



**HAL**  
open science

# Fast construction of self-avoiding polygons and efficient evaluation of closed walk fractions on the square lattice

Jean Fromentin, P.-L Giscard, Yohan Hosten

► **To cite this version:**

Jean Fromentin, P.-L Giscard, Yohan Hosten. Fast construction of self-avoiding polygons and efficient evaluation of closed walk fractions on the square lattice. 2024. hal-04842116

**HAL Id: hal-04842116**

**<https://hal.science/hal-04842116v1>**

Preprint submitted on 17 Dec 2024

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Fast construction of self-avoiding polygons and efficient evaluation of closed walk fractions on the square lattice

Jean Fromentin<sup>a</sup>, Pierre-Louis Giscard<sup>a</sup>, Yohan Hosten<sup>a</sup>

<sup>a</sup>*Université du Littoral Côte d'Opale, UR 2597, LMPA, Laboratoire de Mathématiques Pures et Appliquées Joseph Liouville, 50 rue F. Buisson, F-62100, Calais, France*

---

## Abstract

We build upon a recent theoretical breakthrough by employing novel algorithms to accurately compute the fractions  $F_p$  of all closed walks on the infinite square lattice whose the last erased loop corresponds is any one of the 762,207,869,373 self-avoiding polygons  $p$  of length at most 38. Prior to this work, only 6 values of  $F_p$  had been calculated in the literature. The main computational engine uses efficient algorithms for both the construction of self-avoiding polygons and the precise evaluation of the lattice Green's function. Based on our results, we propose two conjectures: one regarding the asymptotic behavior of sums of  $F_p$ , and another concerning the value of  $F_p$  when  $p$  is a large square. We provide strong theoretical arguments supporting the second conjecture. Furthermore, the algorithms we introduce are not limited to the square lattice and can, in principle, be extended to any vertex-transitive infinite lattice. In establishing this extension, we resolve two open questions related to the triangular lattice Green's function.

*Keywords:* Self-avoiding polygon, computational combinatorics, loop-erasing, discrete version of the Schramm-Loewner Evolution, infinite resistor lattice

---

## 1. Introduction

In a series of seminal works on random generation of self-avoiding paths, G. Lawler introduced the procedure of chronological loop-erasing from a random walk [? ?]. First conceived in the context of percolation theory to randomly generate simple paths—walks where all vertices are distinct—from a sample of random walks, the procedure consists of a chronological removal of cycles (called loops in Lawler's original work) as one walks along on the graph: consider for instance the complete graph on 4 vertices and label these with integers 1 through 4. Walking along the path  $1 \rightarrow 2 \rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 1 \rightarrow 3$  on the graph and removing cycles whenever they appear, we are left with the simple path  $1 \rightarrow 3$  after having successively 'erased' the cycles  $1 \rightarrow 2 \rightarrow 1$ , then  $3 \rightarrow 4 \rightarrow 3$  and finally  $1 \rightarrow 3 \rightarrow 1$ . Note how  $1 \rightarrow 3 \rightarrow 1$  does not appear contiguously in the original walk. Indeed, the edges traversed by the last erased loop might be very far apart from one another in the original walk. Once terminated, Lawler's loop-erasing has eliminated a set of cycles, all of whose internal vertices are distinct, leaving a possibly trivial walk-skeleton behind. If the initial walk was itself a cycle, this skeleton is the empty walk on the initial vertex (also called length-0 walk) and otherwise it is a simple path (also called self-avoiding path). This procedure yields a natural probability measure on self-avoiding paths: it is the probability of obtaining such a path as the self-avoiding skeleton left after erasing all loops from a random walk. For Brownian walks on infinite planar lattices under the continuum limit where edge lengths are sent to 0, this procedure is now known to produce a well-defined, conformally

---

*Email addresses:* [jean.fromentin@univ-littoral.fr](mailto:jean.fromentin@univ-littoral.fr) (Jean Fromentin), [giscard@univ-littoral.fr](mailto:giscard@univ-littoral.fr) (Pierre-Louis Giscard), [yohan.hosten@math.cnrs.fr](mailto:yohan.hosten@math.cnrs.fr) (Yohan Hosten)

invariant probability distribution on self-avoiding paths called the Schramm-Loewner Evolution (SLE) with parameter  $\kappa = 2$  [? ?]. Used on closed random walks (i.e. random walks with identical start and end points) the same procedure leads again to SLE<sub>2</sub> law on self-avoiding polygons. Here the self-avoiding polygon generated from the closed random walk is the very last loop eliminated by the procedure. While much effort has been expended in the study of SLE, the equivalent discrete problems, that is on infinite lattices with *finite edge size* have not been studied in nearly as much details and are comparatively less understood.

Consider an infinite, vertex transitive lattice with edges of unit size. The quantity that plays the role of SLE<sub>2</sub> in this discrete setting is the fraction  $F_p$  of all closed walks whose last erased loop is a chosen self-avoiding polygon  $p$ . By definition,  $0 < F_p < 1$ . Furthermore, since the last erased loop of a walk is unique, the sum of  $F_p$  over all self-avoiding polygons  $p$  is 1,

$$\sum_{p: \text{SAP}} F_p = 1, \quad (1)$$

confirming that the  $F_p$  have a probabilistic interpretation [?]. Note that in this sum polygon orientation is retained and the starting point of a polygon is fixed so that, e.g., on the square lattice the  $1 \times 1$  square is counted 8 times. Beyond these basic assertions, very few results have been established concerning the  $F_p$  numbers. In terms of explicit values, mappings to Abelian sand-pile models have permitted the calculation of  $F_p$  for only the six shortest self-avoiding polygons on the infinite square lattice, owing to the mapping's complexity and ensuing lengthy computations [? ?]. Regarding sums of  $F_p$  values, it is not known how fast the sum of Eq. (1) converges to 1 as the length  $\ell$  of the self-avoiding polygons  $p$  considered goes to infinity  $\ell \rightarrow \infty$ . Yet, one of the authors has shown in [?] that this question is tied to the long open problem of determining the asymptotic growth of the number of self-avoiding polygons of length  $\ell \rightarrow \infty$ .

In this work, we build on a recently derived analytical formula for the fractions  $F_p$  [?], employing advanced computational techniques and novel algorithms to explicitly construct all 762, 207, 869, 373 self-avoiding polygons of length at most 38 on the infinite square lattice and evaluate their corresponding  $F_p$ . Additionally, we develop an algorithm to accurately compute individual  $F_p$  values for self-avoiding polygons of length up to 1000. All source code is available for download [?]. Thanks to these progresses we can conjecture that the sum in Eq. (1) converges to 1 as follows,

$$\sum_{\ell' \leq \ell} \sum_{p \in \text{SAP}_{\ell'}} F_p = 1 - \ell^{-3/5} + O(\ell^{-1}).$$

where  $\text{SAP}_{\ell'}$  is the set of self-avoiding polygons of length  $\ell'$  and starting from a fixed vertex  $O$  on the infinite square lattice. In addition, we can also conjecture that the polygon of length  $4L$  with the largest  $F_p$  value among all polygons of the same length is the  $L \times L$  square  $\text{Sq}_{L \times L}$ . Furthermore we present solid, though incomplete, analytical arguments supporting the conjecture that

$$F_{\text{Sq}_{L \times L}} = \left(\sqrt{2} - 1\right)^{4L} + O(\text{polynom}(L)),$$

which is also supported by numerical results. For the sake of concreteness we work on the square lattice, but our methods are generally valid on all vertex-transitive lattices. In the final section, we solve two open problems pertaining to the Green's function of the triangular lattice.

### 1.1. Fraction $F_p$ of closed walks with last erased polygon $p$

Let  $G$  be an infinite vertex-transitive lattice,  $A_G$  its adjacency matrix and  $\Lambda$  its unique dominant eigenvalue. Let  $p$  be a self-avoiding polygon (SAP) or self-avoiding walk (SAW) on  $G$  of length  $\ell(p)$ .

Let  $\mathcal{N}(p)$  be the distance-one neighborhood of  $p$ , that is, the set of vertices of  $G$  that are at distance *at most* one from any vertex visited by  $p$  on  $G$  (see Fig. 1). Let  $E_{\mathcal{N}(p)}$  be the set of all edges of  $G$  with both ends in  $\mathcal{N}(p)$ . Let  $\mathbf{B}_p$  be the adjacency matrix of the graph  $G_{\mathcal{N}(p)} := (\mathcal{N}(p), E_{\mathcal{N}(p)})$ . This graph is finite provided  $\ell(p) < \infty$  and so its adjacency matrix is of finite size  $\mathcal{N}(p) \times \mathcal{N}(p)$ . In addition, the size of this neighborhood cannot be too-large as compared to  $\ell(p)$ . Indeed, since  $G$  is vertex-transitive it is regular and its dominant eigenvalue is the degree of all its vertices. This implies that  $|\mathcal{N}(p)| \leq (\Lambda - 1)\ell(p)$ .

Let  $\mathbf{R}(z) := (\mathbf{I} - z\mathbf{A}_G)^{-1}$  be the resolvent of  $\mathbf{A}_G$  and let  $\mathbf{P}_\Lambda$  be the projector onto the eigenspace of  $\mathbf{A}_G$  associated with eigenvalue  $\Lambda$ . Define [? ]

$$\mathbf{C} := \lim_{z \rightarrow 1/\Lambda^-} (\mathbf{I} - \mathbf{P}_\Lambda)\mathbf{R}(z). \quad (2)$$

This matrix is directly related with the matrix  $\mathbf{r}$  describing the resistance between pairs of points on the infinite resistor lattice with structure  $G$  and with each edge representing a resistor of 1 Ohm. On infinite regular lattices we have the relation [? ]

$$\mathbf{C} = -\frac{1}{2}\mathbf{r}. \quad (3)$$

The value  $C_{O,v_i}$  between a vertex  $v_i$  of the lattice and the origin is the difference of potentials between the origin and that vertex; while the resistance between these points is, by symmetry, half the potential difference between them. Research on the infinite resistor lattice problem [? ? ? ] together with Equation (3) mean that  $\mathbf{C}$  is known for most vertex-transitive lattices. On the square lattice,  $\mathbf{C}$  is explicitly given by

$$C_{v_i,v_j} = -\frac{1}{\pi} \int_0^\infty \frac{1}{\tau} \left( 1 - \left( \frac{\tau - \mathbf{i}}{\tau + \mathbf{i}} \right)^{x_{ij} - y_{ij}} \left( \frac{\tau - 1}{\tau + 1} \right)^{x_{ij} + y_{ij}} \right) d\tau, \quad (4)$$

where  $\mathbf{i}^2 = -1$ ,  $x_{ij} = |x_{v_i} - x_{v_j}|$  and  $y_{ij} = |y_{v_i} - y_{v_j}|$  are the distance along  $x$  and  $y$  between vertices  $v_i$  and  $v_j$ , respectively. In this expression we have index entries of the matrix  $\mathbf{C}$  by the lattice vertices  $v_i$ . While, in principle, this formula provides  $\mathbf{C}$  explicitly, it is ill-suited to large scale numerical computations owing to the integration. We will present a method to bypass this issue in Section 2.

Let  $\mathbf{C}_p := \mathbf{C}|_{\mathcal{N}(p)}$  be the restriction of  $\mathbf{C}$  to the distance-one neighborhood of  $p$ . In other terms,  $(\mathbf{C}_p)_{v_i,v_j} := C_{v_i,v_j}$  if and only if  $v_i, v_j \in \mathcal{N}(p)$ . The matrix  $\mathbf{C}_p$  is, by definition, finite and of size  $\mathcal{N}(p) \times \mathcal{N}(p)$ .

**Theorem 1.1** ([? ]). *Let  $p$  be a self-avoiding polygon on  $G$  and let  $\mathbf{Id}$  be the  $\mathcal{N}(p) \times \mathcal{N}(p)$  identity matrix. Let  $\text{deg}_p = \text{diag}(\mathbf{B}_p^2)$  be the column vector of vertex degrees on  $G_{\mathcal{N}(p)}$  and let  $\mathbf{1}$  be the  $\mathcal{N}(p) \times 1$  column vector full of 1. Then the asymptotic fraction  $0 < F_p < 1$  of all closed walks on the infinite lattice  $G$  whose last erased loop is  $p$  is well defined and given by*

$$F_p = \frac{1}{\Lambda^{\ell(p)}} \times \frac{1}{\Lambda} \text{deg}_p^T \cdot \text{adj} \left( \mathbf{Id} + \frac{1}{\Lambda} \mathbf{C}_p \cdot \mathbf{B}_p \right) \cdot \mathbf{1}, \quad (5)$$

where  $\text{adj}(\mathbf{M})$  designates the adjugate of a matrix  $\mathbf{M}$ . Furthermore  $F_p \in \mathbb{Q}[\chi]$ , with  $\chi$  an irrational number depending on the lattice ( $\chi = 1/\pi$  for the square lattice).

**Remark 1.** *In spite of  $G$  being infinite, Eq. (5) yields a well defined number as both the  $\mathbf{C}_p$  and  $\mathbf{B}_p$  matrices are of finite size.*

**Remark 2.** *By Theorem 1.1, each  $F_p$  value is accessible only if both the matrix  $\mathbf{C}$  and the self-avoiding polygon  $p$  are known. In particular, we emphasize that since  $\mathbf{B}_p$  is required to evaluate  $F_p$ , the self-avoiding polygon  $p$  must be constructed explicitly. This represents far more information per polygon than had been made available even by the best enumerating techniques [? ? ].*

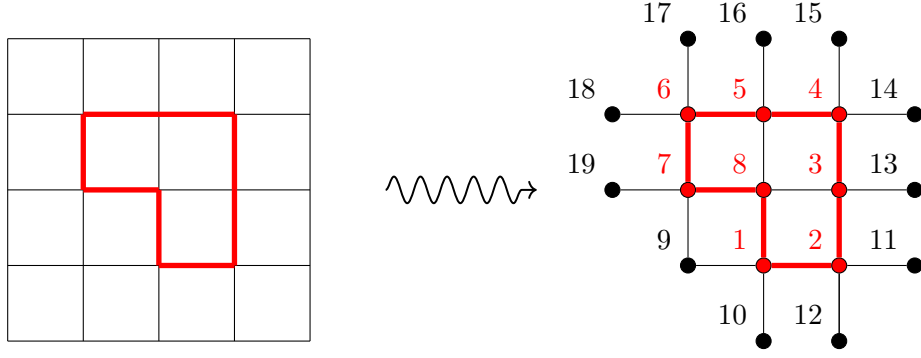


Figure 1: A self-avoiding polygon on the square lattice and its distance-one neighbors. The matrix  $B_{\square}$  is the adjacency matrix of the graph on the right hand side.

As an illustrative example of the Theorem, consider the corner-shape self-avoiding polygon  $\square$  shown in Figure 1. We have  $|\mathcal{N}(\square)| = 19$  so the sieving result providing the exact fraction  $F_{\square}$  involves the adjugate of a  $19 \times 19$  matrix. By construction  $B_{\square}$  and  $C_{\square}$  are symmetric and  $C_{\square}$  further presents the same symmetries as the underlying square lattice  $G$ . This reduces the number of distinct entries of  $C_p$  that need to be computed. In the concrete case of  $\square$ , labeling the vertices of  $G_{\mathcal{N}(\square)}$  as in Fig. 1 (right), the matrix  $C_{\square}$  is given by

$$C_{\square} = \begin{pmatrix} C_1 & C_2 & C_3 \\ C_2 & C_4 & C_5 \\ C_3 & C_5 & C_6 \end{pmatrix},$$

where the  $C_j$  matrices are

- $C_1$  is the  $9 \times 9$  matrix given by

$$C_1 = \begin{pmatrix} 0 & -1 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & -1 & -1 \\ -1 & 0 & -1 & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & \frac{8}{\pi} - 4 \\ -\frac{4}{\pi} & -1 & 0 & -1 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & -1 & 1 - \frac{8}{\pi} \\ 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & -1 & 0 & -1 & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & -\frac{16}{3\pi} \\ \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & -1 & 0 & -1 & -\frac{4}{\pi} & -1 & 1 - \frac{8}{\pi} \\ 1 - \frac{8}{\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & -1 & 0 & -1 & -\frac{4}{\pi} & \frac{8}{\pi} - 4 \\ -\frac{4}{\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & -1 & 0 & -1 & -1 \\ -1 & -\frac{4}{\pi} & -1 & -\frac{4}{\pi} & -1 & -\frac{4}{\pi} & -1 & 0 & -\frac{4}{\pi} \\ -1 & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & -1 & -\frac{4}{\pi} & 0 \end{pmatrix};$$

- $C_2$  is the  $9 \times 1$  vector given by

$$C_2^T = \left(-1 \quad -\frac{4}{\pi} \quad 1 - \frac{8}{\pi} \quad 8 - \frac{92}{3\pi} \quad \frac{48}{\pi} - 17 \quad 8 - \frac{92}{3\pi} \quad 1 - \frac{8}{\pi} \quad \frac{8}{\pi} - 4 \quad -\frac{4}{\pi}\right);$$

- $C_3$  is the  $9 \times 9$  matrix

$$C_3 = \begin{pmatrix} \frac{8}{\pi} - 4 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} & -\frac{16}{3\pi} & 8 - \frac{92}{3\pi} & \frac{48}{\pi} - 17 & 8 - \frac{92}{3\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} \\ -1 & -1 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} & \frac{48}{\pi} - 17 & 8 - \frac{92}{3\pi} & -1 - \frac{8}{\pi} & -1 - \frac{8}{\pi} & 8 - \frac{92}{3\pi} \\ -\frac{4}{\pi} & \frac{8}{\pi} - 4 & -1 & -\frac{4}{\pi} & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & -\frac{16}{3\pi} & 8 - \frac{92}{3\pi} & \frac{48}{\pi} - 17 \\ 1 - \frac{8}{\pi} & \frac{48}{\pi} - 17 & -\frac{4}{\pi} & -1 & -1 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} & \frac{48}{\pi} - 17 & 8 - \frac{92}{3\pi} \\ -\frac{16}{3\pi} & 8 - \frac{92}{3\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & -\frac{4}{\pi} & -1 & -\frac{4}{\pi} & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} \\ -1 - \frac{8}{3\pi} & -1 - \frac{8}{3\pi} & 8 - \frac{92}{3\pi} & \frac{48}{\pi} - 17 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & -1 & -1 & -\frac{4}{\pi} \\ 8 - \frac{92}{3\pi} & -\frac{16}{3\pi} & \frac{48}{\pi} - 17 & 8 - \frac{92}{3\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & -\frac{4}{\pi} & -1 \\ 1 - \frac{8}{\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 & 1 - \frac{8}{\pi} & 1 - \frac{8}{\pi} & \frac{8}{\pi} - 4 \\ \frac{48}{\pi} - 17 & 1 - \frac{8}{\pi} & 8 - \frac{92}{3\pi} & -1 - \frac{8}{3\pi} & -1 - \frac{8}{3\pi} & 8 - \frac{92}{3\pi} & \frac{48}{\pi} - 17 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} \end{pmatrix};$$

- $C_4 = 0$ ;
- $C_5$  is the  $1 \times 9$  vector

$$C_5 = \left( 1 - \frac{8}{\pi} \quad -1 \quad -\frac{16}{3\pi} \quad -1 - \frac{8}{3\pi} \quad 49 - \frac{160}{\pi} \quad \frac{736}{3\pi} - 80 \quad 49 - \frac{160}{\pi} \quad -1 - \frac{8}{3\pi} \quad -\frac{16}{3\pi} \right);$$

- $C_6$  is the  $9 \times 9$  matrix

$$C_6 = \begin{pmatrix} 0 & -\frac{4}{\pi} & -1 & \frac{8}{\pi} - 4 & 8 - \frac{92}{3\pi} & -1 - \frac{8}{3\pi} & -\frac{92}{15\pi} & \frac{472}{15\pi} - 12 & 49 - \frac{160}{\pi} \\ -\frac{4}{\pi} & 0 & 1 - \frac{8}{\pi} & 8 - \frac{92}{3\pi} & \frac{736}{3\pi} - 80 & 49 - \frac{160}{\pi} & \frac{472}{15\pi} - 12 & -\frac{92}{15\pi} & -1 - \frac{8}{3\pi} \\ -1 & 1 - \frac{8}{\pi} & 0 & -1 & 1 - \frac{8}{\pi} & -\frac{16}{3\pi} & -1 - \frac{8}{3\pi} & 49 - \frac{160}{\pi} & \frac{736}{3\pi} - 80 \\ \frac{8}{\pi} - 4 & 8 - \frac{92}{3\pi} & -1 & 0 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} & 8 - \frac{92}{3\pi} & \frac{736}{3\pi} - 80 & 49 - \frac{160}{\pi} \\ 8 - \frac{92}{3\pi} & \frac{736}{3\pi} - 80 & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & 0 & -1 & \frac{8}{\pi} - 4 & 8 - \frac{92}{3\pi} & -1 - \frac{8}{3\pi} \\ -1 - \frac{8}{3\pi} & 49 - \frac{160}{\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} & -1 & 0 & -1 & 1 - \frac{8}{\pi} & -\frac{16}{3\pi} \\ -\frac{92}{15\pi} & \frac{472}{15\pi} - 12 & -1 - \frac{8}{3\pi} & 8 - \frac{92}{3\pi} & \frac{8}{\pi} - 4 & -1 & 0 & -\frac{4}{\pi} & 1 - \frac{8}{\pi} \\ \frac{472}{15\pi} - 12 & -\frac{92}{15\pi} & 49 - \frac{160}{\pi} & \frac{736}{3\pi} - 80 & 8 - \frac{92}{3\pi} & 1 - \frac{8}{\pi} & -\frac{4}{\pi} & 0 & -1 \\ 49 - \frac{160}{\pi} & -1 - \frac{8}{3\pi} & \frac{736}{3\pi} - 80 & 49 - \frac{160}{\pi} & -1 - \frac{8}{3\pi} & -\frac{16}{3\pi} & 1 - \frac{8}{\pi} & -1 & 0 \end{pmatrix}.$$

Now the fraction  $F_{\sqsupset}$  is given by Eq. (5) as

$$F_{\sqsupset} = \frac{1}{576 \pi^6} (3\pi - 8)^2 (8 - \pi) (4 - \pi) (-23\pi^2 + 120\pi - 128) \simeq 3.36 \times 10^{-4},$$

that is, 0.0336% of all closed walks on the infinite square lattice have  $\sqsupset$  as their last erased loop.

### 1.2. Sums of fractions over families of self-avoiding polygons

Fix a vertex  $O$  on the infinite square lattice and denote  $\text{dSAP}_\ell$  the set of self-avoiding polygons of length  $\ell$  from  $O$  to itself. In this set, self-avoiding polygons that differ only through their orientations or through a translation (i.e. changing the starting point) are considered distinct. Let  $\text{SAP}_\ell$  be the set of self-avoiding polygons of length  $\ell$ , up to orientation and translation. Let  $\pi(\ell) = |\text{SAP}_\ell|$ , then  $|\text{dSAP}_\ell| = 2\ell \times \pi(\ell)$  since a self-avoiding polygon has 2 orientations and  $\ell$  possible starting points. Let  $F(\ell)$  be the sum of  $F_p$  values for all  $p \in \text{dSAP}_\ell$ , that is

$$F(\ell) := \sum_{p \in \text{dSAP}_\ell} F_p = 2\ell \sum_{p \in \text{SAP}_\ell} F_p.$$

Of particular interest are the cumulative sums,

$$S(\ell) := \sum_{\ell' \leq \ell} F_{\ell'}.$$

$S(\ell)$  gives the proportions of closed walks on the infinite square lattice whose last erased loop is a self-avoiding polygon of length at most  $\ell$ . The behavior of  $S(\ell)$  as  $\ell \rightarrow \infty$  on the infinite discrete lattices is a determinant signature of the probability law on self-avoiding polygons resulting from loop-erasing. In practice, evaluating the sums  $S(\ell)$  is a major computational challenge because they stem from hundreds of billions of individual  $F_p$  values, each of which, as noted in Remark 2, requires accessing the matrix  $C$  and knowing the self-avoiding polygon  $p$  in full details. In other terms, a numerical study of the behavior of  $S(\ell)$  as  $\ell$  grows requires explicitly constructing all self-avoiding polygons of length  $\ell$  as well as an efficient and accurate method to evaluate the lattice Green's function.

## 2. Efficient computation of the $\mathbf{C}$ matrix

Because of its definition by Eq. (2), the  $\mathbf{C}$  matrix shares many properties with the resolvent  $\mathbf{R}(z) = (\mathbf{I} - z\mathbf{A}_G)^{-1}$  also known as the lattice Green's function. In particular, this implies the existence of recurrence relations between entries of  $\mathbf{C}$ , which we use to calculate  $\mathbf{C}$  in a computationally efficient and numerically stable way.

### 2.1. Recurrence relations

Let  $O := (0, 0)$  denote the origin of the square lattice. For every vertex  $v$ , let the pair  $(i_v, j_v) \in \mathbb{Z}^2$  (or  $(i, j)$  if there is no possible confusion) denote its coordinates and  $\tilde{v}$  the vertex of coordinates  $(j_v, i_v)$  that is, the symmetric of  $v$  with respect to the first bisector. First, observe that  $\mathbf{R}(z)$  satisfies  $\mathbf{R}(z) \cdot (z\mathbf{A}_G) = \mathbf{R}(z) - \mathbf{I}$  and so, by Eq. 2,

$$\begin{aligned} \mathbf{C} \cdot \left( \frac{1}{\Lambda} \mathbf{A}_G \right) &= \lim_{z \rightarrow 1/\Lambda^-} (\mathbf{I} - \mathbf{P}_\Lambda) \mathbf{R}(z) \cdot (z\mathbf{A}_G), \\ &= \lim_{z \rightarrow 1/\Lambda^-} (\mathbf{I} - \mathbf{P}_\Lambda) \mathbf{R}(z) - (\mathbf{I} - \mathbf{P}_\Lambda), \\ &= \mathbf{C} - (\mathbf{I} - \mathbf{P}_\Lambda). \end{aligned} \quad (6)$$

Since  $G$  is regular,  $(\mathbf{P}_\Lambda)_{v,v'} = 1/N \rightarrow 0$  as  $N \rightarrow \infty$ , then the above yields the following recurrence relation between entries of  $\mathbf{C}$ ,

$$\frac{1}{\Lambda} \sum_{v'' \in \mathcal{N}(v')} \mathbf{C}_{v,v''} = \mathbf{C}_{v,v'} - \delta_{v,v'}, \quad (7)$$

here  $\mathcal{N}(v')$  denotes the set of neighbors of vertex  $v'$  on  $G$ , that is the set of vertices at distance 1 of  $v'$ . Multiplying by  $\mathbf{A}$  on the left of  $\mathbf{C}$  in Eq. (6) we obtain the symmetric relation

$$\frac{1}{\Lambda} \sum_{v'' \in \mathcal{N}(v')} \mathbf{C}_{v',v} = \mathbf{C}_{v,v'} - \delta_{v,v'},$$

confirming that  $\mathbf{C} = \mathbf{C}^T$ , as expected from Eq. (4). As the lattice  $G$  is vertex transitive, we may choose  $v$  to be fixed at the origin  $O$  and have  $v'$  move throughout the lattice to determine  $\mathbf{C}$  entirely. Furthermore,  $\mathbf{C}_{v,v'} = \mathbf{C}_{v',v}$ ,  $\mathbf{C}_{O,v} = \mathbf{C}_{O,\tilde{v}}$ ,  $\mathbf{C}_{O,O} = 0$  and all  $\mathbf{C}_{O,(i,i)}$  are well known (see below). Therefore, Eq. (7) is sufficient to determine all entries of  $\mathbf{C}$  by solving linear equations and without computing any integral [? ?]. In the sequel, the coefficient  $\mathbf{C}_{O,v}$  with  $v := (i, j)$  will be denoted by  $c_{i,j}$ , so that we have:

**Proposition 2.1.** *On the square lattice we have  $c_{0,0} = 0$ ,  $c_{0,1} = -1$ ,*

$$c_{i,i} = -\frac{4}{\pi} \sum_{k=1}^i \frac{1}{2k-1} = -\frac{4}{\pi} \sum_{k=0}^{i-1} \frac{1}{2k+1} \quad \text{for } i \geq 1 \quad (8a)$$

$$c_{0,j} = 4c_{0,j-1} - c_{0,j-2} - 2c_{1,j-1} \quad \text{for } j \geq 2 \quad (8b)$$

$$c_{i,j} = 4c_{i,j-1} - c_{i,j-2} - c_{i-1,j-1} - c_{i+1,j-1} \quad \text{for } j > i \geq 1 \quad (8c)$$

As shown by Eq. (8a), the  $c_{i,i}$  values are known exactly and are easily computable. Due to the symmetry  $c_{i,j} = c_{j,i}$ , we only need establish that it is possible to compute  $c_{i,j}$  above the first bisector of the plan to obtain all entries of  $\mathbf{C}$ .

**Lemma 2.2.** *For  $0 \leq i < j$ , it is possible to compute  $c_{i,j}$ .*

*Proof.* We proceed by induction on  $j \geq 1$ .

- If  $j = 1$  then  $i = 0$  and  $c_{1,0} = c_{0,1} = -1$ .
- Let  $j \geq 1$  and assume that for  $0 \leq i < j$ , it is possible to compute  $c_{i,j}$ . We show that for  $0 \leq i < j + 1$ , it is possible to compute  $c_{i,j+1}$ . There are 3 possibilities.
  1. Either  $i = 0$  then  $c_{0,j+1}$  can be computed using Equation (8b) since  $j + 1 \geq 2$ .
  2. Or  $1 \leq i \leq j - 1$  then  $c_{i,j+1}$  can be computed using Equation (8c).
  3. Or  $i = j$  then we use Equation (8c) and the symmetric relation to obtain  $2c_{j,j+1} = 4c_{j,j} - c_{j,j-1} - c_{j-1,j}$ .

Graphically, this iteration implements the following traversal of the square lattice: □

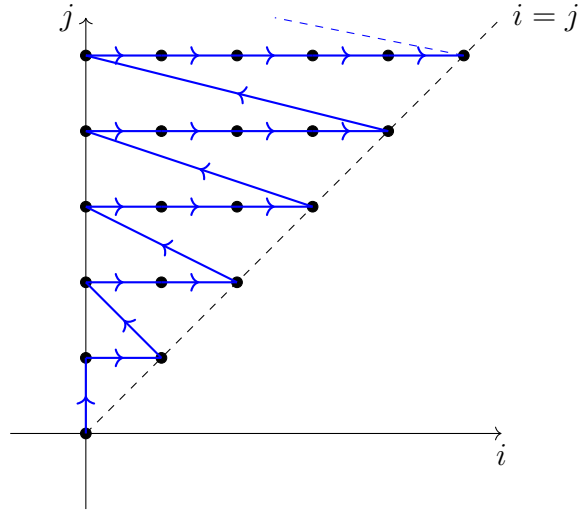


Figure 2: Direction of computations for the calculation of all  $c_{i,j}$  on the square lattice.

While  $C$  is an infinite matrix, for any given self-avoiding polygon  $p$  we need only compute its finite submatrix  $C_p$  to calculate  $F_p$  via Eq. (5). Instead of doing this calculation anew for every  $p$ , in practice, we compute a finite submatrix of  $C$  only once over the finite grid that contains all the self-avoiding polygons to be constructed (see §3.3 for details). The required  $C_p$  matrices are then extracted from this.

## 2.2. Implementation

Firstly remark that two vertices belonging to the distance-one neighborhood  $\mathcal{N}(p)$  of a polygon  $p$  of length  $\ell$ , stand at a distance of at most  $\ell/2 + 2$  from each other. Thus, computations of the  $F_p$  fractions for all self-avoiding polygons of length up to  $\ell$  only necessitates coefficients  $c_{i,j}$  for  $0 \leq i < j \leq \ell/2 + 2$ . Secondly, by Proposition 2.1, the computation of any  $c_{i,j}$  involves a recursion with many intermediate steps and, consequently, is prone to an accumulation of rounding errors. This leads to a poor overall approximation of  $c_{i,j}$ , even when working with *double precision floating point*, i.e. with a precision of  $10^{-16}$ . To bypass this issue, observe that since  $c_{i,j} \in \mathbb{Q}[1/\pi]$ , there is a unique way to write  $c_{i,j} = a_{i,j} + b_{i,j} \times \frac{1}{\pi}$ , where  $a_{i,j}$  and  $b_{i,j}$  are two rational numbers. Computing these rationals exactly is possible so no accumulation of rounding errors can occur. Adapting Proposition 2.1, the rational coefficients are obtained through the following recursion:

1. Set  $a_{0,0} = 0$ ,  $b_{0,0} = 0$ ,  $a_{0,1} = -1$  and  $b_{0,0} = 0$ ;
2. For  $i$  from 1 to  $n$  set  $a_{i,i} = 0$  and compute  $b_{i,i} = b_{i-1,i-1} - 4/(2i - 1)$ ;



- (a) Compute  $a_{0,j} = 4a_{0,j-1} - a_{0,j-2} - 2a_{1,j-1}$  and  $b_{0,j} = 4b_{0,j-1} - b_{0,j-2} - 2b_{1,j-1}$ ;
- (b) For  $i$  from 1 to  $j-2$ , compute  $a_{i,j} = 4a_{i,j-1} - a_{i,j-2} - a_{i-1,j-1} - a_{i+1,j-1}$  and  $b_{i,j} = 4b_{i,j-1} - b_{i,j-2} - b_{i-1,j-1} - b_{i+1,j-1}$ ;
- (c) Compute  $a_{j-2,j} = 2a_{j-1,j-1} - a_{j-1,j-1}$  and  $b_{j-2,j} = 2b_{j-1,j-1} - b_{j-1,j-1}$ .

Once coefficients  $a_{i,j}$  and  $b_{i,j}$  are so obtained, we get a numerical approximation of  $c_{i,j}$ :

$$c_{i,j} \approx \left( a_{i,j} + \frac{1725033}{5419351} \times b_{i,j} \right) + b_{i,j} \times 2.27595720048157 \times 10^{-15},$$

which follows from  $\frac{1}{\pi} \approx \frac{1725033}{5419351} + 2.27595720048157 \times 10^{-15}$ .

Combining these observations and results, and given a maximum length  $\ell$ , we proceed as above to compute  $a_{i,j}$  and  $b_{i,j}$  with  $0 \leq i < j \leq \ell/2 + 2$  once and for all. In a second stage specific to each polygon  $p$ , the  $C_p$  matrices are constructed from the stored rational coefficients. Rational numbers are given by irreducible fractions  $p/q$  of integers, each of which is coded using 64 bits and so must belong to  $\{-2^{63}, \dots, 2^{63} - 1\}$ . If integers outside of that range are needed, an overflow occurs resulting in a missed computation of  $c_{i,j}$ . In practice, we observed that integers coded on 64 bits are enough when computing coefficients  $a_{i,j}$  and  $b_{i,j}$  with  $0 \leq i < j \leq 12$ . This means that we can only deal with self-avoiding polygons of length at most 20. For longer self-avoiding polygons, we must instead use integers coded on 128 bits, allowing for valid computations of  $c_{i,j}$  with  $0 \leq i < j \leq 510$ , thanks to which we can reach self-avoiding polygons of length at most 1016.

### 3. An algorithm for constructing self-avoiding polygons

Once  $C$  is obtained, the second difficulty of Theorem 1.1 is that  $F_p$  depends on the adjacency matrix  $B_p$  of the distance-one neighborhood of the self-avoiding polygon  $p$ . In order to compute large number of  $F_p$  values, we must therefore construct self-avoiding polygons as efficiently as possible.

Let us fix a point  $O$  of the square lattice and a length  $\ell \geq 2$ . The set of all finite path on the square lattice starting at  $O$  will be denoted  $\mathcal{P}$ . To each element of  $\mathcal{P}$  corresponds a finite sequence of steps **Down**, **Left**, **Right** and **Up**. More precisely there is a bijection between elements of  $\mathcal{P}$  and the set  $S^*$  of finite words on the alphabet  $S = \{D, L, R, U\}$ , where D, L, R and U denote steps **Down**, **Left**, **Right** and **Up** respectively. From now on, paths on the square lattice starting at  $O$  will be identified with finite words on the alphabet  $\mathcal{S}$ . The set  $\mathcal{P}$  is then equipped with a tree structure  $\mathcal{T}$ , inherited from that of  $S^*$ . The root of  $\mathcal{T}$  is the empty word  $\varepsilon$  and the sons of a path  $w$  are  $wD$ ,  $wL$ ,  $wR$  and  $wU$ . In order to construct each member of the set  $SAP_\ell$  of all self-avoiding polygons of length  $\ell$ , we will pick out a subtree  $\mathcal{T}_\ell$  of  $\mathcal{T}$  whose leaves at depth  $\ell$  are in bijection with  $SAP_\ell$ .

#### 3.1. Self-avoiding polygon to word

A self-avoiding polygon of length  $\ell$  can be seen as a path starting from one of its vertices. Yet, this correspondence is not one-to-one. For instance, the  $1 \times 1$  square can be represented by 8 words on  $\mathcal{S}$  depending on the starting vertex and the chosen orientation, as depicted on Figure 3,

To avoid multiple constructions of the same self-avoiding polygon, we must fix the starting point and the orientation of the path. Let  $p$  be a self-avoiding polygon. The *base line* of  $p$  is the horizontal line containing the furthest south edge of  $p$ . The *base point* of  $p$  is then the furthest west vertex of  $p$  on its base line, see Figure 4. For the rest of this part, self-avoiding polygons will be drawn on the square lattice in such a way that their base point will be a fixed point  $O$  of the square lattice.

At this point a self-avoiding polygon can be represented by two paths of  $\mathcal{P}$ , depending of the chosen orientation. By construction of the base point these words are of the type  $R \dots D$  and  $U \dots L$ . Hence, forcing the first letter to be a right **R**, we fix the polygon orientation. This leads to:



Since we aim at constructing hundred of billions of self-avoiding polygons, we need a process to check properties (P4) to (P7) as efficiently as possible.

### 3.3. The game board

We construct our  $\ell$ -admissible paths on a *game board*, which can be seen as a finite part of the dual lattice of the square lattice with extra decorations on each cell. The cell containing the base point  $O$  is termed the *base cell* and denoted  $\mathbf{0}$ . We first determine the optimal size of our game board. Denote by  $E_\ell, N_\ell$ , and  $W_\ell$  the  $\ell$ -accepting words that goes furthest east, north and west from the base point  $O$ , respectively. We verify immediately that these words are of the form

$$\begin{aligned} E_\ell &= \underbrace{\mathbf{R} \cdots \mathbf{R}}_{a_\ell} \mathbf{U} \underbrace{\mathbf{L} \cdots \mathbf{L}}_{a_\ell} \mathbf{D} && \text{with } a_\ell = \ell/2 - 1, \\ N_\ell &= \mathbf{R} \underbrace{\mathbf{U} \cdots \mathbf{U}}_{a_\ell} \mathbf{L} \underbrace{\mathbf{D} \cdots \mathbf{D}}_{a_\ell} && \\ W_\ell &= \mathbf{R} \mathbf{U} \mathbf{U} \mathbf{L} \underbrace{\mathbf{L} \cdots \mathbf{L}}_{b_\ell} \mathbf{D} \underbrace{\mathbf{R} \cdots \mathbf{R}}_{b_\ell} \mathbf{D} && \text{with } b_\ell = \ell/2 - 3, \end{aligned}$$

where we consider only even lengths  $\ell$ , since there are no polygon of odd length on the infinite square lattice. It follows that the game board must have at least  $a_\ell$  cells east of the base cell,  $a_\ell$  cells north of it and  $b_\ell - 1$  cells west of the cell that is immediately north of the base cell. It will be useful to make a *border* of forbidden cells to our game board. By definition of base point, any cells west of it are forbidden. Thus the game board has width  $w_\ell = a_\ell + b_\ell + 3 = \ell - 1$  and height  $h_\ell = a_\ell + 3 = \ell/2 + 2$ , see the illustration of Figure 6.

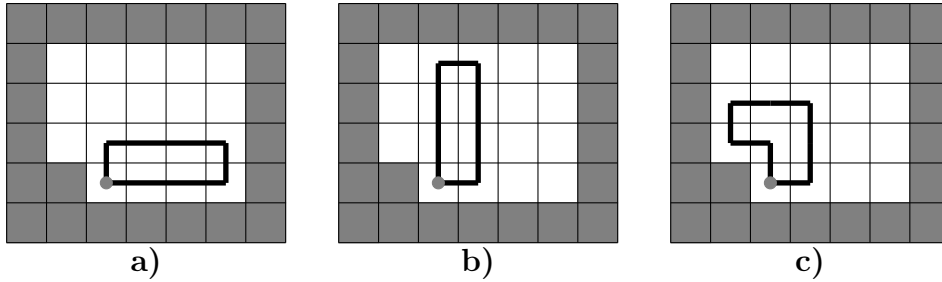


Figure 6: Game board for  $\ell = 8$ . The base cell is depicted with a gray dot. Forbidden cells are in gray. Paths  $E_\ell, N_\ell$  and  $W_\ell$  are drawn on **a)**, **b)** and **c)** respectively.

We identify each cell  $c$  of the game board by a unique integer  $n_c$  in  $\{0, \dots, h_\ell \times w_\ell - 1\}$ : we have  $n_c = w_\ell \times y + x$  for a cell  $c$  at position  $(x, y)$  with the convention that the left-bottom cell has position  $(0, 0)$  and the right-top one has position  $(w_\ell - 1, h_\ell - 1)$ . Consider a non forbidden cell  $c$  and denote by  $c_D, c_L, c_R$  and  $c_U$  the cells visited after a D, L, R and U steps from  $c$ , respectively. Then,

$$n_{c_D} = n_c - w_\ell, \quad n_{c_L} = n_c - 1, \quad n_{c_R} = n_c + 1, \quad \text{and} \quad n_{c_U} = n_c + w_\ell.$$

### 3.4. Cell data

In order to check that a path satisfies property (P7), we add information to the game-board cells as follows. Let  $w$  be a word of length  $t$  on  $\mathcal{S}$ . Starting from the base cell  $\mathbf{0}$ , we construct a sequence of cells  $\phi_w$  of length  $\ell(w)$  accordingly to  $w$ : for  $i \in \{1, \dots, \ell(w)\}$  the cell  $\kappa_w(i)$  is the adjacent cell of  $\kappa_w(i-1)$  with respect to  $w_i$ . Let  $0 \leq t < \ell$  be an integer. In order to be able to complete  $w$  into a closed path of length  $\ell$ , we must complete it with  $\ell - t$  steps linking the cell  $\kappa_w(\ell(w))$  to the base cell  $\mathbf{0}$ . This is only possible if the length of the minimal sequence of steps linking  $\kappa_w(\ell(w))$  to  $\mathbf{0}$  is smaller than  $\ell - t$ .

For each non forbidden cell  $\mathbf{c}$  of the game board, we thus define  $d(\mathbf{c})$  to be the minimal number of steps needed to reach the base cell from  $\mathbf{c}$ . This quantity is the Manhattan distance between  $\mathbf{c}$  and the base cell  $\mathbf{0}$ : if we denote by  $(0, 0)$  and  $(x, y)$  positions of the base cell  $\mathbf{0}$  and  $\mathbf{c}$ , respectively, we have  $d(\mathbf{c}) = |x| + |y|$ . By convention, we set  $d(\mathbf{c}) = +\infty$  for each forbidden cell (in gray in Figure 6). We can now give a precise formulation of (P7) :

**Definition 3.3.** *A word  $w$  of length  $t$  on  $\mathcal{S}$  satisfies (P7) whenever  $t + d(c_w(t)) \leq \ell$  holds.*

Thus, to each cell  $\mathbf{c}$  of the game board we attach two pieces of information:

- the Manhattan distance  $d(\mathbf{c})$  between  $\mathbf{c}$  and the base cell  $\mathbf{0}$ ;
- an integer  $t(\mathbf{c})$ ,  $1 \leq t(\mathbf{c}) \leq \ell$ , that depends on the current word  $w$  being constructed. For  $i \in \{1, \dots, \ell(w)\}$ , the value of  $t(\kappa_w(i))$  is set to  $i$ . Roughly speaking,  $t(\mathbf{c})$  specifies *when* the cell  $\mathbf{c}$  is reached by the word  $w$ .

### 3.5. Initialization

We now describe the main algorithmic engine for the construction of the self-avoiding polygons of a fixed length  $\ell$ . Recall that this is based on the exploration of the tree  $\mathcal{T}_\ell$ , whose leaves are in bijection with  $\text{SAP}_\ell$ . We begin by constructing the game board  $\mathbf{G}$  as described above in § 3.3. We then compute the Manhattan distance  $d(\mathbf{c})$  from the base cell  $\mathbf{0}$  to each cell  $\mathbf{c}$  of  $\mathbf{G}$ . In addition of the game board  $\mathbf{G}$ , we use

1. an array **word** of size  $\ell$  storing the word  $w$  under construction, **word**[ $i$ ] being the  $i$ th letter of  $w$ ;
2. an array **kappa** of size  $\ell$  containing integers in  $\{0, \dots, h_\ell \times w_\ell - 1\}$  or  $+\infty$  (in practice we can replace  $+\infty$  by any integer greater than  $h_\ell \times w_\ell$ );
3. a stack **stack** of triple  $(\mathbf{c}, k, s)$ , where  $\mathbf{c}$  is a cell,  $k$  an integer between 0 and  $\ell$ , and  $s \in \mathcal{S}$  is a step.

We give a complete description of the initialization steps in Algorithm 1.

---

**Algorithm 1** Initialization for the construction of  $\text{SAP}_\ell$ .

---

```

1: procedure INITENUMSAP
2:   Construct the game board  $\mathbf{G}$ 
3:   for each cell  $\mathbf{c}$  of  $\mathbf{G}$  do
4:     if  $\mathbf{c}$  is on the border then
5:        $d(\mathbf{c}) \leftarrow +\infty$ 
6:     else
7:        $d(\mathbf{c}) \leftarrow$  Manhattan distance between  $\mathbf{c}$  and  $\mathbf{0}$ 
8:        $t(\mathbf{c}) \leftarrow -1$ 
9:     end if
10:  end for
11:  word  $\leftarrow$  array steps of length  $\ell$ 
12:  kappa  $\leftarrow$  array of cells of length  $\ell$ 
13:  stack  $\leftarrow$  empty stack
14:  stack.push( $\mathbf{0}, 0, \mathbf{R}$ ) ▷ Start with a step  $\mathbf{R}$  from the base cell  $\mathbf{0}$ , to ensure (P2) and (P4)
15: end procedure

```

---

### 3.6. Exploration of the tree $\mathcal{T}_\ell$

We perform a *depth-first* exploration of tree of  $\mathcal{T}_\ell$  using the stack `stack` which is a container equipped with four operations:

- `stack.empty()` testing if the stack is empty;
- `stack.push(...)` putting an element at the top of the stack;
- `stack.top()` returning the element at the top of the stack;
- `stack.pop()` removing the element at the top of the stack.

Consider for instance the set  $\text{SAP}_8$  of length 8 self-avoiding polygons and suppose that at some point of the exploration of  $\mathcal{T}_8$  we pop the 8-admissible word  $w = \text{RURU}$  from the stack `stack`. As a word,  $w$  can be completed as  $w_1 = w\text{D}$ ,  $w_2 = w\text{L}$ ,  $w_3 = w\text{R}$  or  $w_4 = w\text{U}$ .

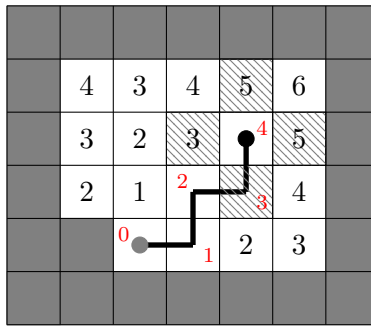


Figure 7: Depth first exploration of the tree  $\mathcal{T}_8$ . The 8-admissible word  $w = \text{RURU}$  of length 4 is represented by the solid black line on the game board. The base cell is that with the gray bullet, while cell  $\kappa_w(4)$  is highlighted with a black bullet. Black numbers in cells  $\mathbf{c}$  stand for  $d(\mathbf{c})$ , red numbers give  $t(\mathbf{c})$ . Shaded cells are those immediately reachable from  $w$  by adding one step. They yield words  $w_1 = w\text{D}$ ,  $w_2 = w\text{L}$ ,  $w_3 = \text{R}$  and  $w_4 = w\text{U}$ .

Using the game board as depicted in Figure 7, we check whether these four words are 8-admissible. Let us denote by  $\mathbf{c}_1, \mathbf{c}_2, \mathbf{c}_3$  and  $\mathbf{c}_4$  the cells reached by  $w_1, w_2, w_3$  and  $w_4$  respectively, *i.e.*,  $\mathbf{c}_i = \kappa_{w_i}(5)$ . In the case of  $w_1$ , noting that  $t(\mathbf{c}_1) = 3$  is lower than the length of  $w_1$ , this word may self-intersect. Remark that  $t(\mathbf{c}_1) = 3 < 5$  is not sufficient to immediately conclude that  $w_1$  self intersects because, for the sake of speed and efficiency, the game board information is not systematically cleared. As a consequence many cells  $\mathbf{c}'$  could share the same value  $t(\mathbf{c}') = t(\mathbf{c}_1)$ , the values taken by  $t$  being correct only for cells visited by the current word. So, to decide if  $w_1$  self intersects the algorithm checks whether cell  $\mathbf{c}_1$  is indeed the third one visited by  $w_1$ . This is what `kappa` is used for: given that  $\kappa_{w_1}(3) = \kappa_w(3) = \mathbf{c}_1$ ,  $\mathbf{c}_1$  is really visited twice and  $w_1$  is rejected because it violates property (P6). Words  $w_2, w_3$  and  $w_4$  are checked similarly and found to satisfy (P6). There remains to verify property (P7) for them. These words are of length 5 leaving only three steps to reach back to the base cell. As the distance information contained in cells  $\mathbf{c}_3$  and  $\mathbf{c}_4$  is 5, which is strictly greater than the three remaining steps, the algorithm rejects words  $w_3$  and  $w_5$  for violating (P7). Hence  $w_2 = w\text{L}$  is the sole 8-admissible word that can be built from  $w$  and  $(\mathbf{c}_2, 5, \text{L})$  is pushed on the stack `stack`. The exploration of  $\mathcal{T}_8$  continues in this fashion until the stack `stack` is empty.

The algorithm for the complete exploration of  $\mathcal{T}_\ell$  is presented in Algorithm 2.

---

**Algorithm 2** Construction of  $SAP_\ell$ .

---

```
1: procedure ENUMSAP
2:   while stack not empty do
3:      $(c, k, s) \leftarrow \text{stack.top}()$ 
4:      $\text{stack.pop}()$ 
5:     APPLY( $c, k, s$ )
6:      $c' \leftarrow$  the cell adjacent to  $c$  with respect to  $s$ 
7:     if  $k = \ell$  then
8:       TREAT(word)
9:     else
10:      for  $s'$  in  $\{D, L, R, U\}$  do
11:        if CANADD( $c', k + 1, s'$ ) then
12:           $\text{stack.push}(c', k + 1, s')$ 
13:        end if
14:      end for
15:    end if
16:  end while
17: end procedure
```

---

▷ In line 5, the current word is augmented with step  $s$ , which corresponds to exploring an edge of  $\mathcal{T}_\ell$ . The updates performed by the code on that occasion are presented in Algorithm 3.

▷ In line 8, the algorithm has reached a leaf of  $\mathcal{T}_\ell$  at depth  $\ell$ . At this point, **word** codes for a valid self-avoiding polygon  $p$  of length  $\ell$ . The function **treat** is called and it implements any desired computations on  $p$ , stores it for future use, etc.

▷ In line 11, the algorithm checks whether the current word augmented with step  $s'$  is  $\ell$ -admissible, a task performed by Algorithm 4. If the answer is positive we push this future exploration on the stack.

---

**Algorithm 3** Apply a new step to the on construction path.

---

```
1: function APPLY( $c, k, s$ )
2:    $t(c) \leftarrow k$ 
3:    $\text{word}[k] \leftarrow s$ 
4:    $\text{kappa}[k] \leftarrow c$ 
5:    $\text{kappa}[k + 1] \leftarrow +\infty$ 
6: end function
```

---

---

**Algorithm 4** Test if we can add the letter  $s$  at position  $k$  in the word under construction

---

```
1: function CANADD( $c, k, s$ )
2:    $c' \leftarrow$  the cell adjacent to  $c$  with respect to  $s$ 
3:   if  $k + d(c') \geq \ell$  then ▷ Checks (P7) and whether a game board boundary is reached
4:     return false
5:   end if
6:   if  $t(c') \leq k$  and  $\text{kappa}[t(c')] = c'$  then ▷ Checks (P6)
7:     return false
8:   end if
9:   return true
10: end function
```

---

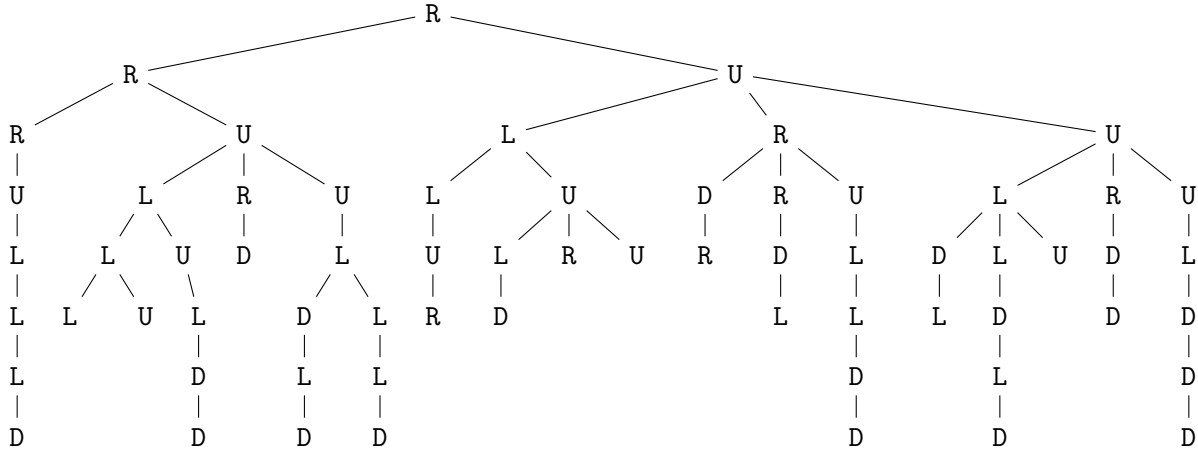


Figure 8: The complete tree  $\mathcal{T}_8$ . To each node  $n$  corresponds the 8-admissible word made of the letters met going from the root of the tree down to node  $n$ . The seven elements of  $\text{SAP}_8$  are the seven leaves at depth 8 of  $\mathcal{T}_8$ .

#### 4. Computational results

Our experiments were carried out on the shared computational platform CALCULCO [? ]. This platform is equipped with 28 nodes for a total of 1852 cores and is designed for distributed computation. In this section we detail the construction of a suite of programs to take advantage of CALCULCO to carry out the computation of the sums  $S(\ell)$  of §1.2 up to  $\ell = 38$ .

##### 4.1. Memory usage and parallelization

Our first task is to construct and store the self-avoiding polygons of a given length  $\ell$ . The process is fully described in Section 3 so we here focus on the storage method. As a self-avoiding polygons of length  $\ell$  is given by a word which we code as an array of steps  $\{D, L, R, U\}$  of length  $\ell$ . Each step can be encoded with two bits :

$$D \rightarrow 00, \quad L \rightarrow 01, \quad R \rightarrow 10, \quad U \rightarrow 11. \tag{9}$$

With this methods a self-avoiding polygons of length  $\ell$  needs  $2\ell$  bits to be stored. Since there are 2 895 432 660 self-avoiding polygons of length 38, this method requires 5.6 Tb to store all the elements of  $\text{SAP}_{38}$ . To decrease this enormous quantity of data, remark that thanks to the depth-first exploration of the tree  $\mathcal{T}_\ell$ , elements of  $\text{SAP}_\ell$  are constructed in lexicographic order. This implies that two consecutive self-avoiding polygons likely share a long prefix. As a consequence, to store two consecutive self-avoiding polygons  $p_1$  and  $p_2$ , we store  $p_1$ , then store the length of the common prefix between  $p_2$  and  $p_1$  as well as the suffix of  $p_2$  with respect to this prefix using the previous encoding. Since we consider SAP of length at most  $38 \leq 64$ , we need no more than 6 bits to store the length of the prefix, considerably reducing the memory requirement for  $p_2$ . Finally, we use the Lempel–Ziv–Markov chain algorithm (LZMA) [? ] to perform on the fly compression. The following table summarize space needed to store  $\text{SAP}_\ell$  using basic encoding Eq. (9); prefix encoding; and LZMA compression on prefix encoding. Empirically, we found that LZMA compression employed directly on basic encoding takes much time and gives terrible results in terms of storage.

$\ell$	20	22	24	26	28	30	32	34	36	38	40
basic	400Kb	2.4Mb	4.4Mb	89Mb	556Mb	3.4Gb	21.6Gb	137Gb	877Gb	5.6Tb	35.4Tb
prefix	148Kb	809Kb	4.4Mb	25.1Mb	145Mb	849Mb	4.9Gb	–	–	–	–
+ lzma	516Kb	524Kb	884Kb	1.8Mb	4.4Mb	14Mb	66Mb	350Mb	1.5Gb	8Gb	43Gb

We can parallelize the exploration of the tree  $\mathcal{T}_\ell$  by selecting nodes  $w_1, \dots, w_k$  such that the leaves of maximal depth in  $\mathcal{T}_\ell$  are in bijection with leaves of maximal depth in the subtrees of  $\mathcal{T}_\ell$  rooted at

$\ell$	$\pi(\ell)$	$F_\ell$	$S(\ell)$
2	1	0.5000000000000000	0.5000000000000000
4	1	0.14727245910375	0.64727245910375
6	2	0.06204664274521	0.70931910184896
8	7	0.04001566383131	0.74933476568027
10	28	0.02805060444094	0.77738537012121
12	124	0.02102490313204	0.79841027325325
14	588	0.01644695527417	0.8148572285274199
16	2 938	0.01329675992709	0.8281539884545099
18	15 268	0.01102242742254	0.8391764158770499
20	81 826	0.00931937541569	0.8484957912927399
22	449 572	0.00800628886867	0.8565020801614099
24	2 521 270	0.00696952442824	0.8634716045896499
26	14 385 376	0.00613458465224	0.8696061892418899
28	83 290 424	0.00545085547657	0.8750570447184599
30	488 384 528	0.00488288725692	0.8799399319753799
32	2 895 432 660	0.00440520148051	0.8843451334558899
34	17 332 874 364	0.00399907262190	0.88834420607779
36	104 653 427 012	0.00365046770131	0.8919946737790999
38	636 737 003 384	0.00334868856902	0.89534336234812

Table 1: Number  $\pi(\ell)$  of self-avoiding polygons of length  $\ell$  constructed by the code (see also [?, Table C.3]), corresponding  $F_\ell = \sum_{p \in \text{SAP}_\ell} F_p$  values and cumulative sums  $S(\ell)$  of the  $F_\ell$ .

$w_1, \dots, w_k$ . The difficulty here is to find a selection of nodes such that the times needed to explore the corresponding rooted subtrees are balanced. Using empirical experiments on  $\mathcal{T}_{28}$ , we were able to parallelize the exploration of  $\mathcal{T}_\ell$  for  $\ell \geq 28$  using 64 threads. As our ultimate goal is the computation of the fractions  $F_p$  for all elements of  $\text{SAP}_\ell$ , the constructed self-avoiding polygons are stored in many different files, typically several thousands.

#### 4.2. Computation of $F_p$

At this point, a self-avoiding polygon  $p$  is given as a word  $w$  of steps. From this word, we draw the polygon back onto the square grid. This yields the graph of the distance one neighborhood  $\mathcal{N}(p)$ , the adjacency matrix of which is  $\mathbf{B}_p$  (see Figure 1). The matrix  $\mathbf{C}_p$  is similarly constructed from the pre-computed coefficients  $c_{i,j}$ , see §2. We then form  $\mathbf{M}_p = \text{Id} + \Lambda^{-1}\mathbf{C}_p \cdot \mathbf{B}_p$  and use a Gauss–Jordan elimination to compute the adjugate of  $\mathbf{M}_p$ . To speed up computations, we use the FMA instruction set [?] to perform fused multiply–add operations on vector of four doubles. Typically, if  $\mathbf{a}$ ,  $\mathbf{b}$  and  $\mathbf{c}$  are arrays of four double floats then the computation of  $\mathbf{a}[i] \times \mathbf{b}[i] + \mathbf{c}[i]$  for  $i = 0, 1, 2, 3$ , is done with only one CPU instruction.

#### 4.3. Numerical results

All the algorithms presented here were implemented in C++. Source codes are available on GitHub [?]. We present the final numerical results in Table. 1, showing the computed values for  $F_\ell$  and their accumulation  $S(\ell)$ . These results are also plotted in Fig. 9. These observations lead us to conjecture that  $S(\ell) = 1 - \ell^{-3/5} + O(\ell^{-1})$  and, from there,  $F_\ell = \ell^{-8/5} + O(\ell^{-2})$ .

We wrote a second numerical code that can evaluate  $F_p$  for one given  $p$  of length less than  $\ell(p) \leq 1400$ . Using this code, we present on Table. 2 computed  $F_{\text{sq}_{L \times L}}$  values for the family of  $L \times L$  squares on the infinite square lattice as a function of the square’s side length  $L$ . Such results were hitherto



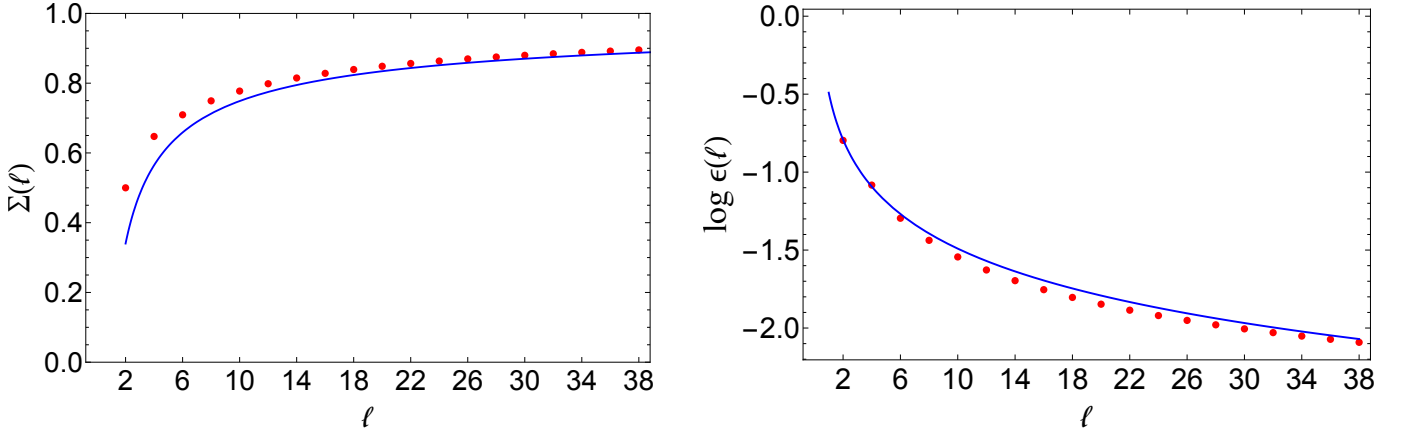


Figure 9: Left figure:  $S(\ell)$  (red dots), the proportion of closed walks on the infinite square lattice whose last erased loop is a self-avoiding polygon of length at most  $\ell$ , as a function of  $\ell$ . The solid blue line is the conjectured fit by  $1 - \ell^{-3/5}$ , which is supposed to hold asymptotically as  $\ell \rightarrow \infty$ . Right figure: logarithm of the error  $\epsilon(\ell) := S(\ell) - (1 - \ell^{-3/5})$  (red dots) and fit by the function  $0.323/\ell$  (solid blue line). Data from Table 1.

completely impossible to access: Theorem 1.1 was unknown until 2021 and only for the six shortest self-avoiding polygons had the fractions  $F_p$  been evaluated [? ? ]. The family of squares offers an interesting testing ground for two reasons: we can conjecture the asymptotic growth  $F_{\text{Sq}_{L \times L}}$  from both numerical observations and analytical arguments (see below); and empirically we found that  $F_{\text{Sq}_{L \times L}}$  is the maximum of  $F_p$  values over the set of all self-avoiding polygons of length  $4L$ . If the length considered is not a multiple of 4, the self-avoiding polygon with maximum  $F_p$  value is a rectangle with a ratio width/height as close to one as possible. At the opposite, the self-avoiding polygons with smallest  $F_p$  value are found to be highly winding with 0 interior area.

The results presented in Table. 2 were also calculated *analytically* by a symbolic program in Sage up to and including  $\text{Sq}_{7 \times 7}$  and by a code for MATHEMATICA up to  $\text{Sq}_{30 \times 30}$ , confirming the numerical results. The analytic formulas are far too cumbersome to be systematically reported here, for example,

$$\begin{aligned}
F_{\text{Sq}_{3 \times 3}} = & 26576424 - \frac{295147905179352825856}{12301875 \pi^{11}} + \frac{191384969764736598016}{2460375 \pi^{10}} - \frac{156074021897315024896}{1366875 \pi^9} \\
& + \frac{136878648694447013888}{1366875 \pi^8} - \frac{8851794332131262464}{151875 \pi^7} + \frac{3588749561696485376}{151875 \pi^6} \\
& - \frac{38289042343284736}{5625 \pi^5} + \frac{870420275786752}{625 \pi^4} - \frac{24779053698384}{125 \pi^3} + \frac{467156948616}{25 \pi^2} - \frac{5246537184}{5 \pi}.
\end{aligned}$$

And the formulas quickly get more involved as the square's side-length increases!

#### 4.4. Analytical considerations

Consider the asymptotic behavior of the fraction of closed walks whose last erased loop is the  $L \times L$  square as  $L \rightarrow \infty$ . Going back to the roots of the sieve of Theorem 1.1 we have that, on a finite lattice  $G_N$  with dominant eigenvalue  $\Lambda$ , the fraction of all closed walks from a vertex  $\bullet$  to itself whose last erased loop is a self-avoiding polygon  $p$  visiting  $\bullet$  is

$$F_p = \frac{1}{\Lambda^{\ell(p)}} \frac{\det(\text{Id} - \frac{1}{\Lambda} \mathbf{A}_{G_N \setminus p})}{\det(\text{Id} - \frac{1}{\Lambda} \mathbf{A}_{G_N \setminus \bullet})}.$$

Here  $\mathbf{A}_{G_N \setminus \bullet}$  is the adjacency matrix of the finite lattice  $G_N$  with vertex  $\bullet$  removed. Similarly,  $\mathbf{A}_{G_N \setminus p}$  is the adjacency matrix of the finite lattice  $G_N$  with all vertices visited by  $p$  removed. Since the vertices

$L$	$F_{\text{Sq}_{L \times L}}$	$L$	$F_{\text{Sq}_{L \times L}}$
1	$1.8409057387969413 \times 10^{-2}$	20	$7.234313040400423 \times 10^{-32}$
2	$4.462339923059934 \times 10^{-4}$	30	$3.2283875735110397 \times 10^{-47}$
3	$1.192983879778077 \times 10^{-5}$	40	$1.4793787629654915 \times 10^{-62}$
4	$3.2824487567509144 \times 10^{-7}$	50	$6.877846988396231 \times 10^{-78}$
5	$9.174122974521936 \times 10^{-9}$	60	$3.226927117230214 \times 10^{-93}$
6	$2.5893979305184303 \times 10^{-10}$	70	$1.523552585714086 \times 10^{-108}$
7	$7.3577883524995755 \times 10^{-12}$	80	$7.226423170276898 \times 10^{-124}$
8	$2.1009188710297932 \times 10^{-13}$	90	$3.4396489661899583 \times 10^{-139}$
9	$6.0210656056096115 \times 10^{-15}$	100	$1.6417501872360198 \times 10^{-154}$
10	$1.730587034739647 \times 10^{-16}$	120	$3.763918325436204 \times 10^{-185}$

Table 2: Numerically computed fraction  $F_{\text{Sq}_{L \times L}}$  of closed walks on the infinite square lattice whose last erased loop is the  $L \times L$  square  $\text{Sq}_{L \times L}$ . These numbers were also calculated analytically up to  $L = 30$ . See Fig. 10 for a plot of these values.

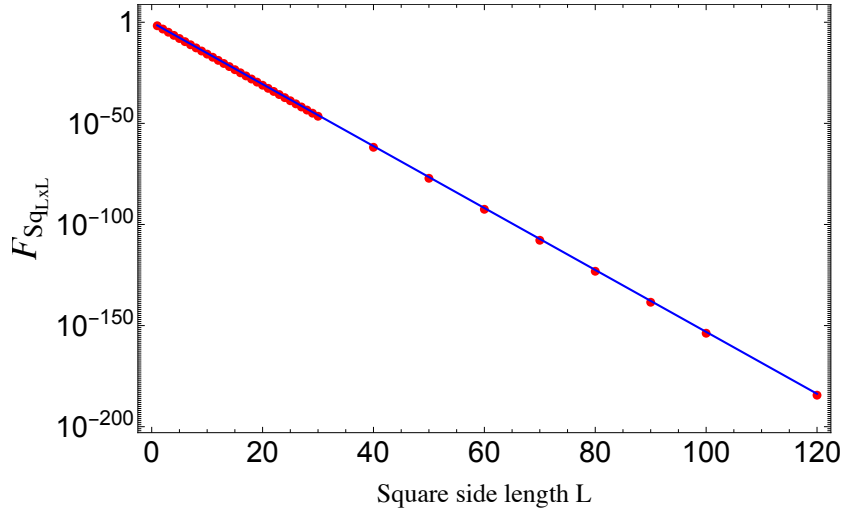


Figure 10: Log-plot of the fraction  $F_{\text{Sq}_{L \times L}}$  of closed walks on the infinite square lattice whose last erased loop is the  $L \times L$  square as a function of the side length  $L$  (red dots); and fit based on analytical considerations §4.4 by the function  $(\sqrt{2} - 1)^{4L}$  (solid blue line). Data from Table 2.

located in the interior of the self-avoiding polygon  $p$  are completely disconnected from those outside of it, the graph  $G_N \setminus p$  is disconnected and  $F_p$  simplifies to

$$F_p = \frac{1}{\Lambda^{\ell(p)}} \frac{\det(\text{Id} - \frac{1}{\Lambda} \mathbf{A}_{\text{out}(G_N \setminus p)})}{\det(\text{Id} - \frac{1}{\Lambda} \mathbf{A}_{G \setminus \bullet})} \det(\text{Id} - \frac{1}{\Lambda} \mathbf{A}_{\text{in}(G_N \setminus p)}). \quad (10)$$

where  $\text{in}(G_N \setminus p)$  and  $\text{out}(G_N \setminus p)$  designate the set of vertices in the interior and outside of  $p$  on  $G_N$ , respectively. The induced subgraph of  $G_N$  with vertices in  $\text{in}(G_N \setminus p)$  remains a finite graph under the limit  $N \rightarrow \infty$  so  $\det(\text{Id} - \Lambda^{-1} \mathbf{A}_{\text{in}(G_N \setminus p)})$  is well defined and usually known exactly. For instance on the square lattice and with  $p$  the  $L \times L$  square,  $\text{in}(G_N \setminus \text{Sq}_{L \times L})$  is the  $(L-2) \times (L-2)$  square, so

$$\log \det \left( \text{Id} - \frac{1}{4} \mathbf{A}_{\text{Sq}_{L-2 \times L-2}} \right) = \prod_{1 \leq i, j \leq L-2} \left( 1 - \frac{1}{2} \cos \left( \frac{\pi i}{L-1} \right) - \frac{1}{2} \cos \left( \frac{\pi j}{L-1} \right) \right).$$

The asymptotic behavior of this when  $L \rightarrow \infty$  is known. Denote  $\mathbf{L}_{\text{Sq}_{L \times L}}$  the Laplacian of the  $L \times L$  square, then

$$\det \left( \text{Id} - \frac{1}{4} \mathbf{A}_{\text{Sq}_{L-2 \times L-2}} \right) = \frac{4^{-(L-2)^2}}{(L-1)^2} \det_0(\mathbf{L}_{\text{Sq}_{L-1 \times L-1}}),$$

where  $\det_0$  is the product of the non-zero eigenvalues. It follows [?, Section 6] [? ? ],

$$\log \det \left( \text{Id} - \frac{1}{4} \mathbf{A}_{\text{Sq}_{L-2 \times L-2}} \right) = \left( \frac{4C}{\pi} - \log(4) \right) \mathcal{A}(\text{Sq}_{L-1 \times L-1}) + \frac{1}{2} \log(4\sqrt{2} - 4) \mathcal{P}(\text{Sq}_{L-1 \times L-1}) + O(\log L),$$

where  $C$  is Catalan's constant,  $\mathcal{A}(\text{Sq}_{L-1 \times L-1}) = (L-1)^2$  and  $\mathcal{P}(\text{Sq}_{L-1 \times L-1}) = 4(L-1)$  are the area and perimeter of  $\text{Sq}_{L-1 \times L-1}$ , respectively. The area and perimeter are quantities that are invariant under conformal transformations of the lattice under the continuum limit, leading to the conformal invariance of SLE measures. Unfortunately, neither of the other two determinants in Eq. (10) make any sense under the limit where  $N := |G_N| \rightarrow \infty$  where  $G_N \rightarrow G$  becomes infinite. The behavior of their ratio is also very difficult to control. Instead of attempting to do so exactly, as Theorem 1.1 achieved, we may only seek an asymptotic description of these determinants as  $N \rightarrow \infty$ . To do so, observe that the two ill-defined determinants' contents can be transformed so as to make graph Laplacians appear. Indeed,

$$\det(\text{Id} - \Lambda^{-1} \mathbf{A}_{\text{out}(G_N \setminus p)}) = \Lambda^{-N} \det(\text{Id} \Lambda - \mathbf{A}_{\text{out}(G_N \setminus p)}),$$

which looks like a graph Laplacian, except that all vertices are effectively given a degree  $\Lambda$ . But vertices of  $\mathcal{N}(p)$  have a strictly smaller degree on  $G_N \setminus p$ , by virtue of the removal of the vertices visited by  $p$ . Let  $\mathbf{D}_p$  be a diagonal matrix accounting for this, that is, for any vertex  $i$ , define  $\mathbf{D}_p$  through

$$\Lambda = \deg(i_{\text{out}(G_N \setminus p)}) + (\mathbf{D}_p)_{i,i},$$

where  $\deg(i_{\text{out}(G_N \setminus p)})$  designates the degree of vertex  $i$  on the subgraph of  $G_N$  induced by the vertex set  $\text{out}(G_N \setminus p)$ . By construction,  $(\mathbf{D}_p)_{i,i} = 0$  whenever  $i \notin \mathcal{N}(p)$ , i.e.  $\mathbf{D}_p$  is a "small-rank and localized" correction, the extent of which does not increase as the lattice size is increased. This leads to

$$\det(\text{Id} - \Lambda^{-1} \mathbf{A}_{\text{out}(G_N \setminus p)}) = \Lambda^{-|\text{out}(G \setminus p)|} \det_0(\mathbf{L}_{\text{out}(G \setminus p)} + \mathbf{D}_p), \quad (11)$$

and similarly,  $\det(\text{Id} - \frac{1}{\Lambda} \mathbf{A}_{G \setminus \bullet}) = \Lambda^{|G_N| - 1} \det(\mathbf{L}_{G \setminus \bullet} + \mathbf{D}_\bullet)$ . The determinant of Eq. (11) can be expanded around the dominant  $\mathbf{L}_{\text{out}(G \setminus p)}$ . Here unfortunately we could not complete that step of the proof by showing that the contribution from  $\mathbf{D}_p$  was  $O(\log \ell(p))$  in  $\log \det(\text{Id} - \Lambda^{-1} \mathbf{A}_{\text{out}(G_N \setminus p)})$ , where  $\ell(p)$  is the length of  $p$ . If we accept that this true we would have

$$\begin{aligned} \log(F_p) \stackrel{?}{=} & (-\ell(p) + |\text{in}(G_N \setminus p)| + \mathcal{P}(p) + 1) \log \Lambda + \log \det_0(\mathbf{L}_{G_N \setminus p}) - \log \det_0(\mathbf{L}_{G_N \setminus \bullet}) \\ & + \log \det(\text{Id} - \Lambda^{-1} \mathbf{A}_{\text{in}(G_N \setminus p)}) + O(\log \ell(p)). \end{aligned}$$

Here we used  $|\text{out}(G \setminus p)| = |G_N| - |\text{in}(G \setminus p)| - \mathcal{P}(p)$ . The two terms involving graph Laplacians are available asymptotically thanks to the remarkable results of Finski [? ], Greenblatt [? ] and [? ]. For example, on the square lattice and in the case of the  $L \times L$  square, we have [? , Eq. (1.4) and Theorem 1.2],

$$\log \det_0(\mathbf{L}_{G_N \setminus \text{Sq}_{L \times L}}) - \log \det_0(\mathbf{L}_{G_N \setminus \bullet}) = \frac{4C}{\pi} \times -\mathcal{A}(\text{Sq}_{L-1 \times L-1}) + \frac{1}{2} \log(\sqrt{2} - 1) \mathcal{P}(\text{Sq}_{L \times L}) + O(\log L).$$

Since  $|\text{in}(G \setminus p)| = (L-1)^2$  and  $\mathcal{P}(\text{Sq}_{L \times L}) = 4L$ , putting everything together yields

$$\log(F_{\text{Sq}_{L \times L}}) \stackrel{?}{=} -4L \log(4) + 4L \log(4\sqrt{2} - 4) + O(\log L). \quad (12)$$

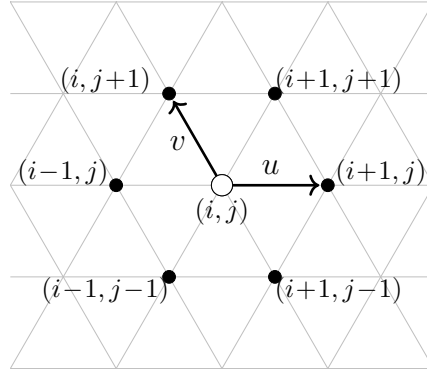


Figure 11: Coordinate system on triangular lattice.

Note how all terms proportional to  $L^2$  stemming from areas have canceled each other out. It cannot be otherwise: if something proportional to  $L^\alpha$  with  $\alpha > 1$  were to remain, then  $F_{S_{q_L \times L}}$  would diverge as  $L \rightarrow \infty$ , in sharp contradiction with the basic combinatorial requirement of Eq. (1). Overall, Eq. (12) allows us to conjecture that

$$F_{S_{q_L \times L}} \stackrel{?}{=} (\sqrt{2} - 1)^{4L} + O(\text{polynom}(L)).$$

This prediction is borne out by the numerical results, see Fig. 10.

The analytical reasoning for the  $-3/5$  exponent appearing the sums  $S(\ell)$  is much weaker. If indeed localized corrections can be neglected, then the sum of  $F_p$  values could possibly be estimated asymptotically using asymptotic determinants of discrete Laplacians. The problem may then be related to that solved by Kenyon [?] as one of the authors of this work had adventurously proposed in [?].

## 5. Generalization to other lattices

The computational procedure described in the case of the square lattice in Sections 2 and 3 extends to other lattices, provided a suitable recursion formula can be found on the entries of the  $C$  matrix. Indeed, while explicit formulas are known for  $C$  via its relation to lattices' Green's functions in all cases [? ? ? ?], these formulas involve integrals of complicated functions and, as such, are ill-suited to fast, large-scale numerical computations requiring high numerical precision. Instead, it is much easier to implement a direct reconstruction of  $C$  exploiting lattice relations [?] and known values for some family of its entries (in the spirit of Proposition 2.1 and Fig. 2 for the square lattice). In this respect the triangular lattice plays a special role since many relations are known relating entries of its resistance matrix to entries of the resistance matrices of other lattices including the Kagomé and dice ones [?]. Unfortunately, as noted by the author of [?], while an explicit integral formula is known giving all resistances on the triangular lattice, no recurrence relation is known between these values. Consequently, these cannot be efficiently reconstructed by a computational procedure similar to that presented here for the square lattice. We remove this roadblock by not only providing such a recurrence but also solving it explicitly thus obtaining a novel expression for the resistance on the triangular lattice. We also solve the open problem of determining the asymptotic behavior of the resistance between two points of this lattice when their distance  $d$  grows to infinity  $d \rightarrow \infty$ . This results not only answer existing question on the infinite resistor triangular lattice but they also establish the validity of our computational approach for the calculation of fractions  $F_p$  to a vast array of lattices.

We first recall the analytical results concerning the resistance values on the triangular lattice:

**Proposition 5.1** ([? ]). *Let  $R(n, m)$  be the resistance from the origin  $O$  to the point with coordinates  $(n, m)$  on the triangular lattice (see Fig. 11). Then*

$$R(n, m) = \frac{1}{\pi} \int_0^{\pi/2} \frac{1 - e^{-|n-m|s} \cos((n+m)s)}{\sinh(s) \cos(x)} dx,$$

where  $\cosh(s) = 2 \sec(x) - \cos(x)$ . In particular for  $n = m$  this simplifies to

$$R(n, n) = \frac{2}{\pi} \int_0^{\pi/2} \frac{\sin(nx)^2}{\sin(x) \sqrt{4 - \cos(x)^2}} dx.$$

Of central importance in the computational procedure for the evaluation of  $\mathbf{C}$  is the determination of these  $R(n, n)$ , ideally without integrals, since from these all  $R(n, m)$  can be recursively calculated using sum rules and lattice symmetries [? ]. The following Proposition identifies a recurrence (conjectured to exist by [? ]) to bypass the integral formulation altogether; identifies its explicit solution in terms of hypergeometric functions and proves the conjectured [? ] logarithmic divergence of  $R(n, n)$ :

**Proposition 5.2.** *Let  $r_n := R(n, n)$  and define  $r_{-1} = 0$  for convenience. Then, on the triangular lattice we have  $r_0 = 0$ ,  $r_1 = 1/3$  and for  $n \geq 2$ ,*

$$r_n = \frac{15n - 22}{n - 1} r_{n-1} - \frac{15n - 23}{n - 1} r_{n-2} + \frac{n - 2}{n - 1} r_{n-3} - \frac{4\sqrt{3}}{\pi(n - 1)}. \quad (13)$$

It follows that  $r_n$  is available exactly from a finite number of sums, ratios and products of rational numbers and a single multiplication by the irrational  $\sqrt{3}/\pi$ . The solution of this is, for  $n \geq 0$ ,

$$r_n = \frac{n}{3} H(n) - \frac{4\sqrt{3}}{\pi} \sum_{m=1}^n (n - m) H(n - m) H(m),$$

where  $H(n) := {}_3F_2\left(\frac{1}{2}, -n + 1, n + 1; 1, \frac{3}{2}; -3\right) \in \mathbb{Q}$ . Here  ${}_3F_2$  designates a generalized hypergeometric function [? ]. Finally we have, asymptotically as  $n \rightarrow +\infty$ ,

$$r_n = \frac{1}{\sqrt{3}\pi} \log(n) + \frac{\gamma + \log(2\sqrt{3})}{\sqrt{3}\pi} + O(1/n),$$

where  $\gamma$  is the Euler–Mascheroni constant.

*Proof.* First we prove the recurrence. Observe that, since  $2 \cos^2(x) - \cos(2x) = 1$ , we have  $r_n = \frac{2\sqrt{2}}{\pi} \int_0^{\pi/2} \frac{\sin(nx)^2}{\sin(x) \sqrt{7 - \cos(2x)}} dx$ . Let  $f_n(x) := \frac{\sin(nx)^2}{\sin(x) \sqrt{7 - \cos(2x)}}$ . We claim

$$\begin{aligned} f_n(x) &= \frac{15n - 22}{n - 1} f_{n-1}(x) - \frac{15n - 23}{n - 1} f_{n-2}(x) + \frac{n - 2}{n - 1} f_{n-3}(x) \\ &\quad + \frac{1}{n - 1} \frac{d}{dx} \left\{ \sqrt{7 - \cos(2x)} \cos((2n - 3)x) \right\}. \end{aligned}$$

Actually, it suffices to show that

$$\begin{aligned} (n - 1) \sin(nx)^2 &= (15n - 22) \sin((n - 1)x)^2 - (15n - 23) \sin((n - 2)x)^2 + (n - 2) \sin((n - 3)x)^2 \\ &\quad + \cos((2n - 3)x) \sin(2x) \sin(x) - (2n - 3)(7 - \cos(2x)) \sin(x) \sin((2n - 3)x). \end{aligned}$$

However, since  $2 \sin(\theta)^2 = 1 - \cos(2\theta)$ , it is equivalent to

$$(n-1) \cos(2nx) = (15n-22) \cos(2(n-1)x) - (15n-23) \cos(2(n-2)x) + (n-2) \cos(2(n-3)x) \\ + \cos((2n-3)x) \sin(2x) \sin(x) - (2n-3)(7 - \cos(2x)) \sin(x) \sin((2n-3)x).$$

This relation is clear using the Product to Sum identities ( $2 \cos(\theta) \cos(\mu) = \cos(\theta - \mu) + \cos(\theta + \mu)$  and  $2 \sin(\theta) \sin(\mu) = \cos(\theta - \mu) - \cos(\theta + \mu)$ ) on the the RHS of the relation.

Second we establish the solution. We begin by considering the homogeneous recurrence equation in  $h_n$  for  $n \geq 2$ ,

$$h_n = \frac{15n-22}{n-1} h_{n-1} - \frac{15n-23}{n-1} h_{n-2} + \frac{n-2}{n-1} h_{n-3}, \quad (14)$$

and with  $h_{-1} = 0$ ,  $h_0 = 0$  and  $h_1 = 1/3$ . Define for  $n, k, \in \mathbb{N}$ ,

$$c(n, k) := \frac{1}{(2k+1)(k!)^2} \frac{(n+k)!}{(n-k-1)!},$$

and observe that for all integers  $n, k \geq 2$ ,

$$c(n, k) = \frac{3n-4}{n-1} c(n-1, k) - \frac{4n-6}{n-1} c(n-1, k-1) \\ - \frac{3n-5}{n-1} c(n-2, k) + \frac{4n-6}{n-1} c(n-2, k-1) + \frac{n-2}{n-1} c(n-3, k), \quad (15)$$

noting that  $c(-1, k) = 0$  for all  $k \geq 1$ . Since furthermore

$$n Q_n(z) := \sum_{k=0}^n c(n, k) z^k = n {}_3F_2 \left( \frac{1}{2}, 1-n, n+1; 1, \frac{3}{2}; z \right),$$

then Eq. (15) implies

$$Q_n(z) = \left( -\frac{4n-6}{n-1} z + \frac{3n-4}{n-1} \right) (n-1) Q_{n-1}(z) \\ + \left( \frac{4n-6}{n-1} z - \frac{3n-5}{n-1} \right) (n-2) Q_{n-2}(z) + \frac{n-2}{n-1} (n-3) Q_{n-3}(z).$$

This becomes Eq. (14) for  $z = -3$ , showing that the solution of the homogeneous recurrence relation Eq. (14) is  $h_n = n Q_n(-3)$  and  $H(n) = Q_n(-3)$ .

The solution to the inhomogeneous equation has a similar origin. Eq. (15) implies that  $P_n(z) := \sum_{m=1}^n (n-m) Q_{n-m}(z) Q_m(z)$  satisfies

$$P_n(z) = \left( -\frac{4n-6}{n-1} z + \frac{3n-4}{n-1} \right) P_{n-1}(z) + \left( \frac{4n-6}{n-1} z - \frac{3n-5}{n-1} \right) P_{n-2}(z) + \frac{n-2}{n-1} P_{n-3}(z) + \frac{1}{n-1},$$

implying that  $-(4\sqrt{3}/\pi) P_n(-3) = -(4\sqrt{3}/\pi) \sum_m (n-m) H(n-m) H(m)$  is a particular solution to the inhomogeneous equation.

Finally, we prove that  $r_n$  diverges logarithmically as  $n \rightarrow +\infty$ . Observe that

$$\frac{1}{\sin(x) \sqrt{4 - \cos(x)^2}} = \frac{1}{\sqrt{3}x} + \frac{4x^3}{45\sqrt{3}} - \frac{2x^5}{63\sqrt{3}} + O(x^6),$$

thus

$$\frac{2}{\pi} \int_0^{\pi/2} \frac{1}{\sqrt{3}x} \sin(nx)^2 dx \leq r_n \leq \frac{2}{\pi} \int_0^{\pi/2} \left( \frac{1}{\sqrt{3}x} + \frac{4x^3}{45\sqrt{3}} \right) \sin(nx)^2 dx.$$

Evaluating both integrals and multiplying everything by  $\sqrt{3}\pi$  for ease of presentation we get

$$\begin{aligned} -\text{Ci}(n\pi) + \gamma + \log(\pi) + \log(n) &\leq \sqrt{3}\pi r_n \leq -\text{Ci}(n\pi) + \gamma + \log(\pi) + \log(n) \\ &\quad + \frac{24((-1)^n - 1)}{720n^4} - \frac{12\pi^2(-1)^n}{720n^2} + \frac{\pi^4}{720}. \end{aligned}$$

Here  $\gamma$  is the Euler–Mascheroni constant and  $\text{Ci}(\cdot)$  is the cos-integral function for which we have  $\lim_{n \rightarrow +\infty} \text{Ci}(n\pi) = 0$ . It follows from the above that asymptotically as  $n \rightarrow +\infty$ ,  $r_n \sim \log(n)/(\sqrt{3}\pi) + c + O(1/n)$ . The value of  $c$  is determined by the observation that for  $q \geq 0$ ,

$$\lim_{n \rightarrow \infty} \int_0^{\pi/2} x^q \sin(nx)^2 dx = \frac{1}{2} \times \frac{(\pi/2)^{q+1}}{q+1}.$$

Given that  $\int x^q dx = x^{q+1}/(q+1)$  and since

$$F(x) := \int \left( \frac{1}{\sin(x)\sqrt{4 - \cos(x)^2}} - \frac{1}{\sqrt{3}x} \right) dx = \frac{-\log(x)}{\sqrt{3}} - \frac{1}{\sqrt{3}} \tanh^{-1} \left( \frac{\sqrt{3} \cos(x)}{\sqrt{4 - \cos(x)^2}} \right),$$

then we have

$$\lim_{n \rightarrow \infty} \int_0^{\pi/2} \left( \frac{1}{\sin(x)\sqrt{4 - \cos(x)^2}} - \frac{1}{\sqrt{3}x} \right) \sin(nx)^2 dx = \frac{1}{2} (F(\pi/2) - F(0)),$$

where we understand  $F(0)$  as the limit  $F(0) := \lim_{x \rightarrow 0} F(x) = -\log(\sqrt{3})/\sqrt{3}$  while  $F(\pi/2) = -\log(\pi/2)/\sqrt{3}$ . Gathering everything we get

$$c = \frac{\gamma + \log(\pi)}{\sqrt{3}\pi} + \frac{2}{\pi} \times \frac{1}{2} \times \left( -\frac{\log(\pi/2)}{\sqrt{3}} + \frac{\log(\sqrt{3})}{\sqrt{3}} \right),$$

which simplifies to the promised result. □

## Acknowledgements

P.-L. G. and Y. H. are supported by the French National Research Agency ANR-19-CE40-0006 project ALgebraic COmbinatorics of Hikes On Lattices (ALCOHOL). Y. H. is further supported by an Université du Littoral Côte d’Opale postdoctoral grant.

## References

- [1] FMA instruction set. [https://en.wikipedia.org/wiki/FMA\\_instruction\\_set](https://en.wikipedia.org/wiki/FMA_instruction_set), 2024.
- [2] Github code, project “square\_sap”. [https://github.com/jfromentin/square\\_sap](https://github.com/jfromentin/square_sap), 2024.
- [3] Lempel–Ziv–Markov chain algorithm. [https://en.wikipedia.org/wiki/Lempel-Ziv-Markov\\_chain\\_algorithm](https://en.wikipedia.org/wiki/Lempel-Ziv-Markov_chain_algorithm), 2024.

- [4] CALCULCO platform, Université du Littoral Côte d’Opale. <https://www-calculco.univ-littoral.fr>, 2024.
- [5] D. Atkinson and F. J. van Steenwijk. Infinite resistive lattices. *American Journal of Physics*, 67(6):486–492, 1999.
- [6] W. N. Bailey. *Generalized hypergeometric series*. Cambridge University Press, 1935.
- [7] A. R. Conway, I. G. Enting, and A. J. Guttmann. Algebraic techniques for enumerating self-avoiding walks on the square lattice. *Journal of Physics A: Mathematical and General*, 26(7):1519, apr 1993.
- [8] J. Cserti. Application of the lattice Green’s function for calculating the resistance of an infinite network of resistors. *American Journal of Physics*, 68(10):896–906, 2000.
- [9] J. Cserti, G. Széchenyi, and G. Dávid. Uniform tiling with electrical resistors. *Journal of Physics A: Mathematical and Theoretical*, 44(21):215201, apr 2011.
- [10] B. Duplantier and F. David. Exact partition functions and correlation functions of multiple Hamiltonian walks on the Manhattan lattice. *Journal of Statistical Physics*, 51(3):327–434, May 1988.
- [11] S. Finski. Spanning trees, cycle-rooted spanning forests on discretizations of flat surfaces and analytic torsion. *Math. Z.*, 301(4):3285–3343, 2022.
- [12] P.-L. Giscard. Counting walks by their last erased self-avoiding polygons using sieves. *Discrete Mathematics*, 344(4):112305, 2021.
- [13] R. L. Greenblatt. Discrete and zeta-regularized determinants of the Laplacian on polygonal domains with Dirichlet boundary conditions. *Journal of Mathematical Physics*, 64(4):043301, 04 2023.
- [14] A. J. Guttmann. Lattice Green’s functions in all dimensions. *Journal of Physics A: Mathematical and Theoretical*, 43(30):305205, jun 2010.
- [15] I. Jensen and A. J. Guttmann. Self-avoiding polygons on the square lattice. *Journal of Physics A: Mathematical and General*, 32(26):4867, jul 1999.
- [16] R. Kenyon. The asymptotic determinant of the discrete Laplacian. *Acta Mathematica*, 185(2):239–286, 2000.
- [17] D. J. Klein. Resistance-distance sum rules\*. *Croatica Chemica Acta*, 75:633–649, 2002.
- [18] D. J. Klein and M. Randić. Resistance distance. *Journal of Mathematical Chemistry*, 12(1):81–95, Dec 1993.
- [19] G. F. Lawler. A self-avoiding random walk. *Duke Mathematical Journal*, 47(3):655–693, 09 1980.
- [20] G. F. Lawler. *Loop-Erased Random Walk*, pages 197–217. Birkhäuser Boston, Boston, MA, 1999.
- [21] G. F. Lawler, O. Schramm, and W. Werner. Conformal invariance of planar loop-erased random walks and uniform spanning trees. *The Annals of Probability*, 32(1B):939 – 995, 2004.
- [22] N. Madras and G. Slade. *The Self-Avoiding Walk*. Modern Birkhäuser Classics. Springer New York, 2012.



- [23] S. N. Majumdar and D. Dhar. Height correlations in the Abelian sandpile model. *Journal of Physics A: Mathematical and General*, 24(7):L357–L362, apr 1991.
- [24] S. S. Manna, D. Dhar, and S. N. Majumdar. Spanning trees in two dimensions. *Physical Review A*, 46:R4471–R4474, Oct 1992.
- [25] T. Morita. Use of a recurrence formula in computing the lattice Green function. *Journal of Physics A: Mathematical and General*, 8(4):478–489, apr 1975.
- [26] C. Pozrikidis. *An Introduction to Grids, Graphs, and Networks*. OUP USA, 2014.
- [27] O. Schramm. Scaling limits of loop-erased random walks and uniform spanning trees. *Israel Journal of Mathematics*, 118(1):221–288, Dec 2000.