unwanted in CFG:

– variables not used in successful derivations $S \Rightarrow^* x \in \Sigma^*$

– $A \to \Lambda$ A variable Λ-productions

– $A \to B$ A, B variables unit productions [chain rules]

unwanted in CFG:
– variables not used in successful derivations $S \Rightarrow^* x \in \Sigma^*$
– $A \to \Lambda$ A variable Λ-productions
– $A \to B$ A, B variables unit productions [chain rules]

restricted CFG, with 'nice' from
Chomsky normalform $A \to BC$, $A \to \sigma$
Greibach normalform (☒) $A \to \sigma B_1 \dots B_k$

CFG $G = (V, \Sigma, S, P)$

Definition

variable A is *live* if $A \Rightarrow^* x$ for some $x \in \Sigma^*$.

variable A is *reachable* if $S \Rightarrow^* \alpha A \beta$ for some $\alpha, \beta \in (\Sigma \cup V^*)$.

variable A is *useful* if there is a derivation of the form $S \Rightarrow^* \alpha A \beta \Rightarrow^* x$ for some string $x \in \Sigma^*$.

useful implies live and reachable.
For $S \to AB \mid b$ and $A \to a$, variable A is live and reachable, not useful.
[M] Exercise 4.51, 4.52, 4.53

Live variables

Construction

- $N_0 = \varnothing$
- $N_{i+1} = N_i \cup \{\, A \in V \mid A \to \alpha \text{ in } P, \text{ with } \alpha \in (N_i \cup \Sigma)^* \,\}$

$N_1 = \{\, A \in V \mid A \to x \text{ in } P, \text{ with } x \in \Sigma^* \,\}$
$N_0 \subseteq N_1 \subseteq N_2 \subseteq \cdots \subseteq V$
there exists a k such that $N_k = N_{k+1}$
A is live iff $A \in \bigcup_{i \geqslant 0} N_i = N_k$
(minimal) depth of derivation tree $A \Rightarrow^* x$

Live variables

Construction

– $N_0 = \varnothing$
– $N_{i+1} = N_i \cup \{ A \in V \mid A \to \alpha \text{ in } P, \text{ with } \alpha \in (N_i \cup \Sigma)^* \}$

Exercise 4.53(c_i).

$S \to ABC \mid BaB \qquad A \to aA \mid BaC \mid aaa$
$B \to bBb \mid a \qquad C \to CA \mid AC$

Reachable variables

Construction

- $N_0 = \{S\}$
- $N_{i+1} = N_i \cup \{\, A \in V \mid B \to \alpha_1 A \alpha_2 \text{ in } P, \text{ with } B \in N_i \,\}$

$N_0 \subseteq N_1 \subseteq N_2 \subseteq \cdots \subseteq V$
there exists a k such that $N_k = N_{k+1}$
A is reachable iff $A \in \bigcup_{i \geqslant 0} N_i = N_k$
(minimal) length of derivation $S \Rightarrow^* \alpha A \beta$

Reachable variables

Construction

- $N_0 = \{S\}$
- $N_{i+1} = N_i \cup \{\, A \in V \mid B \to \alpha_1 A \alpha_2 \text{ in } P,\ \text{with } B \in N_i \,\}$

$N_0 \subseteq N_1 \subseteq N_2 \subseteq \cdots \subseteq V$
there exists a k such that $N_k = N_{k+1}$
A is reachable iff $A \in \bigcup_{i \geqslant 0} N_i = N_k$
(minimal) length of derivation $S \Rightarrow^* \alpha A \beta$

- remove all non-live variables (and productions that contain them)
- remove all unreachable variables (and productions)

then all variables are useful

does not work the other way around . . .

Reachable variables

Construction

- $N_0 = \{S\}$
- $N_{i+1} = N_i \cup \{\, A \in V \mid B \to \alpha_1 A \alpha_2 \text{ in } P, \text{ with } B \in N_i \,\}$

Exercise 4.53(c_i)., ctd

$S \to BaB \qquad A \to aA \mid aaa \qquad B \to bBb \mid a$

– remove all non-live variables (and productions that contain them)
– remove all unreachable variables (and productions)

then all variables are useful

does not work the other way around . . .

Exercise 4.53(c_i)., revisited

$S \to ABC \mid BaB$ $\qquad A \to aA \mid BaC \mid aaa$
$B \to bBb \mid a$ $\qquad C \to CA \mid AC$

Idea:

Example

$A \to BCDCB$
$B \to b \mid \Lambda$
$C \to c \mid \Lambda$
$D \to d$

Definition

variable A is *nullable* iff $A \Rightarrow^* \Lambda$

Theorem

– *if $A \to \Lambda$ then A is nullable*

– *if $A \to B_1 B_2 \dots B_k$ and all B_i are nullable, then A is nullable*

[M] Def 4.26 / Exercise 4.48

Construction

– $N_0 = \varnothing$

– $N_{i+1} = N_i \cup \{ A \in V \mid A \to \alpha \text{ in } P, \text{ with } \alpha \in N_i^* \}$

$N_1 = \{ A \in V \mid A \to \Lambda \text{ in } P \}$

$N_0 \subseteq N_1 \subseteq N_2 \subseteq \cdots \subseteq V$

there exists a k such that $N_k = N_{k+1}$

A is nullable iff $A \in \bigcup_{i \geqslant 0} N_i = N_k$

Construction

– identify nullable variables
– for every production $A \to \alpha$ add $A \to \beta$,
  where β is obtained from α by removing one or more nullable variables
– remove all Λ-productions (and all productions $A \to A$)

Grammar for $\{ a^i b^j c^k \mid i = j \text{ or } i = k \}$

$$
\begin{aligned}
S &\to TU \mid V \\
T &\to aTb \mid \Lambda \\
U &\to cU \mid \Lambda \\
V &\to aVc \mid W \\
W &\to bW \mid \Lambda
\end{aligned}
$$

Grammar for $\{ a^i b^j c^k \mid i = j \text{ or } i = k \}$

$S \to TU \mid V$
$T \to aTb \mid \Lambda$
$U \to cU \mid \Lambda$
$V \to aVc \mid W$
$W \to bW \mid \Lambda$

$N_1 = \{T, U, W\}$, variables with Λ at right-hand side productions

$N_2 = \{T, U, W\} \cup \{S, V\}$, variables with $\{T, U, W\}^*$ at rhs productions

$N_3 = N_2 = \{T, U, W, S, V\}$, all productions found, no new

Example nullable, ctd

add all productions, where (any number of) nullable variables are removed...

$$
\begin{aligned}
S &\to TU \mid V \\
T &\to aTb \mid \Lambda \\
U &\to cU \mid \Lambda \\
V &\to aVc \mid W \\
W &\to bW \mid \Lambda
\end{aligned}
$$

[M] Ex. 4.31

add all productions, where (any number of) nullable variables are removed

$S \to TU \mid V$	$S \to T \mid U \mid \Lambda$
$T \to aTb \mid \Lambda$	$T \to ab$
$U \to cU \mid \Lambda$	$U \to c$
$V \to aVc \mid W$	$V \to ac \mid \Lambda$
$W \to bW \mid \Lambda$	$W \to b$

remove all Λ-productions...

[M] Ex. 4.31

add all productions, where (any number of) nullable variables are removed

$S \to TU \mid V$	$S \to T \mid U \mid \Lambda$
$T \to aTb \mid \Lambda$	$T \to ab$
$U \to cU \mid \Lambda$	$U \to c$
$V \to aVc \mid W$	$V \to ac \mid \Lambda$
$W \to bW \mid \Lambda$	$W \to b$

remove all Λ-productions

$S \to TU \mid V \mid T \mid U$
$T \to aTb \mid ab$
$U \to cU \mid c$
$V \to aVc \mid W \mid ac$
$W \to bW \mid b$

[M] Ex. 4.31

Theorem

For every CFG G *there is* CFG G_1 *without* Λ*-productions such that* $L(G_1) = L(G) - \{\Lambda\}$.

Proof...

[M] Thm 4.27

Assume Λ-productions have been removed

Variable B is *A-derivable*, if
– $B \neq A$, and
– $A \Rightarrow^* B$ (using only unit productions)

Construction

– $N_1 = \{ B \in V \mid B \neq A \text{ and } A \to B \text{ in } P \}$
– $N_{i+1} = N_i \cup \{ C \in V \mid C \neq A \text{ and } B \to C \text{ in } P, \text{ with } B \in N_i \}$

$N_1 \subseteq N_2 \subseteq \cdots \subseteq V$
there exists a k such that $N_k = N_{k+1}$
B is A-derivable iff $B \in \bigcup_{i \geqslant 0} N_i = N_k$

Construction

- for each $A \in V$, identify A-derivable variables
- for every pair (A, B) where B is A-derivable,
  and every production $B \to \alpha$ add $A \to \alpha$
- remove all unit productions

Grammar for $\{ a^i b^j c^k \mid i = j \text{ or } i = k \}$

$$
\begin{aligned}
S &\to TU \mid V \mid T \mid U \\
T &\to aTb \mid ab \\
U &\to cU \mid c \\
V &\to aVc \mid W \mid ac \\
W &\to bW \mid b
\end{aligned}
$$

Example unit productions

$S \to TU \mid V \mid T \mid U$
$T \to aTb \mid ab$
$U \to cU \mid c$
$V \to aVc \mid W \mid ac$
$W \to bW \mid b$

S-derivable: $\{V, T, U\}, \{V, T, U, W\}$ V-derivable: $\{W\}$

New productions:

$S \to aTb \mid ab \quad S \to cU \mid c \quad S \to aVc \mid W \mid ac \quad S \to bW \mid b$
$V \to bW \mid b$

Remove unit productions:

$S \to TU \mid aTb \mid ab \mid cU \mid c \mid aVc \mid ac \mid bW \mid b$
$T \to aTb \mid ab$
$U \to cU \mid c$
$V \to aVc \mid ac \mid bW \mid b$
$W \to bW \mid b$

Definition

CFG in *Chomsky normal form*

productions are of the form

– $A \to BC$ variables A, B, C

– $A \to \sigma$ variable A, terminal σ

Theorem

For every CFG G *there is* CFG G_1 *in CNF such that* $L(G_1) = L(G) - \{\Lambda\}$.

[M] Def 4.29, Thm 4.30

Construction

1. remove Λ-productions
2. remove unit productions
3. introduce variables for terminals $\quad X_\sigma \to \sigma$
4. split long productions

$$A \to aBabA$$

is replaced by

$$X_a \to a \qquad X_b \to b \qquad A \to X_a B X_a X_b A$$

$$A \to ACBA$$

is replaced by

$$A \to AY_1 \qquad Y_1 \to CY_2 \qquad Y_2 \to BA$$

Grammar for $\{\ a^i b^j c^k \mid i = j \text{ or } i = k\ \}$

$S \to TU \mid V$

$T \to aTb \mid \Lambda \qquad U \to cU \mid \Lambda$

$V \to aVc \mid W \qquad W \to bW \mid \Lambda$

After removing Λ-productions and unit productions, we obtain (see before)

$S \to TU \mid aTb \mid ab \mid cU \mid c \mid aVc \mid ac \mid bW \mid b$

$T \to aTb \mid ab \qquad U \to cU \mid c$

$V \to aVc \mid ac \mid bW \mid b \qquad W \to bW \mid b$

Now introduce productions for the terminals. . .

Grammar for $\{ a^i b^j c^k \mid i = j \text{ or } i = k \}$

$S \to TU \mid V$

$T \to aTb \mid \Lambda$ $\qquad U \to cU \mid \Lambda$

$V \to aVc \mid W$ $\qquad W \to bW \mid \Lambda$

After removing Λ-productions and unit productions, we obtain (see before)

$S \to TU \mid aTb \mid ab \mid cU \mid c \mid aVc \mid ac \mid bW \mid b$

$T \to aTb \mid ab$ $\qquad U \to cU \mid c$

$V \to aVc \mid ac \mid bW \mid b$ $\qquad W \to bW \mid b$

Now introduce productions for the terminals:

$X_a \to a$ $\qquad X_b \to b$ $\qquad X_c \to c$

$S \to TU \mid X_a T X_b \mid X_a X_b \mid X_c U \mid c \mid X_a V X_c \mid X_a X_c \mid X_b W \mid b$

$T \to X_a T X_b \mid X_a X_b$

$U \to X_c U \mid c$

$V \to X_a V X_c \mid X_a X_c \mid X_b W \mid b$

$W \to X_b W \mid X_b$

Only a few productions that are too long:

$S \to X_a T X_b \mid X_a V X_c$

$T \to X_a T X_b$

$V \to X_a V X_c$

Split these long productions...

Only a few productions that are too long:

$S \to X_a T X_b \mid X_a V X_c$
$T \to X_a T X_b$
$V \to X_a V X_c$

Split these long productions:

$S \to X_a Y_1 \mid X_a Y_2$
$Y_1 \to T X_b \qquad Y_2 \to V X_c$
$T \to X_a Y_1$
$V \to X_a Y_2$

Note that we can reuse Y_1, Y_2 for two productions

Chomsky NF for pumping lemma (later)

Definition

Regular grammar:
productions are of the form
– $A \to \sigma B$ variables A, B, terminal σ
– $A \to \Lambda$ variable A

[M] Def 4.13

Definition

CFG in *Chomsky normal form*

productions are of the form

– $A \to BC$ variables A, B, C

– $A \to \sigma$ variable A, terminal σ

[M] Def 4.29

$\text{even}(L) = \{ \ w \in L \mid |w| \text{ even } \}$

idea: new variables for even/odd length strings
Chomsky normalform to reduce number of possibilities.

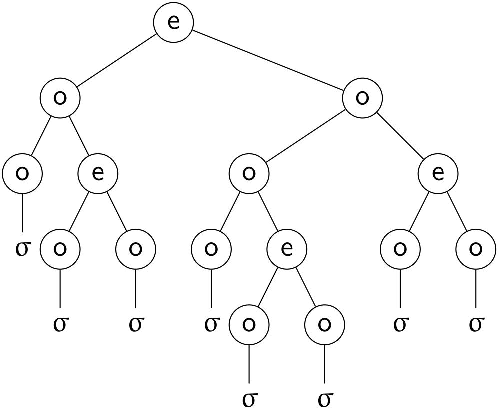
grammar $G = (V, \Sigma, P, S)$ for L, in ChNF
new grammar $G = (V', \Sigma, P', S')$ for $\text{even}(L)$
variables: $V' = \{X_e, X_o \mid X \in V\}$
axiom: $S' = S_e$
productions: – for every $A \to BC$ in P we have in P':

$$A_e \to B_e C_e \mid B_o C_o \qquad\qquad A_o \to B_e C_o \mid B_o C_e$$

– for every $A \to \sigma$ in P we have in P': $A_o \to \sigma$

We consider closure properties: given an operation X show that whenever L is regular/context-free, then also $X(L)$ is regular/context-free.

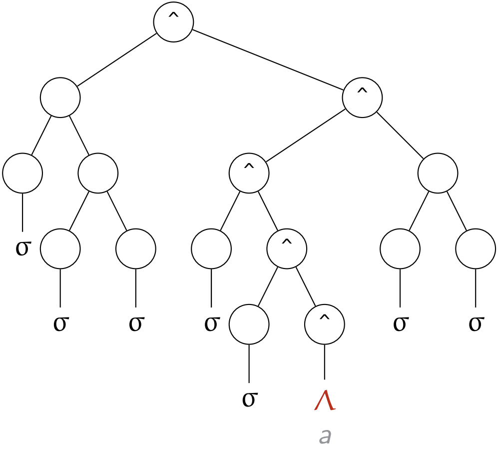
This is done as follows: if L is regular/context-free, then we know there is a regular/context-free grammar G for L, and we show how to construct a new grammar G' (of the same type) for $X(L)$, in terms of the original grammar G.


e
o
o
o
e
o
e
σ
o
o
o
e
o
o
σ
σ
σ
o
o
σ
σ
σ
σ

$L \subseteq \{a, b\}^*$, $\quad$ $\text{chop}(L) = \{\, xy \mid xay \in L\}$ $\quad$ remove some a in each string

idea: new variables for the task of removing letter a

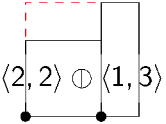
grammar $G = (V, \{a, b\}, P, S)$ for L, in ChNF
new grammar $G = (V', \{a, b\}, P', S')$ for $\text{chop}(L)$
variables: $V' = V \cup \{\hat{X} \mid X \in V\}$
axiom: $S' = \hat{S}$
productions: keep all productions from P, and
– for every $A \to BC$ add $\hat{A} \to \hat{B}C \mid B\hat{C}$
– for every $A \to a$ add $\hat{A} \to \Lambda$

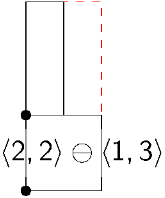

^
^
^
^
^
σ
σ
σ
σ
σ
σ
σ
Λ
a

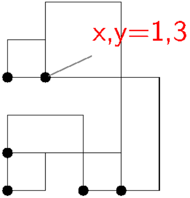
$$E \to E + T \mid T$$
$$T \to T * F \mid F$$
$$F \to (\ E\ ) \mid int$$

$E \to E_1 + T_1$	$E.\mathrm{val} = \mathrm{E_1.val} + \mathrm{T_1.val}$
$E \to T_1$	$E.\mathrm{val} = \mathrm{T_1.val}$
$T \to T_1 * F_1$	$T.\mathrm{val} = \mathrm{T_1.val} \cdot \mathrm{F_1.val}$
$T \to F_1$	$T.\mathrm{val} = \mathrm{F_1.val}$
$F \to (\ E_1\ )$	$F.\mathrm{val} = \mathrm{E_1.val}$
$F \to int$	$F.\mathrm{val} = \mathrm{IntVal}(\ \mathrm{int}\ )$

D.E. Knuth. Semantics of Context-Free Languages.
Math. Systems Theory (1968) 127–145 doi:10.1007/BF01692511



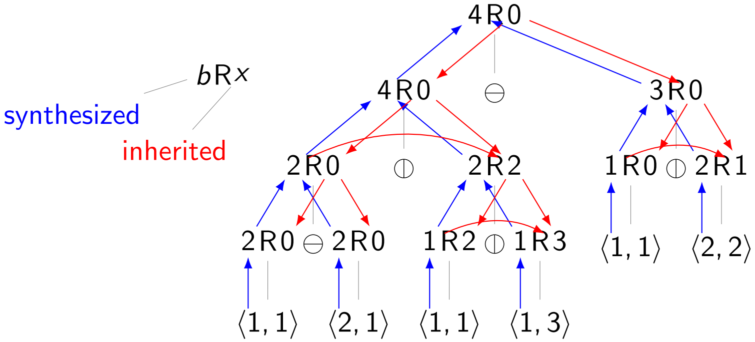




$$(\ ((\langle 1,1\rangle \ominus \langle 2,1\rangle) \obar ((\langle 1,1\rangle \obar \langle 1,3\rangle)\ ) \ominus ((\langle 1,1\rangle \obar \langle 2,2\rangle)$$

production	semantic rule
$R \to \langle E_1, E_2\rangle$	$R.b = E_1.\text{val} \quad \text{R.h} = \text{E}_2.\text{val}$
$R \to (R_1 \obar R_2)$	$R.b = R_1.b + R_2.b$
	$R.h = \max\{R_1.h, R_2.h\}$
	$R_1.x = R.x \quad R_2.x = R.x + R_1.b$
	$R_1.y = R.y \quad R_2.y = R.y$
$R \to (R_1 \ominus R_2)$	$R.b = \max\{R_1.b, R_2.b\}$
	$R.h = R_1.h + R_2.h$
	$R_1.x = R.x \quad R_2.x = R.x$
	$R_1.y = R.y \quad R_2.y = R.y + R_1.h$

$$
\begin{array}{ll}
R \to (R_1 \oplus R_2) & R.b = R_1.b + R_2.b \\
 & R_1.x = R.x \quad R_2.x = R.x + R_1.b \\
R \to (R_1 \ominus R_2) & R.b = \max\{R_1.b, R_2.b\} \\
 & R_1.x = R.x \quad R_2.x = R.x
\end{array}
$$



Section 5

Pushdown Automata

just like FA, PDA accepts strings / language
just like FA, PDA has states
just like FA, PDA reads input one letter at a time
unlike FA, PDA has auxiliary memory: a stack
unlike FA, by default PDA is nondeterministic
unlike FA, by default Λ-transitions are allowed in PDA

Why a stack?

$$AnBn = \{a^i b^i \mid i \geqslant 0\}$$

with $x = aaabbb$

$$SimplePal = \{xcx^r \mid x \in \{a, b\}^*\}$$

with $x = aabcbaa$

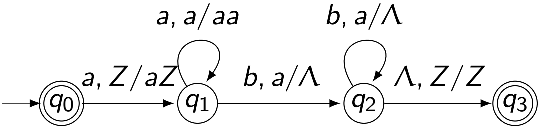
Stack in PDA contains symbols from certain alphabet.
Usual stack operations: pop, top, push
Extra possiblity: replace top element X by string α

$\alpha = \Lambda$	pop
$\alpha = X$	top
$\alpha = YX$	push
$\alpha = \beta X$	push*
$\alpha = \dots$	

Top element X is required to do a move!

$AnBn = \{ a^n b^n \mid n \geqslant 0 \}$

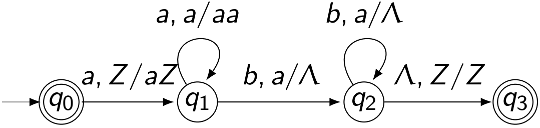
initial q_0, Z, accept $A = \{q_0, q_3\}$



[M] E 5.3

$AnBn = \{ \ a^n b^n \mid n \geqslant 0 \ \}$

initial q_0, Z, accept $A = \{q_0, q_3\}$



[M] E 5.3