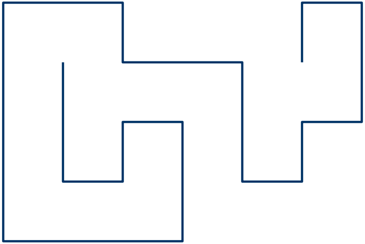
Loop-Erased Random Walks

Baining Zhang
Supervised by Prof Maximilian Nitzschner

May 13, 2025

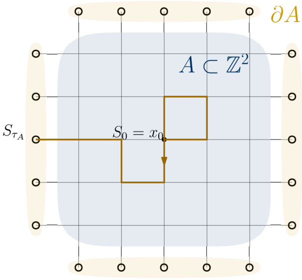
Introduction

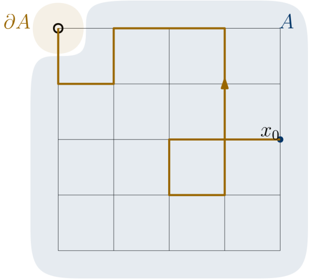


- Self-avoiding walks (SAW) are paths that do not visit the same vertex twice.
- Introduced by Lawler in 1979, the loop-erased random walk (LERW) is a model of self-avoiding walks.
- LERW = Markov chains + loop-erasure.

Markov Chains on Finite Sets

- Let $A, \partial A$ be disjoint finite vertex sets and denote $\overline{A} = A \cup \partial A$. We may also use $A, \partial A, \overline{A}$ to denote their complete graphs.
- A Markov chain S from x_0 to ∂A is given by the measure $\mathbf{P}^{x_0}$ with transition probabilities $(p(x, y))_{x,y \in \overline{A}}$.
- It is stopped at ∂A by setting $p(x, y) = \delta_{xy}$ for $x, y \in \partial A$.
- Its path is given by $S[0, \tau_A] = [S_0, \ldots, S_{\tau_A}]$, where $\tau_A = \inf\{j \geq 0\,;\, S_j \notin A\}$ is the exit time of A.
- Simple random walk: a Markov chain with transition probabilities $p(x, y) = 1/d_A(x)$, the degree of x in A.





Markov Chain as Measure on Path Space

- Given a directed edge $\vec{e} = [x, y]$, define weight $p(\vec{e}) = p(x, y)$.
- We can concatenate directed edges $\vec{e} = [x, y], \vec{f} = [y, z]$ to form a path $\vec{e} \oplus \vec{f} = [x, y, z]$.
- For a path $\omega = [x_0, \dots, x_n] = \vec{e_1} \oplus \cdots \oplus \vec{e_n}$, $p(\omega) = \prod_{j=1}^{n} p(\vec{e_j})$.
- This induces a measure p on sets of paths, e.g. we can write $p(\mathcal{K}_A(x, y))$, where $\mathcal{K}_A(x, y)$ denotes the set of paths from x to y in A.
- Let $\omega = [x_0, \dots, x_n] \in \mathcal{K}_A(x_0, x_n)$. By the Markov property, we can write

$$\mathbf{P}^{x_0}\{S[0, n] = \omega\} = \mathbf{P}^{x_0}\{S_0 = x_0, \dots, S_n = x_n\} = p(\omega).$$

Loop-Erasure

Let $\mathcal{W}_A(x, y)$ denote the set of SAWs from x to y in A.

Definition (Loop-erasure)

We define a deterministic procedure called (chronological) loop-erasure

$$\mathrm{LE} : \omega \in \mathcal{K}_A(x, y) \mapsto \eta \in \mathcal{W}_A(x, y).$$

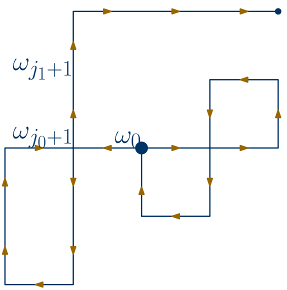
It is given by the following:

1. Let $\omega = [\omega_0, \ldots, \omega_n] \in \mathcal{K}_A(x, y)$, $j_0 = \sup\{k\,;\, \omega_k = \omega_0\}$, and $\eta_0 = \omega_0 = \omega_{j_0}$.
2. If $j_i < n$, set $j_{i+1} = \sup\{k\,;\, \omega_k = \omega_{j_i+1}\}$, and $\eta_{j+1} = \omega_{j_i+1}(= \omega_{j_{i+1}})$. Recursively apply this procedure until $j_m = n$.
3. Set $\mathrm{LE}(\omega) = \eta = [\eta_0, \ldots, \eta_m]$.

Loop-Erasure

Definition (Loop-erasure)

1. Let $\omega = [\omega_0, \ldots, \omega_n] \in \mathcal{K}_A(x, y)$, $j_0 = \sup\{k\,;\,\omega_k = \omega_0\}$, and $\eta_0 = \omega_0 = \omega_{j_0}$.
2. If $j_i < n$, set $j_{i+1} = \sup\{k\,;\,\omega_k = \omega_{j_i+1}\}$, and $\eta_{j+1} = \omega_{j_i+1} (= \omega_{j_{i+1}})$. Recursively apply this procedure until $j_m = n$.
3. Set $\mathrm{LE}(\omega) = \eta = [\eta_0, \ldots, \eta_m]$.

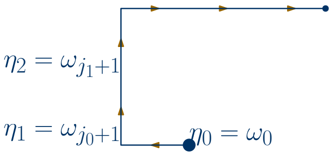


$j_0 = 8, j_1 = 15, j_2 = 16$

Loop-Erasure

Definition (Loop-erasure)

1. Let $\omega = [\omega_0, \ldots, \omega_n] \in \mathcal{K}_A(x, y)$, $j_0 = \sup\{k \,;\, \omega_k = \omega_0\}$, and $\eta_0 = \omega_0 = \omega_{j_0}$.
2. If $j_i < n$, set $j_{i+1} = \sup\{k \,;\, \omega_k = \omega_{j_i+1}\}$, and $\eta_{j+1} = \omega_{j_i+1} (= \omega_{j_{i+1}})$. Recursively apply this procedure until $j_m = n$.
3. Set $\mathrm{LE}(\omega) = \eta = [\eta_0, \ldots, \eta_m]$.



Loop-Erasure

What is the probability that η is the loop-erasure of the path of a Markov chain. That is, what is $p(\mathrm{LE}^{-1}(\eta))$?

Theorem

Let $\eta = [x_0, \ldots, x_m, x_{m+1}]$*, where* $x_{m+1} \in \partial A$*.*

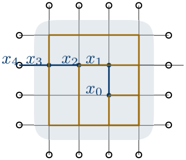
$$p(\mathrm{LE}^{-1}(\eta)) = p(\eta) \prod_{j=0}^{m} p(\mathcal{K}_{A_j}(x_j, x_j)), \textit{ where } A_j = A \setminus \{x_0, \ldots, x_{j-1}\}.$$

The product is a measure of all possible combinations of loops erased. We write

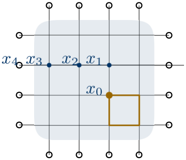
$$F_\eta(A) = \prod_{j=0}^{m} p(\mathcal{K}_{A_j}(x_j, x_j)).$$

Loop-Erasure

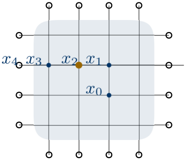
$$\omega = \ell_0 \oplus [x_0, x_1] \oplus \ell_1 \oplus [x_1, x_2] \oplus \ell_2 \oplus [x_2, x_3] \oplus \ell_3 \oplus [x_3, x_4]$$



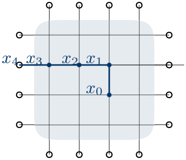
Whole path ω



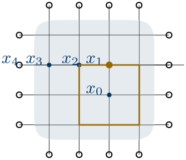
Loop ℓ_0 at x_0



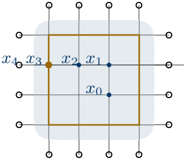
Loop ℓ_2 at x_2



Loop-erasure η



Loop ℓ_1 at x_1



Loop ℓ_3 at x_3

Loop-Erasure

Proof (sketch).

Given some path $\omega \in LE^{-1}(\eta)$, we have the unique decomposition

$$\omega = \ell_0 \oplus [x_0, x_1] \oplus \ell_1 \oplus [x_1, x_2] \oplus \cdots \oplus [x_{m-1}, x_m] \oplus \ell_m \oplus [x_m, x_{m+1}],$$

where $\ell_j \in \mathcal{K}_{A_j}(x_j, x_j)$. From this, we obtain $p(\omega) = p(\eta)p(\ell_0)\dots p(\ell_m)$. Summing over all possible choices of the loops $\ell_0, \dots \ell_m$, we arrive at

$$p(\mathrm{LE}^{-1}(\eta)) = p(\eta) \prod_{j=0}^{m} p(\mathcal{K}_{A_j}(x_j, x_j)).$$

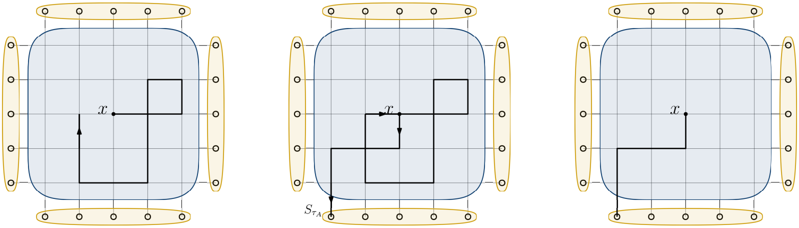
□

LERW in Finite Sets

Definition

The LERW from x to ∂A in A is the path $LE(S[0,\tau_A])$, given by the probability measure

$$\hat{p}_A(\eta) = \mathbf{P}^x \{\mathrm{LE}(S[0,\tau_A]) = \eta\} = \mathbf{P}^x \left\{S[0,\tau_A] \in \mathrm{LE}^{-1}(\eta)\right\}.$$

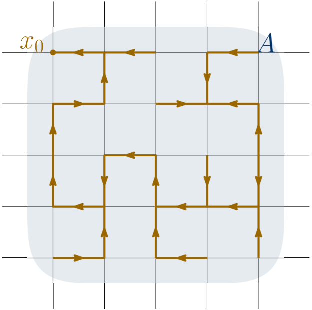


Proposition

$$\hat{p}_A(\eta) = p(\eta) F_\eta(A).$$

Spanning Trees

- A tree T of A is a collection of edges which contains no loops (cycles) and the restriction of the graph A to T is connected.
- The tree T is a spanning tree of A if every vertex in A is incident on some edge in T.
- Given a root x_0, $(T; x_0)$ is a rooted tree for which we can define an orientation of T, where each edge points toward x_0.



Uniform Spanning Tree

We define the weight of T with respect to x_0 by

$$p(T; x_0) = \prod_{e \in T} p(\vec{e}),$$

where $\vec{e}$ is a directed edge with the orientation given by the tree.

Definition (uniform spanning tree)

The uniform spanning tree (UST) on A is random spanning tree with the probability measure μ_A on the set $\mathfrak{T}$ of spanning trees of A given by

$$\mu_A(T) = \frac{p(T; x_0)}{\sum_{T \in \mathfrak{T}} p(T; x_0)}.$$

This is the uniform distribution on $\mathfrak{T}$ if p is given by the simple random walk on A.

Wilson's Algorithm

Wilson's algorithm builds a random spanning tree from LERW.

Definition (Wilson's algorithm)

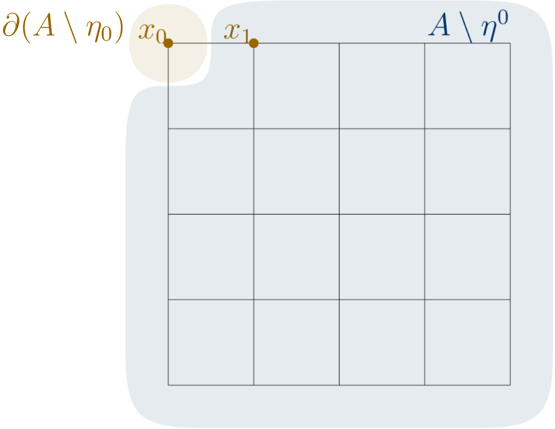
Let $A, \mathbf{P}, x_0 \in A$. Initialise $\mathcal{T}_0 = \emptyset$, $\eta_0 = [x_0]$.

1. Run a LERW in $A_0 = A \setminus \eta^0$ from x_1 to $\partial A_0 = x_0$ to get a path η^1. Add η^1 to the tree by defining $\mathcal{T}_1 = \mathcal{T}_0 \cup \eta^1$ and $A_1 = A_0 \setminus \mathcal{T}_1$.
2. Recursively, if A_k is empty, then we have a spanning tree $\mathcal{T} \stackrel{\text{def}}{=} \mathcal{T}_k$. STOP.
3. Otherwise, let j denote the smallest index with $x_j \notin A_k$. Run a LERW η in A_k from x_j to $\mathcal{T}_k$. Add η to the tree $\mathcal{T}_k$ to form $\mathcal{T}_{k+1} \stackrel{\text{def}}{=} T_k \cup \eta$ and let $A_{k+1} = A_k \setminus \mathcal{T}_{k+1}$, i.e. the vertices that are not yet connected to the tree.

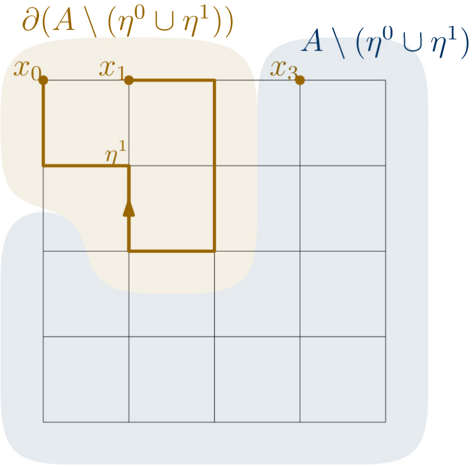
Theorem

Wilson's Algorithm generates spanning trees in the same distribution as the uniform spanning tree.

Wilson's Algorithm

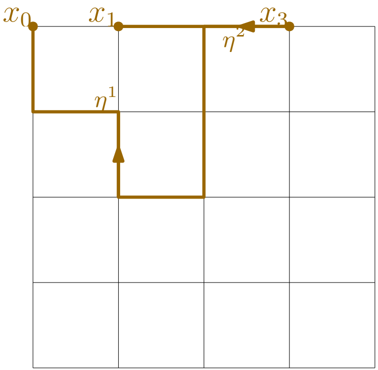


Wilson's Algorithm



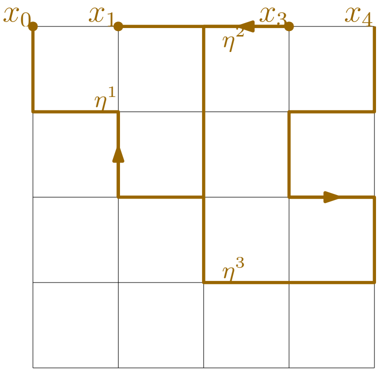
$p(\eta^1) F_{\eta^1}(A \setminus \eta^0)$

Wilson's Algorithm



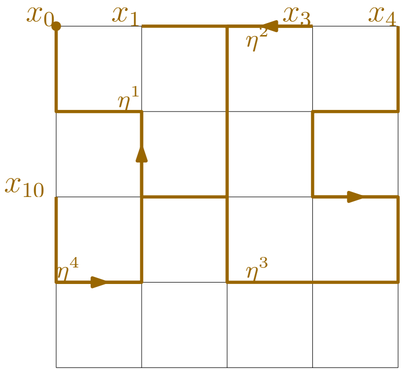
$$p(\eta^1)F_{\eta^1}(A \setminus \eta^0)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1))$$

Wilson's Algorithm



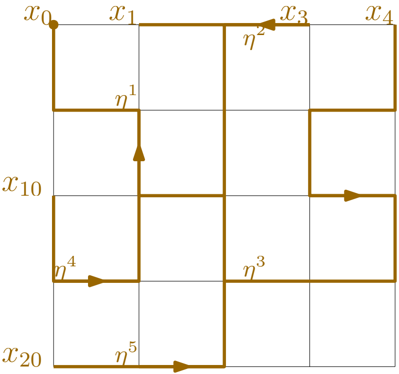
$$p(\eta^1)F_{\eta^1}(A \setminus \eta^0)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1))p(\eta^3)F_{\eta^3}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2))$$

Wilson's Algorithm



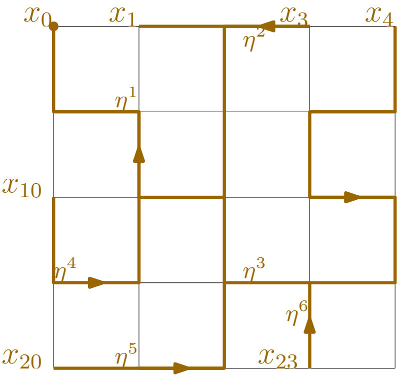
$p(\eta^1)F_{\eta^1}(A \setminus \eta^0)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1))p(\eta^3)F_{\eta^3}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2))p(\eta^4)F_{\eta^4}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3))$

Wilson's Algorithm



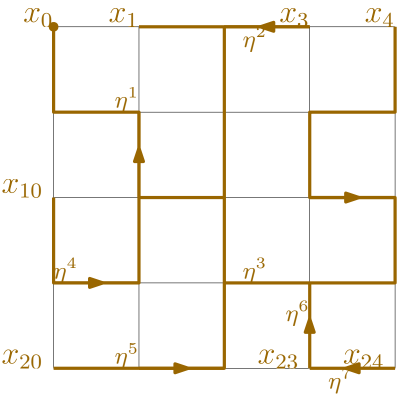
$p(\eta^1)F_{\eta^1}(A \setminus \eta^0)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1))p(\eta^3)F_{\eta^3}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2))p(\eta^4)F_{\eta^4}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3))p(\eta^5)F_{\eta^5}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3 \cup \eta^4))$

Wilson's Algorithm



$p(\eta^1)F_{\eta^1}(A \setminus \eta^0)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1))p(\eta^3)F_{\eta^3}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2))p(\eta^4)F_{\eta^4}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3))p(\eta^5)F_{\eta^5}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3 \cup \eta^4))p(\eta^6)F_{\eta^6}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3 \cup \eta^4 \cup \eta^5))$

Wilson's Algorithm



$p(\eta^1)F_{\eta^1}(A \setminus \eta^0)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1))p(\eta^3)F_{\eta^3}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2))p(\eta^4)F_{\eta^4}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3))p(\eta^5)F_{\eta^5}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3 \cup \eta^4))p(\eta^6)F_{\eta^6}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3 \cup \eta^4 \cup \eta^5))p(\eta^7)F_{\eta^7}(A \setminus (\eta^0 \cup \eta^1 \cup \eta^2 \cup \eta^3 \cup \eta^4 \cup \eta^5 \cup \eta^6))$

Wilson's Algorithm Generates UST

Proof (sketch).

The probability that T is chosen is

$$p(\eta^1)F_{\eta^1}(A)p(\eta^2)F_{\eta^2}(A \setminus (\eta^0 \cup \eta^1)) \cdots p(\eta^k)F_{\eta^k}(A \setminus (\eta^1 \cup \cdots \cup \eta^{k-1})).$$

Note that vertex sets, $A = \eta^1 \cup \cdots \cup \eta^{k-1} \cup \eta^k$. Hence, the above is equal to $p(T; x_0)F_A(A)$. Therefore, the probability that T is chosen is

$$\frac{p(T; x_0)F_A(A)}{\sum_{T \in \mathfrak{T}} p(T; x_0)F_A(A)} = \frac{p(T; x_0)}{\sum_{T \in \mathfrak{T}} p(T; x_0)}.$$

□

Not in this presentation

- This presentation was very combinatorial, only includes LERW on finite graphs.
- We can define LERW as $\text{LE}(S[0,\infty))$ for transient Markov chains on infinite graphs, e.g. simple random walk on $\mathbb{Z}^d$ for $d \geq 3$.
- LERW can only be defined for some recurrent Markov chains: $\text{LE}(S[0,\infty))$ is not well-defined since

$$\sup\{k\,;\,\omega_k = \omega_0\}$$

does not exist.