4. Крайни автомати. Регулярни езици. Теорема на Клини.

Дефиниция: Краен детерминиран автомат е наредената петорка $M = (K, \Sigma, \delta, s, F)$, където:

K е крайно множество от състояния
Σ е крайна азбука
$s \in K$ е началното състояние
$F \subseteq K$ е множеството от крайни състояния
δ е функцията на преходите. Това е функция от $K \times \Sigma$ в K

Функцията на преходите определя това как се сменя текущото състояние според това каква е текущата буква в думата. Например, ако се намираме в състояние $q \in K$ и текущата буква е $a \in \Sigma$, тогава следващото състояние в което трябва да се преместим е $\delta(q,a) \in K$.

Дефиниция: Текуща конфигурация на автомата ще наричаме текущото състояние, в което се намираме и остатъка от думата, който не е прочетен. Както казахме по-рано това по еднозначен начин дефинира текущото състояние на изпълнение на автомата, тъй като по никакъв начин изпълнението не зависи от буквите, които са вече прочетени. Ще записваме конфигурациите с наредени двойки: $(q_2, ababab)$, който означават състояние и остатъка от думата.

Дефиниция: Ако (q,w) и (q',w') са две конфигурации на М, тогава $(q,w) \mapsto_M (q',w')$ е изпълнено тогава и само тогава когато $w = aw'$ за някои символ $a \in \Sigma$ и $\delta(q,a) = q'$. Ако се намираме в конфигурацията (q,e), това означава, че сме прочели цялата дума и работата на автомата е приключила (тук използваме e като означение на празната дума).

Ще означаваме рефлексивната и транзитивно затворена релация на $\mapsto_M$ със $\mapsto_M^*$, което означава, че ако $(q,w) \mapsto_M^* (q',w')$, то тогава (q',w') е достижимо от (q,w) след последователност от преходи или дори без нито един преход. Така можем да дефинираме и кога една дума се разпознава от автомат М.

Дефиниция: една дума w се разпознава от автомат М, тогава и само тогава когато съществува състояние $q \in F$, такова че $(s,w) \mapsto_M^* (q,e)$. Така казваме, че думата е от езика, който автомата разпознава, т.е. $w \in L(M)$, където дефинираме $L(M) = \{w : w \in \Sigma^*, (s,w) \mapsto_M^* (q,e), q \in F\}$.

Дефиниция: Недетерминиран краен автомат е наредената петорка $M = (K, \Sigma, \Delta, s, F)$, където:

K е крайно множество от състояния
Σ е крайна азбука
$s \in K$ е началното състояние

$F \subseteq K$ е множеството от крайни състояния
Δ е множество от преходите. Това е подмножество на $K \times (\Sigma \cup \{e\}) \times K$

Всяка една наредена тройка $(q,u,p) \in \Delta$ се нарича преход в М от q в p при входна буква u. Възможно е да имаме преходи от вида (q,e,p), при които от q преминаваме в p без да четем буква от входа. Конфигурация дефинираме по същият начин както при детерминираните автомати. Релацията $\mapsto_M$, се дефинира по аналогичен начин.

Дефиниция: Два автомата M_1 и M_2 са еквивалентни $\Leftrightarrow$ $L(M_1) = L(M_2)$

Вече можем да въведем и следната теорема:

Теорема: За всеки недетерминиран автомат, съществува еквивалентен детерминиран автомат.

<u>Доказателство</u>:
Нека $M = (K, \Sigma, \Delta, s, F)$ е недетерминиран краен автомат. Ще построим детерминиран краен автомат $M' = (K', \Sigma, \delta', F')$ еквивалентен на M.Идеята е да си представим, че на всяка стъпка не се намираме в едно състояние, а в множество от състояния, което е множеството на всички състояния, в които можем да се намираме в момента за прочетената дума. Така ако имаме 5 състояния $\{q_0, q_1, q_2, q_3, q_4\}$ и за текущо прочетената дума можем да се намираме в състоянията q_0, q_2 и q_3, можем да смятаме, че се намираме в състояние $\{q_0, q_2, q_3\}$. Ако следващият символ от входа премества от q_0 в q_1 или q_2, от q_2 в q_0 и от q_3 в q_2, тогава следващото състояние за тази буква ще е $\{q_0, q_1, q_2\}$.

Използвайки тази идея започваме конструирането. Логично, състоянията на M' ще бъдат $|K'| = 2^K$. Крайните състояния на новата машина ще бъдат всички тези състояния, които съдържат в себе си крайно състояние на недетерминирания автомат. Функцията на преход е малко по сложна, поради това че при всеки един преход в недетерминираната машина е възможно да извършим и произволен брой e-преходи. За да формализираме това, ще въведем следната дефиниция:

Дефиниция: За дадено състояние q, нека $E(q) = \{p \in K : (q,e) \mapsto_M^* (p,e)\}$, т.е. това е множеството от състояния достижими от q, само чрез e-преходи.

Множеството $E(q)$ се пресмята лесно със следният алгоритъм:

1. Първоначално $E(q) := \{q\}$
2. Докато има преход $(p,e,r) \in \Delta$, такъв че $p \in E(q)$ и $r \notin E(q)$ изпълни: $E(q) := E(q) \cup \{r\}$

Този алгоритъм ще завърши след най-много броя на състоянията стъпки.

Така вече можем да дефинираме формално автомата M':

$K' = 2^K$
$s' = E(s)$
$F' = \{Q \subseteq K : Q \cap F \neq \varnothing\}$
и за всяко $Q \subseteq K$ и всяка буква $a \in \Sigma$:
$\delta'(Q,a) = \bigcup\{E(q) : \exists(q,a,p), p \in K, q \in Q\}$

Сега остава да докажем, че така получения автомат е детерминиран и еквивалентен на M. Това, че е детерминиран се вижда лесно, тъй като функцията δ' по дефиниция има единствена стойност във всяка една точка и е дефинирана за всяко едно състояние и всяка една буква от азбуката (това че е възможно $\delta'(Q,a) = \varnothing$, не означава че функцията не е дефинирана, тъй като $\varnothing \in K'$).

Сега ако докажем твърдението
$(q,w) \mapsto^*_M (p,e) \Leftrightarrow (E(q),w) \mapsto^*_{M'} (P,e), p \in P$
ще докажем теоремата много лесно. Това е така, защото ако вземем $w \in \Sigma^*$, то
$w \in L(M) \Leftrightarrow (s,w) \mapsto^*_M (f,e), f \in F \Leftrightarrow (E(s),w) \mapsto^*_{M'} (Q,e), f \in Q \Leftrightarrow (s',w) \mapsto^*_{M'} (Q,e), Q \in F'$
, като последната част на твърдението е дефиницията за това $w \in L(M')$.

Ще докажем горното твърдение по индукция върху $|w|$.

1. За $|w| = 0$, т.е. $w = e$, трябва да покажем че
   $(q,e) \mapsto^*_M (p,e) \Leftrightarrow (E(q),e) \mapsto^*_{M'} (P,e), p \in P$
   Първото твърдение е еквивалентно на това да кажем че $p \in E(q)$,а второто твърдение на това че $P = E(q)$, т.е. $p \in E(q)$, с което твърдението е доказано.
2. Предполагаме, че твърдението е вярно за всички думи с дължина до k.
3. Трябва да докажем, че твърдението е вярно за всички думи w с дължина $k+1$. Нека $w = va$, където $v \in K^*$,а $a \in \Sigma$.
   a. $\Rightarrow$) нека $(q,w) \mapsto^*_M (p,e)$, тогава съществуват две състояния r_1 и r_2, такива че $(q,w) \mapsto^*_M (r_1,a) \mapsto_M (r_2,e) \mapsto^*_M (p,e)$, от където следва, че можем да кажем $(q,v) \mapsto^*_M (r_1,e)$, но тъй като $|v| = k$, от индукционното предположение следва, че $(E(q),v) \mapsto^*_{M'} (R_1,e), r_1 \in R_1$. Тъй като $(r_1,a) \mapsto^*_M (r_2,e)$, то съществува $(r_1,a,r_2) \in \Delta$, следователно от дефиницията на M' следва, че $E(r_2) \subseteq \delta'(R_1,a)$. Тъй като $(r_2,e) \mapsto^*_M (p,e)$, следва че $p \in E(r_2)$, следователно $p \in \delta'(R_1,a)$. От тук следва че $(R_1,a) \mapsto_{M'} (P,e), p \in P$,от където $(E(q),w) \mapsto^*_{M'} (R_1,a) \mapsto_{M'} (P,e)$
   .
   b. $\Leftarrow$) за да докажем твърдението в обратната посока да предположим че $(E(q),va) \mapsto^*_{M'} (R_1,a) \mapsto_{M'} (P,e)$, за някое P, което съдържа p и някое R_1, за което $\delta'(R_1,a) = P$. От дефиницията на δ', знаем че $\delta'(R_1,a)$ е обединението от

всички множества $E(r_2)$, където за някое $r_1 \in R_1$, съществува $(r_1,a,r_2) \in K$. Тъй като $p \in P = \delta'(R_1,a)$, съществува някакво r_2, такова че $p \in E(r_2)$ и за някое $r_1 \in R_1$ има преход $(r_1,a,r_2) \in K$. Следователно от дефиницията на $E(r_2)$, следва че $(r_2,e) \mapsto_M^* (p,e)$. От индукционното предположение следва че $(q,v) \mapsto_M^* (r_1,e)$, от където $(q,va) \mapsto_M^* (r_1,a) \mapsto_M (r_2,e) \mapsto_M^* (p,e)$.

Така теоремата е доказана.

Дефиниция: Нека имаме крайна азбука Σ. Множеството от регулярни изрази във Σ^* е:

1. $\varnothing$, e и всяка буква $a \in \Sigma$ е регулярен израз
2. Ако α и β са регулярни изрази, то $(\alpha\beta)$ също е регулярен израз
3. Ако α и β са регулярни изрази, то $(\alpha \cup \beta)$ също е регулярен израз
4. Ако α е регулярен израз, то α^* също е регулярен израз
5. Няма други регулярни изрази освен тези описани в точки 1 до 4

Дефиниция: Ако имаме даден регулярен израз α, регулярен език е множеството $\Lambda(\alpha)$, където Λ е функция дефинирана по следния начин:

1. 1. $\Lambda(\varnothing) = \varnothing$ и $\Lambda(a) = \{a\}$ за всяко $a \in \Sigma$.
2. Ако α и β са регулярни изрази, тогава $\Lambda((\alpha\beta)) = \Lambda(\alpha)\Lambda(\beta)$
3. Ако α и β са регулярни изрази, тогава $\Lambda((\alpha \cup \beta)) = \Lambda(\alpha) \cup \Lambda(\beta)$
4. Ако α е регулярen израз, тогава $\Lambda(\alpha^*) = \Lambda(\alpha)^*$

Теорема: Класа от езици, които се разпознават от крайни автомати е затворен относно операциите:

1. Обединение
2. Конкатенация
3. Звезда на Клини
4. Допълнение
5. Сечение

Доказателство(тук е добра идея да се приложат и малко диаграми):

За всеки един случай ще покажем как можем да построим автомат, който разпознава езика резултат от съответната операция.

А. Обединение: Нека имаме два недереминирани крайни автомата $M_1 = (K_1,\Sigma,\Delta_1,s_1,F_1)$ и $M_2 = (K_2,\Sigma,\Delta_2,s_2,F_2)$. Ще построим недерминиран краен автомат, който за който $L(M) = L(M_1) \cup L(M_2)$. Това ще направим като направим едно ново състояние и сложим e преходи от това състояние към началните състояния на двата автомата (можем да приложим абсолютно същата техника за да построим обединение на произволен брой автомати). Ето как би изглеждала дефиницията формално: нека новия автомат е $M = (K,\Sigma,\Delta,s,F)$ като,

$K = K_1 \cup K_2 \cup \{s\}$,

$F = F_1 \cup F_2$,

$\Delta = \Delta_1 \cup \Delta_2 \cup \{(s,e,s_1),(s,e,s_2)\}$

Така при всяко едно пускане M недетерминистично решава, кой от двата езика да разпознава, като по този начин разпознава и двата езика. Формално казано: ако $w \in \Sigma^*$, тогава $(s,w) \mapsto_M^* (q,e)$ за някое $e \in F$ $\Leftrightarrow$ $(s_1,w) \mapsto_{M_1}^* (q,e)$, за някое $q \in F_1$ или $(s_2,w) \mapsto_{M_2}^* (q,e)$, за някое $q \in F_2$. Тогава M разпознава w $\Leftrightarrow$ M_1 разпознава w или M_2 разпознава w, т.е. $L(M) = L(M_1) \cup L(M_2)$.

Б. Слепване: Нека пак M_1 и M_2 са два недетерминирани крайни автомати. Тогава за да построим автомат M, за който $L(M) = L(M_1)L(M_2)$, ще обединим състоянията на двата автомата (без ограничение на общността можем да считаме ще състоянията на двата автомата са непресичащи се множества) и от всяко крайно състояние на M_1 ще направим e преход към началното състояние на M_2. Доказателството е аналогично на доказателството от предишната точка.

В. Звезда на Клини: Нека M_1 е недетерминиран краен автомат. Искаме да построим автомат M, такъв че $L(M) = L(M)^*$. За целта ще използваме идея подобна на тази, която използвахме при слепването. Новия автомат M ще има същите състояние и преходи като M_1, но ще има ново начално състояние, което е и крайно, за можем да разпознаваме празната дума. Освен това от всяко едно крайно състояние ще направим e преход към s_1 (началното състояние на M_1), както и e преход от s към s_1. Така когато прочетем дума от $L(M_1)$, можем да започнем от началното състояние на M_1 с нова дума.

Г. Допълнение: Нека $M = (K, \Sigma, \delta, s, F)$ е краен детерминиран автомат. Тогава допълнението на езика на този автомат е $\overline{L} = \Sigma^* - L(M)$ се разпознава от детерминирания краен автомат $\overline{M} = (K, \Sigma, \delta, s, K - F)$, т.е. просто правим всяко не-крайно състояние крайно и всяко крайно, не-крайно.

Д. Сечение: Имайки доказателствата до тук можем просто да си припомним, че

$L_1 \cap L_2 = \Sigma^* - ((\Sigma^* - L_1) \cup (\Sigma^* - L_2))$

Забелязваме, че дясната страна на израза използва само операции, които вече описахме, с което доказателството е завършено.

Теорема(на Клини): Един език е регулярен, тогава и само тогава, когато се разпознава от краен автомат.

<u>Доказателство</u>:

$\Rightarrow$) Очевидно е че крайните автомати разпознават $\varnothing$, както всеки език съставен от една буква. Освен това от предната теорема видяхме, че езиците, които се разпознават от автомати са затворени относно операциите обединение, слепване и звезда на Клини. Следователно всеки един регулярен език се разпознава от някакъв краен автомат.

$\Leftarrow$) Нека $M = (K, \Sigma, \Delta, \delta, F)$ е краен автомат (не задължително детерминиран). Ще конструираме регулярен език R, такъв че $L(R) = L(M)$. За целта ще представим $L(M)$ като обединение на няколко по-прости езика. Нека $K = \{q_1, \ldots, q_n\}$ и $s = q_1$. За $i, j = 1, \ldots, n$ и $k = 0, \ldots, n$, ще дефинираме $R(i, j, k)$ да е множеството от всички думи от Σ^*, които могат да накарат M

да отиде от състояние q_i до състояние q_j, без да се преминава през състояния със номера по-големи от k, като i и j е възможно да са по-големи от k. Забелязваме че при $k = n$:

$R(i,j,n) = \{w \in \Sigma^* : (q_i,w) \mapsto_M^* (q_j,e)\}$

Следователно:

$L(M) = \bigcup\{R(1,j,n) : q_j \in F\}$

Идеята е, да докажем че всяко от множествата $R(i,j,k)$ е регулярно, от където ще следва че $L(M)$ също е регулярен.

Ще докажем това с индукция по k. За $k = 0$, $R(i,j,k)$ е или $\{a \in \Sigma \cup \{e\} : (q_i,a,q_j) \in \Delta\}$ ако $i \neq j$ или $\{e\} \cup \{a \in \Sigma \cup \{e\} : (q_i,a,q_j) \in \Delta\}$ ако $i = j$. Всяко едно от тези множества е крайно, от където следва че е регулярно.

Предполагаме, че за $k-1$, всяко едно множество $R(i,j,k-1)$ е регулярно.

Трябва да докажем, че $R(i,j,k)$ е регулярно. За целта ще представим това множество с операциите обединение, слепване и звезда на Клини по следния начин:

$R(i,j,k) = R(i,j,k-1) \cup R(i,k,k-1)R(k,k,k-1)^* R(k,j,k-1)$

Идеята тук е, че всеки път от q_i до q_j, които не минава през състояния с номер по-голям от k е един от следните типове:

1. Минава през състояния с номера по-малки или равни на $k-1$
2. Минава от q_i до q_k, после се върти от q_k до q_k, нула или повече пъти и накрая отива от q_k до q_j, като всеки един от тези подпътища не минава през състояния с номер по-голям от $k-1$.

От тук следва, че $R(i,j,k)$ е регулярен, което завършва доказателството.

Лема на покачването: Нека L е регулярен език. Тогава съществува цяло число $n \geq 1$, такова че всяка дума $w \in L$, такава че $|w| \geq n$, може да бъде представена като $w = xyz$, като $y \neq e$, $|xy| \leq n$ и $xy^i z \in L$, за всяко $i \geq 0$.

Доказателство(тук май може да се докаже и с картинка):

L е регулярен, значи съществува краен автомат M, които разпознава L. Нека n е броя състояния на M и нека w е дума с дължина n или повече. Да предположим, че първите n стъпки от смятането на M за w са:

$(q_0,w_1w_2...w_n) \mapsto_M (q_1,w_2...w_n) \mapsto_M ... \mapsto_M (q_n,e)$

където q_0 е началното състояние на M и $w_1...w_n$ са първите n букви от w. Тъй като M има n състояния, а минаваме през $n+1$ конфигурации, то съществува състояния q_i и q_j, $0 \leq i < j \leq n$, такива че $q_i = q_j$. Това е поддумата $y = w_i w_{i+1}...w_j$, която води M от състояние q_i обратно в същото състояние, и това не е празната дума тъй като $i < j$. Но тогава тази поддума може да бъде премахната от w или повторена произволен брой пъти след

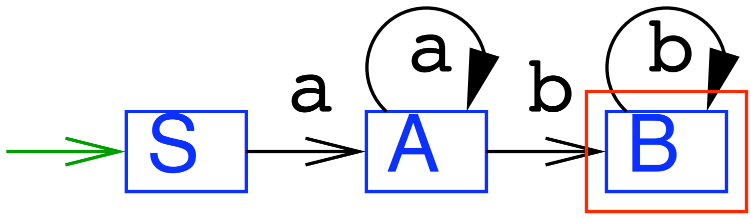
j-тата буква на w и M ще продължи да разпознава новополучената дума. Следователно M ще разпознава $xy^i z \in L$, за всяко $i \geq 0$, където $x = w_1 \ldots w_i$ и $z = w_{j+1} \ldots w_m$. Накрая забелязваме, че дължината на xy е j, което по предположението по-горе е най-много n, така както се изисква в теоремата. С това теоремата е доказана.

Примери за регулярни и нерегулярни езици:

Езика $L = \{a^i b^i : i \geq 0\}$ не е регулярен. Ако беше тогава от горната теорема ще съществува n, за което $a^n b^n = xyz$, такива че $|xy| \leq n$ и $y \neq e$, т.е. $y = a^i, i > 0$, но тогава $xz = a^{n-i} b^n \notin L$, което противоречи на теоремата.

Езика $L = \{a^n : n - prime\}$ не е регулярен. Ако беше тогава то горната теорема съществува n, за което $a^n = xyz$. Тогава $x = a^p, y = a^q, z = a^r, p, r \geq 0, q > 0$. Тогава от теоремата следва че $xy^s z \in L, \forall s \geq 0 \Rightarrow p + sq + r$ е просто, но за $s = p + 2q + r + 2$, получаваме $p + sq + r = (q+1)(p + 2q + r)$, което е произведение на 2 естествени числа всяко от които по голямо от 1.

Езика $L = \{a^n b^m : n > 0, m > 0\}$ е регулярен. И неговия автомат би изглеждал така (автомата не е детерминиран, тъй като няма преход от S със b):



Дефиниция: Нека $L \subseteq \Sigma^*$ е език и нека $x, y \in \Sigma^*$. Казваме че x и y са еквивалентни спрямо L ($x \approx_L y$), ако за всяко $z \in \Sigma^*$, следното е изпълнено: $xz \in L \Leftrightarrow yz \in L$. Забележете, че $\approx_L$ е релация на еквивалентност.

Дефиниция: Нека $M = (K, \Sigma, \delta, s, F)$ е детерминиран краен автомат. Казваме, че две думи $x, y \in \Sigma^*$, са еквивалентни спрямо M ($x \sim_M y$), ако има състояние q, такова че $(s, x) \mapsto_M^* (q, e)$ и $(s, y) \mapsto_M^* (q, e)$.

$\sim_M$ също е релация на еквивалентност и нейните класове на еквивалентност могат да бъдат идентифицирани, чрез състоянията на M, които могат да бъдат достигнати от s, чрез някаква дума. Ще означаваме класа на еквивалентност за дадено състояние q, чрез E_q.

Теорема: За всеки детерминиран краен автомат $M = (K, \Sigma, \delta, s, F)$ и две думи $x, y \in \Sigma^*$, ако $x \sim_M y$, то $x \approx_{L(M)} y$.

От горната теорема следва, че всеки един клас на еквивалентност спрямо $\sim$ се съдържа в клас на еквивалентност на $\approx$ и всеки един клас на еквивалентност на $\approx$ е обединение на един и няколко класа на еквивалентност на $\sim$. Тъй като класовете на еквивалентност на $\sim$ са броя състояния на автомат разпознаващ $L(M)$, то можем да изкажем едно много важно свойство на всеки един автомат, които разпознава $L(M)$: всеки един автомат които разпознава $L(M)$ трябва да има поне толкова състояния, колкото класове на еквивалентност има релацията $\approx$. Следващата теорема доказва, че е възможно да се построи автомат разпознаващ $L(M)$ с точно толкова състояния, което ще рече, че това е автомата с минимален брой състояния за езика $L(M)$.

Теорема (на Михайл-Нероуд): Нека $L \subseteq \Sigma^*$ е регулярен език. Тогава съществува краен детерминиран автомат с брой състояния равен на броя на класове на еквивалентност на $\approx_{L(M)}$, който разпознава L. Това е автомата с минимален брой състояния, който разпознава $L(M)$ и този автомат е единствен с точност до изоморфизъм.

Доказателство:

Нека означим класа на еквивалентност спрямо $\approx_{L(M)}$, породен от x, чрез $[x]$. Ще построим краен детерминиран автомат $M = (K,\Sigma,\Delta,\delta,F)$, за който $L = L(M)$. Дефинираме M по следния начин:

$K = \{[x] : x \in \Sigma^*\}$, т.е. това са класовете на еквивалентност на $\approx_L$

$s = [e]$, класа на еквивалентност на празната дума

$F = \{[x] : x \in L\}$

Накрая дефинираме за всяко състояние $[x] \in K$ и всяка буква $a \in \Sigma$, $\delta([x],a) = [xa]$.

Така дефиниран M има краен брой състояния, защото L е регулярен, т.е. има краен детерминиран автомат M', който го разпознава. От предишната теорема знаем, че има по-малко или равен брой класове на еквивалентност в $\approx_L$ от колкото в $\sim_{M'}$, а $\sim_{M'}$ има краен брой класове на еквивалентност тъй като M' има краен брой състояния, от където следва че и $\approx_L$ има краен брой класове на еквивалентност, т.е. K е крайно.

Освен това δ е добре дефинирана функция на преходите, тъй като е еднозначна и дефинирана за всяко едно състояние на M и всяка една буква от азбуката.

Остава да покажем, че $L = L(M)$. Първо ще покажем че за всеки $x,y \in \Sigma^*$ имаме че:

$$([x],y) \mapsto_M^* ([xy],e) \qquad (1)$$

Това се доказва с индукция по $|y|$. За $y = e$ се доказва тривиално. Ако предположим, че е вярно за всяка една дължина на y до n и $y = y'a$, тогава от индукционното предположение имаме: $([x],y'a) \mapsto_M^* ([xy'],a) \mapsto_M ([xy],e)$.

Използвайки (1), доказателството е директно: за всяка $x \in \Sigma^*$, имаме че $x \in L(M) \Leftrightarrow ([e],x) \mapsto_M^* (q,e), q \in F$, което от (1) е същото като да кажем, че $[x] \in F$ или от дефиницията на F, $x \in L$.

Следствие: Един език L е регулярен тогава и само тогава, когато релацията $\approx_L$ има краен брой класове на еквивалентност.

Алгоритъм за конструиране на минимален детерминиран автомат еквивалентен на даден детерминиран автомат:

Нека $M = (K,\Sigma,\delta,s,F)$ е детерминиран краен автомат. Дефинираме релацията $A_M \subseteq K \times \Sigma^*$, като $(q,w) \in A_M \Leftrightarrow (q,w) \mapsto_M^* (f,e), f \in F$. Казваме, че 2 състояния $q,p \in K$ са еквивалентни (означаваме $q \equiv p$), ако за $\forall z \in \Sigma^* : (q,z) \in A_M \Leftrightarrow (p,z) \in A_M$. Класовете на еквивалентност на $\equiv$ са тези множества от състояния, които дефинират състоянията на минималният автомат, който разпознава $L(M)$. За да намерим тези класове на еквивалентност, ще пресметнем последователно класовете на еквивалентност на $\equiv_0, \equiv_1, \equiv_2, \cdots$, където $p \equiv_n q : (q,z) \in A_M \Leftrightarrow (p,z) \in A_M, z \in \Sigma^*, |z| \le n$. Това ще направим последователно като ще намираме класовете на еквивалентност на една релация от класовете на еквивалентност на предишната релация.

1. За $\equiv_0$ е очевидно, че класовете на еквивалентност са F и $K - F$.
2. Да предположим, че сме намерили класовете на еквивалентност на $\equiv_n$. Тогава за всеки две състояния $q,p \in K$, $p \equiv_{n+1} q \Leftrightarrow p \equiv_n q$ и $\forall a \in \Sigma, \delta(q,a) \equiv_n \delta(p,a)$. По този начин можем да разберем дали 2 състояния са в един клас на еквивалентност или не от където можем да намерим класовете на еквивалентност на $\equiv_n$.
3. Продължаваме докато не получим, че класовете на еквивалентност на $\equiv_n$ са същите като класовете на еквивалентност на $\equiv_{n-1}$.

Алгоритъмът ще завърши след краен брой стъпки, тъй като на всяка една стъпка класовете на еквивалентност се увеличават поне с 1 и не може да има повече класове на еквивалентност от колкото състояния има в M.