Geometric Random Graphs and their Applications to Wireless Networks

Amitabha Bagchi

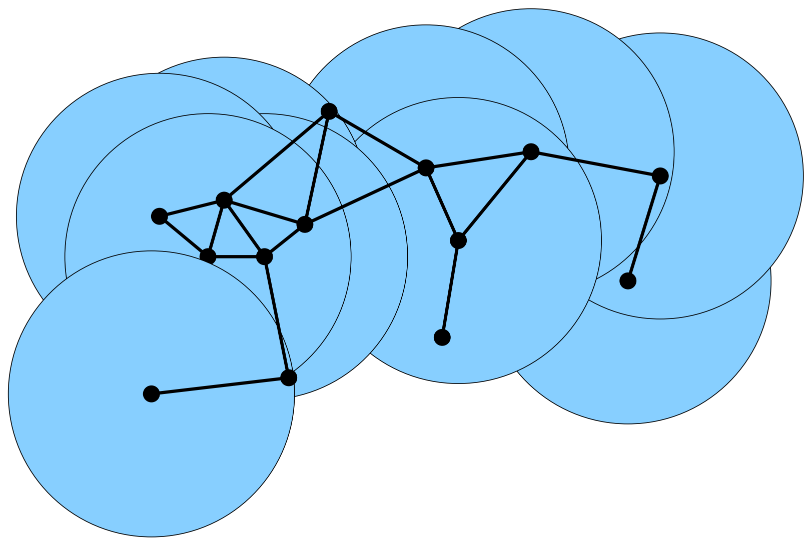
Computer Science & Engineering

Indian Institute of Technology, Delhi

What are geometric graphs?

- The vertices of the graphs are geometric objects.
- The edges are placed based on a geometric relationship between the objects.

Geometric graphs: An example



The *Unit Disk Graph*: Vertices are points, an edge is placed between x and y if $d(x, y) \leq 1$ where $d(\cdot, \cdot)$ is a distance function.

Geometric graphs: More examples

- Vertices: Line segments in $\mathbb{R}^d$. Edges: Between two line segments that intersect.
- Vertices: Voronoi cells of a point set in $\mathbb{R}^d$. Edges: Between two cells that share a $d-1$ dimensional facet.
- Vertices: Points in $\mathbb{R}^d$. Edges: From each point to the k points closest to it.

What are random graphs?

Given a graph $G = (V, E)$, a random graph is a probability distribution over the set of all subgraphs of G.

Random graphs: Some examples

- *The Erdös Rényi graph.* Given a complete graph on n vertices and a parameter $0 < p < 1$, retain each edge independently from all others with probability p.

- *Random d-regular graphs.* A graph on n vertices where each vertex has d randomly chosen neighbors

What are geometric random graphs?

Given a geometric graph $G = (V, E)$ a geometric random graph is a probability distribution over the set of all subgraphs of G.

Geometric random graphs: Some examples

- The unit disk graph on a randomly distributed set of points.
- The Voronoi graph with each Voronoi cell retained in the graph independently with probability p and removed with probability $1 - p$.

Modelling wireless networks

- It is not always possible to deterministically predict the position of the wireless nodes.
- The edges of a wireless network depend on the transmission and reception capabilities of the wireless antennas that nodes are equipped with.

Modelling WNs with Geometric Random Graphs

- We model the placement of wireless nodes (e.g. sensor nodes) as being placed randomly.
- Connection rules depend on the wireless transmission model. Unit disk graphs are the simplest model.

Real-life constraints should be respected.

Analyzing the properties of WNs using GRGs

- Formulate the service requirements of the network and its constraints in mathematical terms (measurable functions).
- Use the tools of probability and algorithmics to analyze these quantities.
- The analysis is useful if it provides insight into the working of the network, or presents demonstrably better ways of performing essential tasks.

Case study: Multihop wireless ad hoc sensor networks

Multihop communication is useful

- System tasks e.g. time synchronization.
- Collaborative tasks e.g. target tracking.

Just like ad hoc wireless networks in general, multihop WASNs require a connected topology. But there is one major difference

> *It is not necessary that every sensor be part of a connected network. It is only necessary that the density of connected sensors is high enough to perform the sensing function.*

Desirable properties of a multihop WASN

Sparsity. The degree of each node should be bounded.

Constant stretch. The distance between a pair of nodes along the edges of the network should be at most a constant times the Euclidean distance between the nodes.

Coverage. The range which has to be sensed must be well covered.

Local Computability. The network should be formed using local computations and exchange of information between each node and its neighbors.

The significance of constant stretch

Given a graph $G = (V, E)$ and a subgraph $H \subseteq G$ the *distance stretch* of H is defined as

$$\delta = \max_{u,v \in V} \frac{d_H(u, v)}{d_G(u, v)},$$

Given a connection network G and a subgraph H with distance stretch δ, the power stretch of H is at most δ^{β} for some $2 \leq \beta \leq 5$ (Li, Wan, Wang, 2001).

The model for sensor placement

Sensor locations are modeled by a point set generated by a homogenous Poisson point process of intensity λ in $\mathbb{R}^2$ i.e.

- Given a region A with area $V(A)$, the number of points in A is a r.v. X_A with distribution

$$\mathrm{P}(X_A = k) = e^{-\lambda V(A)} \cdot \frac{(\lambda V(A))^k}{k!}.$$

- The random variables for disjoint regions are independent.

Two geometric random graph models

Given a set of points S generated by a Poisson point process in $\mathbb{R}^2$ with density λ, we define two random graph models

- UDG$(2, \lambda)$: there is an edge between points $x \in S$ and $y \in S$ if $d(x, y) \leq 1$.
- NN$(2, k)$: there is an (undirected) edge between points $x \in S$ the k points in $S \setminus \{x\}$ that are closest to x.

We will show that there are settings of the parameters λ and k such that both these contain subgraphs with the properties we want.

Critical density for UDG$(2, \lambda)$

- There is a finite value $\lambda_c(2)$ s. t. for $\lambda > \lambda_c(2)$, UDG$(2, \lambda)$ has an infinite connected component.
- Previously, it was known that

$$0.7698 \leq \lambda_c(2) \leq 3.372.$$

  Lower bound due to Kong and Zeh (2008), upper bound due to Hall (1985).
- Upper bound improved to 1.568.

Critical value for NN$(2, k)$

- There is a finite value $k_c(2)$ s. t. for $k > k_c(2)$, NN$(2, k)$ has an infinite connected component (Häggström and Meester, 1996).
- Previously it was known that

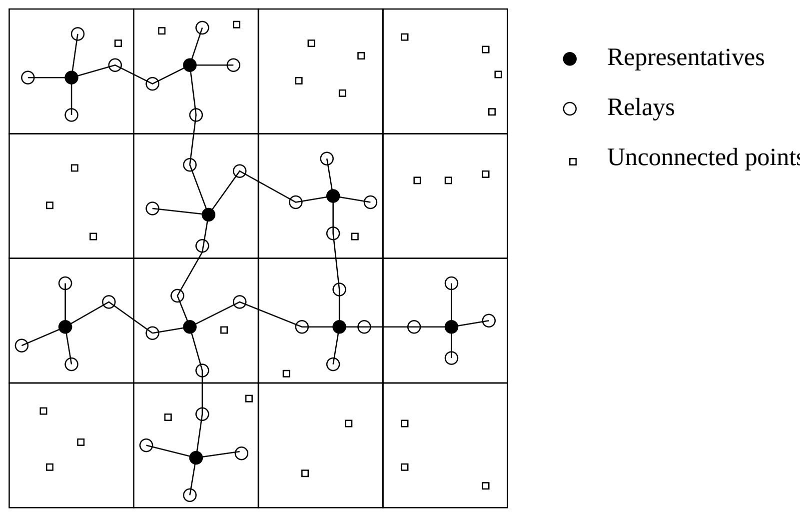
$$1 < k_c(2) < 213.$$

Lower bound due to Eppstein, Paterson and Yao (1997), upper bound due to Teng and Yao (2007).

- Upper bound improved to 188. (Subsequently improved to 11 by Balister and Bollobás).

Overview of our technique

We tile the space with square tiles and look for two kinds of points

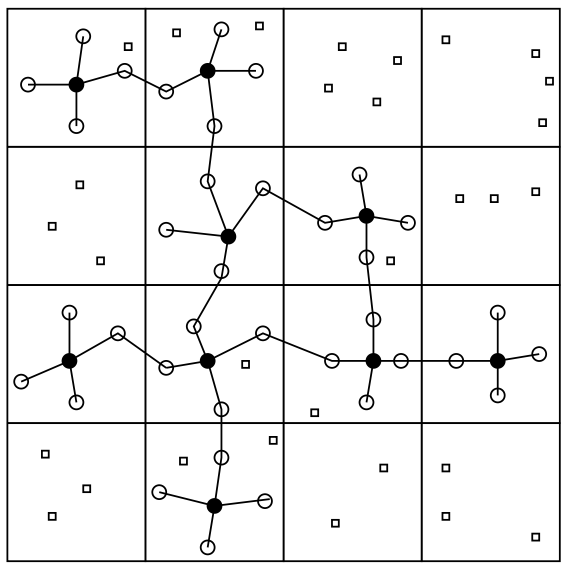


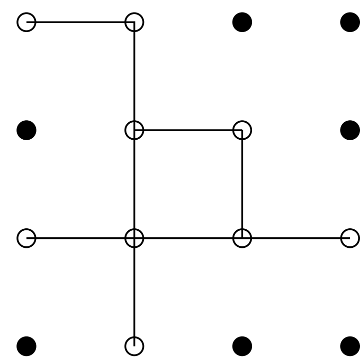
- *representative points* lie roughly at the centre of the tile.
- *relay points* help connect representative points.

We call a tile *good* if it contains both kinds of points.

Coupling with a process on $\mathbb{Z}^2$

We associate each tile in $\mathbb{R}^2$ with a point in $\mathbb{Z}^2$.





We declare a point in $\mathbb{Z}^2$ *open* (non-faulty) if the corresponding tile in $\mathbb{R}^2$ is good and *closed* (faulty) otherwise.

Site percolation in $\mathbb{Z}^2$

Setting. $\mathbb{L}^2$ is an infinite graph with vertex set $\mathbb{Z}^2$ and edges between points x and y such that $\|x - y\|_1 = 1$.

The stochastic process. Each point of $\mathbb{Z}^2$ is taken to be *open* with probability p and *closed* with probability $1 - p$. An edge is open if both its endpoints are open.

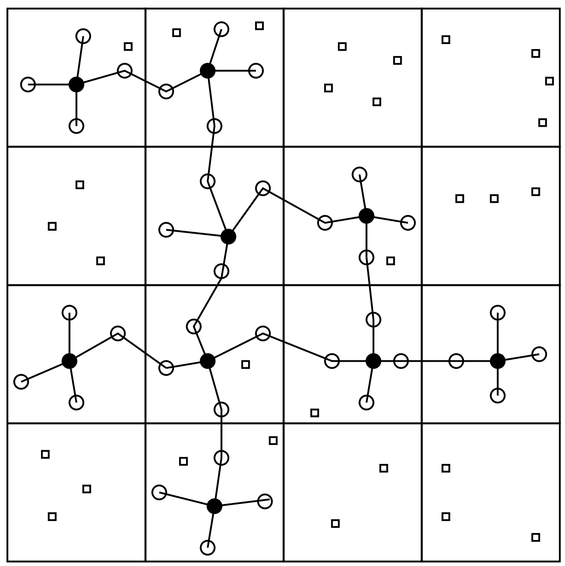
Lemma 1 *There is a p_c s.t. $0 < p_c < 1$ such that for $p > p_c$, $\mathbb{L}^2$ a.s. contains an infinite open cluster and for $p \leq p_c$, $\mathbb{L}^2$ a.s. does not contain an infinite cluster.*

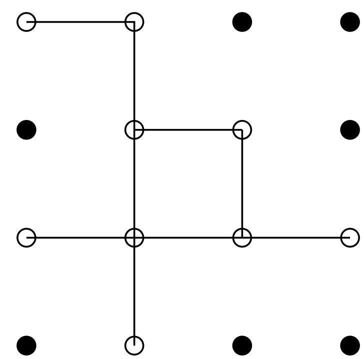
It is known that $p_c \approx 0.592...$

A basic property of the coupling

A path in $\mathbb{Z}^2 \Rightarrow$ A path between representative points in $\mathbb{R}^2$.

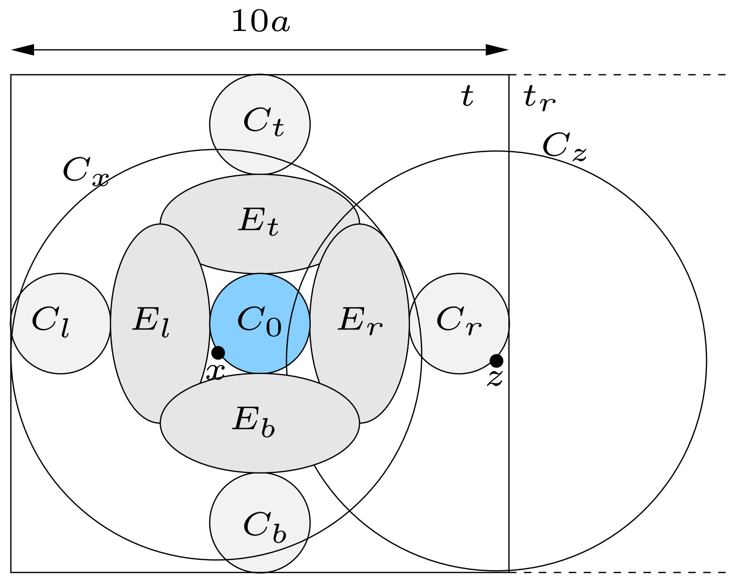
infinite open component in $\mathbb{Z}^2 \Rightarrow$ infinite component in the geometric random graph model.





$\Rightarrow$ if the probability of a tile being good exceeds p_c, the geometric random graph model a.s. has an infinite component.

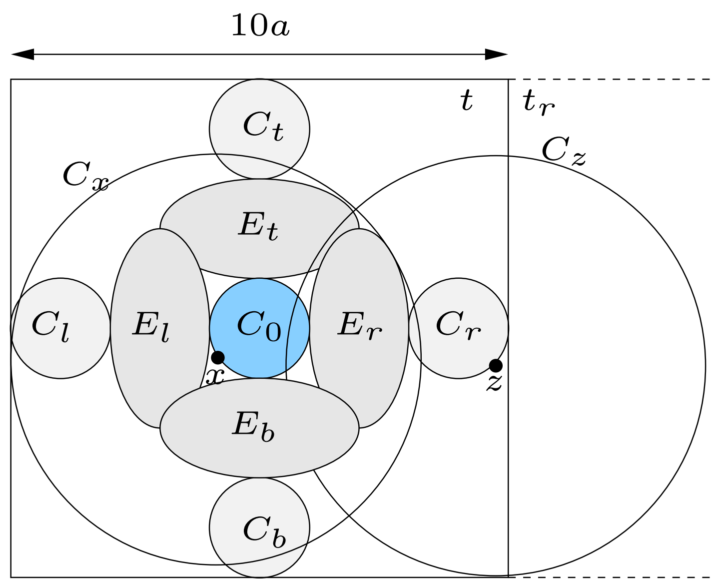
NN$(2, k)$: When is a tile good? Slide I



C_0, C_l, C_r, C_t, C_b are circles of radius a.

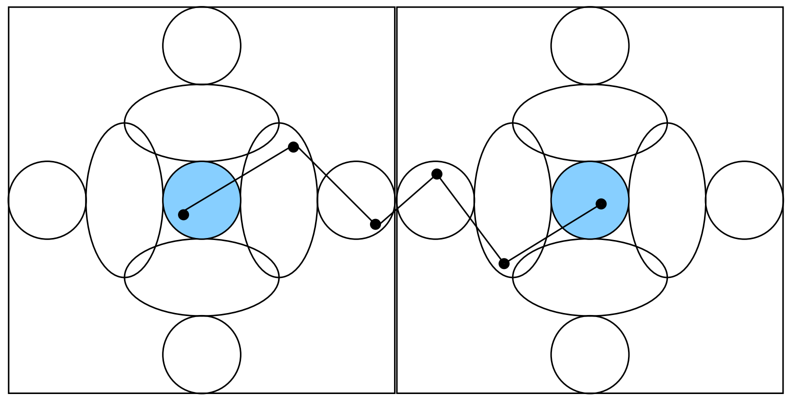
E_r: Consider the largest circle centred at any point in C_0 or C_r that lies wholly within the two tiles t and t_r. E_r is the locus of the points contained in all such circles.

NN$(2, k)$: When is a tile good? Slide II



1. the number of points inside t is at most $k/2$ and
2. the nine regions $C_0, C_r, C_t, C_l, C_b, E_r, E_t, E_l$ and E_b contain at least one point each.

$\mathbb{L}^2$ edges = paths in $\mathbf{NN}(2, k)$



An edge in $\mathbb{L}^2$ between two points x and y means

> There is a path between the representative points $\mathrm{rep}(\phi^{-1}(x))$ and $\mathrm{rep}(\phi^{-1}(y))$.

An upper bound for k_c

Theorem 2 *For* $NN(2, k)$,

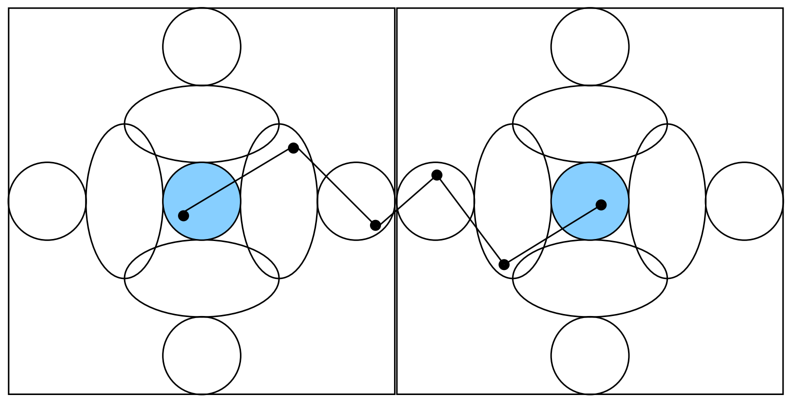
$$k_c(2) \leq 188.$$

Numerical calculations reveal that $k = 188$ is the smallest value for which the probability of a tile being good exceeds p_c for $\mathbb{L}^2$.

For all $k > k_2$ we call the infinite component NN-SENS$(2, k)$.

Constant stretch. Slide I

$\mathbb{L}^2$ edges = *short* paths in $\mathbf{NN}(2, k)$

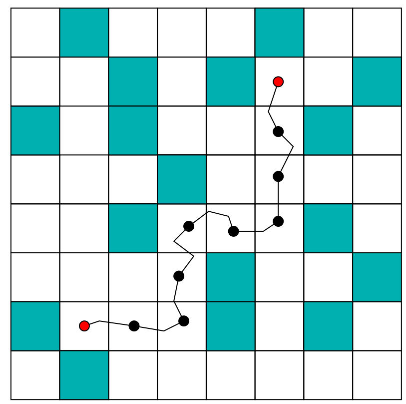


An edge in $\mathbb{L}^2$ between two points x and y means there is a constant c_k such that

$$d_k(\mathrm{rep}(\phi^{-1}(x)), \mathrm{rep}(\phi^{-1}(y))) \leq c_k \cdot d(\mathrm{rep}(\phi^{-1}(x)), \mathrm{rep}(\phi^{-1}(y))).$$

Constant stretch. Slide II

Short paths in the percolated $\mathbb{L}^2$

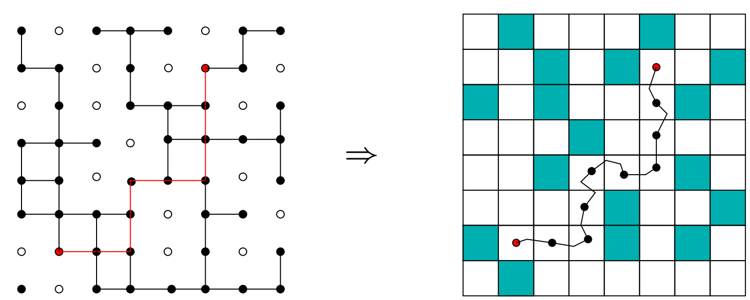


Lemma 3 *(Antal and Pisztora, 1996) For any $p > p_c$ and any x, y connected through an open path in a cube M^d of the infinite lattice. For some $\rho, c_2 > 0$ depending only on the dimension and p and for any $a > \rho \cdot D(x, y)$*

$$pr(D^p(x, y) > a)) < e^{-c_2 a}.$$

Constant stretch. Slide III

Our result



Theorem 4 *For NN-SENS*$(2,k)$*, with* $k \geq 188$ *there are constants* β *and* c_2 *depending only on* k *such that*

$$P(d_k(x,y) > \beta \cdot D(x,y)) < e^{-c_2 \cdot D(x,y)}.$$

Coverage

Theorem 5 *Let us consider a square region of size $\ell \times \ell$, call it $B(\ell)$. For $k \geq 188$ there are constants c_1, c_2 depending only on k and λ such that*

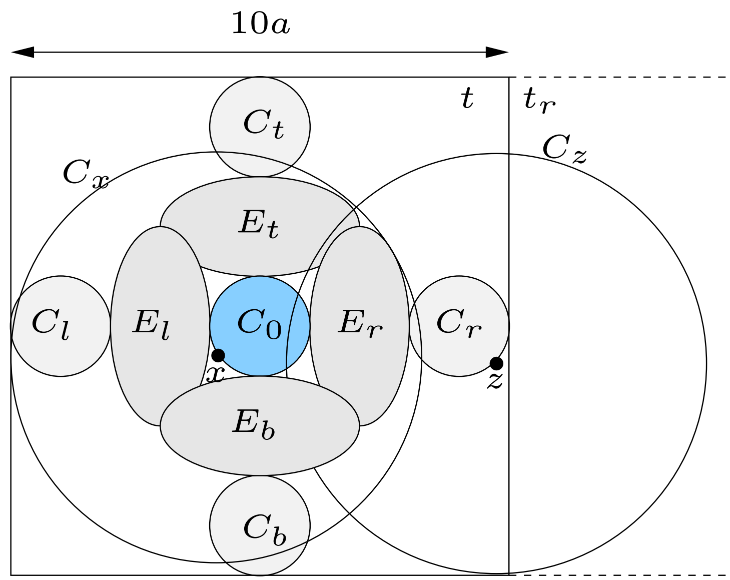
$$P[|B(\ell) \cap \mathit{NN\text{-}SENS}(2,k)| = 0] \leq c_1 \cdot \ell^2 \cdot e^{-c_2 \cdot \ell}.$$

Hence it follows that

Corollary 6 *There is a constant c_3 such that for $\ell \geq c_3 \log n$*

$$P[|B(\ell) \cap \mathit{NN\text{-}SENS}(2,k)| = 0] < \frac{1}{n}.$$

Algorithmic issues I: Constructing NN-SENS$(2, k)$



We begin with a tiling of $\mathbb{R}^2$

1. Each point uses location information to decide which of the 9 regions it is in, if any.
2. Leader election is used to identify one node within each region.
3. Nodes make connections with neighbouring leaders.

Algorithmic issues II: Routing

Representative points of a tile emulate open lattice points in $\mathbb{L}^2$. Any algorithm for routing in a percolated mesh can be used.

1. Try to follow the $x - y$ path between two vertices.
2. If the path is broken at some point, do a distributed BFS in order to find the next reachable vertex on that path.

Algorithm is due to Angel et. al. (2005) who show that the number of probes required to route a packet between two nodes n units apart is $O(n)$.

Conclusion of the case study

1. Similar results can be shown for UDG$(2, \lambda)$.
2. Geometric random graphs have properties well suited for sensor networks: sparsity, constant stretch, coverage and local computability.

Open question 1. Can all these properties be shown for all $k > k_c(2)$ and $\lambda > \lambda_c(2)$?

Open question 2. Can the value of $k_c(2)$ be brought down to somewhere near 3?

Research direction 1: More realistic models of transmission

- Noise and signal fading need to be taken into account.
- Current work along this direction either makes simplistic assumptions about the transmission model and solves a scheduling problem or vice versa.

Research direction 2: Secure routing

- We assume a model of passive eavesdropper presence in the region.
- Secret sharing over disjoint paths can be a basis for secure routing in the presence of eavesdroppers.

What eavesdropper density can a wireless network tolerate and still maintain reasonable secure throughput?

Research direction 3: Intruder detection

- A power-constrained sensor network monitoring a region conserves energy by putting sensors to sleep for periods of time.
- If we consider a random independent sleep schedule then the coverage region of the waking sensors at any point in time forms a random subregion.
- We evaluate the sleep schedule by studying how effective it is in detecting and tracking intruders.

References

1. P. Antal and A. Pisztora. On the chemical distance for supercritical Bernuolli percolation. *Ann. Probab.*, 24(2):1036–1048, 1996.

2. O. Angel, I. Benjamini, E. Ofek, and U. Wieder. Routing complexity of faulty networks. In Proc. PODC '05, pages 209–217, 2005.

3. Amitabha Bagchi. Sparse power-efficient topologies for wireless ad hoc sensor networks. arXiv:0805.4060v5 [cs.NI].

4. Amitabha Bagchi and Sohit Bansal. Nearest-neighbor graphs on random point sets and their applications to sensor networks. In Proc. PODC '08, page 434, 2008.

5. P. N. Balister and B. Bollobás. Percolation in the k-nearest neighbour graph. Submitted, 2009.

6. O. Häggström and R. Meester. Nearest neighbor and hard sphere models in continuum percolation. *Random Struct. Algor.*, 9(3):295–315, 1996.

7. P. Hall. On continuum percolation. *Ann. Probab.*, 13(4):1250–1266, 1985.

8. Z. Kong and E. M. Yeh. Connectivity and latency in large-scale wireless networks with unreliable links. In Proc. INFOCOM '08, pages 394–402, 2008.

9. X.-Y. Li, P.-J. Wan, and Y. Wang. Power efficient and sparse spanner for wireless ad hoc networks. In Proc. ICCCN '01, pages 564–567, 2001.

10. S.-H. Teng and F. F. Yao. k-nearest-neighbor clustering and percolation theory. *Algorithmica*, 49:192–211, 2007.