

Übung *Logik und Komplexität*

Jens Keppeler

22. April 2015

Binärbäume

- ▶ Ein **gewurzelter Baum** G ist ein endlicher gerichteter Graph, der als ungerichteter Graph ein Baum ist (d.h. der ungerichtete Graph, der aus dem gerichteten Graphen G entsteht, indem jede Kante durch eine ungerichtete Kante ersetzt wird, ist ein Baum) und der einen **Wurzelknoten** enthält, von dem aus jeder andere Knoten über einen gerichteten Pfad erreichbar ist.

Binärbäume

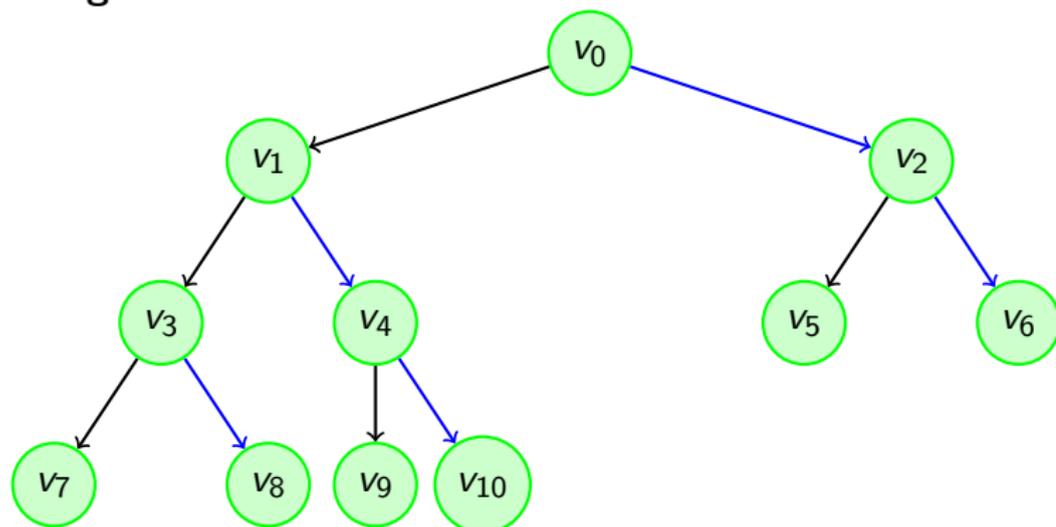
- ▶ Ein **gewurzelter Baum** G ist ein endlicher gerichteter Graph, der als ungerichteter Graph ein Baum ist (d.h. der ungerichtete Graph, der aus dem gerichteten Graphen G entsteht, indem jede Kante durch eine ungerichtete Kante ersetzt wird, ist ein Baum) und der einen **Wurzelknoten** enthält, von dem aus jeder andere Knoten über einen gerichteten Pfad erreichbar ist.
- ▶ Wenn jeder Knoten von G entweder ein Blatt ist oder genau zwei Kinder besitzt, so heißt G **voll**.

Binärbäume

- ▶ Ein **gewurzelter Baum** G ist ein endlicher gerichteter Graph, der als ungerichteter Graph ein Baum ist (d.h. der ungerichtete Graph, der aus dem gerichteten Graphen G entsteht, indem jede Kante durch eine ungerichtete Kante ersetzt wird, ist ein Baum) und der einen **Wurzelknoten** enthält, von dem aus jeder andere Knoten über einen gerichteten Pfad erreichbar ist.
- ▶ Wenn jeder Knoten von G entweder ein Blatt ist oder genau zwei Kinder besitzt, so heißt G **voll**.
- ▶ Ein voller Baum, dessen Kantenmenge E in $E_1 \dot{\cup} E_2$ partitioniert ist, so dass jeder Knoten, der kein Blatt ist, genau ein Kind in E_1 (sein **erstes Kind**) und ein Kind in E_2 (sein **zweites Kind**) besitzt, heißt **geordneter Binärbaum**.

Binärbäume (Beispiel)

Ein geordneter Binärbaum:



E_1 -Kanten: \longrightarrow

E_2 -Kanten: \longrightarrow

Σ -Bäume

Sei Σ ein endliches (nicht-leeres) Alphabet.

- ▶ Ein Σ -**Baum** $t = (B, \lambda)$ besteht aus einem geordneten Binärbaum B und einer Abbildung λ , die jedem Knoten v von B eine **Beschriftung** $\lambda(v) \in \Sigma$ zuordnet.

Σ -Bäume

Sei Σ ein endliches (nicht-leeres) Alphabet.

- ▶ Ein Σ -**Baum** $t = (B, \lambda)$ besteht aus einem geordneten Binärbaum B und einer Abbildung λ , die jedem Knoten v von B eine **Beschriftung** $\lambda(v) \in \Sigma$ zuordnet.
- ▶ Die Menge aller Σ -Bäume bezeichnen wir mit T_Σ .

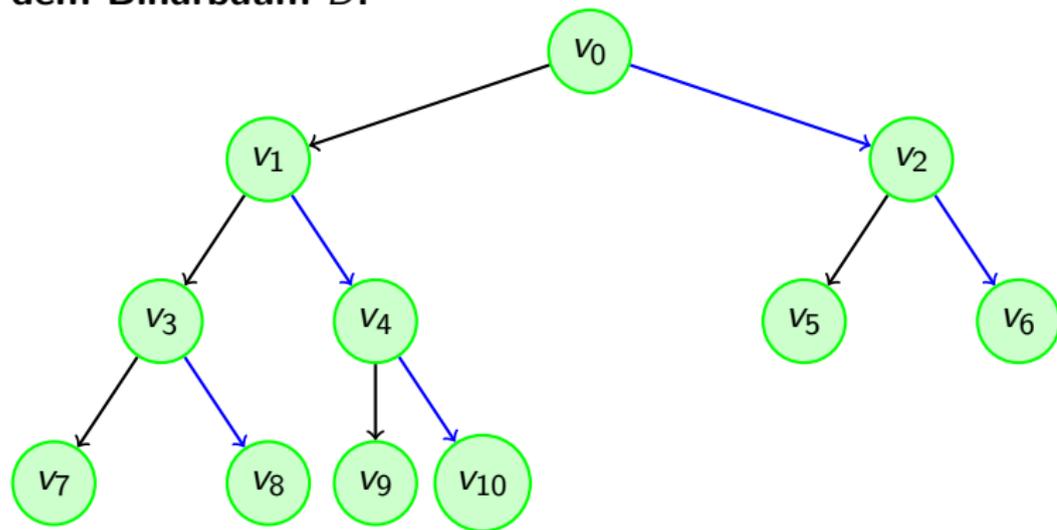
Σ -Bäume

Sei Σ ein endliches (nicht-leeres) Alphabet.

- ▶ Ein **Σ -Baum** $t = (B, \lambda)$ besteht aus einem geordneten Binärbaum B und einer Abbildung λ , die jedem Knoten v von B eine **Beschriftung** $\lambda(v) \in \Sigma$ zuordnet.
- ▶ Die Menge aller Σ -Bäume bezeichnen wir mit T_Σ .
- ▶ Eine **Baumsprache** ist eine Teilmenge L von T_Σ .

Σ -Bäume (Beispiel)

Sei $\Sigma = \{0, 1\}$. Sei $t = (B, \lambda)$ ein Σ -Baum mit dem Binärbaum B :



und

$\lambda(v_1) := \lambda(v_3) := \lambda(v_4) := \lambda(v_5) := \lambda(v_{10}) = 0$ und

$\lambda(v_0) := \lambda(v_2) := \lambda(v_6) := \lambda(v_7) := \lambda(v_8) := \lambda(v_9) := 1$

Baumautomaten

Ein (bottom-up) **Baumautomat** ist ein Tupel $\mathbb{A} := (Q, \Sigma, \Delta, F)$,
wobei

- ▶ Q eine endliche Menge, **Zustandsmenge**
- ▶ Σ ein endliches Alphabet,
- ▶ $F \subseteq Q$ **Menge der akzeptierenden Zustände**
- ▶ $\Delta \subseteq (Q^2 \cup \{\perp\}) \times \Sigma \times Q$ **Überföhrungsrelation**
eine Relation

ist.

Baumautomaten

Ein (bottom-up) **Baumautomat** ist ein Tupel $\mathbb{A} := (Q, \Sigma, \Delta, F)$,
wobei

- ▶ Q eine endliche Menge, **Zustandsmenge**
- ▶ Σ ein endliches Alphabet,
- ▶ $F \subseteq Q$ **Menge der akzeptierenden Zustände**
- ▶ $\Delta \subseteq (Q^2 \cup \{\perp\}) \times \Sigma \times Q$ **Überföhrungsrelation**
eine Relation

ist.

Falls Δ der Graph einer Abbildung ist, die auf ganz $(Q^2 \cup \{\perp\}) \times \Sigma$ definiert ist, so heißt \mathbb{A} **deterministisch**,
ansonsten heißt \mathbb{A} **nichtdeterministisch**.

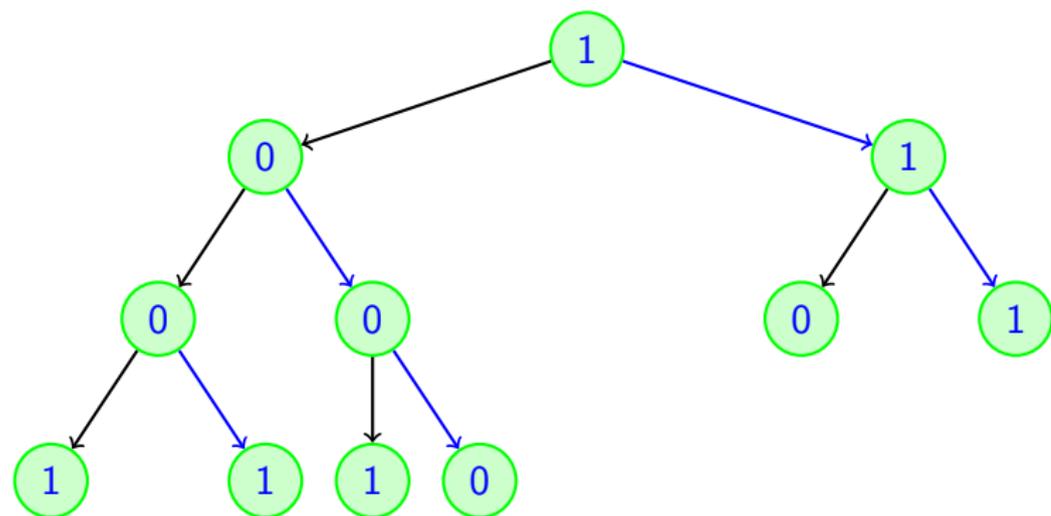
Baumautomaten (Beispiel)

Sei $\mathbb{A} := (Q, \Sigma, \Delta, F)$ ein Baumautomat mit

- ▶ $Q = \{q_0, q_1\}$
- ▶ $\Sigma = \{0, 1\}$
- ▶ $F = \{q_0\}$
- ▶

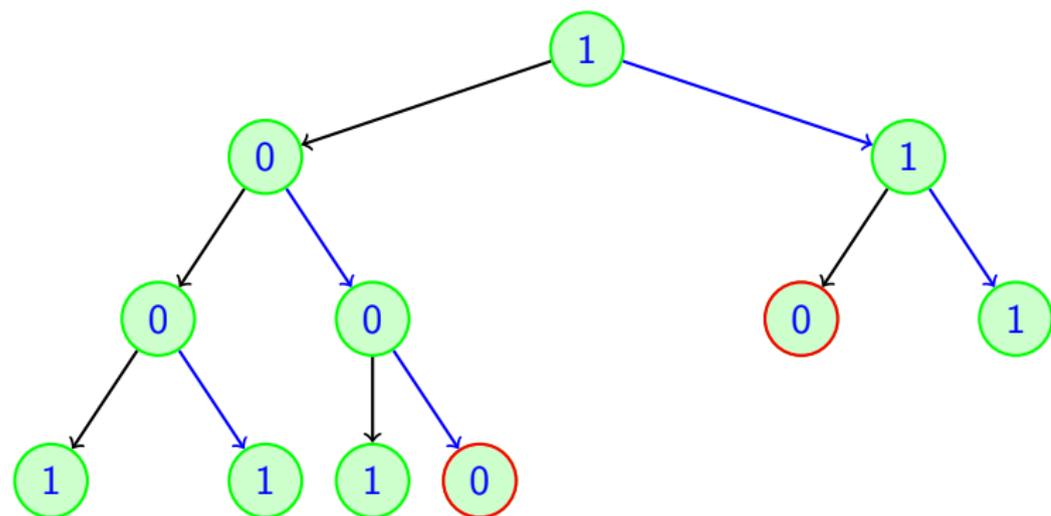
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



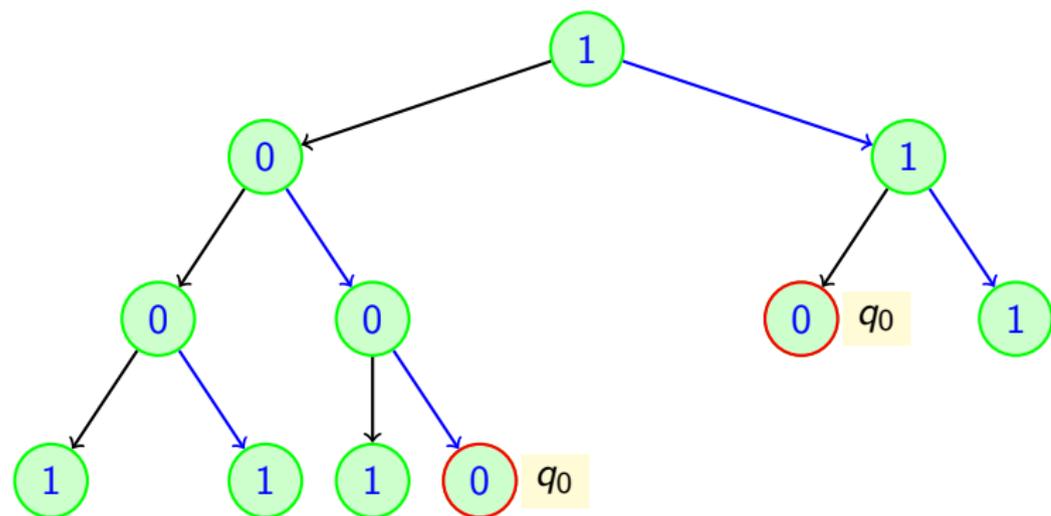
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



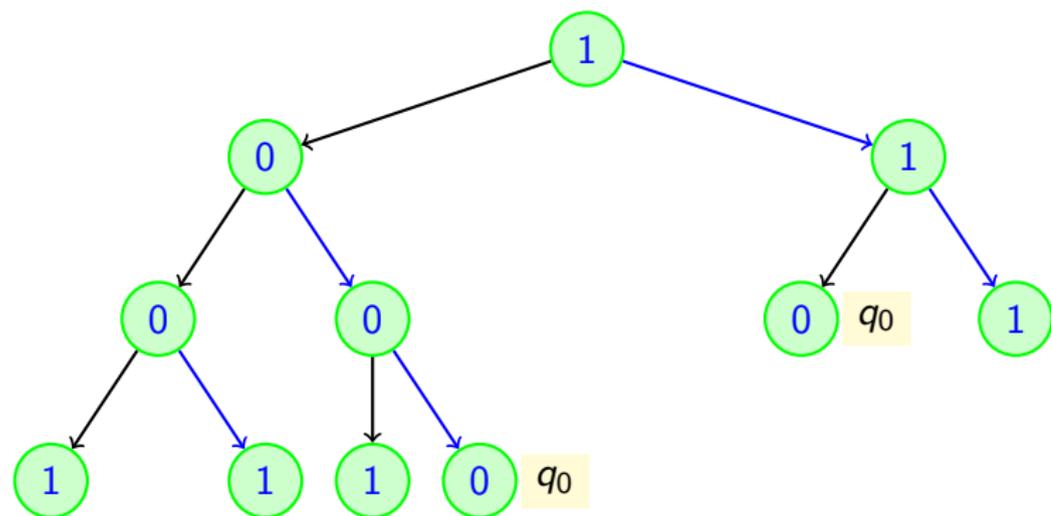
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



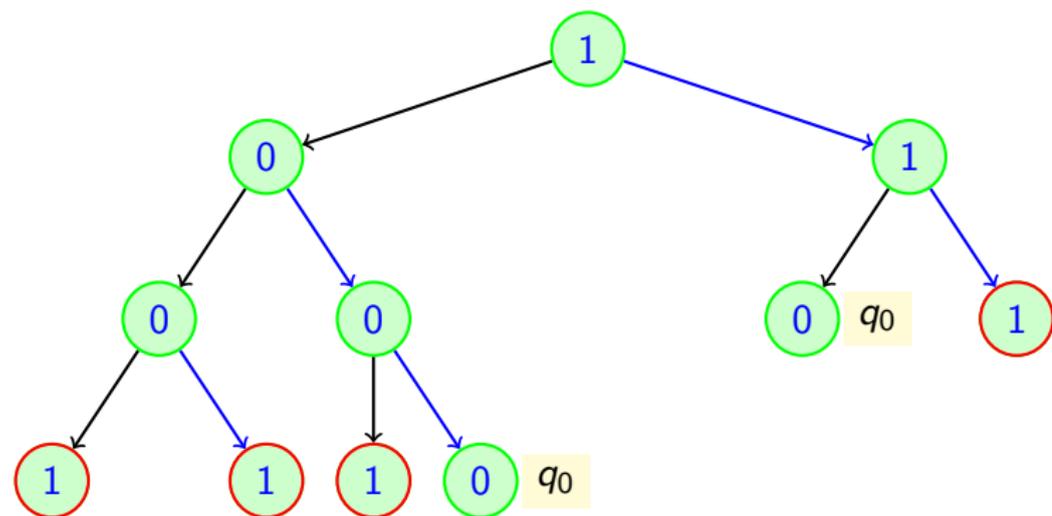
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



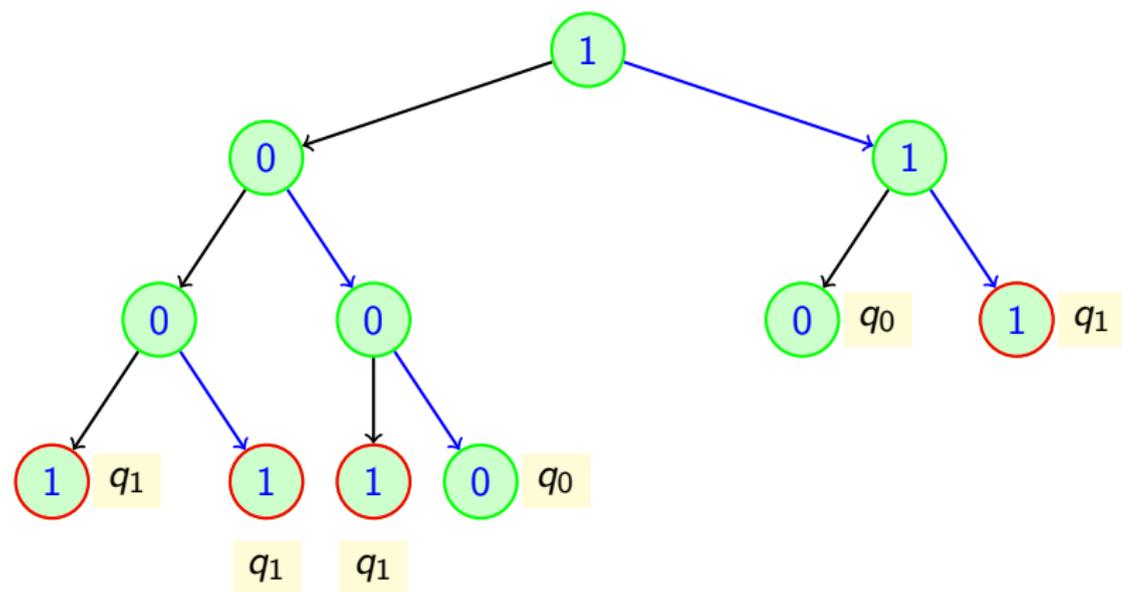
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



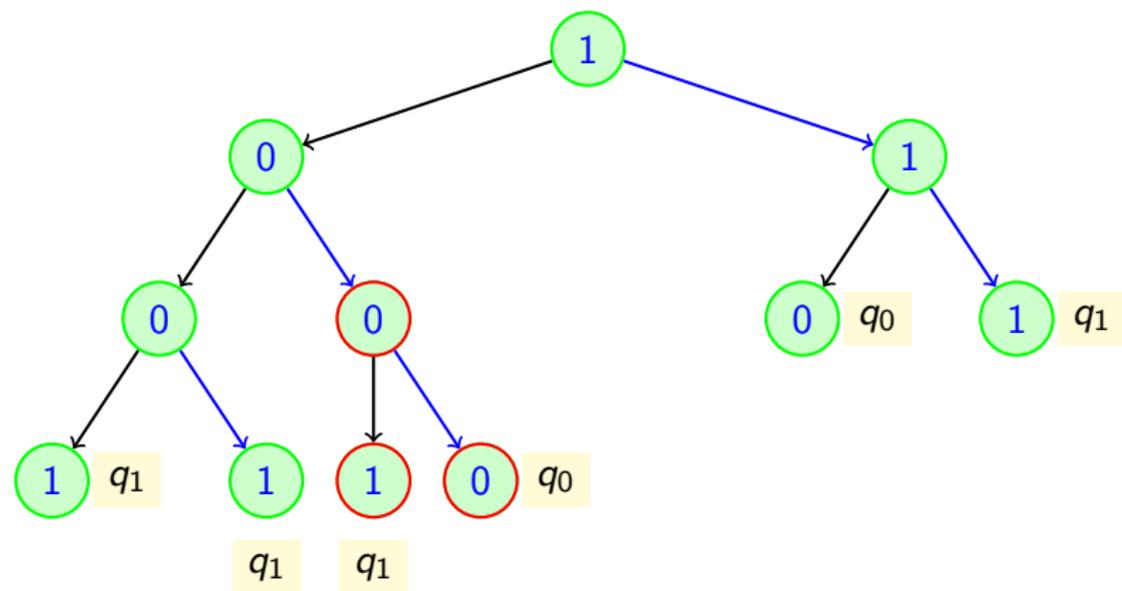
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



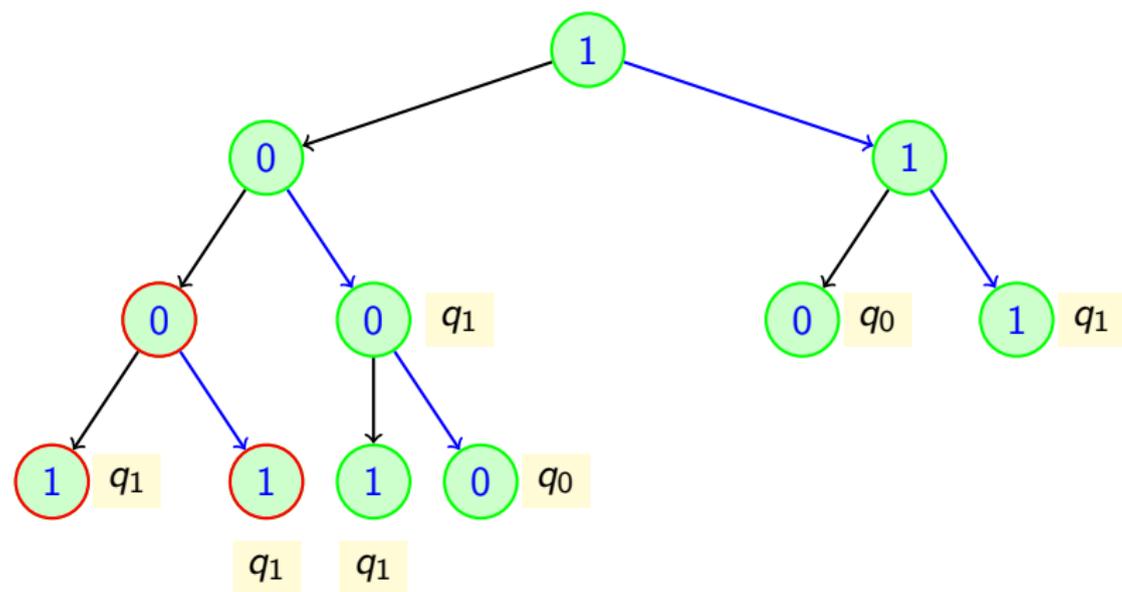
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



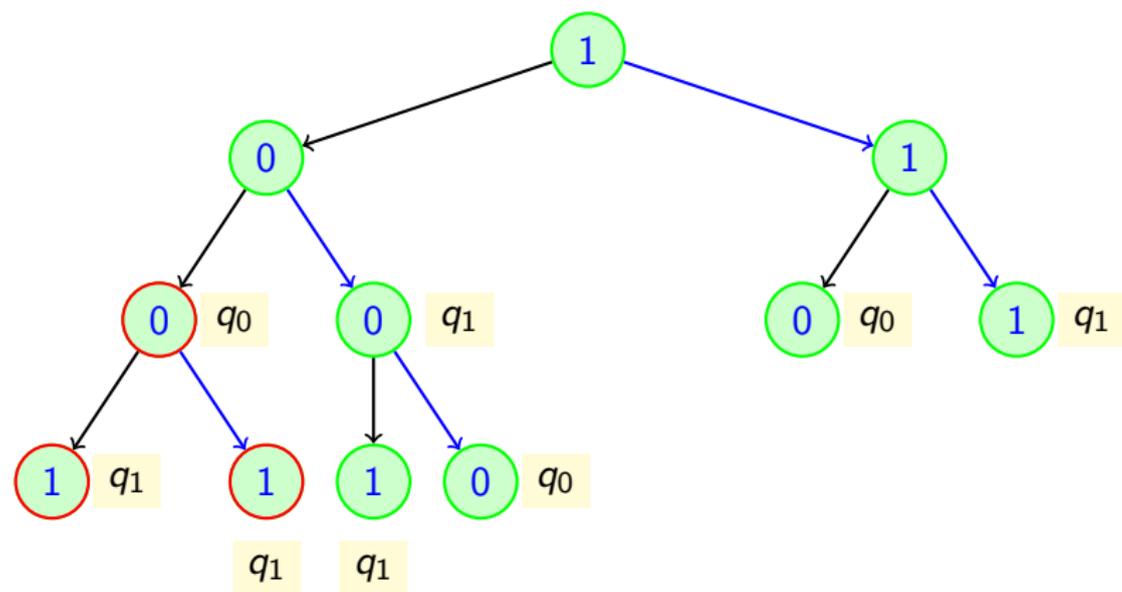
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



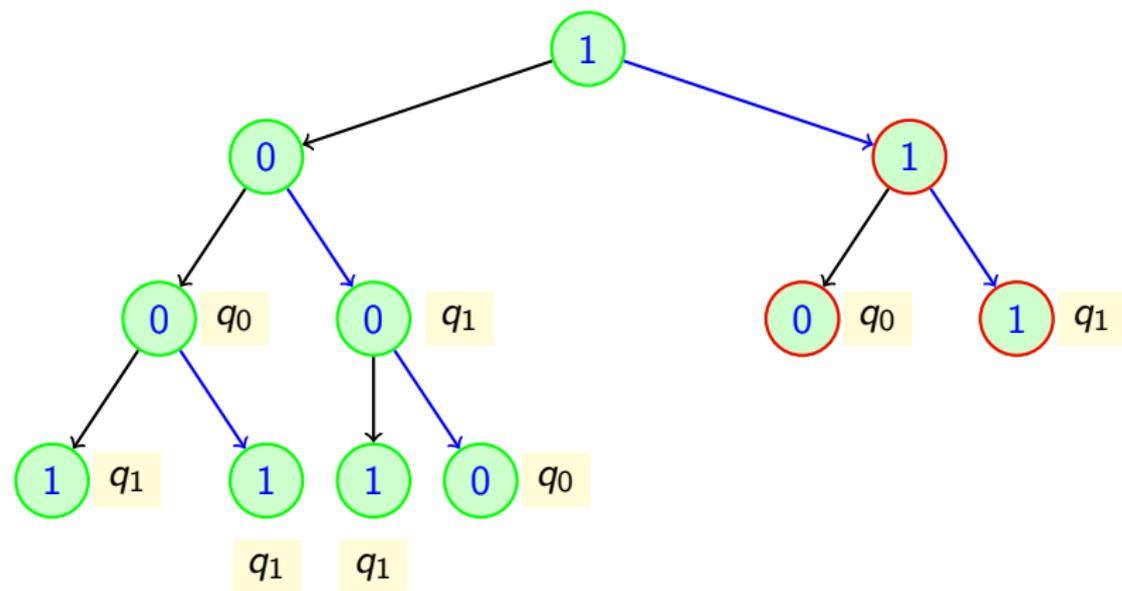
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



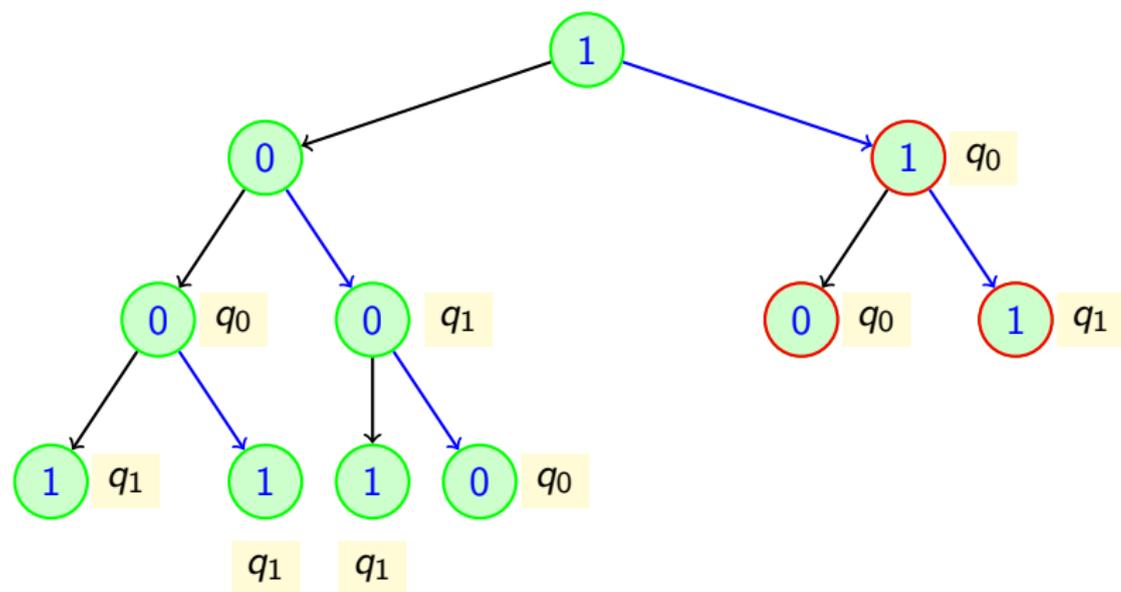
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



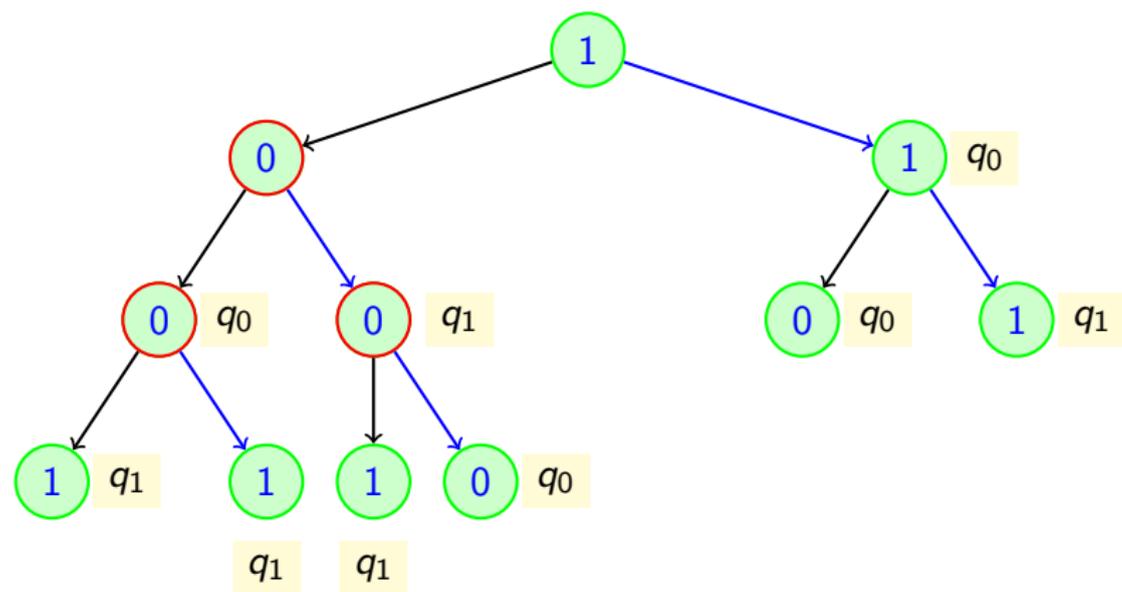
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



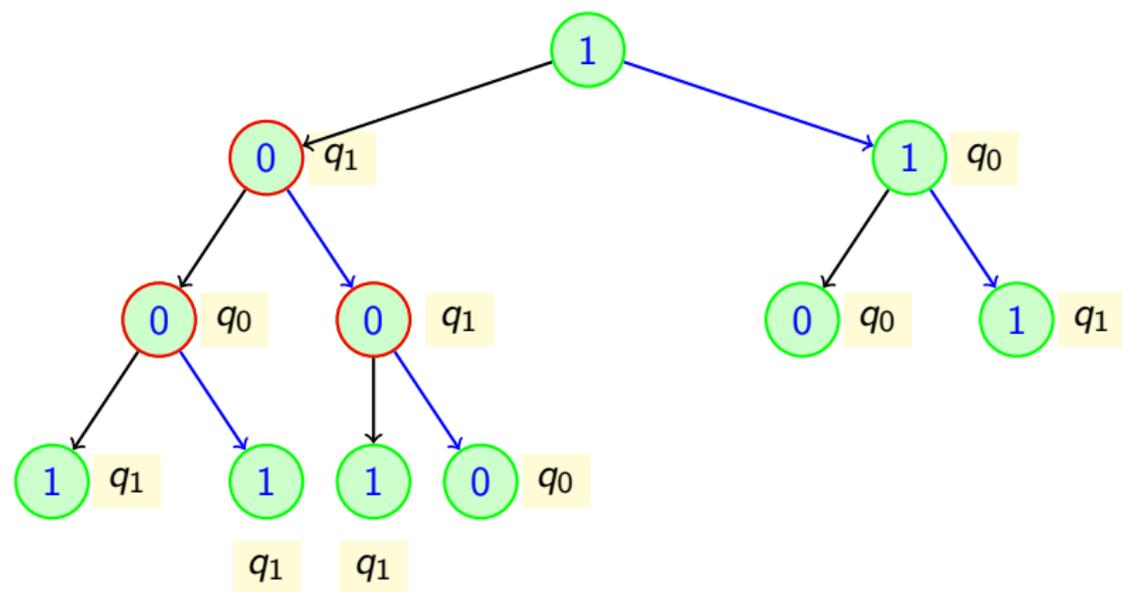
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



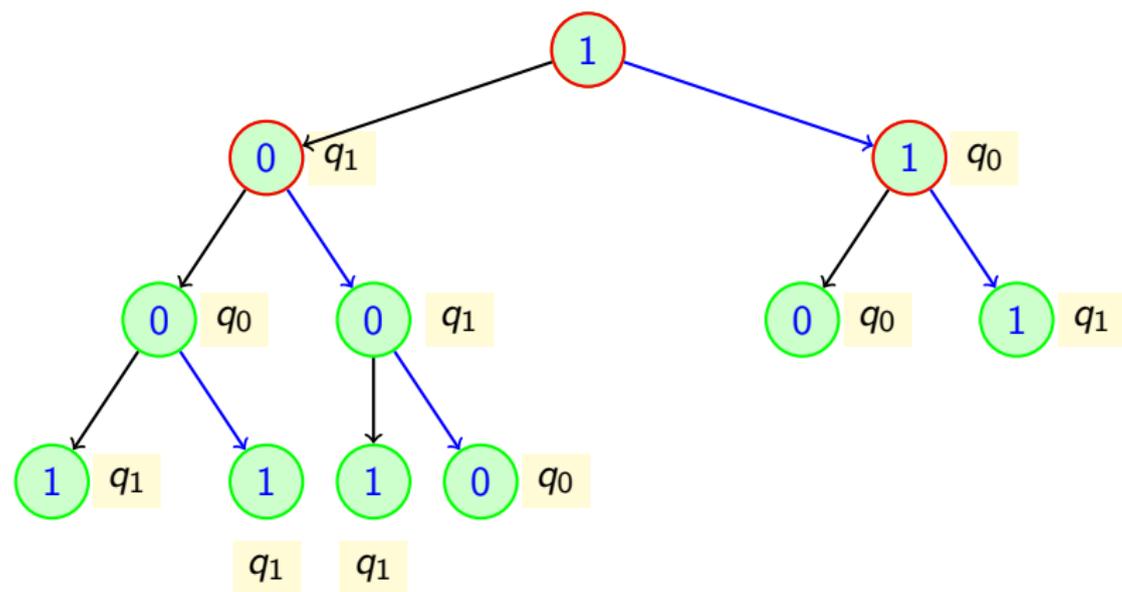
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



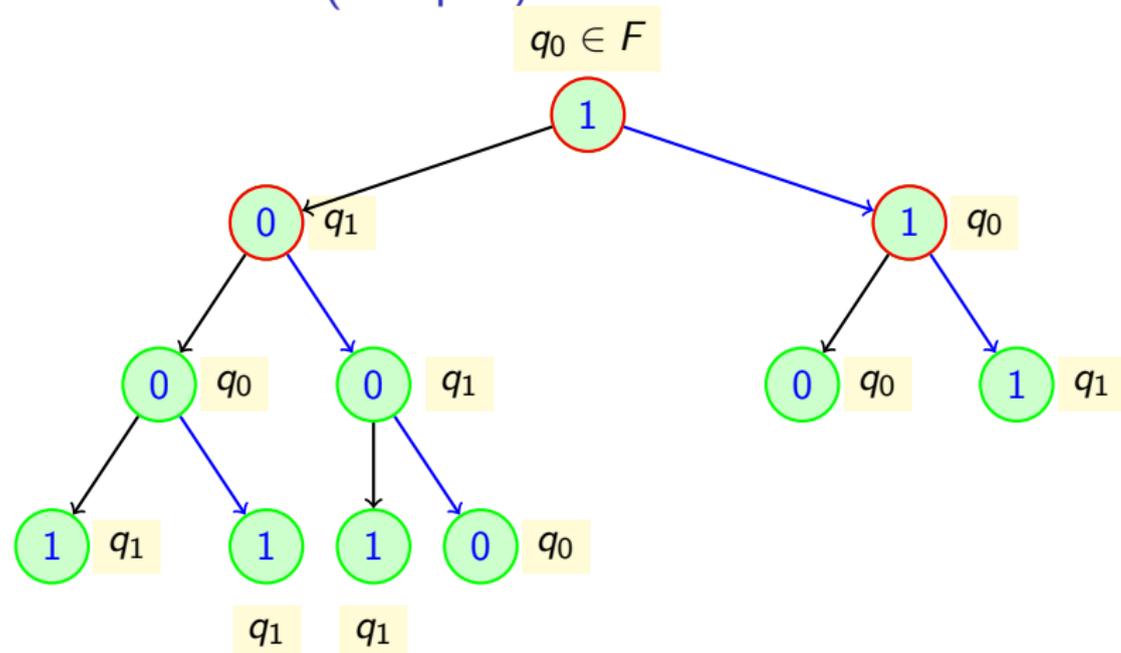
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



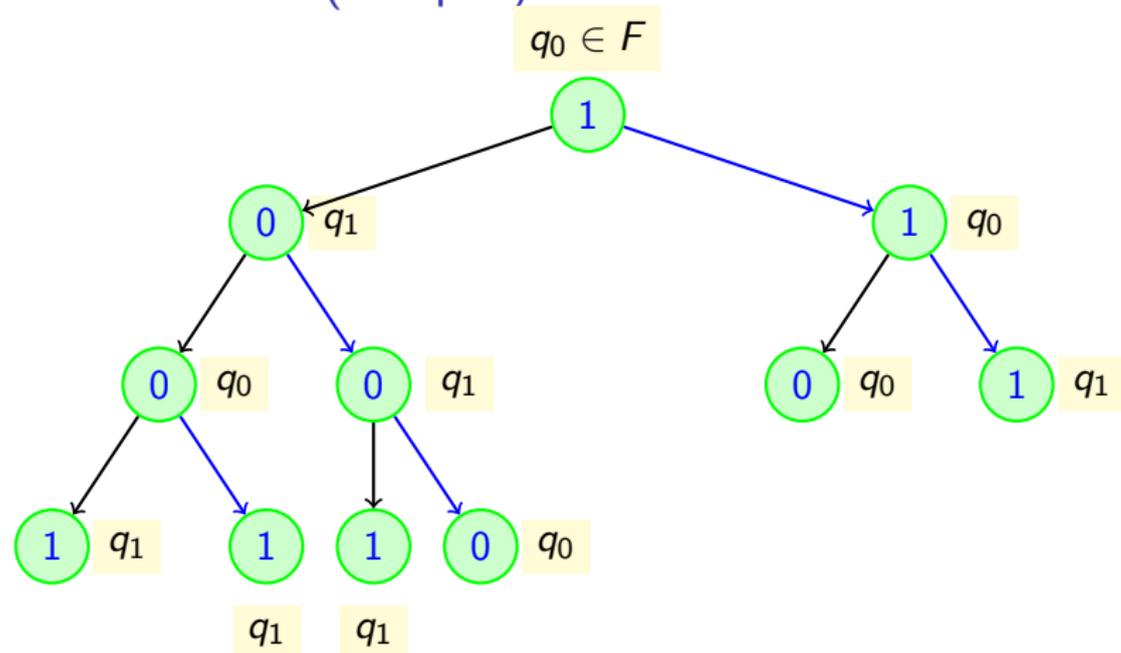
$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)



$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten (Beispiel)

Hinweis: Oft kann es praktisch sein, die Zustandsrelation kompakt aufzuschreiben. Zum Beispiel:

$$\Delta = \{(\perp, i, q_i) : i \in \{0, 1\}\} \cup \\ \{(q_i, q_j, \ell, q_k) : i, j, \ell, k \in \{0, 1\}, i + j + \ell \equiv k \pmod{2}\}$$

anstatt

$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

Baumautomaten

Sei $V(t)$ die Knotenmenge des Binärbaums von t .

- ▶ Der Baumautomat \mathbb{A} baut eine Funktion $q : V(t) \rightarrow Q$ auf, die als **Lauf von \mathbb{A} auf t** bezeichnet wird.

Baumautomaten

Sei $V(t)$ die Knotenmenge des Binärbaums von t .

- ▶ Der Baumautomat \mathbb{A} baut eine Funktion $q : V(t) \rightarrow Q$ auf, die als **Lauf von \mathbb{A} auf t** bezeichnet wird.
- ▶ Zunächst wird jedem Blatt v mit Beschriftung $a \in \Sigma$ ein Zustand $q(v) \in Q$ mit $(\perp, a, q(v)) \in \Delta$ zugewiesen.

Baumautomaten

Sei $V(t)$ die Knotenmenge des Binärbaums von t .

- ▶ Der Baumautomat \mathbb{A} baut eine Funktion $q : V(t) \rightarrow Q$ auf, die als **Lauf von \mathbb{A} auf t** bezeichnet wird.
- ▶ Zunächst wird jedem Blatt v mit Beschriftung $a \in \Sigma$ ein Zustand $q(v) \in Q$ mit $(\perp, a, q(v)) \in \Delta$ zugewiesen.
- ▶ Nun bestimmt \mathbb{A} rekursiv den Zustand $q(v)$ eines mit $a \in \Sigma$ gefärbten Knotens v , der kein Blatt ist, aus den Zuständen seines ersten und zweiten Kindes u_1 und u_2 . Dazu weist \mathbb{A} dem Knoten v einen Zustand $q(v) \in Q$ mit $(q(u_1), q(u_2), a, q(v)) \in \Delta$ zu.

Baumautomaten

Sei $V(t)$ die Knotenmenge des Binärbaums von t .

- ▶ Der Baumautomat \mathbb{A} baut eine Funktion $q : V(t) \rightarrow Q$ auf, die als **Lauf von \mathbb{A} auf t** bezeichnet wird.
- ▶ Zunächst wird jedem Blatt v mit Beschriftung $a \in \Sigma$ ein Zustand $q(v) \in Q$ mit $(\perp, a, q(v)) \in \Delta$ zugewiesen.
- ▶ Nun bestimmt \mathbb{A} rekursiv den Zustand $q(v)$ eines mit $a \in \Sigma$ gefärbten Knotens v , der kein Blatt ist, aus den Zuständen seines ersten und zweiten Kindes u_1 und u_2 . Dazu weist \mathbb{A} dem Knoten v einen Zustand $q(v) \in Q$ mit $(q(u_1), q(u_2), a, q(v)) \in \Delta$ zu.
- ▶ q ist ein **akzeptierender Lauf** genau dann, wenn der Zustand der Wurzel zu F gehört.

Anmerkung: Wenn \mathbb{A} deterministisch ist, ist q eindeutig bestimmt; ansonsten kann \mathbb{A} viele Läufe auf t haben.

reguläre Baumsprachen

- ▶ Ein Σ -Baum t wird von \mathbb{A} genau dann **akzeptiert**, wenn es (mindestens) einen akzeptierenden Lauf von \mathbb{A} auf t gibt.

reguläre Baumsprachen

- ▶ Ein Σ -Baum t wird von \mathbb{A} genau dann **akzeptiert**, wenn es (mindestens) einen akzeptierenden Lauf von \mathbb{A} auf t gibt.
- ▶ Die vom Baumautomat \mathbb{A} **erkannte** Baumsprache $L(\mathbb{A})$ ist die Menge aller Σ -Bäume t , die von \mathbb{A} akzeptiert werden.

reguläre Baumsprachen

- ▶ Ein Σ -Baum t wird von \mathbb{A} genau dann **akzeptiert**, wenn es (mindestens) einen akzeptierenden Lauf von \mathbb{A} auf t gibt.
- ▶ Die vom Baumautomat \mathbb{A} **erkannte** Baumsprache $L(\mathbb{A})$ ist die Menge aller Σ -Bäume t , die von \mathbb{A} akzeptiert werden.
- ▶ Eine Baumsprache L heißt **regulär**, wenn es einen Baumautomaten \mathbb{A} gibt, der L erkennt.

reguläre Baumsprachen

Frage: Welche Baumsprache wird von dem Baumautomaten

$\mathbb{A} := (Q, \Sigma, \Delta, F)$ ein Baumautomat mit

- ▶ $Q = \{q_0, q_1\}$
- ▶ $\Sigma = \{0, 1\}$
- ▶ $F = \{q_0\}$
- ▶

$$\Delta = \{(\perp, 0, q_0), (\perp, 1, q_1), \\ (q_0, q_0, 0, q_0), (q_1, q_0, 0, q_1), (q_0, q_1, 0, q_1), (q_1, q_1, 0, q_0), \\ (q_0, q_0, 1, q_1), (q_1, q_0, 1, q_0), (q_0, q_1, 1, q_0), (q_1, q_1, 1, q_1)\}$$

erkannt?

Weiteres Beispiele:

- ▶ Sei $\Sigma = \{0, 1, 2, 3, 4, 5, 6\}$. Gesucht ist ein deterministischer Baumautomat \mathbb{A}_0 , der die folgende Klasse von Σ -Bäumen akzeptiert:
 - ▶ Die Blätter sind jeweils mit einer Zahl aus Σ beschriftet.
 - ▶ Für alle anderen Knoten v gilt: Der Knoten v ist mit der Zahl aus Σ beschriftet, die sich aus der Summe der Beschriftungen seiner Kinder modulo 7 ergibt.

Σ -Bäume und Logik

Signatur: $\tau_\Sigma := \{E_1, E_2\} \cup \{P_a : a \in \Sigma\}$ wobei

- ▶ E_1 und E_2 2-stellige und
- ▶ P_a 1-stellige Relationssymbole

sind.

Σ -Bäume und Logik

Signatur: $\tau_\Sigma := \{E_1, E_2\} \cup \{P_a : a \in \Sigma\}$ wobei

- ▶ E_1 und E_2 2-stellige und
- ▶ P_a 1-stellige Relationssymbole

sind. Ist t ein Σ -Baum mit Knotenmenge $V(t)$, Kantenmengen $E_1(t)$ und $E_2(t)$ und Beschriftungsfunktion λ , so repräsentieren wir t durch die τ_Σ -Struktur \mathfrak{A}_t

- ▶ mit dem Universum $V(t)$
- ▶ und den Relationen $E_i^{\mathfrak{A}_t} := E_i(t)$ (für jedes $i \in \{1, 2\}$)
- ▶ und $P_a^{\mathfrak{A}_t} := \{v \in V(t) : \lambda(v) = a\}$ (für jedes $a \in \Sigma$).

Σ -Bäume und Logik

Signatur: $\tau_\Sigma := \{E_1, E_2\} \cup \{P_a : a \in \Sigma\}$ wobei

- ▶ E_1 und E_2 2-stellige und
- ▶ P_a 1-stellige Relationssymbole

sind. Ist t ein Σ -Baum mit Knotenmenge $V(t)$, Kantenmengen $E_1(t)$ und $E_2(t)$ und Beschriftungsfunktion λ , so repräsentieren wir t durch die τ_Σ -Struktur \mathfrak{A}_t

- ▶ mit dem Universum $V(t)$
- ▶ und den Relationen $E_i^{\mathfrak{A}_t} := E_i(t)$ (für jedes $i \in \{1, 2\}$)
- ▶ und $P_a^{\mathfrak{A}_t} := \{v \in V(t) : \lambda(v) = a\}$ (für jedes $a \in \Sigma$).

Beispiel: siehe Tafel

Σ -Bäume und Logik

- ▶ Ein **SO** $[\mathcal{T}_\Sigma]$ -Satz φ **beschreibt** eine Baumsprache L , wenn gilt: $L = \{t \in \mathcal{T}_\Sigma : \mathfrak{A}_t \models \varphi\}$.

Σ -Bäume und Logik

- ▶ Ein **SO** $[\mathcal{T}_\Sigma]$ -Satz φ **beschreibt** eine Baumsprache L , wenn gilt: $L = \{t \in \mathcal{T}_\Sigma : \mathfrak{A}_t \models \varphi\}$.
- ▶ Eine Baumsprache $L \subseteq \mathcal{T}_\Sigma$ heißt **MSO-definierbar**, wenn es einen **MSO** $[\mathcal{T}_\Sigma]$ -Satz gibt, der L beschreibt.

Σ -Bäume und Logik

- ▶ Ein **SO** $[\mathcal{T}_\Sigma]$ -Satz φ **beschreibt** eine Baumsprache L , wenn gilt: $L = \{t \in \mathcal{T}_\Sigma : \mathfrak{A}_t \models \varphi\}$.
- ▶ Eine Baumsprache $L \subseteq \mathcal{T}_\Sigma$ heißt **MSO-definierbar**, wenn es einen **MSO** $[\mathcal{T}_\Sigma]$ -Satz gibt, der L beschreibt.

Beispiele: Sei $\Sigma = \{0, 1\}$.

- ▶ Gesucht ist ein **SO** $[\mathcal{T}_\Sigma]$ -Satz φ , der die Klasse aller $\{0, 1\}$ -Bäume beschreibt, für die gilt, dass jeder mit 1 beschriftete Knoten einen mit 0 beschrifteten Nachfolger hat.

Σ -Bäume und Logik

- ▶ Ein **SO** $[\mathcal{T}_\Sigma]$ -Satz φ **beschreibt** eine Baumsprache L , wenn gilt: $L = \{t \in \mathcal{T}_\Sigma : \mathfrak{A}_t \models \varphi\}$.
- ▶ Eine Baumsprache $L \subseteq \mathcal{T}_\Sigma$ heißt **MSO-definierbar**, wenn es einen **MSO** $[\mathcal{T}_\Sigma]$ -Satz gibt, der L beschreibt.

Beispiele: Sei $\Sigma = \{0, 1\}$.

- ▶ Gesucht ist ein **SO** $[\mathcal{T}_\Sigma]$ -Satz φ , der die Klasse aller $\{0, 1\}$ -Bäume beschreibt, für die gilt, dass jeder mit 1 beschriftete Knoten einen mit 0 beschrifteten Nachfolger hat.
- ▶ Welche Klasse aller $\{0, 1\}$ -Bäume beschreibt der folgende Satz?

$$\forall x \left(P_1(x) \rightarrow \forall z \left(\left(Z(x) \wedge \forall y \forall z \left((Z(y) \wedge (E_1(y, z) \vee E_2(y, z))) \rightarrow Z(z) \right) \rightarrow \exists y (Z(y) \wedge P_0(y)) \right) \right) \right)$$