Formale Aspekte des DNA-Computing

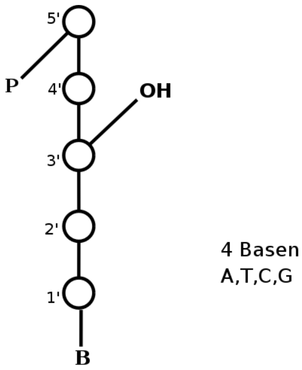
gelesen von
Prof. Dr. Dietrich Kuske

im
Sommersemester 2005

Inhaltsverzeichnis

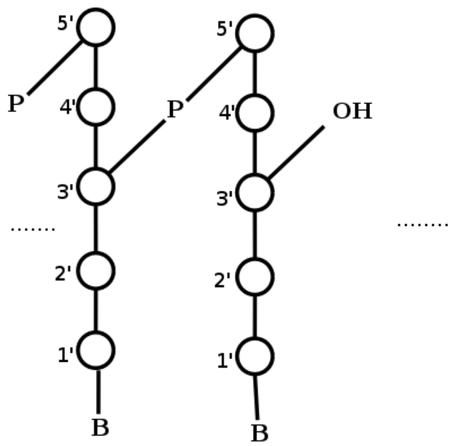
1 Schematisch – Chemischer – Hintergrund

1. Bausteine



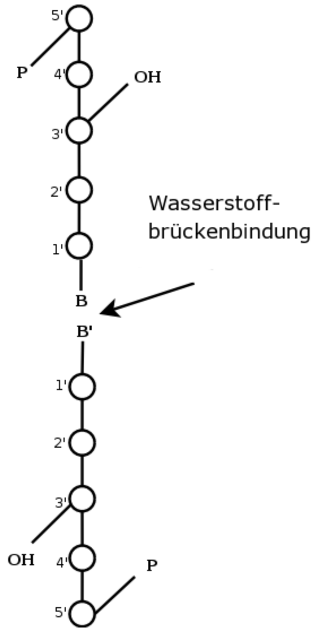
2. Bindungen

   (a) **Verbindung der P-&-OH-Gruppen**

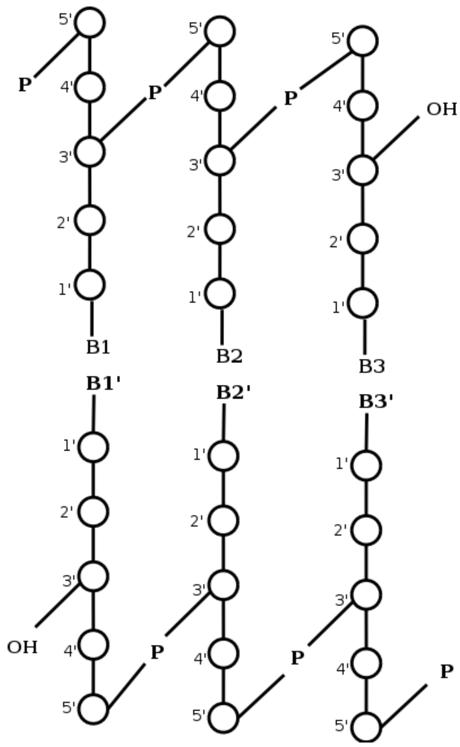


   $\Rightarrow$ Ergebnis Kette von Nukleotiden ($\neq$ einsträngige DNA).
   `Bezeichnung:` $(5'-)BB'B''\ldots$ oder $(3'-)B''B'B\ldots$
   Nach Konvention kann man das 5' am Anfang auch weglassen.
   Verkettung von P und OH $\Rightarrow$ kovalent also stabil, d.h. einsträngige DNA existiert.

   (b) **Verbindung der Basen**
   mit $B, B' = A, T$
   oder $B, B' = C, G$
   Die entstandenen Bindungen sind instabil ($\Rightarrow$ keine Kettenbildung).



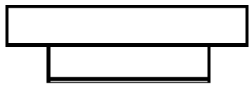
⇒ Gleichzeitige Verwendung beider Verbindungen erzeugt doppelsträngige DNA



1.1 Operationen

1. **(De)-Naturierung**
   `allgemein:`
   - Wärme zerstört Moleküle
   - doppelsträngige DNA zerlegt sich, bei 85C - 95C, in die 2 Eeinzlstränge

`Naturierung:` Annealing liefert wieder 2-Strängige DNA, aber unter Umständen unvollständig (siehe Grafik).



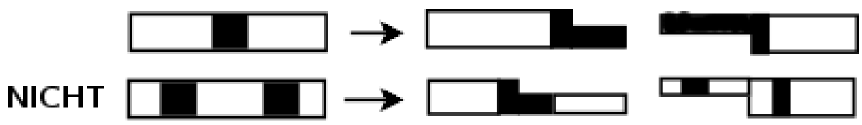
2. **Verkürzen von DNA**

   (a) Exonucleasen: Entfernen der letzten Nukleotide, ein oder Doppelsträngiger DNA

   (b) Endonukleasen: Zerschneiden DNA im Inneren

```
...GATTCG...     ...GA      TTCG...
            =>           &
...CTAAGC...     ...CTAA      GC...
```

Auch für einsträngige DNA möglich
aber nur lokal, d.h.



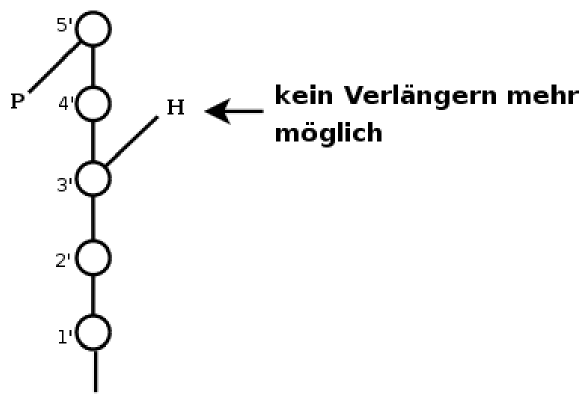
→ Überhänge müssen die gleiche Länge haben und komplementär sein

3. **Verbinden von DNA (Ligation)**

   (a) Führen spezifisch überhänge zusammen

   (b) Führen spezifisch vollständige Molekülenden zusammen. z.B.:
   $\ldots GATT$ & $CG\ldots$
   $\ldots GATT$ & $CG\ldots$

Problem
`Vorhanden:` Lösung mit Molekülen $\alpha\&\beta$ ($\alpha, \beta \in \{A, G, C, T\}^+$)
`Ziel:` Lösung mit Molekül $\alpha\beta$ aber <u>nicht</u> $\alpha\alpha$
`erste Idee:` Zusammenkippen und vollständige Enden verbinden → auch $\alpha\alpha$ existiert. Geht also nicht ;0(
`zweite Idee:` Verhindern, dass sich Ende von α mit Anfang von α verbindet.
ersetze P-Gruppe am 5'-Ende von α und OH-Atom
`Ergebnis:` α' mit linkem Rand



so entstehen nach Ligation $\alpha'\beta$ und $\beta\beta$
dann ersetze OH in $\alpha'\beta$ durch $P \curvearrowright \alpha\beta$

4. **Polymerase**
`Ausgangspunkt:` unvollständigie 2-Strängige DNA
z.B.:

```
  TCG
CGAGCCTAG
```

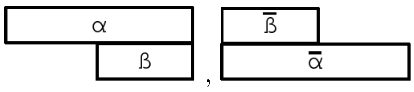
Polymeraseenzyme binden passendes Nukleotid an 3'-Ende des kurzen Strangs
es entsteht:

```
  TCGGATC
CGAGCCTAG
```

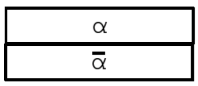
5. **Polymerase-Kettenreaktion**
`Ziel:` Vervielfältigung einer Menge von DNA.
`Ausgangspunkt:` Lösung, die doppelsträngige DNA mit Strängen α bzw. $\overline{\alpha}$ enthält.
`Zugabe:`

- Primer, die koplementär zum 5'-Ende von α bzw. $\overline{\alpha}$ sind (β und $\overline{\beta}$)
- Nukleotide
- Polymerase.

`Vorgehen:`

- mischen DNA und Zugaben: $\binom{\alpha}{\overline{\alpha}}$, β, $\overline{\beta}$
- Erwärmen auf ca 85C - 95C:
$\begin{matrix}\alpha \\ \overline{\alpha}\end{matrix}$, $\beta, \overline{\beta}$
- leichtes Abkühlen: $\begin{matrix}\alpha \\ \overline{\alpha}\end{matrix}$, 

(a) **Polymerase- Reaktion**



`Ergebnis:` Anzahl der Moleküle $\begin{matrix}\alpha \\ \overline{\alpha}\end{matrix}$ (ca.) verdoppelt.

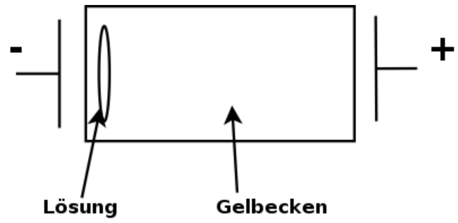
(b) **Herausfiltern**
`Ziel:` Herausfiltern aller DNA-Moleküle, die gegebene Teilsequenz β enthalten.
`Methode:`

- fixiere Komplementärmoleküle $\overline{\beta}$ an z.B. Metallkugel
- danach entferne Metallkugel mit daran klebenden DNA-Molekülen aus der Lösung
- löse durch Denaturierung DNA-Stränge mit β von $\overline{\beta}$

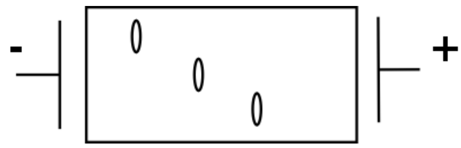
(c) **Gel-Elektorphorese**
`Ziel:` Festellen der vorkommenden Längen von DNA-Molekülen in einer Lösung.
`Grundlage:`

- DNA-Moleküle sind negativ geladen
- kurze DNA-Stränge wandern leichter durch ein Gel zum pos. Pol eines Feldes

`Versuchsanordnung:`



nach gewisser Zeit:

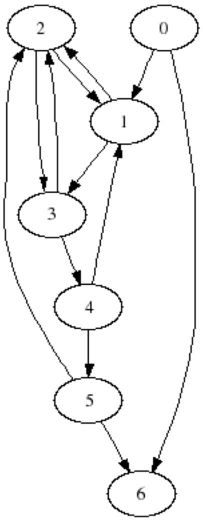


⇒ aus Zeit, Ort und Stromstärke lässt sich die Länge der DNA-Fragmente berechnen oder auch auftragen eines Makers → es wird die relative Lage betrachtet

2 Klassische DNA-Berechnung

2.1 Adlemans Experiment (Since '94)

betrachte folgenden Graphen:
(nicht planar da 1,3,5 und 2,4,6 ein $K_{3,3}$ bilden)



Frage:
Existiert ein Hamiltonscher Pfad (Es gibt einen Pfad durch den Graphen bei dem jeder Knoten genau einmal besucht wird [0,1,2,3,4,5,6]) in diesem Graphen?

Idee:

- Rate alle Tupel von Knoten auf einmal
- sortiere diejenigen Tupel um, die kein Hamiltonscher Pfad sind

Realisierung:

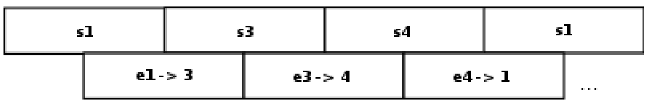
- für jeden Knoten i wähle ein DNA-Molekül der Länge 20
- z.B.

$$s_2 = TATCGGATCG|GTATATCCGA$$
$$s_5 = GGCTAGGTAC|CAGCATGCTT$$

  Bedingung:
  kein Anfangs- oder Endstück der Länge 10 darf 2-mal auftreten

- für jede Kante (z.B. 5→2):
  - zerlege $s_2 = s_2's_2''$ und $s_5's_5''$ der Länge 10
  - bilde Komplement $e_{5\to 2}$ von $s_5''s_2'$
    $\Rightarrow s_{2K}' = ATAGCCTAGC; s_{5K}'' = GTCGTACGAA$ (K steht für Komplement)
    $e_{5\to 2} = GTCGTACGAA|ATAGCCTAGC$
- stelle Lösung her, die genau die Moleküle $s_i (0 \leq i \leq 6)$ und $e_{i\to j}$ für (i,j) ∈ E enthält und führe Ligation herbei

Ergebnis:
in der Lösung befinden sich dann z.B. die Moleküle



und

d.h. jedes Molekül beschreibt einen Pfad und jeder Pfad der Länge n hat gleiche Wahrscheinlichkeit des Vorkommens. Hinreichend viel Lösung sichert, dass jeder Pfad der Länge 7 mit hoher Wahrscheinlichkeit vorkommt. Jetzt werden die folgenden Schritte ausgeführt:

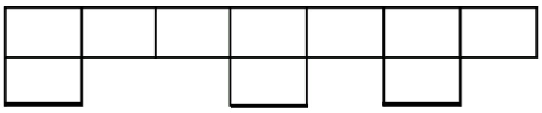
- Herausfiltern der Stränge der Länge $7 \cdot 20 = 140$
- Herausfiltern der Stränge die s_0 enthalten
- Herausfiltern der Stränge die s_1 enthalten
- ...
- Herausfiltern der Stränge die s_6 enthalten

Wenn jetzt ein DNA-Molekül in der Lösung verbleibt, so hat der Graph einen Hamiltonschen Pfad.

2.2 Sticker-Modelle (Roveis et al '96)

Idee:

- betrachte partiell 2-strängige Moleküle der Form

  als Kodierung von 0-1-Vektoren (Position n besitzt Komplement $\triangleq$ Vektor der an Stelle n eine 1 enthält).
- damit sind alle Vektoren in Lösung gleichzeitig kodierbar
- Beispiel: SAT (jeder 0-1-Vektor ist eine geratene Variable)

  1. erzeuge 1-strängige DNA der Länge = Anzahl der Variablen · Stickerlänge
  2. erzeuge Komplemente der Stickerpositionen
  3. hieran erzeuge durch Annealing alle partiellen 2-strängigen DNA-Moleküle
  4. für jede Klausel aus Formel mit n Variablen
     - teile Lösung in n gleiche Teile
     - teile jedes Teil in 2 Teile:
       * im 1. Teil ist Stickerposition i besetzt

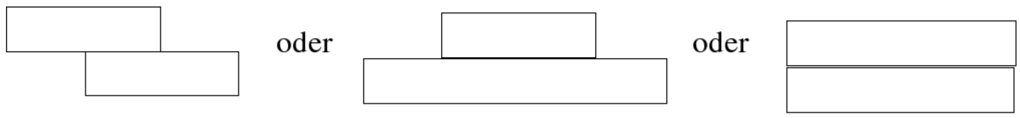
    * im 2. Teil nicht
  - kommt Variable x_i in Klause positiv vor, so behalte ersten Teil sonst den zweiten
  - vermische die verbleibenden Teile

5. teste am Ende, ob ein DNA-Molekül in der Lösung verbleibt
   $\Rightarrow$ dieses kodiert ggf. eine gültige Belegung der Formel

3 Stickersysteme

Haben nichts mit den Stickermodellen zu tun!!!

Grundbausteine der Berechnung:
DNA-Moleküle der Form



`'Berechnung'` kombiniert diese Bausteine zu komplexeren Molekülen. Das `Ergebnis der Berechnung` ist die Menge der vollständigen Moleküle, die so entstehen (bzw. deren obere Schranke).

Verallgemeinerung:

- nicht nur 4 Basen, sondern Basenmenge V
- Komplentarität $\rho \subseteq V \times V$ nicht unbedingt eindeutig, noch notwendig symmetrisch ($\curvearrowright$ A|T ist nicht muss)

3.1 Dominos (Bausteine) & deren Verkettung

V ... (endliches, nichtleeres) Alphabet $\rightarrow$ V{A,G,C,T}

$\rho \subseteq V \times V \ldots$ Komplementaritätsrelation $\rightarrow \rho =$ {(A,T),(T,A),(G,C),(C,G)}

$V^* \ldots$ Menge der Wörter über V inklusive ϵ

$V^* \times V^* \ldots$ Menge der Paare von Wörtern
binär Operation $(u, v) \cdot (\overline{u}, \overline{v}) = (u, \overline{u})(v, \overline{v})$
Schreibweise:

$$(u, v) \text{ für } \binom{u}{v} \in V^* \times V^*$$
$$\binom{X}{Y} \text{ für } X \times Y; \text{ mit } X, Y \subseteq V^*$$
$$\binom{X}{u} \text{ für } \binom{X}{\{u\}}; \text{ falls } u \in V^*, x \leq V^*$$

$\rho^* \ldots$ Menge der Wörter über ρ,
$\rho^+ = \rho^* \backslash \{\epsilon\} \rightarrow \binom{A}{T}, \binom{T}{A}, \binom{G}{C} \rightarrow$ Wort der Länge 3
Schreibweise:

$$\binom{x_1}{y_1}\binom{x_2}{y_2}\cdots\binom{x_n}{y_n} = \left[\binom{x_1}{y_1}\binom{x_2}{y_2}\cdots\binom{x_n}{y_n}\right];$$
$$\text{falls} \binom{x_i}{y_i} \in \rho, \text{ für } 1 \leq i \leq n$$

$$\left.\begin{array}{l}\rho^* \ldots \text{ Wörter von Paaren}\\ \binom{V^*}{V^*} \ldots \text{ Paare von Wörtern}\end{array}\right\} \rho^* \nsubseteq \frac{V^*}{V^*}$$

$O(V) \ldots$ Menge der einsträngigen Dominos

$$\begin{aligned} O(V) &= \begin{pmatrix} V^* \\ \epsilon \end{pmatrix} \cup \begin{pmatrix} \epsilon \\ V^* \end{pmatrix} \\ &= \left\{ \begin{pmatrix} u \\ v \end{pmatrix} : u, v \in V^*, u = \epsilon \text{ oder } v = \epsilon \right\} \\ \text{wobei } & \begin{pmatrix} u \\ \epsilon \end{pmatrix} \neq \begin{pmatrix} \epsilon \\ u \end{pmatrix}; \text{ falls } u \neq \epsilon \end{aligned}$$

$W(V, \rho) = W_\rho(V) = (O(V) \times \rho^+ \times O(V)) \cup O(V)$
(W Watson-Crick Domain (Menge der Dominos))

Beispiel

$$W(V,\rho) \ni \left(\begin{pmatrix} AG \\ \epsilon \end{pmatrix}, \begin{bmatrix} \text{ACT} \\ \text{TGA} \end{bmatrix}, \begin{pmatrix} \epsilon \\ AG \end{pmatrix} \right) = \begin{array}{l} \boxed{\text{AGACT}} \\ \qquad \boxed{\text{TGAAG}} \end{array}$$

$$W(V,\rho) \ni \begin{pmatrix} Ag \\ \epsilon \end{pmatrix} = \begin{array}{c} \boxed{\text{AG}} \\ \boxed{\phantom{\text{AG}}} \end{array} = 5 - AG$$

$$W(V,\rho) \ni \begin{pmatrix} \epsilon \\ GA \end{pmatrix} = \begin{array}{c} \boxed{\phantom{\text{GA}}} \\ \boxed{\text{GA}} \end{array} = 3 - GA = 5 - AG$$

$\underline{\underline{\text{aber}}} \begin{pmatrix} AG \\ \epsilon \end{pmatrix} \neq \begin{pmatrix} \epsilon \\ GA \end{pmatrix}$ obwohl sie das selbe Molekül modellieren.

Schreibweise

$$\begin{aligned} \left(\begin{pmatrix} u_1 \\ v_1 \end{pmatrix}, \begin{bmatrix} u_2 \\ v_2 \end{bmatrix}, \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} \right) &= \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \begin{pmatrix} u_3 \\ v_3 \end{pmatrix} \\ \begin{pmatrix} \epsilon \\ \epsilon \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \begin{pmatrix} u_3 \\ v_3 \end{pmatrix} &= \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \begin{pmatrix} u_3 \\ v_3 \end{pmatrix} \\ \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \begin{pmatrix} \epsilon \\ \epsilon \end{pmatrix} &= \begin{pmatrix} u_1 \\ v_1 \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \\ \begin{pmatrix} \epsilon \\ \epsilon \end{pmatrix} \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \begin{pmatrix} \epsilon \\ \epsilon \end{pmatrix} &= \begin{bmatrix} u_2 \\ v_2 \end{bmatrix} \end{aligned}$$

$\Rightarrow$ wir betrachten ρ^+ als Teilmenge von $W(V, \rho)$
für $m \in W(V, \rho)$ sei h(m) die überhanglänge definiert durch:

$$h(m) = \begin{cases} \max\{|u|,|v|\}; \text{ falls m} = \binom{u}{v} \in O(V) \\ \max\{|u_1|,|v_1|,|u_3|,|v_3|\}; \text{ falls } m = \binom{u_1}{v_1}\begin{bmatrix} u_2 \\ v_2 \end{bmatrix}\binom{u_3}{v_3} \end{cases}$$

Beispiel

$$\begin{bmatrix} \text{AG} \\ \text{TC} \end{bmatrix} \dot{\rho} \begin{bmatrix} \text{GCA} \\ \text{CGT} \end{bmatrix} = \begin{bmatrix} \text{AGGCA} \\ \text{TCCGT} \end{bmatrix}$$
$$\begin{bmatrix} \text{AG} \\ \text{TC} \end{bmatrix} \begin{pmatrix} CA \\ \epsilon \end{pmatrix} \dot{\rho} \begin{pmatrix} \epsilon \\ GT \end{pmatrix} = \begin{bmatrix} \text{AGCA} \\ \text{TCGT} \end{bmatrix}$$
$$\begin{pmatrix} AA \\ \epsilon \end{pmatrix} \dot{\rho} \begin{pmatrix} \epsilon \\ TT \end{pmatrix} \text{ undefiniert}$$
$$\begin{bmatrix} \text{AG} \\ \text{TC} \end{bmatrix} \begin{pmatrix} CA \\ \epsilon \end{pmatrix} \dot{\rho} \begin{pmatrix} \epsilon \\ TG \end{pmatrix} \text{ undefiniert}$$

$\dot{\rho}$ entspricht dem Verknüpfungsoperator

Definition 3.1 (Verkettung von Dominos)
Die Verkettung von Dominos über (V, ρ):

1. $\binom{u_1}{\epsilon}\dot{\rho}\binom{u_2}{\epsilon} = \binom{u_1u_2}{\epsilon}$ und $\binom{\epsilon}{v_1}\dot{\rho}\binom{\epsilon}{v_2} = \binom{\epsilon}{v_1v_2}$

2. $\left(\binom{u_1}{u_2}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}\binom{w_1}{w_2}\right) \dot{\rho} \underbrace{\begin{pmatrix} x_1 \\ x_2 \end{pmatrix}}_{\in O(V)} = \binom{u1}{u2}\begin{bmatrix} v_1y_1 \\ v_2y_2 \end{bmatrix}\binom{z_1}{z_2}$

   - mit $w_1x_1 = y_1z_1$ und $w_2x_2 = y_2z_2$
   - $|y_1| = |y_2| = min\{|w_1x_1|, |w_2x_2|\}$
   - und $\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$

3. $\binom{x_1}{x_2}\dot{\rho}\left(\binom{u_1}{u_2}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}\binom{w_1}{w_2}\right)$ analog

4. $\binom{u_1}{u_2}\begin{bmatrix} v_1 \\ v_2 \end{bmatrix}\binom{w_1}{w_2}\dot{\rho}\binom{x_1}{x_2}\begin{bmatrix} y_1 \\ y_2 \end{bmatrix}\binom{z_1}{z_2} = \binom{u_1}{u_2}\left[\binom{v_1w_1x_1y_1}{v_2w_2x_2y_2}\right]\binom{z_1}{z_2}$; falls $\left[\binom{w_1x_1}{w_2x_2}\right] \in \rho^*$

5. $m \cdot m'$ ist undefiniert wenn keiner der Fälle 1. - 4. zutrifft

Definition 3.2 (Stickersystem)
Ein Stickersystem ist ein Quadrupel $\gamma = (V, \rho, A, D)$, wobei

- V ein Alphabet
- $\rho \subseteq V^2$ eine Komplementaritätsrelation
- $A \subseteq LR_\rho(V) := W_\rho(V) \backslash O(V)$ eine endliche Menge von Axiomen (Startsymbol) und

- $D \subseteq W_\rho(V) \times W_\rho(V)$ eine endliche Menge von Regeln ist

Sei $\gamma \in (V, \rho, A, D)$ Stickersystem und $x, y \in LR_\rho(V)$
$x \rightarrow_\gamma y :\Leftrightarrow$ sei Regel $(u, v) \in D$ mit $y = u \cdot \rho x \cdot \rho v$

Definition 3.3 (Molekülsprache)
Sei $\gamma = (V, \rho, A, D)$ ein Stickersystem. Die Molekülsprache $LM(\gamma)$ ist die Menge der vollständigen Dominos $y \in \rho^+$, die aus Axiomen abgeleitet werden können:

$$LM(\gamma) := \{y \in \rho^+ : \exists x \in A : x \rightarrow_\gamma^* y\}$$

Die Sprache $L(\gamma)$ ist die Menge der oberen Stränge von Elementen von $LM(\gamma)$:

$$L(\gamma) := \left\{ u \in V^+ : \exists v \in V^+ : \begin{bmatrix} \text{u} \\ \text{v} \end{bmatrix} \in LM(\gamma) \right\}$$

Beispiel 3.4

$LM(\gamma)$ mit Stickersystem $\gamma = (V, \rho, A, D)$
$V = \{a, b, c, d\}$
$\rho = \{(a, a), (b, b)(c, c), (d, d)\}$
$A = \left\{ \begin{bmatrix} \text{c} \\ \text{c} \end{bmatrix} \right\}$
$D = \left\{ \left(\binom{b}{\epsilon}, \binom{d}{\epsilon} \right), \left(\binom{a}{\epsilon}, \binom{\epsilon}{\epsilon} \right), \left(\binom{\epsilon}{b}, \binom{\epsilon}{\epsilon} \right), \left(\binom{\epsilon}{a}, \binom{\epsilon}{d} \right) \right\}$
Sei
$L_1 = \left\{ \begin{bmatrix} \text{a} \\ \text{a} \end{bmatrix}^* \begin{bmatrix} \text{b} \\ \text{b} \end{bmatrix}^* \begin{bmatrix} \text{c} \\ \text{c} \end{bmatrix} \begin{bmatrix} \text{d} \\ \text{d} \end{bmatrix} \right\}$ und
$L_2 = \left\{ \begin{bmatrix} \text{a} \\ \text{a} \end{bmatrix}^m \begin{bmatrix} \text{b} \\ \text{b} \end{bmatrix}^m \begin{bmatrix} \text{c} \\ \text{c} \end{bmatrix} \begin{bmatrix} \text{d} \\ \text{d} \end{bmatrix}^m : m \in \mathbb{N} \right\}$

Behauptung
$LM(\gamma) \cap L_1 = L_2$

Beweis
$\leq$

Sei $x = \begin{bmatrix} \text{a} \\ \text{a} \end{bmatrix}^k \begin{bmatrix} \text{b} \\ \text{b} \end{bmatrix}^l \begin{bmatrix} \text{c} \\ \text{c} \end{bmatrix} \begin{bmatrix} \text{d} \\ \text{d} \end{bmatrix}^m \in LM(\gamma) \cap L_1$
wegen $x \in LM(\gamma)$ existiert x_i $W(V, \rho)$ mit $x_0 \in A$, $x_i \rightarrow x_{i+1}, x_n = x$ für $0 \leq i < n$
$\Rightarrow$ da $\left(\binom{\epsilon}{a}, \binom{\epsilon}{d}\right)$ die einzige Regel ist, die a im 2. Strang einführt, folgt m=k.
Da $\left(\binom{b}{\epsilon}, \binom{d}{\epsilon}\right)$ einzige Regel, die b bzw. d im 1. Strang einführt, gilt auch l = m
$\Rightarrow x \in L_2$,
$\Rightarrow$ d.h. $LM(\gamma) \cap L_1 \subseteq L_2$

$\geq$

Schritt	angewendete Regel
$\begin{bmatrix} c \\ c \end{bmatrix}$	$(\binom{\epsilon}{b}, \binom{\epsilon}{\epsilon})$
$\to^m \binom{\epsilon}{b} \begin{bmatrix} c \\ c \end{bmatrix} \binom{\epsilon}{\epsilon}$	$\left(\binom{b}{\epsilon}, \binom{d}{\epsilon}\right)$
$\to^m \begin{bmatrix} b \\ b \end{bmatrix}^n \begin{bmatrix} c \\ c \end{bmatrix} \binom{d}{\epsilon}^m$	$(\binom{\epsilon}{a}, \binom{\epsilon}{d})$
$\to^m \binom{\epsilon}{a}^m \begin{bmatrix} b \\ b \end{bmatrix}^m \begin{bmatrix} c \\ c \end{bmatrix} \begin{bmatrix} d \\ d \end{bmatrix}^m$	$(\binom{a}{\epsilon}, \binom{\epsilon}{\epsilon})$
$\to^m \begin{bmatrix} a \\ a \end{bmatrix}^m \begin{bmatrix} b \\ b \end{bmatrix}^m \begin{bmatrix} c \\ c \end{bmatrix} \begin{bmatrix} d \\ d \end{bmatrix}^m$	

$\Rightarrow L_2 \subseteq LM(\gamma) \cap L_1$

`damit ist die Behauptung gezeigt`

Behauptung
$L(\gamma)$ ist kontextfrei

Beweis
angenommen $L(\gamma)$ wäre kontextfrei
$\Rightarrow LM(\gamma) \cap L_1$ kontextfrei (übung 2.3)
Widerspruch zu L_2 nicht kontextfrei (übung 2.2)

`q.e.d.`

3.2 Erzeugte Sprachen

Jede Stickersprache ist kontextsensitiv $\to$ sie wird somit von einem LBA akzeptiert

Lemma 3.5 (Peter Weigel '04)
Sei γ ein Stickersystem. Dann gilt $L(\gamma) \in NL$.

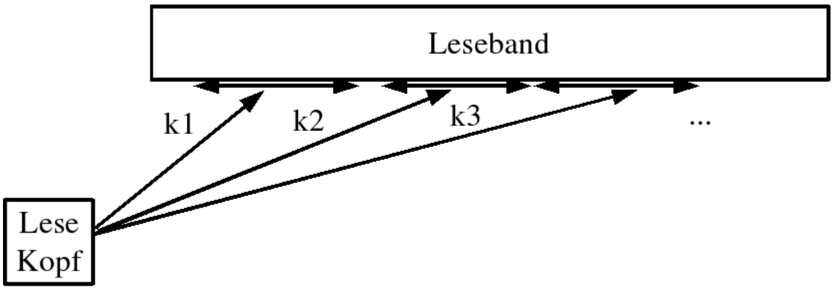
Bemerkung
$NL \subseteq NP \subseteq CS = NSPACE(n)$ und $NL \subsetneq CS$

Beweis
Sei $w = a_1, a_2, \dots a_n \in V^*$
für $1 \leq i \leq j \leq n$ sei $w[i,j] = a_i a_{i+1} \dots a_{j-1}$
für $i < 1$ oder $j > n{+}1$: Aufruf $w(i,j)$ durch return FEHLGESCHLAGEN entspricht Laufzeitfehler (weiter siehe Kopie)
Variable:
$l_1, l_2, r_1, r_2 \in \{-k, -k+1, \dots, n, \dots, n+k\}$ wobei k die Größe des größten Dominos in γ ist.
$x_1, x_2, y_1, y_2, z_1, z_2 \in V^{\leq k}$
$x, y \in W(V, \rho)$; $|x|, |y| \leq k$
Platzbedarf:
$4 \cdot log(2k+n) + c \in O(logn)$ (c hängt nur von γ, aber nicht von der Eingabe w ab)

`q.e.d.`

Jede Stickersprache lässt sich von einem k-Kopfautomaten erkennen.
Nach Bsp. 3.4 $\Rightarrow$ es gibt nicht kontextfrei Stickersprache, also insbesondere eine nicht-reguläre.

Einschub
Erläuterung zu k-Kopfautomaten



- 1 Kopf nur zum Lesen
- ein Kopf weiß nicht ob er rechts oder links von einem anderen ist
- mit 4 Köpfen lassen sich Stickersprachen erkennen
- $L(\gamma)$ lassen sich mit 6 Kopfautomaten erkennen
- $\bigcup_{k \in \mathbb{N}}$ L(k-Kopfautomaten)$\subseteq NL$
- je mehr Köpfe desto schwächer der Automat

Definition
Ein Stickersystem $\gamma = (V, \rho, A, D)$ heißt <u>einseitig</u>, wenn für jede Regel $(x, y) \in D$ gilt:

$$x = \binom{\epsilon}{\epsilon} \text{ oder } y = \binom{\epsilon}{\epsilon}$$

vgl. Grammatik:
G = (N,T,S,R) mit
N = {S,S'}, T = {a,b}, R = $\{S \to S'b, S' \to aS, S \to \epsilon\ \}$
$L(G) = \{a^n b^n : n \in N\}$ nicht regulär.

Lemma3.6
Sei $G = (N, T, S, R)$ kontextfreie Grammatik mit $N = N_l \mathbin{\dot\cup} N_r \mathbin{\dot\cup} \{S\}$ und
$R \subseteqq (\{S\} \times N_l T^* N_r) \cup (N_l \times N_l T^*) \cup (N_r \times T^* N_r) \cup (N \times \{\epsilon\})$.
Dann ist L(G) regulär obwohl die Grammatik nicht regulär ist.

Beweis
für $p = (S, X_l w X_r) \in R$ definiere Grammatik
$G_l^p = (N_l, T, X_l R \cap (N_l \times (N_l \cup T)^*))$
und
$G_r^p = (N_r, T, X_r, R \cap (N_r \times (N_r \cup T)^*))$
$\Rightarrow G_l^p$ ist linksregulär und G_r^p ist rechtsregulär
$L(G) \backslash \{\epsilon\} = \cup_{p=(S, X_l w X_r) \in R} L(G_l^p) \cdot w \cdot L(G_r^p)$
Da G_l^p und G_r^p regulär sind, ist auch $L(G) \backslash \epsilon$ regulär.
$\Rightarrow$ L(G) regulär

q.e.d.

Lemma3.7
Sei $\gamma = (V, \rho, A, D)$ ein einseitiges Stickersystem. Dann ist $L(\gamma)$ regulär.

Beweis
Wir konstruieren Grammatik G mit $L(G) = L(\gamma)$, die die Bedingungen von Lemma 3.6 erfüllt.
Idee:
wähle $d \in N$ mit $h(x) \leq d$ für alle $x \in A, (x, y), (y, x) \in D$, dann ist jedes $x \in LM(\gamma)$ durch

eine Ableitung $A \ni x_0 \rightarrow_\gamma x_1 \rightarrow_\gamma x_2 \rightarrow_\gamma \cdots \rightarrow_\gamma x_n = x$ herleitbar mit $h(x_i) \leq d$ für alle i. Die Überlänge von x_i werden die Nichtterminale:

$$N = \left(\left\{ c \in \binom{V^*}{\epsilon} \cup \binom{\epsilon}{V^*} : |x| \leq d \right\} \times \{l, r\} \right) \cup \{S\}$$

$$T = \rho$$

P enthält folgende Regeln:

1. $S \rightarrow (x, l)y(z, r)$ ist Regel, falls
   $x, z \in \binom{V^*}{\epsilon} \cup \binom{\epsilon}{V^*}$; $|x|, |z| \leq d$; $y \in \rho^+$; $x, y, z \in A$
   $\Rightarrow (x, l)$ und (z, r) sind Überhänge
2. $(y, l) \rightarrow (y', l)z$ ist Regel falls
   $y, y' \in \binom{V^*}{\epsilon} \cup \binom{\epsilon}{V^*}$; $|y|, |y'| \leq d$; $z \in \rho^*$ und es gibt
   $(x, \epsilon) \in D$, und $\begin{bmatrix} \text{a} \\ \text{b} \end{bmatrix} \in \rho$ mit $x \cdot y \cdot \begin{bmatrix} \text{a} \\ \text{b} \end{bmatrix} = y'z \begin{bmatrix} \text{a} \\ \text{b} \end{bmatrix}$
3. $(y, r) \rightarrow x(y', r)$ ist Regel, falls
   $y, y' \in \binom{V^*}{\epsilon} \cup \binom{\epsilon}{V^*}$; $|y|, |y'| \leq d$; $x \in \rho^*$ und es gibt
   (ϵ, z) und $\begin{bmatrix} \text{a} \\ \text{b} \end{bmatrix} \in \rho$ mit $\begin{bmatrix} \text{a} \\ \text{b} \end{bmatrix} y \cdot z = \begin{bmatrix} \text{a} \\ \text{b} \end{bmatrix} x \cdot y'$
4. $(\epsilon, l) \rightarrow \epsilon$
   $(\epsilon, r) \rightarrow \epsilon$

Die Grammatik $G = (N, T, S, P)$ erfüllt die Bedingungen aus Lemma 3.6, also ist L(G) regulär.

`Wir zeigen zunächst:` $L(G) = L(M)(\gamma)$
Sei $x \in L(G) \rightarrow$ es gibt Ableitung in G der Gestalt

$$S \rightarrow_G (x_1, l)y(z_1, r) \rightarrow_G (x_2, l)y_2(z_2, r) \cdots \rightarrow_G (x_n, l)y_n(z_n, r) \rightarrow_G y_n(z_n, r) = x$$
$$\Rightarrow x_1 y_1 z_1 \in A$$
$$x_i y_i z_i \rightarrow_\gamma x_{i+1}, y_{i+1}, z_{i+1} \text{ für } 1 \leq i \leq n-1$$
$$\Rightarrow x_n y_n z_n = y_n \in LM(\gamma), \text{ also } L(G) \leq LM(\gamma)$$

Es sei $x \in LM(\gamma)$
$\Rightarrow$ es gibt Ableitung:

$$A \ni x_1 \rightarrow_\gamma x_2 \rightarrow_\gamma x_3 \cdots \rightarrow_\gamma x_n = x \text{ mit } h(x_i) \leq d$$
$$\text{Sei } x_i^1, x_i^3 \in \binom{V^*}{\epsilon} \cup \binom{\epsilon}{v^*}, x_i^2 \in \rho^+ \text{ mit } x_i = x_i^1 x_i^2 x_i^3$$

Regeln aus P genau so, dass

$S \to G(x_1^1,l)x_1^2(x_1^3,r) \to_G (x_2^1,l)x_2^2(x_2^3,r) \cdots \to_G (x_n^1,l)x_n^2(x_n^3,r)$
$x = x_n = x_n^1 x_n^2 x_n^3 \in \rho^+$
$\Rightarrow x_n^1, x_n^3 = \binom{\epsilon}{\epsilon}$
$\Rightarrow (x_n^1,l)x_n^2(x_n^3,r) \to_G^* x_n^2 = x_n = x$
$\Rightarrow x \in L(G)$, also $LM(\gamma) \leq L(G)$
$\Rightarrow LM(\gamma)$ regulär.
Sei $\eta : \rho^* \to V^*$Homomorphismus mit $\eta\left(\begin{bmatrix} \mathrm{a} \\ \mathrm{b} \end{bmatrix}\right) = a$ für $\begin{bmatrix} \mathrm{a} \\ \mathrm{b} \end{bmatrix} \in \rho$
$\Rightarrow \eta(LM(\gamma))$ ist regulär und gleich $L(\gamma)$

q.e.d.

Lemma3.8
Sei $G = (N,T,S,P)$ rechtsreguläre Grammatik. Dann existiert ein einseitiges Stickersystem γ mit $L(G) = L(\gamma)$.

Beweis
o.E.
$N = \{X_1, X_2, \ldots, X_k\}$
$(X_i w X_j)$ oder (X_i, w) Regel $\to |w| \leq 1$
Idee:
ist $a_1, \ldots, a_n X_i$ ableitbare Satzform, so kodiere diese durch Domino

a_1	a_2	$\ldots$	a_{n-i}	$\ldots$	a_n
a_1	a_2	$\ldots$	a_{m-i}		

Formal:
$V = T$
$\rho = \left\{\binom{a}{a}, a \in V\right\}$
$A = \left\{\begin{bmatrix} \mathrm{x} \\ \mathrm{x} \end{bmatrix}\binom{u}{\epsilon} : S \to_G^* xuX_{|u|}, |xu| = k+1\right\} \cup \left\{\begin{bmatrix} \mathrm{x} \\ \mathrm{x} \end{bmatrix} : x \in L(G), |x| < k+1\right\}$
$D = \left\{\left(\binom{\epsilon}{\epsilon}, \binom{\epsilon}{v}\begin{bmatrix} \mathrm{x} \\ \mathrm{x} \end{bmatrix}\binom{u}{\epsilon}\right) : |xu| = k+1, X_{|v|} \to_G^* xuX_{|u|}\right\} \cup \left\{\left(\binom{\epsilon}{\epsilon}, \binom{\epsilon}{v}\begin{bmatrix} \mathrm{x} \\ \mathrm{x} \end{bmatrix}\right) : |x| \leq k, X_{|v|} \to_G^* x\right\}$
Dann ist $\gamma = (V, \rho, A, D)$ einseitiges Stickersystem.

Nun zu zeigen:

$$L(G) = L(\gamma)$$
$$'L(\gamma) \leq L(G)' \text{ ohne Beweis}$$
$$'L(G) \leq L(\gamma)'$$

Sei $w \in L(G)$
Es gibt folgende Fälle:

1. $|w| \leq k \Rightarrow \begin{bmatrix} \mathrm{w} \\ \mathrm{w} \end{bmatrix} \in A \cap \rho^+ \Rightarrow w \in L(\gamma)$

2. $|w| > k$
$\Rightarrow$ es gibt $w_i \in V^+$ mit $|w_i| = k+1$ für $1 \le i \le n$ und $w = w_1 w_2 w_3 \dots w_n w_{n+1}, w_{|n+1|} \le k$
und es gibt $j_i = k$ mit $S \to_G^* w_1 X_{j_1} \to_G^* w_1 w_2 X_{j_2} \dots \to_G^* w_1 w_2 \dots w_n X_{j_n} \to_G^* w_1 w_2 \dots w_n w_{n+1} = w$
wegen $|w_i| = k+1$ existiert $u_i, x_i \in V^+ : w_i = x_i u_i, |u_i| = j_i$ für $1 \le i \le n$
$\Rightarrow$

   (a) $\begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \binom{u_1}{\epsilon} \in A$

   (b) für $2 \le i \le n$
   $X_{j_{i-1}} \to w_i X_{j_i, j_{i-1}} = |u_{i-1}|, j_i = |u_i|$
   $\Rightarrow \left(\binom{\epsilon}{\epsilon}, \binom{\epsilon}{u_{i-1}} \begin{bmatrix} x_i \\ x_i \end{bmatrix} \binom{u_i}{\epsilon} \right) \in D$

   (c) $X_{j_n} \to_G^* w_{n+1}, |w_{n+1}| \le k$
   $\Rightarrow \left(\binom{\epsilon}{\epsilon}, \binom{\epsilon}{u_n} \begin{bmatrix} w_{n+1} \\ w_{n+1} \end{bmatrix} \right) \in D$
   $\Rightarrow \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \binom{n_1}{\epsilon} \to_\gamma \begin{bmatrix} x_1 u_1 x_2 \\ x_1 u_1 x_2 \end{bmatrix} \binom{u_2}{\epsilon}$
   (Regel der Form 2) $\to_\gamma^* \begin{bmatrix} x_1 u_1 \dots u_{n-1} x_n \\ x_1 u_1 \dots u_{n-1} x_n \end{bmatrix} \binom{u_n}{\epsilon}$
   (Regel der Form 3) $\to_\gamma^* \begin{bmatrix} x_1 u_1 \dots u_n w_{n+1} \\ x_1 u_1 \dots u_n w_{n+1} \end{bmatrix} = \begin{bmatrix} \text{w} \\ \text{w} \end{bmatrix}$
   $\Rightarrow w \in L(\gamma)$

q.e.d.

Satz 3.9
Es gelte die folgende Beziehung

$$\begin{array}{ccc}
 & CS = NSPACE(id) & \\
 & \rotatebox{90}{$\supsetneq$} & \\
\rotatebox{-45}{$\subsetneq$} & \text{SL} := & \rotatebox{45}{$\supsetneq$} \\
 & \{L(\gamma) : \gamma \text{ SSys}\} & \\
\text{SSL} := & \not\supseteq & \text{OSL} = \text{REG} =: \\
\{L(\gamma) : \gamma \text{ einf SSys}\} & \not\subseteq & \{L(\gamma) : \gamma \text{ eins SSys}\} \\
\rotatebox{45}{$\supsetneq$} & & \rotatebox{-45}{$\subsetneq$} \\
 & \text{SOSL} := & \\
 & \{L(\gamma) : \gamma \text{ einf \& eins SSys}\} &
\end{array}$$

(Abkürzungen: SSys = Stickersystem, einf = einfaches, eins = einseitiges)

Beweis
$SL \subsetneq CS$: Lemma 3.5
$OSL = REG$: Lemma 3.7 & 3.8
$SOSL \neq SSL$: übung $ab^*a \in SSL \backslash SOSL$
$SOSL \neq REG$: $ab^*a \in REG \backslash SOSL$
$SSL \not\subseteq REG$: Bsp 3.4 (SSL enthält eine Sprache, die nicht kontextfrei ist)
$\Rightarrow SL \neq OSL$ $REG \not\subseteq SSL$: $a^* \cup b \in REG \backslash SSL \Rightarrow SSL \subsetneq SL$
(Weigel & Keilwagen '04)

q.e.d.

3.3 Turing- mächtige Erweiterungen

nach Lemma 3.5 existiert insbesondere eine rekursiv aufzählbare Menge L, die sich von keinem Stickersystem erzeugen lässt.

Frage:
existiert vielleicht ein 'Übersetzungs-' oder 'Anzeigemachanismus' g so, dass für jede rekursiv aufzählbare (im weiteren Verlauf abgekürzt durch r.e.) Sprache L ein Stickersystem γ existiert mit $L = g(L(\gamma))$.
`hier:` generalized sequential machines (gsm)

Definition3.10
Ein Tupel $g = (Q, V_1, V_2, \iota, F, \delta)$ ist eine det. gsm, falls:

- Q eine endliche Menge von Zuständen
- V_1, V_2 Ein- & Ausgabealphabete
- $\iota \in Q$ der Initialzustand
- F $\subseteq$ Q eine Menge von Finalzuständen und
- $\delta : Q \times V_1 \to (Q \times V_2^*)$ eine Überführungsfunktion ist.

Für die det. gsm g sei $R(g) \subseteq V_1^* \times V_2^*$ definiert durch:

$$\begin{aligned}
&(a_1 a_2 \dots a_n, w) \in R(g) \\
&\text{ mit } a_i \in V_1 \ \& \ w \in V_2^* \Leftrightarrow \exists q_i \in Q, v_i \in V_2^* : \\
&q_0 = \iota, \delta(q_i, a_{i+1}) = (q_{i+1} v_{i+1}) \text{ für } i = 0, \dots, n-1 \\
&q_n \in F \text{ und } w = v_1 v_2 \dots v_n
\end{aligned}$$

Bemerkung
Sind $(v, w), (v, w') \in R(g)$, so gilt $w = w'$
$\Rightarrow R(g)$ ist partielle Funktion von V_1^* nach V_2^*
$\Rightarrow$ Schreibweise $g(v) = w$ für $(v, w) \in R(g)$

$$V_1^* \underrightarrow{g} V_2^* \underrightarrow{h} V_3^*$$
$$V_1^* \underrightarrow{h \circ g} V_3^*$$

Lemma3.11
Seien $g_i = (Q_i, V_i, V_{i+1}, \iota_i, \delta_i)$ für i = 1,2 det. gsms. Dann existiert eine det. gsm g mit $R(g) = R(g_1) \circ R(g_2)$
Rightarrow g ist Verkettung von g_1 und g_2

Bemerkung
$h \circ g(u) := h(g(u))$; $u \in V_1^*$
$R(g_1) \circ R(g_2) = \{(u, v) \in V_1^* \times v_3^* : \exists v \in V_2^* : (u, v) \in R(g_1), (v, w) \in R(g_2)\}$

Beweis
Beweisskizze:

$$\xrightarrow{a_1 a_2 a_3} \boxed{g_1} \rightarrow \begin{array}{|c|}\hline v_1 \\ \hline v_2 \\ \hline v_3 \\ \hline \end{array} \rightarrow \boxed{g_2} \rightarrow g_2(v_1 v_2 v_3)$$

$$\begin{aligned}
&Q = Q_1 \times Q_2 \\
&\delta((q_1, q_2), a) = ((r_1, r_2), w) \text{ mit} \\
&\delta_1(q_1, a) = (r_1, v)\text{und} \\
&\delta_2^*(q_2, v) = (r_2, w) \text{ für ein } v \in V_2^* \\
&\iota = (\iota_1, \iota_2) \\
&F = F_1 \times F_2
\end{aligned}$$

Setze $g = (Q, V_1, V_3, \iota, F, \delta)$ Dann gilt:

$$R(g) = R(g_1) \circ R_(g_2)$$

o.B.
q.e.d.

Bemerkung

$$\begin{aligned}
&\delta_2^*(q_2, v) = (r_2, w) \\
&\delta_2^*(q_2, v) = (r, \epsilon) \\
&\delta_2^*(q_2, au) = (r'', vw) \\
&\text{falls } \delta_2(r, a) = (r', v) \\
&\text{und } \delta_2^*(r', u) = (r'', w)
\end{aligned}$$

weiteres Vorgehen:

1. Konstruktion eines <u>einfachen Stickersystems</u> & einer deterministischen gsm g_1 mit $TS_v = g_1(L(\gamma))$
2. für L r.e. existiert det. gsm g_2 mit $L = g_2(TS_v) = \underbrace{g_2(g_1}_{g}(L(\gamma))) = g(L(\gamma))$

Definition 3.12 Twin-Shuffel-Sprache
V Alphabet

- $\overline{V} = \{\overline{a} : a \in V\}$ wir nehmen an, dass $V \cap \overline{V} = \oslash$
- $u, v \in V^*$
  $u \sqcup\!\sqcup v = \{u_1 v_1 u_2 v_2 \dots u_n v_n : u_1 u_2 \dots u_n = u, v_1 v_2 \dots v_n = v$
  ist der <u>Schuffel</u> von u & v
- $a_1, a_2, \dots a_n \in V : \overline{a_1 a_2 \dots a_n} = \overline{a_1}, \overline{a_2}, \overline{a_3}, \dots, \overline{a_n}$

$$TS_V = \bigcup_{w \in V^*} w \sqcup\!\sqcup \overline{w}$$

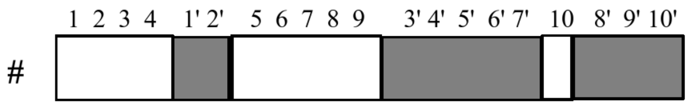
ist die <u>Twin-Schuffel-Sprache</u> über V

Bemerkung

$|v| = 1 : TS_v$ kontextfrei $\qquad$ $|v| > 1 : TS_v$ kontextsensitiv, aber nicht kontextfrei

Beispiel
aab $\sqcup\!\sqcup$ ca $= \{$aabca caaab, caaba, aacba, aacab, $\ldots\}$

Lemma 3.13
Sei X endliche Menge. Dann existiert ein einfaches Stickersystem $\gamma = (V, \rho, A, D)$ und eine det. gsm g mit $TS_X = g(L(\gamma))$



Das Bild zeigt $v \sqcup\!\sqcup \overline{v}$. Die weißen Kästchen sind $v \in X^*$ und die grauen $\overline{v} \in \overline{X^*}$. Es gilt außerdem $|$grau$| = |$weiß$|$. Die gesammte Grafik ist $\in TS_x$

Beweis
Idee:

1. wir erzeugen mit γ die Sprache L aller Wörter $v^{rev}\#w$ mit $v \in X^*, w \in v \sqcup\!\sqcup \overline{v}$
2. $g(v^{rev}\#w) = w$

zu 1.

$$V = X \cup \overline{X} \,\dot\cup\, \{\#\}$$
$$\delta = \left\{ \binom{a}{a} : a \in V \right\}$$
$$A = \left\{ \left[\begin{matrix} \# \\ \# \end{matrix} \right] \right\}$$
$$D = \{R_a, R_{\overline{a}} : a \in X\} \cup \left\{ \left(\binom{\epsilon}{\epsilon}, \binom{\epsilon}{a} \right) : a \in X \cup \overline{X} \right\}$$
$$R_a = \left(\binom{a}{\epsilon}, \binom{a}{\epsilon} \right), R_{\overline{a}} = \left(\binom{\epsilon}{a}, \binom{\overline{a}}{\epsilon} \right)$$

$\gamma = (V, \delta, A, D)$ ist einfaches, aber nicht einseitiges Stickersystem.

`zu zeigen:` $L \subseteq L(\gamma)$

Sei $w^{rev}\#v \in L$, d.h. $w \in X^*$, $v \in w \sqcup\!\sqcup \overline{w}$
definiere Homomorphismus

$$f : \qquad (X \cup \overline{X}) \to X^*$$
$$a \to a$$
$$\overline{a} \to \epsilon (\text{für } a \in X)$$
$$g : (X \cup \overline{X})^* \to X^*$$
$$a \to \epsilon$$
$$\overline{a} \to a$$

für $0 \leq i \leq |v|$ sei v_i der Präfix von v der Länge i

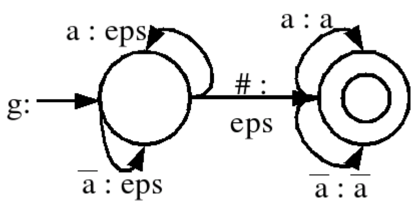
$$W_\delta(V) \ni \begin{array}{c} \boxed{f(v_i)^{rev}\#v_i} \\ \boxed{g(v_i)^{rev}\#} \end{array} \rightarrow_\gamma \begin{array}{c} \boxed{f(v_{i+1})^{rev}\#v_{i+1}} \\ \boxed{g(v_{i+1})^{rev}\#} \end{array}$$

insbesondere also $\begin{bmatrix} \# \\ \# \end{bmatrix} \rightarrow_\gamma^* \begin{bmatrix} f(v)^{rev}\# \\ g(v)^{rev}\# \end{bmatrix} \binom{v}{\epsilon} \rightarrow_\gamma^* \begin{bmatrix} w^{rev}\#v \\ w^{rev}\#v \end{bmatrix}$
$\Rightarrow w^{rev}\#v \in L(\gamma)$, also $L \subseteq L(\gamma)$.

`zu zeigen:` $L(\gamma) \subseteq L$

Bemerkung
alle Elemente von $LM(\gamma)$ haben Form $\begin{bmatrix} w^{rev}\#v \\ w^{rev}\#v \end{bmatrix}$ mit $w \in X^*, v \in (X \cup \overline{X})^*$.

Sei also $w^{rev}\#v \in L(\gamma)$, also
$\underbrace{\begin{bmatrix} \# \\ \# \end{bmatrix} \rightarrow_\gamma^* \begin{bmatrix} w^{rev}\# \\ w^{rev}\# \end{bmatrix} \begin{pmatrix} v \\ \epsilon \end{pmatrix}}_{\text{nur Regel der Form} R_a \text{ und } R_{\overline{a}}} \rightarrow_\gamma^* \begin{bmatrix} w^{rev}\#v \\ w^{rev}\#v \end{bmatrix}$
Reihenfolge dieser Regelanwendungen: v
Reihenfolge der Anwendung von $\{R_a : a \in X\}$: w
$\{R_{\overline{a}} : a \in X\}$: w
$\Rightarrow v \in w \sqcup\!\sqcup \overline{w}$, also $w^{rev}\#v \in L$
$\Rightarrow L(\gamma) \subseteq L$
zu 2.



`q.e.d.`

Lemma 3.14
Sei L eine r.e. Sprache. Dann existiert det. gsm g und eine endliche Menge X mit $L = g(TS_x)$.

Beweis
Sei $G = (N, T, S, P)$ Grammatik mit L = L(G).
Idee:

1. erfinde Kodierung von G – Ableitung durch Wörter über $X \cup \overline{X} \rightarrow$ Menge $C \subseteq (X \cup \overline{X})^*$
2. zeige $C = L_1 \cap TS_x$ für eine reg. Menge L_1
3. für $w \in C$ kann abgeleitetes Wort durch det. gsm berechnet werden.

zum Beweis:

$$X := N \cup T \,\dot\cup\, \{€\}$$

1. Kodierung:
   Sei $S = w_0 \to_G w_1 \to_G w_2 \cdots \to_G w_1 \in T^*$ G – Ableitung
   $\Rightarrow$ es gibt $u_i, v_i \in (N \cup T)^*, (l_i, r_i) \in P$ mit
   $w_i = u_i l_i v_i$ & $w_{i+1} = u_i r_i v_i$ für $0 \leq i < n$
   kodiere Ableitung wie folgt:

$$\begin{array}{l} \text{€} w_n \text{€} \overbrace{u_{n-1} l_{n-1} v_{n-1}}^{w_{n-1}} \text{€} \overbrace{u_{n-2} l_{n-2} v_{n-2}}^{w_{n-2}} \ldots \text{€} \overbrace{u_0 l_0 v_0}^{w_0} \\ \quad \overline{\text{€}} \underbrace{\overline{u_{n-1} r_{n-1} v_{n-1}}}_{\overline{w_n}} \overline{\text{€}} \underbrace{\overline{u_{n-2} r_{n-2} v_{n-2}}}_{\overline{w_{n-1}}} \ldots \overline{\text{€}} \underbrace{\overline{u_0 r_0 v_0}}_{\overline{w_1}} \overline{\text{€} w_0} \end{array}$$

   um Wort über $X \cup \overline{X}$ zu erhalten:

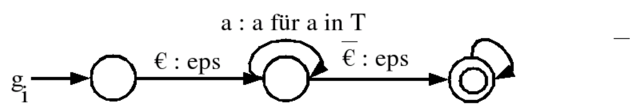
   - mische u_i und $\overline{u_i}$ 'fair' (analog für v_i & $\overline{v_i}$)
   - ersetze $\frac{\text{€}}{\text{€}}$ durch $\overline{\text{€}}\text{€}$

   $f : X^* \to (X \cup \overline{X})^*$ mit $f(a) = a\overline{a}$ für $a \in X$
   $C = \{\text{€} w_n \overline{\text{€}}\text{€} f(u_{n-1}) l_{n-1} \overline{r_{n-1}} f(v_{n-1}) \overline{\text{€}}\text{€} f(u_{n-2}) l_{n-2} \overline{r_{n-2}} f(v_n - 2) \overline{\text{€}}\text{€} \ldots \overline{\text{€}}\text{€} f(n_0) l_0 \overline{r_0} f(v_0) \overline{\text{€} w_0}$
   $S = W_0 \to w_1 \cdots \to w_n \in T^*$ Ableitung in G $\} \subseteq TS_X$

   - nicht regulär
   - nicht kontextfrei
   - kontextsensitiv

2. $C = L_1 \cap TS_x$ für eine reguläre Menge L_1.
   $Y := \{a\overline{a} = f(a) : a \in N \cup T\}$
   $R := \{l\overline{r} : (l, r) \in P\}$
   $L_1 = \text{€} T^* \overline{\text{€}} (\text{€} Y^* R Y^* \overline{\text{€}})^* \overline{S}$
   L_1ist regulär
   $C \subseteq L_1 \cap TS_x$ leicht zu sehen
   `zu zeigen:` $L_1 \cap TS_x \subseteq C$ ohne Beweis

3. Konstruktion der deterministischen gsm g mit $L = g(TS_x)$
   leicht für $L = g_1(TS_x \cap L_1)$:



   L_1 regulär $\Rightarrow$ es gibt einen deterministischen endlichen Automaten $A = (Q, \iota, \delta, F)$ mit $L_1 = L(A)$
   $g_2 = (Q, X \cup \overline{X}, X \cup \overline{X}, \iota, \delta', F)$ mit $\delta'(p, a) = (\delta(p, a), a)$ ist deterministische gsm mit $domg_2 = L_1$ und $g_2(u) = u$ für $u \in L_1$ (d.h. $R(g_2) = \{(u, u) : u \in L_1\}$)
   $\Rightarrow g_1 \circ g_2(TS_x) = g_1(TS_x \cup L_1) = L$
   nach Lemma 3.11 existiert deterministische gsm g mit $g = g_1 \circ g_2$, also $L = g(TS_x)$

`q.e.d`

Satz 3.15
Sei L r.e. Sprache. Dann existiert ein einfaches Stickersystem γ und eine deterministische gsm g mit $L = g(L(\gamma))$.

Beweis
Nach:

- Lemma 3.14: $L = g_1(TS_x)$ für eine endliche Menge X & eine deterministische gsm g_1
- Lemma 3.13: $TS_x = g_2(L(\gamma))$ für ein einfaches Stickersystem γ und eine deterministische gsm g_2

$\Rightarrow$ es gibt deterministische gsm g mit:

$$g(L(\gamma)) = g_1 \circ g_2(L(\gamma)) = g_1(TS_x) = L$$

`q.e.d`

nächstes Ziel: jede Stickersprache ist gsm-Bild einer Stickersprache über natürlicher Komplementarität.

Lemma 3.16
V_1, V_2 Alphabete, $k \in \mathbb{N}$, $L_a \subseteq V_1^k$ für $a \in V_2$ und L_a paarweise disjunkt. Dann existiert eine deterministische gsm g mit:
$g(u) = a_1 a_2 \ldots a_n$ für $u \in L_{a_1} L_{a_2} \ldots L_{a_n}$ und $g(u)$ undefiniert falls u nicht von dieser Gestalt.

Beweis
Idee:

- merke dir die letzten Buchstaben
- nach k Buchstaben weiß man, was ausgegeben werden muss.

$$\begin{aligned}
&L := \bigcup_{a \in V_2} L_a \subseteq V_1^k \text{ endl.} \\
&Q := \{u \in V_1^* : |u| \leq k-1\} \cup \{\perp\} \\
&\iota := \epsilon \\
&\delta(u,a) = \begin{cases} (ua, \epsilon), & \text{falls } |ua| \leq k-1 \\ (\epsilon, b)\ , & \text{falls } ua \in L_b \\ (\perp, \epsilon) & \text{falls } |ua| = k, ua \notin L \end{cases} \\
&\delta(\perp, a) = (\perp, \epsilon) \\
&F = \{\epsilon\}
\end{aligned}$$

$g = (Q, V_1, V_2, \delta, \iota, F)$ deterministische gsm da $\{L_a : a \in V_2\}$ paarweise disjunkt.

`q.e.d`

Ein Stickersystem $\gamma = (V, \delta A', D)$ heißt <u>natürlich</u>, falls $V = \{A, G, C, T\}$ und $\delta = \left\{\binom{A}{T}, \binom{T}{A}\binom{G}{C}\binom{C}{G}\right\}$.

Satz 3.17
Sei $\gamma_2 = (V_2, \delta_2, A_2', D_2)$ ein Stickersystem. Dann existiert ein natürliches Stickersystem $\gamma_1 = (V_1, \gamma_1, A_1', D_1)$ und eine deterministische gsm g mit $L(\gamma_2) = g(L(\gamma_1))$. Ist γ_2 einfach\ einseitig, so kann γ_1 einfach\ einseitig gewählt werden.

Beweis
Sei $\delta_2 = \{(a_i, b_i) 1 \leq i \leq k\}$ ($a_1 = a_2$ oder $a_1 = b_5$mgl.) für $a \in V_2$ sei $L_a = \{G^i C^{k-i} : a_i = a\} \cup \{C^j G^{k-j}\} \subseteq V_1^k$

Sei g deterministische gsm aus Lemma 3.16, d.h.:

$$g : \left(\bigcup_{a \in V_2} L_a \right)^* \to V_2^*$$

Sei $\binom{u_1}{u_2} \left[\begin{array}{c} v_1 \\ v_2 \end{array} \right] \binom{w_1}{w_2} \in W_{\rho_1}(V_1)$ mit $u_i, v_i, w_i \in (\bigcup_{a \in V_2} L_a)^*$
$\Rightarrow \binom{g(u_1)}{g(u_2)} \left[\begin{array}{c} g(v_1) \\ g(v_1) \end{array} \right] \binom{g(w_1)}{g(w_2)} \in W_{\rho_2}(V_2)$
wir schreiben $g^* \left(\binom{u_1}{u_2} \left[\begin{array}{c} v_1 \\ v_2 \end{array} \right] \binom{w_1}{w_2} \right)$ für dieses Molekül.
$V_1 = \{A, G, C, T\}$
$\delta_1 = \{(A, T), (T, A), (G, C), (C, G)\}$
$A_1 = \{x \in W_{\delta_1}(V_1) : g^*(x) \in A_2\}$
$D_2 = \{(x, y) \in W_{\delta_1}(V_1)^2 : (g^*(x), g^*(y)) \in D_2\}$
$\gamma_1 := (V_1, \delta_1, A_1, D_1)$ natürliches Stickersystem.

`zu zeigen:` $g(L(\gamma_1)) = L(\gamma_2)$
Seien $x, y \in W_{\delta_1}(V_1)$ mit $x \to_{\gamma_1} y$ und $g^*(x), g^*(y)$ definiert, da komplementäre Paare aus $\bigcup L_a$ auf komplementäre Paare aus V_2 abgebildet werden, gilt $g^*(x) \to_{\gamma_2} g^*(y)$.

damit:
$y \in LM(\gamma_1) \Rightarrow$ es gibt $x \in A_1$ mit $x \to_{\gamma_1}^* y$
$\Rightarrow$ es gibt $x \in A_1$ mit $g^*(x) \to_{\gamma_2}^* g^*(y) \& g^*(x) \in A_2$
$\Rightarrow g^*(y) \in LM(\gamma_2)$

also:
$w \in L(\gamma_1) \Rightarrow g(w) \in L(\gamma_2)$, d.h. $g(L(\gamma_1)) \subseteq L(\gamma_2)$

umgekehrt sei $w' \in L(\gamma_2)$. Dann existiert $y' = \left[\begin{array}{c} w' \\ w'' \end{array} \right] \in LM(\gamma_2)$
$\Rightarrow$ es gibt $1 \leq i_e \leq k$ für $1 \leq l \leq |w_1|$ mit

$$\begin{aligned} y' &= \left[\begin{array}{c} a_{i_1} \\ b_{i_1} \end{array} \right] \left[\begin{array}{c} a_{i_2} \\ b_{i_2} \end{array} \right] \cdots \left[\begin{array}{c} a_{i_{|w'|}} \\ b_{i_{|w'|}} \end{array} \right] \\ y &:= \left[\begin{array}{c} G^{i_1} C^{k-i_1} \\ C^{i_1} G^{k-i_1} \end{array} \right] \left[\begin{array}{c} G^{i_2} C^{k-i_2} \\ C^{i_2} G^{k-i_2} \end{array} \right] \cdots \left[\begin{array}{c} G^{i_n} C^{k-i_n} \\ C^{i_n} G^{k-i_n} \end{array} \right] \text{ mit } n = |w'| \end{aligned}$$

dann gilt $g^*(y) = y'$, da y' in γ_2 ableitbar aus einem Axiom, ist y in γ_1 aus einem Axiom ableitbar, also $L(\gamma_2) \subseteq g(L(\gamma_1))$
Beobachtung: ist γ_2 einfach bzw. einseitig, so ist unser γ_1 ebenfalls einfach bzw. einseitig.

`q.e.d`

Beobachtung:
γ einseitiges Stickersystem

$\Rightarrow$ es gibt reguläre Grammatik G mit $L(\gamma) = L(G)$ und diese kann berechnet werden (vgl. Beweis von Lemma 3.6 und 3.7)

$\Rightarrow$ es gibt endlichen Automaten A mit $L(A) = L(G)$ und dieser kann berechnet werden.

$L(A) \neq \oslash$ gdw, ein Finalzustand in A von einem Initialzustand aus erreicht werden kann und dies kann effektiv entschieden werden. Also ist die Frage $L(\gamma) = \oslash$? für einseitiges Stickersysteme entscheidbar (in P lösbar).

Frage: Was ist im allgemeinen Fall?

Posts Korrespondenzproblem (PCP):

`Eingabe:` endliche Menge von Paaren (u_i, v_i) von Wörtern
`Frage:` existieren Indizes $i_1, i_2, \ldots, i_n$ für eine Zahl $n > 0$ mit $u_{i_1} u_{i_2} \ldots u_{i_n} = v_{i_1} v_{i_2} \ldots v_{i_n}$

Satz 3.18
PCP ist unentscheidbar.

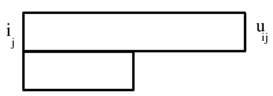
Satz 3.19
Das Leerheitsproblem für einfache Stickersysteme ist unentscheidbar.

Beweis
wir reduzieren PCP auf Leerheitsproblem
Seien $(u_i, v_i) 1 \leq i \leq n$ Paare von Wörtern über X
$V = X \,\dot{\cup}\, \{\#1, 2, \ldots, n\}$
$\delta = \left\{ \binom{a}{a} : a \in V \right\}$ `Idee:`
wir konstruieren einfaches Stickersystem γ mit $L(\gamma) = \{i_e, i_{e-1} \ldots i_1 \# u_{i_1} u_{i_2} \ldots u_{i_e} : 1 \leq i_j \leq n_j; u_{i_1} u_{i_2} \ldots u_{i_e} = v_{i_1} v_{i_2} \ldots v_{i_e}\} =: L$

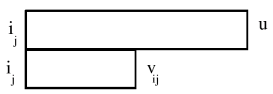
$$A = \left\{ \binom{i}{\epsilon} \begin{bmatrix} \# \\ \# \end{bmatrix} \binom{u_i}{\epsilon} : 1 \leq i \leq n \right\}$$
$$D = \left\{ \left(\binom{i}{\epsilon}, \binom{u_i}{\epsilon} \right), \left(\binom{\epsilon}{i}, \binom{\epsilon}{v_i} \right) : 1 \leq i \leq n \right\}$$

Sei $\gamma = (V, \delta, A, D)$.
`zu zeigen:` $L \subseteq L(\gamma)$
Sei $(i_1, i_2, \ldots, i_e)$ Lösung der PCP-Instanz
$u_{i_1} u_{i_2} \ldots u_{i_n} = v_{i_1} v_{i_2} \ldots v_{i_l} => \binom{u_{i_1} \ldots u_{i_l}}{v_{i_1} \ldots v_{i_n}} \in W_\delta(V)$
für $i \leq j \leq l$
$\boxed{i_{j-1} i_{j-2} \ldots i_1 \# u_{i_1} u_{i_2} \ldots u_{i_{j-1}}}$
$\boxed{i_{j-1} i_{j-2} \ldots i_1 \# V_{i_1} v_{i_2} \ldots v_{i_{j-1}}}$

$\rightarrow_\gamma$



$\rightarrow_\gamma$



$\Rightarrow i_e i_{e-1} \ldots i_1 \# u_{i_1} u_{i_e} \in L(\gamma)$, also $L \subseteq L(\gamma)$. $L(\gamma) \subseteq L$
Sei $x \in LM(\gamma)$. Dann gibt es $a, b \in \{1, \ldots, n\}^*, u, v \in X^*, x = \begin{bmatrix} a^{rev} \# u \\ b^{rev} \# v \end{bmatrix}$ wegen der Gestalt von ??? gilt $a = b$ oder $u = v$.
betrachte eine beliebige Herleitung von x aus $\begin{bmatrix} \# \\ \# \end{bmatrix}$

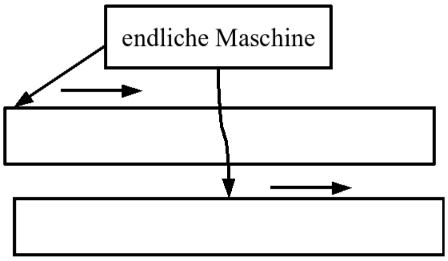
Reihenfolge der Anwendungen von $\left(\binom{i}{\epsilon}, \binom{u_i}{\epsilon}\right)$ ist a
Reihenfolge der Anwendungen von $\left(\binom{\epsilon}{i}, \binom{\epsilon}{v_i}\right)$ ist b
`also:`
$u = u_{a_1} u_{a_2} \dots u_{a_n}$ mit $a = a_1 a_2 \dots a_n$ und $v = v_{a_1} v_{a_2} \dots v_{a_n}$ wegen $a = b$.
Da $u = v$ gilt $a^{rev} \# v \in L \Rightarrow L(\gamma) \subseteq L$
$\Rightarrow L(\gamma) = L$
also $L(\gamma) \neq \oslash$ gdw. PCP Instanz eine Lösung hat.
$\Rightarrow L(\gamma) \neq \oslash$? ist unentscheidbar für einfache Stickersysteme.

Korollar 3.19
Das Leerheitsproblem für nat. und einfache Stickersysteme ist unentscheidbar.

Beweis
Wir reduzieren das Leerheitsproblem einfacher Stickersysteme auf das obige Problem.
Sei γ einfaches Stickersystem
Satz 3.17 $\Rightarrow \exists$ nat. Stickersystem $\gamma^{'}$ (einfach)
$\exists$ deterministische gsm g mit $L(\gamma) = g(L(\gamma^{'}))$ und im Beweis von Satz 3.17 werden
??? γ und g effektiv konstruiert $g(v)$ ist definiert für alle $v \in L(\gamma^{'})$
also $L(\gamma) = \oslash$ gdw. $L(\gamma) = \oslash$

4 Watson – Crick – Automaten

Idee: Automat mit 2 Leseköpfen, die über die Stränge des DNS - Moleküls vom 3'-Ende zum 5'-Ende laufen.



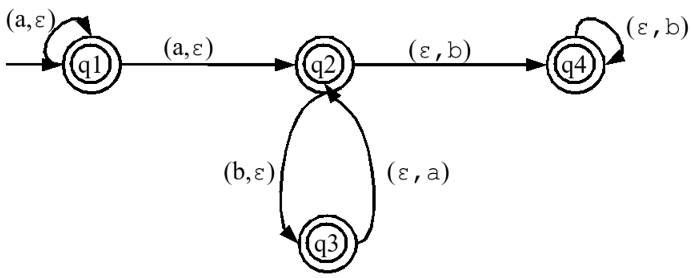
4.1 Akzeptierte Sprachen

Definition 4.1 (Watson - Crick - Automat)
Ein Watson - Crick - Automat (WK - Automat) ist ein Tupel $A = (V, \rho, Q, \iota, T, F)$, wobei

- V ein Alphabet;
- $\rho \subseteq V^2$ eine Komplementaritätsrelation;
- Q endliche Zustandsmenge;
- $\iota \in Q$ Initialzustand;
- $T \subseteq Q \times V^* \times V^* \times Q$ endliche Menge von Transitionen;
- $F \subseteq Q$ eine Menge von Finalzuständen

ist.
Eine Berechnungsfolge in A ist eine endliche Folge von Transitionen $\binom{p=t_1t_2t_3}{t_n}$ aus T mit $t_i = (q_i, u_i, v_i, q_{n+1})$ für $1 \leq i \leq n$.
Sie ist erfolgreich, falls $q_i = i$ und $q_{n+1} \in F$ und $\binom{u_1u_2 \ldots u_n}{v_1v_2 \ldots v_n} \in \rho^*$. Das Paar $\binom{u_1u_2 \ldots v_n}{v_1v_2 \ldots v_n} \in \binom{V^*}{V^*}$ heißt die Beschriftung von p.

Beispiel 4.2
$Q = \{q_1, \ldots, q_4\}, i = q_1, F = Q$
$V = \{a, b\}, \rho = \left\{\binom{a}{a}, \binom{b}{b}\right\}$



$\binom{u}{v}$ Beschriftung eines erfolgreichen Pfades
$v, u \in a * b*$, d.h. $\exists n, m \in \mathbb{N}$ mit $u = v = a^m b^n$
Angenommen dieser Pfad endet im:
$q_1 : v = \epsilon \Rightarrow u = \epsilon$ da (u = v)
$q_2 : m > 0$, da Transition von q_1 nach q_2 in Pfad entahlen wegen $u = v$ wird Transition

von q_1 nach q_2 genau m-mal durchlaufen.
$\Rightarrow$ Transition von q_2 nac q_3 wird genau m-mal durchlaufen
$\Rightarrow n = m > 0$ führt dazu, dass v keine b's enthält.
q_3 : u enthält aber das v nicht $\rightsquigarrow mu = v$
$q_4 : m = n > 0$
$\Rightarrow L(A) \geq \{a^m b^m | m \in \mathbb{N}\}$ sogar $= L(A)$ also nicht regulär.

für $V = \{a_1, \ldots, a_n\}$ und $w \in V^*$ sei $\#aw$ die Anzahl der Vorkommen des Buchstaben $a \in V$ in w und $\psi(w) = (\#a_1 w, \#a_2 w, \ldots, \#a_n w) \subseteq \mathbb{N}^n$
Eine Menge $X \subseteq \mathbb{N}^n$ heißt <u>semilinear</u>, wenn es ein $k \in \mathbb{N} \wedge \overrightarrow{x_i}, \overrightarrow{y_i} \in \mathbb{N}^n$ für $1 \leq i \leq k$ gibt mit

$$X = \bigcup_{1 \leq i \leq k} \overrightarrow{x_i} + \{l \cdot \overrightarrow{y_i} | l \in \mathbb{N}\}$$

Bemerkung
Vereinigung und Schnitt semilinearer Mengen sind semilinear.

Satz von Parikh 4.3
Sei $L \subseteq V^*$ kontextfrei. Dann ist $\psi(L) \subseteq \mathbb{N}^{|V|}$ semilinear.

Lemma 4.4
Sei A WK - Automat mit $L(A) \subseteq a^*$. Dann ist $L(A)$ regulär.

Beweis
Sei $A = (V, \rho, Q, \iota, F, T)$ WL - Automat mit $L(A) \subseteq a^*$
$A := \{b \in V : \binom{a}{b} \in \rho\}$
wir konstruieren eine lineare/kontextfreie Grammatik
$G = (N, \{a, b\}, S, P)$
$\curvearrowright$
$N = Q$
$S = \iota$
$P = \{q \rightarrow \epsilon : q \in F\} \cup \{q \rightarrow a^i q' b^j$: es gibt $(q, \binom{a}{v}, q') \in T, |v| = j, v \in A^*\}$

Behauptung
$\{a^m b^m : a^m \in L(A)\} = L(G) \cap \{a^n b^n : n \in \mathbb{N}\}$

Beweis
"$\subseteq$" Sei $a^m \in L(A)$
$\Rightarrow$ es gibt $v \in V^*$ komplementär zu a^m und ein erfolgreicher Pfad mit Beschriftung $\begin{bmatrix} a_m \\ \text{b} \end{bmatrix}$
$\iota = q_0 \underrightarrow{u_1 v_1} q_1 \underrightarrow{u_2 v_2} q_2 \ldots \underrightarrow{u_n v_n} q_n \in F$
$\curvearrowright u_1 u_2 \ldots u_n = a^m$
$\curvearrowright v_1 v_2 \ldots v_n = v \in A^*$
$\Rightarrow (q_i \rightarrow u_{i+1} q_{i+1} b^{v_{i+1}}) \in P$
$\Rightarrow \iota \rightarrow u_1 q_1 b^{v_1} \rightarrow u_1 u_2 q_2 b^{v_2} b^{v_1} \cdots \rightarrow u_1 u_2 \ldots u_n q_n b^{v_n v_{n-1} \ldots v_1}$
$\rightarrow a^m b^m$
also "$\subseteq$" ist gezeigt.

Beweis
"$\supseteq$"
Sei $\iota = q_0 \rightarrow_G a^{m_1} q_1 b^{n_1} \rightarrow_G a^{m_1} a^{m_2} q_2 b^{n_2} b^{n_1} \cdots \rightarrow_G a^{m_1} a^{m_2} \ldots a^{m_k} q_k b^{n_k} b^{n_{k-1}} \ldots b^{n_1} \rightarrow_G a^{m_1 \ldots m_k} b^{n_1 + n_2 \ldots n_k} \in \{a^n b^n : n \in \mathbb{N}\}$ Ableitung in G
$\Rightarrow q_k \in F$
es gibt Transitionen $(q_i, a^{m_{i+1}}, v_{i+1}, q_{i+1}) \in T$ mit $v_{i+1} \in A^*, |v_{i+1}| = u_{i+1}$
$q_0 \underrightarrow{(a^{m_1 v_1})} q_1 \underrightarrow{(a^{m_2} v_2)} q_2 \ldots \underrightarrow{a^{m_k} v_k} q_k$
ist ein Pfad in A.

$q_0 = \iota$
$q_k \in F$
$v = v_1 v_2 \dots v_k \in A^*$
$|v| = n_1 + n_2 \dots + n_k = m_1 + m_2 + \dots m_k = m$
also ist die Beschriftung $\binom{a^m}{v} \in \rho^*$
$\Rightarrow$ Pfad erfolgreich $\Rightarrow a^m \in L(A)$
$\Rightarrow a^m b^m \in$ linker Seite
für $w \in \{a,b\}^*$ sei $\psi(w) = (\#_a w, \#_m w)$
$\curvearrowright$ Anwendung des Satz von Parikh: $\psi(L(G))$ semilinear
$\Rightarrow \psi(L(G)) \cap \{(p,p) : p \in \mathbb{N}\}$ ist semilinear
$\Rightarrow_{Behauptung}\ \psi(\{a^m b^m : a^m \in L(A)\}) = \{(m,m) : a^m \in L(A)\}$
$\Rightarrow \{m : a^m \in L(A)\}$ semilinear
$\Rightarrow L(A) \cap a^* = L(A)$ regulär

`q.e.d.`

Satz 4.5
$\{L(A)$: A WL - Automat$\} \subseteq CS$

Beweis
$L = \{a^p$: p Primzahl$\} \in CS \backslash REG$
Nach Lemma 4.4 sind die Sprachklassen also verschieden.
Mitgliedschaft von $L(A)$ in CS wird durch linear platzbeschränkte TM (LBA) bewiesen

`q.e.d.`

Ein WK - Automat $A = (V, \rho, Q, \iota, F, T)$ heißt <u>einfach</u>, falls für jedes Tupel $(p, u, v, q) \in T$ gilt $u = \epsilon$ oder $v = \epsilon$

Lemma 4.6
$A' = (V, \rho, Q', \iota', F', T')$ mit $L(A') = L(A) \cup \{\epsilon\}$ und $Q' = F'$

Beweis
`1. Idee:`
Kopf 1 immer auf ungerader Position
Kopf 2 immer auf gerader
(außer am Anfang und bei Akzeptanz)
`2. Idee:`
Köpfe immer schon weiter rechts, unbearbeiteter Teil der Eingabe im Zustand gemerkt.

wähle $m \geq 0$ so, dass alle Wörter in Transitionen von A kürzer sind (d.h. $T \leq Q \times V^{<m} \times V^{<m} \times Q$)
$Q' := Q \times V^{\leq m} \times V^{\leq m} \dot\cup \{\iota', f'\}$
$F' := Q'$
4 Gruppen von Transitionen:

1. Initialisierung:
   $(\iota', \binom{a}{\epsilon}, (\iota, a, \epsilon))$ ist Transition in T' für $a \in V$

2. Vorrausschau:
   für $(p, ux, v) \in Q'$ mit $|x| = 2$:
   $((p, u, v), \binom{x}{\epsilon}, (p, ux, v)) \in T'$
   für $(p, u, v) \in Q'$ mit $|y| = 2$:
   $((p, u, v), \binom{\epsilon}{y}, (p, u, vy)) \in T'$

3. Simulation:
   für $(p.u, v), (p', u', v') \in Q'$
   $((p, u, v), \binom{\epsilon}{\epsilon}, (p', u', v')) \in T'$, falls es $x, y \in V^*$ gibt mit $(p, \binom{x}{y}, p') \in T$ und $u = xu'$ und $v = yv'$

4. Akzeptanz:
   $((p, u, v), \binom{a}{\epsilon}, f')$ falls $(p, \binom{ua}{v}, f) \in T$ für ein $f \in F$
   $((p, u, v), \binom{u}{vb}, f) \in T$ für ein $f \in F$

Dann ist $A' = (V, \rho, Q', \iota', F', T')$ ein einfacher WK-Automat mit $F' = Q'$
man zeigt jetzt:

Behauptung 1
Seien $x, y \in V^*, (p, u, v) \in Q'$. Dann sind äquivalent

- $\exists$ Pfad von ι' nach (p, u, v) mit Beschriftung $\binom{x}{y}$ in A'
- $\exists x', y' \in V^* : x = x'u, y = y'v, |x|$ ungerade, $|y|$ gerade, $\exists$ Pfad in A von ι nach p mit Beschriftung $\binom{x'}{y'}$

Beweis
induktive über jeweilige Pfadlänge

Behauptung 2
Seien $x, y \in V^+$. Dann sind äquivalent

- $\exists$ Pfad in A' von ι' nach f' mit Beschriftung $\binom{x}{y}$
- $\exists$ Pfad in A von ι nach $f \in F$ mit Beschriftung $\binom{x}{y}$ und $|x| \equiv |y| mod 2$

  Beweis
  verwende die Behauptung 1 und die spezielle Gestalt der Akzeptanztransition.

damit folgt Behauptung des Lemmas:
Sei $x \in V^+$
$x \in L(A) \Leftrightarrow \exists y \in V^+ : \binom{x}{y} \in \rho^*$ und $\exists$ Pfad in A von ι nach $f \in F$ mit Beschriftung $\binom{x}{y}$
$\Leftrightarrow y \in V^+ : \binom{x}{y} \in \rho^*$ und $\exists$ Pfad in A' von ι' nach f' mit Beschriftung $\binom{x}{y}$
$\Leftrightarrow x \in L(A')$

q.e.d

Lemma 4.7
Sei $L \subseteq v^*$ regulär $\Rightarrow$ es gibt einfachen WK - Automaten A mit $L(A) = L \cup \{\epsilon\}$ und $F = Q$

Beweis
Übung

q.e.d

Lemma 4.8
A einfacher WK - Automat mit $|Q| = 1$ (es gibt nur einen Zustand). Dann ist $L(A)$ regulär.

Beweis
Sei $W_1 = \{w \in V^* : (p, \binom{w}{\epsilon}, p) \in T\}$ endlich
$W_2 = \{w \in V^* : (p, \binom{\epsilon}{W}, p) \in T\}$ endlich
$L(A) = \{x \in W_1^* : \exists y \in W_2^*, \binom{x}{y} \in \rho^*$
$H := \{x \in V^* : \exists y \in W_2^* : \binom{x}{y} \in \rho^*\}$ regulär und $L(A) = W_1^* \cap H$

q.e.d

Satz 4.9
Es gelte die folgenden Beziehungen:

	CS	
	$\subsetneq$	
	$\{L^+(A)\ :\ A$ WK - Automat$\}\ =$	
	AWK	
	$=$	
	$\{L^+(A)\ :$	
	A einfacher WK - Automat mit $Q =$	
	$F\} = FSWK$	
$\subsetneq$		$\subsetneq$
REG	$\not\subseteq$	$\{L^+(A)\ :$ A WK - Automat, $\lvert Q\rvert\ =\ 1\}\ =$ NWK
$\subsetneq$	$\not\supseteq$	$\subsetneq$
	$\{L^+(A)\ :$	
	Aeinfacher WK -Automat$\lvert Q\rvert\ =$	
	$1\} = NSWK$	

Beweis

$$AWK \subsetneq CS : S4.5$$
$$AWK \subseteq FSWK : L4.6$$
$$AWK \supseteq trivial$$
$$REG \subseteq FSWK : L4.7$$
$$NSWK \subseteq REG : L4.8$$
$$REG \not\subseteq NWK$$
$$L \in NWK \Rightarrow L = L^+ \text{ gilt nicht für alle reg. Mengen}$$
$$NWK \not\subseteq REG : TS_V \in NWK(\ddot{U}bung); TS_V! \in REG$$

aus der Unvergleichbarkeit von NWK und REG folgt die Echtheit der restlichen Inklusionen.

q.e.d.