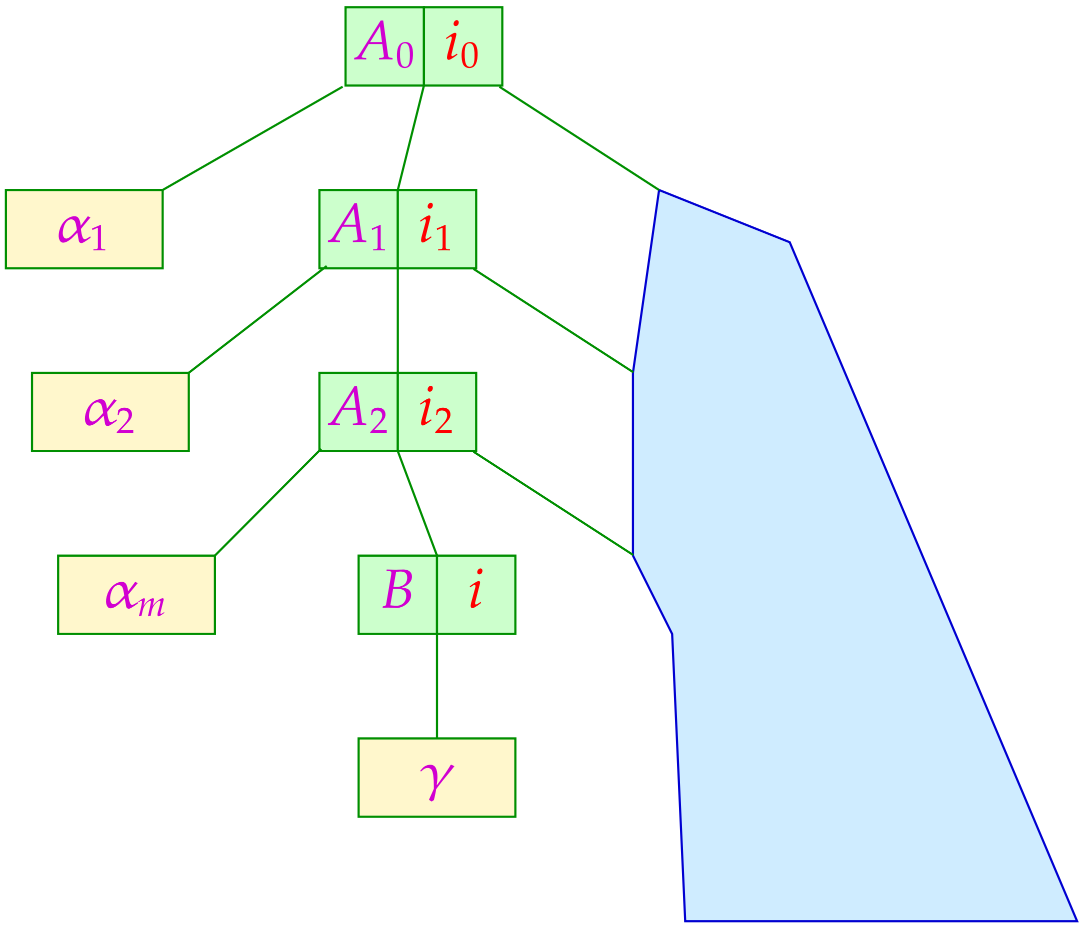
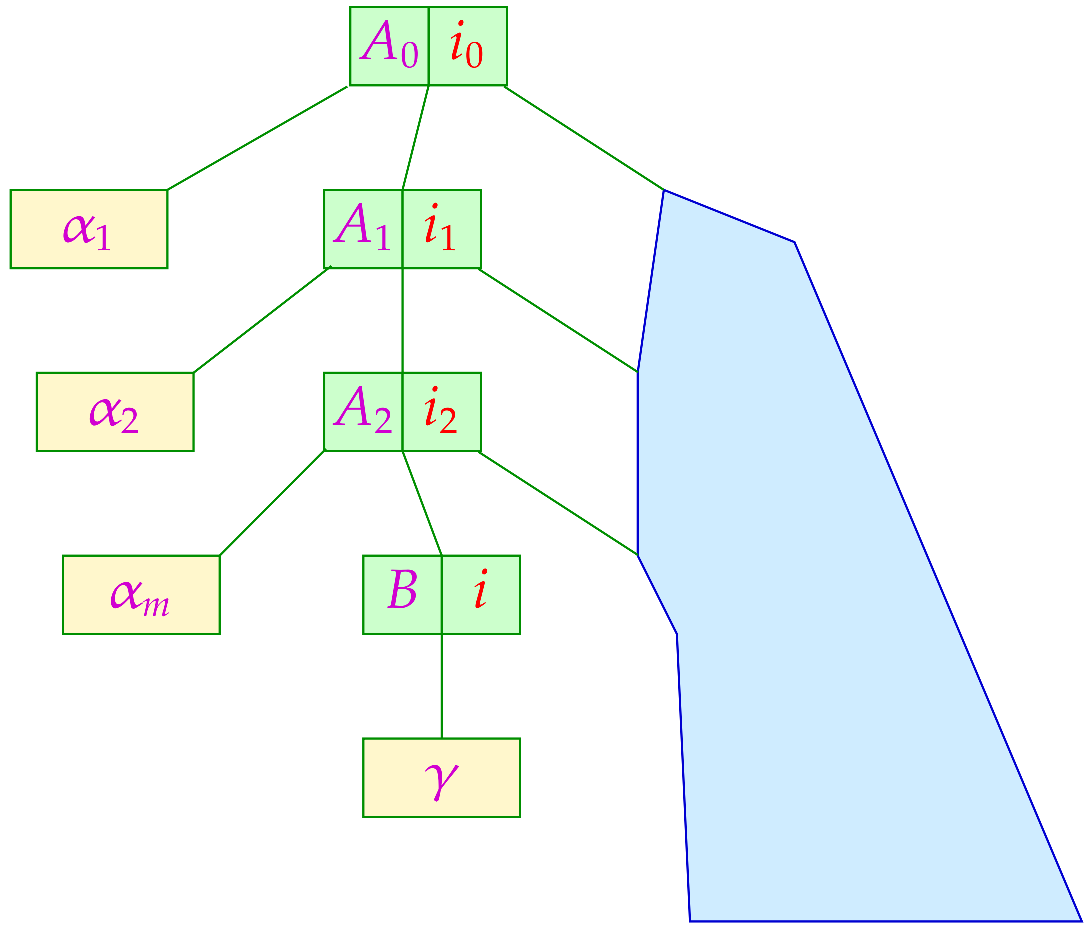


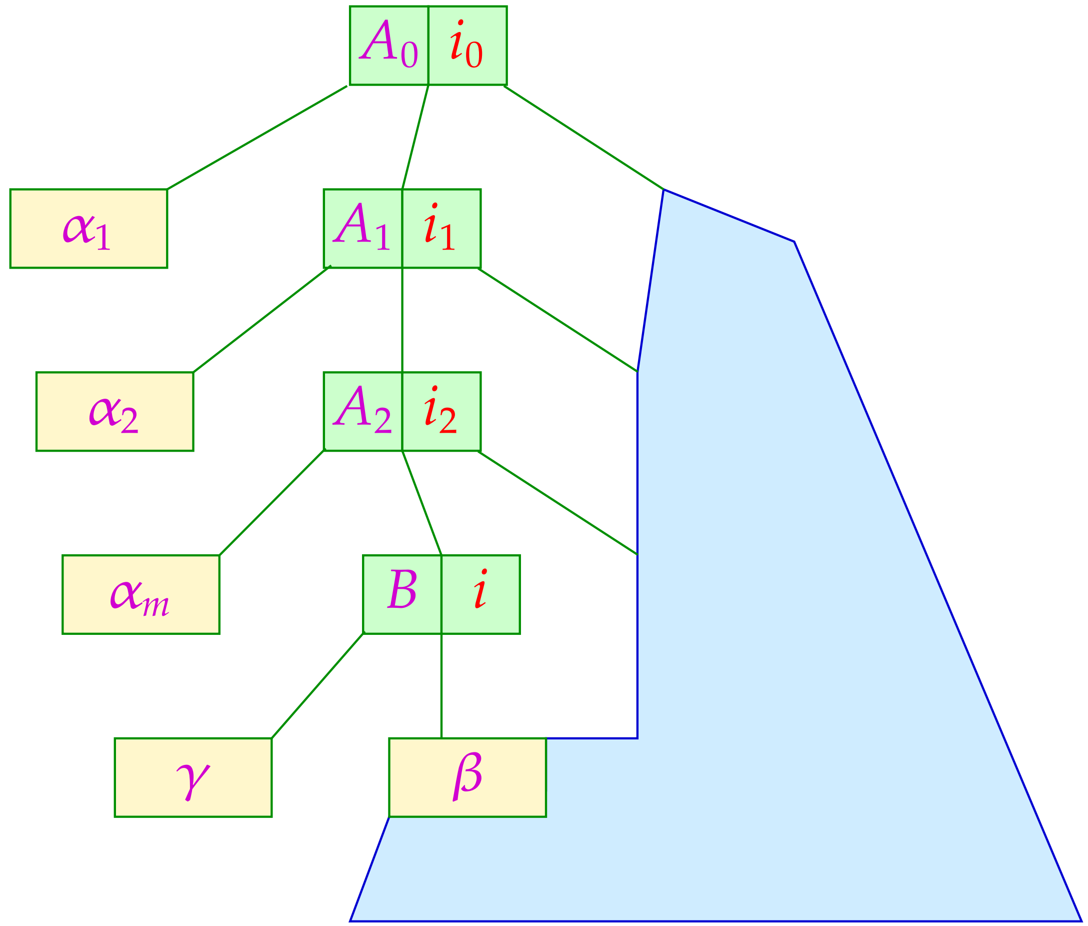
... wobei $\alpha = \alpha_1 \ \dots \ \alpha_m$:-)



... wobei $\alpha = \alpha_1 \ldots \alpha_m$:-)

Umgekehrt können wir zu jedem möglichen Wort α' die Menge aller möglicherweise später passenden Regeln ermitteln ...

Das Item $[B \to \gamma \bullet \beta]$ heißt gültig für α' gdw. $S \to_R^* \alpha\, B\, v$ mit
$\alpha' = \alpha\,\gamma$:



... wobei $\alpha = \alpha_1 \ldots \alpha_m$:-)

Beobachtung:

Die Menge der zuverlässigen Präfixe aus $(N \cup T)^*$ für (vollständige) Items kann mithilfe eines endlichen Automaten berechnet werden :-)

Zustände: Items :-)

Anfangszustand: $[S' \to \bullet\, S]$

Endzustände: $\{[B \to \gamma \bullet] \mid B \to \gamma \in P\}$

Übergänge:

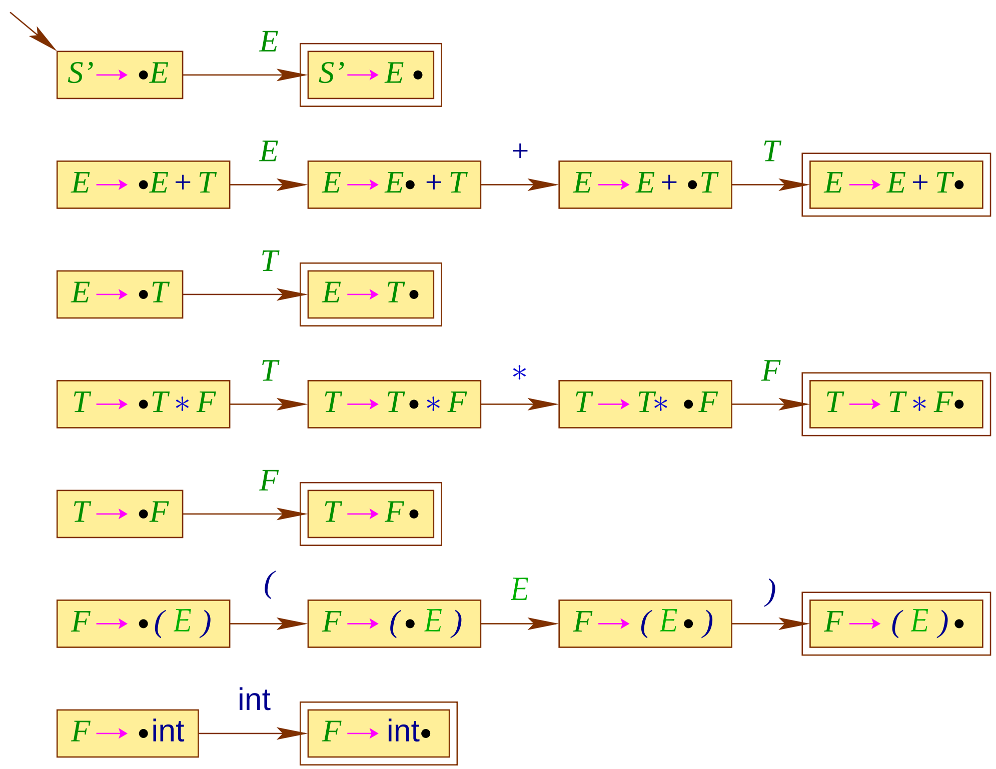
(1) $([A \to \alpha \bullet X\, \beta], X, [A \to \alpha\, X \bullet \beta])$, $X \in (N \cup T), A \to \alpha\, X\, \beta \in P$;

(2) $([A \to \alpha \bullet B\, \beta], \epsilon, [B \to \bullet\, \gamma])$, $A \to \alpha\, B\, \beta\;,\;\; B \to \gamma \in P$;

Den Automaten $c(G)$ nennen wir charakteristischen Automaten für G.

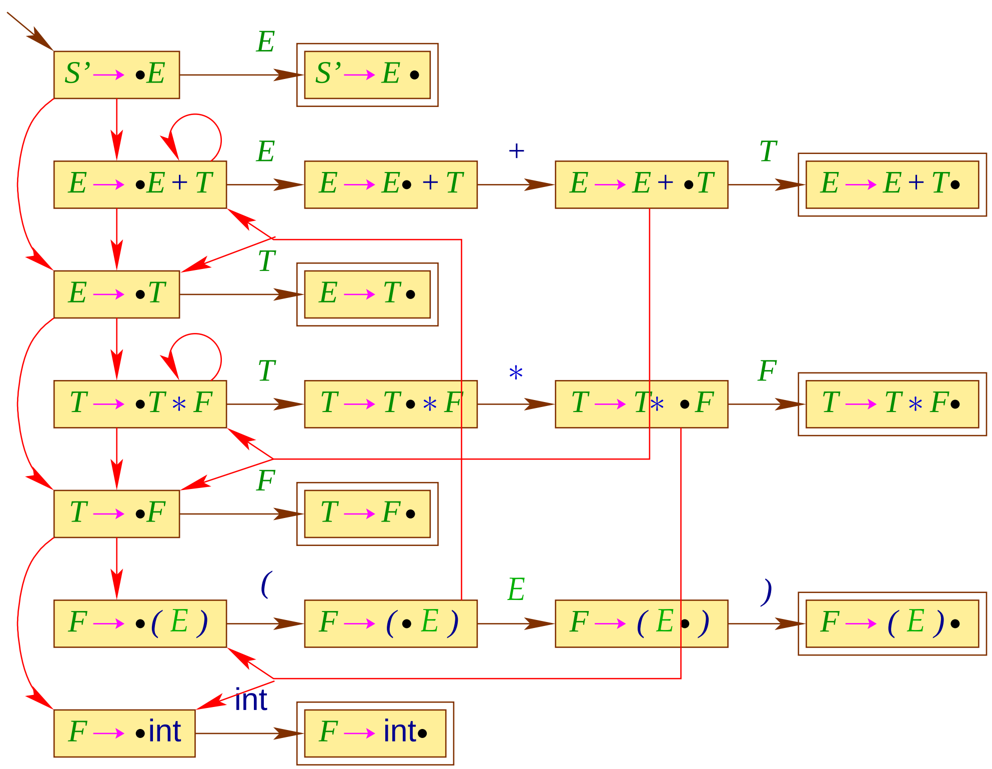
Beispiel:

$$
\begin{array}{lcll}
E & \to & E + T & \mid \; T \\
T & \to & T * F & \mid \; F \\
F & \to & (\, E \,) & \mid \; \mathsf{int}
\end{array}
$$



Beispiel:

$$
\begin{array}{lcll}
E & \to & E + T & \mid \; T \\
T & \to & T * F & \mid \; F \\
F & \to & (\, E \,) & \mid \; \mathsf{int}
\end{array}
$$

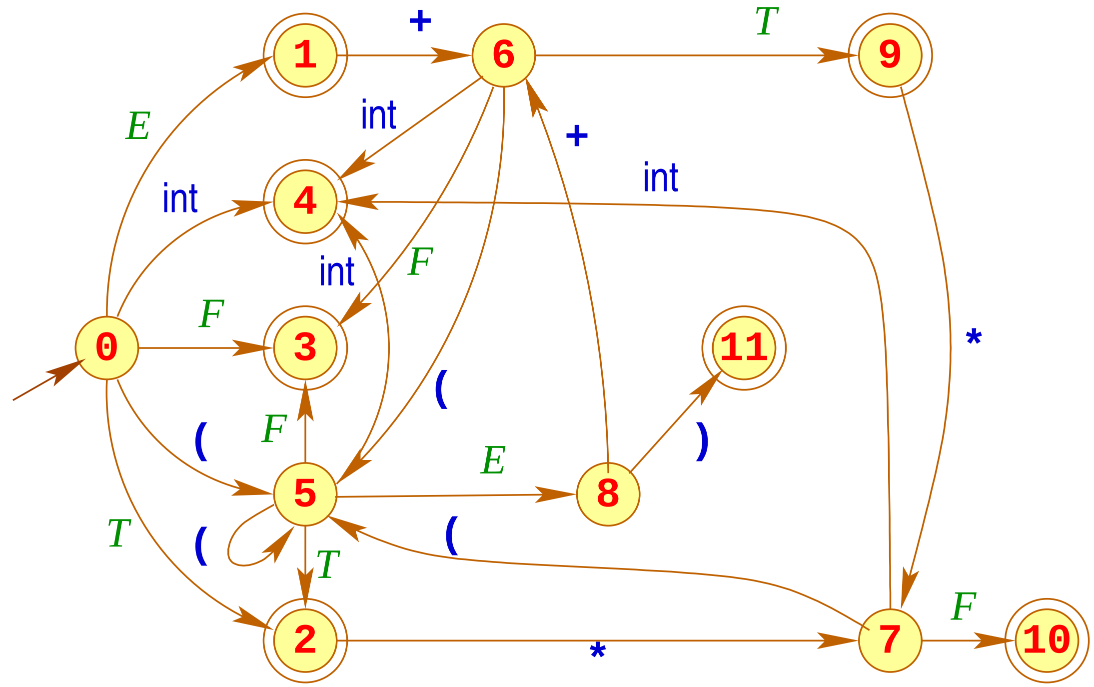


Den kanonischen $LR(0)$-Automaten $LR(G)$ erhalten wir aus $c(G)$, indem wir:

(1) nach jedem lesenden Übergang beliebig viele ϵ-Übergänge einschieben (unsere Konstruktion 1 zur Beseitigung von ϵ-Übergängen :-)

(2) die Teilmengenkonstruktion anwenden.

... im Beispiel:



Dazu konstruieren wir:

$$
\begin{array}{rcl}
q_0 & = & \{[S' \to \bullet\, E], \\
 & & [E \to \bullet\, E + T], \\
 & & [E \to \bullet\, T], \\
 & & [T \to \bullet\, T * F]\} \\
 & & [T \to \bullet\, F], \\
 & & [F \to \bullet\, (\, E\,)\,], \\
 & & [F \to \bullet\, \mathsf{int}]\}
\end{array}
$$

$$
\begin{array}{rcccl}
q_1 & = & \delta(q_0, E) & = & \{[S' \to E \bullet], \\
 & & & & [E \to E \bullet + T]\} \\
 & & & & \\
q_2 & = & \delta(q_0, T) & = & \{[E \to T \bullet], \\
 & & & & [T \to T \bullet * F]\} \\
 & & & & \\
q_3 & = & \delta(q_0, F) & = & \{[T \to F \bullet]\} \\
 & & & & \\
q_4 & = & \delta(q_0, \mathsf{int}) & = & \{[F \to \mathsf{int} \bullet]\}
\end{array}
$$

$$
\begin{aligned}
q_5 &= \delta(q_0, () = \{[F \rightarrow (\bullet E)],\\
&\qquad [E \rightarrow \bullet E + T],\\
&\qquad [E \rightarrow \bullet T],\\
&\qquad [T \rightarrow \bullet T * F],\\
&\qquad [T \rightarrow \bullet F],\\
&\qquad [F \rightarrow \bullet (E)],\\
&\qquad [F \rightarrow \bullet\, \mathsf{int}]\}
\end{aligned}
$$

$$
\begin{aligned}
q_6 &= \delta(q_1, +) = \{[E \rightarrow E + \bullet T],\\
&\qquad [T \rightarrow \bullet T * F],\\
&\qquad [T \rightarrow \bullet F],\\
&\qquad [F \rightarrow \bullet (E)],\\
&\qquad [F \rightarrow \bullet\, \mathsf{int}]\}
\end{aligned}
$$

$$
\begin{aligned}
q_7 &= \delta(q_2, *) = \{[T \rightarrow T * \bullet F],\\
&\qquad [F \rightarrow \bullet (E)],\\
&\qquad [F \rightarrow \bullet\, \mathsf{int}]\}
\end{aligned}
$$

$$
\begin{aligned}
q_8 &= \delta(q_5, E) = \{[F \rightarrow (E \bullet)]\\
&\qquad [E \rightarrow E \bullet + T]\}
\end{aligned}
$$

$$
\begin{aligned}
q_9 &= \delta(q_6, T) = \{[E \rightarrow E + T \bullet],\\
&\qquad [T \rightarrow T \bullet * F]\}
\end{aligned}
$$

$$
q_{10} = \delta(q_7, F) = \{[T \rightarrow T * F \bullet]\}
$$

$$
q_{11} = \delta(q_8,)) = \{[F \rightarrow (E) \bullet]\}
$$

Beachte:

Der kanonische $LR(0)$-Automat kann auch direkt aus der Grammatik konstruiert werden :-)

Man benötigt die Hilfsfunktion:

$$\delta^*_\epsilon(q) = q \cup \{[B \to \bullet\, \gamma] \mid \exists\, [A \to \alpha \bullet B'\, \beta'] \in q\,, \\ \beta \in (N \cup T)^* : \quad B' \to^* B\, \beta\}$$

Zustände: Mengen von Items;

Anfangszustand: $\delta^*_\epsilon\, \{[S' \to \bullet\, S]\}$

Endzustände: $\{q \mid \exists\, A \to \alpha \in P : \;\; [A \to \alpha\, \bullet] \in q\}$

Übergänge: $\delta(q, X) = \delta^*_\epsilon\, \{[A \to \alpha\, X \bullet\, \beta] \mid [A \to \alpha \bullet X\, \beta] \in q\}$

Idee zu einem Parser:

- Der Parser verwaltet ein zuverlässiges Präfix $\alpha = X_1 \ldots X_m$ auf dem Keller und benutzt $LR(G)$, um Reduktionsstellen zu entdecken.
- Er kann mit einer Regel $A \to \gamma$ reduzieren, falls $[A \to \gamma \bullet]$ für α gültig ist :-)
- Damit der Automat nicht immer wieder neu über den Kellerinhalt laufen muss, kellern wir anstelle der X_i jeweils die Zustände !!!

Achtung:

Dieser Parser ist nur dann deterministisch, wenn jeder Endzustand des kanonischen $LR(0)$-Automaten keine Konflikte enthält ...

... im Beispiel:

$$q_1 = \{[S' \to E \bullet], \; [E \to E \bullet + T]\}$$

$$q_2 = \{[E \to T \bullet], \; [T \to T \bullet * F]\}$$

$$q_9 = \{[E \to E + T \bullet], \; [T \to T \bullet * F]\}$$

$$q_3 = \{[T \to F \bullet]\}$$

$$q_{10} = \{[T \to T * F \bullet]\}$$

$$q_4 = \{[F \to \mathsf{int} \bullet]\}$$

$$q_{11} = \{[F \to (E) \bullet]\}$$

Die Endzustände q_1, q_2, q_9 enthalten mehr als ein Item :-(

Aber wir haben ja auch noch nicht Vorausschau eingesetzt :-)

Die Konstruktion des $LR(0)$-Parsers:

Zustände: $Q \cup \{f\}$ (f neu :-)

Anfangszustand: q_0

Endzustand: f

Übergänge:

Shift:	$(p, a, p\,q)$	falls	$q = \delta(p, a) \neq \emptyset$
Reduce:	$(p\,q_1 \ldots q_m, \epsilon, p\,q)$	falls	$[A \to X_1 \ldots X_m \bullet] \in q_m$,
			$q = \delta(p, A)$
Finish:	$(q_0\,p, \epsilon, f)$	falls	$[S' \to S \bullet] \in p$

wobei $LR(G) = (Q, T, \delta, q_0, F)$.

Zur Korrektheit:

Man zeigt:

Die akzeptierenden Berechnungen des $LR(0)$-Parsers stehen in eins-zu-eins Beziehung zu denen des Shift-Reduce-Parsers $M_G^{(1)}$.

Wir folgern:

$\Longrightarrow$ Die akzeptierte Sprache ist genau $\mathcal{L}(G)$:-)

$\Longrightarrow$ Die Folge der Reduktionen einer akzeptierenden Berechnung für ein Wort $w \in T$ liefert eine reverse Rechts-Ableitung von G für w :-)

Leider ist der $LR(0)$-Parser i.a. nicht-deterministisch :-(

Wir identifizieren zwei Gründe:

Reduce-Reduce-Konflikt:

$$[A \to \gamma \bullet]\,,\; [A' \to \gamma' \bullet] \;\in\; q \quad \text{mit} \quad A \neq A' \vee \gamma \neq \gamma'$$

Shift-Reduce-Konflikt:

$$[A \to \gamma \bullet]\,,\; [A' \to \alpha \bullet a\, \beta] \;\in\; q \quad \text{mit} \quad a \in T$$

für einen Zustand $q \in Q$.

Solche Zustände nennen wir ungeeignet.

Idee:

Benutze k-Vorausschau, um Konflikte zu lösen.

Wir definieren:

Die reduzierte kontextfreie Grammatik G heißt $LR(k)$-Grammatik, falls für $\mathsf{First}_k(w) = \mathsf{First}_k(x)$ aus:

$$\left.\begin{array}{lllll} S & \to_R^* & \alpha\, A\, w & \to & \alpha\, \beta\, w \\ S & \to_R^* & \alpha'\, A'\, w' & \to & \alpha\, \beta\, x \end{array}\right\} \quad \text{folgt:} \qquad \alpha = \alpha' \;\wedge\; A = A' \;\wedge\; w' = x$$

Beispiele:

(1) $S \rightarrow A \mid B \qquad A \rightarrow a\,A\,b \mid 0 \qquad B \rightarrow a\,B\,b\,b \mid 1$

... ist nicht $LL(k)$ für jedes k — aber $LR(0)$:

Sei $S \rightarrow_R^* \alpha\,X\,w \rightarrow \alpha\,\beta\,w$. Dann ist $\alpha\,\underline{\beta}$ von einer der Formen:

$$\underline{A}\;,\;\underline{B}\;,\;a^n\,\underline{a\,A\,b}\;,\;a^n\,\underline{a\,B\,b\,b}\;,\;a^n\,\underline{0}\;,\;a^n\,\underline{1} \qquad (n \geq 0)$$

Beispiele:

(1) $S \to A \mid B \qquad A \to a\,A\,b \mid 0 \qquad B \to a\,B\,b\,b \mid 1$

... ist nicht $LL(k)$ für jedes k — aber $LR(0)$:

Sei $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Dann ist $\alpha\,\underline{\beta}$ von einer der Formen:

$$\underline{A}\,,\ \underline{B}\,,\ a^n\,\underline{a\,A\,b}\,,\ a^n\,\underline{a\,B\,b\,b}\,,\ a^n\,\underline{0}\,,\ a^n\,\underline{1} \qquad (n \geq 0)$$

(2) $S \to a\,A\,c \qquad A \to A\,b\,b \mid b$

... ist ebenfalls $LR(0)$:

Sei $S \to_R^* \alpha\,X\,w \to \alpha\,\beta\,w$. Dann ist $\alpha\,\underline{\beta}$ von einer der Formen:

$$a\,\underline{b}\,,\ a\,\underline{A\,b\,b}\,,\ \underline{a\,A\,c}$$

(3) $S \rightarrow a\, A\, c \qquad A \rightarrow b\, b\, A \mid b$ ... ist nicht $LR(0)$, aber $LR(1)$:

Für $S \rightarrow^*_R \alpha\, X\, w \rightarrow \alpha\, \beta\, w$ mit $\{y\} = \text{First}_k(w)$ ist $\alpha\, \underline{\beta}\, y$ von einer der Formen:

$$a\, b^{2n}\, \underline{b}\, c \;,\; a\, b^{2n}\, \underline{b\, b\, A}\, c \;,\; \underline{a\, A\, c}$$

(3) $S \to a\, A\, c \qquad A \to b\, b\, A \;|\; b$ ... ist nicht $LR(0)$, aber $LR(1)$:

Für $S \to_R^* \alpha\, X\, w \to \alpha\, \beta\, w$ mit $\{y\} = \text{First}_k(w)$ ist $\alpha\, \underline{\beta}\, y$ von einer der Formen:

$$a\, b^{2n}\, \underline{b}\, c\ ,\ a\, b^{2n}\, \underline{b\, b\, A}\, c\ ,\ \underline{a\, A\, c}$$

(4) $S \to a\, A\, c \qquad A \to b\, A\, b \;|\; b$ ... ist nicht $LR(k)$ für jedes $k \geq 0$:

Betrachte einfach die Rechtsableitungen:

$$S \to_R^* a\, b^n\, A\, b^n\, c \to a\, b^n\, \underline{b}\, b^n\, c$$

In der Tat gilt:

Satz:

Die reduzierte Grammatik G ist genau dann $LR(0)$ wenn der kanonische $LR(0)$-Automat $LR(G)$ keine ungeeigneten Zustände enthält.

In der Tat gilt:

Satz:

Die reduzierte Grammatik G ist genau dann $LR(0)$ wenn der kanonische $LR(0)$-Automat $LR(G)$ keine ungeeigneten Zustände enthält.

Beweis:

Enthalte G einen ungeeigneten Zustand q.

Fall 1: $[A \to \gamma\bullet], [A' \to \gamma'\bullet] \in q$ mit $A \to \gamma \neq A' \to \gamma'$

Dann gibt es ein zuverlässiges Präfix $\alpha\,\gamma = \alpha'\,\gamma'$ mit

$$S \to_R^* \alpha\, A\, w \to \alpha\,\gamma\, w \quad \wedge \quad S \to_R^* \alpha'\, A'\, x \to \alpha'\,\gamma'\, x$$

$$\Longrightarrow \qquad G \text{ ist nicht } LR(0) \quad \text{:-)}$$

Fall 2: $[A \rightarrow \gamma\bullet]\,,\ [A' \rightarrow \beta \bullet a\, \beta'] \ \in\ q$

Dann gibt es ein zuverlässiges Präfix $\alpha\,\gamma = \alpha'\,\beta$ mit

$$S \rightarrow_R^* \alpha\, A\, w \rightarrow \alpha\,\gamma\, w \quad \wedge \quad S \rightarrow_R^* \alpha'\, A'\, x \rightarrow \alpha'\,\beta\, a\, \beta'\, x$$

Ist $\beta' \in T^*$, dann ist G nicht $LR(0)$:-)

Andernfalls $\beta' \rightarrow_R^* v_1\, X\, v_2 \rightarrow v_1\, u\, v_2$. Damit erhalten wir:

$$S \rightarrow_R^* \alpha'\,\beta\, a\, v_1\, X\, v_2\, x \rightarrow \alpha'\,\beta\, a\, v_1\, u\, v_2\, x$$

$\Longrightarrow$ G ist nicht $LR(0)$:-)

Fall 2: $[A \to \gamma \bullet]\,,\ [A' \to \beta \bullet a\, \beta'] \ \in\ q$

Dann gibt es ein zuverlässiges Präfix $\alpha\,\gamma = \alpha'\,\beta$ mit

$$S \to_R^* \alpha\, A\, w \to \alpha\,\gamma\, w \quad \wedge \quad S \to_R^* \alpha'\, A'\, x \to \alpha'\,\beta\, a\, \beta'\, x$$

Ist $\beta' \in T^*$, dann ist G nicht $LR(0)$:-)

Andernfalls $\beta' \to_R^* v_1\, X\, v_2 \to v_1\, u\, v_2$. Damit erhalten wir:

$$S \to_R^* \alpha'\,\beta\, a\, v_1\, X\, v_2\, x \to \alpha'\,\beta\, a\, v_1\, u\, v_2\, x$$

$\Longrightarrow$ G ist nicht $LR(0)$:-)

Enthalte $LR(G)$ keine ungeeigneten Zustände. Betrachte:

$$S \to_R^* \alpha\, A\, w \to \alpha\,\gamma\, w \qquad\qquad S \to_R^* \alpha'\, A'\, w' \to \alpha'\,\gamma'\, x$$

Sei $\delta(q_0, \alpha\,\gamma) = q$. Insbesondere ist $[A \to \gamma \bullet] \in q$.

Annahme: $(\alpha, A, w') \neq (\alpha', A', x)$.

Fall 1: $w' = x$. Dann muss q $[A' \to \gamma' \bullet]$ enthalten :-)

Fall 2: $w' \neq x$. Weitere Fallunterscheidung :-))