Bibliographie

- Danièle BEAUQUIER, Jean BERSTEL et Philippe CHRETIENNE:

Eléments d'algorithmique, Masson 1992

(ce livre est épuisé, mais téléchargeable sur le Web à l'adresse

http://www-igm.univ-mlv.fr/~berstel/Elements/Elements.html)

Il comporte plusieurs chapitres qui peuvent (doivent ?) vous intéresser à différents titres (cours d'algorithmique, maths discrètes) et le chapitre 9 qui concerne les automates.

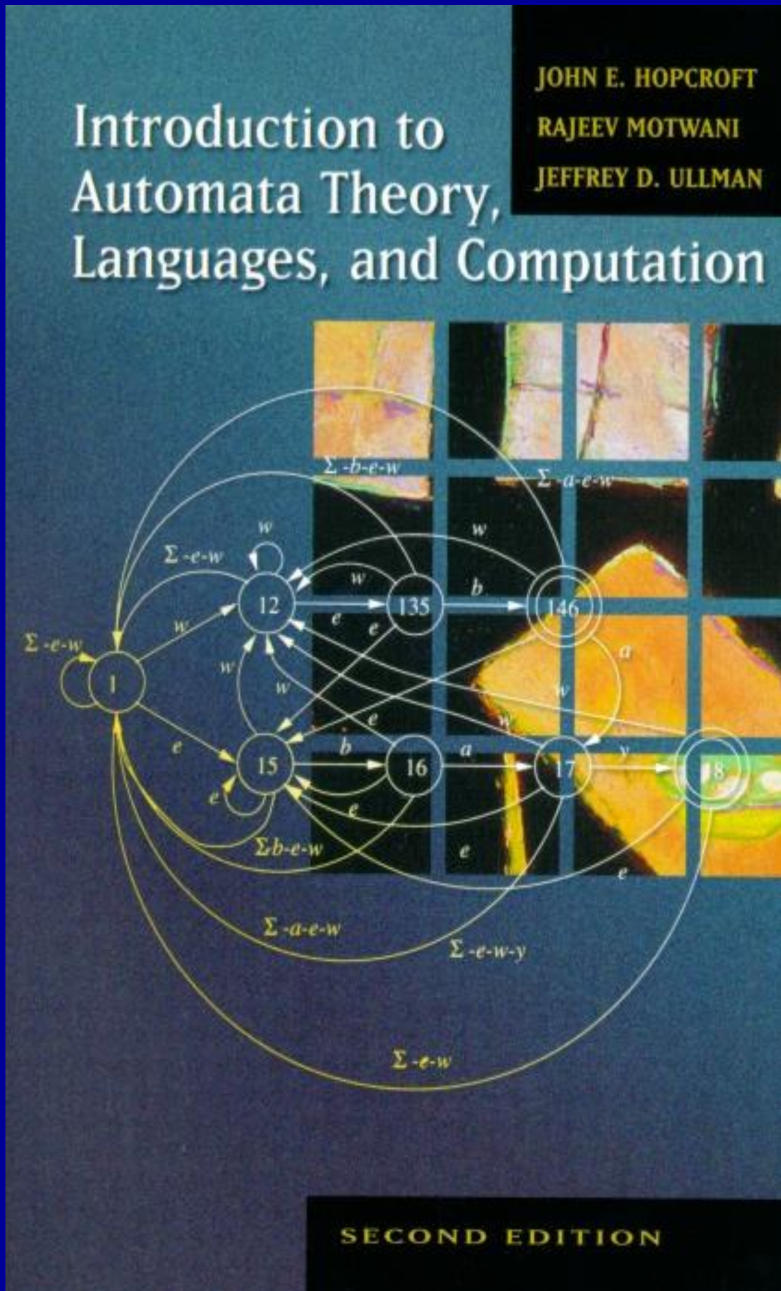
Bibliographie

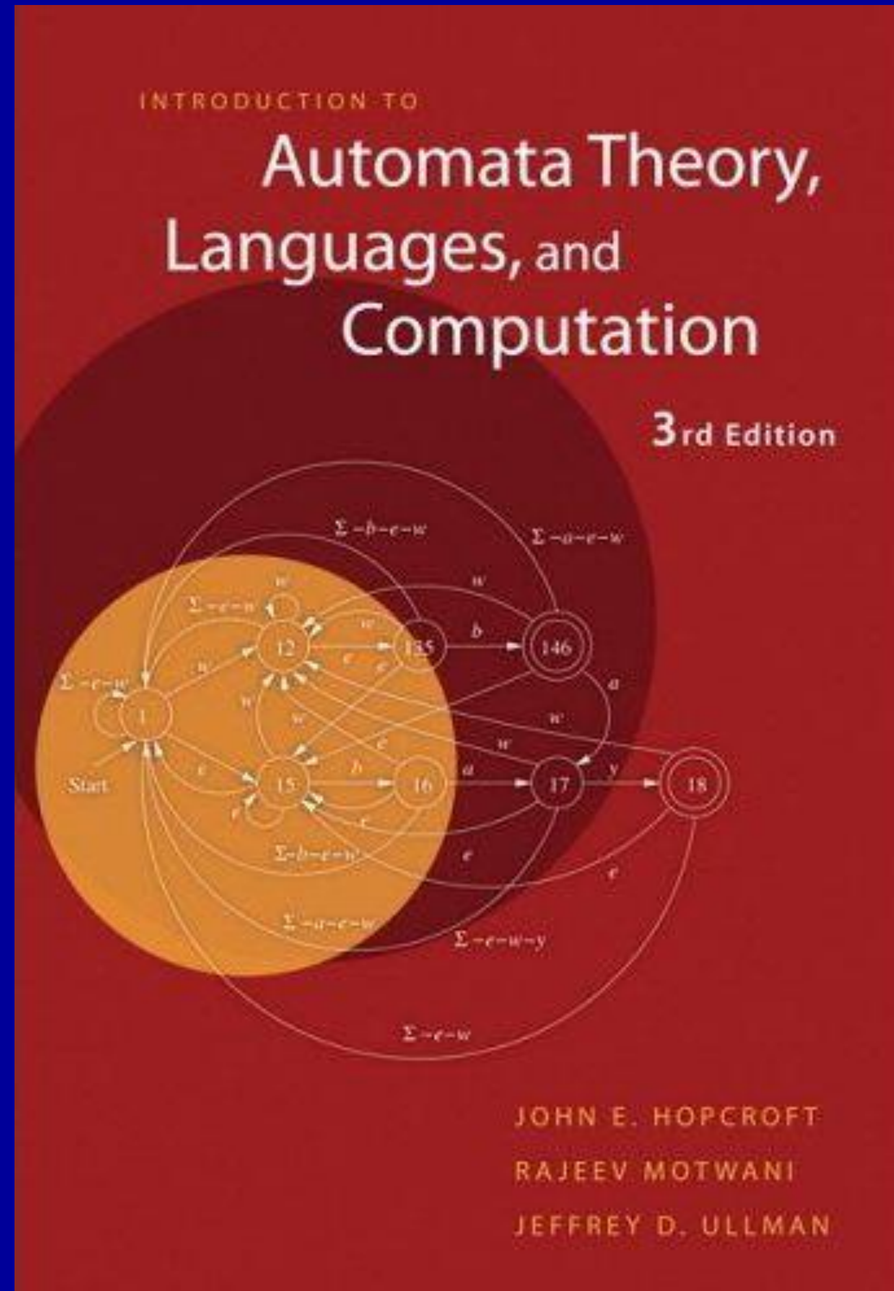
- John HOPCROFT, Jeffrey ULLMAN : *Introduction to Automata Theory and Computation*, Addison Wesley, 1979.

Nouvelle édition, revue et corrigée :

- John HOPCROFT, Rajeev MOTWANI, Jeffrey ULLMAN : *Introduction to Automata Theory, Languages and Computation*, Addison Wesley, 2001.

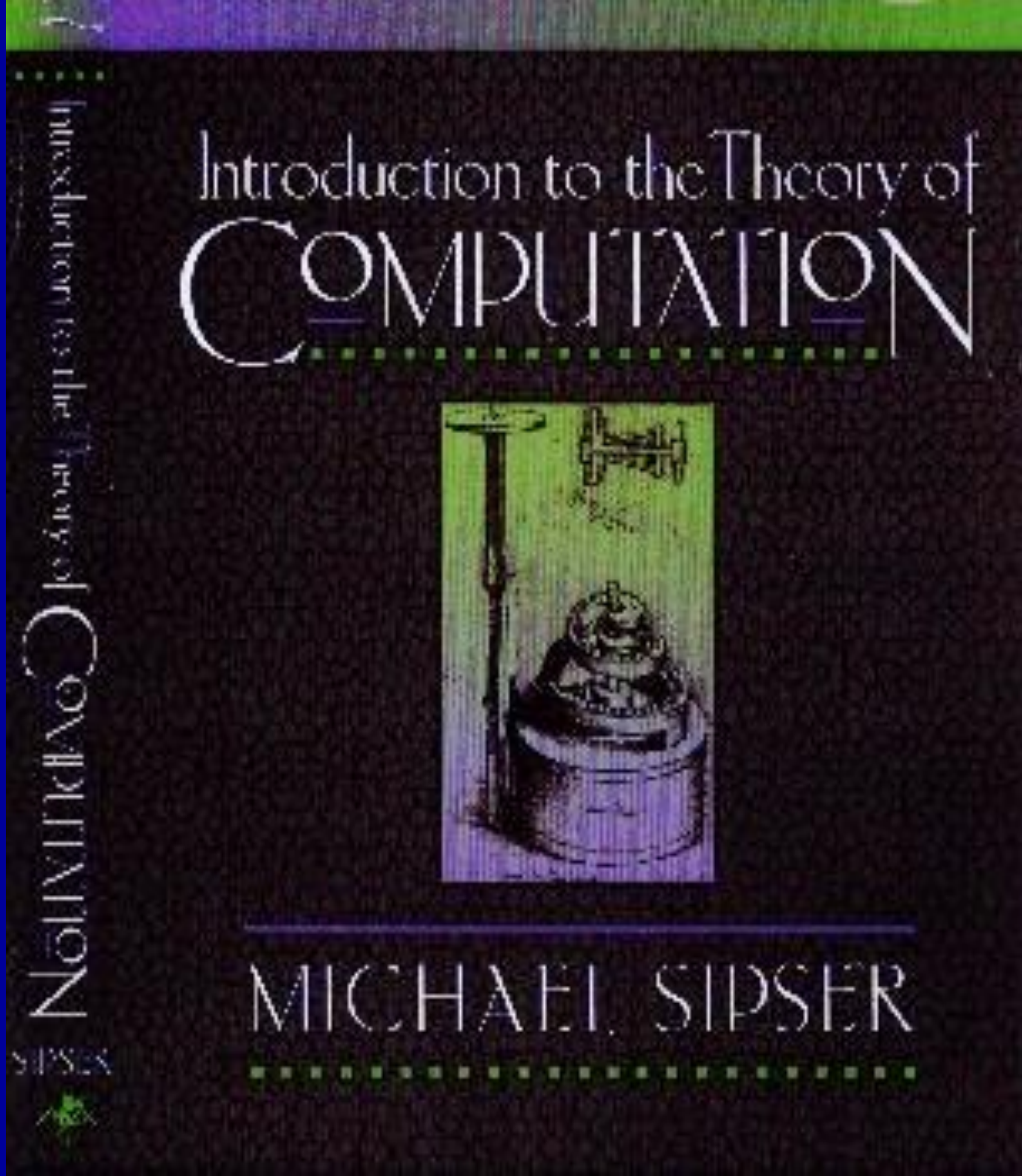
  Dernière révision 2006.


Introduction to
Automata Theory,
Languages, and Computation
JOHN E. HOPCROFT
RAJEEV MOTWANI
JEFFREY D. ULLMAN
SECOND EDITION


INTRODUCTION TO
Automata Theory,
Languages, and
Computation
3rd Edition
JOHN E. HOPCROFT
RAJEEV MOTWANI
JEFFREY D. ULLMAN

Bibliographie

- Michael SIPSER : *Introduction to the Theory of Computation*, PWS publishing comp. 1997. (nouvelle édition : mai 2012)

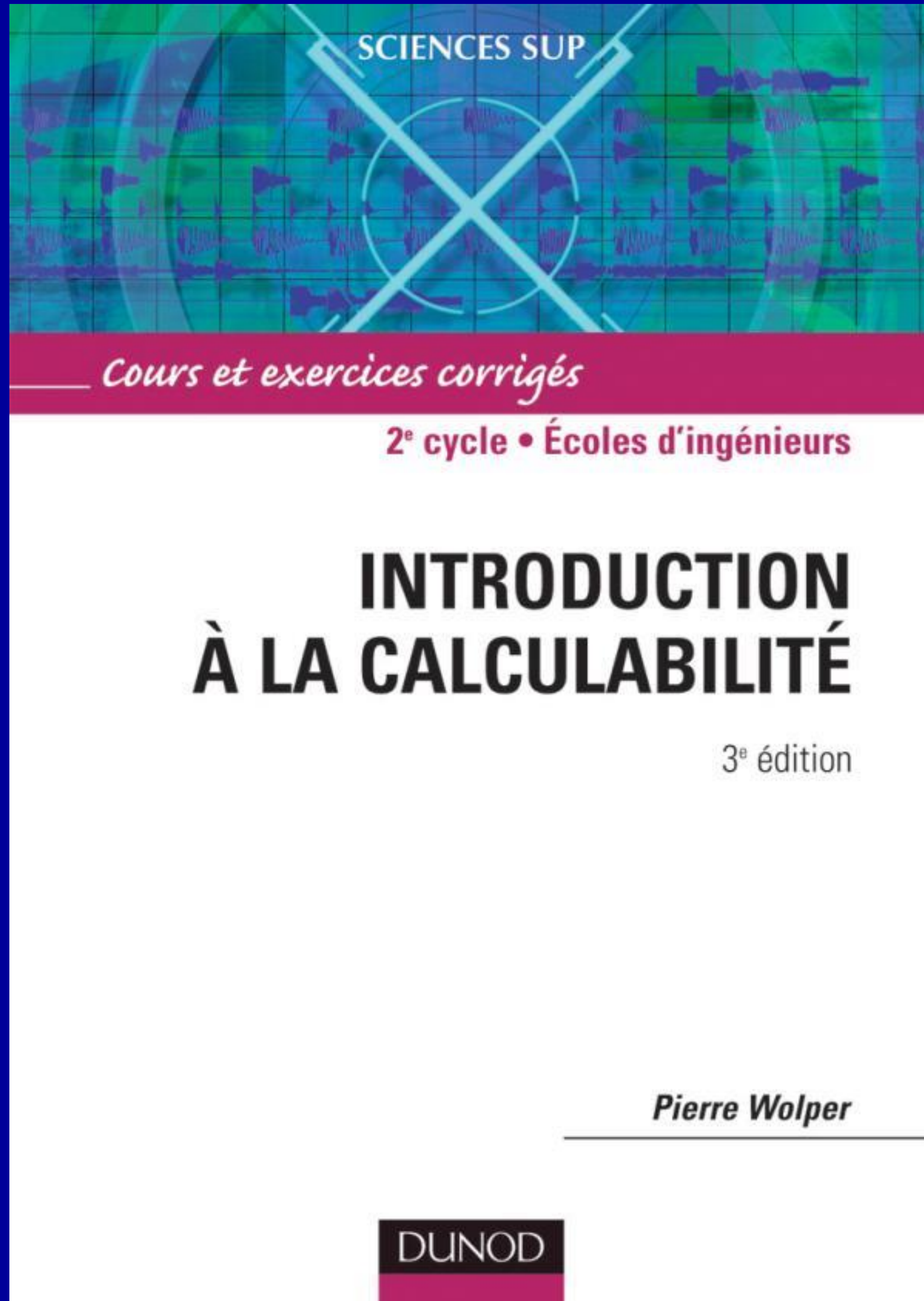

Introduction to the Theory of
COMPUTATION
MICHAEL SIPSER
Introduction to the Theory of COMPUTATION
SIPSER

Bibliographie

- Jacques STERN : *Fondements mathématiques de l'informatique*, McGraw Hill, 1990.

Bibliographie

- Pierre WOLPER : *Introduction à la calculabilité*, Inter Éditions 1991 (troisième édition : Dunod, 2006).


SCIENCES SUP
Cours et exercices corrigés
2e cycle • Écoles d'ingénieurs
INTRODUCTION
À LA CALCULABILITÉ
3e édition
Pierre Wolper
DUNOD

Concaténation

- Σ^* = collection de tous les mots finis sur Σ
  = ensemble de tous les mots finis
- Opération interne associée: concaténation "."
  $$\Sigma^* \times \Sigma^* \to \Sigma^*$$
  $$(u,v) \to u.v$$
  u =ES, v =SI, $u.v$ = ESSI
- Élément neutre: mot vide $m.\varepsilon = \varepsilon.m = m$
- concaténation = opération associative :
  $$(u.v).w = u.(v.w)$$
- $(\Sigma^*,.)$ est un monoïde

Monoïde

de Wikipédia :

- Un **monoïde** est une structure algébrique consistant en un ensemble muni d'une loi de composition interne associative et d'un élément neutre.
- En d'autres termes, $(E, *)$ est un **monoïde** si :
  - $\forall\ x,y \in E,\ x*y \in E$ (composition interne)
  - $\forall\ x,y,z \in E,\ x*(y*z) = (x*y)*z$ (associativité)
  - $\exists\ e \in E$ t.q. : $\forall\ x \in E,\ x*e=e*x=x$

Autre vision des langages

- Langage = ensemble de mots (infini?)
- Langage = sous-ensemble de Σ^*
  - ensemble des nombres ordinaires
  - ensemble des programmes Java (syntaxiquement corrects)
- Langage vide $L = \{ \} = \emptyset \neq \{\varepsilon\}$
- $L = \{\varepsilon\}$, langage du mot vide
- Langage fini de mots finis

$$L=\{ab,ba,aca\}$$

- Langage infini dénombrable de mots finis

$$L=\{\text{mots binaires pairs}\}$$

Opération * de Kleene

- L un langage, L^* = concaténation de mots de L

  $L^0 = \{\varepsilon\}$, $L^1 = L$, $L^{i+1} = L^i . L \quad \forall i \geq 0$

  $L^* = \cup_{i \geq 0} L^i$, $L^+ = \cup_{i \geq 1} L^i$
- $L=\{a,0\}$
  $L^2=\{aa,a0,0a,00\}$
  $L^3=\{aaa,aa0,a0a,a00,0aa,0a0,00a,000\}$
- L^* = plus petit langage de Σ^* clos pour la concaténation contenant ε et L.
  C'est un sous-monoïde de Σ^*
- Opération idempotente: $(L^*)^*=L^*$

Langages rationnels

- Intérêt particulier pour la suite
- sous-ensemble de l'ensemble des langages
- définition inductive
- Notation simplifiée par expressions rationnelles (recherche sur le Web, etc…)

Définition inductive

- Base :
  - $\emptyset$ est un langage rationnel
  - $\{\varepsilon\}$ est un langage rationnel
  - $\forall a \in \Sigma$, $\{a\}$ est un langage rationnel
- Induction :
  - Si R et S sont deux langages rationnels,

$$R \cup S \,,\, R \,.\, S \text{ et } R^*$$

sont aussi rationnels

Expressions rationnelles (ER)

- Base :
  - $\emptyset$ est une expression rationnelle (ER)
  - ε est une ER qui représente $\{\varepsilon\}$
  - $\forall a \in \Sigma$, a est un ER qui représente {a} (le mot a)
- Induction : Si *r* et *s* sont des ER,
  - ($r + s$) est une ER qui représente $R \cup S$
  - (rs) est une ER qui représente $R.S$
  - (r^*) est une ER qui représente R^*

Exemples

- $(a+b)^*$ tous les mots avec des a et des b
- $(a+b)^*ab(a+b)^*=(b^*a^*)^*ab(a+b)^*$
- $(b+ba)^*$ mots sans facteur aa et qui ne commencent pas par un a
- $(a+\varepsilon)(b+ba)^*$ mots sans facteur aa

Résumé

ensembliste

syntaxique

Langages rationnels

Expressions rationnelles

Définition inductive

Définition inductive

Le théorème de Kleene

Théorème de Kleene

- $Rat(\Sigma^*)$=classe des ER sur Σ
- $Rec(\Sigma^*)$=classe des langages reconnus par AF sur Σ

Théorème : Un langage sur Σ est rationnel si et seulement si il est reconnu par un automate fini.

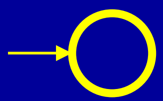
- On veut montrer que $Rat(\Sigma^*) \subseteq Rec(\Sigma^*)$
  i.e. étant donnée une ER, on peut construire un AF qui la reconnaît
- Et que $Rec(\Sigma^*) \subseteq Rat(\Sigma^*)$
  i.e. étant donné un AF, on peut trouver une ER qui le dénote (prochaine fois)

Preuve $Rat(\Sigma^*) \subseteq Rec(\Sigma^*)$

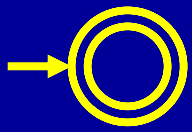
- Par induction sur le nombre d'opérateurs de l'ER

- Base

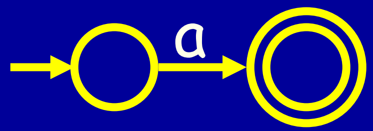
  - $\emptyset$ est une ER,



  - ε est une ER,

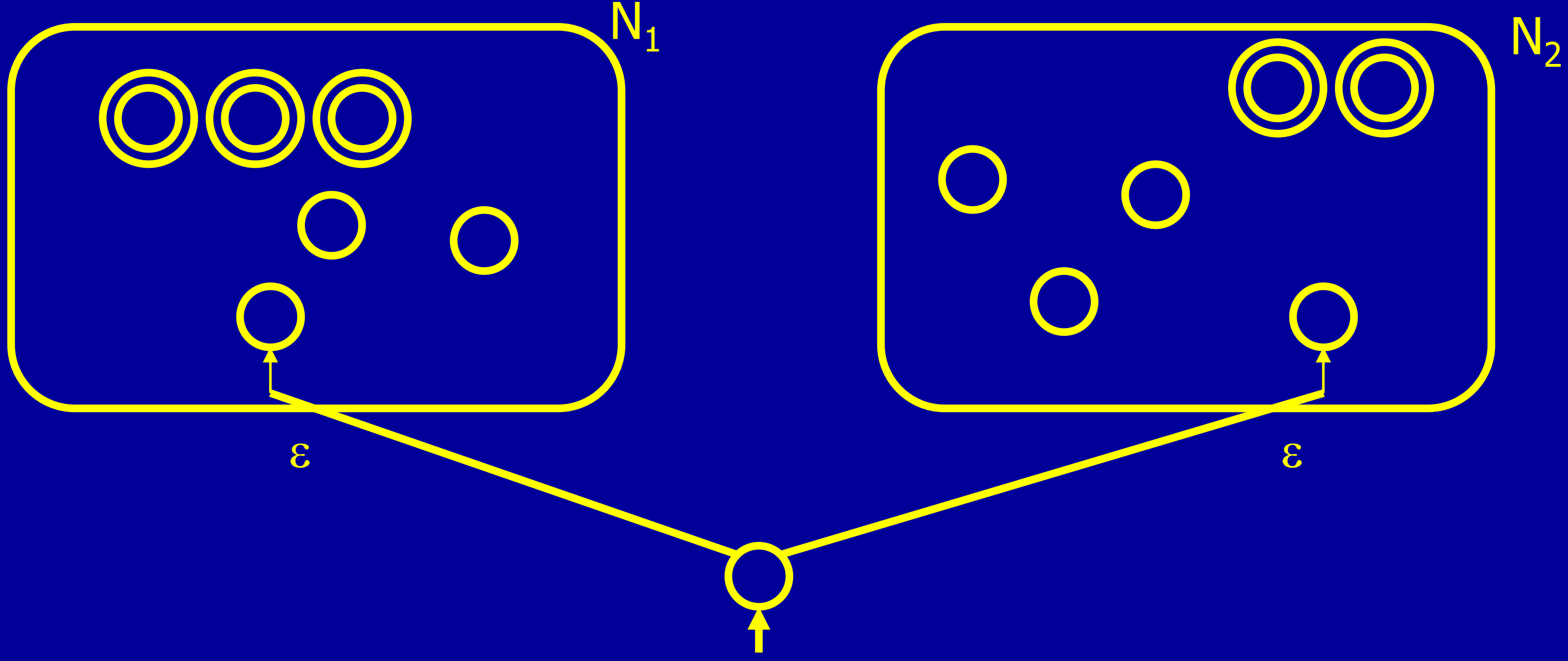


  - $\forall a \in S$, a est une ER



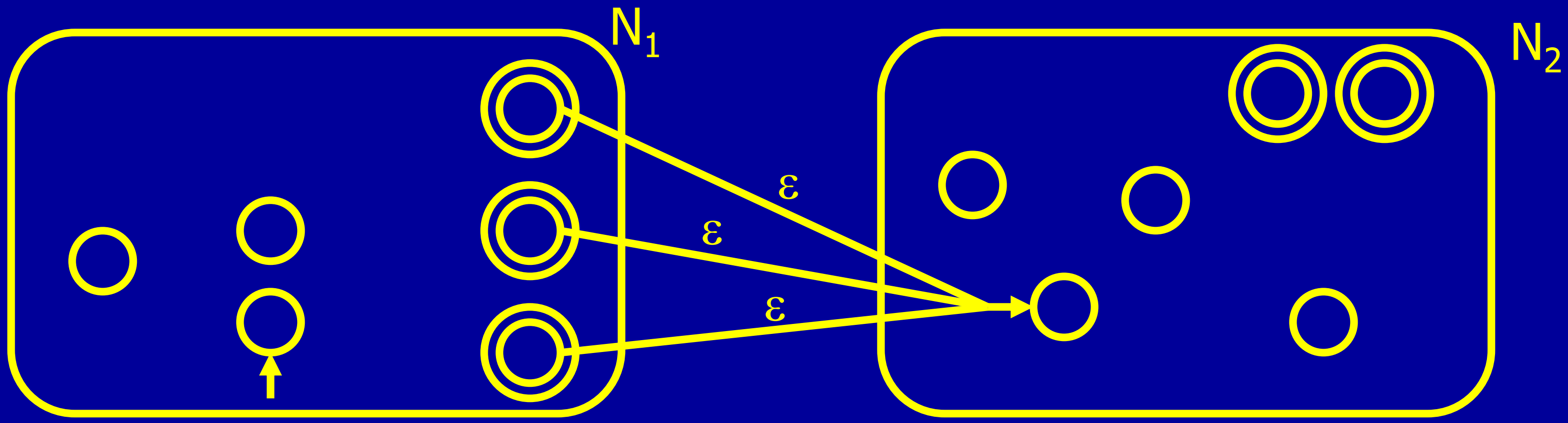
Preuve pour t=(r+s)

- r et s ont strictement moins d'opérateurs que t; par HR, il existe N_1 et N_2, deux AFND tq $L(N_1)=r$ et $L(N_2)=s$.



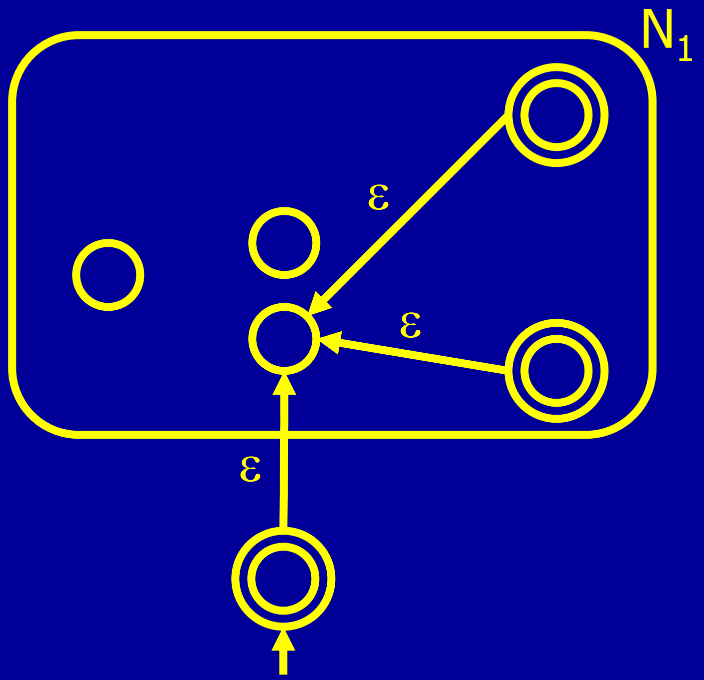
Preuve pour t=(r.s)

- r et s ont strictement moins d'opérateurs que t; par HR, il existe N_1 et N_2, deux AFND tq $L(N_1)=r$ et $L(N_2)=s$.



Preuve pour t=(r)*

- r a strictement moins d'opérateurs que t; par HR, il existe N_1 un AFND tq $L(N_1)$=r.



Le théorème de Kleene

- $Rec(\Sigma^*)$ = langages reconnus par AF
- $Rat(\Sigma^*)$ = ensemble des ER (construit inductivement)
  - Base : $\emptyset$, ε, et $a \in \Sigma$ sont des ER
  - Induction : *r* et *s* des ER, $(r+s)$, $(r.s)$ et $(r)^*$ sont des ER

Théorème : Un langage sur Σ est rationnel si et seulement si il est reconnu par un automate fini.

- On a montré (cours 2) que $Rat(\Sigma^*) \subseteq Rec(\Sigma^*)$
  (étant donnée une ER, on peut construire un AF qui la reconnaît)
- On montre la réciproque : $Rec(\Sigma^*) \subseteq Rat(\Sigma^*)$
  (étant donné un AF, on peut trouver une ER qui le dénote)

Le problème

- **Donnée** : *A* un automate fini déterministe
- **Problème** : trouver une expression rationnelle qui représente le langage reconnu par *A*, L(*A*).

Idée de résolution

Étant donné un AFD

- On considère les chemins de *i* vers tout *t* $\in T$
- L'ER correspondant à chacun de ces chemins est obtenue en concaténant les étiquettes des transitions en traitant les boucles par une *
- L'ER finale est l'union des différentes ER ainsi obtenues.

Idée de résolution (suite)

- Arcs étiquetés par des lettres et il faut prendre en compte les boucles.
- Mots reconnus en partant de *i* et arrivant dans l'état *j* en ne passant que par les états $\{1,2,\ldots,k\}$:

$$R_{ij}^{k} = \{m \in \Sigma^* \mid \delta(i,m) = j \text{ et } \forall p <_{pref} m,\ \delta(i,p) = n,\ n \leq k\}$$

- Algorithme de McNaughton-Yamada

Intuitivement

R_{ij}^{k} = ensemble des mots permettant d'aller de *i* à *j* en ne passant que par {1,…,k}. Ces mots sont soit

- dans R_{ij}^{k-1} i.e. ils ne passent que par états ≤ *k*-1
- composés de R_{ik}^{k-1}
  (mènent *A* dans l'état *k* pour la première fois)
  suivis de l'itération des mots de R_{kk}^{k-1}
  (forment un cycle pour *k* sans passer par des états d'un numéro supérieur à *k*)
  suivis des mots de R_{kj}^{k-1}
  (qui mènent *A* de l'état *k* à l'état *j*).

Définition inductive des R_{ij}^k

- **Base** :
  - $R_{ij}^0=\{a \mid \delta(i,a)=j\}$ pour $i \neq j$

    R_{ij}^0 peut être $\varnothing$ si la transition n'est pas définie
  - $R_{ii}^0= \{a \mid \delta(i,a)=i\} \cup \{\varepsilon\}$

    $R_{ii}^0 = \varepsilon$ si i sans boucle

- **Règle** :
  - $R_{ij}^{k} = R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1}$

Reste à prouver que les R_{ij}^k sont rationnels !

les R_{ij}^k sont rationnels

$R_{ij}^0 = \{a : \delta(i,a)=j\}$ pour $i \neq j$
$R_{ii}^0 = \{a : \delta(i,a)=i\} \cup \{\varepsilon\}$
$R_{ij}^k = R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1}$

- On montre par induction sur ***k***, que pour chaque ***i,j,k*** il existe r_{ij}^k, ER qui représente le langage R_{ij}^k

- **Base** : R_{ij}^0 : ensemble fini de chaînes composées soit de $a \in \Sigma$ soit de ε
  - pour i=j : $r_{ii}^0 = \varepsilon + a_1 + \ldots + a_p$ (ε, s'il n'y a pas de boucle sur *i*)
  - pour $i \neq j$: $r_{ij}^0 = a_1 + \ldots + a_p$ $\{a_1, \ldots, a_p\} = \{a \in \Sigma : \delta(i,a)=j\}$ ($\emptyset$, s'il n'y a pas de transition de *i* vers *j*.)

les R_{ij}^k sont rationnels

$R_{ij}^0 = \{a : \delta(i,a)=j\}$ pour $i \neq j$
$R_{ii}^0 = \{a : \delta(i,a)=i\} \cup \{\varepsilon\}$
$R_{ij}^k = R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1}$

- **Induction** (HR) : pour tout *l* et *m* r_{lm}^{k-1}, est une ER qui représente R_{lm}^{k-1}. L'ER pour r_{ij}^k est

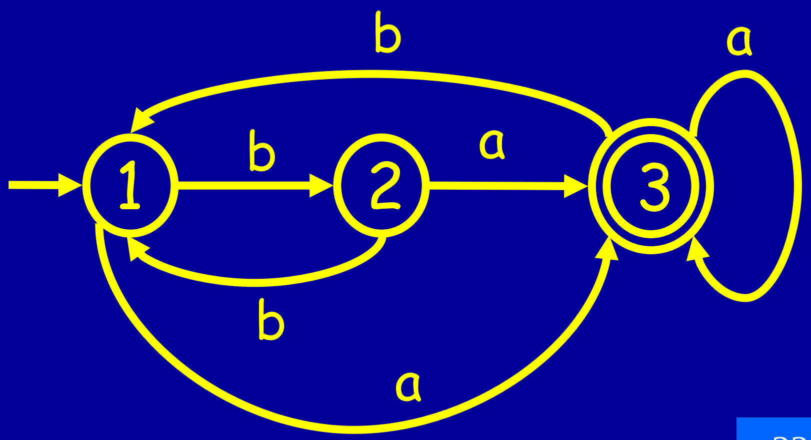
$$r_{ij}^k = r_{ik}^{k-1}.(r_{kk}^{k-1})^*.\ r_{kj}^{k-1} + r_{ij}^{k-1}$$

- R_{1j}^n représente les chemins qui conduisent de l'état initial (état 1) vers les états de reconnaissance de *A*, l'ER qui représente L(*A*) est :

$$\Sigma_{m \in F}\, r_{1m}^n$$

Exemple

$R_{ij}^0 = \{a : \delta(i,a)=j\}$ pour $i \neq j$
$R_{ii}^0 = \{a : \delta(i,a)=i\} \cup \{\varepsilon\}$
$R_{ij}^k = R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1}$



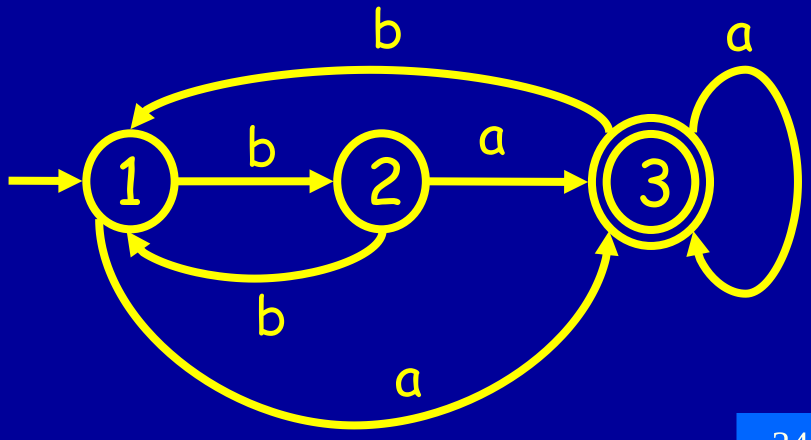
	k=0	k=1	k=2	k=3
r_{11}^k				
r_{12}^k				
r_{13}^k				
r_{21}^k				
r_{22}^k				
r_{23}^k				
r_{31}^k				
r_{32}^k				
r_{33}^k				

Exemple

$R_{ij}^{0}= \{a : \delta(i,a)=j \}$ pour $i\neq j$
$R_{ii}^{0}= \{a : \delta(i,a)=i \} \cup \{\varepsilon\}$
$R_{ij}^{k} = R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1}$

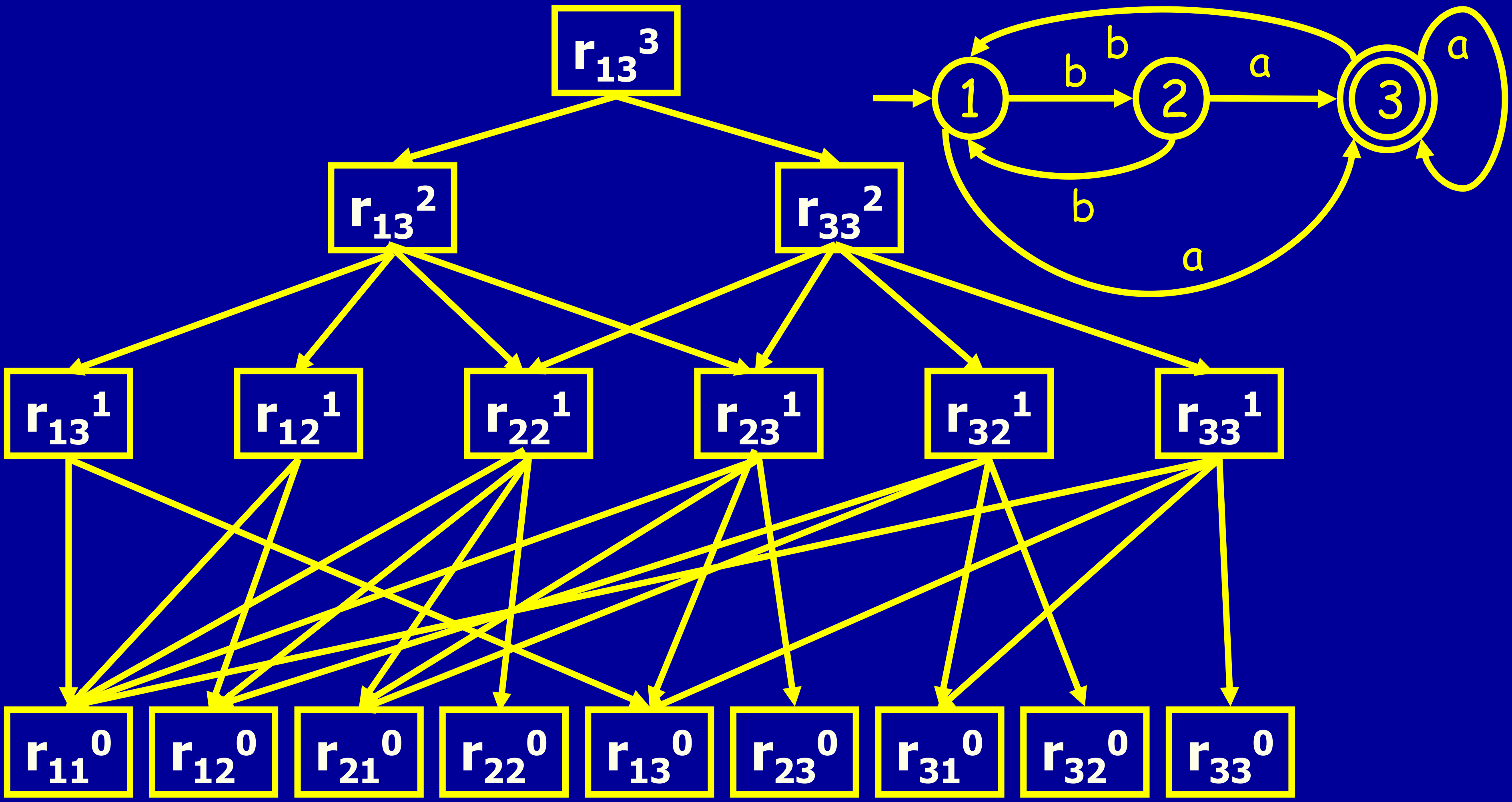
- $r_{13}^{3}=r_{13}^{2}(r_{33}^{2})^*r_{33}^{2}+r_{13}^{2}=r_{13}^{2}((r_{33}^{2})^{+}+\varepsilon)=r_{13}^{2}(r_{33}^{2})^*$
- $r_{13}^{2} = r_{12}^{1}(r_{22}^{1})^* r_{23}^{1} + r_{13}^{1}$
- $r_{33}^{2} = r_{32}^{1}(r_{22}^{1})^* r_{23}^{1} + r_{33}^{1}$

- $r_{12}^{1} = r_{11}^{0}(r_{11}^{0})^* r_{12}^{0}+ r_{12}^{0} = (r_{11}^{0})^* r_{12}^{0}$
- $r_{13}^{1} = r_{11}^{0}(r_{11}^{0})^* r_{13}^{0}+ r_{13}^{0} = (r_{11}^{0})^* r_{13}^{0}$
- $r_{22}^{1} = r_{21}^{0}(r_{11}^{0})^* r_{12}^{0}+ r_{22}^{0}$
- $r_{23}^{1} = r_{21}^{0}(r_{11}^{0})^* r_{13}^{0}+ r_{23}^{0}$
- $r_{32}^{1} = r_{31}^{0}(r_{11}^{0})^* r_{12}^{0}+ r_{32}^{0}$
- $r_{33}^{1} = r_{31}^{0}(r_{11}^{0})^* r_{13}^{0}+ r_{33}^{0}$



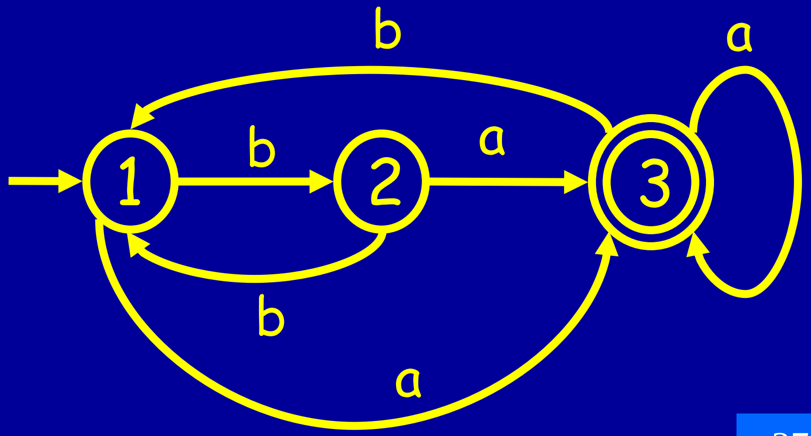
Exemple

$R_{ij}^0 = \{a : \delta(i,a)=j\}$ pour $i \neq j$
$R_{ii}^0 = \{a : \delta(i,a)=i\} \cup \{\varepsilon\}$
$R_{ij}^k = R_{ik}^{k-1}(R_{kk}^{k-1})^* R_{kj}^{k-1} \cup R_{ij}^{k-1}$



	k=0	k=1	k=2	k=3
r_{11}^k				
r_{12}^k				
r_{13}^k				
r_{21}^k				
r_{22}^k				
r_{23}^k				
r_{31}^k				
r_{32}^k				
r_{33}^k				

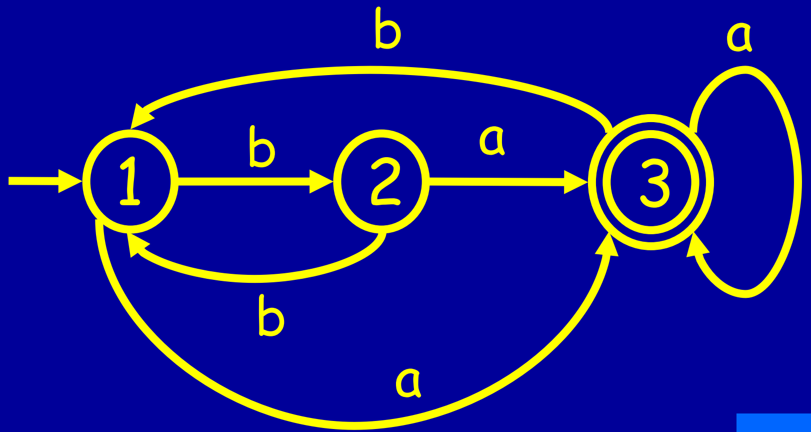
- **Base** : R_{ij}^0 : ensemble fini de chaînes composées soit de $a \in \Sigma$ soit de ε
  - pour i=j : $r_{ii}^0 = \varepsilon + a_1 + \ldots + a_p$ (ε, s'il n'y a pas de boucle sur *i*)
  - pour $i \neq j$: $r_{ij}^0 = a_1 + \ldots + a_p$ $\{a_1, \ldots, a_p\} = \{a \in \Sigma : \delta(i,a) = j\}$
    ($\emptyset$, s'il n'y a pas de transition de *i* vers *j*.)
- On obtient :
  - $r_{32}^0 = \emptyset$
  - $r_{11}^0 = r_{22}^0 = \varepsilon$
  - $r_{13}^0 = r_{23}^0 = a$
  - $r_{12}^0 = r_{21}^0 = r_{31}^0 = b$
  - $r_{33}^0 = \varepsilon + a$



	k=0	k=1	k=2	k=3
r_{11}^k	ε			
r_{12}^k	b			
r_{13}^k	a			
r_{21}^k	b			
r_{22}^k	ε			
r_{23}^k	a			
r_{31}^k	b			
r_{32}^k	∅			
r_{33}^k	ε+a			

$$r_{ij}^{k} = r_{ik}^{k-1}.(r_{kk}^{k-1})^*.\ r_{kj}^{k-1} + r_{ij}^{k-1}$$

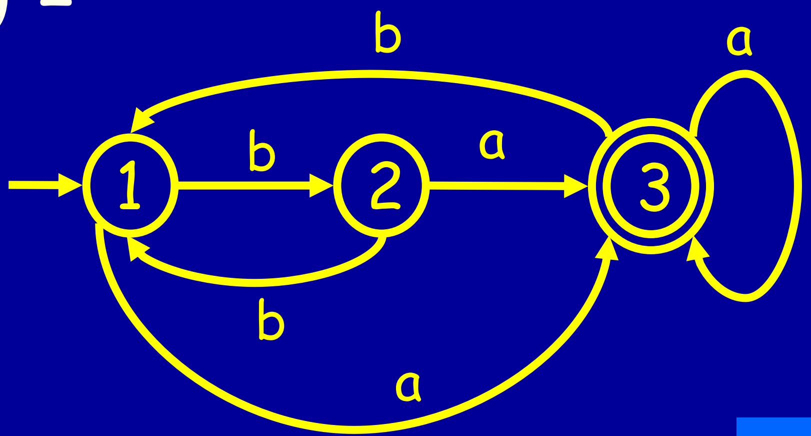
- $r_{12}^{1} = (r_{11}^{0})^*\ r_{12}^{0} = \varepsilon^* b = b$
- $r_{13}^{1} = (r_{11}^{0})^*\ r_{13}^{0} = \varepsilon^* a = a$
- $r_{22}^{1} = r_{21}^{0}(r_{11}^{0})^*\ r_{12}^{0} + r_{22}^{0} = b\varepsilon^* b+\varepsilon\ =\ \varepsilon+bb$
- $r_{23}^{1} = r_{21}^{0}(r_{11}^{0})^*\ r_{13}^{0} + r_{23}^{0} = b\varepsilon^* a+a\ =\ \mathbf{a}+ba$
- $r_{32}^{1} = r_{31}^{0}(r_{11}^{0})^*\ r_{12}^{0} + r_{32}^{0} = b\varepsilon^* b+\emptyset = bb$
- $r_{33}^{1} = r_{31}^{0}(r_{11}^{0})^*\ r_{13}^{0} + r_{33}^{0} = b\varepsilon^* a+\varepsilon+a = \varepsilon+a+ba$



	k=0	k=1	k=2	k=3
r_{11}^k	ε			
r_{12}^k	b	b		
r_{13}^k	a	a		
r_{21}^k	b			
r_{22}^k	ε	ε+bb		
r_{23}^k	a	a+ba		
r_{31}^k	b			
r_{32}^k	∅	bb		
r_{33}^k	ε+a	ε+a+ba		

$$r_{ij}^{k} = r_{ik}^{k-1}.(r_{kk}^{k-1})^{*}.\ r_{kj}^{k-1} + r_{ij}^{k-1}$$

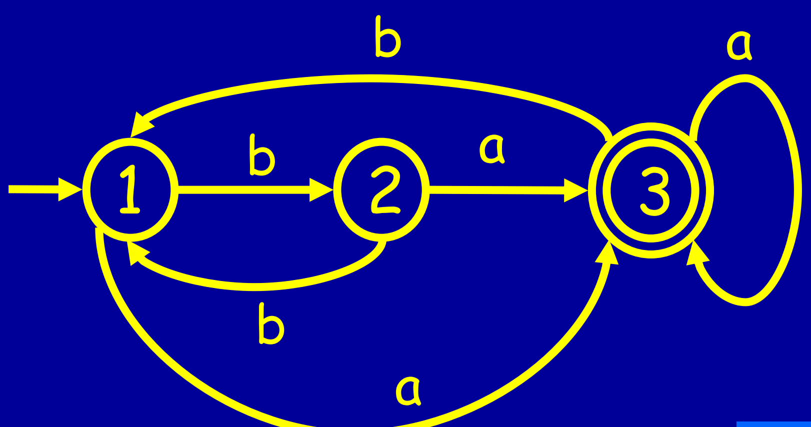
- $r_{13}^{2} = r_{12}^{1}(r_{22}^{1})^{*}\ r_{23}^{1} + r_{13}^{1} =$
  $b(\varepsilon+bb)^{*}(a+ba)+a =$
  $b(bb)^{*}(\varepsilon+b)a+a =$
  $b^{+}a+a =$
  $(b^{+}+\varepsilon)a = b^{*}a$
- $r_{33}^{2} = r_{32}^{1}(r_{22}^{1})^{*}\ r_{23}^{1} + r_{33}^{1} =$
  $bb(\varepsilon+bb)^{*}(a+ba)+(\varepsilon+a+ba) =$
  $b^{+}ba+ba+a+\varepsilon =$
  $b^{*}a+\varepsilon$



	k=0	k=1	k=2	k=3
r_{11}^k	ε			
r_{12}^k	b	b		
r_{13}^k	a	a	b*a	
r_{21}^k	b			
r_{22}^k	ε	ε+bb		
r_{23}^k	a	a+ba		
r_{31}^k	b			
r_{32}^k	∅	bb		
r_{33}^k	ε+a	ε+a+ba	ε+b*a	

$$r_{ij}^{k} = r_{ik}^{k-1} . (r_{kk}^{k-1})^* . r_{kj}^{k-1} + r_{ij}^{k-1}$$

- $r_{13}^{3} = r_{13}^{2}(r_{33}^{2})^* =$
  $b^*a\,(\varepsilon + b^*a)^* =$
  $(b^*a)^+$



	k=0	k=1	k=2	k=3
r_{11}^k	ε			
r_{12}^k	b	b		
r_{13}^k	a	a	b*a	b^*a^+
r_{21}^k	b			
r_{22}^k	ε	ε+bb		
r_{23}^k	a	a+ba		
r_{31}^k	b			
r_{32}^k	∅	bb		
r_{33}^k	ε+a	ε+a+ba	ε+b*a	

Complexité

- Il faut calculer pour k=0,1,2,…,*n*
  - Pour chaque paire d'états

$$R_{ij}^{k} = R_{ik}^{k-1}(R_{kk}^{k-1})^{*} R_{kj}^{k-1} \cup R_{ij}^{k-1}$$

- Soit *n* fois pour chaque paire d'états
- Au total n^3 opérations

- Complexité $O(n^3)$

(p.e. le cas où il y a O(n) états d'acceptation …)

Questions

- Est-ce que l'ER dépend de la numérotation des états?

Non, puisqu'on parcourt la totalité du graphe

- Est-ce qu'il faut tout calculer?

Seulement ce qui nous sert…

$Rec(\Sigma^*) = Rat(\Sigma^*)$

- On a donc montré que les langages rationnels sont reconnus par AF et seulement par ceux-ci

- Les AF **caractérisent** les langages rationnels