

Lecture 17 – Deterministic Pushdown Automata (DPDA)

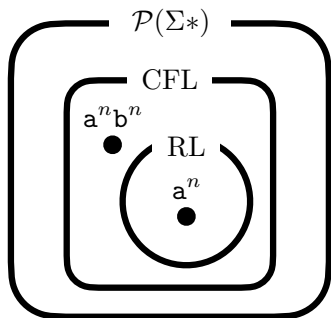
COSE215: Theory of Computation

Jihyeok Park



2026 Spring

- A **pushdown automaton (PDA)** is an extension of ϵ -NFA with a **stack**. Thus, PDA is **non-deterministic**.
 - Acceptance by **final states**
 - Acceptance by **empty stacks**
- Then, how about **deterministic PDA (DPDA)**?
- What is the **language class** of DPDA? Still, CFL?



PDA_{FS}
 (by final states)

\equiv

PDA_{ES}
 (by empty stacks)

\equiv

CFG

1. Deterministic Pushdown Automata (DPDA)

2. Deterministic Context-Free Languages (DCFLs)

Fact 1: $\text{DCFL} \subsetneq \text{CFL}$

Fact 2: $\text{RL} \subsetneq \text{DCFL}$

3. Languages Accepted by Empty Stacks of DPDA (DCFL_{ES})

Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

Fact 4: $\text{DCFL}_{\text{ES}} = \text{DCFL}$ having Prefix Property

Fact 5: $\text{RL} \not\subseteq \text{DCFL}_{\text{ES}}$

4. Inherent Ambiguity of DCFLs

Fact 6: $\text{DCFL} \subsetneq \text{Non Inherently Ambiguous Languages}$

1. Deterministic Pushdown Automata (DPDA)

2. Deterministic Context-Free Languages (DCFLs)

Fact 1: $\text{DCFL} \subsetneq \text{CFL}$

Fact 2: $\text{RL} \subsetneq \text{DCFL}$

3. Languages Accepted by Empty Stacks of DPDA (DCFL_{ES})

Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

Fact 4: $\text{DCFL}_{\text{ES}} = \text{DCFL}$ having Prefix Property

Fact 5: $\text{RL} \not\subseteq \text{DCFL}_{\text{ES}}$

4. Inherent Ambiguity of DCFLs

Fact 6: $\text{DCFL} \subsetneq \text{Non Inherently Ambiguous Languages}$

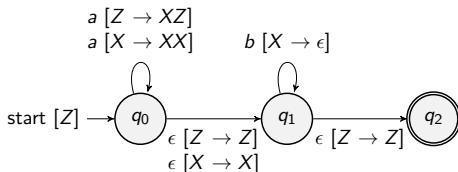
Definition (Deterministic Pushdown Automata (DPDA))

A **deterministic pushdown automaton (DPDA)** is a pushdown automaton having **at most one** one-step move (\vdash) from any configuration.

We can check it with the following conditions:

- ① $|\delta(q, a, X)| \leq 1$ for all $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, and $X \in \Gamma$.
- ② If $\delta(q, \epsilon, X) \neq \emptyset$, then $\delta(q, a, X) = \emptyset$ for all $a \in \Sigma$.

For example, is the following PDA deterministic?



No, because it has multiple transitions for (q_0, ab, Z) .

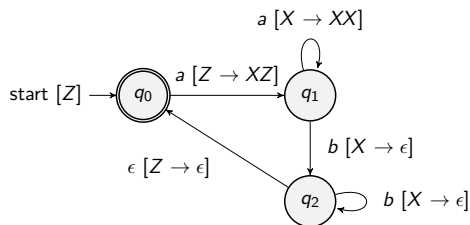
Definition (Deterministic Pushdown Automata (DPDA))

A **deterministic pushdown automaton (DPDA)** is a pushdown automaton having **at most one** one-step move (\vdash) from any configuration.

We can check it with the following conditions:

- 1 $|\delta(q, a, X)| \leq 1$ for all $q \in Q$, $a \in \Sigma \cup \{\epsilon\}$, and $X \in \Gamma$.
- 2 If $\delta(q, \epsilon, X) \neq \emptyset$, then $\delta(q, a, X) = \emptyset$ for all $a \in \Sigma$.

However, the following PDA is **deterministic**:



$$\begin{aligned}
 (q_0, aabb, Z) &\vdash (q_1, abb, XZ) \\
 &\vdash (q_1, bb, XXZ) \\
 &\vdash (q_2, b, XZ) \\
 &\vdash (q_2, \epsilon, Z) \\
 &\vdash (q_0, \epsilon, \epsilon)
 \end{aligned}$$

1. Deterministic Pushdown Automata (DPDA)

2. Deterministic Context-Free Languages (DCFLs)

Fact 1: $\text{DCFL} \subsetneq \text{CFL}$

Fact 2: $\text{RL} \subsetneq \text{DCFL}$

3. Languages Accepted by Empty Stacks of DPDA (DCFL_{ES})

Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

Fact 4: $\text{DCFL}_{\text{ES}} = \text{DCFL}$ having Prefix Property

Fact 5: $\text{RL} \not\subseteq \text{DCFL}_{\text{ES}}$

4. Inherent Ambiguity of DCFLs

Fact 6: $\text{DCFL} \subsetneq \text{Non Inherently Ambiguous Languages}$

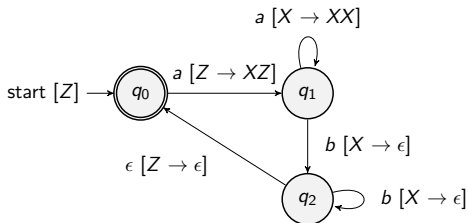
Definition (Deterministic Context-Free Languages (DCFLs))

A language L is a **deterministic context-free language (DCFL)** if and only if there exists a DPDA P such that $L = L_F(P)$ where $L_F(P)$ is the language accepted by **final states** of P .

For example, the following language is a DCFL:

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

because it is accepted by **final states** of the following **DPDA**:



Fact 1: $DCFL \subsetneq CFL$

① All DCFLs are CFLs **BUT** ② there exists a CFL that is not a DCFL.

- ① $DCFL \subseteq CFL$: By definition of DCFL, we can easily prove it.
- ② $CFL \setminus DCFL \neq \emptyset$: What is an example of a CFL that is not a DCFL?

The following language is a CFL but not a DCFL:

$$L = \{ww^R \mid w \in \{a, b\}^*\} \in CFL \setminus DCFL$$

The formal proof is complex, but we can intuitively understand it with the following two example words in L :

- $ww^R = abba \in L$ where $w = ab$
- $ww^R = abbbbba \in L$ where $w = abb$

When we read b after ab , we need to consider two possible actions:

- ① pop Y for b (for $abba$) or
- ② push Y for b (for $abbbbba$).

Fact 2: $RL \subsetneq DCFL$

① All RLs are DCFLs **BUT** ② there exists a DCFL that is not an RL.

① $RL \subseteq DCFL$: For a given RL L , consider its corresponding DFA D :

$$D = (Q, \Sigma, \delta, q_0, F)$$

Then, we can construct a DPDA P that accepts L as follows:

$$P = (Q, \Sigma, \{Z\}, \delta_P, q_0, Z, F)$$

where $\forall q \in Q. \forall a \in \Sigma. \delta_P(q, a, Z) = \{(\delta(q, a), Z)\}$ because

$$(q_0, w, Z) \vdash^* (q, \epsilon, Z) \iff \delta^*(q_0, w) = q \quad \square$$

② $DCFL \setminus RL \neq \emptyset$: We already know that the following language is a DCFL but not an RL:

$$L = \{a^n b^n \mid n \geq 0\} \in DCFL \setminus RL$$

1. Deterministic Pushdown Automata (DPDA)

2. Deterministic Context-Free Languages (DCFLs)

Fact 1: $\text{DCFL} \subsetneq \text{CFL}$

Fact 2: $\text{RL} \subsetneq \text{DCFL}$

3. Languages Accepted by Empty Stacks of DPDA (DCFL_{ES})

Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

Fact 4: $\text{DCFL}_{\text{ES}} = \text{DCFL}$ having Prefix Property

Fact 5: $\text{RL} \not\subseteq \text{DCFL}_{\text{ES}}$

4. Inherent Ambiguity of DCFLs

Fact 6: $\text{DCFL} \subsetneq \text{Non Inherently Ambiguous Languages}$

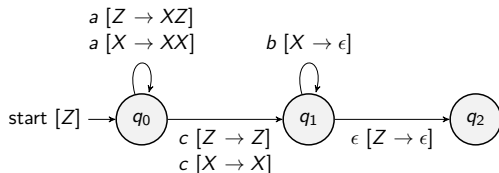
Definition (DCFL_{ES})

A language L is a **deterministic context-free language by empty stacks (DCFL_{ES})** if and only if there exists a DPDA P such that $L = L_E(P)$ where $L_E(P)$ is the language accepted by **empty stacks** of P .

For example, the following language is a DCFL_{ES}:

$$L = \{a^n cb^n \mid n \geq 0\}$$

because it is accepted by empty stacks of the following DPDA:

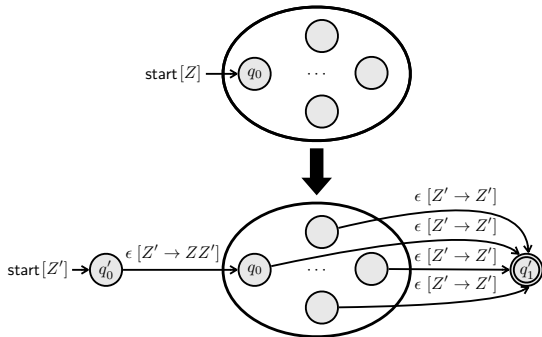


Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

① All DCFL_{ES} s are DCFLs **BUT** ② there is a DCFL but not a DCFL_{ES} .

- ① $\text{DCFL}_{\text{ES}} \subseteq \text{DCFL}$: For a given DCFL_{ES} L , consider its corresponding DPDA P that accepts L by **empty stacks**.

Then, we can construct a DPDA P' that accepts L by **final states** as:



Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

① All DCFL_{ES} s are DCFLs **BUT** ② there is a DCFL but not a DCFL_{ES} .

② $\text{DCFL} \setminus \text{DCFL}_{\text{ES}} \neq \emptyset$: The following is a DCFL but not a DCFL_{ES} :

$$L = \{a^n b^n \mid n \geq 0\} \in \text{DCFL} \setminus \text{DCFL}_{\text{ES}}$$

Why?

The DPDA needs to accept the following two words by empty stacks:

- $w = \epsilon \in L$
- $w = ab \in L$

However, if a DPDA accepts the ϵ by empty stacks, then the stack must become empty without reading any input symbols.

Thus, the PDA **cannot accept** ab by empty stacks.

We can generalize it as **prefix property** of DCFL_{ES} .

Definition (Prefix Property)

A language L has the **prefix property** if and only if for any word $w \in L$, any proper prefix of w is not in L :

$$\forall x, y \in \Sigma^*. ((xy \in L \wedge y \neq \epsilon) \implies x \notin L)$$

Fact 4: $\text{DCFL}_{\text{ES}} = \text{DCFL}$ having Prefix Property

A language L is a DCFL_{ES} if and only if the language L is a DCFL having the **prefix property**.

For example, the following language is a **DCFL** but does **NOT** have the **prefix property** because $\epsilon \in L$ is a proper prefix of

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

Thus, L is a **DCFL** but **NOT** a DCFL_{ES} .

Fact 5: $RL \not\subseteq DCFL_{ES}$

There exists a RL that is not a $DCFL_{ES}$.

- $RL \setminus DCFL_{ES} \neq \emptyset$: For example, the following language is a **RL** but does **NOT** have the **prefix property**:

$$L = \{a^n \mid n \geq 0\} \in RL \setminus DCFL_{ES}$$

because $aa \in L$ is a proper prefix of $aaaa \in L$.

Thus, L is a **RL** but **NOT** a **DCFL_{ES}**.

1. Deterministic Pushdown Automata (DPDA)

2. Deterministic Context-Free Languages (DCFLs)

Fact 1: $\text{DCFL} \subsetneq \text{CFL}$

Fact 2: $\text{RL} \subsetneq \text{DCFL}$

3. Languages Accepted by Empty Stacks of DPDA (DCFL_{ES})

Fact 3: $\text{DCFL}_{\text{ES}} \subsetneq \text{DCFL}$

Fact 4: $\text{DCFL}_{\text{ES}} = \text{DCFL}$ having Prefix Property

Fact 5: $\text{RL} \not\subseteq \text{DCFL}_{\text{ES}}$

4. Inherent Ambiguity of DCFLs

Fact 6: $\text{DCFL} \subsetneq \text{Non Inherently Ambiguous Languages}$

Definition (Inherent Ambiguity)

A language L is **inherently ambiguous** if all CFGs whose languages are L are ambiguous. (i.e. there is no unambiguous grammar for L)

What is the relationship of **inherently ambiguous languages** and DCFLs?

It satisfies the following fact:

$$\text{DCFL} \subsetneq \text{Non Inherently Ambiguous Languages}$$

We prove this fact by the following three steps:

- ① $\text{DCFL}_{ES} \subseteq \text{Non Inherently Ambiguous Languages}$
- ② $\text{DCFL} \subseteq \text{Non Inherently Ambiguous Languages}$ (using ①)
- ③ $\text{Non Inherently Ambiguous Languages} \setminus \text{DCFL} \neq \emptyset$

Fact 6: DCFL \subsetneq Non Inherently Ambiguous Languages

All DCFLs have at least one corresponding unambiguous grammars **BUT** there exists a non inherently ambiguous language that is not a DCFL.

- ① A DCFL_{ES} has an unambiguous grammar: For a given DCFL_{ES} L and its corresponding DPDA P , we can define a CFG for P as follows:

- For all $0 \leq j < n$,

$$S \rightarrow A_{0,j}^Z$$

- For all transition $(q_j, X_1 \cdots X_m) \in \delta(q_i, a, X)$ where $q_i \in Q$, $a \in \Sigma \cup \{\epsilon\}$, and $X \in \Gamma$ and combinations $0 \leq k_1, \dots, k_m < n$:

$$A_{i,k_m}^X \rightarrow a A_{j,k_1}^{X_1} A_{k_1,k_2}^{X_2} \cdots A_{k_{m-1},k_m}^{X_m}$$

For any word $w \in L$, w has a unique moves from the initial configuration to the final configuration in P . And, we know that:

$$A_{i,j}^X \Rightarrow^* w \quad \text{if and only if} \quad (q_i, w, X) \vdash^* (q_j, \epsilon, \epsilon)$$

Since the leftmost derivation of w simulates the unique moves of w in P , there is only one leftmost derivation for w . Thus, CFG is **unambiguous**.

Fact 6: DCFL \subsetneq Non Inherently Ambiguous Languages

All DCFLs have at least one corresponding unambiguous grammars **BUT** there exists a non inherently ambiguous language that is not a DCFL.

- ② A DCFL has an unambiguous grammar: For a given DCFL L , we can define another DCFL L' with a special symbol $\$$ as follows:

$$L' = L\$ = \{w\$ \mid w \in L\}$$

Then, L' is a DCFL_{ES} because it has the **prefix property**, and L' has an **unambiguous grammar** G' . Now, we can define an **unambiguous grammar** G for L by treating $\$$ as a variable with a rule $\$ \rightarrow \epsilon$.

For example, $L = \{a^n b^n \mid n \geq 0\}$ is DCFL, then $L' = \{a^n b^n \$ \mid n \geq 0\}$ is a DCFL_{ES} and its **unambiguous grammar** G' is:

$$S \rightarrow X\$ \quad X \rightarrow aXb \mid \epsilon$$

Then, the **unambiguous grammar** G for L is:

$$S \rightarrow X\$ \quad X \rightarrow aXb \mid \epsilon \quad \$ \rightarrow \epsilon$$

Fact 6: DCFL \subsetneq Non Inherently Ambiguous Languages

All DCFLs have at least one corresponding unambiguous grammars **BUT** there exists a non inherently ambiguous language that is not a DCFL.

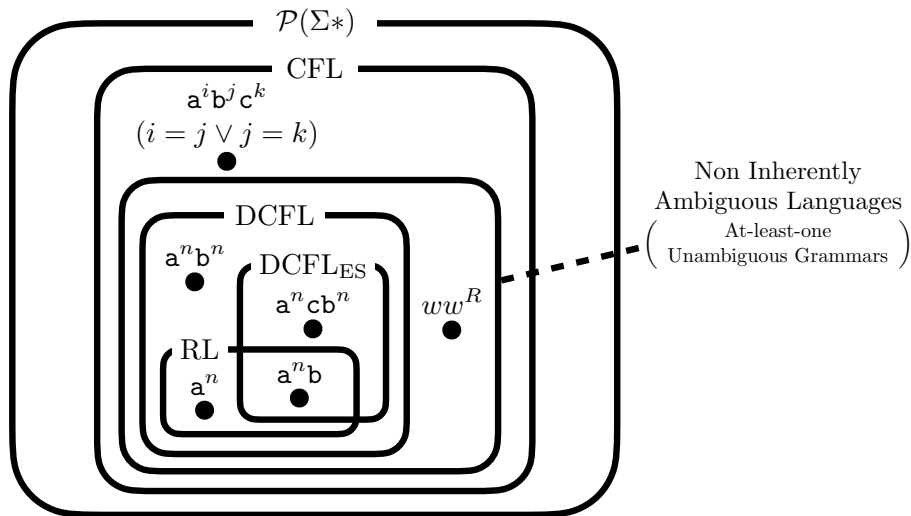
- ③ Non Inherently Ambiguous Languages \setminus DCFL $\neq \emptyset$: The following language is a **non inherently ambiguous language** but **not** a **DCFL**:

$$L = \{ww^R \mid w \in \{a, b\}^*\}$$

because the following **unambiguous grammar** G represents L :

$$S \rightarrow aSa \mid bSb \mid \epsilon$$

but we already know that L is **not** a **DCFL**.



- Please see this document on GitHub:

<https://github.com/ku-plrg-classroom/docs/tree/main/cose215/pda-examples>

- The due date is 23:59 on May 18 (Mon.).
- Please only submit `Implementation.scala` file to [LMS](#).

- Normal Forms of Context-Free Grammars

Jihyeok Park

jihyeok_park@korea.ac.kr

<https://plrg.korea.ac.kr>