Recapitulation:

Büchi pushdown systems (BPDS) is

$BP = (Q, \Gamma, \rightarrow, Q_F)$

where

- $(Q, \Gamma, \rightarrow)$ is a PDS
- $Q_F \subseteq Q$ set of final states.

Accepting runs

$r = (q_0, w_0) \rightarrow (q_1, w_1) \rightarrow \ldots$ with $q_i \in Q_F$ for infinitely many $i \in \mathbb{N}$.

Goal: Solve accepting run problem

Compute set of all configurations c so that BP has an accepting run from c.

Following proposition relates accepting runs to reachability in PDS

Proposition:

Let c a configuration of a BPDS $BP = (Q, \Gamma, \rightarrow, Q_F)$

Then BP has an accepting run starting from c

if and only if

there are configurations (q, γ), (q_F, u), $(q, \gamma . v)$ with $q_F \in Q_F$ so that

(1) $c \rightarrow^* (q, \gamma w)$ for some $w \in \Gamma^*$

(2) $(q, \gamma) \rightarrow^+ (q_F, u) \rightarrow^* (q, \gamma . v)$

Note the beauty:

Statement about emptiness (language / set-theoretic) turned into algorithmic problem via mathematical reasoning.

Proof:

$\Rightarrow$ Let

$$r: c = c_0 \to c_1 \to \dots$$

an accepting run of D.P. Denote by $(c_k)_{k \in \mathbb{N}}$.

Let

$$r^i = c_i \to c_{i+1} \to \dots$$

suffix of r starting in c_i.

Define

$$len(c) = \text{length of stack content in } c.$$

Define

$$m^i = \text{minimal length of stack content in } r^i.$$

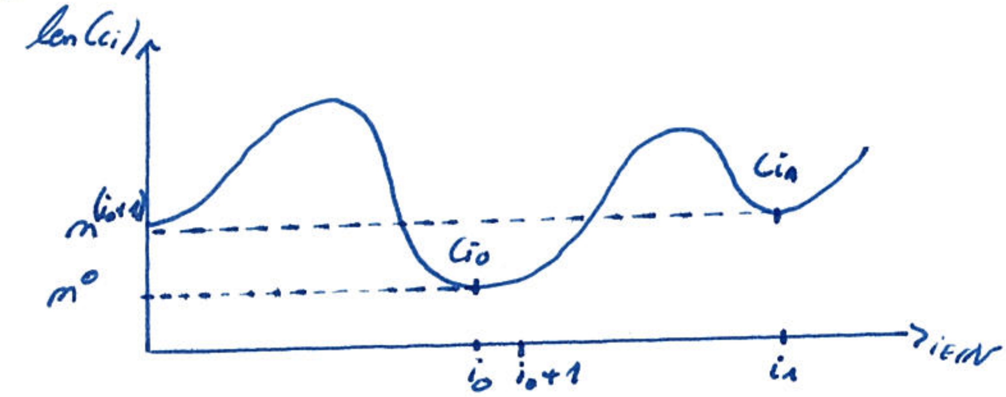
Has to grow as sequences get shorter, $m^i \leq m^j$ for $i \leq j$.

Build subsequence $(c_{i_k})_{k \in \mathbb{N}}$:

c_{i_0} = first configuration of length m_0.

$c_{i_{(k+1)}}$ = first configuration of $r^{(i_k+1)}$ of length $m^{(i_k+1)}$, $k \in \mathbb{N}$.

Illustration:



As the number of states and the number of stack symbols is finite, there is a subsequence

$$(c_{j_k})_{k \in \mathbb{N}} \text{ of } (c_{i_k})_{k \in \mathbb{N}}$$

so that all elements in $(c_{j_k})_{k \in \mathbb{N}}$ have

- same state and
- same topmost stack symbol.

As the run is accepting, we find $l \in \mathbb{N}$ so that
between c_{j_0} and c_{j_l} there is a "final" configuration
$$c_F = (q_F, u) \quad \text{with} \quad q_F \in Q_F$$
in $(c_k)_{k \in \mathbb{N}}$.

Moreover, we can pick l so as to guarantee
$$c \to^+ c_F.$$

To investigate the shape of c_{j_0}, c_F, and c_{j_l}
let $c_{j_0} = (q, \gamma.w)$.

As (c_{j_k}) was chosen minimal, stack content w
is never changed between c_{j_0} and c_{j_l}.

In particular
$$c_F = (q_F, u) = (q_F, u'.w) \quad \text{for some } u' \in \Gamma^*$$
$$c_{j_l} = (q, \gamma.v.w).$$

By construction:
$$c \to^* c_{j_0} \quad \text{and} \quad (q, \gamma) \to^+ (q_F, u') \to^* (q, \gamma.v).$$

"$\Leftarrow$" By (1):
$$c \to (q, \gamma.w).$$
By (2):
$$(q, \gamma.v^i.w) \to^+ (q_F, u.v^i.w) \to^* (q, \gamma.v^{i+1}.w) \quad \text{f.a. } i \in \mathbb{N}$$
This yields an accepting run. $\square$

To check existence of an accepting run algorithmically,
reformulate (1) and (2):

(1') $c \in pre^*(\{q\} \times \gamma\Gamma^*)$

(2') $(q, \gamma) \in pre^+((Q_F \times \Gamma^*) \cap pre^*(\{q\} \times \gamma\Gamma^*))$

Theorem (Bouajjani, Esparza, Maler '97):
The accepting run problem of BPDS can be solved in polynomial time.

Algorithm:

- Find all configurations (q, γ) for which (2') holds. ($|Q| \times |\Gamma|$ many)

  How?

  ↳ Construct BP-NFA for $pre^*(\{q\} \times \gamma\Gamma^*)$

  ↳ $\underbrace{\text{Intersect with } Q_F \times \Gamma^*}$

  Keep those stack contents from s_{q_F} with $q_F \in Q_F$.

  ↳ Compute $pre^*((Q_F \times \Gamma^*) \cap pre^*(\{q\} \times \gamma\Gamma^*))$

  ↳ Compute another single pre:

  $$pre(pre^*((Q_F \times \Gamma^*) \cap pre^*(\{q\} \times \gamma\Gamma^*)))$$
  $$= pre^+((Q_F \times \Gamma^*) \cap pre^*(\{q\} \times \gamma\Gamma^*)).$$

  ↳ Check $(q, \gamma) \in pre^+((Q_F \times \Gamma^*) \cap pre^*(\{q\} \times \gamma\Gamma^*))$.

- For all (q, γ) that satisfy (2'):

  ↳ Compute $pre^*(q, \gamma\Gamma^*)$

- Take union of all these sets $pre^*(q, \gamma\Gamma^*)$.

Note:

If we have an initial configuration c_I, emptiness of BP with initial configuration c_I reduces to accepting run problem.

Check $c_I \in CF(A)$

where $CF(A)$ is the union computed above.