Unendliche Spiele

gehalten von Dr. rer. nat. Christof Löding
im Wintersemester 2005/06 an der RWTH Aachen

eine studentische Mitschrift von
Florian Heller
florian@heller-web.net
Diese Mitschrift erhebt keinen Anspruch auf Richtigkeit oder Vollständigkeit.
- Think Different -

6. Februar 2006

1 Unendliche Spiele

1.1 Grundzüge

- 2 Spieler
- Züge Abwechselnd
- deterministisch
- volle/perfekte Information
- Gewinn/Verlust (Nullsumme)
- unendliche Dauer

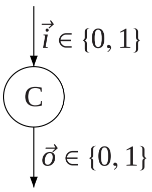
1.2 Enstehung

- in der Mengenlehre Anfang des 20. Jahrhunderts (Banach, Mazur, Mycielski,..)
- Eigenschaften von Teilmengen reeler Zahlen.
- Grale/Stewart (1953) ”Infinite Games of Perfect Information”

$$\begin{array}{lcccc} \text{Spieler I} & x_0 & & x_2 & \\ \text{Spieler II} & & x_1 & & x_3 \end{array} \quad Win \subseteq \{0,1\}^{\omega}$$

Spieler I gewinnt $\Leftrightarrow x_0, x_1, x_2, x_3 \cdots \in Win$ Binärdarstellung einer reellen Zahl aus $[0, 1]$
Wesentliche Frage: Für welche Menge Win hat einer der Spieler eine Gewinnstrategie oder ist dieses Spiel determiniert?

Schaltkreissynthese: Chruch 1936 ”Logic, Arithmetic and Automata”



Berechnung: $\binom{\vec{i}_1}{\vec{\sigma}_1}, \binom{\vec{i}_2}{\vec{\sigma}_2}, \cdots \in \left(\{0, 1\}^{n+1}\right)^{\omega}$
Spezifikation für das gewünschte Verhalten des Schaltkreises

McNaughton 1965, Büchi/Landweber 1969: Sicht als Spiel:
Spieler 0 = Umgebung, zieht in jedem Zug einen Vektor $\vec{i}$ von Eingabesignalen
Spieler 1 gewinnt $\Leftrightarrow$ die unendliche Sequenz von Ein-/Ausgabesignalen die Spezifikation erfüllt.

Wesentliche Fragen: Welcher Spieler hat eine Gewinnstrategie?
Wieviel Speicher/Gedächtnis braucht man um die Strategie umzusetzen?

Heute: Modellierung von reaktiven Systemen (Interaktion mit Umgebung). Weitere Anwendungen:

- Model-Checking (modaler μ-Kalkül)
- Vergleiche von Strukturen (Bisimulation)

Grobgliederung:

1. Grundlegende Definitionen
2. Erreichbarkeitsspiele und Attraktorkonstruktion
3. Spezielle Strategien, Strategieautomaten
4. Staiger-Wagner-Spiele, Spielreduktion
5. Büchi-Spiele
6. Paritäts- und Mullerspiele
7. Determiniertheit, Algorithmen,Anwendungen
8. Rabin-/Streett-Spiele
9. Nebenläufe, stochastische Spiele

2 Grundlegende Definitionen

Definition
Spielgraph $G = (Q, E)$ Q: Kanten E:Kanten
Spieler 0-/Spieler 1-Knoten $Q = Q_1 \cup Q_2$
$Q_0 \stackrel{A}{=} \bigcirc, Q_1 \stackrel{A}{=} \square$
E ist vollständige Kantenrelation: $E \subseteq Q \times Q$

Definition
Partie Folge: $\rho = q_0 q_1 q_2 \cdots \in Q^\omega$ alternativ Abbildung: $\rho : \mathbb{N} \to Q$ mit $(\rho(i), \rho(i+1)) \in E$ für alle $i \in \mathbb{N}$

Definition
Spiel (G, φ) G: Spielgraph φ: Gewinnbedingung
Normalerweise ist φ Bedingung an Knoten die in ρ vorkommen bzw. ∞ oft vorkommen. Dazu definieren wir
$Occ(\rho) = \{q \in Q | \exists i \in \mathbb{N} : \rho(i) = q\}$
$Inf(\rho) = \{q \in Q | \rho(i) = q$ für ∞ viele $i\}$

Beispiel:
für Gewinnbedingungen

- $3 \in Occ(\rho)$
- $3 \in Inf(\rho) \Leftrightarrow 5 \in Inf(\rho)$
- $\{2, 6\} \subseteq Occ(\rho)$

Spieler 0 gewinnt ρ, falls ρ die Gewinnbedingung φ erfüllt.

Definition
Gewinnstrategie für Spieler 0 Funktion $f : Q^* Q_0 \to Q$ Q: bisheriger Spielverlauf
Q_0 aktueller Knoten
sodass für jedes Partiepräfix $q_0, \ldots q_k$ mit $q_k \in Q_0 gilt (q_k, f(q_0, \ldots, q_k)) \in E$ ρ ist gemäß f gespielt / mit f verträglich / f-Partie gdw.
$\forall i \in \mathbb{N} : \rho(i) \in Q_0 \Rightarrow \rho(i+1) = f(\rho(0) \ldots \rho(i))$ f ist Gewinnstrategie für Spieler 0 von q aus gdw. jede f-Partie die in q beginnt, von Spieler 0 gewonnen wird. (analog für Spieler 1)

Beispiel:
a) $3 \in Occ(\rho)$ Gewinnstrategie für 1: $f(wq) = \begin{cases} 5 & \text{falls } q = 2 \\ 4 & \text{falls } q = 6 \end{cases}$

b) $3 \in Inf(\rho) \Leftrightarrow 6 \in Inf(\rho)$ Gewinnstrategie für 0.

$f(w1) = \begin{cases} 2 & \text{falls 3 noch nicht vorkam (in } w\text{) oder falls das letzte Vorkommen von 6 in w nach dem letzten Vork} \\ 7 & \text{sonst} \end{cases}$

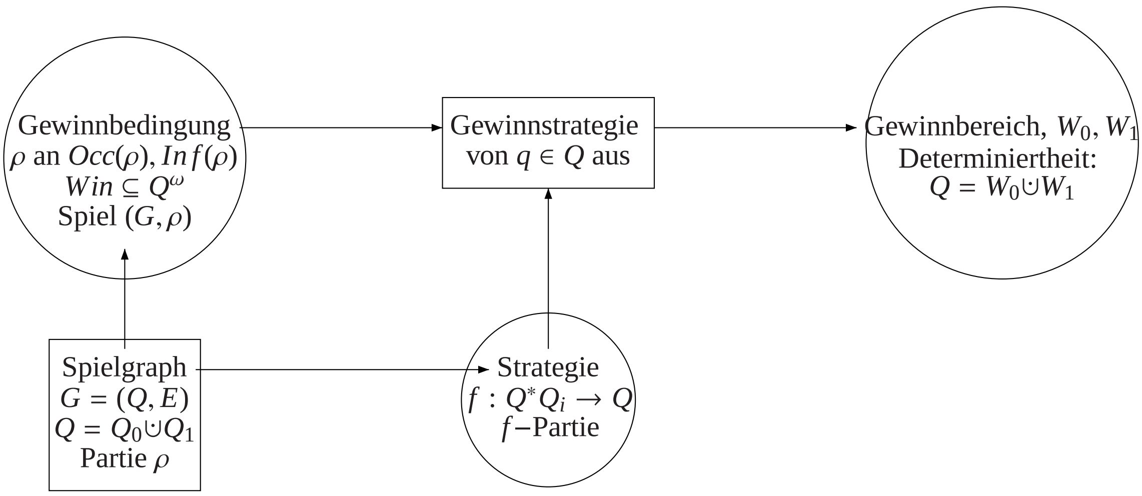
$f(w5) = \begin{cases} 1 & \text{falls der letzte Knoten in w die 2 ist} \\ 6 & \text{Sonst} \end{cases}$

3 Erreichbarkeitsspiele und Attraktorkonstruktion

Definition
Erreichbarkeitsspiel (G, F) mit $F \subseteq Q$.
Spieler 0 gewinnt ρ gdw. $Occ(\rho) \cap F \neq \emptyset$
$W_0\{q \in Q|$ Spieler 0 hat Gewinnstrategie von q aus$\}$
$W_1\{q \in Q|$ Spieler 1 hat Gewinnstrategie von q aus$\}$

Definition
Determiniertheit $W_0 \dot\cup W_1 = Q$
$F \subseteq W_0$

31.10.05

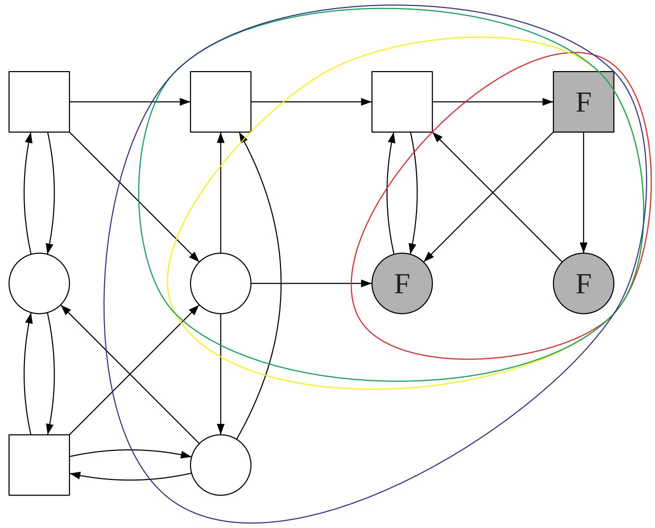


Spieler i gewinnt von q auf $\triangleq$ Spieler i hat eine Gewinnstrategie von q aus.
$\triangleq$ $q \in Q_i$

Lösen eines Spiels $\triangleq$ Berechnen der Gewinnbereiche und entsprechender Gewinnstrategien.

3.1 Erreichbarkeitsspiele

(G, F) mit $F \subseteq Q$
Gewinnbedingung (für SP. 0): $Occ(\rho) \cap F \neq 0$



Rot: $Attr_0^0(F)$
Gelb: $Attr_0^1(F)$
Grün $Attr_0^2(F)$
Blau $Attr_0^3(F)$
$Attr_0^i(F) = \{q \in Q |$ Sp. 0 kann die Partie in $\leq i$ nach F bringen $\}$

$i \geq 0$

$Attr_0^0(F) = F$

$$\begin{aligned} Attr_0^{i+1}(F) = Attr_0^i(F) &\cup \{q \in Q_0 | \exists (q,p) \in E : p \in Attr_0^i(F)\} \\ &\cup \{q \in Q_1 | \forall p \in Q : (q,p) \in E \to p \in Attr_0^i(F)\} \end{aligned}$$

Es gilt: $Attr_0^0(F) \subseteq Attr_0^1(F) \subseteq Attr_0^2(F) \subseteq \cdots \subseteq Attr_0^{|Q|}(F)$
Also bricht die Berechnung irgendwann ab.

$$Attr_0(F) := \bigcup_{i=0}^{|Q|} Attr_0^i(F) = Attr_0^{|Q|}(F)$$

Ist $q \in Attr_0(F)$, dann ist $q \in Attr_0^i(F)$ für irgendein i.
Für $q \in Q_0$ **kann** Sp 0 zu einer niedrigeren Attraktorstufe ziehen.
Für $q \in Q_1$ **muss** Sp 1 zu einer niedrigeren Attraktorstufe ziehen.
Somit $Attr_0(F) \subseteq W_0$
Ist $q \in Q \setminus Attr_0(F)$, so gilt für $q \in Q_1$ existiert $p \in Q$ mit $(q,p) \in E$ und $p \in Q \setminus Attr_0(F)$
für $q \in Q_0$ sind alle Nachfolger von $q \in Q \setminus Attr_0(F)$.
$\Rightarrow Q \setminus Attr_0(F) \subseteq W_1$
Somit gilt $W_0 = Attr_0(F)$ und $W_1 = Q \setminus Attr_0(F)$
Insbesondere gilt $Q = W_0 \cup W_1$ (Erreichbarkeitsspiele sind determiert)

Satz 3.1: Zu einem Erreichbarkeitsspiel (G, F) kann man in Polynomzeit die Gewinnbereiche W_0, W_1 und entsprechende Gewinnstrategien für die beiden Spieler berechnen.

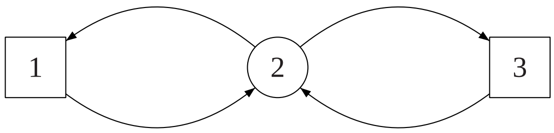
Die Strategie von Sp. 0 nennen wir Attraktorstrategie
Die Berechnung kann in Zeit $O(|Q| + |E|)$ durchgeführt werden (Skript)

3.1.1 Spezielle Strategien / Strategieautomaten

Uniforme Strategien

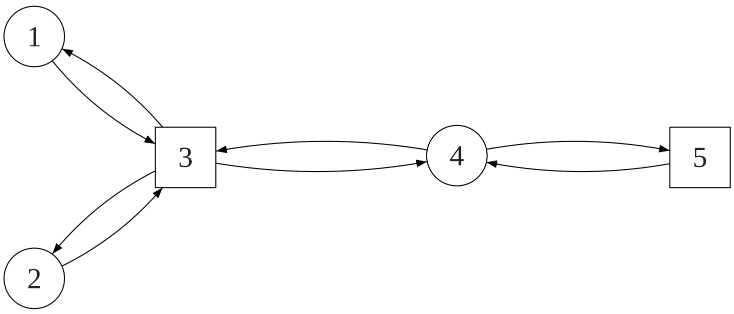
Definition
Eine Strategie f für Sp i ist eine uniforme Gewinnstrategie gdw. f eine Gewinnstrategie von jedem $q \in W_i$ aus ist.

Die Attraktorstrategie ist eine uniforme Gewinnstrategie für Sp. 0
Für die Zugauswahl ist nur der aktuelle Knoten relevant.
Solche Strategien heißen positional.
Positionale Strategien genügen nicht



$Inf(\rho) = \{1, 2, 3\}$

Formal ist f positional gdw. $\forall q \in Q_i \forall w_1, w_2 \in Q^*\colon f(w_1, q) = f(w_2, q)$
Funktion $f : Q_i \to Q$
Kantenmenge $\{(q_i f(q)|q \in Q_i\}$
Frage: Reicht es sich die letzen n Knoten zu merken? (für festes n)
Nein



$\{1, 4\} \subseteq Inf(\rho) \Leftrightarrow 5 \in Inf(\rho)$
Gewinnstrategie für 0: Falls seit dem letzten Besuch von 4 der Knoten 1 besucht wurde, ziehe zu 5, Sonst ziehe zu 3.
3232323432313231345432323234 3. . .

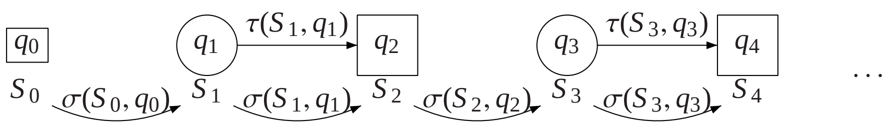
Strategieautomaten

Ein ”Strategieautomat” liest das bisherige Präfix der Partie (Eingabealphabet Q)
Ist die Partie in $q \in Q_0$, so gibt die Ausgabefunktion des Automaten in Abhängigkeit von q und dem Zustand des Automaten den nächsten Zug vor.

Definition (Strategieautomat für Spieler 0)
$\mathcal{A} = (S, Q, s_0, \sigma, \tau)$

S	endliche Zustandsmenge (Speicher)
Q	Eingabealphabet
s_0	Anfangszustand (initialer Speicherzinhalt)
$\sigma : S \times Q \to S$	Transitionsfunktion (Aktualisierun des Speichers)
$\tau : S \times Q_0 \to Q$	Ausgabefunktion (nächster Zug)

Definition
$f_{\mathcal{A}}$-Partie:



formale Definition von $f_{\mathcal{A}}$: Erweiterung von σ auf Eingabefelder $\sigma^* : S \times Q^* \to S$
$\sigma^*(S, \varepsilon) = S$ und $\sigma^*(s, wq) = \sigma(\sigma^*(s, w))$
Dann ist $f_{\mathcal{A}}(wq) = \tau(\sigma^*(s_0, w), q)$

Definition (Formale Def. des Beispielautomaten:)
$S = \{S_0, S_1\}, Q = \{1, 2, 3, 4, 5\}$

$$\sigma(S_0, q) = \begin{cases} S_1 & \text{falls } q = 1 \\ S_0 & \text{sonst} \end{cases}$$

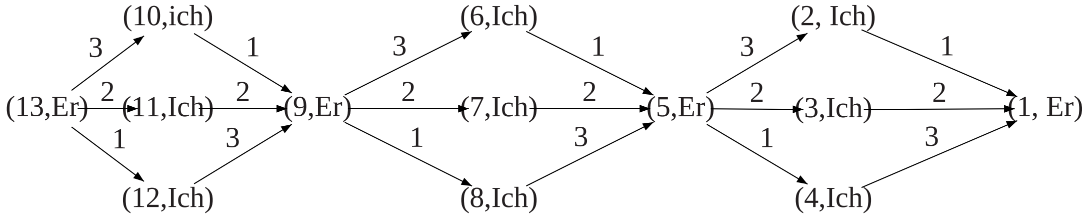
$$\sigma(S_1, q) = \begin{cases} S_0 & \text{falls } q = 5 \\ S_1 & \text{sonst} \end{cases}$$

$\tau(S_0, 4) = 3,\ \tau(S_1, 4) = 5$

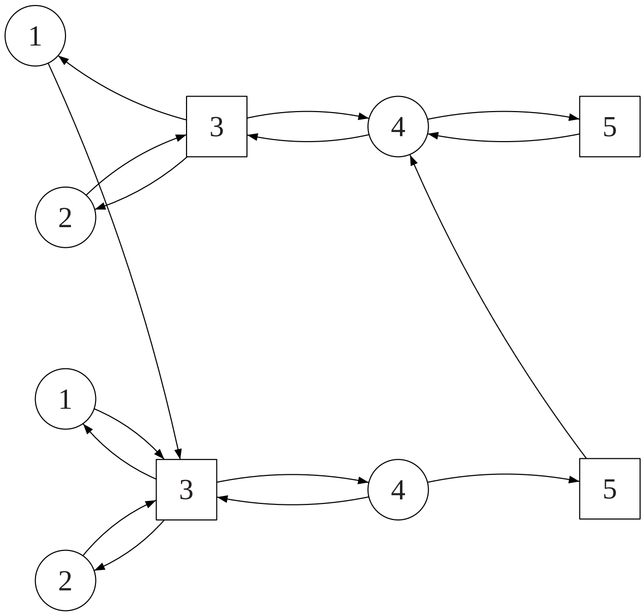
14. 11.05

Wiederholung: Attraktorkonstruktion. **endliches Spiel:**

- Haufen Streichhölzer
- Spieler entfernen abwechselnd 1,2 oder 3 Streichhölzer
- Wer das letzte Hölzchen nimmt verliert.



Nachtrag zu Strategieautomaten $S\sigma$ und G definieren einen Produktgraphen $S \circ_\sigma G$
Knoten: $S \times Q$, Kanten $(s, q) \to (\sigma(s, q), p)$ mit $(q, p) \in E$



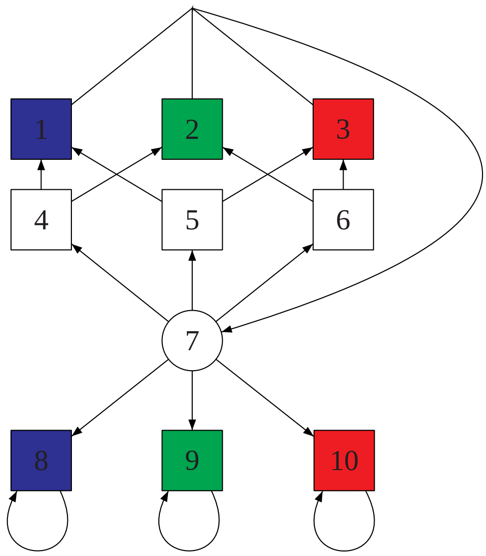
Beispiel: τ definiert eine positionale Strategie aus $S \circ_\sigma G$

3.2 Wie kann man (systematisch) Strategieautomaten konstruieren.

3.2.1 Spielreduktion

Am Bsp. von verallgemeinerten Erreichbarkeitsspielen (VE-Sielen)
Ein VE-Spiel hat die Form $(G, (F_1, \ldots, F_k))$ mit $F_i \subseteq Q$
Spieler 0 gewinnt $\rho \leftrightarrow Occ(\rho) \cap F_i \neq \emptyset$ für alle $i \in \{1, \ldots, k\}$

Ziel Entwickle ein allgemeines Verfahren, dass zu gegebenem VE-Spiel Strategieautomaten für Gewinnstrategien der beiden Spieler berechnet.



Allgemeine Vorgehensweise:

Schritt 1: Welcher Speicher wird benötigt?
Definiere S, σ und S_0
Für VE-Spiele: Welche F_i wurden bereits besucht?
$S = 2^{1,\ldots,k} = Pot\{1,\ldots,k\}, \sigma(t,q) = S \cup \{i \in \{1,\ldots,k\} | q \in F_i\}, t_0 \neq \emptyset$
Bsp.: $t = \{B\}, q = 3$. Dann: $\sigma(\{B\},3) = \{B\} \cup \{R\} = \{B,R\}, t \in S, d.h. t \subseteq |\{1,\ldots,k\}$
Dies definiert $G' := S \circ_\sigma G$
Für VE-Spiele: Graph in dem mitprotokolliert wird, welche Mengen bereits besucht wurden.

Schritt 2: Definiere eine "äquivalente" Gewinnbedingung auf G'
Eine positionale Gewinnstrategie für Spieler 0 entspricht dann einer Ausgabefunktion τ
Für WE-Spiele: Erreiche Speicherinhalt $\{1,\ldots,k\}$
Wir haben VE-Spiele auf E-Spiele reduziert.

Satz 3.2: Seien (G,φ) und (G',φ') Spiele mit $(G,\varphi) \leq (G',\varphi')$ (und Bezeichnungen wie in der Def.): Hat für $q_0 \in Q$ Spieler 0 eine Gewinnstrategie f' von (s_0,q_0) aus die positional ist, dann hat Spieler 0 von q_0 aus (in G) eine Automatengewinnstrategie.

Beweis:
Definiere den Strategieautomaten für Sp.0 in G wie folgt:
$\mathcal{A} = (S,Q,s_0,\sigma,\tau)$ mit $\tau(s,q) = q'$, falls $f'(s,q) = (s',q')$
$G'(s,q) \to (s',q')$
Zeige: $\mathcal{A}$ definiert Gewinnstrategie für Sp.0 in (G,φ)
Sei q_0, q_1, q_2 eine $f_\mathcal{A}$Partie in G
Sei s_0, s_1, s_2 die entsprechende Folge von Speicherinhalten, also $S_i + 1 = \sigma(s_i,q_i)$
Nach Def. von $G' = S \circ_\sigma G$ ist dann $(s_0,q_0)(s_1,q_1)\ldots$ eine Partie in G'
Nach Def . von τ (anhand von f') ist diese Partie sogar eine f'-Partie
Somit wird sie von Sp0 in $G',\varphi')$ gewonnen
Also wird q_0,q_1,q_2 von Sp. 0 in (G,φ) gewonnen und deshalb ist $f_\mathcal{A}$ eine Gewinnstrategie $\square$
Entsprechendes gilt fpr Sp. 1

Korollar 1 *Gilt $(G,\varphi) \leq (G',\varphi')$ und gewinnen beide Spieler auf ihren Gewinnbereichen W_0', W_1' mit nicht positioalen Strategien in G', dann können wir di eGewinnbereiche W_0, W_1 und entsprechende Automatenstrategien in G berechnen.*

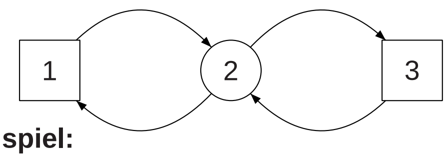
Für zwei Klassen K und K' von Spielen sagen wir, dass K-Spiele auf K'-Spiele reduzierbar sind, wenn es eine allgemeine Konstruktion gibt, die zu jedem Spiel (G, φ) aus K ein Spiel (G', φ') aus K' liefert mit $(G, \varphi) \leq (G', \varphi')$

Satz 3.3: VE-Spiele sind auf E-Spiele reduzierbar

3.3 Staiger-Wagner-Spiele

Definition
Klasse der Spiele, deren Gewinnbedingungen nur von den Occ-Mengen abhängen. Ein SW-Spiel hat die Form $(G, \mathcal{F})$ mit $\mathcal{F} \subseteq 2^Q$
$\mathcal{F} = \{F_1, \ldots, F_k\}$ mit $F_i \subseteq Q$

Sp. 0 gewinnt ρ genau dann, wenn $Occ(\rho) \in \mathcal{F}$ (also $Occ(\rho) = F_i$ für ein i)



Beispiel:

$\mathcal{F} = \{\{1, 2\}, \{2, 3\}\}$

$\mathcal{F} = \{\{1, 2, 3\}\}$

$\mathcal{F}$ listet die guten Occ-Mengen für Spieler 0 auf.

Bemerkung:
Ist $Win \subseteq Q^\omega$ mit $\forall \rho_1, \rho_2 \in Q^\omega \cdot Occ(\rho_1) = Occ(\rho_2) \rightarrow (\rho_1 \in Win \Leftrightarrow \rho_2 \in Win)$ so giibt es auch eine SW-Bedingung die Win definiert.

Beweis:
Setze $\mathcal{F} = \{Occ(\rho) | \rho \in Win\} \square$

Ziel: berechne Gewinnbereiche und gewinnstrategien für SW-Spiele
Erinnerung zu Spielreduktion
Bild

3.3.1 Schwache Paritätsspiele

Ein schwaches Paritätsspiel hat die Form (G, c) mit Prioritätsfunktion $c : Q \rightarrow \{0, 1, \ldots, k\}$ mit $k \in \mathbb{N}$
Eine Partie $\rho \in Q^\omega$ induziert eine entsprechende Folge von Prioritäten $c(\rho) = c(\rho(0))c(\rho(1))\ldots$ Spieler 0 gewinnt ρ gdw. $max(Occ(c(\rho)))$ ist gerade.
Beispielpartien: $\rho_1 = q_4 q_3 q_5 q_2 (q_1 q_0)^\omega$
$c(\rho_1) = 2\ 2\ 3\ 1\ (0\ 0)^\omega$
$max(Occ(\rho_1)) = 3$ ungerade, also wird ρ_1 von Spieler 1 gewonnen.
$\rho_2 = q_4 q_7 q_6 q_8 q_5 q_2 (q_1 q_0)^\omega$
$c(\rho_2) = 2\ 3\ 3\ 4\ 3\ 1 (0\ 0)^\omega$

Bemerkung:
E-Spiele sind spezielle schwache Paritätsspiele. Definiere Paritätsfunktion $c(q) = \begin{cases} 2 & q \in F \\ 1 & q \notin F \end{cases}$

Satz 3.4: In schwachen Paritätspielen haben beide Spieler uniforme positionale Strategien auf ihren Gewinnbereichen. Solche Strategien können in Zeit $O((k+1)(|Q|) + |E|))$ berechnet werden.

Beweis:
Per Induktion über k.
$k = 0$: Spieler 0 gewinnt jede Partie. Wähle beliebige Strategie.
$k \geq 1$ Hier nur der Fall k gerade. Anderer Fall analog mit Rollen von Sp. 0 und Sp.1 vertauscht.
Sei $C_k = \{q \in Q | c(q) = k\}$ Setze $A_k := Attr_0(c_k)$
Dann ist $A_k \neq \emptyset$ (wenn $C_k \neq \emptyset$) Sei G_{k-1} der Graph, der durch das Löschen der Knoten in A_k entsteht.
Nach I.V. haben Sp. 0 und Sp. 1 auf ihren Gewinnbereichen u_0, U_1 in G_{k-1} uniforme positionale Strategien (und G_{k-1} ist als Komplement eines Attraktors wieder ein Spielgraphen)
Zeige: $W_1 = U_1, W_0 = U_0 \cup A_k$
Seien dazu g_0, g_1 die Strategien auf u_0 und U_1 in G_{k-1}.
Definition von $f_1 \dots f_n := g_1$ Eine f_1-Partie von U_1 aus kann G_{k-1} nicht verlassen. Da g_1 eine Geinnstrategie für Sp. 1 auf G_{k-1} ist, wird die Partie auch in G gewonnen. Definition von f_0:
Auf U_0 spiele gemäß g_0 und auf A_k gemäß der Attraktorstrategie. (mit beliebiger Wahl auf C_k).
Dann bleibt eine f_0-Partie entweder in G_{k-1} und wird von Sp. 0 gewonnen, da g_0 eine Gewinnstrategie für Sp. 0 in G_{k-1} ist, aber die Partie erreicht irgendwann A_k und somit auch C_k. Dann ist k die größte vorkommede Priorität und Sp. 0 gewinnt.

28.11.200

Eine Partie $\rho' = (R_0, q_0)(R_1, q_1), \dots$ in $G' = S \circ_\sigma G$ hat die Eigenschaft, dass die R_i sich irgendwann nicht mehr ändern, d.h. ex. $i \in \mathbb{N}$ mit $R_j = R_i$ für alle $j \geq i$ und $|R_i|2$ falls ... , gewinnt Sp.0 ρ genau dann wenn Sp.0 ρ gewinnt.

Reduktion liefert Strategieautomaten, deren größe eponentiell in der Größe des Spielgraphen ist.

Satz 3.5: Es gibt eine Folge $(G_1, \mathcal{F}_1), (G_2, \mathcal{F}_2), \dots$ von SW-Spielen, sodass die Größe von G_n linear in n ist und jeder Strategieautomat der eine Gewinnstrategie für Sp. 0 auf seinen Gewinnbereich in $(G_n, \mathcal{F}_n)$ realisiert, hat mindestend 2^n Zustände.

Beweis:
Ang. es gibt einen Strategieautomaten für Sp. 0 mit weniger als 2^n Zuständen der eine Gewinnstrategie ind $(G_n, \mathcal{F}_n)$ realisiert. Dann exisitieren zwei Partiepräfixe w_1, w_2 die von q_0 nach q führen (mit $w_1 \neq w_2$) sodass der Strategieautomat $\mathfrak{A}$ ist nach w_1, w_2 im gleichen Zustand. Ab q ist die weitere $f_\mathfrak{A}$-Partie eindeutig bestimmt. Somit verliert Sp. mindestens eine der beiden Partien und $f_\mathfrak{A}$ ist keine Gewinnstrategie für Sp. 0. □

3.4 Büchi-Spiele

Ein Büchi-Spiel hat die Form (G, F) mit $F \subseteq Q$. Spieler 0 gewinnt eine Partie ρ gdw. $Inf(\rho) \cap F \neq \emptyset$
Idee zur Lösung von Büchi-Spielen:
Zwischenziel für Sp. 0:

- Erreiche $f(Attr_0(F))$
- In F angekommen, versuche wieder in $Att_0(F)$ zu ziehen.

Berechne die Knoten aus F, von denen Sp. 0 mindestens i Wiederbesuche in F garantieren kann. Diese Menge nennen wir $Recur_0^i(F)$ Dazu zunächst für $P \subseteq Q$ $Next_0(P)= \{q \in Q_0 | \exists(q,p) \in E$ mit $p \in P\} \cup \{q \in Q | \forall(q,p) \in E$ gilt $p \in P\}$ (Alle Knoten bei denen wir im nächstens Zug in P landen).
Setze $Att_0^+(F) = Next(Attr_0(F))$

Definiere $Recur_0^0(F) = F$
$Recur_0^1(F) = F \cap Attr_0^+(F)$
$Recr_0^{i+1}(F) = F \cap Attr_0^+(Recur_0^i(F))$

Im Beispiel: $F = \{3,6,9\}$
$Recur_0^0(F) = \{3,6,9\}$
$Attr_0(F) = \{3,4,5,6,7,8,9,10\}$
$Attr_0^+(F) = \{4,5,6,7,8,9,10\}$
$Recur_0^1(F) = \{6,9\}$
$Attr_0(Recur_0^1(F)) = \{5,6,8,9,10\}$
$Attr_0^+(Recur_0^1(F)) = \{5,8,9,10\}$
$Recur_0^2(F) = \{9\}$
$Attr_0(Recur_0^2(F)) = \{9,10\}$
$Attr_0^+(recur_0^2(F) = \{9,10\}$
$Recur_0^3(F) = \{9\}$ Es gilt $F \supseteq Recur_0^1(F) \supseteq Recur_0^2(F) \supseteq \cdots \supseteq Recur_0^{|F|}(F) = Recur_0^{|F|+1}(F)$
Setze $Recur_0(F) \coloneqq Recur_0^{|F|}(F) = \bigcap_{i\geq 0} Recur_0^i(F)$
$Recur(F)$ ist die Menge der Knoten von F, von denen Sp. 0 unendlich viele Besuche in F garantieren kann.

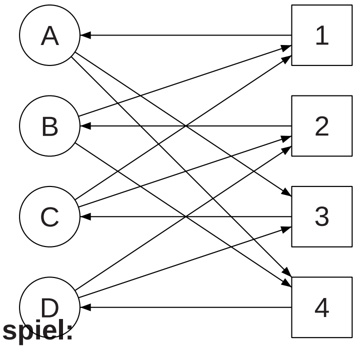
Lemma In einem Büchi-Spiel (G,F) gilt, $W_0 = Attr_0(Recur_0(F))$ und beide Spieler haben uniforme positionale Strategien auf ihrem Gewinnbereich.

Beweis:
Es gilt $F \cap Attr_0^+(Recur_0(F)) = Recur_0(F)$ also $Recur_0(F) = F \cap Next_0(Attr_0(Recur_0(F)))$. Wenn die Partie alsoin $Recur_0(F)$ ist, dann kann Spieler 0 im nächsten Zug garantieren, dass die Partie in $Attr_0(Recur_0(F)$

5.12.2005

3.5 Muller und Paritätsspiele

Muller-Spiele sind von der Form $(G, \mathfrak{F})$ mit $\mathfrak{F} \subseteq 2^Q$, also $\mathfrak{F} = \{S_1, \ldots, S_k\}$ mit $S_i \subseteq Q$
Sp. 0 gewinnt $\rho \Leftrightarrow Inf(\rho) \in \mathfrak{F}(\Leftrightarrow Inf(\rho) = S_i$ für ein $i)$



Beispiel:

$$\forall x \in \{A, B, C, D\}, y \in \{1, 2, 3, 4\}; (x, y), (y, x) \in E$$

Gewinnbedingung $|Inf(\rho) \cap \{A, B, C, D\}| = max(Inf(\rho) \cap \{1, 2, 3, 4\}$
Hat einer der Spieler eine Gewinnstrategie? Wenn ja, welcher und mit wieviel Speicher?

3.5.1 Paritätsspiele

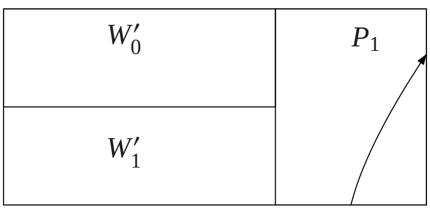
Parritätsspiele haben die Form (G, c) mit Prioritätsfunktion $c : Q \to \{0, \dots, k\}$
Sp. 0 gewinnt $\rho \Leftrightarrow max(Inf(c(\rho)))$ gerade. Partie $\rho := (q_6, q_2, q_5)^\omega$
$max(Inf(c(\rho))) = 3$ $W_0 = \{q_1, q_3, q_4, q_7, q_8\}$ $W_1 = \{q_2, q_5, q_6, q_9, q_{10}\}$

Nächste Schritte:

- Positionale Determiniertheit von Paritätsspielen
- Reduktion von Muller-Spielen auf Paritätspielen
- Algorithmen zum Lösen von Paritätsspielen
- Anwendung der Paritätsspiele in der Theorie der Baumautomaten

Satz 3.6: In einem Paritätsspiel hat von jedem Knoten einer der beiden Spieler eine Gewinnstrategie. Auf ihren Gewinnbereichen haben die beiden Spieler jeweils uniforme positionale Gewinnstrategien.

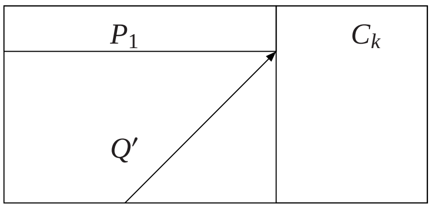
Beweis:
Notation: UPGS
Induktiv über die Anzahl der Prioritäten
$k = 0$: Spieler 0 gewinnt immer mit beliebiger Strategie. Sei P_1 die Menge der Knoten, von denen Spieler 1 eine positionale Gewinnstrategie hat. Diese GS kann also UPGS f_1 gewählt werden.
Setze $C_k := \{q \in Q | c(q) = k\}$
Wir betrachten nur den Fall dass k gerade ist. Der andere Fall verläuft analog mit den Rollen der Spieler vertauscht.
Zeige $Q' := Q \setminus P_1$ ist der Gewinnbereich von Sp. 0 (mit UPGS).



Der durch Q' induzierte Teilgraph $G'(Q', E')$ definiert ein Subspiel (G', c') mit $c' = c_{|Q|}$
Begründung:Q' ist eine Falle für Sp.1:
$\forall q \in Q' : q \in Q_0 \Rightarrow \exists (q, p) \in E : p \in Q'$
$und q \in Q_1 \Rightarrow \forall (q, p) \in E : p \in Q'$

1. Fall $C_k \subseteq P_1$
   Dann können wir die I.V. auf (G', c') anwenden. Diese liefert W_0', W_1' mit UPGS f_0', f_1'. Es gilt $W_1' \neq \emptyset$, da Sp.1 von dort aus positional gewinnen kann indem er auf W_1' nach f_1' spielt und auf P_1 nach f_1
   Spielt Sp. 0 auf $W_0' = Q'$ in G nach f_0', dann gewinnt er, da Q' eine Falle für Sp.1 ist.
   Somit ist $W_0 = W_0'$ und $W_1 = P_1$ mit UPGS.

2. Fall $C_k \nsubseteq P_1$
   Setze $A := Attr_0(C_k \setminus P_1)$



   Setze $Q'' := Q' \setminus A$. Dann ist (G'', c'') mit $G'' = (Q'', E'')$ und $c'' = c|_{Q''}$ ein Subspiel
   Wir können die I.V. auf (G'', c'') anwenden und erhalten W_0'', W_1'' und UPGS f_0'', f_1''. Wie im Fall 1 muss $W_1'' \neq \emptyset$ gelten (nach Whal von P_1)
   Definiere nun f_0 auf $Q' = W_0'' \cup A$ wie folgt.

   - Auf W_0'' spiele gemäß f_0''
   - Auf A spiele gemäß der Attraktorstrategie, die Sp. 0 nach $C_k \setminus P_1$ bringt
   - Auf $C_k$wähle beliebige Kante nach Q' (existiert, da Q' eine Falle ist für Sp. 1)

   Eine f_0'-Partie bleibt entweder schliesslich in W_0'' (dann gewinnt Sp. 0 nach I.V.) oder sie besucht A unendlich oft und somit auch C_k In diesem Fall ist k die höchste Priorität, die unendlich oft auftritt. Da k gerade ist, gewinnt Sp. 0.
   Somit ist f_0 eine UPGS auf Q' aso gilt $W_0 = Q'$ und $W_1 = P_1$

□

Da es nur endlich viele positionale Strategien zu einem festen Graphen gibt, kann man im Prinzip alle diese Strategien durchprobieren.

Lemma
Gegeben ein Paritätsspiel (G, c), eine positionale Strategie f für Sp. 0 und $q \in Q$, kann man in Polynomzeit entscheiden, ob f eine GS von q aus ist.

Beweis:
- Streiche aus G alle Kanten aus $Q_0 \times Q$, die nicht von f benutzt werden.
- Berechne dann alle Knoten, die noch von q aus erreichbar sind. Streiche alle anderen Knoten. Bezeichne diesen Graphen mit $G_f = (Q_f, E_f)$

- Sp. 1 kann genau dann gegen f gewinnen wenn es in G_f noch einen Zykel gibt, dessen höchste Priorität ungerade ist.
  Dies ist genau dann der Fall, wenn es eine ungerade Priorität $m \in \{0, \ldots, k\}$ gibt, so dass in G_f eingeschränkt auf $Q_m = \{q \in Q | c(q) \leq m\}$ eine starke Zusammenhangskomponente existiert die nichttrivial ist (enhält mindestens eine Kante) und einen Knoten der Priorität m enhtalten.

Da SCCs om Polynomzeit berchnen kann, kann der gesamte Test in Polynomzeit durchgeführt werden

□

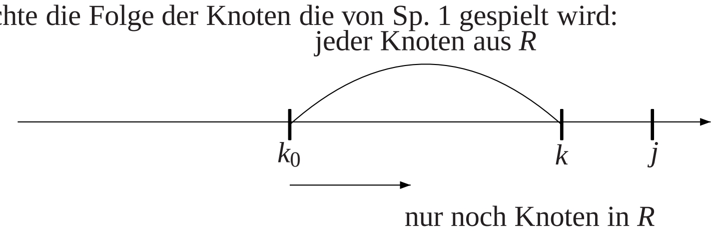
Satz 3.7: Das Problem: Gegeben Paritätsspiel (G, c) und Knoten $q \in Q$
Frage: $q \in W_0$ ist in $NP \cap co - NP$

Beweis:
In NP: Rate pos. Strategie für Sp. 0. Teste in Polynomzeit, ob dies eine GWS für Sp.0 vo q aus ist.
Komplementproblem: $q \notin W_0$? $\Leftrightarrow q \in W_0$
Verfahre analog für Sp. 1.

Ein Schwieriges Muller-Spiel: Spieler 0 hat eine GWS:

$$A \quad A \quad B \quad C \quad D \quad A \quad D \quad C \quad C_1 \quad D_2 \quad C_2 \quad A_3 \quad \cdots$$

Für $X \in \{A, B, C, D\}$ betrachte die Menge S der Knoten in $\{A, B, C, D\}$ die seit dem letzten Besuch von X gesehen wurden. Ziehe zu $|S|$
Ist das eine GWS?
Betrachte die Folge der Knoten die von Sp. 1 gespielt wird:



Sei R die Menge der Knoten die dabei unendlich oft auftreten Sei $j > k$ und X der j-te Knoten.
Die Menge der Knoten die seit dem letzten Besuch von X gesehen wurden, ist eine Teilmenge von R.
Sei $Y \in \{A, B, C, D\}$ der Knoten aus R, dessen Besuch an j am längsten zurückliegt.
Da $Y \in R$ gilt, ex. ein $l \geq j$, sodass Y an Position l besucht wird. Seit dem letzten Besuch von Y werden dann alle anderen Knoten aus R besucht, als zieht Sp. 0 zu $|R|$.
somit haben wir eine GWS für Sp.0 gefunden.
Finde endliche Datenstruktur die die benötigte Information für Sp. 0 zur Verfügung stellt.
Merke die Knoten in der Reihenfolge ihrer letzten Besuche (LAR≃ latest appearance record)
Erweiterung dieser Idee für die Reduktion von Muller- auf Paritätsspiele.
Zu Q sei die Menge $LAR(Q)$ definiert als $LAR(Q) = \{q_1, \ldots, q_n | n = |Q|, q_i \neq q_j$ für $i \neq j\}$
Weiterhin sei $\sigma_{LAR} : LAR(Q) \times Q \to LAR(Q)$ definiert durch $\sigma_{LAR}((q_1 \ldots q_n, q) = (qq_0 \ldots q_{i-1}1_{i+1} \ldots q_n$ für das i mit $q_i = q$
$\sigma_{LAR}((CDAB), A) = ACDB$
Für $((q_1 \ldots q_n), q) \in LAR(Q) \times Q$ sei die Treffermenge $\{q_1, \ldots, q_i\}$ für das i mit $q_i = q$

Zu einer Partie $\rho \in Q^\omega$ und einem (beliebiegen, fest gewählten) $s_0 \in LAR(Q)$ ist die entsprechende LAR-Partie $\rho' \in (LAR(Q) \times Q)^\omega$ definiert durch $\rho(0) = (s_0, \rho(0))$ und $\rho'(i+1) = \sigma_{LAR}(\underbrace{S_i, \rho(i)}_{=\rho'(i)}), \rho(i+1))$

Lemma (LAR-Lemma)
Ist $\rho \in Q^\omega$ und ρ' die entsprechende LAR-Partie, dann ist die Treffermenge in ρ' unendlich oft gleich $Inf(\rho)$ und von einer bestimmten Stelle an immer Teilmenge von $Inf(\rho)$.

Beweis:
Wähle $k_0 < k$ so dass $\rho(j) \in Inf(\rho)$ für alle $j \geq k_0$ und $\{\rho(k_0) \dots \rho(k)\} = Inf(\rho)$
Sei $j > k$ Die Treffermenge von $\rho'(j) = (S_k, \rho(j))$ enthält genau die Zustände, die seit dem letzten Besuch von $\rho(j)$ besucht wurden. Nach Wahl von k_0 und k ist dies eine Teilmenge von $Inf(\rho)$
Sei $q \in Inf(\rho)$ der Zustand aus $Inf(\rho)$ der in S_i am weitesten rechts steht.
Da $q \in Inf(\rho)$ ex. eine Position $l > j$, an der q in ρ auftritt. Dann ist die Treffermenge von $\rho'(l)$ gleich $Inf(\rho)$.□

Satz 3.8: Muller-Spiele können auf Paritätsspiele reduziert werden.

Beweis:
Sei $(G, \mathfrak{F})$ mit $G = (Q, E)$ ein Muller-Spiel. Definiere die für die Reduktion benötigten Komponenten

- $S = LAR(Q)$
- $S_0 \in LAR(Q)$ beliebig
- $\sigma = \sigma_{LAR}$

In $G' = S \circ_\sigma G$ sind die Partien die LAR-Partien zu Partien in G.
Definiere also $c(s,q) = \begin{cases} |F| \cdot 2 & fallls\ F \in \mathfrak{F} \\ |F| \cdot 2 - 1 & falls\ F\ /\!\!\int \mathfrak{F} \end{cases}$ mit F= Treffermenge von (s, q).
Sei $\rho \in Q^\omega$ eine Partie in G und ρ' die entsprechende LAR-Partie in G'.
Die höchste Priorität, die in ρ' auftritt ist nach dem LAR-Lemma anhand von $Ind(\rho)$ definiert, nämlich $2|Inf(\rho)|$ falls $Inf(\rho) \in \mathfrak{F}$ und $2|Inf(\rho)| - 1$ sonst.
Somit gewinnt Sp.0. ρ gdw. er ρ' gewinnt. □

Mit dem Satz über Spielreduktionen folgt:

Satz 3.9: In Muller-Spielen kann man die Gewinnbereiche der beiden Spieler berechnen und entsprechende Automatenstrategien konstruieren.

Die reduktion liefert zu einem Muller-Spiel $(G, \mathfrak{F})$ mit n Zuständen Strategieauomaten mit $n!$ Zuständen.
Dies kann nicht verbessert werden (asymptotisch), unter der Voraussetzung, dass die Konstruktion von G' in der Reduktion nur vom Graphen G abhängt.
Folie: Ein Schwieriges Muller-Spiel Sei dazu $G^{(n)} = (Q^{(n)}, E^{(n)})$ mit $Q^{(n)} = \{-1, \dots, -n, 1, \dots, n\}$
$Q_0^{(n)} = \{-1, \dots, -n\}$
$Q_1^{(n)} = \{1, \dots, n\}$
$E^{(n)} = (Q_0^{(n)} \times Q_1^{(n)}) \cup (Q_1^{(n)} \times Q_0^{(n)})$ und es sei $F \in \mathfrak{F}^{(n)} \subseteq q^{q^{(n)}}$, falls $F \cap \{-1, \dots, -n\} = max(F \cap \{1, \dots, n\})$

Satz 3.10: In den Muller-Spielen $(G^{(n)}, \mathfrak{F}^{(n)})$ hat Sp.0 eine Gewinnstrategie und jeder Automat der eine GWS für SP. 0. reailisert hat $n!$ Zustände

Beweis:
Induktiv über n:
$n = 1$: ✓
$n > 1$ Skript!

19.12.05
9.1.06

4 Baumautomaten

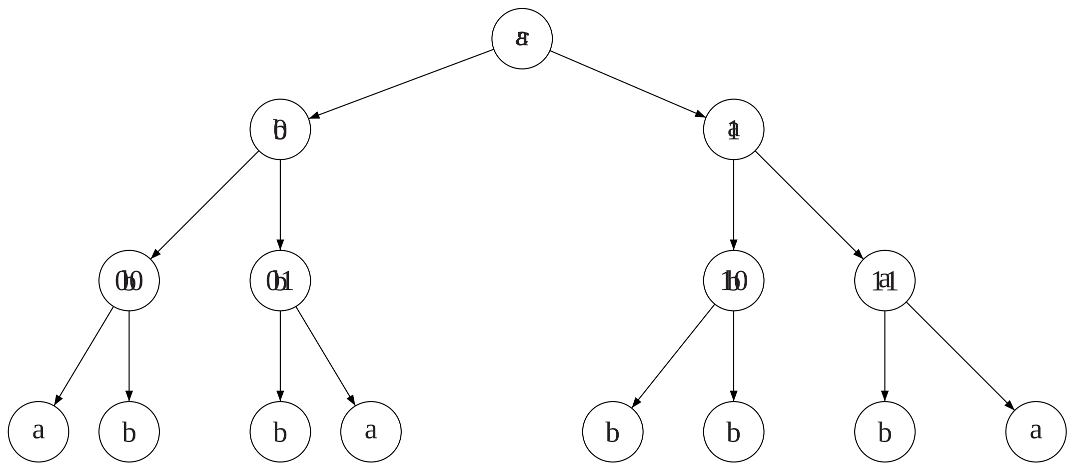
4.1 Hintergründe

Erweiterte Sprachtheorie,
endliche Wörter unendliche Wörter
endliche Bäume unendliche Bäume
Lösung logischer Entscheidungsprobleme (Rabin 1969).
Bäume zur Systembeschreibung, zur Beschreibung des Verhaltens von Systemen.
Im Folgenden:

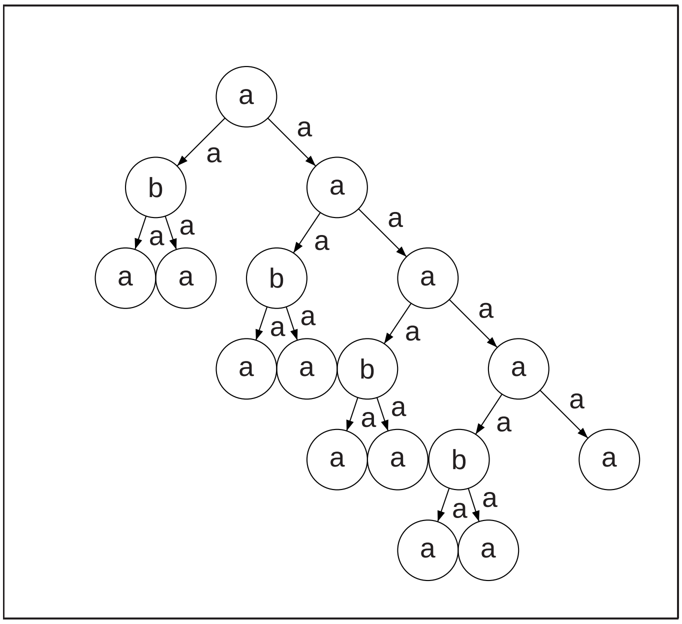
- Bäume, Baumsprachen
- Automaten für unendliche Bäume

4.2 Bäume und Baumsprachen

Hier: Binäre Bäume, alle Pfade unendlich. beschriftungen an den Knoten über endlichem Alphabet Σ



Knoten: Name + Beschriftung
Name: endliches Wort über $\{0, 1\}$ Linksabzweigung: 0 anhängen
Rechtsabzweigung: 1 anhängen
Durch diese Konvention: Knoten und Kantenrelation festegelegt.
Zur Angabe eines Baumes genügt die Angabe der Beschriftung.
Ein Σ-Beschrifteter Baum t ist eine Abbildung $T : \{0, 1\}* \to \Sigma$
In obigem Beispiel:
$t(\varepsilon) = a$
$t(01) = b$



Beispiel:
Nach erster Linksabzweigung ein b sonst nur a

$t(n) = \begin{cases} b & \text{falls } u = 1^n0 \text{ für } n \geq 0 \\ a & \text{sonst} \end{cases}$

4.3 Begriffe und Notationen

T_Σ Menge aller Σ-beschrifteten Bäume
Baumsprache: $T \subseteq T_\Sigma$
Ein Pfad π ist eine (unendliche) Folge$u_0u_1u_2\ldots$ mit $u_0 = \varepsilon$ und $U_{i+1} = u_i0$ oder $u_{i+1} = u_i1$
Ist t ein Baum und π ein Pfad, dann ist $t_\pi = t(u_0)t(u_1)t(u_2)\ldots$ für $\pi = u_0u_1u_2$
Ein Pfad π entspricht einem unendlichen Wort über $\{0,1\}$ z.B. entspricht $(01)^\omega$ dem Pfad, der abwechselnd link und rechts verzweigt. ε 0 01 010 0101 ...Für obigen Beispielbaum und $\pi = 1101^\omega$ ist $t_{|\pi} = aaaba^\omega$

Beispiel:
eine Baumsprache:
$T_1 := \{t \in T_{\{a,b\}} | \exists$ Pfad π mit $t_{|\pi}$ enthält unendlich viele b $\}$

4.4 Baumautomaten

Idee: Lauf ist eine Beschriftung des Eingabebaums mit Zuständen.
Transitionen
Automat in Zustand q, Knoten mit a beschriftet. Transistion: (q, a, q_1, q_2)

4.4.1 Definition

$\mathcal{A} = (Q, \Sigma, q_0, \Delta, Acc)$

- Q: endliche Zustandsmenge
- Σ: Beschriftungsalphabet

- $q_0 \in Q$ Anfangszustand
- $\Delta \subseteq Q \times \Sigma \times Q \times Q$: Transitionsrelation
- Acc: Akzeptanzkomponente (beschreibt unendliche Zustandsfolgen)

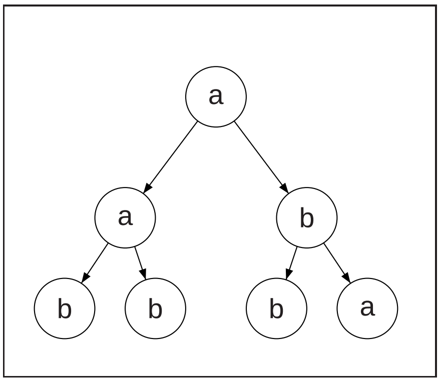
Ein Lauf ρ von $\mathcal{A}$ auf dem Beum t ist eine Abbildung $\rho : \{0, 1\} \to Q$, also ein Baum aus T_q, mit

- $\rho(\varepsilon) = q_0$
- Für alle $u \in \{0, 1\}^*$ gilt : $(\rho(u), t(u), \rho(u0), \rho(u1)) \in \Delta$

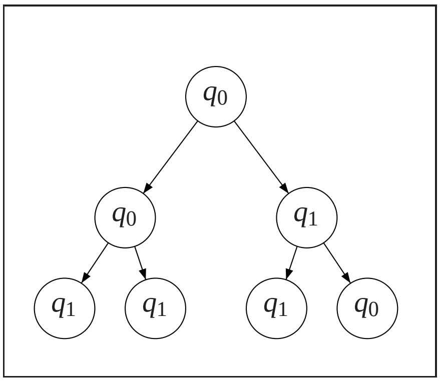
Beispiel:
Zustände: q_0, q_1

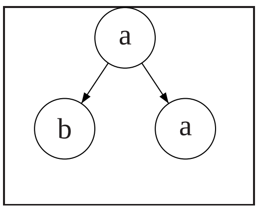
Transitionen: (q_0, a, q_0, q_1)
(q_0, a, q_1, q_1)
(q_1, b, q_0, q_0)
(q_1, b, q_1, q_0)

Anfangsstück eines Baums:



Anfangsstück eines Laufs

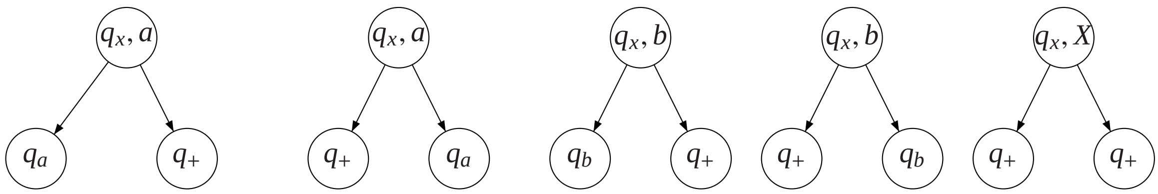


Auf einem Baum der mit 

beginnt kann man mit obigen Transitionen keinen gültigen Lauf bilden.

Zur Akzeptanzkomponente: Beschreibt unendliche Zustandsfolgen, wie z.B. Büchi- Muller oder Paritätsbedingungen.
ein Lauf ρ ist akzeptierend, wenn für jeden Pfad π gilt: $\rho_{|\pi}$ erfüllt die Akzeptierbedingunge (Büchi, Muller, ...)
Die von $\mathcal{A}$ akzeptierte Sprache ist $T(\mathcal{A}) = \{t \in T_\Sigma |$ es gibt einen akzpetierenden Lauf von $\mathcal{A}$ auf $t\}$

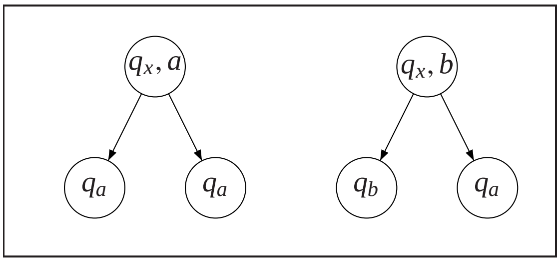
Beispiel:
Büchi-BA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ für T_1
Idee: Rate den Pfad π, auf dem unendlich viele b vorkommen.
Auf diesem Pfad: für jedes b gehe in Zustand $q_b \in F$
Rest des Baumes beschriftet mit q_+
Formal: $Q = \{q_a, q_b, q_+\}, F = \{q_b, q_+\}$
$q_0 = q_a$ und Transitionen



Für $x \in \{a, b\}$

Für einen Lauf ρ von $\mathcal{A}$ auf t gilt dass genau ein Pfad nur mit q_a und q_b beschriftet ist. Auf allen anderen Pfaden steht schließlich nur noch q_+ Auf diesem Pfad kommt genau dann unendlich of q_b vor, wenn in t auf diesem Pfad unendliche oft b vorkommt. Somit gilt $T(\mathcal{A}) = T_1$

Beispiel:
Muller-BA:
$\overline{T}_1 = \{t \in T_{\{a,b\}} | \forall$ Pfade $\pi . t_{|\pi}$ enthält nur endlich viele $b\}$
$Q = \{q_a, q_b\}, q_0 = q_a, \mathcal{F} = \{\{q_a\}\}$
Transitionen:



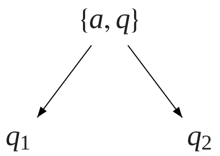
Im Baum kommen unendlich viele b vor , aber auf jedem Pfad höchstens 1 mal.

Lemma
Die Sprache $\overline{T}_1$ kann nicht durch einen Büchi-BA akzeptiert werden.

Satz 4.1: Die Klasse der Büchi-erkennbaren Baumsprachen ist nicht unter Komplement abgeschlossen.

16.01.06

$t : \{0, 1\}* \to \Sigma$



$\in Q \times \Sigma \times Q \times Q$

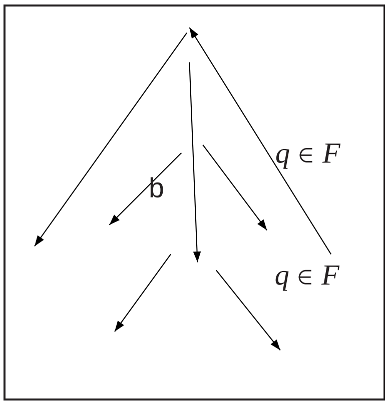
Büchi-Baumautomaten (BBA)
Muler-Baumautomaten (MBA)
Paritäts-Baumautomaten (PBA)

Lemma

Die Sprache: $T = \{t \in T_{\{a,b\}} | \forall$ Pfade $\pi : t_{|\pi}$ enthällt nur endlich viele b $\}$ ist nicht BBA-erkennbar:

Beweis:
Angenommen, der BBA $\mathcal{A} = (Q, \Sigma, q_0, \Delta, F)$ akzeptiert T.
Generelle Vorgehensweise:

(i) Betrachte einen bestimmten Baum $t \in T$ un einen akzeptierenden Lauf ρ



(ii) Zeige dass in t und ρ folgendes vorkommt.

(iii) Erhalte duch Iteration das Stücks zwischen den beiden Endzuständen einen Baum t' und einen entsprechenden Lauf ρ'. t' ist nicht in T, wird aber von $\mathcal{A}$ akzeptiert.

(i) Zu $n \in \mathbb{N}$ definiere den Baum $t_n \in T$ durch: $t_n(u) = \begin{cases} b & \text{falls} a \in (1^*, 0)^i \\ a & \text{sonst} \end{cases}$ Ein b jeweils nach den ersten n Linksabzweigungen
Setze $n := |F| + 1$ und $t := t_n$
Sei ρ ein akz. Lauf auf t.

(ii) Gehe nach rechts bis ein Endzustand in ρ auftritt, und dann nach links usw. D.h. wähle $i_1, \ldots, i_n > 0$ mit
$\rho(1^{i_1} = q_1 \in F$
$\rho(1^{i_1}01^{1_2} = q_2 \in F$
$\rho(1^{i_1}01^{i_2}0 \ldots 1^{i_{n-1}}01^{i_n} = q_{n \in F}$
Da $n = |F| + 1$, existieren $j < k$ mit $q_j = q_k$ Setze $u_j = 1^{1_1}01^{1_2}0 \ldots 1^{i_j}$ und $u_k = 1^{1_1}01^{1_2}0 \ldots 1^{i_k}$.
Zwischen u_j und u_k kommt in t mindestens ein b vor.

(iii) Erhalte t' aus t und ρ' aus ρ durch Iteration des Stücks zwischen u_j und u_k. Dann ist ρ' ein akzeptierender Laus auf t'. Aber $t' \notin T$. Widerspruch □

Folgerung: Die Klasse der BBA erkennbaren Spraachen ist nicht unter Komplement abgeschlossen.

4.5 Muller- und Paritätsbaumautomaten

Satz 4.2:
1. Zu jedem PBA ex. ein äquivalenter MBA
2. Zu jedem MBA ex. ein äquivalenter PBA

Beweis:
Zu $c : Q \to \{0, \ldots, k\}$ definiere $\mathcal{F} = \{\rho \subseteq Q | max\{c(q)|q \in S\}\ gerade\}$
Zu 2. LAR-Konstrukution wie bei Spiuelreduktion von Muller- auf Paritätsspiele.
Zuständе: $LAR(Q) \times Q$
Transitionsfunktion entsprechend σ_{LAR}
Prioritätsfunktion wie in Spiel-reduktion. □

Lemma
Zu zwei PBAs $\mathcal{A}_1, \mathcal{A}_2$ kann man einen PBA für die Sprache $T(\mathcal{A}_1) \cup T(\mathcal{A}_2)$ konstruieren.

Beweis:
Vereinige Zustands und Transitionsmengen und füge neuen Anfangszustand hinzu. $cmitc_{|Q_1} = c_1 undc_{|Q_2} = c_2$□

Nächstes Ziel: Komplementabschluss von PBA's
Vorgehensweise Stelle das Akzeptieren $G_{\mathcal{A},t}$ dar, sodass Sp. 0 eine Gewinnstrategie hat, wenn t von $\mathcal{A}$ akzeptiert wird. Der Komplementautomat testet, ob Sp. 1 eine Gewinnstrategie hat.
Sei $\mathcal{A} = (Q, \Sigma, q_0, \Delta, c)$ ein PBA unt $t \in T_\Sigma$
$\mathcal{A}$ akz. $t \Leftrightarrow \exists$ Lauf ρ $\forall$ Pfad $\pi : \rho_{|\pi}$ erfüllt Paritätsbedingung. ($\exists$ Sp.0, $\forall$ Sp. 1)
Sp. 0 = Automat
Sp. 1 = Pfadsucher
Idee: Bild fehlt
Das Spiel $G_{\mathcal{A},t} = (V, E)$ mit $V = V_0 \cup V_1$
$V_0 = \{(w, t(w), q)|w \in \{0, 1\}^*, q \in Q\}$
$V_1 = \{(w, t(w), \underbrace{(q, t(w), q_1, q_2)}_{\in\Delta})\}$
$E = \{((w, t(w), q), (w, t(w), q_1, q_2))\} \cup \{((w, t(w), (q, t(w), q_1, q_2)), (w_i, t(wi), q_{i+1}))\}$
$i \in \{01, \}$
Prioritätsfunktion: $C_{\mathcal{A},t}((w, t(w), q)) = c(q)$
$c_{\mathcal{A},t}(v) = 0$ für alle $v \in V_1$

Lemma
Automat hat eine Gewinnstrategie in $(G_{\mathcal{A},t}, c_{\mathcal{A},t})$ mit Anfangsknoten $(\varepsilon, t(\varepsilon)q_0)$ genau dann, wenn $t \in T_\mathcal{A}$

Beweis:
Sei ρ ein akz. Lauf von $\mathcal{A}$ auf t. Definiere die Gewinnstrategie wie folgt:

Automat stellt sicher, dass nur Knoten der Form $(w, t(w), \rho(w))$ vorkommen. Automat wählt $(w, t(w), (\rho(W), t(W), \rho($
Eine solche Partie entspricht einem Pfad durch ρ. Da ρ akzpetierend ist, gewinnt Automat. Ist f eine Gewinnstrategie für Automat, definiere ρ induktiv wie folgt: $\rho(\varepsilon) = q_0$
$\rho(w0) = q_1$, falls $f(w, t(w), \rho(w)) = (w, t(w), (\rho(w), t(w), q_1, q_2)$
$\rho(w1) = q_2$, falls $f(w, t(w), \rho(w)) = (w, t(w), (\rho(w), t(w), q_1, q_2)$
Dann entspricht jeder Pfad durch ρ einer f-Partie. Somit ist ρ akzeptierend □.

23.01.06

Folgerung: Pfadsucher hat eine PGS in $G_{\mathcal{A},t} \Leftrightarrow t \notin T(\mathcal{A})$
Idee: Konstruiere PBA, der testet ob Pfadsucher eine PGS in $G_{\mathcal{A},t}$ hat.
Dazu: Hilfsinformationen an den Knoten von t.
Seien Σ, Γ Alphabete und $t \in T_\Sigma$, $s \in T_\Gamma$. Der Baum $t\hat{\ }s \in T_{\Sigma\times\Gamma}$ ist definiert durch $t\hat{\ }s(w) = (t(w), s(w))$.
Die Σ-Projektion von $t\hat{\ }s$ ist der Baum t
Ist $T \subseteq T_{\Sigma\times\Gamma}$, dann ist $proj_\Sigma(T) = \{t \in T_\Sigma | \exists s \in T_\Gamma\ t\hat{\ }s \in T|$

Lemma (Projektionslemma)
Für ein PBA $\mathcal{A}$ mit $T(\mathcal{A}) \subseteq T_{\Sigma\times\Gamma}$ kann man einen PBA $\mathcal{B}$ mit $T(\mathcal{B}) = proj_\Sigma(T(\mathcal{A}))$.

Beweis:
Erhalte $\mathcal{B}$ aus $\mathcal{A}$ wie folgt: ersetze jede Transition $(q, (a, a'), q_1, q_2)$ aus $\mathcal{A}$ durch (q, a, q_1, q_2)

□

Wir benutzen das Hilfsalphabet Γ = Menge der Abbildungen $h : \Delta \to \{0, 1\}$
Ein Baum $s \in T_\Gamma$ kodiert eine Strategie f_s für Pfadsucher in $G_{\mathcal{A},t}$
Sei dazu $w \in \{0, 1\}^*$ und $h = s(w)$.
Für $h((q, a, q_1, q_2)) = \begin{matrix} 0 \\ 1 \end{matrix}$ setze $f_s(w, t(w), (q, a, q_1, q_2)) = \begin{matrix} (w0, t(w0)q_1) \\ (w1, t(w1), q_2) \end{matrix}$ falls $t(w) = a$ Umgekehrt kann jede PGS für Pfadsucher durch einen Baum $s \in T_\Gamma$ kodiert werden. **Ziel:** Konstruiere PBA, der die Bäume $t\hat{\ }s$ akzeptiert, für die f_s eine Gewinnstrategie in $G_{\mathcal{A},t}$ ist.
Dazu konstruieren wir zunächst einen Wortautomaten, der für einzelne Pfade testet, ob die Strategie auf diesen Pfaden ok ist. Wir betrachten zu einem Pfad $\pi \in \{0, 1\}^\omega$ alle Transitionsfolgen die diesen Pfad generieren, wenn wir die Strategie f_s auf diese Transitionsfolgen anwenden.
Sei $\pi \in \{0, 1\}^\omega$, $t \in T_\Sigma$, $s \in T_\Gamma$.
Das Wort $\pi\hat{\ }t_{|\pi}\hat{\ }s_{|\pi}$ ist von der Form $(x_0, a_0, h_0)(x_1, a_1, h_1)\dots$ mit $x_i \in \{0, 1\}, a_i \in \Sigma, h_i \in \Gamma$.
Eine Folge $\tau_0, \tau_1, \tau_2 \dots \in \Delta^\omega$ mit $\tau_i = (q_i, b_i, q_i^0, q_i^1)$ heißt kompatibel mit $\pi, t_{|\pi}$ und $s_{|\pi}$ falls

1. $a_i = b_i$
2. $h_i(\tau_i) = x_i$
3. $q_{i+1} = q_i^{x_i}$

für alle $i \geq 0$
Die Transitionsfolge heißt gut, falls $q_0q_1q_2\dots$ die Paritätsbedingung von $\mathcal{A}$ nicht erfüllt.
Lemma: Pfadsucher gewinnt mit f_s genau dann, wenn für alle Pfade π gilt: alls Transitionsfolgen, die kompatibel sind mit $\pi, t_{|\pi}, s_{|\pi}$ auch gut sind.

Lemma
Wir können einen nichtdet. Muller-Wortautomaten $\mathcal{A}_1$ konstruieren der alle Worte der Form $\pi\hat{\ }\alpha\hat{\ }\beta \in (\{0, 1\} \times \Sigma \times \Gamma)^\omega$ akzeptiert, wenn es eine Transitionsfolge gibt, die kompatibel mit π, α, β ist und nicht gut.

Beweis:
$\mathcal{A}_1$ rät die Traistionsfolge nichtdet., verifiziert, ob sie kompatibel ist (nur lokale Bedingungen) und testet mit seiner Akzeptierbedingung ob die Folge nicht gut ist.

Lemma
Es gibt einen det. Muller-Wortautomaten $\mathcal{A}_2$, der allle Wörter der Form $\pi\hat{\ }\alpha\hat{\ }\beta$ (wie oben) akzeptiert für die jeden kompatible Transitionsfolge gut ist.

Beweis:
Determinisiere und komplementiere $\mathcal{A}_1$ (siehe VL von Prof. Thomas)

Lemma
Es gibt einen det. MBA, der alle Bäume der Formm $t \wedge s \in T_{\Sigma\times\Gamma}$ akzeptiert, für die f_s eine GS in $G_{\mathcal{A},t}$ für Pfadsucher ist.

Beweis:
Wende $\mathcal{A}_2$ auf alle Pfade an (vgl. Blatt 11, Aufgabe 21).

Satz 4.3: Zu einem PBA $\mathcal{A}$ kann man einen PBA $\bar{\mathcal{A}}$ konstruieren mit $T(\bar{\mathcal{A}}) = T_\Sigma \setminus T(\mathcal{A})$

Beweis:
Wende Projektionslemme auf $\mathcal{A}_3$ an und wandle diesen MBA in einen PBA um

□

4.6 MSO-Logik über dem unendlichen Binärbaum (S2S)

Erinnerung: (AiW) S1S: 1 Nachfolger - S2S: 2 Nachfolger

Beispiel:
$root(x) = \neg\exists y(S_0(y,x) \vee S_1(y,x))$
$x \sqsubseteq y$: x ist Präfix von y
$\forall x\,[(y \in X \wedge \forall z\,[(z0 \in X \vee z1 \in X) \rightarrow t \in X]) \rightarrow x \in X]$ (X ist unter Vorgängern abgeschlossen).
chain(x): Die Knoten in X sind linear geordnet $\forall x, y\,[(x \in X \wedge y \in X) \rightarrow (x \sqsubseteq y \vee y \sqsubseteq x)]$
$Path(X) : Chain(x) \wedge \neg\exists Y[Chain(Y) \wedge X \subseteq Y \wedge Y \neq Y]$
$X \subseteq Y : \forall x(x \in X \rightarrow x \in Y)$
$X = Y : X \subseteq Y \wedge Y \subseteq X$
$Sing(X) : \exists x(x \in X \wedge \forall y(y \in X \rightarrow x = y))$ Wenn wir $\subseteq$ und $Sing$ als atomare Formeln erlauben, dann können wir erste Stufe Variablen durch Mengen Kodieren:
$x \in Y \rightsquigarrow Sing(X) \wedge X \subseteq Y$
$X = Y \rightsquigarrow Sing(X) \wedge Sing(Y) \wedge X = Y$

Atomare Formeln: $S_0(X,Y)$ und $S_1(X,Y)$ bedeuten $X = \{x\}$, $Y = \{y\}$ und $S_1(x,y)$ bzw. $S_1(x,y)$
Ist $\varphi(X_1,\ldots,X_n)$ eine Formel mit n freien Mengenvariablen, wird diese in einem erweiterten Modell $t(T_2, P_1 \ldots P_n)$ mit $P_i \subseteq \{0,1\}^*$ interpretiert.
Wir schreiben $t \vDash \varphi(X_1,\ldots,X_n)$ falls φ in T_2 gibt, wobei die x_i jeweils durch P_i interpretiert werden. Das Modell t entspricht einem Baum $t \in T_{\{0,1\}^n}$ mit $t(n) = (c_1,\ldots,c_n)$ mit $c_i = 1 \Leftrightarrow w \in P_i$
30.1.06 Die durch $\varphi(X_1,\ldots,X_n)$ definiere Baumsprache ist $T(\varphi) = \{t \in T_{\{0,1\}^n} | t \vDash \varphi\}$

S1S äquivalent zu Büchi-Automaten: ω-Sprache über $\{0,1\}^n$: L S1S-def.bar $\Leftrightarrow$ L ist Büchi-erkennbar. **Plan:** Übertrage diese Ergebnisse auf S2S
Ziel: Äquivalenz von S2S und Paritätsbaumautomaten.

4.6.1 Binärbaum als relationale Struktur

$T_2 = (\{0,1\}^*, S_0, S_1)$ mit $S_0(u,v)$ für $v = u0$ und $S_1(u,v)$ für $v = u1$
Formeln

- Variablen $x, y, \ldots$ (interpretiert durch Knoten)

- Variablen $X, Y, \ldots$ (interpretiert durch Mengen von Knoten)
- Atomare Formeln $x \in X, S_0(x,y), S_1(x,y), x = y$
- Boolesche Kombinationen $\neg\varphi, \varphi \vee \psi, \ldots$
- Quantoren $\exists x(\varphi(x)), \exists X(\varphi(X))$

Schreibweisen

- Übliche Abkürzung für Boolesche Junktoren und $\forall$
- Direktes Anhängen von 0/1-Folgen an Variablen, z.B. $z0 \in X$ für $\exists(S_0(x,y) \wedge y \in X)$

Definierte Formeln: $root(x)$, $x \sqsubseteq y$, $Path(X)$
$\varphi(X_y, \ldots, X_n)$ freie Mengenvariable $X_1, \ldots, X_n$ erw. Struktur $\underline{t} = (T_2, P_1, \ldots, P_n)$, $P_i \subseteq \{0,1\}^*$
Label von n, $b_i = 1 \Leftrightarrow u \in P_i$
$T(\varphi) = \{t \in T_{\{0,1\}^n} | \underline{t} \vDash \varphi(X_1, \ldots, X_n)\}$

Von PBA zu S2S

Sei PBA $\mathcal{A} = (Q, \{0,1\}^*, q_1, \Delta, c)$ mit $Q = \{q_1, \ldots, q_m\}$
Lauf $\rho : \{0,1\}^* \to Q$
Lauf ρ ist akzeptierend, falls für alle Pfade $\pi \in \{0,1\}^\omega : \rho_{|\pi}$ erfüllt die Paritätsbedingung, d.h. $max(Inf(c(\rho_{|\pi})))$ ist gerade
$\mathcal{A}$ akzpetiert Baum $t \in T_{\{0,1\}^*}$ gdw. es ex. ein akzeptierender Lauf von $\mathcal{A}$ auf t.

Kodiere akz. Lauf durch Mengenvar. $Y_1 \ldots Y_m$

Lemma
Zu jedem PBA $\mathcal{A}$ mit $T(\mathcal{A}) \subseteq T_{\{0,1\}^n}$ existiert eine S2S-Formel $\varphi_{\mathcal{A}}(X_1, \ldots, X_n)$ mit $T(\varphi_{\mathcal{A}} = T(\mathcal{A})$

Sei $\mathcal{A} = (Q, \{0,1\}^n, q_1, \Delta, c)$ ein PBA mit $Q = \{q_1, \ldots, q_m\}$ und $c : Q \to \{0, \ldots, k\}$
S2S-Formel Partition $(Y_1, \ldots, Y_m) \sim Y_1 \ldots Y_m$ bilden eine Partition von $\{0,1\}^*$
$\forall x(\bigvee_{i=1}^m x \in Y_i \wedge \bigwedge_{i,j \in \{1,\ldots,m\} i \neq j} \neg(x \in Y_i \wedge x \in Y_j))$
Für $a = (b_1, \ldots, b_n) \in \{0,1\}^n$ sei "$x \in X_a$" eine Abkürzung für $\begin{cases} \neg x \in X_1 & \text{falls } b_1 = 0 \\ x \in X_1 & \text{falls } b_1 = 1 \end{cases} \wedge \cdots \wedge \begin{cases} \neg x \in X_n & \text{falls } b_n = 0 \\ x \in X_n & \text{falls } b_n = 1 \end{cases}$
Sei Z ein Pfad ("$Path(Z)$"), $0 \leq l \leq k$
S2S-Formel Parität$_l(Z) \sim$ die größte unendl. oft gesehene Priorität auf Z ist l.

$$\begin{aligned} \text{Parität}_l(Z) = \forall x \Big[x \in Z \to \exists y(x \sqsubseteq y \wedge x \neq y \wedge y \in Z \textstyle\bigvee_{i:c(q_j)=l} y \in Y_j) \Big] & \text{ "unendlich oft Priorität } l\text{"} \\ \wedge \exists x \Big[x \in Z \wedge \forall y(x \sqsubseteq y \wedge y \in Z \to \textstyle\bigvee_{l' \leq l} \bigvee_{j:c(q_j)=l'} y \in Y_j) \Big] & \text{ "nur endlich of höhere Prioritäten } > l'\text{"} \end{aligned}$$

Definiere $\varphi_{\mathcal{A}}(X_1, \ldots, X_n) := \exists Y_1 \ldots \exists Y_m(\; Partition(Y_1, \ldots, Y_m) \wedge \exists x(root(x) \wedge x \in Y_1) \wedge \forall z(\bigvee_{(q_i, a, q_{j1}, q_{j2} \in \Delta} z \in Y_i \wedge z \in X_a \wedge z0 \in Y_{j1} \wedge z1 \in Y_{j2})) \wedge$
$\forall Z \; (\; Path(Z) \to \bigvee_{0 \leq l \leq k, l gerade} \text{Parität}_l(Z) \;)$

□

von S2S zu PBA

Lemma
Sei $\varphi(X_1, \ldots, X_n)$ eine S2S Formel. Dann existiert ein PBA $\mathcal{A}_\varphi$ über $\{0,1\}^n$ so dass $T(\mathcal{A}_\varphi) = T(\varphi)$.
O.E. sei φ eine S2S$_0$-Formel, d,h.

- Quantifierzierung nur über Mengenvar.
- Atomare Formeln:
  - $Sing(X) \sim X$ ist Eine Menge
  - $X \subseteq Y$
  - $S_0(X, Y) \sim$ implizit: X und Y sind Einermengen
    "$X = \{x\}$", "$Y = \lfloor y\}$" und $S_0(x, y)$
  - $S_1(X, Y)$ analog

Bemerkung:
S2S und S2S$_0$ haben gleiche Ausdrucksstärke.

Bew des Lemma über Induktionnach Formelaufbau.

Beweis:
Ind-Anfang: $Sing(X)$ Sei PBA $\mathcal{A} = (\{q_+, q_0\}, \{0, 1\}, q_+, \Delta, c)$
$(q_+, 0, q_+, q_0), (q_+, 0, q_0, q_+), (q_+, 1, q_0, q_0), (q_0, 0, q_0, q_0)$

$X \subseteq Y$ O.E. Beschriftungen $\binom{b_1}{b_2}\begin{matrix}\sim X\\ \sim Y\end{matrix}$ [$b_1 = 1$ gdw. entspr. Knoten in X]
PBA $\mathcal{A} = (\{q_0\}, \{0, 1\}^2, q_0, \Delta, c), \Delta : c(q_0) = 0$

$S_0(X, Y)$: O.E. Beschriftungen $\binom{b_1}{b_2}\begin{matrix}\sim X\\ \sim Y\end{matrix}$
PBA $\mathcal{A} = (\{q_+, q_0, q_1\}, \{0, 1\}^2, q_+, \Delta, c), \Delta : c(q_0) = 2, c(q_+) = c(q_1) = 1$ Induktionsschritt: Für $\psi, \varphi_1, \varphi_2$ seien bereits äquivalente PBAs $\mathcal{A}_\psi, \mathcal{A}_{\varphi_1}, \mathcal{A}_{\varphi_2}$ gegeben.

1. $\varphi = \varphi_1(X_1, \ldots, X_n) \vee \varphi_2(X_1, \ldots, X_n) : \mathcal{A}_{\varphi_\infty \vee \varphi_\in}$ Automat ist Vereinigungsautomat von $\mathcal{A}_{\varphi_1}$ und $\mathcal{A}_{\varphi_2}$
2. $\varphi(X_1, \ldots, X_n) = \neg\psi(X_1, \ldots, X_n\ \mathcal{A}_\varphi$ ist Komplementautomat zu $\mathcal{A}_\varphi$ (geht nicht mit BBA)
3. $\varphi(X_1, \ldots, X_n) = \exists\psi(X_1, \ldots, X_n\ \mathcal{A}_\varphi$ gemäß Projektionslemma

Zusammen:

Satz 4.4: Eine Baumsprache $T \subseteq T_{\{0,1\}^n}$ ist genau dann S2S-definierbar falls sie PBA-erkennbar ist.

6.02.06 **Anwendung:** "Rabin Baum-Theorem": das Erfüllbarkeitsproblem für S2S ist entscheidbar

Anwendung: Erfüllbarkeitstest für S2S-Formeln
Das Erfüllbarkeitsproblem: Gegeben: S2S-Formel $\varphi(X_1, \ldots, X_n)$
Frage: Hat φ ein Modell (ist $T(\varphi) \neq \emptyset$) Ansatz: Konstruiere den PBA A_φ und teste, ob $T(A_\varphi) \neq \emptyset$

4.6.2 Leerheitstest für PBA's

Erinnerung: Für festes t, kann die Frage $t \in T(\mathcal{A})$ auf ein Spiel $(G_{\mathcal{A},t}, c_{\mathcal{A},t})$ reduziert werden. In $(G_{\mathcal{A},t})$ konstruiert der Automat einen akzeptierenden Lauf auf t.
In dem Spiel $G_{\mathcal{A}}$, das wir angeben werden, konstruiert der Automat gleichzeitig einen Lauf und einen Eingabe-baum. Wir benötigen dafür im Spielgraphen nicht mehr explizit die Baumknoten. Die aktuelle Position im Baum wird durch den bisherigen Spielverlauf gegeben.
Sei $\mathcal{A}$ ein PBA mit Prioritätsfunktion c. Das Paritätsspiel $(G_{\mathcal{A}}, c_{\mathcal{A}})$ ist definiert wie folgt: $G_{\mathcal{A}} = (V, E)$ mit $V = Q \cup \Delta, V_0 = Q, V_1 = \Delta$
$E = \{(q, (q, a, q_1, q_2)), ((q, a, q_1, q_2), q_1), ((q, a, q_1, q_2), q_2) | (q, a, q_1, q_2) \in \Delta\}$ $c_{\mathcal{A}}(q) = c(q)$ für alle $q \in Q$
$c_{\mathcal{A}}(\tau) = 0$ für alle $\tau \in \Delta$
Beispiel:
Akzeptierte Sprache

- Linker Pfad: nur a
- Vom linken Pfad immer wieder Abzweigungen nach rechts, die Schließlich zu einem b führen

Satz 4.5: Der Spieler Automat hat in $(G_{\mathcal{A}}, c_{\mathcal{A}})$ eine Gewinnstrategie gdw. $T(\mathcal{A}) \neq \emptyset$ gilt.

Beweis:
Wie für das Spiel $(G_{\mathcal{A},t}, c_{\mathcal{A},t})$

Folgerung: Das Leerheitsproblem für PBA's ist entscheidbar.

Beweis:
Konstruiere zu PBA $\mathcal{A}$ das Spiel $(G_{\mathcal{A}}, c_{\mathcal{A}})$ und löse dies. □

Folgerung: Das Erfüllbarkeitsproblem für S2S ist entscheidbar.

Beweis:
Konstruiere zu $\varphi(X_1, \ldots, X_n)$ den Automaten $\mathcal{A}_\varphi$ und löse das Leerheitsproblem für diesen □

Ein Satz ist eine Formel ohne freie Variablen. **Folgerung:** Die Theorie S2S (Menge der wahren S2S Sätze) ist entscheidbar

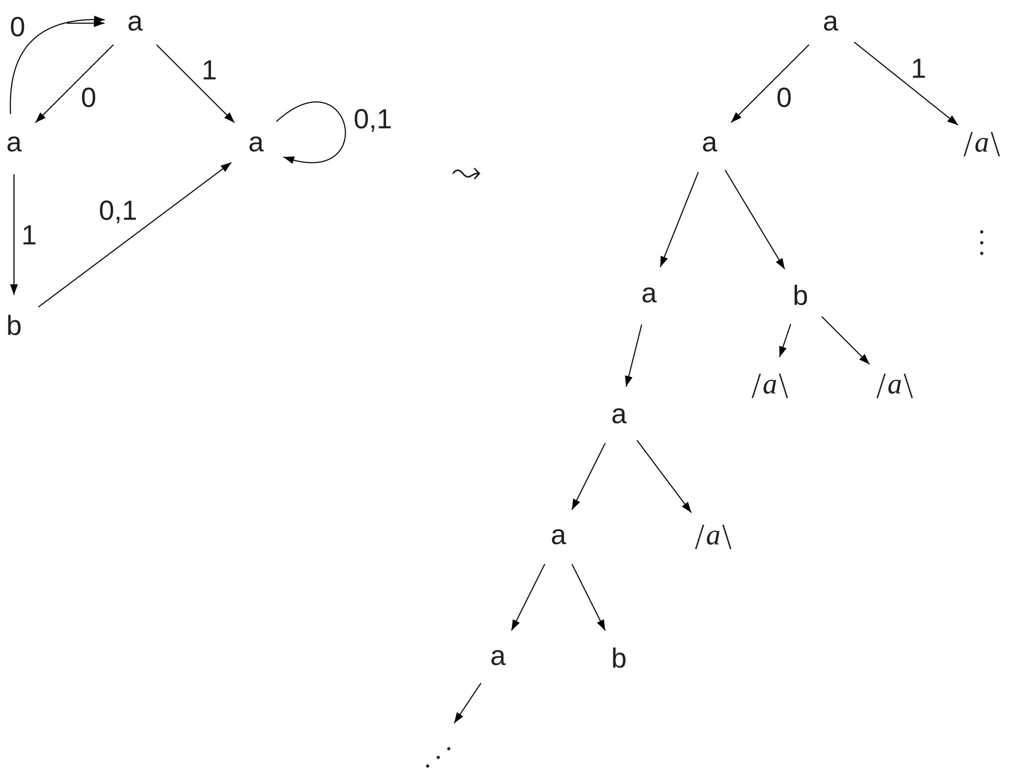
Beweis:
Ein Satz definiert $\{0, 1\}^0$-beschriftete Bäume. $\{0, 1\}^0$ ist eine einelementige Menge. Der äquivalente PBA hat also nur einen Eingabebaum. Ob dieser akzeptiert wird, kann mit dem Leerheitstest überprüft werden.

4.7 Reguläre Bäume

Entsprechung zu schließlich periodischen Wörtern uv^ω
Als Abrollung eines endlichen Graphen, z.B. für $abb(cab)^\omega$

$$\begin{tikzcd} a \arrow[r] & b \arrow[r] & b \arrow[r] & c \arrow[r] & a \arrow[r] & b \arrow[ll, bend left] \end{tikzcd}$$

Erweiterung für Binäre Bäume:

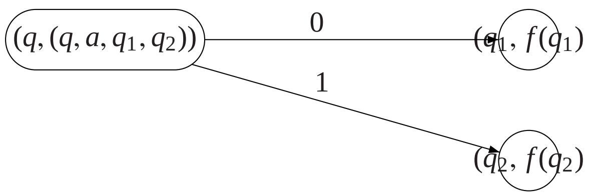


Beispiel:

Ein baum heißt regulär, wenn er der Abrollung eines endlichen Graphen mit fogenden Eigenschaften entspricht:

- Es gibt einen Anfangsknoten, bei dem die Abrollung beginnt
- Die Knoten sind mit Σ beschriftet
- Jeder Knoten hat genau zwei ausgehende Kanten, eine mit 0, die andere mit 1 beschriftet.

Aus einer pos. Gewinstrategie für Automat in $(G_{\mathcal{A}}, c_{\mathcal{A}})$ kann man einen regulären Baum aus $T(\mathcal{A})$ konstruieren.
Sei f eine pos. Gewinnstrategie in $(G_{\mathcal{A}}, c_{\mathcal{A}})$ für Automat. Der reguläre Baum in $T(\mathcal{A})$ entspricht der Abrollung des folgenden Graphen:
Die Knoten des Graphen sind die Paare $(q, f(q))$ für alle $q \in Q$



Kanten:
Beschriftung von $(q, (q, a, q_1, q_2))$ ist a.

Satz 4.6: Jede nicht leere Baumsprache die durch einen PBA akzeptiert werden kann, enthält einen regulären Baum.

Folgerung: Zu jeder erfüllbaren S2S-Formel kann man einen endliche darstellbares Modell konstruieren.

Inhaltsverzeichnis