Efficient Random-Bit Algorithms for Uniform Involutions

Olivier Bodini[1], Francis Durand[12]
EREN, Université Sorbonne Paris-Nord[1], LIP6, Sorbonne Université[2]

In large combinatorial structures, randomness dominates over deterministic computation. We present entropy-optimal algorithms for uniform sampling of involutions, minimizing the costly resource of random bit generation while maintaining $O(n \log n)$ time complexity. Our approach matches the Shannon entropy lower bound of $\frac{n \log n}{2} + O(n)$ random bits.

Definition and Properties of Involutions

An **involution** $\sigma : [n] \to [n]$ satisfies $\sigma \circ \sigma = \mathrm{id}$.

- **Structure:** Disjoint union of:
  - Fixed points (1-cycles)
  - Transpositions (2-cycles)
- **Recurrence:**

$$I_n = I_{n-1} + (n-1)I_{n-2} \quad (n \geq 2), \quad \text{with} \quad I_0 = I_1 = 1$$

- **Exponential Generating Function:**

$$I(x) = \sum \frac{I_n x^n}{n!} = e^{x + x^2/2}$$

Entropic Sampling

Asymptotic count: For n-element involutions:

$$I_n \sim \frac{1}{\sqrt{2}} \left(\frac{n}{e}\right)^{n/2} e^{\sqrt{n} - \frac{1}{4}}$$

Shannon entropy:

$$H_n = \log_2 I_n = \frac{n \ln n}{2 \ln 2} + O(\sqrt{n})$$

Why entropy matters:

- Measures the *randomness*: the limiting factor for Sampling
- Algorithms must use $\geq H_n$ bits to encode involutions
- Our algorithms achieve $\frac{n \ln n}{2 \ln 2} + O(n)$ random bit consumption
  $\Rightarrow$ *entropy-optimal* up to lower-order terms

Sampling by using Ghost Elements

Other Decomposition: $\boxed{I_n = \frac{1}{\sqrt{e}} \sum_{2k \geq n} \frac{(2k)^{\underline{n}}}{2^k k!}}$

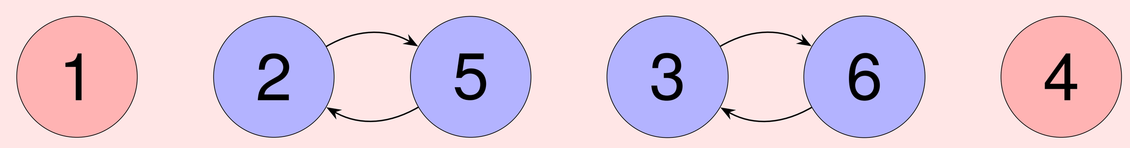
with $(2k)^{\underline{n}} = (2k)(2k-1)\ldots(2k-n+1)$ the falling factorial.
Random Sampling Algorithm:

1. Sample ghost count G with $P(G = 2k - n) \propto \frac{(2k)^{\underline{n}}}{2^k k!}$
2. Create G ghost elements with n real elements
3. Generate uniform pairing of $G + n$ elements
4. Transform pairs:
   - Two real elements → transposition
   - One real + one ghost → fixed point
   - Two ghosts → discard
5. Returns a Uniform Involution

Number of discarded ghost pairs: Poisson of parameter $\frac{1}{2}$, independent of the Involution sampled

An Involution of size 6

An Involution: $\sigma = (1)(4)(2\ 5)(3\ 6)$



Sampling via the Classic Decomposition Scheme

Counting Formula: For n-element involutions with k fixed points ($n-k$ even):

$$\boxed{I_{n,k} = \binom{n}{k} \frac{(n-k)!}{2^{(n-k)/2} \left(\frac{n-k}{2}\right)!}}$$

Total involutions: $I_n = \sum_{\substack{0 \leq k \leq n \\ k \equiv n \pmod 2}} I_{n,k}$

Random Sampling Algorithm:

1. Sample k from distribution of fixed point counts
2. Select k fixed points
3. Pair remaining elements uniformly at random

Complexity of the Algorithms

- Time: $O(n \log n)$ (optimal up to a constant factor)
- Random bits: $\log_2 I_n + O(n)$ (entropy-optimal)
- Sampling k *with a log-concave sampler* [Dev87]
- Allows to sample Involutions of sizes up to 10^9

Generating Functions Interpretation

- Pair configurations: $\phi(x) = \frac{e^{x^2/2}}{\sqrt{e}} = \sum_{k \geq 0} \frac{x^{2k}}{2^k k!}$
- Ghost element GF: $G_n(x) = \frac{1}{\sqrt{e}} \sum_{k \geq 0} \frac{(2k)^{\underline{n}} x^{2k-n}}{2^k k!}$

Hermite Decomposition:

$$\left(\frac{d}{dx}\right)^n \phi(x) = G_n(x) = H_n(x)\phi(x)$$

where $H_n(x) = \sum_{k=0}^{\lfloor n/2 \rfloor} I_{n,n-2k} x^{n-2k}$ (variant of Hermite polynomials)
Combinatorial Meaning:

- LHS: n pointing-erasing operations on pairs: the Matching
- RHS: Product of involution GF $H_n(x)$ and ghost pair GF $\phi(x)$

Reference

Luc Devroye.
A simple generator for discrete log-concave distributions.
Computing, 39(1):87–91, 1987.