Παραλλαγές και επεκτάσεις αυτομάτων I

Ορισμός

Ένα **two-way deterministic FA** (2DFA) είναι μία πεντάδα $M = (Q, \Sigma, \delta, q_0, F)$, όπου τα Q, Σ, q_0 και F είναι όπως προηγουμένως και δ είναι μία απεικόνιση από το $Q \times \Sigma$ στο $Q \times \{L, R\}$, δηλαδή $\delta \colon Q \times \Sigma \to Q \times \{L, R\}$.
Εάν $\delta(q, a) = (p, L)$, τότε στην κατάσταση q, διαβάζοντας το σύμβολο a, το 2DFA πάει στην κατάσταση p και μετακινεί την κεφαλή προς τα αριστερά κατά μία θέση. Εάν $\delta(q, a) = (p, R)$, τότε το 2DFA πάει στην κατάσταση p και μετακινεί την κεφαλή προς τα δεξιά κατά μία θέση.

Παραλλαγές και επεκτάσεις αυτομάτων II

Configuration ή instantaneous description (ID) ενός 2DFA: περιγράφει το **string εισόδου**, την **τρέχουσα κατάσταση** και την **τρέχουσα θέση της κεφαλής εισόδου**.

Ένα ID του M είναι ένα string στο $\Sigma^* Q \Sigma^*$.
Το ID wqx, όπου $w, x \in \Sigma^*$, αντιπροσωπεύει τα εξής:

1. wx είναι το string εισόδου,
2. q είναι η τρέχουσα κατάσταση και
3. η κεφαλή εισόδου διαβάζει το πρώτο σύμβολο του x.

Εάν το $x = \varepsilon$, τότε η κεφαλή εισόδου έχει μετακινηθεί στο δεξί άκρο της ταινίας εισόδου.

Παραλλαγές και επεκτάσεις αυτομάτων III

Στη συνέχεια εισάγουμε την σχέση $\vdash_M$ στα ID's, έτσι ώστε $I_1 \vdash_M I_2$, αν και μόνο αν το M μπορεί να πάει από την instantaneous description I_1 στην I_2 με μία κίνηση.

Ορίζουμε τη σχέση $\vdash_M$, ή απλά $\vdash$, αν το M εννοείται, ως εξής:

1. $a_1 a_2 \ldots a_{i-1} q a_i \ldots a_n \vdash a_1 a_2 \ldots a_{i-1} a_i p a_{i+1} \ldots a_n$, όταν $\delta(q, a_i) = (p, R)$ και
2. $a_1 a_2 \ldots a_{i-2} a_{i-1} q a_i \ldots a_n \vdash a_1 a_2 \ldots a_{i-2} p a_{i-1} a_i \ldots a_n$, όταν $\delta(q, a_i) = (p, L)$ και $i > 1$.

Παραλλαγές και επεκτάσεις αυτομάτων IV

- Στα διαγράμματα, μία ακμή εκτός από το σύμβολο εισόδου έχει επίσης επιγραφή L ή R.
- Το ανακλαστικό, μεταβατικό κλείσιμο της σχέσης $\vdash$ είναι το $\vdash^*$.
- Computation is a legal finite sequence of configurations.

Ορισμός

$L(M) = \{w \mid q_0 w \vdash^* wp$ για κάποιο p στο $F\}$.

Θεώρημα

Έστω M ένα 2DFA. Τότε $L(M)$ είναι κανονική.

FA με έξοδο I

Μηχανή Moore

Ορισμός

Μία **μηχανή Moore** είναι μία εξάδα $M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$, όπου τα Q, Σ, δ και q_0 είναι όπως στα DFA, με $\delta\colon Q \times \Sigma \to Q$. Επιπλέον:

- Δ: αλφάβητο εξόδου
- λ: απεικόνιση από το Q στο Δ, που δίνει την έξοδο που σχετίζεται με κάθε κατάσταση, δηλαδή $\lambda\colon Q \to \Delta$.

input		a_1	a_2	...	a_n		
states	q_0	q_1	q_2	...	q_{n-1}	q_n	μήκος $n+1$
output	$\lambda(q_0)$	$\lambda(q_1)$	$\lambda(q_2)$	...	$\lambda(q_{n-1})$	$\lambda(q_n)$	μήκος $n+1$

FA με έξοδο II

Μηχανή Moore

Παρατηρήσεις:

- Κάθε μηχανή Moore δίνει έξοδο $\lambda(q_0) = \flat$ για είσοδο ε.
- Το DFA μπορεί να θεωρηθεί ως μία ειδική περίπτωση μίας μηχανής Moore όπου το αλφάβητο είναι $\{0,1\}$ και η κατάσταση q αποδέχεται αν και μόνο αν $\lambda(q) = 1$.
- Μία μηχανή Moore δεν έχει τελικές καταστάσεις και έχει μήκος εξόδου $n+1$.
- Τα διαγράμματα μηχανών Moore είναι όπως αυτά των FA, αλλά επιπλέον σημειώνουμε μαζί με την κατάσταση το αντίστοιχο σύμβολο εξόδου ως εξής $q_i/(q_i)$, π.χ.:

$$\cdots \quad \boxed{q/B} \xrightarrow{a} \boxed{p/C} \quad \cdots$$

FA με έξοδο I

Μηχανή Mealy

Ορισμός

Μία **μηχανή Mealy** είναι μια εξάδα $M = (Q, \Sigma, \Delta, \delta, \lambda, q_0)$, όπου όλα είναι όπως στη μηχανή Moore, με $\delta\colon Q \times \Sigma \to Q$, εκτός από το λ που είναι απεικόνιση από το $Q \times \Sigma$ στο Δ, δηλαδή $\lambda\colon Q \times \Sigma \to \Delta$.

Η έξοδος του M με είσοδο το $a_1 a_2 \ldots a_n$ είναι $\lambda(q_0, a_1)\lambda(q_1, a_2) \ldots \lambda(q_{n-1}, a_n)$, όπου $q_0, q_1, \ldots, q_n$ είναι η ακολουθία των καταστάσεων έτσι ώστε $\delta(q_{i-1}, a_i) = q_i$, για $1 \leqslant i \leqslant n$.

FA με έξοδο II

Μηχανή Mealy

Παρατηρήσεις:

- Η ακολουθία εξόδου έχει μήκος n και όχι $n+1$ όπως η μηχανή Moore και με είσοδο ε μία μηχανή Mealy δίνει έξοδο ε.
- Τα διαγράμματα μηχανών Mealy είναι όπως αυτά των FA, αλλά επιπλέον σημειώνουμε στις μεταβάσεις την αντίστοιχη έξοδο, π.χ.:

$$\cdots \quad (q) \xrightarrow{a/B} (p) \xrightarrow{b/C} (r) \quad \cdots$$

- Μία γενίκευση των μηχανών Mealy είναι το **Finite Transducer** (ή finite state machine, FSM) με συνάρτηση: $\lambda\colon Q \times \Sigma \to \Delta^*$.

Ισοδυναμία μηχανών Moore και Mealy I

Ορισμός

Έστω M μία μηχανή Moore ή Mealy. Ορίζουμε $T_M(w)$, για είσοδο w να είναι η έξοδος που παράγεται από την M με είσοδο w.

Παρατήρηση

Δεν είναι δυνατόν ποτέ να είναι ίδιες οι συναρτήσεις $T_M(w), T_{M'}(w)$, αν M είναι μία μηχανή Moore και M' μία μηχανή Mealy, επειδή $|T_M(w)|$ είναι ένα περισσότερο από $|T_{M'}(w)|$, για κάθε w. Ωστόσο, μπορούμε να δώσουμε τον παρακάτω ορισμό ισοδυναμίας:

Ορισμός

Μία μηχανή Mealy M και μία μηχανή Moore M' είναι ισοδύναμες:
$\forall w$: $\flat T_M(w) = T_{M'}(w)$, όπου $\lambda'(q_0) = \flat$ και $T_{M'}(\varepsilon) = \flat$,
$T_M(w) =$ η έξοδος μετά την επεξεργασία του w.

Ισοδυναμία μηχανών Moore και Mealy II

Μπορούμε να αποδείξουμε το ακόλουθο θεώρημα για την ισοδυναμία μεταξύ μηχανών Moore και Mealy:

Θεώρημα

$\forall M$ Moore $\Rightarrow$ $\exists M'$ Mealy ισοδύναμη.
$\forall M$ Mealy $\Rightarrow$ $\exists M'$ Moore ισοδύναμη.

Απόδειξη.

1. Έστω μία μηχανή Moore M_1. Κατασκευάζουμε μία ισοδύναμη μηχανή Mealy M_2: ορίζουμε $\lambda'(q,a) = \lambda(\delta(q,a))$ για όλες τις καταστάσεις q και για όλα τα σύμβολα εισόδου a.
2. Έστω M_1 μηχανή Mealy. Τότε $M_2 = \{Q \times \Delta, \Sigma, \Delta, \delta', \lambda', [q_0, b_0]\}$, όπου b_0 είναι τυχαίο στοιχείο του . Αυτό σημαίνει ότι οι καταστάσεις του M_2 είναι ζευγάρια της μορφής $[q,b]$, αποτελούμενα από μία κατάσταση της M_1 και ένα σύμβολο εξόδου. Ορίζουμε $\delta'([q,b],a) = [\delta(q,a), \lambda(q,a)]$ και $\lambda'([q,b]) = b$. Το δεύτερο στοιχείο μίας κατάστασης $[q,b]$ της M_2 είναι η έξοδος που παράγεται από την M_1 σε κάποια μετάβαση στην κατάσταση q. Μόνο τα πρώτα στοιχεία των καταστάσεων της M_2 καθορίζουν τις κινήσεις της M_2. Εύκολα με επαγωγή στο n δείχνεται ότι αν η M_1 πηγαίνει στις καταστάσεις $q_0, q_1, \dots, q_n$ με είσοδο $a_1 a_2 \dots a_n$ και δίνει έξοδο $b_1, b_2, \dots, b_n$, τότε η M_2 πηγαίνει στις καταστάσεις $[q_0, b_0], [q_1, b_1], \dots, [q_n, b_n]$ και δίνει έξοδο $b_0, b_1, \dots, b_n$.

$\square$

Pumping lemma για κανονικά σύνολα I

Λήμμα (Pumping lemma)

Εάν L είναι regular τότε:
$\exists n \in \mathbb{N}, \forall z \in L$ με $|z| \geqslant n, \exists u, v, w \in \Sigma^* :$
$[z = uvw \ \wedge |uv| \leqslant n \ \wedge |v| \geqslant 1 \wedge \forall i \in \mathbb{N}\ (uv^i w \in L)]$

Pumping lemma για κανονικά σύνολα II

Χρησιμοποιώντας το λήμμα δείχνουμε ότι ένα δοσμένο σύνολο δεν είναι regular. Η μέθοδος είναι η ακόλουθη:

1. Διαλέγεις τη γλώσσα που θέλεις να αποδείξεις πως δεν είναι regular.
2. Ο αντίπαλος (PL) επιλέγει ένα n. Θα πρέπει να μπορείς για οποιοδήποτε πεπερασμένο ακέραιο n διαλέξει, να αποδείξεις ότι η L δεν είναι regular, αλλά από τη στιγμή που ο αντίπαλος έχει διαλέξει ένα n αυτό είναι σταθερό στην απόδειξη.
3. Διαλέγεις ένα string z της L έτσι ώστε $|z| \geqslant n$.
4. Ο αντίπαλος (PL) σπάει το z σε u, v και w που ικανοποιούν τους περιορισμούς $|uv| \leqslant n$ και $|v| \geqslant 1$.
5. Φτάνεις σε αντίφαση δείχνοντας ότι για κάθε u, v, w που καθορίζονται από τον αντίπαλο, υπάρχει ένα i για το οποίο uv^iw δεν ανήκει στην L. Τότε μπορούμε να συμπεράνουμε ότι η L δεν είναι regular. Η επιλογή του i μπορεί να εξαρτάται από τα n, u, v, w.

Pumping lemma για κανονικά σύνολα III

Παράδειγμα

- $L = \{a^k b^k \mid k \in \mathbb{N}\} = \{\varepsilon, ab, aabb, aaabbb, \ldots\}$
  1. Υποθέτουμε ότι L είναι regular και χρησιμοποιούμε το pumping lemma.
  2. PL: $\exists n \in N$
  3. Διαλέγουμε $z = a^n b^n$. Εντάξει επιλογή, διότι $z \in L$, $|z| = 2n \geqslant n$.
  4. PL: z μπορεί να γραφεί: $z = uvw$ με $|uv| \leqslant n \ \wedge |v| \geqslant 1$, ώστε $v = a^l$ με $l \geqslant 1$.
  5. Διαλέγουμε $i = 2$: $uvvw = a^{n+l}b^n \in L$.

Άτοπο

Παράδειγμα

Το σύνολο $L = \{0^{k^2} \mid k \geqslant 1\} = \{0, 0000, 000000000, \ldots\}$, δεν είναι regular.

1. Υποθέτουμε ότι L είναι regular και χρησιμοποιούμε το pumping lemma.
2. PL: $\exists n \in N$.
3. Διαλέγουμε $z = 0^{n^2}$. Εντάξει επιλογή, διότι $|z| = n^2 \geqslant n$.
4. PL: z μπορεί να γραφεί: $z = uvw$ με $|uv| \leqslant n \ \wedge |v| \geqslant 1$, ώστε $v = 0^l$ με $1 \leqslant l \leqslant n$.
5. Διαλέγουμε $i = 2$: $uvvw = 0^{n^2+l} \in L$, αλλά $n^2 + l$ δεν είναι τέλειο τετράγωνο, διότι: $\underline{n^2} < n^2 + 1 \leqslant n^2 + l \leqslant n^2 + n = n(n+1) < \underline{(n+1)^2}$.

Άτοπο

Ιδιότητες κλειστότητας για κανονικά σύνολα

Λογικές (boolean) πράξεις

Υπενθύμιση: Οι boolean (λογικές) πράξεις είναι *η ένωση, η τομή και το συμπλήρωμα*.

Θεώρημα

*Η κλάση των regular sets είναι μηχανιστικά κλειστή (effectively closed) ως προς τις λογικές πράξεις, την παράθεση (concatenation) και το Kleene *.*

Ιδιότητες κλειστότητας για κανονικά σύνολα Ι

Αντικαταστάσεις και ομομορφισμοί

Αντικατάσταση: $f\colon \Sigma \to \mathsf{Pow}(\Delta^*)$

Ειδική περίπτωση:
Ομομορφισμός: $h\colon \Sigma \to \Delta^*$

Αντίστροφος ομομορφισμός:

$h^{-1}(w) = \{x \mid h(x) = w\}$ για string και
$h^{-1}(L) = \{x \mid h(x) \in L\}$ για γλώσσα.

Ιδιότητες: $h(h^{-1}(L)) \subseteq L$ ενώ $h^{-1}(h(L)) \supseteq L$, για κάθε γλώσσα L.

Η απεικόνιση f μπορεί να επεκταθεί και σε strings ως εξής:

$$\begin{cases} \tilde{f}(\varepsilon) = \varepsilon \\ \tilde{f}(wa) = \tilde{f}(w)f(a) \end{cases}$$

Η f επεκτείνεται και σε γλώσσες: $f^{\times}(L) = \bigcup_{x \in L} f(x)$.

Ιδιότητες κλειστότητας για κανονικά σύνολα II

Αντικαταστάσεις και ομομορφισμοί

Ορισμός

Κανονική αντικατάσταση (regular substitution):
αντικατάσταση (substitution) τέτοια ώστε $\forall a \in \Sigma$: $f(a)$ είναι regular.

Θεώρημα

Η κλάση των regular sets είναι μηχανιστικά κλειστή (effectively closed) ως προς τις πράξεις της κανονικής αντικατάστασης, ομομορφισμού και αντίστροφου ομομορφισμού.

Ιδιότητες κλειστότητας για κανονικά σύνολα

Πηλίκα Γλωσσών (quotients of languages)

Πηλίκο γλωσσών: $L_1/L_2 = \{w \mid \exists v \in L_2 : \; wv \in L_1\}$

Παράδειγμα

$L_1 = ab^*c$ και $L_2 = b^*c$ τότε $L_1/L_2 = ab^*$.

Θεώρημα

Η κλάση των regular sets είναι κλειστή ως προς πηλίκο με τυχαίο σύνολο.

Αλγόριθμοι απόφασης για κανονικά σύνολα Ι

Θεώρημα

Έστω M ένα DFA με $|Q| = n$. Τότε

1. $L(M) \neq \varnothing \Leftrightarrow \exists z \in L(M)$ *με* $|z| < n$.
2. $|L(M)| = \infty \Leftrightarrow \exists z \in L(M)$ *με* $n \leqslant |z| < 2n$.

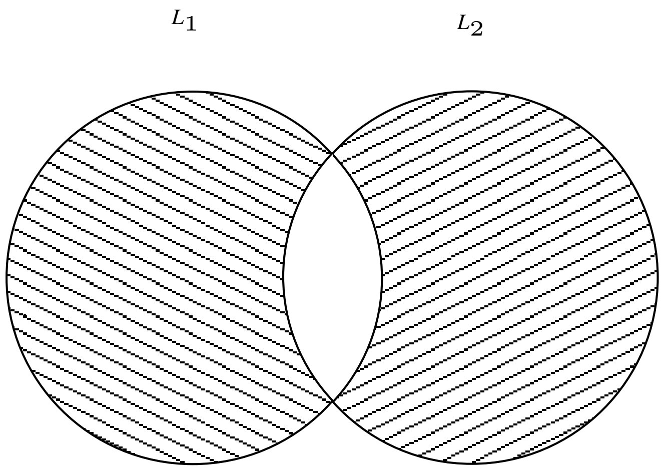
Θεώρημα

Υπάρχει μηχανιστικός αλγόριθμος που αποφασίζει αν δύο πεπερασμένα αυτόματα αποδέχονται το ίδιο σύνολο, δηλαδή αν είναι ισοδύναμα.

Απόδειξη.

Έστω M_1, M_2 τα αυτόματα που αποδέχονται την L_1, L_2 αντίστοιχα. Γνωρίζουμε από θεώρημα ότι υπάρχει αυτόματο M που αποδέχεται το $L = (L_1 \cap \overline{L_2}) \cup (\overline{L_1} \cap L_2)$. Ισχύει $L = \varnothing \Leftrightarrow L_1 = L_2$. Έτσι από το προηγούμενο θεώρημα υπάρχει αλγόριθμος που αποφασίζει αν $L_1 = L_2$. □

Αλγόριθμοι απόφασης για κανονικά σύνολα II



Σχήμα: Ισοδυναμία πεπερασμένων αυτομάτων

Το θεώρημα Myhill-Nerode I

Για κάθε γλώσσα L θα ορίσουμε μία αντίστοιχη σχέση ισοδυναμίας R_L ως εξής:

Ορισμός

Ορίζουμε για $x, y \in \Sigma^*$, $x\ R_L\ y$, αν και μόνον αν για κάθε $z \in \Sigma^*$, $xz \in L$ ακριβώς όταν $yz \in L$ (δηλαδή για κάθε $z \in \Sigma^*$ είτε αμφότερα τα xz, yz ανήκουν στην γλώσσα είτε κανένα από τα δύο δεν ανήκει στην γλώσσα).

Ο **δείκτης** μίας σχέσης ισοδυναμίας είναι το πλήθος των κλάσεων ισοδυναμίας αυτής. Έτσι, λέμε ότι μία ισοδυναμία είναι *πεπερασμένου δείκτη* αν το πλήθος των κλάσεων ισοδυναμίας της είναι πεπερασμένο.

Παρατήρηση: Όπως θα δείξουμε παρακάτω, κάτι που χαρακτηρίζει κάθε κανονική γλώσσα L είναι ότι η R_L είναι πεπερασμένου δείκτη.

Το θεώρημα Myhill-Nerode II

Επίσης, για κάθε πεπερασμένο αυτόματο M με αλφάβητο Σ ορίζουμε μία αντίστοιχη σχέση R_M στο Σ^* ως εξής:

Ορισμός

$x \; R_M \; y$ αν και μόνον αν $\tilde{\delta}(q_0, x) = \tilde{\delta}(q_0, y)$.

Πρόταση

Η R_M είναι ***δεξιά αναλλοίωτη*** *(right invariant), ως προς την παράθεση, δηλαδή για κάθε x, $y \in \Sigma^*$ με $x \; R_M \; y$ έχουμε ότι για κάθε $z \in \Sigma^*$ ισχύει $xz \; R_M \; yz$.*

Απόδειξη.

$\delta(q_0, xz) = \delta(\delta(q_0, x), z) = \delta(\delta(q_0, y), z) = \delta(q_0, yz)$. □

Το θεώρημα Myhill-Nerode III

Θεώρημα (Myhill-Nerode)

Τα παρακάτω είναι ισοδύναμα για μία γλώσσα L:

(i) *η L είναι κανονική*

(ii) *η L είναι ένωση κάποιων από τις κλάσεις ισοδυναμίας μίας δεξιά αναλλοίωτης σχέσης ισοδυναμίας πεπερασμένου δείκτη*

(iii) *η σχέση R_L είναι πεπερασμένου δείκτη.*

Απόδειξη: Θα δείξουμε τις εξής κατευθύνσεις: (i) → (ii) → (iii) → (i).

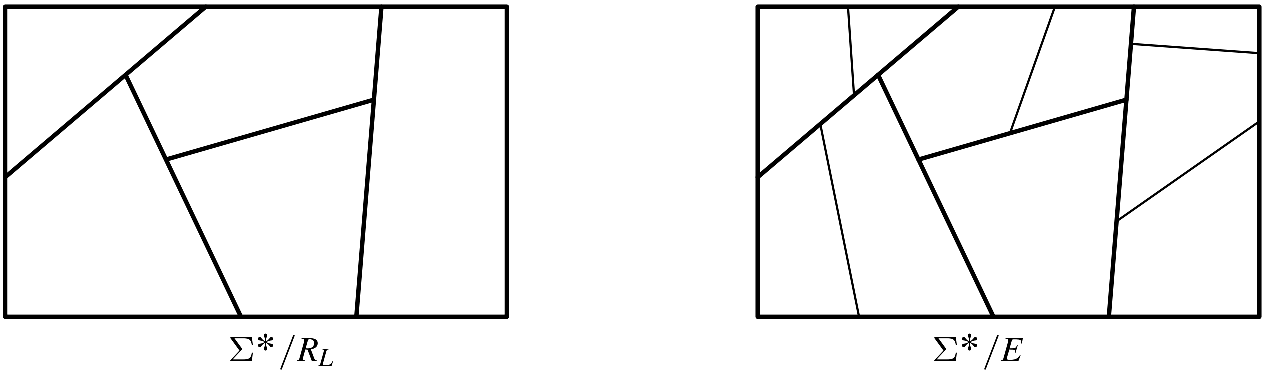
(i) $\implies$ (ii). Έστω DFA $M = (Q, \Sigma, \delta, q_0, F)$ που αποδέχεται την L. Όπως είδαμε, η R_M είναι δεξιά αναλλοίωτη. Οι κλάσεις ισοδυναμίας της R_M είναι όσες και οι καταστάσεις του αυτομάτου M, δηλαδή η R_M είναι επιπλέον πεπερασμένου δείκτη. Προφανώς, η γλώσσα που γίνεται αποδεκτή από το αυτόματο M είναι ένωση των κλάσεων ισοδυναμίας που αντιστοιχούν σε τελικές καταστάσεις του αυτομάτου.

Το θεώρημα Myhill-Nerode IV

(ii) $\implies$ (iii). Έστω ισοδυναμία E που ικανοποιεί την ιδιότητα που δίνεται στο (ii). Θα δείξουμε την συνεπαγωγή: Αν για $x, y \in \Sigma^*$ ισχύει $x\ E\ y$, τότε $x\ R_L\ y$. Πράγματι έστω $x\ E\ y$. Αφού η E είναι δεξιά αναλλοίωτη, έχουμε ότι για κάθε $z \in \Sigma^*$ ισχύει $xz\ E\ yz$, δηλαδή είτε αμφότερα τα xz, yz ανήκουν στην γλώσσα L, είτε κανένα από τα δύο δεν ανήκει στην γλώσσα L, δηλαδή (από τον ορισμό της R_L) $x\ R_L\ y$.

Επομένως, κάθε κλάση ισοδυναμίας της E περιέχεται σε μία κλάση ισοδυναμίας της R_L. Αυτό σημαίνει ότι η σχέση E αποτελεί *εκλέπτυνση* (refinement, βλέπε σχήμα) της σχέσης R_L και αφού η E είναι πεπερασμένου δείκτη, δεν μπορεί παρά και η R_L να είναι πεπερασμένου δείκτη.

Το θεώρημα Myhill-Nerode V



Σχήμα: Εκλέπτυνση (refinement) σχέσης ισοδυναμίας (fine-coarse=λεπτή-αδρή)

Το θεώρημα Myhill-Nerode VI

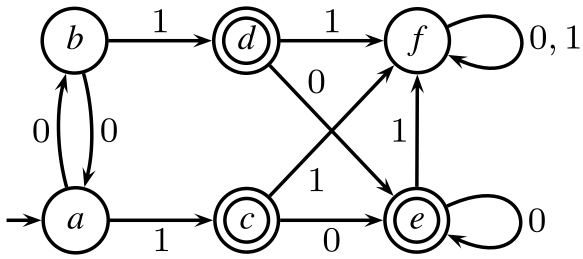
(iii) $\implies$ (i). Αποδεικνύουμε πρώτα τον εξής ισχυρισμό: Η R_L είναι δεξιά αναλλοίωτη. Πράγματι, αν $x\ R_L\ y$ τότε για κάθε $u \in \Sigma^*$ έχουμε $xu\ R_L\ yu$ αφού από τον ορισμό της R_L για κάθε $v \in \Sigma^*$ έχουμε $xuv \in L \iff yuv \in L$ (αρκεί να θέσουμε $z = uv$).

Τελικά, θα κατασκευάσουμε αυτόματο $M = (Q, \Sigma, \delta, q_0, F)$ που αποδέχεται την γλώσσα L, δεδομένης της σχέσης R_L. Την κλάση ισοδυναμίας στην R_L του τυχόντος στοιχείου $x \in \Sigma^*$ συμβολίζουμε με $[x]$. Το σύνολο των καταστάσεων Q είναι οι κλάσεις ισοδυναμίας της R_L, δηλαδή $Q = \{[x] \mid x \in \Sigma^*\}$. Αρχική κατάσταση είναι η $[\varepsilon]$ (η κλάση ισοδυναμίας στην οποία ανήκει η κενή συμβολοσειρά). Το σύνολο των τελικών καταστάσεων είναι $F = \{[x] \mid x \in L\}$. Η συνάρτηση μετάβασης ορίζεται $\delta([x], a) = [xa]$ (ο ορισμός δεν εξαρτάται από το x και είναι συνεπής δεδομένου ότι η R_L είναι δεξιά αναλλοίωτη). Το αυτόματο αποδέχεται την L αφού $\delta(q_0, x) = [x]$ κι επομένως $x \in L(M)$ αν και μόνον αν $[x] \in F$. $\square$

Ελαχιστοποιώντας πεπερασμένα αυτόματα I

Myhill-Nerode ⇒ μοναδικό DFA ελάχιστων καταστάσεων.

Παράδειγμα:
Έστω το αυτόματο M που φαίνεται στο σχήμα, το οποίο αποδέχεται την γλώσσα $L = 0^*10^*$.



Στην $L(M)$ υπάρχουν 6 κλάσεις ισοδυναμίας, οι ακόλουθες:

$$C_a = (00)^*, \quad C_b = (00)^*0, \quad C_c = (00)^*1,$$
$$C_d = (00)^*01, \quad C_e = 0^*100^*, \quad C_f = 0^*10^*1(0+1)^*,$$

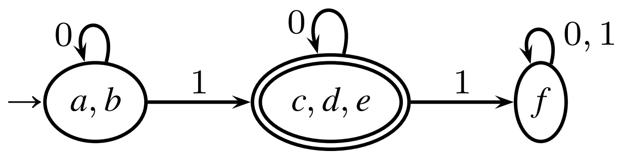
οπότε και $L = C_c \cup C_d \cup C_e$.

Ελαχιστοποιώντας πεπερασμένα αυτόματα II

Τα C_a, C_b είναι το C_1 (χωρίς 1) $= 0^*$. Τα C_c, C_d, C_e είναι το C_2 (με ένα 1) $= 0^*10^*$ και το C_f είναι το C_3 (περισσότερα του ενός 1) $= 0^*10^*1(0+1)^*$.

Ελαχιστοποιώντας πεπερασμένα αυτόματα III

Το ελάχιστο αυτόματο φαίνεται στο παρακάτω σχήμα.



Η ίδια μέθοδος συστηματικά με πίνακα:

1. Εξαλείφουμε όλες τις απρόσιτες καταστάσεις.
2. Συγχωνεύουμε ισοδύναμες καταστάσεις που δεν διακρίνονται με κανένα επόμενο string.

Ελαχιστοποιώντας πεπερασμένα αυτόματα IV

Μέθοδος για το δεύτερο:

- Το p είναι διακρίσιμο από το q εάν υπάρχει ένα x τέτοιο ώστε $\delta(p, x)$ είναι στο F και $\delta(q, x)$ δεν είναι ή αντίστροφα.
- Φτιάχνουμε ένα πίνακα για να συγκρίνουμε κάθε ζεύγος καταστάσεων. Βάζουμε ένα X σε κάθε θέση του πίνακα κάθε φορά που ανακαλύπτουμε ότι δύο καταστάσεις δεν είναι ισοδύναμες. Αρχικά εγγράφουμε X σε όλα τα ζεύγη που προφανώς διακρίνονται γιατί η μία είναι τελική και η άλλη δεν είναι. Μετά προσπαθούμε να δούμε αν διακρίνονται δύο καταστάσεις, διότι από αυτές με ένα σύμβολο a οδηγούμαστε σε διακρίσιμες καταστάσεις. Επαναλαμβάνουμε την πιο πάνω προσπάθεια ώσπου να μην προστίθεται κανένα X πια στον πίνακα. Τα υπόλοιπα ζευγάρια είναι μη διακρίσιμα και συνεπώς συγχωνεύσιμα.

Στο πίνακα οι δείκτες 1 και 2 του X δείχνουν σε ποια επανάληψη εγγράφουμε το X.

b					
c	X_1	X_1			
d	X_1	X_1			
e	X_1	X_1			
f	X_2	X_2	X_1	X_1	X_1
	a	b	c	d	e

Ελαχιστοποιώντας πεπερασμένα αυτόματα V

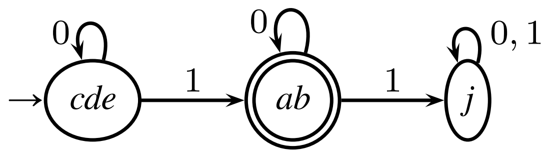
Τελικά οι ισοδύναμες καταστάσεις είναι $a \equiv b$, $c \equiv d \equiv e$.

Ελαχιστοποιώντας πεπερασμένα αυτόματα I

Μέθοδος Beckmann

Άλλη μέθοδος για την κατασκευή ενός ελάχιστου DFA είναι η **μέθοδος Beckmann**.

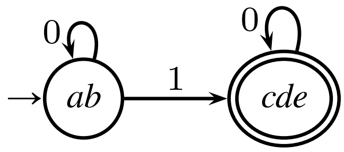
1. Κατασκεύασε το καθρέφτισμα του FA: δηλαδή ανάστρεψε τη φορά των τόξων. Αντάλλαξε τελικές με αρχικές καταστάσεις.
2. Κατασκεύασε DFA ισοδύναμο με το προκύπτον.



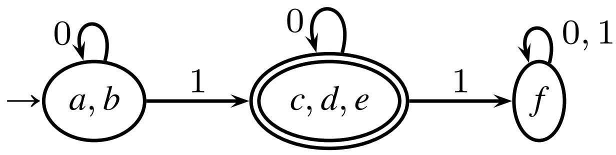
Ελαχιστοποιώντας πεπερασμένα αυτόματα II

Μέθοδος Beckmann

3. Ξανακατασκεύασε το καθρέφτισμα.



4. Ξανακατασκεύασε ισοδύναμο DFA. Αυτό τώρα είναι το ελάχιστο DFA που βρήκαμε και παραπάνω, δηλ.



Τυπικές Γραμματικές I

Μια **τυπική γραμματική (formal grammar)** $G = (V, T, P, S)$ αποτελείται από:

- ένα πεπερασμένο αλφάβητο V από **μη τερματικά σύμβολα (non terminals)** ή **μεταβλητές (variables)**,
- ένα πεπερασμένο αλφάβητο T από **τερματικά σύμβολα (terminals)** ή **σταθερές (constants)**, τ.ώ. $V \cap T = \emptyset$,
- ένα πεπερασμένο σύνολο P από **κανόνες παραγωγής (production rules)** ή απλούστερα **παραγωγές (productions)**, δηλαδή διατεταγμένα ζεύγη (α, β) όπου $\alpha, \beta \in (V \cup T)^*$ και $\alpha \neq \varepsilon$
  (Σύμβαση: γράφουμε $\alpha \to \beta$ αντί για (α, β)),
- ένα **αρχικό σύμβολο (start symbol)** ή **αξίωμα** $S \in V$.

Τυπικές Γραμματικές ΙΙ

Θα χρησιμοποιήσουμε την εξής σύμβαση για τη χρήση γραμμάτων:

$$a, b, c, d, \ldots \in T$$
$$A, B, C, D, \ldots \in V$$
$$z, y, x, w, v, u, \ldots \in T^*$$
$$\alpha, \beta, \gamma, \delta, \ldots \in (V \cup T)^*$$

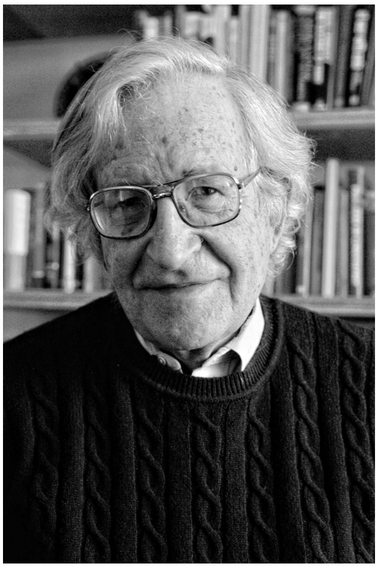
- Θα γράφουμε $\alpha \to \beta|\gamma|\delta$ ως ένα κανόνα στο P αντί για τους τρεις κανόνες $\alpha \to \beta$, $\alpha \to \gamma$, $\alpha \to \delta$ στο P .
- Για μια γραμματική G λέμε ότι το $\gamma_1\beta\gamma_2$ **προκύπτει (is derived from)** από το $\gamma_1\alpha\gamma_2$ και γράφουμε $\gamma_1\alpha\gamma_2 \underset{G}{\Rightarrow} \gamma_1\beta\gamma_2$, αν ο $\alpha \to \beta$ είναι κανόνας παραγωγής (δηλαδή $(\alpha, \beta) \in P$).

Τυπικές Γραμματικές III

- Συμβολίζουμε με $\stackrel{*}{\Rightarrow}$ το ανακλαστικό, μεταβατικό κλείσιμο του $\Rightarrow$, δηλαδή $\alpha \stackrel{*}{\Rightarrow} \beta$ σημαίνει ότι υπάρχει μια ακολουθία παραγωγών (derivation): $\alpha \Rightarrow \alpha_1 \Rightarrow \alpha_2 \Rightarrow \cdots \alpha_k \Rightarrow \beta$.
- Μια συμβολοκολουθία $\alpha \in (V \cup T)^*$ ονομάζεται **προτασιακή μορφή (sentential form)** αν ισχύει $S \stackrel{*}{\Rightarrow} \alpha$.
- Ως **γλώσσα που παράγεται από τη γραμματική** G ορίζουμε την $L(G) := \{w \in T^* | S \stackrel{*}{\Rightarrow} w\}$.
- Δύο γραμματικές G_1 , G_2 θα ονομάζονται **ισοδύναμες** αν $L(G_1) = L(G_2)$.

Ιεραρχία Chomsky I

Ο Noam Chomsky (1956) ταξινόμησε τις τυπικές γραμματικές σε μια ιεραρχία σύμφωνα με τη μορφή των κανόνων παραγωγής τους.



Noam Chomsky