info633 : Théorie des langages
TD 2 : expressions régulières – 2, dérivation

Pierre Hyvernat
Laboratoire de mathématiques de l'université de Savoie
bâtiment Chablais, bureau 22, poste : 94 22
email : `Pierre.Hyvernat@univ-savoie.fr`
www : `http://www.lama.univ-savoie.fr/~hyvernat/`

Exercice 1 : Constructions d'expressions régulières

Question 1. Construisez des expressions régulières sur $\Sigma = \{a, b\}$ dont les langages associés sont exactement :
- l'ensemble des mots ayant un nombre pair de symboles,
- l'ensemble des mots contenant un nombre pair de a,
- l'ensemble des mots contenant un nombre pair de a ou un nombre pair de b (ou les deux),
- l'ensemble des mots contenant un nombre pair de a et un nombre pair de b.

Question 2. Construisez une expression régulière sur $\Sigma = \{a, b\}$ dont le langage associé est exactement l'ensemble des mots où chaque préfixe contient "presque autant" de a que de b : la différence entre le nombre de a et de b est au plus 1.

Par exemple, *abababababa* est dans le langage, ainsi que *ababbabaab*. Par contre, *aabb* n'est pas dans le langage...

Exercice 2 : Équivalences d'expressions régulières

Question 1. Essayez de simplifier les expressions régulières suivantes en donnant une expression équivalente qui utilise moins de symboles. :
- $1 + ab + aba(ba)^*b$
- $(a^*b)^* + (b^*a)^*$
- $(abc)^* + ((abc)^*d(abc)^*)^*$

Question 2. On note $R_1 \leq R_2$ pour signifier que $R_1 + R_2 = R_2$. Cela signifie que $\mathcal{L}(R_1) \subseteq \mathcal{L}(R_2)$.

C'est une relation d'ordre : si $R_1 \leq R_2$ et $R_2 \leq R_3$, alors $R_1 \leq R_3$.

Cela permet de simplifier des expressions régulières car on peut supprimer des termes dans les sommes.

Simplifiez les expressions suivantes :
- $a + aa + (aa)^*$
- $(ab^* + c)^* + (ac)^*$
- $a(b + c)^* + a(bc)^*$

Exercice 3 : Dérivées d'expressions régulières

Question 1. Calculez les dérivées R/a, R/b et R/ab pour $R = a^*(aab + bb^*a + bb)^*$.

Est-ce que ces dérivées contiennent le mot vide ?

Que peut-on en conclure sur l'appartenance de a, b et ab au langage $\mathcal{L}(R)$?

Question 2. Calculez les dérivées R/a, R/aa et R/aaa pour les expressions suivantes :
- $R = (aaa)^*$
- $R = (aaa)^* + (aaaaa)^*$
- $R = (aaa + aaaaa)^*$

Regardez ce qui se passe quand on essaie de vérifier si a^k est dans le langage associé de R. Pour ceci, essayer de calculer la forme de toutes les dérivées possibles de R/a^k, en précisant lesquelles contiennent le mot vide.

Que constatez-vous ?

Question 3. Calculez toutes les dérivées (pour l'alphabet $\Sigma = \{a, b\}$) des expressions

- $R = (aaa)^* + (bbbbb)^*$
- $R = (aaa + bbbbb)^*$

Comment expliquez-vous que la situation soit plus simple que pour la question précédente ?

Question 4. Dessinez le graphe des dérivées pour les expressions régulières de la question précédente : les sommets sont les dérivées de R (y compris la dérivée le long de ε), et il y a un arc "a" de D/w vers D/wa et un arc "b" de D/w vers D/wb.

Comment peut-on utiliser ce graphe pour savoir si un mot w est dans le langage de R ?

Question 5. Décrivez une stratégie pour implanter le test "appartenance au langage $\mathcal{L}(R)$" lorsque l'expression R ne change pas.

Quelle est la complexité "amortie" de ce test ?

Exercice 4 : Extension des expressions régulières

On ajoute un opérateur $\neg$ aux expressions régulières, avec la définition

$$w \in \mathcal{L}(\neg R) \quad \text{si et seulement si} \quad w \notin \mathcal{L}R$$

La dérivée s'étend naturellement avec $(\neg R)/a = \neg(R/a)$ car

$$w \in \mathcal{L}(\neg R)/a) \iff aw \notin \mathcal{L}(R) \iff w \notin \mathcal{L}(R/a) \iff w \in \mathcal{L}(\neg(R/a))$$

Question 1. Que pensez-vous du langage associé à l'expression $\neg 0$?

Question 2. En utilisant la technique des dérivées, montrez que le mot $aabb$ est dans le langage de l'expression $(a + b)^* \neg(bb^*)$ (sur l'alphabet $\Sigma = \{a, b\}$).

Que pensez-vous du langage de cette expression ?

Question 3. Donnez une expression "étendue" dont le langage associé est exactement l'ensemble des mots sur $\Sigma = \{a, b, c\}$ qui ne contiennent ni aa ni bb.

Comparez avec l'expression régulière "normale" donnée pour ce langage dans le TD1.