

Percolation thresholds on two-dimensional Voronoi networks and Delaunay triangulations

Adam M. Becker*

Department of Physics, University of Michigan, Ann Arbor, Michigan 48109-1040, USA

Robert M. Ziff†

Center for the Study of Complex Systems and Department of Chemical Engineering, University of Michigan, Ann Arbor, Michigan 48109-2136, USA

(Received 26 June 2009; revised manuscript received 26 August 2009; published 1 October 2009)

The site percolation threshold for the random Voronoi network is determined numerically, with the result $p_c = 0.714\,10 \pm 0.000\,02$, using Monte Carlo simulation on periodic systems of up to 40 000 sites. The result is very close to the recent theoretical estimate $p_c \approx 0.7151$ of Neher *et al.* For the bond threshold on the Voronoi network, we find $p_c = 0.666\,931 \pm 0.000\,005$ implying that, for its dual, the Delaunay triangulation $p_c = 0.333\,069 \pm 0.000\,005$. These results rule out the conjecture by Hsu and Huang that the bond thresholds are $2/3$ and $1/3$, respectively, but support the conjecture of Wierman that, for fully triangulated lattices other than the regular triangular lattice, the bond threshold is less than $2 \sin \pi/18 \approx 0.3473$.

DOI: [10.1103/PhysRevE.80.041101](https://doi.org/10.1103/PhysRevE.80.041101)

PACS number(s): 64.60.ah, 64.60.aq, 46.65.+g

I. INTRODUCTION

The Voronoi diagram [1] for a given set of points on a plane (Fig. 1) is simple to define. Given some set of points P on a plane \mathbb{R}^2 , the Voronoi diagram divides the plane \mathbb{R}^2 into polygons, each containing exactly one member of P . Each point's polygon cordons off the portion of \mathbb{R}^2 that is closer to that point than to any other member of P . More precisely, the Voronoi polygon around $p_i \in P$ contains all locations on \mathbb{R}^2 that are closer to p_i than to any other element of P . The total Voronoi diagram is the set of all the Voronoi polygons for P on \mathbb{R}^2 ; the Voronoi *network* is the set of vertices and edges of the Voronoi diagram.

The dual to the Voronoi diagram is interesting in its own right. Known as the Delaunay triangulation [2] (see Fig. 2), it can be defined independent of the Voronoi diagram for the same set of points P on \mathbb{R}^2 : it is simply the set of all possible triangles formed from triples chosen out of P whose circumscribed circles do not contain any other members of P (Fig. 3). The Delaunay triangulation and the Voronoi diagram for the same set of points can be seen in Fig. 4. Note that, while the members of P are sites in the Delaunay triangulation, they are not sites in the Voronoi network, whose sites are the vertices of the polygons; also note that the edges of the Voronoi diagram lie along the perpendicular bisectors of the edges of the Delaunay triangulation—however, the edges of the Voronoi diagram do not always intersect the edges of the Delaunay triangulation, as seen in Fig. 4. The Delaunay triangulation represents the connectivity of the Voronoi *tessellation* of the surface.

There are many algorithms for constructing these networks. The fastest ones run in $O(n \log n)$ time for general distributions of points [3–6], where n is the number of generating sites, and this has been proven to be the optimal worst-case performance [4]. For a Poisson distribution of

points on the plane, there are many $O(n)$ expected-time algorithms [7–10].

In addition to being theoretically interesting [11–15], both the Voronoi diagram and the Delaunay triangulation are widely used in modeling and analyzing physical systems. They have seen use in lattice field theory and gauge theories [16], analyzing molecular dynamics of glassy liquids [17], detecting galaxy clusters [18], modeling the atomic structure and folding of proteins [19,20], modeling plant ecosystems and plant epidemiology [21], solving wireless signal routing problems [22,23], assisting with peer-to-peer network construction [24], the finite-element method of solving differential equations [25], game theory [26,27], modeling fragmentation [28], and numerous other areas [29,30].

Percolation theory is used to describe a wide variety of natural phenomena [31,32]. In the nearly 70 years since the first papers on percolation appeared [33,34], it has become a paradigmatic example of a continuous phase transition. For a given network, finding the critical probability p_c at which the percolation transition occurs is a problem of particular interest. p_c has been found analytically for certain two-dimensional networks [35–37]; however, most networks remain analytically intractable. Numerical methods have been used to find p_c for many such networks, e.g., [31,38–43].

In this paper we consider the percolation thresholds of the Voronoi and the Delaunay networks for a Poisson distribution of generating points, as represented in Figs. 1, 2, and 4. There are four percolation thresholds related to these two networks: the site and the bond percolation on each. Being a fully triangulated network, the site percolation threshold of the Delaunay network is exactly $p_c^{\text{site,Del}} = \frac{1}{2}$ [35,44–46]. This result has recently been proven rigorously by Bollobás and Riordan [47]. Somewhat surprisingly, a search of the literature revealed no prior calculation of the site percolation threshold of the Voronoi network at all, despite the widespread use of such networks. A prediction for its value has recently been made by Neher *et al.* [48]; they use an empirical formula to predict $p_c^{\text{site,Vor}} = 0.7151$, but they too were unable to find any previous calculation of this value, either

*beckeram@umich.edu

†rziff@umich.edu

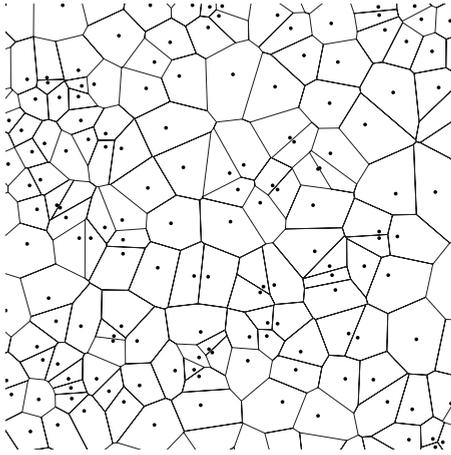


FIG. 1. Voronoi diagram with a Poisson distribution of generating points.

analytically or numerically. [There are places in the literature (e.g., [29]) where the Voronoi “site” threshold is listed as $1/2$. This is true if the sites are taken to be the generating points from which the diagram is created, rather than the vertices of the diagram. Thus, this is actually the Voronoi tiling threshold, i.e. the percolation threshold of the Voronoi *polygons*, which is in turn equivalent to the Delaunay site threshold, well known to be $1/2$.]

The bond thresholds of the Voronoi and Delaunay networks are complementary,

$$p_c^{bond,Vor} = 1 - p_c^{bond,Del}, \quad (1)$$

because these networks are dual to one another [49]. The first numerical measurement of the bond threshold for either network seems to be that of Jerauld *et al.* [50], who in 1984 found $p_c^{bond,Del} = 0.332$. Shortly thereafter, Yuge and Hori [51] performed a renormalization-group calculation which yielded $p_c^{bond,Del} = 0.3229$. In 1999, Hsu and Huang [52] found $p_c^{bond,Del} = 0.3333(1)$ and $p_c^{bond,Vor} = 0.6670(1)$ through Monte Carlo methods. (The numbers in parentheses repre-

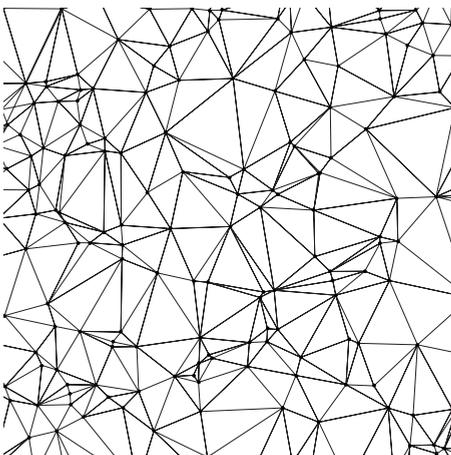


FIG. 2. Delaunay triangulation for the same set of generating points as in Fig. 1. The generating points become the vertices in this network.

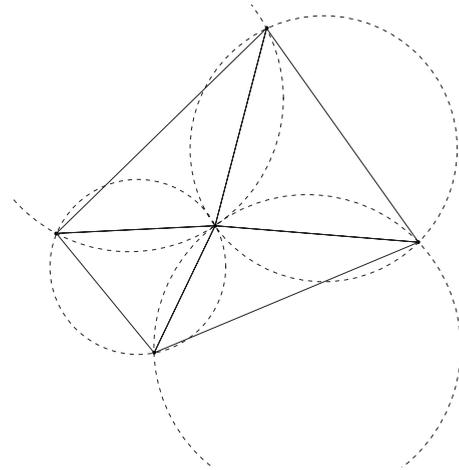


FIG. 3. The Delaunay triangulation for a set of five points, along with the associated circumcircles.

sent the errors in the last digits.) These values led them to make the intriguing conjecture that the thresholds are exactly $1/3$ and $2/3$, respectively. There is, however, no known theoretical reason to believe that this conjecture is true. In order to test this conjecture, and to find the site percolation threshold of the Voronoi network, we have carried out extensive numerical simulations, as detailed below. In Sec. II, we describe our methods, and in Sec. III we discuss our results and compare them to the thresholds of several related lattices and also discuss the covering graph and further generalizations of the Voronoi system. Conclusions are given in Sec. IV.

II. GENERATING ALGORITHMS AND ANALYSIS TECHNIQUES

A. Delaunay-Voronoi generation algorithm

In order to avoid edge effects in the networks when growing percolation clusters, and to make it possible to use more sites as seeds for those clusters (see Sec. II B), we wished to create Voronoi and Delaunay networks with periodic bound-

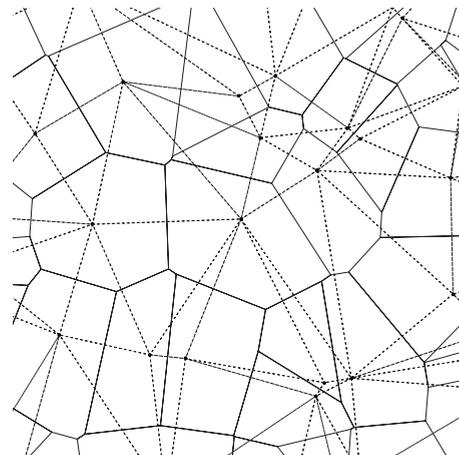


FIG. 4. The Delaunay triangulation (dotted lines) superposed on the Voronoi diagram (solid lines), its dual graph, for a set of Poisson-distributed generating points.

ary conditions. While popular fast algorithms for generating Voronoi and Delaunay networks exist, most notably the Quickhull algorithm [53], these do not generally support periodic boundary conditions. We therefore created our own fairly straightforward algorithm for generating the desired networks. After coming up with it independently, we later found that it falls into the class of expected-linear-time algorithms known as Incremental Search [8]. The basic outline of the algorithm is as follows:

(1) Divide the region in which the generating points (vertices of the Delaunay triangulation) are located into squares of equal size (“bins”).

(2) Find a single Delaunay edge by picking a point at random and searching through its bin and neighboring bins to find the point which is its nearest neighbor.

(3) Given a Delaunay edge and a “side” to look on (immediately above or below the edge), determine the third point in the Delaunay triangle by looking at the radii of the circumcircles of the triangles formed by that edge with each point in its bin and all of the neighboring bins.

(4) Look at the other Delaunay triangles that have been found in that bin and neighboring bins to make sure this new triangle is not a duplicate of one that has already been found. If it is not, find which of its neighbors have already been discovered and mark them as its neighbors, and vice versa.

(5) From the list of neighbors of the current triangle, figure out which of its edges are not already shared with neighbors (if any), and—if there are any unshared edges—whether the missing neighbor should be above or below the edge.

(6) Repeat steps (3)–(5) until there are no unprocessed edges left, at which point the Delaunay triangulation is finished. Because the neighbors of each triangle are known, this algorithm also yields an adjacency list of the sites on the Voronoi diagram (because the Voronoi diagram is dual to the Delaunay triangulation).

This algorithm is significantly easier to implement with periodic boundary conditions, because every triangle is guaranteed to have exactly three neighbors. Furthermore, the imposition of periodic boundary conditions also gives the Delaunay and Voronoi networks a very useful property: there are always exactly twice as many Delaunay triangles (Voronoi sites) as there are generating points (vertices of the Delaunay network or polygons in the Voronoi network) for a given diagram. This is a consequence of the more general fact that the number of faces (triangles) must be double the number of vertices (sites) for any fully triangulated network with doubly periodic boundary conditions in two dimensions. This fact follows from Euler’s formula and is proven in the Appendix. This simple relation makes it easier to spot certain kinds of errors in the code, because improperly written code is rather unlikely to consistently produce the proper number of sites for the given number of generating points. Using this algorithm, we generated thousands of Voronoi diagrams of 40 000 sites each. Figure 5 shows an example of a smaller Delaunay triangulation created with this algorithm.

B. Percolation cluster growth and finding p_c

The Leath-type epidemic growth method [54] that we used involves growing a large number of percolation clusters

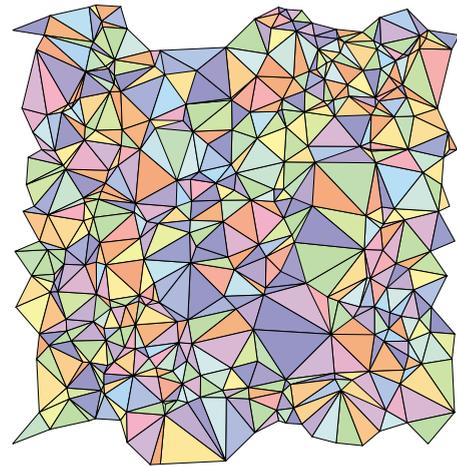


FIG. 5. (Color online) A Delaunay triangulation with periodic boundary conditions (i.e., on a torus), created from $n=300$ generating points. Because the surface has periodic boundary conditions, there are exactly $2n=600$ triangles here. Note the corresponding shapes of the outline on opposite edges, because this diagram has been “unrolled” from a torus.

in order to find p_c . For site percolation clusters, we start with a seed site somewhere on the network. Each of its neighbors is turned on with probability p or off with probability $1-p$. Neighbors of active sites are then visited and the procedure is repeated for all their previously unvisited neighbors; the cluster either dies out naturally or is stopped by the program when it hits a cutoff size of 1000 sites. For bond percolation clusters, an analogous algorithm is used in which sites are simply never turned off, as it is the bonds between sites that are pertinent.

Due to the fact that our Voronoi diagrams are finite and are generated from a random Poisson distribution of points, each diagram yields a slightly different effective value of p_c ; therefore, we had to generate many diagrams. This could have been very computationally expensive, but the choice of periodic boundary conditions helped here as well. Because there are no edges to the diagrams, we were able to place the seed point for a cluster at *any* site on a diagram, rather than being limited to a small subset of sites near the center. This meant that we were able to use many widely separated seed points to grow clusters on each diagram, which reduced the impact of each seed point’s immediate neighborhood upon the value of p_c obtained for each diagram. This, in turn, dramatically reduced the number of distinct diagrams we needed to obtain a particular level of precision. Specifically, we grew 8×10^5 clusters of up to 1000 sites on each of 800 diagrams, for a total of 6.4×10^8 clusters grown at each value of p . We then repeated this process at each of various p near p_c to generate the plots in the next section. Finally, this process was done twice—once for site percolation and once for bond percolation, both on the Voronoi network.

Because the percolation clusters are cut off before they can become large enough to wrap around the network, the clusters effectively see the diagram as infinite in size. Thus, their size distribution can be used to obtain an unbiased estimate of P_s , the probability that a percolation cluster will grow to be at least size s (for $s \leq 1000$) on an infinite net-

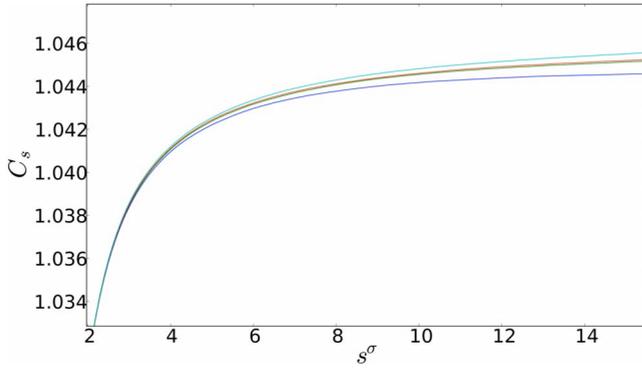


FIG. 6. (Color online) Epidemic site percolation cluster growth on the Voronoi diagram, for $p=0.714\ 07, 0.714\ 09, 0.714\ 11,$ and $0.714\ 13$, from bottom to top on the right.

work. At the critical threshold p_c , $P_s \sim s^{2-\tau}$ as $s \rightarrow \infty$, where $\tau=187/91$ for the two-dimensional percolation cluster universality class [31]. (It is expected that the critical exponents here are the same as for regular two-dimensional lattices.) In the scaling region, where s is large and $p-p_c$ is small such that $s^\sigma(p-p_c)$ is constant (with $\sigma=36/91$), P_s behaves as

$$P_s \sim A s^{2-\tau} f[B(p-p_c)s^\sigma], \quad (2)$$

where A and B are nonuniversal metric constants specific to the system being considered and $f(x)$ is a universal scaling function analytic about $x=0$. If we operate close to p_c such that $B(p-p_c)s^\sigma \ll 1$, then we can make a Taylor-series expansion of $f(x)$ to find

$$P_s \sim s^{2-\tau} [A + D(p-p_c)s^\sigma + \dots], \quad (3)$$

where D is another constant. Thus, plotting $C_s \equiv P_s s^{\tau-2}$ vs s^σ should yield a straight line at large s when p is near p_c , and that line will have a slope of zero when $p=p_c$. Figure 6 shows several such plots for site percolation clusters on the Voronoi network, and Fig. 7 shows several plots for bond percolation clusters on the same (see also Fig. 8). C_s does

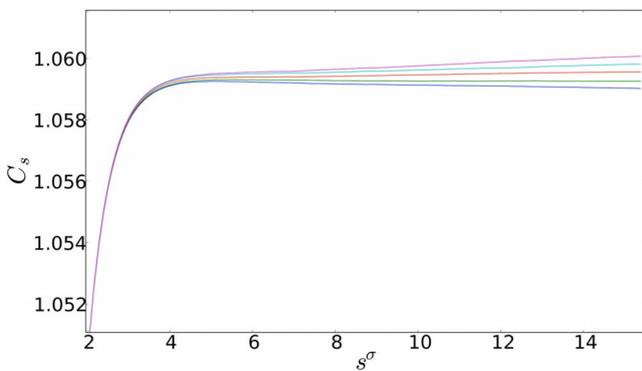


FIG. 7. (Color online) Epidemic bond percolation cluster growth on the Voronoi diagram, for $p=0.666\ 91, 0.666\ 93, 0.666\ 95, 0.666\ 97,$ and $0.666\ 99$, from bottom to top on the right. Note that these plots approach their linear asymptotes far more rapidly than those for site percolation clusters, as in Fig. 6; also note the difference in the vertical scale between the two figures.

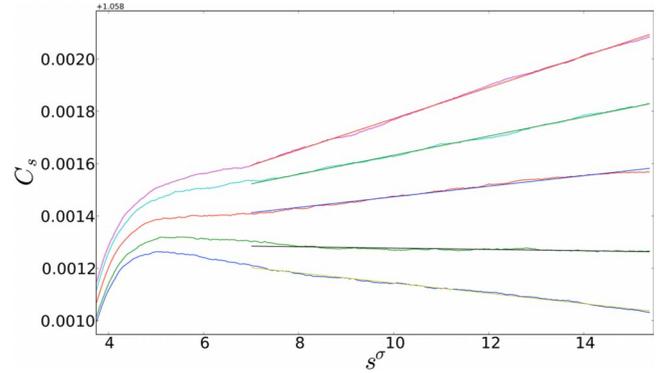


FIG. 8. (Color online) A zoomed-in portion of Fig. 7. The deviations from the horizontal in the asymptotes for each curve can be seen more clearly here. (The y-axis values are given from a reference of $C_s=1.058$). The least-squares linear fits for the curves are also on this plot.

indeed approach a linear function for large s in these plots, albeit far more quickly for bond percolation than for site percolation, with $p_c^{site,Vor} \approx 0.7141$ and $p_c^{bond,Vor} \approx 0.66693$.

Unfortunately, for smaller s there are deviations in C_s due to finite-size effects, and these are quite apparent for site percolation, even at the largest values of s that we were able to investigate. Exactly at p_c , one expects

$$P_s \sim s^{2-\tau} (A + E s^{-\Omega} + \dots) \quad (4)$$

as $s \rightarrow \infty$, where E is a constant and $\Omega \approx 0.6-0.8$ is the correction-to-scaling exponent [55]. Similar deviations should occur when p is close to p_c . In the case of site percolation on the Voronoi network, these finite-size effects make it difficult to determine when C_s has a truly horizontal asymptote; thus, it is not possible to use the above method to find $p_c^{site,Vor}$ to much greater precision than four digits when $s \leq 1000$.

The most straightforward way to solve this problem would be to grow larger site percolation clusters, using a larger system to insure that wrap around does not occur. However, because of the computational time that would be required to do that, we instead used a more sensitive method to find p_c that takes the finite-size corrections in Eq. (4) into account.

Equation (4) implies that, at p_c , $C_s - C_{s/2} = E(1 - 2^\Omega)s^{-\Omega}$ to leading order. This means that it is possible to estimate Ω directly from [55]

$$\Omega_s^{est} = -\log_2 \left(\frac{C_s - C_{s/2}}{C_{s/2} - C_{s/4}} \right). \quad (5)$$

Thus, in the regime where s is small enough that the finite-size effects of Eq. (4) matter, yet large enough that higher-order corrections are unimportant, Ω_s^{est} should approach a constant Ω when $p=p_c$. When $p \neq p_c$, there will be deviations due to scaling. Plots of Ω_s^{est} vs $\ln s$ for several values of p can be seen in Fig. 9; these yield the result for $p_c^{site,Vor}$ found in the following section.

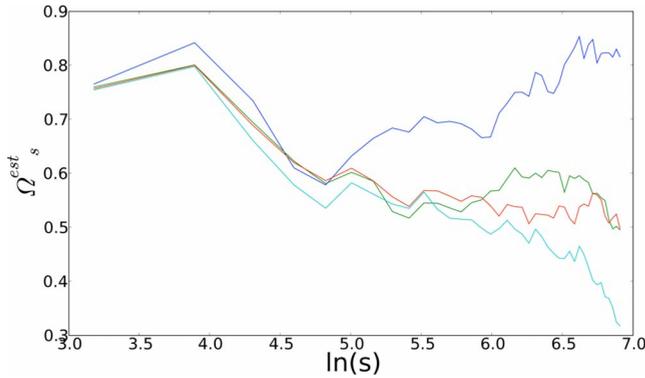


FIG. 9. (Color online) Ω_s^{est} at $p=0.71407, 0.71409, 0.71411,$ and 0.71413 (top to bottom on right) for site percolation on the Voronoi diagram.

III. RESULTS AND COMPARISON WITH RELATED LATTICES

A. p_c for site and bond percolation on the Voronoi diagram

Examining Fig. 9, it can be seen that Ω_s^{est} approaches a constant for large s for $p \approx 0.71409-0.71411$, and we conclude

$$p_c^{site,Vor} = 0.71410 \pm 0.00002, \tag{6}$$

where the error bars are meant to indicate one standard deviation of error. This plot also gives us a rough value of 0.65 for Ω —close to the value of Ω found for the Penrose rhomb quasilattice [55].

We used the method of plotting $C_s \equiv P_s s^{\tau-2}$ vs s^σ , outlined in the previous section, to find the bond percolation threshold of the Voronoi diagram. Taking the results shown in Figs. 7 and 8, we see immediately that $p_c^{bond,Vor} \approx 0.66693$. Because finite-size effects were not significant for bond percolation, we were able to find excellent least-squares linear fits to the asymptotic portions of the curves in Fig. 8. By plotting the slopes of these lines against the values of p used (see Fig. 10), we were able to solve for the value of

p that would yield a slope of zero; this should be p_c . This technique yielded a more accurate estimate,

$$p_c^{bond,Vor} = 0.666931 \pm 0.000005, \tag{7}$$

which by Eq. (1) implies $p_c^{bond,Del} = 0.333069 \pm 0.000005$. We considered various contributions to the stated error. First of all, it is unclear precisely where the linear regime begins in Fig. 8, and this leads to some uncertainty in the slopes we measured from the best-fit lines. Statistical effects of course are a source of error. However, a somewhat larger source of uncertainty turned out to be the error involved in reusing the same diagram multiple times during cluster growth—even with different seed points, there is a distinct likelihood that the same part of the nonuniform diagram will be sampled. To estimate this error, we considered our usual runs of 800 000 samples on ten different diagrams at $p=0.666931$ and looked at the variation in the curves of C_s vs s^σ (Fig. 11). In contrast, we also looked at ten runs of 800 000 samples each on the same diagram to gauge the purely statistical error. We found that the errors in the previous case are larger than in the latter. Using the measured standard deviation of 1.5×10^{-4} and dividing by $\sqrt{800}$ for the 800 runs we actually used in our simulations for each value of p , we estimate a final error of ± 0.000005 in the slopes of C_s , as indicated in the error bars of Fig. 10. Finally, because the slope of the fitted line in Fig. 10 [which equals the coefficient D in Eq. (3)] is nearly 1, we estimate that the final error in p_c is ± 0.000005 . Note that the runs for the five values of p were each done on 800 *different* diagrams, so there is no systematic error among the least-squares fit lines drawn in Fig. 8. Because of this, we believe that our error bars are conservative. The results for the thresholds are summarized in Table I and discussed further below.

B. Comparison with thresholds of related lattices

The Voronoi network has a uniform coordination number z equal to 3. In Table II, we compare the site and bond thresholds of the Voronoi diagram with several other lattices with $z=3$, listed in descending order of threshold values. In the Grünbaum-Shepard notation, $(3^{a_3}, 4^{a_4}, \dots)$ describes a

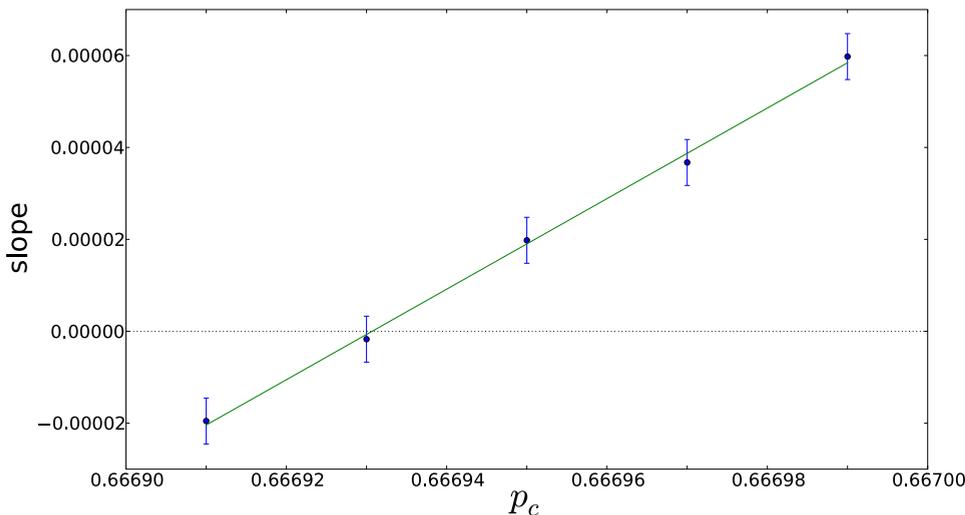


FIG. 10. (Color online) Slopes of the lines fitted in Fig. 8 versus the values of p used for each line, along with a best-fit line.

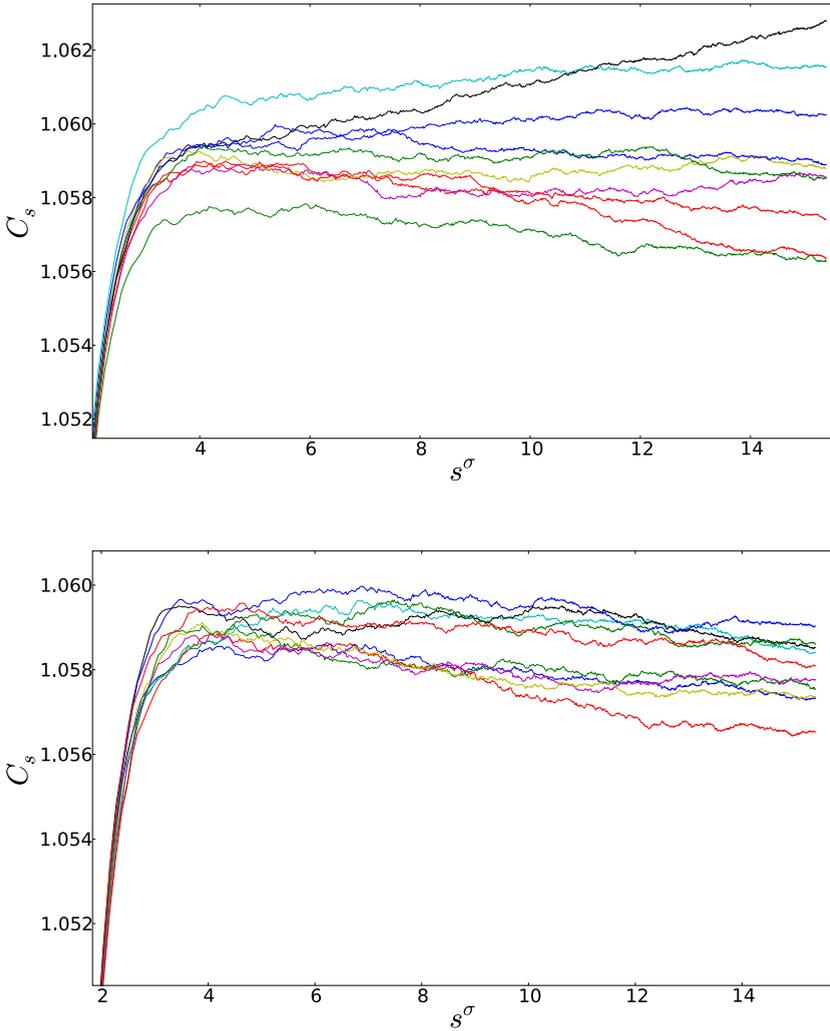


FIG. 11. (Color online) Comparison of multiple- and single-diagram bond percolation cluster growth on finite periodic Voronoi networks. Each curve represents 8×10^5 clusters grown on ten different diagrams (upper) or on ten identical diagrams (lower), at $p = 0.666\,931$. The mean values of the slopes are -1.8×10^{-5} (upper) and -1.1×10^{-4} (lower), and the standard deviations are 1.5×10^{-4} (upper) and 4.7×10^{-5} (lower).

lattice with a_3 triangles, a_4 quadrilaterals, etc., per vertex. The Archimedean lattices $(3, 12^2)$, $(4, 6, 12)$, and $(4, 8^2)$ are illustrated in [56–58]. The martini lattice was introduced in [59] and can be represented by $(3/4)(3, 9^2) + (1/4)(9^3)$.

We also list in Table II for each lattice the generalized filling factor f , defined as [56]

$$f = \pi \left[\sum_{n \geq 3} a_n \cot \frac{\pi}{n} \right]^{-1}, \quad (8)$$

which generalizes Scher and Zallen’s definition of f for lattices not necessarily composed of regular polygons [60]. f has been shown to provide a good correlation to site percolation thresholds for a variety of lattices. To calculate

TABLE I. Results for percolation thresholds of Voronoi and Delaunay networks. Numbers in parentheses represent errors in last digits.

Network	z	p_c^{site}	p_c^{bond}
Voronoi	3	0.71410(2)	0.666931(5)
Delaunay	6 (avg.)	0.5 (exact)	0.333069(2)

f for the Voronoi network, we use $b_3 = 0.011\,240\,0$, $b_4 = 0.106\,845\,4$, etc., from [61], where $b_n = 2a_n/n$ is the fraction of n -sided polygons in the system, satisfying $\sum_n b_n = 1$ and $\bar{n} = \sum_n n b_n = 6$ for $z = 3$.

Table II also lists the fluctuations in the number of the sides of the polygons for each lattice,

TABLE II. Thresholds of lattices with uniform coordination number $z = 3$, also showing the filling factor f and the polygon variance μ/\bar{n}^2 .

Lattice	μ/\bar{n}^2	f	p_c^{site}	p_c^{bond}
$(3, 12^2)$	0.5	0.39067	0.807901 ^{a,b}	0.740422 ^c
Martini	0.25	0.47493	0.764826 ^{a,d}	0.707107 ^{a,e}
$(4, 6, 12)$	0.222222	0.48601	0.747806 ^b	0.693734 ^d
$(4, 8^2)$	0.111111	0.53901	0.729724 ^b	0.676802 ^d
Voronoi	0.049468	0.57351	0.71410 ^f	0.666931 ^f
Honeycomb	0.0	0.60460	0.697040 ^{b,g}	0.652704 ^{a,h}

^aExact.

^bReference [56].

^cReference [64].

^dReference [59].

^eReference [65].

^fThis work.

^gReference [40].

^hReference [66].

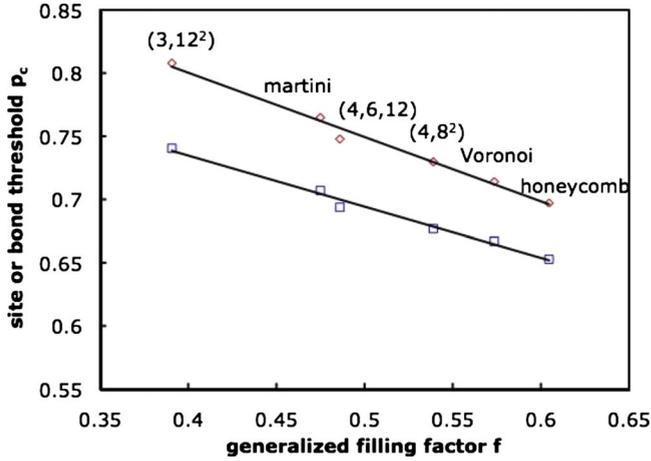


FIG. 12. (Color online) Thresholds vs generalized filling factor of Eq. (8) for site (top) and bond (bottom) percolation for the systems of Table II. The lines show least-squares fits to all of the data points.

$$\frac{\mu}{\bar{n}^2} \equiv \frac{\bar{n}^2 - \bar{n}^2}{\bar{n}^2}, \tag{9}$$

which is equal to the fluctuations in the coordination number of the dual lattice. It follows from Euler’s formula that the average number of sides of the polygons, $\bar{n} = \sum_n n b_n$, in any three-coordinated network is exactly six. For the Voronoi diagram, $\mu \approx 1.780\,811\,699\,0$ is known exactly as an integral [61–63].

In Fig. 12 we plot the thresholds given in Table II as a function of f . The thresholds fit well to a linear relation, as can be seen in the figure. In general, for bond percolation, f is not effective in correlating thresholds, which depend strongly on the coordination number z . However, for networks with fixed $z=3$, we find that the correlation of the bond thresholds with f is quite good.

We can fit the linear behavior of $p_c(f)$ using just data from exact results, with no numerical input. For site percolation, we use the exactly known thresholds for the $(3, 12^2)$ and martini lattices, while for bond percolation we use the martini and honeycomb lattice results, and find

$$\begin{aligned} p_c^{site} &= -0.5116f + 1.0078, \\ p_c^{bond} &= -0.4195f + 0.9063. \end{aligned} \tag{10}$$

These equations imply that, for the Voronoi diagram (where $f=0.573\,51$), $p_c^{site,Vor} = 0.7143$ and $p_c^{bond,Vor} = 0.6657$, which are evidently excellent estimates. Thus, the thresholds for the Voronoi diagram are consistent with other lattices with respect to the filling factor. A similar plot of thresholds versus the fluctuations also shows a consistent behavior between the Voronoi results and those for these other lattices.

Finally, the result for the bond threshold for the Voronoi network implies the site threshold for the Voronoi covering graph, shown in Fig. 13. The covering graph (or line graph) for a given network is defined as the graph that connects the centers of the bonds together and converts the bond percolation problem on that network to a site problem. Thus,

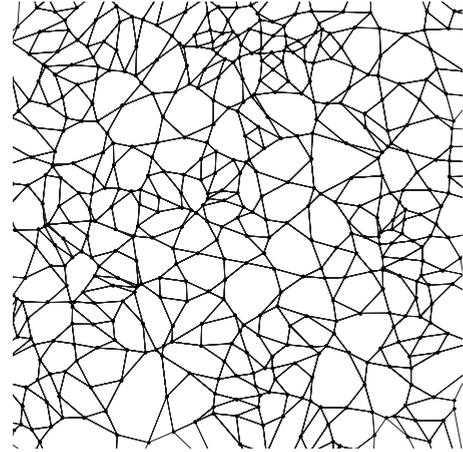


FIG. 13. Covering graph of a Voronoi network.

$p_c^{site,VorCov} = p_c^{bond,Vor} \approx 0.666\,931$. The covering graph is a kind of randomized kagomé diagram, consisting of triangles connected together. Using similar arguments given in [67] for generalized kagomé lattices, one can find an estimate for the bond threshold of the covering lattice, with the prediction $p_c^{bond,VorCov} \approx 0.536\,18$, as well as an estimate for the site-bond threshold for the Voronoi diagram. Details will be given elsewhere [68].

IV. CONCLUSIONS

We have determined the site percolation threshold for the Voronoi network to a high precision, with the result $p_c^{site,Vor} = 0.714\,10(2)$. We reiterate that this is not the well-known threshold ($1/2$) of the polygonal tiles of the Voronoi tessellation, which is equivalent to site percolation on the Delaunay triangulation, but rather the threshold for the three-coordinated diagram of all the Voronoi polygons. Our Monte Carlo result is very close to the prediction of 0.7151 of Neher *et al.* [48] and confirms their empirical procedure based on the Euler characteristic.

We also determined the bond threshold of the Voronoi network. Our result $p_c^{bond,Del} = 0.333\,069(5)$ is consistent with the result of Jerauld *et al.*, 0.332 [50], and close to Hsu and Huang’s value of 0.3333(1) [52], but runs counter to the latter authors’ conjecture that this threshold is exactly $1/3$. It is interesting to note that $1/3$ is the value predicted by the general (approximate) bond-threshold correlation $p_c \approx d / [(d-1)z]$ given by Vyssotsky *et al.* [69] for $z=6$ and dimension $d=2$.

We made comparisons of our results with thresholds of other lattices with the same coordination number ($z=3$) and found that the Voronoi thresholds are what one would expect based on correlations with the filling factor f for both the site and bond problems. Wierman has conjectured [70] that $2 \sin \pi/18 \approx 0.3473$, the bond threshold of the regular triangular lattice, is the maximum possible bond threshold for any fully triangulated network and that no other fully triangulated network has a bond threshold greater than or equal to that value. We indeed find that the bond threshold of the fully triangulated Delaunay network is consistent with this conjecture.

For future work, it would also be interesting to look at thresholds for other random systems, such as Johnson-Mehl tessellations [71] or the graph formed by the random distribution of lines in a plane [72]. Finding thresholds in Voronoi systems of higher dimensions is another interesting open problem.

ACKNOWLEDGMENT

This work was supported in part by the U.S. National Science Foundation Grant No. DMS-0553487.

APPENDIX: PROOF THAT $F=2V$ for any fully triangulated network with doubly periodic boundary conditions in two dimensions

We take advantage of the Euler relation for polyhedra to prove the desired fact about fully triangulated networks. A network on a square surface with doubly periodic boundary

conditions is topologically equivalent to placing the network on the surface of a torus; this network, in turn, can be seen as a polyhedron on the surface of the torus. Thus, the Euler relation for polyhedra applies,

$$V - E + F = \chi_{\text{torus}} = 0, \quad (\text{A1})$$

where V is the number of vertices on the polyhedron, E is the number of edges, F is the number of faces, and χ_{torus} is the Euler characteristic for the 2-torus, which is zero. Because every face has exactly three edges (i.e., the network is fully triangulated), and every edge is shared by exactly two faces (the network has no boundary), we have $E=3F/2$, and we can rewrite the Euler relation as follows:

$$V - \frac{3F}{2} + F = V - \frac{F}{2} = 0 \quad (\text{A2})$$

and thus $F=2V$.

Q.E.D.

-
- [1] G. Voronoi, *J. Reine Angew. Math.* **134**, 198 (1908).
 [2] B. Delaunay, *Bulletin de l'Académie des Sciences de l'URSS, Class des Sciences Mathématiques et Naturelles VII Serie* (1934), pp. 793–800.
 [3] S. Fortune, *Algorithmica* **2**, 153 (1987).
 [4] M. I. Shamos and D. Hoey, *16th Annual Symposium on Foundations of Computer Science, 1975* (IEEE, Piscataway, NJ, 1975), pp. 151–162.
 [5] L. J. Guibas, D. E. Knuth, and M. Sharir, *Algorithmica* **7**, 381 (1992).
 [6] D. T. Lee and B. J. Schachter, *Int. J. Parallel Programming* **9**, 219 (1980).
 [7] R. A. Dwyer, *Discrete Comput. Geom.* **6**, 343 (1991).
 [8] P. Su and S. Drysdale, *Proceedings of the 11th ACM Computational Geometry Conference, 1995* (ACM, New York, 1995), pp. 61–70.
 [9] A. Maus, *BIT* **24**, 151 (1984).
 [10] J. L. Bentley, B. W. Weide, and A. C. Yao, *ACM Trans. Math. Softw.* **6**, 563 (1980).
 [11] H. J. Hilhorst, *J. Stat. Mech.: Theory Exp.* (2005) P09005.
 [12] H. J. Hilhorst, *J. Stat. Mech.: Theory Exp.* (2009) P05007.
 [13] H. J. Hilhorst, *Eur. Phys. J. B* **64**, 437 (2008).
 [14] V. Lucarini, *J. Stat. Phys.* **130**, 1047 (2008).
 [15] M. M. de Oliveira, S. G. Alves, S. C. Ferreira, and R. Dickman, *Phys. Rev. E* **78**, 031133 (2008).
 [16] N. H. Christ, R. Friedberg, and T. D. Lee, *Nucl. Phys. B* **202**, 89 (1982).
 [17] F. W. Starr, S. Sastry, J. F. Douglas, and S. C. Glotzer, *Phys. Rev. Lett.* **89**, 125501 (2002).
 [18] M. Ramella, W. Boschin, D. Fadda, and M. Nonino, *Astron. Astrophys.* **368**, 776 (2001).
 [19] M. Gerstein, J. Tsai, and M. Levitt, *J. Mol. Biol.* **249**, 955 (1995).
 [20] A. Poupon, *Curr. Opin. Struct. Biol.* **14**, 233 (2004).
 [21] O. Deussen, P. Hanrahan, B. Lintermann, R. Měch, M. Pharr, and P. Prusinkiewicz, *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques* (ACM, New York, 1998), pp. 275–286.
 [22] S. Meguerdichian, F. Koushanfar, M. Potkonjak, and M. Srivastava, in *IEEE INFOCOM Conference, 2001* (IEEE, Piscataway, NJ, 2001), Vol. 3, pp. 1380–1387.
 [23] S. Bandyopadhyay and E. Coyle, in *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies, 2003* (IEEE, Piscataway, NJ, 2003), Vol. 3.
 [24] M. Naor and U. Wieder, *ACM Trans. Algorithms* **3**, 34 (2007).
 [25] O. C. Zienkiewicz, R. L. Taylor, and J. Z. Zhu, *The Finite-Element Method: Its Basis and Fundamentals*, 6th edition (Elsevier Butterworth-Heinemann, Oxford, 2005).
 [26] H. K. Ahn, S. W. Cheng, O. Cheong, M. Golin, and R. van Oostrum, *Theor. Comput. Sci.* **310**, 457 (2004).
 [27] O. Cheong, S. Har-Peled, N. Linial, and J. Matousek, *Discrete Comput. Geom.* **31**, 125 (2004).
 [28] T. Kiang, *Z. Astrophys.* **64**, 433 (1966).
 [29] A. Okabe, B. Boots, K. Sugihara, and S. N. Chiu, *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*, 2nd ed. (Wiley, New York, 2000).
 [30] F. Aurenhammer, *ACM Comput. Surv.* **23**, 345 (1991).
 [31] D. Stauffer and A. Aharony, *Introduction to Percolation Theory*, 2nd ed. (CRC Press, Boca Raton, FL, 1994).
 [32] M. Sahimi, *Applications of Percolation Theory* (Taylor & Francis, London, 1994).
 [33] P. J. Flory, *J. Am. Chem. Soc.* **63**, 3083 (1941).
 [34] S. R. Broadbent and J. M. Hammersley, *Proc. Cambridge Philos. Soc.* **53**, 629 (1957).
 [35] M. F. Sykes and J. W. Essam, *J. Math. Phys.* **5**, 1117 (1964).
 [36] J. C. Wierman, *J. Phys. A* **17**, 1525 (1984).
 [37] R. M. Ziff and C. R. Scullard, *J. Phys. A* **39**, 15083 (2006).
 [38] R. M. Ziff and P. N. Suding, *J. Phys. A* **30**, 5351 (1997).
 [39] C. D. Lorenz and R. M. Ziff, *J. Chem. Phys.* **114**, 3659 (2001).
 [40] X. Feng, Y. Deng, and H. W. J. Blöte, *Phys. Rev. E* **78**, 031136 (2008).
 [41] S. Quintanilla, S. Torquato, and R. M. Ziff, *J. Phys. A* **33**,

- L399 (2000).
- [42] H. G. Ballesteros, L. A. Fernandez, V. Martín-Mayor, A. Muñoz Sudupe, G. Parisi, and J. J. Ruiz-Lorenzo, *J. Phys. A* **32**, 1 (1999).
- [43] P. Grassberger, *Phys. Rev. E* **67**, 036101 (2003).
- [44] H. Kesten, *Percolation Theory for Mathematicians* (Birkhäuser, Boston, 1982).
- [45] M. V. Menshikov, *Sov. Math. Dokl.* **33**, 856 (1986).
- [46] M. Aizenman and D. J. Barsky, *Commun. Math. Phys.* **108**, 489 (1987).
- [47] B. Bollobás and O. Riordan, *Probab. Theory Relat. Fields* **136**, 417 (2006).
- [48] R. A. Neher, K. Mecke, and H. Wagner, *J. Stat. Mech.: Theory Exp.* (2008) P01011.
- [49] B. Bollobás and O. Riordan, *Random Struct. Algorithms* **32**, 463 (2008).
- [50] G. R. Jerauld, J. C. Hatfield, L. E. Scriven, and H. T. Davis, *J. Phys. C* **17**, 1519 (1984).
- [51] Y. Yuge and M. Hori, *J. Phys. A* **19**, 3665 (1986).
- [52] H.-P. Hsu and M.-C. Huang, *Phys. Rev. E* **60**, 6361 (1999).
- [53] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, *ACM Trans. Math. Softw.* **22**, 469 (1996).
- [54] P. L. Leath, *Phys. Rev. B* **14**, 5046 (1976).
- [55] R. M. Ziff and F. Babalievski, *Physica A* **269**, 201 (1999).
- [56] P. N. Suding and R. M. Ziff, *Phys. Rev. E* **60**, 275 (1999).
- [57] B. Grünbaum and G. C. Shephard, *Tilings and Patterns* (Freeman, New York, 1987).
- [58] http://en.wikipedia.org/wiki/Percolation_threshold
- [59] C. R. Scullard, *Phys. Rev. E* **73**, 016107 (2006).
- [60] H. Scher and R. Zallen, *J. Chem. Phys.* **53**, 3759 (1970).
- [61] H. J. Hilhorst, *J. Phys. A* **40**, 2615 (2007).
- [62] K. A. Brakke, *Statistics of Random Plane Voronoi Tessellations* (unpublished), <http://www.susqu.edu/brakke/aux/downloads/papers/vorplane.pdf>
- [63] S. R. Finch, *Mathematical Constants* (Cambridge University Press, Cambridge, England, 2003); <http://algo.inria.fr/csolve/vi.pdf>
- [64] R. Parviainen, *J. Phys. A* **40**, 9253 (2007).
- [65] R. M. Ziff, *Phys. Rev. E* **73**, 016134 (2006).
- [66] M. F. Sykes and J. W. Essam, *Phys. Rev. Lett.* **10**, 3 (1963).
- [67] R. M. Ziff and H. Gu, *Phys. Rev. E* **79**, 020102(R) (2009).
- [68] R. M. Ziff, A. M. Becker, and H. Gu (unpublished).
- [69] V. A. Vyssotsky, S. B. Gordon, H. L. Frisch, and J. M. Hammersley, *Phys. Rev.* **123**, 1566 (1961).
- [70] J. C. Wierman, *J. Phys. A* **35**, 959 (2002).
- [71] B. Bollobás and O. Riordan, *Probab. Theory Relat. Fields* **140**, 319 (2008).
- [72] S. Goudsmit, *Rev. Mod. Phys.* **17**, 321 (1945).