Automaten en Berekenbaarheid

Eerste gekwoteerde oefenzitting 3 november 2016

1 Algebra van talen

Hieronder zijn L_1, L_2 en L_3 talen over hetzelfde alfabet Σ. Stel dat L_1 en L_2 verschillende niet-reguliere talen zijn, en dat L_3 een oneindige reguliere taal is die verschilt van Σ^*. Zijn volgende talen dan regulier, niet-regulier of hangt dit af van de talen in kwestie ? Bewijs of geef (tegen)voorbeelden.
(a) $L_1 \cap L_3$ (b) $L_1 \cup L_3$ (c) $L_1 \cup L_2$ (d) $\overline{L_1} \cup L_3$.

Antwoord Neem steeds $\Sigma = \{a, b, c\}$ in de voorbeelden.

(a) (1) neem $L_1 = \{a^n b^n\}$ en $L_3 = \{c^n\}$: hun doorsnede is leeg, dus regulier
 (2) neem $L_1 = \{a^n b^n\}$ en $L_3 = \Sigma^* \setminus \{c^n\}$: hun doorsnede is L_1 en niet regulier

(b) (1) neem $L_1 = \{a^n b^n\}$ en $L_3 = \{c^n\}$: in hun unie kunnen lange genoege strings in L_1 niet regulier in die unie gepompt worden, dus niet regulier
 (2) neem $L_1 = \{a^n b^n\}$ en $L_3 = \Sigma^* \setminus \{c^n\}$: hun unie is L_3 en regulier

(c) (1) neem $L_1 = \{a^n b^n\}$ en $L_2 = \overline{L_1}$: hun unie is Σ^* en regulier
 (2) neem $L_1 = \{a^n b^n\}$ en $L_2 = \{a^n c^n\}$: hun unie is niet regulier

(d) zie geval (b)

In elk geval hangt het van de talen af of het resultaat van de algebraische operatie regulier is of niet.

2 NFA, DFA, RegExp ...

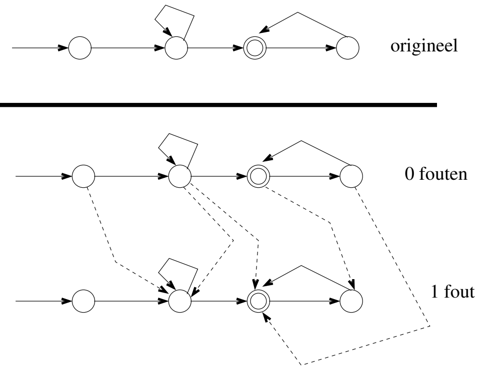
We nemen een vast alfabet met 3 tekens (maar in feite heeft het weinig belang)/
Definieer de afstand d tussen twee strings s en t van gelijke lengte als volgt: laat $s == s_1 s_2 ... s_n$ en $t == t_1 t_2 ... t_n$, dan is $d(s,t) = \#\{i | s_i \neq t_i\}$ dus het aantal posities waarop s en t verschillen.
Voor een gegeven taal L definiëren we nu de taal: $fout_1(L) = \{s | \exists t \in L \ zodat\ |s| == |t| \ en\ d(s,t) \leq 1\}$
Informeel is dat de taal met strings met hoogstens één foutje t.o.v. een string in L.
Bewijs nu het volgende: als L regulier is, dan is $fout_1(L)$ ook regulier.
Je kan misschien van een reguliere expressie voor L en een reguliere expressie voor $fout_1(L)$ maken, of vertrekken van een DFA (of NFA) voor L; je mag ook het concept *transducer* gebruiken, en/of de vele stellingen/lemmas uit de cursus.

Antwoord Gegeven een DFA $(Q, \Sigma, \delta, q_s, F)$ voor L maken we een NFA $((Q_1, \Sigma, \delta_1, q_{s1}, F_1)$ voor $fout_1(L)$. De NFA heeft geen ϵ-bogen, wel keuzes. We kiezen de DFA zo dat δ totaal is.

- $Q_1 = Q \times \{0, 1\})$
- $q_{s1} = (q_s, 0)$
- $F_1 = (F \times \{0, 1\})$
- definieer δ_1 als volgt:
  $\forall x \in \Sigma : \delta_1((q, 0), x) = \{(\delta(q, x), 0)\} \cup \{(\delta(q, z), 1) | z \in \Sigma\}$
  $\forall y \in \Sigma : \delta_1((q, 1), x) = \{(\delta(q, x), 1)\}$

Informeel: we maken een copie van de eerste DFA; de eerste heet *nul fouten*, de copie heet *hoogstens één fout*. Binnenin die twee DFAs gaan de bogen zoals in het origineel, maar in de nul-fouten-DFA maken we ook overgangen naar de 1-fout-DFA voor elk teken (vandaar het niet-determinisme). Eens in de 1-fout-DFA gelden de regels van de originele DFA.

Hieronder een voorbeeldje: op elke boog in stippellijn staan alle symbolen uit Σ^*.



Je moet nog wel argumenteren dat die $fout_1$-machine $fout_1(L)$ als taal heeft.

3 Pompende lemma's I

Voor een taal L definiëren we $lengtes(L) = \{|s| \mid s \in L\}$.
$lengtes(L) \subseteq \mathbb{N}$ en we beschouwen $lengtes(L)$ als een geordende rij van getallen $l_i, i \in \mathbb{N}$, dus $l_{i+1} > l_i$.
Beschouw nu de rij van getallen $g_i = l_{i+1} - l_i$.
Tijd voor een voorbeeld: stel dat $L = \{ab, ba, a, bbbb, ababa\}$, dan is $lengtes(L) = [1, 2, 4, 5]$ en de rij $g = [1, 2, 1]$
Voor een bepaalde **oneindige** taal L is nu gegeven dat $g_{i+1} > g_i$, $\forall i \in \mathbb{N}$.

(a) Kan L regulier zijn ? Waarom (bv. een voorbeeld) of waarom niet (bewijs !) ?

(b) Kan L contextvrij zijn ? Waarom (bv. een voorbeeld) of waarom niet (bewijs !) ?

Antwoord Vermits de taal oneindig is, zijn er voor een veronderstelde pomplengte p zeker strings s die in de taal zitten en langer zijn dan p. Het pompen van die string levert stringlengtes op van lengte $|s| + l^i = |s| + i \times |l|$ met $i \in \mathbb{N}$ en l het pompbaar stuk dat niet leeg is. De lengtes van de strings kunnen dus niet sneller stijgen dan *lineair* met een constante factor. Bijgevolg is die taal niet CF en niet Reg.

4 Pompende lemma's II

Gegeven is de CFG over alfabet $\{a\}$ met startsymbool S en regels
$S \to aSB$
$B \to aSa$
$S \to \epsilon$
Is L_{CFG} regulier ? Geef voldoende detail van het bewijs van je antwoord.

Antwoord Laten we eerst de CFG vereenvoudigen tot:
$S \to aSaSa$
$S \to \epsilon$
We hebben B niet meer nodig. Nu merk je dat de strings die van S kunnen afgeleid worden exact de verzameling $\{a^{3n} | n \in \mathbb{N}\}$ vormen, ofwel gegenereerd door de reguliere expressie $(aaa)^*$. Die L_{CFG} is dus regulier.

5 Partities

Gegeven de volgende twee partities:

$$P_1 = \{\{a, b\}, \{c\}, \{d\}, \{e, f, g\}, \{h\}\} \quad (1)$$
$$P_2 = \{\{a, e\}, \{b, c\}, \{d, h\}, \{f, g\}\} \quad (2)$$

Bepaal het supremum van de twee partities volgens de theorie in de cursus (sectie 13.3): laat de tussenliggende stappen zien.

Antwoord

Kies een willekeurige letter, bv. a. Wat hoort bij a ?

- a zit in dezelfde deelverzameling als b en e
  b zit in dezelfde deelverzameling als c
  e zit in dezelfde deelverzameling als f en g
  dus: $\{a, b, c, e, f, g\} \in$ het supremum

Kies een willekeurige letter die nog niet in een element van het supremum voorkomt, bv. d. Wat hoort bij d ?

- d zit in dezelfde deelverzameling als h
  dus: $\{d, h\} \in$ het supremum

Er zijn geen letters meer te beschouwen, dus het supremum is de partitie $\{\{a, b, c, e, f, g\}, \{d, h\}\}$

6 Oneindige unie van talen

Laat $R_i, i \in \mathbb{N}$ een rij reguliere talen zijn, en $C_i, i \in \mathbb{N}$ een rij contextvrije talen. Vermits de unie van twee reguliere talen regulier is, en de unie van twee contextvrije talen contextvrij, kan men gemakkelijk bewijzen dat voor elke $n \in \mathbb{N}$, $\cup_{i=1}^{n} R_i$ regulier is en $\cup_{i=1}^{n} C_i$ contextvrij.
Bewijs of geef een tegenvoorbeeld van volgende uitspraken

(a) $\cup_{i>0} R_i$ is regulier

(b) $\cup_{i>0} R_i$ is contextvrij

(c) $\cup_{i>0} C_i$ is contextvrij

Antwoord Het volstaat op te merken dat *elke* taal de (aftelbaar) oneindige unie is van singleton talen, of meer formeel: $L = \cup_{s \in L} \{s\}$. Elke $\{s\}$ is regulier en context-vrij. Dus: in elk van de drie gevallen is de uitspraak mogelijk waar (neem bijvorbeeld alle R_i gelijk, of alle C_i gelijk), maar hoeft niet waar te zijn.

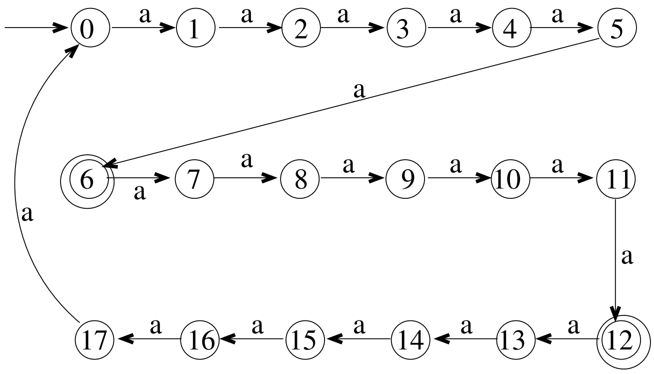
7 Minimale DFA

$\Sigma = \{a\}$, en $L = \{a^n | n \in \mathbb{N}, n \text{ is deelbaar door 6 maar niet door 9}\}$.
Argumenteer waarom L niet kan beslist worden door een DFA met strikt minder dan 18 toestanden.

Antwoord 18 is niet toevallig het kleinste gemeen veelvoud van 6 en 9 ...
Definieer een DFA met 18 toestanden als volgt:

- nummer de toestanden van 0 tot 17
- maak een boog van toestand i naar $((i + 1) \mod 18)$ met label het teken a
- de starttoestand is toestand 0
- zet $F = \{6, 12\}$

Vergewis je er eerst van dat deze DFA L als taal heeft. Hieronder een tekening.



Alle toestanden zijn bereikbaar. We bewijzen nu dat elke twee verschillende toestanden ook f-verschillend zijn.

- 6 en 12 zijn f-verschillend want vanuit 6 geraak je met a^6 in F maar met a^6 geraak je vanuit 12 in 0 en $0 \notin F$
- 6 en 12 zijn eindtoestanden en dus f-verschillend van elke niet-eindtoestand
- neem $0 \leq i < j < 18$ en beide geen eindtoestand; dan zijn er de volgende mogelijkheden:

    1. $i < j < 6$: dan $\delta^*(a^{6-j}, i) = i - j + 6 < 6 = \delta^*(a^{6-j}, j)$
    2. $i < 6 < j < 12$: dan $\delta^*(a^{12-i}, i) = 12$ en $0 < \delta^*(a^{12-i}, j) < 6$
    3. $i < 6 < 12 < j$: ...
    4. $6 < i < j < 12$: zie geval 1
    5. $6 < i < 12 < j$: ...
    6. $12 < i < j$: zie geval 1

    Niet alles is uitgeschreven, maar we hebben telkens een string getoond die vanuit i F bereikt en vanuit j niet, of omgekeerd. Dus zijn i en j f-verschillend.

Bijgevolg is de bovenstaande DFA een minimale DFA: er bestaat er geen met strikt minder toestanden die L bepaalt.