

AN UPPER BOUND FOR THE CONNECTIVE CONSTANT OF THE SQUARE LATTICE

TEJASVI SINGH TOMAR

ABSTRACT. Let μ be the connective constant of self-avoiding walks on the square lattice $\mathbb{Z}^2$, the exponential growth rate of the number of n -step self-avoiding walks. We prove $\mu \leq 2.6600756$. The previous best upper bound, $\mu \leq 2.662342426$, is due to Couronné. As in the work of Pönitz and Tittmann and of Couronné, the bound comes from a finite automaton whose states remember a bounded part of the past of the walk. Our automaton counts bridges, and, as in Couronné's automaton, its states may contain cells that the walk has not visited; here these are cells that no infinite self-avoiding continuation of the walk can visit, recognized by three tests that depend only on the state and the step. Distinct bridges, preceded by a fixed infinite history, then give distinct paths in the automaton, so the Hammersley–Welsh theorem turns a bound for the number of paths into a bound for μ . The bound follows from a certificate, consisting of the automaton (4,124,965,633 states) and a vector of integer weights, which a program checks in exact integer arithmetic.

1. INTRODUCTION

Let c_n be the number of n -step self-avoiding walks on the square lattice $\mathbb{Z}^2$ that start at the origin. Since $c_{n+m} \leq c_n c_m$, the limit

$$\mu = \lim_{n \rightarrow \infty} c_n^{1/n}$$

exists [3]. It is called the connective constant of $\mathbb{Z}^2$ (constant 38a in the repository of optimization constants [10]). Its value is not known; numerical estimates give $\mu \approx 2.63815853032790$ [5].

Upper bounds for μ come from counting walks with a bounded memory. Alm [1] obtained $\mu \leq 2.69576$, and Noonan [8] obtained $\mu \leq 2.6939$. Pönitz and Tittmann [9] built finite automata whose states are the last steps of the walk, with the steps that can no longer matter removed, and obtained $\mu \leq 2.679192495$. Couronné [2] added several simplification rules to their automaton and obtained $\mu \leq 2.662342426$, considering loops of length up to 26. Lower bounds come from counting irreducible bridges; see [6] and the references there.

Theorem 1.1. *The connective constant of the square lattice satisfies*

$$\mu \leq \frac{7653257817410405}{2877082878637577} = 2.660075548826 \dots < 2.6600756.$$

As in [9, 2], the bound comes from a finite automaton. Here it is the largest ratio of a vector of weights (Lemma 5.1), which is an upper bound for the growth rate of the automaton. Unlike these works, we count bridges. Each bridge is preceded by the same infinite history (Section 2), and history and bridge together are part of a two-sided infinite self-avoiding path. As in some of the simplification rules of [2], a state may contain cells that the walk has not visited; here each such cell is dead, that is, no infinite self-avoiding continuation of the current past can visit it. A state of the automaton is a set of cells, written as a lattice path ending at the current position,

2020 *Mathematics Subject Classification.* 82B41; 05A15, 60K35, 68Q45, 68W30.

Key words and phrases. self-avoiding walk, connective constant, square lattice, bridges, finite automata, computer-assisted proof.

and every cell of the state is either visited or dead. When the walk takes a step, the automaton passes to a state that consists of cells of the old state, the new position and new dead cells, found by the three tests of Definition 3.1. The automaton is described by a finite certificate: a list of states, an entry (a child or the symbol $\emptyset$) for every state and step, and a vector of nonnegative integers. By Theorem 4.1, distinct bridges give distinct paths in the automaton. Lemma 5.1 bounds the number of paths in terms of the largest ratio of the weights. The Hammersley–Welsh theorem [4] gives $\mu = \lim b_n^{1/n}$ for the number b_n of bridges, which proves Theorem 1.1.

We built the certificate for Theorem 1.1 with a separate program (Section 7), but its correctness depends only on the checks described in Section 6, which a checker carries out in exact integer arithmetic.

2. BRIDGES

We write $e_1 = (1, 0)$ and $e_2 = (0, 1)$, and $x(z)$ for the first coordinate of $z \in \mathbb{Z}^2$. An n -step self-avoiding walk is a sequence $\omega_0, \dots, \omega_n$ of distinct points of $\mathbb{Z}^2$ with $|\omega_i - \omega_{i-1}| = 1$. It is a bridge if

$$x(\omega_0) < x(\omega_i) \leq x(\omega_n) \quad (1 \leq i \leq n).$$

Let b_n be the number of n -step bridges with $\omega_0 = 0$. The sequence (b_n) is supermultiplicative, and Hammersley and Welsh [4] proved that $c_n \leq b_{n+1}e^{O(\sqrt{n})}$. Hence (see also [7, Section 3.1])

$$(2.1) \quad \mu = \lim_{n \rightarrow \infty} b_n^{1/n}.$$

Let $H = \{(x, 0) : x \leq 0\}$. If ω is a bridge with $\omega_0 = 0$, then $\omega_i \notin H$ for $i \geq 1$, because $x(\omega_i) \geq 1$. Concatenating ω with translates of itself gives an infinite self-avoiding path that starts with ω and stays in the half-plane $\{x \geq 1\}$ after its first step. Hence the two-sided sequence that runs through H from $x = -\infty$ to 0 and then follows this path is a two-sided infinite self-avoiding path. We call H the history of the bridge. We call the points of $\mathbb{Z}^2$ cells. For $0 \leq k \leq n$, a *continuation* of ω at time k is a sequence $(\gamma_i)_{i \in \mathbb{Z}}$ of distinct cells with $|\gamma_i - \gamma_{i-1}| = 1$ for all i , $\gamma_i = (i, 0)$ for $i \leq 0$ and $\gamma_i = \omega_i$ for $0 \leq i \leq k$. It *visits* a cell z after time k if $\gamma_i = z$ for some $i > k$. By the above, continuations exist for every k , and a continuation at time j is also a continuation at every time $k \leq j$.

3. AUTOMATA WITH DEAD CELLS

3.1. States and steps. Let D_4 be the group of the eight symmetries of $\mathbb{Z}^2$ that fix the origin. A *state* is a self-avoiding lattice path $S = (s_0, \dots, s_{m-1})$ with $m \geq 2$, $s_{m-1} = 0$ and $s_{m-2} = -e_1$. We write $|S|$ for its set of cells. The origin is the current position, and the last step of the path is $+e_1$. The three *steps* of a state are $t \in \{e_1, e_2, -e_2\}$; the fourth neighbour $-e_1$ of the origin is a cell of the state. The *start state* S_{st} is the straight path $s_i = (i - K/2, 0)$, $0 \leq i \leq K/2$, of $K/2 + 1$ cells, for an even integer $K \geq 2$, which we call the order of the automaton. The order enters the definitions below only through S_{st} ; it mainly governs the construction of the automaton (Section 7).

For a finite set $E \subset \mathbb{Z}^2$, the complement $\mathbb{Z}^2 \setminus E$ has exactly one infinite connected component (for 4-connectivity). A cell of $\mathbb{Z}^2 \setminus E$ is *connected to infinity in $\mathbb{Z}^2 \setminus E$* if it lies in this component. This is the case if and only if it is connected, within $\mathbb{Z}^2 \setminus E$, to a cell outside the bounding box of E .

Definition 3.1. Let S be a state, t a step with $t \notin |S|$, and $E = |S| \cup \{t\}$. A cell $z \notin E$ is *dead for (S, t)* if

- (a) at least three of the four neighbours of z lie in $|S|$; or
- (b) z is not connected to infinity in $\mathbb{Z}^2 \setminus E$; or

- (c) there is a cell $w \notin E \cup \{z\}$ such that, in $\mathbb{Z}^2 \setminus (E \cup \{w\})$, the cell z is connected neither to infinity nor to any neighbour of t outside $E \cup \{w\}$.

3.2. Certificates.

Definition 3.2. A *certificate* of order K consists of a finite set $\mathcal{S}$ of states that contains S_{st} , an integer $V_S \geq 0$ for each $S \in \mathcal{S}$, and, for each $S \in \mathcal{S}$ and each step t , an entry that is either the symbol $\emptyset$ or a state $S' \in \mathcal{S}$, such that the following hold.

- (C1) If the entry for (S, t) is $\emptyset$, then $t \in |S|$, or t is not connected to infinity in $\mathbb{Z}^2 \setminus |S|$.
- (C2) If the entry for (S, t) is a state S' and $t \notin |S|$, then there is $g \in D_4$ such that every cell of the set $g(|S'|) + t$ lies in $E = |S| \cup \{t\}$ or is dead for (S, t) .
- (C3) $V_{S_{\text{st}}} > 0$; every state with $V_S = 0$ has only children with $V = 0$; and there is no sequence $S_1, \dots, S_r = S_1$ ($r \geq 2$) of states with $V = 0$ in which each S_{i+1} is a child of S_i .

Here the *children* of S are the states S' that occur in its entries. The *ratio* of the certificate is

$$\lambda = \max \left\{ V_S^{-1} \sum_t V_{S'_t} : S \in \mathcal{S}, V_S > 0 \right\},$$

where the sum runs over the steps t whose entry is a state S'_t .

An entry S' for a step $t \in |S|$ is allowed and need not satisfy any condition. Such an entry is never used by a walk and can only increase the counts below. The set $g(|S'|) + t$ in (C2) contains both $t = g(0) + t$ and the cell $g(-e_1) + t$, which is the image of the second-to-last cell of S' ; so (C2) requires the cell in the excluded direction $-e_1$ of the next state to lie in E or to be dead. The *automaton graph* of a certificate is the directed graph on $\mathcal{S}$ with one edge from S to S' for each step of S whose entry is S' . A *path* of length n in it is a sequence of n edges in which each edge starts at the state where the previous one ends; states and edges may repeat.

4. SOUNDNESS

Theorem 4.1. *Let a certificate of order K be given. For every $n \geq 1$, b_n is at most the number of paths of length n in the automaton graph that start at S_{st} .*

Proof. Fix an n -step bridge ω with $\omega_0 = 0$. For $0 \leq k \leq n$ the *past* at time k is $P_k = H \cup \{\omega_0, \dots, \omega_k\}$. A cell is *dead at time k* if no continuation of ω at time k visits it after time k . The cells of P_k are dead at time k , because the cells of a continuation are distinct. A cell that is dead at time k is dead at every later time j , because a continuation at time j is a continuation at time k , and a visit after time j is a visit after time k .

We define states $S^k \in \mathcal{S}$ and isometries $\Phi_k(z) = h_k(z) + \omega_k$ of $\mathbb{Z}^2$, with $h_k \in D_4$, such that

$$(4.1) \quad \text{every cell of } \Phi_k(|S^k|) \text{ is dead at time } k.$$

Let $S^0 = S_{\text{st}}$ and $\Phi_0 = \text{id}$. Then $\Phi_0(|S^0|) \subset H \subset P_0$, so (4.1) holds for $k = 0$.

Suppose that S^k and Φ_k satisfy (4.1), and let $k < n$. Put $t = h_k^{-1}(\omega_{k+1} - \omega_k)$, a neighbour of the origin. Let γ be a continuation of ω at time n . It is also a continuation at time k , and it visits $\omega_{k+1} = \gamma_{k+1}$ after time k , so ω_{k+1} is not dead at time k . By (4.1), $t \notin |S^k|$. In particular $t \neq -e_1$, so t is a step of S^k . The cells γ_i with $i \geq k+1$ are not dead at time k , by the definition of dead cells, so they avoid $\Phi_k(|S^k|)$. Hence the cells $\Phi_k^{-1}(\gamma_i)$, $i \geq k+1$, form an infinite self-avoiding path that starts at t and avoids $|S^k|$, and t is connected to infinity in $\mathbb{Z}^2 \setminus |S^k|$. By (C1), the entry of (S^k, t) is a state S' . Let g be the first element, in a fixed list of the elements of D_4 , that satisfies (C2) for this entry. This choice makes the construction deterministic. Let $S^{k+1} = S'$ and $\Phi_{k+1}(z) = \Phi_k(g(z) + t)$. Then $h_{k+1} = h_k g$ and $\Phi_{k+1}(0) = \Phi_k(t) = \omega_{k+1}$.

We show (4.1) for $k+1$. Let $E = |S^k| \cup \{t\}$. Every cell of $\Phi_k(E)$ is dead at time $k+1$: the cells of $\Phi_k(|S^k|)$ were dead at time k , and $\Phi_k(t) = \omega_{k+1} \in P_{k+1}$. Let z be a cell of $g(|S'|) + t$. If

$z \in E$, then $\Phi_k(z)$ is dead at time $k + 1$. Otherwise z is dead for (S^k, t) by (C2), and we show that $\Phi_k(z)$ is dead at time $k + 1$. Let δ be a continuation of ω at time $k + 1$, and suppose that δ visits $\Phi_k(z)$ after time $k + 1$. The cells δ_i with $i > k + 1$ are not dead at time $k + 1$, so they avoid $\Phi_k(E)$. Apply Φ_k^{-1} , which is an isometry of $\mathbb{Z}^2$ that maps ω_{k+1} to t . The images of the cells δ_i , $i > k + 1$, form a self-avoiding path δ' that starts at a neighbour of t , avoids E , passes through z and then goes to infinity. In case (a), at most one neighbour of z lies outside $|S^k|$. If there is none, then δ' can neither start at z , which is then not a neighbour of t , nor enter z . Otherwise let u be this neighbour. If z is the first cell of δ' , then z is a neighbour of t , so $u = t$, all four neighbours of z lie in E , and δ' cannot leave z . If not, δ' enters z from a neighbour outside E , which can only be u (so $u \neq t$), and it cannot leave z , because the other neighbours of z lie in E and δ' has already visited u . In case (b), δ' goes from z to infinity in $\mathbb{Z}^2 \setminus E$, which is impossible. In case (c), let w be the cell of the test and Q the component of z in $\mathbb{Z}^2 \setminus (E \cup \{w\})$. The first cell of δ' is a neighbour of t outside E , so it is not in Q (it may be w), Q is finite, and every cell adjacent to Q but not in Q lies in $E \cup \{w\}$. Hence δ' passes through w before it reaches z and again after it leaves z , which contradicts self-avoidance. In each case we have a contradiction, so $\Phi_k(z)$ is dead at time $k + 1$. This proves (4.1) for $k + 1$.

Let t_k be the step t used at time k . The edges of the automaton graph given by S^k and t_k , for $0 \leq k < n$, form a path of length n that starts at S_{st} . The bridge can be recovered from this path: $\omega_{k+1} = \Phi_k(t_k)$, and Φ_{k+1} is determined by Φ_k and by the symmetry g , which depends only on S^k , t_k and the entry of (S^k, t_k) . Hence distinct bridges give distinct paths, which proves the theorem. $\square$

Remark 4.2. Two monotonicity facts are used in the proof. First, every cell of $\Phi_k(E)$ is dead at time $k + 1$, so a continuation at time $k + 1$ avoids $\Phi_k(E)$ after time $k + 1$. Second, if z is dead for (S^k, t) , then $\Phi_k(z)$ is dead at time $k + 1$, although the tests (a)–(c) look only at E : the case analysis uses only that δ' is an infinite self-avoiding path that starts at a neighbour of t and avoids E . The cells of the past outside $\Phi_k(E)$ play no role; self-avoidance is used at cells outside E (the neighbour u in case (a), the cell w in case (c)).

5. COUNTING

Lemma 5.1. *Let a certificate have ratio λ , and let L be the number of its states with $V = 0$. The number of paths of length n in the automaton graph that start at S_{st} is at most*

$$V_{S_{\text{st}}} \lambda^n + 3^{L+1} V_{S_{\text{st}}} \sum_{k=0}^{n-1} \lambda^k.$$

If no state has $V = 0$, it is at most $V_{S_{\text{st}}} \lambda^n$.

Proof. For $k \geq 0$ and a state S with $V_S > 0$, let $W_k(S)$ be the number of paths of length k from S_{st} to S that visit only states with $V > 0$. Then $\sum_S W_k(S) V_S \leq \lambda^k V_{S_{\text{st}}}$. This holds for $k = 0$, and

$$\sum_S W_{k+1}(S) V_S \leq \sum_{S'} W_k(S') \sum_t V_{S'_t} \leq \lambda \sum_{S'} W_k(S') V_{S'},$$

where the outer sums run over states with $V > 0$ and the inner sum runs over the steps t of S' whose entry is a state S'_t , as in the definition of λ . Since $V_S \geq 1$ when $V_S > 0$, the number of such paths of length k is at most $\lambda^k V_{S_{\text{st}}}$. A path that leaves the states with $V > 0$ does so at its first edge into a state with $V = 0$, and from then on it only visits states with $V = 0$, by (C3). These states form an acyclic graph with out-degrees at most 3, so the rest of the path has length less than L , and there are at most 3^L possible rests for each first edge. The first edge

into a state with $V = 0$ follows a path of some length $k \leq n - 1$ that stays in $V > 0$, and there are at most 3 choices for it. This gives the bound. $\square$

Proof of Theorem 1.1. By Theorem 4.1 and Lemma 5.1, $b_n \leq C n \max(\lambda, 1)^n$ for the ratio λ of any certificate, with C independent of n . By (2.1), $\mu \leq \max(\lambda, 1)$. By the computation of Section 6, the files described there define a certificate of order 28 in the sense of Definition 3.2, and its ratio λ is the fraction in Theorem 1.1, which is larger than 1. $\square$

6. THE CERTIFICATE AND ITS VERIFICATION

The certificate for Theorem 1.1 has order $K = 28$ and 4,124,965,633 states, none with $V = 0$. It is stored in three binary files. A state is stored as the 64-bit key $m 2^{58} + \sum_{i=1}^{m-2} \tau_i 3^{m-2-i}$, where τ_i is 0, 1 or 2 when the path goes straight, turns left or turns right at s_i . The keys are sorted. The entries are stored as 32-bit indices into the list of keys, in the order $e_1, e_2, -e_2$ of the steps, with $2^{32} - 1$ for $\emptyset$. The weights are stored as numbers $v_S \in [0, 1]$ in double precision, and $V_S = \lfloor 2^{60} v_S \rfloor$; multiplying a double by a power of 2 is exact, so V_S is computed exactly.

The checker reads the three files and verifies the following, for every state S and every step t .

- The keys are strictly increasing and decode to self-avoiding paths that end at the origin with last step $+e_1$, the start state is present, every v_S lies in $[0, 1]$, and every index is smaller than the number of states. The checker also enforces two range limits (at most 60 cells in a state, and coordinates of absolute value at most 90 for the cells of a placed child outside E); exceeding one causes a rejection, never an acceptance.
- If the entry is $\emptyset$, then $t \in |S|$, or a flood fill from t in $\mathbb{Z}^2 \setminus |S|$ does not reach the boundary of the bounding box of $|S|$ enlarged by one cell. This is (C1).
- If the entry is a child S' , the checker tries the eight symmetries $g \in D_4$ (the symmetry is not stored) and accepts the entry if for one of them every cell of $g(|S'|) + t$ lies in E or passes one of the tests (a)–(c) of Definition 3.1. It does this also when $t \in |S|$, where Definition 3.2 requires nothing. For (b) it uses a flood fill in $\mathbb{Z}^2 \setminus E$ to the boundary of the bounding box of E enlarged by one cell. For (c) it tries as w the cells of one shortest path in $\mathbb{Z}^2 \setminus E$ from z to that boundary; since (b) has failed, such a path exists. A cell outside the bounding box of E stays connected to infinity when any one other cell is removed, so a flood fill with w removed decides whether z is connected to infinity in $\mathbb{Z}^2 \setminus (E \cup \{w\})$, and every cell w for which (c) holds lies on every path from z to the boundary, in particular on the chosen one. This is (C2).
- With the integers $V_S \leq 2^{60}$, it computes the ratio λ as an exact fraction in 128-bit integer arithmetic, in which the products that it compares are below 2^{122} , and checks (C3).

The checker accepts the certificate in 846 seconds on 176 threads. Table 1 lists the counts that it reports: the entries $\emptyset$, by the reason that allows them; the entries with a child for which the first symmetry that passes, in the checker's fixed order of D_4 , maps every cell of the child into E ; and the cells outside E of the other children, by the first of the tests (a), (b), (c) that they pass, where a cell is counted once for each entry whose child contains it. The ratio is

$$\lambda = \frac{7653257817410405}{2877082878637577} = 2.660075548826 \dots$$

The checker prints λ unreduced, as the sum of the weights of the children over the weight of a state where the maximum is attained.¹ A second checker is an independent re-implementation: its checking logic was written from the description of the checked conditions before its author

¹In our run it printed 30613031269641620/11508331514550308. If several states attain the maximum, this pair may differ between runs; the reduced fraction does not.

read the first checker.² It uses flood fills on rows stored as bit masks, and for (c) it tries as w the cells of the component of z in $\mathbb{Z}^2 \setminus E$ that share a side or a corner with a cell of E . No valid w is missed: if none of the eight cells around a cell w lies in E , these cells connect the four neighbours of w , so removing w does not disconnect z from infinity, which (b) has shown to be connected; removing a cell outside the component of z does not change that component; and a smaller set of candidates could in any case only cause a rejection. It accepts the certificate in 6657 seconds on 48 threads and reports the same numbers of entries $\emptyset$ and of dead cells of each kind, the number of entries with a child in Table 1, and the same ratio. It lists D_4 in a different order, so the agreement of the counts, which depend on the first symmetry that passes, is not forced by a common order. Certificates of order 10, 12 and 14, built by the same construction program with the same rules and the default values of its other parameters (in particular without the tables of bonuses used at order 28; see Section 7), are also checked by a program in Python, which, unlike the two checkers, tries as w every cell of the bounding box of E enlarged by one cell; it gives the same ratios as the two checkers. Both checkers accept the certificates of order 10 to 16 built in this way. To test the checkers themselves, we also ran them on certificates that must fail: both reject the certificates of order 12, 14 and 16 built by a variant of the program without the guard for loops with a gap cell (Section 7), and the accepted certificate of order 16 after one child is removed or some children are redirected; after one weight of that certificate is halved, both report a larger ratio.

TABLE 1. Counts for the certificate of order 28. The third row is 3 times the number of states minus the entries $\emptyset$; the other rows are counts reported by the checker.

Entries $\emptyset$ for a step into $ S $	1,837,522,690
Entries $\emptyset$ for a step into a finite component of $\mathbb{Z}^2 \setminus S $	558,549,870
Entries with a child	9,978,824,339
of these, mapped into E by the first symmetry that passes	8,528,323,731
Cells outside E of the other children, passing (a)	1,255,135,841
passing (b) but not (a)	312,989,071
passing (c) but neither (a) nor (b)	475,332,839

7. CONSTRUCTION OF THE AUTOMATON

This section describes how we found the certificate; none of it is needed for the proof.

Our construction program starts from the start state and explores the states reachable by steps. For a state S and a step t it lists several candidate children: subsets of $E = |S| \cup \{t\}$ together with dead cells, written as paths ending at t and placed by a symmetry so that they end at the origin with last step $+e_1$, as in Section 3. The size of a path $(s_0, \dots, s_{m-1})$ is $m - 1 + |s_{m-1} - s_0|_1$, the length of the loop obtained by closing it with a shortest lattice path. The candidates come from rules of the kind introduced by Pönitz and Tittmann [9] and Couronné [2]: old cells are forgotten when the size exceeds the order K , or K plus a bonus of at most 8 that depends on the shape of the path and is read from two tables at order 28 (Couronné’s parameter k plays the role of K), cells cut off by bridge configurations in the sense of [2] (not related to the bridges of Section 2) are recognized as dead, loops sealed by the walk are treated as blocked regions, and steps into trapped regions get no child. Couronné also uses a rule based on the enclosure of the oldest cell of a state (Section 4.2 of the arXiv version of [2]).

²The version used here differs from that program only in its interface, its messages, the choice of the printed state and a limit on the number of candidates for w that the certificate never reaches.

Our program does not use this rule: it gives a step no child only in the two cases allowed by (C1). Our program allows the end of a loop sealed except for one gap cell to be the current position, but only under a guard: no neighbour of the current position lies inside the loop. The small-loop rule of Couronné (Section 3.4 of the arXiv version of [2]) requires that neither end of such a loop is the current position. A variant of our program without the guard produces children that violate (C2), as the rejected certificates of Section 6 show. The program explores the states twice; in the second pass, when it shortens a candidate child, it may stop at any state found in the first pass. The program then alternates power iteration, which gives approximate positive weights, with a new choice, for every state and step, of the candidate child of smallest weight; for the certificate of Theorem 1.1 we ran nine such rounds. The weights in the certificate are those of the last power iteration, computed before the last new choice of children, which can only lower the sums; the ratio λ is computed from them exactly.

For comparison, the arXiv version of [2] reports 2.664196283 with 79,510,267 states at order 24 and 2.662342426 with 430,365,791 states at order 26, and our automaton of order 28 has ratio 2.660075548826... with 4,124,965,633 states.

DATA AVAILABILITY

The certificate files (compressed with xz), the two checkers, the program in Python, the construction program and the programs for the tests of Section 6 are archived at doi:10.5281/zenodo.23540769. The archive also contains an earlier certificate of order 26, with 960,470,402 states and ratio $21295442098601775/8001377286078544 = 2.661472061272\dots$, which both checkers accept. The programs are also available at <https://github.com/tejasvi/connective-constant-bound>.

USE OF AI TOOLS

Generative AI tools were used to find and prove the results, to write the programs, and to draft the text. The author reviewed the arguments and the programs and takes responsibility for the content.

REFERENCES

- [1] S. E. Alm, *Upper bounds for the connective constant of self-avoiding walks*, Combin. Probab. Comput. **2** (1993), 115–136.
- [2] O. Couronné, *New upper bound for the connective constant for square-lattice self-avoiding walks*, J. Stat. Phys. **192** (2025), 153; arXiv:2211.16146.
- [3] J. M. Hammersley and K. W. Morton, *Poor man's Monte Carlo*, J. Roy. Statist. Soc. Ser. B **16** (1954), 23–38.
- [4] J. M. Hammersley and D. J. A. Welsh, *Further results on the rate of convergence to the connective constant of the hypercubical lattice*, Quart. J. Math. Oxford Ser. (2) **13** (1962), 108–110.
- [5] J. L. Jacobsen, C. R. Scullard and A. J. Guttmann, *On the growth constant for square-lattice self-avoiding walks*, J. Phys. A **49** (2016), 494004.
- [6] I. Jensen, *Improved lower bounds on the connective constants for two-dimensional self-avoiding walks*, J. Phys. A **37** (2004), 11521–11529.
- [7] N. Madras and G. Slade, *The Self-Avoiding Walk*, Birkhäuser, Boston, 1993.
- [8] J. Noonan, *New upper bounds for the connective constants of self-avoiding walks*, J. Stat. Phys. **91** (1998), 871–888.
- [9] A. Pönitz and P. Tittmann, *Improved upper bounds for self-avoiding walks in $\mathbb{Z}^d$* , Electron. J. Combin. **7** (2000), R21.
- [10] T. Tao et al., *Optimization constants in mathematics*, <https://teorth.github.io/optimizationproblems/>, 2026.

ORCID: <https://orcid.org/0000-0003-4668-3639>