Multiple kontextfreie Grammatiken

Daniel Götzmann

Seminar Formal Grammars
Lehrstuhl für Programmiersysteme

2007-03-22

Generalisierte kontextfreie Grammatiken

Eine generalisierte kontextfreie Grammatik (gcfg) ist ein 5-Tupel $\mathcal{G} = (N, O, F, P, S)$.

- N ist eine endliche Menge von Nichtterminalen.
- O ist eine Menge von n-Tupeln von Strings $(n \geq 1)$ über einer endlichen Menge von Terminalsymbolen.
- F ist eine endliche Menge partieller Funktionen von $O \times \cdots \times O$ nach O.
- P ist eine endliche Menge von Produktionsregeln.
- $S \in N$ ist das Startsymbol.

Produktionsregeln

Produktionsregeln haben die Form

$$A \to f[B^{(1)}, B^{(2)}, \ldots, B^{(q)}]$$

wobei $A, B^{(1)}, \ldots, B^{(q)} \in N$ Nichtterminale sind und $f \in F$ eine partielle Funktion von O^q nach O ist.

Gilt $q = 0$, hat man eine terminierende Regel der Form

$$A \to \theta$$

wobei $\theta \in O$.

Ableitungen

Für $A \in N$, ist $L_{\mathcal{G}}(A)$ die Menge, die alle Tupel enthält, die in $\mathcal{G}$ von A ableitbar sind.

$L_{\mathcal{G}}(A)$ ist die kleinste Menge, die folgende Eigenschaften erfüllt:

- $A \to \theta \in P \Rightarrow \theta \in L_{\mathcal{G}}(A)$
- $A \to f[B^{(1)}, B^{(2)}, \ldots, B^{(q)}] \in P$
  und $\theta_i \in L_{\mathcal{G}}(B^{(i)})$
  und $f[\theta_1, \theta_2, \ldots, \theta_q]$ ist definiert
  $\Rightarrow f[\theta_1, \theta_2, \ldots, \theta_q] \in L_{\mathcal{G}}(A)$

$L(\mathcal{G}) = L_{\mathcal{G}}(S)$ heißt die von $\mathcal{G}$ erzeugte generalisierte kontextfreie Sprache.

Beispiel

$\mathcal{G} = (\{S, A, B\}, (\{a, b, c\}^*)^+, \{f, g_a, g_b, \theta_a, \theta_b\}, P, S)$

$A \to g_a[A]$ $g_a[(x)] = (xa)$	$B \to g_b[B]$ $g_b[(x_1, x_2)] = (bx_1, bx_2)$	$S \to f[A, B]$ $f[(x), (y_1, y_2)] = (y_1 c^{\lvert x \rvert^2} y_2)$
$A \to \theta_a$ $\theta_a = (\varepsilon)$	$B \to \theta_b$ $\theta_b = (\varepsilon, \varepsilon)$	
$L_{\mathcal{G}}(A)$ $= \{(a^m) \mid m \in \mathbb{N}\}$	$L_{\mathcal{G}}(B)$ $= \{(b^n, b^n) \mid n \in \mathbb{N}\}$	$L_{\mathcal{G}}(S)$ $= \{(b^n c^{m^2} b^n) \mid m, n \in \mathbb{N}\}$

$L(\mathcal{G}) = L_{\mathcal{G}}(S) = \{(b^n c^{m^2} b^n) \mid m, n \in \mathbb{N}\}$

Expressivität von gcfg's

- Wenn beliebige Funktionen $f \in F$ erlaubt sind, kann jede von einer Typ-0-Grammatik genenerierte Sprache von einer gcfg generiert werden.

- gcfg's mit beliebigen Funktionen sind sehr expressiv.
  $\longrightarrow$ Einschränkung auf interessante Teilklasse.

- Idee: Nur Funktionen zulassen, die als Konkatenation von konstanten Strings und Komponenten ihrer Argumente definiert sind.

Multiple kontextfreie Grammatiken

Eine m-multiple kontextfreie Grammatik (m-mcfg) ist eine gcfg $\mathcal{G} = (N, O, F, P, S)$, welche die folgenden Eigenschaften hat:

- Jede Funktion $f \in F$ ist eine Funktion, die definiert ist als Konkatenation von konstanten Strings und Komponenten ihrer Argumente.
  Jede Komponente kommt dabei höchstens einmal im Ergebnis vor.

- Für alle $A \in N$ gilt, dass alle von A ableitbaren Tupel die gleiche Größe haben.

- Alle vom Startsymbol S ableitbaren Tupel sind 1-Tupel.

- In einer m-mcfg haben alle Tupel in O höchstens m Komponenten.

Beispiel

$\mathcal{G} = (\{S, A, B\}, \bigcup_{m=1}^{2}(\{a, b, c\}^*)^i, \{f_2, g_a, g_b, \theta_a, \theta_b\}, P, S)$

$A \to g_a[A]$ $g_a[(x)] = (xa)$	$B \to g_b[B]$ $g_b[(x_1, x_2)] = (bx_1, bx_2)$	$S \to f_2[A, B]$ $f_2[(x), (y_1, y_2)] = (y_1 x y_2)$
$A \to \theta_a$ $\theta_a = (\varepsilon)$	$B \to \theta_b$ $\theta_b = (\varepsilon, \varepsilon)$	
$L_{\mathcal{G}}(A)$ $= \{(a^m) \mid m \in \mathbb{N}\}$	$L_{\mathcal{G}}(B)$ $= \{(b^n, b^n) \mid n \in \mathbb{N}\}$	$L_{\mathcal{G}}(S)$ $= \{(b^n a^m b^n) \mid m, n \in \mathbb{N}\}$

$L(\mathcal{G}) = L_{\mathcal{G}}(S) = \{(b^n a^m b^n) \mid m, n \in \mathbb{N}\}$

Pumping-Lemma

Für jede unendliche m-mcfl L existieren $u_j \in T^*$ $(1 \leq j \leq m+1)$, v_j, w_j, $s_j \in T^*$ $(1 \leq j \leq m)$, so dass gilt:

1. $\sum_{j=1}^{m} |v_j s_j| > 0$

2. $\forall i \geq 0 : z_i = u_1 \underline{v_1^i} w_1 \underline{s_1^i} u_2 \underline{v_2^i} w_2 \underline{s_2^i} u_3 \ldots u_m \underline{v_m^i} w_m \underline{s_m^i} u_{m+1} \in L$

Zum Vergleich: Pumping-Lemma für kontextfreie Sprachen:

1. $|v_1 s_1| > 0$

2. $\forall i \geq 0 : z_i = u_1 \underline{v_1^i} w_1 \underline{s_1^i} u_2 \in L$

Expressivität von m-mcfg's

$L = \{a_1^n a_2^n \dots a_{2m+2}^n | n \geq 1, a_i \in T, 1 \leq i \leq 2m+2\}$ ist keine m-mcfl.

Beweis mit dem Pumping-Lemma:

- $a_1 \dots a_1 a_2 \dots a_2 \dots a_k \dots \underbrace{a_k a_{k+1}} \dots a_{k+1} \dots a_{2m+2} \dots a_{2m+2}$
  Ein pumpbarer Teilstring darf nicht zwei verschiedene Terminalsymbole enthalten.

- $\underbrace{a_1 \dots a_1}\underbrace{a_2 \dots a_2}\underbrace{a_3 \dots a_3}\underbrace{a_4 \dots a_4} \dots$ (an $2m+2$ Stellen muss gepumpt werden)

  $u_1 \overbrace{v_1^i} w_1 \overbrace{s_1^i} u_2 \overbrace{v_2^i} w_2 \overbrace{s_2^i} u_3 \dots$ (an höchstens $2m$ Stellen kann gepumpt werden.)

Expressivität von m-mcfg's

$L = \{a_1^n a_2^n \dots a_{2m+2}^n | n \geq 1, a_i \in T, 1 \leq i \leq 2m+2\}$ ist eine (m+1)-mcfl.

Die folgende (m+1)-mcfg $\mathcal{G}$ generiert L:

$\mathcal{G} = (\{S, X\}, \bigcup_{i=1}^{m+1} (\{a_1, a_2, \dots a_{2m+2}\}^*)^i, \{f, g, \theta\}, P, S)$

$X \to g[X]$ $g[(x_1, x_2, \dots, x_{m+1})]$ $= (a_1 x_1 a_2, a_3 x_2 a_4, \dots, a_{2m+1} x_{m+1} a_{2m+2})$	$S \to f[X]$ $f[(x_1, x_2, \dots, x_{m+1})]$ $= (x_1 x_2 \dots x_{m+1})$
$X \to \theta$ $\theta = (a_1 a_2, a_3 a_4, \dots, a_{2m+1} a_{2m+2})$	
$L_{\mathcal{G}}(X)$ $= \{(a_1^n a_2^n, a_3^n a_4^n, \dots, a_{2m+1}^n a_{2m+2}^n) \mid n \geq 1\}$	$L_{\mathcal{G}}(S)$ $= \{(a_1^n a_2^n \dots a_{2m+2}^n) \mid n \geq 1\}$

Expressivität von m-mcfg's

- m-MCFL $\subsetneq$ (m+1)-MCFL
- CFL = 1-MCFL
- TAL $\subsetneq$ 2-MCFL

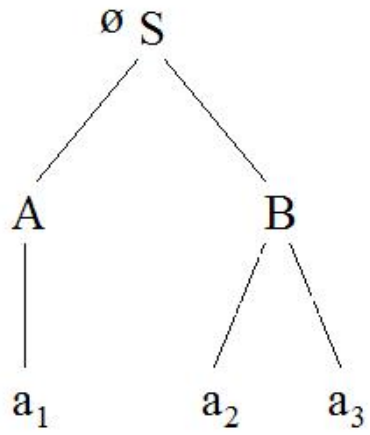
TAL $\subseteq$ 2-MCFL

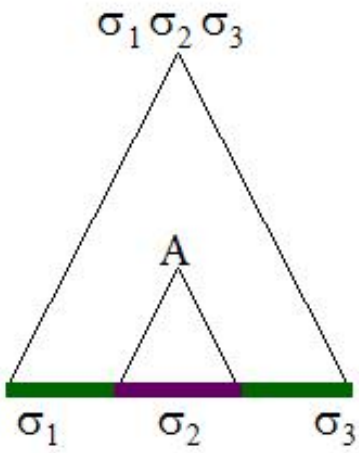
- Idee: Normalform für TAG's
- Initialbäume der TAG's in Produktionsregeln von 2-mcfg umwandeln
- Auxiliarbäume der TAG's in Produktionsregeln von 2-mcfg umwandeln

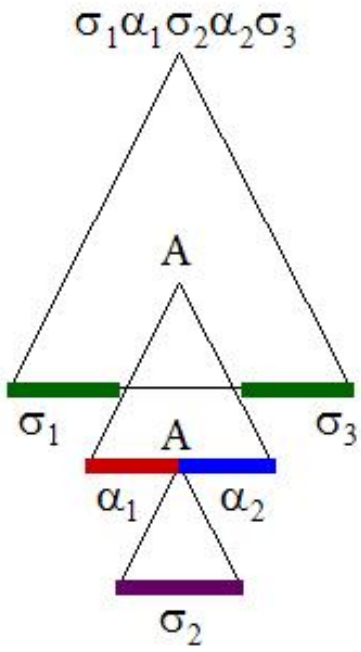
Normalform für TAG's

- Die Wurzel jedes Initialbaums ist S.
- Kein Auxiliarbaum hat S als Wurzel.
- An der Wurzel ist kein Auxiliarbaum adjungierbar.
- Am Fußknoten ist kein Auxiliarbaum adjungierbar.

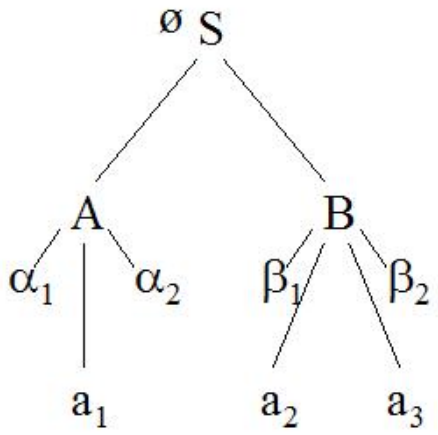
Initialbäume







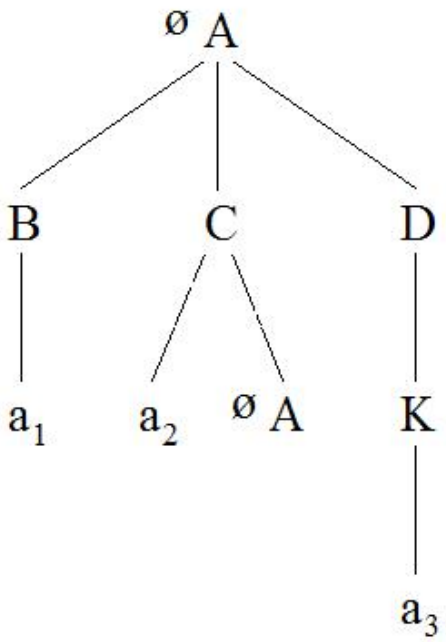
Initialbäume

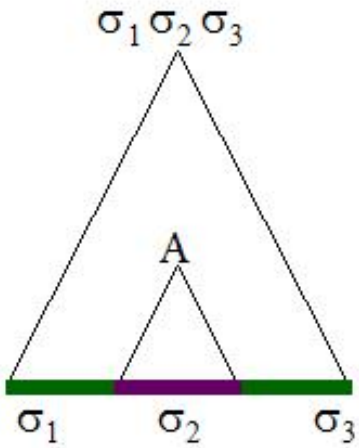


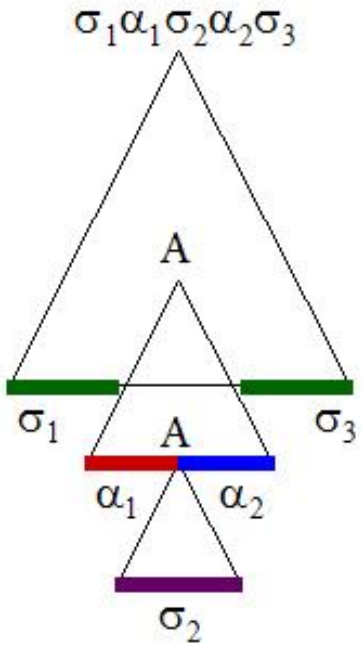
$$S \to f[A, B] \in P$$

$$f[(\alpha_1, \alpha_2), (\beta_1, \beta_2)]$$
$$= (\alpha_1 a_1 \alpha_2 \beta_1 a_2 a_3 \beta_2)$$

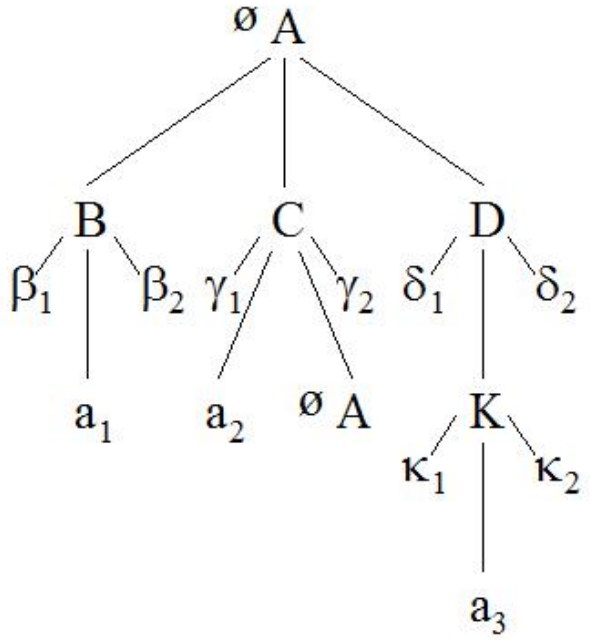
Auxiliarbäume







Auxiliarbäume



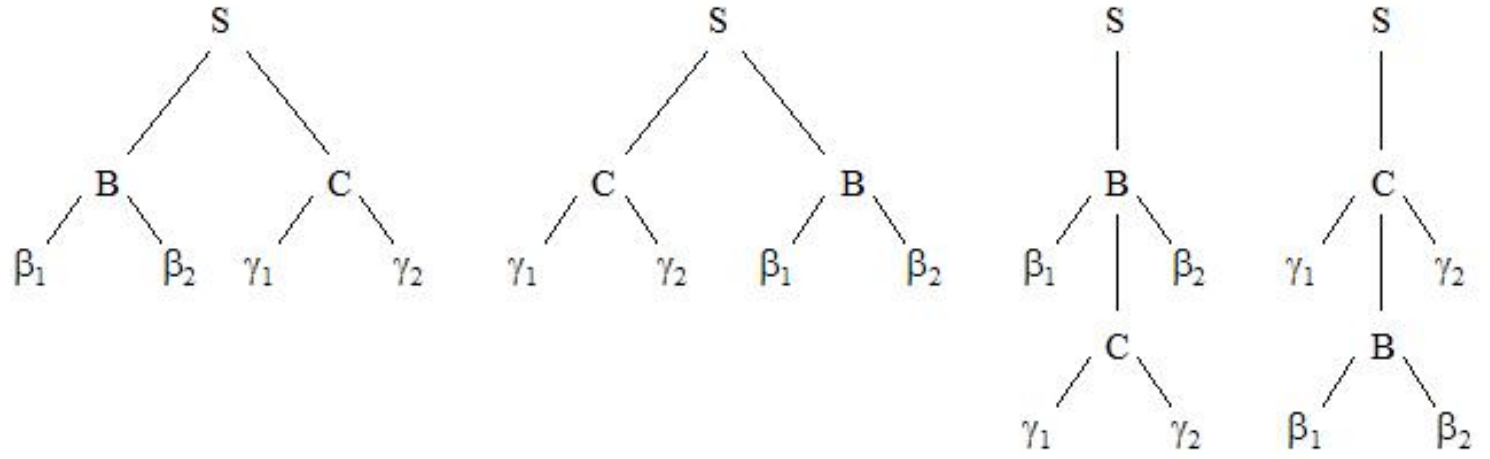
$$A \to f[B, C, D, K] \in P$$

$$f[(\beta_1, \beta_2), (\gamma_1, \gamma_2), (\delta_1, \delta_2), (\kappa_1, \kappa_2)]$$
$$= (\beta_1 a_1 \beta_2 \gamma_1 a_2,\ \gamma_2 \delta_1 \kappa_1 a_3 \kappa_2 \delta_2)$$

$$A \to (\varepsilon, \varepsilon) \in P$$

TAL $\neq$ 2-MCFL

$f[(\beta_1, \beta_2), (\gamma_1, \gamma_2)] = (\beta_1\gamma_1\beta_2\gamma_2)$ in 2-mcfg möglich.



$$\text{2-MCFL} \ni \{\underbrace{a^m b^m}_{(}\ \underbrace{c^n d^n}_{[}\ \underbrace{e^m f^m}_{)}\ \underbrace{g^n h^n}_{]} \mid m, n \in \mathbb{N}\} \notin \text{TAL}$$

Zusammenfassung

- mcfg ist eingeschränkte Form von gcfg.
- m-MCFL $\subsetneq$ (m+1)-MCFL
- CFL = 1-MCFL
- TAL $\subsetneq$ 2-MCFL

Literatur

Hiroyuki Seki, Takashi Matsumura, Mamoru Fujii und Tadao Kasami. On multiple context-free grammars. *TCS: Theoretical Computer Science*, 88, 1991

Normalform

Jede m-mcfg $\mathcal{G} = (N, O, F, P, S)$ kann in eine m-mcfg $\mathcal{G}' = (N', O', F', P', S')$ umgeformt werden, für die gilt:

- Jede terminierende Regel hat die Form $A \to (a)$, $a \in T$.
- Von $A \neq S'$ ableitbare Tupel enthalten keine leere Komponente.
- S' kommt auf keiner rechten Seite in einer Produktion in P' vor.
- Alle Komponenten bleiben erhalten.
  (z.B.: $f[(x_1, x_2), (y_1, y_2)] = (x_1 y_2)$ ist nicht möglich)
- Unterschied zu Chomsky-NF: keine Beschränkung auf zwei Argumente.

Parsing

- $T \to (t) \in P$

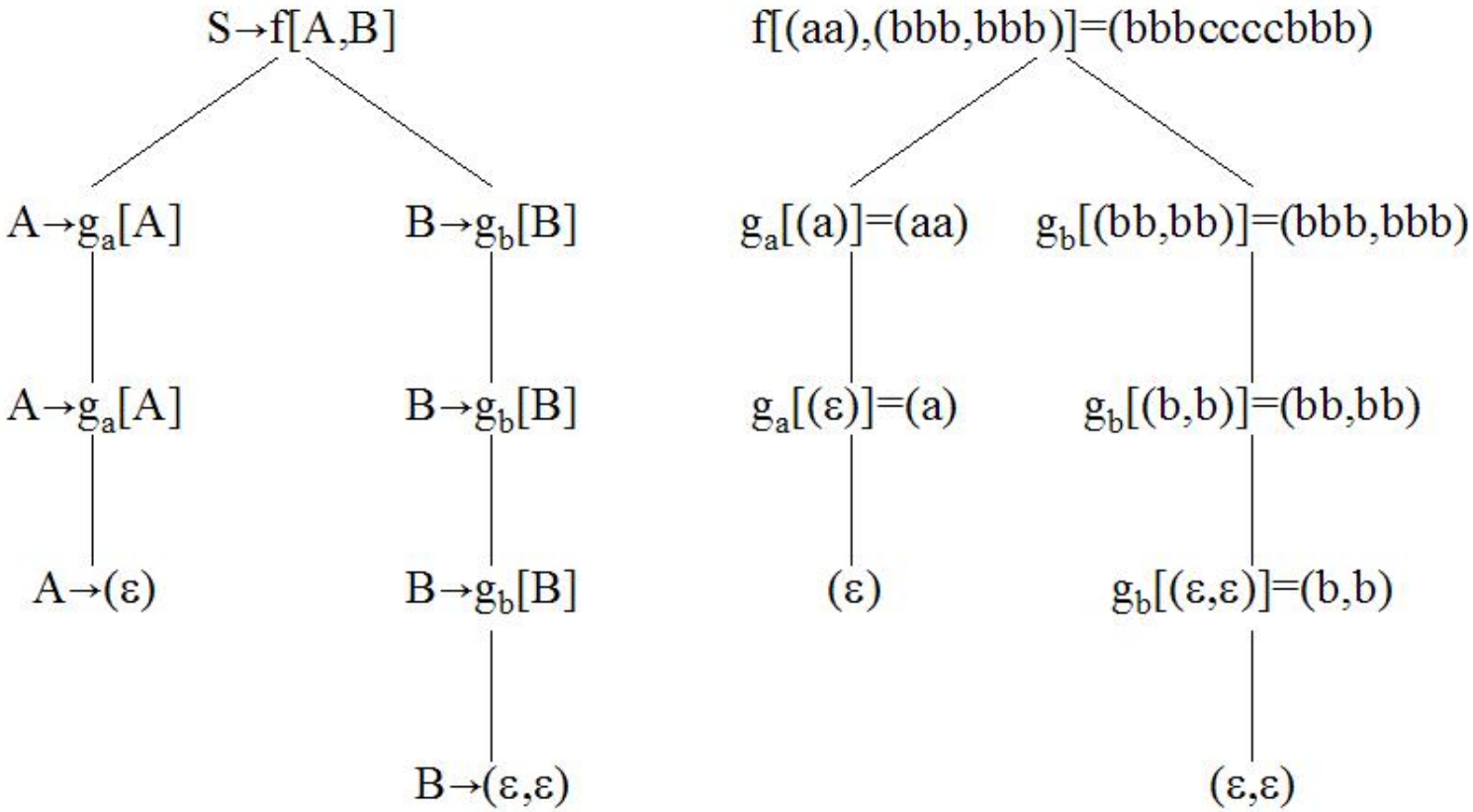
  $$\frac{}{(i, i, T)}\ \sigma_i : t$$

- $X \to f[A, B] \in P$
  $f[(a_1, a_2, a_3), (b_1, b_2)] = (a_1 b_1, a_2 b_2 a_3)$

  $$\frac{(i_1, i_2, i_3, i_4, j_4 + 1, i_6, A) \qquad (i_2 + 1, j_2, i_4 + 1, j_4, B)}{(i_1, j_2, i_3, i_6, X)}$$

- $O(n^d)$ mit $d = d(X) + d(A) + d(B) = 2 + 3 + 2 = 7$

Ableitungen



Funktionen in mcfg's

$$f : (T^*)^{d_1(f)} \times (T^*)^{d_2(f)} \times \cdots \times (T^*)^{d_{a(f)}(f)} \longrightarrow (T^*)^{r(f)}$$
$$(\overline{x_1}, \overline{x_2}, \ldots, \overline{x_{a(f)}}) \mapsto (\varphi_1, \varphi_2, \ldots, \varphi_{r(f)})$$

- $\overline{x_i} = (x_{(i,1)}, x_{(i,2)}, \ldots, x_{(i,d_i)})$
- $\varphi_h = \alpha_{(h,0)} z_{(h,1)} \alpha_{(h,2)} z_{(h,2)} \cdots z_{(h,v_h)} \alpha_{(h,v_h)}$
- $\alpha_{(h,k)} \in T^*$ und $z_{(h,k)} \in \{(x_{(i,j)} | 1 \leq i \leq d_{a(f)}, 1 \leq j \leq d_i\}$
- Jedes $x_{(i,j)}$ ist höchstens einmal im Ergebnis enthalten.