MPI 4 - Cours 2

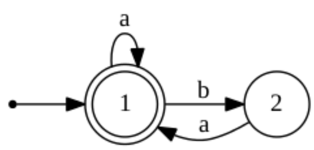
Un peu de calcul non commutatif : séries rationnelles et langages rationnels

Considérons l'alphabet $A = \{a, b\}$, et intéressons nous au langage $L = (a + ba)^*$.

L'ensemble $C = \{a, ba\}$ est ce qu'on appelle un *code préfixe*, c'est à dire que s'il contient un mot w, il ne contient aucun préfixe u de w (un mot tel que $w = uv$ avec $u, v \neq \epsilon$). C'est un *code*, car tout mot de C^* n'a qu'une seule écriture comme concaténation d'éléments de C (unicité du déchiffrage, ou encore, non-ambiguïté de la grammaire) :

$$w = baaabaaababaaa \rightarrow ba.a.a.ba.a.a.ba.ba.a.a$$

Le langage L est reconnu par l'automate



Soit u_n le nombre de mots de longueur n dans L. On a

$$L = \{\epsilon, a, ba, aa, baa, aba, aaa, baba, baaa, abaa, aaba, aaaa, \cdots\}$$

et donc

$$u_0 = 1,\ u_1 = 1,\ u_2 = 2,\ u_3 = 3,\ u_4 = 5, \cdots$$

Si on note L_n l'ensemble des mots de longueur n dans L, on voit que pour $n \geq 2$,

$$L_n = aL_{n-1} + baL_{n-2}$$

puisqu'un mot de L commence soit par a, soit par ba. Donc, $u_n = u_{n-1} + u_{n-2}$ avec $u_0 = 1$, $u_1 = 1$, c'est donc la suite de Fibonacci décalée de 1.

Une technique classique pour étudier ce genre de problème est d'identifier les mots à des monômes en variables non commutatives, la concaténation devenant alors la multiplication et son élément neutre ϵ le scalaire 1. Le code C s'identifie alores au polynôme

$$C = a + ba$$

et le langage L s'identifie à la série formelle

$$L = C^* = \sum_{n\geq 0} C^n = 1 + C + C^2 + \cdots = \frac{1}{1-C} = \frac{1}{1-a-ba} = 1 + a + ba + baa + aba + aaa + \cdots.$$

On trouve alors automatiquement la série génératrice de u_n en remplaçant a et b par x dans cette expression, car les mots de longueur n deviennent tous x^n

$$U(x) = \frac{1}{1-x-x^2} = 1 + x + 2x^2 + 3x^3 + 5x^4 + 8x^3 + \cdots$$

C'est donc la suite de Fibonacci décalée $u_n = F_{n+1}$.

D'une manière générale, l'expression C^* se traduit par $(1-C)^{-1}$, mais on ne peut en déduire le dénombrement des mots par longueur que si l'expression est non ambiguë. Par exemple, $C = a + ab + ba$ n'est pas un code, puisque $w = aba$ possède deux dérivations : $w = a.ba = ab.a$. Dans la série non commutative

$$(1 - a - ab - ba)^{-1} = 1 + a + ab + ba + 2aba + aab + baa + aaa + \cdots$$

le coefficient d'un mot est le nombre de ses factorisations en éléments de C. On peut montrer en construisant l'automate minimal de C^* qu'une expression non ambigue serait

$$((b + a(a+ba)^* bb)a)^*(\epsilon + a((a+ba)^*(\epsilon + b))$$

En remplaçant a et b par x et les X^* par des $(1-X)^{-1}$, on trouve pour la série génératrice

$$\frac{1}{1-x-2x^2+x^3} = 1 + x + 3x^2 + 4x^3 + 9x^4 + 14x^5 + 28x^6 + 47x^7 + 89x^8 + 155x^9 + 286x^{10} + 507x^{11} + O\left(x^{12}\right)$$

Si on avait fait cela avec l'expression ambiguë, on aurait trouvé le résultat faux

$$\frac{1}{1-x-2x^2} = 1 + x + 3x^2 + 5x^3 + 11x^4 + 21x^5 + 43x^6 + 85x^7 + 171x^8 + 341x^9 + 683x^{10} + 1365x^{11} + O\left(x^{12}\right)$$

La décomposition en éléments simples

Lorsque la série génératrice d'une suite représente une fraction rationnelle, on peut la décomposer en éléments simples, pourvu que l'on connaisse les racines du dénominateur : si

$$U(x) = \frac{A(x)}{B(x)}$$

avec

$$B(x) = (x-\beta_1)^{m_1}(x-\beta_2)^{m_2}\cdots(x-\beta_n)^{m_n}$$

alors, on peut trouver une décomposition de la forme

$$U(x) = P(x) + \sum_{k=1}^{n}\sum_{i_k=1}^{m_k}\frac{a_{ik}}{(x-\beta_k)^{i_k}}$$

où $P(x)$ est un polynôme (qui n'apparaît que si le degré de A est supérieur ou égal à celui de B, auquel cas c'est le quotient de A par B), et pour chaque racine β, on a un élément simple $\frac{1}{(x-\beta)^k}$ pour k de 1 jusqu'à la multiplicité m de β.

In [44]:
```
from sympy import *
init_printing()
var('x')
```
Out[44]:

$$x$$

In [45]:
```
# Exemple au hasard
U = (2*x**12-x**3+x+2)/((1-2*x)**3*(1+3*x)**2*(1-x)**4);U
```
Out[45]:

$$\frac{2x^{12}-x^3+x+2}{(-2x+1)^3(-x+1)^4(3x+1)^2}$$

In [46]:
```
U.apart()
```
Out[46]:

$$-\frac{x^3}{36}-\frac{29x^2}{216}-\frac{85x}{216}-\frac{3485}{3888}+\frac{1054999}{11664000x+3888000}+\frac{45271}{388800(3x+1)^2}-\frac{153003}{8000x-4000}-\frac{1389}{80(2x-1)^2}-\frac{973}{160(2x-1)^3}+\frac{2213}{128x-128}-\frac{411}{64(x-1)^2}\cdot$$

Notons que cette décomposition peut faire intervenir des racines complexes ou irrationnelles. Pour que `sympy` accepte de les calculer (s'il ne doit résoudre que des équations de degré au plus 4), il faut utiliser l'option `full=True` et la méthode `doit`.

In [47]:
```
V = 1/(1+x+x**2);V
```
Out[47]:

$$\frac{1}{x^2+x+1}$$

In [48]:
```
V.apart()
```
Out[48]:

$$\frac{1}{x^2+x+1}$$

In [49]:
```
V.apart(full=True).doit()
```
Out[49]:

$$\frac{\sqrt{3}i}{3x+\frac{3}{2}+\frac{3i}{2}\sqrt{3}}-\frac{\sqrt{3}i}{3x+\frac{3}{2}-\frac{3i}{2}\sqrt{3}}$$

In [50]:
```python
# Avec des équations du 3ème degré, le résultat n'est pas très exploitable ...
www=1/(1+x+x**3)
www.apart(full=True).doit()
```

Out[50]:

$$\frac{1}{x - \frac{1}{\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}}} + \frac{1}{3}\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}}\left(-\frac{9}{31\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}}} + \frac{6}{31}\left(-\frac{1}{3}\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}} + \frac{1}{\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}}}\right)^2 + \frac{4}{31} + \frac{3}{31}\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}}\right)} + \frac{\quad}{x - \frac{1}{\left(-\frac{1}{2} + \frac{\sqrt{3}i}{2}\right)\sqrt[3]{\frac{27}{2} + \frac{3\sqrt{93}}{2}}}}$$

La formule du binôme généralisé

On voit qu'il faut être capable de développer en série entière les fractions simples

$$\frac{1}{(1-x)^k}.$$

Pour $k = 1$, c'est fait. On peut traiter $k = 2$ en l'élevant au carré. En utilisant la formule générale de produit

$$\sum_{p\ge 0} a_p x^p \sum_{q \ge 0} b_q x^q = \sum_{n\ge 0}\left(\sum_{p=0}^{n} a_p b_{n-p}\right)x^n$$

on a

$$\left(\frac{1}{1-x}\right)^2 = \sum_{n\ge 0}\left(\sum_{p=0}^{n} 1 \times 1\right)x^n = \sum_{n\ge 0}(n+1)x^n$$

Pour peu que l'on sache que

$$1 + 2 + \cdots + n = \frac{n(n+1)}{2}$$

on peut remultiplier par $\frac{1}{1-x}$ et obtenir

$$\left(\frac{1}{1-x}\right)^3 = \sum_{n\ge 0}\left(\sum_{p=0}^{n}(p+1)\times 1\right)x^n = \sum_{n\ge 0}\frac{(n+1)(n+2)}{2}x^n$$

Avec un effort supplémentaire, on peut encore trouver $k = 4$ et deviner la formule générale

$$\left(\frac{1}{1-x}\right)^k = \sum_{n\ge 0}\binom{n+k-1}{n}x^n$$

et la prouver par récurrence.

Ou encore, la prouver combinatoirement : si on fait le produit de k séries géométriques

$$\frac{1}{(1-x_1)(1-x_2)\cdots(1-x_k)} = \sum_{n_1\ge 0} x_1^{n_1}\sum_{n_2\ge 0} x_2^{n_2}\cdots\sum_{n_k \ge 0} x_k^{n_k},$$

on voit que le membre droit est la somme de tous les monômes en $x_1, \ldots, x_k$. Si l'on pose

$$x_1 = x_2 = \cdots = x_k = x$$

dans cette égalité, le membre gauche devient $\left(\frac{1}{1-x}\right)^k$, et le coefficient de x^n dans le membre droit est le nombre de monômes de degré total n que l'on peut former avec k variables, c'est à dire, le nombre de sélections avec répétitions de n objets parmi k, connu pour être égal à $\binom{n+k-1}{n}x^n$ d'après le cours de L2.

In [54]:
```python
series(1/(1-x)**2,x,0,10)
```

Out[54]:

$$1 + 2x + 3x^2 + 4x^3 + 5x^4 + 6x^5 + 7x^6 + 8x^7 + 9x^8 + 10x^9 + \mathcal{O}\left(x^{10}\right)$$

In [51]:
```
series(1/(1-x)**3,x,0,10)
```
Out[51]:

$$1 + 3x + 6x^2 + 10x^3 + 15x^4 + 21x^5 + 28x^6 + 36x^7 + 45x^8 + 55x^9 + \mathcal{O}\left(x^{10}\right)$$

In [52]:
```
series(1/(1-x)**4,x,0,10)
```
Out[52]:

$$1 + 4x + 10x^2 + 20x^3 + 35x^4 + 56x^5 + 84x^6 + 120x^7 + 165x^8 + 220x^9 + \mathcal{O}\left(x^{10}\right)$$

On peut faire encore mieux. Rappelons qu'on définit $x^{p/q}$ comme $\sqrt[q]{x^p}$, et qu'on peut définir x^α pour α réel quelconque comme $\exp(\alpha \ln x)$.

On peut donc donner un sens à la série formelle $(1+x)^\alpha$ pour α quelconque. Posons

$$(1+x)^\alpha = \sum_{n\ge 0} a_n x^n$$

et cherchons les coefficients a_n.

On *définit* la dérivation des séries formelles en posant $(x^n)' = \frac{d}{dx}x^n = nx^{n-1}$
opération vérifie les propriétés usuelles de la dérivée des fonctions. Par exemple, pour montrer $(fg)' = f'g + fg'$
$g = x^q$
quelconques.

On en déduit ensuite par récurrence $(f^n)' = nf^{n-1}f'$

Comme $(1+x)^\alpha = \exp(\alpha\ln(1+x))$

$$\frac{d}{dx}(1+x)^\alpha = \exp(\alpha\ln(1+x))\alpha\frac{1}{1+x} = \alpha(1+x)^{\alpha-1},$$

On peut maintenant écrire

$$(1+x)\frac{d}{dx}(1+x)^\alpha = (1+x)\sum_{n\ge 0} na_n x^{n-1} = \alpha(1+x)^\alpha = \sum_{n\ge 0}\alpha a_n x^n.$$

$$\sum_{n\ge 0}(n+1)a_{n+1}x^n + \sum_{n\ge 0} na_n x^n = \sum_{n\ge 0}\alpha a_n x^n$$

$$(n+1)a_{n+1} = (\alpha-n)a_n,\ \text{d'où } a_{n+1} = \frac{\alpha-n}{n+1}\frac{\alpha-n+1}{n}\frac{\alpha-n+2}{n-1}\cdots\frac{\alpha}{1}a_0.$$

$$a_n = \frac{\alpha(\alpha-1)\cdots(\alpha-n+1)}{n!} =: \binom{\alpha}{n}.$$

ordinaires.

Exemple : $\alpha = \frac{1}{2}$

On a

$$\binom{1/2}{n} = \frac{\frac{1}{2}(\frac{1}{2}-1)(\frac{1}{2}-2)\cdots(\frac{1}{2}-n+1)}{n!} = \frac{1(1-2)(1-4)\cdots(1-2n+2)}{2^n n!} = (-1)^{n-1}\frac{1\cdot 3\cdot 5\cdots(2n-3)}{2^n n!}$$

$$= (-1)^{n-1}\frac{(2n-2)!}{2^{n-1}(n-1)!2^n n!} = \frac{(-1)^{n-1}}{2^{2n-1}}\frac{1}{n}\binom{2n-2}{n-1}.$$

Application : dénombrement des arbres binaires

Un *arbre binaire complet*, c'est soit un sommet unique o, soit

```
   o
  / \
 A   B
```

où A et B sont des arbres binaires complets. Soit c_n

```
o         o                             o            o
         / \                           / \          / \
        o   o                         o   o        o   o
                                     / \              / \
                                    o   o            o   o
```

```
    _o__           __o___           __o__           __o___          _o__
   /    \         /      \         /     \         /      \        /    \
  o     _o_      o       _o_      o       o      _o_      o      _o_     o
       /   \            /   \    / \     / \    /   \            /   \
      o     o          o     o  o   o   o   o  o     o          o     o
           / \        / \                           / \        / \
          o   o      o   o                         o   o      o   o
```

On a donc $c_0 = 0$

Comment calculer les c_n

Un arbre à n
p, q

$$c_n = \sum_{p=1}^{n-1} c_p c_{n-p} = c_1 c_{n-1} + c_2 c_{n-2} + \cdots + c_{n-1} c_1.$$

$$c_2 = c_1 c_1 = 1,$$

$$c_3 = c_1 c_2 + c_2 c_1 = 1 + 1 = 2,$$

$$c_4 = c_1 c_3 + c_2 c_2 + c_3 c_1 = 2 + 1 + 2 = 5,$$

$$c_5 = c_1 c_4 + c_2 c_3 + c_3 c_2 + c_4 c_1 = 5 + 2 + 2 + 5 = 14,$$

$$c_6 = c_1 c_5 + c_2 c_4 + c_3 c_3 + c_4 c_2 + c_5 c_1 = 14 + 5 + 4 + 5 + 14 = 42, \ldots$$

Euler qui les appelait lui-même *nombres de Segner*. Ce [livre](https://books.google.fr/books/about/Catalan_Numbers.html?id=i5QSBwAAQBAJ&redir_esc=y) (https://books.google.fr/books/about/Catalan_Numbers.html?id=i5QSBwAAQBAJ&redir_esc=y) de Richard Stanley en recense 214 interprétations différentes.

Formons la série génératrice

$$C(x) = \sum_{n \geq 0} c_n x^n = 0 + x + x^2 + 2x^3 + 5x^4 + 14x^5 + 42x^6 + \cdots$$

$$c_n = \sum_{p=0}^{n} c_p c_{n-p} \quad \text{pour } n \geq 2$$

$$C(x) = x + C(x)^2.$$

$$X^2 - X + x = 0,$$

$$C(x) = \frac{1 \pm \sqrt{1-4x}}{2}$$

terme soit x

$$C(x) = \frac{1-\sqrt{1-4x}}{2} = \frac{1}{2}\left[1 - \sum_{n \geq 0} \binom{1/2}{n}(-4x)^n\right] = \sum_{n \geq 1} \frac{(-1)^n}{2^{2n}} \frac{1}{n} \binom{2n-2}{n-1}(-4x)^n = \sum_{n \geq 1} \frac{1}{n} \binom{2n-2}{n-1} x^n$$

$$c_n = \frac{1}{n}\binom{2n-2}{n-1}.$$

$$c_6 = \frac{1}{6}\binom{10}{5} = \frac{1}{6} 252 = 42.$$

In [53]:
```
[binomial(2*n-2,n-1)//n for n in range(1,12)]
```
Out[53]:

$$[1, \quad 1, \quad 2, \quad 5, \quad 14, \quad 42, \quad 132, \quad 429, \quad 1430, \quad 4862, \quad 16796]$$

In [ ]: