# Computation of the Ising partition function for two-dimensional square grids

Roland Häggkvist,[1,*]
Anders Rosengren,[2,†] Daniel Andrén,[1,‡] Petras Kundrotas,[2,§] Per Håkan Lundow,[2,‖] and Klas Markström[1,¶]
[1]*Department of Mathematics, Umeå University, SE-901 87 Umeå, Sweden*
[2]*Department of Physics, AlbaNova University Center, KTH, SE-106 91 Stockholm, Sweden*
(Received 17 October 2003; published 16 April 2004)

An improved method for obtaining the Ising partition function of $n\times n$ square grids with periodic boundary is presented. Our method applies results from Galois theory in order to split the computation into smaller parts and at the same time avoid the use of numerics. Using this method we have computed the exact partition function for the $(320\times320)$ grid, the $(256\times256)$ grid, and the $(160\times160)$ grid, as well as for a number of smaller grids. We obtain scaling parameters and compare with what theory prescribes.

DOI: 10.1103/PhysRevE.69.046104 PACS number(s): 05.50+q, 05.10.−a, 02.70.Wz, 02.70.Rr

## I. INTRODUCTION

The Lenz-Ising model [1,2] of ferromagnetism was solved in the one-dimensional case by Ernst Ising in 1925 and in the (infinite) two-dimensional case without an external field by Lars Onsager [3] in 1944. Somewhat later Bruria Kaufman [4] gave the zero field partition function for the $(m\times n)$ grid with periodic boundary. Beale [5] has made an easy-to-use program in MATHEMATICA which implements this solution. Beale used this program to compute the partition function for the $(32\times32)$ grid, and with a modern desktop computer one can use this program to compute the partition function for the $(64\times64)$ grid in about 30 h. We present an improved form of the algorithm where this computation now runs in under 6 h, using a simple implementation in MATHEMATICA. We use a FORTRAN implementation of this algorithm to compute the partition function for a large number of grids of side up to 128, as well as the $(160\times160)$, $(256\times256)$, and $(320\times320)$ grids.

The graphs we are dealing with are the $(m\times n)$ grids with periodic boundary, i.e., the Cartesian product $C_m\times C_n$ of a cycle on $m$ and $n$ vertices, respectively. The total number of vertices is then $mn$ and the number of edges is $2mn$. A state $\sigma$ is a function from the vertices to the set $\{-1,+1\}$. We let $\sigma_v$ denote the state, or spin, of the vertex $v$. The energy of a state $\sigma$ is $E(\sigma)=\Sigma_{uv}\sigma_u\sigma_v$ where the sum is taken over all edges. We then have that $-2mn\leqslant E\leqslant 2mn$, but note that the energy can not take any value in this interval. If both $m$ and $n$ are even then $i$ can take the values $0,\pm4,\pm8,\ldots,\pm2mn$ except for $\pm(2mn-4)$. The relative energy is defined as $\nu(\sigma)=E(\sigma)2mn$, that is $-1\leqslant\nu\leqslant1$. We now define the partition function as the formal Laurent polynomial

$$Z(z)=\sum_{\sigma}z^{E(\sigma)}=\sum_i a_i z^i,$$

where the first sum is taken over all the $2^{mn}$ states. The second sum defines the coefficients $a_i$ as the number of states at energy $i$. In graph theoretical language, $a_i$ is the number of edge cuts of size $(2mn-i)/2$. However, it is common to work not with $Z(z)$ as defined here but rather $Z_0(z)=z^{mn}Z(z^{1/2})$, which gives a polynomial with positive exponents between 0 and $2mn$. In order to be consistent with our references we will do so too.

Whenever we need to distinguish between quantities for different grids we will subscript them with just an $n$ or both $m,n$. Evaluating the partition function $Z$ in the point $z=e^K$, where $K$ is the coupling, gives the partition function $\mathcal{Z}(K)$ typically studied in statistical physics. Here $K=J/k_BT$ where $J$ is the interaction energy, $k_B$ is Boltzmann's constant and $T$ is the absolute temperature. To avoid cluttering up our formulae we set $k_B=J=1$. From $\mathcal{Z}(K)$ we obtain, e.g., the free energy $\mathcal{F}(K)$, the internal energy $\mathcal{U}(K)$ and the specific heat $\mathcal{C}(K)$; we shall define them properly later.

## II. THE FINITE SIZE SOLUTION IN TERMS OF CHEBYSHEV POLYNOMIALS

Following Kaufman [4] and Kasteleyn [6], we know that the partition function for the square grid graph $C_m\times C_n$ can be expressed as a linear combination of four polynomials. These polynomials in turn are given by the Pfaffians of four matrices and can be calculated as the square roots of the corresponding determinants. So if $A_i$ denote the mentioned determinants we have that

$$Z_0(C_m\times C_n,z)=c_1\sqrt{A_1}+c_2\sqrt{A_2}+c_3\sqrt{A_3}+c_4\sqrt{A_4}.$$

Each $A_i$ is a polynomial given by a double product over its roots. A comprehensive description of how to obtain these products and the general Pfaffian method is given in Ref. [7].

Let $\alpha_t=\cos(\pi t/n)$, $\beta_t=\cos(\pi t/m)$, $a=1+z^2$, and $b=z(1-z^2)$. In terms of these variables the four $A_i$'s are

$$A_1=\prod_{i=1}^{n}\prod_{j=1}^{m}(a^2-2b\alpha_{2i}-2b\beta_{2j}),$$

*Electronic address: Roland.Haggkvist@math.umu.se
†Electronic address: ar@theophys.kth.se
‡Electronic address: Daniel.Andren@math.umu.se
§Electronic address: Petras.Kundrotas@csb.ki.se
‖Electronic address: phl@kth.se
¶Electronic address: Klas.Markstrom@math.umu.se

 

$$A_2 = \prod_{i=1}^{n}\prod_{j=1}^{m}(a^2 - 2b\alpha_{2i} - 2b\beta_{2j+1}),$$

$$A_3 = \prod_{i=1}^{n}\prod_{j=1}^{m}(a^2 - 2b\alpha_{2i+1} - 2b\beta_{2j}),$$

$$A_4 = \prod_{i=1}^{n}\prod_{j=1}^{m}(a^2 - 2b\alpha_{2i+1} - 2b\beta_{2j+1}).$$

Computing these products directly and then taking formal square roots is a quite arduous task and we want to find more efficient ways to do this. A first step in this direction was taken by Beale [5] who made use of the fact that most of the roots of the $A_i$ can easily be seen to be double roots and that one can avoid having to take square roots simply by restricting the index range in the products. Using a MATHEMATICA program which evaluated the cosines numerically before performing the simplified products Beale computed the partition function of the $(32\times 32)$ square grid.

Our goal is to perform these products even more efficiently, and with less risk for numerical errors, by using some further observations about the products which both allow us to avoid numerics and use fewer polynomial multiplications. We start out by noting that the roots of the $A_i$ are in fact sums of roots of Chebyshev polynomials, see Eqs. (A4) and (A5) of Appendix A. Here $T_n$ and $U_n$ are the Chebyshev polynomials of the first and second kind. Let $Y=a^2/b$ and $X_t=Y-2\alpha_t$, we can now rewrite the products as

$$\begin{aligned} A_1 &= \prod_{i=1}^{n}\prod_{j=1}^{m}(a^2 - 2b\alpha_{2i} - 2b\beta_{2j}) \\ &= b^{nm}\prod_{i=1}^{n}\prod_{j=1}^{m}(Y - 2\alpha_{2i} - 2\beta_{2j}) \\ &= b^{nm}\prod_{i=1}^{n}\prod_{j=1}^{m}(X_{2i} - 2\beta_{2j}) \\ &= b^{nm}\prod_{i=1}^{n}\prod_{j=1}^{m}2\left(\frac{X_{2i}}{2} - \beta_{2j}\right) \\ &= b^{nm}\prod_{i=1}^{n}2\left[T_m\left(\frac{X_{2i}}{2}\right) - 1\right], \end{aligned} \tag{1}$$

$$\begin{aligned} A_4 &= \prod_{i=1}^{n}\prod_{j=1}^{m}(a^2 - 2b\alpha_{2i+1} - 2b\beta_{2j+1}) \\ &= b^{nm}\prod_{i=1}^{n}\prod_{j=1}^{m}(Y - 2\alpha_{2i+1} - 2\beta_{2j+1}) \\ &= b^{nm}\prod_{i=1}^{n}\prod_{j=1}^{m}(X_{2i+1} - 2\beta_{2j+1}) \\ &= b^{nm}\prod_{i=1}^{n}\prod_{j=1}^{m}2\left(\frac{X_{2i+1}}{2} - \beta_{2j+1}\right) \\ &= b^{nm}\prod_{i=1}^{n}2\left[T_m\left(\frac{X_{2i+1}}{2}\right) + 1\right]. \end{aligned} \tag{2}$$

The expressions for $A_2$ and $A_3$ are similar to $A_4$ and $A_1$, the difference being the index of $X_t$. In $A_2$ the index is $2i$ and in $A_3$ it is $2i+1$.

We now restrict our discussion to the case where $n=m=2p$. The cases for equal, but odd, sides or unequal sides are very similar to the current case and can be handled in the same way. We now use lemma A.4 of Appendix A to simplify the products further. From Eq. (1) we get

$$\begin{aligned} A_1 &= b^{4p^2}\prod_{i=1}^{2p}2\left[T_{2p}\left(\frac{X_{2i}}{2}\right) - 1\right] = b^{4p^2}\prod_{i=1}^{2p}(X_{2i}^2 - 4)\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) = b^{4p^2}\prod_{i=1}^{2p}(X_{2i}^2 - 4)\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) \\ &= b^{4p^2}\prod_{i=1}^{2p}(X_{2i}+2)(X_{2i}-2)\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) = b^{4p^2}\prod_{i=1}^{2p}(Y - 2\alpha_{2i}+2)(Y-2\alpha_{2i}-2)\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) \\ &= b^{4p^2}\prod_{i=1}^{2p}[(Y+2)-2\alpha_{2i}][(Y-2)-2\alpha_{2i}]\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) \\ &= b^{4p^2}2\left[\mathrm{T}_{2p}\left(\frac{Y+2}{2}\right) - 1\right]2\left[\mathrm{T}_{2p}\left(\frac{Y-2}{2}\right) - 1\right]\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) \\ &= b^{4p^2}4\left[\left(\frac{Y+2}{2}\right)^2 - 1\right]\mathrm{U}_{p-1}^2\left(\frac{Y+2}{2}\right)4\left[\left(\frac{Y-2}{2}\right)^2 - 1\right]\mathrm{U}_{p-1}^2\left(\frac{Y-2}{2}\right)\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right) \end{aligned}$$

$$= b^{4p^2}Y^2(Y^2-16)\mathrm{U}_{p-1}^2\left(\frac{Y+2}{2}\right)\mathrm{U}_{p-1}^2\left(\frac{Y-2}{2}\right)\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right)$$
$$= b^{4p^2}\frac{(1+z^2)^4(-1-2z+z^2)^2(-1+2z+z^2)^2}{b^4}\mathrm{U}_{p-1}^2\left(\frac{Y+2}{2}\right)\mathrm{U}_{p-1}^2\left(\frac{Y-2}{2}\right)\prod_{i=1}^{2p}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right).$$

Since all the terms are raised to an even power, we can now take a formal square root by dividing each exponent by 2; the correctness of this choice of sign in the square roots will be discussed in connection with the final linear combination of the polynomials.

$$\sqrt{A_1} = b^{2p^2}\frac{(1+z^2)^2(-1-2z+z^2)(-1+2z+z^2)}{b^2}\mathrm{U}_{p-1}\left(\frac{Y+2}{2}\right)\times\mathrm{U}_{p-1}\left(\frac{Y-2}{2}\right)\prod_{i=1}^{2p}\mathrm{U}_{p-1}\left(\frac{X_{2i}}{2}\right)[\text{Since } X_t = X_{4p-t}]$$
$$= b^{2p^2-2}(1+z^2)^2(-1-2z+z^2)(-1+2z+z^2)\times\mathrm{U}_{p-1}^2\left(\frac{Y+2}{2}\right)\mathrm{U}_{p-1}^2\left(\frac{Y-2}{2}\right)\prod_{i=1}^{p-1}\mathrm{U}_{p-1}^2\left(\frac{X_{2i}}{2}\right)$$
$$= b^{2p^2-2}(1+z^2)^2(-1-2z+z^2)(-1+2z+z^2)\times\mathrm{U}_{p-1}^2\left(\frac{Y+2}{2}\right)\mathrm{U}_{p-1}^2\left(\frac{Y-2}{2}\right)\left[\prod_{i=1}^{p-1}\mathrm{U}_{p-1}\left(\frac{X_{2i}}{2}\right)\right]^2. \quad (3a)$$

Working the same way, we can rewrite identity (2) as

$$A_4 = b^{4p^2}\prod_{i=1}^{2p}2\left[\mathrm{T}_{2p}\left(\frac{X_{2i+1}}{2}\right)+1\right] = b^{4p^2}\prod_{i=1}^{2p}4\mathrm{T}_p^2\left(\frac{X_{2i+1}}{2}\right).$$

Once again can we take a formal square root

$$\sqrt{A_4} = b^{2p^2}\prod_{i=1}^{2p}2\mathrm{T}_p\left(\frac{X_{2i+1}}{2}\right) = b^{2p^2}\left[\prod_{i=0}^{p-1}2\mathrm{T}_p\left(\frac{X_{2i+1}}{2}\right)\right]^2. \quad (3b)$$

Likewise for $A_2$ and $A_3$ we find that

$$\sqrt{A_2} = b^{2p^2}\prod_{i=1}^{2p}2\mathrm{T}_p\left(\frac{X_{2i}}{2}\right)[\text{Since } X_t = X_{4p-t}]$$
$$= b^{2p^2}2\mathrm{T}_p\left(\frac{Y-2}{2}\right)2\mathrm{T}_p\left(\frac{Y+2}{2}\right)\prod_{i=1}^{p-1}\left[2\mathrm{T}_p\left(\frac{X_{2i}}{2}\right)\right]^2$$
$$= b^{2p^2}2\mathrm{T}_p\left(\frac{Y-2}{2}\right)2\mathrm{T}_p\left(\frac{Y+2}{2}\right)\left[\prod_{i=1}^{p-1}2\mathrm{T}_p\left(\frac{X_{2i}}{2}\right)\right]^2, \quad (3c)$$

$$\sqrt{A_3} = b^{2p^2}2\mathrm{T}_p\left(\frac{Y-2}{2}\right)2\mathrm{T}_p\left(\frac{Y+2}{2}\right)\prod_{i=1}^{2p}\mathrm{U}_{p-1}\left(\frac{X_{2i+1}}{2}\right). \quad (3d)$$

Note that we have not simplified $A_3$ quite as much as $A_2$ due to the fact that we have to mix Chebyshev polynomials of the first and second kind.

In fact we find that when $n=m$, $A_2$ and $A_3$ are equal and we could have worked with only one of them above, but we include both cases separately in order to simplify for readers wishing to work on more general cases. Since the expression for $A_2$ is somewhat simpler than that for $A_3$, we shall use the former in our calculations. Here it is computationally very favorable to compute the products first and then square the resulting polynomials.

## III. AVOIDING NUMERICS: A DETOUR DE GALOIS

In order to calculate the $A_i$ we see that we need to evaluate expressions such as $\mathrm{U}_{p-1}(X_{2i}/2)$ and $2\mathrm{T}_p(X_{2i+1}/2)$ for several values of $i$. The most direct route here is of course to evaluate the cos terms of the $X_{2i+1}$ to very high precision and perform the products with floating point numbers as coefficients, and later round all coefficients to integers. Doing this performs well in comparison to Beale's method and using an Alpha workstation and MATHEMATICA one of us was able to compute $Z(C_{128}\times C_{128},z)$ already a few years ago.

The drawback with this numerical, by which we mean using floating point arithmetic, approach is twofold. First we must make sure that we use high enough precision, linear in the number of vertices in the graph, to get a correct answer and it is not a trivial matter to choose a suitable precision which guarantees that both the products and the final additions behave well. Second, the computational effort increases with increasing precision, thus making the size of the graph work against us in two ways. With this in mind our next step is to remove the need for numerical calculations and as far as possible stick to integer coefficients throughout the entire process.

### A. When to use only integers

The first question we need to answer is at which point of our calculation we actually will have integer coefficients. The time when one would usually resort to numerics is when one wants to compute one of the three large products

$$P_1\prod_{i=1}^{p-1}\mathrm{U}_{p-1}\left(\frac{X_{2i}}{2}\right), \quad (4)$$

$$P_2 = \prod_{i=1}^{p-1} 2\mathrm{T}_p\left(\frac{X_{2i}}{2}\right), \tag{5}$$

$$P_4 = \prod_{i=0}^{p-1} 2\mathrm{T}_p\left(\frac{X_{2i+1}}{2}\right), \tag{6}$$

where $P_i$ is the product part of our expression for $\sqrt{A_i}$. Let us focus on $P_2$ for a moment. Every zero of $P_2$ is of the form $2\alpha_i+2\beta_j$, where $2\beta_j$ is a zero of $\mathrm{U}_{p-1}(x/2)$ and $2\alpha_i$ is a zero of $\mathrm{T}_p(x/2)$, see Appendix A, Eq. (A4), and lemma A.4. In fact the set of zeros of $P_2$ consists of all such pairwise sums of zeros of $\mathrm{U}_{p-1}(x/2)$ and $\mathrm{T}_p(x/2)$.

We now make use of the following theorem (the theorem is not new but we include a proof for completeness). Recall that a polynomial is said to be *monic* if its leading coefficient is 1.

*Theorem III.1. Let $P(x)$ and $Q(x)$ be monic polynomials with integer coefficients and define $P \oplus Q$ to be*

$$(P \oplus Q)(x) = \prod_{\alpha \in \mathcal{Z}(P)} \prod_{\beta \in \mathcal{Z}(Q)} (x - \alpha - \beta),$$

*where $\mathcal{Z}(P)$ is the set of zeros of $P$ and $\mathcal{Z}(Q)$ is the set of zeros of $Q$, here the zeros are not necessarily distinct. Then $P \oplus Q$ is a polynomial with integer coefficients.*

*Proof.* From Ref. [8], p. 177, we know that there exist matrices $M_P$ and $M_Q$, with integer entries, such that $P$ and $Q$ are the characteristic polynomials of $M_P$ and $M_Q$, respectively. From Ref. [9], p. 30, we know that the eigenvalues of the matrix $M_{PQ}=M_P \oplus M_Q$, where $\oplus$ denote the Kronecker sum, is the set of pairwise sums of zeros from $P$ and $Q$. Thus we know that $P \oplus Q$ is the characteristic polynomial of $M_{PQ}$ and since all entries of $M_{PQ}$ are integers it follows that $P \oplus Q$ has integer coefficients. ■

*Corollary III.2. Let $P$, $Q_1$, and $Q_2$ be polynomials with integer coefficients. Then*

$$(Q_1 Q_2) \oplus P = (Q_1 \oplus P)(Q_2 \oplus P),$$

*where both $Q_1 \oplus P$ and $Q_2 \oplus P$ are polynomials with integer coefficients.*

From corollary A.3 of Appendix A we know that both $\mathrm{U}_{p-1}(x/2)$ and $2\mathrm{T}_p(x/2)$ have integer coefficients and so the theorem implies that $P_2$ has integer coefficients too. Identical arguments show that $P_1$ and $P_4$ have integer coefficients as well.

This result is very useful in our context since it means that if we use numerics we can round our coefficients to integers once the $P_i$'s have been computed. Since the final polynomial is obtained after squaring the $P_i$'s we have effectively halved the precision needed in our numerics. This also means that if we can compute the $P_i$'s without numerics we can avoid numerics at all stages of our computation.

### B. Galois theory: Basic facts

Before we proceed let us recall some of the basic facts of Galois theory (for a nice introduction to this topic see Ref. [10]). Let $K$ denote a field [20], such as $\mathbb{Q}$ or $\mathbb{R}$ and let $K[x]$ be the ring of polynomials in the indeterminate $x$. A polynomial is said to be *monic* if its leading coefficient is 1. A polynomial in $K[x]$ is said to be *irreducible* if it can not be written as a product of two nonconstant polynomials from $K[x]$. Thus every polynomial in $K[x]$ can be written as a product of irreducible polynomials from $K[x]$.

Given a number $\alpha$ such that $p(\alpha)=0$ for some $p \in K[x]$ we can find a unique irreducible monic polynomial $q \in K[x]$ of minimal degree such that $q(\alpha)=0$; we call this polynomial the *minimum polynomial* of $\alpha$ over $K$. The minimum polynomial of $\alpha$ will divide any polynomial of which $\alpha$ is a zero.

Given a polynomial $p \in K[x]$ we can form a new field by adding the zeros of $p$ to $K$. The smallest field formed in this way is called the *splitting field* of $p$ and in this field $p$ can be factored into linear factors. Given a number $\alpha$ such that $p(\alpha)=0$ for some $p \in K[x]$ we denote by $K(\alpha)$ the splitting field of the minimum polynomial of $\alpha$. Given a polynomial $p$ there is always a zero $\alpha$ of $p$ such that the first $\deg(p)$ powers of $\alpha$ form a basis for $K(\alpha)$ as a vector space over $K$.

We let $G(\alpha)$ denote the set of automorphisms of $K(\alpha)$ which fixes the elements of $K$. From Galois theory we know that $G(\alpha)$ acts as a permutation of the zeros of the minimum polynomial of $\alpha$ and it acts transitively on the set of zeros.

### C. When Galois theory is not needed, $n=2^q$

In each of our three products we want to evaluate a polynomial in $X_{2i}$ or $X_{2i+1}$. We recall that $X_t = Y - 2\alpha_t$ and for later convenience we denote $\gamma_t = 2\alpha_t$. Since our $\gamma_t$ represent $2\cos(t\pi/n)$ we have the following multiplication rule for $\gamma_t$:

$$\gamma_t \gamma_u = \gamma_{t+u} + \gamma_{t-u}$$

which for squaring means that

$$\gamma_t^2 = \gamma_t \gamma_t = \gamma_{2t} + \gamma_0 = \gamma_{2t} + 2. \tag{7}$$

Furthermore, we find that if we multiply $\gamma_t$ and $\gamma_{n-t}$ we get

$$\gamma_{n-t}\gamma_t = \gamma_{(n-t)+t} + \gamma_{(n-t)-t} = \gamma_n + \gamma_{(n-2t)} = -2 + \gamma_{(n-2t)}. \tag{8}$$

Here we should keep in mind that $\gamma_p=0$ and use this to eliminate terms where $\gamma_p$ appears. In both Eqs. (7) and (8) we find that we now have indices of $\gamma$ which correspond to a term of the form

$$\cos\left(\frac{t\pi}{n/2}\right),$$

meaning that we have halved the denominator.

Let us now look at the product $P_1$ and assume that $n$ is of the form $2^q$. Rather than computing $P_1$ directly we compute a sequence of auxiliary polynomials, using the multiplication rules to simplify the products

$$p_{1,n-2k} = \begin{cases} \mathrm{U}_{p-1}\left(\frac{X_{2k}}{2}\right), & 1 \leqslant k \leqslant p-1, \\ 1, & \text{otherwise}, \end{cases}$$

$$p_{t,n-2k} = \begin{cases} p_{t-1,k}\, p_{t-1,n-k}, & 0 \leq k \leq p-1, \\ p_{t-1,p}, & k = p, \\ 1, & \text{otherwise}. \end{cases}$$

From our observations above it follows that each $p_{t,k}$ will be a polynomial in $Y$ and terms of the form $\cos(j\pi/2)/(n/2^{t-1})$, i.e., when we increase $t$ by 1 we halve the denominators in the cos terms. Thus our final polynomial $p_{q+1,n}$ will have only cos terms of the form $\cos(j\pi/2)$, that is, it will have only integer coefficients. Now $p_{q+1,n}$ is our entire product and so is actually $P_1$. This means that the product for $P_1$ will have no remaining cos terms and there is no need for numerical evaluations. That this result will hold for any order of multiplication follows from the commutativity of polynomial multiplication. The same argument applies for $P_2$ and $P_4$.

### D. When Galois theory comes into use

Let us now look at $n$ of the form $n=2p$, where $p$ is not a power of 2. In this case we find that each of our three products can be rewritten as

$$P_1 = \mathrm{U}_{p-1}(Y/2) \oplus \mathrm{U}_{p-1}(Y/2), \tag{9}$$

$$P_2 = [2\mathrm{T}_p(Y/2)] \oplus \mathrm{U}_{p-1}(Y/2), \tag{10}$$

$$P_4 = [2\mathrm{T}_p(Y/2)] \oplus [2\mathrm{T}_p(Y/2)] \tag{11}$$

or, in the terminology of Appendix A 3,

$$P_1 = S_{p-1}(Y) \oplus S_{p-1}(Y), \tag{12}$$

$$P_2 = C_p(Y) \oplus S_{p-1}(Y), \tag{13}$$

$$P_4 = C_p(Y) \oplus C_p(Y). \tag{14}$$

We now have several choices regarding how to compute our $P_i$'s.

A first way to compute our products is to use the observation of corollary III. 2 in combination with our knowledge of the irreducible factors of $C_p$ and $S_p$ to define several intermediate polynomials

$$P_{1,h} = S_{p-1}(Y) \oplus G_h(Y) = \prod S_{p-1}(X_{2i}), \tag{15}$$

$$P_{2,h} = C_p(Y) \oplus G_h(Y) = \prod C_p(X_{2i}), \tag{16}$$

$$P_{4,h} = C_p(Y) \oplus F_h(Y) = \prod S_{p-2}(X_{2i-1}), \tag{17}$$

where the products range over the set of $i$'s corresponding to $h$; see Appendix A. We now have that

$$P_1 = \prod_h P_{1,h}\,, \quad P_2 = \prod_h P_{2,h}\,, \quad P_4 = \prod_h P_{4,h}.$$

The corollary implies that each $P_{i,h}$ will be a polynomial with integer coefficients and so we can return to integer coefficients after each $P_{i,h}$ has been computed. We also note that all the computations performed when computing $P_{1,h}$, and similarly for the other products can be performed in the splitting field of $G_h(x)$. Here we can use the multiplication rule defined earlier to compute products of our $\gamma_t$ as formal variables. We recall that the splitting field $K(\alpha)$ of $G_h$ is generated by some root $\alpha$ of $G_h$. This means that once we have expanded the product for $P_{1,h}$ we will have a polynomial in $Y$ and $\alpha$ with integer coefficients. Since the Galois group $G(\alpha)$ acts transitively on the powers of $\alpha$ and the value of $P_{1,h}$ is invariant under this action, we find that the coefficients of the powers of $\alpha$ in the coefficient of $Y^k$ must all be equal and our polynomial thus has terms of the form

$$\left[a + b\left(\sum_j b_j \alpha^j\right)\right] Y^k,$$

where the $b_j$ are either 0 or 1. We can now evaluate each sum $\Sigma_j b_j \alpha^j$ to an integer and we will have our desired polynomial, computed without need for numerics.

As a second alternative we can make full use of the factorizations of $C_p$ and $S_{p-1}$ to define products

$$P_{1,h,k} = G_h \oplus G_k, \tag{18}$$

$$P_{2,h,k} = F_h \oplus G_k, \tag{19}$$

$$P_{4,h,k} = F_h \oplus F_k, \tag{20}$$

with

$$P_1 = \prod_{h,k} P_{1,h\ ,k}, \quad P_2 = \prod_{h,k} P_{2,h\ ,k}, \quad P_4 = \prod_{h,k} P_{4,h,k}.$$

As before, each of these polynomials will have integer coefficients and we can compute each polynomial either using the multiplication rule as above or, for low degree polynomials, making use of the methods described in the proof of theorem III.1. Breaking the polynomials into small pieces like this will save us a lot in memory usage and we will be able to return to integer coefficients at the earliest possible stage. If we look at the products for $P_1$ and $P_4$ we can note another possible optimization. These products can be rewritten as

$$P_1 = \prod_{h,k} P_{1,h,k} = \left(\prod_{h<k} P_{1,h,k}\right)^2 2^{p-2} S_{p-2}(Y/2), \tag{21}$$

$$P_4 = \prod_{h,k} P_{4,h,k} = \left(\prod_{h<k} P_{1,h,k}\right)^2 2^{p} C_p(Y/2). \tag{22}$$

We can thus compute only about half as many products and then square the resulting polynomials instead. In case $p$ is an odd number we can take this even further by noticing that now the factors of $\mathrm{U}_{p-1}(x/2)$ come in pairs, so that if $q(x)$ is a factor then $q(-x)$ is also a factor. Thus we can compute half of the products just by evaluating the other half in $-x$.

## IV. SUMMING IT UP, BOTH STRAIGHT AND ROUND

Our final step is to take the proper linear combination of the $\sqrt{A_i}$'s in order to get $Z_0$. Here we are faced with two choices. There is one choice of signs which gives us the generating function for the set of Euler subgraphs [21] of size $k$ of $C_m \times C_n$, this is the classical approach following Kasteleyn and Kaufman, however, there is also another choice of signs which gives us the generating function for the number of states of energy $k$ on $C_m \times C_n$. For a fixed energy $k$ these numbers will be equal, apart from a factor 2, for a large enough grid, $k < \min\{m,n\}$, but for a finite grid they will differ for most values of $k$.

The first thing to consider here is the fact that we have to take a formal square root $\sqrt{A_i}$ in order to get the polynomials we wish to add. The square root of a polynomial is unique up to the choice of sign, just as it is for numbers, and we need some way to see which sign is right in our context. This problem is solved as soon as we realize that $\sqrt{A_i}$ is in fact a generating function in itself, counting weighted Euler subgraphs of our grid [7]. Using this fact we see that the first $k=\min\{m,n\}-1$ coefficients should be positive for all four $\sqrt{A_i}$'s and so our earlier choice of sign is correct.

In order not to make our presentation too long we will now make use of some facts from chapters 4 and 5 of Ref. [7]. From Ref. [7] we know that if we take the linear combination

$$\frac{1}{2}(-\sqrt{A_1}+\sqrt{A_2}+\sqrt{A_3}+\sqrt{A_4})$$

we get the generating function for the number of Euler subgraphs of $C_m \times C_n$ and that these are typically considered equal in number to Ising states of a corresponding energy by virtue of the purported self-duality of the square grid. What is typically not mentioned is that this duality work only for self-dual planar graphs such as $P_m \times P_n$, the product of two paths, and in this particular case only for the infinite grid. (Note that a finite self-dual graph on $N$ vertices has $2N-2$ edges, which is not the case for $P_m \times P_n$.) To see this let us consider a cycle in $C_m \times C_n$ which "goes around" the torus on which the graph is naturally embedded, a noncontractible cycle in the language of topology. In the dual graph this cycle will correspond to a set of edges which does not form an edge cut and thus not to an Ising state on the dual graph. For cycles shorter than $k=\min\{m,n\}$ this can not occur and so, by duality, the first and last $k-1$ coefficients will be equal.

However, the problem just described can be remedied in a quite simple way. From basic topological graph theory [11] we know that an Euler subgraph of our grid will correspond to an Ising state on the dual graph if it either does not contain a noncontractible cycle, being of kind (0,0) in the terminology of Ref. [7], p. 66, or contains an even number of such cycles in each of the two possible directions, being of kind (even,even). Making use of this observation and the sign table on p. 66 of Ref. [7] we deduce that

$$\frac{1}{2}(\sqrt{A_1}+\sqrt{A_2}+\sqrt{A_3}+\sqrt{A_4})$$

will give us the generating function for the set of Euler subgraphs of the right kind and so, by duality, the generating function for Ising states with a given energy.

## V. IMPLEMENTATION: MORE OF THE PRACTICAL DETAILS

Here we comment on how to perform some of the calculations described so far in practice and how to verify that we have in the end the correct answer.

### A. Making the initial polynomials

To calculate the product (3a)–(3d) we first calculate the Chebyshev polynomials $U_{n-1}(x/2)$ and $2T_n(x/2)$, then evaluate them in $Y-\gamma_t$ where $\gamma_t=2\alpha_t$ and $Y$ are considered formal variables. That is, we do not choose a value for $t$ at this stage. We end up with a polynomial with integer coefficients and in two variables $Y$ and $\gamma_t$. Since $\gamma_t$ represents $2\cos(t\pi/n)$ we have the following multiplication rule, as we already noted in Sec. III C:

$$\gamma_t\gamma_u=\gamma_{t+u}+\gamma_{t-u}$$

and for squaring this simplifies to

$$\gamma_t^2=\gamma_t\gamma_t=\gamma_{2t}+\gamma_0=\gamma_{2t}+2.$$

Using this rule we can transform the polynomial to a polynomial linear in $\gamma_{t_1},\gamma_{t_2},\ldots$ .

By using the symmetries of the cos function we can further reduce the index of $\gamma_t$ to the interval $0\leqslant t\leqslant n/2$. This reduces the number of $\gamma$ variables and the memory consumption of our calculation. This means that we are now working with signed roots rather than the orginal roots.

In order to make sure that all the $\gamma_{t_j}$ represent nonrational zeros, as required for our conclusions based on the Galois group to apply, we also make use of the rules

$$\gamma_0=2,\quad \gamma_{n/2}=0,\quad \gamma_{n/3}=1.$$

These are the only indices which correspond to rational values of the cos function, see, e.g., Ref. [12].

Should we wish to use one of the more optimized versions of the algorithm, and work with $G_h$ and $F_h$ instead, we can obtain the needed polynomials, e.g., by factoring the respective Chebyshev polynomials in MATHEMATICA.

### B. Multiplying the polynomials

Next we multiply all the polynomials and use the above rules to multiply $\gamma_t$. In this way we will end up with a polynomial in $Y$ and our $\gamma_{t_j}$'s with terms of the form

$$\left[a+b\left(\sum_j b_j\gamma_{t_j}\right)\right]Y^k.$$

We now evaluate the appearing sums of the form $\Sigma_j b_j\gamma_{t_j}$, either using known formulas for trigonometric sums such as

$$\sum_{k=0}^{n}\cos(kx)=\frac{\cos\left(\frac{nx}{2}\right)\sin\left(\frac{(n+1)x}{2}\right)}{\sin\frac{x}{2}},$$

or "cheating" by evaluating them numerically, rounding to the actual integer, and substituting the values back into the polynomial. Using numerics at this stage is actually safe since the sums have few terms, all of similar and small size.

The specific order of multiplication described earlier for the case when $n$ is a power of 2 has some practical advantages as well. Since at each stage we halve the denominator we also reduce the number of cos terms in our polynomials. This means that memory usage is reduced and since there are fewer terms we also save some time in the multiplication of coefficients.

When $n$ is not a power of 2 it is noteworthy that since the number of irreducible factors of the Chebyshev polynomials depend on the divisors of the side length of our grid we can end up with large differences in the amount of work needed to compute the partition function for grids of nearly equal sides. For example, we expect the 510 grid to be significantly easier to handle than the 512 grid. Thus some care should be taken in the choice of grid side, when one is free to do so.

### C. Substituting back to z

To get back to a polynomial in $z$ we have to substitute back

$$Y=\frac{a^2}{b}=\frac{(1+z^2)^2}{z(1-z^2)}.$$

This is a rational function in $z$ and we would like to avoid working with rational functions and work only with polynomials. This is accomplished by using the Horner form of the polynomial [13]. Since we know that the answer is a polynomial and we multiply by a large enough power of $b=z(1-z^2)$, we have the following scenario:

$$\begin{aligned}&b^{2p^2}Y(c_0+Y\{c_1+\cdots Y[c_{2p^2-1}+c_{2p^2}(Y)]\})\\&=b^{2p^2}c_0\left(\frac{a^2}{b}+c_1\left\{\frac{a^2}{b}+\cdots\left[c_{2p^2-1}+c_{2p^2}\left(\frac{a^2}{b}\right)\right]\right\}\right)\\&=c_0(a^2b^{p^2-1}+c_1\{a^2b^{2p^2-2}+\cdots[a^2b+c_{2p^2}(a^2)]\}),\end{aligned}$$

and by using the Horner rule for multiplication of polynomials we end up only using polynomial arithmetic.

### D. Squaring

We now square our polynomials. After that we multiply $A_1$ and $A_2$ with appropriate factors according to Eqs. (3a) and (3c).

When $n$ is large, say 200 or more, some care should be taken here. First, this stage is very suitable for parallelization; second, since the coefficients of the polynomials now become very large one should use an FFT-based multiplication algorithm when multiplying the coefficients, such as the one implemented in Ref. [14].

When $n$ is very large, say 500 or more with present day machines, it becomes hard to handle the full polynomial. The Ising polynomial for $n=512$ would need around 8 Gb of disk space. However, since one is usually interested in some specific range of coefficients rather than the whole polynomial one can settle for computing only the needed range in the squaring process.

### E. The final linear combination

Finally we add our polynomials with either of the choices of signs and we are now done.

### F. Checksums

In order to be reasonably certain that our calculated polynomial is correct we will also make some checksums. Here we focus on $Z$ as the generating function for the number of Ising states of a given energy, with exponents running between $-2mn$ and $2mn$.

The first test to make is of course that the coefficients sum to $2^{mn}$, and more generally we make use of the moments $\mu_k$ of the density of states to verify our calculations. The generating function for the number of states with a given energy is $Z(G,z)$ and thus the moment generating function is $Z(G,\exp(K))=\mathcal{Z}(G,K)$.

Since the first $k=\min\{m,n\}-1$ Taylor coefficients of the free energy $\mathcal{F}(K)$ for four finite $m\times n$ grid coincide with the first $k$ Taylor coefficients of $\mathcal{F}_\infty(K)$ for the infinite grid (see, e.g., Ref. [15]) and $\mathcal{F}(K)$ is the exponential generating function for the moments, see Ref. [16], we have that the first $k$ derivatives of $\exp[mn\mathcal{F}_\infty(K)]$ are equal to the first $k$ moments of our $\mathcal{F}(x)$.

In fact we have

$$\mu_j=\sum_{i=-2mn}^{2mn}a_i i^j=\left.\frac{\mathrm{d}^j\mathcal{Z}(K)}{\mathrm{d}K^j}\right|_{K=0}$$

for $j\leqslant k$. We can now calculate these moments both for the Onsager solution for the infinite grid and for our polynomial and if the first $k$ moments agree we have a very strong indicator that no computational error has occurred.

In practice it seems easier to calculate $[d^j\mathcal{Z}(K)/dK^j]|_{K=0}$ by using the Taylor expansion of the internal energy $\mathcal{U}(K)$ and evaluate

$$\frac{\mathrm{d}^j}{\mathrm{d}K^j}\exp\left(mn\int\mathcal{U}(K)\mathrm{d}K\right)$$

in the ring of formal power series.

A final test can be obtained by observing that the first $k$ Taylor coefficients of $(1/mn)\ln A_1,\ldots,(1/mn)\ln A_4$ are all equal to those of $\mathcal{F}_\infty(x)$. This is the case since each of the three polynomials count the small Euler subgraphs with the same weight.

### G. What we have done

We have implemented our method for both $n=2^k$ as well as general even $n$ using formal variables for $\gamma_t$ but not utilizing full factorization of the Chebyshev polynomials. We began by evaluating Chebyshev polynomials in the formal variables in MATHEMATICA. Next the $P_i$ are computed, substitution is made, squaring is done and finally multiplication with the appropriate prefactors, all using four separate F90 programs. In this way we have computed the Ising partition function for the following $n$: all multiples of 4 up to 80, all multiples of 16 from 80 to 128, all multiples of 32 from 128 up to 160, and finally for $n=256$ and $n=320$.

The smaller cases were handled on ordinary workstations. For $n$ from 160 and upwards we used a linux cluster for the squaring stage. Computation of the $P_i$ for the 256 grid was done on an SGI Origin 3800, using the large integer libraries or Ref. [14]. The squaring stage for the 256 grid took the equivalent of 30 CPU days on an Athlon MP2000 +(1.667 GHz).

For $n=160$ and $n=320$ we used the full Galois method. The factor polynomials were computed using MATHEMATICA on a Macintosh, the larger products giving the $P_{i,k}$ and the substitutions were done on a Linux workstation, and the final multiplications and squarings were done on a Linux cluster. For $n=320$ the final multiplications and squarings took a total of 165 CPU days. The polynomial itself takes up 1.86 Gb of disk space. The polynomials can be downloaded via the papers homepage at URL http://abel.math.umu.se/Combinatorics/ising.html

Here we can also mention that in the course of computing these polynomials our checksums as described above have identified one faulty compiler, a malfunctioning hard disk as well as a bug in a well used standard FORTRAN package—a testimony to how sensitive to software and hardware errors an exact computation like this is, as well as to the accuracy of our checksums.

## VI. DEFINITION OF QUANTITIES

Having computed the partition function for a number of grids, our aim is now to do an analysis of the data. The quantities divide into two groups: those expressed in terms of the coupling $K$ and those expressed in terms of the energy $\nu$. To the first category belongs the free energy $\mathcal{F}(K)$ and its derivatives, the second category contains the entropy $S(\nu)$ and its derivatives. Since the free energy depends on the entire sequence of coefficients $a_i$ whereas the entropy depends on only one $a_i$, we will see some different behavior. Note that asymptotically we may translate between $K$ and a corresponding $\nu$ through the relation $\nu=\mathcal{U}(K)/2$. For example, we may write $S(\nu_c)=\mathcal{F}(K_c)-K_c\,\mathcal{U}(K_c)$ to obtain the asymptotic value of the entropy at the critical point, but this does not throw any light on how this value scales with the size of the grid. Quantities depending on coupling $K$ are written in script, e.g., $\mathcal{F}(K)$, while those depending on energy, e.g., $S(\nu)$ are written in a normal style. Whenever logarithms are used they are natural logarithms in base $e$.

### A. Entropy and coupling

We define the entropy at relative energy $\nu=i/2mn$ as

$$S(\nu)=\frac{\ln a_i}{mn}. \tag{23}$$

Should we desire the entropy at some energy where $a_i$ is not defined, then we will happily circumvent this problem with linear interpolation. The coupling $K$ is defined as

$$K=\frac{-1}{2}S'(\nu). \tag{24}$$

This is in line with the maximum term method (see Ref. [17], Vol. 1, Chap. 2.6) which could give us an alternative definition. Consider the terms in the sum $Z=\Sigma_i a_i z^i$. Given a number $z$ we assume that there is a maximum term $a_i z^i$ such that

$$a_{i-k}z^{i-k}\leqslant a_i z^i \geqslant a_{i+k}z^{i+k},$$

where $k$ is the difference in energy between two consecutive levels of energy. From this inequality we obtain

$$\frac{a_{i-k}}{a_i}\leqslant z^k\leqslant \frac{a_i}{a_{i+k}}.$$

It also follows, as an aside, that $a_{i-k}a_{i+k}\leqslant a_i^2$, i.e., the sequence is log concave at energy $i$. Assuming now that $z=e^K$ we have that $K$ is a number in the interval

$$\frac{1}{k}\ln\frac{a_{i-k}}{a_i}\leqslant K\leqslant \frac{1}{k}\ln\frac{a_i}{a_{i+k}},$$

where we let the lower bound be denoted by $\underline{K}$ and the upper bound $\overline{K}$. Consider now the derivative $S'$ which we define to be

$$S'\left(\frac{i+k/2}{2mn}\right)=\frac{S\left(\frac{i+k}{2mn}\right)-S\left(\frac{i}{2mn}\right)}{k/2mn} =\frac{2mn}{mnk}(\ln a_{i+k}-\ln a_i)=\frac{-2}{k}\ln\frac{a_i}{a_{i+k}}=-2\overline{K}.$$

Note that we will associate the derivative with the middle of $i/2mn$ and $(i+k)/2mn$ since we are dealing with data at discrete points, though this will make little difference for large grids.

As the grid grows we expect that $\overline{K}\to\underline{K}$ making $K$ a well-defined number in the limit. Alternatively we may, as we have done, associate $K$ with the upper bound $\overline{K}$. This has the benefit of making the coupling well defined for all finite systems rather than a number in an interval that exists (possibly) only in the limit.

### B. Physical quantities

For the physical quantities we evaluate the partition function $Z$ in $e^K$ and write $\mathcal{Z}(K)$ for simplicity. We assume the Boltzmann distribution on the states, that is,

$$\Pr(\sigma)=\frac{e^{KE(\sigma)}}{\mathcal{Z}} \quad \text{and} \quad \mathcal{Z}=\sum_\sigma e^{KE(\sigma)},$$

so that the sum of probabilities becomes 1. The derivative then becomes

$$\frac{\partial \ln \mathcal{Z}(K)}{\partial K}=\frac{\mathcal{Z}'}{\mathcal{Z}}=\frac{\Sigma_i a_i i e^{iK}}{\mathcal{Z}}=\sum_i i\,\Pr(i)=\langle E\rangle,$$

where $\langle\cdots\rangle$ denotes the expected value. Analogously for the second derivative we get

$$\frac{\partial^2 \ln \mathcal{Z}(K)}{\partial K^2} = \frac{\mathcal{Z}''}{\mathcal{Z}} - \left(\frac{\mathcal{Z}'}{\mathcal{Z}}\right)^2 = \langle E^2\rangle - \langle E\rangle^2 = \mathrm{var}(E),$$

that is, the variance of $E$. We define the following physical quantities:

$$\text{free energy} \quad \mathcal{F}(K) = \frac{1}{mn}\ln \mathcal{Z}(K),$$

$$\text{internal energy} \quad \mathcal{U}(K) = \frac{\partial \mathcal{F}}{\partial K},$$

$$\text{specific heat} \quad \mathcal{C}(K) = K^2\frac{\partial \mathcal{U}}{\partial K},$$

$$\text{entropy} \quad \mathcal{S}(K) = \mathcal{F} - K\mathcal{U}.$$

We try the reader's patience here somewhat by using a nonstandard, yet clean, simple, and dimensionless definition of the free energy and entropy. That they are internally consistent follows, again, from the maximum-term method. For a large system we simply expect a given coupling $K$ to correspond to a certain energy $E$ and a term that dominates the partition function, thus having $\log \mathcal{Z}(K) \approx \ln a_E e^{KE}$. This gives

$$\mathcal{F}(K) \approx \frac{1}{mn}\ln a_E e^{KE} = \frac{\ln a_E}{mn} + K\frac{E}{mn} = S + K\mathcal{U}$$

so that $S(E/2mn) \approx \mathcal{S}(K) = \mathcal{F}(K) - K\mathcal{U}(K)$.

### C. The Onsager solutions

For completeness we shall state the Onsager solutions which we will view as the limit functions as $m,n \to \infty$. Let $\mathcal{K}_1$ be the complete elliptic integral of the first kind defined by

$$\mathcal{K}_1(x) = \int_0^{\pi/2} (1 - x\sin\theta)^{-1/2}\,\mathrm{d}\theta.$$

Let $\mathcal{K}_2$ be the complete elliptic integral of the second kind defined by

$$\mathcal{K}_2(x) = \int_0^{\pi/2} (1 - x\sin\theta)^{1/2}\,\mathrm{d}\theta.$$

The free energy for the infinite grid, depicted in Fig. 1, is

$$\mathcal{F}(K) = \ln 2 + \frac{1}{2\pi^2}\int_0^{\pi}\int_0^{\pi} \ln[\cosh^2(2K) - \sinh(2K) \times(\cos u + \cos v)]\mathrm{d}u\,\mathrm{d}v.$$

Define $z$ as

$$z = \frac{2\sinh(2K)}{\cosh^2(2K)}.$$

Then the internal energy for the infinite grid, depicted in Fig. 1, is

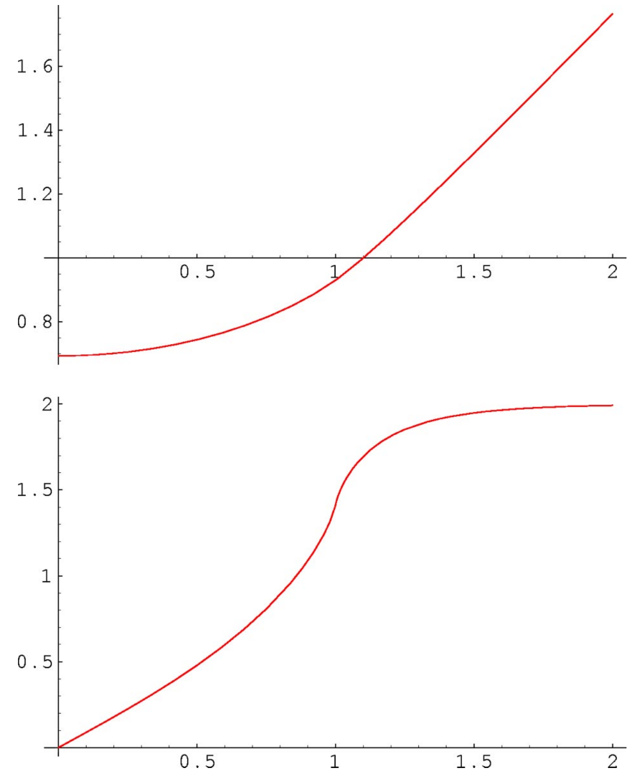


FIG. 1. (Color online) Free energy $\mathcal{F}(K)$ (top) and internal energy $\mathcal{U}(K)$ (bottom) vs $K/K_c$ for the infinite grid.

$$\mathcal{U}(K) = \coth(2K)\left(1 + \frac{2}{\pi}\mathcal{K}_1(z^2)[2\tanh^2(2K) - 1]\right)$$

and the specific heat for the infinite grid, depicted in Fig. 2, is

$$\mathcal{C}(K) = \frac{2}{\pi}K^2\coth^2(2K) \times\Big[2\mathcal{K}_1(z^2) - 2\mathcal{K}_2(z^2) - 2[1 - \tanh^2(2K)] \times\left(\frac{\pi}{2} + \mathcal{K}_1(z^2)[2\tanh^2(2K) - 1]\right)\Big]$$

We shall need the following constants, where $K_c$ is the critical coupling and $G \approx 0.915966$ is Catalan's constant:

$$K_c = \frac{1}{2}\ln(1 + \sqrt{2}) \approx 0.440687,$$

$$F_c = \mathcal{F}(K_c) = \frac{\ln 2}{2} + \frac{2G}{\pi} \approx 0.929695,$$

$$U_c = \mathcal{U}(K_c) = \sqrt{2} \approx 1.414214,$$

$$S_c = \mathcal{S}(K_c) = \frac{\ln 2}{2} + \frac{2G}{\pi} - \sqrt{2}K_c \approx 0.306470.$$

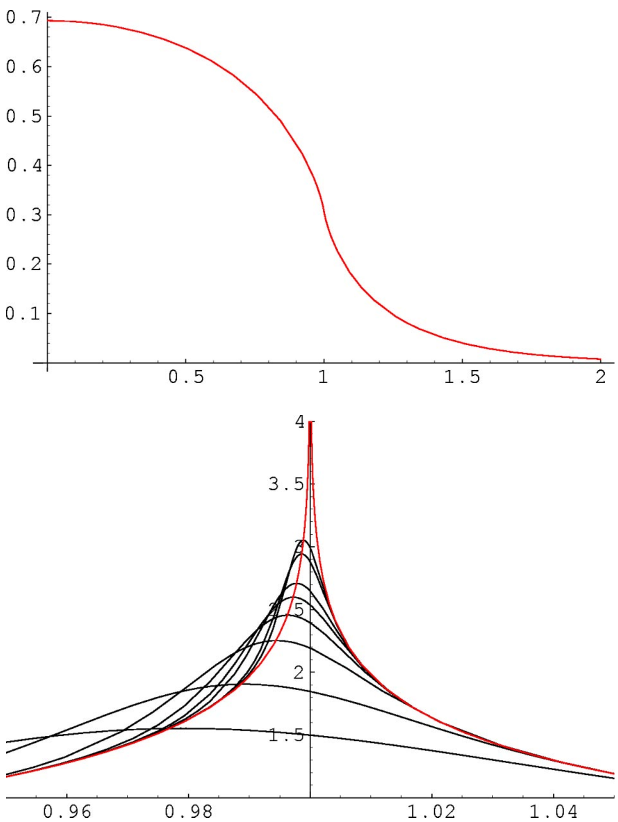


FIG. 2. (Color online) Entropy $\mathcal{S}(K)$ (top) and specific heat $\mathcal{C}_n(K)$ (bottom) for $n=16,32,64,96,128,160,256,320$ and the infinite grid vs $K/K_c$.

## VII. THE FREE ENERGY AND ITS DERIVATIVES

Henceforth we will only consider the case $m=n$. The values at $K_c$ of the free energy etc. are shown in Table I along with the maximum value of $\mathcal{C}$ and the location of the maximum. We denote by $K_n^*$ the location of the maximum of $\mathcal{C}_n$.

In Fig. 3 we show how $\mathcal{F}$ and $\mathcal{U}$ differ from their respective critical values as $n$ increases. It was shown by Ferdinand and Fisher [18] how these differences should behave:

$$\mathcal{F}_n(K_c) - F_c \sim \frac{1}{n^2}\ln(2^{1/4}+2^{-1/2}) \approx \frac{0.639912}{n^2},$$

$$\mathcal{U}_n(K_c) - U_c \sim \frac{2}{n}\frac{\theta_2\theta_3\theta_4}{\theta_2+\theta_3+\theta_4} \approx \frac{0.622439}{n},$$

$$\mathcal{S}_n(K_c) - S_c \approx -\frac{0.274301}{n} + \frac{0.639912}{n^2},$$

where the last formula follows from our definition of entropy $\mathcal{S}=\mathcal{F}-K\mathcal{U}$. For the constants $\theta_2,\theta_3,\theta_4$ we have used the elliptic theta functions $\theta_2=\theta_2(0,e^{-\pi})\approx 0.913579$, $\theta_3=\theta_3(0,e^{-\pi})\approx 1.08643$, and $\theta_4=\theta_4(0,e^{-\pi})\approx 0.913579$.

TABLE I. Values at $K_c$ and extremal data on $\mathcal{C}$.

| $n$ | $\mathcal{F}_n(K_c)$ | $\mathcal{U}_n(K_c)$ | $\mathcal{S}_n(K_c)$ | $\mathcal{C}_n(K_c)$ | max $\mathcal{C}_n$ | $K_n^*$ |
|---|---|---|---|---|---|---|
| 4 | 0.970120 | 1.56562 | 0.280170 | 0.78327 | 0.81646 | 0.410012 |
| 8 | 0.939715 | 1.49159 | 0.282392 | 1.14556 | 1.19184 | 0.423374 |
| 12 | 0.934143 | 1.46596 | 0.288114 | 1.35295 | 1.40391 | 0.428687 |
| 16 | 0.932196 | 1.45306 | 0.291850 | 1.49870 | 1.55220 | 0.431498 |
| 20 | 0.931296 | 1.44531 | 0.294367 | 1.61116 | 1.66628 | 0.433239 |
| 24 | 0.930807 | 1.44013 | 0.296159 | 1.70273 | 1.75898 | 0.434424 |
| 28 | 0.930512 | 1.43643 | 0.297494 | 1.77997 | 1.83706 | 0.435282 |
| 32 | 0.930320 | 1.43366 | 0.298526 | 1.84677 | 1.90451 | 0.435933 |
| 36 | 0.930189 | 1.43150 | 0.299346 | 1.90561 | 1.96386 | 0.436444 |
| 40 | 0.930095 | 1.42977 | 0.300014 | 1.95818 | 2.01686 | 0.436855 |
| 44 | 0.930026 | 1.42836 | 0.300568 | 2.00570 | 2.06473 | 0.437194 |
| 48 | 0.929973 | 1.42718 | 0.301034 | 2.04906 | 2.10839 | 0.437477 |
| 52 | 0.929932 | 1.42618 | 0.301432 | 2.08891 | 2.14850 | 0.437718 |
| 56 | 0.929899 | 1.42533 | 0.301777 | 2.12579 | 2.18561 | 0.437925 |
| 60 | 0.929873 | 1.42459 | 0.302077 | 2.16012 | 2.22013 | 0.438106 |
| 64 | 0.929852 | 1.42394 | 0.302341 | 2.19221 | 2.25239 | 0.438264 |
| 68 | 0.929834 | 1.42337 | 0.302575 | 2.22235 | 2.28269 | 0.438403 |
| 72 | 0.929819 | 1.42286 | 0.302784 | 2.25076 | 2.31123 | 0.438528 |
| 76 | 0.929806 | 1.42240 | 0.302972 | 2.27762 | 2.33822 | 0.438639 |
| 80 | 0.929795 | 1.42199 | 0.303142 | 2.30310 | 2.36381 | 0.438740 |
| 96 | 0.929765 | 1.42070 | 0.303682 | 2.39362 | 2.45470 | 0.439060 |
| 112 | 0.929746 | 1.41977 | 0.304072 | 2.47010 | 2.53145 | 0.439289 |
| 128 | 0.929734 | 1.41908 | 0.304366 | 2.53633 | 2.59789 | 0.439462 |
| 160 | 0.929720 | 1.41810 | 0.304781 | 2.64695 | 2.70880 | 0.439705 |
| 256 | 0.929705 | 1.41664 | 0.305408 | 2.87979 | 2.94210 | 0.440071 |
| 320 | 0.929701 | 1.41616 | 0.305619 | 2.99027 | 3.05275 | 0.440193 |

If we fit a straight line through the origin and the last point ($n=320$) for the free energy it will have formula $0.639913x$, where $x=1/n^2$, which matches well indeed with the value in Ref. [18]. Analogously, for the internal energy we get $0.622437x$, where $x=1/n$, again only a slight deviation in the sixth decimal.

### A. Specific heat

The specific heat should go to infinity with logarithmic speed if we stay close to $K_c$. It was shown by Onsager [3] that

$$\max \mathcal{C}_{n,\infty} = A \ln n + B_\infty + o(1),$$

$$A = \frac{2}{\pi}\left(\ln \cot\frac{\pi}{8}\right)^2 \approx 0.494539,$$

$$B_\infty = A\left(\ln\frac{2^{5/2}}{\pi} + \gamma_E - \frac{\pi}{4}\right) \approx 0.187903,$$

where $\gamma_E \approx 0.5772$ is Euler's gamma. However, it should be noted that the B-constant depends on the shape of the

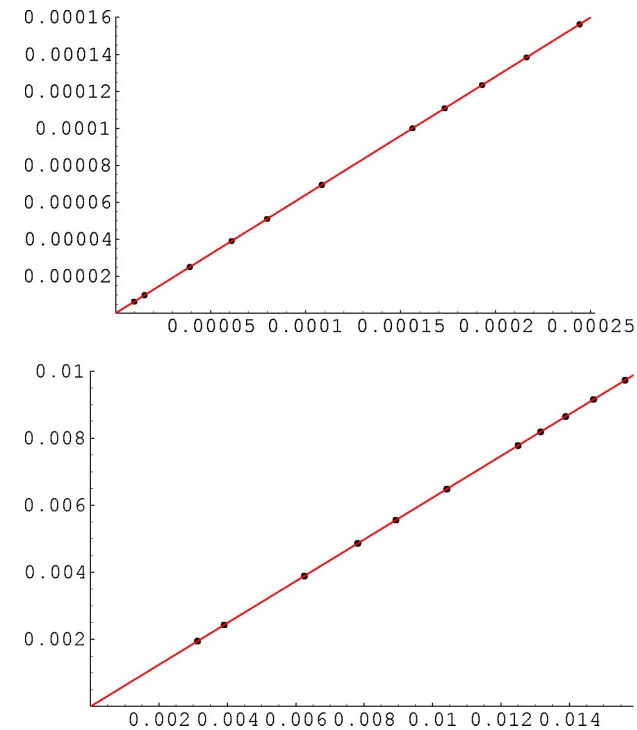


FIG. 3. (Color online) Top: $\mathcal{F}_n(K_c)-F_c$ vs $1/n^2$. Bottom: $\mathcal{U}_n(K_c)-U_c$ vs $1/n$.

grid. Onsager's grid has shape $n\times\infty$. Also, it will depend on whether we are looking at the critical point or at the maximum. For an $(n\times n)$ grid we have, and we quote this from Ref. [18],

$$\max \mathcal{C}_n = A \ln n + B_{\max} + o(1),$$

$$\mathcal{C}_n(K_c) = A \ln n + B_c + o(1),$$

$$K_n^* - K_c \sim \frac{-0.36029K_c}{n} = \frac{-0.15878}{n},$$

where $B_{\max}\approx 0.201359$ and

$$B_c = B_\infty - \frac{\left(\ln \cot\dfrac{\pi}{8}\right)^2}{\theta_2+\theta_3+\theta_4}\left(\frac{4}{\pi}\sum_{i=2}^{4}\theta_i \ln \theta_i + \frac{\theta_2^2\theta_3^2\theta_4^4}{\theta_2+\theta_3+\theta_4}\right)$$

$$\approx 0.138150.$$

The authors of Ref. [18] do not give exact expressions for $B_{\max}$ or the constant $-0.36029$ above.

A curious fact which we would like to mention (see Refs. [3] and [18]) is that for an oblong grid such as an $(n\times\infty)$ grid or indeed, perhaps surprisingly, an $(n\times 3.1393\,n)$ grid the difference between $K^*$ and $K_c$ is of the order $\ln n/n^2$ rather than $1/n$.

So let us compare our data with theory. The upper curve of the top panel in Fig. 4 shows $\max \mathcal{C}_n - A \ln n$ versus $1/n$. A straight line fitted through the last two points ($n=256,320$) gives $0.201274-0.377915x$, where $x=1/n$. Our constant deviates in the fourth decimal from the $B_{\max}$ given in Ref. [18]. The lower curve shows $\mathcal{C}_n(K_c)-A \ln n$ together with its similarly fitted line $0.138149-0.170816x$, a near-perfect match with the constant prescribed above. The bottom panel of Fig. 4 shows how $K_n^*$ differs from $K_c$. A straight line fitted through the origin and the last point ($n=320$) gives $-0.157888x$, again a small deviation. In the plot we use the line $-0.15878x$, a very good fit.

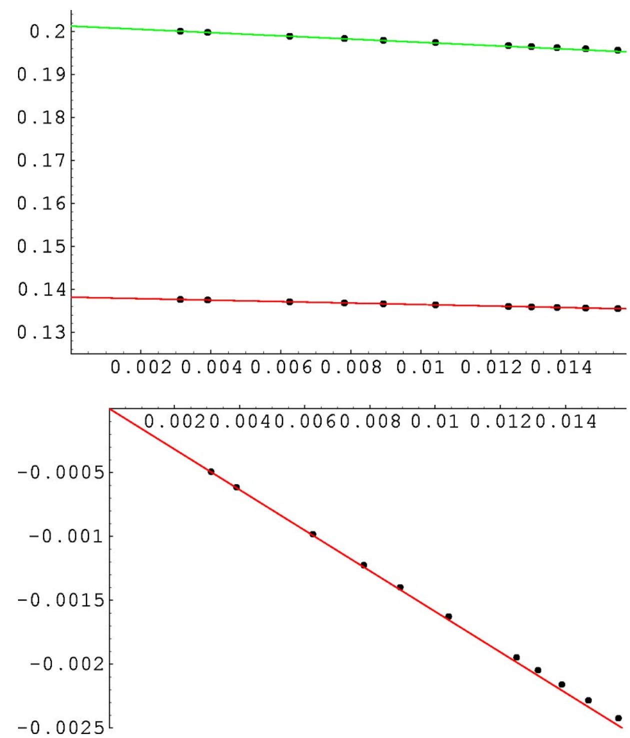


FIG. 4. (Color online) Top: $\max \mathcal{C}_n - A \ln n$ and $\mathcal{C}_n(K_c) - A \ln n$ vs $1/n$. Bottom: $K_n^* - K_c$ vs $1/n$.

## VIII. THE ENTROPY AND ITS DERIVATIVES

In this section we will do a more thorough investigation of the entropy as defined in Eq. (23). To obtain limit curves we will need to translate between relative energy $\nu$ and coupling $K$. This is done with the relation $\nu=\mathcal{U}(K)/2$ where $\mathcal{U}(K)$ is Onsager's formula and this also gives us the critical energy $\nu_c=1/\sqrt{2}\approx 0.7071$. The plots in Figs. 5 and 6 shows the entropy and its derivatives with respect to the relative energy $\nu$. By definition we have $\mathcal{S}=\mathcal{F}-K\mathcal{U}$. If we then associate $\mathcal{S}(K)$ with $\nu(K)$ then we can plot a limit curve of the entropy versus the relative energy. By definition, we also have

$$K = \frac{-1}{2}S'(\nu)$$

and since

$$\mathcal{C} = K^2\frac{\partial \mathcal{U}}{\partial K} = \left(\frac{-1}{2}S'(\nu)\right)^2\frac{1}{\partial K/\partial \mathcal{U}},$$

it follows that

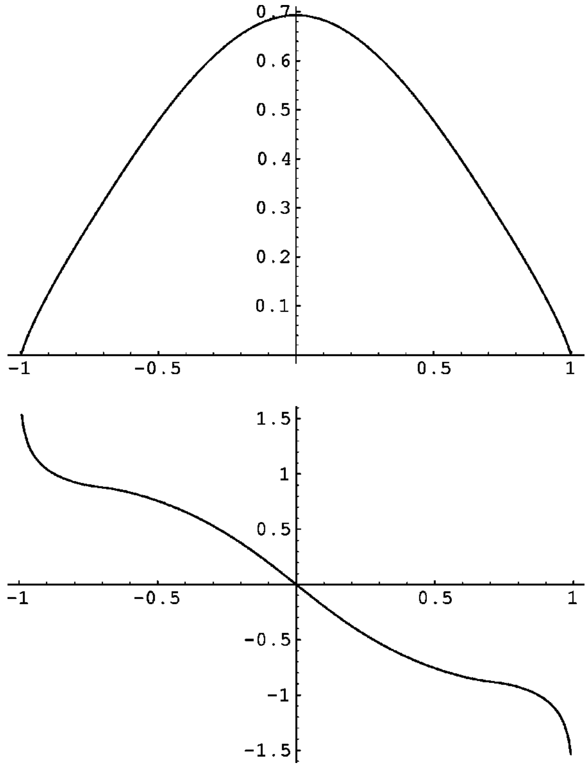


FIG. 5. $S_{320}(\nu)$ and $S'_{320}(\nu)$.

$$\mathcal{C}(\nu)=\frac{1}{4}[S'(\nu)]^2\frac{1}{\frac{-1}{2}S''(\nu)\,\partial\nu/\partial\mathcal{U}}=\frac{-[S'(\nu)]^2}{S''(\nu)},$$

though this is of course only valid for an infinite grid. We can use this last formula, though, to give us a limit curve for the second derivative of the entropy, i.e., asyptotically we have

$$S''(\nu)=\frac{-4}{\mathcal{U}'(K)},$$

which is then plotted versus the energy $\nu(K)$. Continuing in the same spirit with the third derivative we obtain the limit

$$S^{(3)}(\nu)=\frac{8\mathcal{U}''(K)}{[\mathcal{U}'(K)]^3}.$$

These last two formulas are used in the plots of Fig. 6.

Figure 6 shows how the second and third derivatives behave near $\nu_c$. Apparently the second derivative approaches 0 from below. Since the specific heat goes to infinity as $K\to K_c$ for an infinite grid, which corresponds to $\nu\to\nu_c=1/\sqrt{2}$, while $S'\to-2K_c$ it is clear that $S''\to0$ at that point also. Actually, the formula above suggests the following rough estimate:

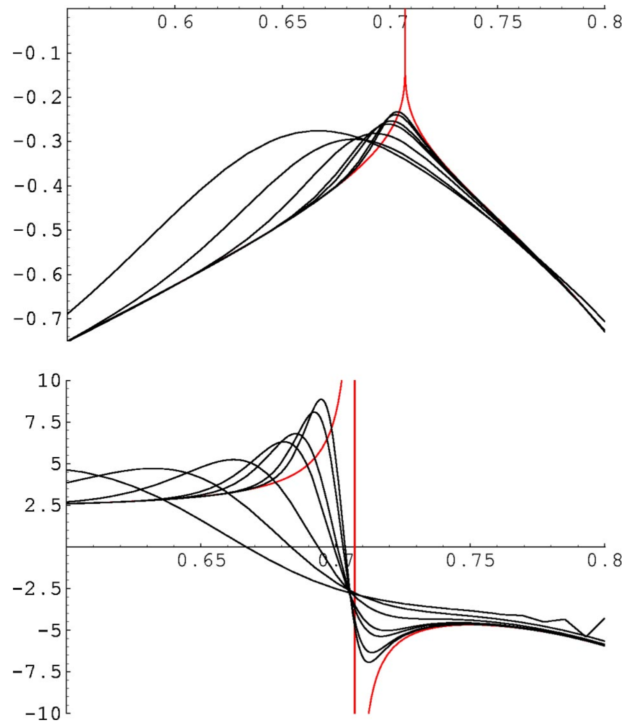


FIG. 6. (Color online) $S''_n(\nu)$, top, and $S^{(3)}_n(\nu)$, bottom, for $n=16,32,64,128,160,256,320,\infty$.

$$S''_n(\nu_c)=\frac{-[S'_n(\nu_c)]^2}{\mathcal{C}_n(K_c)}\sim\frac{-4K_c^2}{A\ln n}=\frac{-\pi}{2\ln n}$$

and of course the same result for the maximum $S''_n$. Figure 8 gives that this could be a reasonable estimate for very large grids though not for $n<1000$. In fact, the maximum has only started to approach zero when $n=32$.

In Fig. 7 we see how the entropy at the critical point $\nu_c$ and its derivative approaches their limits $S_c$ and $-2K_c$, respectively. Beginning with the entropy $S_n(\nu_c)$ one might expect that its behavior would be similar to that of the free energy. However, whereas the difference between the free energy and its critical value is of the order $1/n^2$, the corresponding difference for the entropy seems to be slightly larger, possibly $n^{-9/5}$. For the derivative this difference seems to be of the order of $n^{-5/4}$. In the top panel of Fig. 7 the difference $S_n(\nu_c)-S_c$ versus $n^{-9/5}$ is displayed together with the straight line $-1.91\,x$. The bottom panel shows $S'_n(\nu_c)+2K_c$ versus $n^{-5/4}$ together with $0.425x$.

The top panel of Fig. 8 shows max $S''_n$ versus $1/\ln n$ with the fitted polynomial $-1.56x+0.32x^2+5.4x^3$ and the straight line $-\pi x/2$. A similar behavior is of course found for $S''_n(\nu_c)$ but is better fitted by the polynomial $-1.56x+0.17x^2+4.3x^3$. The bottom panel shows $\nu_n^*-\nu_c$ versus $n^{-5/6}$ and the line $-0.44x$, fitted through the origin and the last point. It should also be stated that the fourth derivative at $\nu^*$ obviously grows to the negative infinity; see the bottom plot of Fig. 6 and the corresponding column in Table II. Its growth rate seems to be on the order of $n^{19/15}$ or thereabout. Assuming this, a straight line fitted through the last four points gives that the fourth

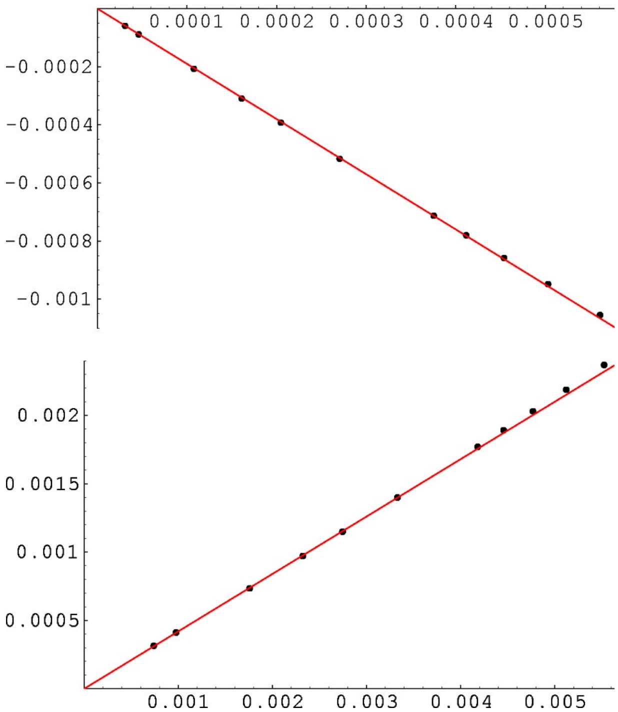


FIG. 7. (Color online) Top: $S_n(\nu_c)-S_c$ vs $n^{-9/5}$. Bottom: $S_n'(\nu_c)+2K_c$ vs $n^{-5/4}$.

derivative at $\nu_n^*$ is $-65-1.03n^{19/15}$; see top plot of Fig. 9. The bottom plot shows $K_n(\nu^*)-K_c$ for each grid versus $n^{-21/20}$ and the line $-0.249x$.

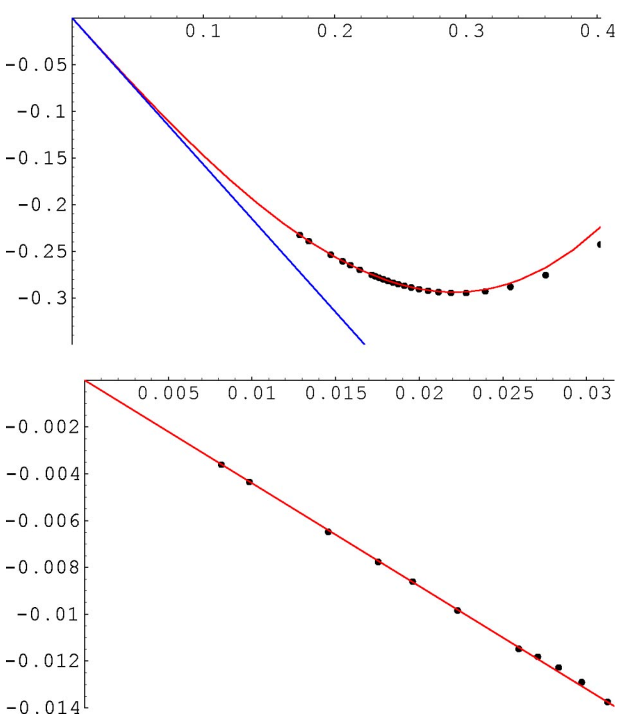


FIG. 8. (Color online) Top: max $S_n''$ vs $1/\ln n$. Bottom: $\nu_n^*-\nu_c$ vs $n^{-5/6}$.

TABLE II. Entropy data.

| $n$ | $S_n(\nu_c)$ | $-S_n'(\nu_c)$ | $-S_n''(\nu_c)$ | $-\max S_n''$ | $-S_n^{(4)}(\nu_n^*)$ | $\nu_n^*$ |
|---|---|---|---|---|---|---|
| 12 | 0.289122 | 0.855602 | 0.328177 | 0.242751 | 87.6930 | 0.652778 |
| 16 | 0.295499 | 0.864776 | 0.340533 | 0.275587 | 95.2653 | 0.664062 |
| 20 | 0.298843 | 0.869474 | 0.341183 | 0.288223 | 104.431 | 0.675000 |
| 24 | 0.300829 | 0.872249 | 0.338620 | 0.293102 | 119.379 | 0.677083 |
| 28 | 0.302111 | 0.874054 | 0.334909 | 0.294729 | 132.302 | 0.681122 |
| 32 | 0.302991 | 0.875309 | 0.330880 | 0.294656 | 146.386 | 0.683594 |
| 36 | 0.303622 | 0.876225 | 0.326890 | 0.293712 | 161.361 | 0.685185 |
| 40 | 0.304091 | 0.876920 | 0.323030 | 0.292304 | 175.026 | 0.687500 |
| 44 | 0.304450 | 0.877463 | 0.319365 | 0.290661 | 189.577 | 0.689050 |
| 48 | 0.304732 | 0.877897 | 0.315910 | 0.288886 | 204.967 | 0.690104 |
| 52 | 0.304957 | 0.878252 | 0.312658 | 0.287076 | 221.071 | 0.690828 |
| 56 | 0.305139 | 0.878546 | 0.309620 | 0.285273 | 236.062 | 0.691964 |
| 60 | 0.305290 | 0.878793 | 0.306759 | 0.283497 | 251.792 | 0.692778 |
| 64 | 0.305416 | 0.879004 | 0.304070 | 0.281763 | 268.220 | 0.693359 |
| 68 | 0.305522 | 0.879186 | 0.301537 | 0.280082 | 283.638 | 0.694204 |
| 72 | 0.305612 | 0.879344 | 0.299146 | 0.278455 | 299.752 | 0.694830 |
| 76 | 0.305690 | 0.879482 | 0.296885 | 0.276878 | 316.558 | 0.695291 |
| 80 | 0.305757 | 0.879604 | 0.294745 | 0.275359 | 333.999 | 0.695625 |
| 96 | 0.305954 | 0.879974 | 0.287189 | 0.269811 | 402.297 | 0.697266 |
| 112 | 0.306077 | 0.880224 | 0.280900 | 0.265004 | 473.012 | 0.698501 |
| 128 | 0.306161 | 0.880403 | 0.275552 | 0.260798 | 547.528 | 0.699341 |
| 160 | 0.306263 | 0.880639 | 0.266858 | 0.253768 | 703.101 | 0.700625 |
| 256 | 0.306381 | 0.880962 | 0.249667 | 0.239297 | 1219.53 | 0.702759 |
| 320 | 0.306411 | 0.881060 | 0.242063 | 0.232702 | 1603.41 | 0.703496 |

## IX. THE LOG-CONCAVITY POINT

Here we take a quick look at a finite-size phenomena which occurs at high energies. If we consider the plot in Fig. 10 of the coupling $K_{16}(\nu)=-S_{16}'(\nu)/2$ we note an irregular behavior at about $\nu\approx 0.87$. For larger grids this will move closer to 1.

This is the energy where the sequence $a_i$ stops being log concave. We will define this point as the largest $\nu=i/2n^2$ such that $a_{i-4}a_{i+4}\leqslant a_i^2$ and denote it by $\tilde{\nu}_n$. The table in Fig. 10 shows where this energy is located. In Fig. 11 we see $1-\tilde{\nu}_n$ versus $n^{-19/15}$ together with the line $3.96x$. The coupling $K_n=-S_n'/2$ corresponding to this energy is displayed in the bottom plot with the line (through $n=256, 320$) $0.030+0.155x$.

That $K$ in this case grows as $O(\ln n)$ is perhaps not very surprising. Note that for high energies we know the sequence of $a_i$. Counting backwards from $i=2n^2$ the $a_i$ sequence begins $2,0,2n^2,4n^2,n^4+9n^2,\ldots$ . It seems also that the largest value of $\frac{1}{4}\ln[a_i/(a_{i+4})]$ is obtained for $i=2n^2-16$ giving the coupling value $\frac{1}{4}\ln[(n^2+9)/4]\sim\frac{1}{2}\ln n$.

## X. THE LARGEST COEFFICIENT

In this section we will take a look at the largest coefficient of the partition function. For all grids we have looked at, this position is held by coefficient $a_0$. However, proof that this is

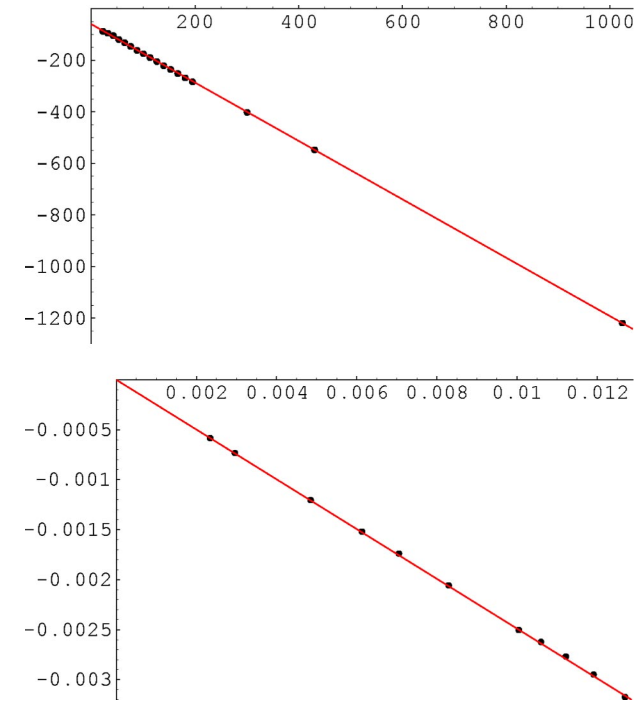


FIG. 9. (Color online) Top: $S^{(4)}(\nu_n^*)$ vs $n^{19/15}$. Bottom: $K_n(\nu_n^*) - K_c$ vs $n^{-21/20}$.

generally true is still lacking. It seems fairly safe though to assume, as we will here, that $\max_i a_i = a_0$. We begin by setting up two easy bounds. First, obviously we have

$$a_0 \leqslant \sum_i a_i = 2^{n^2}.$$

Second, the energy levels can take the values $0, \pm 4, \ldots, \pm(2n^2-8), \pm 2n^2$, i.e., there are $n^2-1$ energies. If we distribute the mass $2^{n^2}$ on these levels then some coefficient must be at least average, i.e.,

$$\frac{2^{n^2}}{n^2} \leqslant \frac{2^{n^2}}{n^2-1} \leqslant a_0.$$

It would seem appropriate to guess that $a_0$ is of the intermediate order $2^{n^2}/n$. As we will see, mutatis mutandis, this is just about perfect. The correct quantity to study is

$$Q_n = \frac{a_0}{\binom{n^2}{n^2/2}},$$

where, by Stirling's formula

$$\binom{n^2}{n^2/2} \sim \sqrt{\frac{2}{\pi}}\frac{2^{n^2}}{n},$$

that is, the guess from above.

The table and the plot in Fig. 12 give rather strong evidence that $Q_n \to \sqrt{2}$. They are well fitted by the line $\frac{7}{8}\sqrt{2}x$. To conclude, we conjecture that

| $n$ | $\tilde{\nu}_n$ | $K_n(\tilde{\nu}_n)$ |
|---|---|---|
| 32 | 0.949219 | 0.583526 |
| 40 | 0.962500 | 0.616059 |
| 48 | 0.970486 | 0.642342 |
| 64 | 0.979492 | 0.683126 |
| 80 | 0.984375 | 0.714141 |
| 96 | 0.987847 | 0.743054 |
| 112 | 0.989796 | 0.763426 |
| 128 | 0.991455 | 0.784164 |
| 160 | 0.993594 | 0.818101 |
| 256 | 0.996460 | 0.888827 |
| 320 | 0.997344 | 0.923393 |

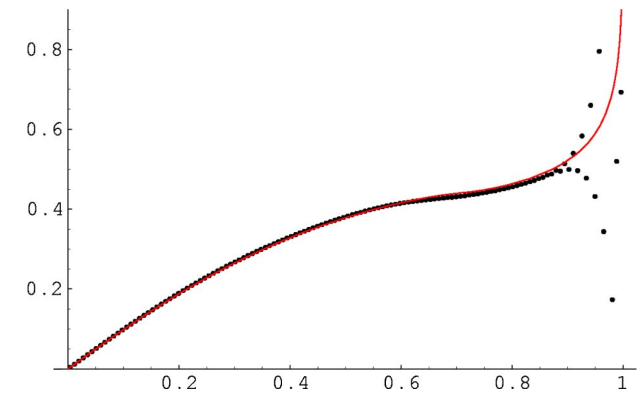


FIG. 10. (Color online) Top: Data on $\tilde{\nu}_n$. Bottom: $K_n(\nu)$ for $n=16$ and $n=\infty$.

$$a_0 = \frac{2}{\sqrt{\pi}}\frac{2^{n^2}}{n}\left[1 + \frac{7}{8n^2} + O\left(\frac{1}{n^3}\right)\right].$$

## XI. ASYMPTOTICS

Here we collect all statements on asymptotic behavior which are spread out through the text. Exact formulas for the first four are given elsewhere in the article.

$$\mathcal{F}_n(K_c) - F_c \sim 0.639912n^{-2},$$

$$\mathcal{U}_n(K_c) - U_c \sim 0.622439\ n^{-1},$$

$$\mathcal{S}_n(K_c) - S_c = -0.274301n^{-1} + 0.639912n^{-2} + O(n^{-3}),$$

$$\mathcal{C}_n(K_c) = 0.494539 \ln n + 0.138150 + o(1),$$

$$\max \mathcal{C}_n = 0.494539 \ln n + 0.201359 + o(1),$$

$$K_n^* - K_c \sim -0.15878n^{-1}.$$

The following asymptote approximations should be considered conjectural, i.e., guessed up to the given precision. A

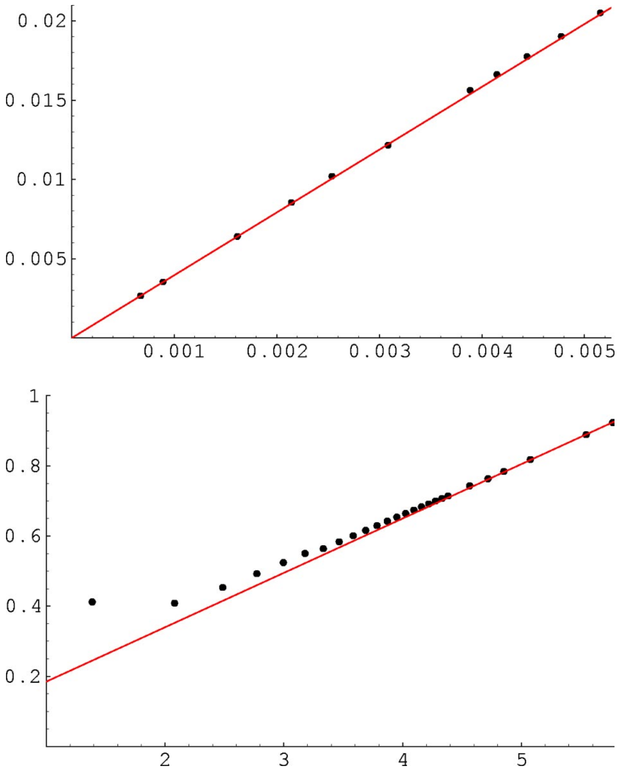


FIG. 11. (Color online) Top: $1-\tilde{\nu}_n$ vs $n^{-19/15}$. Bottom: $K_n(\tilde{\nu}_n)$ vs $\ln n$.

similar caveat applies to the exponents on $n$; they are simply chosen among the rationals with small denominator:

$$S_n(\nu_c) - S_c \sim -1.91 n^{-9/5},$$

$$S_n'(\nu_c) + 2K_c \sim 0.425 n^{-5/4},$$

$$S_n''(\nu_c) \approx \frac{-1.56}{\ln n} + \frac{0.17}{\ln^2 n} + \frac{4.3}{\ln^3 n},$$

$$\max S_n''(\nu) \approx \frac{-1.56}{\ln n} + \frac{0.32}{\ln^2 n} + \frac{5.4}{\ln^3 n},$$

$$\max S_n''(\nu) \sim S_n''(\nu_c) \sim \frac{-\pi}{2\ln n},$$

$$S_n^{(4)}(\nu_n^*) \sim -1.03 n^{19/15},$$

$$K_n(\nu_n^*) - K_c \sim -0.249 n^{-21/20},$$

$$\nu_n^* - \nu_c \sim -0.442 n^{-5/6},$$

$$1 - \tilde{\nu}_n \sim 3.96 n^{-19/15},$$

$$K_n(\tilde{\nu}_n) \sim 0.155 \ln n,$$

| $n$ | $Q_n$ |
|---|---|
| 32 | 1.415424 |
| 40 | 1.414988 |
| 48 | 1.414751 |
| 64 | 1.414516 |
| 80 | 1.414407 |
| 96 | 1.414348 |
| 112 | 1.414312 |
| 128 | 1.414289 |
| 160 | 1.414262 |
| 256 | 1.414232 |
| 320 | 1.414226 |

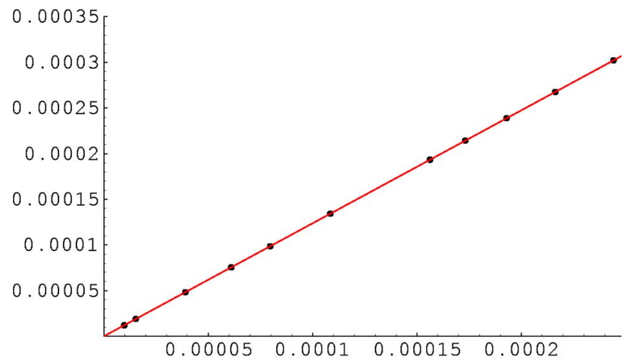


FIG. 12. (Color online) Top: Data on $Q_n$. Bottom: $Q_n-\sqrt{2}$ vs $1/n^2$.

$$a_0 = \frac{2}{\sqrt{\pi}} \frac{2^{n^2}}{n}\left[1 + \frac{7}{8n^2} + O\left(\frac{1}{n^3}\right)\right].$$

## ACKNOWLEDGMENTS

This work was supported by the Swedish Natural Science Research Council, The Göran Gustafsson Foundation, and the National Graduate School in Scientific Computing (NGSSC). We used the computing facilities at Department of Mathematics, Umeå University, HPC2N Umeå, NSC Linköping, Sweden, and PDC Stockholm, Sweden.

## APPENDIX A: CHEBYSHEV POLYNOMIALS

We will now develop some facts about Chebyshev polynomials that we make use of in the main body of the paper. For further information we recommend Ref. [19]. We begin with some basics.

*Definition A.1.* The Chebyshev polynomials of the first kind are defined as

$$\mathrm{T}_n(x) = \cos(n \arccos x) = \cos n\theta, \quad x = \cos\theta. \quad \text{(A1)}$$

*Definition A.2.* The Chebyshev polynomials of the second kind are defined as

$$\mathrm{U}_{n-1}(x)=\frac{\sin(n\ \arccos\ x)}{\sqrt{1-x^2}}=\frac{1}{n}\mathrm{T}_n'(x)=\frac{\sin\ n\theta}{\sin\ \theta},$$

$$x=\cos\ \theta.$$

A useful fact which follows directly from the definition is that

$$T_n[\cos(x)]=\cos(nx).$$

Since $\mathrm{T}_n(x)=\cos\ n\theta$ and $\cos\ n\theta_j=0$ for

$$\theta_j=\theta_j^{(n)}=\frac{(2j-1)\pi}{2n},\quad j=1,\ldots,n,$$

we see that the points

$$\xi_j=\xi_j^{(n)}=\cos\ \theta_j^{(n)}=\cos\frac{(2j-1)\pi}{2n},\quad j=1,\ldots,n$$

satisfy

$$\mathrm{T}_n(\xi_j)=0,\quad j=1,\ldots,n.$$

From this we can factor $T_n(x)$ as

$$T_n(x)=2^{n-1}\prod_{j=1}^{n}\left(x-\cos\frac{(2j-1)\pi}{2n}\right)\tag{A2}$$

and $U_n(x)$ as

$$U_n(x)=2^{n}\prod_{j=1}^{n}\left(x-\cos\frac{j\pi}{n+1}\right).\tag{A3}$$

#### 1. Extremal points

It is also clear from Eq. (A1) that $|\mathrm{T}_n(x)|\leqslant 1$ if $|x|\leqslant 1$. The points in this interval, when $|T_n(x)|=1$, are called the *extrema* of $T_n(x)$. We know that $\cos\ k\pi=(-1)^k$ for any integer $k$ so if

$$\phi_k=\phi_k^{(n)}=\frac{k\pi}{n},\quad k=0,1,\ldots,n,$$

the points

$$\eta_k=\eta_k^{(n)}=\cos\ \phi_k^{(n)}=\cos\frac{k\pi}{n},\quad k=0,1,\ldots,n$$

satisfy

$$T_n(\eta_k)=(-1)^k,\quad k=0,1,\ldots,n.$$

This gives us the following products on closed form:

$$\prod_{k=1}^{n}2\left(x-\cos\frac{2\pi k}{n}\right)=2[\mathrm{T}_n(x)-1]\tag{A4}$$

and

$$\prod_{k=1}^{n}2\left(x-\cos\frac{\pi(2k-1)}{n}\right)=2[\mathrm{T}_n(x)+1].\tag{A5}$$

#### 2. The coefficients

If $|t|<1$ then

$$\sum_{n\geqslant 0}t^n e^{in\theta}=\sum_{n\geqslant 0}(te^{i\theta})^n=\frac{1}{1-te^{i\theta}}$$

$$=\frac{1}{1-t(\cos\ n\theta)+i\ \sin\ n\theta}$$

$$=\frac{1-t\ \cos\ n\theta+ti\ \sin\ n\theta}{(1-t\ \cos\ n\theta)^2+t^2\ \sin\ n\theta}$$

$$=\frac{1-t\ \cos\ n\theta+ti\ \sin\ n\theta}{1-2t\ \cos\ n\theta+t^2}.$$

On equating the real parts, we obtain

$$\sum_{n\geqslant 0}t^n\cos\ n\theta=\frac{1-t\ \cos\ \theta}{1+t^2-2t\ \cos\ \theta}$$

or

$$\frac{1-tx}{1-2tx+t^2}=\sum_{n\geqslant 0}t^n\mathrm{T}_n(x),$$

the generating function for $\mathrm{T}_n(x)$. Using the definition we find the generating function for $\mathrm{U}_n(x)$:

$$\frac{1}{1-2tx+t^2}=\sum_{n\geqslant 0}t^n\mathrm{U}_n(x).$$

From this we obtain the following lemma.

*Lemma A.3. The polynomials $2T_n(x/2)$ and $U_n(x/2)$ have integer coefficients.*

*Proof*. Using the generating function for $\mathrm{U}_n(x/2)$ we have

$$\frac{1}{1-2t\frac{x}{2}+t^2}=\frac{1}{1-tx+t^2}=\frac{1}{1-t(x-t)}$$

$$=\sum_{k\geqslant 0}[t(x-t)]^k=\sum_{k\geqslant 0}t^k(x-t)^k$$

and for fixed $n$ the coefficients for $t^n$ are polynomials in $x$ with integer coefficients. Multiplying by $1-t(x/2)$ gives the result for $2\mathrm{T}_n(x/2)$. ■

We can use the formula in the proof above to explicitly give the coefficients for the Chebyshev polynomials as

$$\mathrm{T}_n(x)=\frac{1}{2}\sum_{k=0}^{\lfloor n/2\rfloor}(-1)^k\frac{n}{n-k}\binom{n-k}{k}(2x)^{n-2k},$$

$$\mathrm{U}_n(x)=\sum_{k=0}^{\lfloor n/2\rfloor}(-1)^k\binom{n-k}{k}(2x)^{n-2k}.$$

#### 3. The irreducible factors

We will now describe the irreducible factors of the Chebyshev polynomials. We will state the results without proofs, which the interested reader can find in Ref. [19].

Rather than factoring the Chebyshev polynomials themselves we will give the irreducible factors of $C_k(x) = 2\mathrm{T}_k(x/2)$ and $S_k(x) = \mathrm{U}_k(x/2)$, for $k > 0$. From lemma A.3 we know that these polynomials are monic and have integer coefficients.

Given an odd divisor $h$ of $k$, let

$$F_{h,k}(x) = \prod_{\mathrm{GCD}(2j-1,2k)=h, 1\leqslant j\leqslant k} \left[x - 2\cos\left(\frac{(2j-1)\pi}{2k}\right)\right].$$

Now $F_{h,k}(x)$ will be an irreducible monic polynomial with integer coefficients and

$$C_k(x) = \prod_{h|k, h \text{ odd}} F_{h,k}(x).$$

Given a divisor $h$ of $2(k+1)$, let

$$G_{h,k}(x) = \prod_{\mathrm{GCD}[j,2(k+1)]=h, 1\leqslant j\leqslant k} \left[x - 2\cos\left(\frac{j\pi}{k+1}\right)\right].$$

Here $G_{h,k}(x)$ will be an irreducible monic polynomial with integer coefficients and

$$S_k(x) = \prod_{h|[2(k+1)], 1\leqslant h\leqslant k} G_{h,k}(x).$$

#### 4. Two useful identities

We will also need the following facts about the Chebyshev polynomials.

*Lemma A.4. Let $T_n(x)$ and $U_n(x)$ be the Chebyshev polynomials of the first and second kind. Then for $n \geqslant 1$ we have the following. For even indices of $T_n(x)$*

$$2[\mathrm{T}_{2(n+1)}(x) - 1] = 4(x^2-1)\mathrm{U}_n^2(x)2[\mathrm{T}_{2n}(x) + 1] = 4\mathrm{T}_n^2(x)$$

*and for odd indices of $T_n(x)$*

$$1 + \mathrm{T}_{2n+1}(x) = (1+x)[\mathrm{U}_n(x) - \mathrm{U}_{n-1}(x)]^2,$$

$$1 - \mathrm{T}_{2n+1}(x) = (1-x)[\mathrm{U}_n(x) + \mathrm{U}_{n-1}(x)]^2.$$

*Proof.* Even indices:

$$\begin{aligned}
2[\mathrm{T}_{2(n+1)}(x) - 1] &= 2^{2(n+1)} \prod_{k=1}^{2(n+1)} \left(x - \cos\frac{2\pi k}{2(n+1)}\right) \\
&= 2^n \prod_{k=1}^{n} \left(x - \cos\frac{\pi k}{n+1}\right) 2^n \prod_{k=n+2}^{2n+1} \left(x - \cos\frac{\pi k}{n+1}\right) 2^2 \left(x - \cos\frac{\pi(n+1)}{(n+1)}\right)\left(x - \cos\frac{2\pi(n+1)}{(n+1)}\right) \\
&= 4(x^2-1)\mathrm{U}_n^2(x) 2[T_{2n}(x) + 1] = 2^{2n} \prod_{k=1}^{2n} \left(x - \cos\frac{\pi(2k-1)}{2n}\right) \\
&= 4(2^{n-1}) \prod_{k=1}^{n} \left(x - \cos\frac{\pi(2k-1)}{2n}\right) 2^{n-1} \prod_{k=n+1}^{2n} \left(x - \cos\frac{\pi(2k-1)}{2n}\right) = 4\mathrm{T}_n^2(x).
\end{aligned}$$

Odd indices:

$$\begin{aligned}
(1 \pm x)[\mathrm{U}_n(x) \mp \mathrm{U}_{n-1}(x)]^2 &= (1\pm x)[\mathrm{U}_n^2(x) + \mathrm{U}_{n-1}^2(x) \mp 2\mathrm{U}_n(x)\mathrm{U}_{n-1}(x)] \\
&= \frac{1\pm x}{1-x^2}\left\{[1 - \mathrm{T}_{n+1}^2(x)] + [1 - \mathrm{T}_n^2(x)] \mp 2\tfrac{1}{2}[T_1(x) - T_{2n+1}(x)]\right\} \\
&= \frac{1\pm x}{1-x^2}\left\{2 - \tfrac{1}{2}[\mathrm{T}_{2n+2}(x) + 1] - \tfrac{1}{2}[\mathrm{T}_{2n}(x) + 1] \mp x \pm \mathrm{T}_{2n+1}(x)\right\} \\
&= \frac{1\pm x}{1-x^2}\left\{(1 \mp x) - \tfrac{1}{2}[\mathrm{T}_{2n+2}(x) + \mathrm{T}_{2n}(x)] \pm \mathrm{T}_{2n+1}(x)\right\} \\
&= \frac{1\pm x}{1-x^2}\left\{(1 \mp x) - \tfrac{1}{2}[\mathrm{T}_{2n+1}(x) - \mathrm{T}_{2n}(x) + \mathrm{T}_{2n}(x)] \pm \mathrm{T}_{2n+1}(x)\right\} \\
&= \frac{1\pm x}{1-x^2}[(1 \mp x) - x\mathrm{T}_{2n+1}(x) \pm \mathrm{T}_{2n+1}(x)]
\end{aligned}$$

$$= \frac{1 \pm x}{1 - x^2}[(1 \mp x) + (1 \mp x)\mathrm{T}_{2n+1}(x)]$$

$$= \frac{(1+x)(1-x)}{1-x^2}[1 \pm \mathrm{T}_{2n+1}(x)] = 1 \pm \mathrm{T}_{2n+1}(x) . \qquad \blacksquare$$

## APPENDIX B: AN IMPLEMENTATION IN MATHEMATICA

In this appendix we demonstrate a MATHEMATICA program which implements some of the calculations discussed in the main text. This implementation works for grids with even side $n$ and uses numerical evaluation at the stage where $\gamma_t$ is eliminated. The only optimization from the paper used here is our formulation of the products in terms of Chebyshev polynomials. For sides less than about $n=80$ it is actually faster to use a direct numerical evaluation of the cosine-terms before the multiplication is performed. However, once we get to around $n=80$ the need for high precision numerics makes that numerical version slower than the version shown here.

A notebook demonstrating the Galois method is also available at the paper's homepage at http://abel.math.umu.se/Combinatorics/ising.html

Let us start with the definition of the functions we will use later to compute our three products $P_1$, $P_2$, and $P_4$.

```
Multiply [p_c,c_]: =
(* Applying the product rule for gamma_t *)
Module [{i, j, m, b},
FixedPoint [
Expand [#]
/.{
Power [c[i_,m_],b_]:>
(c[i,m])∧Mod[b,2]*(2+c[zi, m])∧Floor[b/2]/;b>=2,
c[i_,m_]*c[j_,m_]:>c[i+j,m]+c[i−j,m]}&,
p]
]
SymReduce[p_,c_]:= (*Reducing by symmetries*)
Module [{i,m},
p
/.c[i_,m_]:>c[−i,m]/;i<0
//.c[i_,m_]:>c[i−2m,m]/;i>m2m
/.c[i_,m_]:>c[2m−i,m]/;i>m
/.{
c[i_,m_]:>0/;2i==m,c[i_,m_]:>1/;3i==m,
c[i_,m_]:>−1/;3i==2m,c[m_,m_]−>−2,c[0,_}−>2
}
]
RemoveCos[p_,c−,acc_]:=
Module[{i,j,x},
p/.c[i_,j_]:>N[2*Cos[i*Pi/j},acc]/.x_Real:>Round[x]
]
TakeProduct[polys_,e_,Y_,c_,z_,acc_]:=
Module[{a,b,prod},
prod=Fold[SymReduce[Multiply[#1*#2,c],c]&,1,polys];
Cancel[b∧e*RemoveCos[prod,c,acc]/.Y−>a∧2/b]
/.{a−>(1+z∧2),b−>z(1−z∧2)}
]
```

The function "Multiply" implements the multiplication and squaring rules for $\lambda_t$. "SymReduce" uses the symmetries of cos to reduce the number of $\lambda_t$ variables needed. "RemoveCos" uses high precision floating point arithmetic (of accuracy "acc") to evaluate the cos functions and then rounds the answer to the nearest integer. Finally, "TakeProduct" takes a list of polynomials in the variables Y and c (where c [i,j] represents 2 $\cos(i\pi/j)$, multiplies them together, evaluates the cos functions using "RemoveCos," and finally does the substitution $Y\rightarrow(1+z^2)^2/z(1-z^2)$ while multiplying with a high enough power of $z(1-z^2)$:

To be able to check the result later we also need the function $U(K)$ defined as follows:

```
U[K_]=Fullsimplify[
Coth[2K](1+2/Pi*EllipticK[z∧2](2*Tanh[2K]∧2−1))
/.z−>2*Sinh[2K]/Cosh[2K]∧2,Element[K,Reals]
]
```

Let us do a worked example of how to use these functions to compute a partition function and check the result. We begin by defining the size of our square grid:

```
(************Input*************)
n=50.
```

This is the only parameter we need to set ourself, everything else can now be calculated from this. We next calculate some constants and the two polynomials $\mathrm{U}_{p-1}(X_t/2)$ and $2\mathrm{T}_p(X_t/2)$ for a general $t$:

```
(*************** I **************)
p=n/2;
acc=Floor[N[p∧2*Log[10,2]]];
U[Y_,t_]=
SymReduce[Multiply[ChebyshevU[p−1,(y−a[t,n])/
2],a],a];
T[Y_,t]=
SymReduce[Multiply[2*ChebyshevT[p,(Y−a[t,n])/
2],a],a]
```

We can now calculate $A_1$, making use of our functions "TakeProduct" and "SymReduce". This is done in two steps since we need to multiply with the appropriate "prefactors." In this case $p^2-1$ is a large enough power of $z(1-z^2)$.

```
(************* A1) *************)
Module[{A1prod,A1},
A1prod=TakeProduct[
Table[SymReduce[U[Y,2i],a],{i,0,p}]
,p∧2−1,Y,a,z,acc
];
A1=Expand[
```

```
(1+z∧2)2(−1−2z+z∧2)(−1+2z+z∧2)*A1prod∧2
];
Z=A1;
]
```

We now calculate $A_2$ in much the same way as $A_1$. The differences are the prefactors and that now the power of $z(1-z^2)$ is $p^2-p$ for the bulk of the polynomials and $n=2p$ for the factors. We also add $2A_2$ to $Z$ since $A_2=A_3$ for a square grid and we do not want to waste precious time calculating $A_3$ separately:

```
(**********A2)*********)
Module[{A2prod,A2pre,A2},
A2prod=TakeProduct[
Table[SymReduce[T[Y,2i],a],{i,1,p−1}],
p∧2−p,Y,a,z,acc
];
A2pre,TakeProduct[
{SymReduce[T[Y,n],a],SymReduce[T[Y,0],a]},
n,Y,a,z,acc
];
A2=Expand[A2pre*A2prod∧2];
Z=Z+2*A2;
]
```

$A_4$ is the simplest term to calculate since it does not need any prefactors and such. The power of $z(1-z^2)$ is $p^2$:

```
(**********A4)*********)
Module[{A4prod,A4},
A4prod=TakeProduct[
Table[SymReduce[T[Y,2i+1],a],{i,0,p−1}],
p∧2,Y,a,z,acc
];
A4=Expand[A4prod∧2];
Z=Expand[(Z+A4/2];
]
```

Finally we verify the correctness of our resulting polynomial by calculating the moment generating function for the distributions of energies and compare it with the infinite grid:

```
(************** Check *************)
s1=Simplify/@Integrate[Series[U[K],{K,0,n−1}],K];
s2=Simplify/@Series[Exp[n∧2*s1],{K,0,n−1}];
s3=Simplify/@Series[Z/(2z)∧(n∧2)/.z−
>Exp[K]∧2,{K,0,n−1}];
s2==s3
True.
```

As you can see, the two expressions are equal and it is unlikely that any computational errors have occurred. In Table III we give timings for various grid sizes run on a Linux machine with an Athlon 2000+ and 2 Gb RAM. We have also included timings of Beale's implementation, run on the same machine.

TABLE III. Timing data.

| $n$ | Our's | Beale's | Ratio |
|---|---|---|---|
| 8 | 0.1 | 0.5 | 5 |
| 16 | 2.0 | 11.0 | 5.5 |
| 24 | 22.0 | 95.0 | 4.3 |
| 32 | 143.1 | 622.7 | 4.4 |
| 40 | 835.5 | 3010.5 | 3.6 |
| 48 | 2675.2 | 11223.9 | 4.2 |
| 56 | 8111.4 | 38118.9 | 4.7 |
| 64 | 20006.5 | 108331.3 | 5.4 |

[1] W. Lenz, Phys. Z. **21**, 613 (1920).
[2] E. Ising, Z. Phys. **31**, 253 (1925).
[3] L. Onsager, Phys. Rev. **65**, 117 (1944).
[4] B. Kaufman, Phys. Rev. **76**, 1232 (1949).
[5] P. Beale, Phys. Rev. Lett. **76**, 78 (1996).
[6] P. Kasteleyn, in *Graph Theory and Theoretical Physics* (Academic, New York, 1967).
[7] B. McCoy and T. Wu, *The Two-dimensional Ising Model* (Harvard University Press, Boston, 1973).
[8] B. Hartley and T. O. Hawkes, *Rings, Modules and Linear Algebra* (Chapman and Hall, London, 1980). A further course in algebra describing the structure of abelian groups and canonical forms of matrices through the study of rings and modules, a reprinting.
[9] A. Graham, *Kronecker Products and Matrix Calculus: With Applications*, Ellis Horwood Series in Mathematics and its Applications (Ellis Horwood, Chichester, 1981).
[10] I. Stewart, *Galois Theory*, 2nd ed. (Chapman and Hall, London, 1989).
[11] C. P. Bonnington and C. H. C. Little, *The Foundations of Topological Graph Theory* (Springer-Verlag, New York, 1995).
[12] M. Laczkovich, *Conjecture and Proof* (TypoTEX Kft. Elektronikus Kiadó, Budapest, 1998).
[13] D. E. Knuth, *The Art of Computer Programming*, 2nd ed. (Addison-Wesley, Reading, MA, 1981), Vol. 2.
[14] D. H. Bailey, ACM Trans. Math. Softw. **21**(4), 379 (1995).
[15] B. Cipra, Am. Math. Monthly **94**, 937 (1987).
[16] G. R. Grimmett and D. R. Stirzaker, *Probability and Random Processes*, 2nd ed. (Clarendon Press, New York, 1992).
[17] D. Lavis and G. Bell, *Statistical Mechanics of Lattice Systems* (Springer, Berlin, 1999).
[18] A. Ferdinand and M. Fisher, Phys. Rev. **185**, 832 (1969).
[19] T. J. Rivlin, *Chebyshev Polynomials, Pure and Applied Mathematics*, 2nd ed. (Wiley, New York, 1990).
[20] In the algebraic meaning of the word, not a physical field.
[21] A Euler subgraph is a subgraph where all vertices have even degree.