

LF – Théorie des langages formels

*Sylvain Brandel*

2022 – 2023

[sylvain.brandel@univ-lyon1.fr](mailto:sylvain.brandel@univ-lyon1.fr)



CM 2

# ALPHABETS ET LANGAGES

# Alphabets et langages

## Mot

- **Alphabet** : *ensemble fini, non vide*, de symboles  
Généralement noté  $\Sigma$
- **Mot** ou **chaîne** (ang. *string*) sur un alphabet  $\Sigma$  :  
*suite finie* d'éléments de  $\Sigma$
- On note  $\Sigma^*$  l'ensemble de **tous** les mots (y compris le mot vide) définis sur  $\Sigma$

# Alphabets et langages

## *Mot*

- **Longueur** : nombre de symboles d'un mot
- Deux mots  $u$  et  $v$  sont **égaux** ssi
  - ils ont même longueur
  - $\forall i \in \{1, \dots, |u|\} : u_i = v_i$

# Alphabets et langages

## Mot

- **Concaténation** de 2 mots  $u$  et  $v$  de  $\Sigma^*$  : mot noté  $uv$  et défini par :
  - $u = u_1u_2\dots u_n, v = v_1v_2\dots v_n \rightarrow w = u_1\dots u_nv_1\dots v_n$ 
    - $\forall i \in \{1, |u|\} \quad (uv)_i = u_i$
    - $\forall i \in \{|u|+1, \dots, |u|+|v|\} \quad (uv)_i = v_{i-|u|}$
- **Propositions**
  - la concaténation est régulière à droite et à gauche
    - $wu = wv \Rightarrow u = v$
    - $uw = vw \Rightarrow u = v$
  - $|uv| = |u| + |v|$

# Alphabets et langages

## Mot

- **Facteur gauche** de  $w$  : mot  $u$  tel que  $uv = w$
- **Facteur droit** de  $w$  : mot  $v$  tel que  $uv = w$
- **Facteur** de  $w$  : mot  $u$  tel que il existe  $v$  et  $v'$  tels que  $vuv' = w$
- **Miroir (Reverse)**
  - Fonction miroir  $_R : \Sigma^* \rightarrow \Sigma^*$  définie par récurrence :
    - $w$  tq  $|w| = 0 : w^R = \varepsilon^R = \varepsilon$
    - $w$  tq  $|w| > 0 : \exists a \in \Sigma$  tq  $w = au$   
et  $w^R = (au)^R = u^R a$
- **Propriété**
  - $\forall u, v \in \Sigma^* : (uv)^R = v^R u^R$

# Alphabets et langages

## Langage

- Langage sur  $\Sigma$  : *ensemble* de mots sur  $\Sigma$
- Remarques de cardinalité
  - $\Sigma$  fini
  - $\Sigma^*$  infini dénombrable (rappel : dont on peut énumérer les éléments)
  - $P(\Sigma^*)$  est infini non dénombrable
- Opérations sur les langages
  - $\cup, \cap, \neg$  (complément),  $\setminus \Rightarrow$  comme d'habitude  
(complément :  $\neg A = \Sigma^* \setminus A$ )
  - Concaténation :  $L_1 \subset \Sigma^*, L_2 \subset \Sigma^*$ 
    - $L = L_1.L_2$  ou  $L_1L_2$   
est défini par  $L = \{w \mid \exists w_1 \in L_1 \text{ et } \exists w_2 \in L_2 : w = w_1w_2\}$
  - Clôture de Kleene (Kleene star) ou étoile de  $L$ 
    - $L^* = \{w \in \Sigma^* \mid \exists k \in \mathbb{N}, \exists w_1, w_2, \dots, w_k \in L : w = w_1w_2\dots w_k\}$

# Représentation finie des langages

- Langages définis par
  - Éléments de base :
    - L'ensemble  $\emptyset$
    - Le mot vide  $\varepsilon$
    - Les singletons sur  $\Sigma$
  - Opérations :
    - La concaténation de langages
    - La réunion de deux langages
    - La fermeture de Kleene
- De tels langages sont appelés **langages rationnels**

# Représentation finie des langages

- Description et manipulation des langages **rationnels**
  - **Génération** *Grammaires régulières*
    - Un moyen de générer un langage
  - **Reconnaissance** *Automates finis*
    - Un moyen de décider l'appartenance
  - Caractérisation **algébrique** *Expressions régulières*
    - Notations et équations *Systèmes*



# Représentation des langages

- Description et manipulation des langages
  - Génération *Grammaires*
    - Un moyen de générer un langage
  - Reconnaissance *Automates*
    - Un moyen de décider l'appartenance
  - Caractérisation algébrique
    - Notations et équations