Formale Sprachen: Graphtransformation

Hans-Jörg Kreowski

SS 2004

die folgenden Folien betreffen den Arbeitsgegenstand der Lehrveranstaltung am 4., 6., 11. und 13. Mai:

Graphgrammatik und erzeugte Graphsprache

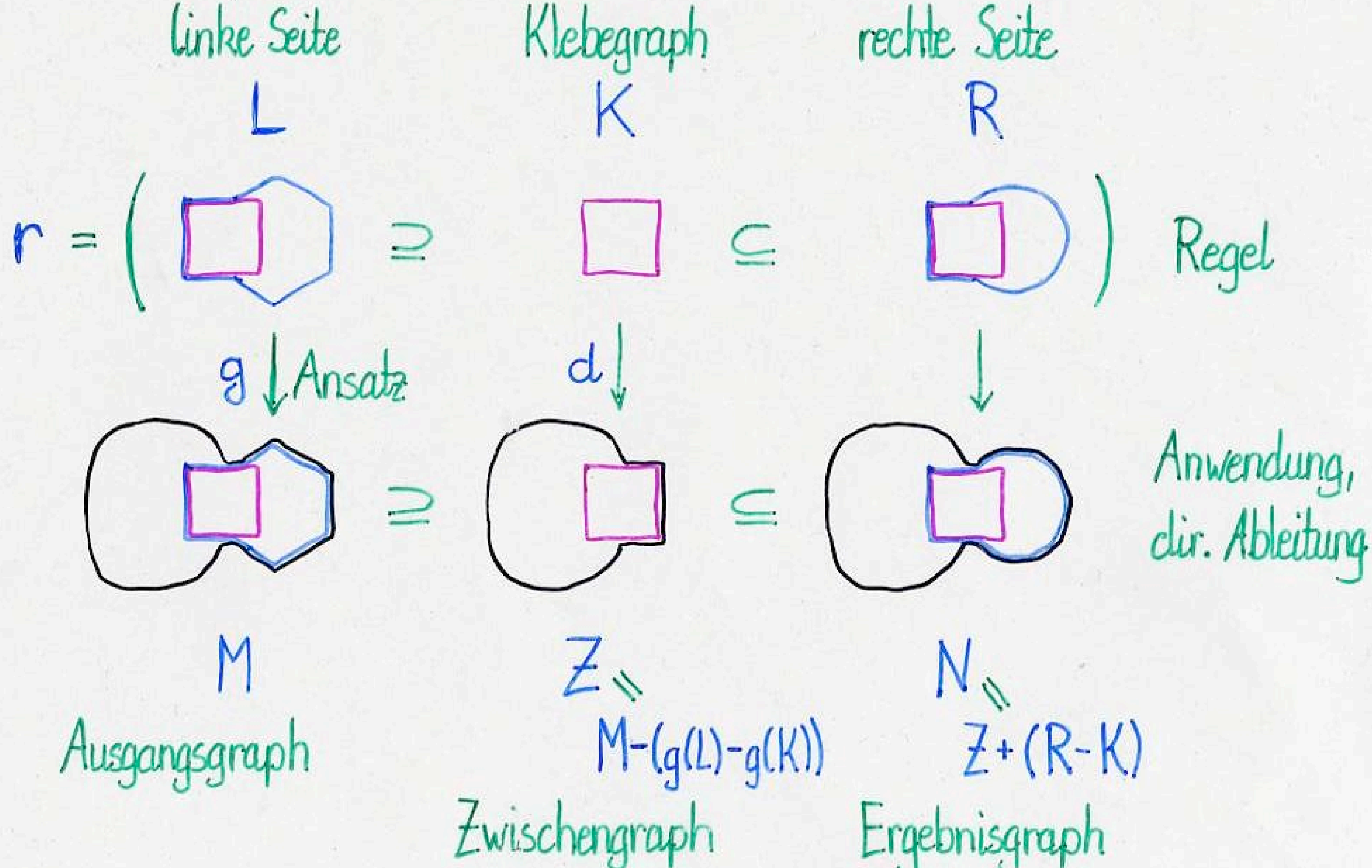
Graphgrammatik-Maschine

Übersetzung von Chomsky-Grammatiken zu Graphgrammatiken

Übersetzung von Postschen Korrespondenzproblemen zu Graphgrammatiken

diverse Nichtentscheidbarkeitsergebnisse

Regelanwendung $M \underset{r/g}{\Longrightarrow} N$ (Schema)



Graphgrammatik

Sei C Markierungsalphabet
und $\mathcal{G}_X$ die Menge der über $X \subseteq C$ markierten Graphen.

Syntax

▷ $G = (T, P, Z)$ Graphgrammatik:

- $T \subseteq C$ Menge terminaler Zeichen
- P endl. Menge von Regeln
  der Form $r = (L \supseteq K \subseteq R)$

$$L \in \mathcal{G}_C, \quad K \in \mathcal{G}_C, \quad R \in \mathcal{G}_C$$

- $Z \in \mathcal{G}_C$ Startgraph

Semantik

▷ $L(G) = \{ M \in \mathcal{G}_T \mid Z \overset{*}{\underset{P}{\Rightarrow}} M \}$ erzeugte Graphsprache

Welche Sprache erzeugt $G_? = (T_?, P_?, Z_?)$?

$P_?$:

$r_1 =$

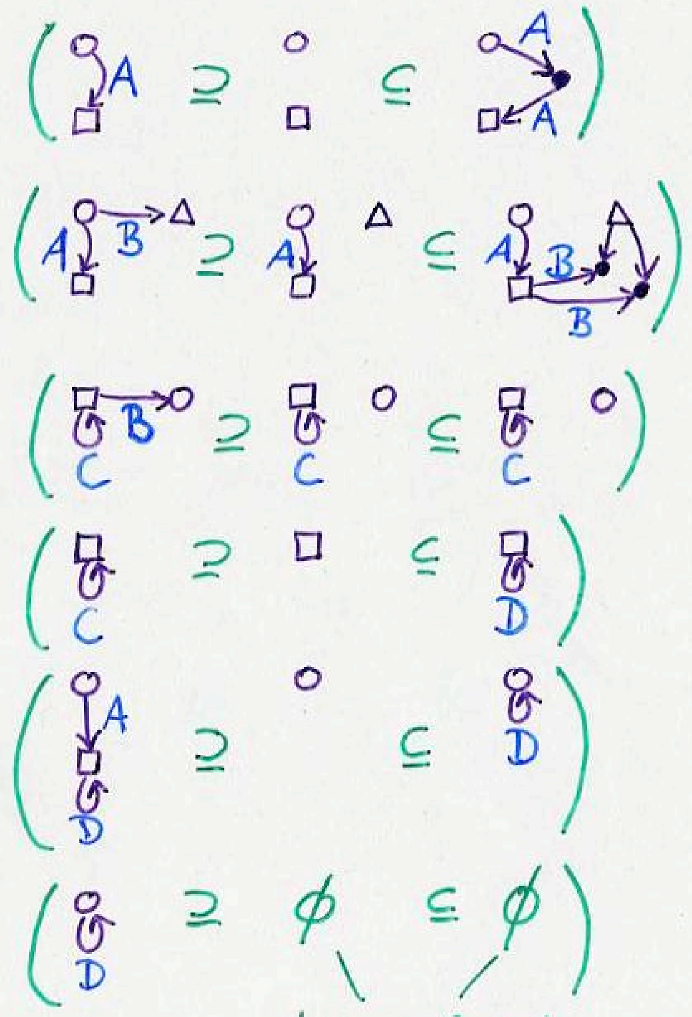
$r_2 =$

$r_3 =$

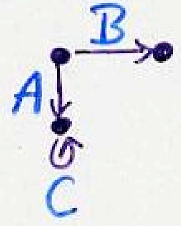
$r_4 =$

$r_5 =$

$r_6 =$



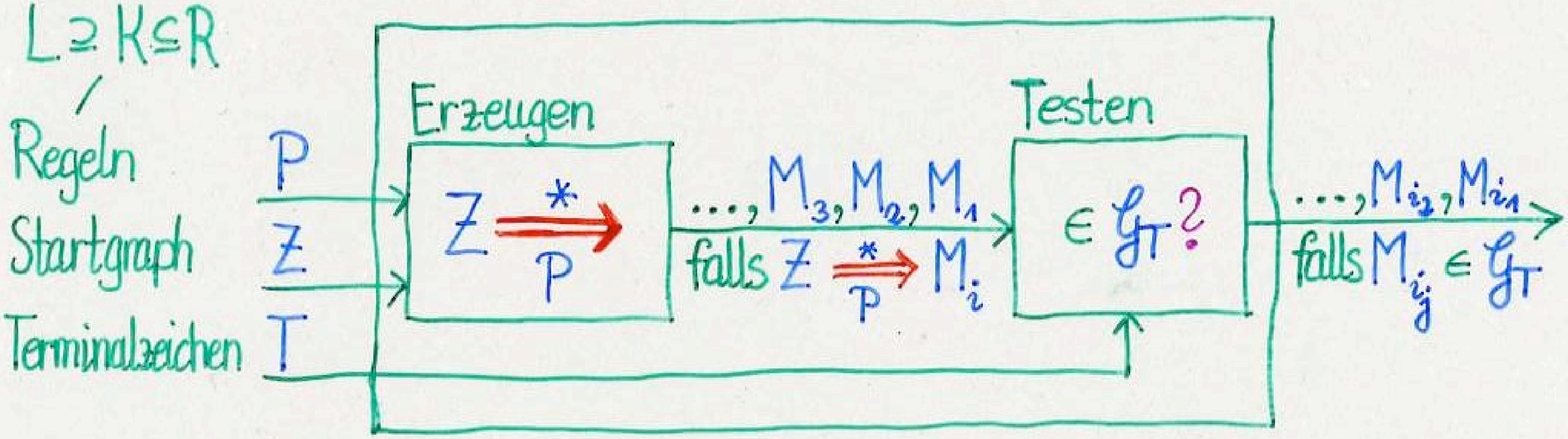
leerer Graph

$Z_? =$ 

$A, B, C, D \notin T_? = \{*\}$

einziges Terminalzeichen, wird nicht gemalt

Die GRAGRA-Maschine



Prozess der Spracherzeugung

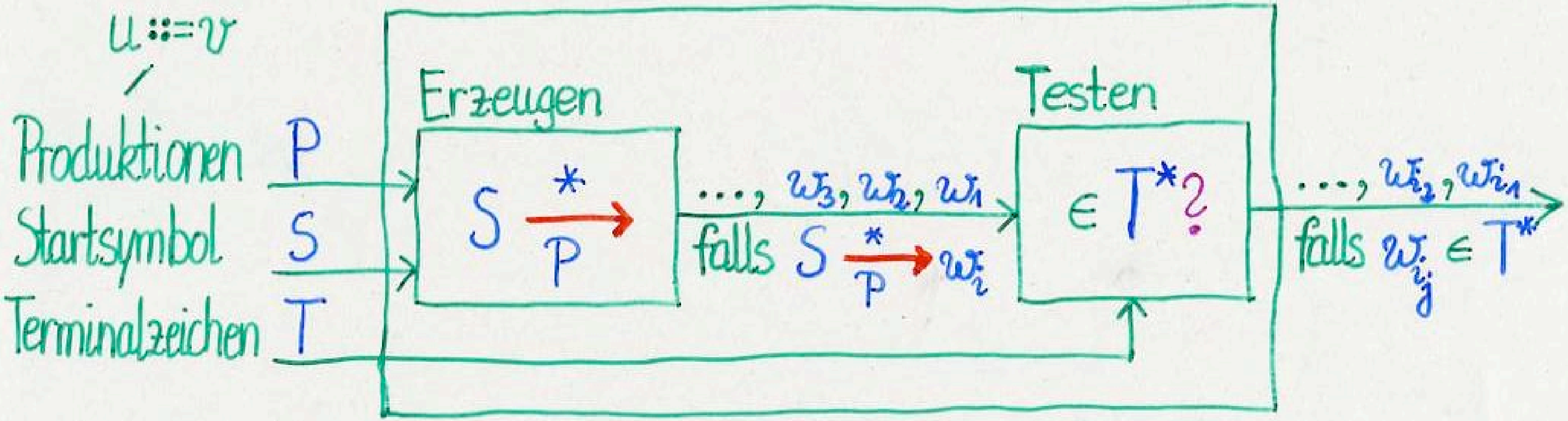
operationelle Semantik

Chomsky-Grammatik

Syntax ▷ $G=(N,T,P,S)$ Chomsky-Grammatik:

- N Menge nichtterminaler Zeichen
- T Menge terminaler Zeichen
- P endl. Menge von Produktionen der Form $p=(u ::= v)$, $u \in (N \cup T)^* \setminus T^*$, $v \in (N \cup T)^*$
- $S \in N$ Startsymbol oder $S \in (N \cup T)^*$ Startwort

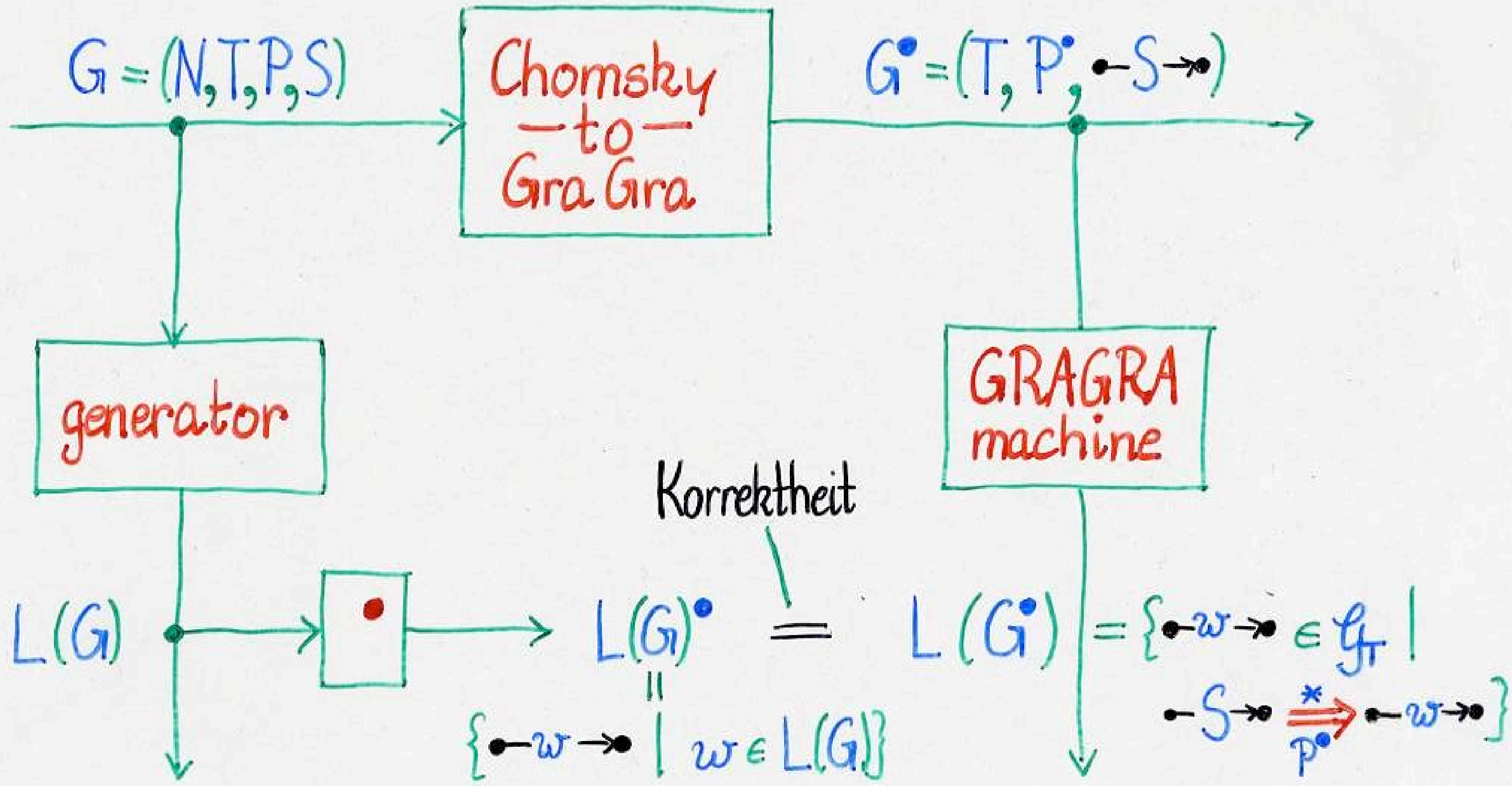
Semantik ▷ $L(G)=\{\, w \in T^* \mid S \xrightarrow[P]{*} w \,\}$ erzeugte Sprache



Prozess der Spracherzeugung

operationelle Semantik

Übersetzung von Chomsky-Grammatiken in Graphgrammatiken



$$\bullet\!-x_1 x_2 \cdots x_n\!\rightarrow\!\bullet \;=\; \bullet \xrightarrow{x_1} \bullet \xrightarrow{x_2} \cdots \xrightarrow{x_n} \bullet$$

Übersetzung von Wörtern zu Graphen

$$\begin{array}{ccc}
w = x_1 x_2 \cdots x_n & \longmapsto & \bullet\!\xrightarrow{\;w\;}\!\bullet = \bullet\xrightarrow{x_1}\bullet\xrightarrow{x_2}\bullet \cdots \bullet\xrightarrow{x_n}\bullet \\
\rotatebox{90}{$\in$} & & \rotatebox{90}{$\in$} \\
C^* & & \mathcal{G}_C
\end{array}$$

C^* Wort $\qquad$ $\mathcal{G}_C$ String-Graph

Spezialfall:

$$w = \lambda \quad \longmapsto \quad \bullet\!\xrightarrow{\;\lambda\;}\!\bullet = \bullet$$

leeres Wort

Folgerungen aus der Übersetzung von Chomsky-Grammatiken

Übersetzung: $G=(N,T,P,S) \longmapsto G^\bullet=(T,P^\bullet,\bullet\!-S\!\to\!\bullet)$

Beobachtung:

$$\begin{array}{c} L(G)^\bullet \\ \| \\ \{\bullet\!-w\!\to\!\bullet \mid w\in L(G)\} \end{array} = \{\bullet\!-w\!\to\!\bullet \mid w\in T^*,\ S\overset{*}{\underset{P}{\Rightarrow}} w\} = \begin{array}{c} L(G^\bullet) \\ \| \\ \{\bullet\!-w\!\to\!\bullet \in \mathcal{G}_T \mid \bullet\!-S\!\to\!\bullet \overset{*}{\underset{P^\bullet}{\Rightarrow}} \bullet\!-w\!\to\!\bullet\} \end{array}$$

Insbesondere:

- $L(G) \left\{\begin{array}{l}\text{leer}\\ \text{endlich}\end{array}\right\}$ gdw. $L(G^\bullet) \left\{\begin{array}{l}\text{leer}\\ \text{endlich}\end{array}\right\}$
- $w\in L(G)$ gdw. $\bullet\!-w\!\to\!\bullet \in L(G^\bullet)$

Folgerung: Für Graphgrammatiken GG sind unentscheidbar:

(1) Leerheitsproblem $\quad GG \to \boxed{L(GG)=\emptyset\,?} \to$ JA / NEIN

(2) Endlichkeitsproblem $\quad GG \to \boxed{L(GG)\text{ endlich}\,?} \to$ JA / NEIN

(3) Mitgliedschaftsproblem $\quad GG, M \to \boxed{M\in L(GG)\,?} \to$ JA / NEIN

Was verrät die Syntax über die Semantik?

z.B. $G = (T, P, Z) \longrightarrow \boxed{\text{GraGra Machine}} \longrightarrow L(G)$ *)

- typische Fragen bzgl. einer Graph-eigenschaft **prop**:
  - existiert $M \in L(G)$ mit $prop(M)$?
  - gilt $prop(M)$ für alle $M \in L(G)$?

-----

*) explizit erzeugt ist zu jeder Zeit nur endlicher Ausschnitt

Postsches Korrespondenzproblem

PCP

$K = \begin{bmatrix} u_1 \\ v_1 \end{bmatrix}, \dots, \begin{bmatrix} u_n \\ v_n \end{bmatrix}$

$u_i, v_i \in A^*$

A endl. Alphabet

ex. $i_1 i_2 \cdots i_k$ $(k \geq 1)$

mit

$u_{i_1} u_{i_2} \cdots u_{i_k} = v_{i_1} v_{i_2} \cdots v_{i_k}$

?

JA

(dann heißt $i_1 \cdots i_k$ eine <u>Lösung</u>)

NEIN

Fakt: Das Postsche Korrespondenzproblem ist unentscheidbar.

Übersetzung von PCPs in Graphgrammatiken

$K = \begin{bmatrix} u_1 \\ v_1 \end{bmatrix} \cdots \begin{bmatrix} u_n \\ v_n \end{bmatrix}$ → PCP-2-GraGra → $G_K = (A, P_K, \overset{\bullet}{\underset{\bullet}{\downarrow}} S)$

solvable?

JA, falls $u_{i_1} \cdots u_{i_k} = v_{i_1} \cdots v_{i_k}$ für $i_1 \cdots i_k \neq \lambda$

NEIN sonst

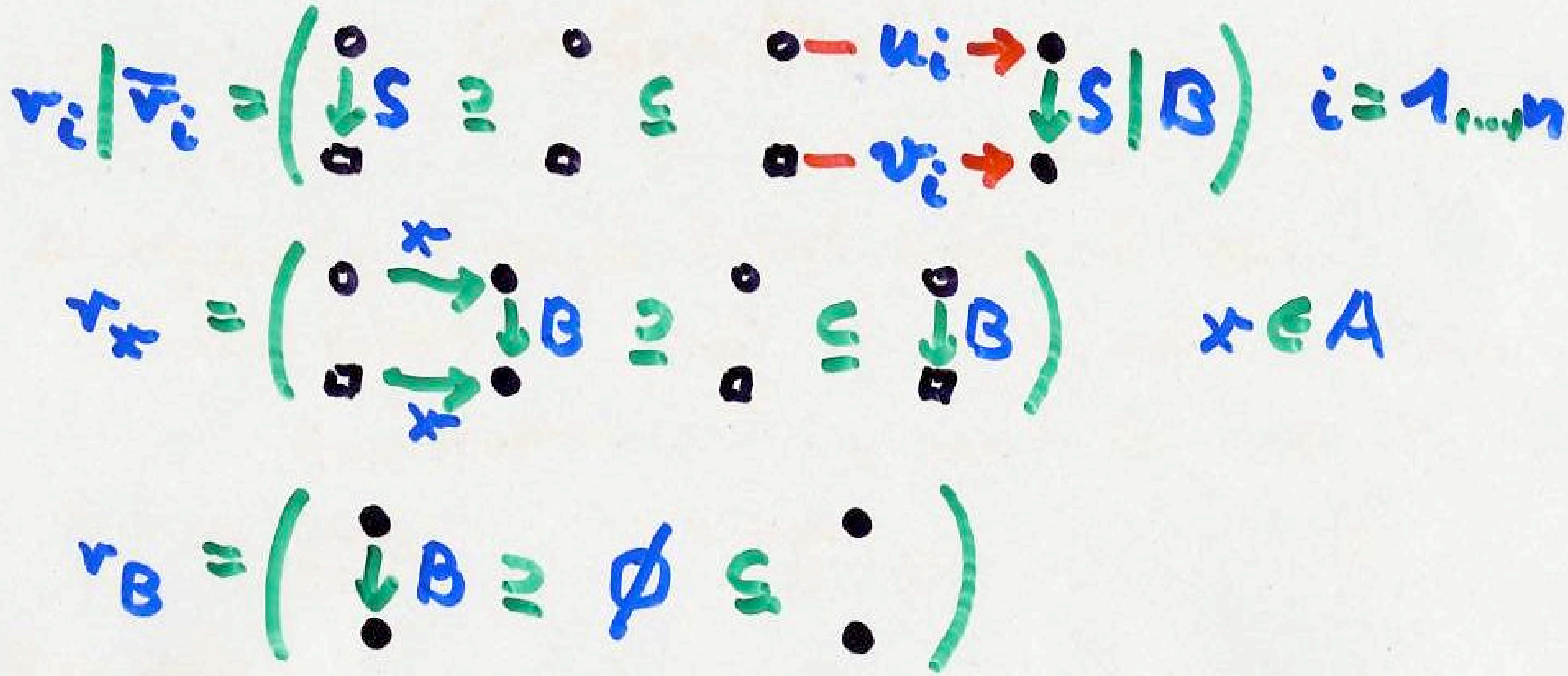
GraGra Machine

$L(G_K)$

Korrektheit: K lösbar gdw $L(G_K) = \{:\}$
(K nicht lösbar gdw. $L(G_K) = \emptyset$)

Konsequenz: Es ist nicht entscheidbar, welcher der beiden Fälle für $L(G_K)$ eintritt

Regeln in P_K


$r_i \mid \bar{r}_i =$
S
$\supseteq$
$\subseteq$
u_i
v_i
S | B
$i = 1, \ldots, n$
$r_x =$
x
B
$\supseteq$
$\subseteq$
B
$x \in A$
$r_B =$
B
$\supseteq \emptyset \subseteq$

Beobachtung

Es ist für Graphgrammatiken G nicht entscheidbar, ob in L(G) ein Graph existiert, der diskret | nicht zusammenhängend | kreisfrei | kein Baum | nicht Hamiltonsch | .... ist oder ob alle Graphen in L(G) zusammenhängend | Bäume | Hamiltonsch | .... sind

Beweis

sonst wäre die Lösbarkeit des Postschen Korrespondenz-Problems entscheidbar