Regulated Array Grammars of Finite Index

C. Gruber, J. Reiter

TU Wien

3.März, 2010

Table of contents

n-dimensional array

Ein n-dimensionales Array A über ein Alphabet V (Menge aller non-terminal und terminal Symbole) ist eine Funktion

$$A : Z^n \rightarrow V \cup \{\#\} \quad n \in N = \{1, 2, ...\}$$

wobei

$$shape(A) = \{v \in Z^n | A(v) \neq \#\}$$

endlich ist und $\# \notin V$ als *background* oder *blank Symbol* bezeichnet wird. Das Array A kann nun so definiert werden

$$A = \{(v, A(v)) \mid v \in shape(A)\}.$$

n-dimensional array production and grammar

Eine n-dimensionale Array Produktion p über dem Alphabet V ist ein Tripel (W, A_1, A_2) wobei $W \subseteq Z^n$ eine endliche Menge von Koordinaten ist und A_1 und A_2 Abbildungen von W auf $V \cup \{\#\}$ sind.

Eine n-dimensionale Array Grammatik kann nun als Sechstupel

$$G = (n, V_N, V_T, \#, P, \{(v_0, S)\})$$

definiert werden. $\{(v_0, S)\}$ wird als Startarray (Axiom), v_0 als Startvektor und S als das Startsymbol bezeichnet.

matrix grammar

Eine Matrixgrammatik G_M ist ein 4-Tupel

$$G_M = (V_N, V_T, (M, F), S),$$

F kann auch als Fehlermenge bezeichnet werden. Ist $F = \emptyset$, dann kann G_M als Matrixgrammatik ohne appearence checking bezeichnet werden.

graph controlled grammar

Eine graph-controlled Grammatik G_P ist ein 4-Tupel

$$G_M = (V_N, V_T, (R, L_{in}, L_{fin}), S),$$

R ist eine endliche Menge von Regeln r der Form

$$(l(r) : p(l(r)), \sigma(l(r)), \varphi(l(r))).$$

Falls alle Felder $\varphi(l(r))$ leer sind fr alle $r \in R$, dann kann G_P als graph-controlled Grammatik ohne appearence checking bezeichnet werden.

Matrix- und graph-controlled Grammatiken knnen in Arraygrammatiken direkt bergefhrt werden, indem ihre Produktionen durch Arrayproduktionen ersetzt werden.

graph controlled grammar

Eine graph-controlled Grammatik G_P ist ein 4-Tupel

$$G_M = (V_N, V_T, (R, L_{in}, L_{fin}), S),$$

R ist eine endliche Menge von Regeln r der Form

$$(l(r) : p(l(r)), \sigma(l(r)), \varphi(l(r))).$$

Falls alle Felder $\varphi(l(r))$ leer sind fr alle $r \in R$, dann kann G_P als graph-controlled Grammatik ohne appearence checking bezeichnet werden.

Matrix- und graph-controlled Grammatiken knnen in Arraygrammatiken direkt bergefhrt werden, indem ihre Produktionen durch Arrayproduktionen ersetzt werden.

bounded derivations

Index einer Ableitung

Der Index einer Ableitung D eines Terminalobjekts w in einer Grammatik G ist mit der maximalen Anzahl von non-terminal Symbolen, die in einem Zwischenableitungsschritt vorkommen, definiert und wird mit $ind_{G,D}(w)$ bezeichnet.

Weiters bezeichnet $ind_{G,min}(w)$ bzw. $ind_{G,max}(w)$ das Minimum bzw. das Maximum aus der Menge

$$\{ind_{G,D}(w) \mid w \text{ is generated by } G\}$$

Entsprechend gibt es nun die Definition fr die Grammatik

$$ind_Y(G) = sup\{ind_{G,Y}(w) \mid w \text{ is generated by } G\} \quad Y \in \{min, max\}$$

Bei einer endlichen Index Restriktion nur Objekte w, die von einer Grammatik G mit einer Ableitung $ind_{G,Y}(w) \leq k$ fr $Y \in \{min, max\}$.

bounded derivations

Index einer Ableitung

Der Index einer Ableitung D eines Terminalobjekts w in einer Grammatik G ist mit der maximalen Anzahl von non-terminal Symbolen, die in einem Zwischenableitungsschritt vorkommen, definiert und wird mit $ind_{G,D}(w)$ bezeichnet.

Weiters bezeichnet $ind_{G,min}(w)$ bzw. $ind_{G,max}(w)$ das Minimum bzw. das Maximum aus der Menge

$$\{ind_{G,D}(w) \mid w \text{ is generated by } G\}$$

Entsprechend gibt es nun die Definition fr die Grammatik

$$ind_Y(G) = sup\{ind_{G,Y}(w) \mid w \text{ is generated by } G\} \quad Y \in \{min, max\}$$

Bei einer endlichen Index Restriktion nur Objekte w, die von einer Grammatik G mit einer Ableitung $ind_{G,Y}(w) \leq k$ fr $Y \in \{min, max\}$.

grammar with prescribed teams

Eine Grammatik mit prescribed teams G_t ist ein 4-Tupel

$$G_t = (V_N, V_T, (P, R, F), S),$$

$G = (V_N, V_T, P, S)$ ist eine kontextfreie Grammatik, R ist eine endliche Menge von Teams aus P und F ist die Menge von Produktionen, die bei dem appearence checking bersprungen werden knnen. Ist $F = \emptyset$, dann kann G_t als Grammatik mit prescribed teams ohne appearence checking bezeichnet werden.

Beispiel fr eine Grammatik mit prescribed teams(1)

2-dimensionale Array-Grammatik mit prescibed teams:

$$G = (n, \{D, E, L, Q, R, S, U\}, \{a\}, \#, (P, R, F), \{((0,0), S)\})$$

$$P = \left\{ \begin{array}{l} \# \\ S\# \end{array} \rightarrow \begin{array}{l} L \\ aD \end{array}, \begin{array}{l} \# \\ L \end{array} \rightarrow \begin{array}{l} L \\ a \end{array}, D\# \rightarrow aD, \begin{array}{l} \#\# \\ L \end{array} \rightarrow \begin{array}{l} aU \\ a \end{array}, \right.$$

$$D\# \rightarrow aR, \begin{array}{l} \# \\ R \end{array} \rightarrow \begin{array}{l} R \\ a \end{array}, U\# \rightarrow aU,$$

$$\left. U\# \rightarrow aU, U \rightarrow E, \begin{array}{l} \# \\ R \end{array} \rightarrow \begin{array}{l} Q \\ a \end{array}, R \rightarrow a, E \rightarrow a \right\}$$

Beispiel fr eine Grammatik mit prescribed teams(2)

$$R = \left\{ \left\langle \begin{matrix} \# \\ S\# \end{matrix} \rightarrow \begin{matrix} L \\ aD \end{matrix} \right\rangle, \left\langle \begin{matrix} \# \\ L \end{matrix} \rightarrow \begin{matrix} L \\ a \end{matrix}, \; D\# \rightarrow aD \right\rangle, \right.$$

$$\left\langle \begin{matrix} \#\# \\ L \end{matrix} \rightarrow \begin{matrix} aU \\ a \end{matrix}, \; D\# \rightarrow aR \right\rangle, \left\langle \begin{matrix} \# \\ R \end{matrix} \rightarrow \begin{matrix} R \\ a \end{matrix}, \; U\# \rightarrow aU \right\rangle,$$

$$\left. \left\langle U\# \rightarrow aU, U \rightarrow E, \begin{matrix} \# \\ R \end{matrix} \rightarrow \begin{matrix} Q \\ a \end{matrix} \right\rangle, \langle R \rightarrow a, E \rightarrow a \rangle \right\}$$

$$F = \left\{ \begin{matrix} \# \\ R \end{matrix} \rightarrow \begin{matrix} Q \\ a \end{matrix} \right\}$$

unnumbered lists

- Introduction to $\LaTeX$
- Course 2
- Termpapers and presentations with $\LaTeX$
- Beamer class

numbered lists

1. Introduction to LaTeX
2. Course 2
3. Termpapers and presentations with LaTeX
4. Beamer class

numbered lists with pause

1. Introduction to LaTeX
2. Course 2
3. Termpapers and presentations with LaTeX
4. Beamer class

numbered lists with pause

1. Introduction to LaTeX
2. Course 2
3. Termpapers and presentations with LaTeX
4. Beamer class

numbered lists with pause

1. Introduction to LaTeX
2. Course 2
3. Termpapers and presentations with LaTeX
4. Beamer class

numbered lists with pause

1. Introduction to LaTeX
2. Course 2
3. Termpapers and presentations with LaTeX
4. Beamer class

Tables

Date	Instructor	Title
WS 04/05	Sascha Frank	First steps with LaTeX
SS 05	Sascha Frank	LaTeX Course serial

Tables with pause

A	B	C
1	2	3
A	B	C

Tables with pause

A	B	C
1	2	3
A	B	C

Tables with pause

A	B	C
1	2	3
A	B	C

blocs

title of the bloc

bloc text

title of the bloc

bloc text

title of the bloc

bloc text

splitting screen

- Beamer
- Beamer Class
- Beamer Class Latex

Instructor	Title
Sascha Frank	LaTeX Course 1
Sascha Frank	Course serial