Introduction aux langages formels

Stages LIESSE IPP

David A. Madore
Télécom Paris
david.madore@enst.fr

27 avril 2021

http://perso.enst.fr/madore/20210427-liesse.pdf

Git: cdf8d68 Fri Apr 30 12:21:34 2021 +0200

Plan

Qu'est-ce que la théorie des langages ?

- Discipline à l'interface entre **mathématiques** discrètes (algèbre, combinatoire, logique), **informatique** (décidabilité, algorithmique) et même **linguistique** (étude des langues naturelles).

- Cherche à développer des outils algorithmiques et concepts abstraits pour l'étude des **chaînes de caractères** (= **mots**) et fichiers texte, p.ex. :
  - recherche de motifs / remplacement,
  - différences, substitutions,
  - analyse syntaxique (p.ex. avant compilation).

- Cet exposé se concentrera sur deux aspects (classes de langages) :
  - langages et expressions rationnels, automates finis,
  - grammaires hors contexte, automates à pile.

Quelques noms

Sélection *très* partielle !

- **S. C. Kleene** (1956) : étude du concept d'automate, notion de langages (« événements ») rationnels, équivalence rationnel⇔reconnaissable.

- **N. Chomsky** : « Three models for the description of language » (1956) : *hiérarchie de Chomsky*.

- **M. O. Rabin** : lien avec calculabilité (Rabin & Scott, « Finite Automata and Their Decision Problem » (1959)).

- **M.-P. Schützenberger** : lien avec semigroupes & combinatoire ; transducteurs ; séries (multiplicités)... Nombreux étudiants.

- **S. Eilenberg** : lien avec l'algèbre abstraite (variétés au sens de Birkhoff), exposition/synthèse (*Automata, Languages and Machines* (1974)).

Quelques livres

Sélection *très* partielle !

- J. Sakarovitch, *Eléments de théorie des automates* (Vuibert, 2003 ; traduction anglaise : Cambridge, 2009)
- J. Berstel & Ch. Reutenauer, *Noncommutative Rational Series With Applications* (Cambridge, 2011)
- G. Rozenberg, A. Salomaa (eds.), *Handbook of Formal Languages* (1997)
- J. Berstel, *Transductions and Context-Free Languages* (1979)
- A. Salomaa & M. Soittola, *Automata-Theoretic Aspects of Formal Power Series* (1978)
- M. Harrison, *Introduction to Formal Language Theory* (1978)
- S. Eilenberg, *Automata, Languages & Machines* (vol. A 1974, vol. B 1976)
- J. H. Conway, *Regular Algebra and Finite Machines* (1971)
- S. Ginsburg, *The Mathematical Theory of Context-Free Languages* (1966)

Quelques autres références

Sélection *très* partielle !

- J. Sakarovitch, « Automata and Rational Expressions » (`arXiv:1502.03573`)
- J. Sakarovitch, « Rational and Recognizable Power Series », *in*: Droste, Kuich & Vogler, *Handbook of Weighted Automata* (2009)

- M. Bousquet-Mélou, « Rational and Algebraic Series in Combinatorial Enumeration » (`arXiv:0805.0588`)
→ lien avec la combinatoire énumérative

- D. Madore, « THL (Théorie des Langages) », notes de cours (Télécom Paris, INF105) `http://perso.enst.fr/madore/inf105/notes-inf105.pdf`
Vidéos : `http://peertube.r2.enst.fr/video-channels/inf105`
→ proche d'un cours de niveau prépa

Alphabet, lettres, mots, langages

Concepts basiques :

- L'**alphabet** (= jeu de caractères) est un ensemble *fini*, généralement noté Σ ou S, sans structure particulière.

Souvent fixé pour toute la discussion, p.ex., $S = \{a, b, c\}$.
En informatique, on parle de **jeu de caractères**, p.ex., ASCII ou Unicode. Ou $S = \{0, 1\}$ (alphabet **binaire**).

- Les éléments de S s'appellent **lettres** (ou **caractères** ou **symboles**).
- Un **mot** est une suite *finie* de lettres.

L'ensemble des mots possibles se note S^*, et s'appelle **monoïde libre** sur S.

- Un **langage** est un ensemble (quelconque) de mots, i.e., une partie de S^*.

Un langage avec **multiplicités** dans $\mathbb{K}$ (un *semi-anneau*, p.ex., $\mathbb{N}$) est une fonction $S^* \to \mathbb{K}$.

Mots

(Rappel : S est un ensemble *fini*.)

- Un **mot** sur S est une suite finie de lettres (= éléments de S).

En informatique, on parle de **chaîne de caractères**.

- On notera sans séparateur : $x_1 \cdots x_n$ le mot de n lettres (= longueur n) formé par $x_1, \ldots, x_n$.
- **Longueur** d'un mot : $|x_1 \cdots x_n| = n$. Aussi notée $\ell(x_1 \cdots x_n)$.
- Il existe un unique mot de longueur 0 : le **mot vide**, généralement noté ε ou 1 (en informatique, `""`).

Le symbole ε ne fait pas partie de l'alphabet ! C'est un symbole spécial.

- L'ensemble de tous les mots sur S sera noté S^*.

L'ensemble S^* est toujours infini (sauf si $S = \varnothing$). P.ex. $\{a\}^* = \{\varepsilon, a, aa, aaa, aaaa, \ldots\}$.

- Si $x \in S$, on *identifie* x à un mot de longueur 1.
- Si $y \in S$ et $x_1, \ldots, x_n \in S^*$ on note $|x_1 \cdots x_n|_y := \#\{1 \leq i \leq n : x_i = y\}$.

Concaténation de mots

(Rappel : S est un ensemble *fini*, et S^* l'ensemble des **mots** sur S = suites finies.)

- Si $u = x_1 \cdots x_m$ est un mot de longueur m,
et $v = y_1 \cdots y_n$ est un mot de longueur n,
alors $uv := x_1 \cdots x_m y_1 \cdots y_n$ mot de longueur $m + n$.
- On parle de **concaténation** ou simplement **produit** des mots u et v.
- Opération associative ($u(vw) = (uv)w$), le mot vide ε est neutre, et on a $|uv| = |u| + |v|$. Idem $|uv|_y = |u|_y + |v|_y$ (nombre de y dans le mot).
- Puissances d'un mot : $u^n := u \cdots u$ (répété n fois) si $u \in S^*$, $n \in \mathbb{N}$, avec convention $u^0 := \varepsilon$. On a $u^{m+n} = u^m u^n$ et $u^{mn} = (u^m)^n$ et $|u^n| = n|u|$.
- Si $w = uv$, on dit que u est un **préfixe** de w, que v est le **suffixe** correspondant.

« Monoïde libre »

Un **monoïde** est un ensemble M muni d'une opération $\cdot$ (multiplication) et d'un élément 1 vérifiant :

- la multiplication est **associative** ($u(vw) = (uv)w$),
- l'élément 1 est **neutre** ($1 \cdot u = u \cdot 1 = u$).

(Comme un groupe mais sans les inverses.)

Un **morphisme de monoïdes** est une application $\varphi\colon M \to M'$ préservant 1 (i.e., $\varphi(1_M) = 1_{M'}$) et la multiplication (i.e., $\varphi(u \cdot v) = \varphi(u) \cdot \varphi(v)$).

L'ensemble S^* des mots sur S est un monoïde. C'est même le **monoïde libre** sur S, ce qui signifie :

- Pour tout monoïde M et toute application $\varphi\colon S \to M$, il existe un *unique* morphisme de monoïdes $\varphi^*\colon S^* \to M$ tel que $\varphi^*(x) = \varphi(x)$ pour tout $x \in S$.

Appliquer cette propriété universelle à $M = \mathbb{N}$ et $\varphi(x) = 1$ pour $x \in S$ donne le morphisme « longueur ». Appliqué à $1_{\{y\}}$ donne $|\cdot|_y$.

Facteurs, sous-mots, miroir

- **Facteur** = préfixe d'un suffixe = suffixe d'un préfixe = choix de lettres consécutives.

Si $w = x_1 \cdots x_n$, un facteur de w est un $x_i x_{i+1} \cdots x_{j-1}$ où $i \leq j$ (longueur $j - i$), y compris le mot vide.

P.ex. ε, bbc, cab et $abbcab$ sont facteurs de $abbcab$ mais pas abc.

- **Sous-mot** = choix de lettres non nécess$^{\text{t}}$ consécutives (mais dans l'ordre).

Si $w = x_1 \cdots x_n$, un facteur de w est un $x_{i_1} x_{i_2} \cdots x_{i_k}$ où $1 \leq i_1 < \cdots < i_k \leq n$ (longueur k), y compris le mot vide.

P.ex. ε, abc et $abbcab$ sont sous-mots de $abbcab$ mais pas cba. Tout facteur est un sous-mot.

- **Mot miroir** = inverser l'ordre des lettres.

Si $w = x_1 \cdots x_n$, alors $w^{\text{R}} := x_n \cdots x_1$.

Noter $(uv)^{\text{R}} = v^{\text{R}} u^{\text{R}}$.

Palindrome : mot w tel que $w = w^{\text{R}}$.

Notion de langage (sans multiplicité)

- Un **langage** L sur S est un ensemble (quelconque !) de mots sur S.

Autrement dit, $L \subseteq S^*$, ou $L \in \mathscr{P}(S^*)$ (ensemble des parties).

- Un langage peut être fini ou infini.

Deux cas extrêmes : $\varnothing$ (langage vide, ne contient aucun mot), S^* (langage plein, contient tous les mots).

On distinguera bien $\{\varepsilon\}$ (contient un seul mot, le mot vide), parfois noté 1, et $\varnothing$ (ne contient aucun mot), aussi noté 0.

- Autres exemples sur $S = \{a, b\}$: le langage $\{abb, abab, abbbb\}$, celui des mots commençant par a, celui des mots finissant par b, celui des mots ne contenant aucun a, celui des mots dont la longueur est un nombre premier. Aussi S = mots de longueur 1.

- Opérations booléennes : $L_1 \cup L_2$ (réunion), $L_1 \cap L_2$ (intersection), $S^* \setminus L$ (complémentaire, parfois noté $\overline{L}$).
- Remarque : on a $L_1 \cap L_2 = \overline{\overline{L_1} \cup \overline{L_2}}$.
- On peut identifier un langage à une **propriété** des mots, p.ex. « commencer par a ».

Concaténation, puissances, étoile de Kleene

- Concaténation de langages : $L_1 L_2$ est l'ensemble $\{u_1 u_2 : u_1 \in L_1, u_2 \in L_2\}$ des concaténations d'un mot quelconque de L_1 et d'un mot quelconque de L_2.
- Puissances : $L^n := L \cdots L$ (avec n facteurs) est l'ensemble des concaténations de n mots de L. (Convention : $L^0 = \{\varepsilon\}$.)
  - Attention : L^n n'est pas $\{w^n : w \in L\}$ en général.
- **Étoile de Kleene** : $L^* := \bigcup_{n \in \mathbb{N}} L^n$ est l'ensemble des concaténations (d'un nb. fini quelconque, éventuellement zéro) de mots de L.

  On retrouve bien $S^* = \{\text{mots sur } S\}$.
- Variante : $L^+ := \bigcup_{n \geq 1} L^n = LL^* = L^*L$ est l'ensemble des concaténations d'au moins un mot de L.

  Notamment, $S^+ = \{\text{mots non vides}\}$.

Classes de langages

Une **classe de langages** est un ensemble de langages. On ne fera pas leur théorie générale.

Mais on étudiera ici deux grandes classes de langages :

- Les **langages rationnels** définis par les expressions rationnelles. Cette classe coïncidera avec celle des **langages reconnaissables** définis par les automates finis.
- Les **langages algébriques** définis par les grammaires hors contexte. Coïncide avec celle des langages définis par les automates à pile.

Langages avec multiplicités = séries formelles : généralisation des langages à des multi-ensembles (hors programme en prépa, mais apporte du recul).

Multiplicités : semi-anneaux

Un **semi-anneau** est un « anneau sans soustraction » : un ensemble K muni d'opérations binaires $+$ (addition) et $\cdot$ (multiplication) et d'éléments 0 et 1 t.q. :

- $(K, +, 0)$ est un monoïde commutatif, $(K, \cdot, 1)$ est un monoïde,
- $\cdot$ est distributive à g. et à d. sur $+$, et 0 est absorbant pour $\cdot$.

On dit qu'il est « commutatif » lorsque $\cdot$ l'est.

Exemples (commutatifs) :

- les anneaux, p.ex. $\mathbb{Z}$ (anneau des entiers relatifs), $\mathbb{Q}$ (corps des rationnels), $\mathbb{Z}/m\mathbb{Z}$ (anneau des entiers modulo m),
- $\mathbb{N}$ (entiers naturels) ou $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$ (avec $0 \cdot \infty = 0$, le reste évident),
- $\mathbb{B} = \{0, 1\}$ avec $1 + 1 = 1$ (**semi-anneau booléen**),
- $\mathbb{T}_\mathbb{Z} = \mathbb{Z} \cup \{+\infty\}$ muni de $\min$ comme addition et $+$ comme multiplication (semi-anneau **tropical**).

Exemple non-commutatif : matrices $n \times n$ sur un semi-anneau.

Multiplicités : langages ou séries formelles

Si $\mathbb{K}$ est un semi-anneau, un **langage avec multiplicités dans** $\mathbb{K}$ est une fonction $L\colon S^* \to \mathbb{K}$ ici notée $w \mapsto (L!w)$ (un $\mathbb{K}$-**multiensemble** de mots).

- Union avec multiplicités : $w \mapsto (L_1!w) + (L_2!w)$
- Concaténation avec multiplicités : $w \mapsto \sum_{u_1u_2=w}(L_1!u_1)(L_2!u_2)$

Mieux vaut considérer qu'on a affaire à une **série formelle** (sur l'ensemble S d'indéterminées *non-commutatives*), formellement :

$$\sum_{w\in S^*} (L!w)\cdot w$$

On note $\mathbb{K}\langle\langle S\rangle\rangle$ le semi-anneau de ces séries avec opérations : addition $L_1 + L_2\colon w \mapsto (L_1!w) + (L_2!w)$ (= union avec multiplicité) et multiplication $L_1L_2\colon w \mapsto \sum_{u_1u_2=w}(L_1!u_1)(L_2!u_2)$ (= concaténation avec multiplicité).

On identifie $c \in \mathbb{K}$ à la série $\varepsilon \mapsto c,\ \varepsilon \neq w \mapsto 0$, et $u \in S^*$ à $u \mapsto 1,\ u \neq w \mapsto 0$.

Si $\mathbb{K} = \mathbb{C}$ et $S = \{z\}$, on retrouve bien $\mathbb{C}[\![z]\!]$.

Multiplicités : ajout et oubli de multiplicités

- **Fonction indicatrice** : si $L \subseteq S^*$ est un langage sans multiplicité, et $\mathbb{K}$ un semi-anneau, on associe à L la série (= langage avec multiplicité) $\underline{L} := \sum_{w \in L} w$ c'est-à-dire $L \ni w \mapsto 1, L \not\ni w \mapsto 0$.

- **Support** : si $L \in \mathbb{K}\langle\langle S \rangle\rangle$, on lui associe son support $\mathrm{supp}(L) := \{w \in S^* : (L!w) \neq 0\}$.

- **Le cas sans multiplicité** : lorsque $\mathbb{B} := \{0, 1\}$ (semi-anneau booléen où $1 + 1 = 1$), les opérations $L \mapsto \underline{L}$ et $L \mapsto \mathrm{supp}(L)$ permettent d'identifier $\mathscr{P}(S^*)$ (avec union, concaténation, $\varnothing$, $\{\varepsilon\}$) et $\mathbb{B}\langle\langle S \rangle\rangle$ (avec somme, produit, 0, 1).

Multiplicités : le problème de l'étoile de Kleene

Problème : si $L = \sum_{w \in S^*} (L!w) \cdot w$, quel sens donner à $L^* = \sum_{n=0}^{+\infty} L^n$?

- Si le terme constant $c := (L!\varepsilon)$ de L est nul (série dite **propre**), c'est clair : $(L^*!w) = \sum_{u_1 \cdots u_r = w} (L!u_1) \cdots (L!u_r)$, somme sur toutes les factorisations de w en mots de longueur ≥ 1.
- Plus généralement, si on a une « étoile » sur $\mathbb{K}$ telle que c^* ait un sens, poser $(L^*!w) = \sum_{u_1 \cdots u_r = w} c^*(L!u_1)c^* \cdots c^*(L!u_r)c^*$.
- Ou si $\mathbb{K}$ est muni d'une topologie « compatible » (distributivité des produits sur les sommes infinies) faisant que $\sum_{n=0}^{+\infty} L^n$ converge.

On cherchera à n'appliquer l'étoile qu'à des séries propres (sans terme constant, i.e., « ne contenant pas le mot vide »).

Si $\mathbb{K} = \mathbb{B}$ (semi-anneau booléen, avec $1 + 1 = 1$), pas de problème : on pose $1^* = 1$. (Pas non plus sur $\mathbb{N}_\infty$, avec $n^* = \infty$ si $n \geq 1$.)

Multiplicités : étoile et morphismes de monoïdes

- Propriété universelle du monoïde libre : si $\varphi\colon S \to \mathbb{K}$ est une fonction quelconque, $\varphi^*(x_1 \cdots x_n) = \varphi(x_1) \cdots \varphi(x_n)$ définit l'unique morphisme $\varphi^*\colon S^* \to \mathbb{K}$ prolongeant φ.

- Si $f := \sum_{x \in S} \varphi(x)\, x \in \mathbb{K}\langle\langle S \rangle\rangle$ (somme finie !), alors pour $w = x_1 \cdots x_n$ on a $(f^*!w) = (f!x_1) \cdots (f!x_n) = \varphi(x_1) \cdots \varphi(x_n) = \varphi^*(w)$, bref

$$\Big(\sum_{x \in S} \varphi(x)\, x\Big)^* = \sum_{w \in S^*} \varphi^*(w)\, w$$

Notion de langage rationnel

Sans multiplicités, pour commencer.

- Les **langages rationnels** (= **réguliers**) sur S sont ceux qui s'obtiennent à partir des langages $\varnothing$, $\{\varepsilon\}$ et $\{x\}$ (pour $x \in S$) à partir des opérations de
  - réunion (de deux langages),
  - concaténation (de deux langages),
  - étoile de Kleene (d'un langage),

  appliquées un nombre fini de fois.

- Intuitivement, ce sont les langages qui admettent une description à partir des lettres de l'alphabet et des connecteurs « ou bien » (disjonction), « suivi de » (concaténation) et « répétitions illimitées de » (étoile).

- Exemple : $\{d\}(\{c\}^*) = \{d, dc, dcc, dccc, \ldots\}$ est le langage constitué des mots formés d'un d suivi d'une répétition quelconque de la lettre c.

Expressions régulières

- Les **expressions régulières** (= **rationnelles**) sont un moyen pour *(dé)noter* commodément un langage rationnel.
- P.ex. : $dc*$ est une r.e. dénotant le langage rationnel $\{d\}(\{c\}^*) = \{d, dc, dcc, dccc, \ldots\}$. À lire comme « un d suivi d'un nombre quelconque de c ».
- Souvent utilisées en informatique pour spécifier des motifs de recherche. Nombreuses variations de syntaxe, ci-dessous syntaxe inspirée de « `egrep` ».
- Ce sont des mots sur l'alphabet « étendu » $S \cup \{\bot, \underline{\varepsilon}, (,), |, *\}$, les derniers étant appelés **métacaractères** (n'appartiennent pas à S).
- Le $|$ dénote la disjonction (réunion des langages).
- Le $*$ dénote l'étoile de Kleene.
- Les parenthèses corrigent la priorité (étoile prioritaire sur concaténation prioritaire sur disjonction).
- $\underline{\varepsilon}$ dénote le (langage formé du seul) mot vide.
- $\bot$ dénote le langage vide (rarement utile...).

Expressions rationnelles (suite)

Exemples d'expressions rationnelles sur $S = \{a, b, c, d\}$:

- $a|b|cd$ dénote le langage $\{a\} \cup \{b\} \cup \{c\}\{d\} = \{a, b, cd\}$.
- $a(b|cd)$ dénote le langage $\{a\}(\{b\} \cup \{c\}\{d\}) = \{a\}\{b, cd\} = \{ab, acd\}$, et est équivalente à $ab|acd$.
- $a*$ dénote $\{a\}^* = \{\varepsilon, a, aa, aaa, \ldots\}$.
- $aa*$ dénote $\{a\}\{a\}^* = \{a, aa, aaa, aaaa, \ldots\} = \{a\}^+$.
- $(a|b)*$ dénote le langage $(\{a\} \cup \{b\})^* = \{a, b\}^*$ des mots quelconques sur l'alphabet $\{a, b\}$ (= mots ne contenant que des a et des b).
- $(a|bb)*$ dénote le langage $\{a, bb\}^*$ des mots du précédent dont les b viennent par paires.
- $aba(a|b|c|d)*$ ...mots ayant aba pour préfixe.
- $(a|b|c|d)*aba$ ...mots ayant aba pour suffixe.
- $(a|b|c|d)*aba(a|b|c|d)*$ ...mots ayant aba pour facteur.

Expressions rationnelles (suite)

Autres exemples d'expressions rationnelles sur $S = \{a, b, c, d\}$:

- $(a|b|c|d)*a(a|b|c|d)*b(a|b|c|d)*a(a|b|c|d)*$ ...mots ayant aba pour sous-mot.
- $(b|c|d)*(a(b|c|d)*a(b|c|d)*a(b|c|d)*)*$ ...mots w dont le nombre $|w|_a$ de a soit multiple de 3.
- $(ba*)*$ dénote le langage $(\{b\}\{a\}^*)^* = \{b, ba, baa, \ldots\}^*$ des mots formés d'un certain nombre de b chacun suivi d'un certain nombre de a.
- $a*(ba*)*$ est en fait équivalente à $(a|b)*$ (dénotant le langage $\{a, b\}^*$).
- $(a|b)\underline{\varepsilon}(c|d)$ est équivalente à $(a|b)(c|d)$ et dénote $\{ac, bc, ad, bd\}$.
- $(a|b)\perp(c|d)$ dénote le langage $\{a, b\}\varnothing\{c, d\}$ qui est vide.

Expressions rationnelles : définition formelle

Formellement, par induction sur la complexité :

- $\bot$ est une r.e. dénotant $\varnothing$,
- $\underline{\varepsilon}$ est une r.e. dénotant $\{\varepsilon\}$,
- si $x \in S$ alors x est une r.e. dénotant $\{x\}$,
- si r_1, r_2 sont deux r.e. dénotant L_1 et L_2 alors $r_1 r_2$ est une r.e. dénotant $L_1 L_2$,
- ...et $(r_1|r_2)$ est une r.e. dénotant $L_1 \cup L_2$,
- si r est une r.e. dénotant L alors $(r)*$ est une r.e. dénotant L^*.

- Un langage rationnel est un langage dénoté par une r.e. ; si r est une r.e., on note $L(r)$ le langage qu'elle dénote.

(+ règles sur l'omission des parenthèses...)

- On dit que w **vérifie** r quand $w \in L(r)$.
- On dit que r et r' sont **équivalentes** quand $L(r) = L(r')$.

Questions algorithmiques

On veut maintenant se poser les questions suivantes :

- Donnés w (un mot) et r (une r.e.), comment décider algorithm$^{\text{t}}$ si $w \in L(r)$?
- Données r et r' (deux r.e.), comment décider algorithm$^{\text{t}}$ si $L(r) = L(r')$?
- Donnée r, existe-t-il une r.e. r' telle que $L(r') = S^* \setminus L(r)$ (i.e., le langage complémentaire d'un rationnel est-il rationnel) et si oui, comment la trouver algorithm$^{\text{t}}$?
- Données r_1 et r_2, existe-t-il une r.e. r telle que $L(r) = L(r_1) \cap L(r_2)$ (i.e., l'intersection de deux langages rationnels est-il rationnel) et si oui, comment la trouver algorithm$^{\text{t}}$?

La résolution de ces problèmes passera par la notion d'**automate fini**.

Rationnels avec multiplicités

- Les **séries rationnelles** sont la partie de $\mathbb{K}\langle\langle S\rangle\rangle$ engendrée par les éléments de $\mathbb{K}$ et de S par les opérations de :
  - addition
  - multiplication, et
  - étoile de Kleene appliquée à des séries propres (= sans terme constant).

N.B. : 0 correspond ici à $\varnothing$ et 1 à ε.

Le **terme constant** (= multiplicité du mot vide) $c(L) := (L!\varepsilon)$ est calculable *a priori* :

- $c(x) = 0$ si $x \in S$; et $c(k) = k$ si $k \in \mathbb{K}$
- $c(L_1 + L_2) = c(L_1) + c(L_2)$
- $c(L_1 L_2) = c(L_1)\, c(L_2)$
- $c(L^*) = c(L)^*$ (donc 1 si on n'autorise L^* que lorsque $c(L) = 0$)

Si $L = c + L_0$ avec $c = c(L)$, et L_0 propre, on peut poser $L^* = c^*(L_0\, c^*)^*$ si on a donné un sens à c^*.

Étoile de Kleene sur les matrices

L'identité

$$\begin{pmatrix} A & B \\ C & D \end{pmatrix}^* = \begin{pmatrix} (A+BD^*C)^* & A^*B(D+CA^*B)^* \\ D^*C(A+BD^*C)^* & (D+CA^*B)^* \end{pmatrix}$$

vaut pour A, B, C, D des matrices de tailles compatibles sur un semi-anneau topologique où $M^* = \sum_{n=0}^{+\infty} M^n$ s'il y a distributivité du produit sur de telles sommes, p.ex., pour des matrices de séries propres dans $\mathbb{K}\langle\langle S \rangle\rangle$.

On peut s'en servir pour définir le membre de gauche si celui de droite est défini.

Avec $M^* = (1-M)^{-1}$ sur un corps, ceci revient à une formule d'inversion par blocs (cf. « complément de Schur »).

Exemples :

$$\begin{pmatrix} 0 & r_1 & r_{12} \\ 0 & s & r_2 \\ 0 & 0 & 0 \end{pmatrix}^* = \begin{pmatrix} 1 & r_1 s^* & r_{12} + r_1 s^* r_2 \\ 0 & s^* & s^* r_2 \\ 0 & 0 & 1 \end{pmatrix}$$

$$\begin{pmatrix} 0 & e & 0 \\ 0 & 0 & e \\ e & 0 & 0 \end{pmatrix}^* = \begin{pmatrix} (e^3)^* & e(e^3)^* & e^2(e^3)^* \\ e^2(e^3)^* & (e^3)^* & e(e^3)^* \\ e(e^3)^* & e^2(e^3)^* & (e^3)^* \end{pmatrix}$$

Lemme d'Arden

L'équation linéaire

$$X = MX + P \text{ resp. } Y = YM + Q$$

dans $\mathbb{K}\langle\langle S \rangle\rangle$, où M est une matrice à coefficients propres (= terme constant nul), et P, Q, X, Y vecteurs de taille compatible, a pour unique solution

$$X = M^*P \text{ resp. } Y = QM^*$$

Résolution algorithmique en « éliminant » les variables, en utilisant le cas 1×1. P.ex., pour résoudre

$$\begin{cases} y = ya + y'c + q \\ y' = yb + y'd + q' \end{cases}$$

on a $y' = y'd + (yb + q')$ donc $y' = (yb + q')d^*$ donc $y = y(a + bd^*c) + q + q'd^*c$ donc $y = (q + q'd^*c)(a + bd^*c)^*$.

Quelques identités rationnelles

On préfère ici noter $+$ que $|$, et 0 que $\bot$, et 1 que ε.

On a les identités suivantes entre séries/langages rationnel(le)s, donc équivalences entre expressions régulières, *même avec multiplicités* :

- Identités exprimant le fait qu'on a affaire à un semi-anneau :
$e+0 \equiv 0+e \equiv e$ et $e_1+(e_2+e_3) \equiv (e_1+e_2)+e_3$ et $e_1+e_2 \equiv e_2+e_1$ et $e0 \equiv 0e \equiv 0$ et $e_1(e_2e_3) \equiv (e_1e_2)e_3$ et $e1 \equiv 1e \equiv e$ et $e_1(e_2+e_3) \equiv e_1e_2+e_1e_3$ et $(e_1+e_2)e_3 \equiv e_1e_3+e_2e_3$.
- Identités « apériodiques » concernant l'étoile : $(e_1e_2)^* \equiv 1+e_1(e_2e_1)^*e_2$ et $(e_1+e_2)^* \equiv e_1^*(e_2e_1^*)^* \equiv (e_1^*e_2)^*e_1^*$.
- Identités dues à Conway : si e_g sont indicées par les éléments g d'un *groupe fini* G, et si M est la matrice $M_{h,g} := e_{h^{-1}g}$, alors

$$\sum_g (M^*)_{1,g} \equiv \Big(\sum_g e_g\Big)^*$$

pour G cyclique d'ordre m ceci équivaut à $e^* \equiv (1+e+e^2+\cdots+e^{m-1})(e^m)^*$.
- Identités spécifiques à $\mathbb{B}$ (i.e. *sans* multiplicités) : $e+e \equiv e$ et $(e^*)^* \equiv e^*$.

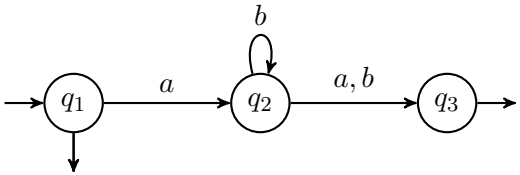
Krob (1990) : Sur $\mathbb{B}$, les identités ci-dessus « suffisent ».

Automates finis : thème

- **Automate fini** non-déterministe (« NFA ») (sans multiplicités) sur S :
  - Q ensemble fini d'« **états** »,
  - I, F deux parties de Q (états **initiaux** et **finaux**),
  - $\delta \subseteq Q \times S \times Q$ **relation de transition**.

Langage reconnu $L(A)$:= ensemble des mots étiquetant *un* chemin d'*un* état initial à *un* état final.

Exemple :



Ici $I = \{q_1\}$ et $F = \{q_1, q_3\}$ et $\delta = \{(q_1, a, q_2), (q_2, b, q_2), (q_2, a, q_3), (q_2, b, q_3)\}$.
On a $L(A) = \{\varepsilon, aa, ab, aba, abb \ldots\} = L(\underline{\varepsilon}|ab*(a|b))$.

Automates finis : variations

- Automate fini **déterministe** (« DFA ») : $I = \{q_0\}$ est un singleton, et pour chaque état $q \in Q$ et chaque lettre $x \in S$ il existe un *unique* q' t.q. $(q, x, q') \in \delta$ (i.e., δ est une fonction $Q \times S \to Q$; selon les auteurs : « au plus un q' », « δ est une fonction partielle »).

- Transitions **spontanées** (= ε-transitions) : on autorise (q, ε, q') dans δ (uniquement dans un NFA $\rightsquigarrow$ « ε-NFA »).
À ignorer dans la lecture d'un chemin (empruntées spontanément).

- Transitions étiquetées par des expressions régulières.
(Parcourables en consommant un mot du langage dénoté.)

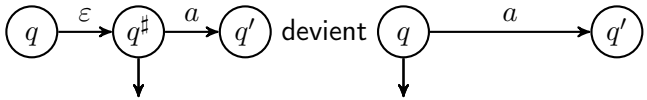
- NFA **avec multiplicités** dans $\mathbb{K}$: défini par $I, F \colon Q \to \mathbb{K}$ et $\delta \colon Q \times S \times Q \to \mathbb{K}$. (Se mélange mal avec les transitions spontanées !)
Multiplicité de $x_1 \cdots x_n$ = sommer les $I(q_0)\delta(q_0, x_1, q_1) \cdots \delta(q_{n-1}, x_n, q_n)F(q_n)$.

- Avec multiplicités *et* transitions étiquetées par des expressions régulières...

Élimination des transitions spontanées

Sans multiplicités.

- La ε-**fermeture** $C(q)$ d'un état q est l'ensemble des états accessibles depuis q (y compris q) par une succession de ε-transitions (fermeture réflexive-transitive).
- Élimination des ε-transitions : si $q^\sharp \in C(q)$, pour chaque $(q^\sharp, x, q') \in \delta$, introduire (q, x, q') dans δ, et si $q^\sharp \in F$, introduire q dans F.
Puis oublier les ε-transitions (et les états non-initiaux accessibles seul[t] par elles).



Permet de ramener un ε-NFA à un NFA équivalent.

(Adaptable avec multiplicités si pas de boucle de ε-transitions.)

Automates finis : représentation matricielle

- NFA avec multiplicités dans $\mathbb{K}$:
  - Q ensemble fini,
  - I vecteur-ligne et F vecteur-colonne, indicés par Q (à valeurs dans $\mathbb{K}$),
  - $\delta(x)$ matrice carrée, indicée par $Q \times Q$ (dans $\mathbb{K}$), pour chaque $x \in S$.

De $\delta\colon S \to \mathbb{M}_{Q\times Q}(\mathbb{K})$ on déduit $\delta^*\colon S^* \to \mathbb{M}_{Q\times Q}(\mathbb{K})$ donné par (la propriété universelle du monoïde libre) : $\delta^*(x_1 \cdots x_n) = \delta(x_1) \cdots \delta(x_n)$.

- Série définie : $(L!w) := I\,\delta^*(w)\,F$, c'est-à-dire $L = \sum_{w\in S^*} (I\,\delta^*(w)\,F)\,w$.

Or $\sum_{w\in S^*} \delta^*(w)\,w = \big(\sum_{x\in S} \delta(x)\,x\big)^*$. Donc :

- On pose $M = \sum_{x\in S} \delta(x)\,x$ (matrice carrée indicée par $Q \times Q$, à coefficients de la forme ax où ici $a = \delta(q, x, q')$).
- On a alors $L = IM^*F$.

Automates finis : représentation matricielle, de nouveau

On se donne

- Q ensemble fini,
- I vecteur-ligne et F vecteur-colonne, indicés par Q,
- M matrice carrée, indicée par $Q \times Q$,

à coefficients dans $\mathbb{K}\langle\langle S \rangle\rangle$: de la forme ax avec $a \in \mathbb{K}, x \in S$ pour un NFA avec multiplicités, ou même définis par une expression régulière telle que $c(r) = 0$.

La série (langage avec multiplicité) définie est alors

$$IM^*F = \sum_{n=0}^{+\infty} \sum_{q_0,\dots,q_n \in Q} I_{q_0} M_{q_0,q_1} \cdots M_{q_{n-1},q_n} F_{q_n}$$

(c'est IM^nF pour les chemins de longueur n).

Théorème (Kleene-Schützenberger) : cette série est rationnelle.

Démonstration : formule donnant M^*.

Reconnaissable implique rationnel : algorithmique

On vient de voir que **reconnaissable** (par NFA) **implique rationnel** : tout NFA reconnaît le langage dénoté par une expression régulière (même avec multiplicités).

Approches algorithmiques possibles (il s'agit de calculer $L = IM^*F$) :

- calcul de M^* par la formule par blocs,
- calcul de M^* par l'algorithme de McNaughton-Yamada : (après numérotation des états de 1 à N)

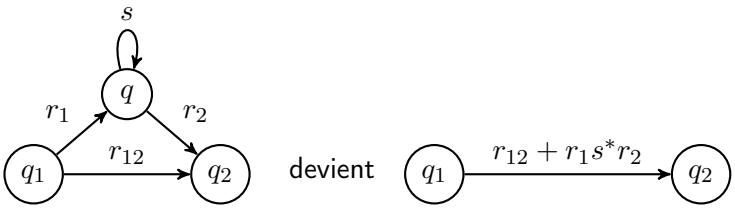
$$M_{p,q}^{(0)} = M_{p,q} \quad \text{et} \quad M_{p,q}^{(k+1)} = M_{p,q}^{(k)} + M_{p,k}^{(k)} \, (M_{k,k}^{(k)})^* \, M_{k,q}^{(k)}$$
$$(M^*)_{p,q} = M_{p,q}^{(N)} \quad \text{si } p \neq q, \quad \text{et} \quad (M^*)_{p,p} = 1 + M_{p,p}^{(N)}$$

- résolution de $Y = YM + I$ (cf. lemme d'Arden) et calcul de YF, ou symétriquement $X = MX + F$ et calcul de IX,
- élimination des états (Brzozowski-McCluskey, page suivante).

Élimination des états

- S'arranger pour avoir un unique état initial q_i sans transition y aboutissant, et un unique état final q_f sans transition qui en part.

- Pour tous les états $q \notin \{q_i, q_f\}$, pour tout couple d'état (q_1, q_2) distincts de q (mais *y compris* $q_1 = q_2$) tels qu'il existe une transition $q_1 \to q$ et une $q \to q_2$, remplacer *simultanément*



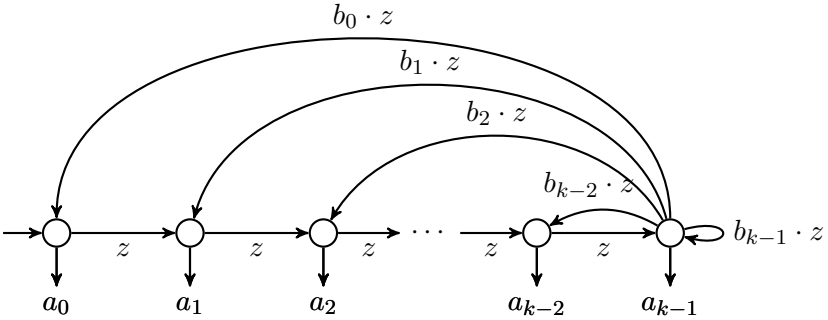
(ignorer le terme r_{12} si la transition $q_1 \to q_2$ n'existe pas déjà, ignorer le facteur s^* si la transition $q \to q$ n'existe pas), et *effacer* l'état q.

Reconnaissable implique rationnel : conséquences

- Sans multiplicités : un langage reconnu par un NFA peut être dénoté par une expression régulière.
- Avec multiplicités : la série définie par un NFA avec multiplicités est rationnelle. Notamment :
- Sa série génératrice (remplacer toutes les lettres par une même lettre z, i.e. $w \rightsquigarrow z^{|w|}$) est rationnelle. (Idem pour une série à plusieurs variables commutatives.)
$\rightarrow$ voir plus loin.
- Dans un graphe orienté où on a choisi deux ensembles I, F de sommets, si a_n est le nombre de chemins orientés de longueur n d'un sommet de I vers un sommet de F, alors $\sum_{n=0}^{+\infty} a_n z^n$ est une fonction rationnelle.

Reconnaissable implique rationnel : conséquences (suite)

- Une suite linéairement récurrente $a_{n+k} = b_0\, a_n + b_1\, a_{n+1} + \cdots + b_{k-1}\, a_{n+k-1}$ définit une série $\sum_{n=0}^{+\infty} a_n z^n$ rationnelle.

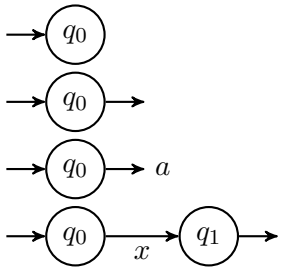


Automate de Glushkov

On veut associer à toute expression régulière r (éventuellement avec multiplicités) un **automate standard**, qui :

- reconnaît le langage / la série dénoté(e) par r,
- a un unique état initial (multiplicité 1), sans transition y aboutissant,
- a autant d'états non-initiaux qu'il y a de lettres dans r, chaque transition y aboutissant étant étiquetée par la lettre en question.

Cas de base :



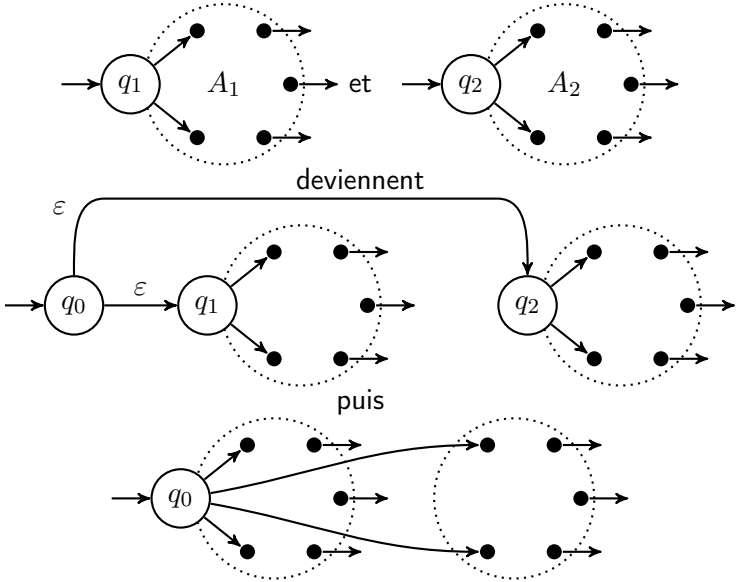
pour 0 (i.e., $\bot$),

pour 1 (i.e., $\underline{\varepsilon}$),

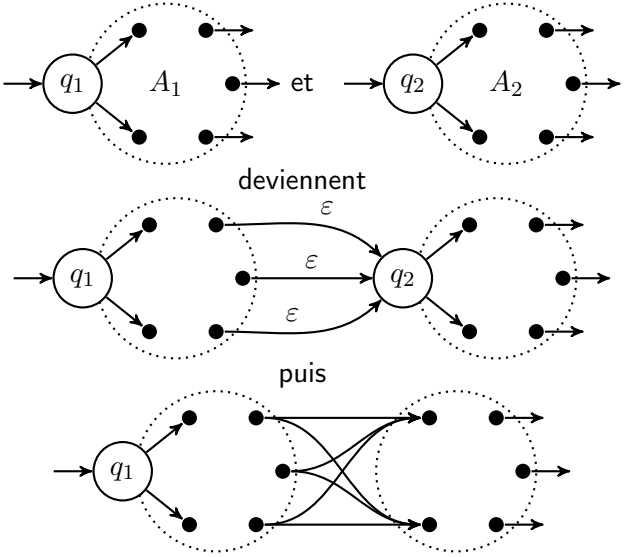
pour $a \in \mathbb{K}$ en général,

pour $x \in S$.

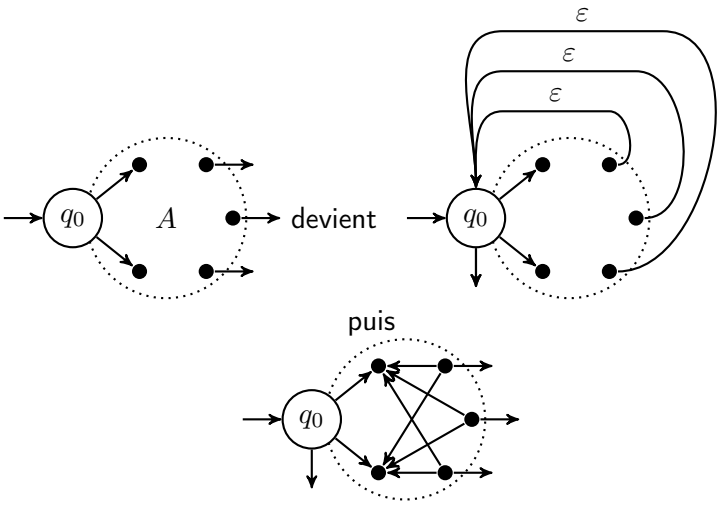
Automate de Glushkov : somme (disjonction)



Automate de Glushkov : produit (concaténation)



Automate de Glushkov : étoile de Kleene



Automate de Glushkov : description matricielle

Multiplicités possibles ici.

- Un automate standard ayant $n+1$ états est défini matriciellement par

$$I = \begin{pmatrix} 1 & 0 \end{pmatrix} \ , \ M = \begin{pmatrix} 0 & J \\ 0 & N \end{pmatrix} \ , \ F = \begin{pmatrix} c \\ G \end{pmatrix}$$

(décomposition en blocs de taille 1 et n ; langage $L = IM^*F = c + JN^*G$).

	J	N	c	G
$r_1 + r_2$	$\begin{pmatrix} J_1 & J_2 \end{pmatrix}$	$\begin{pmatrix} N_1 & 0 \\ 0 & N_2 \end{pmatrix}$	$c_1 + c_2$	$\begin{pmatrix} G_1 \\ G_2 \end{pmatrix}$
$r_1 r_2$	$\begin{pmatrix} J_1 & c_1 J_2 \end{pmatrix}$	$\begin{pmatrix} N_1 & G_1 J_2 \\ 0 & N_2 \end{pmatrix}$	$c_1 c_2$	$\begin{pmatrix} G_1 c_2 \\ G_2 \end{pmatrix}$
r^*	$c^* J$	$N + G c^* J$	c^*	$G c^*$

$(c = 0 \ \Rightarrow \ c^* = 1)$

(Si pas de multiplicités et lettres distinctes, J donne les premières lettres possibles, N les paires, G les dernières, et c si le mot vide est accepté.
$\rightarrow$ Algorithme de Berry-Sethi.)

Automate miroir

- Sans multiplicités : si $L \subseteq S^*$ est rationnel alors $L^{\mathsf{R}} := \{w^{\mathsf{R}} : w \in L\}$ est rationnel.

Preuve : Inverser toutes les flèches (y compris échanger états initiaux et finaux) dans un NFA reconnaissant L.

- Avec multiplicités : si $L \in \mathbb{K}\langle\langle S \rangle\rangle$ est rationnel et si $\mathbb{K}$ est commutatif alors $L^{\mathsf{R}} = \sum_{w \in S^*} (L!w)\, w^{\mathsf{R}}$ (c'est-à-dire $(L^{\mathsf{R}}!w) = (L!w^{\mathsf{R}})$) est rationnel.

Preuve : Idem : si $L = \sum_{w \in S^*} (I\, \delta^*(w)\, F)\, w$ alors $L = \sum_{w \in S^*} (F^{\mathsf{R}}\, \delta^*(w)^{\mathsf{R}}\, I^{\mathsf{R}})\, w$ où M^{R} est la transposée de M, et $\delta^*(x_1 \cdots x_n)^{\mathsf{R}} = \delta(x_n)^{\mathsf{R}} \cdots \delta(x_1)^{\mathsf{R}}$, donc L^{R} est défini par $(F^{\mathsf{R}}, I^{\mathsf{R}}, \delta^{\mathsf{R}})$.

Produit d'automates

Supposons $\mathbb{K}$ *commutatif* ici.

Donnés deux automates $(Q_1, I_1, F_1, \delta_1)$ et $(Q_2, I_2, F_2, \delta_2)$, on définit leur **produit** par :

- $Q = Q_1 \times Q_2$ (produit des ensembles d'états),
- $I(q_1, q_2) = I_1(q_2)\, I_2(q_2)$, si on veut $I = I_1 \otimes I_2$,
- $F(q_1, q_2) = F_1(q_2)\, F_2(q_2)$, si on veut $F = F_1 \otimes F_2$,
- $\delta((q_1, q_2), x, (q_1', q_2')) = \delta_1(q_1, x, q_1')\, \delta_2(q_2, x, q_2')$, si on veut $\delta(x) = \delta_1(x) \otimes \delta_2(x)$, pour chaque $x \in S$.

« Faire travailler les deux automates en parallèle, en multipliant les multiplicités. »

Ceci définit $L_1 \odot L_2$ avec $(L_1 \odot L_2\,!\,w) = (L_1!w)\,(L_2!w)$ (**produit de Hadamard** des séries), qui est donc rationnelle si L_1, L_2 le sont.

Notamment, sans multiplicités, $L_1 \cap L_2$ est rationnel si L_1, L_2 le sont.

Déterminisation

Si $\mathbb{K}$ est *fini*, p.ex. $\mathbb{K} = \mathbb{B}$, l'ensemble des vecteurs $\mathbb{K}^Q$ est aussi fini.

On peut donc transformer un NFA (Q, I, F, δ) en DFA à un coût exponentiel :

- $Q' = \mathbb{K}^Q$ (vus comme des vecteurs-lignes),
- $\mathbf{q}'_0 = I$ (état initial),
- $\delta'(\mathbf{q}, x) = \mathbf{q}\ \delta(x)$ (action à droite sur les vecteurs-lignes),
- $F'(\mathbf{q}) = \mathbf{q}\ F$ (produit scalaire vecteur-ligne par vecteur-colonne).

Les seules multiplicités restantes sont dans F' (multiplicité des états finaux).

Algorithmiquement, ne créer que les états utiles (accessibles depuis l'état initial).

Sur $\mathbb{B}$ (= sans multiplicités), les DFA permettent de calculer le *complémentaire* (échanger états finaux et non-finaux), donc aussi l'*intersection*.

Désambiguïsation

- Un NFA *sans multiplicités* reconnaissant un langage L est dit **inambigu** quand chaque mot de L est accepté par un *unique* chemin.

I.e., considéré comme automate sur $\mathbb{N}$ au lieu de $\mathbb{B}$, il définit la série $\underline{L} := \sum_{w \in L} w \in \mathbb{N}\langle\langle S \rangle\rangle$ (multiplicité 1 pour chaque mot de L).

- *Un DFA est inambigu*, donc tout langage rationnel est reconnu par un automate inambigu.

- On peut en déduire une expression régulière inambiguë :
  - les disjonctions sont inambiguës, i.e., réunions disjointes,
  - les concaténations sont inambiguës, i.e. $L_1 \times L_2 \to L_1 L_2,\ (u_1, u_2) \mapsto u_1 u_2$ est bijective,
  - les étoiles de Kleene sont inambiguës, i.e., $\biguplus_{n \in \mathbb{N}} L^{\times n} \to \bigcup_{n \in \mathbb{N}} L^n,\ (u_1, \ldots, u_n) \mapsto u_1 \cdots u_n$ est bijective.

Exemple : $(a|b)*a(a|b)*$ est ambiguë, $b*a(a|b)*$ ne l'est pas.

Automate canonique : définition

Sans multiplicités ($\mathbb{K} = \mathbb{B}$) ici.

Si L est un langage, on définit, pour $w \in S^*$:

$$w^{-1}L := \{t \in S^* : wt \in L\}$$
$$w_1 \equiv_L w_2 \iff w_1^{-1}L = w_2^{-1}L$$
$$\iff \forall t \in S^*\big(w_1 t \in L \Leftrightarrow w_2 t \in L\big)$$

Automate déterministe **canonique** de L :

- $Q = \{w^{-1}L : w \in S^*\}$ (ou, si on préfère, $S^*/(\equiv_L)$),
- état initial $q_0 = L$ (ou $[\varepsilon]$),
- états finaux $F = \{w^{-1}L : w \in L\} = \{q \in Q : \varepsilon \in q\}$ (ou $L/(\equiv_L)$),
- transitions : $\delta(w^{-1}L, x) = x^{-1}w^{-1}L = (wx)^{-1}L$ (ou $[w] \mapsto [wx]$).

Myhill-Nerode (1958) : Cet automate reconnaît L, il est fini (i.e., Q est fini) ssi L est reconnaissable et, quand c'est le cas, il est sous-quotient de tout DFA reconnaissant L. Il est donc l'unique tel DFA ayant le plus petit nombre d'états.

Automate canonique : algorithmique

À partir d'un DFA sans état inaccessible reconnaissant L, on peut construire l'automate minimal (=canonique) de L en le quotientant par :

$$q_1 \equiv q_2 \ :\Longleftrightarrow\ \forall t \in S^*\big(\delta^*(q_1,t) \in F \Leftrightarrow \delta^*(q_2,t) \in F\big)$$

On calcule $\equiv$ par approximations successives (relations $\equiv_i$ de plus en plus fines, jusqu'à ce que $(\equiv_{i+1}) = (\equiv_i)$) :

$$q_1 \equiv_0 q_2 \iff (q_1 \in F \Leftrightarrow q_2 \in F)$$

$$q_1 \equiv_{i+1} q_2 \iff (q_1 \equiv_i q_2 \text{ et } \forall x \in S\,(\delta(q_1,x) \equiv_i \delta(q_2,x)))$$

soit en fait

$$q_1 \equiv_i q_2 \iff \forall t\,(|t| \leq i \Rightarrow (\delta^*(q_1,t) \in F \Leftrightarrow \delta^*(q_2,t) \in F))$$

(algorithme de Moore).

Automate canonique : conséquences

- Grâce à l'automate canonique, on sait décider si deux expressions rationnelles/automates sont équivalents.

Algorithme : (construire l'automate de Glushkov si expression régulière,) déterminiser puis minimiser, et comparer les deux automates.

Le test d'isomorphisme des automates est facile parce qu'ils sont convenablement étiquetés.

- Conséquences : on sait décider la vacuité, l'universalité, l'inclusion.

Ceci ne vaut pas avec des multiplicités. Sur $\mathbb{T}_{\mathbb{Z}} = (\mathbb{Z} \cup \{+\infty\}, \min, +)$ ou $\mathbb{T}_{\mathbb{N}} = (\mathbb{N} \cup \{+\infty\}, \min, +)$, l'égalité de deux expressions rationnelles est *indécidable* en général. (Krob, 1992)

(Le cas de $\mathbb{B}$ se généralise cependant à un semi-anneau *fini*.)

Le cas d'un corps : représentation canonique

On suppose ici que $\mathbb{K}$ est un corps ou corps-gauche (= algèbre à divisions).

Si $L \in \mathbb{K}\langle\langle S \rangle\rangle$, on définit, pour $w \in S^*$:

$$w^{-1}L := \sum_{t \in S^*} (L!wt)\, t \quad \text{c'est-à-dire} \quad (w^{-1}L\,!\,t) = (L\,!\,wt)$$

$$V = \mathrm{LVect}(\{w^{-1}L : w \in S^*\}) = \sum_{w \in S^*} \mathbb{K} w^{-1}L \subseteq \mathbb{K}\langle\langle S \rangle\rangle$$

Alors V est de $\mathbb{K}$-dimension (gauche) finie ssi L est $\mathbb{K}$-reconnaissable.

$$\delta(x) \colon V \to V, \;\; w^{-1}L \mapsto x^{-1}w^{-1}L = (wx)^{-1}L$$
$$F \colon V \to \mathbb{K}, \;\; w^{-1}L \mapsto (w^{-1}L\,!\,\varepsilon) = (L\,!\,w)$$

définissent un automate « canonique » reconnaissant L (voir $I = L$ comme un vecteur-ligne, $\delta(x)$ comme une matrice, et F comme un vecteur-colonne).

Comme dans le cas de Myhill-Nerode il est le NFA reconnaissant L ayant la plus petite dimension, unique cette fois à conjugaison linéaire près.

Le cas d'un corps : algorithmique

Partant de (I, F, δ) (où I vecteur-ligne, F vecteur-colonne et $\delta\colon S \to \mathbb{M}_{Q\times Q}(\mathbb{K})$) définissant L, on peut calculer « la » représentation minimale :

- Calculer les $I\,\delta(w)$ en parcourant S^* en largeur, en collectant tous ceux qui sont linéair[t] (à gauche) indépendants des précédents, jusqu'à ne plus rien ajouter.

$\widetilde{\leftrightarrow}$ calcul d'un DFA sans état inaccessible.

- Calculer les $\delta(w)\,F$ en parcourant S^* en largeur, en collectant tous ceux qui sont linéair[t] (à droite) indépendants des précédents, idem.

$\widetilde{\leftrightarrow}$ algorithme de Moore (recherche du quotient minimal séparant les transitions).

Pour le **test d'égalité**, le plus simple est de calculer la *différence* des séries, et de tester l'égalité à 0 (par dimension minimale).

Rationalité : ajout et oubli de multiplicités

- **Fonction indicatrice** : si $L \subseteq S^*$ est un langage rationnel, et $\mathbb{K}$ un semi-anneau, la série associée $\underline{L} := \sum_{w \in L} w$ est encore rationnelle.

Preuve : construire un automate inambigu (p.ex. un DFA) reconnaissant L.

Conséquence : la série génératrice d'un langage rationnel L, c'est-à-dire $\sum_{n=0}^{+\infty} \#\{w \in L : |w| = n\}\, z^n$, est rationnelle. (Preuve : $w \rightsquigarrow z^{|w|}$ dans $\underline{L}$.)

- **Support** : si $L \in \mathbb{K}\langle\langle S \rangle\rangle$ est rationnelle, son support $\mathrm{supp}(L) := \{w \in S^* : (L!w) \neq 0\}$ n'est *pas forcément rationnel*.

La série $\sum_{w \in \{a,b\}^*} (|w|_a - |w|_b)\, w \in \mathbb{Z}\langle\langle a, b \rangle\rangle$ est rationnelle, mais son support $\{w \in \{a,b\}^* : |w|_a \neq |w|_b\}$ ne l'est pas.

Mais *c'est le cas* si $\mathbb{K}$ est un semi-anneau « positif » c'est-à-dire que $\mathbb{K} \to \mathbb{B}, 0 \mapsto 0, 0 \neq c \mapsto 1$ est un morphisme. Notamment pour $\mathbb{N}$.

Comparer : Skolem-Mahler-Lech : si $f = \sum_{n=0}^{+\infty} a_n z^n \in \mathbb{C}[\![z]\!]$ est rationnelle, alors $\mathrm{supp}(f)$ est rationnel (réunion finie de suites arithmétiques et de singletons). Idem sur corps de caract. 0.

Différence de séries positives

Soit $\mathbb{K}$ un sous-anneau de $\mathbb{R}$ et $\mathbb{K}_+ := \mathbb{K} \cap [0; +\infty[$. Tout élément $a \in \mathbb{K}$ s'écrit sous la forme $a_+ - a_-$ où $a_+, a_- \in \mathbb{K}_+$ (prendre $a_\pm := \max(\pm a, 0)$).

Lemme : $(a_+ - a_-) + (b_+ - b_-) = (a_+ + b_+) - (a_- + b_-)$ et
$(a_+ - a_-)(b_+ - b_-) = (a_+ b_+ + a_- b_-) - (a_- b_+ + a_+ b_-)$.

- Proposition : $L \in \mathbb{K}\langle\langle S \rangle\rangle$ est $\mathbb{K}$-rationnelle ssi il existe $L_+, L_- \in \mathbb{K}_+\langle\langle S \rangle\rangle$ et $\mathbb{K}_+$-rationnelles telles que $L = L_+ - L_-$.

Preuve : Le « si » est clair. Si (I, F, δ) définit L, on peut écrire $I = I_+ - I_-$ et $F = F_+ - F_-$ et $\delta = \delta_+ - \delta_-$. On pose alors

$$I'_+ = \begin{pmatrix} I_+ & 0 \end{pmatrix}\ ,\ I'_- = \begin{pmatrix} 0 & I_- \end{pmatrix}\ ,\ \delta' = \begin{pmatrix} \delta_+ & \delta_- \\ \delta_- & \delta_+ \end{pmatrix}\ ,\ F' = \begin{pmatrix} F_+ \\ F_- \end{pmatrix}$$

D'après le lemme, $(I'_+ - I'_-, F', \delta')$ définit L, donc $L = L_+ - L_-$ où $L_\pm$ est définie par $(I'_\pm, F', \delta')$.

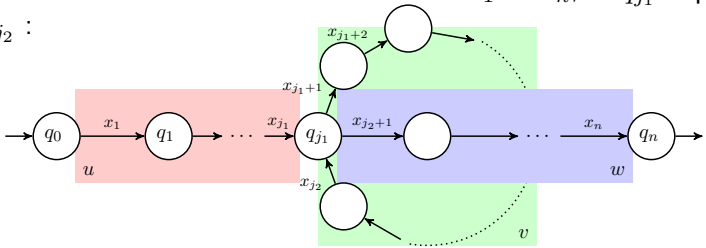
Attention : $L \in \mathbb{K}_+\langle\langle S \rangle\rangle$ et $\mathbb{K}$-rationnelle *n'entraîne pas* $\mathbb{K}_+$-rationnelle.
Contre-exemple : $\sum_{w \in \{a,b\}^*} (|w|_a - |w|_b)^2\, w \in \mathbb{Z}\langle\langle a, b \rangle\rangle$.

Lemme de pompage pour les langages rationnels

« Lemme d'itération », « lemme de l'étoile », « lemme de pompage »...

- Si $L \subseteq S^*$ est rationnel, il existe $k \in \mathbb{N}$ tel que tout mot $t \in L$ de longueur $|t| \geq k$ se factorise en $t = uvw$ tels que :
  - $|uv| \leq k$
  - $|v| \geq 1$
  - $\forall i \in \mathbb{N}\ (uv^i w \in L)$

Preuve : partir d'un automate reconnaissant L, soit k son nombre d'états, $q_0, \ldots, q_n$ les états traversés en consommant $t = x_1 \cdots x_n$, et q_{j_1} le premier état répété $= q_{j_2}$:



Un exemple de langage non rationnel

« Un automate ne peut pas compter. »

Affirmation : $L := \{a^n b^n\}$ n'est pas rationnel.

Preuve : par l'absurde, s'il l'est, le lemme de pompage donne un k comme décrit. On considère $t = a^k b^k$.
La factorisation $t = uvw$ doit vérifier $u = a^\ell$ et $v = a^m$ car $|uv| \leq k$, donc $w = a^{k-\ell-m} b^k$, et $m \neq 0$ car $|v| \geq 1$. Alors $uv^i w = a^{k+(i-1)m} b^k$ est dans L, ce qui est faux pour $i \neq 1$.

Grammaires hors-contexte

- Une **grammaire hors-contexte** G (ou « non contextuelle », CFG) sur S est la donnée de :
  - N fini disjoint de S (ensemble des **nonterminaux**),
  - $X_0 \in N$ (**axiome** ou **symbole initial** ; on peut noter $\to X_0$),
  - un ensemble fini de couples (X, α) où $X \in N$ et $\alpha \in (S \cup N)^*$ (**règles** ou **productions** ; on note $X \to \alpha$, et on abrège $X \to \alpha_1, \dots, X \to \alpha_n$ en $X \to \alpha_1 | \cdots | \alpha_n$).

Pseudo-mot = mot sur $S \cup N$.

- On définit $\gamma X \gamma' \Rightarrow \gamma \alpha \gamma'$ (**dérivation immédiate**) lorsque $X \to \alpha$ est une règle : X est le symbole réécrit, (γ, γ') le contexte. Et $\Rightarrow^*$ la clôture réflexive-transitive de $\Rightarrow$ (**dérivation**).

- **Langage engendré** : $L(G) = \{w \in S^* : X_0 \Rightarrow^* w\}$. Plus généralement, $L(G, X) = \{w \in S^* : X \Rightarrow^* w\}$

Arbres d'analyse

- Un **arbre d'analyse** (ou **de dérivation**), resp. **arbre d'analyse incomplet** pour la grammaire G est un arbre (fini, ordonné, enraciné) dont les nœuds sont étiquetés par des éléments de $S \cup N \cup \{\varepsilon\}$, tel que :
  - la racine est étiquetée X_0 (axiome),
  - si un nœud n'est pas une feuille et est étiqueté X, alors $X \in N$ et ses fils sont étiquetés $Z_1, \ldots, Z_n$ pour $Z_1, \ldots, Z_n \in S \cup N$ et $n \geq 1$, ou bien ε, et il existe une règle $X \to Z_1 \cdots Z_n$ ou $X \to \varepsilon$ selon le cas,
  - pour un arbre complet : toute feuille est étiquetée par un élément de $S \cup \varepsilon$.

Les ε servent uniquement à marquer la fin des dérivations $X \to \varepsilon$ et peuvent être omis dans un arbre complet.

- Le **mot analysé** d'un arbre d'analyse (complet) s'obtient par lecture en profondeur de ses feuilles.

- Bijection naturelle entre arbres d'analyse, dérivations gauches (= on réécrit le nonterminal le plus à gauche) et dérivations droites d'un même mot.

Langage de Dyck

Sur l'alphabet $S := \{a_1, b_1, \ldots, a_m, b_m\}$ représentant m types de parenthèses, on définit le **langage de Dyck** ou des **expressions bien-parenthésées** par la grammaire :

$$X_0 \to \varepsilon \mid a_1 X_0 b_1 X_0 \mid \cdots \mid a_m X_0 b_m X_0$$

Exemple : $a_1a_2b_2b_1a_1b_1a_2b_2$ mais pas $a_1a_2b_1b_2$.

Pour $S = \{a, b\}$ (soit $m = 1$ type de parenthèses), c'est aussi :

$$\{w \in S^* : |w|_b = |w|_a \text{ et } |u|_b \leq |u|_a \text{ pour tout préfixe } u \text{ de } w\}$$

Le nombre de tels mots de longueur $2n$ (toujours pour $m = 1$ type de parenthèses) s'appelle n-ième **nombre de Catalan** $C_n = \frac{1}{n+1}\binom{2n}{n}$. On a $\sum_{n=0}^{+\infty} C_n z^n = \frac{1-\sqrt{1-4z}}{2z}$.

Multiplicités

- Une CFG avec **multiplicités** (dans $\mathbb{K}$ semi-anneau) est donnée par N, X_0 comme avant, et des règles de la forme $X \to c\alpha$ où $c \in \mathbb{K}$ (multiplicité de la règle). On abrège $X \to c_1\alpha_1, \ldots, X \to c_n\alpha_n$ en $X \to c_1\alpha_1 + \cdots + c_n\alpha_n$.
- La multiplicité $v(\mathscr{T})$ d'un arbre d'analyse $\mathscr{T}$ vaut $c\, v(\mathscr{T}_1) \cdots v(\mathscr{T}_n)$ si la règle à la racine est $X \to c\, Z_1 \cdots Z_n$ (multiplicité c) et $\mathscr{T}_1, \ldots, \mathscr{T}_n$ les sous-arbres de racines les fils étiquetés $Z_1, \ldots, Z_n$ de la racine de $\mathscr{T}$ (si Z_i est terminal, $v(\mathscr{T}_i) = 1$).
- Mieux vaut prendre $\mathbb{K}$ commutatif : $\rightsquigarrow$ produit des toutes les multiplicités.
- Si chaque mot n'a qu'un *nombre fini* d'arbres d'analyse ou si la somme a un sens pour d'autres raisons (e.g. $\mathbb{N}_\infty$), la multiplicité $(L!w)$ d'un mot dans la série L définie par G sera :

$$\sum_{\mathscr{T} \text{ analysant } w} v(\mathscr{T})$$

Les tracas du mot vide

- Si G est une CFG, on appelle **évanescents** (« nullable ») le plus petit ensemble de nonterminaux tels que si $X_1, \ldots, X_n$ sont évanescents et qu'il existe une règle $X \to X_1 \cdots X_n$ (y compris si $X \to \varepsilon$) alors X est évanescent.

  Ce sont les nonterminaux tels que $X \Rightarrow^* \varepsilon$. Calculables effectivement.

- On dira que G est **sans boucle essentielle** quand on peut ordonner les nonterminaux $N = \{X_1, \ldots, X_k\}$ de façon qu'il n'y ait aucune règle $X_i \to \eta X_j \eta'$ avec η, η' produits d'évanescents et $j \leq i$.

  Cette condition (ou d'autres...) assure que chaque mot $w \in S^*$ n'a qu'un *nombre fini* d'arbres d'analyse, qu'on peut calculer effectivement (par récurrence sur $|w|$ et, pour $|w|$ donné, récurrence descendante sur le i du X_i à la racine).

  (Pour analyser w selon une règle $X_i \to Z_1 \cdots Z_n$, considérer toutes les partitions de w en produit de n facteurs, non-vides si le Z_ℓ correspondant n'est pas évanescent, et appliquer la même analyse à chaque facteur.)

Séries et langages algébriques

- Si G est une grammaire avec multiplicités sans boucle essentielle *ou* sur $\mathbb{B}$ ou $\mathbb{N}_\infty$, on a donné un sens à la série $L(G)$ définie (et $L(G, X)$ pour X nonterminal). Pour G sans multiplicités :
  - sur $\mathbb{B}$, c'est la fonction indicatrice du langage engendré,
  - sur $\mathbb{N}_\infty$, elle compte le nombre d'arbres d'analyse de chaque mot.
- Les séries et langages ainsi obtenus sont dits **algébriques**.
- On vient de voir qu'on peut décider algorithmiquement si $w \in L$, resp. calculer $(L!w)$ pour L algébrique et $w \in S^*$ (complexité épouvantable ici), en énumérant les arbres d'analyse.
- Les dérivations $X \to c_1\alpha_1 + \cdots + c_n\alpha_n$ de X définissent un système d'équations algébriques sur les séries (noter $L(G) = L(G, X_0)$) :

$$L(G, X) = c_1 L(G, \alpha_1) + \cdots + c_n L(G, \alpha_n)$$

où $L(G, \alpha) := L(G, Z_1) \cdots L(G, Z_r)$ si $\alpha = Z_1 \cdots Z_r$ (et $L(G, s) := s$ si $s \in S$).

Grammaires régulières

Soit (Q, I, F, δ) un NFA éventuellement avec multplicités. On sait qu'on peut supposer un seul état initial q_0 (i.e., I vaut 1 en q_0 et 0 ailleurs).

On définit une grammaire par :

- nonterminaux = états du NFA ($N = Q$),
- axiome q_0,
- des règles $q \to cxq'$ lorsque $\delta(q, x, q') = c$, et $q \to c\varepsilon$ lorsque $F(q) = c$.

Un arbre d'analyse dans cette grammaire est essentiellement un chemin dans le NFA, avec la même multiplicité. Notamment, ils engendrent le même langage y compris avec multiplicités. Donc :

- Tout langage/série rationnel est algébrique.

Formes normales de Chomsky et de Greibach

- Une CFG est dite en **forme normale de Chomsky** lorsque toutes ses règles sont de la forme $X \to YZ$ avec Y, Z (exactement) deux nonterminaux autres que l'axiome, ou $X \to t$ avec t un terminal, ou éventuellement $X_0 \to \varepsilon$ avec X_0 l'axiome. (Possible avec multiplicités.)

On peut assurer cette forme après diverses réécritures (p.ex. : si X est évanescent, recopier toute règle faisant intervenir X sans celui-ci pour supprimer les règles $X \to \varepsilon$; court-circuiter les règles $X \to Y$ en reproduisant sur X les règles partant de Y).

- **Forme normale de Greibach** : toutes les règles sont de la forme $X \to tY_1 \cdots Y_k$ où $k \geq 0$ et Y_i sont des nonterminaux, plus éventuellement $X_0 \to \varepsilon$.

(Identifier et extraire tous les premiers terminaux produisibles par X.) Ceci permet un début d'analyse.

Ambiguïté

- Une CFG sans multiplicités est dite **inambiguë** (resp. **finiment ambiguë**) quand chaque mot a *au plus un* arbre d'analyse (resp. un *nombre fini*).

Autrement dit, la série qu'elle définit sur $\mathbb{N}_\infty$ (qui compte le nombre d'arbres d'analyse) est à valeurs dans $\{0,1\}$ (resp. $\mathbb{N}$).

Si on veut, $L(G)$ est inambiguë ssi $L(G) = \underline{L(G)}$ (fonction indicatrice) sur $\mathbb{N}_\infty$.

- $L \subseteq S^*$ langage algébrique est dit **intrinsèquement inambigu** lorsqu'il existe une CFG inambiguë qui l'engendre, i.e., $\underline{L}$ est algébrique sur $\mathbb{N}$.

L'ambiguïté intrinsèque existe : $\{a^m b^m c^n : m,n \in \mathbb{N}\} \cup \{a^m b^n c^n : m,n \in \mathbb{N}\}$ est un exemple.

En revanche, tout langage algébrique est engendré par une grammaire sans boucle essentielle, donc finiment ambiguë.

Automates à pile

Un automate à pile (non-déterministe) sur S est la donnée de :

- Q ensemble fini d'états, $q_0 \in Q$ état initial, $F \subseteq Q$ états finaux,
- Γ ensemble fini « alphabet de pile »,
- $\Delta \subseteq Q \times \Gamma^* \times (S \cup \{\varepsilon\}) \times Q \times \Gamma^*$ ensemble fini de transitions.

Comprendre $(q, \alpha, x, q', \alpha') \in \Delta$ comme « dans l'état q avec α au sommet de la pile, on peut consommer x, dépiler α, passer dans l'état q' et empiler α' ». Le mot est accepté quand une exécution le consomme et finit dans un état final avec une pile vide.
(Énormément de variations possibles.)

- Contrairement aux automates finis, le déterminisme a son importance ici : les automates à pile déterministes reconnaissent un sous-ensemble strict des automates à pile non-déterministes (les langages $LR(*)$).

Grammaires hors-contexte et automates à pile

- Les grammaires hors-contexte et les automates à pile non déterministes définissent la même classe de langages (algébriques).

Deux approches pour définir un automate pour une grammaire G :

- Approche « descendante » : le langage de pile de l'automate est formé des terminaux et nonterminaux de la grammaire : il commence par empiler l'axiome de la grammaire, il peut dépiler un nonterminal X le remplacer par la droite d'une règle $X \to \alpha$; il peut aussi dépiler un terminal t et consommer la lettre t.
- Approche « ascendante » : le langage de pile de l'automate est formé des terminaux et nonterminaux de la grammaire : il peut consommer un terminal en l'empilant, il peut dépiler le miroir de la partie droite d'une règle $X \to \alpha$ et la remplacer par X ; il peut aussi dépiler l'axime et terminer.

Propriétés de clôture

- Sans multiplicités : les langages algébriques sont stables par union, concaténation, et étoile de Kleene.

  De plus, l'intersection d'un langage algébrique et d'un langage rationnel est algébrique.

- Avec multiplicités : les séries algébriques sont stables par somme, multiplication par les constantes, produit, et étoile de Kleene des séries propres.

  De plus, sur un semi-anneau commutatif, le produit de Hadamard (= produit terme à terme des coefficients) d'une série algébrique et d'une série rationnelle est algébrique.

Propriétés de clôture : esquisse de preuve

- Somme : données G_1, G_2 d'axiomes X_1, X_2 définissant L_1, L_2, on fabrique G d'axiome X_0 avec $X_0 \to X_1 + X_2$ (c'est-à-dire $X_0 \to X_1$ et $X_0 \to X_2$) et les règles de G_1, G_2.
- Produit : $X_0 \to X_1 X_2$.
- Étoile : $X_0 \to 1 + X_1 X_0$ (c'est-à-dire $X_0 \to \varepsilon$ et $X_0 \to X_0 X_1$).
- Produit de Hadamard : soit $\delta \colon S \to \mathbb{M}_{n\times n}(\mathbb{K})$ la matrice de transitions d'un NFA (I, F, δ), et soit G une CFG. Pour chaque nonterminal X_t de G on introduit une matrice $n \times n$ de nonterminaux $(X_{t,i,j})_{i,j}$. Dans chaque règle de G, remplacer chaque nonterminal X_t par sa matrice, et chaque terminal t par la matrice $\delta(t)\, t$. Pour axiome on prend (une production vers) $\sum_{i,j} I_j F_i X_{0,i,j}$ où X_0 est l'axiome de G.

Chomsky-Schützenberger

On a vu que l'intersection d'un langage algébrique et d'un rationnel est algébrique. Ceci est une sorte de réciproque :

- Chomsky-Schützenberger : si L est un langage algébrique sur S, il existe R un langage rationnel sur l'alphabet $\Delta := \{a_1, b_1, \ldots, a_m, b_m\}$, et $h\colon \Delta \to S^*$, tels que si $D \subseteq \Delta^*$ est le langage de Dyck sur m types de parenthèses, on ait

$$L = h^*(D \cap R)$$

où $h^*\colon \Delta^* \to S^*$ est l'extension de h en un morphisme de monoïdes (substitution de $h(t)$ pour t).

(Une variante avec multiplicités est aussi énonçable.)

Algébricité : ajout et oubli de multiplicités

- **Fonction indicatrice** : si $L \subseteq S^*$ est un langage algébrique, la série associée $\underline{L} := \sum_{w \in L} w$ n'est *pas forcément algébrique*, même sur $\mathbb{N}$. Elle l'est ssi L est intrinsèquement inambigu.

Conséquence : la série génératrice d'un langage algébrique inambigu L, c'est-à-dire $\sum_{n=0}^{+\infty} \#\{w \in L : |w| = n\}\, z^n$, est algébrique.

- **Support** : si $L \in \mathbb{K}\langle\langle S \rangle\rangle$ est algébrique, son support $\mathrm{supp}(L) := \{w \in S^* : (L!w) \neq 0\}$ n'est *pas forcément algébrique*.

Mais *c'est le cas* si $\mathbb{K}$ est un semi-anneau « positif » c'est-à-dire que $\mathbb{K} \to \mathbb{B}, 0 \mapsto 0, 0 \neq c \mapsto 1$ est un morphisme. Notamment pour $\mathbb{N}$.

- Si $\mathbb{K}$ un sous-anneau de $\mathbb{R}$ et $\mathbb{K}_+ := \mathbb{K} \cap [0; +\infty[$, alors $L \in \mathbb{K}\langle\langle S \rangle\rangle$ est $\mathbb{K}$-algébriques ssi il existe $L_+, L_- \in \mathbb{K}_+\langle\langle S \rangle\rangle$ et $\mathbb{K}_+$-algébriques telles que $L = L_+ - L_-$.

Lemme de pompage pour les langages algébrique

Lemme de Bar-Hillel :

- Si $L \subseteq S^*$ est algébrique, il existe $k \in \mathbb{N}$ tel que tout mot $t \in L$ de longueur $|t| \geq k$ se factorise en $t = uvwxy$ tels que :
  - $|vwx| \leq k$
  - $|vx| \geq 1$
  - $\forall i \in \mathbb{N}\ (uv^i wx^i y \in L)$

Esquisse de preuve : on suppose G en forme normale de Chomsky, $k = 2^r$ où r est le nombre de nonterminaux : alors l'arbre d'analyse de tout mot de longueur $\geq k$ fait apparaître un même nonterminal X comme descendant de lui-même : soit w est ce qui résulte du X le plus bas, et vwx d'un X ancêtre.

Exemple d'application : $\{a^n b^n c^n : n \in \mathbb{N}\}$ n'est pas algébrique.

L'intersection des langages algébriques $\{a^m b^m c^n : m, n \in \mathbb{N}\}$ et $\{a^m b^n c^n : m, n \in \mathbb{N}\}$ n'est pas algébrique.

Questions décidables et indécidables

Les questions suivantes sont indécidables sur les CFG :

- décider si le langage engendré par G est plein (i.e. tout S^*),
- décider si G_1 et G_2 engendrent le même langage,
- décider si le langage engendré par G_1 est inclus dans celui engendré par G_2,
- décider si le langage engendré par G est régulier,
- décider si G est ambiguë.

Les questions suivantes sont décidables sur les CFG :

- décider si le langage engendré par G est vide,
- décider si le langage engendré par G est infini,
- décider si G est ambiguë sachant que $L(G) = L(G_0)$ avec G_0 inambiguë,
- décider si $L(G_1) = L(G_2)$ sachant que $L(G_1) \subseteq L(G_2)$ et G_1, G_2 inambiguës (notamment, décider si $L(G) = S^*$ sachant G inambiguë).