

Génération aléatoire de structures combinatoires: méthode de Boltzmann effective

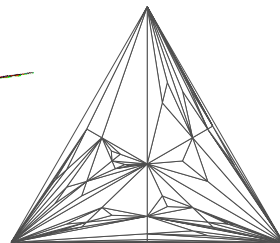
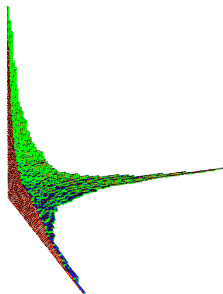
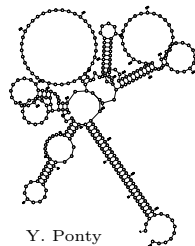
Carine Pivoteau

LIP6 - UPMC

3 décembre 2008

Motivations

- observation, expérimentation, étude statistique
 - conjectures,
 - distributions de paramètre,
 - propriétés limites,
 - validation : correction/adéquation de modèle.
- simulations, tests
 - validité des programmes,
 - robustesse des logiciels.
- applications
 - combinatoire,
 - bio-informatique,
 - test de logiciel,
 - physique statistique,
 - ...



Besoin de générateurs **efficaces** et **automatiques** !

Engendrer des circuits série-parallèle

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

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

$\sum_{n \geq 0} c_n z^n$  séries génératrices

$$S(z) = z + \frac{1}{1 - P(z)} - 1 - P(z)$$

$$P(z) = z + \exp\left(\sum_{k \geq 0} \frac{S(z^k)}{k}\right) - 1 - S(z)$$

coefficients 

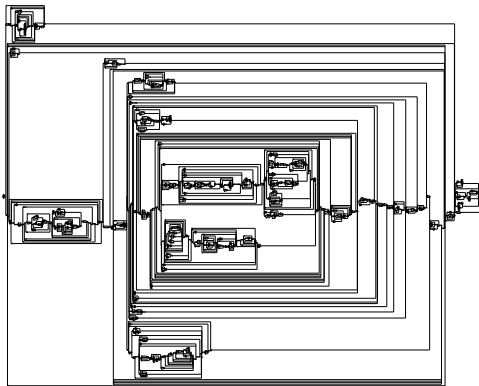
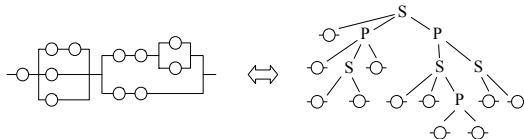
Génération
Récursive

[FIZiVC94]

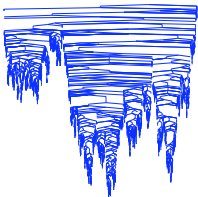
oracle 

Méthode de
Boltzmann

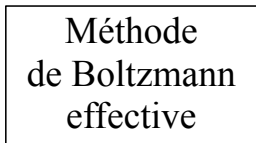
[DuFILoSc04]



Contributions de la thèse



**Générateurs
non étiquetés**

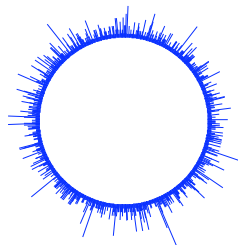
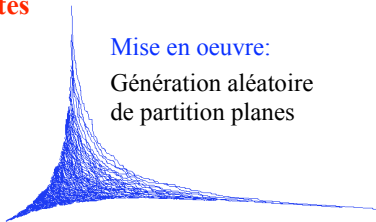


Algorithmes

Oracle

Mise en oeuvre:

Génération aléatoire
de partition planes



**Oracle numérique
Itération de Newton**

Bonus:

Calcul des coefficients
des séries génératrices

▷ validation théorique + expérimentale

Classes décomposables

[Flajolet, Sedgewick]

Méthode symbolique [FlSe08] :

- **spécification combinatoire** (grammaire récursive) :

$$\mathcal{Y} = \mathcal{H}(\mathcal{Z}, \mathcal{Y}) \equiv \begin{cases} \mathcal{Y}_1 = \mathcal{H}_1(\mathcal{Z}, \mathcal{Y}_1, \mathcal{Y}_2, \dots, \mathcal{Y}_m), \\ \mathcal{Y}_2 = \mathcal{H}_2(\mathcal{Z}, \mathcal{Y}_1, \mathcal{Y}_2, \dots, \mathcal{Y}_m), \\ \vdots \\ \mathcal{Y}_m = \mathcal{H}_m(\mathcal{Z}, \mathcal{Y}_1, \mathcal{Y}_2, \dots, \mathcal{Y}_m), \end{cases}$$

- **constructions** combinatoires usuelles (étiquetées ou non) :

$\mathcal{E}$, $\mathcal{Z}$, $+$, $\times$, séquence, cycle, ensemble

- **série génératrice** automatique pour toute classe combinatoire $\mathcal{C}$:

$$\text{ordinaire : } C(z) = \sum_{n \geq 0} c_n z^n \qquad \text{exponentielle : } \hat{C}(z) = \sum_{n \geq 0} c_n \frac{z^n}{n!}$$

où c_n est le nombre de structures de $\mathcal{C}$ de **taille** n .

Dictionnaire des séries génératrices

	construction	s.g. ordinaire (non étiquetée)	s.g. exponentielle (étiquetée)
$\mathcal{E}$ / atome	$1 / \mathcal{Z}$	$1 / z$	$1 / z$
Union	$\mathcal{A} + \mathcal{B}$	$A(z) + B(z)$	$A(z) + B(z)$
Produit	$\mathcal{A} \times \mathcal{B}$	$A(z) \times B(z)$	$A(z) \times B(z)$
Séquence	$\text{SEQ}(\mathcal{A})$	$\frac{1}{1 - A(z)}$	$\frac{1}{1 - A(z)}$
Ensemble	$\text{PSET}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k} A(z^k)\right)$	$\exp(A(z))$
Multi-ensemble	$\text{MSET}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty} \frac{1}{k} A(z^k)\right)$	—
Cycle	$\text{CYC}(\mathcal{A})$	$\sum_{k=1}^{\infty} \frac{\varphi(k)}{k} \log \frac{1}{1 - A(z^k)}$	$\log \frac{1}{1 - A(z)}$

1 Introduction

2 Générateurs de Boltzmann

- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

1 Introduction

2 Générateurs de Boltzmann

- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

Modèle de Boltzmann

La distribution des tailles des structures s'étale sur l'ensemble d'une classe combinatoire mais reste **uniforme** pour chaque sous-classe de structures de **même taille**.

- génération en **taille approchée** (paramètre de contrôle).
- de **très grandes structures** peuvent être engendrées.

Définition (Duchon, Flajolet, Louchard, Schaeffer – 2004)

Pour $x < \rho_{\mathcal{C}}$, le **modèle de Boltzmann** *non étiqueté* assigne à tout objet c de la classe $\mathcal{C}$ la probabilité :

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

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

2 structures de même taille $\rightarrow$ même probabilité.

Générateurs

[DuFlLoSc04]

	s.g. ordinaire (non étiquetée)	s.g. exponentielle (étiquetée)	
$1 / \mathcal{Z}$	$1 / z$	$1 / z$	
$\mathcal{A} + \mathcal{B}$	$A(z) + B(z)$	$A(z) + B(z)$	→ Bernoulli
$\mathcal{A} \times \mathcal{B}$	$A(z) \times B(z)$	$A(z) \times B(z)$	→ pas de choix
$\text{SEQ}(\mathcal{A})$	$\frac{1}{1 - A(z)}$	$\frac{1}{1 - A(z)}$	→ loi géométrique
$\text{PSET}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k} A(z^k)\right)$	$\exp(A(z))$	→ loi de Poisson
$\text{MSET}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty} \frac{1}{k} A(z^k)\right)$	—	
$\text{CYC}(\mathcal{A})$	$\sum_{k=1}^{\infty} \frac{\varphi(k)}{k} \log \frac{1}{1 - A(z^k)}$	$\log \frac{1}{1 - A(z)}$	→ loi logarithmique

Méthode de Boltzmann

[DuFLLoSc04]

Théorème (Générateurs étiquetés – DuFLLoSc04)

Pour toutes classe $\mathcal{C}$ **étiquetée** spécifiée (éventuellement récursivement) à partir des constructions suivantes :

$$\varepsilon, \mathcal{Z}, +, \times, \text{SEQ}, \text{SET}, \text{CYC},$$

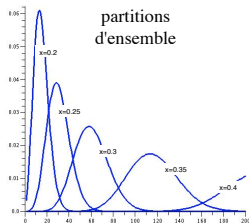
le **générateur de Boltzmann libre** $\Gamma_{\mathcal{C}}(x)$ opère en **temps linéaire** par rapport à la taille de la structure produite.

- Générateur **libre** : structures de **n'importe quelle taille** !
▷ **viser** la taille n : choisir x tel que l'espérance de la taille est n .

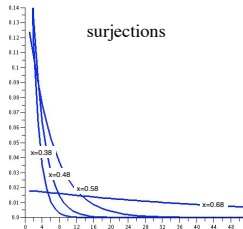
$$\mathbb{E}_x(N) = x \frac{C'(x)}{C(x)}$$

- Générateur en taille **approchée** ou **exacte** : utiliser du **rejet**.
▷ la distribution des tailles détermine le **coût du rejet**.

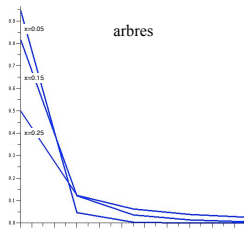
Génération en taille approchée/exacte



$$C(x) = e^{(e^x - 1)}$$



$$C(x) = \frac{1}{2 - e^x}$$



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

Théorème (Complexités – DuFllLoSc04)

type de distribution	concentrée	plate	piquée
taille approchée	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$o(n^2)$
taille exacte	$o(n^2)$	$o(n^2)$	—

▷ pointage
gen. singulier

1 Introduction

2 Générateurs de Boltzmann

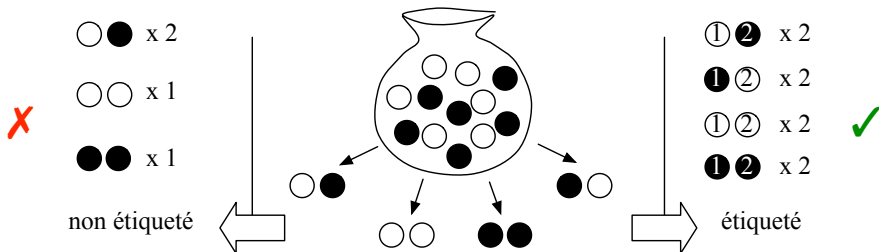
- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

Le problème des symétries



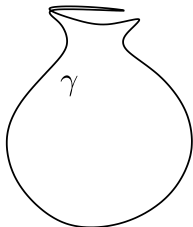
Opérateurs de Pólya

	construction	s.g. ordinaire (non étiquetée)	s.g. exponentielle (étiquetée)
$\mathcal{E}$ / atome	1 / $\mathcal{Z}$	1 / z	1 / z
Union	$\mathcal{A} + \mathcal{B}$	$A(z) + B(z)$	$A(z) + B(z)$
Produit	$\mathcal{A} \times \mathcal{B}$	$A(z) \times B(z)$	$A(z) \times B(z)$
Séquence	$\text{SEQ}(\mathcal{A})$	$\frac{1}{1 - A(z)}$	$\frac{1}{1 - A(z)}$
Ensemble	$\text{PSET}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty} \frac{(-1)^{k-1}}{k} A(z^k)\right)$	$\exp(A(z))$
Multi-ensemble	$\text{MSET}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty} \frac{1}{k} A(z^k)\right)$	—
Cycle	$\text{CYC}(\mathcal{A})$	$\sum_{k=1}^{\infty} \frac{\varphi(k)}{k} \log \frac{1}{1 - A(z^k)}$	$\log \frac{1}{1 - A(z)}$

L'exemple du multi-ensemble

 $\mathbf{MSet}(\mathcal{A})$

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

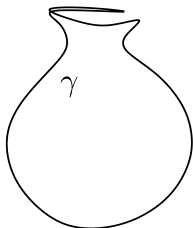


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le nombre d'éléments à engendrer – suivant une loi de Poisson de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis dupliquer i fois.

L'exemple du multi-ensemble

 $\mathbf{MSet}(\mathcal{A})$

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

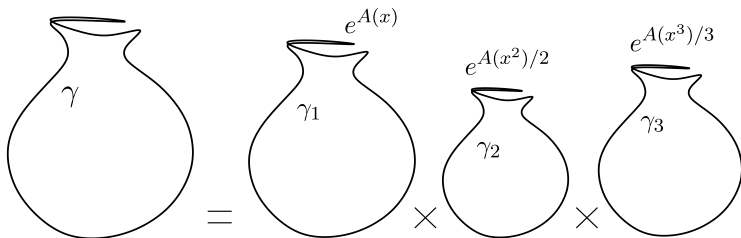


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le nombre d'éléments à engendrer – suivant une loi de Poisson de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis dupliquer i fois.

L'exemple du multi-ensemble

 $\mathbf{MSet}(\mathcal{A})$

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

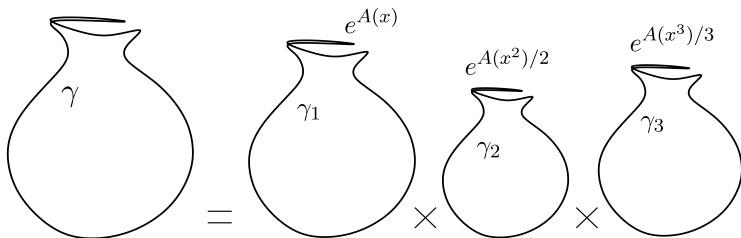


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le nombre d'éléments à engendrer – suivant une loi de Poisson de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis dupliquer i fois.

L'exemple du multi-ensemble

 $MSet(\mathcal{A})$

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

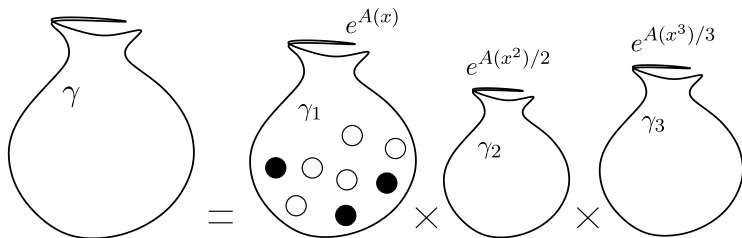


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le nombre d'éléments à engendrer – suivant une loi de Poisson de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis dupliquer i fois.

L'exemple du multi-ensemble

 $MSet(\mathcal{A})$

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

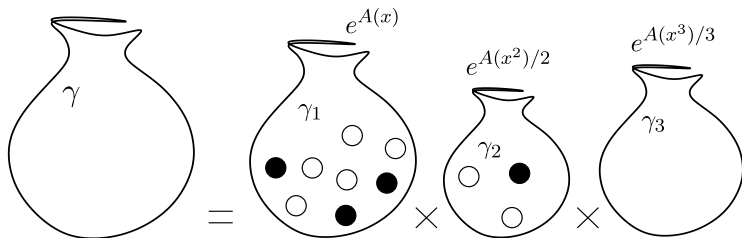


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le **nombre d'éléments à engendrer** – suivant une loi de **Poisson** de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis **dupliquer i fois**.

L'exemple du multi-ensemble

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

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

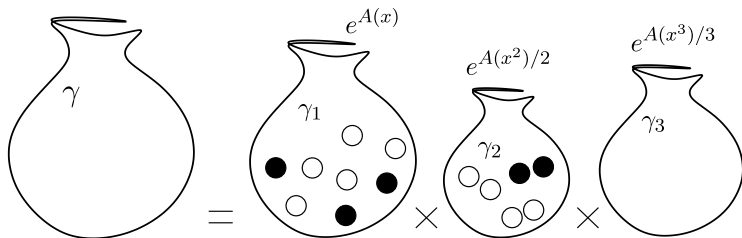


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le **nombre d'éléments à engendrer** – suivant une loi de **Poisson** de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis **dupliquer i fois**.

L'exemple du multi-ensemble

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

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

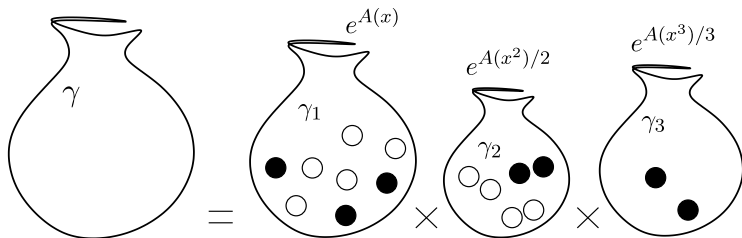


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le **nombre d'éléments à engendrer** – suivant une loi de **Poisson** de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis **dupliquer i fois**.

L'exemple du multi-ensemble

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

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

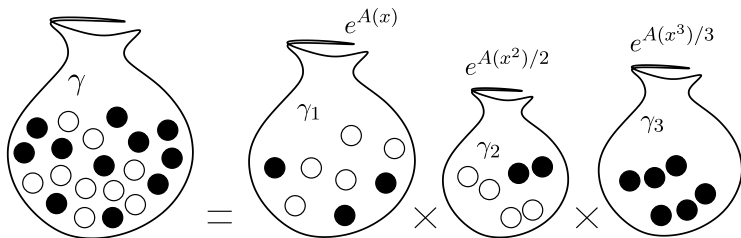


- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le **nombre d'éléments à engendrer** – suivant une loi de **Poisson** de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis **dupliquer i fois**.

L'exemple du multi-ensemble

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

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



- Tirer k tel que : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp(A(x^j)/j)$
- Pour chaque indice i jusqu'à k
 - Tirer p – le **nombre d'éléments à engendrer** – suivant une loi de **Poisson** de paramètre $A(x^i)/i$.
 - Faire p appels indépendants au générateur de la classe $\mathcal{A}$ avec le paramètre x^i , puis **dupliquer i fois**.

Générateurs de Boltzmann non étiquetés

Théorème (Générateurs non étiquetés – FlFuPi07)

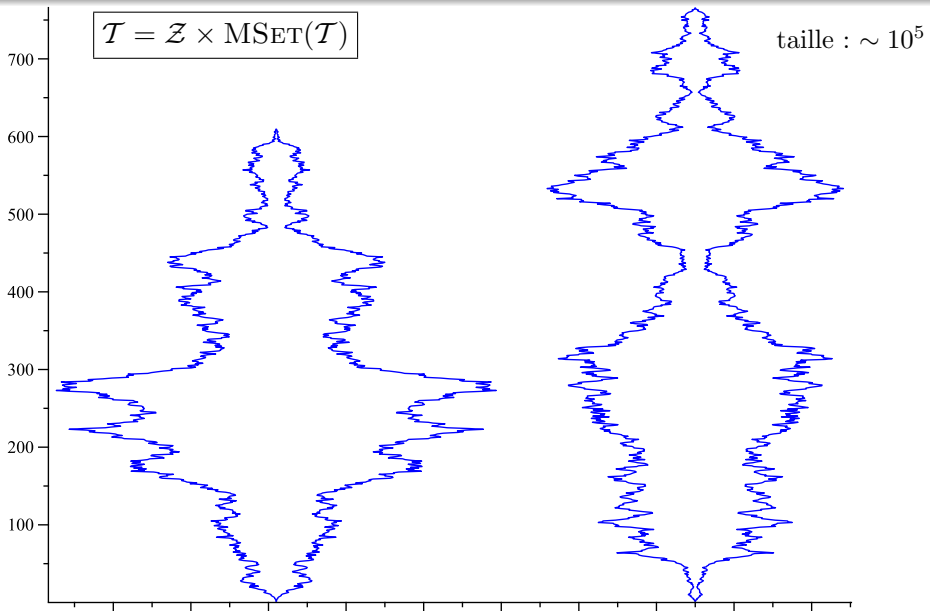
Pour toute classe $\mathcal{C}$ **non étiquetée** spécifiée (éventuellement récursivement) à partir des constructions suivantes :

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

le **générateur de Boltzmann libre** $\Gamma_{\mathcal{C}}(x)$ opère en **temps linéaire** par rapport à la taille de la structure produite.

[FlFuPi07] *Boltzmann sampling of unlabelled structures*, P. Flajolet, É. Fusy et C. Pivoteau

Arbres généraux non planaires enracinés



Résultats expérimentaux

Génération en taille approchée $\pm 10\%$ (temps en secondes)

	distribution concentrée	distribution plate		distribution piquée		
	Partitions d'entier	Compositions circulaires	Graphes fonctionnels	Arbres d'Otter	Arbres généraux	Circuits
taille	$O(\sqrt{n})$	$O(n)$	$O(n)$	$O(n)$	$O(n)$	$O(n)$
10^2	0.0020	0.0047	0.0084	0.024	0.018	0.047
10^3	0.039	0.051	0.080	0.20	0.17	0.43
10^4	0.078	0.57	0.82	2.2	1.6	4.7
10^5	0.17	6.2	8.4	22	18	43
10^6	0.47	60	86	240	170	530
10^7	1.5	570	—	—	—	—
10^8	5.2	—	—	—	—	—

jusqu'à 10^{10}

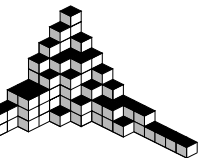
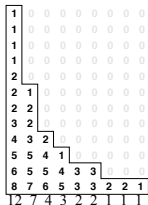
en Maple

en OCaml, sur un processeur Intel à 3.2 GHz avec 2 Go de mémoire

Application à la génération de partitions planes

$$\mathcal{P} = \text{MSET}(\mathcal{Z} \times \text{SEQ}(\mathcal{Z})^2)$$

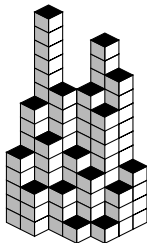
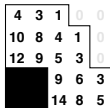
$$P(z) = \prod_{i,j \geq 0} \frac{1}{1 - z^{i+j+1}}$$



partition plane

$$S_D \prod_{(i,j) \in D} \text{SEQ}(\mathcal{Z}^{h(i,j)})$$

$$S_D(z) \prod_{(i,j) \in D} \frac{1}{1 - z^{h(i,j)}}$$



partition tordue (skew)

Théorème (Complexité moyenne)

Partitions planes :

- taille approchée : $O(n \ln(n)^3)$
- taille exacte : $O(n^{\frac{4}{3}})$

*Partitions encadrées ou tordues
(sur un domaine D) :*

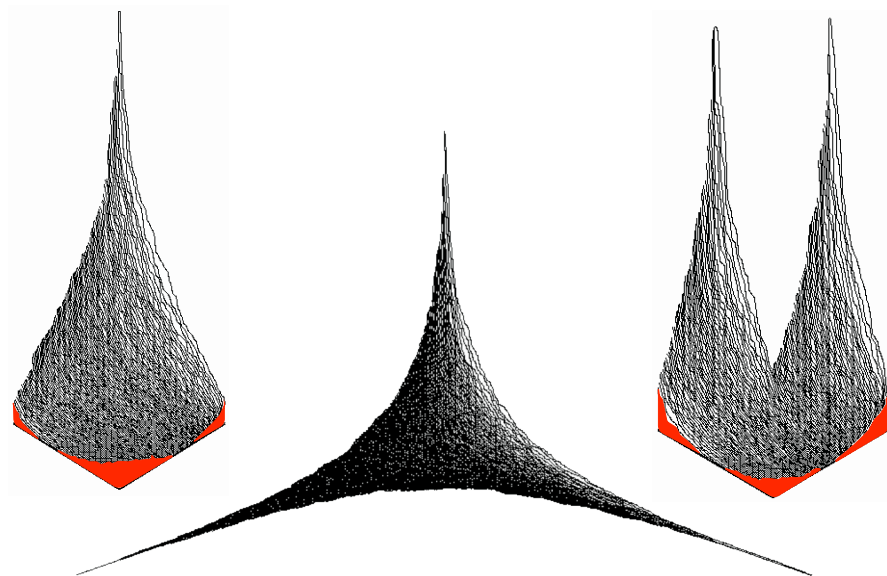
- taille approchée :
 $O(1)$ quand $n \rightarrow \infty$
- taille exacte :
bornée par $C|D|.n$,
avec C une constante.

[BoFuPi06]

Random sampling of plane partitions,

O. Bodini, E. Fusy et C. Pivoteau

Partitions géantes



1 Introduction

2 Générateurs de Boltzmann

- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

Exemple

Spécification combinatoire d'un langage algébrique :

$$\mathcal{C}_0 = \mathcal{Z}\mathcal{C}_1\mathcal{C}_2\mathcal{C}_3(\mathcal{C}_1 + \mathcal{C}_2)$$

$$\mathcal{C}_1 = \mathcal{Z} + \mathcal{Z}\text{SEQ}(\mathcal{C}_1^2\mathcal{C}_3^2)$$

$$\mathcal{C}_2 = \mathcal{Z} + \mathcal{Z}^2\text{SEQ}(\mathcal{Z}\mathcal{C}_2^2\text{SEQ}(\mathcal{Z}))\text{SEQ}(\mathcal{C}_2)$$

$$\mathcal{C}_3 = \mathcal{Z} + \mathcal{Z}(3\mathcal{Z} + \mathcal{Z}^2 + \mathcal{Z}^2\mathcal{C}_1\mathcal{C}_3)\text{SEQ}(\mathcal{C}_1^2)$$

Le système de séries génératrices associé :

$$C_0(z) = zC_1(z)C_2(z)C_3(z)(C_1(z) + C_2(z))$$

$$C_1(z) = z + z/(1 - C_1(z)^2C_3(z)^2)$$

$$C_2(z) = z + z^2/((1 - zC_2(z)^2/(1 - z))(1 - C_2(z)))$$

$$C_3(z) = z + z(3z + z^2 + z^2C_1(z)C_3(z))/(1 - C_1^2(z))$$

Exemple

Spécification combinatoire d'un langage algébrique :

$$\mathcal{C}_0 = \mathcal{Z}\mathcal{C}_1\mathcal{C}_2\mathcal{C}_3(\mathcal{C}_1 + \mathcal{C}_2)$$

$$\mathcal{C}_1 = \mathcal{Z} + \mathcal{Z}\text{SEQ}(\mathcal{C}_1^2\mathcal{C}_3^2)$$

$$\mathcal{C}_2 = \mathcal{Z} + \mathcal{Z}^2\text{SEQ}(\mathcal{Z}\mathcal{C}_2^2\text{SEQ}(\mathcal{Z}))\text{SEQ}(\mathcal{C}_2)$$

$$\mathcal{C}_3 = \mathcal{Z} + \mathcal{Z}(3\mathcal{Z} + \mathcal{Z}^2 + \mathcal{Z}^2\mathcal{C}_1\mathcal{C}_3)\text{SEQ}(\mathcal{C}_1^2)$$

Le système de séries génératrices associé : avec $z = 0.27$

$$C_0(z) = zC_1(z)C_2(z)C_3(z)(C_1(z) + C_2(z))$$

$$C_1(z) = z + z/(1 - C_1(z)^2C_3(z)^2)$$

$$C_2(z) = z + z^2/((1 - zC_2(z)^2/(1 - z))(1 - C_2(z)))$$

$$C_3(z) = z + z(3z + z^2 + z^2C_1(z)C_3(z))/(1 - C_1^2(z))$$

Exemple

Spécification combinatoire d'un langage algébrique :

$$C_0 = ZC_1C_2C_3(C_1 + C_2)$$

$$C_1 = Z + Z\text{SEQ}(C_1^2C_3^2)$$

$$C_2 = Z + Z^2\text{SEQ}(ZC_2^2\text{SEQ}(Z))\text{SEQ}(C_2)$$

$$C_3 = Z + Z(3Z + Z^2 + Z^2C_1C_3)\text{SEQ}(C_1^2)$$

Le système de séries génératrices associé : avec $z = 0.27$

$$C_0 = 0.27C_1C_2C_3(C_1 + C_2)$$

$$C_1 = 0.27 + 0.27/(1 - C_1^2C_3^2)$$

$$C_2 = 0.27 + 0.0729/((1 - 0.2754211138C_2^2)(1 - C_2))$$

$$C_3 = 0.27 + 0.27(0.8829 + 0.0729C_1C_3)/(1 - C_1^2)$$

en Maple

$$\text{sys} := \left[\begin{array}{l} C0 = x \, C3 \, C1 \, C2 \, (C1 + C2), \, C3 = x + \frac{x \, (3 \, x + x^2 + C1 \, C3 \, x^2)}{1 - C1^2}, \\ C2 = x + \frac{x^2}{\left(1 - \frac{x \, C2^2}{1 - x}\right) (1 - C2)}, \, C1 = x + \frac{x}{1 - C1^2 \, C3^2} \end{array} \right]$$

solve(subs(x=0.27,sys));

{ C2 = 0.3988484105, C3 = 0.643292746, C0 = 0.03981177934, C1 = 0.5844488219 },
 { C2 = 0.3988484105, C0 = 0.08483583330, C1 = 0.7669225413, C3 = 0.881136046 },
 { C2 = 0.3988484105, C1 = 3.827601486, C3 = 0.25115016, C0 = 0.4375296896 },
 { C3 = 0.643292746, C1 = 0.5844488219, C2 = 0.8243443421, C0 = 0.1178894124 },
 { C1 = 0.7669225413, C3 = 0.881136046, C2 = 0.8243443421, C0 = 0.2393370429 },
 { C1 = 3.827601486, C3 = 0.25115016, C2 = 0.8243443421, C0 = 0.9953303530 },
 { C3 = 0.643292746, C1 = 0.5844488219, C2 = 1.702776766, C0 = 0.3953535314 },
 { C1 = 0.7669225413, C3 = 0.881136046, C2 = 1.702776766, C0 = 0.7672908697 },
 { C1 = 3.827601486, C3 = 0.25115016, C2 = 1.702776766, C0 = 2.444198592 }

1 Introduction

2 Générateurs de Boltzmann

- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

Principe

Calcul par itération

Oracle de Boltzmann :

itération **numérique** qui **converge**
vers l'unique solution pertinente

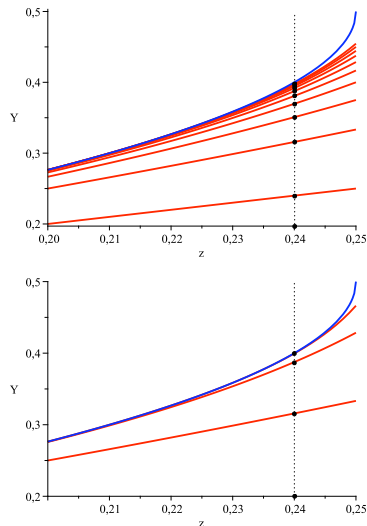


convergence de l'itération sur
les séries de **dénombrement**



convergence de l'itération pour
les systèmes d'équations **combinatoires**

Oracle efficace : itération de Newton



Arbres binaires : $B(z) = z + B(z)^2$

Théorème (Transfert de convergence – PiSaSo08)

Soit $\mathcal{Y} = \mathcal{H}(\mathcal{Z}, \mathcal{Y})$ un système tel que $\mathcal{H}(0, 0) = 0$ et $\partial\mathcal{H}/\partial\mathcal{Y}(0, 0)$ est nilpotente. Si l'itération combinatoire :

$$\mathcal{Y}^{[n+1]} = \mathcal{F}(\mathcal{Z}, \mathcal{Y}^{[n]}), \quad \text{avec } \mathcal{Y}^{[0]} = 0,$$

converge vers la famille d'espèces $\mathcal{Y}$, solution de $\mathcal{Y} = \mathcal{H}(\mathcal{Z}, \mathcal{Y})$, alors l'itération :

$$\mathbf{Y}^{[n+1]}(z) = \mathbf{F}(z, \mathbf{Y}^{[n]}(z)), \quad \text{avec } \mathbf{Y}^{[0]}(z) = 0,$$

converge vers le vecteur de séries génératrices $\mathbf{Y}(z)$ de $\mathcal{Y}$. De plus, si $\mathcal{F}$ est analytique, pour tout α tel que $|\alpha| < \rho$, l'itération :

$$\mathbf{y}^{[n+1]} = \mathbf{F}(\alpha, \mathbf{y}^{[n]}), \quad \text{avec } \mathbf{y}^{[0]} = 0,$$

converge vers le vecteur $\mathbf{Y}(\alpha)$ des valeurs des séries $\mathbf{Y}(z)$ au point α .

[PiSaSo08] *Boltzmann oracle for combinatorial systems*, C. Pivoteau, B. Salvy et M. Soria.

Validité de l'approche

Convergence combinatoire

- Théorème des espèces implicites [Joyal 81] : **existence** et **unicité** de la solution du système $\mathcal{Y} = \mathcal{H}(\mathcal{Z}, \mathcal{Y})$:
 - $\mathcal{H}(\mathbf{0}, \mathbf{0}) = \mathbf{0}$,
 - la **matrice jacobienne** $\frac{\partial \mathcal{H}}{\partial \mathcal{Y}}(\mathbf{0}, \mathbf{0})$ est **nilpotente**.
- Mesure de convergence : **contact** entre les classes combinatoires.

Séries génératrices

- Énumération des structures combinatoires.
- Contact $\Leftrightarrow$ **valuation** des séries.

Itération numérique

- Valeurs numériques $\Leftrightarrow$ évaluation des séries au point α .
- Analyticité de $\mathbf{H}$ en α et $\mathbf{y}^{[n]}$.

Exemple des arbres généraux planaires (Newton)

$$\mathcal{Y}^{[0]} = \emptyset$$

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

$$\mathcal{Y}^{[1]} = \boxed{\bullet} \boxed{\bullet} \begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array} \dots$$

$$\mathcal{Y}^{[2]} = \boxed{\bullet} \boxed{\bullet} \boxed{\begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array}} \boxed{\begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array}} \boxed{\begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array} \begin{array}{c} \bullet \\ \bullet \end{array}} \dots$$

$$Y^{[0]}(z) = \mathbf{0}$$

$$Y^{[1]}(z) = z + z^2 + z^3 + z^4 + z^5 + z^6 + z^7 + z^8 + z^9 + \dots$$

$$Y^{[2]}(z) = z + z^2 + \mathbf{2} z^3 + \mathbf{5} z^4 + \mathbf{14} z^5 + \mathbf{42} z^6 + 131 z^7 + 417 z^8 + 1341 z^9 \dots$$

$$Y^{[3]}(z) = z + z^2 + \mathbf{2} z^3 + \mathbf{5} z^4 + \mathbf{14} z^5 + \mathbf{42} z^6 + \mathbf{132} z^7 + \dots + \mathbf{742900} z^{14} + \dots$$

$$Y^{[4]}(z) = z + z^2 + \mathbf{2} z^3 + \mathbf{5} z^4 + \mathbf{14} z^5 + \mathbf{42} z^6 + \dots + \mathbf{1002242216651368} z^{30} + \dots$$

$$y^{[0]} = \mathbf{0}$$

pour $z = 0.2$

$$y^{[1]} = \mathbf{0.2}500000000000000000000000000000 \dots$$

$$y^{[2]} = \mathbf{0.27}586206896551724137931034482758 \dots$$

$$y^{[3]} = \mathbf{0.27639}296187683284457478005865104 \dots$$

$$y^{[4]} = \mathbf{0.2763932022}4997168102435963404908 \dots$$

$$y^{[5]} = \mathbf{0.27639320225002103035908263}104683 \dots$$

1 Introduction

2 Générateurs de Boltzmann

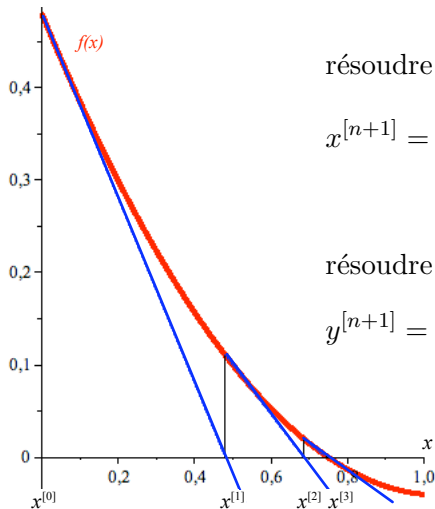
- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

Itération de Newton numérique “classique”



résoudre $f(x) = 0$:

$$x^{[n+1]} = x^{[n]} - \frac{f(x^{[n]})}{f'(x^{[n]})}$$

résoudre $y = f(y)$:

$$y^{[n+1]} = y^{[n]} + \frac{1}{1 - f'(y^{[n]})} (f(y^{[n]}) - y^{[n]})$$

Itération de Newton combinatoire

Pour une seule équation [Décoste, Labelle, Leroux 82] :

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \text{SEQ}\left(\frac{\partial \mathcal{H}}{\partial \mathbf{y}}(\mathcal{Z}, \mathbf{y}^{[n]})\right) \times \left(\mathcal{H}(\mathcal{Z}, \mathbf{y}^{[n]}) - \mathbf{y}^{[n]}\right), \quad \mathbf{y}^{[0]} = \mathbf{0}$$

▷ dérivée combinatoire

Pour un système :

$$\mathbf{y}^{[n+1]} = \mathbf{y}^{[n]} + \left(\mathbf{Id} - \frac{\partial \mathcal{H}}{\partial \mathbf{y}}(\mathcal{Z}, \mathbf{y}^{[n]})\right)^{-1} \left(\mathcal{H}(\mathcal{Z}, \mathbf{y}^{[n]}) - \mathbf{y}^{[n]}\right), \quad \mathbf{y}^{[0]} = \mathbf{0}$$

▷ matrice jacobienne

▷ éclosions combinatoires [Labelle 85]

Optimisation

Calcul de la matrice $\left(\mathbf{Id} - \frac{\partial \mathcal{H}}{\partial \mathbf{y}}(\mathcal{Z}, \mathbf{y}^{[n]})\right)^{-1}$ par itération de Newton

▷ itérations en parallèle.

Dérivée combinatoire

$\partial\mathcal{H}/\partial\mathcal{T}$: *dérivée* de $\mathcal{H}(\mathcal{Z}, \mathcal{Y}_1, \dots, \mathcal{Y}_m)$ par rapport à $\mathcal{T}$.

	notation		dérivée	
atome	$\mathcal{Z}$ ou $\mathcal{E}$		$\emptyset$	
$\mathcal{Y}_i$	$\mathcal{Y}_i = \mathcal{T}$	$\mathcal{Y}_i \neq \mathcal{T}$	$\mathcal{E}$	$\mathcal{Y}'_i$
union	$\mathcal{Y}_i + \mathcal{Y}_j$		$\mathcal{Y}_i + \mathcal{Y}'_i + \mathcal{Y}'_j$	
produit	$\mathcal{Y}_i \times \mathcal{Y}_j$		$\mathcal{Y}'_i \times \mathcal{Y}_j + \mathcal{Y}_i \times \mathcal{Y}'_j$	
séquence	$\text{SEQ}(\mathcal{Y}_i)$		$\text{SEQ}(\mathcal{Y}_i) \times \mathcal{Y}'_i \times \text{SEQ}(\mathcal{Y}_i)$	
cycle	$\text{CYC}(\mathcal{Y}_i)$		$\text{SEQ}(\mathcal{Y}_i) \times \mathcal{Y}'_i$	
ensemble	$\text{SET}(\mathcal{Y}_i)$		$\text{SET}(\mathcal{Y}_i) \times \mathcal{Y}'_i$	

Exemple : $\mathcal{H}(\mathcal{Z}, \mathcal{T}) = \mathcal{T}^2$, $\partial\mathcal{H}/\partial\mathcal{T}(\mathcal{Z}, \mathcal{T}) = 2\mathcal{T}$.

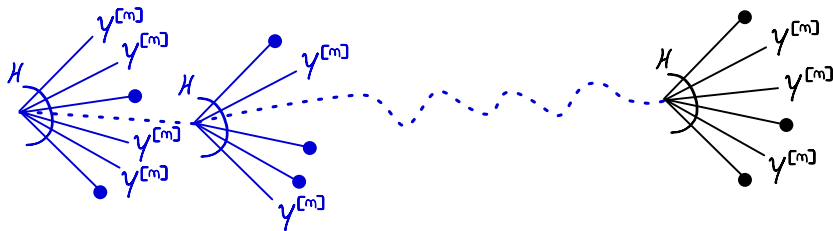
▷ Interprétation combinatoire du produit par une dérivée

$\partial\mathcal{H}/\partial\mathcal{Y}$: la *matrice Jacobienne* de $\mathcal{H}(\mathcal{Z}, \mathcal{Y})$ par rapport à $\mathcal{Y}$,
ses entrées sont les dérivées partielles $\partial\mathcal{H}_i/\partial\mathcal{Y}_j(\mathcal{Z}, \mathcal{Y})$.

Convergence de l'itération de Newton combinatoire

Interprétation combinatoire de l'itération de Newton
[Décoste, Labelle, Leroux 82]

$$\mathcal{Y}^{[n+1]} = \mathcal{Y}^{[n]} + \text{SEQ}\left(\frac{\partial \mathcal{H}}{\partial \mathcal{Y}}(\mathcal{Z}, \mathcal{Y}^{[n]})\right) \times \left(\mathcal{H}(\mathcal{Z}, \mathcal{Y}^{[n]}) - \mathcal{Y}^{[n]}\right), \quad \mathcal{Y}^{[0]} = 0$$



▷ convergence quadratique !

Itération de Newton pour les séries

$$\mathbf{Y}^{[k+1]}(z) = \mathbf{Y}^{[k]}(z) + \left(I - \frac{\partial \mathbf{H}}{\partial \mathbf{Y}}(z, \mathbf{Y}^{[k]}(z)) \right)^{-1} \left(\mathbf{H}(z, \mathbf{Y}^{[k]}(z)) - \mathbf{Y}^{[k]}(z) \right)$$

Algorithme efficace pour le calcul des coefficients !

Complexités arithmétique et binaire **quasi-optimales**

Théorème (Complexité arithmétique)

Coût du calcul des N premiers termes de $\mathbf{Y}(z)$: $O(M(N))$

- ▷ $M(N)$: coût de la multiplication de séries à l'ordre N .
- ▷ avec la FFT : $O(N \log N)$.
- ▷ amélioration de la complexité de la méthode récursive.

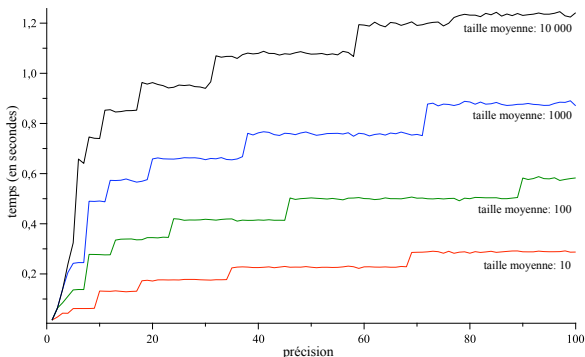
Oracle numérique

Théorème (Oracle de Boltzmann – PiSaSo08)

Soit $\mathcal{Y} = \mathcal{H}(\mathcal{Z}, \mathcal{Y})$ une spécification combinatoire (sans opérateur de Pólya). L'itération numérique suivante converge vers $\mathbf{Y}(\alpha)$:

$$\mathbf{y}^{[k+1]} = \mathbf{y}^{[k]} + \left(\mathbf{I} - \frac{\partial \mathbf{H}}{\partial \mathbf{Y}}(\alpha, \mathbf{y}^{[k]}) \right)^{-1} \left(\mathbf{H}(\alpha, \mathbf{y}^{[k]}) - \mathbf{y}^{[k]} \right), \quad \mathbf{y}^{[0]} = \mathbf{0}$$

Évolution du temps de calcul de l'oracle pour les circuits SP en fonction de la précision souhaitée, pour différentes valeurs de α .



Implantation

Prototype en Maple

(pour les spécifications sans opérateurs de Pólya) :

- tests sur des grammaires aléatoires,
- grammaires XML, $\sim 10^3$ équations [Darrasse 08],
- chemins dans des grands graphes (systèmes concurrents) [Oudinet 07],
- ...

# équations	4	10	50		100		500
# constructions/eqn	10	10	10	50	10	50	50
taille moy. cfc max.	2.47	3.42	7.95	18.62	10.93	67.18	339.1
temps (0.99 ρ)	0.05	0.11	0.17	0.47	0.23	7.29	61.73
temps (0.999999 ρ)	0.08	0.16	0.19	0.56	0.25	8.11	61.86
espérance moy. taille	$4.1 \cdot 10^{14}$	$1.4 \cdot 10^7$	$2.2 \cdot 10^5$	$1.0 \cdot 10^5$	$1.2 \cdot 10^6$	$5.0 \cdot 10^4$	$3.3 \cdot 10^4$

en secondes, avec Maple 11, sur un processeur Intel à 3.2 GHz avec 2 Go de mémoire.

1 Introduction

2 Générateurs de Boltzmann

- Méthode de Boltzmann
- Générateurs non étiquetés

3 Calcul de l'oracle

- Transfert de convergence
- Itérations de Newton

4 Conclusion

Conclusion et perspectives

- Méthode de Boltzmann effective !
- Implantation d'une bibliothèque de générateurs de Boltzmann génériques.
- Applications à grande échelle.

Perspectives :

- Accélération de convergence.
- Calcul des singularités
 - ▷ analyse asymptotique automatique.
- Réglage du paramètre de Boltzmann.