Boltzmann Sampling

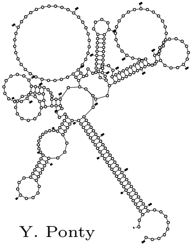
Carine Pivoteau

Université PARIS EST

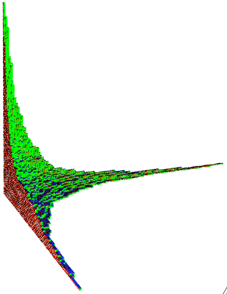
december 2010

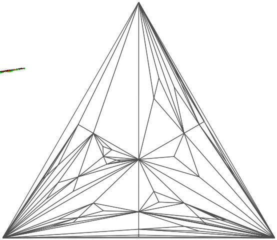
Motivations

- observation, experimentation, statistical study
  - conjectures,
  - parameters distribution,
  - limit properties,
  - validation : correctness/model adequacy.
- simulations, tests
  - programs validity,
  - software robustness.
- applications
  - combinatorics,
  - bio-informatics,
  - software testing,
  - statistical physics,
  - ...



Y. Ponty



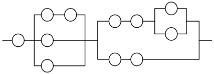


A. Darrasse

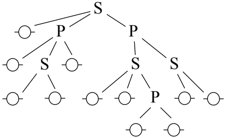
Need of efficient and automatic samplers !

Generate series-parallel circuits

$$\mathcal{S} = \mathcal{Z} + \textsc{Seq}_{\geq 2}(\mathcal{P})$$
$$\mathcal{P} = \mathcal{Z} + \textsc{MSet}_{\geq 2}(\mathcal{S})$$



$\Longleftrightarrow$



$\sum_{n\geq 0} c_n z^n$ generating series

$$\boxed{\begin{aligned} S(z) &= z + \frac{1}{1-P(z)} - 1 - P(z) \\ P(z) &= z + \exp\Big(\sum_{k\geq 0} \frac{S(z^k)}{k}\Big) - 1 - S(z) \end{aligned}}$$

coefficients

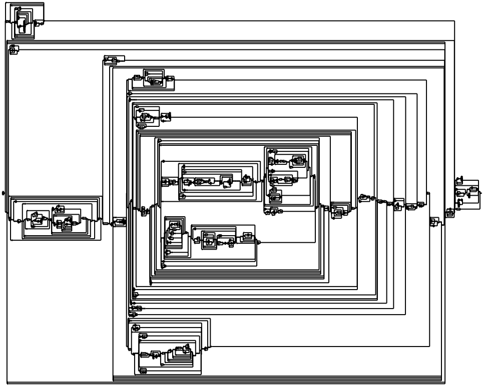
oracle

Recursive Sampling

[FlZiVC94]

Boltzmann Method

[DuFlLoSc04]



Decomposable classes [Flajolet, Sedgewick]

Symbolic method [FlSe08] :

- Combinatorial specification (recursive grammar) :

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

- usual combinatorial constructions (labelled or not) :

$\mathcal{E}$, $\mathcal{Z}$, $+$, $\times$, sequence, cycle, set

- automatic generating series for any combinatorial class $\mathcal{C}$:

ordinary : $C(z) = \sum_{n \geq 0} c_n z^n$ exponential : $\hat{C}(z) = \sum_{n \geq 0} c_n \frac{z^n}{n!}$

where c_n is the number of structures of $\mathcal{C}$ that are of size n.

Generating series dictionary

	construction	ordinary g.s. (unlabelled)	exponential g.s. (labelled)
$\mathcal{E}$ / atom	1 / $\mathcal{Z}$	1 / z	1 / z
Union	$\mathcal{A}+\mathcal{B}$	$A(z)+B(z)$	$A(z)+B(z)$
Product	$\mathcal{A}\times\mathcal{B}$	$A(z)\times B(z)$	$A(z)\times B(z)$
Sequence	$\textsc{Seq}(\mathcal{A})$	$\dfrac{1}{1-A(z)}$	$\dfrac{1}{1-A(z)}$
Set	$\textsc{PSet}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty}\frac{(-1)^{k-1}}{k}A(z^k)\right)$	$\exp\big(A(z)\big)$
Multi-Set	$\textsc{MSet}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty}\frac{1}{k}A(z^k)\right)$	–
Cycle	$\textsc{Cyc}(\mathcal{A})$	$\sum_{k=1}^{\infty}\frac{\varphi(k)}{k}\log\frac{1}{1-A(z^k)}$	$\log\dfrac{1}{1-A(z)}$

The recursive method on an example

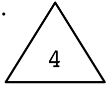
Rooted binary trees $\mathcal{T} = \mathcal{E} + \mathcal{Z} \times \mathcal{T}^2$

$1 + z + 2z^2 + 5z^3 + 14z^4 + \cdots$

The recursive method on an example

Rooted binary trees $\mathcal{T} = \mathcal{E} + \mathcal{Z} \times \mathcal{T}^2$

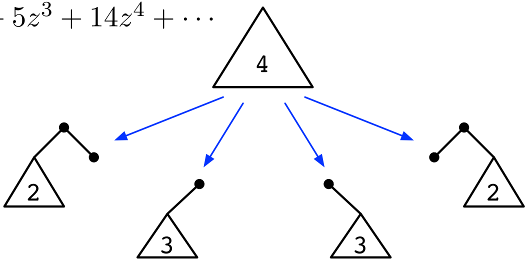
$1 + z + 2z^2 + 5z^3 + 14z^4 + \cdots$



The recursive method on an example

Rooted binary trees $\mathcal{T} = \mathcal{E} + \mathcal{Z} \times \mathcal{T}^2$

$1 + z + 2z^2 + 5z^3 + 14z^4 + \cdots$

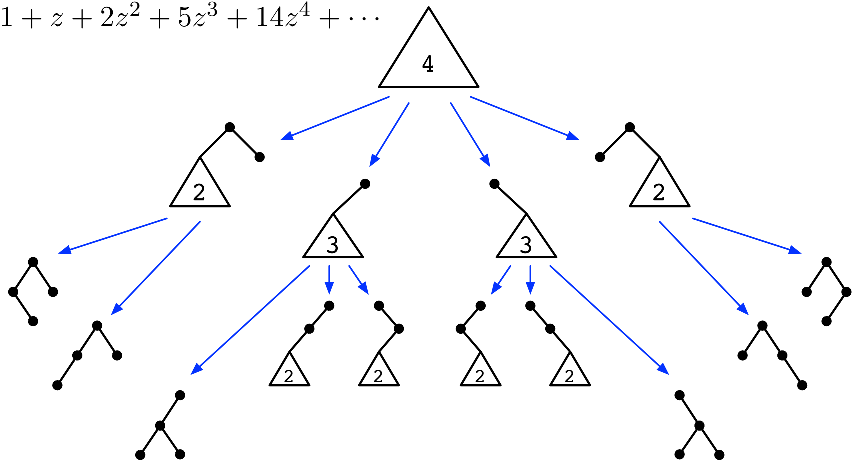


The recursive method on an example

Rooted binary trees

$$1 + z + 2z^2 + 5z^3 + 14z^4 + \cdots$$

$$\mathcal{T} = \mathcal{E} + \mathcal{Z} \times \mathcal{T}^2$$

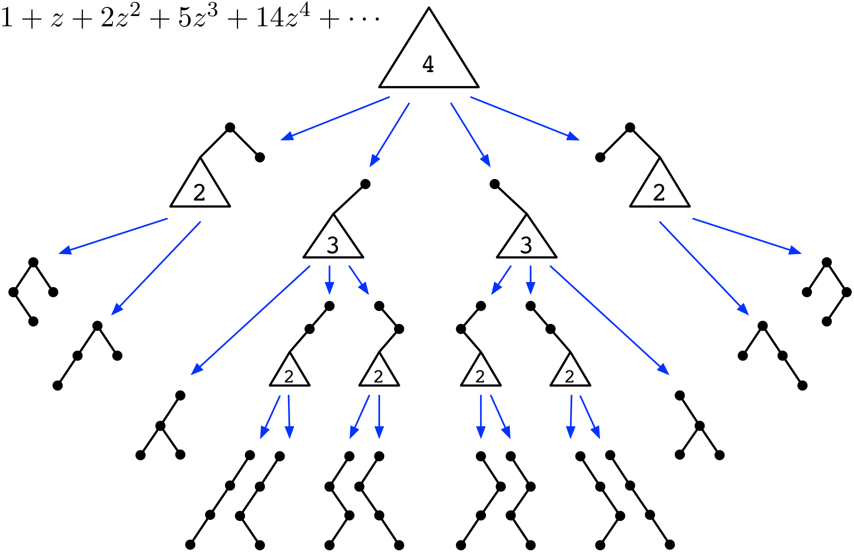


The recursive method on an example

Rooted binary trees

$1 + z + 2z^2 + 5z^3 + 14z^4 + \cdots$

$\mathcal{T} = \mathcal{E} + \mathcal{Z} \times \mathcal{T}^2$

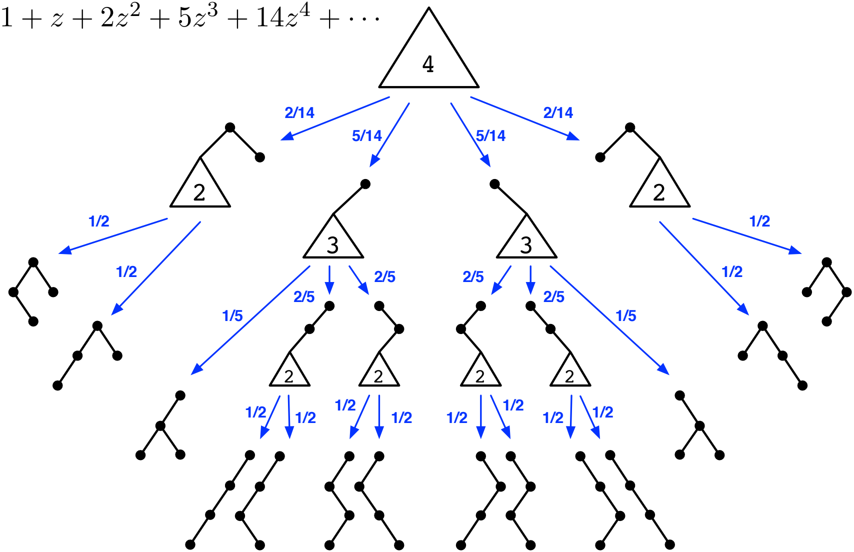


The recursive method on an example

Rooted binary trees

$1 + z + 2z^2 + 5z^3 + 14z^4 + \cdots$

$\mathcal{T} = \mathcal{E} + \mathcal{Z} \times \mathcal{T}^2$



Boltzmann model

Size distribution spread over the whole combinatorial class, but uniform for a sub-class of objects of the same size.

- approximate size sampling (control parameter).
- very large structures can be generated.

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

For a given $x < \rho_{\mathcal{C}}$, the *unlabelled* **Boltzmann model** associates to any object c in the class $\mathcal{C}$ the probability :

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

A Boltzmann sampler $\Gamma C(x)$ for the class $\mathcal{C}$ is an algorithm that produces objects of $\mathcal{C}$ according to this model.

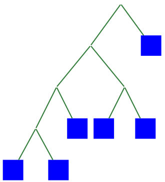
2 structures of the same size $\rightarrow$ same probability.

Générateurs [DuFlLoSc04]

	ordinary g.s. (unlabelled)	exponential g.s. (labelled)	
1 / $\mathcal{Z}$	1 / z	1 / z	
$\mathcal{A}+\mathcal{B}$	$A(z)+B(z)$	$A(z)+B(z)$	$\rightarrow$ Bernoulli law
$\mathcal{A}\times\mathcal{B}$	$A(z)\times B(z)$	$A(z)\times B(z)$	$\rightarrow$ no choice
$\textsc{Seq}(\mathcal{A})$	$\dfrac{1}{1-A(z)}$	$\dfrac{1}{1-A(z)}$	$\rightarrow$ geometric law $\triangleright\ \mathbb{P}(X=k)=(1-\lambda)\lambda^k$
$\textsc{PSet}(\mathcal{A})$	...	$\exp\big(A(z)\big)$	$\rightarrow$ Poisson law $\triangleright\ \mathbb{P}(X=k)=e^{-\lambda}\frac{\lambda^k}{k!}$
$\textsc{MSet}(\mathcal{A})$	...	–	
$\textsc{Cyc}(\mathcal{A})$	...	$\log\dfrac{1}{1-A(z)}$	$\rightarrow$ logarithmic law $\triangleright\ \mathbb{P}(X=k)=\frac{1}{\log(1-\lambda)^{-1}}\frac{\lambda^k}{k}$

Example : binary trees

$$\mathcal{B} = \mathcal{Z} + \mathcal{B} \times \mathcal{B}$$
$$B(z) = z + B(z)^2 = \frac{1-\sqrt{1-4z}}{2}$$



Algorithm : $\Gamma B(x)$

$b \leftarrow \text{Bern}(x/B(x))$;
if $b = 1$ **then**
  Return ■
else
  Return $\langle\ \Gamma B(x)\ ,\ \Gamma B(x)\ \rangle$;
end if

Boltzmann method [DuFlLoSc04]

Theorem (Labelled samplers – DuFlLoSc04)

For any labelled class $\mathcal{C}$ specified (possibly recursively) using the following constructions :

$$\varepsilon,\ \mathcal{Z},\ +,\ \times,\ \textsc{Seq},\ \textsc{Set},\ \textsc{Cyc},$$

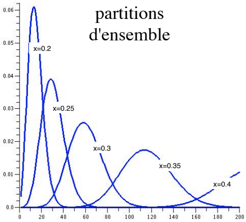
the free Boltzmann sampler $\Gamma C(x)$ works in linear time with respect to the size of the output.

- Free sampler : structures of any sizes !
  - ▷ target size n : choose x such that the **expected** size is n.

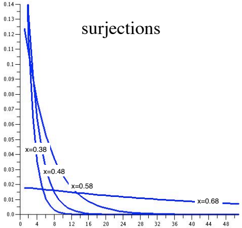
$$\mathbb{E}_x(N) = x\frac{C'(x)}{C(x)}$$

- Approximate or exact size sampler : use rejection.
  - ▷ the size distribution determine the cost of rejection.

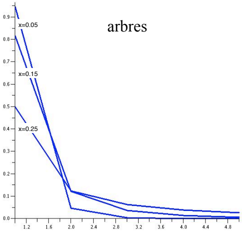
Approximate/exact size sampling



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



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



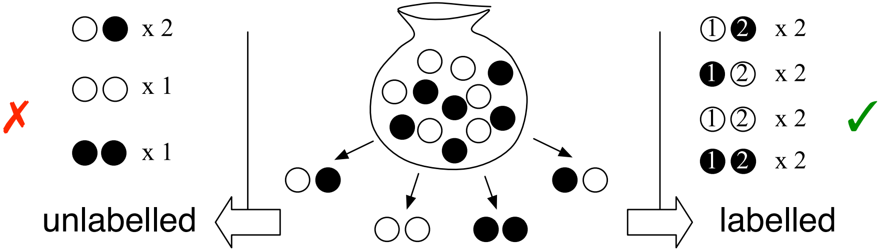
$$C(x) = \frac{1 - \sqrt{1-4x}}{2}$$

Theorem (Complexities – DuFlLoSc04)

distribution type	bumpy	flat	picked
approximate size	$\mathcal{O}(n)$	$\mathcal{O}(n)$	$o(n^2)$
exact size	$o(n^2)$	$o(n^2)$	–

▷ pointing
singular sampler

The symmetries issue



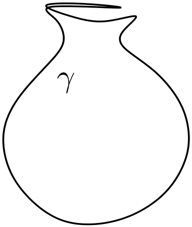
Pólya operators

	construction	ordinary g.s. (unlabelled)	exponential g.s. (labelled)
$\mathcal{E}$ / atom	1 / $\mathcal{Z}$	1 / z	1 / z
Union	$\mathcal{A}+\mathcal{B}$	$A(z)+B(z)$	$A(z)+B(z)$
Product	$\mathcal{A}\times\mathcal{B}$	$A(z)\times B(z)$	$A(z)\times B(z)$
Sequence	$\textsc{Seq}(\mathcal{A})$	$\dfrac{1}{1-A(z)}$	$\dfrac{1}{1-A(z)}$
Set	$\textsc{PSet}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty}\dfrac{(-1)^{k-1}}{k}A(z^k)\right)$	$\exp\big(A(z)\big)$
Multi-set	$\textsc{MSet}(\mathcal{A})$	$\exp\left(\sum_{k=1}^{\infty}\dfrac{1}{k}A(z^k)\right)$	–
Cycle	$\textsc{Cyc}(\mathcal{A})$	$\sum_{k=1}^{\infty}\dfrac{\varphi(k)}{k}\log\dfrac{1}{1-A(z^k)}$	$\log\dfrac{1}{1-A(z)}$

The multi-set example

MSet($\mathcal{A}$)

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

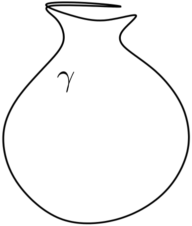


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet($\mathcal{A}$)

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

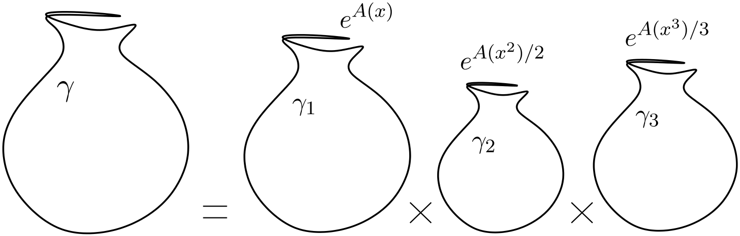


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet($\mathcal{A}$)

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

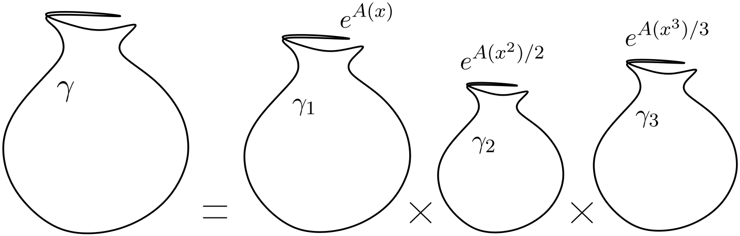


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet$(\mathcal{A})$

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

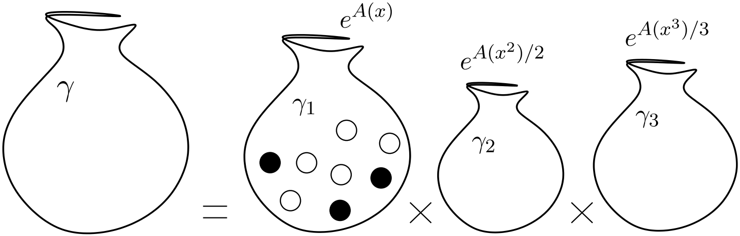


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet($\mathcal{A}$)

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

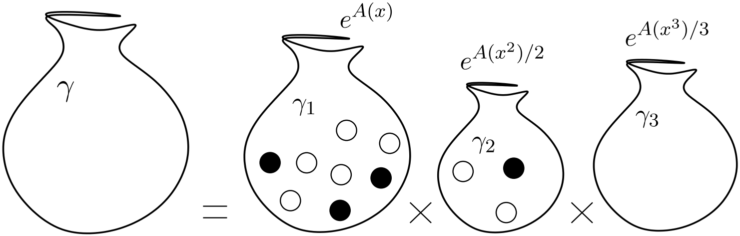


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet($\mathcal{A}$)

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

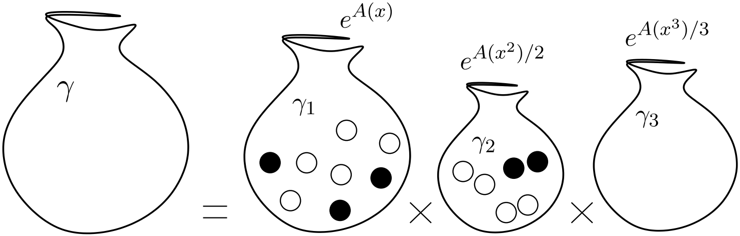


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet$(\mathcal{A})$

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

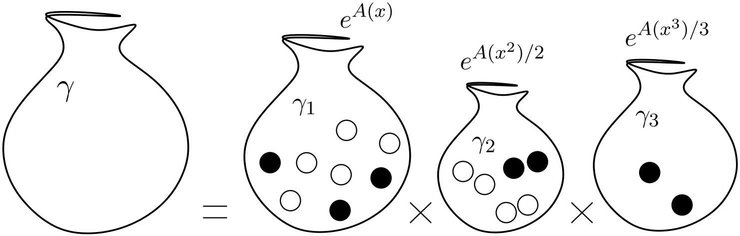


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet$(\mathcal{A})$

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

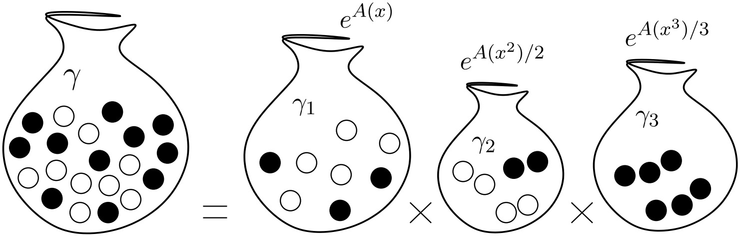


- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

The multi-set example

MSet($\mathcal{A}$)

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



- Draw k such that : $\mathbb{P}(K \leq k) = \prod_{j \leq k} \exp\left(A(x^j)/j\right)$
- For any index i up to k
  - Draw p – the number of elements to genearte – according to a Poisson law of parameter $A(x^i)/i$.
  - Perform p independent call to the sampler of the class $\mathcal{A}$, with parameter x^i, then make i copies of the result.

Unlabelled Boltzmann samplers

Theorem (Unlabelled samplers – FlFuPi07)

For any unlabelled class $\mathcal{C}$ specified (possibly recursively) using the following constructions :

$$\varepsilon,\ \mathcal{Z},\ +,\ \times,\ \textsc{Seq},\ \textsc{Seq}_k,\ \textsc{MSet},\ \textsc{MSet}_k,\ \textsc{Cyc},\ \textsc{Cyc}_k,$$

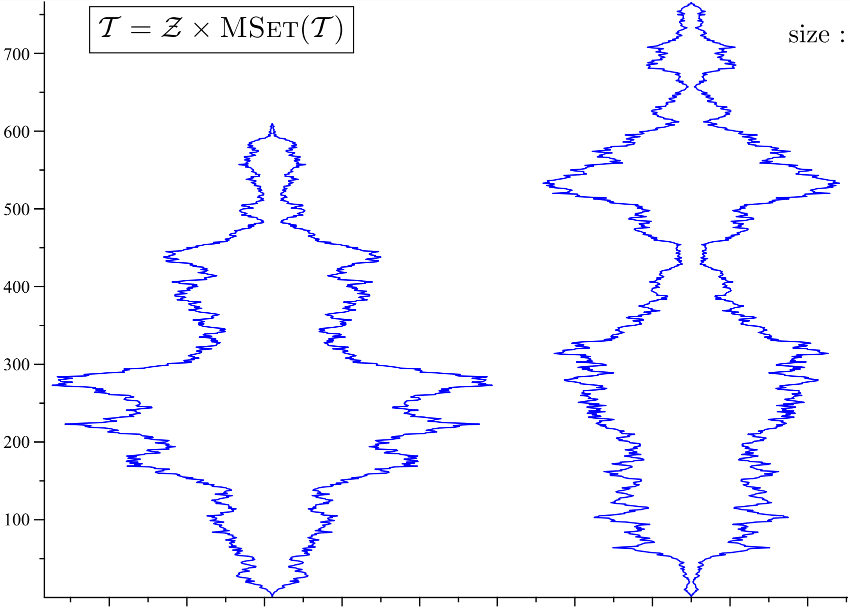
the free Boltzmann sampler $\Gamma C(x)$ works in linear time with respect to the size of the output.

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

General non planar rooted trees

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

size : $\sim 10^5$



Experimental results

Approximate size sampling $\pm 10\%$ (times in seconds)

	bumpy distribution	flat distribution		picked distribution		
	Integer partitions	Cyclic compositions	Functional graphs	Otter trees	General trees	Circuits
size	$O(\sqrt{n})$	$O(n)$	$O(n)$	$O(n)$	$O(n)$	$O(n)$
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	–	–	–	–	–

up to 10^{10}

with Maple

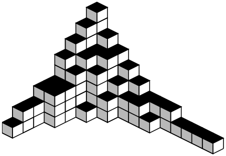
with OCaml, on an Intel processor, 3.2 GHz with a 2 GB RAM

Application to plane partitions

$$\mathcal{P} = \mathrm{MS}\textsc{et}(\mathcal{Z} \times \mathrm{S}\textsc{eq}(\mathcal{Z})^2)$$

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

1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0
2	1	0	0	0	0	0	0	0
2	2	0	0	0	0	0	0	0
3	2	0	0	0	0	0	0	0
4	3	2	0	0	0	0	0	0
5	5	4	1	0	0	0	0	0
6	5	5	4	3	3	0	0	0
8	7	6	5	3	3	2	2	1
12	7	4	3	2	2	1	1	1

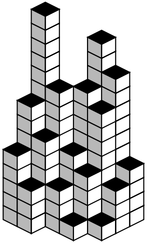


plane partition

$$\mathcal{S}_D \prod_{(i,j)\in D} \mathrm{S}\textsc{eq}(\mathcal{Z}^{h(i,j)})$$

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

4	3	1	0	0
10	8	4	1	0
12	9	5	3	0
		9	6	3
		14	8	5



skew plane partition

Theorem (Average complexity)

Plane partitions :

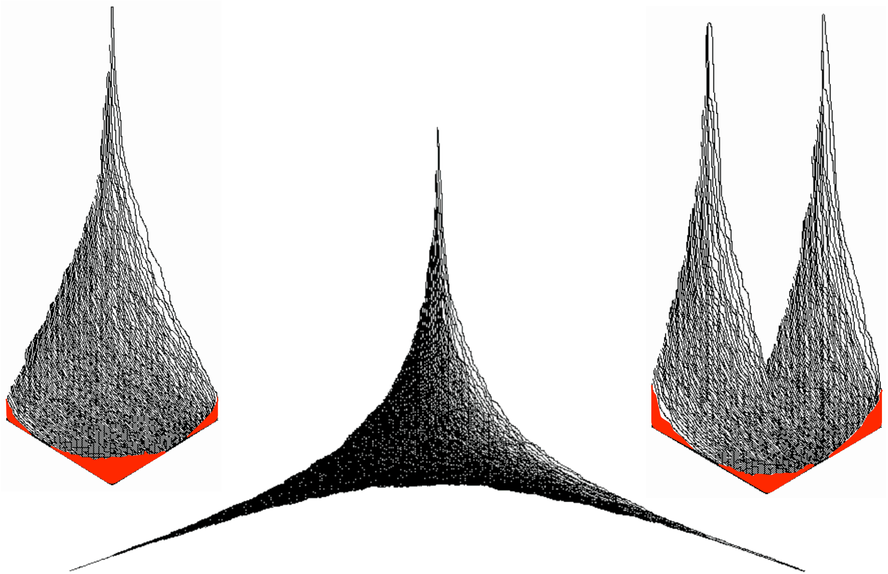
- *approx. size :* $O(n\ln(n)^3)$
- *exact size :* $O(n^{\frac{4}{3}})$

Boxed or skew partitions (on a domain D) :

- *approximate size :* $O(1)$ *when* $n \to \infty$
- *exact size :* *bounded by* $C|D|.n$, *C being a constant.*

[BoFuPi06]
Random sampling of plane partitions,
O. Bodini, E. Fusy and C. Pivoteau

Giant partitions



Related work (non exhaustive list)

Applications

- Accessible det. automata : *F. Bassino, J. David, C. Nicaud.*
- Planar graphs, *E. Fusy.*
- RNA structures, *Y. Ponty.*
- Apollonian network, *A. Darrasse, M. Soria.*
- Software testing, *J. Oudinet.*

Extensions

- Unbiased pointing operator for unlabeled structures, *M. Bodirsky, E. Fusy, M. Kang, S. Vigerske.*
- Colored structures, *O. Bodini, A. Jacquot.*
- Ordered product, Shuffle, Hadamard product, *A. Darrasse, K. Panagiotou, O. Roussel, M. Soria.*
- Multivariate Boltzmann samplers, *O. Bodini, Y. Ponty.*

Miscellaneous

- Properties of Random Graphs via Boltzmann Samplers, *K. Panagiotou, A. Weißl.*

Example

Combinatorial specification for an algebraic language :

$$\begin{aligned}
\mathcal{C}_0 &= \mathcal{Z}\mathcal{C}_1\mathcal{C}_2\mathcal{C}_3(\mathcal{C}_1+\mathcal{C}_2)\\
\mathcal{C}_1 &= \mathcal{Z}+\mathcal{Z}\textsc{Seq}(\mathcal{C}_1^2\mathcal{C}_3^2)\\
\mathcal{C}_2 &= \mathcal{Z}+\mathcal{Z}^2\textsc{Seq}(\mathcal{Z}\mathcal{C}_2^2\textsc{Seq}(\mathcal{Z}))\textsc{Seq}(\mathcal{C}_2)\\
\mathcal{C}_3 &= \mathcal{Z}+\mathcal{Z}(3\mathcal{Z}+\mathcal{Z}^2+\mathcal{Z}^2\mathcal{C}_1\mathcal{C}_3)\textsc{Seq}(\mathcal{C}_1^2)
\end{aligned}$$

The associated system on generating series :

$$\begin{aligned}
&C_0(z) = zC_1(z)C_2(z)C_3(z)(C_1(z)+C_2(z))\\
&C_1(z) = z + z/(1-C_1(z)^2C_3(z)^2)\\
&C_2(z) = z + z^2/((1-zC_2(z)^2/(1-z))(1-C_2(z)))\\
&C_3(z) = z + z(3z+z^2+z^2C_1(z)C_3(z))/(1-C_1^2(z))
\end{aligned}$$

Example

Combinatorial specification for an algebraic language :

$$\begin{aligned}
\mathcal{C}_0 &= \mathcal{Z}\mathcal{C}_1\mathcal{C}_2\mathcal{C}_3(\mathcal{C}_1+\mathcal{C}_2)\\
\mathcal{C}_1 &= \mathcal{Z}+\mathcal{Z}\,\mathrm{S}\textsc{eq}(\mathcal{C}_1^2\mathcal{C}_3^2)\\
\mathcal{C}_2 &= \mathcal{Z}+\mathcal{Z}^2\mathrm{S}\textsc{eq}(\mathcal{Z}C_2^2\mathrm{S}\textsc{eq}(\mathcal{Z}))\mathrm{S}\textsc{eq}(\mathcal{C}_2)\\
\mathcal{C}_3 &= \mathcal{Z}+\mathcal{Z}(3\mathcal{Z}+\mathcal{Z}^2+\mathcal{Z}^2\mathcal{C}_1\mathcal{C}_3)\mathrm{S}\textsc{eq}(\mathcal{C}_1^2)
\end{aligned}$$

The associated system on generating series : with $z = 0.27$

$$\begin{aligned}
&C_0(z) = zC_1(z)C_2(z)C_3(z)(C_1(z)+C_2(z))\\
&C_1(z) = z + z/(1-C_1(z)^2C_3(z)^2)\\
&C_2(z) = z + z^2/((1-zC_2(z)^2/(1-z))(1-C_2(z)))\\
&C_3(z) = z + z(3z+z^2+z^2C_1(z)C_3(z))/(1-C_1^2(z))
\end{aligned}$$

Example

Combinatorial specification for an algebraic language :

$$
\begin{aligned}
\mathcal{C}_0 &= \mathcal{Z}\mathcal{C}_1\mathcal{C}_2\mathcal{C}_3(\mathcal{C}_1+\mathcal{C}_2)\\
\mathcal{C}_1 &= \mathcal{Z}+\mathcal{Z}\textsc{Seq}(\mathcal{C}_1^2\mathcal{C}_3^2)\\
\mathcal{C}_2 &= \mathcal{Z}+\mathcal{Z}^2\textsc{Seq}(\mathcal{Z}\mathcal{C}_2^2\textsc{Seq}(\mathcal{Z}))\textsc{Seq}(\mathcal{C}_2)\\
\mathcal{C}_3 &= \mathcal{Z}+\mathcal{Z}(3\mathcal{Z}+\mathcal{Z}^2+\mathcal{Z}^2\mathcal{C}_1\mathcal{C}_3)\textsc{Seq}(\mathcal{C}_1^2)
\end{aligned}
$$

The associated system on generating series : with $z = 0.27$

$$
\begin{aligned}
C_0 &= 0.27C_1C_2C_3(C_1+C_2)\\
C_1 &= 0.27+0.27/(1-C_1^2C_3^2)\\
C_2 &= 0.27+0.0729/((1-0.2754211138C_2^2)(1-C_2))\\
C_3 &= 0.27+0.27(0.8829+0.0729C_1C_3)/(1-C_1^2)
\end{aligned}
$$

with Maple

$$sys := \left[C0 = x\, C3\, C1\, C2\, (C1 + C2),\ C3 = x + \frac{x\,(3\,x + x^2 + C1\, C3\, x^2)}{1 - C1^2},\right.$$

$$\left. C2 = x + \frac{x^2}{\left(1 - \frac{x\, C2^2}{1 - x}\right)(1 - C2)},\ C1 = x + \frac{x}{1 - C1^2\, C3^2} \right]$$

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

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

Idea

By iteration

Boltzmann Oracle :
numerical iteration that converges
to the unique **relevant** solution

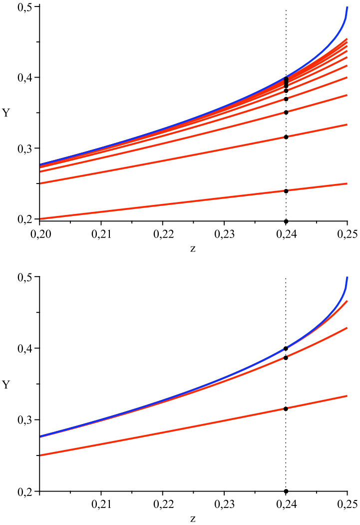
$\Uparrow$

convergence of the iteration
on counting series

$\Uparrow$

convergence of the iteration on
combinatorial functional systems

Fast oracle : Newton iteration



Binary trees : $B(z) = z + B(z)^2$

Theorem (Transfer of convergence – PiSaSo08)

Let $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{H}}(\mathcal{Z}, \boldsymbol{\mathcal{Y}})$ be a system such that $\boldsymbol{\mathcal{H}}(0, \mathbf{0}) = \mathbf{0}$ and $\partial\boldsymbol{\mathcal{H}}/\partial\boldsymbol{\mathcal{Y}}(0, \mathbf{0})$ is nilpotent. If the combinatorial iteration :

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

converges to the family of species $\boldsymbol{\mathcal{Y}}$, solution of $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{H}}(\mathcal{Z}, \boldsymbol{\mathcal{Y}})$, then the iteration :

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

converges to the vector of generating series $\boldsymbol{Y}(z)$ of $\boldsymbol{\mathcal{Y}}$. Besides, if $\boldsymbol{\mathcal{F}}$ is analytic, for any α such that $|\alpha| < \rho$, the iteration :

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

converges to the vector $\boldsymbol{Y}(\alpha)$ of values of the series $\boldsymbol{Y}(z)$ at point α.

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

Validity of this approach

Combinatorial convergence

- Implicit Species Theorem [Joyal 81] : existence and unicity of the solution for the system $\boldsymbol{\mathcal{Y}} = \boldsymbol{\mathcal{H}}(\mathcal{Z}, \boldsymbol{\mathcal{Y}})$:
  - $\boldsymbol{\mathcal{H}}(\mathbf{0}, \mathbf{0}) = \mathbf{0}$,
  - the jacobian matrix $\frac{\partial \boldsymbol{\mathcal{H}}}{\partial \boldsymbol{\mathcal{Y}}}(\mathbf{0}, \mathbf{0})$ is nilpotent.
- Measure of convergence : contact between combinatorial classes.

Generating series

- Counting of combinatorial structures.
- Contact $\Leftrightarrow$ valuation of series.

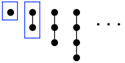
Numerical iteration

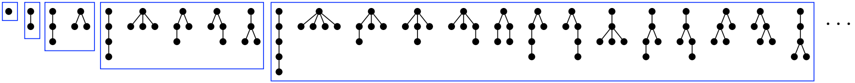
- Numerical values $\Leftrightarrow$ evaluation of series at point α.
- Analyticity of $\boldsymbol{H}$ in α and $\boldsymbol{y}^{[n]}$.

Example of general plane trees (Newton)

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

$$\boxed{\mathcal{T} = \mathcal{Z} \times \textsc{Seq}(\mathcal{T})}$$

$\mathcal{Y}^{[1]} =$ 

$\mathcal{Y}_{[2]} =$ 

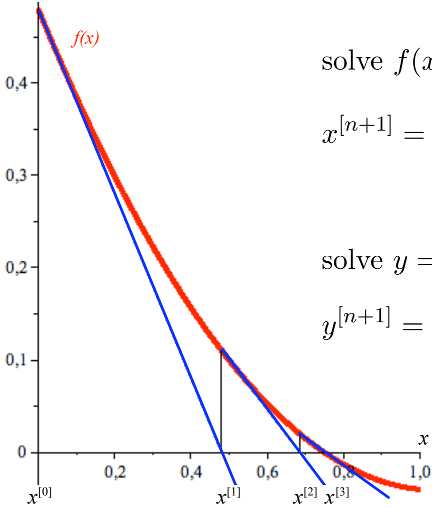
$$Y^{[0]}(z) = \mathbf{0}$$
$$Y^{[1]}(z) = \mathbf{z + z^2} + z^3 + z^4 + z^5 + z^6 + z^7 + z^8 + z^9 + \cdots$$
$$Y^{[2]}(z) = \mathbf{z + z^2 + 2\,z^3 + 5\,z^4 + 14\,z^5 + 42\,z^6} + 131\,z^7 + 417\,z^8 + 1341\,z^9 \cdots$$
$$Y^{[3]}(z) = \mathbf{z + z^2 + 2\,z^3 + 5\,z^4 + 14\,z^5 + 42\,z^6 + 132\,z^7} + \cdots + \mathbf{742900\,z^{14}} + \cdots$$
$$Y^{[4]}(z) = \mathbf{z + z^2 + 2\,z^3 + 5\,z^4 + 14\,z^5 + 42\,z^6} + \cdots + \mathbf{1002242216651368\,z^{30}} + \cdots$$

with $z = 0.2$

$$y^{[0]} = \mathbf{0}$$
$$y^{[1]} = \mathbf{0.2}5000000000000000000000000000000\ldots$$
$$y^{[2]} = \mathbf{0.27}586206896551724137931034482758\ldots$$
$$y^{[3]} = \mathbf{0.27639}296187683284457478005865104\ldots$$
$$y^{[4]} = \mathbf{0.2763932022}4997168102435963404908\ldots$$
$$y^{[5]} = \mathbf{0.27639320225002103035908263}104683\ldots$$

1. Introduction
2. Boltzmann Samplers
   - Labelled Boltzmann Samplers
   - Unlabelled samplers
3. Computing the oracle
   - Transfer of convergence
   - Newton iterations
4. Conclusion

Classical Newton numerical iteration



solve $f(x) = 0$:

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

solve $y = f(y)$:

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

Combinatorial Newton iteration

For a unique equation [Décoste, Labelle, Leroux 82] :

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

- ▷ combinatorial derivative

For a system :

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

- ▷ jacobian matrix
- ▷ "combinatorial bloomings" [Labelle 85]

Optimization

Compute the matrix $\left(\mathbf{Id} - \frac{\partial\boldsymbol{\mathcal{H}}}{\partial\boldsymbol{\mathcal{Y}}}(\boldsymbol{\mathcal{Z}},\boldsymbol{\mathcal{Y}}^{[n]})\right)^{-1}$ by Newton iteration

- ▷ iterations in parallel.

Combinatorial derivative

$\partial\mathcal{H}/\partial\mathcal{T}$: *derivative* of $\mathcal{H}(\mathcal{Z},\mathcal{Y}_1,\ldots,\mathcal{Y}_m)$ with respect to $\mathcal{T}$.

	notation		derivative	
atom	$\mathcal{Z}$ ou $\mathcal{E}$		$\varnothing$	
$\mathcal{Y}_i$	$\mathcal{Y}_i = \mathcal{T}$	$\mathcal{Y}_i \neq \mathcal{T}$	$\mathcal{E}$	$\mathcal{Y}_i'$
union	$\mathcal{Y}_i + \mathcal{Y}_j$		$\mathcal{Y}_i + \mathcal{Y}_i' + \mathcal{Y}_j'$	
product	$\mathcal{Y}_i \times \mathcal{Y}_j$		$\mathcal{Y}_i' \times \mathcal{Y}_j + \mathcal{Y}_i \times \mathcal{Y}_j'$	
sequence	$\textsc{Seq}(\mathcal{Y}_i)$		$\textsc{Seq}(\mathcal{Y}_i) \times \mathcal{Y}_i' \times \textsc{Seq}(\mathcal{Y}_i)$	
cycle	$\textsc{Cyc}(\mathcal{Y}_i)$		$\textsc{Seq}(\mathcal{Y}_i) \times \mathcal{Y}_i'$	
set	$\textsc{Set}(\mathcal{Y}_i)$		$\textsc{Set}(\mathcal{Y}_i) \times \mathcal{Y}_i'$	

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

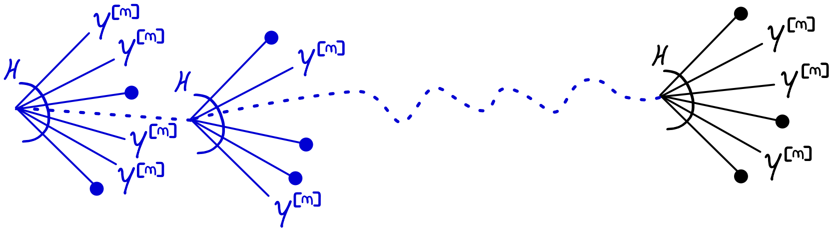
▷ Combinatorial interpretation of the product by a derivative

$\boldsymbol{\partial\mathcal{H}/\partial\mathcal{Y}}$: la *jacobian matrix* de $\boldsymbol{\mathcal{H}}(\mathcal{Z},\boldsymbol{\mathcal{Y}})$ with respect to $\boldsymbol{\mathcal{Y}}$,
its entries are the partial derivatives $\partial\mathcal{H}_i/\partial\mathcal{Y}_j(\mathcal{Z},\boldsymbol{\mathcal{Y}})$.

Convergence of the combinatorial Newton iteration

Combinatorial interpretation of Newton iteration
[Décoste, Labelle, Leroux 82]

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



▷ quadratic convergence !

Newton iteration on series

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

Fast algorithm to compute the coefficients !

Arithmetic and binary complexity : quasi-optimal

Theorem (Arithmetic complexity)

Cost of the calculation of the first N terms of $\boldsymbol{Y}(z)$: $O(\mathsf{M}(N))$

- ▷ $\mathsf{M}(N)$: cost of multiplying series of order N.
- ▷ with FFT : $O(N \log N)$.

- ▷ better complexity for the recursive method.

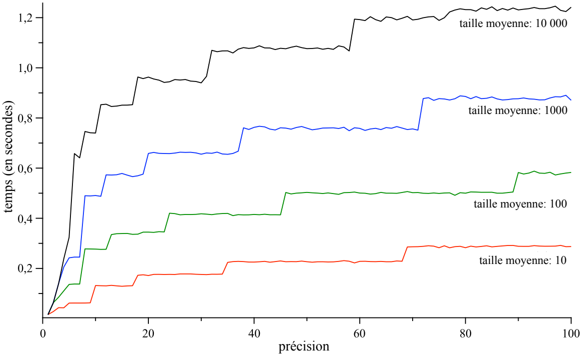
Numerical oracle

Theorem (Boltzmann oracle– PiSaSo08)

Let $\mathcal{Y} = \mathcal{H}(\mathcal{Z}, \mathcal{Y})$ be a combinatorial specification (without Pólya operator). The followinf numerical iteration converges to $\boldsymbol{Y}(\alpha)$:

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

Evolution of the computation time of the oracle for the SP circuits with respect to the precision, for different values of α.



Implementation

Maple prototype
(specifications without Pólya operators) :

- tests on random grammars,
- XML grammars, $\sim 10^3$ equations [Darrasse 08],
- Uniform random walks in very large models (software testing) [Oudinet 07],
- ...

# equations	4	10	50		100		500
# constructions/eqn	10	10	10	50	10	50	50
$\sim$ size max scc	2.47	3.42	7.95	18.62	10.93	67.18	339.1
time (0.99ρ)	0.05	0.11	0.17	0.47	0.23	7.29	61.73
time (0.999999ρ)	0.08	0.16	0.19	0.56	0.25	8.11	61.86
$\sim$ expected size	$4.1\,10^{14}$	$1.4\,10^{7}$	$2.2\,10^{5}$	$1.0\,10^{5}$	$1.2\,10^{6}$	$5.0\,10^{4}$	$3.3\,10^{4}$

in seconds, with Maple 11, on an Intel processor, 3.2 GHz, with 2 GB RAM.

Conclusion

- Efficient sampling method, with large scale applications
- Toward a generic package of Boltzmann samplers :
  Maple ? Sage ? C ?

Work in progress :

- Pólya operators
- specifications involving $\mathcal{E}$
- binary complexity

Perspectives :

- Convergence acceleration
- Singularities computation
  - ▷ automatic asymptotic analysis
- Boltzmann parameter tuning