

Langages et Automates

Introduction et quelques mots sur les langages

Engel Lefauchaux

Prépas des INP

Organisation du cours

- 8 CM de 1h30
- 5 TD en demi-groupe
- 1 DM (30% de la note finale)
- 1 Examen (70% de la note finale)
- L'ensemble des slides se trouve sur ma page web
<https://elefauch.github.io/>

Objectifs du cours

- Méthodologie et approche scientifique :
 - La modélisation mathématique de problèmes informatiques.
 - L'analyse des modèles mathématiques.
- Connaissances spécifiques :
 - Plusieurs formalismes de modélisation (expression régulière, automates, . . .)
 - Les capacités de ces modèles, ainsi que leurs limitations.

Objectif du jour

- Définir ce qu'est un langage
- Apprendre à créer et manipuler un langage
 - Opérations sur les mots et les langages
- Langage régulier
 - Expression régulière
 - Critères de régularité

Plan

- 1 Qu'est-ce qu'un langage ?
- 2 Construction et opérations sur les langages
- 3 Expressions régulières
- 4 Critères de régularité

Outline

- 1 Qu'est-ce qu'un langage ?
- 2 Construction et opérations sur les langages
- 3 Expressions régulières
- 4 Critères de régularité

Qu'est-ce qu'un langage ?

- Un nombre incroyable de langage
 - Français, anglais, chinois, Russe,...
 - Braille, SMS, Morse,...
 - Pascal, Ocaml, C++, Python,...
- Origine du langage
 - Parlé \approx -50000 ?
 - Écrit \approx -6000 (écriture cunéiforme)
 - Informatique : 1951, A0
- Un langage est *structuré*:
 - Symboles (lettre, hiéroglyphe, chiffre,...)
 - Mots
 - Ordonnancement (phrase, structure "if, then, else", ...)

L'importance des règles

- Sans règle, tout est un langage
- L'ensemble des nombres premiers.
- Les programmes Python qui compilent correctement.
- L'ensemble des théorèmes mathématiquement vrai.

Résoudre un problème, c'est identifier un langage.

Entrée : un système \mathcal{A}

Question : \mathcal{A} satisfait-il la propriété P ?

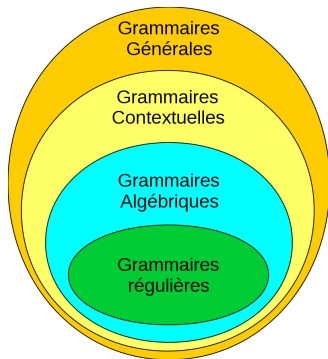
Entrée : un mot \mathcal{A}

Question : est-ce que \mathcal{A} appartient au langage L_P ?

→ Certains langages ne sont pas étudiables !

Hiérarchie des langages (Chomsky, 1956)

Chomsky propose des limitations pour différentes grammaires, définissant ainsi des classes de langages formels



Les grammaires Générales ne sont pas toute puissantes.

→ Machine de Turing

Langage et machine

- L'**automate** (ordinateur) « comprend » un langage que l'ingénieur lui soumet :
 - Comme une **commande** (« ordre »)
 - Il « **reconnaît** » ou « **accepte** » un langage
 - Il ne « reconnaît pas » ou « n'accepte pas » les autres langages
- Ceci a besoin d'être formalisé pour :
 - Créer de **nouveaux langages**, sous certaines contraintes :
 - **Efficacité** : rapidité des traitements
 - **Expressivité** : capacité à formuler des « choses »
 - **Précision** : degré de contrôle sur ce que fait la machine
 - Savoir à l'avance **si un message / code sera reconnu** ou non par l'automate (sinon : erreurs)

Exemple d'un langage

Comment modéliser les exécutions d'un système simple ?

- Attribuer un symbole ou mot à chaque action du système
- N'autoriser que les ordres correspondant au système.

Comment représenter l'achat d'une baguette ou d'un croissant dans une boulangerie ?

- *entrer, commander_pain, commander_croissant, payer, sortir*

Exemple d'un langage

Comment modéliser les exécutions d'un système simple ?

- Attribuer un symbole ou mot à chaque action du système
- N'autoriser que les ordres correspondant au système.

Comment représenter l'achat d'une baguette ou d'un croissant dans une boulangerie ?

- *entrer, commander_pain, commander_croissant, payer, sortir*
- *entrer commander_pain payer sortir*

Exemple d'un langage

Comment modéliser les exécutions d'un système simple ?

- Attribuer un symbole ou mot à chaque action du système
- N'autoriser que les ordres correspondant au système.

Comment représenter l'achat d'une baguette ou d'un croissant dans une boulangerie ?

- *entrer, commander_pain, commander_croissant, payer, sortir*
- *entrer commander_pain payer sortir*
ou
entrer commander_croissant payer sortir

Exemple d'un langage (2)

Nous voulons modéliser une machine à café possédant les propriétés suivantes :

- Si à l'arrêt, on peut cliquer sur un bouton pour l'activer.
- En cours de fonctionnement, la machine fait du bruit pendant une durée aléatoire.
- Si la machine a été activé, elle va éventuellement produire du café et s'éteindre.

Exemple d'un langage (2)

Nous voulons modéliser une machine à café possédant les propriétés suivantes :

- Si à l'arrêt, on peut cliquer sur un bouton pour l'activer.
- En cours de fonctionnement, la machine fait du bruit pendant une durée aléatoire.
- Si la machine a été activé, elle va éventuellement produire du café et s'éteindre.

Quels symboles / mots pour ce modèle ?

Exemple d'un langage (2)

Nous voulons modéliser une machine à café possédant les propriétés suivantes :

- Si à l'arrêt, on peut cliquer sur un bouton pour l'activer.
- En cours de fonctionnement, la machine fait du bruit pendant une durée aléatoire.
- Si la machine a été activé, elle va éventuellement produire du café et s'éteindre.

Quels symboles / mots pour ce modèle ?

Quelles phrases représentent un bon fonctionnement du modèle ?

Outline

- 1 Qu'est-ce qu'un langage ?
- 2 Construction et opérations sur les langages
- 3 Expressions régulières
- 4 Critères de régularité

Comment construit-on un langage ?

- Description extentionnelle : $\{\text{mot}_1, \text{mot}_2, \text{mot}_3\}$
 - Description intentionnelle : "tous les mots qui..."
 - Description définitoire : $\{xyz \mid z = yx\}$
 - Par opération sur des langages déjà définis
- une structure d'anneau pour les langages

L'alphabet, une brique de base

Alphabet Σ : l'ensemble fini des éléments minimaux du langage

- Lettre : $\Sigma = \{a, b, \dots, z\}$
- Playstation : $\Sigma = \{haut, bas, gauche, droite, \square, \dots\}$
- $\Sigma = \{abc, a, tub\}$
- $\Sigma = \emptyset$

Pas de répétitions ni d'ordre : $\{a, b, c, b\} = \{a, b, c\}$

Mot = concaténation de symboles

On fixe un alphabet Σ ,

- Tout élément de Σ est un mot
- Concaténation : Si w et v sont des mots, alors $w \cdot v$ est un mot
 - Associativité : $w \cdot (v \cdot u) = (w \cdot v) \cdot u$
 - Non-commutativité : $0 \cdot 1 \neq 1 \cdot 0$
- ε représente le mot vide
 - $w \cdot \varepsilon = \varepsilon \cdot w = w$
- w^n représente le mot $\underbrace{w \cdot w \dots w \cdot w}_{n \text{ times}}$

Pour $\Sigma = \{abc, a, tub\}$, $a \cdot abc \cdot a \cdot tub$ est un mot

On omettra souvent le \cdot quand l'alphabet ne crée pas d'ambiguïté:

Pour $\Sigma = \{a, b, c\}$, $a \cdot b \cdot c \cdot b = abcb$

Sous-mots et taille d'un mot

Si $z = uvw$

- u est un préfixe de z
- w est un suffixe de z
- u, v et w sont des facteurs de z .

Taille du mot z

- Notée $|z|$
- Nombre d'éléments dans z
- $|a \cdot l \cdot e \cdot s \cdot t \cdot o \cdot r \cdot m| = 8$
- $|\varepsilon| = 0$
- $|z| = |u| + |v| + |w|$

Combien existe-t-il de mots de taille 2 sur l'alphabet $\Sigma = \{a, b\}$?

Exercice

Lemma

Deux mots u et v commutent s'ils sont puissances d'un même troisième, i.e., s'il existe un mot w et des entiers i, j tels que $u = w^i$ et $v = w^j$.

Langages = Ensemble de mots

Tout ensemble de mots de Σ est un langage sur Σ .

Si L_1 et L_2 sont des langages, alors

- Union : $L_1 \cup L_2$ est un langage
 - $\{a, b, c, ab\} \cup \{a, b, cd\} = \{a, b, c, ab, cd\}$
 - associative et commutative
- Intersection : $L_1 \cap L_2$ est un langage
 - $\{a, b, c, ab\} \cap \{a, b, cd\} = \{a, b\}$
 - associative et commutative
- Différence : $L_1 \setminus L_2$ est un langage
 - $\{a, b, c, ab\} \setminus \{a, b, cd\} = \{c, ab\}$
 - non-associative et non-commutative

Autres opérations sur les langages

- concaténation : $L_1 \cdot L_2$ est un langage
 - Tout mot de L_1 concaténé à un mot de L_2
 - $\{a, b, c, ab\} \cdot \{a, b, cd\} =$
 $\{aa, ab, acd, ba, bb, bcd, ca, cb, ccd, aba, abb, abcd\}$
 - associative, non-commutative
- Puissance : L_1^n pour n entier est un langage
 - Correspond à $\underbrace{L_1 \cdot L_1 \dots L_1 \cdot L_1}_{n \text{ times}}$

Étoile de Kleene

L^* est l'ensemble des mots obtenus par concaténation arbitraire

- $L^* = L^0 \cup L^1 \cup L^2 \cup L^3 \cup \dots$
- Aussi appelé itéré ou fermeture de L
- $\{a, b\}^*$ est l'ensemble des mots écrits dans l'alphabet $\{a, b\}$
- idempotent : $(L^*)^* = L^*$
- $L^0 = \varepsilon \in L^*$
- $L^+ = L^* \setminus \{\varepsilon\}$

Exercice

Quel langage est décrit par les structures suivantes :

- $(\{a, b\} \setminus \{a\}) \cup \{a\}$
- $(\{a, b\} \cap \{a\}) \cup \{a\}$
- $\{a, b\}^2 \cap \{a\}^*$

Est-ce que les mots suivants appartiennent au langage $(\{a, b\} \cdot \{\varepsilon, r\})^*$

- ε
- a
- $babar$

Notion d'induction

Soit L un langage satisfaisant

- $\varepsilon \in L$
- pour tout $w \in L$ et $a \in \Sigma$, $wa \in L$.

Que pouvez-vous dire sur L ?

Notion d'induction

Soit L un langage satisfaisant

- $\varepsilon \in L$
- pour tout $w \in L$ et $a \in \Sigma$, $wa \in L$.

Que pouvez-vous dire sur L ?

Lemma

Soit P une propriété définie sur les mots de Σ^* et telle que

- $P(\varepsilon)$
- pour $w \in \Sigma^*$ et $a \in \Sigma$, si $P(w)$ alors $P(wa)$.

Alors $P(w)$ pour tout $w \in \Sigma^*$.

Notion d'induction

Soit L un langage satisfaisant

- $\varepsilon \in L$
- pour tout $w \in L$ et $a \in \Sigma$, $wa \in L$.

Que pouvez-vous dire sur L ?

Lemma

Soit P une propriété définie sur les mots de Σ^* et telle que

- $P(\varepsilon)$
- pour $w \in \Sigma^*$ et $a \in \Sigma$, si $P(w)$ alors $P(wa)$.

Alors $P(w)$ pour tout $w \in \Sigma^*$.

Prouvez que tout mot du langage $\{aa, bb\}^*$ est de longueur paire.

Outline

- 1 Qu'est-ce qu'un langage ?
- 2 Construction et opérations sur les langages
- 3 Expressions régulières**
- 4 Critères de régularité

Un formalisme pour générer certains langages

Expressions régulières

- Parfois appelées expressions rationnelles
- Génère un langage "régulier"

Définition récursive sur un alphabet $\Sigma = \{a, b\}$:

- ε , a et b sont des expressions régulières pour $\{\varepsilon\}$, $\{a\}$ et $\{b\}$
- Si r_1 et r_2 sont des expressions régulières générant L_1 et L_2 , alors
 - $r_1 \cdot r_2$ génère $L_1 \cdot L_2$
 - $r_1 + r_2$ génère $L_1 \cup L_2$
 - r_1^* génère L_1^*
 - (r) est une expression régulière générant L_1

→ les parenthèses servent à ordonner l'application des opérations

Quelques exemples

Quelles langages pour les expressions régulières suivantes :

- $(a + b)^*$
- $a + b^*$
- $a(a)^*$
- $(a^*b^*)^*$
- $(a + ab^*a)^*$

Quelles expressions rationnelles pour les langages suivants :

- les mots n'ayant que des a ou que des b
- $\{am, bm, an, cn\}$
- les mots de $\{a, i, m, o, u\}^*$ ayant *miaou* en facteur
- $\{a^n b^n \mid n \in \mathbb{N}\}$.

Outline

- 1 Qu'est-ce qu'un langage ?
- 2 Construction et opérations sur les langages
- 3 Expressions régulières
- 4 Critères de régularité

Une règle d'or

Si des relations existent entre les exposants apparaissant dans la description du langage, alors celui-ci n'est pas régulier.

- $\{a^n b^n \mid n \in \mathbb{N}\}$
- $\{a^n b^m c^k \mid n, m, k \in \mathbb{N} \wedge k \geq n + m\}$

ne sont pas réguliers.

Plusieurs critères formels de non-régularité

- Théorème de Myhill-Nerode (complexe)
- Lemme de l'étoile (simple, mais ne marche pas tout le temps)

Lemme de l'étoile

Theorem

Soit L un langage régulier. Il existe un entier N tel que tout mot w de L de longueur $|w| \geq N$ possède une factorisation $w = xyz$ avec $0 < |y|$ telle que

- 1 $0 < |xy| \leq N$ et
- 2 $xy^n z \in L$ pour tout entier $n \geq 0$.

Lemme de l'étoile

Theorem

Soit L un langage régulier. Il existe un entier N tel que tout mot w de L de longueur $|w| \geq N$ possède une factorisation $w = xyz$ avec $0 < |y|$ telle que

- 1 $0 < |xy| \leq N$ et
- 2 $xy^n z \in L$ pour tout entier $n \geq 0$.

Quid de $\{a^n b^n \mid n \in \mathbb{N}\}$?

Exercice

Les langages suivants sont-ils réguliers ?

- $\{a^n \mid n \text{ est un nombre premier}\}$
- $\{a^n b^m \mid n \neq m\}$
- Le langage des palindromes
- $(ab)^* \cap \{w \mid |w|_a = |w|_b\}$
- $ab(a + b)^* \cap \{w \mid |w|_a = |w|_b\}$

Quelques références

Quelques liens vers des documents ayant aidé à réaliser ce cours :

- <https://perso.liris.cnrs.fr/christine.solnon/langages.pdf>
- http://www.discmath.ulg.ac.be/cours/main_autom.pdf
- <https://pageperso.lis-lab.fr/frederic.olive/Materiel/langagesL2/cours.pdf>
- <https://damien.nouvel.net/fr/enseignement>
- https://www.i3s.unice.fr/nlt/cours/licence/it/s6_itdut_poly.pdf