

Grammars

Grammars

Grammars express languages

Example: the English language

$\langle \textit{sentence} \rangle \rightarrow \langle \textit{noun_phrase} \rangle \langle \textit{predicate} \rangle$

$\langle \textit{noun_phrase} \rangle \rightarrow \langle \textit{article} \rangle \langle \textit{noun} \rangle$

$\langle \textit{predicate} \rangle \rightarrow \langle \textit{verb} \rangle$

$\langle \textit{article} \rangle \rightarrow a$

$\langle \textit{article} \rangle \rightarrow \textit{the}$

$\langle \textit{noun} \rangle \rightarrow \textit{cat}$

$\langle \textit{noun} \rangle \rightarrow \textit{dog}$

$\langle \textit{verb} \rangle \rightarrow \textit{runs}$

$\langle \textit{verb} \rangle \rightarrow \textit{walks}$

A derivation of "the dog walks":

$\langle \textit{sentence} \rangle \Rightarrow \langle \textit{noun_phrase} \rangle \langle \textit{predicate} \rangle$
 $\Rightarrow \langle \textit{noun_phrase} \rangle \langle \textit{verb} \rangle$
 $\Rightarrow \langle \textit{article} \rangle \langle \textit{noun} \rangle \langle \textit{verb} \rangle$
 $\Rightarrow \textit{the} \langle \textit{noun} \rangle \langle \textit{verb} \rangle$
 $\Rightarrow \textit{the dog} \langle \textit{verb} \rangle$
 $\Rightarrow \textit{the dog walks}$

A derivation of "a cat runs":

$\langle \textit{sentence} \rangle \Rightarrow \langle \textit{noun_phrase} \rangle \langle \textit{predicate} \rangle$
 $\Rightarrow \langle \textit{noun_phrase} \rangle \langle \textit{verb} \rangle$
 $\Rightarrow \langle \textit{article} \rangle \langle \textit{noun} \rangle \langle \textit{verb} \rangle$
 $\Rightarrow a \langle \textit{noun} \rangle \langle \textit{verb} \rangle$
 $\Rightarrow a \textit{ cat} \langle \textit{verb} \rangle$
 $\Rightarrow a \textit{ cat runs}$

Language of the grammar:

$L = \{$ "a cat runs",
"a cat walks",
"the cat runs",
"the cat walks",
"a dog runs",
"a dog walks",
"the dog runs",
"the dog walks" $\}$

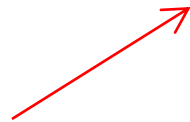
Notation

Production Rules



$\langle noun \rangle \rightarrow cat$

$\langle noun \rangle \rightarrow dog$



Variable

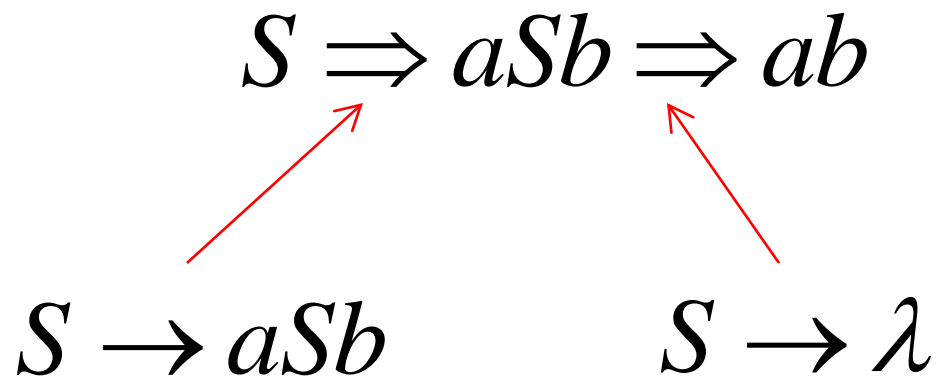
Terminal

Another Example

Grammar: $S \rightarrow aSb$

$S \rightarrow \lambda$

Derivation of sentence ab :



Grammar: $S \rightarrow aSb$

$S \rightarrow \lambda$

Derivation of sentence $aabb$:

$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aabb$



$S \rightarrow aSb$

$S \rightarrow \lambda$

Other derivations:

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbbb$$

$$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \\ \Rightarrow aaaaSbbbb \Rightarrow aaabbbbb$$

Language of the grammar

$$S \rightarrow aSb$$

$$S \rightarrow \lambda$$

$$L = \{a^n b^n : n \geq 0\}$$

More Notation

Grammar $G = (V, T, S, P)$

V : Set of variables

T : Set of terminal symbols

S : Start variable

P : Set of Production rules

Example

Grammar G : $S \rightarrow aSb$
 $S \rightarrow \lambda$

$$G = (V, T, S, P)$$

$$V = \{S\}$$

$$T = \{a, b\}$$

$$P = \{S \rightarrow aSb, S \rightarrow \lambda\}$$

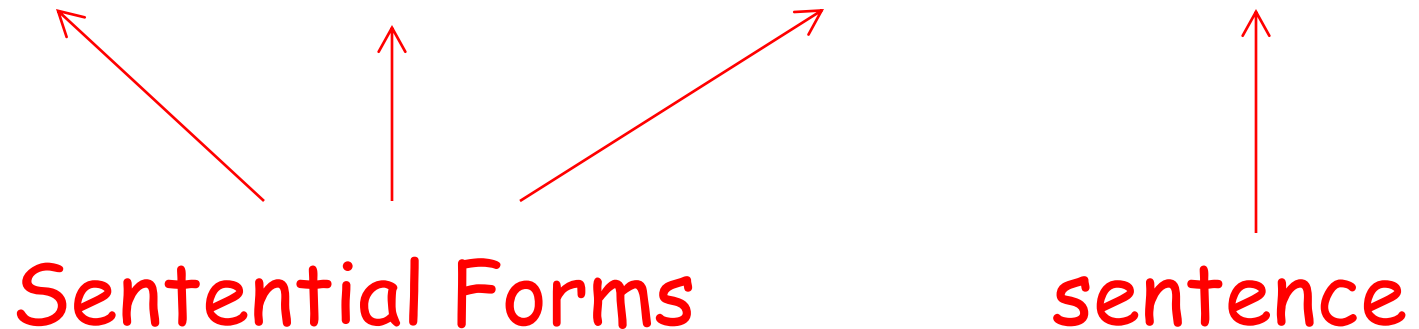
More Notation

Sentential Form:

A sentence that contains variables and terminals

Example:

$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbbb$



We write: $S \stackrel{*}{\Rightarrow} aaabbb$

Instead of:

$S \Rightarrow aSb \Rightarrow aaSbb \Rightarrow aaaSbbb \Rightarrow aaabbb$

In general we write: $w_1 \stackrel{*}{\Rightarrow} w_n$

If: $w_1 \Rightarrow w_2 \Rightarrow w_3 \Rightarrow \cdots \Rightarrow w_n$

By default:

$$w \stackrel{*}{\Rightarrow} w$$

Example

Grammar

$$S \rightarrow aSb$$

$$S \rightarrow \lambda$$

Derivations

$$\begin{array}{c} * \\ S \Rightarrow \lambda \end{array}$$

$$\begin{array}{c} * \\ S \Rightarrow ab \end{array}$$

$$\begin{array}{c} * \\ S \Rightarrow aabb \end{array}$$

$$\begin{array}{c} * \\ S \Rightarrow aaabbb \end{array}$$

Example

Grammar

$$S \rightarrow aSb$$

$$S \rightarrow \lambda$$

Derivations

$$S \xRightarrow{*} aaSbb$$

$$aaSbb \xRightarrow{*} aaaaaaSbbbbbb$$

Another Grammar Example

Grammar G :

$$S \rightarrow Ab$$
$$A \rightarrow aAb$$
$$A \rightarrow \lambda$$

Derivations:

$$S \Rightarrow Ab \Rightarrow b$$

$$S \Rightarrow Ab \Rightarrow aAbb \Rightarrow abb$$

$$S \Rightarrow Ab \Rightarrow aAbb \Rightarrow aaAbbb \Rightarrow aabbbb$$

More Derivations

$$S \Rightarrow Ab \Rightarrow aAbb \Rightarrow aaAbbb \Rightarrow aaaAbbbb \\ \Rightarrow aaaaAbbbbb \Rightarrow aaaaabbbbb$$

$$* \\ S \Rightarrow aaaaabbbbb$$

$$* \\ S \Rightarrow aaaaaabbbbbbb$$

$$* \\ S \Rightarrow a^n b^n b$$

Language of a Grammar

For a grammar G
with start variable S :

$$L(G) = \{w : S \overset{*}{\Rightarrow} w\}$$

String of terminals

Example

For grammar G : $S \rightarrow Ab$

$$A \rightarrow aAb$$

$$A \rightarrow \lambda$$

$$L(G) = \{a^n b^n b : n \geq 0\}$$

Since: $S \xRightarrow{*} a^n b^n b$

A Convenient Notation

$A \rightarrow aAb$

$A \rightarrow \lambda$



$A \rightarrow aAb \mid \lambda$

$\langle \textit{article} \rangle \rightarrow a$

$\langle \textit{article} \rangle \rightarrow \textit{the}$



$\langle \textit{article} \rangle \rightarrow a \mid \textit{the}$

Linear Grammars

Linear Grammars

Grammars with
at most one variable at the right side
of a production

Examples: $S \rightarrow aSb$

$S \rightarrow \lambda$

$S \rightarrow Ab$

$A \rightarrow aAb$

$A \rightarrow \lambda$

A Non-Linear Grammar

Grammar G :

$$S \rightarrow SS$$
$$S \rightarrow \lambda$$
$$S \rightarrow aSb$$
$$S \rightarrow bSa$$

$$L(G) = \{w : n_a(w) = n_b(w)\}$$

Number of a in string w

Another Linear Grammar

Grammar G :

$$S \rightarrow A$$
$$A \rightarrow aB \mid \lambda$$
$$B \rightarrow Ab$$

$$L(G) = \{a^n b^n : n \geq 0\}$$

Right-Linear Grammars

All productions have form: $A \rightarrow xB$

or

$$A \rightarrow x$$



Example: $S \rightarrow abS$

$$S \rightarrow a$$

string of
terminals

Left-Linear Grammars

All productions have form: $A \rightarrow Bx$

or

$$A \rightarrow x$$



Example: $S \rightarrow Aab$

$A \rightarrow Aab \mid B$

$B \rightarrow a$

string of
terminals

Regular Grammars

Regular Grammars

A regular grammar is any right-linear or left-linear grammar

Examples:

G_1

$$S \rightarrow abS$$

$$S \rightarrow a$$

G_2

$$S \rightarrow Aab$$

$$A \rightarrow Aab \mid B$$

$$B \rightarrow a$$

Observation

Regular grammars generate regular languages

Examples:

G_1

$S \rightarrow abS$

$S \rightarrow a$

$L(G_1) = (ab)^* a$

G_2

$S \rightarrow Aab$

$A \rightarrow Aab \mid B$

$B \rightarrow a$

$L(G_2) = aab(ab)^*$

Regular Grammars
Generate
Regular Languages

Theorem

$$\left\{ \begin{array}{l} \text{Languages} \\ \text{Generated by} \\ \text{Regular Grammars} \end{array} \right\} = \left\{ \begin{array}{l} \text{Regular} \\ \text{Languages} \end{array} \right\}$$

Theorem - Part 1

$$\left\{ \begin{array}{l} \text{Languages} \\ \text{Generated by} \\ \text{Regular Grammars} \end{array} \right\} \subseteq \left\{ \begin{array}{l} \text{Regular} \\ \text{Languages} \end{array} \right\}$$

Any regular grammar generates
a regular language

Theorem - Part 2

$$\left\{ \begin{array}{l} \text{Languages} \\ \text{Generated by} \\ \text{Regular Grammars} \end{array} \right\} \equiv \left\{ \begin{array}{l} \text{Regular} \\ \text{Languages} \end{array} \right\}$$

Any regular language is generated by a regular grammar

Proof - Part 1

$$\left\{ \begin{array}{l} \text{Languages} \\ \text{Generated by} \\ \text{Regular Grammars} \end{array} \right\} \subseteq \left\{ \begin{array}{l} \text{Regular} \\ \text{Languages} \end{array} \right\}$$

The language $L(G)$ generated by any regular grammar G is regular

The case of Right-Linear Grammars

Let G be a right-linear grammar

We will prove: $L(G)$ is regular

Proof idea: We will construct NFA M
with $L(M) = L(G)$

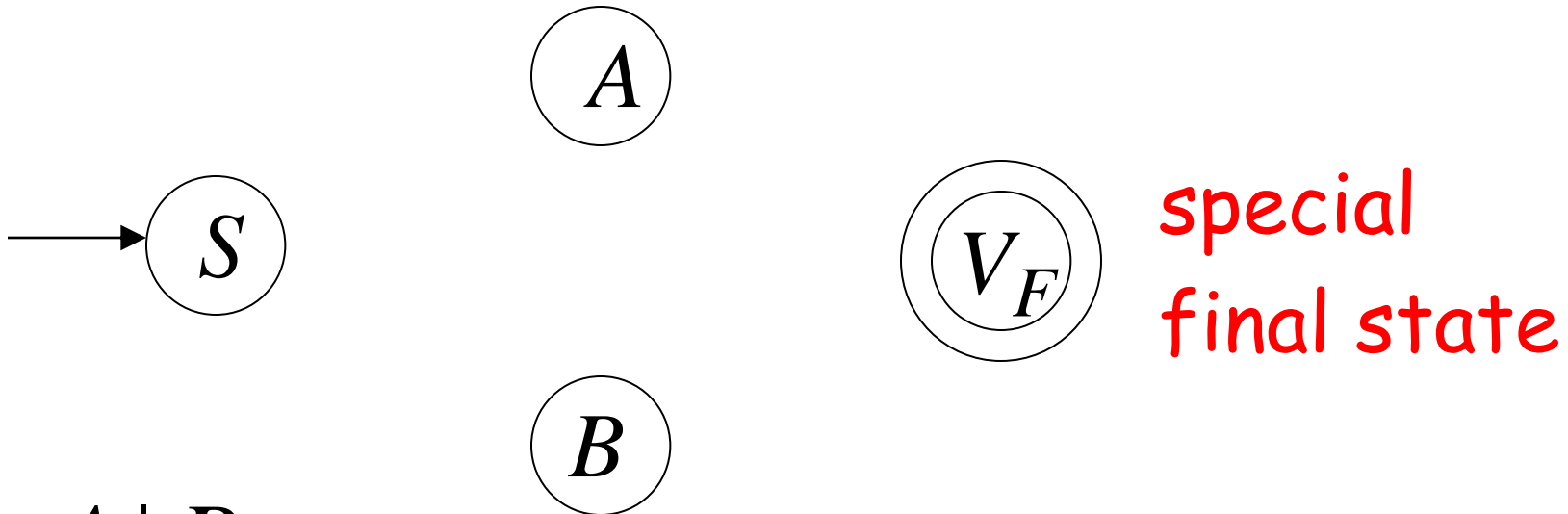
Grammar G is right-linear

Example: $S \rightarrow aA \mid B$

$A \rightarrow aa B$

$B \rightarrow b B \mid a$

Construct NFA M such that every state is a grammar variable:

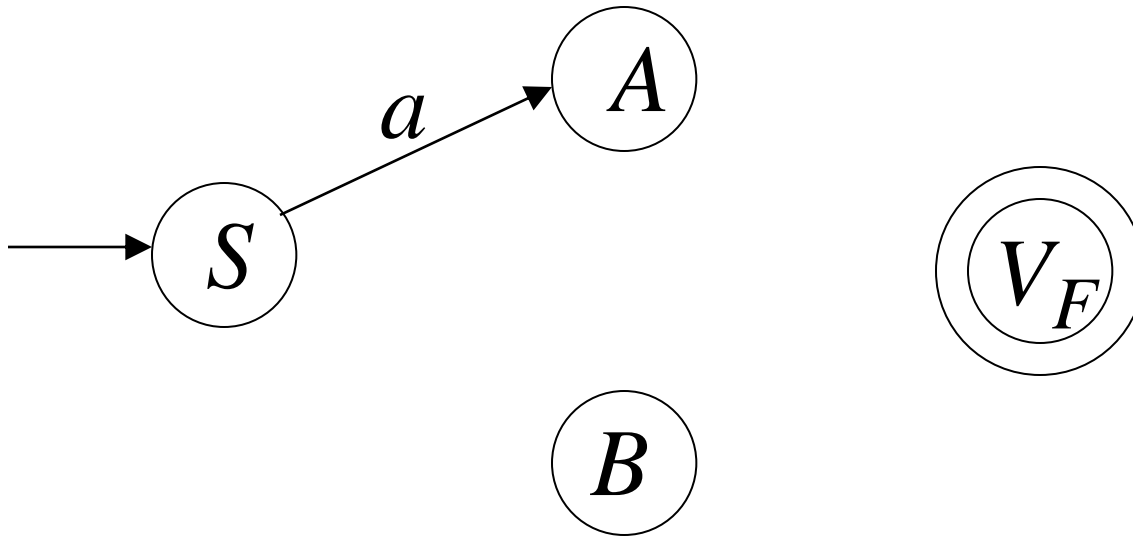


$$S \rightarrow aA \mid B$$

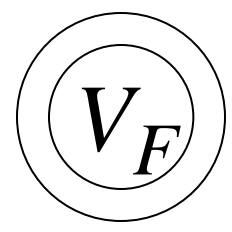
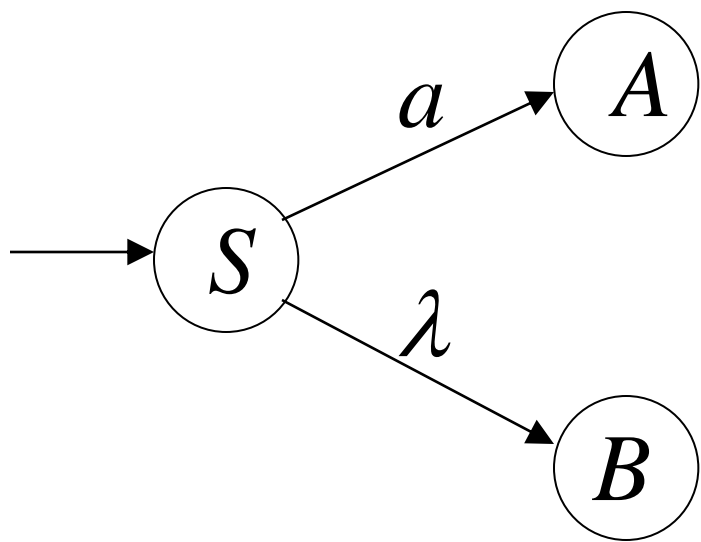
$$A \rightarrow aa B$$

$$B \rightarrow b B \mid a$$

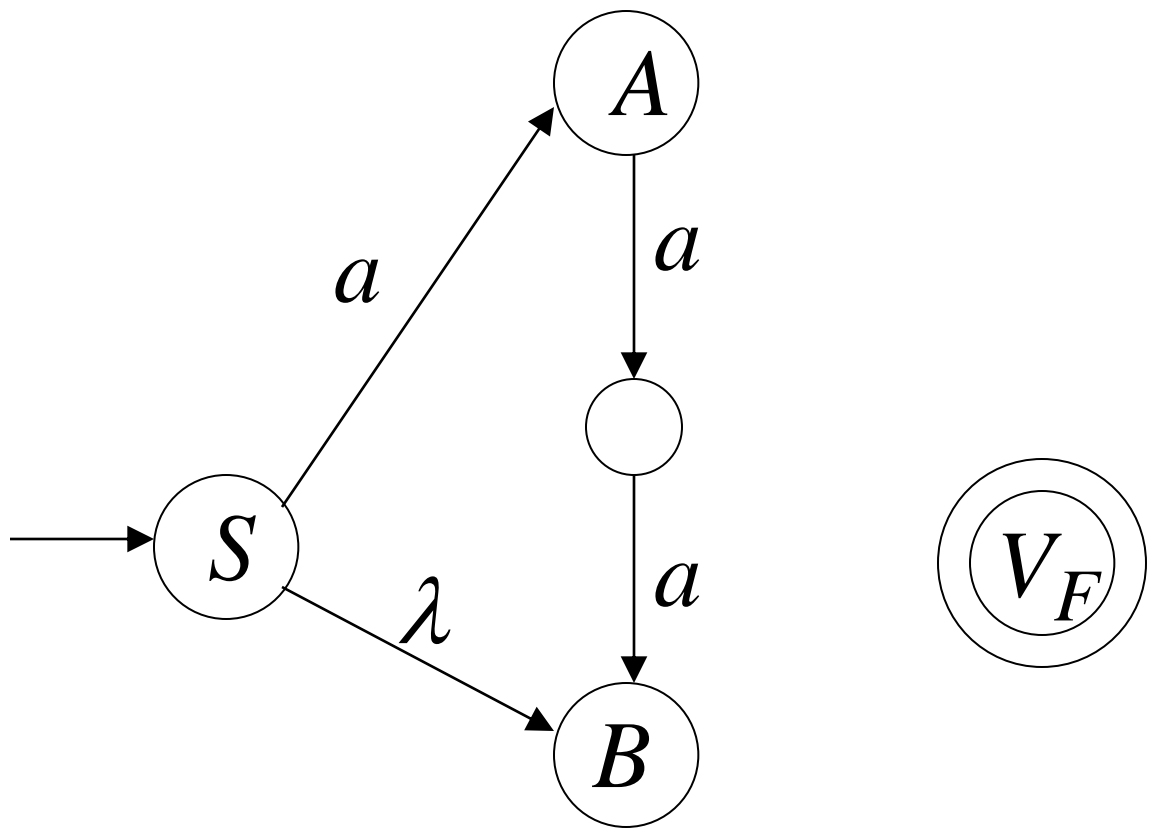
Add edges for each production:



$S \rightarrow aA$

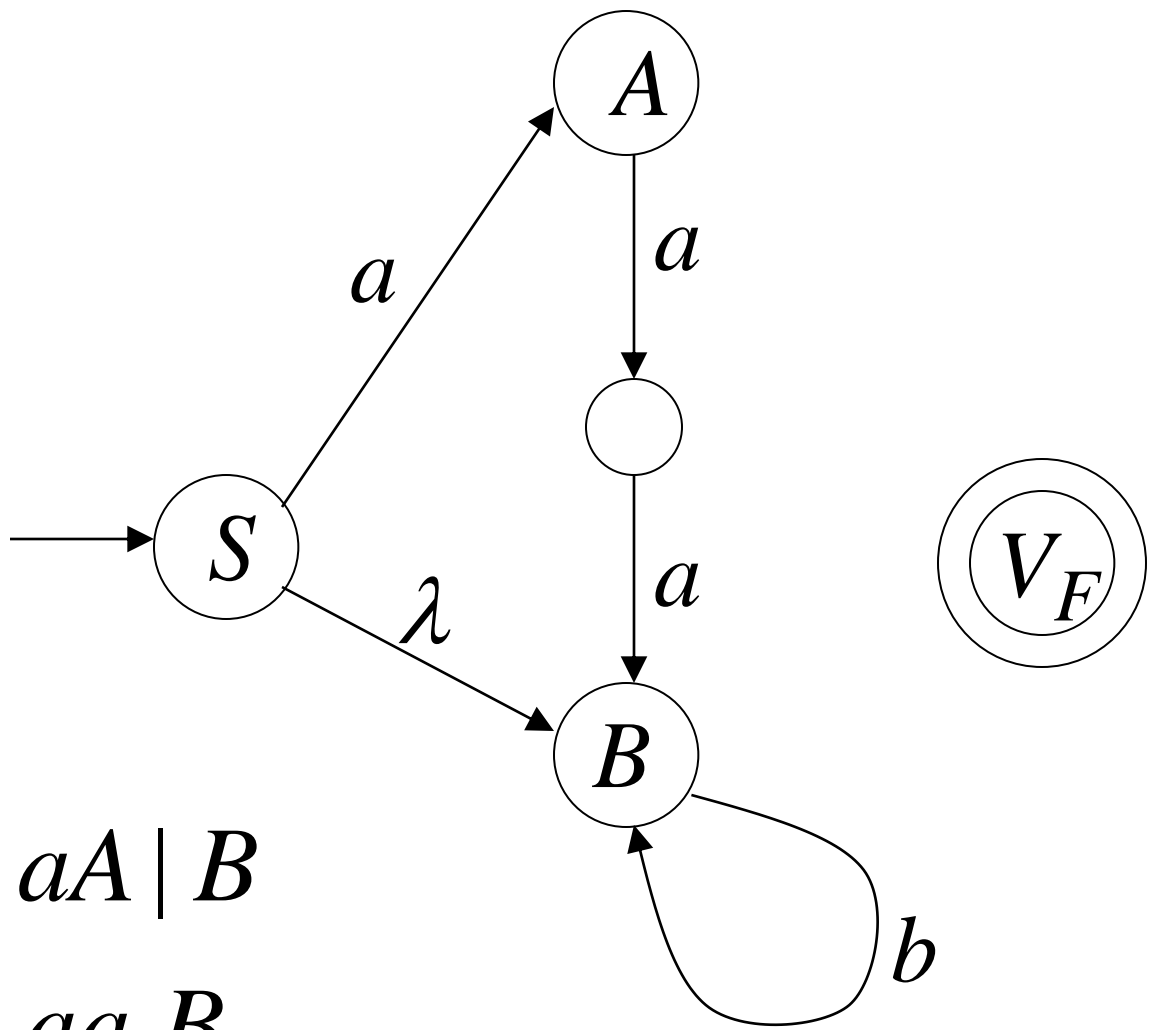


$S \rightarrow aA \mid B$



$$S \rightarrow aA \mid B$$

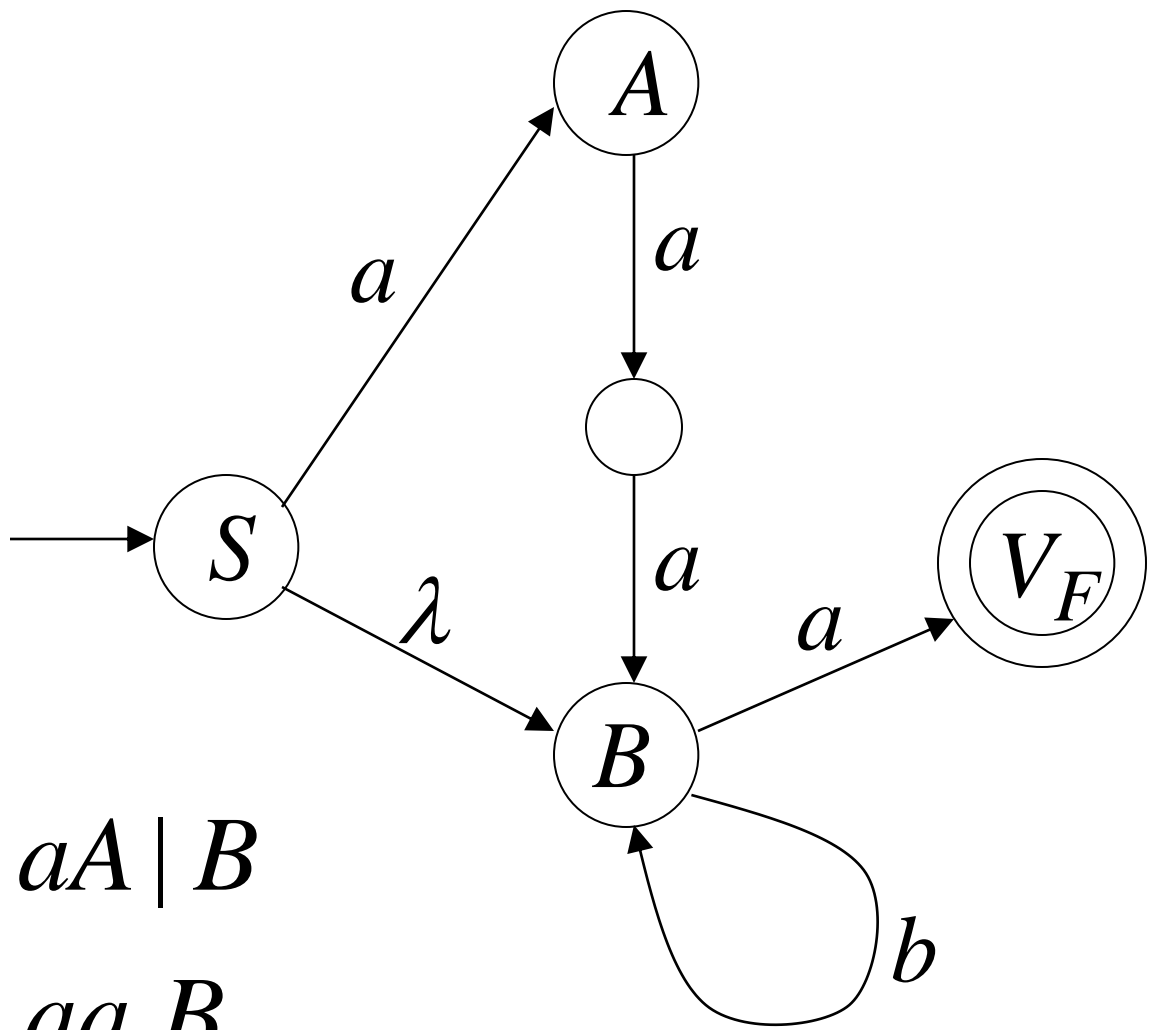
$$A \rightarrow aa B$$



$$S \rightarrow aA \mid B$$

$$A \rightarrow aa B$$

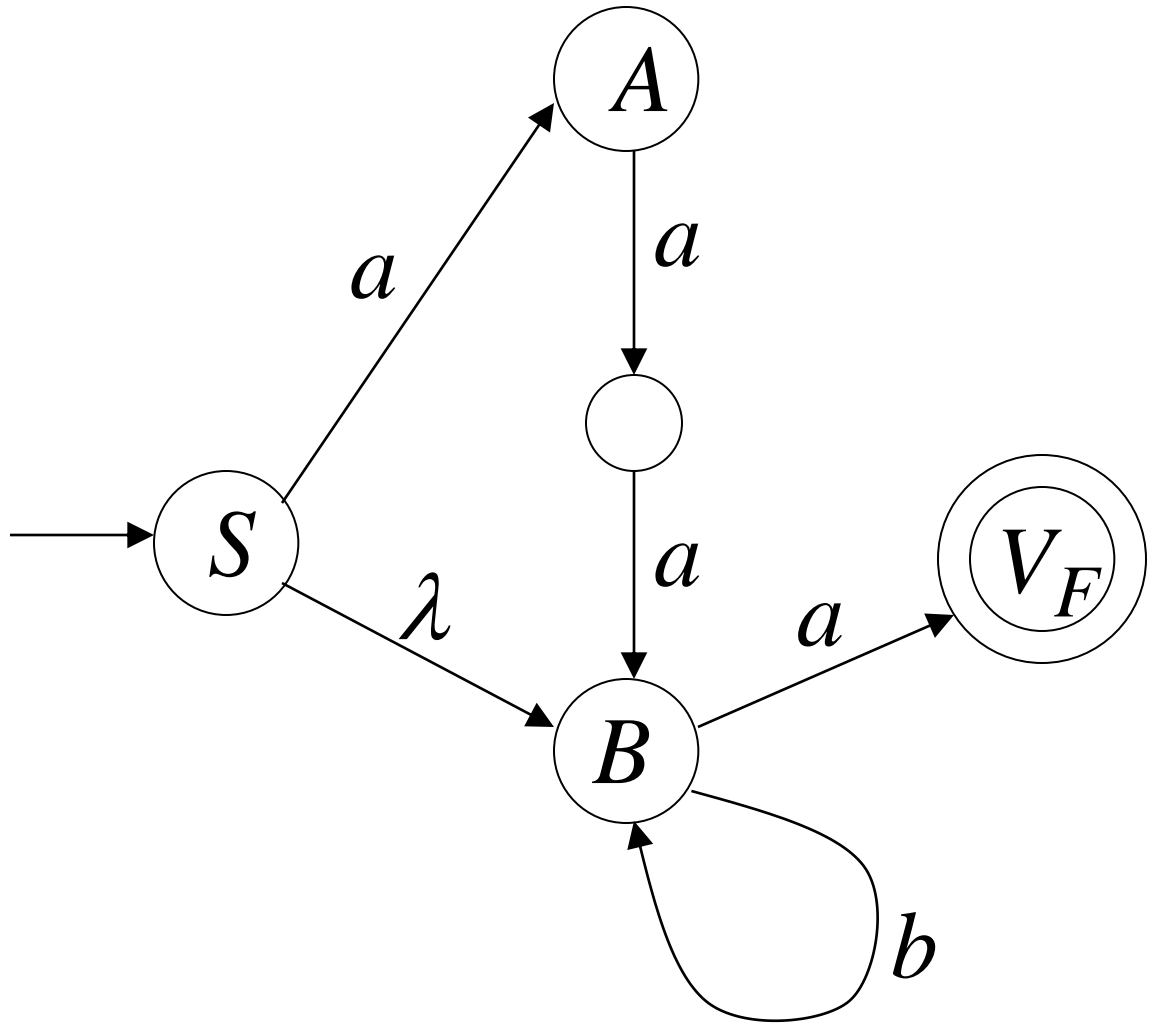
$$B \rightarrow bB$$



$$S \rightarrow aA \mid B$$

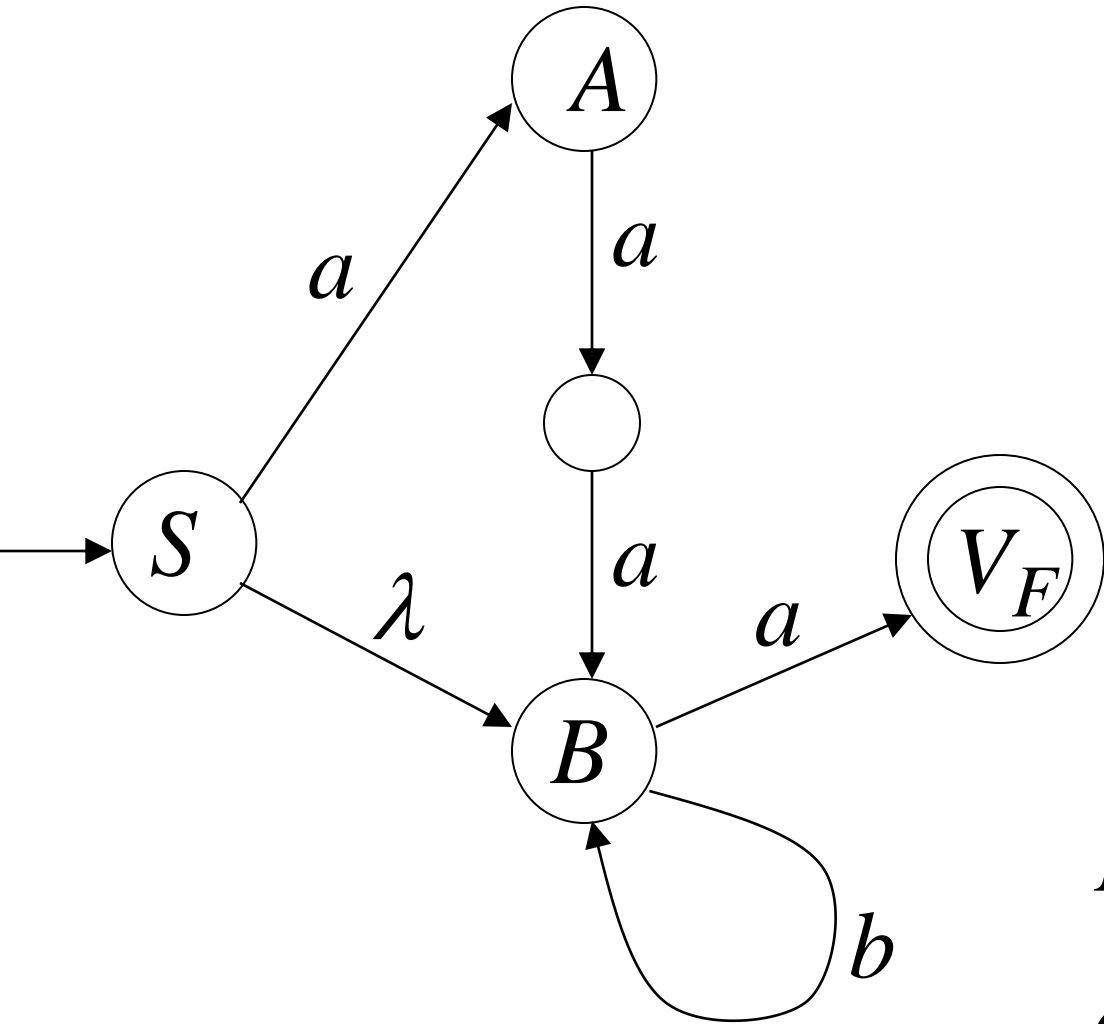
$$A \rightarrow aa B$$

$$B \rightarrow bB \mid a$$



$S \Rightarrow aA \Rightarrow aaaB \Rightarrow aaabB \Rightarrow aaaba$

NFA M



Grammar

G

$S \rightarrow aA \mid B$

$A \rightarrow aaB$

$B \rightarrow bB \mid a$

$L(M) = L(G) =$
 $aaab^*a + b^*a$

In General

A right-linear grammar G

has variables: V_0, V_1, V_2, \dots

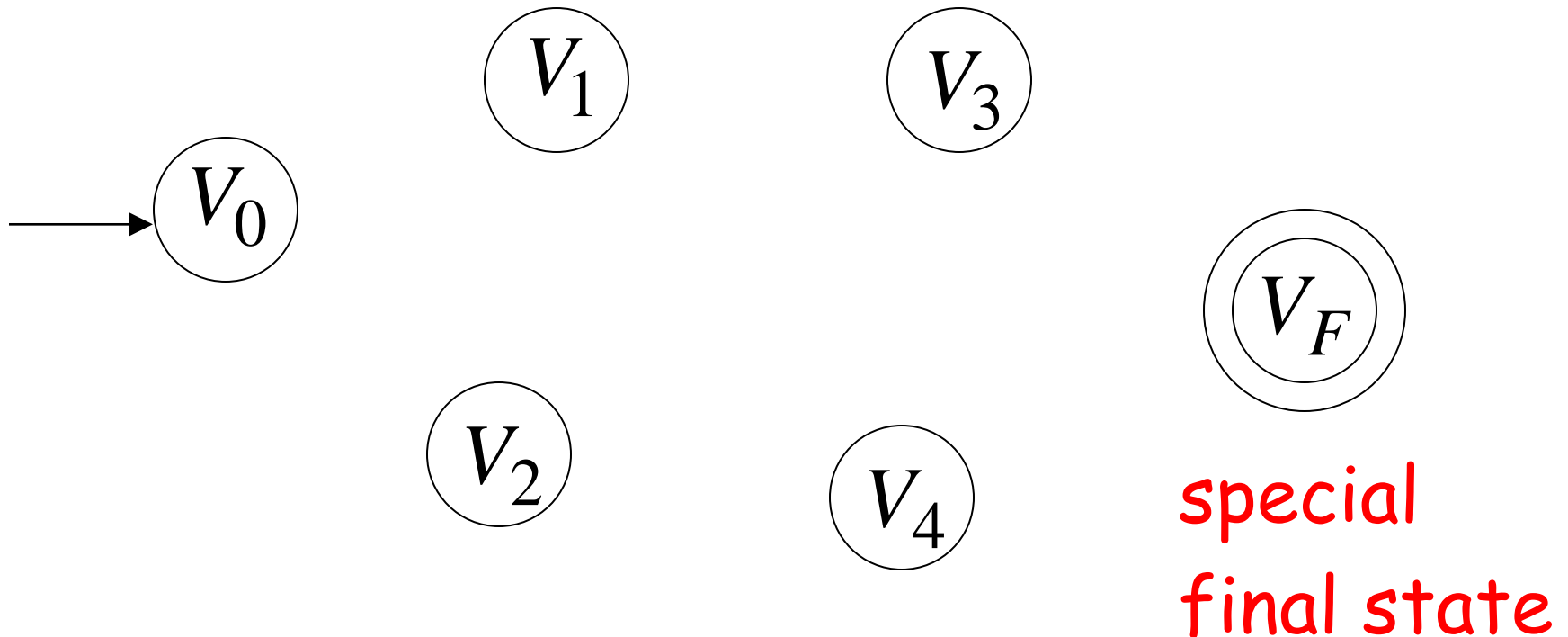
and productions: $V_i \rightarrow a_1 a_2 \cdots a_m V_j$

or

$V_i \rightarrow a_1 a_2 \cdots a_m$

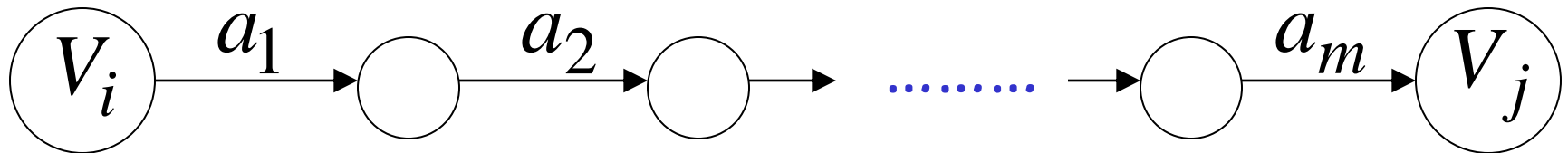
We construct the NFA M such that:

each variable V_i corresponds to a node:



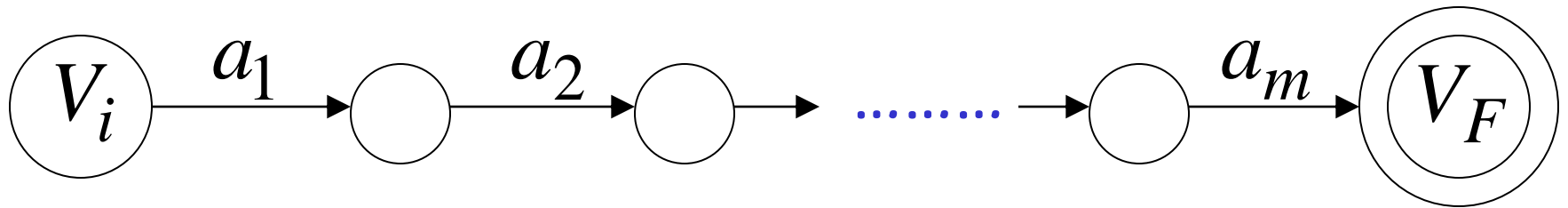
For each production: $V_i \rightarrow a_1 a_2 \cdots a_m V_j$

we add transitions and intermediate nodes

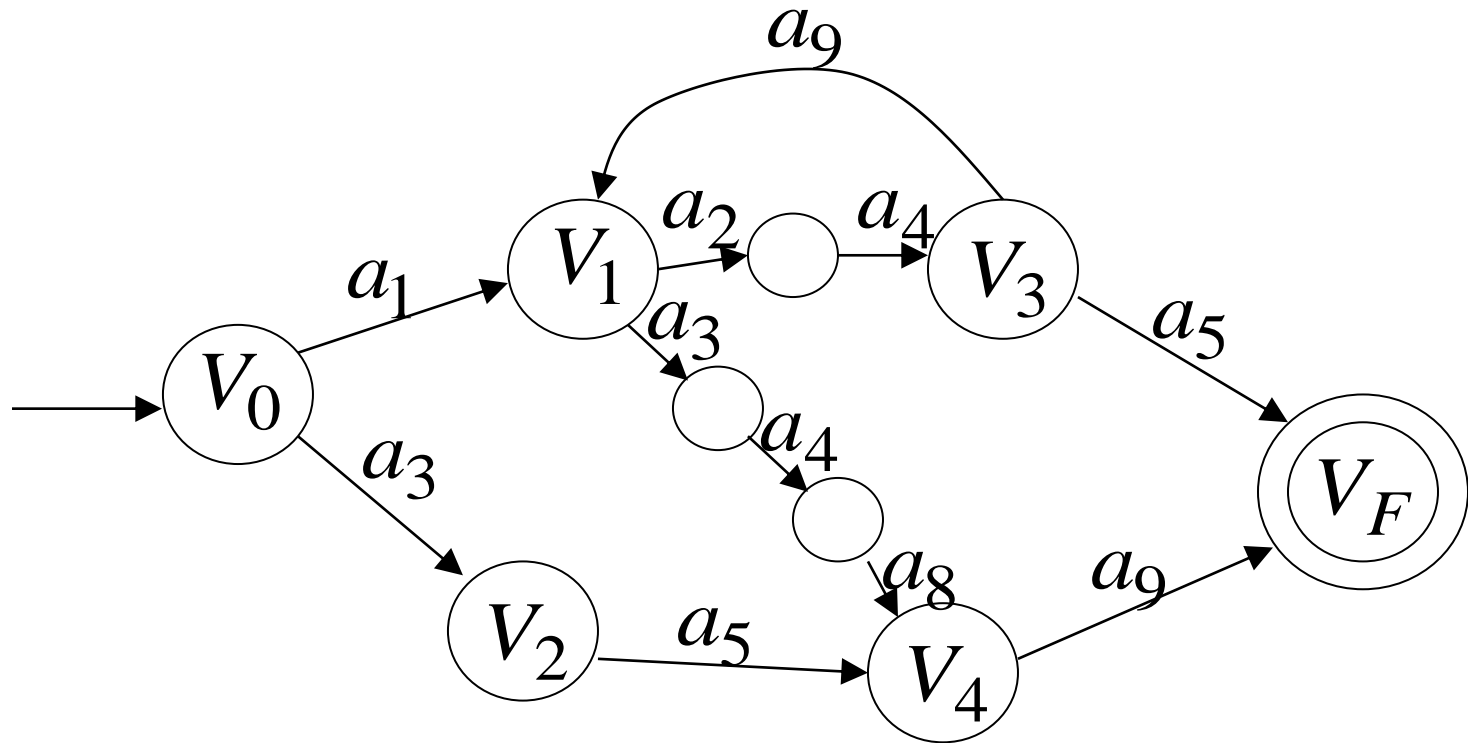


For each production: $V_i \rightarrow a_1 a_2 \cdots a_m$

we add transitions and intermediate nodes



Resulting NFA M looks like this:



It holds that: $L(G) = L(M)$

The case of Left-Linear Grammars

Let G be a left-linear grammar

We will prove: $L(G)$ is regular

Proof idea:

We will construct a right-linear grammar G' with $L(G) = L(G')^R$

Since G is left-linear grammar
the productions look like:

$$A \rightarrow Ba_1a_2 \cdots a_k$$

$$A \rightarrow a_1a_2 \cdots a_k$$

Construct right-linear grammar G'

Left
linear

G

$$A \rightarrow Ba_1a_2 \cdots a_k$$

$$A \rightarrow Bv$$



Right
linear

G'

$$A \rightarrow a_k \cdots a_2a_1B$$

$$A \rightarrow v^R B$$

Construct right-linear grammar G'

Left
linear

G

$$A \rightarrow a_1 a_2 \cdots a_k$$

$$A \rightarrow v$$



Right
linear

G'

$$A \rightarrow a_k \cdots a_2 a_1$$

$$A \rightarrow v^R$$

It is easy to see that: $L(G) = L(G')^R$

Since G' is right-linear, we have:



Proof - Part 2

$$\left\{ \begin{array}{l} \text{Languages} \\ \text{Generated by} \\ \text{Regular Grammars} \end{array} \right\} \equiv \left\{ \begin{array}{l} \text{Regular} \\ \text{Languages} \end{array} \right\}$$

Any regular language L is generated
by some regular grammar G

Any regular language L is generated
by some regular grammar G

Proof idea:

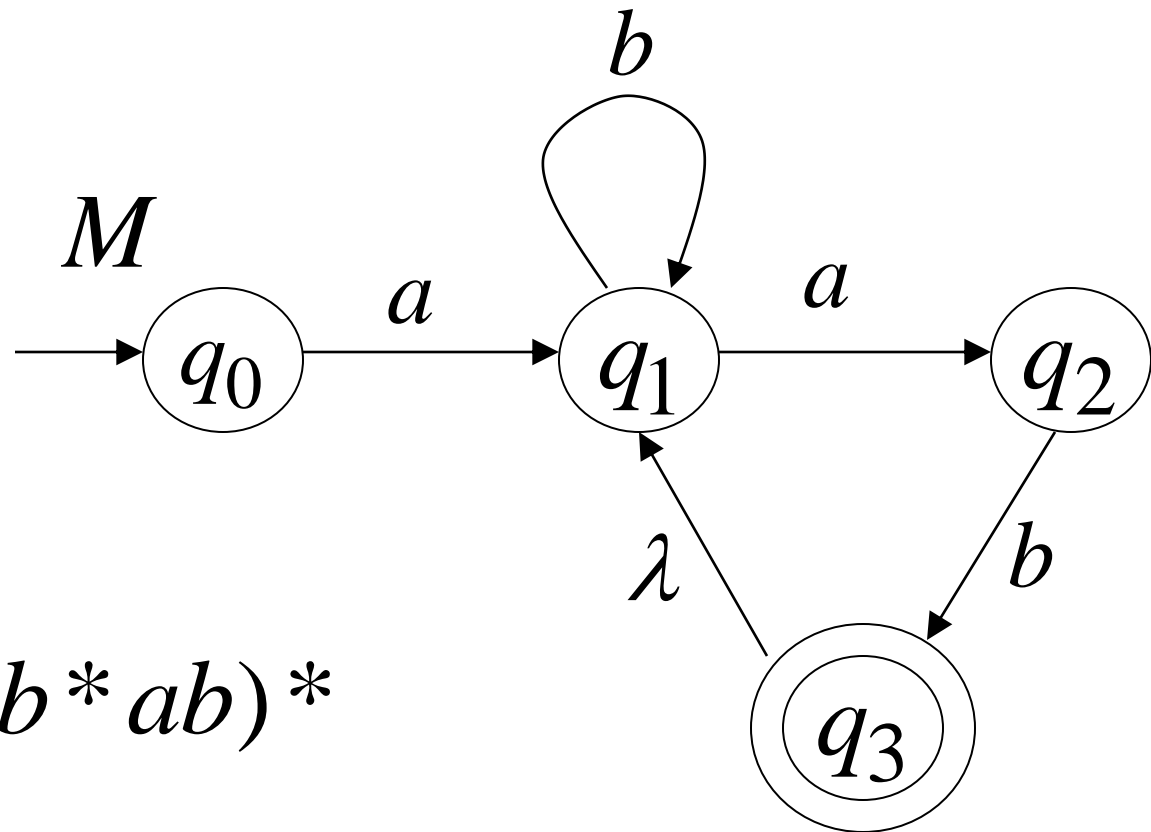
Let M be the NFA with $L = L(M)$.

Construct from M a regular grammar G
such that $L(M) = L(G)$

Since L is regular

there is an NFA M such that $L = L(M)$

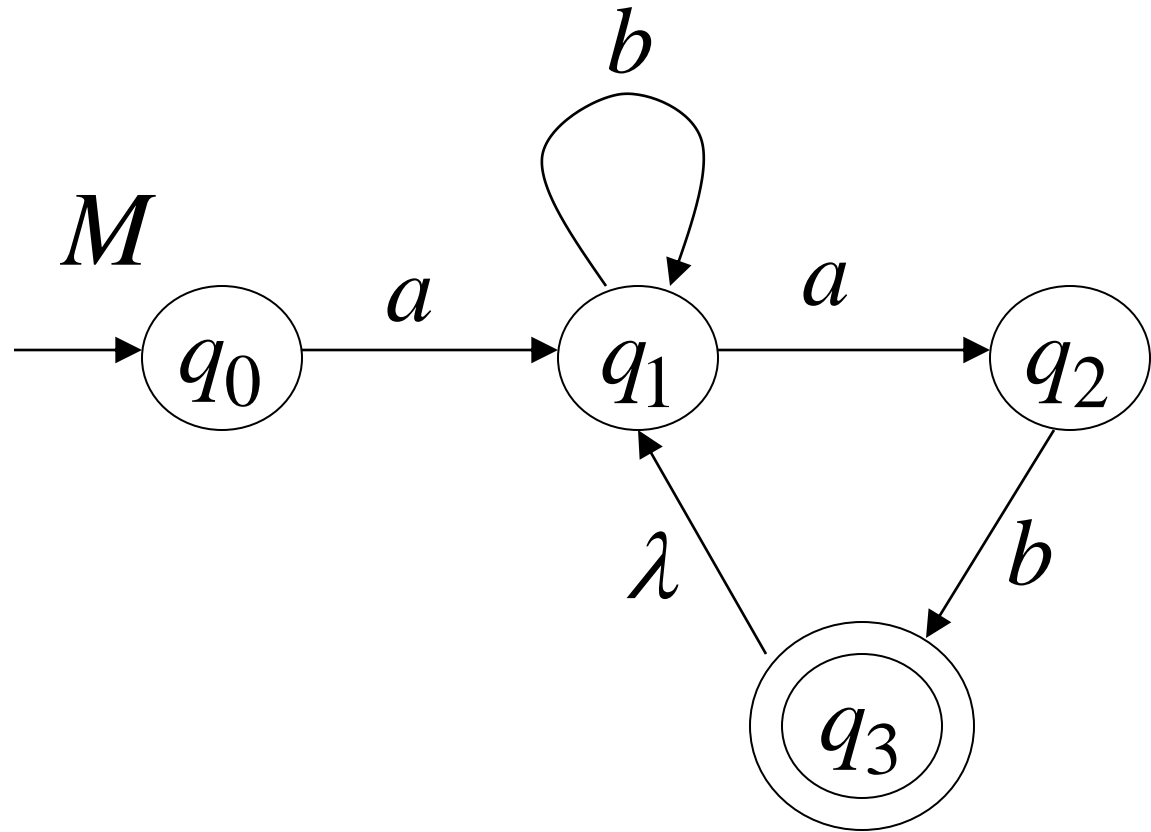
Example:



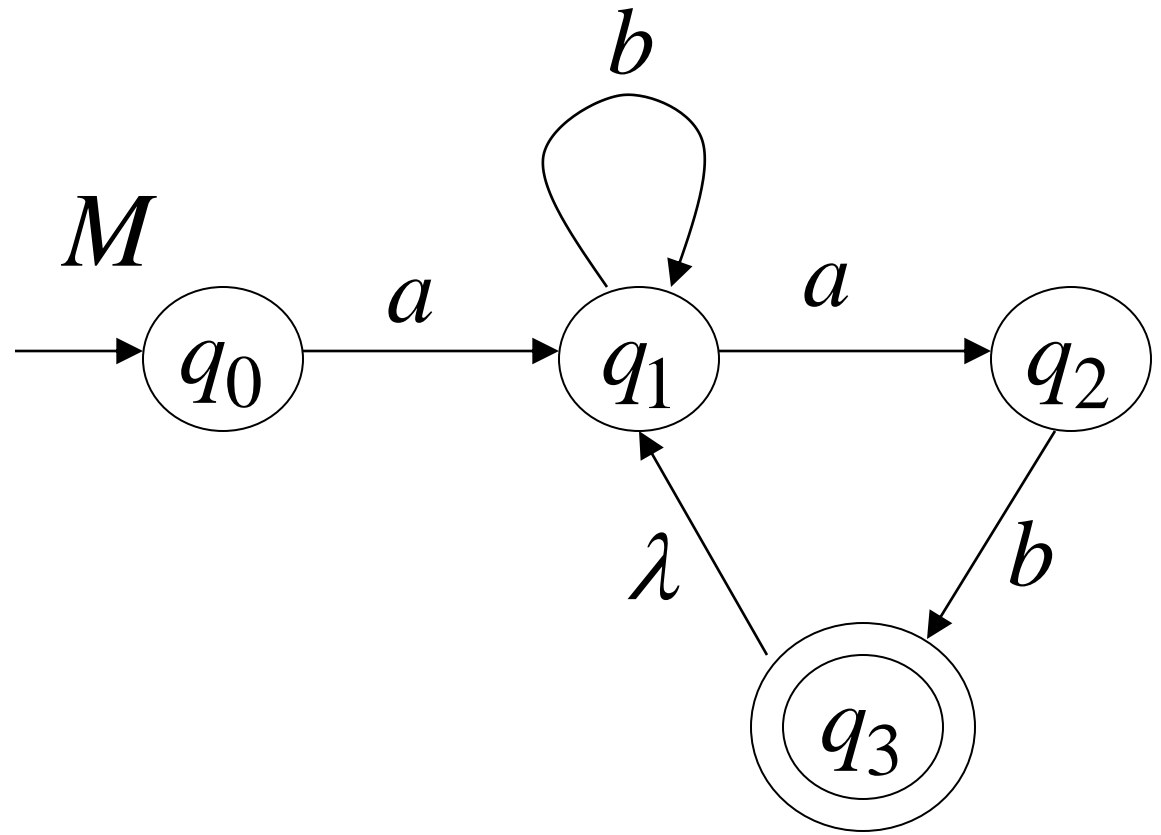
$$L = ab^*ab(b^*ab)^*$$

$$L = L(M)$$

Convert M to a right-linear grammar



$$q_0 \rightarrow aq_1$$



$$q_0 \rightarrow aq_1$$

$$q_1 \rightarrow bq_1$$

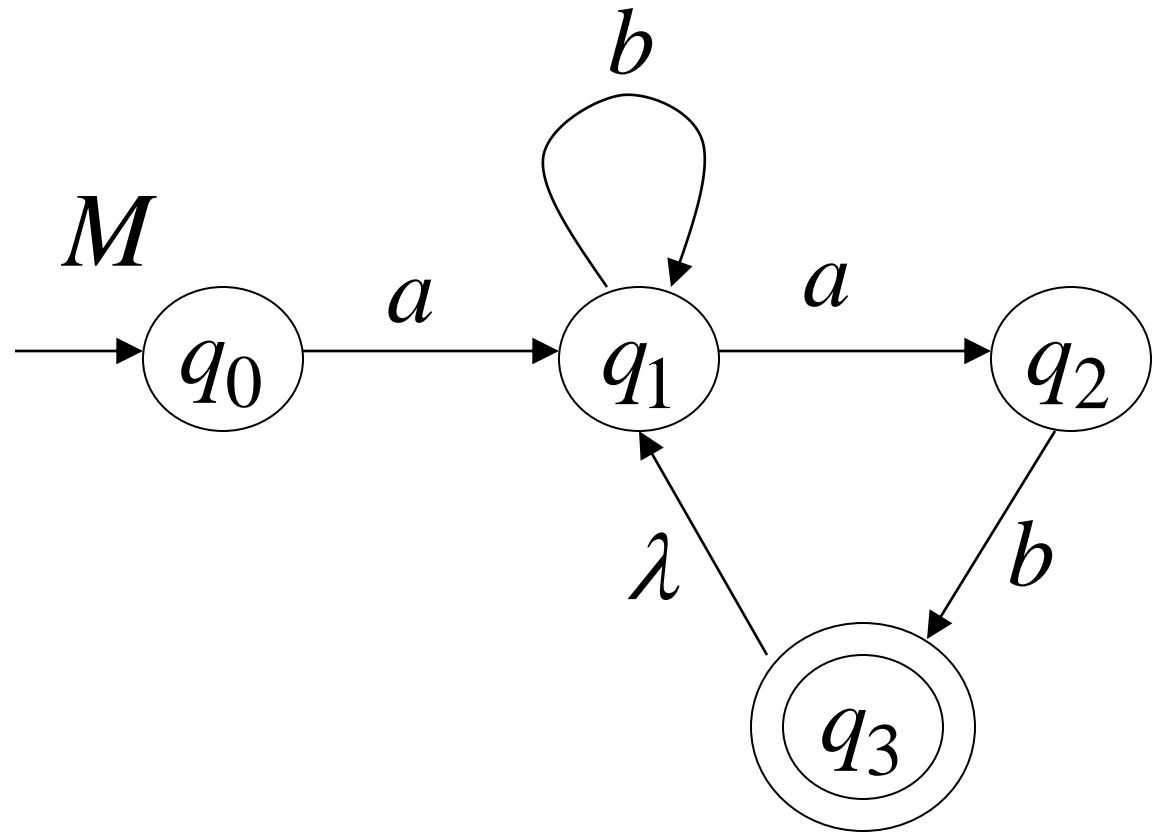
$$q_1 \rightarrow aq_2$$

$q_0 \rightarrow aq_1$

$q_1 \rightarrow bq_1$

$q_1 \rightarrow aq_2$

$q_2 \rightarrow bq_3$



$$L(G) = L(M) = L$$

G

$$q_0 \rightarrow aq_1$$

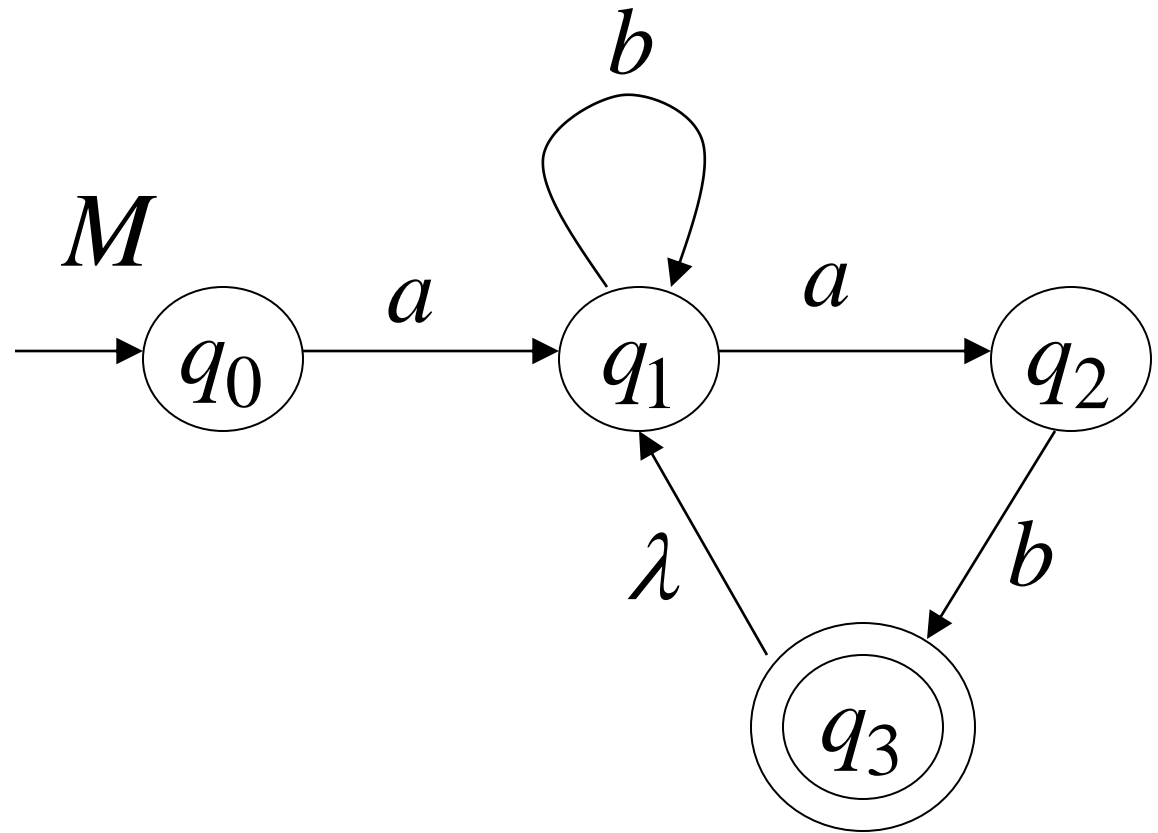
$$q_1 \rightarrow bq_1$$

$$q_1 \rightarrow aq_2$$

$$q_2 \rightarrow bq_3$$

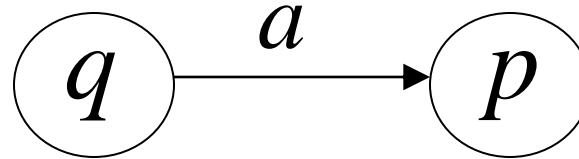
$$q_3 \rightarrow q_1$$

$$q_3 \rightarrow \lambda$$

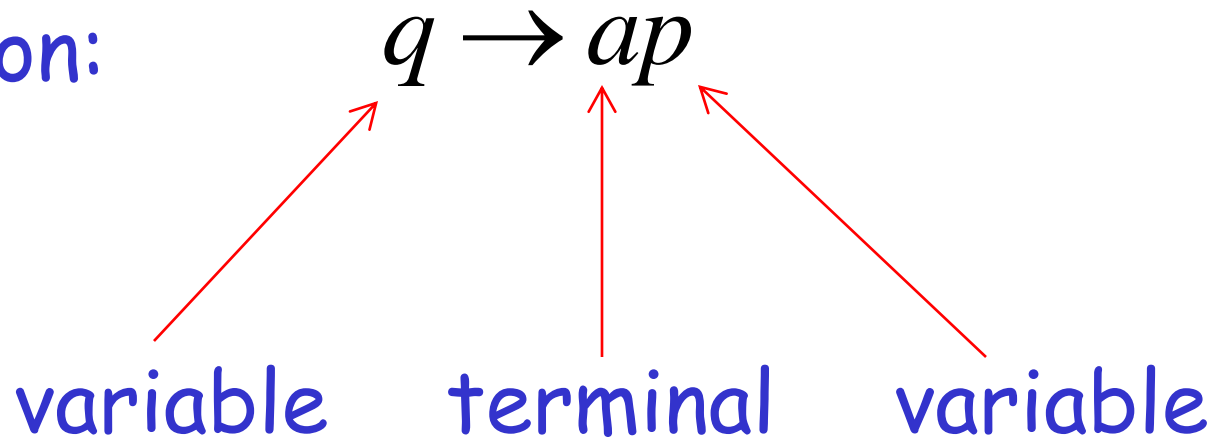


In General

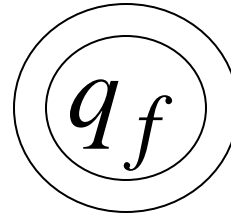
For any transition:



Add production:



For any final state:



Add production:

$$q_f \rightarrow \lambda$$

Since G is right-linear grammar

G is also a regular grammar

with $L(G) = L(M) = L$

The End