Logică și structuri discrete

Expresii regulate

Marius Minea
marius@cs.upt.ro

http://www.cs.upt.ro/~marius/curs/lsd/

2 decembrie 2013

Cum descriem un limbaj ?

Un limbaj = o mulțime de cuvinte peste un alfabet

Adesea suntem interesați în cuvinte cu structură simplă:

un întreg: o secvență de cifre, eventual cu semn

un real: parte întreagă, parte zecimală (una opțională), exponent opțional

un identificator: litere, cifre, _ începând cu literă sau _

fișiere cu numele `01-`*titlu*`.mp3`, `02-`*alttitlu*`.mp3`, ...

Am văzut că anumite limbaje pot fi recunoscute eficient de *automate finite*

dar scrierea automatului ia efort

$\Rightarrow$ se poate face mai simplu ?

Recapitulăm: operații pe limbaje

Am văzut:

reuniunea, intersecția și complementul limbajelor regulate sunt limbaje regulate

Mai putem defini:

Concatenarea limbajelor

$$L1 \cdot L2 = \{w_1 w_2 \mid w_1 \in L_1, w_2 \in L_2\}$$

Închiderea Kleene (repetiția)

$$L^* = \{w \mid \exists n \in \mathbb{N}. w = w_1 w_2 \ldots w_n, w_i \in L\}$$

nu repetiția *aceluiași* șir, ci concatenarea *oricăror* șiruri

luând $n = 0$, rezultă $\epsilon \in L^*$ pentru orice $L \neq \emptyset$

ϵ reprezintă *șirul vid* (niciun simbol, lungime 0)

Expresii regulate: definiție formală

O expresie regulată peste un alfabet Σ e fie:

3 cazuri de bază:

$\emptyset$	limbajul vid
ϵ	denotă limbajul $\{\epsilon\}$ (cu șirul vid)
a	denotă limbajul $\{a\}$ cu $a \in \Sigma$

3 cazuri recursive:

dacă e_1, e_2 sunt expresii regulate, atunci și următoarele sunt:

$(e_1 + e_2)$	reuniunea limbajelor
în practică, notată adesea $e_1 \vert e_2$ (alternativă, "sau")	
$(e_1 \cdot e_2)$	concatenarea limbajelor
e_1^*	închiderea Kleene a limbajului

Reguli de scriere și exemple

Omitem paranteze când sunt clare din relațiile de precedență
- cel mai prioritar: *, apoi concatenare și apoi reuniune $+$
- punctul pentru concatenare se omite

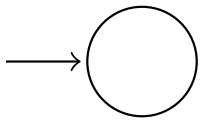
În practică se mai folosesc abrevierile
- $e?$ pentru $e + \epsilon$ (e, opțional)
- e^+ pentru $e^* \setminus \epsilon$ (e, cel puțin o dată)

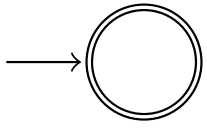
$(0 + 1)^*$ mulțimea tuturor șirurilor din 0 sau 1
$(0 + 1)^*0$ ca mai sus, încheiat cu 0 (biții pentru numere pare)
$1(0 + 1)^* + 0$ numere binare, fără zerouri inițiale

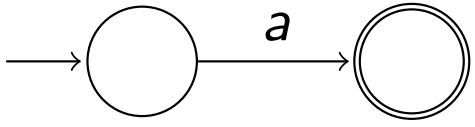
Orice expresie regulată se poate recunoaște de un automat

Construim prin *inducție structurală*

Pentru cazurile de bază

$\emptyset$  nu are stare acceptoare

ϵ  starea inițială e acceptoare

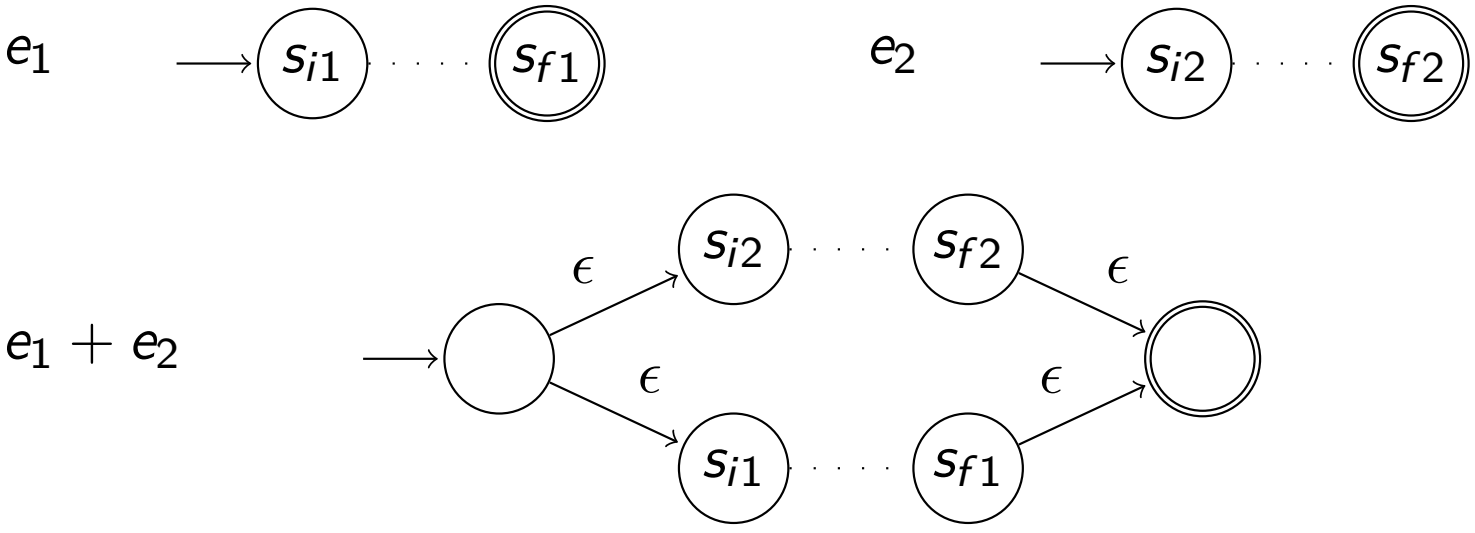
a 

acceptă simbolul a

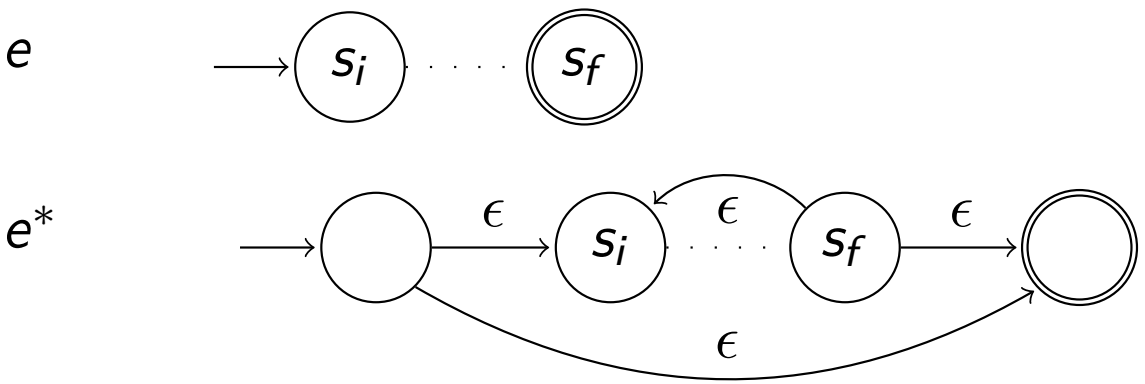
în celelalte trei cazuri, *combinăm* automatele limbajelor date
rezultă un *automat finit nedeterminist* cu tranzițiiϵ

Conversia expresie regulată $\longrightarrow$ automat

Construcția pentru reuniune

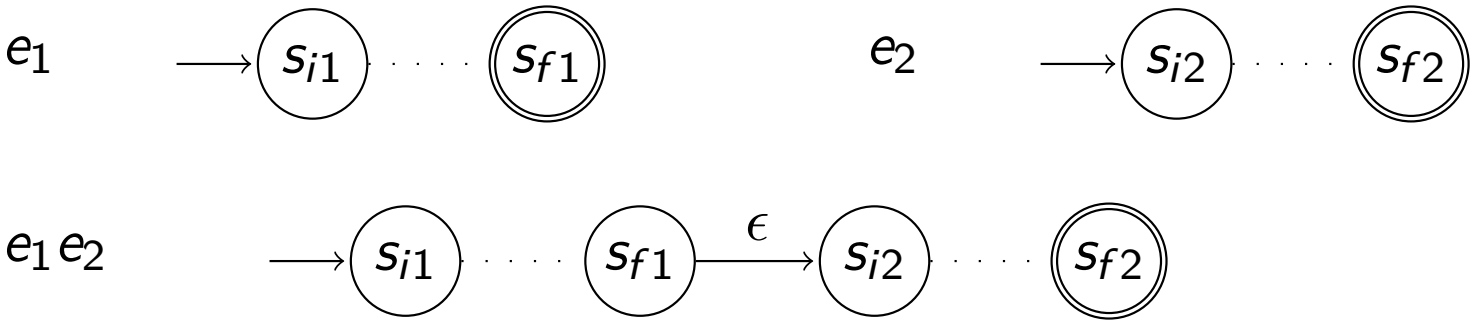


Construcția pentru închiderea Kleene



Conversia expresie regulată $\longrightarrow$ automat

Construcția pentru concatenare



În general, nu putem contopi s_{f1} și s_{i2}: ajunși în s_{f1} nu avem voie să revenim în e_1.

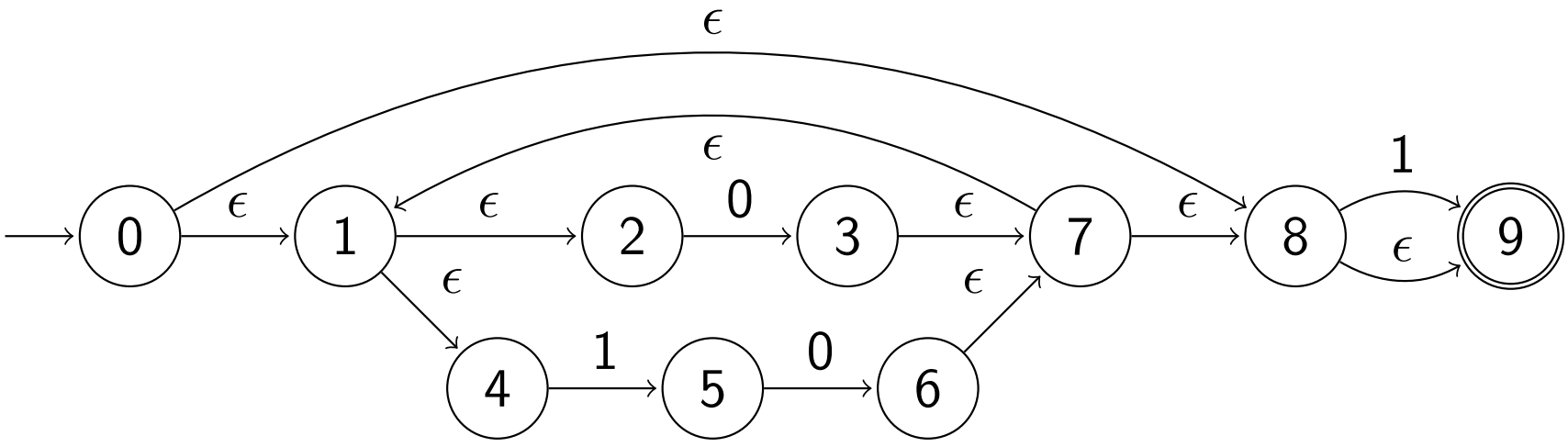
Însă construcțiile de până acum asigură:

- o *unică* stare inițială, în care nu se revine
- o *unică* stare acceptoare, din care nu ies tranziții

În aceste condiții putem simplifica și contopi capetele lui e_1 și e_2 la concatenare

Conversie din expresie regulată în automat

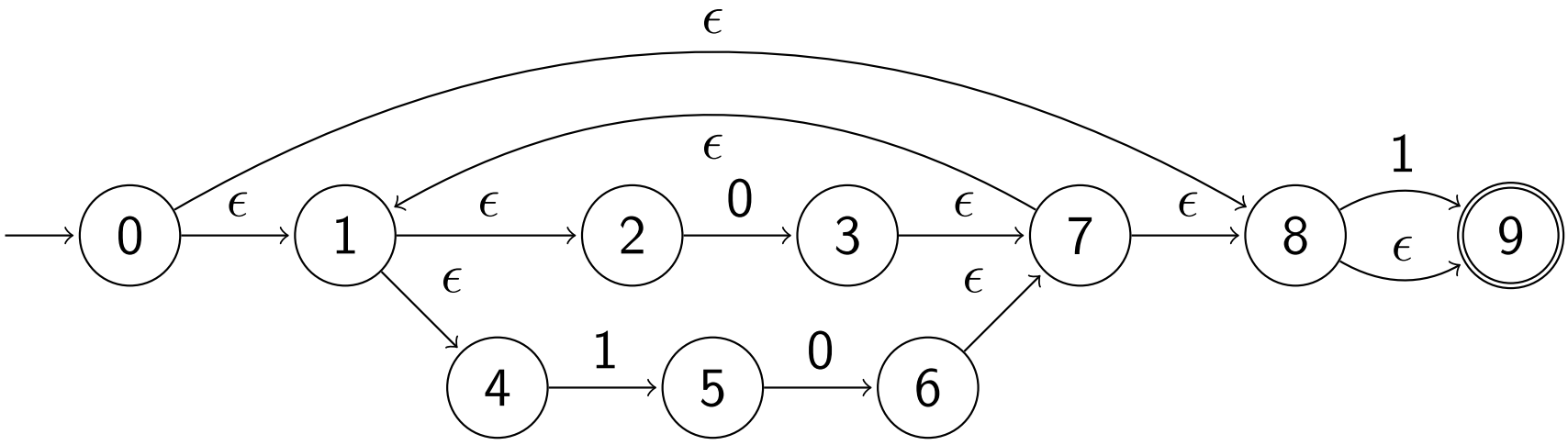
Fie expresia (0+10)*(1+ϵ). Construim (comasând pașii triviali):



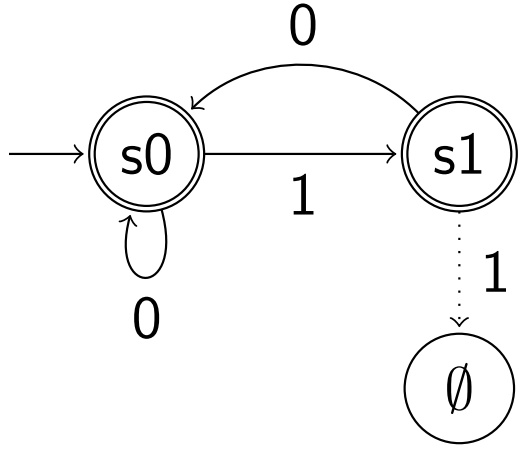
O tranziție ϵ se face spontan, fără a consuma un simbol de intrare
$\Rightarrow$ ajuns într-o stare s, e ca și ajuns în orice stare s' legată de s doar prin ϵ-tranziții (închiderea tranzitivă a relației definite de ϵ)

Deci, aflat în 0, e ca și cum s-ar afla în 1, 2, 4, 8 sau 9
Aflat în 7, e ca și cum s-ar afla în 1, 2, 4, 8, 9

Conversie din NFA cu tranziții ϵ în DFA



	0	1
012489	1234789	59
1234789	1234789	59
59	1246789	$\emptyset$
1246789	1234789	59



Liniile 1, 2 și 4 au destinații identice $\Rightarrow$ stările sunt echivalente.
$\Rightarrow$ automat cu doar două stări (ignorând starea de eroare $\emptyset$)

Conversia din automat în expresie regulată

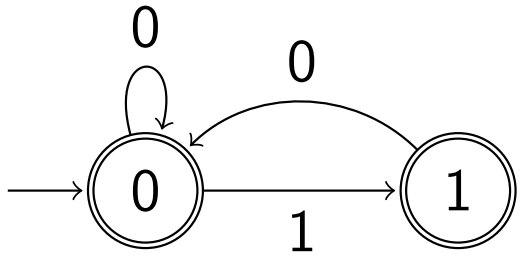
Dacă sunt mai multe noduri acceptoare, se adaugă un unic nod acceptor (cu tranziții ϵ spre el)

Se elimină pe rând fiecare nod (în afară de cel inițial și acceptor):

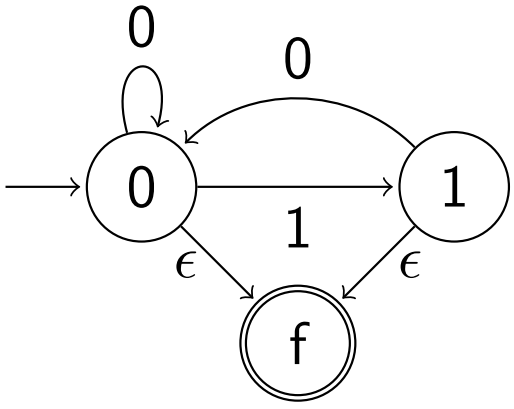
pentru orice nod intermediar i de eliminat
  pentru orice pereche (s, d)
    adaugă la muchia $s \to d$ limbajul $L_{si} L_{ii}^* L_{id}$

Conversie din automat în expresie regulată

Șiruri de 0 și 1 care nu au doi 1 consecutivi pe 1, trece în stare cu tranziție doar pe 0



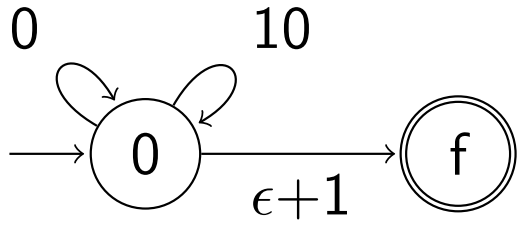
Ambele stări sunt acceptoare $\Rightarrow$ adăugăm o unică stare acceptoare



Eliminăm 1:

$0 \xrightarrow{10} 0$

$0 \xrightarrow{1\epsilon} f$



Obținem astfel limbajul (0+10)*(1+ϵ)