Theoretical Computer Science

Lab Session 10

April 08, 2021

Agenda

- Non-determinism (cont.):
  - PDA
  - TM

Non-deterministic PDA.

Non-deterministic Pushdown Automaton (NDPDA)

Definition: NDPDA

A NDPDA is a tuple $\langle Q, I, \Gamma, \delta, q_0, Z_0, F\rangle$, where $Q, I, \Gamma, q_0, Z_0, F$ are defined as in (D)PDA and the transition function is defined as

$$\delta : Q \times (I \cup \{\epsilon\}) \times \Gamma \to \mathbb{P}_{\mathtt{F}}(Q \times \Gamma^*)$$

where $\mathbb{P}_{\mathtt{F}}$ indicates finite subsets.

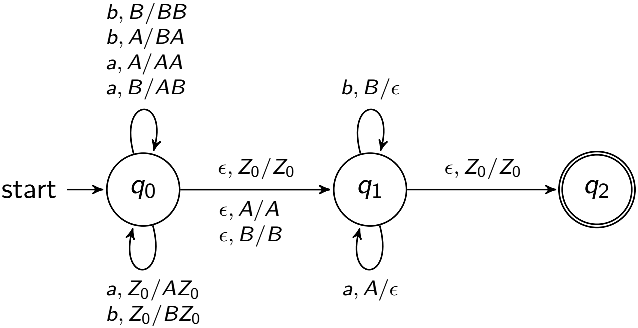
Exercises

Build NDPDAs that recognise the following languages:

1. $L_1 = \{ww^R \mid w \in \{a, b\}^*\}$ where w^R is the reversed string w.
2. $L_2 = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$.
3. The language of well-parenthesised strings (including empty string). E.g. a string in the language: (()()))(), a string that does not belong to the language: (()()(). The alphabet is $I = \{\text{“(”}, \text{“)”}\}$.
4. $L_4 = \{w \in \{a, b\}^* \mid \phi(w, a) = \phi(w, b)\}$ where $\phi(s, c)$ is the number of occurrences of the character c in the string s.

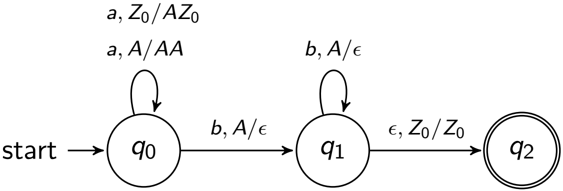
Solution (1)

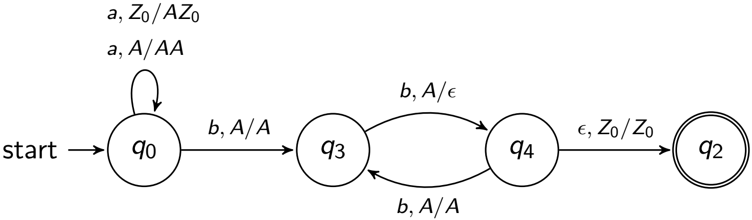
NDPDA accepting $L_1 = \{ww^R \mid w \in \{a, b\}^*\}$ where w^R is the reversed string w.



Solution (2)

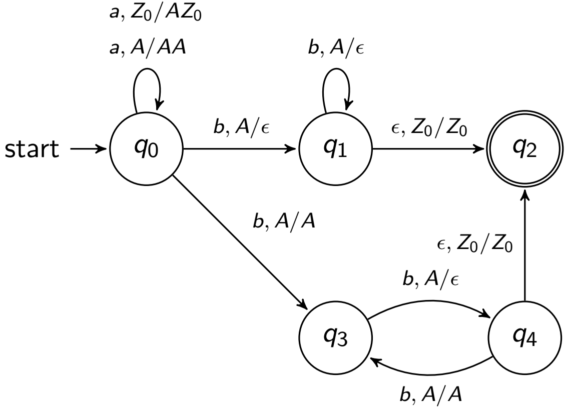
PDA accepting $\{a^n b^n \mid n \geq 1\}$ and $\{a^n b^{2n} \mid n \geq 1\}$.





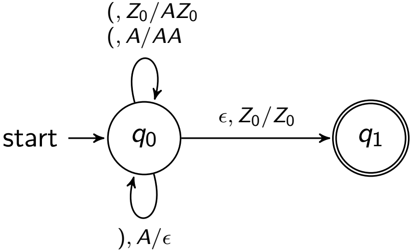
Solution (2)

NDPDA accepting $L_2 = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$.



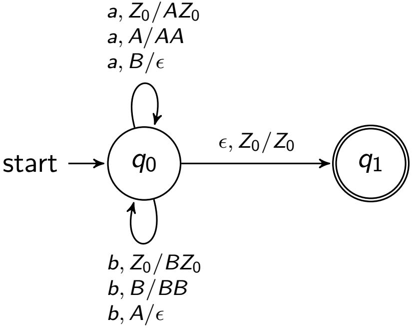
Solution (3)

NDPDA accepting The language of well-parenthesised strings (including empty string). E.g. a string in the language: (()())(), a string that does not belong to the language: (()()() – the alphabet is $I = \{\text{“(”}, \text{“)”}\}$.



Solution (4)

NDPDA accepting the language $L_4 = \{w \in \{a,b\}^* \mid \phi(w,a) = \phi(w,b)\}$ where $\phi(s,c)$ is the number of occurrences of the character c in the string s.

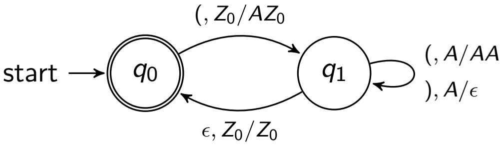


NDPDA & (D)PDA

NDPDAs are more powerful than (D)PDA. Let's try to build (D)PDAs that recognise the languages previously defined. (D)PDA accepting the language of well-parenthesised strings (including empty string). E.g. a string in the language: (()())(), a string that does not belong to the language: (()()() – the alphabet is $I = \{\text{“(”}, \text{“)”}\}$.

NDPDA & (D)PDA

NDPDAs are more powerful than (D)PDA. Let's try to build (D)PDAs that recognise the languages previously defined. (D)PDA accepting the language of well-parenthesised strings (including empty string). E.g. a string in the language: (()())(), a string that does not belong to the language: (()()() – the alphabet is $I = \{\text{"("}, \text{")"}\}$. We did it in one of the previous lab. sessions:

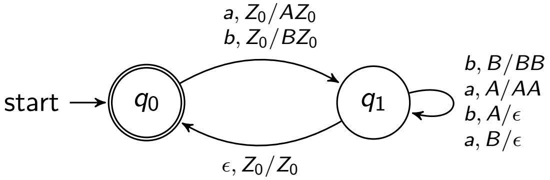


NDPDA & (D)PDA (2)

(D)PDA accepting the language $L_4 = \{w \in \{a, b\}^* \mid \phi(w, a) = \phi(w, b)\}$ where $\phi(s, c)$ is the number of occurrences of the character c in the string s.

NDPDA & (D)PDA (2)

(D)PDA accepting the language
$L_4 = \{w \in \{a, b\}^* \mid \phi(w, a) = \phi(w, b)\}$ where $\phi(s, c)$ is the number of occurrences of the character c in the string s.
We did it in one of the previous lab. sessions:



Exercises (2)

What about (D)PDAs accepting the languages

1. $L_1 = \{ww^R \mid w \in \{a, b\}^*\}$ where w^R is the reversed string w.
2. $L_2 = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$.

Exercises (2)

What about (D)PDAs accepting the languages

1. $L_1 = \{ww^R \mid w \in \{a,b\}^*\}$ where w^R is the reversed string w.
2. $L_2 = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$.

These languages cannot be accepted by any (D)PDA.

Exercises (2)

What about (D)PDAs accepting the languages

1. $L_1 = \{ww^R \mid w \in \{a, b\}^*\}$ where w^R is the reversed string w.
2. $L_2 = \{a^n b^n \mid n \geq 1\} \cup \{a^n b^{2n} \mid n \geq 1\}$.

These languages cannot be accepted by any (D)PDA.
Homework: Show that language L_1 and L_2 cannot be accepted by a (D)PDA.

Non-deterministic TM.

Non-deterministic Turing Machine (NDTM)

To define a NDTM, we need to change the transition function (all the other elements remain as in a (D)TM):

Definition: NDTM

A NDTM is a tuple $\langle Q, I, \Gamma, \delta, q_0, Z_0, F \rangle$, where $Q, I, \Gamma, q_0, Z_0, F$ are defined as in (D)TM and the transition function is defined as

$$\delta : (Q - F) \times (I \cup \{\_\}) \times (\Gamma \cup \{\_\})^k \to \mathbb{P}\left(Q \times (\Gamma \cup \{\_\})^k \times \{R, L, S\}^{k+1} \right)$$

Acceptance: Among the various possible runs (with the same input) of the NDTM, it is sufficient that one of the run (instances) reaches the final state. The input string may not be fully consumed to be considered as 'accepted'.

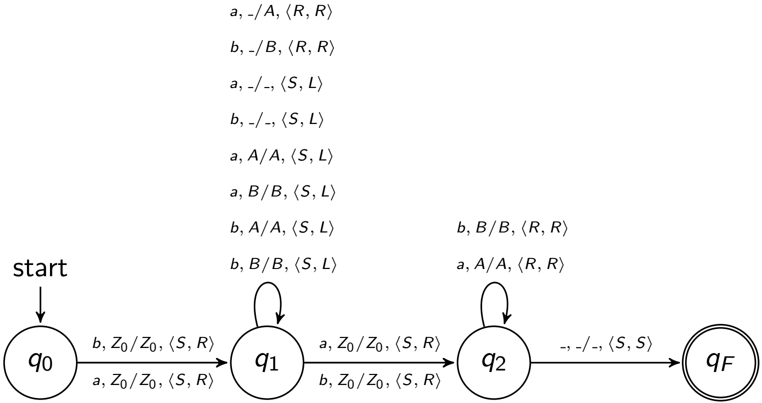
Exercises

Build NDTMs that recognise the following languages:

- $L_1 = \{ww \mid w \in \{a, b\}^+\}$
- $L_2 = \{ww^R \mid w \in \{a, b\}^+\}$, where w^R is the reversed string w
- $L_3 = \{a^n b^n | n \geq 0\} \cup \{a^n b^{2n} | n \geq 0\}$ (the homework)

Solution (1)

TM that recognises the language $L_1 = \{ww \mid w \in \{a, b\}^+\}$



Solution (2)

TM that recognises the language $L_2 = \{ww^R \mid w \in \{a, b\}^+\}$, where w^R is the reversed string w