Πεπερασμένα Αυτόματα
Finite Automata

Διάλεξη 11: Θεωρία Υπολογισμού · ΗΥ-280

Χαράλαμπος Ε. Τσουρακάκης

HY-280: Η γνώμη σας μετράει!

Θα ήθελα να ακούσω τη γνώμη σας για την πρόοδο και για το μάθημα γενικότερα. Οι απαντήσεις σας θα με βοηθήσουν να βελτιώσω το μάθημα για τη συνέχεια του εξαμήνου.

Σκανάρετε το QR code και απαντήστε ανώνυμα!

https://surl.li/cnatja

Χρόνος συμπλήρωσης: περίπου 5λεπτά.

Θεώρημα του Kleene

Θεώρημα (Kleene)

Για κάθε γλώσσα $L \subseteq \Sigma^*$, οι παρακάτω προτάσεις είναι ισοδύναμες:

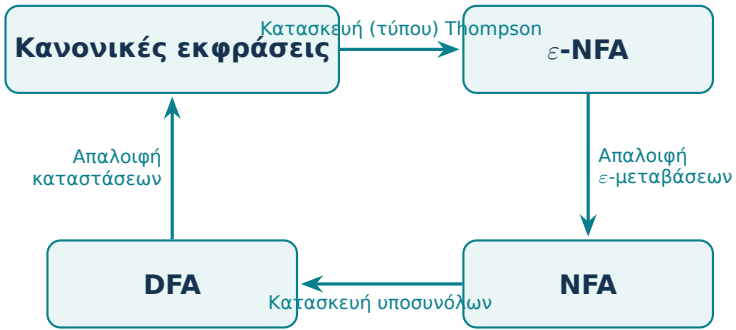
1. Η L περιγράφεται από μια **κανονική έκφραση** (regular expression).
2. Η L αναγνωρίζεται από ένα **DFA**.
3. Η L αναγνωρίζεται από ένα **NFA**.
4. Η L αναγνωρίζεται από ένα **ε-NFA**.

$$\boxed{\text{RegExp} \equiv \text{DFA} \equiv \text{NFA} \equiv \varepsilon\text{-NFA}}$$

Συμπέρασμα

Όλα αυτά τα μοντέλα περιγράφουν **ακριβώς την ίδια κλάση γλωσσών**: τις κανονικές γλώσσες.

Θεώρημα Kleene: Ισοδυναμία μοντέλων

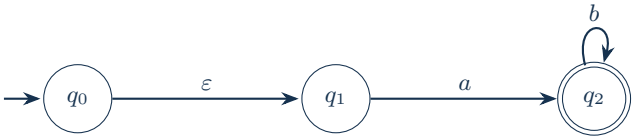


Η βασική ιδέα της απόδειξης

Αρκεί να δείξουμε ότι μπορούμε να μετατρέψουμε **κάθε μοντέλο στο επόμενο**, χωρίς να αλλάξουμε τη γλώσσα που περιγράφει.

Παράδειγμα: Από ε-NFA σε NFA

Θεωρούμε το παρακάτω ε-NFA με αλφάβητο $\Sigma = \{a, b\}$.



Στόχος

Να κατασκευάσουμε ένα ισοδύναμο NFA **χωρίς ε-μεταβάσεις**.

Βασική ιδέα: Κάθε κατάσταση πρέπει να μπορεί να εκτελεί τις μεταβάσεις που ήταν διαθέσιμες μετά από οποιονδήποτε αριθμό ε-μεταβάσεων.

Ερώτημα

Ποιες καταστάσεις μπορούμε να προσεγγίσουμε από την q_0 χωρίς να διαβάσουμε κανένα σύμβολο;

Βήμα 1: Υπολογισμός ε-κλεισιμάτων

Το ε-κλείσιμο μιας κατάστασης q είναι το σύνολο των καταστάσεων που μπορούμε να προσεγγίσουμε από την q χωρίς να καταναλώσουμε σύμβολα.

$$\mathrm{EC}(q) = \{p : \ q \xrightarrow{\varepsilon^*} p\}.$$

Για το αυτόματό μας:

Κατάσταση	ε**-κλείσιμο**
q_0	$\{q_0, q_1\}$
q_1	$\{q_1\}$
q_2	$\{q_2\}$

Παρατήρηση

Από την q_0 μπορούμε να μεταβούμε στην q_1 χωρίς να διαβάσουμε σύμβολο.
Άρα, αν από την q_1 υπάρχει μετάβαση με a, θα πρέπει να υπάρχει αντίστοιχη δυνατότητα και από την q_0 στο νέο NFA.

Βήμα 2: Νέα συνάρτηση μετάβασης

Ορίζουμε τη νέα συνάρτηση μετάβασης ως

$$\boxed{\delta'(q,x) = \mathrm{E}\left(\bigcup_{p\in \mathrm{E}(q)} \delta(p,x)\right)}$$

για κάθε $q \in Q$ και $x \in \Sigma$.

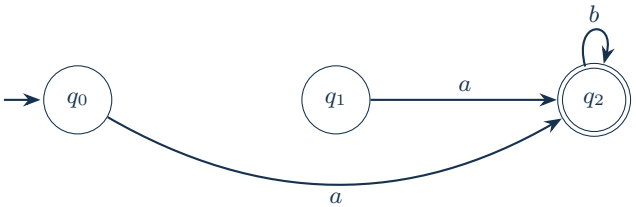
Παράδειγμα: Για την q_0 και το σύμβολο a,

$$\begin{aligned}\delta'(q_0,a) &= \mathrm{E}\big(\delta(q_0,a)\cup\delta(q_1,a)\big)\\ &= \mathrm{E}(\varnothing\cup\{q_2\})\\ &= \{q_2\}.\end{aligned}$$

	a	b
q_0	$\{q_2\}$	$\varnothing$
q_1	$\{q_2\}$	$\varnothing$
q_2	$\varnothing$	$\{q_2\}$

Βήμα 3: Το ισοδύναμο NFA

Αφαιρούμε τις ε-μεταβάσεις και προσθέτουμε τις νέες μεταβάσεις που υπολογίσαμε.



Ποια γλώσσα αναγνωρίζουν;

Και τα δύο αυτόματα αναγνωρίζουν

$$\boxed{L = \{ab^k : k \geq 0\} = ab^*.}$$

Πώς αποδεικνύουμε την ισοδυναμία;

1. **Κανονική έκφραση $\longrightarrow$ ε-NFA**
   Κατασκευάζουμε αναδρομικά ένα αυτόματο χρησιμοποιώντας τις πράξεις ένωσης, συνένωσης και Kleene star.

2. **ε-NFA $\longrightarrow$ NFA**
   Απαλείφουμε τις ε-μεταβάσεις, λαμβάνοντας υπόψη τα ε-κλεισίματα.

3. **NFA $\longrightarrow$ DFA**
   Χρησιμοποιούμε την κατασκευή υποσυνόλων: κάθε κατάσταση του DFA αντιστοιχεί σε ένα σύνολο καταστάσεων του NFA.

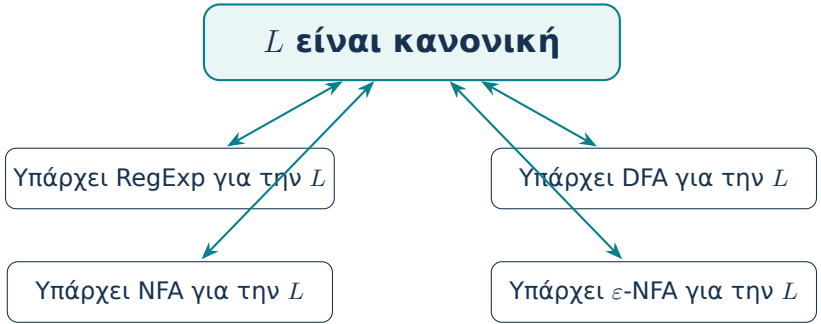
4. **DFA $\longrightarrow$ Κανονική έκφραση**
   Χρησιμοποιούμε τον αλγόριθμο **απαλοιφής καταστάσεων**. Στο τέλος απομένουν δύο καταστάσεις, s και t, και μια κανονική έκφραση στη μετάβαση $s \to t$.

Κοινή ιδιότητα

Κάθε μετατροπή **διατηρεί τη γλώσσα**. Μπορεί να αλλάζει η αναπαράσταση, αλλά όχι οι συμβολοσειρές που γίνονται αποδεκτές.

Τι σημαίνει ότι μια γλώσσα είναι κανονική;

Έστω $L \subseteq \Sigma^*$.



Συγκρατούμε

Για να αποδείξουμε ότι μια γλώσσα είναι **κανονική**, αρκεί να κατασκευάσουμε *ένα* από τα παραπάνω μοντέλα. Για να αποδείξουμε ότι μια γλώσσα **δεν είναι κανονική**;

Οι κανονικές γλώσσες είναι κλειστές ως προς την τομή

Θεώρημα

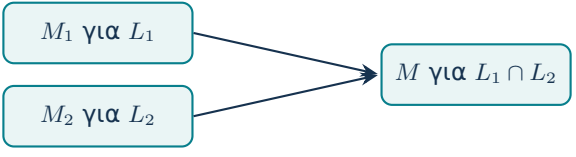
Αν $L_1, L_2 \subseteq \Sigma^*$ είναι κανονικές γλώσσες, τότε και η γλώσσα

$$\boxed{L_1 \cap L_2}$$

είναι κανονική.

Πώς θα το αποδείξουμε;

1. Από το **Θεώρημα Kleene**, υπάρχουν DFA M_1, M_2 που αναγνωρίζουν τις L_1, L_2.
2. Κατασκευάζουμε ένα νέο DFA M που προσομοιώνει **ταυτόχρονα** τα M_1, M_2.
3. Το M αποδέχεται μια λέξη μόνο αν **και τα δύο** αρχικά αυτόματα την αποδέχονται.



Η ιδέα: Γινόμενο αυτομάτων

Έστω δύο DFA:

$$M_1 = (Q_1, \Sigma, \delta_1, q_1, F_1), \qquad M_2 = (Q_2, \Sigma, \delta_2, q_2, F_2).$$

Κατασκευάζουμε το DFA

$$\boxed{M = (Q_1 \times Q_2, \Sigma, \delta, (q_1, q_2), F_1 \times F_2).}$$

Πώς λειτουργεί;

- Κάθε κατάσταση του M είναι ζεύγος (p, r).
- Το p καταγράφει την κατάσταση του M_1.
- Το r καταγράφει την κατάσταση του M_2.

Για κάθε σύμβολο $a \in \Sigma$:

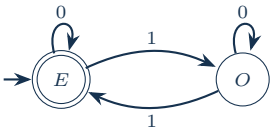
$$\boxed{\delta((p, r), a) = \big(\delta_1(p, a), \delta_2(r, a)\big).}$$

Πότε αποδέχεται το M;

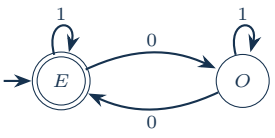
Ακριβώς όταν **και οι δύο συνιστώσες** βρίσκονται σε τελικές καταστάσεις:

Παράδειγμα: Δύο DFA και το γινόμενό τους

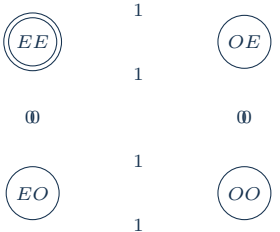
M_1: Άρτιο πλήθος από 1



M_2: Άρτιο πλήθος από 0



M: Άρτιο πλήθος από 0 και 1



Παρατήρηση

Το γινόμενο έχει $2 \times 2 = 4$ καταστάσεις. Μόνο η EE είναι τελική, αφού απαιτούμε και οι δύο συνθήκες να ισχύουν.

Απόδειξη ορθότητας της κατασκευής

Θα δείξουμε ότι το νέο DFA M αναγνωρίζει ακριβώς την τομή $L_1 \cap L_2$.

Βασική ιδιότητα

Για κάθε $w \in \Sigma^*$,

$$\boxed{\delta^*((q_1, q_2), w) = \big(\delta_1^*(q_1, w), \delta_2^*(q_2, w)\big).}$$

Η ιδιότητα αποδεικνύεται με **επαγωγή στο μήκος της λέξης** w.

Επομένως:

$$\begin{aligned} w \in L(M) &\iff \delta^*((q_1, q_2), w) \in F_1 \times F_2 \\ &\iff \delta_1^*(q_1, w) \in F_1 \ \wedge \ \delta_2^*(q_2, w) \in F_2 \\ &\iff w \in L_1 \ \wedge \ w \in L_2 \\ &\iff w \in L_1 \cap L_2. \end{aligned}$$

Συμπέρασμα

Κατασκευάσαμε ένα DFA M με $L(M) = L_1 \cap L_2$.
Από το Θεώρημα Kleene, η $L_1 \cap L_2$ είναι **κανονική**.

Κλειστότητα κανονικών γλωσσών: Ασκήσεις

Έστω $L_1, L_2 \subseteq \Sigma^*$ κανονικές γλώσσες.

Οι κανονικές γλώσσες είναι επίσης κλειστές ως προς:

1. **Συμπλήρωμα:** $\Sigma^* \setminus L_1$.
2. **Διαφορά:** $L_1 \setminus L_2$.
3. **Συμμετρική διαφορά (XOR):**

$$L_1 \triangle L_2 = (L_1 \setminus L_2) \cup (L_2 \setminus L_1).$$

4. **Αντιστροφή συμβολοσειρών:**

$$L_1^R = \{w^R : w \in L_1\}.$$

Άσκηση

Αποδείξτε τις παραπάνω ιδιότητες κλειστότητας, χρησιμοποιώντας το Θεώρημα Kleene και κατάλληλες κατασκευές αυτομάτων.

Το πρόβλημα τερματισμού (Halting)

Γενικό πρόβλημα τερματισμού

Δίνονται ένα πρόγραμμα P και μια είσοδος w.

Ερώτημα: Θα τερματίσει η εκτέλεση του $P(w)$;

$$\mathrm{HALT}(P, w) = \begin{cases} 1, & \text{αν το } P(w) \text{ τερματίζει,} \\ 0, & \text{αν το } P(w) \text{ δεν τερματίζει.} \end{cases}$$

Γνωστό θεώρημα

Δεν υπάρχει αλγόριθμος που να τερματίζει πάντοτε και να αποφασίζει σωστά το HALT για **κάθε** πρόγραμμα P και είσοδο w.

Τι συμβαίνει όμως αν περιοριστούμε σε πεπερασμένα αυτόματα (DFA);

Τερματισμός πεπερασμένων αυτομάτων

Έστω ένα DFA

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

και μια συμβολοσειρά

$$w = a_1 a_2 \cdots a_n \in \Sigma^*.$$

Πρόβλημα τερματισμού για DFA

Δοθέντων των M, w, αποφασίστε αν η επεξεργασία της w από το M ολοκληρώνεται.



Βασική παρατήρηση

Ένα DFA διαβάζει **ακριβώς ένα σύμβολο** σε κάθε βήμα.
Επομένως, σε μια είσοδο μήκους n, η επεξεργασία ολοκληρώνεται μετά από n **μεταβάσεις**.

Γιατί το Halting είναι αποφασίσιμο για DFA;

Θεώρημα

Το πρόβλημα τερματισμού για DFA είναι αποφασίσιμο.

Απόδειξη.

Έστω οποιοδήποτε DFA M και οποιαδήποτε συμβολοσειρά w μήκους n.

1. Ξεκινάμε από την αρχική κατάσταση q_0.
2. Διαβάζουμε διαδοχικά τα σύμβολα της w, ακολουθώντας τη συνάρτηση μετάβασης δ.
3. Μετά από ακριβώς n μεταβάσεις, έχουμε καταναλώσει ολόκληρη την είσοδο.
4. Η επεξεργασία ολοκληρώνεται, ανεξάρτητα από την τελική κατάσταση.

$$\boxed{\forall M \text{ DFA},\ \forall w \in \Sigma^* : \quad M(w)\downarrow}$$

Συμπέρασμα

Ο αλγόριθμος που απαντά πάντοτε **ΝΑΙ** αποφασίζει το πρόβλημα τερματισμού για DFA.

Κάθε κανονική γλώσσα είναι αποφασίσιμη

Θεώρημα

Αν μια γλώσσα L είναι κανονική, τότε υπάρχει αλγόριθμος που αποφασίζει αν μια οποιαδήποτε λέξη w ανήκει στην L.

Απόδειξη.

Από το **Θεώρημα Kleene**, υπάρχει DFA M τέτοιο ώστε

$$L = L(M).$$

Για να αποφασίσουμε αν $w \in L$:

1. Εκτελούμε το M πάνω στην w.
2. Η εκτέλεση ολοκληρώνεται σε $|w|$ μεταβάσεις.
3. Απαντάμε ΝΑΙ αν η τελική κατάσταση ανήκει στο F, διαφορετικά ΟΧΙ.



Υπάρχουν γλώσσες που δεν είναι κανονικές!

Θεωρούμε τη γλώσσα του γενικού προβλήματος τερματισμού:

$$L_{\text{HALT}} = \{\langle P, w\rangle : P \text{ τερματίζει στην είσοδο } w\}.$$

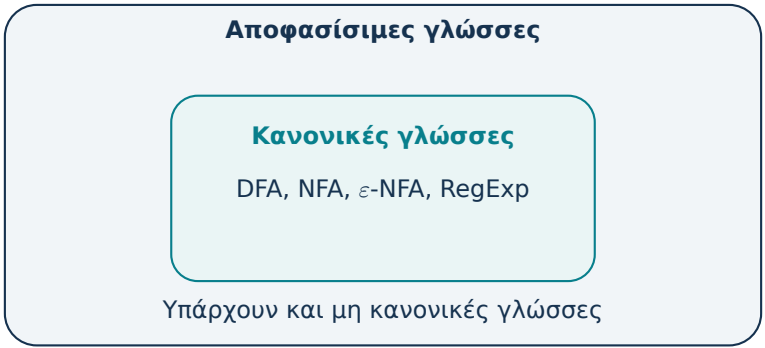
Εδώ $\langle P, w\rangle$ είναι μια πεπερασμένη κωδικοποίηση του προγράμματος και της εισόδου του.

Απόδειξη με άτοπο

1. Υποθέτουμε ότι η L_{HALT} είναι **κανονική**.
2. Από το Θεώρημα Kleene, υπάρχει DFA M που αναγνωρίζει την L_{HALT}.
3. Επειδή το M τερματίζει πάντοτε, μπορούμε να αποφασίσουμε αν $\langle P, w\rangle \in L_{\text{HALT}}$.
4. Άρα το γενικό Halting είναι αποφασίσιμο. **Άτοπο!**

L_{HALT} δεν είναι κανονική.

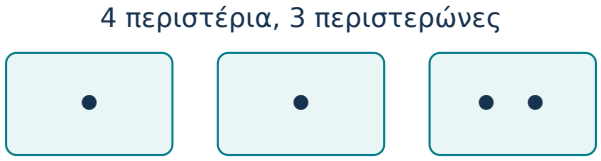
Η θέση των κανονικών γλωσσών



L_{HALT}: μη αποφασίσιμη

3. Λήμμα άντλησης (Pumping Lemma)

Υπενθύμιση: Η Αρχή του Περιστερώνα



Αρχή του Περιστερώνα

Αν τοποθετήσουμε $n+1$ περιστέρια σε n περιστερώνες, τότε **τουλάχιστον ένας** περιστερώνας περιέχει **τουλάχιστον δύο περιστέρια**.

Το λήμμα άντλησης για κανονικές γλώσσες

Λήμμα άντλησης (pumping lemma)

Αν η L είναι κανονική, τότε υπάρχει ακέραιος $p \geq 1$ τέτοιος ώστε κάθε $w \in L$ με $|w| \geq p$ γράφεται $w = xyz$, όπου:

$$\boxed{|y| \geq 1}, \qquad \boxed{|xy| \leq p}, \qquad \boxed{xy^i z \in L \text{ για κάθε } i \geq 0}.$$

$|y| \geq 1$

Αντλούμε ένα **μη κενό** τμήμα.

$|xy| \leq p$

Το τμήμα βρίσκεται μέσα στα πρώτα p σύμβολα.

Κάθε $i \geq 0$

Επιτρέπεται και **διαγραφή**: $xy^0 z = xz$.

Τα x, z επιτρέπεται να είναι κενά.

Πέντε εναλλασσόμενοι ποσοδείκτες

Η δομή του λήμματος

$$\forall L \text{ κανονική} \quad \exists p \geq 1 \quad \forall w \in L,\ |w| \geq p \quad \exists\, (x, y, z) \text{ νόμιμη} \quad \forall i \geq 0 :\ xy^i z \in L.$$

$\forall$ Για **κάθε** κανονική γλώσσα L,
$\exists$ **υπάρχει** $p \geq 1$ τέτοιο ώστε
$\forall$ για **κάθε** $w \in L$ με $|w| \geq p$,
$\exists$ **υπάρχουν** x, y, z με $w = xyz$, $|y| \geq 1$, $|xy| \leq p$, τέτοια ώστε
$\forall$ για **κάθε** $i \geq 0$ ισχύει $xy^i z \in L$.

Απόδειξη του λήμματος (1/3): $p+1$ επισκέψεις σε p καταστάσεις

Έστω DFA M με p καταστάσεις που αναγνωρίζει την L. Πάρτε $w = a_1 \cdots a_m \in L$, με $m \geq p$, και την εκτέλεσή του:

$$r_0 \xrightarrow{a_1} r_1 \xrightarrow{a_2} \cdots \xrightarrow{a_p} r_p \xrightarrow{a_{p+1}} \cdots \xrightarrow{a_m} r_m \in F.$$

Οι $r_0, r_1, \ldots, r_p$ είναι $p+1$ επισκέψεις σε σύνολο p καταστάσεων. Από την αρχή του περιστερώνα υπάρχουν

$$0 \leq j < k \leq p \qquad \text{με} \qquad r_j = r_k.$$

Προσοχή

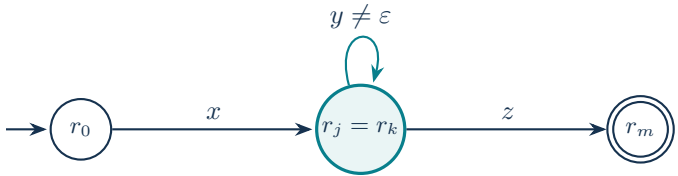
Μετράμε και την αρχική κατάσταση r_0. Η r_p δεν είναι απαραίτητα αποδεκτή: μπορεί να απομένει ολόκληρο το επίθημα $a_{p+1} \cdots a_m$.

Αν $m = p$, το τελευταίο αυτό επίθημα είναι κενό.

Απόδειξη του λήμματος (2/3): πού βρίσκονται τα x, y, z;

Για τις θέσεις $0 \leq j < k \leq p$ με $r_j = r_k$, θέτουμε:

$$x = a_1 \cdots a_j, \qquad y = a_{j+1} \cdots a_k, \qquad z = a_{k+1} \cdots a_m.$$



$$|y| = k - j \geq 1, \qquad |xy| = k \leq p.$$

Κάθε βέλος συνοψίζει ένα τμήμα της εκτέλεσης. Αν $x = \varepsilon$ ή $z = \varepsilon$, οι αντίστοιχες καταστάσεις συμπίπτουν.

Απόδειξη του λήμματος (3/3): γιατί όλες οι αντλήσεις δουλεύουν

Έχουμε $\delta^*(r_0, x) = r_j$, $\delta^*(r_j, y) = r_j$ και $\delta^*(r_j, z) = r_m \in F$.

$$\delta^*(r_j, y^i) = r_j \quad \text{για κάθε } i \geq 0.$$

Επομένως, με το λήμμα συνένωσης:

$$\delta^*(r_0, xy^iz) = \delta^*(r_j, z) = r_m \in F.$$

$i = 0$	xz	παραλείπουμε το κλειστό τμήμα
$i = 1$	xyz	η αρχική λέξη
$i = 3$	$xyyyz$	τρεις διαδοχικές διελεύσεις

Άρα $xy^iz \in L$ για κάθε $i \geq 0$. Μαζί με τους περιορισμούς στα μήκη, αυτό αποδεικνύει το λήμμα. □

Μια κανονική γλώσσα: καλή και κακή διάσπαση της ίδιας λέξης

Στο DFA για άρτιο πλήθος b, επιλέγουμε $p = 2$ και τη λέξη $w = \texttt{bba}$.

Μια διάσπαση που δουλεύει

$x = \varepsilon$, $y = bb$, $z = a$.

$$xy^i z = (bb)^i a.$$

Το πλήθος b είναι $2i$, άρα η λέξη ανήκει στη γλώσσα για κάθε $i \geq 0$.

Μια διάσπαση που δεν δουλεύει

$x = \varepsilon$, $y = b$, $z = ba$.

$$xy^0 z = ba.$$

Η λέξη έχει περιττό πλήθος b, άρα δεν ανήκει στη γλώσσα.

Συγκρατούμε τα εξής:

Και οι δύο διασπάσεις έχουν $|y| \geq 1$, $|xy| \leq 2$. Το λήμμα υπόσχεται **μία κατάλληλη διάσπαση**, όχι ότι όλες οι διασπάσεις αντλούνται.

Πώς χρησιμοποιούμε το λήμμα για να αποκλείσουμε όλα τα DFA

1. Υποθέτουμε προς άτοπο ότι η L είναι κανονική. Έστω p ένα μήκος άντλησης.
2. Επιλέγουμε συγκεκριμένη $w = w(p)$ με $w \in L$ και $|w| \geq p$.
3. Πάρτε **οποιαδήποτε** διάσπαση $w = xyz$ με $|y| \geq 1$ και $|xy| \leq p$. Χρησιμοποιούμε τους περιορισμούς για να συμπεράνουμε τη μορφή του y.
4. Βρίσκουμε $i \geq 0$ για το οποίο $xy^iz \notin L$. Αυτό αντιφάσκει με την ιδιότητα που υποτίθεται ότι έχει η διάσπαση του λήμματος.

Η σειρά που χρειαζόμαστε

$$\forall p \; \exists w \; \forall \text{ νόμιμες διασπάσεις } (x,y,z) \; \exists i : \quad xy^iz \notin L.$$

Η αποτυχία μίας μόνο επιλεγμένης από εμάς διάσπασης δεν αρκεί.

Πρώτη εφαρμογή: δύο μπλοκ ίσου μήκους

$$L_= = \{0^n 1^n : n \geq 0\}.$$

Υποθέτουμε ότι είναι κανονική, με μήκος άντλησης p. Επιλέγουμε $w = 0^p 1^p \in L_=$, με $|w| = 2p \geq p$.

Για **κάθε** νόμιμη διάσπαση, το y βρίσκεται στα πρώτα p σύμβολα (**γιατί;....). Άρα**

$$y = 0^t, \qquad 1 \leq t \leq p.$$

Με $i = 0$ **παίρνουμε**

$$xz = 0^{p-t} 1^p \notin L_=, \qquad \textbf{αφού } p - t \neq p.$$

Άτοπο. Επομένως η $L_=$ **δεν είναι κανονική.** □

Συγκρατούμε τα εξής:

... Επειδή το $|xy| \leq p$ **αναγκάζει το αντλούμενο τμήμα να επηρεάζει μόνο το πρώτο μπλοκ.**

Ασκήσεις: πώς επιλέγουμε λέξη για άντληση

Άσκηση 1: ίσο πλήθος μηδενικών και άσων

$$C = \{w \in \{0,1\}^* : \#_0(w) = \#_1(w)\}.$$

Ερώτημα

Είναι η C κανονική; Χρησιμοποιήστε το λήμμα της άντλησης για να αιτιολογήσετε την απάντησή σας.

Ερώτημα

Ποια από τις $s = (01)^p$ και $s = 0^p 1^p$ μάς βοηθά να ελέγξουμε **όλες** τις επιτρεπτές διασπάσεις;

Η ιδέα

Θέλουμε το y να αποτελείται αναγκαστικά από **ένα μόνο είδος συμβόλου**. Το πετυχαίνουμε με ένα μεγάλο αρχικό μπλοκ.

Λύση 1: εκμεταλλευόμαστε το $|xy| \leq p$

Υποθέτουμε ότι η C είναι κανονική και έστω p το μήκος άντλησης. Επιλέγουμε

$$s = 0^p 1^p \in C.$$

Για **οποιαδήποτε** επιτρεπτή διάσπαση $s = xyz$:

$$|xy| \leq p, \quad |y| > 0 \quad \Longrightarrow \quad y = 0^t \text{ για κάποιο } 1 \leq t \leq p.$$

Αν αντλήσουμε με $i = 0$:

$$xy^0 z = xz = 0^{p-t} 1^p.$$

Πλέον $\#_0(xz) = p - t \neq p = \#_1(xz)$. Άρα $xz \notin C$: **άτοπο**.

Συγκρατούμε τα εξής:

Η λέξη $(01)^p$ δεν είναι καλή επιλογή: π.χ. $x = \varepsilon$, $y = 01$ επιτρέπει άντληση χωρίς να χαλά η ισότητα.

Άσκηση 2: η ίδια λέξη δύο φορές

$$F = \{ww : w \in \{0,1\}^*\}.$$

Ερώτημα

Να αποδειχθεί ότι η F δεν είναι κανονική.

Ερώτημα

Γιατί η 0^p0^p είναι ακατάλληλη για εύκολη αντίφαση; Τι θα συμβεί αν τοποθετήσουμε ένα 1 σε κάθε αντίγραφο;

Η ιδέα

Χρειαζόμαστε **σημάδια** που να εντοπίζουν τα όρια των δύο πανομοιότυπων μισών.

Γιατί η επιλογή $s = 0^p0^p$ δεν λειτουργεί;

Η πρώτη μας προσπάθεια

Επιλέγουμε $s = 0^p0^p = 0^{2p} \in F$.
Από το Λήμμα Άντλησης, $s = xyz$, όπου $|xy| \leq p, \qquad |y| > 0$.
Άρα $y = 0^t$, για κάποιο $1 \leq t \leq p$.

Το πρόβλημα: τι συμβαίνει αν t είναι άρτιο;

Για οποιοδήποτε $k \geq 0$, η άντληση δίνει $xy^kz = 0^{2p+(k-1)t}$.
Αν t είναι άρτιο, τότε $2p + (k-1)t$ είναι άρτιος αριθμός για κάθε $k \geq 0$.
Επομένως, $\boxed{xy^kz = \underbrace{0^{p+(k-1)t/2}}_{w'}\underbrace{0^{p+(k-1)t/2}}_{w'} \in F \quad \forall k \geq 0.}$

Δεν υπάρχει κανένα k που να δίνει αντίφαση!

Συμπέρασμα

Στις αποδείξεις με το Λήμμα Άντλησης δεν επιλέγουμε απλώς μια μεγάλη λέξη. Επιλέγουμε μια λέξη με δομή που επιτρέπει να αποδείξουμε ότι κάθε επιτρεπτή διάσπαση αποτυγχάνει για κάποια άντληση.

Λύση 2: δύο αντίγραφα, δύο σημάδια

Ας είναι p το μήκος άντλησης. Διαλέγουμε

$$s = (0^p 1)(0^p 1) \in F.$$

Επειδή $|xy| \leq p$ και $|y| > 0$, κάθε επιτρεπτό y έχει μορφή 0^t, $1 \leq t \leq p$. Με $i = 0$ παίρνουμε

$$xz = 0^{p-t} 1\, 0^p 1.$$

Γιατί δεν ανήκει στην F;

Η λέξη έχει **ακριβώς δύο** σύμβολα 1 και τελειώνει σε 1. Αν ήταν uu, τότε κάθε u θα είχε ακριβώς ένα 1, αναγκαστικά στο τέλος του. Τα δύο 1 θα απείχαν $|u|$ θέσεις. Όμως εδώ απέχουν $p+1$ θέσεις, ενώ

$$|u| = \frac{2p + 2 - t}{2} \neq p + 1.$$

Άτοπο: η F δεν είναι κανονική.

Άσκηση 3: μήκη που είναι τέλεια τετράγωνα

$$D = \{1^{n^2} : n \geq 0\}.$$

Ερώτημα

Να δείξετε ότι η D δεν είναι κανονική, παρόλο που χρησιμοποιεί μόνο ένα σύμβολο.

Ερώτημα

Ποια είναι η απόσταση ανάμεσα στα διαδοχικά τετράγωνα p^2 και $(p+1)^2$;

Η ιδέα

Με το λήμμα, το μήκος του y είναι το πολύ p. Προσπαθούμε να πέσουμε **αυστηρά ανάμεσα** σε δύο διαδοχικά αποδεκτά μήκη.

Λύση 3: το κενό ανάμεσα στα τετράγωνα

Έστω μήκος άντλησης $p \geq 1$. Επιλέγουμε $s = 1^{p^2} \in D$. Σε κάθε νόμιμη διάσπαση $s = xyz$ ισχύει

$$y = 1^t, \qquad 1 \leq t \leq p.$$

Αν βάλουμε $i = 2$, το μήκος αυξάνεται κατά t:

$$|xy^2z| = p^2 + t.$$

Όμως

$$\underbrace{p^2}_{\text{τετράγωνο}} < p^2 + t \leq p^2 + p < \underbrace{(p+1)^2}_{\text{επόμενο τετράγωνο}}.$$

Άρα το $p^2 + t$ **δεν** είναι τέλειο τετράγωνο: αντίφαση.

Συγκρατούμε τα εξής:

Σε μονοσυμβολικές γλώσσες, η άντληση μεταβάλλει μόνο το **μήκος**.

Σύγκριση: τέσσερις διαφορετικές ιδέες

Γλώσσα	Λέξη s	i	Τι καταστρέφουμε;
$\#_0 = \#_1$	$0^p 1^p$	0	Την ισότητα πληθών
$\{ww\}$	$0^p 1\, 0^p 1$	0	Την ταύτιση των δύο μισών
$\{1^{n^2}\}$	1^{p^2}	2	Το τέλειο τετράγωνο μήκος
$\{0^i 1^j : i > j\}$ (φροντιστήριο)	$0^{p+1} 1^p$	0	Την αυστηρή ανισότητα

Συγκρατούμε τα εξής:

Κοινή δομή: **μία κατάλληλη λέξη** s - **κάθε νόμιμη διάσπαση** - **μία αποτυχημένη τιμή** i. Αυτό παραβιάζει την παραδοχή οτι η γλώσσα είναι κανονική.

Εφαρμογές του Λήμματος Άντλησης

Μέχρι τώρα χρησιμοποιήσαμε το Λήμμα Άντλησης για να δείξουμε ότι ορισμένες γλώσσες **δεν είναι κανονικές**.

Θα δούμε τώρα μια διαφορετική εφαρμογή του: την κατασκευή αλγορίθμων για προβλήματα σχετικά με πεπερασμένα αυτόματα. Για κάθε ερώτημα θέλουμε έναν αλγόριθμο που **τερματίζει πάντοτε** και απαντά σωστά.

Τρία ερωτήματα

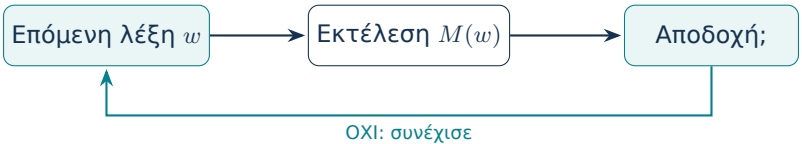
Δοθέντος ενός DFA M, μπορούμε να αποφασίσουμε:

1. Αν η γλώσσα του είναι κενή:
$$L(M) = \varnothing \ ?$$
2. Αν η γλώσσα του είναι άπειρη:
$$|L(M)| = \infty \ ?$$
3. Αν δύο DFA αναγνωρίζουν την ίδια γλώσσα:
$$L(M_1) = L(M_2) \ ?$$

Μια πρώτη ιδέα: Απαρίθμηση όλων των λέξεων

Έστω ένα DFA M πάνω στο $\Sigma = \{0, 1\}$. Μπορούμε να απαριθμήσουμε τις λέξεις της Σ^* κατά αύξον μήκος:

$$\varepsilon, \quad 0,\ 1, \quad 00,\ 01,\ 10,\ 11, \quad 000,\ 001, \ldots$$



Τι πετυχαίνουμε και Πού βρίσκεται το πρόβλημα;

Αν βρούμε μια αποδεκτή λέξη, τότε γνωρίζουμε ότι $L(M) \neq \varnothing$. Όμως α η γλώσσα είναι κενή, η αναζήτηση **δεν θα τερματίσει ποτέ**.

Δύο σημαντικές συνέπειες του Λήμματος Άντλησης

Έστω DFA M με $n \geq 1$ καταστάσεις.

Θεώρημα

1. Η γλώσσα $L(M)$ είναι μη κενή αν και μόνο αν υπάρχει αποδεκτή λέξη w με
$$\boxed{|w| < n.}$$
2. Η γλώσσα $L(M)$ είναι άπειρη αν και μόνο αν υπάρχει αποδεκτή λέξη w με
$$\boxed{n \leq |w| < 2n.}$$



Η ουσία: Και οι δύο ιδιότητες μπορούν να ελεγχθούν εξετάζοντας **πεπερασμένα πολλές λέξεις**.

Απόδειξη: Μη κενότητα και σύντομες λέξεις

Θέλουμε να δείξουμε:

$$L(M) \neq \varnothing \iff \exists w \in L(M) : |w| < n.$$

Η κατεύθυνση ($\Leftarrow$) είναι προφανής.

Για την ($\Rightarrow$):

1. Έστω w μια **συντομότερη** αποδεκτή λέξη.
2. Υποθέτουμε προς άτοπο ότι $|w| \geq n$.
3. Από το Λήμμα Άντλησης υπάρχει διάσπαση $w = xyz$ τέτοια ώστε

$$|xy| \leq n, \qquad |y| > 0, \qquad xy^i z \in L(M) \quad (\forall i \geq 0).$$

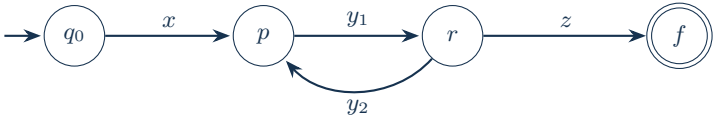
4. Επιλέγοντας $i = 0$, παίρνουμε $xz \in L(M)$ και $|xz| = |w| - |y| < |w|$.

Άτοπο

Βρήκαμε αποδεκτή λέξη μικρότερη από τη w, παρότι η w είχε ελάχιστο μήκος.
Άρα $|w| < n$.

Γιατί εμφανίζεται το όριο n;

Ένα DFA με n καταστάσεις που διαβάζει μια λέξη μήκους τουλάχιστον n επισκέπτεται κάποια κατάσταση δύο φορές.



Τι σημαίνει αυτό;

Η επανάληψη μιας κατάστασης δημιουργεί έναν κύκλο που διαβάζει μια μη κενή λέξη $y = y_1 y_2$.

Μπορούμε να επαναλάβουμε τον κύκλο, ή να τον παραλείψουμε, διατηρώντας την ίδια τελική κατάσταση.

$$xyz, \quad xyyz, \quad xyyyz, \quad \ldots$$

Απόδειξη: Πότε μια κανονική γλώσσα είναι άπειρη;

Θέλουμε να δείξουμε:

$$|L(M)| = \infty \iff \exists w \in L(M) : n \leq |w| < 2n.$$

Κατεύθυνση ($\Leftarrow$)

Υποθέτουμε ότι υπάρχει αποδεκτή λέξη w με $n \leq |w| < 2n$.

Από το Λήμμα Άντλησης:

$$w = xyz, \qquad |y| > 0$$

και

$$xy^i z \in L(M), \qquad \forall i \geq 0.$$

Οι λέξεις $xz, \quad xyz, \quad xyyz, \quad xyyyz, \quad \ldots$ είναι όλες αποδεκτές. Επειδή $|y| > 0$, έχουν διαφορετικά μήκη, άρα είναι διαφορετικές μεταξύ τους.

Συμπέρασμα

Υπάρχουν άπειρες αποδεκτές λέξεις. Επομένως, $|L(M)| = \infty$.

Απόδειξη: Γιατί αρκεί το όριο $2n$;

Κατεύθυνση $(\Rightarrow)$**.**
Υποθέτουμε ότι η $L(M)$ είναι άπειρη. Άρα υπάρχει αποδεκτή λέξη μήκους τουλάχιστον n. Επιλέγουμε μια τέτοια λέξη w με **ελάχιστο μήκος**.

Υποθέτουμε προς άτοπο $|w| \geq 2n$

Από το Λήμμα Άντλησης:

$$w = xyz, \qquad 1 \leq |y| \leq n, \qquad xz \in L(M).$$

Επειδή $|y| > 0$,

$$|xz| = |w| - |y| < |w|.$$

Όμως

$$|xz| \geq |w| - n \geq n.$$

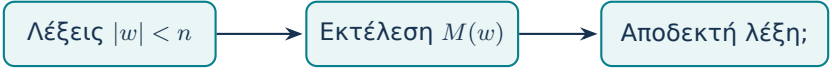
Άτοπο

Η xz είναι αποδεκτή, έχει μήκος τουλάχιστον n και είναι μικρότερη από τη w.
Αυτό αντιφάσκει με την επιλογή της w.

Αλγόριθμος 1: Είναι η γλώσσα κενή;

Είσοδος: DFA M με n καταστάσεις

1. Απαριθμούμε όλες τις λέξεις $w \in \Sigma^*$ με $|w| < n$.
2. Εκτελούμε το M σε καθεμία από αυτές.
3. Αν κάποια γίνεται αποδεκτή, απαντάμε **ΟΧΙ** στο ερώτημα $L(M) = \varnothing$.
4. Αν καμία δεν γίνεται αποδεκτή, απαντάμε **ΝΑΙ**.

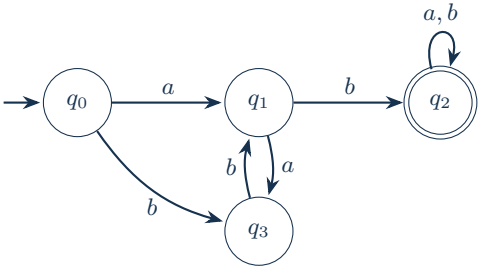


Γιατί είναι αποφασιστής;

Εξετάζουμε πεπερασμένα πολλές λέξεις. Κάθε εκτέλεση DFA τερματίζει.
Η ορθότητα ακολουθεί από το Θεώρημα 8.1: αν $L(M) \neq \varnothing$, υπάρχει αποδεκτή λέξη μήκους μικρότερου από n.

Μη κενότητα ως πρόβλημα προσβασιμότητας

Ένα DFA μπορεί να θεωρηθεί ως κατευθυνόμενος γράφος.



Ισοδυναμία

$$L(M) \neq \varnothing \iff \exists f \in F : q_0 \rightsquigarrow f.$$

Δηλαδή, κάποια τελική κατάσταση είναι προσβάσιμη από την αρχική.

Αλγόριθμος: Εκτελούμε έναν reachability αλγόριθμο (πχ BFS) από την q_0 και ελέγχουμε αν συναντάμε τελική κατάσταση. Έτσι, δεν χρειάζεται να απαριθμήσουμε συμβολοσειρές. Αρκεί μια αναζήτηση στον πεπερασμένο γράφο.

Αλγόριθμος 2: Είναι η γλώσσα άπειρη;

Είσοδος: DFA M με n καταστάσεις

1. Απαριθμούμε όλες τις λέξεις $w \in \Sigma^*$ με
$$n \leq |w| < 2n.$$
2. Εκτελούμε το M σε καθεμία από αυτές.
3. Αν κάποια γίνεται αποδεκτή, απαντάμε **ΝΑΙ**.
4. Διαφορετικά, απαντάμε **ΟΧΙ**.

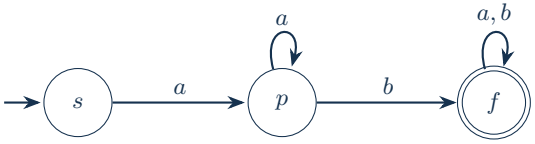


Ορθότητα

Από το Θεώρημα: $|L(M)| = \infty \iff \exists w \in L(M) : n \leq |w| < 2n$.
Ο έλεγχος είναι πεπερασμένος, άρα ο αλγόριθμος τερματίζει πάντοτε.

Απειρία και κύκλοι στο αυτόματο

Υπάρχει και μια πιο αποδοτική προσέγγιση που χρησιμοποιεί τη δομή του γράφου του DFA.



Κριτήριο απειρίας

Η $L(M)$ είναι άπειρη αν και μόνο αν υπάρχει κύκλος στον γράφο του DFA ο οποίος:

1. Είναι προσβάσιμος από την αρχική κατάσταση.
2. Από κάποια κατάστασή του μπορούμε να φτάσουμε σε τελική κατάσταση.

Γιατί; Μπορούμε να διασχίσουμε τον κύκλο οποιονδήποτε αριθμό φορών και στη συνέχεια να φτάσουμε σε τελική κατάσταση.

$$\boxed{xy^k z \in L(M), \quad k \geq 0, \quad |y| > 0.}$$

Αλγόριθμος 3: Αναγνωρίζουν δύο DFA την ίδια γλώσσα;

Έστω δύο DFA M_1, M_2 πάνω στο ίδιο αλφάβητο.

Θέλουμε να αποφασίσουμε αν

$$\boxed{L(M_1) = L(M_2).}$$

Βασική παρατήρηση

Δύο γλώσσες είναι διαφορετικές αν και μόνο αν υπάρχει λέξη που ανήκει σε ακριβώς μία από τις δύο.

$$L_1 \neq L_2 \iff L_1 \triangle L_2 \neq \varnothing,$$

όπου

$$L_1 \triangle L_2 = (L_1 \cap L_2^c) \cup (L_1^c \cap L_2).$$



Απόδειξη: Η ισοδυναμία DFA είναι αποφασίσιμη

Κατασκευή αλγορίθμου:

1. Κατασκευάζουμε DFA N που αναγνωρίζει τη συμμετρική διαφορά:
$$L(N) = L(M_1) \triangle L(M_2).$$
2. Η κατασκευή είναι δυνατή επειδή οι κανονικές γλώσσες είναι κλειστές ως προς το συμπλήρωμα, την τομή και την ένωση.
3. Εφαρμόζουμε τον αλγόριθμο κενότητας στο N.
4. Αν $L(N) = \varnothing$, απαντάμε
$$L(M_1) = L(M_2).$$
Διαφορετικά, απαντάμε ότι οι γλώσσες διαφέρουν.

Γιατί τερματίζει;

Η κατασκευή του N γίνεται σε πεπερασμένο χρόνο.
Ο έλεγχος κενότητας ενός DFA είναι αποφασίσιμος.
Άρα και η ισοδυναμία δύο DFA είναι αποφασίσιμη.