

Génération aléatoire sous modèle de Boltzmann

le cas non étiqueté

Carine Pivoteau
en collaboration avec P. Flajolet et E. Fusy

Laboratoire d'Informatique de Paris 6 (LIP6)

INRIA Rocquencourt

Plan de l'exposé

- 1 Le modèle de Boltzmann
- 2 Constructions propres au cas non étiqueté
- 3 Complexité

Plan de l'exposé

- 1 Le modèle de Boltzmann
- 2 Constructions propres au cas non étiqueté
- 3 Complexité

Plan de l'exposé

- 1 Le modèle de Boltzmann
- 2 Constructions propres au cas non étiqueté
- 3 Complexité

Introduction

- Génération aléatoire d'objets appartenant à des classes combinatoires **décomposables**
- Génération en **taille fixée**, avec une **distribution uniforme**.
 - Méthode bijective
 - Méthode récursive
 - *Nijenhuis, Wilf, 1978*
 - *Flajolet, Zimmermann, Van Cutsem, 1994*
- Génération sous modèle de Boltzmann
 - Génération en **taille approchée, paramétrée**
 - **Distribution uniforme sur chaque ensemble d'objets de même taille**, répartie sur l'ensemble de la classe combinatoire

Le modèle de Boltzmann

Définition du modèle

Definition

Dans le cas non étiqueté, le modèle de Boltzmann assigne à tout objet $c \in \mathcal{C}$ la probabilité suivante :

$$\mathbb{P}_x(c) = \frac{x^{|c|}}{C(x)}$$

Un générateur de Boltzmann $\Gamma C(x)$ pour la classe $\mathcal{C}$ est un algorithme qui produit des objets de $\mathcal{C}$ suivant ce modèle.

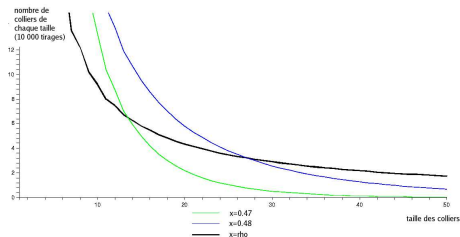
→ 2 objets de même taille ont la même probabilité d'être tirés.

Distribution en taille

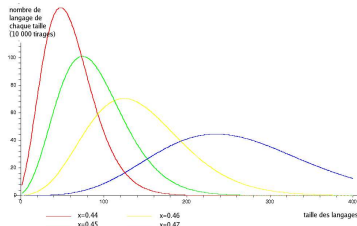
La probabilité de tirer un objet de taille N est alors :

$$\mathbb{P}_x(N = n) = \sum_{|c|=n} \mathbb{P}_x(c) = \frac{C_n x^n}{C(x)}$$

→ distribution en taille des objets engendrés, en fonction de x .



Colliers bicolores



Langages finis

Générateurs pour ε , $\mathcal{Z}$, $+$, $\times$ et Seq

construction	générateur
$\mathcal{C} = \emptyset$	$\Gamma C(x) := \varepsilon$
$\mathcal{C} = \mathcal{Z}$	$\Gamma C(x) := z$
$\mathcal{C} = \mathcal{A} + \mathcal{B}$	$\Gamma C(x) := \left(\text{Bern } \frac{A(x)}{C(x)} \longrightarrow \Gamma A(x) \mid \Gamma B(x) \right)$
$\mathcal{C} = \mathcal{A} \times \mathcal{B}$	$\Gamma C(x) := (\Gamma A(x); \Gamma B(x))$
$\mathcal{C} = \text{SEQ}(\mathcal{A})$	$\Gamma C(x) := (\text{Geom } A(x) \Longrightarrow \Gamma A(x))$

Remarque : Les valeurs de $A(x)$ et $C(x)$ sont données par des oracles.

Des arbres binaires

$$B(z) = \sum_{b \in \mathcal{B}} z^{|b|} = z + B(z)^2$$

$$B(z) = \frac{1 - \sqrt{1 - 4z}}{2}$$

- processus de
branchement binaire
- en la singularité $\rho = \frac{1}{4}$,
on tire une feuille avec
probabilité $\frac{1}{2}$

Algorithme : $\Gamma B(x)$

```
 $b \leftarrow \text{Bern}(x/B(x))$ ; ← B(x) est calculable : oracle  
if  $b = 1$  then  
    Retourner ■  
else  
    Retourner □  $\langle \Gamma B(x), \Gamma B(x) \rangle$ ;  
end if
```

Des compositions d'entier

$$\mathcal{C} = \text{SEQ}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

$$C(z) = \frac{1}{1 - \frac{z}{1-z}} = \frac{1-z}{1-2z}$$

Algorithme : $\Gamma C(x)$

$k \leftarrow \text{Geom}(x/(1-x))$;

$C \leftarrow \emptyset$;

for i from 1 to k do

$s \leftarrow \text{Geom}_{\geq 1}(x)$;

$C \leftarrow C, s$;

end for

Return C ;

Des compositions d'entier

$$\mathcal{C} = \text{SEQ}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

$$C(z) = \frac{1}{1 - \frac{z}{1-z}} = \frac{1-z}{1-2z}$$

- tirage d'une loi géométrique

Algorithme : $\Gamma C(x)$

$k \leftarrow \text{Geom}(x/(1-x));$

$C \leftarrow \emptyset;$

for i from 1 to k do

$s \leftarrow \text{Geom}_{\geq 1}(x);$

$C \leftarrow C, s;$

end for

Return C ;

Des compositions d'entier

$$\mathcal{C} = \text{SEQ}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

$$C(z) = \frac{1}{1 - \frac{z}{1-z}} = \frac{1-z}{1-2z}$$

- tirage d'une loi géométrique

Algorithme : $\Gamma C(x)$

$k \leftarrow \text{Geom}(x/(1-x));$

$C \leftarrow \emptyset;$

for i from 1 to k do

$s \leftarrow \text{Geom}_{\geq 1}(x);$

$C \leftarrow C, s;$

end for

Return C ;

Des compositions d'entier

$$\mathcal{C} = \text{SEQ}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

$$C(z) = \frac{1}{1 - \frac{z}{1-z}} = \frac{1-z}{1-2z}$$

- tirage d'une loi géométrique
- décalage

Algorithme : $\Gamma C(x)$

$k \leftarrow \text{Geom}(x/(1-x));$

$C \leftarrow \emptyset;$

for i **from** 1 **to** k **do**

$s \leftarrow \text{Geom}_{\geq 1}(x);$

$C \leftarrow C, s;$

end for

Return $C;$

Des compositions d'entier

$$\mathcal{C} = \text{SEQ}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

$$C(z) = \frac{1}{1 - \frac{z}{1-z}} = \frac{1-z}{1-2z}$$

- tirage d'une loi géométrique
- décalage

Algorithme : $\Gamma C(x)$

$k \leftarrow \text{Geom}(x/(1-x));$

$C \leftarrow \emptyset;$

for i **from** 1 **to** k **do**

$s \leftarrow \text{Geom}_{\geq 1}(x);$

$C \leftarrow C, s;$

end for

Return $C;$

Constructions propres au cas non étiqueté

Théorème

- *Il existe un **procédé de construction effectif** qui, partant d'une classe combinatoire non étiquetée quelconque, spécifiée au moyen des constructions :*

$$\{\varepsilon, \mathbb{Z}, +, \times, \text{SEQ}, \text{MSET}, \text{PSET}, \text{CYC}\}$$

*produit un **générateur aléatoire conforme au modèle de Boltzmann.***

- *On peut compléter cette liste par des **restrictions de cardinalité** sur les constructions SEQ, MSET, PSET, CYC*

On couvre ainsi de nombreuses classes combinatoires :

- partitions, compositions, colliers,...
- arbres (et autres structures arborescentes),
- langage réguliers,
- graphes fonctionnels,...

Théorème

- *Il existe un **procédé de construction effectif** qui, partant d'une classe combinatoire non étiquetée quelconque, spécifiée au moyen des constructions :*

$$\{\varepsilon, \mathbb{Z}, +, \times, \text{SEQ}, \text{MSET}, \text{PSET}, \text{CYC}\}$$

*produit un **générateur aléatoire conforme au modèle de Boltzmann.***

- *On peut compléter cette liste par des **restrictions de cardinalité** sur les constructions SEQ, MSET, PSET, CYC*

On couvre ainsi de nombreuses classes combinatoires :

- partitions, compositions, colliers,...
- arbres (et autres structures arborescentes),
- langage réguliers,
- graphes fonctionnels,...

- 1 Le modèle de Boltzmann
- 2 **Constructions propres au cas non étiqueté**
 - Les multi-ensembles
 - Les ensembles sans répétitions
 - Les cycles
 - Restrictions de cardinalité
- 3 Complexité

Pour commencer : MSet₂

$\text{MSET}_2(\mathcal{A}) \cong$ ensemble non ordonné de deux objets de $\mathcal{A}$

$$\begin{aligned} \mathcal{C} &= \text{MSET}_2(\mathcal{A}) \\ C(z) &= \frac{1}{2}A^2(z) + \frac{1}{2}A(z^2) \quad \left[2\text{MSET}_2(\mathcal{A}) = \mathcal{A}^2 + \Delta\mathcal{A}^2 \right] \end{aligned}$$

Algorithme : $\Gamma C(x)$

```

if  $\text{Bern}\left(\frac{1}{2} \frac{A^2(x)}{C(x)}\right) = 1$  then
  Return (  $\Gamma A(x), \Gamma A(x)$  )
else
   $a \leftarrow \Gamma A(x^2)$ ;
  Return (  $a, a$  );
end if

```

Correction de l'algorithme

Démonstration.

Soit $\gamma = \langle \gamma_1, \gamma_2 \rangle$, un multi-ensemble de deux éléments de $\mathcal{A}$. Sa probabilité d'être engendré par $\Gamma C(x)$ est :

❶ $\gamma_1 \neq \gamma_2$

$$\mathbb{P}_x(\gamma) = \frac{1}{2} \frac{A(x)^2}{C(x)} \frac{x^{|\gamma_1|}}{A(x)} \frac{x^{|\gamma_2|}}{A(x)} \times 2 = \frac{x^{|\gamma|}}{C(x)}$$

❷ $\gamma_1 = \gamma_2$

$$\mathbb{P}_x(\gamma) = \frac{1}{2} \frac{A(x)^2}{C(x)} \frac{x^{|\gamma_1|}}{A(x)} \frac{x^{|\gamma_1|}}{A(x)} + \frac{1}{2} \frac{A(x)^2}{C(x)} \frac{x^{2|\gamma_1|}}{A(x^2)} = \frac{x^{|\gamma|}}{C(x)}$$



Un exemple simple

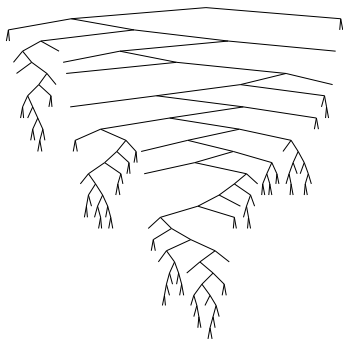
Les arbres binaires non planaires
 (arbres d'Otter)

$$\mathcal{B} = \mathcal{Z} + \text{MSET}_2(\mathcal{B})$$

$$B(z) = z + \frac{1}{2}B^2(z) + \frac{1}{2}B(z^2)$$

```

→ □
→ ■⟨ ΓB(x), ΓB(x) ⟩
→ ot ← ΓB(x2);
return ■⟨ ot, ot ⟩;
```



Un exemple simple

Les arbres binaires non planaires
 (arbres d'Otter)

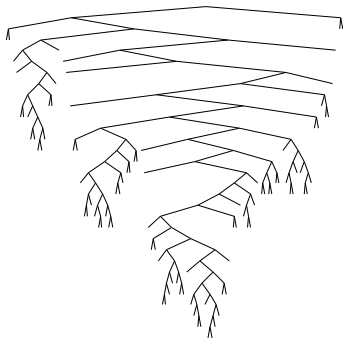
$$\mathcal{B} = \mathcal{Z} + \text{MSET}_2(\mathcal{B})$$

$$B(z) = z + \frac{1}{2}B^2(z) + \frac{1}{2}B(z^2)$$

→ □

→ ■ $\langle \Gamma B(x), \Gamma B(x) \rangle$

→ ot $\leftarrow \Gamma B(x^2);$
return ■ $\langle \text{ot}, \text{ot} \rangle;$



Un exemple simple

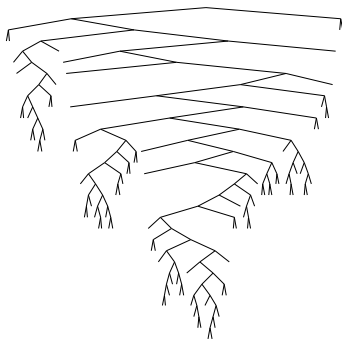
Les arbres binaires non planaires
 (arbres d'Otter)

$$\mathcal{B} = \mathcal{Z} + \text{MSET}_2(\mathcal{B})$$

$$B(z) = z + \frac{1}{2}B^2(z) + \frac{1}{2}B(z^2)$$

```

→ □
→ ■⟨ ΓB(x), ΓB(x) ⟩
→ ot ← ΓB(x2);
   return ■⟨ ot, ot ⟩;
```



Un exemple simple

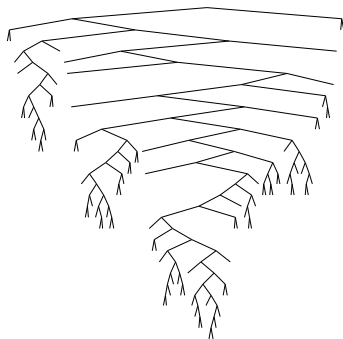
Les arbres binaires non planaires
(arbres d'Otter)

$$\mathcal{B} = \mathcal{Z} + \text{MSET}_2(\mathcal{B})$$

$$B(z) = z + \frac{1}{2}B^2(z) + \frac{1}{2}B(z^2)$$

```

→ □
→ ■⟨ ΓB(x), ΓB(x) ⟩
→ ot ← ΓB(x2);
   return ■⟨ ot, ot ⟩;
```



MSet généraux : une première approche

$$\mathcal{M} = \text{MSET}(\mathcal{A}) \cong \prod_{\gamma \in \mathcal{A}} \text{SEQ}(\gamma) \Rightarrow M(z) = \prod_{\gamma \in \mathcal{A}} (1 - z^{|\gamma|})^{-1}$$

Un algorithme intuitif : $\Gamma M(x)$

```
for  $\gamma \in \mathcal{A}$  do  
   $\mu \leftarrow \text{Geom}(x^{|\gamma|})$     $\mu$  est la multiplicité de  $\gamma$  dans le MSET  
  Ajouter  $\mu$  fois  $\gamma$  au MSET  
end for
```

difficultés :

- Énumération exhaustive des objets de $\mathcal{A}$
- Terminaison de l'algorithme

Un générateur de partitions

Definition

Une **partition de l'entier n** est une décomposition de n sous forme d'une somme d'entiers non nuls. On peut donc décrire la classe des partitions d'entier comme un multi-ensemble d'entiers non nuls :

$$\mathcal{P} = \text{MSET}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

solutions :

- On **énumère** les entiers facilement
- On **calcule** la **distribution du plus grand sommant** dans le modèle de Boltzmann

Tirage du plus grand sommant

On veut calculer la **distribution de la variable aléatoire K qui donne la taille du plus grand sommant** :

- On connaît :

$$\mathbb{P}_x(K = k) = \frac{P^{(k)}(x)}{P(x)}$$

où $P^{(k)}(x)$ est la série génératrice des partitions dont le plus grand sommant est k :

$$P^{(k)}(x) = \frac{x^k}{1 - x^k} \prod_{i < k} \frac{1}{1 - x^i}$$

- On cherche j , pour U aléatoire dans $[0, 1]$, tel que :

$$\sum_{k=1}^{j-1} \mathbb{P}_x(K = k) \leq U \leq \sum_{k=1}^j \mathbb{P}_x(K = k)$$

Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

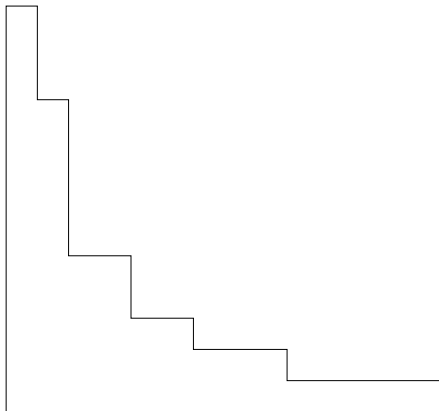
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$

$\mu_\gamma = 1;$

$\gamma = 12, 11;$

$\mu_\gamma = 0;$

$\gamma = 10;$

$\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$

$\mu_\gamma = 0;$

$\gamma = 5;$

$\mu_\gamma = 2;$

$\gamma = 4;$

$\mu_\gamma = 0;$

$\gamma = 3;$

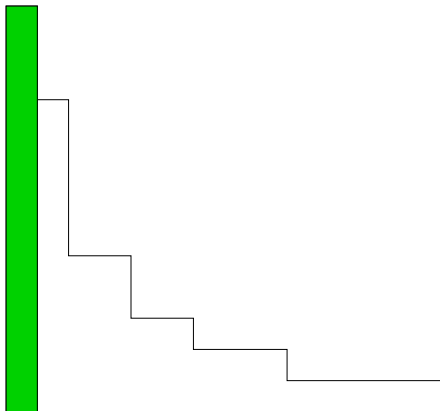
$\mu_\gamma = 2;$

$\gamma = 2;$

$\mu_\gamma = 3;$

$\gamma = 1;$

$\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

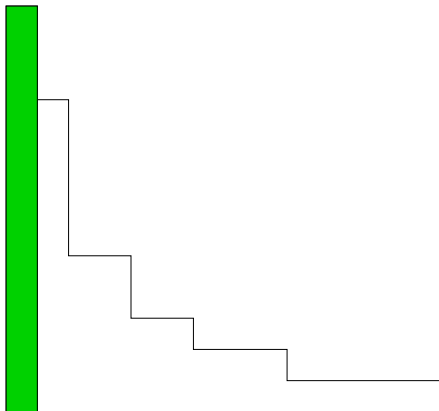
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

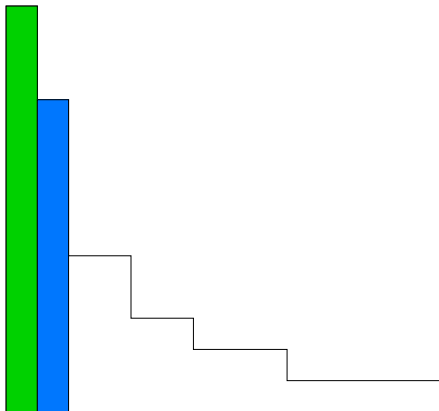
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13; \quad \mu_\gamma = 1;$

$\gamma = 12, 11; \quad \mu_\gamma = 0;$

$\gamma = 10; \quad \mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6; \quad \mu_\gamma = 0;$

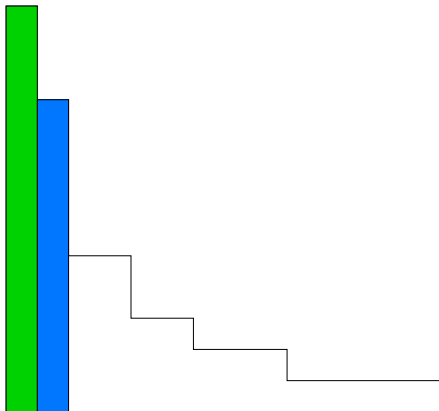
$\gamma = 5; \quad \mu_\gamma = 2;$

$\gamma = 4; \quad \mu_\gamma = 0;$

$\gamma = 3; \quad \mu_\gamma = 2;$

$\gamma = 2; \quad \mu_\gamma = 3;$

$\gamma = 1; \quad \mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

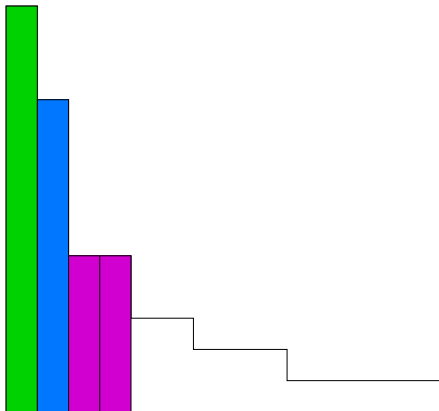
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

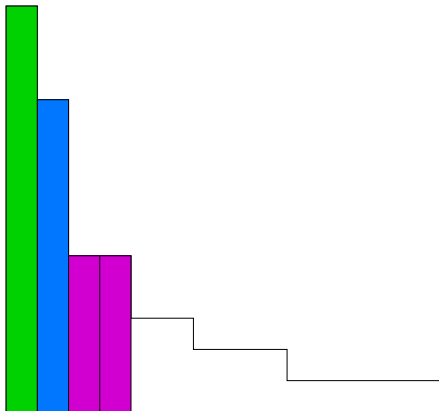
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

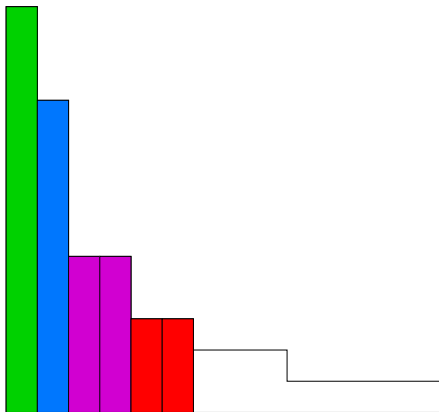
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

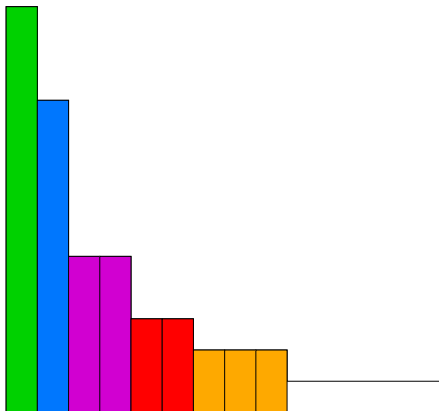
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$



Génération d'une partition

$k = 13;$

$\gamma = 13;$ $\mu_\gamma = 1;$

$\gamma = 12, 11;$ $\mu_\gamma = 0;$

$\gamma = 10;$ $\mu_\gamma = 1;$

$\gamma = 9, 8, 7, 6;$ $\mu_\gamma = 0;$

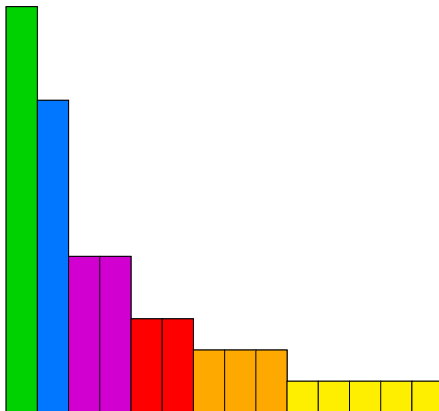
$\gamma = 5;$ $\mu_\gamma = 2;$

$\gamma = 4;$ $\mu_\gamma = 0;$

$\gamma = 3;$ $\mu_\gamma = 2;$

$\gamma = 2;$ $\mu_\gamma = 3;$

$\gamma = 1;$ $\mu_\gamma = 5;$

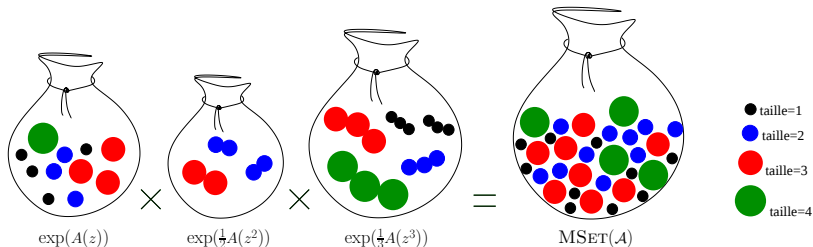


Algorithme pour un MSet quelconque (1)

$$M(z) = \exp \left(\sum_{k=1}^{\infty} \frac{1}{k} A(z^k) \right) = \prod_{k=1}^{\infty} \exp \left(\frac{1}{k} A(z^k) \right)$$

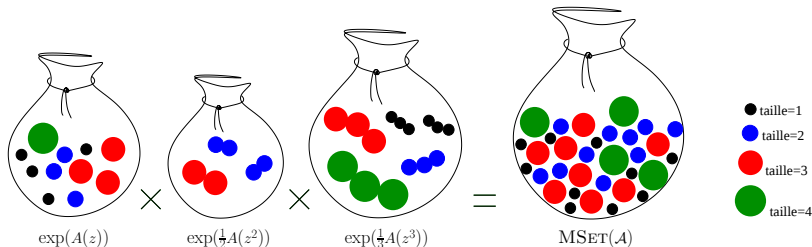
Algorithme pour un MSet quelconque (1)

$$M(z) = \exp \left(\sum_{k=1}^{\infty} \frac{1}{k} A(z^k) \right) = \prod_{k=1}^{\infty} \exp \left(\frac{1}{k} A(z^k) \right)$$



Algorithme pour un MSet quelconque (1)

$$M(z) = \exp \left(\sum_{k=1}^{\infty} \frac{1}{k} A(z^k) \right) = \prod_{k=1}^{\infty} \exp \left(\frac{1}{k} A(z^k) \right)$$



Lemme

Pour $\lambda > 0$, $\text{Geom}(\lambda) \simeq \text{Pois}(\lambda) + 2 \text{Pois}(\frac{\lambda^2}{2}) + \dots$

Algorithme pour un MSet quelconque (2)

$\Gamma MSet[\mathcal{A}](x)$

for $\gamma \in \mathcal{A}$ **do**

for $i \geq 1$ **do**

$\mu \leftarrow i \text{ Pois}(\frac{x^{i|\gamma|}}{i})$

 Ajouter μ fois γ au MSET

On inverse l'ordre des boucles !

$\Gamma MSet[\mathcal{A}](x)$

for $i \geq 1$ **do**

On peut tirer le plus grand indice

for $\gamma \in \mathcal{A}$ **do**

$\mu \leftarrow i \text{ Pois}(\frac{x^{i|\gamma|}}{i})$

 Ajouter μ fois γ au MSET

Remarque : $\forall i$, la multiplicité d'un $\gamma \in \mathcal{A}$ est un multiple de i

Algorithme pour un MSet quelconque (2)

$\Gamma MSet[\mathcal{A}](x)$

for $\gamma \in \mathcal{A}$ do

 for $i \geq 1$ do

$\mu \leftarrow i \text{ Pois}(\frac{x^{i|\gamma|}}{i})$

 Ajouter μ fois γ au MSET

On inverse l'ordre des boucles !

$\Gamma MSet[\mathcal{A}](x)$

for $i \geq 1$ do

On peut tirer le plus grand indice

 for $\gamma \in \mathcal{A}$ do

$\mu \leftarrow i \text{ Pois}(\frac{x^{i|\gamma|}}{i})$

 Ajouter μ fois γ au MSET

Remarque : $\forall i$, la multiplicité d'un $\gamma \in \mathcal{A}$ est un multiple de i

Algorithme pour un MSet quelconque (2)

$\Gamma MSet[\mathcal{A}](x)$

for $\gamma \in \mathcal{A}$ do

 for $i \geq 1$ do

$\mu \leftarrow i \text{ Pois}(\frac{x^{i|\gamma|}}{i})$

 Ajouter μ fois γ au MSET

On inverse l'ordre des boucles !

$\Gamma MSet[\mathcal{A}](x)$

for $i \geq 1$ do

On peut tirer le plus grand indice

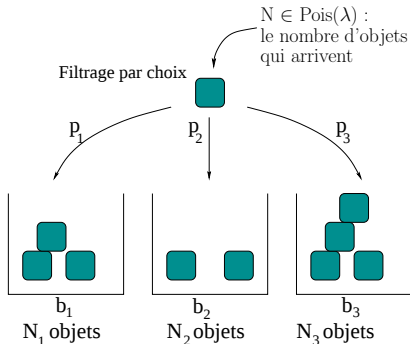
 for $\gamma \in \mathcal{A}$ do

$\mu \leftarrow i \text{ Pois}(\frac{x^{i|\gamma|}}{i})$

 Ajouter μ fois γ au MSET

Remarque : $\forall i$, la multiplicité d'un $\gamma \in \mathcal{A}$ est un multiple de i

Distributivité de la loi de Poisson par filtrage



pour $k = 1, 2, 3$ $N_k \in \text{Pois}(\lambda p_k)$
et les N_k sont indépendants.

Pour tirer :
des N_k indépendants tels
que $N_k \in \text{Pois}(\lambda_k)$
il suffit de :

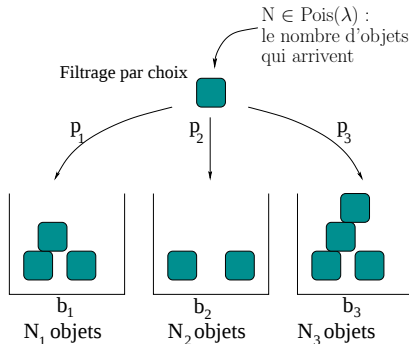
- tirer

$$N \in \text{Pois} \left(\sum_k \lambda_k \right)$$

- filtrer, i.e.,
choisir une boîte b_k
avec probabilité :

$$p_k = \frac{\lambda_k}{\sum_k \lambda_k}$$

Distributivité de la loi de Poisson par filtrage



pour $k = 1, 2, 3$ $N_k \in \text{Pois}(\lambda p_k)$
et les N_k sont indépendants.

Pour tirer :
des N_k indépendants tels
que $N_k \in \text{Pois}(\lambda_k)$
il suffit de :

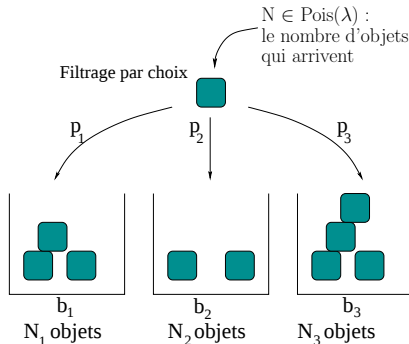
- tirer

$$N \in \text{Pois} \left(\sum_k \lambda_k \right)$$

- filtrer, i.e.,
choisir une boîte b_k
avec probabilité :

$$p_k = \frac{\lambda_k}{\sum_k \lambda_k}$$

Distributivité de la loi de Poisson par filtrage



pour $k = 1, 2, 3$ $N_k \in \text{Pois}(\lambda p_k)$
et les N_k sont indépendants.

Pour tirer :
des N_k indépendants tels
que $N_k \in \text{Pois}(\lambda_k)$
il suffit de :

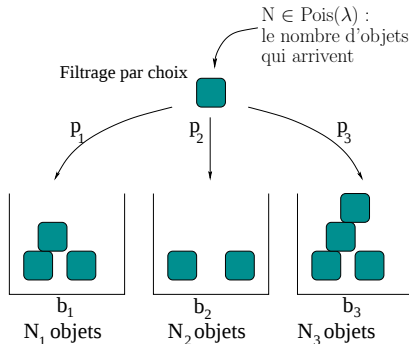
- tirer

$$N \in \text{Pois} \left(\sum_k \lambda_k \right)$$

- filtrer, i.e.,
choisir une boîte b_k
avec probabilité :

$$p_k = \frac{\lambda_k}{\sum_k \lambda_k}$$

Distributivité de la loi de Poisson par filtrage



pour $k = 1, 2, 3$ $N_k \in \text{Pois}(\lambda p_k)$
et les N_k sont indépendants.

Pour tirer :
des N_k indépendants tels
que $N_k \in \text{Pois}(\lambda_k)$
il suffit de :

- tirer

$$N \in \text{Pois} \left(\sum_k \lambda_k \right)$$

- filtrer, i.e.,
choisir une boîte b_k
avec probabilité :

$$p_k = \frac{\lambda_k}{\sum_k \lambda_k}$$

Utilisation du lemme de distributivité ($i = 1$)

Pour $\gamma \in \mathcal{A}$, on souhaite tirer des μ_γ indépendants avec :

$$\mu_\gamma \in \text{Pois}(x^{|\gamma|})$$

Or

$$\sum_{\gamma \in \mathcal{A}} x^{|\gamma|} = A(x)$$

Donc, il suffit de :

- tirer $N \in \text{Pois}(A(x))$,
- filtrer

→ choisir $\gamma \in \mathcal{A}$ avec une probabilité $p_\gamma = \frac{x^{|\gamma|}}{A(x)}$,
ce qui revient à faire un tirage de $\mathbb{P}\mathcal{A}(x)$.

Utilisation du lemme de distributivité ($i = 1$)

Pour $\gamma \in \mathcal{A}$, on souhaite tirer des μ_γ indépendants avec :

$$\mu_\gamma \in \text{Pois}(x^{|\gamma|})$$

Or

$$\sum_{\gamma \in \mathcal{A}} x^{|\gamma|} = A(x)$$

Donc, il suffit de :

- tirer $N \in \text{Pois}(A(x))$,
- filtrer

→ choisir $\gamma \in \mathcal{A}$ avec une probabilité $p_\gamma = \frac{x^{|\gamma|}}{A(x)}$,
ce qui revient à faire un tirage de $\mathbb{P}\mathcal{A}(x)$.

Utilisation du lemme de distributivité ($i = 1$)

Pour $\gamma \in \mathcal{A}$, on souhaite tirer des μ_γ indépendants avec :

$$\mu_\gamma \in \text{Pois}(x^{|\gamma|})$$

Or

$$\sum_{\gamma \in \mathcal{A}} x^{|\gamma|} = A(x)$$

Donc, il suffit de :

- tirer $N \in \text{Pois}(A(x))$,
- filtrer

→ choisir $\gamma \in \mathcal{A}$ avec une probabilité $p_\gamma = \frac{x^{|\gamma|}}{A(x)}$,
ce qui revient à faire un tirage de $\mathbb{P}\mathcal{A}(x)$.

Utilisation du lemme de distributivité ($i = 1$)

Pour $\gamma \in \mathcal{A}$, on souhaite tirer des μ_γ indépendants avec :

$$\mu_\gamma \in \text{Pois}(x^{|\gamma|})$$

Or

$$\sum_{\gamma \in \mathcal{A}} x^{|\gamma|} = A(x)$$

Donc, il suffit de :

- tirer $N \in \text{Pois}(A(x))$,
- filtrer

→ choisir $\gamma \in \mathcal{A}$ avec une probabilité $p_\gamma = \frac{x^{|\gamma|}}{A(x)}$,
ce qui revient à faire un tirage de $\Gamma A(x)$.

Utilisation du lemme de distributivité ($i = 1$)

Pour $\gamma \in \mathcal{A}$, on souhaite tirer des μ_γ indépendants avec :

$$\mu_\gamma \in \text{Pois}(x^{|\gamma|})$$

Or

$$\sum_{\gamma \in \mathcal{A}} x^{|\gamma|} = A(x)$$

Donc, il suffit de :

- tirer $N \in \text{Pois}(A(x))$,

- filtrer

→ choisir $\gamma \in \mathcal{A}$ avec une probabilité $p_\gamma = \frac{x^{|\gamma|}}{A(x)}$,
ce qui revient à faire un tirage de $\Gamma A(x)$.

Utilisation du lemme de distributivité ($i = 1$)

Pour $\gamma \in \mathcal{A}$, on souhaite tirer des μ_γ indépendants avec :

$$\mu_\gamma \in \text{Pois}(x^{|\gamma|})$$

Or

$$\sum_{\gamma \in \mathcal{A}} x^{|\gamma|} = A(x)$$

Donc, il suffit de :

- tirer $N \in \text{Pois}(A(x))$,
- filtrer
→ choisir $\gamma \in \mathcal{A}$ avec une probabilité $p_\gamma = \frac{x^{|\gamma|}}{A(x)}$,
ce qui revient à faire un tirage de $\Gamma A(x)$.

Algorithme pour un MSet quelconque (3)

Algorithme $\Gamma MSet[\mathcal{A}](x)$

```

 $k \leftarrow \text{IndiceMax}(x);$ 
for  $i$  from 1 to  $k - 1$  do {
     $p \leftarrow \text{Pois}(\frac{1}{i}A(x^i));$ 
    repeat  $p$  {
         $\gamma \leftarrow \Gamma A(x^i);$ 
        Ajouter  $i$  fois  $\gamma$  au MSET
    }
}

 $p \leftarrow \text{Pois}_{\geq 1}(\frac{1}{k}A(x^k));$ 
repeat  $p$  {
     $\gamma \leftarrow \Gamma A(x^k);$ 
    Ajouter  $k$  fois  $\gamma$  au MSET
}

```

Algorithme pour un MSet quelconque (3)

Algorithme $\Gamma MSet[\mathcal{A}](x)$

```
 $k \leftarrow \text{IndiceMax}(x);$ 
for  $i$  from 1 to  $k - 1$  do {
   $p \leftarrow \text{Pois}(\frac{1}{i}A(x^i));$ 
  repeat  $p$  {
     $\gamma \leftarrow \Gamma A(x^i);$ 
    Ajouter  $i$  fois  $\gamma$  au MSET
  }
}
 $p \leftarrow \text{Pois}_{\geq 1}(\frac{1}{k}A(x^k));$ 
repeat  $p$  {
   $\gamma \leftarrow \Gamma A(x^k);$ 
  Ajouter  $k$  fois  $\gamma$  au MSET
}
```

Algorithme pour un MSet quelconque (3)

Algorithme $\Gamma MSet[\mathcal{A}](x)$

```
 $k \leftarrow \text{IndiceMax}(x);$ 
for  $i$  from 1 to  $k - 1$  do {
   $p \leftarrow \text{Pois}(\frac{1}{i}A(x^i));$ 
  repeat  $p$  {
     $\gamma \leftarrow \Gamma A(x^i);$ 
    Ajouter  $i$  fois  $\gamma$  au MSET
  }
}
 $p \leftarrow \text{Pois}_{\geq 1}(\frac{1}{k}A(x^k));$ 
repeat  $p$  {
   $\gamma \leftarrow \Gamma A(x^k);$ 
  Ajouter  $k$  fois  $\gamma$  au MSET
}
```

Algorithme pour un MSet quelconque (3)

Algorithme $\Gamma MSet[\mathcal{A}](x)$

```

 $k \leftarrow \text{IndiceMax}(x);$ 
for  $i$  from 1 to  $k - 1$  do {
     $p \leftarrow \text{Pois}(\frac{1}{i}A(x^i));$ 
    repeat  $p$  {
         $\gamma \leftarrow \Gamma A(x^i);$ 
        Ajouter  $i$  fois  $\gamma$  au MSET
    }
}
 $p \leftarrow \text{Pois}_{\geq 1}(\frac{1}{k}A(x^k));$ 
repeat  $p$  {
     $\gamma \leftarrow \Gamma A(x^k);$ 
    Ajouter  $k$  fois  $\gamma$  au MSET
}
    
```

for $\gamma \in \mathcal{A}$ do
 $\mu \leftarrow i \text{Pois}(\frac{x^{i|\gamma|}}{i})$
 Ajouter μ fois γ au MSET
}

Algorithme pour un MSet quelconque (3)

Algorithme $\Gamma MSet[\mathcal{A}](x)$

```

 $k \leftarrow \text{IndiceMax}(x);$ 
for  $i$  from 1 to  $k - 1$  do {
     $p \leftarrow \text{Pois}(\frac{1}{i}A(x^i));$ 
    repeat  $p$  {
         $\gamma \leftarrow \Gamma A(x^i);$ 
        Ajouter  $i$  fois  $\gamma$  au MSET
    }
}
 $p \leftarrow \text{Pois}_{\geq 1}(\frac{1}{k}A(x^k));$ 
repeat  $p$  {
     $\gamma \leftarrow \Gamma A(x^k);$ 
    Ajouter  $k$  fois  $\gamma$  au MSET
}

```

for $\gamma \in \mathcal{A}$ do
 $\mu \leftarrow i \text{Pois}(\frac{x^{i|\gamma|}}{i})$
Ajouter μ fois γ au MSET
}

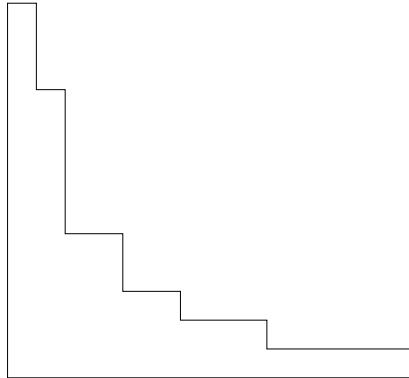
Génération d'une partition

$$k = 3;$$

$$p_1 = 5;$$

$$p_2 = 3;$$

$$p_3 = 1;$$



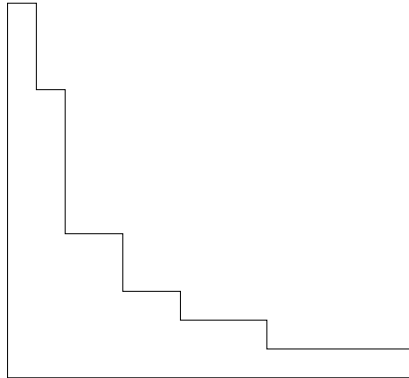
Génération d'une partition

$$k = 3;$$

$$p_1 = 5;$$

$$p_2 = 3;$$

$$p_3 = 1;$$



Génération d'une partition

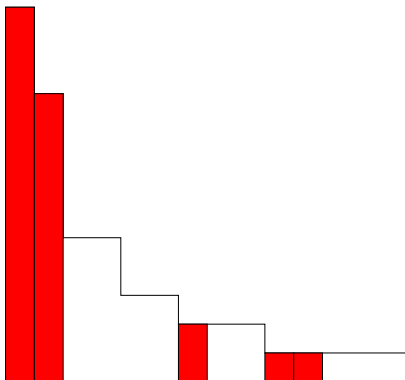
$$k = 3;$$

$$p_1 = 5;$$

$$p_2 = 3;$$

$$p_3 = 1;$$

$$\rightarrow 13, 10, 2, 1, 1$$



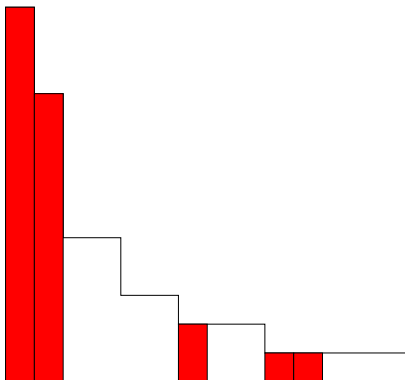
Génération d'une partition

$$k = 3;$$

$$p_1 = 5;$$

$$p_2 = 3;$$

$$p_3 = 1;$$



Génération d'une partition

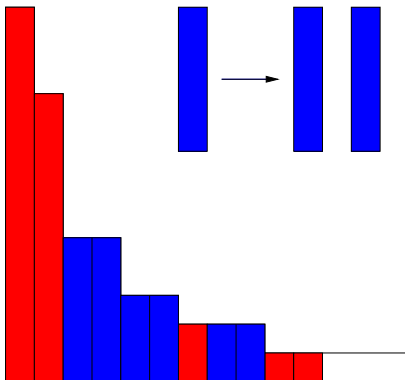
$$k = 3;$$

$$p_1 = 5;$$

$$p_2 = 3;$$

$$p_3 = 1;$$

$$\rightarrow 5, 3, 2$$



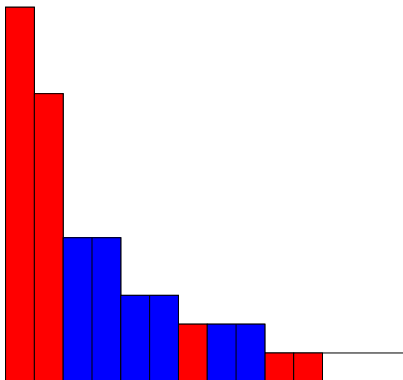
Génération d'une partition

$$k = 3;$$

$$p_1 = 5;$$

$$p_2 = 3;$$

$$p_3 = 1;$$



Génération d'une partition

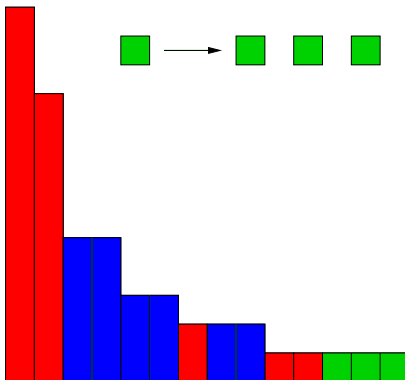
$k = 3;$

$p_1 = 5;$

$p_2 = 3;$

$p_3 = 1;$

$\rightarrow 1$



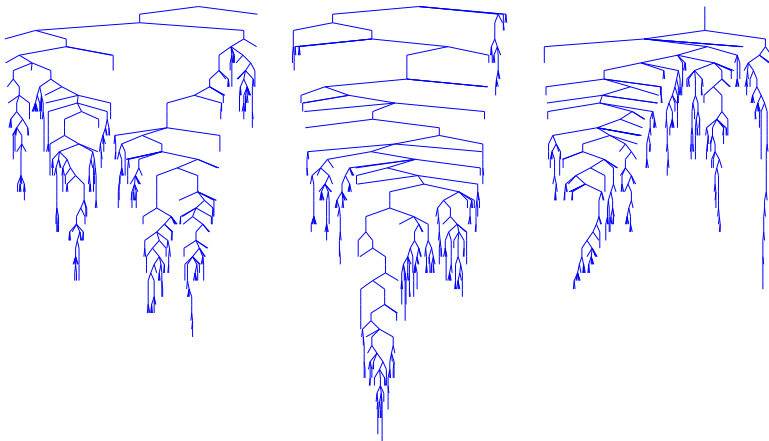
Résultat

Théorème

Si on possède un générateur de Boltzmann $\Gamma A(x)$ pour $\mathcal{A}$, alors l'algorithme $\Gamma MSet[\mathcal{A}](x)$ fournit un générateur de Boltzmann pour $MSET(\mathcal{A})$.

Des arbres généraux

$$\mathcal{T} = \mathcal{Z} \times \text{MSET}(\mathcal{T})$$



1 Le modèle de Boltzmann

2 Constructions propres au cas non étiqueté

- Les multi-ensembles
- Les ensembles sans répétitions
- Les cycles
- Restrictions de cardinalité

3 Complexité

Du MSet au PSet

Principe : On utilise la décomposition non ambiguë suivante :

$$\begin{aligned} \text{MSET}(\mathcal{A}) &= \text{PSET}(\mathcal{A}) \times \text{MSET}(\mathcal{A}^{(2)}) \\ \prod_{\gamma} \frac{1}{1-z^{|\gamma|}} &= \prod_{\gamma} \frac{1+z^{|\gamma|}}{1-z^{2|\gamma|}} = \prod_{\gamma} (1+z^{|\gamma|}) \prod_{\gamma} \frac{1}{1-z^{2|\gamma|}} \end{aligned}$$

L'algorithme $\Gamma\text{PSet}[\mathcal{A}](x)$ pour générer un ensemble sans répétitions d'objets de la classe $\mathcal{A}$ consiste à :

- Générer un multi-ensemble d'objets de $\mathcal{A}$,
- Extraire l'ensemble sans répétitions induit :
 - en supprimant les objets de multiplicité paire,
 - en ne gardant qu'une seule occurrence des objets de multiplicité impaire.

Preuve

Lemme (de division)

$$\mathcal{H} = \mathcal{K} \times \mathcal{L}$$

La procédure consistant à *extraire la première composante* de l'objet engendré par $\Gamma H(x)$ est un générateur de Boltzmann pour $\mathcal{K}$

Démonstration.

$$\begin{aligned}\mathbb{P}(\langle \kappa, \lambda \rangle) &= \frac{x^{|\kappa|} x^{|\lambda|}}{H(x)} \\ \mathbb{P}(\kappa) &= \sum_{\lambda} \frac{x^{|\kappa|} x^{|\lambda|}}{H(x)} = \frac{x^{|\kappa|}}{K(x)}\end{aligned}$$



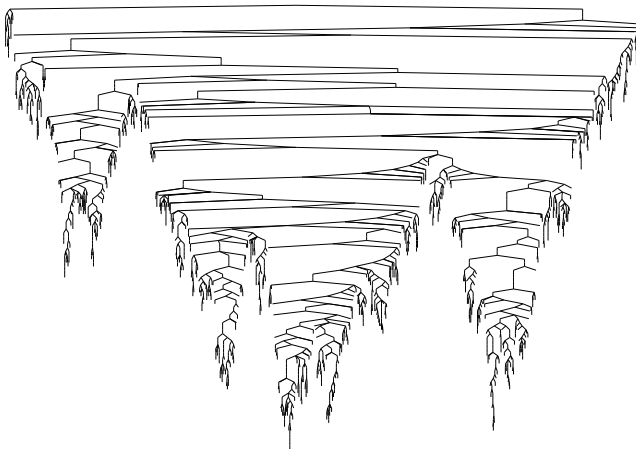
Résultat

Théorème

Si on possède un générateur de Boltzmann $\Gamma A(x)$ pour $\mathcal{A}$, alors l'algorithme $\Gamma PSet[\mathcal{A}](x)$ fournit un générateur de Boltzmann pour $PSET(\mathcal{A})$.

Des arbres sans jumeaux

$$\mathcal{T} = \mathcal{Z} \times \text{PSET}(\mathcal{T})$$



1 Le modèle de Boltzmann

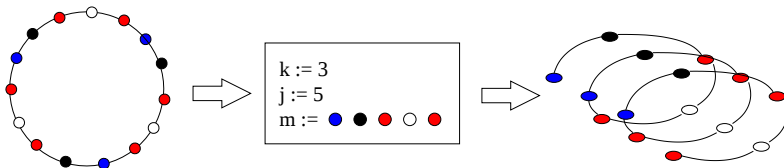
2 Constructions propres au cas non étiqueté

- Les multi-ensembles
- Les ensembles sans répétitions
- Les cycles
- Restrictions de cardinalité

3 Complexité

Des cycles

$$\mathcal{C} = \text{Cyc}(\mathcal{A}) \Rightarrow C(z) = \sum_{k \geq 1} \frac{\varphi(k)}{k} \log \frac{1}{1 - A(z^k)}$$



Algorithme

$\Gamma_{Cyc}[\mathcal{A}](x)$

$k \leftarrow \text{OrdreSymetrie}(x);$	# répétitions de m
$j \leftarrow \text{Loga}(A(x^k));$	longueur de m
$m \leftarrow \underbrace{\Gamma A(x^k), \dots, \Gamma A(x^k)}_{j \text{ fois}};$	motif
Return ($\underbrace{m, \dots, m}_{k \text{ fois}} $);	

Remarque : X suit une loi logarithmique de paramètre λ si :

$$\mathbb{P}(X = k) = \frac{1}{\log(1 - \lambda)^{-1}} \frac{\lambda^k}{k}$$

Preuve (1)

On tire K et L tels que :

$$\mathbb{P}(K = k) = \frac{1}{C(x)} \frac{\varphi(k)}{k} \log \frac{1}{1 - A(x^k)}$$

$$\mathbb{P}(J = j) = \frac{A(x^k)^j}{j} \left(\log \frac{1}{(1 - A(x^k))} \right)^{-1}$$

$$\text{i.e. } \mathbb{P}(K = k, J = j) = \frac{\varphi(k)}{kj} \frac{A(x^k)^j}{C(x)}$$

Propriété

$$\sum_{k|m} \varphi(k) = m$$

Preuve (2)

Soit un cycle u^r de longueur l , de taille n , avec $u = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $v = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\mathbb{P}(\gamma) = s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{A(x^k)^j}{C(x)} \frac{x^{k|v_1|}}{A(x^k)} \cdots \frac{x^{k|v_j|}}{A(x^k)}$$

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\mathbb{P}(\gamma) = s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{A(x^k)^j}{C(x)} \frac{x^n}{A(x^k)^j}$$

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\mathbb{P}(\gamma) = s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{x^n}{C(x)}$$

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\begin{aligned} \mathbb{P}(\gamma) &= s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{x^n}{C(x)} \\ &= \frac{s}{l} \frac{x^n}{C(x)} \sum_{k|r} \varphi(k) \end{aligned}$$

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\begin{aligned} \mathbb{P}(\gamma) &= s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{x^n}{C(x)} \\ &= \frac{s}{l} \frac{x^n}{C(x)} \sum_{k|r} \varphi(k) \\ &= \frac{sr}{l} \frac{x^n}{C(x)} \end{aligned}$$

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\begin{aligned} \mathbb{P}(\gamma) &= s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{x^n}{C(x)} \\ &= \frac{s}{l} \frac{x^n}{C(x)} \sum_{k|r} \varphi(k) \\ &= \frac{sr}{l} \frac{x^n}{C(x)} \end{aligned}$$

Preuve (2)

Soit un cycle $\mathbf{u}^r$ de longueur l , de taille n , avec $\mathbf{u} = [u_1, \dots, u_s]$ primitif.

Il peut être engendré comme une séquence de k répétitions d'un cycle $\mathbf{v} = [v_1, \dots, v_j]$, avec $k|r$, avec une probabilité :

$$\begin{aligned} \mathbb{P}(\gamma) &= s \sum_{kj=l, k|r} \frac{\varphi(k)}{kj} \frac{x^n}{C(x)} \\ &= \frac{s}{l} \frac{x^n}{C(x)} \sum_{k|r} \varphi(k) \\ &= \frac{x^n}{C(x)} \end{aligned}$$

Résultat

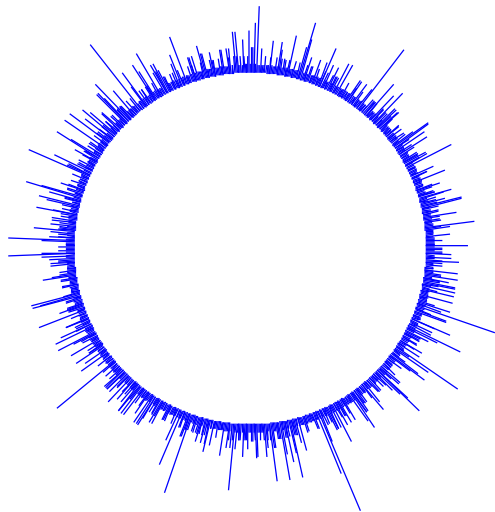
Théorème

Si on possède un générateur de Boltzmann $\Gamma A(x)$ pour $\mathcal{A}$, alors l'algorithme $\Gamma \text{Cyc}[\mathcal{A}](x)$ fournit un générateur de Boltzmann pour $\text{CYC}(\mathcal{A})$.

Des compositions circulaires

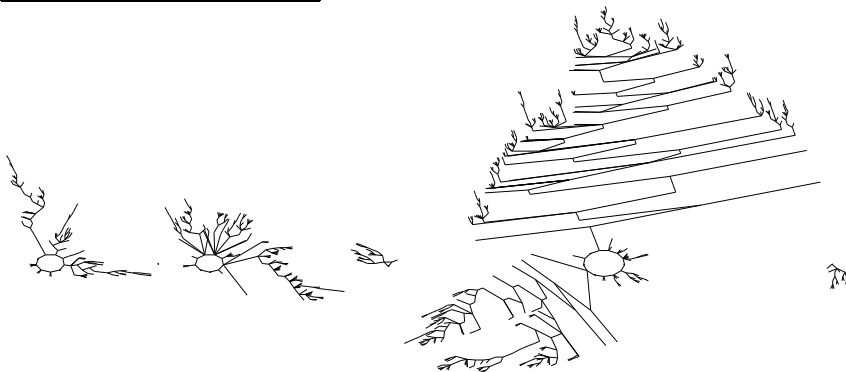
$$\mathcal{C} = \text{CYC}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z}))$$

$$C(z) = \sum_{k=1}^{\infty} \frac{\varphi(k)}{k} \log \frac{1}{1 - \frac{z^k}{1-z^k}}$$



Des graphes fonctionnels

$$\begin{aligned}\mathcal{M} &= \text{MSET}(\mathcal{C}) \\ \mathcal{C} &= \text{CYC}(\mathcal{G}) \\ \mathcal{G} &= \mathbb{Z} \times \text{MSET}(\mathcal{G})\end{aligned}$$



1 Le modèle de Boltzmann

2 Constructions propres au cas non étiqueté

- Les multi-ensembles
- Les ensembles sans répétitions
- Les cycles
- Restrictions de cardinalité

3 Complexité

$$\mathcal{M} = \text{MSet}(\mathcal{A})$$

- MSET_k

On utilise la formule :

$$M(z) = [u^k] \exp \left(uB(z) + \frac{u^2}{2}B(z^2) + \frac{u^3}{3}B(z^3) + \dots \right)$$

- $\text{MSET}_{<k}$

On se ramène au cas précédent :

$$\text{MSET}_{<k}(\mathcal{A}) = \bigcup_{i=1}^{k-1} \text{MSET}_i(\mathcal{A})$$

- $\text{MSET}_{>k}$

On procède par **rejet**.

$$\mathcal{C} = \text{Cyc}(\mathcal{A})$$

- Cyc_k

On utilise la formule :

$$C(z) = [u^k] \sum_{l=1}^{\infty} \frac{\varphi(l)}{l} \log \frac{1}{1 - u^l A(z^l)}$$

- $\text{Cyc}_{<k}$

On se ramène au cas précédent :

$$\text{Cyc}_{<k}(\mathcal{A}) = \bigcup_{i=1}^{k-1} \text{Cyc}_i(\mathcal{A})$$

- $\text{Cyc}_{>k}$

On procède par **rejet**.

$$\mathcal{P} = \text{PSet}(\mathcal{A})$$

- PSET_k

Pas de problème de symétrie :

- On engendre une séquence $\text{SEQ}_k(\mathcal{A})$
- On **rejete** s'il y a des répétitions.

- $\text{PSET}_{<k}$

On se ramène au cas précédent :

$$\text{PSET}_{<k}(\mathcal{A}) = \bigcup_{i=1}^{k-1} \text{PSET}_i(\mathcal{A})$$

- $\text{PSET}_{>k}$

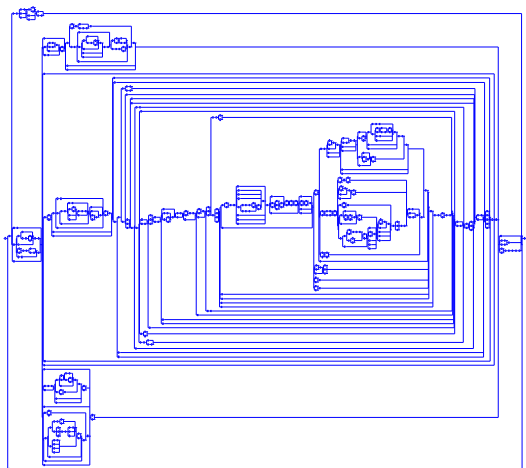
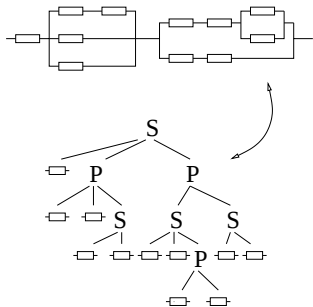
On procède par **rejet**.

Des circuits série-parallèle

$$\mathcal{C} = \mathcal{P} + \mathcal{S} + \mathcal{Z}$$

$$\mathcal{S} = \text{SEQ}_{\geq 2}(\mathcal{P} + \mathcal{Z})$$

$$\mathcal{P} = \text{MSET}_{\geq 2}(\mathcal{S} + \mathcal{Z})$$



Complexité

Coût des différents algorithmes (1)

Proposition

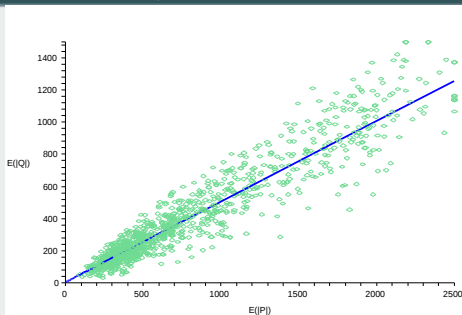
*Pour les constructions de **Multi-Ensemble** et de **Cycle**, les générateurs de Boltzmann proposés ont un **coût linéaire en la taille de l'objet engendré**, dans le pire des cas.*

Remarque : La complexité de l'oracle n'est pas prise en compte dans le coût des algorithmes.

Coût des différents algorithmes (2)

PSET : → **surcoût** dû au MSET

Exemple (Partitions en sommants distincts)



- *Rapport constant*
- Coût en $O(\sqrt{n} \log n)$

Cas général : à déterminer.

Conclusion (1)

- On dispose désormais d'algorithmes génériques de générateurs de Boltzmann pour les constructions : $+$, $\times$, SEQ, MSET, PSET et CYC, ainsi que pour leurs restrictions de cardinalité.
- On peut ainsi implémenter des générateurs de Boltzmann pour des dizaines de classes combinatoires non étiquetées.
- Lorsque la décomposition ne comporte pas de PSET, le coût est linéaire en la taille de l'objet engendré, dans le pire cas.

Conclusion (2)

Théorème

*Pour les classes combinatoires suivantes, pour des valeurs du paramètre x optimisées, et en utilisant une méthode par rejet, on peut obtenir des générateurs de Boltzmann en taille **approchée** (resp. **exacte**), ayant les complexités moyennes suivantes :*

<i>Partitions d'entier de taille n</i>	$O(\sqrt{n} \log n)$	$O(n)$
<i>Arbres non planaires de taille n</i>	$O(n)$	$O(n\sqrt{n})$
<i>Colliers, compositions circulaires de taille n</i>	$O(n)$	$O(n)$
<i>Mobles de taille n</i>	$O(n)$	$O(n\sqrt{n})$
<i>Graphes fonctionnels de taille n</i>	$O(n)$	$O(n\sqrt{n})$