Random site percolation on honeycomb lattices with complex neighborhoods

Krzysztof Malarz

AGH University, Faculty of Physics and Applied Computer Science, al. Mickiewicza 30, 30-059 Kraków, Poland
(Dated: February 13, 2024)

We present a rough estimation—up to four significant digits, based on the scaling hypothesis and the probability of belonging to the largest cluster vs. the occupation probability—of the critical occupation probabilities for the random site percolation problem on a honeycomb lattice with complex neighborhoods containing sites up to the fifth coordination zone. There are 31 such neighborhoods with their radius ranging from one to three and containing from three to 24 sites. For two-dimensional regular lattices with compact extended-range neighborhoods, in the limit of the large number $z$ of sites in the neighborhoods, the site percolation thresholds $p_c$ follow the dependency $p_c \propto 1/z$, as recently shown by Xun, Hao and Ziff [Physical Review E 105, 024105 (2022)]. On the contrary, noncompact neighborhoods (with holes) destroy this dependence due to the degeneracy of the percolation threshold (several values of $p_c$ corresponding to the same number $z$ of sites in the neighborhoods). An example of a single-value index $\xi = \sum_i z_i r_i^{-1}$, where $z_i$ and $r_i$ are the number of sites and radius of the $i$-th coordination zone, respectively characterizing the neighborhood and allowing avoiding the above-mentioned degeneracy is presented. The percolation threshold obtained follows the inverse square root dependence $p_c \propto 1/\sqrt{\xi}$. The functions boundaries() (written in C) for basic neighborhoods (for the unique coordination zone) for the Newman and Ziff algorithm [Physical Review E 64, 016706 (2001)] are also presented.

I. INTRODUCTION

The percolation thresholds [1–4] (see Wikipedia page 5 for the most current and comprehensive data set and Refs. 6 and 7 for recent reviews) were initially estimated for the nearest neighbor interactions [8–10] but later also complex neighborhoods (called extended in case of compact one) were studied for 2D (square [11–17], triangular [11, 21, 18–21], honeycomb [11, 21] and other Archimedean [21]), 3D (simple cubic [17, 21–23]) and 4D (simple hypercubic [24]) lattices. As the exact values of $p_c$ are known only in several cases [25] the most of effort in their calculations is computational.

Very recently, Xun, Hao, and Ziff [21] numerically estimated the site and bond percolation thresholds for all eleven Archimedean lattices with extended compact neighborhoods containing sites up to the tenth coordination zone. They found that for the site percolation problem, the critical site occupation probability $p_c$ follows asymptotically

$$p_c = a/z$$

with the total number $z$ of sites in the neighborhood and $a \approx 4.51235$. This dependence should be reached exactly for the percolation of discs, that is, for compact neighborhoods with a large number $z$ of sites that make up the neighborhood. The finite-$z$ effect may be taken into account by an additional including term $b$ in the denominator of Equation (1) [26]

$$p_c = c/(z + b)$$

with $b = 3$ for the two-dimensional lattices [21]. The third universal scaling studied in Ref. 21 was

$$p_c = 1 - \exp(d/z)$$

proposed by Koza et al. [27, 28].

In contrast, noncompact neighborhoods (with holes) destroy dependencies (1), (2), (3) due to the degeneracy of the percolation threshold (several values of $p_c$ corresponding to the same number $z$ of sites in the neighborhoods). This degeneracy is observed for square [16] and triangular [20] lattices. Here, we show that this degeneracy is also present for the honeycomb lattice, for neighborhoods containing sites up to the fifth coordination zone. These neighborhoods (see Figure 1 in Appendix B) are combined from five basic neighborhoods presented in Figure 1. The lattice names follow the convention proposed in Ref. 17 reflecting the lattice topology (here HC, i.e., honeycomb lattice) and a numerical string specifying the coordination zones $i$, where the sites constituting the neighborhood come from.

To avoid the above-mentioned degeneracy, the weighted squared distance $r_i^2$ of $z_i$ sites in the given neighborhood that belong to the $i$-th coordination zone

$$\xi = \sum_i z_i r_i^2 / i$$

has been proposed in Ref. 20. For a triangular lattice, this index $\xi$ allows differentiation among various neighborhoods and the association of the percolation threshold with its value according to the power law:

$$p_c \propto \xi^{-\gamma}$$

with $\gamma \approx 0.710(19)$ [20].

Here, $i$, $j$ we estimate the site percolation thresholds $p_c$ for the honeycomb lattice with complex neighborhoods...
containing sites up to the 5-th coordination zone. The second aim of this paper is to check ii) if the index $\xi$ (4) suppresses $p_c(z)$ degeneracy and iii) if the power law dependence (5) proposed for complex neighborhoods and the triangular lattice also holds for the honeycomb lattice.

The paper is organized as follows. In Section II we describe the details of the calculation of the percolation thresholds and the computer implementation of the proposed methodology. Section III contains the results obtained, that is, the percolation thresholds for 31 neighborhoods. Section IV is devoted to a discussion of the results and to present the possible directions of further studies. Finally, three appendices provide: numerical procedures applied here (Appendix A); shapes of neighborhoods considered here (Appendix B); and figures utilized for estimation of percolation thresholds (Appendix C).

II. COMPUTATIONS

To evaluate the percolation threshold $p_c$ we rely on the finite-size scaling hypothesis [3, p. 17], [29], [30, p. 77]. According to this hypothesis, near a geometrical phase transition, many quantities $A$ obey a scaling relation

$$A(p; L) = L^{-z} F \left( \left( p - p_c \right)^{L^{1/\nu}} \right),$$

where $p$ is an occupation probability, $L$ is the linear size of the system, $x$ and $\nu$ are characteristic exponents and $F$ is the universal scaling function. For $p = p_c$, the product $A(p; L) L^z = F(0)$—independently of $L$. To find $p_c$, we need to plot $L^z A(p; L)$ for various linear sizes of the system $L$ and the common point of intersection of all these curves predict values of $p_c$. As an observable $A$ we choose the probability $P_{\text{max}}$, that a randomly selected site belongs to the largest cluster of occupied sites. For the two-dimensional percolation problem and the quantity $A = P_{\text{max}}$ the characteristic exponent $x = \frac{5}{36}/\frac{4}{3} = \frac{5}{48}$ [3, p. 54]. The probability $P_{\text{max}}$ can be geometrically calculated as the average size of the largest cluster of occupied sites ($S_{\text{max}}$) divided by the total number of sites $N$. The brackets $\langle \cdot \cdot \cdot \rangle$ represent the averaging procedure in the $R$ lattice realization.

To estimate the size of the largest cluster of occupied sites $S_{\text{max}}$ we utilize the Newman–Ziff algorithm [31]. The algorithm allows you to quickly find $S_{\text{max}}$ depending on the number $n$ ($1 \leq n \leq N$) of occupied sites. This immediately leads to the dependence of $S_{\text{max}}(p; N)$ with $p = n/N$, and the size of the system $N$ controls the natural separation $\Delta p = 1/N$ between the available values of the probabilities of occupation $p$. These separations are clearly visible in Figure 2a for the smallest system size, $N = 127^2$. To overcome the problem of building $S_{\text{max}}(p; N)$ for arbitrarily chosen values of $p$, we again use the idea of Newman and Ziff [31]:

$$S_{\text{max}}(p; N) = \sum_{n=1}^{N} \langle S_{\text{max}}(n; N) \rangle B(n; N, p),$$

where $B(n; N, p)$ are the binomial distribution coefficients

$$B(n; N, p) = \binom{N}{n} p^n (1 - p)^{(N-n)}.$$  

For large enough systems, these coefficients may be successfully approximated by the normal distribution

$$G(n; \mu, \sigma) = \frac{1}{\sqrt{2\pi} \sigma^2} \exp \left( -\frac{(n - \mu)^2}{2\sigma^2} \right),$$

with $\mu = pN$ and $\sigma^2 = p(1-p)N$. An example of such calculations of the probability that an arbitrarily chosen site belongs to the largest cluster

$$P_{\text{max}}(p; N) = S_{\text{max}}(p; N)/L^2.$$
In their original paper [31]) are responsible for providing boundaries() procedures defining the network topology. Here, we consider the honeycomb lattice with periodic boundary conditions and complex neighborhoods. There are 31 such neighborhoods (see Figure X1 in Appendix B). These complex neighborhoods are combined with the basic neighborhoods presented in Figure 1.

The underlying neighborhoods can be mapped into a square brick wall lattice [10]. For the HC-1, HC-3, HC-4 lattices, the implementation of neighboring sites must be separated into odd and even site labels, as the honeycomb lattice is excluded from the set of two-dimensional Bravais lattices [32, p. 8]. These computerized versions of lattices with basic neighborhoods (Figure 1) mapped on a square brick wall lattice are presented in Figure 3. The boundaries() functions necessary for the Newman–Ziff algorithm are presented in Appendix A.

III. RESULTS

In Figure X2 (in Appendix C) the dependencies $P_{\text{max}}L^{5/48}$ on the probability of occupation $p$ obtained by the procedure described in Section II (see Figure 2) are presented. Data are based on $\langle S_{\text{max}}(n,L^2) \rangle$ simulated with the Newman–Ziff algorithm for $N = L^2$ sites and $L = 127, 255, 511, 1023, 2047$, and $4095$ averaged over $R = 10^4$ simulations with $\Delta p = 10^{-4}$ data separation. The percolation thresholds $p_c$ predicted by a common point of six curves for various $L$ with an accuracy given by an assumed data separation constant $\Delta p = 10^{-4}$ are collected in Table I. The only exception is HC-3, where the curves for various $L$ do not intersect each other at a single point (see Figure X2d). Fortunately, $p_c$(HC-3) = $p_c$(HC-1) as the HC-3 lattice is equivalent to HC-1 due to the symmetry of the neighborhood analysis (see Figure 1c).

IV. CONCLUSIONS

For complex and noncompact neighborhoods, the total number $z$ of sites in the neighborhood is insufficient to differentiate between neighborhoods in terms of the percolation thresholds associated with this type of neighborhood (see Figure 4a). The limiting case [of discs, according to Equation (1)] with $a = 4.512$ [21] and its finite-$z$ corrections [Equation (2)] are presented there with black dashed and orange solid lines, respectively. In the latter case, the least squares fit gives $c = 4.630(75)$ and $b = 3.64(13)$. The fitting procedure were performed via five points corresponding to the compact neighborhoods (HC-1, HC-12, HC-123, HC-1234, HC-12345) marked by the open circles.

The degeneracy of $p_c(z)$ mentioned above can be solved by introducing the index $\xi$ [Equation (4)] as proposed in Ref. 20. In fact, for the triangular lattice and complex neighborhoods, the various neighborhoods are characterized by various values of the $\xi$ index [20].
of the $\xi$ index to distinguish between various neighborhoods is also the same for the honeycomb lattice (see Figure 4b). Unfortunately, the power law dependency of $p_c$ on $\xi$ does not apply to the honeycomb lattice (see Figure 4b). Thus, we propose yet another heuristic index

$$\zeta = \sum_i z_i r_i, \quad (11)$$

which seems to be much more appropriate here and which gives a much better (but not perfect) power fit

$$p_c = c_1 \zeta^{-\gamma} \quad (12)$$

with $\gamma \approx \frac{1}{2}$ (the fit according to the least squares method gives $\gamma = 0.4981(90)$). The $p_c$ values for the equivalents of the HC-1 (HC-3, marked with a cross) and TR-1 (HC-2 and HC-5, marked with triangles) and HC-2,5 (marked with an open circle) neighborhoods are excluded from the fitting. Basing on the obtained value of $\gamma$ we have also checked the inverse square root dependence of the percolation threshold on the $\zeta$ index

$$p_c = c_2/\sqrt{\zeta} \quad (13)$$

with a fitted value of constant $c_2 = 1.2251(99)$. The dependence of the percolation threshold $p_c$ on the newly proposed index $\zeta$ and the fits according to Equations (12) and (13) are presented in Figure 4c with solid orange and black dashed lines, respectively.

The values of $\xi$ and $\zeta$ are also presented in Table I.

In conclusion, basing on the Newman–Ziff algorithm and finite-size scaling analysis we have calculated the random site percolation thresholds $p_c$ for complex neighbourhoods on honeycomb lattice. We propose a scalar quantity $\zeta$ which may be helpful for differentiating among various neighbourhoods. The quantity is based on sites number $z_i$ and sites distances $r_i$ to the central site in the sites $i$th coordination zone. The dependency of $p_c$ on this newly proposed index $\zeta$ follows roughly a inverse square root dependence $p_c \propto 1/\sqrt{\zeta}$.

[1] S. R. Broadbent and J. M. Hammersley, “Percolation processes: I. Crystals and mazes,” Mathematical Proceedings of the Cambridge Philosophical Society 53, 629–641 (1957).
[2] J. M. Hammersley, “Percolation processes: II. The connective constant,” Mathematical Proceedings of the Cambridge Philosophical Society 53, 642–645 (1957).
[3] D. Stauffer and A. Aharony, Introduction to Percolation Theory, 2nd ed. (Taylor and Francis, London, 1994).
[4] J. Wierman, “Percolation theory,” in Wiley StatsRef: Statistics Reference Online (American Cancer Society, 2014) pp. 1–9.
[5] “en.wikipedia.org/wiki/percolation_threshold,” (2020).
[6] M. Li, R.-R. Liu, L. Lü, M.-B. Hu, S. Xu, and Y.-C. Zhang, “Percolation on complex networks: Theory and application,” Physics Reports, 1–68 (2021).
[7] A. A. Saberi, “Recent advances in percolation theory and its applications,” Physics Reports 578, 1–32 (2015).
[8] P. Dean, “A new Monte Carlo method for percolation problems on a lattice,” Mathematical Proceedings of the Cambridge Philosophical Society 59, 397–410 (1963).
[9] P. Dean and N. F. Bird, “Monte Carlo estimates of critical percolation probabilities,” Mathematical Proceedings of the Cambridge Philosophical Society 63, 477–479 (1967).
[10] P. N. Suding and R. M. Ziff, “Site percolation thresholds for Archimedean lattices,” Physical Review E 60, 275–283 (1999).
[11] N. W. Dalton, C. Domb, and M. F. Sykes, “Dependence of critical concentration of a dilute ferromagnet on the range of interaction,” Proceedings of the Physical Society 83, 496–498 (1964).
[12] C. Domb and N. W. Dalton, “Crystal statistics with long-range forces: I. The equivalent neighbour model,” Proceedings of the Physical Society 89, 859–871 (1966).
[13] M. Gouker and F. Family, “Evidence for classical critical behavior in long-range site percolation,” Physical Review B 28, 1449–1452 (1983).
[14] K. Malarz and S. Galam, “Square-lattice site percolation at increasing ranges of neighbor bonds,” Physical Review
FIG. 4: Percolation thresholds $p_c$ for complex neighborhoods in the honeycomb lattice. (a) Dependence of degenerated values of $p_c$ on the total number $z$ of sites in the neighborhood. The dependence (1) is also presented by a dashed line. (b) Dependence of $p_c$ vs. index $\xi$ (4) for complex neighborhoods. The $p_c$ values for equivalents of $hc^{-1}$ and $hc^{-2}$ are marked with crosses. (c) Dependence of $p_c$ vs. index $\zeta$ (11) for complex neighborhoods. The $p_c$ values for equivalents of $hc^{-1}$, $hc^{-2}$ and $hc^{-5}$, marked with triangles) neighborhoods are excluded from the fitting. The percolation threshold for $hc^{-2,5}$ (marked with an open circle) is also excluded from a fitting procedure.

### Table I: Estimated values of percolation thresholds $p_c$ for various complex neighborhoods.

| Lattice | $z$ | $\xi$ | $\zeta$ | $p_c$ |
|---------|-----|-------|--------|-------|
| $hc^{-1}$ | 3  | 3     | 3      | 0.6970$^a$ |
| $hc^{-2}$ | 6  | 9     | 10.3923| 0.5000$^b$ |
| $hc^{-1,2}$ | 9  | 12    | 13.3923| 0.3630$^d$ |
| $hc^{-3}$ | 3  | 4     | 6      | 0.6970$^a$ |
| $hc^{-1,3}$ | 6  | 7     | 9      | 0.4132 |
| $hc^{-2,3}$ | 9  | 13    | 16.3923| 0.3139 |
| $hc^{-1,2,3}$ | 12 | 16    | 19.3923| 0.3030$^f$ |
| $hc^{-4}$ | 6  | 10.125| 15.5885| 0.3154 |
| $hc^{-1,4}$ | 9  | 13.125| 18.5885| 0.2704 |
| $hc^{-2,4}$ | 12 | 19.125| 25.9808| 0.2374 |
| $hc^{-3,4}$ | 9  | 14.125| 21.5885| 0.2556 |
| $hc^{-1,2,4}$ | 15 | 22.125| 28.9808| 0.2278 |
| $hc^{-1,3,4}$ | 12 | 17.125| 24.5885| 0.2161 |
| $hc^{-1,2,3,4}$ | 18 | 26.125| 34.9808| 0.2113$^g$ |
| $hc^{-5}$ | 6  | 10.8  | 18     | 0.5000$^h$ |
| $hc^{-1,5}$ | 9  | 13.8  | 21     | 0.2654 |
| $hc^{-2,5}$ | 12 | 19.8  | 23.3923| 0.2903 |
| $hc^{-3,5}$ | 9  | 14.8  | 24     | 0.2560 |
| $hc^{-4,5}$ | 12 | 20.925| 33.5885| 0.2014 |
| $hc^{-1,2,5}$ | 15 | 22.8  | 31.3923| 0.2147 |
| $hc^{-1,3,5}$ | 12 | 17.8  | 27     | 0.2290 |
| $hc^{-1,4,5}$ | 15 | 23.925| 36.5885| 0.1913 |
| $hc^{-2,3,5}$ | 15 | 23.8  | 34.9808| 0.2043 |
| $hc^{-2,4,5}$ | 18 | 29.925| 43.9808| 0.1752 |
| $hc^{-3,4,5}$ | 15 | 24.925| 39.5885| 0.1863 |
| $hc^{-1,2,3,5}$ | 18 | 26.8  | 37.3923| 0.1973 |
| $hc^{-1,2,4,5}$ | 21 | 32.925| 46.9808| 0.1720 |
| $hc^{-1,3,4,5}$ | 18 | 27.925| 42.5885| 0.1795 |
| $hc^{-2,3,4,5}$ | 21 | 33.925| 49.9808| 0.1673 |
| $hc^{-1,2,3,4,5}$ | 24 | 36.925| 52.9808| 0.1655$^b$ |

---

$^a$ 0.697043(3) [10], 0.700 [18], 0.6962(6) [33], 0.697040 [34], 0.697040230(5) [34], 0.6970402(1) [35]

$^b$ equivalent TR-1

$^c$ $\xi$ [5, p. 17]

$^d$ 0.359 [18]

$^e$ equivalent $hc^{-1}$

$^f$ 0.300 [11], 0.302960 [21]

$^g$ 0.210 [18]

$^h$ 0.164 [18]

---

[15] S. Galam and K. Malarz, “Restoring site percolation on damaged square lattices,” Physical Review E 72, 027103 (2005).

[16] M. Majewski and K. Malarz, “Square lattice site per-
Appendix A: Boundaries procedures

Set of boundaries() functions (written in C) to be replaced in the Newman–Ziff program published in Ref. 31 to obtain the single realization of $S_{\text{max}}(n; L)$ with the honeycomb lattice and the corresponding neighborhoods presented in Figure 1 and their computerized versions (see Figure 3).

1. hc-1

```c
void boundaries() {
int i, row, col;
for (row=0; row<L; row++)
for (col=0; col<L; col++) {
    i = row*L+col;
    // 1NN core:
    nn[i][0] = (N+i -1)%N;
    nn[i][1] = (N+i +1)%N;
    if (row%2==0) {
        if (col%2==0) {
            nn[i][2] = (N+1 +L)%N;
        } else {
            nn[i][2] = (N+1 -L)%N;
        }
    } else {
        if (col%2==0) {
            nn[i][2] = (N+i +L)%N;
        } else {
            nn[i][2] = (N+i -L)%N;
        }
    }
    // 1NN left border:
    if (i%L==0) {
        nn[i][0] = (N+L+i -1)%N;
    }
    // 1NN right border:
    if (((i+1)%L==0) {
        nn[i][1] = (N-L+i +1)%N;
    }
}
}
```

2. hc-2

```c
void boundaries() {
int i, row, col;
for (row=0; row<L; row++)
for (col=0; col<L; col++) {
    i = row*L+col;
    // 2NN core:
    nn[i][0] = (N+i -2 )%N;
    nn[i][1] = (N+i +2 )%N;
    nn[i][2] = (N+i +L-1)%N;
    nn[i][3] = (N+i +L+1)%N;
    nn[i][4] = (N+i -L-1)%N;
    nn[i][5] = (N+i -L+1)%N;
    // 2NN left border:
    if (i%L==0) {
        nn[i][0] = (N+L+1 -2 )%N;
    }
    // 2NN right border:
    if (((i+1)%L==0) {
        nn[i][0] = (N+L+i -2 )%N;
    }
}
```
3. hc-3

```c
void boundaries () {
  int i,row , col ;
  for ( row =0; row <L; row ++) {
    for ( col =0; col <L; col ++) {
      i= row *L+ col ;
      // 3NN core :
      if( row %2==0) {
        if( col %2==0) {
          nn[i][0] = (N+L+i +L)%N;
          nn[i][1] = (N+i +L)%N;
        }
        else {
          nn[i][0] = (N+i -L)%N;
          nn[i][1] = (N+i +L -2)%N;
        }
      }
      else {
        if( col %2==0) {
          nn[i][0] = (N+i -L)%N;
          nn[i][1] = (N+i +L -2)%N;
        }
        else {
          nn[i][0] = (N+i +L)%N;
          nn[i][1] = (N+i -L-2)%N;
        }
      }
      // 3NN left border :
      if(i%L ==0 || i%L ==1) {
        if( row %2==0) {
          if( col %2==0) {
            nn[i][1] = (N+i +L)%N;
            nn[i][2] = (N+i +L+2)%N;
          }
          else {
            nn[i][1] = (N+i -L)%N;
            nn[i][2] = (N+i +L-2)%N;
          }
        }
        else {
          if( col %2==0) {
            nn[i][1] = (N+i -L)%N;
            nn[i][2] = (N+i +L-2)%N;
          }
          else {
            nn[i][1] = (N+i +L)%N;
            nn[i][2] = (N+i -L+2)%N;
          }
        }
      }
      // 3NN right border :
      if (((i+1)%L==0 || (i+2)%L==0) {
        if( row %2==0) {
          if( col %2==0) {
            nn[i][2] = (N-L+i -L+2)%N;
          }
          else {
            nn[i][2] = (N-L+i +L+2)%N;
          }
        }
        else {
          if( col %2==0) {
            nn[i][2] = (N-L+i -L+2)%N;
          }
          else {
            nn[i][2] = (N-L+i +L+2)%N;
          }
        }
      }
    }
  }
}
```

4. hc-4

```c
void boundaries () {
  int i,row , col ;
  for ( row =0; row <L; row ++) {
    for ( col =0; col <L; col ++) {
      i= row *L+ col ;
      // 4NN core :
      if( row %2==0) {
        if( col %2==0) {
          nn[i][0] = (N+L+i +3)%N;
          nn[i][1] = (N+i +3)%N;
          nn[i][2] = (N+i +3-L)%N;
        }
        else {
          nn[i][0] = (N+i -3)%N;
          nn[i][1] = (N+i -3-L)%N;
          nn[i][2] = (N+i -3-L+3)%N;
        }
      }
      else {
        if( col %2==0) {
          nn[i][0] = (N+i +L-2)%N;
          nn[i][1] = (N+i +L+2)%N;
          nn[i][2] = (N+i +L+2-L)%N;
        }
        else {
          nn[i][0] = (N+i -L-2)%N;
          nn[i][1] = (N+i -L+2)%N;
          nn[i][2] = (N+i -L+2-L)%N;
        }
      }
      // 4NN left border :
      if(1%L==0 || 1%L==1 || 1%L==2) {
        if( row %2==0) {
          if(1%L==0) {
            nn[i][1] = (N+L+i +L-2)%N;
            nn[i][2] = (N+L+i -L-2)%N;
          }
          else {
            nn[i][1] = (N+L+i -2*L -1)%N;
            nn[i][2] = (N+L+i +2*L -1)%N;
          }
        }
        else {
          if(1%L==0) {
            nn[i][1] = (N+L+i +L-2)%N;
            nn[i][2] = (N+L+i -L-2)%N;
          }
          else {
            nn[i][1] = (N+L+i -2*L -1)%N;
            nn[i][2] = (N+L+i +2*L -1)%N;
          }
        }
      }
      // 4NN right border :
      if (((i+1)%L==0 || (i+2)%L==0) {
        if( row %2==0) {
          if( col %2==0) {
            nn[i][2] = (N-L+i -2*L-1)%N;
          }
          else {
            nn[i][2] = (N-L+i +2*L-1)%N;
          }
        }
        else {
          if( col %2==0) {
            nn[i][2] = (N-L+i +2*L-1)%N;
          }
          else {
            nn[i][2] = (N-L+i -2*L-1)%N;
          }
        }
      }
    }
  }
```

if (col % 2 == 0) {
    nn[i][5] = (N - L + i - L + 2) % N;
} else {
    nn[i][4] = (N - L + i + L + 2) % N;
}

if (((i + 1) % L == 0) || ((row % 2 == 0) {
    if (col % 2 == 0) {
        nn[i][5] = (N - L + i - 2 * L + 1) % N;
    } else {
        nn[i][4] = (N - L + i + 2 * L + 1) % N;
    }
} else {
    if (col % 2 == 0) {
        nn[i][4] = (N - L + i + 2 * L + 1) % N;
    } else {
        nn[i][5] = (N - L + i - 2 * L + 1) % N;
    }
}

void boundaries() {
    int i, row, col;
    for (row = 0; row < L; row++) {
        for (col = 0; col < L; col++) {
            i = row * L + col;
            // 5NN core:
            nn[i][0] = (i - L) % N;
            nn[i][1] = (i + 2 * L) % N;
            nn[i][2] = (i + L - 3) % N;
            nn[i][3] = (i + L + 3) % N;
            nn[i][4] = (i - L - 3) % N;
            nn[i][5] = (i - L + 3) % N;
            // 5NN left border:
            if (i % L == 0 || i % L == 1 || i % L == 2) {
                nn[i][2] = (N + L + i + L - 3) % N;
                nn[i][4] = (N + L + i - L - 3) % N;
            }
            // 5NN right border:
            if (((i + 1) % L == 0) || ((i + 2) % L == 0) || (i % L == 0)) {
                nn[i][3] = (N - L + i + L + 3) % N;
                nn[i][5] = (N - L + i - L + 3) % N;
            }
        }
    }
    // Appendix B: Complex neighborhoods shapes
    In Figure X1 31 complex neighborhoods in a honeycomb lattice are presented. The central site is marked by a red open circle, whereas the sites in its neighborhoods are marked by solid black circles. The names of the neighborhoods are presented in subfigure headlines.

    // Appendix C: Finite-size scaling
    In Figure X2 we show \( P_{\text{max}} \cdot L^{5/48} \) versus \( p \) for neighborhoods containing sites up to the fifth coordination zone. The names of the neighborhoods are presented in subfigure headlines.
FIG. X1: Complex neighborhoods in a honeycomb lattice
\( P_{\text{max}} \cdot L_{\beta/\nu} \)

(a) nc-1

(b) nc-2

(c) nc-1,2

(d) nc-3

(e) nc-1,3

(f) nc-2,3

(g) nc-1,2,3

(h) nc-4

(i) nc-1,4

(j) nc-2,4

(k) nc-3,4

(l) nc-1,2,4

(m) nc-1,3,4

(n) nc-2,3,4

(o) nc-1,2,3,4

(p) nc-5

(q) nc-1,5

(r) nc-2,5
FIG. X2: (Color online). $P_{\text{max}} \cdot L^{5/48}$ vs. $p$ for the honeycomb lattice and for the neighborhoods containing sites up to the fifth coordination zone.