3.6. Decision Problems

Theorem 3.28

The following Emptiness Problem is decidable in polynomial time:

INSTANCE: A context-free grammar G.
QUESTION: Is $L(G) \neq \emptyset$?

Proof.

Let $G = (N, T, S, P)$ be a context-free grammar. In polynomial time we can determine the set V_{term} of usefull nonterminals of G (Lemma 3.3), where

$$V_{\mathrm{term}} = \{ A \in N \mid L(G, A) \neq \emptyset \}.$$

As $L(G) \neq \emptyset$ iff $S \in V_{\mathrm{term}}$, this yields the desired algorithm. □

Theorem 3.29

The following Finiteness Problem is decidable:

INSTANCE: A context-free grammar G.
QUESTION: Is $L(G)$ a finite language?

Proof.

First we transform the given grammar into a grammar $G_1 = (N, T, S, P)$ in CNF s.t. $L(G_1) = L(G) \cap T^+$ (Theorem 3.9).

In addition, we can assume that G_1 is proper and that it contains no ε-productions.

From G_1 we construct a directed graph (V, E) as follows:

- $V := N$, that is, there exists a node for each nonterminal, and
- $(A \to B) \in E$ iff there exists a nonterminal C such that P contains the production $A \to BC$ or $A \to CB$ (or both).

Proof of Theorem 3.29 (cont.)

Claim:

The language $L(G)$ is infinite iff the graph (V, E) contains a cycle.

Proof.

If $A_0, A_1, \ldots, A_m, A_0$ is a cycle in (V, E), then we have a derivation of the following form in G_1:

$$\begin{aligned} A_0 &\to_P \alpha_1 A_1 \beta_1 \to_P \alpha_1 \alpha_2 A_2 \beta_2 \beta_1 \to_P \cdots \to_P \alpha_1 \cdots \alpha_m A_m \beta_m \cdots \beta_1 \\ &\to_P \alpha_1 \cdots \alpha_m \alpha_{m+1} A_0 \beta_{m+1} \cdots \beta_1, \end{aligned}$$

where $\alpha_i, \beta_i \in N^*$ and $|\alpha_i| + |\beta_i| = 1$ for all $i = 1, 2, \ldots, m+1$.

As G_1 is proper, $S \to_P^* u_0 A_0 v_0$ for some $u_0, v_0 \in T^*$
and $\alpha_i \to_P^* u_i \in T^*$, $\beta_i \to_P^* v_i \in T^*$, $i = 1, 2, \ldots, m+1$.
As G_1 contains no ε-production, it follows that $|u_i v_i| \geq |\alpha_i| + |\beta_i| = 1$.

Proof of Theorem 3.29 (cont.)

Proof of Claim (cont.)

Thus, we obtain the following derivations in G_1:

$$\begin{aligned} S &\to_P^* u_0 A_0 v_0 \to_P^* u_0 u_1 \cdots u_{m+1} A_0 v_{m+1} \cdots v_1 v_0 \\ &\to_P^* u_0 (u_1 \cdots u_{m+1})^k A_0 (v_{m+1} \cdots v_1)^k v_0 \\ &\to_P^* u_0 (u_1 \cdots u_{m+1})^k x (v_{m+1} \cdots v_1)^k v_0 \in T^* \end{aligned}$$

for some $x \in T^*$ and all $k \geq 1$. As $|u_1 \cdots u_{m+1} v_{m+1} \cdots v_1| \geq m + 1$, all these words differ from one another, that is, $L(G)$ is infinite.

If (V, E) does not contain any cycle, then each path in each syntax tree of each derivation from S to some word $v \in T^*$ has length at most $|N| + 1$. Thus, there are only finitely many syntax trees for G, and hence, $L(G)$ is finite. □

It is decidable in time $O(|N|^2)$ whether (V, E) contains a cycle. □

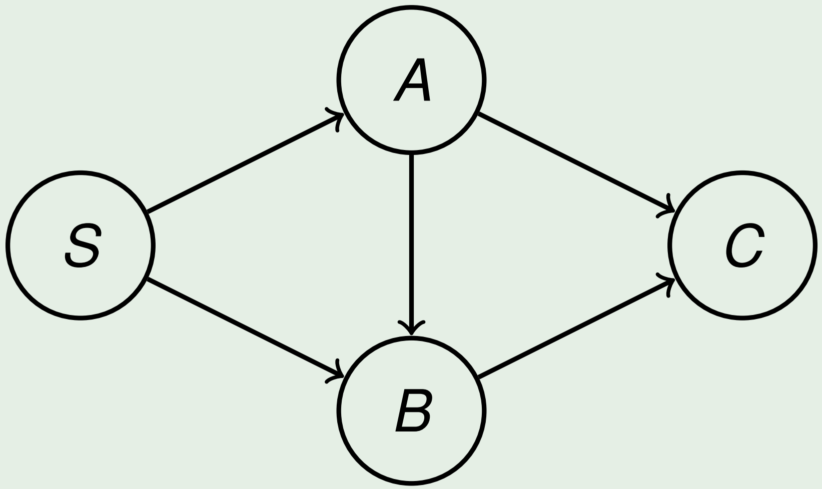
Example:

Let $G = (\{S, A, B, C, \}, \{a, b\}, S, P)$, where

$$P = \{S \to AB, A \to BC, A \to a, B \to CC, B \to b, C \to a\}.$$

G is a proper context-free grammar in CNF without ε-productions.

The directed graph(V, E) for G looks as follows:



(V, E) does not contain any cycle, that is, $L(G)$ is finite.

In fact, it is easily seen that $L(G) = \{ab, a^3, a^3b, ba^3, bab, a^5\}$. □

Theorem 3.30

The following Membership Problem is decidable in polynomial time:

INSTANCE: *A context-free grammar G in CNF, and a word $w \in T^*$.*

QUESTION: *Is $w \in L(G)$?*

Proof.

Let $G = (N, T, S, P)$ be a context-free grammar in CNF, let $N = \{A_1, A_2, \ldots, A_m\}$, and let $S = A_1$.

We can assume that $(S \to \varepsilon)$ is the only ε-production (if any), and that A_1 does not occur on the righthand side of any production.

Hence, $\varepsilon \in L(G)$ iff $(A_1 \to \varepsilon) \in P$.

Proof of Theorem 3.30 (cont.)

Let $w = x_1 x_2 \cdots x_n$, where $x_1, x_2, \ldots, x_n \in T$.

For all $i \in \{1, 2, \ldots, n\}$ and $j \in \{1, 2, \ldots, n+1-i\}$,

let $V_{i,j} \subseteq N$ be defined as follows:

$$V_{i,j} := \{ A \in N \mid A \to_P^* x_i x_{i+1} \cdots x_{i+j-1} \}.$$

Then $w \in L(G)$ iff $S = A_1 \in V_{1,n}$.

We now compute all the sets $V_{i,j}$ through the method of dynamic programming.

This algorithm goes back to J. Cocke, T. Kasami, and D. Younger (1967).

Proof of Theorem 3.30 (cont.)

procedure CKY;
begin
  for $i := 1$ **to** n **do** $V_{i,1} := \{ A \mid (A \rightarrow x_i) \in P \}$;
  for $j := 2$ **to** n **do**
    for $i := 1$ **to** $n - j + 1$ **do**
    begin
      $V_{i,j} := \emptyset$;
$(*)$ **for** $k := 1$ **to** $j - 1$ **do**
        $V_{i,j} := V_{i,j} \cup \{ A \mid (A \rightarrow BC) \in P, B \in V_{i,k}, C \in V_{i+k,j-k} \}$
    end
end.

As $A \rightarrow_P^* x_i \cdots x_{i+j-1}$ iff $\exists (A \rightarrow BC) \in P$ s.t. $B \rightarrow_P^* x_i \cdots x_{i+k-1}$ and $C \rightarrow_P^* x_{i+k} \cdots x_{i+j-1}$, the set $V_{i,j}$ is computed correctly in $(*)$.

Obviously, this algorithm runs in polynomial time. □

Example:

Let $G = (\{S, A, B, C, D, E, F\}, \{a, b, c\}, P, S)$,
where $P = \{\, S \to AB, A \to CD, A \to CF, B \to c, B \to EB,$
$C \to a, D \to b, E \to c, F \to AD \,\}$

Let $x = aaabbbcc$.

$x =$ / $j \downarrow$	a	a	a	b	b	b	c	c
1	C	C	C	D	D	D	B,E	B,E
2			A				B	
3			F					
4		A						
5		F						
6	A							
7	S							
8	S							

As $S \in V_{1,8}$, it follows that $x \in L(G)$. □

Theorem 3.31

The following problems cannot be solved algorithmically:

INSTANCE: Two context-free grammars G_1, G_2.

(1.) *QUESTION: Is $L(G_1) \cap L(G_2) = \emptyset$?*

(2.) *QUESTION: Is $|L(G_1) \cap L(G_2)| = \infty$?*

(3.) *QUESTION: Is $L(G_1) \cap L(G_2)$ context-free ?*

(4.) *QUESTION: Is $L(G_1) \subseteq L(G_2)$?*

(5.) *QUESTION: Is $L(G_1) = L(G_2)$?*

(6.) *QUESTION: Is G_1 unambiguous?*

(7.) *QUESTION: Is $L(G_1)^c$ context-free?*

(8.) *QUESTION: Is $L(G_1)$ regular?*

(9.) *QUESTION: Is $L(G_1)$ deterministic context-free?*

Proof: Later!

3.7. Deterministic Context-Free Languages

A PDA $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ is deterministic, that is, M is a DPDA, if in each configuration there is at most one applicable transition.

This is equivalent to the following two conditions, where $q \in Q$ and $z \in \Gamma$:

(1) For all $a \in \Sigma \cup \{\varepsilon\}$, $|\delta(q, a, z)| \leq 1$.
(2) If $\delta(q, \varepsilon, z) \neq \emptyset$, then $\delta(q, a, z) = \emptyset$ for all $a \in \Sigma$.

A language L is deterministic context-free, if there exists a DPDA M such that $L = L(M)$.

DCFL denotes the class of deterministic context-free languages.

Remark:

For a DPDA M, if $L' = N(M)$, that is, L' is accepted by empty pushdown, then L' is prefix-free: for all $u \in L'$, $u \cdot \Sigma^+ \cap L' = \emptyset$.

Hence, $a^+ \neq N(M)$ for each DPDA M.

Example:

$L := \{ a^m b^m a^n b^n \mid m, n \geq 0 \} \in \text{DCFL}$,
but $L \neq N(M)$ for each DPDA M.

A DPDA $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ is in normal form, if one of the following statements holds for each $\delta(q, a, z) = (p, \gamma)$:

(i) $\gamma = \varepsilon$, that is, the symbol z is popped from the pushdown,
(ii) $\gamma = z$, that is, the pushdown is not changed, or
(iii) $\gamma = zz'$ for some $z' \in \Gamma$, that is, the symbol z' is pushed onto the pushdown.

Theorem 3.32

If $L \in \text{DCFL}(\Sigma)$, then there exists a DPDA M in normal form such that $L = L(M)$.

Let M be a DPDA, and let $u \in L(M)$.
Then M reads input u completely and accepts.

For $v \in \Sigma^* \smallsetminus L(M)$, M will in general not read input v completely.
In fact, one of the following cases can occur before M has read v completely:

- M reaches a configuration to which no transition applies,
- M empties its pushdown completely,
- M enters an infinite computation consisting entirely of ε-transitions.

Lemma 3.33

For each DPDA M, there exists an equivalent DPDA M' that always reads its input completely.

Proof of Lemma 3.33.

Let $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a DPDA in normal form.

- We introduce a new pushdown symbol X_0 and a new initial state q_0' together with the transition $\delta(q_0', \varepsilon, Z_0) = (q_0, X_0 Z_0)$.

- We add a new state d s.t. $\delta(d, a, z) = (d, z)$ for all $a \in \Sigma$ and $z \in \Gamma$. If $\delta(q, a, z) \cup \delta(q, \varepsilon, z) = \emptyset$ for some $q \in Q$, $a \in \Sigma$, and $z \in \Gamma$, then we take $\delta(q, a, z) = (d, z)$.

The resulting DPDA M_1 accepts the same language as M.

If M_1 does not read an input completely, this means that, starting in some state q, M_1 executes an infinite sequence of ε-transitions without removing the topmost symbol z from the pushdown.

- In this situation we take $\delta(q, \varepsilon, z) = (d, z)$, provided that no final state is reached through this sequence of ε-steps.
If, however, a final state is reached, then we take $\delta(q, \varepsilon, z) := (e, z)$ for a new final state e and $\delta(e, \varepsilon, z) := (d, z)$.

The DPDA M' obtained in this way has the desired property. □

Remark:

The construction in the proof of Lemma 3.33 is effective.

Theorem 3.34

The language class DCFL *is closed under complementation, that is, for each* $L \in \text{DCFL}(\Sigma)$, $L^c := (\Sigma^* \smallsetminus L) \in \text{DCFL}(\Sigma)$, *too.*

Proof.

Let $M = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, F)$ be a DPDA such that $L(M) = L$, and assume that M always reads its input completely.

Unfortunately, L^c does in general not coincide with the language accepted by the DPDA $M' = (Q, \Sigma, \Gamma, \delta, q_0, Z_0, Q \smallsetminus F)$, as M may have a computation of the following form:

$$(q_0, Z_0, w) \vdash_M^* (q, \alpha, \varepsilon) \vdash_M (q', \beta, \varepsilon) \text{ for some } q \in F \text{ and } q' \notin F.$$

Proof of Theorem 3.34 (cont.)

We must ensure that the DPDA M cannot execute ε-transitions in a final state.

We define a DPDA $M_1 = (Q_1, \Sigma, \Gamma, \delta_1, q_1, Z_0, F_1)$ as follows:

– $Q_1 := \{\, [q, k] \mid q \in Q \text{ and } k \in \{1, 2, 3\} \,\}$,

– $q_1 := \left\{ \begin{array}{ll} [q_0, 1], & \text{if } q_0 \in F, \\ [q_0, 2], & \text{if } q_0 \notin F, \end{array} \right\}$, – $F_1 := \{\, [q, 3] \mid q \in Q \,\}$,

– and the transition function δ_1 is defined by:

$$\begin{array}{lll} (1) & \delta_1([q,k],\varepsilon,z) := ([p,k'],\gamma) & \text{if } \delta(q,\varepsilon,z) = (p,\gamma),\ k \in \{1,2\} \\ & & \text{and } k' = \left\{ \begin{array}{ll} 1, & \text{if } k = 1 \text{ or } p \in F, \\ 2, & \text{otherwise,} \end{array} \right. \\ (2) & \delta_1([q,2],\varepsilon,z) := ([q,3],z) & \text{if } \delta(q,a,z) = (p,\gamma), \\ (3) & \left. \begin{array}{l} \delta_1([q,1],a,z) := ([p,k],\gamma) \\ \delta_1([q,3],a,z) := ([p,k],\gamma) \end{array} \right\} & \begin{array}{l} \text{if } \delta(q,a,z) = (p,\gamma) \\ \text{and } k = \left\{ \begin{array}{ll} 1, & \text{if } p \in F, \\ 2, & \text{otherwise.} \end{array} \right. \end{array} \end{array}$$

Proof of Theorem 3.34 (cont.)

Claim:

$L(M_1) = \Sigma^* \smallsetminus L$.

Proof.

Let $u = a_1 a_2 \cdots a_n \in L$. On input u, M executes a computation of the following form:

$$(q_0, Z_0, u) \vdash_M^* (p_1, \alpha, a_n) \vdash_M (p_2, \beta, \varepsilon) \vdash_M^* (p_3, \gamma, \varepsilon) \text{ where } p_3 \in F,$$

but p_2 and the subsequent states before p_3 are from $Q \smallsetminus F$.

For M_1, we have the following computation:

$$([q_0, .], Z_0, u) \vdash_{M_1}^* ([p_1, .], \alpha, a_n) \vdash_{M_1}^{\leq 2} ([p_2, 2], \beta, \varepsilon) \vdash_{M_1}^* ([p_3, 1], \gamma, \varepsilon),$$

that is, further ε-transitions cannot lead to a final state of M_1.

Thus, $L(M_1) \subseteq \Sigma^* \smallsetminus L$.

Proof of Theorem 3.34 (cont.)

Proof of Claim (cont.)

Conversely, for $u = a_1 a_2 \cdots a_n \notin L$, M has a computation of the form:

$(q_0, Z_0, u) \vdash_M^* (p_1, \alpha, a_n) \vdash_M (p_2, \beta, \varepsilon) \vdash_M^* (p_3, \gamma, \varepsilon)$, where $p_2, \ldots, p_3 \notin F$,

and in $(p_3, \gamma, \varepsilon)$, no further ε-transition is applicable.

As M reads each input completely, there exists a letter $a \in \Sigma$ s.t. $\delta(p_3, a, \mathrm{top}(\gamma))$ is defined. Hence, M_1 executes the following computation:

$$([q_0, .], Z_0, u) \vdash_{M_1}^* ([p_2, 2], \beta, \varepsilon) \vdash_{M_1}^* ([p_3, 2], \gamma, \varepsilon) \vdash_{M_1} ([p_3, 3], \gamma, \varepsilon)$$

where $[p_3, 3] \in F_1$, that is, $\Sigma^* \smallsetminus L = L(M_1)$. □

This shows that $L^c \in \mathrm{DCFL}(\Sigma)$. □

Corollary 3.35

Each language $L \in \mathrm{DCFL}$ is accepted by a DPDA that does not execute any ε-transitions in any final state.

Corollary 3.36

The class DCFL is closed under intersection with regular languages.

Proof.

Let $L_1 \in \mathrm{DCFL}$. The there exists a DPDA M that accepts L_1 and that reads each input completely.

For $L_2 \in \mathrm{REG}$, there exsists a DFA A such that $L(A) = L_2$.

From M and A, one can construct a DPDA for $L_1 \cap L_2$, that is, $L_1 \cap L_2 \in \mathrm{DCFL}$. □

Obviously, DCFL $\subseteq$ CFL. Now we will prove that this is a proper inclusion. Let

$$L_{\mathrm{Gl}} := \{ w ¢ w^R ¢ w \mid w \in \{a, b\}^* \}$$

be the so-called Gladkij language [Gladkij 1964].

Using the Pumping Lemma 3.14 it is easily shown that L_{Gl} is not context-free.

Let $L_{\mathrm{Gl}}^c := (\{a, b, ¢\}^* \smallsetminus L_{\mathrm{Gl}})$.

Lemma 3.37

$L_{\mathrm{Gl}}^c \in \mathrm{CFL} \smallsetminus \mathrm{DCFL}$.

Proof.

If $L_{\mathrm{Gl}}^c \in \mathrm{DCFL}$, then by Theorem 3.34, $L_{\mathrm{Gl}} \in \mathrm{DCFL}$.
As $L_{\mathrm{Gl}} \notin \mathrm{CFL}$, we see that $L_{\mathrm{Gl}}^c \notin \mathrm{DCFL}$.

It remains to prove that $L_{\mathrm{Gl}}^c \in \mathrm{CFL}$.

For a word $w \in \{a, b, ¢\}^*$, we have $w \in L_{\mathrm{Gl}}^c$ iff one of the following conditions is met:

(1) $|w|_{¢} \neq 2$, or

(2) $w = w_1 ¢ w_2 ¢ w_3$, where $w_1^R \neq w_2$, or

(3) $w = w_1 ¢ w_2 ¢ w_3$, where $w_3^R \neq w_2$.

Let
$$\begin{aligned} H_1 &:= \{ u \in \{a, b, ¢\}^* \mid |w|_{¢} \neq 2 \}, \\ H_2 &:= \{ w_1 ¢ w_2 ¢ w_3 \mid w_1, w_2, w_3 \in \{a, b\}^*, w_1^R \neq w_2 \} \text{ and} \\ H_3 &:= \{ w_1 ¢ w_2 ¢ w_3 \mid w_1, w_2, w_3 \in \{a, b\}^*, w_3^R \neq w_2 \}. \end{aligned}$$

Then $H_1 \in \mathrm{REG}$ and $H_2, H_3 \in \mathrm{CFL}$, implying that $L_{\mathrm{Gl}}^c = H_1 \cup H_2 \cup H_3 \in \mathrm{CFL}$. □

Corollary 3.38

DCFL $\subsetneq$ CFL.

Each DFA can be interpreted as a DPDA that does not use its pushdown. As $\{ a^n b^n \mid n \geq 1 \} \in \text{DCFL} \setminus \text{REG}$, we obtain the following proper inclusion.

Corollary 3.39

REG $\subsetneq$ DCFL.

Theorem 3.40 (Ogden's Lemma for DCFL (see Har78))

Let $L \in \text{DCFL}(\Sigma)$. Then there exists a constant k that depends on L such that each word $z \in L$ containing $\delta(z) \geq k$ marked positions has a factorization $z = uvwxy$ that satisfies all of the following properties:

(1) $v \neq \varepsilon$,

(2) $uv^iwx^iy \in L$ for all $i \geq 0$,

(3) u, v and w or w, x and y contain marked positions,

(4) $\delta(vwx) \leq k$,

(5) if $y \neq \varepsilon$, then the following equivalence holds for all $m, n \geq 0$ and all $\alpha \in \Sigma^*$: $uv^{m+n}wx^n\alpha \in L$ iff $uv^mw\alpha \in L$.

Theorem 3.41

$L := \{ a^n b^n, a^n b^{2n} \mid n \geq 1 \} \in \text{CFL} \setminus \text{DCFL}.$

Proof.

It is easily seen that L is context-free. In fact, L is the union of the two deterministic context-free languages $L_1 := \{ a^n b^n \mid n \geq 1 \}$ and $L_2 := \{ a^n b^{2n} \mid n \geq 1 \}$.

We claim that L is not deterministic context-free.

Assume that L is deterministic context-free.

Let k be the corresponding constant from Thm. 3.40, and let $p := k!$.

Let $z := a^p b^p \in L$, where we mark all occurrences of b.

Then $z = a^p b^p$ has a factorization $z = a^p b^p = uvwxy$ that satisfies conditions (1) to (5) of the theorem.

As $z' = uwy \in L$, we have $v = a^i$ und $x = b^i$ for some $i \in \{1, 2, \ldots, k\}$.

Proof of Theorem 3.41 (cont.)

Because of (3) this implies that w, x and y contain marked positions, that is,

$$u = a^{n_1},\ v = a^i,\ w = a^{p-n_1-i}b^j,\ x = b^i \text{ and}$$
$$y = b^{p-i-j} \text{ for some } j \in \{1, 2, \ldots, k\}.$$

In fact, we have $i + j < k$. In particular, we have $y \neq \varepsilon$.

Now we choose $m = 1$ and $n = 1$, and we take $\alpha = b^{p-j+p+i}$. Then

$$uv^{n+m}wx^n\alpha = a^{n_1}a^{i\cdot 2}a^{p-n_1-i}b^jb^ib^{p-j+p+i} = a^{p+i}b^{p+i+p+i} \in L,$$

but $uv^mw\alpha = a^{n_1}a^ia^{p-n_1-i}b^jb^{p-j+p+i} = a^pb^{p+p+i} = a^pb^{2p+i} \notin L$, a contradiction!

Thus, it follows that $L \notin$ DCFL. □

Corollary 3.42

The language class DCFL *is not closed under union.*

By using the same technique it can be shown that

$$L' := \{\, a^n b^n c, a^n b^{2n} d \mid n \geq 1 \,\}$$

is not in DCFL. On the other hand, the language

$$L'^R = \{\, c b^n a^n, d b^{2n} a^n \mid n \geq 1 \,\}$$

belongs obviously to DCFL.

Corollary 3.43

The class DCFL *is not closed under reversal.*

Also $L'' := \{ ca^nb^n, da^nb^{2n} \mid n \geq 1 \}$ is in DCFL.
The morphism $\varphi : c \mapsto \varepsilon, d \mapsto \varepsilon, a \mapsto a, b \mapsto b$ maps L'' onto L.

Corollary 3.44

The class DCFL *is not closed under morphisms.*

Theorem 3.45

The class DCFL *is closed under inverse morphisms.*

Theorem 3.46

The class DCFL *is not closed under product and Kleene star.*

Theorem 3.47

The following problems are decidable:

(1) *INSTANCE:* $L \in \mathrm{DCFL}(\Sigma)$ *and* $R \in \mathrm{REG}(\Sigma)$.
QUESTION: *Is* $L = R$*?*

(2) *INSTANCE:* $L \in \mathrm{DCFL}(\Sigma)$ *and* $R \in \mathrm{REG}(\Sigma)$.
QUESTION: *Is* $R \subseteq L$*?*

(3) *INSTANCE:* $L \in \mathrm{DCFL}(\Sigma)$.
QUESTION: *Is* $L^c = \emptyset$*?*

(4) *INSTANCE:* $L \in \mathrm{DCFL}(\Sigma)$.
QUESTION: *Is* L *regular?*

(5) *INSTANCE:* $L_1, L_2 \in \mathrm{DCFL}(\Sigma)$.
QUESTION: *Is* $L_1 = L_2$*?*

Proof.

(1) Let $L_1 := (L \cap R^c) \cup (L^c \cap R)$.

Then $L = R$ iff $L_1 = \emptyset$.

From a DPDA for L and a DFA for R one can construct a PDA for L_1.
By Theorem 3.28 it is decidable whether $L_1 = \emptyset$.

(2) $R \subseteq L$ iff $L^c \cap R = \emptyset$. In analogy to (1) this is decidable.

(3) This is obvious, as by Theorem 3.28 emptiness of context-free languages is decidable.

(4) See (Stearns 1967).

(5) See (Senizergues 1997). □