Informatique – MPI

Romain Bricout

9 octobre 2023

Introduction

Ce document réunit l'ensemble de mes cours d'Informatique de MPI, ainsi que les exercices les accompagnant. Le professeur était M. Carcenac. J'ai adapté certaines formulations me paraissant floues ou ne me plaisant pas mais le contenu pur des cours est strictement équivalent.

Les éléments des tables des matières initiale et présentes au début de chaque chapitre sont cliquables (amenant directement à la partie cliquée). C'est également le cas des références à des éléments antérieurs de la forme, par exemple, « Démonstration 5.22 ».

Table des matières

I Cours **2**

1 Langages réguliers **3**

Introduction 3

1.1 Langage formel 3

1.1.1 Alphabet, mot 3

1.1.2 Langage 6

1.2 Langage régulier 8

2 Automates finis **11**

3 Théorème de Kleene **12**

II Exercices **13**

1 Langages réguliers **14**

2 Automates finis **15**

3 Théorème de Kleene **16**

Première partie

Cours

Chapitre 1

Langages réguliers

Sommaire

Introduction **3**

1.1 Langage formel **3**

1.1.1 Alphabet, mot 3

1.1.2 Langage 6

1.2 Langage régulier **8**

Introduction

Cf. diaporama.

Dans ce cours, on nommera « langages réguliers » et « langages rationnels » les mêmes objets.

1.1 Langage formel

1.1.1 Alphabet, mot

Définition 1.1
Un alphabet Σ est un ensemble fini de symboles.

Un mot (sur Σ) est une suite finie de symboles (de Σ).

Exemple 1.2
$m = m_1 m_2 \dots m_n$ est un mot de n symboles.

On note $m = \varepsilon$ le mot vide.

Notation 1.3
L'ensemble de tous les mots (sur Σ) est noté Σ^* (étoile de Kleene).

Notation 1.4
On note $|m|$ la longueur du mot m.

Proposition 1.5
On a $|m_1 m_2 \dots m_n| = n$ *et* $|\varepsilon| = 0$.

Notation 1.6
On note $|m|_x$ le nombre d'occurrences de $x \in \Sigma$ dans m.

Définition 1.7 (Concaténation)
Soient $u, v \in \Sigma^*$.

On note $u.v$ ou uv la concaténation de u et v.

On pose

$$\varepsilon\varepsilon = \varepsilon \qquad \varepsilon u = u\varepsilon = u.$$

Proposition 1.8
Soient $u, v \in \Sigma^*$ *et* $x \in \Sigma$.

On a

$$|uv| = |u| + |v| \qquad |uv|_x = |u|_x + |v|_x \,.$$

Remarque 1.9 (HP, lien avec les structures algébriques)
La concaténation est une loi de composition interne sur Σ^*.

Elle est non-commutative : on n'a pas $\forall u, v \in \Sigma^*,\ \ uv = vu$.

Elle est associative : on a $\forall u, v, w \in \Sigma^*,\ \ u\,(vw) = (uv)\,w$.

Elle admet ε comme élément neutre.

On obtient alors que $(\Sigma^*, .)$ est un monoïde.

Définition 1.10 (Puissance)
Soient $u \in \Sigma^*$ et $n \in \mathbb{N}^*$.

On définit $u^n = \underbrace{uu\dots u}_{n \text{ facteurs}}$ et on pose $u^0 = \varepsilon$.

Remarque 1.11
Pour $x, y, z \in \Sigma^*$, on a $(xyz)^2 = xyzxyz \neq x^2y^2z^2$.

Définition 1.12
Soit $m \in \Sigma^*$.

- $u \in \Sigma^*$ est un préfixe de m ssi il existe $v \in \Sigma^*$ tel que $m = uv$.
  $u \in \Sigma^*$ est un préfixe propre de m ssi il existe $v \in \Sigma^* \setminus \{\varepsilon\}$ tel que $m = uv$.

- $u \in \Sigma^*$ est un suffixe de m ssi il existe $v \in \Sigma^*$ tel que $m = vu$.
  $u \in \Sigma^*$ est un suffixe propre de m ssi il existe $v \in \Sigma^* \setminus \{\varepsilon\}$ tel que $m = vu$.

- $u \in \Sigma^*$ est un facteur de m ssi il existe $v, w \in \Sigma^*$ tel que $m = vuw$.
  $u \in \Sigma^*$ est un facteur propre de m ssi il existe $v, w \in \Sigma^* \setminus \{\varepsilon\}$ tel que $m = vuw$.

- $u \in \Sigma^*$ est un sous-mot de m si c'est une suite extraite de m (*cf.* ?THM? **??** pour une définition formelle).

Exemple 1.13
Cf. exercices.

Remarque 1.14
ε est un préfixe, un suffixe, un facteur et un sous-mot de tout mot $m \in \Sigma^*$.

Remarque 1.15
Soient $u, m \in \Sigma^*$.

On a les implications suivantes :

$$u \text{ est un préfixe de } m \implies u \text{ est un facteur de } m$$

$$u \text{ est un suffixe de } m \implies u \text{ est un facteur de } m$$

$$u \text{ est un facteur de } m \implies u \text{ est un sous-mot de } m$$

Notation 1.16
Si $u \in \Sigma^*$ est un préfixe de $m \in \Sigma^*$, on note $u \leqslant_p m$.

1.1.2 Langage

Définition 1.17
Un langage L (sur Σ) est un ensemble de mots, *i.e.* une partie de Σ^*.

Exemple 1.18
On pose $\Sigma = \{a, b\}$. On peut alors définir les langages sur Σ suivants :

- $L_1 = \{ab, aab, aaab\}$
- $L_2 = \{a^n \mid n \in \mathbb{N}^*\}$
- $L_3 = \varnothing$
- $L_4 = \{\varepsilon\}$
- $$\begin{aligned} L_5 &= \{\text{mots qui contiennent un nombre pair de } a\} \\ &= \{m \in \Sigma^* \mid |m|_a \equiv 0\ [2]\} \\ &\neq \{(aa)^n \mid n \in \mathbb{N}\} \end{aligned}$$
- $L_6 = \{m \in \Sigma^* \mid |m|_a = |m|_b\}$

Si on considère maintenant l'ensemble des symboles ASCII comme alphabet, alors toute chaîne de caractères ASCII est un mot. Par exemple, tout programme informatique est un mot.

Définition 1.19 (Opérations sur les langages)
Soient Σ un alphabet et $L_1, L_2 \in \mathscr{P}(\Sigma^*)$.

On définit les langages ci-dessous :

- union : $L_1 \cup L_2 = \{m \in \Sigma^* \mid m \in L_1 \text{ ou } m \in L_2\}$;
- intersection : $L_1 \cap L_2 = \{m \in \Sigma^* \mid m \in L_1 \text{ et } m \in L_2\}$;
- différence : $L_1 \setminus L_2 = \{m \in \Sigma^* \mid m \in L_1 \text{ et } m \notin L_2\}$;
- différence symétrique : $L_1 \,\Delta\, L_2 = \{m \in \Sigma^* \mid m \in L_1 \text{ et } m \notin L_2 \text{ ou } m \notin L_1 \text{ et } m \in L_2\}$;
- concaténation : $L_1 L_2 = \{uv \mid (u, v) \in L_1 \times L_2\}$.

Exemple 1.20
On pose $\Sigma = \{a, b\}$, $L_1 = \{a, ab, \varepsilon\}$ et $L_2 = \{b, a, \varepsilon\}$.

On a $L_1 L_2 = \{ab, aa, abb, aba, b, a, \varepsilon\}$.

De plus, on a $L_1^2 = \{aa, aab, a, aba, abab, ab, \varepsilon\} \neq \{aa, abab, \varepsilon\}$.

Définition 1.21
Soit $L \in \mathscr{P}(\Sigma^*)$.

Pour $n \in \mathbb{N}^*$, on définit $L^n = \{u_1 u_2 \dots u_n \mid \forall i \in [\![1 \; ; n]\!], \;\; u_i \in L\}$ et on pose $L^0 = \{\varepsilon\} \neq \varnothing$.

On définit $L^* = \bigcup\limits_{n \in \mathbb{N}} L^n$.

On a les propriétés suivantes :

- $L^0 \subseteq L^*$ donc $\varepsilon \in L^*$;
- Pour tout $n \in \mathbb{N}, \;\; L^n \subseteq L^*$;
- $m \in L^* \iff \left| \begin{array}{l} \exists n \in \mathbb{N}^*, \;\; \exists (u_1, \dots, u_n) \in L^n, \;\; m = u_1 \dots u_n \\ \text{ou} \\ m = \varepsilon \end{array} \right.$

On définit $L^+ = \bigcup\limits_{n \in \mathbb{N}^*} L^n$.

On a $L^+ \subseteq L^*$.

Remarque 1.22
La définition de Σ^* est cohérente en voyant Σ comme un langage.

Proposition 1.23
Soient Σ un alphabet et $L, L_1, L_2 \in \mathscr{P}(\Sigma^*)$.

On a :

- $L.\varnothing = \varnothing.L = \varnothing$;
- $L.\{\varepsilon\} = \{\varepsilon\}.L = L$;
- $L^n.L^m = L^m.L^n = L^{n+m}$;
- $L.(L_1 \cup L_2) = L.L_1 \cup L.L_2$;

- ▹ $L.(L_1 \cap L_2) \subseteq L.L_1 \cap L.L_2$;
- ▹ $L^*.L = L.L^* = L^+$;
- ▹ $L^* = L^+ \cup \{\varepsilon\}$;
- ▹ $(L^*)^* = L^*$

Démonstration 1.24
Cf. ?THM? **??**. ∎

1.2 Langage régulier

Soit Σ un alphabet.

Définition 1.25
On définit par induction l'ensemble des langages réguliers LRat et l'ensemble des expressions régulières (représentation symbolique des langages) R.

Cas de base :
$$\varnothing \in \mathrm{LRat} \qquad \{\varepsilon\} \in \mathrm{LRat} \qquad \{a \mid a \in \Sigma\} \in \mathrm{LRat}\,.$$

Cas induits : pour $(L_1, L_2) \in \mathrm{LRat}^2$, on a les propriétés de stabilité suivantes :
$$L_1 \cup L_2 \in \mathrm{LRat} \qquad L_1 L_2 \in \mathrm{LRat} \qquad L_1^* \in \mathrm{LRat}\,.$$

De même, on a les cas de bases :
$$\varnothing \in \mathrm{R} \qquad \varepsilon \in \mathrm{R} \qquad a \in \mathrm{R} \text{ avec } a \in \Sigma$$

et les cas induits : pour $(r_1, r_2) \in \mathrm{R}^2$, on a les propriétés de stabilité suivantes :
$$r_1 \mid r_2 \in \mathrm{R} \qquad r_1 r_2 \in \mathrm{R} \qquad r_1^* \in \mathrm{R}\,.$$

Remarque 1.26 (HP)
LRat et R sont isomorphes.

Définition 1.27
On dit qu'une expression régulière dénote un langage régulier et on note $\mathscr{L}(r)$ le langage dénoté par $r \in \mathrm{R}$.

Exemple 1.28
On pose $\Sigma = \{a, b\}$ et $r = a\,(a \mid b)^* \in \mathrm{R}$.

Alors $\mathscr{L}\,(r) = \{a\}\,.\,(\{a\} \cup \{b\})^*$ est le langage contenant les mots qui commencent par un a.

Proposition 1.29
Tout langage contenant un unique mot est régulier.

Exemple 1.30
On a $\{aba\} = \{a\}\,.\,\{b\}\,.\,\{a\} \in \mathrm{LRat}$.

Proposition 1.31
Tout langage fini est régulier.

Exemple 1.32
On a $\{aa, abc, c\} = \{aa\} \cup \{abc\} \cup \{c\} \in \mathrm{LRat}$.

Proposition 1.33
Le langage Σ^* est régulier.

Démonstration 1.34
L'alphabet Σ un langage fini donc c'est un langage régulier donc Σ^* est régulier. ■

Définition 1.35
On dit qu'une expression régulière $r \in \mathrm{R}$ s'apparie à (ou « match ») tout mot de $\mathscr{L}\,(r)$.

Quelques cas particuliers :

- a s'apparie au mot a ;
- $r_1 r_2$ s'apparie à un mot de $\mathscr{L}\,(r_1)$ concaténé à un mot de $\mathscr{L}\,(r_2)$;
- $r_1 \mid r_2$ s'apparie à un mot de $\mathscr{L}\,(r_1)$ ou à un mot de $\mathscr{L}\,(r_2)$;
- r^* s'apparie à une succession de mots de $\mathscr{L}\,(r)$ (incluant la succession de zéro mot ε) ;

- ε s'apparie au mot vide ε ;
- $\varnothing$ ne s'apparie à aucun mot ;
- Σ s'apparie à n'importe quel symbole ;
- Σ^* s'apparie à n'importe quel mot.

Remarque 1.36 (Priorités)
S'appliquent d'abord les étoiles, puis les concaténations et enfin les unions. On peut parenthéser les expressions si nécessaire.

Exemple 1.37
Avec $\Sigma = \{a, b\}$:

- on pose $r_1 = (a \mid b)^*$.
  On a $\mathscr{L}(r_1) = \{\text{mots composés de } a \text{ ou de } b\}$;
  $$= \{a, b\}^*$$
- on pose $r_2 = aa^*$.
  On a $\mathscr{L}(r_2) = \{a^n \mid n \in \mathbb{N}^*\}$;
- on pose $r_3 = (aa)^*$.
  On a $\mathscr{L}(r_3) = \left\{a^{2k} \mid k \in \mathbb{N}\right\}$;
- on pose $r_4 = a\Sigma^*$.
  On a $\mathscr{L}(r_4) = \{\text{mots commençant par } a\}$.

Avec $\Sigma = \{a, \ldots, z, \_\}$ (où _ désigne une espace) :

- l'expression régulière $ch\,(at \mid ien)$ s'apparie aux mots *chat* et *chien* ;
- l'expression régulière $(tres\_)^*\,bien$ s'apparie aux mots *bien*, *tres_bien*, *tres_tres_bien*, ...

Chapitre 2

Automates finis

⋆⋆ À venir ⋆⋆

Chapitre 3

Théorème de Kleene

$\star\star$ À venir $\star\star$

Deuxième partie

Exercices

Chapitre 1

Langages réguliers

⋆⋆ À venir ⋆⋆

Chapitre 2

Automates finis

⋆⋆ À venir ⋆⋆

Chapitre 3

Théorème de Kleene

⋆⋆ À venir ⋆⋆