Percolation thresholds on high-dimensional math xmlns="http://www.w3.org/1998/Math/MathML">msub>mi >D/mi>mi>n/mi>/msub>/math> and math xmlns="http://www.w3.org/1998/Math/MathML">msub>mi >E/mi>mn>8/mn>/msub>/math>-related lattices

Yi Hu and Patrick Charbonneau

Phys. Rev. E **103**, 062115 — Published 11 June 2021

DOI: 10.1103/PhysRevE.103.062115

Percolation thresholds on high-dimensional D_n and E_8-related lattices

Yi Hu[1] and Patrick Charbonneau[1,2]

[1] *Department of Chemistry, Duke University, Durham, North Carolina 27708, USA*
[2] *Department of Physics, Duke University, Durham, North Carolina 27708, USA*

(Dated: May 27, 2021)

The site and bond percolation problems are conventionally studied on (hyper)cubic lattices, which afford straightforward numerical treatments. The recent implementation of efficient simulation algorithms for high-dimensional systems now also facilitates the study of D_n root lattices in n dimension as well as E_8-related lattices. Here, we consider the percolation problem on D_n for $n = 3$ to 13 and on E_8 relatives for $n = 6$ to 9. Precise estimates for both site and bond percolation thresholds obtained from invasion percolation simulations are compared with dimensional series expansion based on lattice animal enumeration for D_n lattices. As expected, the bond percolation threshold rapidly approaches the Bethe lattice limit as n increases for these high-connectivity lattices. Corrections, however, exhibit clear yet unexplained trends. Interestingly, the finite-size scaling exponent for invasion percolation is found to be lattice and percolation-type specific.

I. INTRODUCTION

Percolation being one of the simplest critical phenomena, its models play a particularly important role in statistical physics [1]. Minimal models—lattice-based ones, in particular—have indeed long been used to test notions of universality as well as the applicability of mean-field and renormalization group predictions to physical systems. On lattices, two covering fractions p can be defined: (i) the probability that a vertex is occupied, and (ii) the probability that an edge between nearest-neighbor vertices is occupied. As p increases, a percolating cluster forms at a threshold p_c^{site} or p_c^{bond}, depending on the covering choice [1]. These thresholds are therefore lattice specific. Because precise thresholds values are prerequisite for stringently assessing criticality [2–5] yet lack analytical expressions [6], substantial efforts have been directed at estimating them through numerical simulations [2, 7–10] and graph-based polynomial methods [11–14]. The strong dependence of criticality on spatial dimension n further motivates expanding these efforts over an extended range of n [1, 15].

In this context, the invasion percolation algorithm recently introduced by Mertens and Moore to determine lattice percolation thresholds [2, 16] is particularly interesting. In short, the algorithm directly grows a percolating cluster, and thus provides both the universal asymptotic critical behavior and the lattice-specific finite-size scaling correction. Most crucially, by avoiding the explicit construction of a lattice grid, the scheme preserves a polynomial space complexity as n increases. Threshold values with ten significant digits of precision have thus been obtained on hypercubic lattices ($\mathbb{Z}^n$) up to $n = 13$ [2].

Hypercubic lattices, although geometrically straightforward, are in some ways not natural systems to study as dimension increases. In order to better visualize this effect, recall that lattices can be seen as discretizations of Euclidean space $\mathbb{R}^n$, in which each lattice site is centered in a cell in that tessellation. As n increases, the cubic cells that tile $\mathbb{Z}^n$ become increasingly dominated by the spikiness of their corners. By contrast, the cells of root lattices, D_n (for $n \geq 3$) have smoother features. In $n = 3$, for instance, this construction gives rise to the face-centered cubic lattice ($D_3 \equiv$ fcc), whose rhombic dodecahedron cells are much closer to spheres than cubes are. A way to quantify this effect is to compare the maximal sphere packing fraction of different lattices (with a sphere centered on every lattice site). In this measure, D_n packings are $2^{n/2-1}$ times denser than their $\mathbb{Z}^n$ counterparts [17]. Similarly, the eight-dimensional E_8 lattice corresponds to a sphere packing fraction twice that of D_8 (and 16 times that of $\mathbb{Z}^8$); E_8-related lattices, E_6, E_7 and Λ_9 are also the densest known sphere packings in their corresponding dimension. This advantage has motivated the recent consideration of D_n and E_8-related periodic boundary conditions for high-dimensional numerical simulations [18–20]. For a same computational cost, these periodic boxes indeed have a larger inscribed radius than hypercubes and thus present less pronounced finite-size corrections. Because these lattices provide a more compact and symmetric tessellation of the space, they may similarly help suppress obfuscating pre-asymptotic corrections to percolation criticality [2, 4], which are especially challenging to handle around the upper critical dimension, $n_u = 6$. Yet percolation threshold values, which are prerequisite for any criticality study, have not been previously reported for these lattices.

In this work, we investigate the two canonical lattice percolation thresholds on D_n lattice for $n = 3$ to 13 as well as on E_8-related lattices in $n = 6 \sim 9$. We first describe the high-dimensional lattices considered in Section II. In Section III, we derive the series expansion for both p_c^{site} and p_c^{bond} on D_n lattices based on lattice animal enumeration. We then describe the invasion percolation algorithm in Section IV, and analyze the numerical threshold results in Section V. We briefly conclude in Section VI.

II. LATTICE CONSTRUCTION

In this section, we review the construction of the high-dimensional D_n and E_8-related lattices following the description in Ref. 21. Construction algorithms for these lattices were developed in 1980s in the context of signal processing in information theory [17, 21], and have recently found use in high-dimensional molecular simulations [18–20].

Before describing these lattices, recall that the conventional n-dimensional hypercubic lattices, $\mathbb{Z}^n$, are defined as a set of n-dimensional vectors of integer components. The nearest-neighbor vectors $(a_1, a_2, ..., a_n)$ in $\mathbb{Z}^n$ are $(\pm 1, 0^{n-1})$. Note that for notational convenience, $(\pm 1, 0, ...)$ denotes a group of vectors with all index permutations, and 0^m denotes m subsequent 0s as vector components. The number of nearest neighbors (or kissing number) is thus $2n$. D_n (or checkboard) lattices can be viewed as the subset of $\mathbb{Z}^n$ for which the sum of coordinates is even. The nearest-neighbor vectors are then $(\pm 1^2, 0^{n-2})$, thus resulting in each vertex having $2n(n-1)$ nearest neighbors in total. In $n = 3$, for example, the 12 nearest-neighbor vectors for the D_3 (fcc) lattice read

$$\begin{array}{llll} (1,1,0), & (1,-1,0), & (-1,1,0), & (-1,-1,0), \\ (1,0,1), & (1,0,-1), & (-1,0,1), & (-1,0,-1), \\ (0,1,1), & (0,1,-1), & (0,-1,1), & (0,-1,-1). \end{array}$$

D_3, D_4 and D_5 lattices are the densest sphere packings in their respective dimensions. The densest sphere packings for $n = 6$ to 9 are E_6, E_7, E_8 and Λ_9 lattices, respectively. In particular, the E_8 lattice consists of two D_8 lattice points with offset $(\frac{1}{2}^8)$. The nearest-neighbor vectors of E_8 can then be viewed as four groups,

$$\begin{cases} \pm(\frac{1}{2}^8), & 2 \text{ vectors}, \\ (\frac{1}{2}^4, -\frac{1}{2}^4), & 70 \text{ vectors}, \\ \pm(\frac{1}{2}^2, -\frac{1}{2}^6), & 56 \text{ vectors}, \\ (\pm 1^2, 0^6), & 112 \text{ vectors}, \end{cases} \tag{1}$$

and thus each vertex has 240 nearest neighbors in total.

The E_7 lattice is a cross-section of E_8 in $n = 7$. One of the choices to generate nearest-neighbor vectors in E_7 is the subset of Eq. (1) with zero sum, which corresponds to a seven-dimensional hyperplane orthogonal to (1^8). Each vertex then has 126 nearest-neighbor vectors,

$$\begin{cases} (\frac{1}{2}^4, -\frac{1}{2}^4), & 70 \text{ vectors}, \\ (1, -1, 0^6), & 56 \text{ vectors}. \end{cases} \tag{2}$$

The E_6 lattice is a cross-section of E_7 in $n = 6$ that is orthogonal to $(1; 0^6; 1)$ (with first and the last components fixed). The E_6 nearest-neighbor vectors are constructed by further constraining $a_1 + a_8 = 0$ from E_7, namely,

$$\begin{cases} \pm(\frac{1}{2}; \frac{1}{2}^3, -\frac{1}{2}^3; -\frac{1}{2}), & 40 \text{ vectors}, \\ (0; 1, -1, 0^4; 0), & 30 \text{ vectors}, \\ \pm(1; 0^6; -1), & 2 \text{ vectors}, \end{cases} \tag{3}$$

thus giving 72 such vectors.

Infinite numbers of equally dense packing exist in $n = 9$ [22], but one of its forms, Λ_9, can be constructed similarly to E_8. This construction consists of two D_9 lattices, offset by $(\frac{1}{2}, ..., \frac{1}{2}, 0)$, which results in 272 nearest-neighbor vectors,

$$\begin{cases} \pm(\frac{1}{2}^8; 0), & 2 \text{ vectors}, \\ (\frac{1}{2}^4, -\frac{1}{2}^4; 0), & 70 \text{ vectors}, \\ \pm(\frac{1}{2}^2, -\frac{1}{2}^6; 0), & 56 \text{ vectors}, \\ (\pm 1^2, 0^7), & 144 \text{ vectors}. \end{cases} \tag{4}$$

Equations (1)-(4), after rescaling by a factor two, generate integer vectors that can be easily implemented using integer arithmetics.

Because for $n = 10$ the (presumed) densest sphere packing is a nonlattice [22], $n = 9$ offers a natural end to our consideration of dense sphere packings. We should note, however, that even for $d \leq 9$ other uniform packings that are equivalently dense to lattice packings can be obtained. For example, in $n = 3$ the face-centered cubic lattice (D_3) is intimately related to the hexagonal closed-packed (hcp) structure. As a result, the two have site percolation thresholds that are numerically close yet not identical [23]. Similarly, in $n = 5, 6, 7$ four (and in $n = 9$ a continuum of) related uniform packings can be constructed [22]. Their p_c are expected to differ slightly from those of the related simple lattices, but their construction is not considered here. Note also that percolation on D_n and E_8-related lattices is related to that on hypercubic lattices with extended-nearest-neighbor connectivity [5, 9, 10]. Some of these constructions are even equivalent to the lattices considered here (see Table II).

III. SERIES EXPANSION

In this section we derive high-dimensional series expansions for both site and bond percolation thresholds on D_n lattices by counting lattice animals embedded on these lattices [24, 25]. Of all possible approaches for deriving such series, this one has thus far achieved the most extended expansion for hypercubic lattices [25], which motivates us to consider it here. The method, however, involves heuristic assumption on lattice animal polynomials and is thus not deemed rigorous. (The first three terms of the site percolation series for hypercubic lattices have been formally validated through a different scheme [26].)

For site percolation a *site animal* of size v is a cluster of v lattice vertices connected after linking all neighboring vertex pairs. Similarly, for bond percolation a *bond animal* of size e consists of a connected set of e lattice edges. In both cases, the perimeter t is the number of incident vertices (or edges) for the lattice animal. Because for a given v (or e), lattice animals with different t exist, we index them as $t_{v,i}$ (or $t_{e,i}$). We further denote the number of distinct (not related by translation) site and

bond animals of perimeter t on a n-dimensional lattice as $g_{v,i}(n)$ (or $g_{e,i}(n)$).

A. Site percolation

We first consider the site percolation threshold. Following Mertens *et al.* [25] we define the polynomial (with implicit n dependence)

$$A_v(q) = \sum_{\{t_{v,i}\}} g_{v,i} q^{t_{v,i}}, \tag{5}$$

in terms of $q = 1 - p$. In particular, $A_v(1) \equiv A_v$ gives the total number of lattice animals of size v in an n-dimensional lattice. At covering fraction p, the expected site cluster size on the lattice is

$$S = \sum_v v^2 p^{v-1} A_v(1-p) \equiv \sum_{\ell=0}^{\infty} b_\ell(n) p^\ell, \tag{6}$$

where we have expanded S as a power series in p. Because $A_v(q)$ is associated with a factor of p^{v-1}, obtaining b_ℓ only requires $A_1, ..., A_{\ell+1}$, i.e., counting $g_{1,i}$ to $g_{\ell+1,i}$. Once these terms are known, p_c^{site} can be approximated by re-summing the terms using a $(\ell-1, 1)$ Padé approximant $b_{\ell-1}/b_\ell$ (see Ref. 25).

In order to obtain a series expansion, the objective is to count $g_{v,i}$ in different dimensions, and express $g_{v,i}(n)$ as a polynomial in n. On hypercubic lattices the computational cost of this enumeration is greatly simplified by introducing *proper dimension* to account for the number of dimensions spanned by a lattice animal [24, 25], but this approach is not obviously generalizable for D_n lattices. We instead implement a more generic, brute-force algorithm [27], which traverses every possible lattice animal via a breadth first search (BFS) of the lattice vertices. For notational convenience, we also define the lexicographical relation between two points. In particular, $\mathbf{x} > \mathbf{y}$ if the first nonzero coordinate elements in $\mathbf{x} - \mathbf{y}$ is greater than zero.

Starting at the origin, we add every nearest-neighbor site (as described in Section II) to the perimeter set. In that set, we then choose one site $\mathbf{x}$ and add it to the site animal set according to the following criteria:

1. if $\mathbf{x}$ is lexicographically greater than the origin, (0^n);

2. if $\mathbf{x}$ was newly added to the perimeter set at the previous iteration, or if $\mathbf{x}$ is lexicographically greater than all sites in the site animal set.

The first constraint ensures that the origin is the site with the lexicographically smallest coordinates in the cluster. The second constraint ensures that a cluster is generated site-by-site with definitive order. These two conditions guarantee that a site animal—after properly accounting for translational invariance—is counted exactly once.

Once a new site is added, the perimeter set is updated with the nearest neighbors of this site. New sites are then iteratively selected until the pre-assigned size v is reached. Therefore, by running the algorithm once with assigned v and n, a series of integer values of $(t_{v,i}(n), g_{v,i}(n))$ can be obtained. For example, numerical enumeration results for $v = 3$ site animals in $n = 2$ to 5 are reported in Table I.

TABLE I. Numerical results for site animal enumeration at $v = 3$

n	2	3			4			5		
$\{t_{3,i}\}$	7	8	22	23	24	48	51	52	86	91	92
$g_{3,i}$	4	2	8	12	30	32	72	108	80	280	260

The next step of the series construction entails obtaining the analytical polynomial forms of all of the t and g polynomials from their numerical values. Let's first consider $t_{v,i}(n)$. Because every site of a D_n lattice has $z = 2n(n-1)$ neighbors, the perimeter for a cluster of size v cannot exceed zv, and hence $t_{v,i}(n) \leq 2n(n-1)v$. However, this upper bound double counts certain sites, namely, those multiply-shared as neighbors and that are part of the cluster. For two neighboring sites, the number of shared neighbors is of $\mathcal{O}(n)$, hence the doubly-counted sites are of $\mathcal{O}(n)$ per cluster site. As a result, $t_{v,i}(n) = 2vn(n-1) - \mathcal{O}(vn)$, which is quadratic in n. Therefore, site animals in different dimensions can be related by a linear fit of t,

$$t_{v,i}(n) = 2n(n-1)v + c_1^{(t)} n + c_0^{(t)}, \tag{7}$$

Although results from only two different dimensions are required to determine the coefficients $c_0^{(t)}$ and $c_1^{(t)}$, fitting results for a larger number of dimensions leaves no residual, which furthers our confidence in this heuristic scheme.

The polynomial $g_{v,i}(n)$ can also be obtained by solving a linear system. The (upper bound of the) order of this polynomial must, however, be determined in advance. Because the total number of lattice animals is $\sim [2n(n-1)]^{v-1}$, the order of $g_{v,i}(n)$ is also at most $n^{2(v-1)}$. And because the orientational degeneracy under D_n symmetry requires that $g_{v,i}(n)$ always has roots $n(n-1)$, the order is further reduced to $n^{2(v-2)}$. Therefore, we require the numerical $g_{v,i}(n)$ results for at most $2v-3$ different dimensions, and solve the equation to obtain coefficients,

$$g_{v,i}(n)/[n(n-1)] = \sum_{k=0}^{2(v-2)} c_k^{(g)} n^k. \tag{8}$$

While the validity of this fitting form has yet to be mathematically demonstrated, the correctness of $g_{v,i}$ polynomials can be empirically tested by checking that the residual vanishes when fitting the results of a (larger-than-necessary) number of dimensions.

In order to illustrate this procedure, we take $v = 3$ as an example. Inserting the results for n = 3-5 from Table I into Eq. (7) gives three t polynomials,

$$\begin{cases} t_{3,1} &= 6n(n-1) - 10n + 16 \\ t_{3,2} &= 6n(n-1) - 8n + 11 \\ t_{3,3} &= 6n(n-1) - 8n + 12. \end{cases} \tag{9}$$

For each $t_{3,1}(n)$ we take the associated $g_{3,1}(n)$ and solve for Eq. (8). This also generates three g polynomials,

$$\begin{cases} g_{3,1}(n) &= \frac{4}{3}n(n-1)(n-2) \\ g_{3,2}(n) &= 2n(n-1)(n^2-5n+7) \\ g_{3,3}(n) &= n(n-1)(4n-7). \end{cases} \tag{10}$$

Note that $\{g_{3,1}(2), g_{3,2}(2), g_{3,3}(2)\} = \{0, 4, 2\}$, which is consistent with the numerical results. (In particular, in $n = 2$ only two distinct t is are reported in Table I, which is consistent with $g_{3,1}(2) = 0$.) In general, it is preferable to solve first for t polynomials, in order to index enumeration results in different dimension, and then solve for g polynomials, in order to obtain a functional mapping $t_{v,i}(n) \mapsto g_{v,i}(n)$. For example, in $v = 3$ we obtain three pairs of (t, g) polynomials in Eqs. (9) and (10).

Evaluating site animals up to $n = 15$ is then sufficient to solve Eq. (8) for $v \leq 6$. (Because the total number of site animals, $A_v \sim n^{2(v-1)}$, grows exponentially with v, results for $v > 6$ lie beyond current computational reach.) We thereby identify $12, 36, 83$ pairs of (t, g) polynomials for $v = 4, 5, 6$, respectively [28], and with $t_{v,i}$ and $g_{v,i}$ for $v \leq 6$, we obtained the first six terms in the expansion for S [Eq. (6)],

$$\begin{aligned} b_0 &= 1, \\ b_1 &= 2n(n-1), \\ b_2 &= 2n(n-1)(2n^2-6n+7), \\ b_3 &= 2n(n-1)(4n^4-24n^3+57n^2-53n+12), \\ b_4 &= 2n(n-1)(8n^6-72n^5+272n^4-552n^3+804n^2 \\ &\quad -1102n+857), \\ b_5 &= 2n(n-1)(16n^8-192n^7+1004n^6-3028n^5+ \\ &\quad 6018n^4-\frac{17710}{3}n^3-11851n^2+\frac{284075}{6}n-43202). \end{aligned} \tag{11}$$

For large ℓ, the Padé approximant $b_{\ell-1}/b_\ell$ converges to p_c [25]. However, it is not a priori known to what accuracy a finite-order approximant should agree with the actual (unknown) expansion form. What we do know is that the leading order of that expansion should agree with the Bethe lattice threshold for a branching tree of degree z [1],

$$p_{c,\text{Bethe}} = \frac{1}{z-1} \equiv \frac{1}{\sigma}, \tag{12}$$

where for D_n lattices $z = 2n(n-1)$. In the following we denote $1/\sigma$ the Bethe lattice limit of the percolation threshold.

We observe that the lowest-order approximant b_0/b_1 already agrees with $p_{c,\text{Bethe}}$ at leading order. We also observe that in general b_ℓ has a leading order of $n^{2\ell}$, and $b_{\ell-1}/b_\ell$ provides an approximation for p_c with an error that vanishes asymptotically as $\mathcal{O}(n^{-(\ell+2)})$. For comparison, $b_\ell \sim n^\ell$ for a hypercubic lattice and the approximant $b_{2\ell}/b_{2\ell+1}$ have the same order of error $\mathcal{O}(n^{-(\ell+2)})$ [25]. Expanding b_4/b_5, in particular, gives

$$p_c^{\text{site}} = \frac{1}{\sigma} + \frac{1}{n^3} + \frac{23}{8n^4} + \frac{17}{2n^5} + \frac{999}{32n^6} + \mathcal{O}(n^{-7}). \tag{13}$$

The numerical accuracy of this series is evaluated in Sec. V B.

B. Bond percolation

For the bond percolation, we similarly define the bond polynomial

$$A_e(q) = \sum_{\{t_{e,i}\}} g_{e,i} q^{t_{e,i}}, \tag{14}$$

which gives the expected bond cluster size

$$S = \sum_e e^2 p^{e-1} A_e(1-p) \equiv \sum_{\ell=0}^{\infty} b_\ell(n) p^\ell \tag{15}$$

as a polynomial in p. The enumeration scheme for bond animals is essentially the same as for site animals, with the exception that we now maintain bonds, which are indexed as the coordinates of the lexicographically smaller vertex on this bond, in addition to the orientation index—from 1 to $n(n-1)$—of the bond. The bond animal enumeration is then used to obtain a series of numerical values $(t_{e,i}(n), g_{e,i}(n))$. The perimeter polynomials $t_{e,i}(n)$ for bond animal are also quadratic with n, but the leading prefactor is not fixed. The t polynomial is thus obtained by fitting $t(n)$ in at least three dimensions. A bond animal of size e includes at most $e + 1$ sites, hence the order of $g_{e,i}(n)$ is at most n^{2e}, including roots $n(n-1)$. This leads to $2e-1$ different dimensions being required for solving the linear equation for $g_{e,i}(n)$, similar to Eq. (8),

$$g_{e,i}(n)/[n(n-1)] = \sum_{k=0}^{2(e-1)} c_k^{(g)} n^k. \tag{16}$$

Bond animals can thus be evaluated up to dimension $n = 12$ and Eq. (16) can be solved up to $e = 5$. We thereby identify $2, 5, 10, 19$ pairs of (t, g) polynomials for $e = 2, 3, 4, 5$, respectively [28]. Here as well, because the total number of bond animals, $A_e \sim n^{2e}$, grows exponentially with e, results for $e > 5$ lie beyond current computational reach.

Invoking Eq. (15) we obtain

$$\begin{aligned}
b_0 &= n(n-1),\\
b_1 &= 2n(n-1)(2n^2-2n-1),\\
b_2 &= 2n(n-1)(4n^4-8n^3+9),\\
b_3 &= 2n(n-1)(8n^6-24n^5+12n^4\\
&\quad -8n^3+27n^2+131n-218),\\
b_4 &= 2n(n-1)(16n^8-64n^7+64n^6-48n^5\\
&\quad +56n^4+328n^3+1534n^2-7778n+7499).
\end{aligned} \tag{17}$$

Note that we have $b_\ell \sim \mathcal{O}(n^{2\ell+2})$ which is two orders (in n) higher than for site percolation. Note also that unlike for site percolation, b_0/b_1 here has yet to converge to the Bethe lattice limit at leading order. For $\ell \geq 2$, however, the Padé approximant $p_c^{\text{bond}} \approx b_{\ell-1}/b_\ell$ has an error of $\mathcal{O}(n^{-(\ell+3)})$ one order smaller than for p_c^{site}. In particular, expanding b_3/b_4 gives

$$p_c^{\text{bond}} = \frac{1}{\sigma} + \frac{1}{n^5} + \frac{81}{16n^6} + \mathcal{O}(n^{-7}). \tag{18}$$

The accuracy of this series is also evaluated in Sec. V B.

IV. INVASION PERCOLATION

In this section we describe the invasion percolation algorithm by Mertens and Moore [2] (itself derived from Ref. 29) for an arbitrary lattice structure, and then analyze its complexity for the considered lattices.

As stated in the introduction, the algorithm grows a single cluster without explicitly storing the lattice grid. This strategy requires a moderate memory usage, even in high dimension, especially compared to the widely used Hoshen-Kopelman algorithm which maintains a periodic lattice grid [30]. It is also less computational expensive than the Leath algorithm, which grows clusters according to standard percolation statistics, and therefore samples the whole cluster size distribution [31]. Although invasion percolation and ordinary percolation differ in general, they both give the same percolating critical cluster for the nearest-neighbor site or bond percolation [16]. The method thus provides a direct estimate of the percolation threshold. It may also be used to efficiently extract critical exponents associated with percolating cluster, such as the fractal dimension d_f and the subdiffusion exponent d_w [1].

In our implementation, two data structures are used: (i) a set (collection of unique elements) $\mathcal{S}$ to maintain all sites (or bonds) that belong to the cluster as well as those incident to them; and (ii) a priority queue $\mathcal{Q}$ for the stepwise growth of the cluster. The priority queue is a sorted data structure that maintains key-value pairs and keeps the keys sorted. Insertion of a pair has a logarithmic time complexity, and extraction (pop) of the pair with the smallest key takes a constant time.

For site percolation—starting from the origin—every neighboring vertex is inserted (following Sec. II) into $\mathcal{S}$. For each of these new vertices, a random weight $w_i \in [0,1)$ is assigned and the vertex is inserted into $\mathcal{Q}$ with w_i as the key. The priority queue then contains references of all perimeter sites (or bonds) of the current cluster in $\mathcal{S}$ and its size $|\mathcal{Q}| = t$. For the next step, the vertex of minimum weight in $\mathcal{Q}$ is popped, and the cluster size, N, is incremented. The previous steps are repeated until the pre-assigned cluster size $N = N_0$ is attained. The expected set size at a certain N, which we denote $B(N) = \langle|\mathcal{S}(N)|\rangle$, is computed by averaging the set size among independent realizations. For bond percolation, we start with an arbitrary bond incident to the origin and otherwise follow the same procedure.

The cluster obtained by invasion percolation simultaneously approaches the giant component at p_c with the scaling form [2, 16]

$$\frac{N}{B(N)} \approx p_c(1 - cN^{-\delta}) \tag{19}$$

where δ is the convergence exponent of finite-size scaling and c is a fitting constant.

For each instance, the space complexity is

$$n|\mathcal{S}| + |\mathcal{Q}| \sim nN/p_c \sim O(n\sigma N)$$

where the factor of n accounts for the size of an n-dimensional vector. For D_n lattice the space complexity is thus $\mathcal{O}(n^3 N)$. Although the space complexity is larger, by a factor of n, than for $\mathbb{Z}^n$ lattices [$\mathcal{O}(n^2 N)$], the memory requirement remains moderate for contemporary computers. The time complexity depends on the number of insertions to $\mathcal{S}$ which is $\mathcal{O}(N/p_c) \sim \mathcal{O}(n^2 N)$, in addition to the complexity of the insertion to $\mathcal{Q}$ which is at most $N \times \mathcal{O}(n + \log|\mathcal{Q}|) \approx \mathcal{O}(N \log N)$, and thus $\mathcal{O}(N \log N)$ in total. In practice, we can grow clusters up to $N_0 = 1.5 \times 10^7$ in $n = 3$ and up to 2×10^5 in $n = 13$, within a memory usage of less than 10 GB. At least 10^4 independent clusters are obtained for each lattice, with each realization is usually taking less than a minute on an AMD Ryzen 3900x processor. Therefore each given model and dimension costs ~ 200 h of CPU time.

V. RESULT AND DISCUSSION

In this section we compare the numerical threshold values obtained from the invasion percolation described in Sec. IV with the series expansion results obtained in Sec. III.

A. Numerical thresholds

Table II reports both site and bond percolation thresholds for D_n as well as for E_8-related lattices obtained by fitting the numerical $N/B(N)$ results with Eq. 19.

TABLE II. Site and bond percolation thresholds on D_n and E_8-related lattices along with available reference values

Lattice	p_c^{site}	p_c^{bond}	Eq. (13) (site)	Eq. (18) (bond)	Eq. (12) (Bethe lattice)
D_3	0.199 236(4)	0.120 169(2)	0.241 243	0.101 969	0.090 909
	0.199 235 17(20) [8]	0.120 163 5(10) [7]			
D_4	0.084 200 1(11)	0.049 519 3(8)	0.086 256	0.045 690	0.043 478
	0.084 10(23) [9]	0.049 517(1) [5]			
D_5	0.043 591 3(6)	0.027 181 3(2)	0.042 959	0.026 285	0.025 641
	0.043 1(3) [32]	0.026(2) [32]			
D_6	0.026 026 74(12)	0.017 415 56(5)	0.025 559	0.017 186	0.016 949
	0.025 2(5) [32]				
D_7	0.017 167 30(5)	0.012 217 868(13)	0.016 932	0.012 151	0.012 048
D_8	0.012 153 92(4)	0.009 081 804(6)	0.012 043	0.009 059	0.009 009
D_9	0.009 058 70(2)	0.007 028 457(3)	0.009 006	0.007 019	0.006 993
D_{10}	0.007 016 353(9)	0.005 605 579(6)	0.006 990	0.005 602	0.005 587
D_{11}	0.005 597 592(4)	0.004 577 155(3)	0.005 584	0.004 575	0.004 566
D_{12}	0.004 571 339(4)	0.003 808 960(2)	0.004 564	0.003 807 996	0.003 802
D_{13}	0.003 804 565(3)	0.003 219 701 3(14)	0.003 801	0.003 219 176	0.003 215
E_6	0.021 940 21(14)	0.014 432 05(8)			0.014 085
E_7	0.011 623 06(4)	0.008 083 68(2)			0.008 000
E_8	0.005 769 91(2)	0.004 202 07(2)			0.004 184
Λ_9	0.004 808 39(2)	0.003 700 865(11)			0.003 690

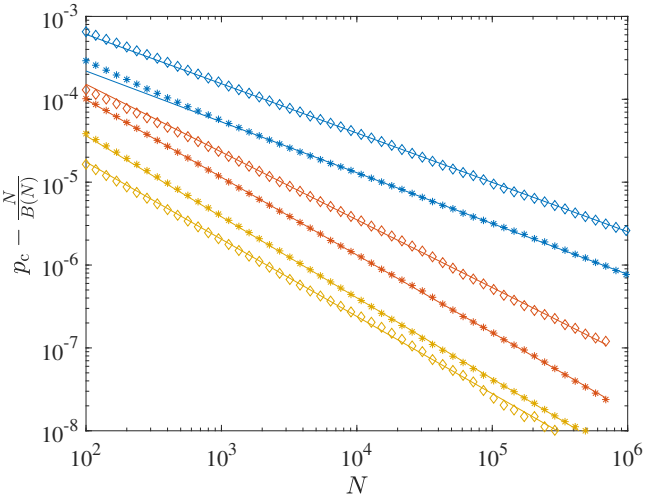

FIG. 1. Convergence of $N/B(N)$ to site (diamonds) and bond (asterisks) percolation thresholds on D_n lattices in $n = 4$ (blue), 8 (red) and 12 (yellow). Finite-size correction scales as $\mathcal{O}(N^{-\delta})$ (solid lines), where $\delta \to 1$ as $n \to \infty$.

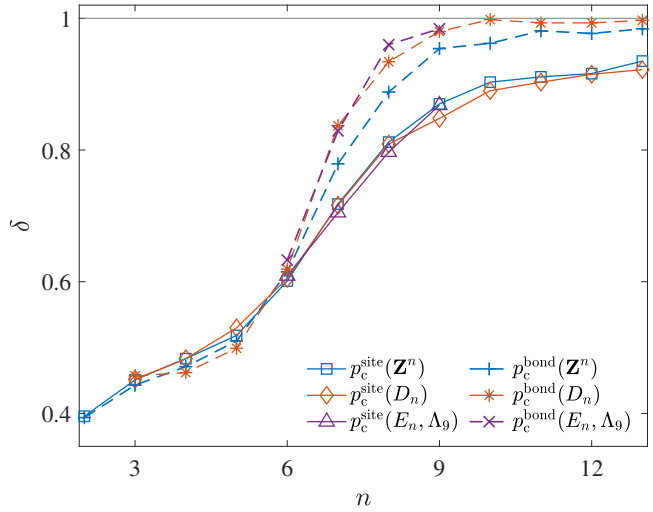

FIG. 2. Convergence exponent δ for site and bond percolation on $\mathbb{Z}^n$, D_n and E_8-related lattices, E_6, E_7, E_8 and Λ_9. (Results for $\mathbb{Z}^n$ site percolation in $n = 4$ to 13 are from Ref. 2.) Error bars from fitting are smaller than (comparable to) the marker size in $n \leq 6$ (in $n > 6$). Lines are guides to the eye. Note that δ generically grows with n but its value is not universal.

For comparison, the series expansion forms of Eqs. (13) and (18) as well as Bethe lattice approximation from Eq. (12) are also included. Results for $n = 3$-6 are consistent with published values, and except for $n = 3$ our results are at least an order of magnitude more accurate. For $6 < n \leq 13$ no prior results are known. Remarkably, as for $\mathbb{Z}^n$ lattices [2], p_c results for D_n lattices are obtained with higher precision—for comparable computational efforts—as n increases (Fig. 1). Because the convergence exponent δ [Eq. (19)] increases with n, finite-size corrections indeed then decay faster. For the range of n considered, this advantage compensates the decrease in N_0 imposed by the growing memory cost. As a result, the method achieves a higher absolute accuracy in higher dimensions, and a relative uncertainty of 10^{-6} to 10^{-7} is obtained for all investigated systems.

Because δ controls the convergence rate of invasion percolation, it is interesting to compare its behavior for different lattices (Fig. 2). At first glance, δ increases with n for both $\mathbb{Z}^n$ and D_n lattices and tends to 1 as dimension increases, as expected from the Bethe lattice analysis [2, 16]. While for site percolation on $\mathbb{Z}^n$, D_n and E_8-related lattices δ appears similar, the exponent evolves differently for bond percolation on different lattices as well as for either type of percolation on a same

lattice. Its evolution further appears to change around the upper critical dimension, $n_u = 6$, with δ for different percolation criteria separating in $n > 6$. Because δ is not universal—depending on the type of percolation as well as on lattice geometry, it is not clear whether this phenomenon is merely a coincidence or intimately related to the mean-field physics. Further theoretical consideration of this quantity would be needed to say more definitely. In any event, δ may be a useful quantity for selecting a lattice to study percolation criticality. A greater δ indeed implies a faster decay of certain finite-size corrections.

B. Comparison with series expansion

Our precise numerical thresholds for D_n lattices can be compared with the series prediction for both site and bond percolation in Sec. III. The relative error of the expansion up to $n^{-\ell}$ term, defined as

$$\eta_p^{(\ell)} = \left| p_{\mathrm{c,simulation}} - p_{\mathrm{c,series}}^{(\ell)} \right| / p_{\mathrm{c,simulation}}, \tag{20}$$

is shown in Fig. 3(a). As expected, the various thresholds converges gradually to the Bethe lattice value, $1/\sigma$, in the large n limit. For site percolation, this convergence rate is fairly slow—a $\sim 10\%$ deviation persists even in $n = 13$—but introducing higher-order terms in the series dramatically reduces that error. In particular, including terms of order up to n^{-6} leads to a relative error of $\sim 0.1\%$ in $n = 13$. For bond percolation, because the prefactors for both n^{-3} and n^{-4} in the expansion form are zero, the deviation is already down to $\sim 0.1\%$ in $n = 13$. Including two more terms in Eq. (18) further divides the error by a factor $\sim n^2$. The series expansion in Eqs. (13) and (18) is thus expected to predict percolation thresholds with very high accuracy for $n > 13$.

Percolation thresholds for $\mathbb{Z}^n$, D_n and E_8-related lattices are compared with the Bethe lattice result in Eq. (12). Although a dimensional series expansion is not available for E_8-related lattices, their large site connectivity (see Sec. II) brings them reasonably close to the Bethe lattice result already. In all three cases, the Bethe lattice prediction indeed better matches the bond than the site percolation threshold [Fig. 3(b)]. For $\mathbb{Z}^n$ and D_n lattices, as discussed above, this result was expected from the vanishing first subleading coefficients in series expansion. For E_8-related lattices, for which no such series exist, the same trend is observed. More specifically, the deviation is $\lesssim 1\%$ for bond percolation and $\lesssim 40\%$ for site percolation. This feature thus appears to be generic for lattices beyond $\mathbb{Z}^n$, for which it was first reported [25, 33, 34]. Yet it lacks a physical explanation. A generic scaling form for the percolation threshold beyond the Bethe lattice approximation might be informative in this respect, but is still found lacking.

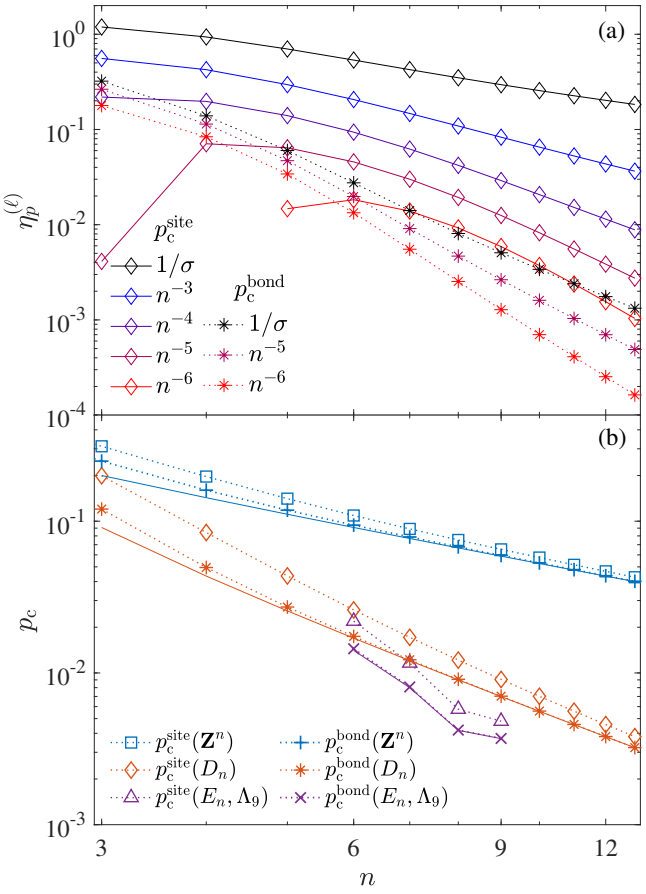

FIG. 3. (a) Relative error for the site (diamonds) and bond (asterisks) percolation thresholds on D_n lattices predicted by series expansion for various highest-order terms. Note that for site percolation the high-order lines are truncated in small n because the relative error then changes sign. Lines are guides to the eye. (b) Percolation thresholds on $\mathbb{Z}^n$, D_n and E_8-related lattices (markers with dotted line) compared to the Bethe lattice limit $1/\sigma$ (solid line), which matches well the bond percolation threshold in all three lattice types.

VI. CONCLUSION

We have reported the series expansion and numerical percolation thresholds for D_n lattices as well as the numerical thresholds for E_8-related lattices from $n = 6$ to 9. The excellent agreement between the two independent approaches cross-validates the methods used and their results. More interestingly, the Bethe lattice approximation to the percolation threshold generically presents a markedly higher precision for bond than for site percolation for all lattices considered, as was first observed for hypercubic lattices. This finding should motivate further theoretical studies of a generic scaling form.

The invasion percolation scheme itself presents some interesting physical features. Although its convergence exponent δ evolves similarly in low dimensions, bond percolation presents much faster decaying finite-size corrections than site percolation in $n > 6$. Whether the

effect is related to the upper critical dimension, however, remains unclear. This finding nevertheless suggests that pre-asymptotic corrections might be most efficiently suppressed for bond percolation models, and thus that these lattices may be preferable for evaluating certain critical exponents. Our findings therefore identify unresolved critical features of percolation theory, and set the stage for investigating percolation criticality on high-dimensional lattices beyond the conventional hypercubic geometry.

ACKNOWLEDGMENTS

We thank R. M. Ziff for carefully maintaining the Percolation threshold Wikipedia page, which has greatly facilitated our literature search. This work was supported by a grant from the Simons Foundation (#454937). The computations were carried out on the Duke Compute Cluster and Open Science Grid [35, 36], supported by National Science Foundation award 1148698, and the U.S. Department of Energy's Office of Science. Data relevant to this work have been archived and can be accessed at the Duke Digital Repository [28].

[1] D. Stauffer and A. Aharony, *Introduction To Percolation Theory* (Taylor & Francis, 1994).
[2] S. Mertens and C. Moore, Phys. Rev. E **98**, 022120 (2018).
[3] W. Huang, P. Hou, J. Wang, R. M. Ziff, and Y. Deng, Phys. Rev. E **97**, 022107 (2018).
[4] G. Biroli, P. Charbonneau, and Y. Hu, Phys. Rev. E **99**, 022118 (2019).
[5] Z. Xun and R. M. Ziff, Phys. Rev. Research **2**, 013067 (2020).
[6] J. C. Wierman, Phys. Rev. E **66**, 027105 (2002).
[7] C. D. Lorenz and R. M. Ziff, Phys. Rev. E **57**, 230 (1998).
[8] X. Xu, J. Wang, J.-P. Lv, and Y. Deng, Front. Phys. **9**, 113 (2014).
[9] M. Kotwica, P. Gronek, and K. Malarz, Int. J. Mod. Phys. C **30**, 1950055 (2019).
[10] Z. Xun and R. M. Ziff, Phys. Rev. E **102**, 012102 (2020).
[11] C. R. Scullard and R. M. Ziff, Phys. Rev. Lett. **100**, 185701 (2008).
[12] C. R. Scullard and R. M. Ziff, J. Stat. Mech. Theory Exp. **2010**, P03021 (2010).
[13] J. L. Jacobsen, J. Phys. A **48**, 454003 (2015).
[14] C. R. Scullard and J. L. Jacobsen, Phys. Rev. Research **2**, 012050(R) (2020).
[15] S. Kirkpatrick, Phys. Rev. Lett. **36**, 69 (1976).
[16] S. Mertens and C. Moore, Phys. Rev. E **96**, 042116 (2017).
[17] J. H. Conway and N. J. A. Sloane, "Certain important lattices and their properties," in *Sphere Packings, Lattices and Groups* (Springer New York, New York, NY, 1988) pp. 94–135.
[18] L. Berthier, P. Charbonneau, and J. Kundu, Phys. Rev. Lett. **125**, 108001 (2020).
[19] G. Biroli, P. Charbonneau, E. I. Corwin, Y. Hu, H. Ikeda, G. Szamel, and F. Zamponi, Phys. Rev. E **103**, L030104 (2021).
[20] P. Charbonneau and P. K. Morse, Phys. Rev. Lett. **126**, 088001 (2021).
[21] J. H. Convay and N. J. A. Sloane, IEEE Trans. Inf. Theory **28**, 227 (1982).
[22] J. H. Conway and N. J. A. Sloane, Discrete Comput. Geom. **13**, 383 (1995).
[23] C. D. Lorenz, R. May, and R. M. Ziff, J. Stat. Phys. **98**, 961 (2000).
[24] W. F. Lunnon, Comput. J **18**, 366 (1975).
[25] S. Mertens and C. Moore, J. Phys. A **51**, 475001 (2018).
[26] M. Heydenreich and K. Matzke, arXiv preprint (2019), arXiv:1912.04584.
[27] S. Mertens, J. Stat. Phys. **58**, 1095 (1990).
[28] "Duke digital repository," https://doi.org/10.7924/xxxxxxxxx.
[29] D. Wilkinson and J. F. Willemsen, J. Phys. A **16**, 3365 (1983).
[30] J. Hoshen and R. Kopelman, Phys. Rev. B **14**, 3438 (1976).
[31] P. Leath, Phys. Rev. B **14**, 5046 (1976).
[32] S. C. Van der Marck, Int. J. Mod. Phys. C **9**, 529 (1998).
[33] D. Gaunt, M. Sykes, and H. Ruskin, J. Phys. A **9**, 1899 (1976).
[34] D. Gaunt and H. Ruskin, J. Phys. A **11**, 1369 (1978).
[35] R. Pordes, D. Petravick, B. Kramer, D. Olson, M. Livny, A. Roy, P. Avery, K. Blackburn, T. Wenaus, F. Würthwein, I. Foster, R. Gardner, M. Wilde, A. Blatecky, J. McGee, and R. Quick, in *J. Phys. Conf. Ser.*, Vol. 78 (2007) p. 012057.
[36] I. Sfiligoi, D. C. Bradley, B. Holzman, P. Mhashilkar, S. Padhi, and F. Wurthwein, in *2009 WRI World Congress on Computer Science and Information Engineering*, Vol. 2 (2009) pp. 428–432.