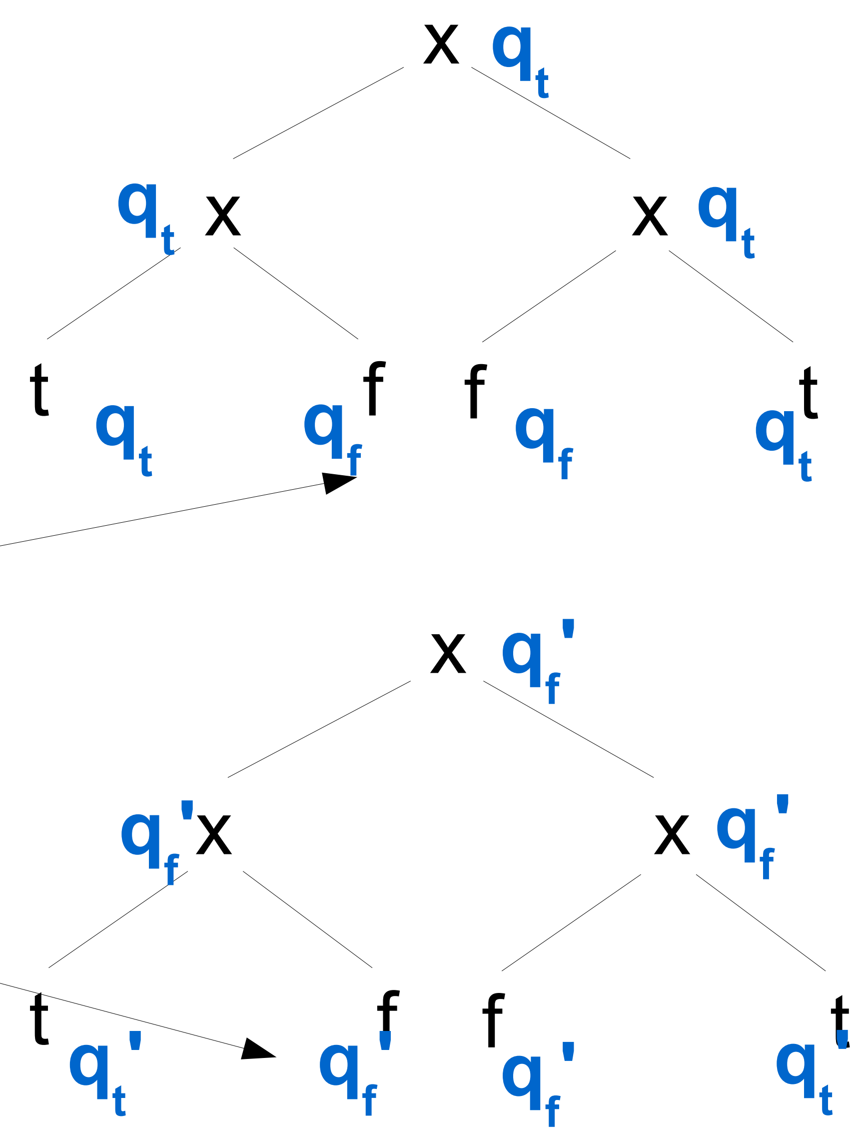


Die Konstruktion:

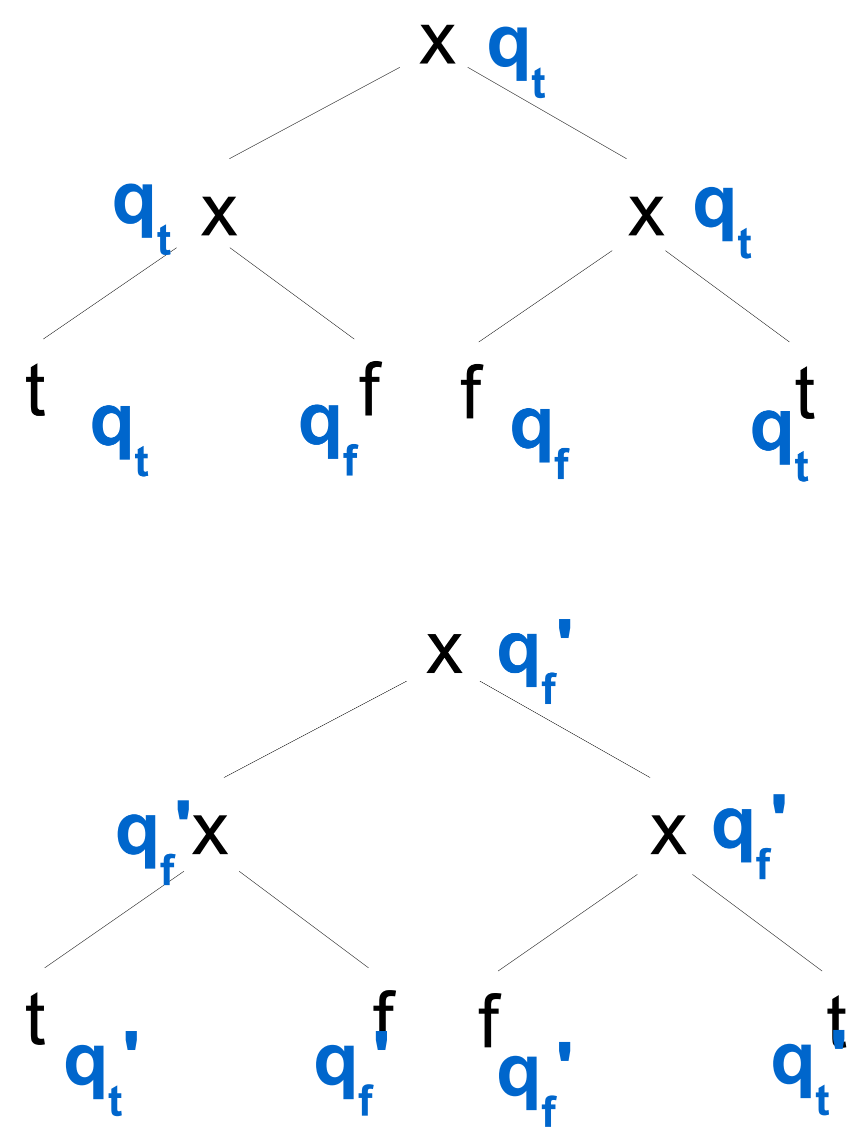


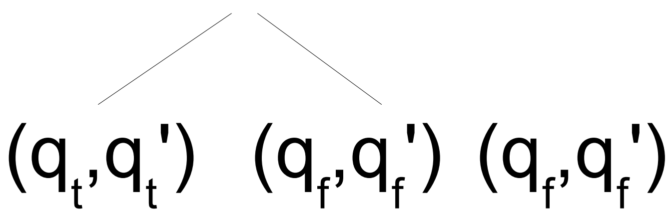
Die Konstruktion:

(q_t, q_t') (q_f, q_f')

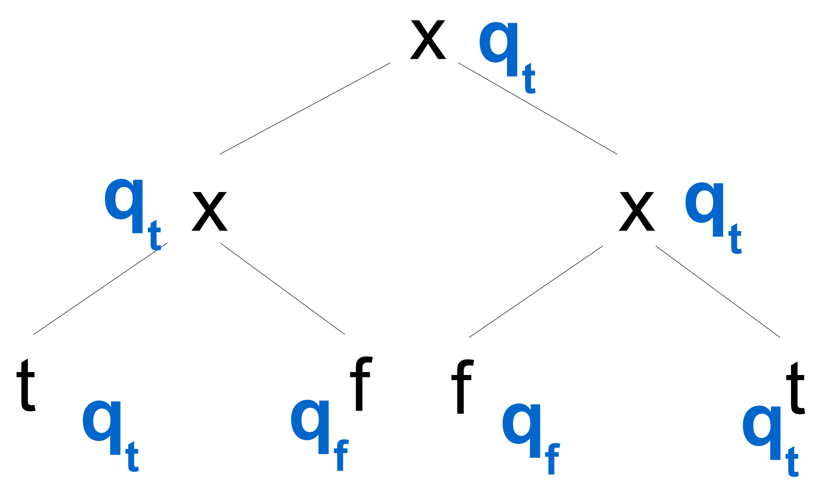


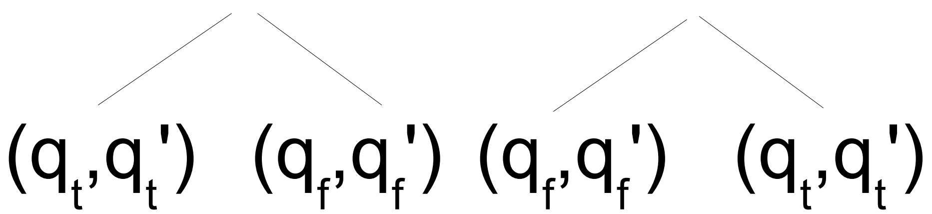
Die Konstruktion:

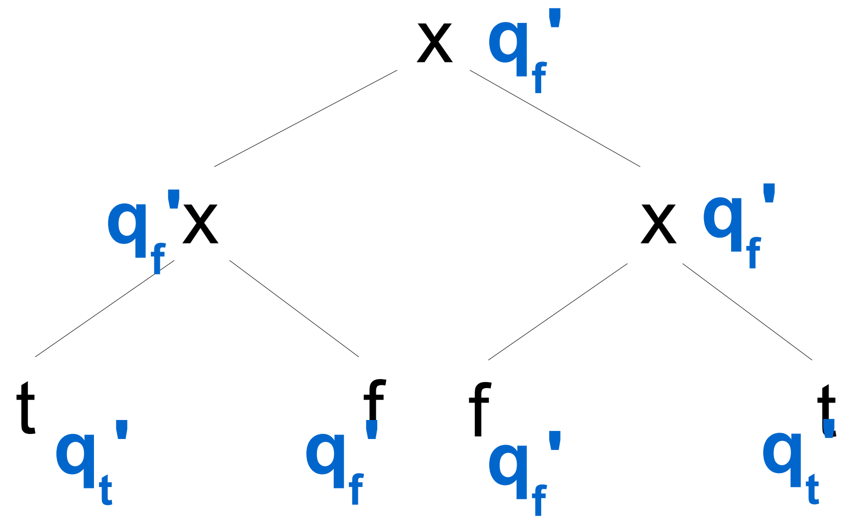




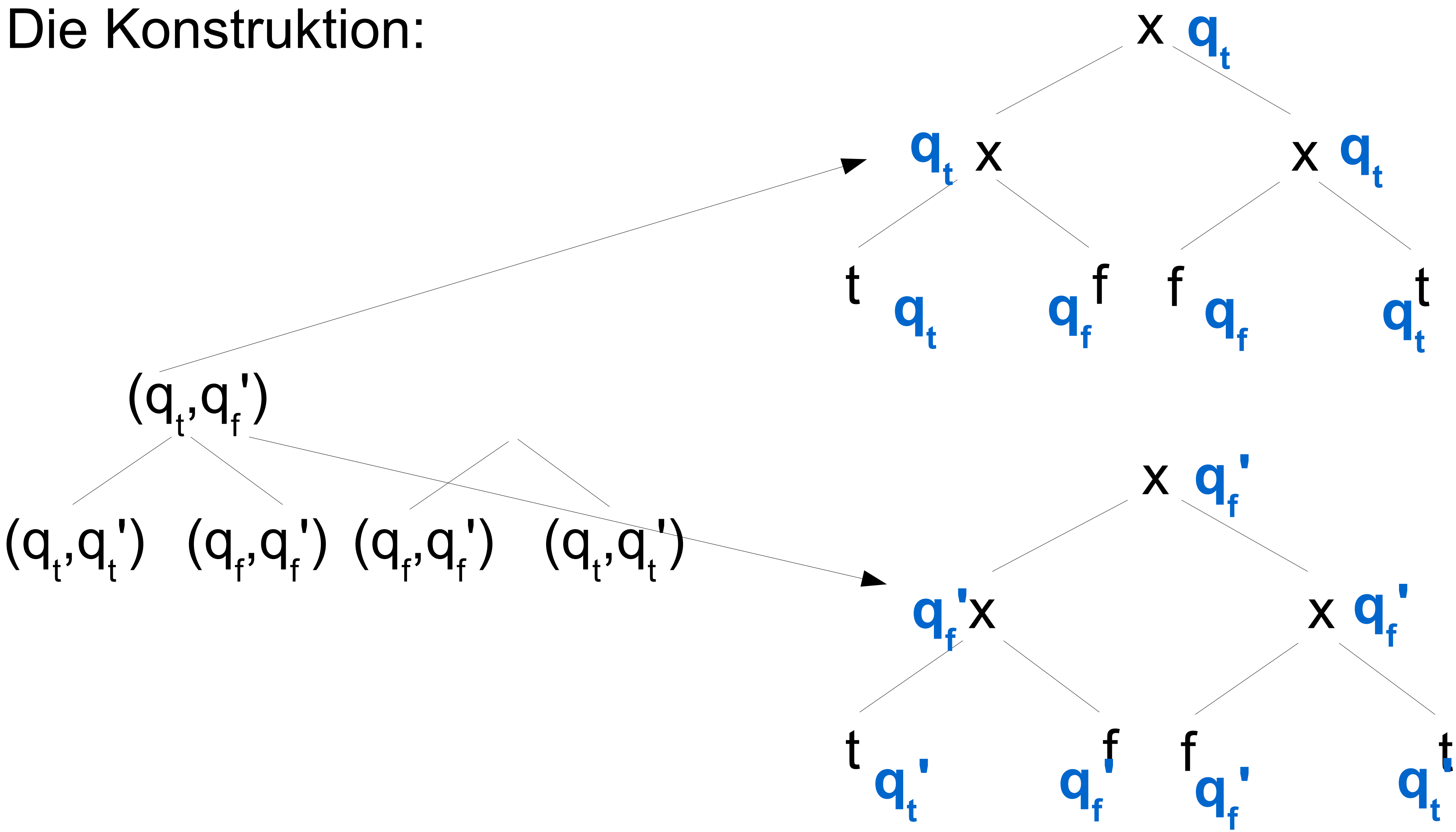
Die Konstruktion:



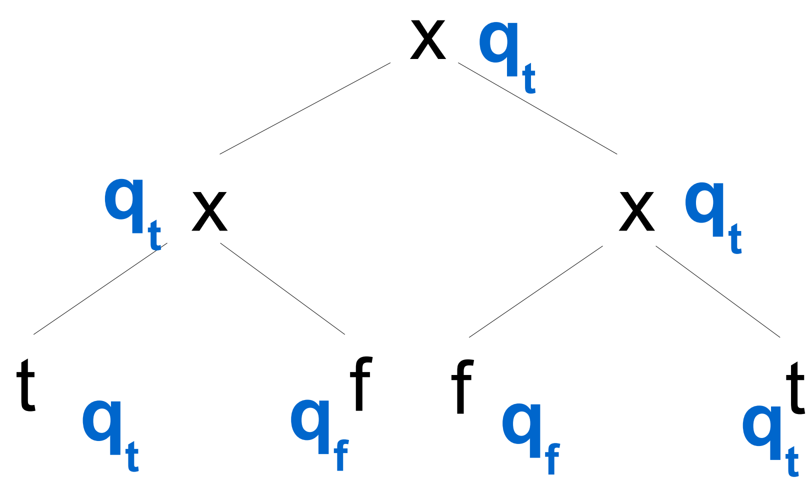


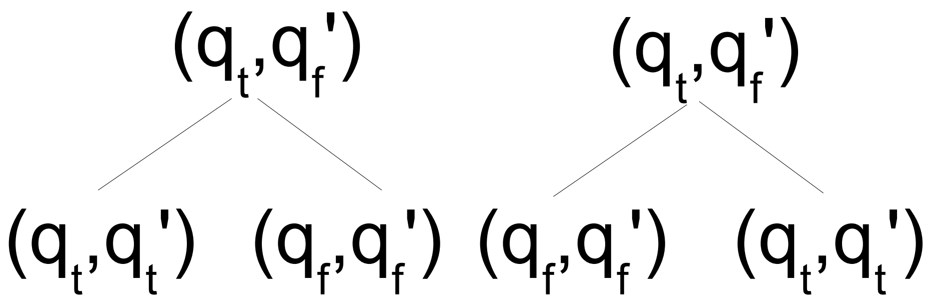


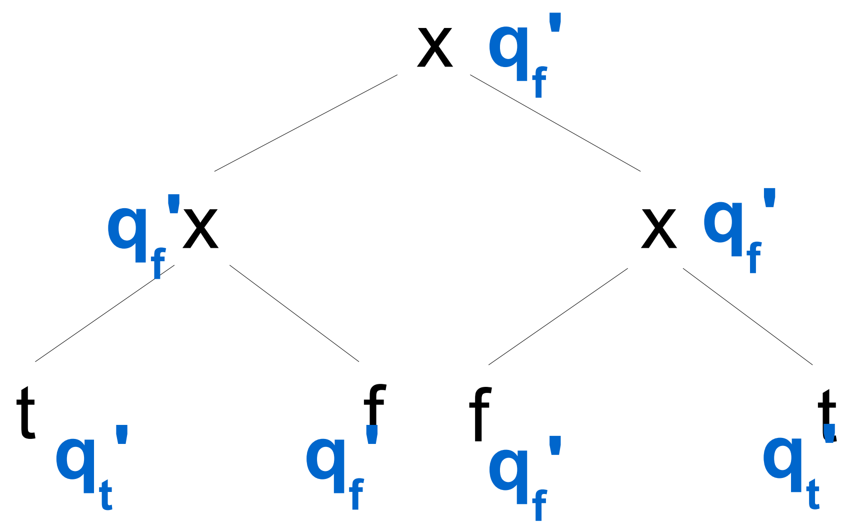
Die Konstruktion:



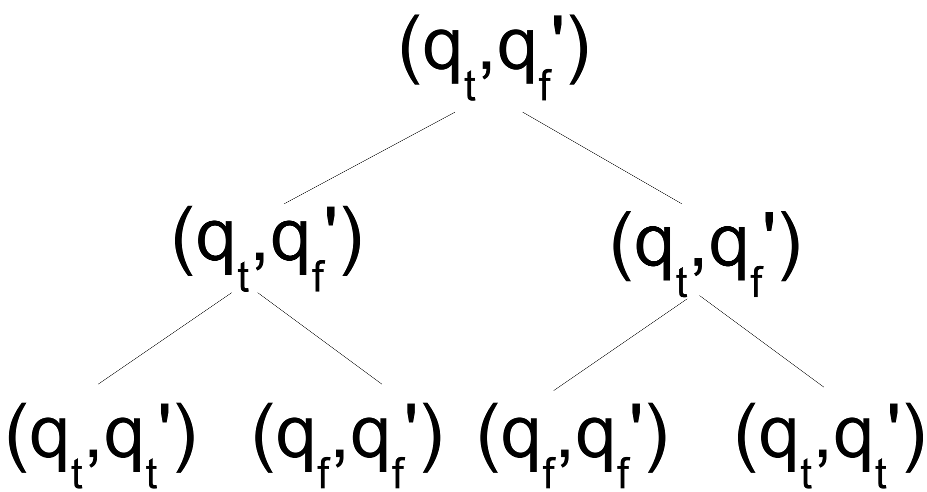
Die Konstruktion:

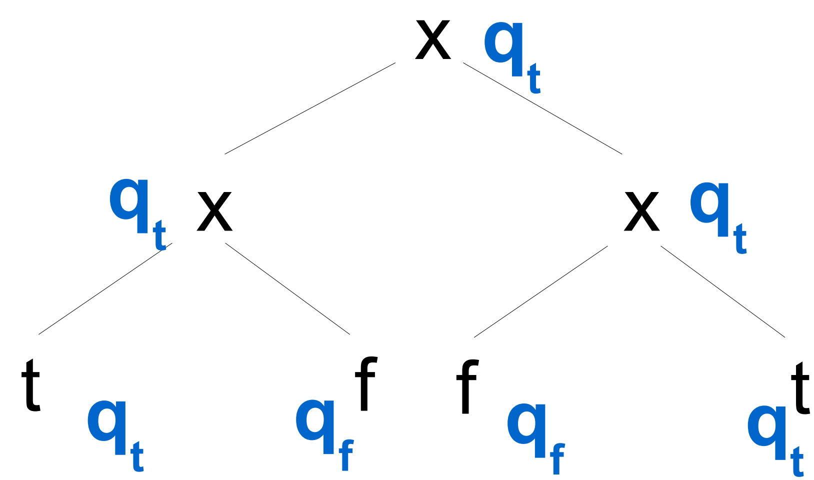


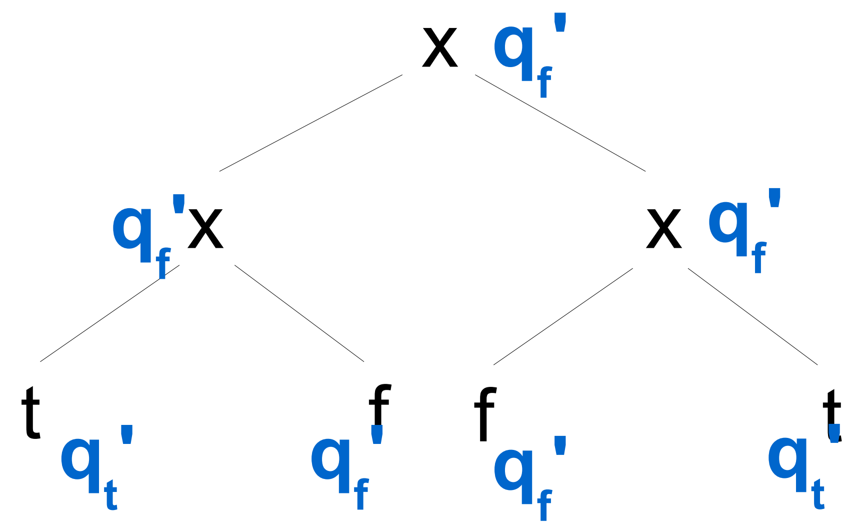




Die Konstruktion:







Abschluss unter Vereinigung

Für die Zustandsübergänge gilt:

$$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$$

$$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$$

Abschluss unter Vereinigung

Für die Zustandsübergänge gilt:

$$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$$
$$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$$

$$((q_t,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t') \qquad ((q_t,q_f'),(q_f,q_t')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t') \qquad ((q_t,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_t,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f') \qquad ((q_t,q_t'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_t,q_t'),(q_f,q_t')) \xrightarrow{x} (q_t,q_t') \qquad ((q_f,q_f'),(q_f,q_t')) \xrightarrow{x} (q_f,q_f')$$
$$((q_t,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f') \qquad ((q_f,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f') \qquad ((q_f,q_t'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$
$$((q_f,q_t'),(q_f,q_t')) \xrightarrow{x} (q_f,q_t') \qquad ((q_t,q_f'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f') \qquad ((q_f,q_f'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$

Für die Zustandsübergänge gilt:

$$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$$
$$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$$

$$((q_t,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$$
$$((q_f,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$$
$$((q_t,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$$
$$((q_t,q_t'),(q_f,q_t')) \xrightarrow{x} (q_t,q_t')$$
$$((q_t,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_t'),(q_f,q_t')) \xrightarrow{x} (q_f,q_t')$$
$$((q_f,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_f'),(q_f,q_t')) \xrightarrow{x} (q_t,q_f')$$
$$((q_t,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_t,q_t'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_f'),(q_f,q_t')) \xrightarrow{x} (q_f,q_f')$$
$$((q_f,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_t'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$
$$((q_t,q_f'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$
$$((q_f,q_f'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$

Eigenschaften:

Eigenschaften:

→ benötigt Vollständigkeit
(d.h. Für alle Zustand-Symbol-Kombinationen gibt es einen Zustandsübergang)

Abschluss unter Vereinigung

Eigenschaften:

→ benötigt Vollständigkeit
(d.h. Für alle Zustand-Symbol-Kombinationen gibt es einen Zustandsübergang)

→ Vollständigkeit und Determinismus bleiben erhalten

Abschluss unter Vereinigung

Eigenschaften:

→ benötigt Vollständigkeit
(d.h. Für alle Zustand-Symbol-Kombinationen gibt es einen Zustandsübergang)

→ Vollständigkeit und Determinismus bleiben erhalten

→ P-TIME

Abschluss unter Komplement

Gegeben: Vollständiger Baumautomat A_1

Gesucht: $L(A) = \mathcal{T}(\mathcal{F}) - L(A_1)$, wobei

T(F) := Menge aller Bäume

Gegeben: Vollständiger Baumautomat A_1

Gesucht: $L(A) = \mathcal{T}(\mathcal{F}) - L(A_1)$, wobei

$T(F)$:= Menge aller Bäume

Algorithmus:

Komplementieren der Finalzustände

Abschluss unter Komplement

Wir erinnern uns an den Automaten von der Vereinigung:

$$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$$

$$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$$

$$((q_t,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$$

$$((q_f,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$$

$$((q_t,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_t'),(q_f,q_t')) \xrightarrow{x} (q_t,q_t')$$

$$((q_t,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_t'),(q_f,q_t')) \xrightarrow{x} (q_f,q_t')$$

$$((q_f,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_f'),(q_f,q_t')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_t'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_f'),(q_f,q_t')) \xrightarrow{x} (q_f,q_f')$$

$$((q_f,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_t'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$

$$((q_t,q_f'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_f'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$

Und fertig:

$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$

$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$

$((q_t,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$

$((q_f,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$

$((q_t,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_t'),(q_f,q_t')) \xrightarrow{x} (q_t,q_t')$

$((q_t,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_t'),(q_f,q_t')) \xrightarrow{x} (q_f,q_t')$

$((q_f,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_f'),(q_f,q_t')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_t'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_f'),(q_f,q_t')) \xrightarrow{x} (q_f,q_f')$

$((q_f,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_t'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$

$((q_t,q_f'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_f'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$

Eigenschaften:

Eigenschaften:

→ benötigt Vollständigkeit & Determinismus

Abschluss unter Komplement

Eigenschaften:

→ benötigt Vollständigkeit & Determinismus

→ Vollständigkeit und Determinismus bleiben erhalten

Eigenschaften:

→ benötigt Vollständigkeit & Determinismus

→ Vollständigkeit und Determinismus bleiben erhalten

→ P-TIME für DTFA

Eigenschaften:

→ benötigt Vollständigkeit & Determinismus

→ Vollständigkeit und Determinismus bleiben erhalten

→ P-TIME für DTFA

→ EXP-TIME für NTFA

Gegeben: Zwei Baumautomaten A_1 und A_2

Gesucht: $L(A) = L(A_1) \cap L(A_2)$

Gegeben: Zwei Baumautomaten A_1 und A_2

Gesucht: $L(A) = L(A_1) \cap L(A_2)$

Algorithmus:

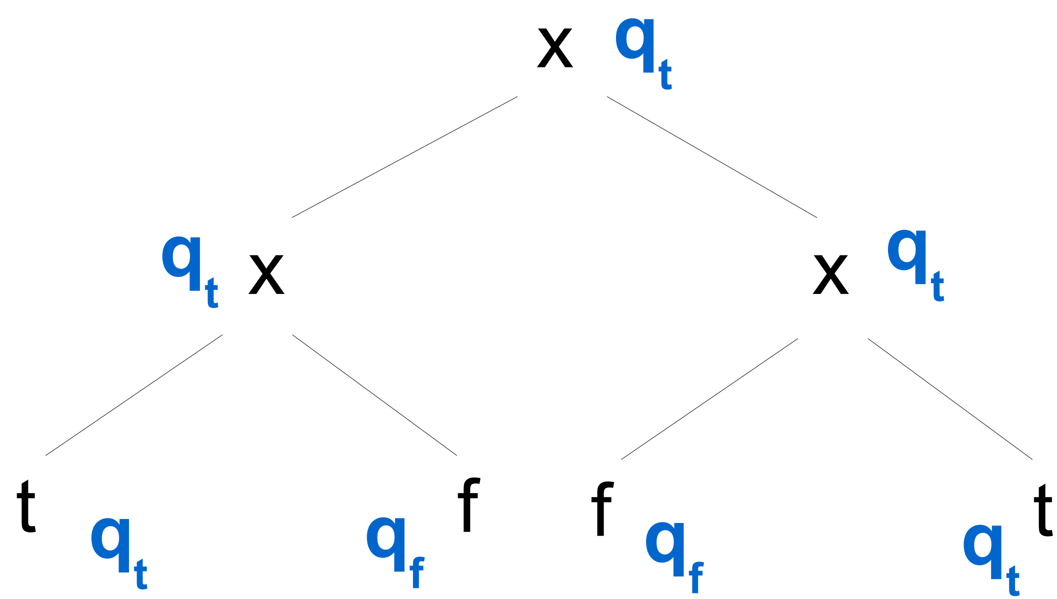
Bildung des Produktautomaten, aber Akzeptanz nur wenn genau beide Automaten akzeptieren

Abschluss unter Schnitt

Wir erinnern uns an A_1:

$$\varepsilon \xrightarrow{t} q_t \qquad \varepsilon \xrightarrow{f} q_f$$

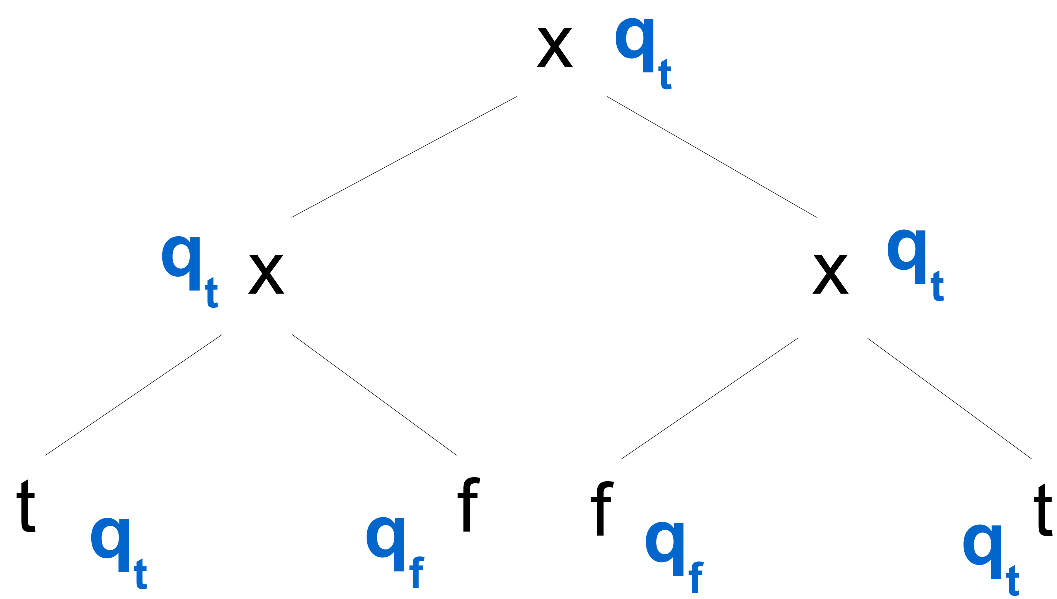
$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Wir erinnern uns an A_1:

$$\varepsilon \xrightarrow{t} q_t \qquad \varepsilon \xrightarrow{f} q_f$$

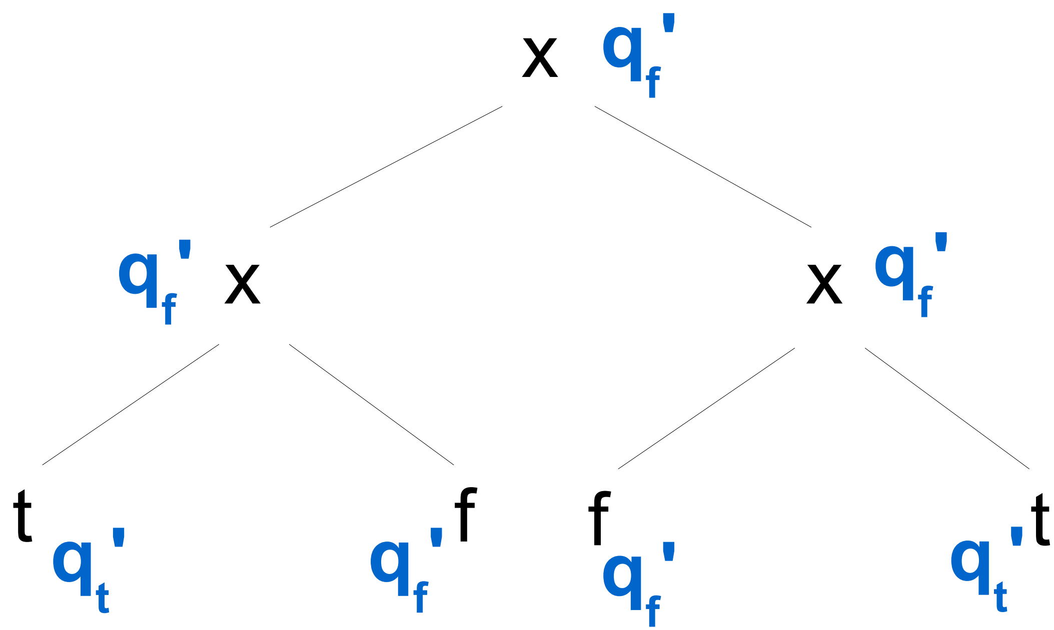
$$(q_t,q_t) \xrightarrow{x} q_t$$
$$(q_t,q_f) \xrightarrow{x} q_t$$
$$(q_f,q_t) \xrightarrow{x} q_t$$
$$(q_f,q_f) \xrightarrow{x} q_f$$



Automat A_2:

$$\varepsilon \xrightarrow{t} q_t' \qquad \varepsilon \xrightarrow{f} q_f'$$

$$(q_t',q_t') \xrightarrow{x} q_t'$$
$$(q_t',q_f') \xrightarrow{x} q_f'$$
$$(q_f',q_t') \xrightarrow{x} q_f'$$
$$(q_f',q_f') \xrightarrow{x} q_f'$$



Dies war unser Vereinigungs-Produkt:

$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$

$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$

$((q_t,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$

$((q_f,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$

$((q_t,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_t'),(q_f,q_t')) \xrightarrow{x} (q_t,q_t')$

$((q_t,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_t'),(q_f,q_t')) \xrightarrow{x} (q_f,q_t')$

$((q_f,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_f'),(q_f,q_t')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_t,q_t'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_f'),(q_f,q_t')) \xrightarrow{x} (q_f,q_f')$

$((q_f,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_t'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$

$((q_t,q_f'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$

$((q_f,q_f'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$

Und dies ergibt sich als Schnitt-Produkt:

$$(\varepsilon,\varepsilon) \xrightarrow{t} (q_t,q_t')$$

$$(\varepsilon,\varepsilon) \xrightarrow{f} (q_f,q_f')$$

$$((q_t,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$$

$$((q_f,q_t'),(q_t,q_t')) \xrightarrow{x} (q_t,q_t')$$

$$((q_t,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_t'),(q_f,q_t')) \xrightarrow{x} (q_t,q_t')$$

$$((q_t,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_f'),(q_t,q_t')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_t'),(q_f,q_t')) \xrightarrow{x} (q_f,q_t')$$

$$((q_f,q_t'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_f'),(q_f,q_t')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_t,q_t'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_f'),(q_f,q_t')) \xrightarrow{x} (q_f,q_f')$$

$$((q_f,q_f'),(q_t,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_t'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$

$$((q_t,q_f'),(q_f,q_f')) \xrightarrow{x} (q_t,q_f')$$

$$((q_f,q_f'),(q_f,q_f')) \xrightarrow{x} (q_f,q_f')$$

Eigenschaften:

Eigenschaften:

$\rightarrow$ benötigt Vollständigkeit

Eigenschaften:

$\rightarrow$ benötigt Vollständigkeit

$\rightarrow$ Determinismus und Vollständigkeit bleiben erhalten

Eigenschaften:

→ benötigt Vollständigkeit

→ Determinismus und Vollständigkeit bleiben erhalten

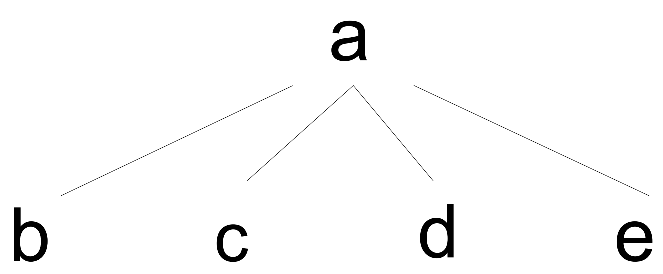
→ P-TIME

Wir nutzen eine Kodierung um unbeschränkt verzweigte Bäume durch binär verzweigte darstellen zu können:

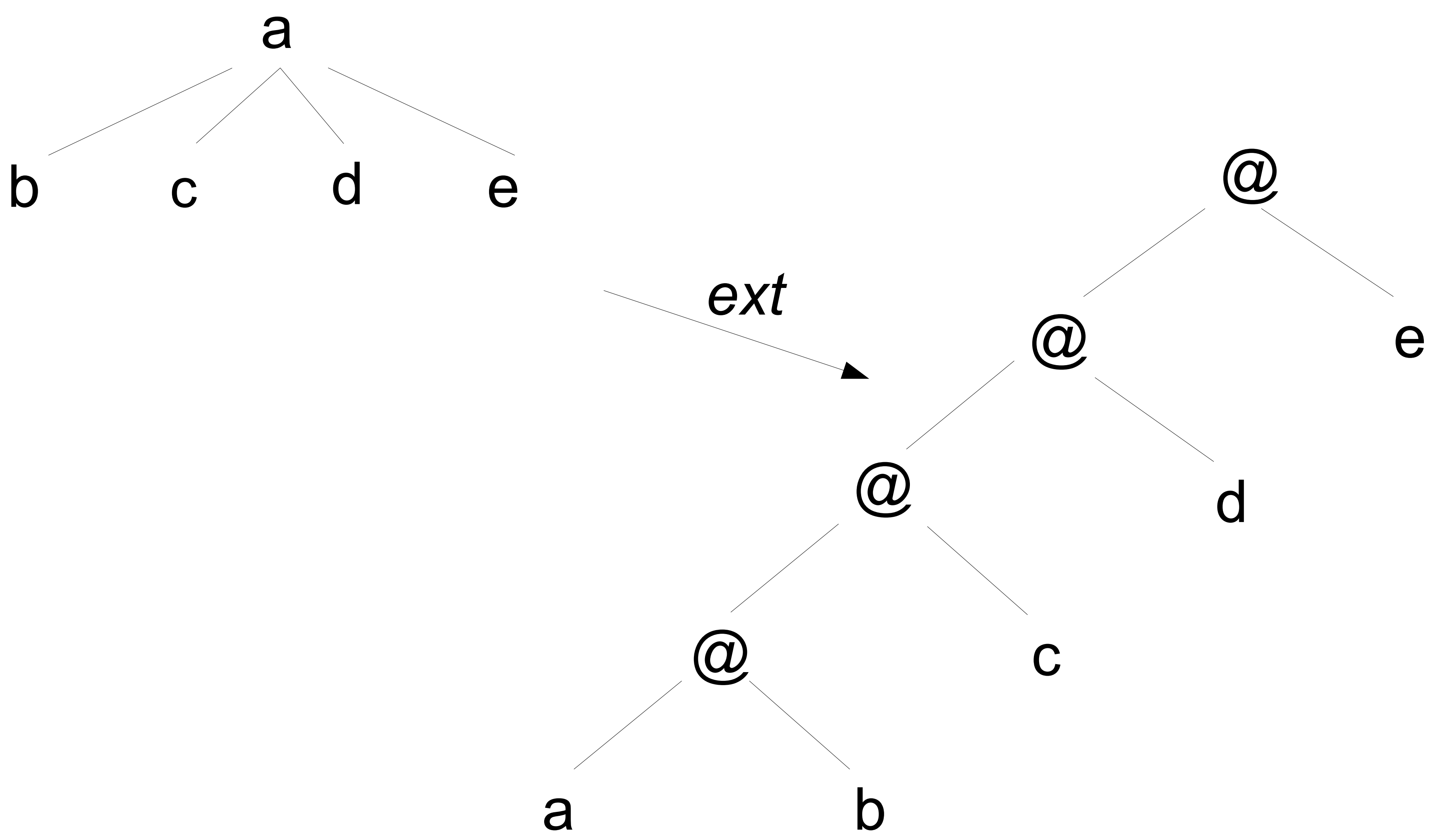
Wir nutzen eine Kodierung um unbeschränkt verzweigte Bäume durch binär verzweigte darstellen zu können:

→ Extension Kodierung *ext* mit
Extension Operator @

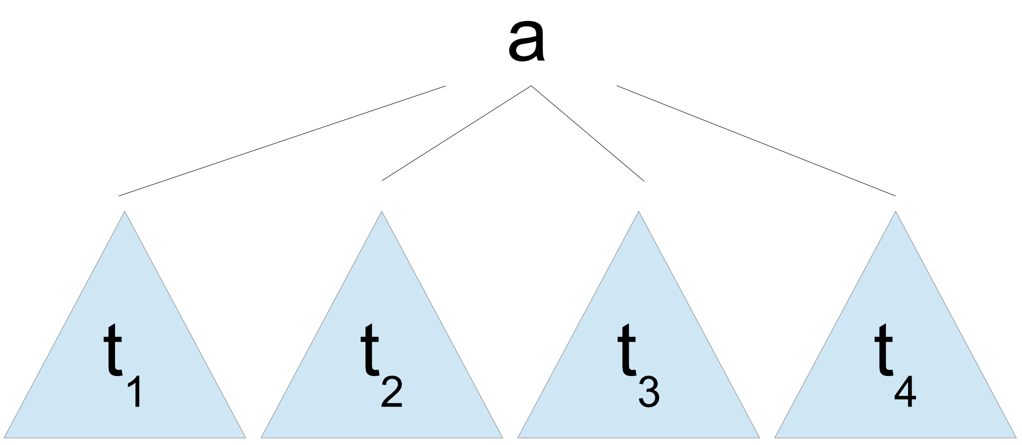
Die Konstruktion:



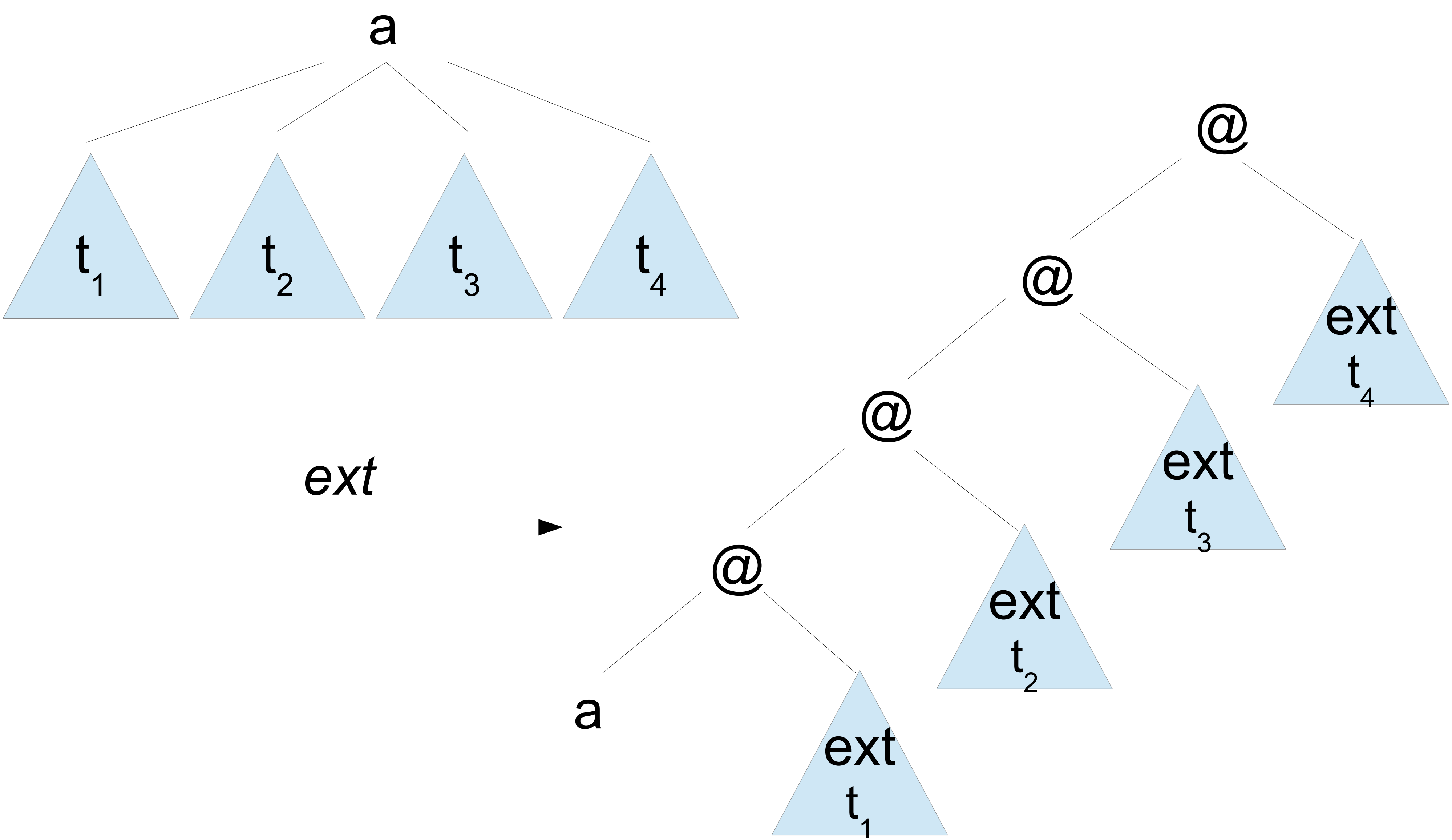
Die Konstruktion:



Die Konstruktion(2):

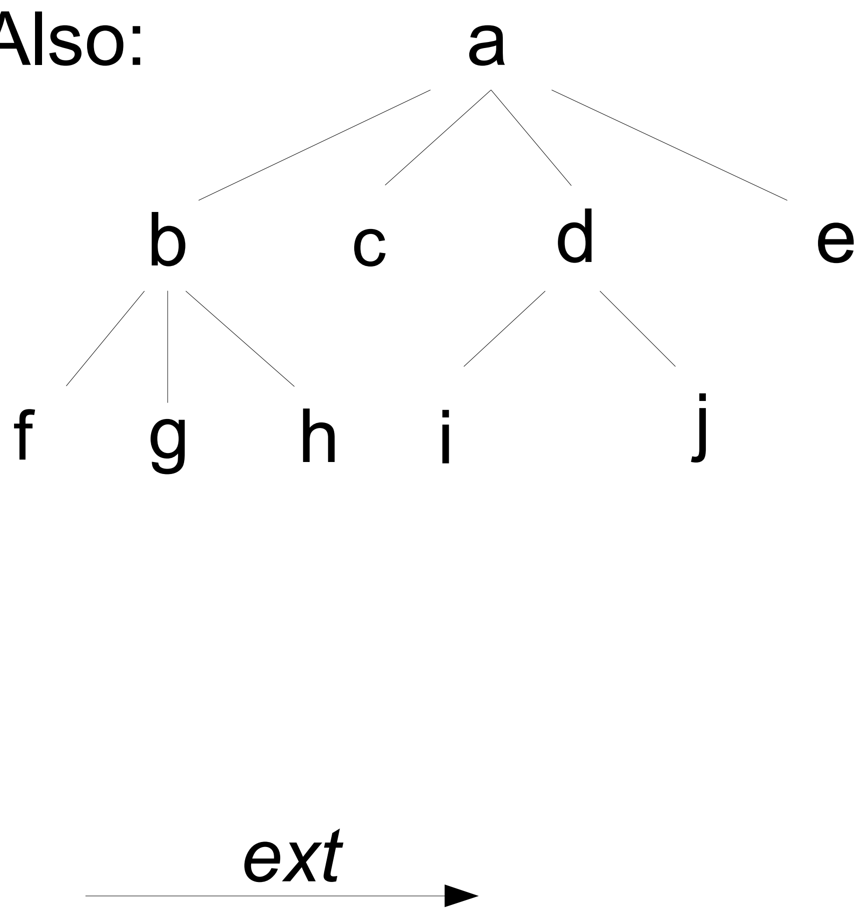


Die Konstruktion(2):

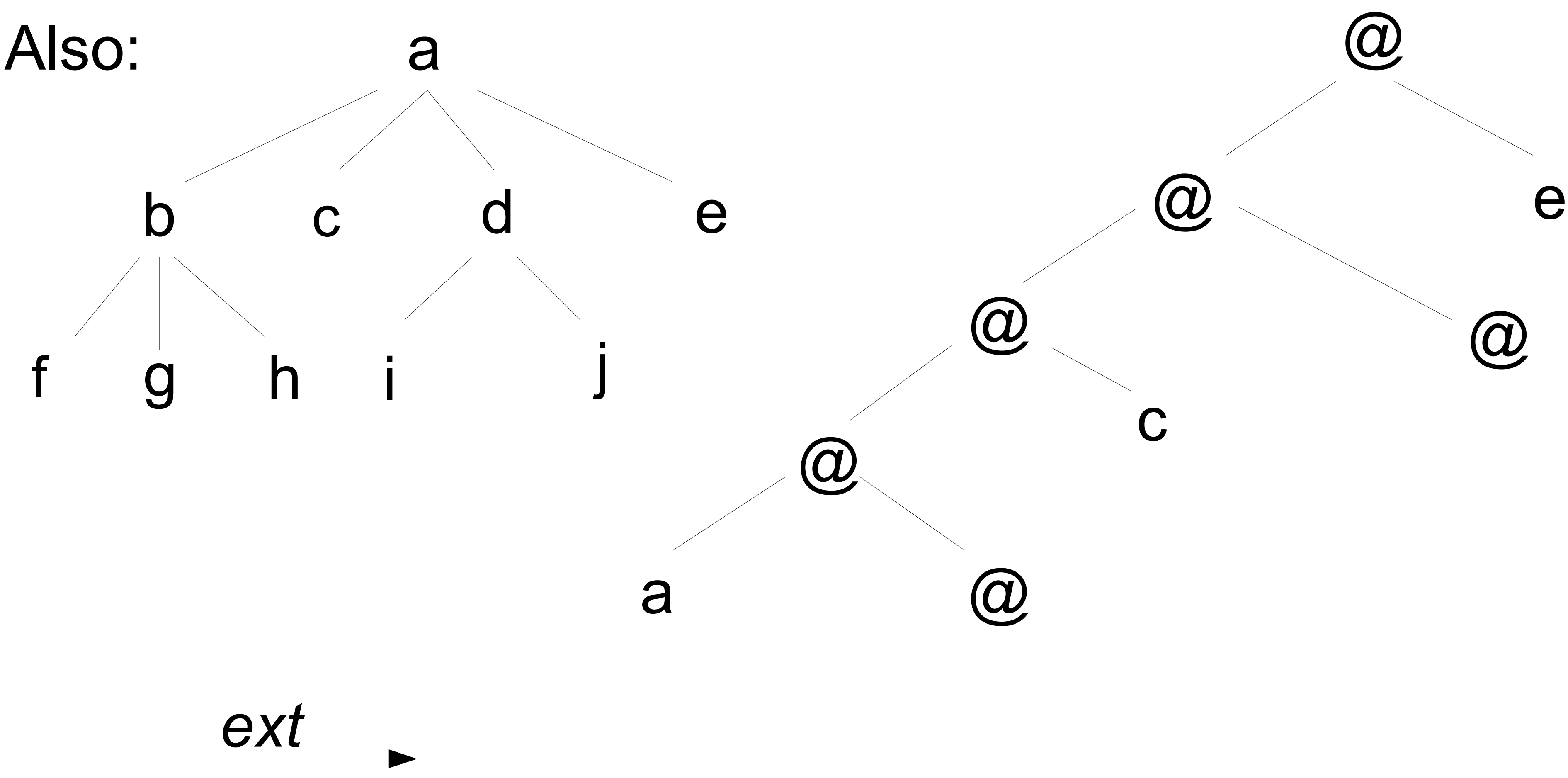


Extension Operator Kodierung

Also:

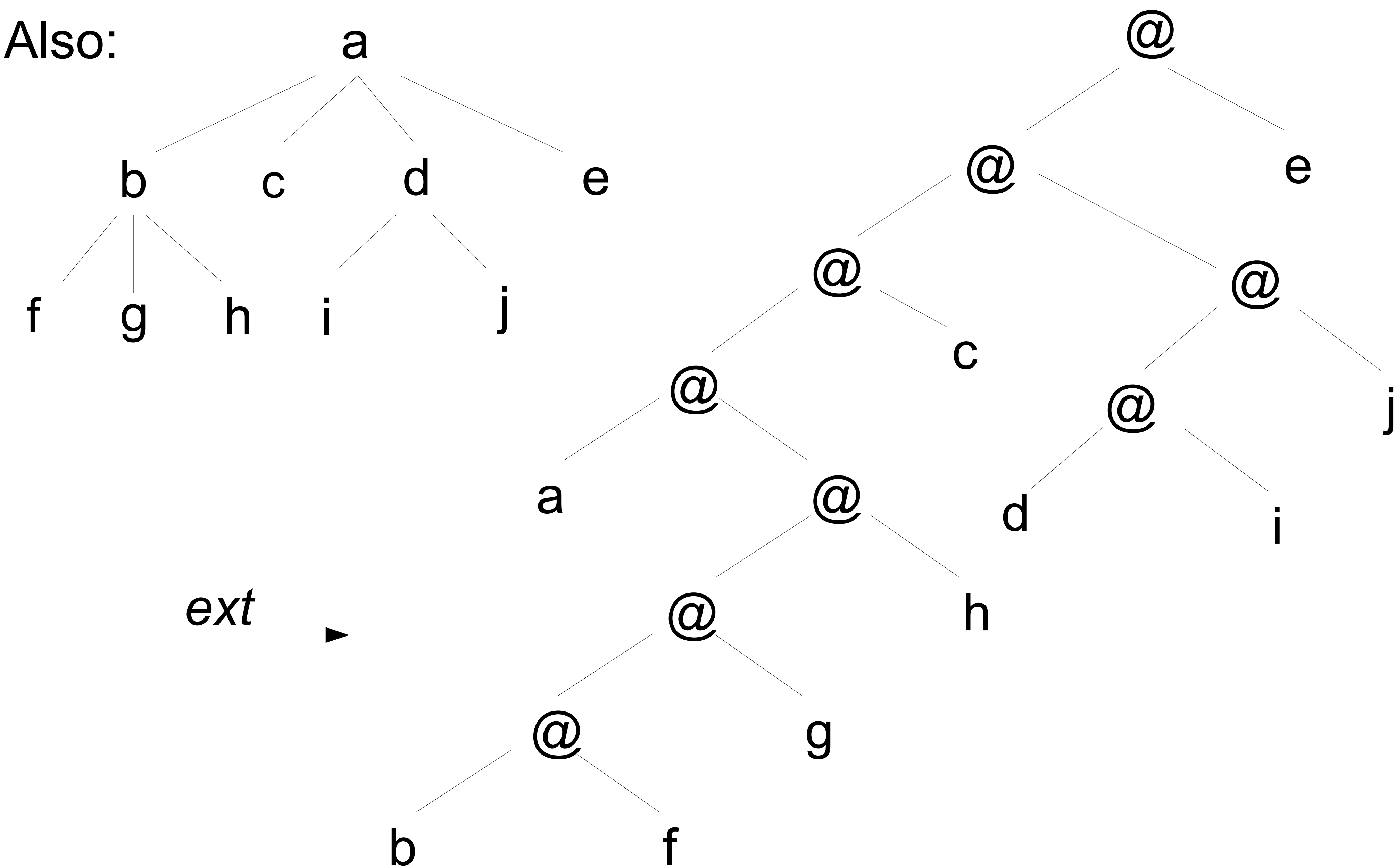


Also:



Extension Operator Kodierung

Also:



Extension Operator Kodierung

Auf alle Bäume eines Baumautomaten angewandt:
$\rightarrow$ Bijektion zweier Baumsprachen

Extension Operator Kodierung

Auf alle Bäume eines Baumautomaten angewandt:
→ Bijektion zweier Baumsprachen

Sie ist weiterhin:
→ Bijektion zwischen regulären Baumsprachen

Auf alle Bäume eines Baumautomaten angewandt:
$\rightarrow$ Bijektion zweier Baumsprachen

Sie ist weiterhin:
$\rightarrow$ Bijektion zwischen regulären Baumsprachen

Es gilt somit:

L(A) ist regulär $\Leftrightarrow$ *ext*(L(A)) regulär

Extension Operator Kodierung

Auf alle Bäume eines Baumautomaten angewandt:
→ Bijektion zweier Baumsprachen

Sie ist weiterhin:
→ Bijektion zwischen regulären Baumsprachen

Es gilt somit:

L(A) ist regulär ⇔ *ext*(L(A)) regulär

Außerdem:

→ Komplexität der Kodierung ist PTIME

Theorem 2:
Die Klasse von regulären unbeschränkt verzweigten Baumsprachen ist abgeschlossen unter:

Vereinigung

Komplement

Schnitt

Theorem 2:
Die Klasse von regulären unbeschränkt verzweigten Baumsprachen ist abgeschlossen unter:

Vereinigung

Komplement

Schnitt

→ gilt, da *ext*-Kodierung eine Bijektion der regulären Baumsprachen ist

Outline

1. Abschlusseigenschaften

2. Algorithmen

Basistermproblem

Gegeben: Ein Automat A und ein Baum t

Gesucht: Ist t $\in$ L(A)?

Basistermproblem

Gegeben: Ein Automat A und ein Baum t

Gesucht: Ist t $\in$ L(A)?

Algorithmus für binär verzweigte Bäume:
Berechne „bottom-up" alle erreichbaren Zustände

Basistermproblem

Gegeben: Ein Automat A und ein Baum t

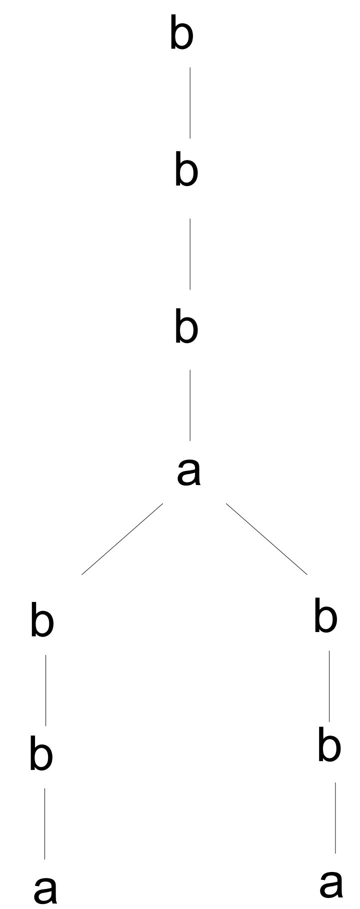
Gesucht: Ist t $\in$L(A)?

Algorithmus für binär verzweigte Bäume:
Berechne „bottom-up“ alle erreichbaren Zustände

Komplexität:
$\rightarrow$ PTIME

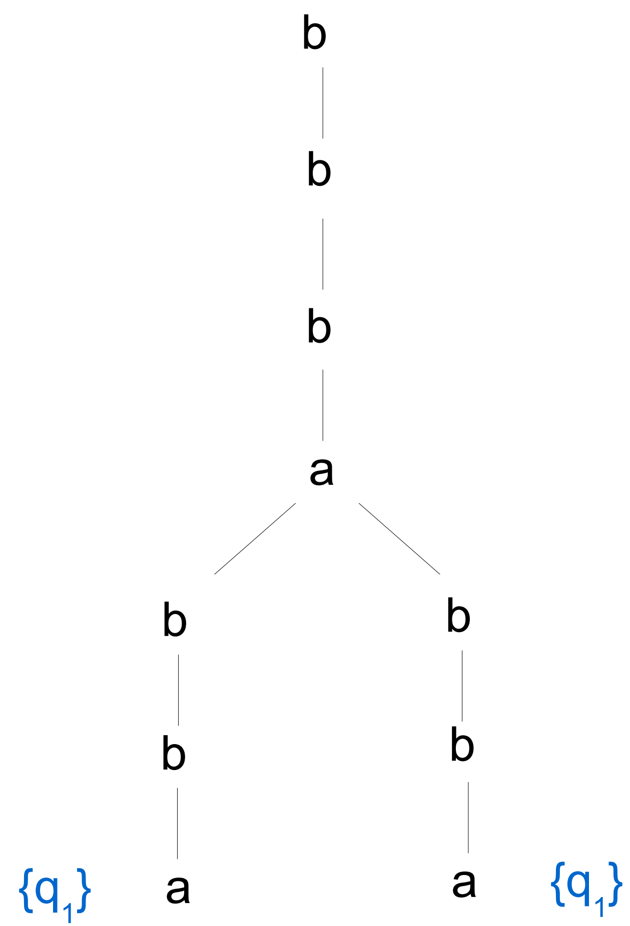
Automat A wie folgt definiert:

$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



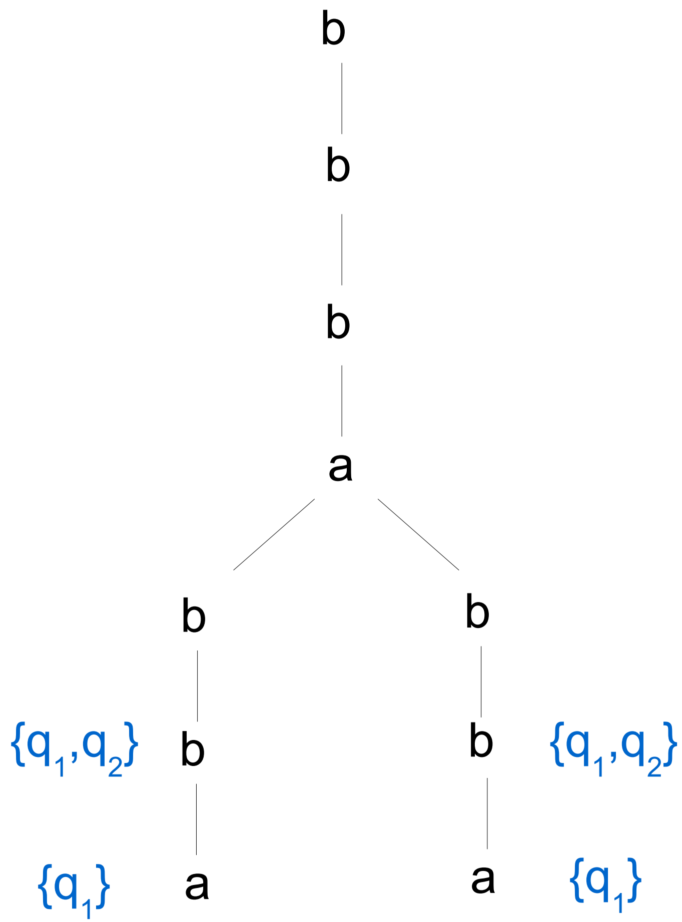
Automat A wie folgt definiert:

$$
\begin{aligned}
\varepsilon &\xrightarrow{a} q_1 \\
q_1 &\xrightarrow{b} q_1 \\
q_1 &\xrightarrow{b} q_2 \\
q_2 &\xrightarrow{b} q_f \\
(q_2,q_2) &\xrightarrow{a} q_1
\end{aligned}
$$



Automat A wie folgt definiert:

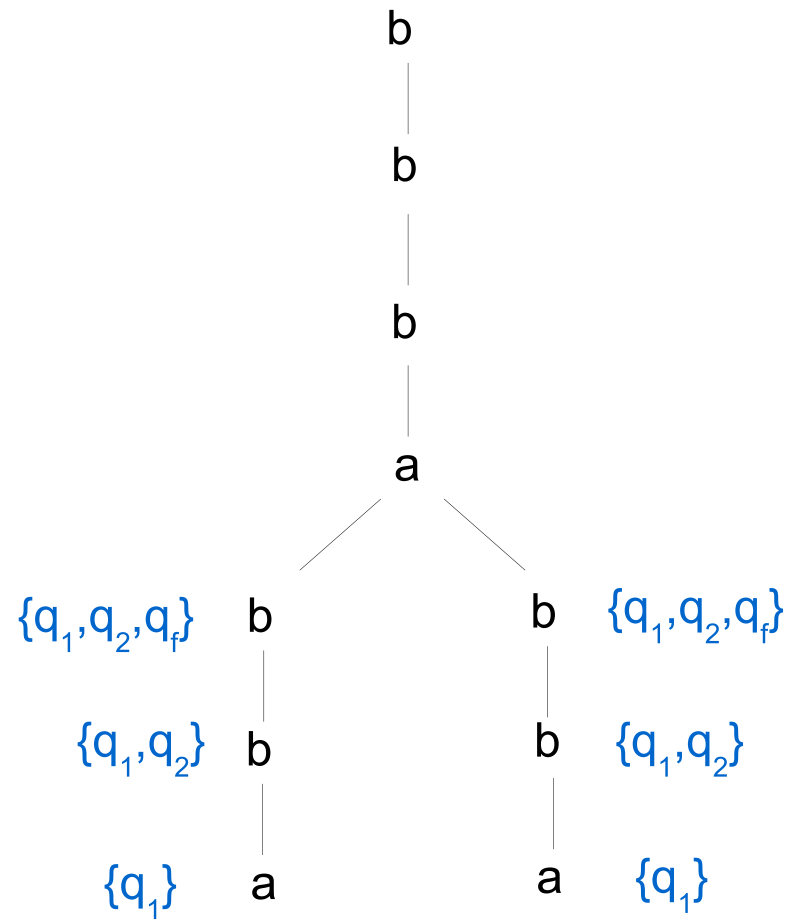
$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



Basistermproblem

Automat A wie folgt definiert:

$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



Basistermproblem

Automat A wie folgt definiert:

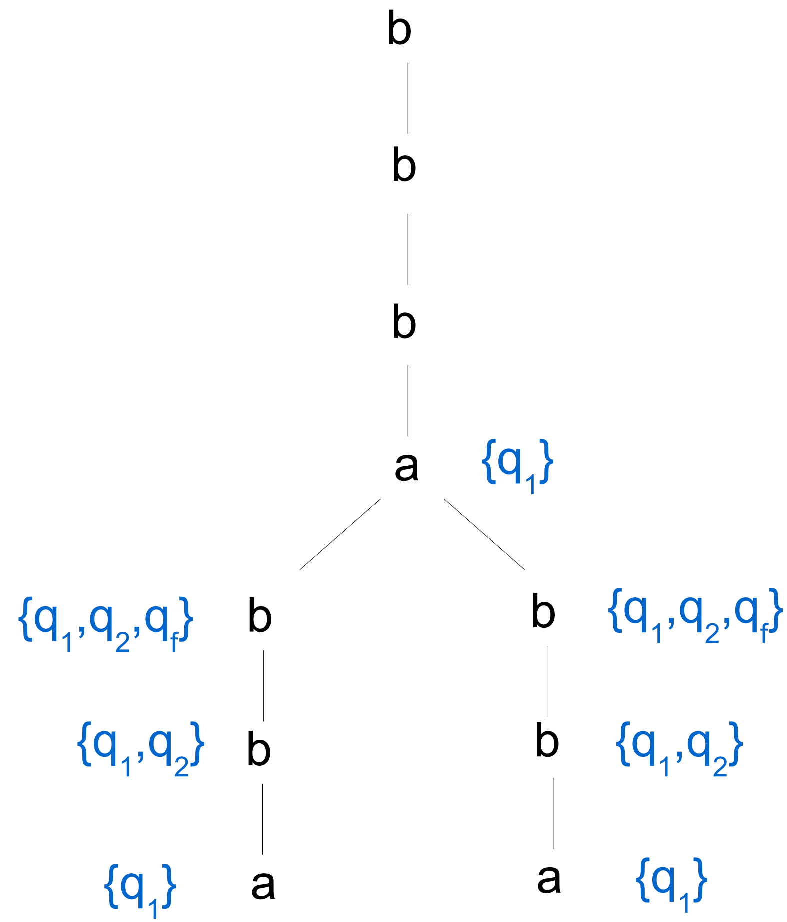
$$\varepsilon \xrightarrow{a} q_1$$

$$q_1 \xrightarrow{b} q_1$$

$$q_1 \xrightarrow{b} q_2$$

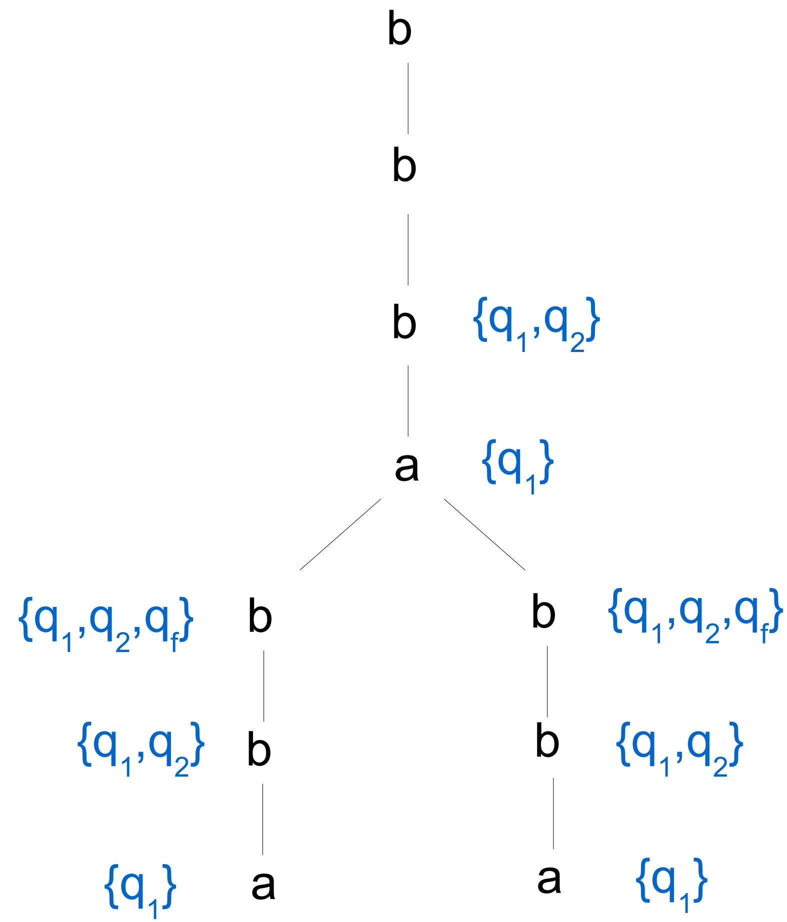
$$q_2 \xrightarrow{b} q_f$$

$$(q_2,q_2) \xrightarrow{a} q_1$$



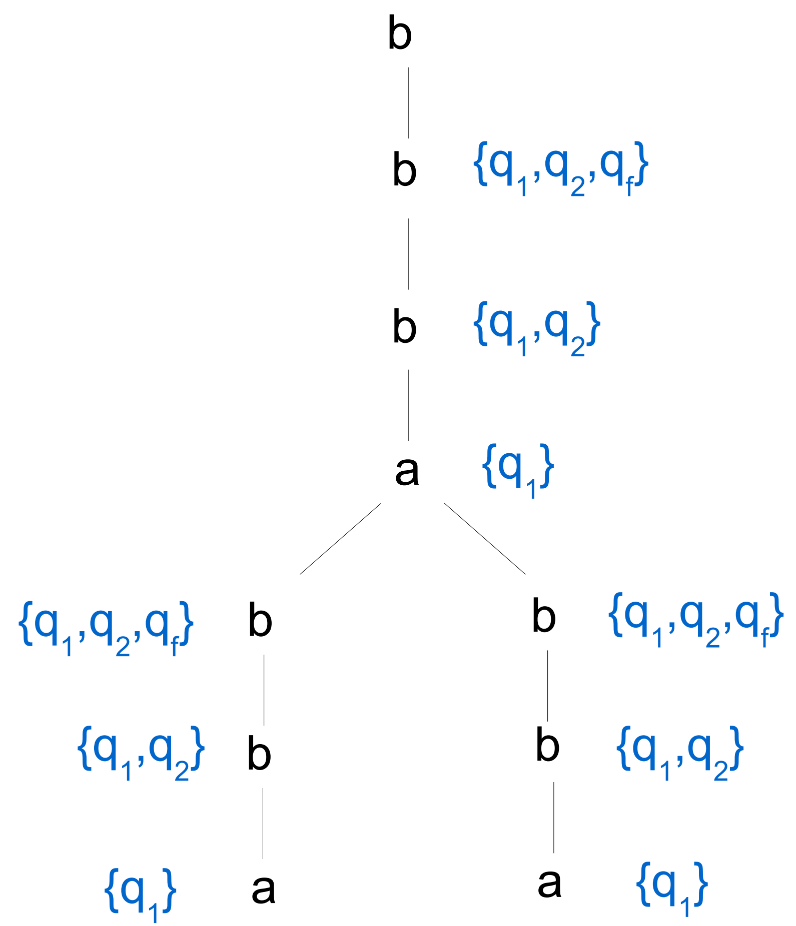
Automat A wie folgt definiert:

$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



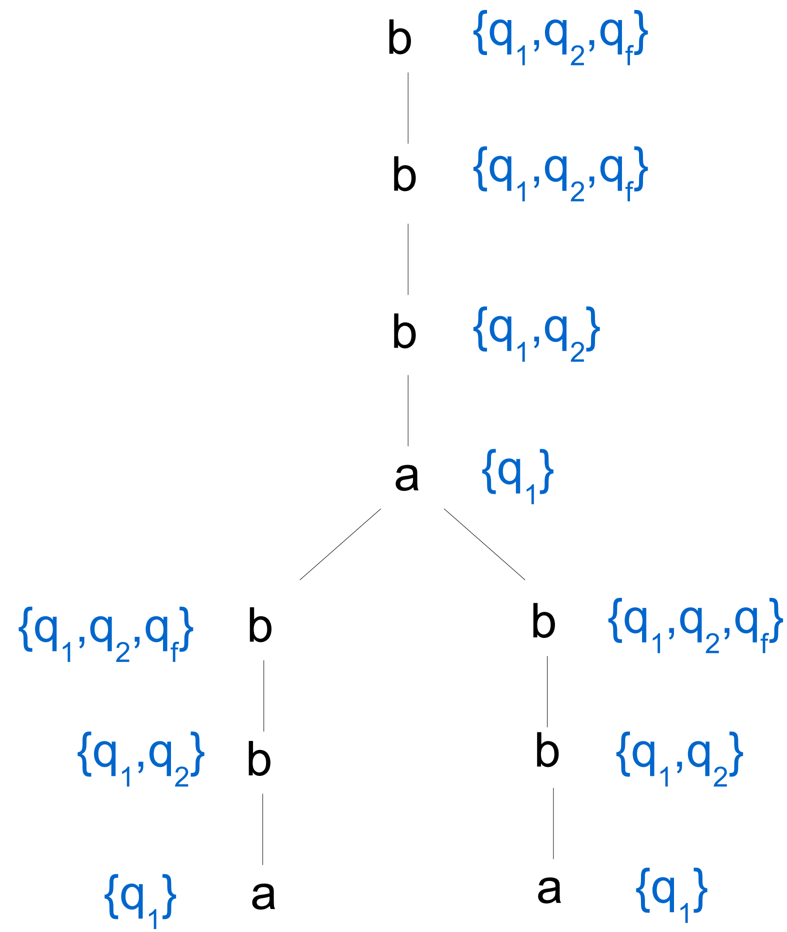
Automat A wie folgt definiert:

$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



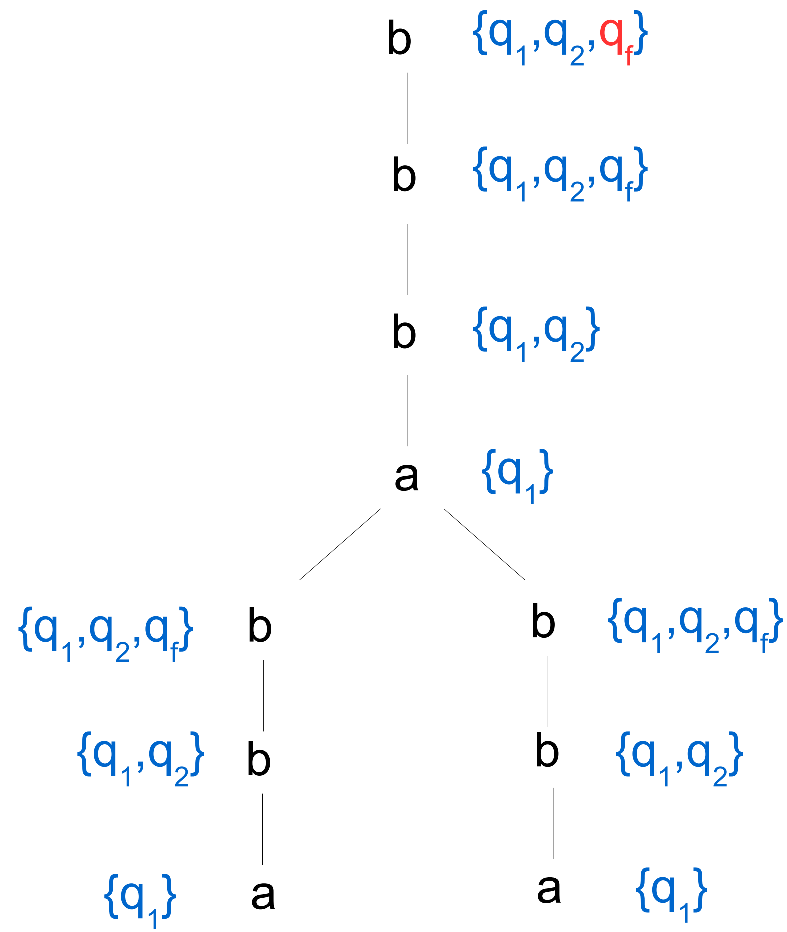
Automat A wie folgt definiert:

$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



Automat A wie folgt definiert:

$$\varepsilon \xrightarrow{a} q_1$$
$$q_1 \xrightarrow{b} q_1$$
$$q_1 \xrightarrow{b} q_2$$
$$q_2 \xrightarrow{b} q_f$$
$$(q_2,q_2) \xrightarrow{a} q_1$$



q_f ist in der Zustandsmenge der Wurzel enthalten

Nicht-Leerheitsproblem

Gegeben: Ein Automat A

Gesucht: Ist $L(A) \neq \emptyset$?

Gegeben: Ein Automat A

Gesucht: Ist $L(A) \neq \emptyset$?

Algorithmus für binär verzweigte Bäume:

Suche einen Weg zum akzeptierenden Zustand
(vgl. Erreichbarkeit in Graphen!)

Nicht-Leerheitsproblem

Gegeben: Ein Automat A

Gesucht: Ist $L(A) \neq \emptyset$?

Algorithmus für binär verzweigte Bäume:
  Suche einen Weg zum akzeptierenden Zustand
  (vgl. Erreichbarkeit in Graphen!)

Komplexität:
  $\rightarrow$ PTIME

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$ 001	100	$\xrightarrow{a}$ 011	(001,001)	$\xrightarrow{b}$ 001
001	$\xrightarrow{a}$ 001	011	$\xrightarrow{a}$ 011	(010,010)	$\xrightarrow{b}$ 010
001	$\xrightarrow{a}$ 010	011	$\xrightarrow{a}$ 101	(100,100)	$\xrightarrow{b}$ 100
001	$\xrightarrow{a}$ 100	011	$\xrightarrow{a}$ 110	(011,011)	$\xrightarrow{b}$ 011
010	$\xrightarrow{a}$ 001	101	$\xrightarrow{a}$ 011	(101,101)	$\xrightarrow{b}$ 101
010	$\xrightarrow{a}$ 010	101	$\xrightarrow{a}$ 101	(110,110)	$\xrightarrow{b}$ 110
010	$\xrightarrow{a}$ 100	101	$\xrightarrow{a}$ 110	(111,111)	$\xrightarrow{b}$ 101
100	$\xrightarrow{a}$ 001	110	$\xrightarrow{a}$ 011	(101,110)	$\xrightarrow{b}$ 111
100	$\xrightarrow{a}$ 010	110	$\xrightarrow{a}$ 101	(101,100)	$\xrightarrow{b}$ 110
100	$\xrightarrow{a}$ 100	110	$\xrightarrow{a}$ 110	(110,100)	$\xrightarrow{b}$ 100

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$ 001	100	$\xrightarrow{a}$ 011	(001,001)	$\xrightarrow{b}$ 001
001	$\xrightarrow{a}$ 001	011	$\xrightarrow{a}$ 011	(010,010)	$\xrightarrow{b}$ 010
001	$\xrightarrow{a}$ 010	011	$\xrightarrow{a}$ 101	(100,100)	$\xrightarrow{b}$ 100
001	$\xrightarrow{a}$ 100	011	$\xrightarrow{a}$ 110	(011,011)	$\xrightarrow{b}$ 011
010	$\xrightarrow{a}$ 001	101	$\xrightarrow{a}$ 011	(101,101)	$\xrightarrow{b}$ 101
010	$\xrightarrow{a}$ 010	101	$\xrightarrow{a}$ 101	(110,110)	$\xrightarrow{b}$ 110
010	$\xrightarrow{a}$ 100	101	$\xrightarrow{a}$ 110	(111,111)	$\xrightarrow{b}$ 101
100	$\xrightarrow{a}$ 001	110	$\xrightarrow{a}$ 011	(101,110)	$\xrightarrow{b}$ 111
100	$\xrightarrow{a}$ 010	110	$\xrightarrow{a}$ 101	(101,100)	$\xrightarrow{b}$ 110
100	$\xrightarrow{a}$ 100	110	$\xrightarrow{a}$ 110	(110,100)	$\xrightarrow{b}$ 100

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$ 001	100	$\xrightarrow{a}$ 011	(001,001)	$\xrightarrow{b}$ 001
001	$\xrightarrow{a}$ 001	011	$\xrightarrow{a}$ 011	(010,010)	$\xrightarrow{b}$ 010
001	$\xrightarrow{a}$ 010	011	$\xrightarrow{a}$ 101	(100,100)	$\xrightarrow{b}$ 100
001	$\xrightarrow{a}$ 100	011	$\xrightarrow{a}$ 110	(011,011)	$\xrightarrow{b}$ 011
010	$\xrightarrow{a}$ 001	101	$\xrightarrow{a}$ 011	(101,101)	$\xrightarrow{b}$ 101
010	$\xrightarrow{a}$ 010	101	$\xrightarrow{a}$ 101	(110,110)	$\xrightarrow{b}$ 110
010	$\xrightarrow{a}$ 100	101	$\xrightarrow{a}$ 110	(111,111)	$\xrightarrow{b}$ 101
100	$\xrightarrow{a}$ 001	110	$\xrightarrow{a}$ 011	(101,110)	$\xrightarrow{b}$ 111
100	$\xrightarrow{a}$ 010	110	$\xrightarrow{a}$ 101	(101,100)	$\xrightarrow{b}$ 110
100	$\xrightarrow{a}$ 100	110	$\xrightarrow{a}$ 110	(110,100)	$\xrightarrow{b}$ 100

Nicht-Leerheitsproblem

Erreicht dieser „Zähler" den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$		$100 \xrightarrow{a} 011$		$(001,001) \xrightarrow{b} 001$	
$001 \xrightarrow{a} 001$		$011 \xrightarrow{a} 011$		$(010,010) \xrightarrow{b} 010$	
$001 \xrightarrow{a} 010$		$011 \xrightarrow{a} 101$		$(100,100) \xrightarrow{b} 100$	
$001 \xrightarrow{a} 100$		$011 \xrightarrow{a} 110$		$(011,011) \xrightarrow{b} 011$	
$010 \xrightarrow{a} 001$		$101 \xrightarrow{a} 011$		$(101,101) \xrightarrow{b} 101$	
$010 \xrightarrow{a} 010$		$101 \xrightarrow{a} 101$		$(110,110) \xrightarrow{b} 110$	
$010 \xrightarrow{a} 100$		$101 \xrightarrow{a} 110$		$(111,111) \xrightarrow{b} 101$	
$100 \xrightarrow{a} 001$		$110 \xrightarrow{a} 011$		$(101,110) \xrightarrow{b} 111$	
$100 \xrightarrow{a} 010$		$110 \xrightarrow{a} 101$		$(101,100) \xrightarrow{b} 110$	
$100 \xrightarrow{a} 100$		$110 \xrightarrow{a} 110$		$(110,100) \xrightarrow{b} 100$	

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$ 001	100	$\xrightarrow{a}$ 011	(001,001)	$\xrightarrow{b}$ 001
001	$\xrightarrow{a}$ 001	011	$\xrightarrow{a}$ 011	(010,010)	$\xrightarrow{b}$ 010
001	$\xrightarrow{a}$ 010	011	$\xrightarrow{a}$ 101	(100,100)	$\xrightarrow{b}$ 100
001	$\xrightarrow{a}$ 100	011	$\xrightarrow{a}$ 110	(011,011)	$\xrightarrow{b}$ 011
010	$\xrightarrow{a}$ 001	101	$\xrightarrow{a}$ 011	(101,101)	$\xrightarrow{b}$ 101
010	$\xrightarrow{a}$ 010	101	$\xrightarrow{a}$ 101	(110,110)	$\xrightarrow{b}$ 110
010	$\xrightarrow{a}$ 100	101	$\xrightarrow{a}$ 110	(111,111)	$\xrightarrow{b}$ 101
100	$\xrightarrow{a}$ 001	110	$\xrightarrow{a}$ 011	(101,110)	$\xrightarrow{b}$ 111
100	$\xrightarrow{a}$ 010	110	$\xrightarrow{a}$ 101	(101,100)	$\xrightarrow{b}$ 110
100	$\xrightarrow{a}$ 100	110	$\xrightarrow{a}$ 110	(110,100)	$\xrightarrow{b}$ 100

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$	001	100	$\xrightarrow{a}$	011	(001,001)	$\xrightarrow{b}$	001
001	$\xrightarrow{a}$	001	011	$\xrightarrow{a}$	011	(010,010)	$\xrightarrow{b}$	010
001	$\xrightarrow{a}$	010	011	$\xrightarrow{a}$	101	(100,100)	$\xrightarrow{b}$	100
001	$\xrightarrow{a}$	100	011	$\xrightarrow{a}$	110	(011,011)	$\xrightarrow{b}$	011
010	$\xrightarrow{a}$	001	101	$\xrightarrow{a}$	011	(101,101)	$\xrightarrow{b}$	101
010	$\xrightarrow{a}$	010	101	$\xrightarrow{a}$	101	(110,110)	$\xrightarrow{b}$	110
010	$\xrightarrow{a}$	100	101	$\xrightarrow{a}$	110	(111,111)	$\xrightarrow{b}$	101
100	$\xrightarrow{a}$	001	110	$\xrightarrow{a}$	011	(101,110)	$\xrightarrow{b}$	111
100	$\xrightarrow{a}$	010	110	$\xrightarrow{a}$	101	(101,100)	$\xrightarrow{b}$	110
100	$\xrightarrow{a}$	100	110	$\xrightarrow{a}$	110	(110,100)	$\xrightarrow{b}$	100

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$		$100 \xrightarrow{a} 011$		$(001,001) \xrightarrow{b} 001$	
$001 \xrightarrow{a} 001$		$011 \xrightarrow{a} 011$		$(010,010) \xrightarrow{b} 010$	
$001 \xrightarrow{a} 010$		$011 \xrightarrow{a} 101$		$(100,100) \xrightarrow{b} 100$	
$001 \xrightarrow{a} 100$		$011 \xrightarrow{a} 110$		$(011,011) \xrightarrow{b} 011$	
$010 \xrightarrow{a} 001$		$101 \xrightarrow{a} 011$		$(101,101) \xrightarrow{b} 101$	
$010 \xrightarrow{a} 010$		$101 \xrightarrow{a} 101$		$(110,110) \xrightarrow{b} 110$	
$010 \xrightarrow{a} 100$		$101 \xrightarrow{a} 110$		$(111,111) \xrightarrow{b} 101$	
$100 \xrightarrow{a} 001$		$110 \xrightarrow{a} 011$		$(101,110) \xrightarrow{b} 111$	
$100 \xrightarrow{a} 010$		$110 \xrightarrow{a} 101$		$(101,100) \xrightarrow{b} 110$	
$100 \xrightarrow{a} 100$		$110 \xrightarrow{a} 110$		$(110,100) \xrightarrow{b} 100$	

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$ 001	100	$\xrightarrow{a}$ 011	(001,001)	$\xrightarrow{b}$ 001
001	$\xrightarrow{a}$ 001	011	$\xrightarrow{a}$ 011	(010,010)	$\xrightarrow{b}$ 010
001	$\xrightarrow{a}$ 010	011	$\xrightarrow{a}$ 101	(100,100)	$\xrightarrow{b}$ 100
001	$\xrightarrow{a}$ 100	011	$\xrightarrow{a}$ 110	(011,011)	$\xrightarrow{b}$ 011
010	$\xrightarrow{a}$ 001	101	$\xrightarrow{a}$ 011	(101,101)	$\xrightarrow{b}$ 101
010	$\xrightarrow{a}$ 010	101	$\xrightarrow{a}$ 101	(110,110)	$\xrightarrow{b}$ 110
010	$\xrightarrow{a}$ 100	101	$\xrightarrow{a}$ 110	(111,111)	$\xrightarrow{b}$ 101
100	$\xrightarrow{a}$ 001	110	$\xrightarrow{a}$ 011	(101,110)	$\xrightarrow{b}$ 111
100	$\xrightarrow{a}$ 010	110	$\xrightarrow{a}$ 101	(101,100)	$\xrightarrow{b}$ 110
100	$\xrightarrow{a}$ 100	110	$\xrightarrow{a}$ 110	(110,100)	$\xrightarrow{b}$ 100

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$	$100 \xrightarrow{a} 011$	$(001,001) \xrightarrow{b} 001$
$001 \xrightarrow{a} 001$	$011 \xrightarrow{a} 011$	$(010,010) \xrightarrow{b} 010$
$001 \xrightarrow{a} 010$	$011 \xrightarrow{a} 101$	$(100,100) \xrightarrow{b} 100$
$001 \xrightarrow{a} 100$	$011 \xrightarrow{a} 110$	$(011,011) \xrightarrow{b} 011$
$010 \xrightarrow{a} 001$	$101 \xrightarrow{a} 011$	$(101,101) \xrightarrow{b} 101$
$010 \xrightarrow{a} 010$	$101 \xrightarrow{a} 101$	$(110,110) \xrightarrow{b} 110$
$010 \xrightarrow{a} 100$	$101 \xrightarrow{a} 110$	$(111,111) \xrightarrow{b} 101$
$100 \xrightarrow{a} 001$	$110 \xrightarrow{a} 011$	$(101,110) \xrightarrow{b} 111$
$100 \xrightarrow{a} 010$	$110 \xrightarrow{a} 101$	$(101,100) \xrightarrow{b} 110$
$100 \xrightarrow{a} 100$	$110 \xrightarrow{a} 110$	$(110,100) \xrightarrow{b} 100$

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$		$100 \xrightarrow{a} 011$		$(001,001) \xrightarrow{b} 001$	
$001 \xrightarrow{a} 001$		$011 \xrightarrow{a} 011$		$(010,010) \xrightarrow{b} 010$	
$001 \xrightarrow{a} 010$		$011 \xrightarrow{a} 101$		$(100,100) \xrightarrow{b} 100$	
$001 \xrightarrow{a} 100$		$011 \xrightarrow{a} 110$		$(011,011) \xrightarrow{b} 011$	
$010 \xrightarrow{a} 001$		$101 \xrightarrow{a} 011$		$(101,101) \xrightarrow{b} 101$	
$010 \xrightarrow{a} 010$		$101 \xrightarrow{a} 101$		$(110,110) \xrightarrow{b} 110$	
$010 \xrightarrow{a} 100$		$101 \xrightarrow{a} 110$		$(111,111) \xrightarrow{b} 101$	
$100 \xrightarrow{a} 001$		$110 \xrightarrow{a} 011$		$(101,110) \xrightarrow{b} 111$	
$100 \xrightarrow{a} 010$		$110 \xrightarrow{a} 101$		$(101,100) \xrightarrow{b} 110$	
$100 \xrightarrow{a} 100$		$110 \xrightarrow{a} 110$		$(110,100) \xrightarrow{b} 100$	

Erreicht dieser „Zähler" den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$		$100 \xrightarrow{a} 011$		$(001,001) \xrightarrow{b} 001$	
$001 \xrightarrow{a} 001$		$011 \xrightarrow{a} 011$		$(010,010) \xrightarrow{b} 010$	
$001 \xrightarrow{a} 010$		$011 \xrightarrow{a} 101$		$(100,100) \xrightarrow{b} 100$	
$001 \xrightarrow{a} 100$		$011 \xrightarrow{a} 110$		$(011,011) \xrightarrow{b} 011$	
$010 \xrightarrow{a} 001$		$101 \xrightarrow{a} 011$		$(101,101) \xrightarrow{b} 101$	
$010 \xrightarrow{a} 010$		$101 \xrightarrow{a} 101$		$(110,110) \xrightarrow{b} 110$	
$010 \xrightarrow{a} 100$		$101 \xrightarrow{a} 110$		$(111,111) \xrightarrow{b} 101$	
$100 \xrightarrow{a} 001$		$110 \xrightarrow{a} 011$		$(101,110) \xrightarrow{b} 111$	
$100 \xrightarrow{a} 010$		$110 \xrightarrow{a} 101$		$(101,100) \xrightarrow{b} 110$	
$100 \xrightarrow{a} 100$		$110 \xrightarrow{a} 110$		$(110,100) \xrightarrow{b} 100$	

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$		$100 \xrightarrow{a} 011$		$(001,001) \xrightarrow{b} 001$	
$001 \xrightarrow{a} 001$		$011 \xrightarrow{a} 011$		$(010,010) \xrightarrow{b} 010$	
$001 \xrightarrow{a} 010$		$011 \xrightarrow{a} 101$		$(100,100) \xrightarrow{b} 100$	
$001 \xrightarrow{a} 100$		$011 \xrightarrow{a} 110$		$(011,011) \xrightarrow{b} 011$	
$010 \xrightarrow{a} 001$		$101 \xrightarrow{a} 011$		$(101,101) \xrightarrow{b} 101$	
$010 \xrightarrow{a} 010$		$101 \xrightarrow{a} 101$		$(110,110) \xrightarrow{b} 110$	
$010 \xrightarrow{a} 100$		$101 \xrightarrow{a} 110$		$(111,111) \xrightarrow{b} 101$	
$100 \xrightarrow{a} 001$		$110 \xrightarrow{a} 011$		$(101,110) \xrightarrow{b} 111$	
$100 \xrightarrow{a} 010$		$110 \xrightarrow{a} 101$		$(101,100) \xrightarrow{b} 110$	
$100 \xrightarrow{a} 100$		$110 \xrightarrow{a} 110$		$(110,100) \xrightarrow{b} 100$	

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$	$100 \xrightarrow{a} 011$	$(001,001) \xrightarrow{b} 001$
$001 \xrightarrow{a} 001$	$011 \xrightarrow{a} 011$	$(010,010) \xrightarrow{b} 010$
$001 \xrightarrow{a} 010$	$011 \xrightarrow{a} 101$	$(100,100) \xrightarrow{b} 100$
$001 \xrightarrow{a} 100$	$011 \xrightarrow{a} 110$	$(011,011) \xrightarrow{b} 011$
$010 \xrightarrow{a} 001$	$101 \xrightarrow{a} 011$	$(101,101) \xrightarrow{b} 101$
$010 \xrightarrow{a} 010$	$101 \xrightarrow{a} 101$	$(110,110) \xrightarrow{b} 110$
$010 \xrightarrow{a} 100$	$101 \xrightarrow{a} 110$	$(111,111) \xrightarrow{b} 101$
$100 \xrightarrow{a} 001$	$110 \xrightarrow{a} 011$	$(101,110) \xrightarrow{b} 111$
$100 \xrightarrow{a} 010$	$110 \xrightarrow{a} 101$	$(101,100) \xrightarrow{b} 110$
$100 \xrightarrow{a} 100$	$110 \xrightarrow{a} 110$	$(110,100) \xrightarrow{b} 100$

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$	$100 \xrightarrow{a} 011$	$(001,001) \xrightarrow{b} 001$
$001 \xrightarrow{a} 001$	$011 \xrightarrow{a} 011$	$(010,010) \xrightarrow{b} 010$
$001 \xrightarrow{a} 010$	$011 \xrightarrow{a} 101$	$(100,100) \xrightarrow{b} 100$
$001 \xrightarrow{a} 100$	$011 \xrightarrow{a} 110$	$(011,011) \xrightarrow{b} 011$
$010 \xrightarrow{a} 001$	$101 \xrightarrow{a} 011$	$(101,101) \xrightarrow{b} 101$
$010 \xrightarrow{a} 010$	$101 \xrightarrow{a} 101$	$(110,110) \xrightarrow{b} 110$
$010 \xrightarrow{a} 100$	$101 \xrightarrow{a} 110$	$(111,111) \xrightarrow{b} 101$
$100 \xrightarrow{a} 001$	$110 \xrightarrow{a} 011$	$(101,110) \xrightarrow{b} 111$
$100 \xrightarrow{a} 010$	$110 \xrightarrow{a} 101$	$(101,100) \xrightarrow{b} 110$
$100 \xrightarrow{a} 100$	$110 \xrightarrow{a} 110$	$(110,100) \xrightarrow{b} 100$

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

$\varepsilon \xrightarrow{a} 001$	$100 \xrightarrow{a} 011$	$(001,001) \xrightarrow{b} 001$
$001 \xrightarrow{a} 001$	$011 \xrightarrow{a} 011$	$(010,010) \xrightarrow{b} 010$
$001 \xrightarrow{a} 010$	$011 \xrightarrow{a} 101$	$(100,100) \xrightarrow{b} 100$
$001 \xrightarrow{a} 100$	$011 \xrightarrow{a} 110$	$(011,011) \xrightarrow{b} 011$
$010 \xrightarrow{a} 001$	$101 \xrightarrow{a} 011$	$(101,101) \xrightarrow{b} 101$
$010 \xrightarrow{a} 010$	$101 \xrightarrow{a} 101$	$(110,110) \xrightarrow{b} 110$
$010 \xrightarrow{a} 100$	$101 \xrightarrow{a} 110$	$(111,111) \xrightarrow{b} 101$
$100 \xrightarrow{a} 001$	$110 \xrightarrow{a} 011$	$(101,110) \xrightarrow{b} 111$
$100 \xrightarrow{a} 010$	$110 \xrightarrow{a} 101$	$(101,100) \xrightarrow{b} 110$
$100 \xrightarrow{a} 100$	$110 \xrightarrow{a} 110$	$(110,100) \xrightarrow{b} 100$

Nicht-Leerheitsproblem

Erreicht dieser „Zähler“ den Finalzustand 111?

ε	$\xrightarrow{a}$	001	100	$\xrightarrow{a}$	011	(001,001)	$\xrightarrow{b}$	001
001	$\xrightarrow{a}$	001	011	$\xrightarrow{a}$	011	(010,010)	$\xrightarrow{b}$	010
001	$\xrightarrow{a}$	010	011	$\xrightarrow{a}$	101	(100,100)	$\xrightarrow{b}$	100
001	$\xrightarrow{a}$	100	011	$\xrightarrow{a}$	110	(011,011)	$\xrightarrow{b}$	011
010	$\xrightarrow{a}$	001	101	$\xrightarrow{a}$	011	(101,101)	$\xrightarrow{b}$	101
010	$\xrightarrow{a}$	010	101	$\xrightarrow{a}$	101	(110,110)	$\xrightarrow{b}$	110
010	$\xrightarrow{a}$	100	101	$\xrightarrow{a}$	110	(111,111)	$\xrightarrow{b}$	101
100	$\xrightarrow{a}$	001	110	$\xrightarrow{a}$	011	→ (101,110)	$\xrightarrow{b}$	111
100	$\xrightarrow{a}$	010	110	$\xrightarrow{a}$	101	(101,100)	$\xrightarrow{b}$	110
100	$\xrightarrow{a}$	100	110	$\xrightarrow{a}$	110	(110,100)	$\xrightarrow{b}$	100

→ Es gibt einen Weg zu 111

Was gilt für unbeschränkt verzweigte Automaten?

Algorithmen für binäre Baumautomaten sind in PTIME

Was gilt für unbeschränkt verzweigte Automaten?

Algorithmen für binäre Baumautomaten sind in PTIME
→ über *ext*-Operator können unbeschränkt verzweigte Automaten binär kodiert werden

Was gilt für unbeschränkt verzweigte Automaten?

Algorithmen für binäre Baumautomaten sind in PTIME

→ über *ext*-Operator können unbeschränkt verzweigte Automaten binär kodiert werden

→ *ext*-Kodierung überträgt die Problemstellung:

Was gilt für unbeschränkt verzweigte Automaten?

Algorithmen für binäre Baumautomaten sind in PTIME

$\longrightarrow$ über *ext*-Operator können unbeschränkt verzweigte Automaten binär kodiert werden

$\longrightarrow$ *ext*-Kodierung überträgt die Problemstellung:

- $L(A) \neq \emptyset \quad \Leftrightarrow ext(L(A)) \neq \emptyset$
- $t \in L(A) \quad \Leftrightarrow ext(t) \in ext(L(A))$

Was gilt für unbeschränkt verzweigte Automaten?

Algorithmen für binäre Baumautomaten sind in PTIME

→ über *ext*-Operator können unbeschränkt verzweigte Automaten binär kodiert werden

→ *ext*-Kodierung überträgt die Problemstellung:

- $L(A) \neq \emptyset \quad \Leftrightarrow ext(L(A)) \neq \emptyset$
- $t \in L(A) \quad \Leftrightarrow ext(t) \in ext(L(A))$

→ Komplexität der Kodierung ist auch in PTIME

Was gilt für unbeschränkt verzweigte Automaten?

Algorithmen für binäre Baumautomaten sind in PTIME

$\rightarrow$ über *ext*-Operator können unbeschränkt verzweigte Automaten binär kodiert werden

$\rightarrow$ *ext*-Kodierung überträgt die Problemstellung:

- $L(A) \neq \emptyset \quad \Leftrightarrow ext(L(A)) \neq \emptyset$
- $t \in L(A) \quad \Leftrightarrow ext(t) \in ext(L(A))$

$\rightarrow$ Komplexität der Kodierung ist auch in PTIME

=> Algorithmen behalten ihre Komplexität und gelten auch für unbeschränkt verzweigte Bäume

Endlichkeit:

Gegeben: Ein Automat A

Gesucht: Ist L(A) endlich?

Endlichkeit:

Gegeben: Ein Automat A
Gesucht: Ist L(A) endlich?

Ein Algorithmus:
→ Finde nützliche Zustände, d.h. Zustände, die in einem akzeptierenden Lauf erscheinen
→ Suche nach Schleifen

Komplexität:
→ PTIME

Universalität:

Gegeben: Ein Automat A

Gesucht: Ist L(A) die Sammlung aller binär verzweigten Bäume über das Alphabet (A)?

Universalität:

Gegeben: Ein Automat A
Gesucht: Ist L(A) die Sammlung aller binär verzweigten Bäume über das Alphabet (A)?

Ein Algorithmus:
→ Determinisiere A
→ Vervollständige und bilde Komplement
→ Untersuche Leerheit

Komplexität:
→ EXPTIME für NFTA
→ PTIME für DFTA

Enthaltensein:

Gegeben: Zwei Automaten A und B

Gesucht: Ist $L(A) \subseteq L(B)$?

Weitere Probleme

Enthaltensein:

Gegeben: Zwei Automaten A und B
Gesucht: Ist $L(A) \subseteq L(B)$?

Betrachte folgendes Problem:
→ $L(A) \cap \overline{L(B)} = \emptyset$?

Komplexität:
→ EXPTIME für NFTA
→ PTIME für DFTA

Gleichheit:

Gegeben: Zwei Automaten A und B

Gesucht: Ist L(A)=L(B)?

Gleichheit:

Gegeben: Zwei Automaten A und B
Gesucht: Ist $L(A)=L(B)$?

Betrachte folgende Probleme:

→ $L(A)\subseteq L(B)$?
→ $L(B)\subseteq L(A)$?

Komplexität:

→ EXPTIME für NFTA
→ PTIME für DFTA

<u>Leerheit des Schnittes:</u>

Gegeben: Automaten $A_1,...,A_n$

Gesucht: Ist der Schnitt aller Automaten $\emptyset$?

Leerheit des Schnittes:

Gegeben: Automaten $A_1,...,A_n$

Gesucht: Ist der Schnitt aller Automaten $\emptyset$?

Ein Algorithmus:

→ Bilde Produktautomaten

→ Untersuche Leerheit

Komplexität:

→ EXPTIME

Zusammenfassung

→ Binäre und unbeschränkt verzweigte Baumautomaten sind unter Vereinigung, Schnitt und Komplement abgeschlossen

→ Extension Operator Kodierung als Mittel zur Darstellung von unbeschränkt verzweigten Baumautomaten durch binär verzweigte

→ Algorithmen und Komplexität einiger Probleme:

- Basistermproblem
- Nicht-Leerheitsproblem
  - Endlichkeit
  - Universalität
  - Enthaltensein
  - Gleichheit
  - Leerheit des Schnittes

Für formale Erläuterungen und Beweise siehe:

Hubert Comon, et al.
Tree Automata Techniques and Applications, 2007.
https://gforge.inria.fr/frs/download.php/10994/tata.pdf

Vielen Dank für die Aufmerksamkeit!