Session 19

Deterministic Pushdown Automata

Three versions of *Pal*

- $Pal_{mark} = \{xcx^r \mid x \in \{a, b\}^*\}$
  - *abbacabba*
- $Pal_{even} = \{xx^r \mid x \in \{a, b\}^*\}$
  - *aabbbbaa*
- $Pal = \{x \mid x = x^r \in \{a, b\}^*\}$
  - *aabbbbaa*
  - *aabbabbaa*

A machine for accepting *Pal*

- The language:

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

- Define M_{pal}:

  $M = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, q_0, Z_0, \{q_2\}, \delta)$

- Two kinds of non-determinism:
  - For a given state, input symbol and top of the stack there is more than one move
  - There are states in which the machine has the choice between consuming a symbol or making a Λ-transition

Transition function for *Pal*

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0), (q_1, Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0), (q_1, Z_0)$
3	q_0	0	0	$(q_0, 00), (q_1, 0)$
4	q_0	1	0	$(q_0, 10), (q_1, 0)$
5	q_0	0	1	$(q_0, 01), (q_1, 1)$
6	q_0	1	1	$(q_0, 11), (q_1, 1)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

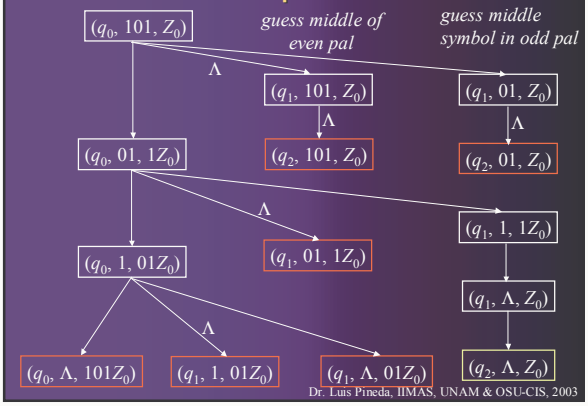
More than one next state *Pal*

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0), (q_1, Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0), (q_1, Z_0)$
3	q_0	0	0	$(q_0, 00), (q_1, 0)$
4	q_0	1	0	$(q_0, 10), (q_1, 0)$
5	q_0	0	1	$(q_0, 01), (q_1, 1)$
6	q_0	1	1	$(q_0, 11), (q_1, 1)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

Λ-Transition

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0), (q_1, Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0), (q_1, Z_0)$
3	q_0	0	0	$(q_0, 00), (q_1, 0)$
4	q_0	1	0	$(q_0, 10), (q_1, 0)$
5	q_0	0	1	$(q_0, 01), (q_1, 1)$
6	q_0	1	1	$(q_0, 11), (q_1, 1)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

Accept 101



A machine for accepting Pal_{even}

- The language:

$$Pal_{even} = \{xx^r \mid x \in \{a, b\}^*\}$$

- Define $M_{pal\text{-}even}$:

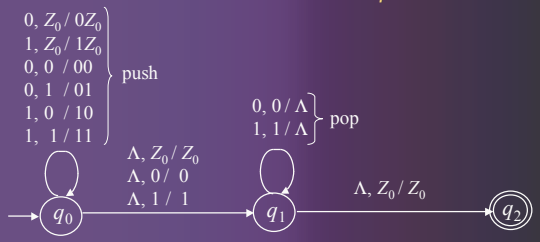
$$M_{pal\text{-}even} = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\}, q_0, Z_0, \{q_2\}, \delta)$$

- Only one kind of non-determinism:
  - There are states in which the machine has the choice of consuming an input symbol or make a Λ-transition

Transition function for $M_{pal\text{-}even}$

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0)$
3	q_0	0	0	$(q_0, 00)$
4	q_0	1	0	$(q_0, 10)$
5	q_0	0	1	$(q_0, 01)$
6	q_0	1	1	$(q_0, 11)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

Graphical rep. of $M_{pal\text{-}even}$



- A label on an arc of form $a, X/\alpha$ from p to q means that on the input a, the top of the stack X is substituted by α:

$$\delta(p, a, X) = (q, \alpha)$$

A choice of Λ-Transition

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0)$
3	q_0	0	0	$(q_0, 00)$
4	q_0	1	0	$(q_0, 10)$
5	q_0	0	1	$(q_0, 01)$
6	q_0	1	1	$(q_0, 11)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

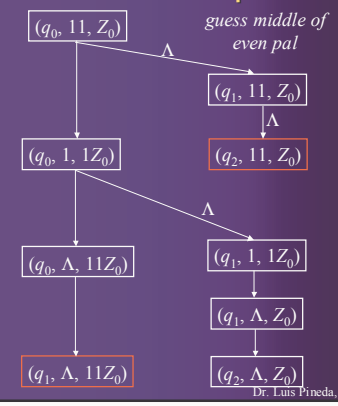
Transition function for $M_{pal\text{-}even}$

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0)$
3	q_0	0	0	$(q_0, 00)$
4	q_0	1	0	$(q_0, 10)$
5	q_0	0	1	$(q_0, 01)$
6	q_0	1	1	$(q_0, 11)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

Transition function for $M_{pal\text{-}even}$

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0)$
3	q_0	0	0	$(q_0, 00)$
4	q_0	1	0	$(q_0, 10)$
5	q_0	0	1	$(q_0, 01)$
6	q_0	1	1	$(q_0, 11)$
7	q_0	Λ	Z_0	(q_1, Z_0)
8	q_0	Λ	0	$(q_1, 0)$
9	q_0	Λ	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

Accept 11



A machine for accepting $M_{pal\text{-}mark}$

- The language:

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

- Define $M_{pal\text{-}mark}$:

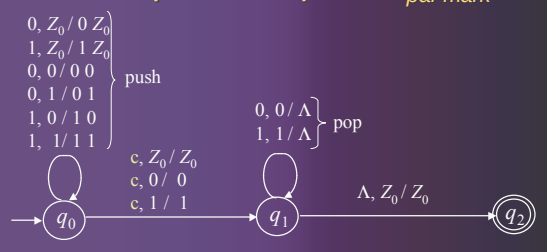
  $M_{pal\text{-}mark} = (\{q_0, q_1, q_2\}, \{0, 1\}, \{0, 1, Z_0\},$
  $q_0, Z_0, \{q_2\}, \delta)$

- No non-determinism at all!

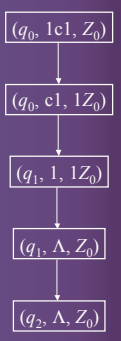
Transition function for $M_{pal\text{-}mark}$

Id	State	Input	Stack symbol	Move(s)
1	q_0	0	Z_0	$(q_0, 0Z_0)$
2	q_0	1	Z_0	$(q_0, 1Z_0)$
3	q_0	0	0	$(q_0, 00)$
4	q_0	1	0	$(q_0, 10)$
5	q_0	0	1	$(q_0, 01)$
6	q_0	1	1	$(q_0, 11)$
7	q_0	c	Z_0	(q_1, Z_0)
8	q_0	c	0	$(q_1, 0)$
9	q_0	c	1	$(q_1, 1)$
10	q_1	0	0	(q_1, Λ)
11	q_1	1	1	(q_1, Λ)
12	q_1	Λ	Z_0	(q_2, Z_0)
Other combinations				non

A Graphical Rep. of $M_{pal\text{-}mark}$



Accept 1c1



Definition of Deterministic PDA

- Let $M = (Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ be a PDA. M is deterministic if there is no configuration for which M has a choice of more than one move. For this, M has to satisfy two conditions:
  1. For any $q \in Q$, $a \in \Sigma \cup \{\Lambda\})$ and $X \in \Gamma$, the set $\delta(q, a, X)$ has at most one element
  2. For any $q \in Q$ and $X \in \Gamma$, if $\delta(q, \Lambda, X) \neq \Phi$, then $\delta(q, a, X) = \Phi$ for every $a \in \Sigma$

  A language L is a *deterministic context-free language* (DCFL) if there is a deterministic PDA (DPDA) accepting L

Definition of Deterministic PDA

- Let $M = (Q, \Sigma, \Gamma, q_0, Z_0, A, \delta)$ be a PDA. M is deterministic if there is no configuration for which there is:
  - No more than one move from one configuration
  - No choice between consume a symbol and make a Λ-transition

Causes of non-determinisms

- In the language?
  - $L_{pal} = \{x \mid x = x^r \in \{a, b\}^*\}$
  - $Pal_{even} = \{xx^r \mid x \in \{a, b\}^*\}$
  - $Pal_{mark} = \{xcx^r \mid x \in \{a, b\}^*\}$
- The description of languages with complex structure requires the use of powerful expressive devices in the grammar:
  - Λ
  - Choice of productions
  - Ambiguity?

Causes of non-determinisms

- In the grammar
  - $G_{pal} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid 0 \mid 1 \mid \Lambda\}$
  - $G_{pal\text{-}even} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid \Lambda\}$
  - $G_{pal\text{-}mark} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid c\}$
- What production was used to generate a symbol in a particular derivation?
  - 001100 $P \rightarrow 0P0$ or $P \rightarrow 0$?
  - 001Λ100 $P \rightarrow \Lambda$ is used, but when?

Removing non-determinism?

- Removing Λ-productions:
  - $G_{pal} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid 0 \mid 1 \mid 00 \mid 11\}$
  - $G_{pal\text{-}even} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid 00 \mid 11\}$
- This is not enough!
  - For a given left-side variable we still have different right-sides producing the same next symbol in a derivation!

    $P \rightarrow 0P0 \mid 0 \mid 00$
  - During interpretation we cannot tell what production was in used (in generation) relying only in local information: interpretation state, input symbol and symbol on top of the stack

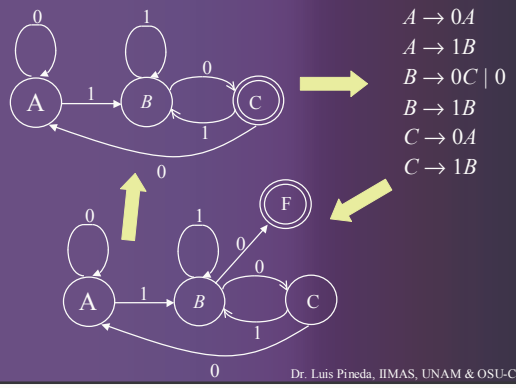
The PDA for Pal_{mark} is deterministic!

- $Pal_{mark} = \{xcx^r \mid x \in \{a, b\}^*\}$
  - $G_{pal\text{-}mark} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid c\}$
- No productions induce non-determinism:
  - $P \rightarrow 0P0 \mid 1P1$: One push and one pop!
  - $P \rightarrow c$: Reaching the first half
- No production with the same left-side variable produce the same next symbol in the right-side!

DPDA and Regular languages

- Regular grammars have a corresponding NFA
- NFA has an equivalent FA
- FA is deterministic
- A DPDA can simulate deterministic FA
  - Define a PDA that only uses its states (i.e. it has a stack and stack symbol, but these are not used!)
  - If $A = (Q, \Sigma, q_0, A, \delta_A)$ is a FA construct a PDA
    $P = (Q, \Sigma, \Gamma, q_0, Z_0, A, \delta_P)$
    such that
    $\delta_P(q, a, Z_0) = \{(p, Z_0)\}$
    for all states p and q in Q such that $\delta_A(q, a) = p$

DPDA and Regular languages

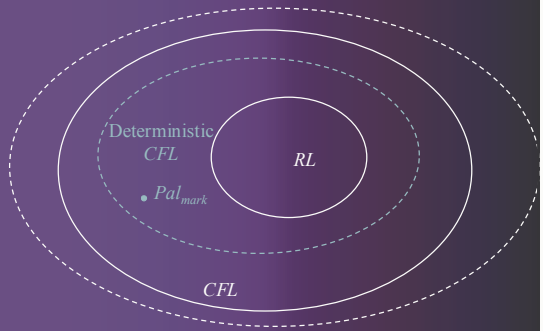


$A \rightarrow 0A$
$A \rightarrow 1B$
$B \rightarrow 0C \mid 0$
$B \rightarrow 1B$
$C \rightarrow 0A$
$C \rightarrow 1B$

Deterministic *CFL*

- Pal_{mark} has is a *DCFL* (has a corresponding DPDA)
- Pal_{mark} is not a *RL*:
  - The pumping lemma: Consider $w = 0^n c 0^n$: choose uv two groups of 0's of the first half: pump down the v group, and the resulting string is not in the language!
- On the other hand, *Pal and* Pal_{even} are *CFL* for which there is no DPDA
- The set of *deterministic CFL* properly include *RL* and is properly included in *CFL*

There is a class of Deterministic CFL

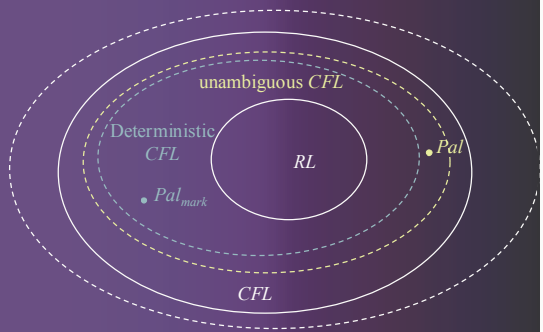


- Is the class of Deterministic *CFL* equivalent with the class of unambiguous *CFL*?

Deterministic and ambiguous CFL

- $L_{pal} = \{x \mid x = x^r \in \{a, b\}^*\}$
  - $G_{pal} = (\{P\}, \{0, 1\}, P, P \rightarrow 0P0 \mid 1P1 \mid 0 \mid 1 \mid 00 \mid 11\}$
- PDA for G_{pal} is Non-deterministic!
- G_{pal} is unambiguous: Leftmost derivations are unique!
- Several derivations but one tree!
  - We might make wrong guesses along the way (which eventually will die before reaching the accepting state)
  - There may be several derivations (e.g. rightmost, leftmost)
  - But there is only one structure for the input string!

Deterministic and ambiguous CFL



- Is the class of Deterministic CFL equivalent with the class of unambiguous CFL: NO!