Dirichlet Random Samplers for Multiplicative Structures

Olivier Bodini
LIPN, Univ. Villetaneuse

Jérémie Lumbroso
LIP6 / INRIA Rocquencourt

January 16th, 2012
京都市 (Kyōto), Japan

1. RANDOM combinatorial structures

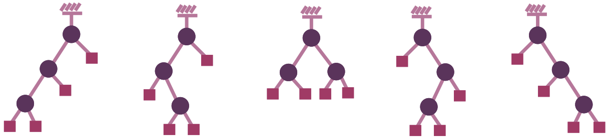
combinatorial structures: symbolically specified (with “grammars”) using operators

- $+$ (disjoint union), $\times$ (Cartesian product)
- Seq (sequence), Set (set), etc.

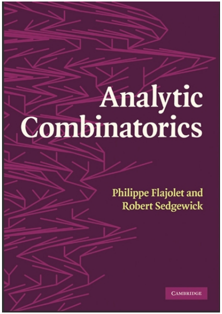
automatically get (counting) generating function [Flajolet & Sedgewick 2009]

random generation: given specification, draw these objects randomly [randomly = say there are C_n objects of size n, I want to pick/**construct** one with probability $1/C_n$]

- binary trees: $\mathcal{B} = \mathcal{Z} + \mathcal{B} \times \mathcal{B}$
  $\Rightarrow B(z) = z + B(z)^2 = \frac{1-\sqrt{1-4z}}{2} = \sum_{n=0}^{\infty} b_n\, z^n$

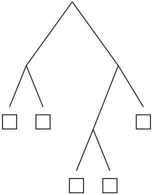
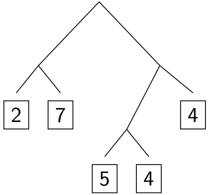


1/5 1/5 1/5 1/5 1/5



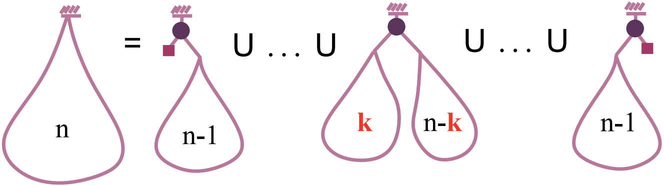
some of the many applications

- **analysis:** study specific properties/statistics of huge
  - generate many random objects, and empirically study properties
  - compare real data with (randomly generated) uniform data: in genetics, in **poetry** [Gasparov 1987]
- **testing:** generate input for algorithm/server to test robustness and ability to withstand heavy loads [Mougenot *et al.* 2009]
- **entertainment:** create objects (trees, trains, etc.) or environments (buildings, forests, cities, etc.) for video games or movies

ADDITIVE (traditional objects)	MULTIPLICATIVE[1] (this talk)
$\alpha \in \mathcal{A},\ \beta \in \mathcal{B} \quad \lvert(\alpha,\beta)\rvert = \lvert\alpha\rvert + \lvert\beta\rvert$	$\alpha \in \mathcal{A},\ \beta \in \mathcal{B} \quad \lvert(\alpha,\beta)\rvert = \lvert\alpha\rvert \cdot \lvert\beta\rvert$
unique atom $\mathcal{Z}$ of unit size 1	**infinity** of atoms, $\mathcal{Z}_m$ $(m \in \mathbb{Z}_{>0})$
$\mathcal{A} = \mathcal{Z} + \mathcal{A} \times \mathcal{A}$	$\mathcal{M} = \mathcal{I} \setminus \mathcal{Z}_1 + \mathcal{M} \times \mathcal{M}$
	
$1+1+1+1+1 = 5$	$2 \times 7 \times 5 \times 4 \times 4 = 1120$
Ordinary GF or Exponential GF	Dirichlet GF
$\sum_{k=0}^{\infty} a_k z^k \qquad \sum_{k=0}^{\infty} \frac{a_k}{k!} z^k$	$\sum_{k=1}^{\infty} a_k \frac{1}{k^s}$

[1] First considered from a symbolic/combinatoric perspective by Hwang (1994).

2. INTUITIVE WAY of generating trees



$$b_n = \sum_{k=1}^{n-1} b_k \cdot b_{n-k}$$

k with prob. $(b_k \cdot b_{n-k})/b_n$

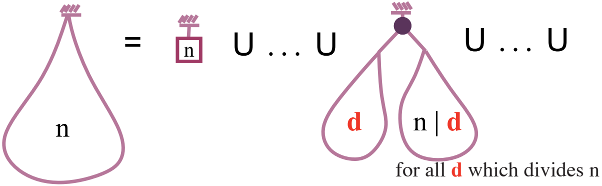
To generate tree of size n:

- pick k (following a certain law)
- recursively generate subtrees of size k and $n - k$

Called "recursive method" [Nijenhuis & Wilf 1978], [Flajolet *et al.* 1994].

Precalculate $b_1, ..., b_n$ (# trees of size n) to generate obj. up to size n
$\Longrightarrow O(n^2)$ time preprocessing, $O(n^2)$ aux. memory, $O(n)$ generation.

cannot be extended (efficiently) to multiplicative objects



$$b_n = 1 + \sum_{\substack{d|n \\ 1<d<n}} b_d \cdot b_{n|d}$$

PROBLEMS of efficiency

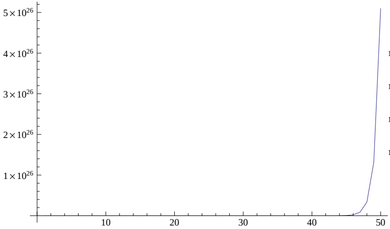
- sizes (wrt. number of "nodes") exponentially larger than for additive objects
- requires **factor decomposition** which is (too) costly

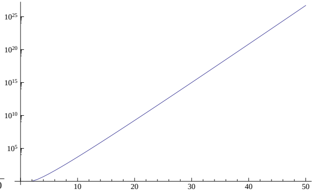
PROBLEMS of quality

- size distribution is highly irregular

size distributions (# obj. of given size)

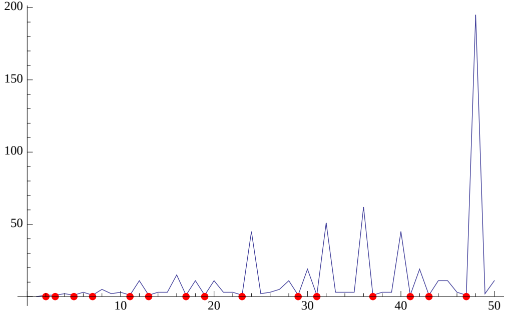
additive binary trees (plot then logplot)





---

multiplicative (binary) branching factorizations



3. ANALYTIC RANDOM SAMPLING

best way of calculating b_n coefficients: extract from generating function[2]

$$\mathcal{B} = \mathcal{Z} + \mathcal{B} \times \mathcal{B} \quad \Rightarrow \quad B(z) = z + B(z)^2 = \frac{1 - \sqrt{1-4z}}{2} = \sum_{n=0}^{\infty} b_n \cdot z^n$$

Boltzmann sampling consists instead in taking a biased average of the coefficients by evaluating the function

---

RECURSIVE version [Flajolet *et al.* 1994]

RTree(n) := {
  if $n = 1$ **then** return Leaf
  else
    k from distr. $\mathbb{P}[K = k] = (b_k \cdot b_{n-k})/b_n$
    return Node(RTree(k), RTree($n - k$))
}

"BOLTZMANN" vers. [Duchon *et al.* 02]

ATree(z) := {
  if $\mathrm{Ber}(z/B(z)) = 1$ **then** return Leaf
  else
    return Node(ATree(z), ATree(z))
}

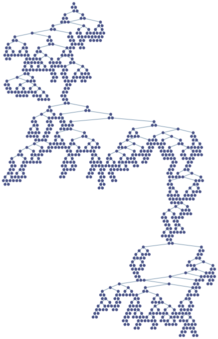
- in "Boltzmann"/analytic random sampling, the randomization is **global**: the same law is calculated in all recursive calls
- size is approximate, but uniformity given size is preserved
- no preprocessing, $O(n)$ generation complexity

[2] Typically using the holonomic decomposition.

extending the idea to multiplicative objects

Theorem [Bodini & Lumbroso 2012]. Let $\mathcal{C}$ be a multiplicative combinatorial class described with: disjoint union, cartesian product, sequence, well-founded recursion, etc.

Under some hypotheses on the generating function, a Dirichlet sampler for $\mathcal{C}$ can generate an object of size n, with some error $\varepsilon \in (0, 1)$, in $O(\log(n)^2)$ worst-case time complexity.

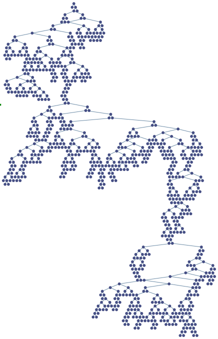


extending the idea to multiplicative objects

Theorem [Bodini & Lumbroso 2012]. Let $\mathcal{C}$ be a multiplicative combinatorial class described with: disjoint union, cartesian product, sequence, well-founded recursion, etc.

Under some hypotheses on the generating function, a Dirichlet sampler for $\mathcal{C}$ can generate an object of size n, with some error $\varepsilon \in (0,1)$, in $O(\log(n)^2)$ worst-case time complexity.

- Zeta-distributed atoms sampled in $O(1)$ [Devroye 1986]
- resorts to analytic number theory: specifically Delange's Tauberian theorem, as equivalent of Flajolet-Odlyzko transfer theorem in additive combinatorics
- tuning of control parameter completely different: in Boltzmann sampling, direct inversion of expected value; here expected value is infinite and requires *ad-hoc* tuning informed from theorem



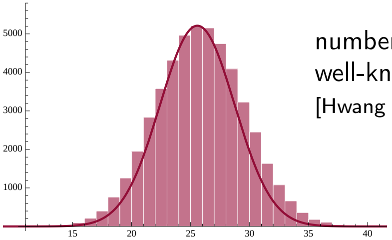
ordered factorizations, $\mathcal{F} := \mathsf{Seq}\left(\mathcal{I} \setminus \mathcal{Z}_1\right)$

$$
\begin{aligned}
&\Gamma\mathrm{D}_s\left[\mathcal{F}\right] := \{\\
&\qquad \lambda \leftarrow \zeta(s) - 1;\\
&\qquad K \in \mathrm{Geo}(\lambda);\\
&\qquad \text{return } (\underbrace{\Gamma\mathrm{D}_s\left[\mathcal{I} \setminus \mathcal{Z}_1\right], \ldots, \Gamma\mathrm{D}_s\left[\mathcal{I} \setminus \mathcal{Z}_1\right]}_{K \text{ times}})\\
&\}
\end{aligned}
$$

```
OrderedFactorization[10^200, 0.5] // AbsoluteTiming

{17.002826, {598,
  94 315 438 343 755 964 449 064 464 145 270 360 907 587 302 431 535 020 906 407 \
   589 438 865 191 662 481 620 456 946 846 202 450 914 444 733 710 252 639 029 \
   394 242 922 918 929 394 271 546 094 283 086 276 198 942 107 362 365 753 807 \
   339 520 000 000 000 000 000 000 000,
  {4, 3, 5, 131, 2, 9, 5, 3, 4, 3, 4, 3, 51, 5, 2, 7, 3, 3, 3, 2, 2, 2, 4,
   2, 3, 5, 3, 3, 23, 3, 3, 6, 5, 10, 2, 6, 6, 2, 22, 2, 2, 3, 18, 242,
   3, 7, 3, 4, 2, 379, 4, 2, 7, 2, 9, 3, 12, 2, 46, 7, 2, 4, 9, 2, 3, 7,
   2, 11, 2, 3, 2, 3, 5, 6, 2, 2, 9, 9, 5, 20, 24, 35, 4, 2, 4, 2, 4, 2,
   2, 2, 5, 2, 2, 3, 6, 3, 2, 5, 22, 3, 13, 16, 2, 3, 2, 3, 4, 2, 21, 4,
   2, 2, 6, 3, 4, 4, 6, 70, 13, 3, 10, 3, 2, 3, 894, 4, 14, 2, 2, 22, 6,
   4, 2, 3, 13, 3, 11, 2, 3, 7, 53, 4, 2, 3, 47, 3, 77, 2, 2, 2, 4, 6, 6,
   6, 3, 2, 7, 4, 2, 8, 2, 3, 2, 53, 3, 4, 33, 2, 2, 6, 4, 3, 7, 15, 3, 7,
   222, 9, 7, 3, 3, 18, 2, 12, 2, 2, 2, 2, 2, 29, 5, 9, 2, 305, 904, 2,
   2, 12, 7, 2, 2, 4, 2, 3, 2, 54, 2, 27, 9, 18, 2, 3, 41, 8, 2, 44, 2,
   3, 2, 4, 2, 3, 2, 3, 2, 4, 17, 4, 5, 2, 5, 2, 53, 8, 2, 40, 2, 2, 4,
   2, 3, 3, 4, 6, 3, 2, 2, 2, 15, 13, 14, 7, 14, 3, 2, 3, 7, 3, 8, 2, 2,
   33, 3, 4, 3, 7, 523, 3, 10, 3, 3, 2, 12, 3, 86, 67, 4, 2, 2, 2, 2}}}
```

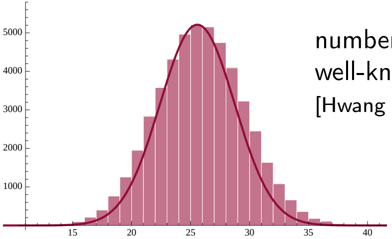
ordered factorizations, $\mathcal{F} := \mathrm{Seq}\,(\mathcal{I} \setminus \mathcal{Z}_1)$



number of factors in random ordered factorizations well-known to be **normally distributed**

[Hwang 1999] [Hwang and Janson 2009]

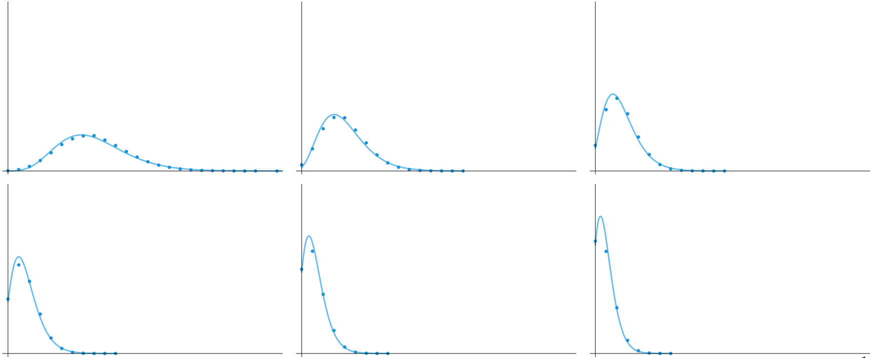
ordered factorizations, $\mathcal{F} := \text{Seq}(\mathcal{I} \setminus \mathcal{Z}_1)$



number of factors in random ordered factorizations well-known to be **normally distributed**

[Hwang 1999] [Hwang and Janson 2009]

number of factors equal to $m = 2, 3, 4, \ldots$ is **gamma distributed** (conjecture)



4. parting words

- non-trivial extension of Boltzmann sampling to multiplicative combinatorics
- first automatic random generation method for multiplicative objects (where previous techniques were limited to exhaustive generation of specific objects)
- could assist in their exploration
- through this work we have gained a lot of insight into what makes Boltzmann sampling tick, and hope to extend the concept in generality to what we suggest be called "**Analytics Random Sampling**"

The slides + Mathematica notebook with simulations:
`http://lip6.fr/Jeremie.Lumbroso/Talks/Analco2012/` (case sens.)