CFG ⇒ PDA

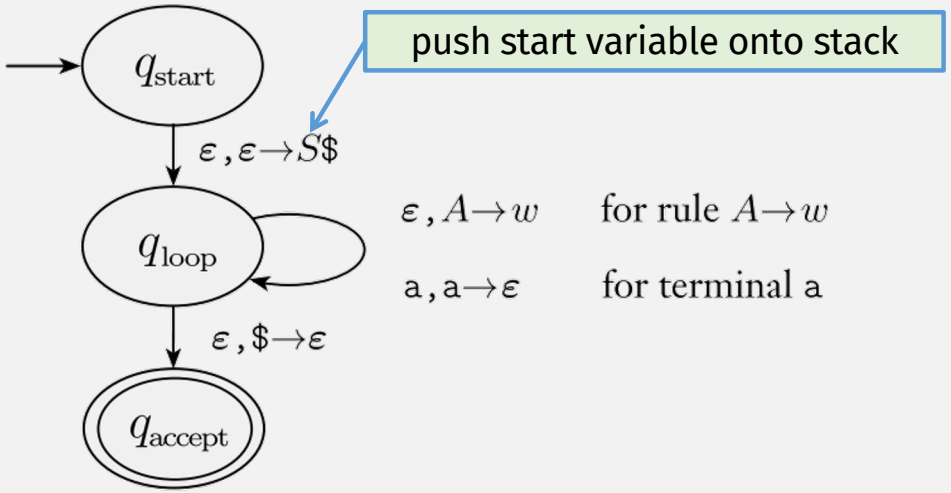
A lang is a CFL iff some PDA recognizes it

- => If a language is a CFL, then a PDA recognizes it
  - (Easier)
  - We know: A CFL has a CFG describing it (definition of CFL)
  - To prove forward dir: Convert CFG -> PDA
- <= If a PDA recognizes a language, then it's a CFL

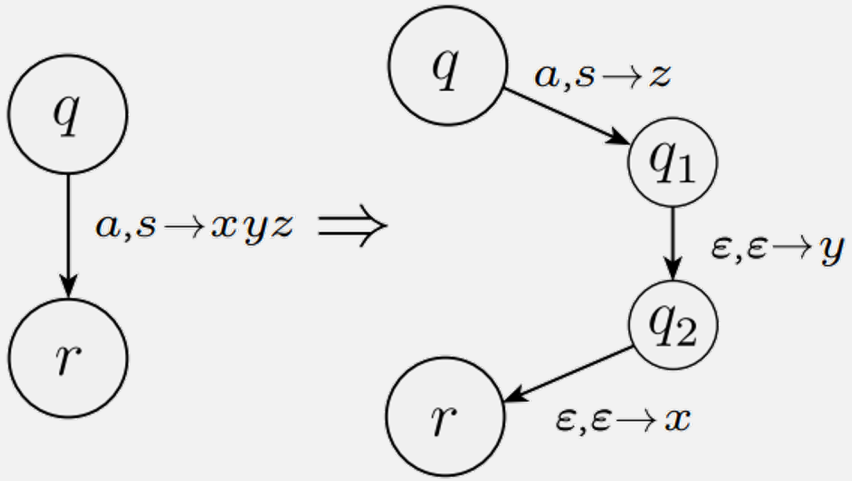
CFG -> PDA

- Construct a PDA from CFG such that:
  - PDA accepts input string only if the CFG can generate that string

- Intuitively, PDA will <u>nondeterministically</u> try all rules



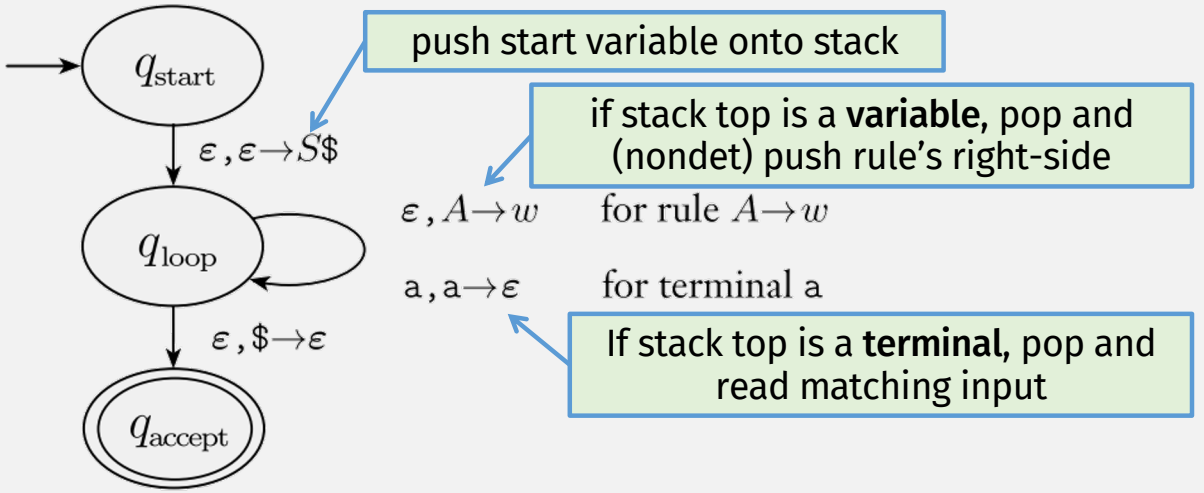
Transition with multiple stack pushes



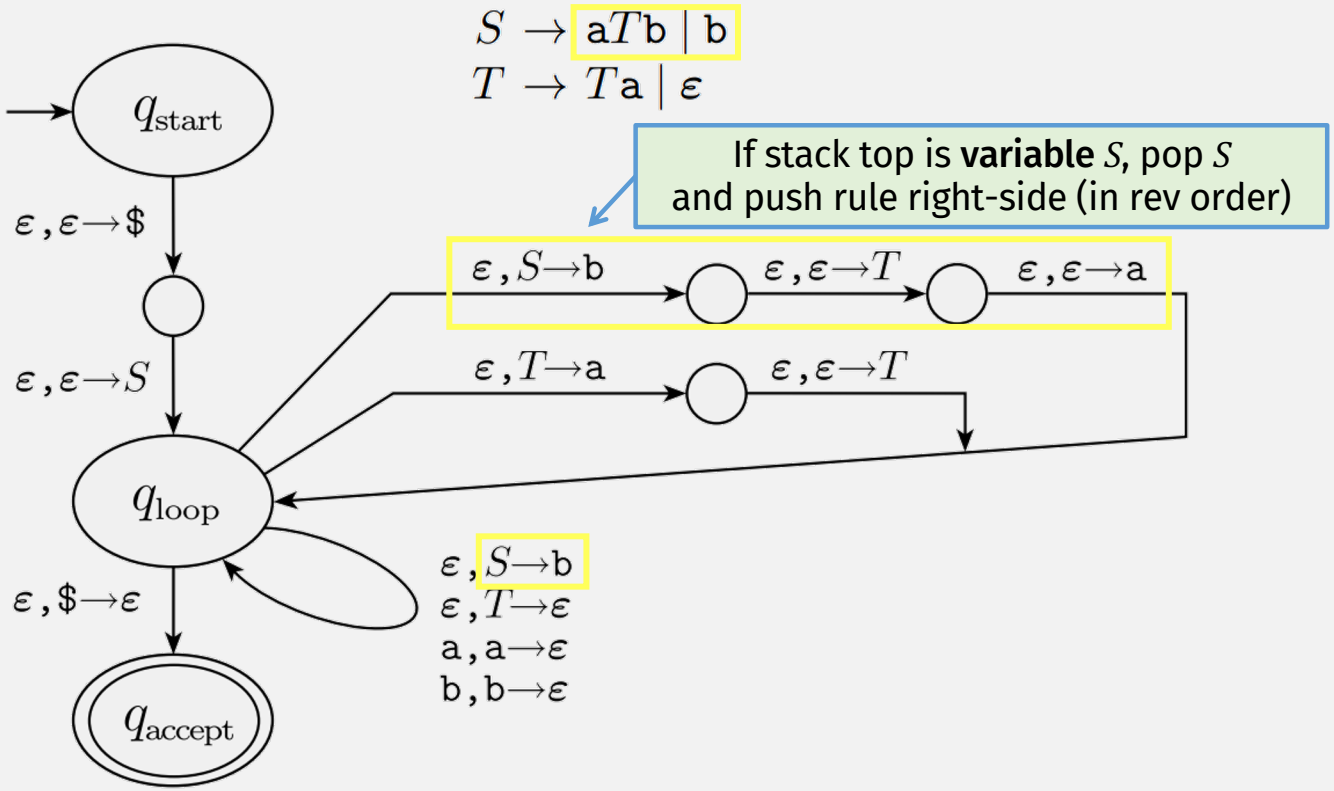
CFG -> PDA

- Construct PDA from CFG such that:
  - PDA accepts input string only if the CFG can generate that string

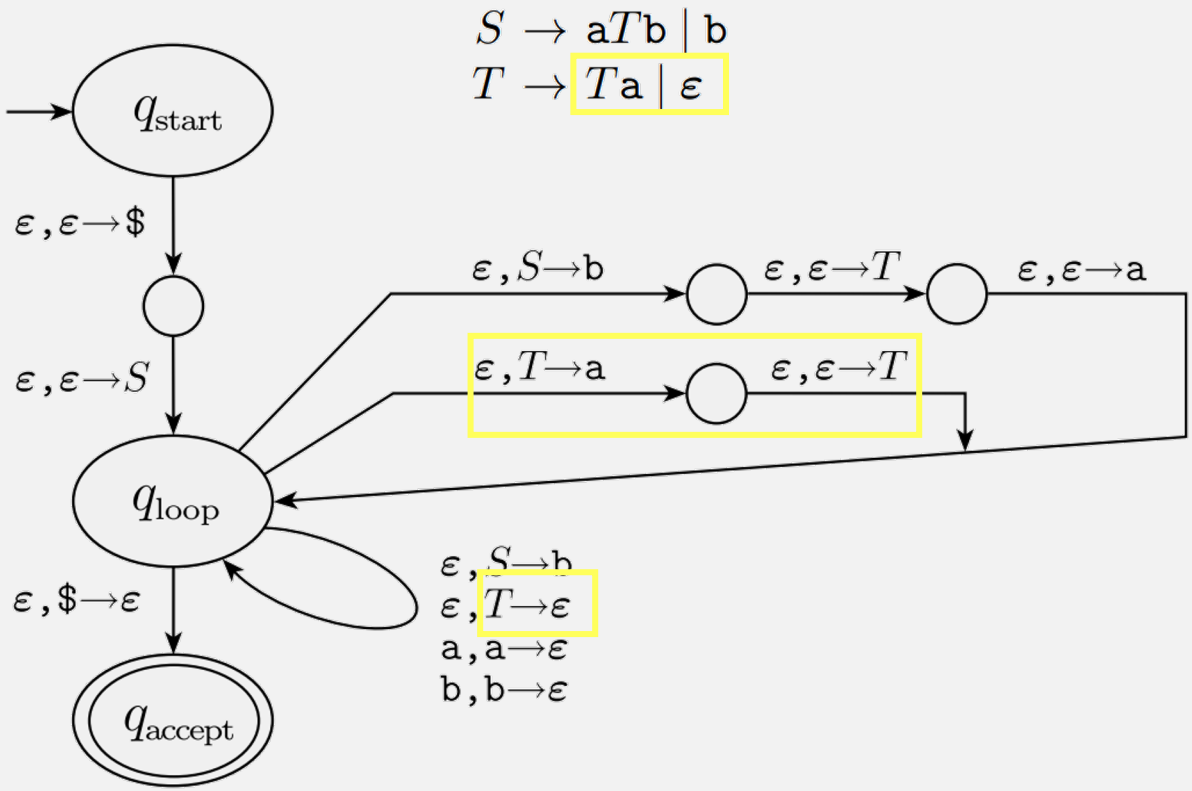
- Intuitively, PDA will <u>nondeterministically</u> try all rules



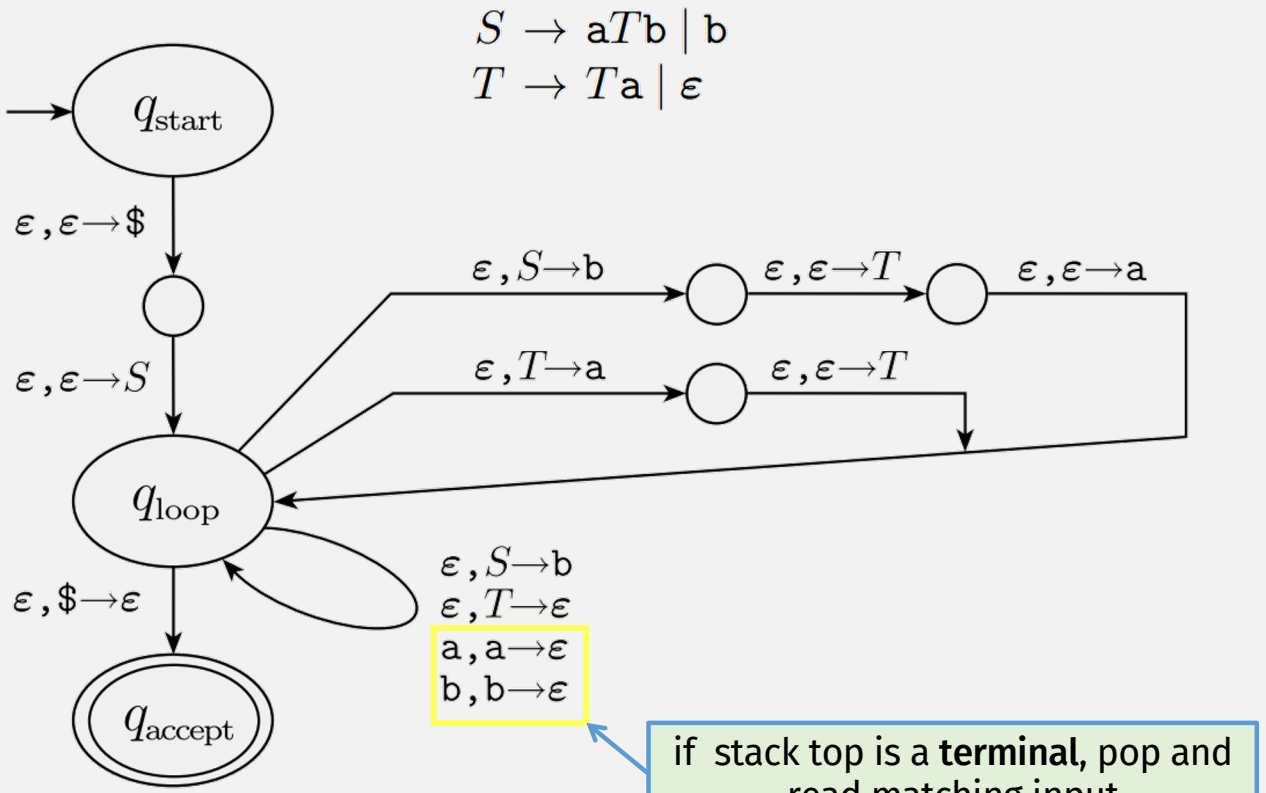
Example CFG -> PDA



Example CFG -> PDA

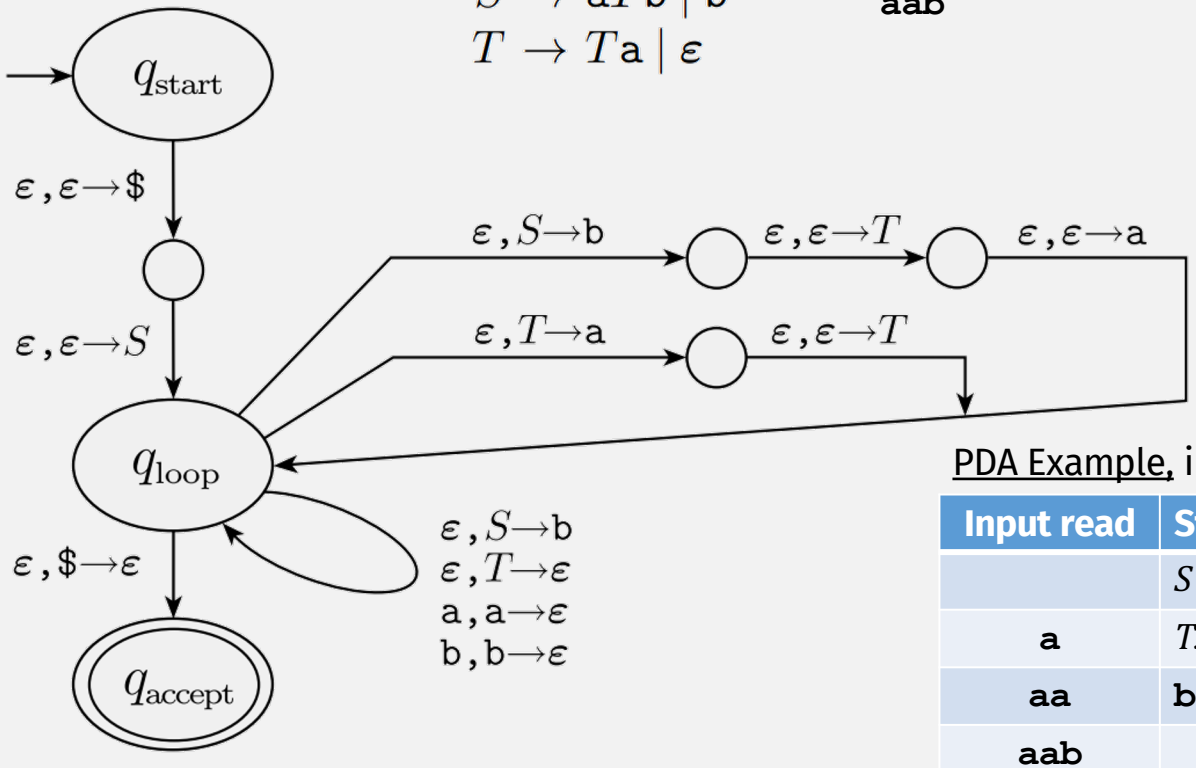


Example CFG -> PDA



Example CFG -> PDA

$S \rightarrow \mathtt{a}T\mathtt{b} \mid \mathtt{b}$
$T \rightarrow T\mathtt{a} \mid \varepsilon$



Example Derivation using CFG:
S ->
aT**b** ->
aT**ab** ->
aab

PDA Example, input **aab**

Input read	Stack
	S -> **a**T**b** ->
a	T**b** -> T**ab** -> **ab** ->
aa	**b** ->
aab	

A lang is a CFL iff some PDA recognizes it

- => If a language is a CFL, then a PDA recognizes it
  - (Easier)
  - We know: A CFL has a CFG describing it (definition of CFL)
  - Need to: Convert CFG -> PDA **(DONE!)**
- <= If a PDA recognizes a language, then it's a CFL
  - (Harder)
  - Need to: Convert PDA -> CFG