Random directed and on-line spatial graphs

Andrew Wade

Department of Mathematical Sciences

Durham University

June 2022

A partial survey, including some joint work with
Nicholas Georgiou, Jonathan Jordan, Mathew Penrose, Qasem Tawhari

Introduction: Spatial networks

- In real-world networks, nodes often carry spatial information relevant to network structure and/or properties, e.g., communications, transport, and geographical networks. The "space" can also be abstract rather than physical, e.g., social networks, etc.

Introduction: Spatial networks

- In real-world networks, nodes often carry spatial information relevant to network structure and/or properties, e.g., communications, transport, and geographical networks. The “space” can also be abstract rather than physical, e.g., social networks, etc.

- It is natural to study simple models of random spatial networks, for inference or prediction, for insight into typical behaviour, or for assessing performance of algorithms or processes that take place on networks.

Introduction: Spatial networks

- In real-world networks, nodes often carry spatial information relevant to network structure and/or properties, e.g., communications, transport, and geographical networks. The "space" can also be abstract rather than physical, e.g., social networks, etc.
- It is natural to study simple models of random spatial networks, for inference or prediction, for insight into typical behaviour, or for assessing performance of algorithms or processes that take place on networks.
- In this talk I will survey some results on some networks constructed on random points in Euclidean space with a connectivity rule that incorporates proximity and some ordering constraint.

Total length asymptotics over uniform random points

- In this talk I will consider graphs with n vertices, labelled $1, 2, \ldots, n$, each associated with a random spatial location in the d-dimensional unit cube $[0, 1]^d$.

Total length asymptotics over uniform random points

- In this talk I will consider graphs with n vertices, labelled $1, 2, \ldots, n$, each associated with a random spatial location in the d-dimensional unit cube $[0, 1]^d$.
- Vertex i has location $U_i \in [0, 1]^d$. The random variables $U_1, U_2, \ldots, U_n$ are independent and uniformly distributed on the cube.

Total length asymptotics over uniform random points

- In this talk I will consider graphs with n vertices, labelled $1, 2, \ldots, n$, each associated with a random spatial location in the d-dimensional unit cube $[0,1]^d$.
- Vertex i has location $U_i \in [0,1]^d$. The random variables $U_1, U_2, \ldots, U_n$ are independent and uniformly distributed on the cube.
- There will be some rule for inserting (directed) edges. Edge from vertex i to j has an associated (Euclidean) length $\|U_i - U_j\|$.

Total length asymptotics over uniform random points

- In this talk I will consider graphs with n vertices, labelled $1, 2, \ldots, n$, each associated with a random spatial location in the d-dimensional unit cube $[0, 1]^d$.
- Vertex i has location $U_i \in [0, 1]^d$. The random variables $U_1, U_2, \ldots, U_n$ are independent and uniformly distributed on the cube.
- There will be some rule for inserting (directed) edges. Edge from vertex i to j has an associated (Euclidean) length $\|U_i - U_j\|$.
- The main quantity of interest (here) is the total edge length of the graph: $\mathcal{L}_n := \sum_{\text{edges } (i,j)} \|U_i - U_j\|$ (a random variable, whose distribution depends on n and whatever the rule for edges is).

Total length asymptotics over uniform random points

- In this talk I will consider graphs with n vertices, labelled $1, 2, \ldots, n$, each associated with a random spatial location in the d-dimensional unit cube $[0, 1]^d$.
- Vertex i has location $U_i \in [0, 1]^d$. The random variables $U_1, U_2, \ldots, U_n$ are independent and uniformly distributed on the cube.
- There will be some rule for inserting (directed) edges. Edge from vertex i to j has an associated (Euclidean) length $\|U_i - U_j\|$.
- The main quantity of interest (here) is the total edge length of the graph: $\mathcal{L}_n := \sum_{\text{edges } (i,j)} \|U_i - U_j\|$ (a random variable, whose distribution depends on n and whatever the rule for edges is).
- More generally, α-weighted total length: $\sum_{\text{edges } (i,j)} \|U_i - U_j\|^\alpha$, for a fixed parameter $\alpha > 0$.

Total length asymptotics over uniform random points

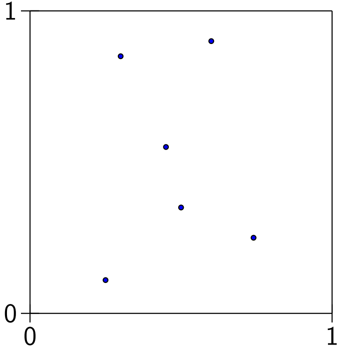
- In this talk I will consider graphs with n vertices, labelled $1, 2, \ldots, n$, each associated with a random spatial location in the d-dimensional unit cube $[0, 1]^d$.
- Vertex i has location $U_i \in [0, 1]^d$. The random variables $U_1, U_2, \ldots, U_n$ are independent and uniformly distributed on the cube.
- There will be some rule for inserting (directed) edges. Edge from vertex i to j has an associated (Euclidean) length $\|U_i - U_j\|$.
- The main quantity of interest (here) is the total edge length of the graph: $\mathcal{L}_n := \sum_{\text{edges } (i,j)} \|U_i - U_j\|$ (a random variable, whose distribution depends on n and whatever the rule for edges is).
- More generally, α-weighted total length: $\sum_{\text{edges } (i,j)} \|U_i - U_j\|^\alpha$, for a fixed parameter $\alpha > 0$.
- Main question: What can we say about the (asymptotic) distribution of $\mathcal{L}_n$ in the large sample limit $n \to \infty$?

Outline

Outline

Nearest-neighbour graph

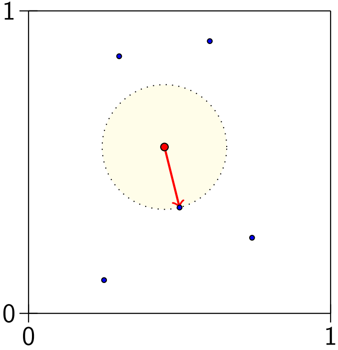
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



Points are distributed uniformly at random.

Nearest-neighbour graph

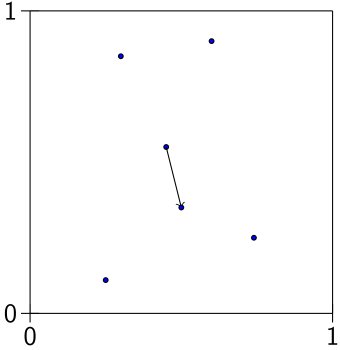
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

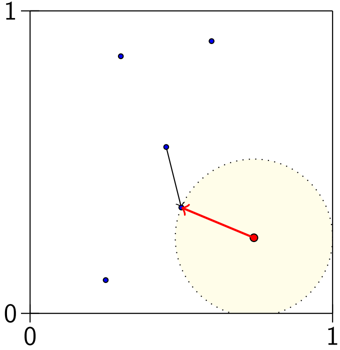
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

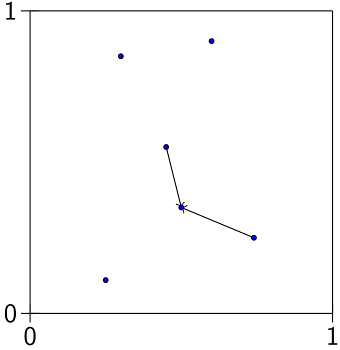
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

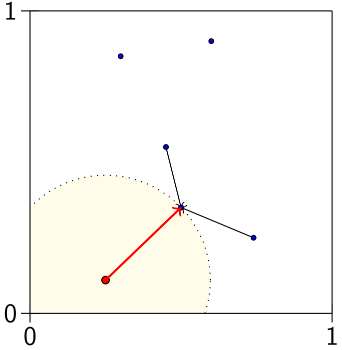
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

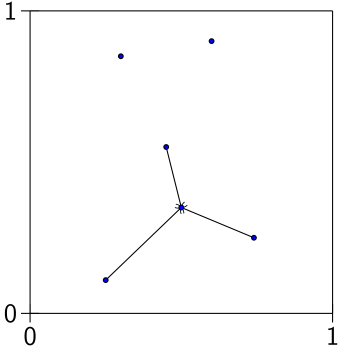
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

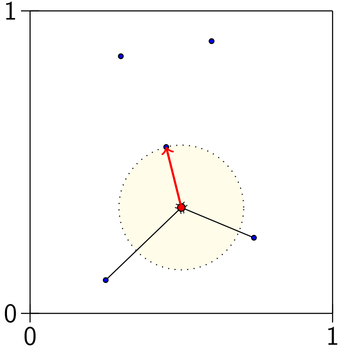
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

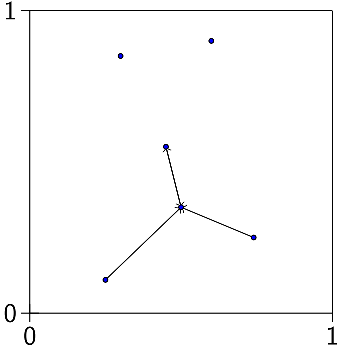
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

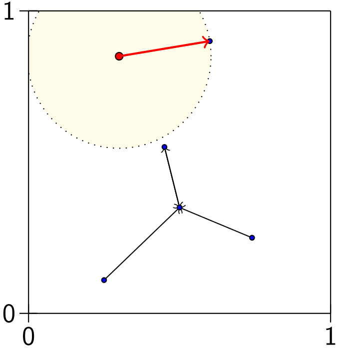
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

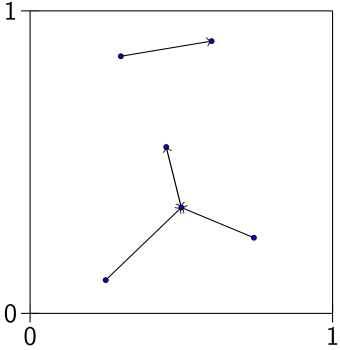
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

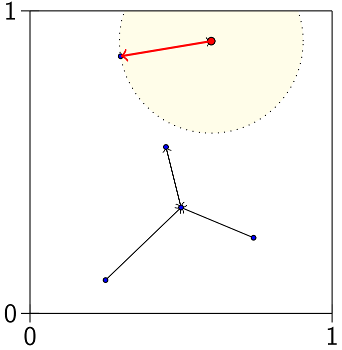
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph

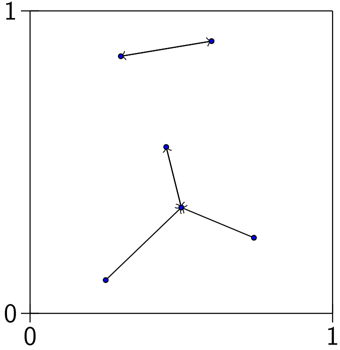
Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

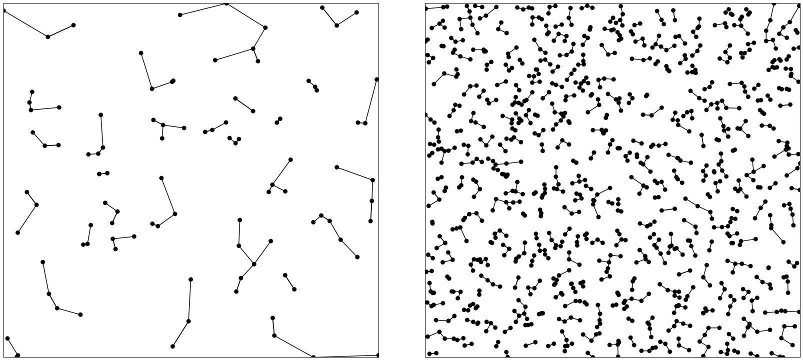
Nearest-neighbour graph

Vertices $1, \ldots, n$ are given independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its nearest neighbour, i.e., the closest (Euclidean distance) other vertex. (Ties have probability 0.) Here is a picture for $d = 2$, $n = 6$:



For each point, identify its nearest neighbour, and add an edge.

Nearest-neighbour graph



Simulations of nearest-neighbour graph (NNG) in $[0,1]^2$, on $n = 100$ points (*left*) and $n = 1000$ points (*right*).

Nearest-neighbour graph

Nearest-neighbour graphs and distances have numerous applications in statistics, machine learning, computational geometry, ecology (1950s–), etc., for example in classification and clustering (1970s–).

Nearest-neighbour graph

Nearest-neighbour graphs and distances have numerous applications in statistics, machine learning, computational geometry, ecology (1950s–), etc., for example in classification and clustering (1970s–).

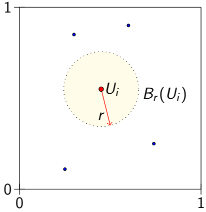
We consider the total power-weighted edge length

$$\mathcal{N}_n^{d,\alpha} := \sum_{i=1}^{n} \min_{1 \le j \le n,\, j \neq i} \|U_j - U_i\|^{\alpha},$$

where $\alpha > 0$ (e.g., $\alpha = 1$ is the total edge length).

The limit theory of $\mathcal{N}_n^{d,\alpha}$ was considered by Bickel & Breiman (1983), Avram & Bertsimas (1993), and Penrose & Yukich (2001), among others.

Nearest-neighbour graph: Heuristics

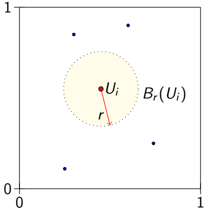


What is the n-dependence in $\mathcal{N}_n^{d,\alpha}$? Say in $\mathbb{E}\mathcal{N}_n^{d,\alpha}$?

Consider vertex $i \in \{1, \ldots, n\}$ at $U_i \in [0,1]^d$.

What is the distribution of its nearest-neighbour distance $D_{n,i} := \min_{j \neq i} \|U_j - U_i\|$?

Nearest-neighbour graph: Heuristics



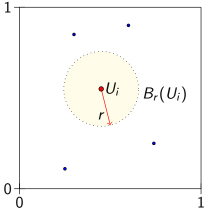
What is the n-dependence in $\mathcal{N}_n^{d,\alpha}$? Say in $\mathbb{E}\mathcal{N}_n^{d,\alpha}$?

Consider vertex $i \in \{1, \dots, n\}$ at $U_i \in [0,1]^d$.

What is the distribution of its nearest-neighbour distance $D_{n,i} := \min_{j \neq i} \|U_j - U_i\|$?

Observe that $D_{n,i} > r$ if and only if all of U_j, $j \neq i$, lie in $[0,1]^d \setminus B_r(U_i)$, where $B_r(x) := \{y \in \mathbb{R}^d : \|y - x\| \leq r\}$.

Nearest-neighbour graph: Heuristics



What is the n-dependence in $\mathcal{N}_n^{d,\alpha}$? Say in $\mathbb{E}\mathcal{N}_n^{d,\alpha}$?

Consider vertex $i \in \{1, \ldots, n\}$ at $U_i \in [0,1]^d$.

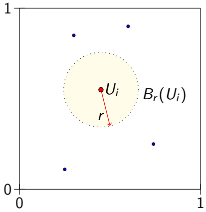
What is the distribution of its nearest-neighbour distance $D_{n,i} := \min_{j \neq i} \|U_j - U_i\|$?

Observe that $D_{n,i} > r$ if and only if all of U_j, $j \neq i$, lie in $[0,1]^d \setminus B_r(U_i)$, where $B_r(x) := \{y \in \mathbb{R}^d : \|y - x\| \leq r\}$.

So, at least for $r < 1/2$, say,

$$\mathbb{P}\left(D_{n,i} > r\right) \asymp \left(1 - cr^d\right)^n .$$

Nearest-neighbour graph: Heuristics



What is the n-dependence in $\mathcal{N}_n^{d,\alpha}$? Say in $\mathbb{E}\mathcal{N}_n^{d,\alpha}$?

Consider vertex $i \in \{1, \ldots, n\}$ at $U_i \in [0,1]^d$.

What is the distribution of its nearest-neighbour distance $D_{n,i} := \min_{j \neq i} \|U_j - U_i\|$?

Observe that $D_{n,i} > r$ if and only if all of U_j, $j \neq i$, lie in $[0,1]^d \setminus B_r(U_i)$, where $B_r(x) := \{y \in \mathbb{R}^d : \|y - x\| \leq r\}$.

So, at least for $r < 1/2$, say,

$$\mathbb{P}(D_{n,i} > r) \asymp \left(1 - cr^d\right)^n .$$

From here it is not hard to see that $\operatorname{med}(D_{n,i}) \asymp n^{-1/d}$, and indeed

$$\mathbb{E}\, D_{n,i}^{\alpha} \asymp n^{-\alpha/d}, \text{ and } \mathbb{V}\mathrm{ar}\, D_{n,i}^{\alpha} \asymp n^{-2\alpha/d}.$$

Nearest-neighbour graph: Heuristics

We claimed $\mathbb{E}\, D_{n,i}^{\alpha} \asymp n^{-\alpha/d}$, and $\mathbb{V}\mathrm{ar}\, D_{n,i}^{\alpha} \asymp n^{-2\alpha/d}$.

The total α-weighted length is $\mathcal{N}_n^{d,\alpha} = \sum_{i=1}^{n} D_{n,i}^{\alpha}$.

Nearest-neighbour graph: Heuristics

We claimed $\mathbb{E}\, D_{n,i}^{\alpha} \asymp n^{-\alpha/d}$, and $\mathbb{V}\mathrm{ar}\, D_{n,i}^{\alpha} \asymp n^{-2\alpha/d}$.

The total α-weighted length is $\mathcal{N}_n^{d,\alpha} = \sum_{i=1}^{n} D_{n,i}^{\alpha}$.

However, the $D_{n,i}$ are not independent, so this is not a sum of i.i.d. random variables. The dependence is in some sense only local: $D_{n,i}$ and $D_{n,j}$ are close to independent if U_i and U_j are far apart.

Nearest-neighbour graph: Heuristics

We claimed $\mathbb{E}\, D_{n,i}^{\alpha} \asymp n^{-\alpha/d}$, and $\mathbb{V}\mathrm{ar}\, D_{n,i}^{\alpha} \asymp n^{-2\alpha/d}$.

The total α-weighted length is $\mathcal{N}_n^{d,\alpha} = \sum_{i=1}^{n} D_{n,i}^{\alpha}$.

However, the $D_{n,i}$ are not independent, so this is not a sum of i.i.d. random variables. The dependence is in some sense only local: $D_{n,i}$ and $D_{n,j}$ are close to independent if U_i and U_j are far apart.

An appropriate formulation of this local dependence allows one to obtain laws of large numbers and central limit theorems, involving scale factors $\mathbb{E}\,\mathcal{N}_n^{d,\alpha} \asymp n^{1-(\alpha/d)}$ and $\mathbb{V}\mathrm{ar}\,\mathcal{N}_n^{d,\alpha} \asymp n^{1-(2\alpha/d)}$.

Nearest-neighbour graph: Heuristics

We claimed $\mathbb{E}\, D_{n,i}^{\alpha} \asymp n^{-\alpha/d}$, and $\mathbb{V}\mathrm{ar}\, D_{n,i}^{\alpha} \asymp n^{-2\alpha/d}$.

The total α-weighted length is $\mathcal{N}_n^{d,\alpha} = \sum_{i=1}^{n} D_{n,i}^{\alpha}$.

However, the $D_{n,i}$ are not independent, so this is not a sum of i.i.d. random variables. The dependence is in some sense only local: $D_{n,i}$ and $D_{n,j}$ are close to independent if U_i and U_j are far apart.

An appropriate formulation of this local dependence allows one to obtain laws of large numbers and central limit theorems, involving scale factors $\mathbb{E}\, \mathcal{N}_n^{d,\alpha} \asymp n^{1-(\alpha/d)}$ and $\mathbb{V}\mathrm{ar}\, \mathcal{N}_n^{d,\alpha} \asymp n^{1-(2\alpha/d)}$.

Now an extensive theory of stabilization to obtain such results for spatial random graphs with local dependence: Aldous & Steele (1992), Avram & Bertsimas (1993), Kesten & Lee (1996), Penrose & Yukich (2001–), many others.

Nearest-neighbour graph: Limit theory

Theorem (Bickel & Breiman 1983, Avram & Bertsimas 1993, Penrose & Yukich 2001, etc.)

For $d \in \mathbb{N}$ and $\alpha > 0$, as $n \to \infty$,

$$n^{(\alpha/d)-1} \mathcal{N}_n^{d,\alpha} \to c_{d,\alpha} := \pi^{-\alpha/2} \Gamma(1 + \tfrac{\alpha}{d}) \Gamma(1 + \tfrac{d}{2})^{\alpha/d}, \text{ in } L^1.$$

Moreover, as $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{N}^{d,\alpha} - \mathbb{E}\mathcal{N}^{d,\alpha} \right) \xrightarrow{d} Z,$$

where Z has a mean zero, finite variance normal distribution.*

* Variance depends on d and α; explicit values known only for $d = 1$ (Penrose & Wade, 2008).

Nearest-neighbour graph: Limit theory

Theorem (Bickel & Breiman 1983, Avram & Bertsimas 1993, Penrose & Yukich 2001, etc.)

For $d \in \mathbb{N}$ and $\alpha > 0$, as $n \to \infty$,

$$n^{(\alpha/d)-1}\mathcal{N}_n^{d,\alpha} \to c_{d,\alpha} := \pi^{-\alpha/2}\Gamma(1+\tfrac{\alpha}{d})\Gamma(1+\tfrac{d}{2})^{\alpha/d}, \text{ in } L^1.$$

Moreover, as $n \to \infty$,

$$n^{(\alpha/d)-(1/2)}\left(\mathcal{N}^{d,\alpha} - \mathbb{E}\mathcal{N}^{d,\alpha}\right) \xrightarrow{d} Z,$$

where Z has a mean zero, finite variance normal distribution.*

Example. When $d = 2$, $\alpha = 1$, we have that the total edge length of the nearest-neighbour graph on $[0,1]^2$ satisfies

$$\mathbb{E}\mathcal{N}^{2,1} \sim \frac{\sqrt{n}}{2}, \text{ and } \mathcal{N}^{2,1} - \mathbb{E}\mathcal{N}^{2,1} \xrightarrow{d} \text{non-degenerate normal.}$$

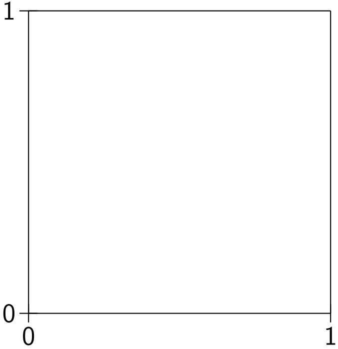
---

* Variance depends on d and α; explicit values known only for $d = 1$ (Penrose & Wade, 2008).

Outline

On-line nearest-neighbour graph

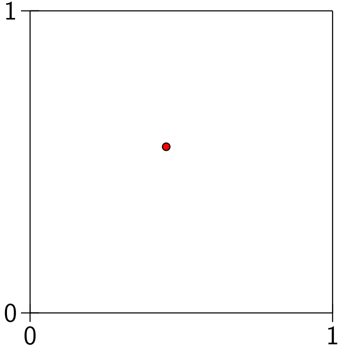
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0, 1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n = 0$: empty graph!

On-line nearest-neighbour graph

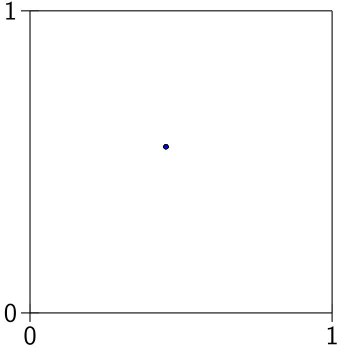
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n = 1$: first vertex arrives at a uniform random location.

On-line nearest-neighbour graph

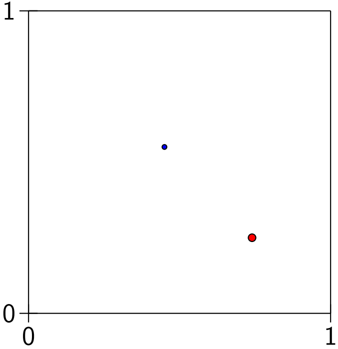
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0, 1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n = 1$: first vertex has no neighbours, so no edge.

On-line nearest-neighbour graph

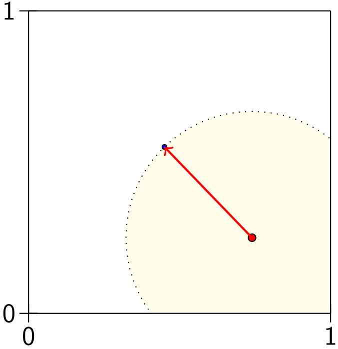
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n = 2$: next vertex arrives.

On-line nearest-neighbour graph

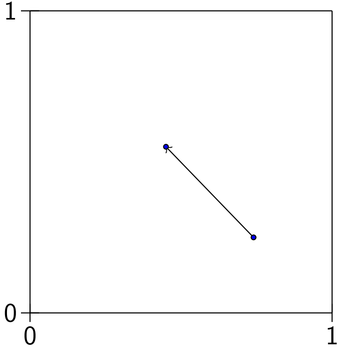
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n = 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

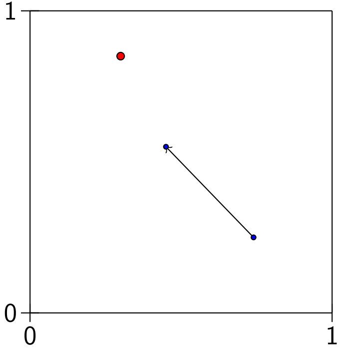
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

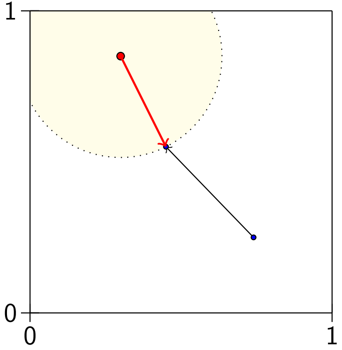
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

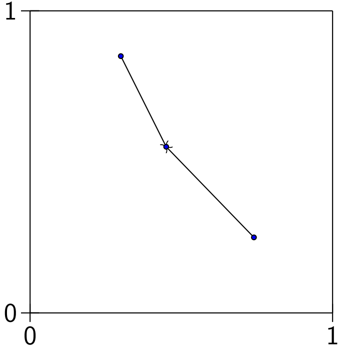
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0, 1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

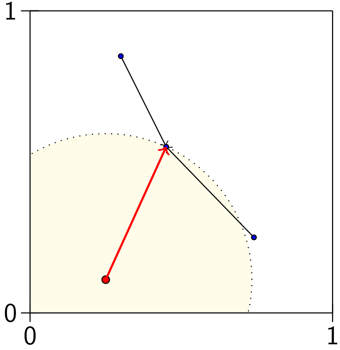
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0, 1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

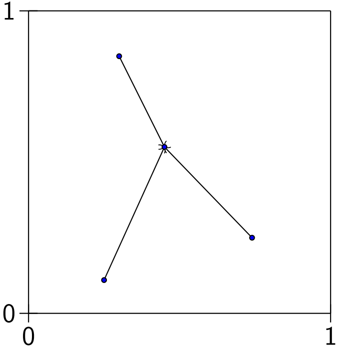
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

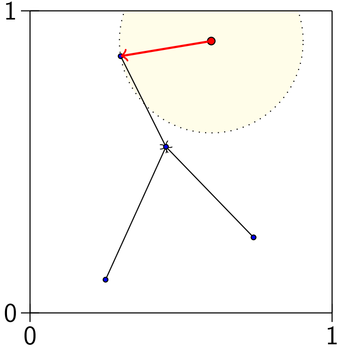
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

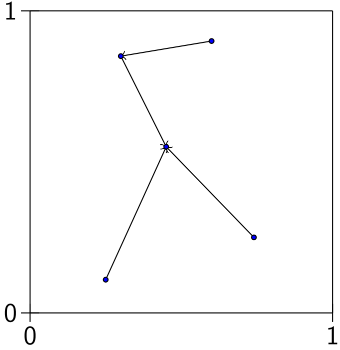
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

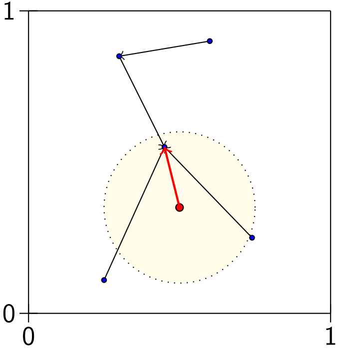
Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0, 1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

On-line nearest-neighbour graph

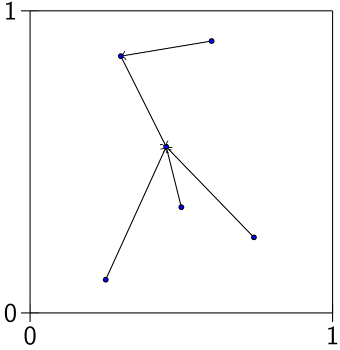
Vertices $1, \dots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \dots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



$n \geq 2$: new vertex joins to nearest existing vertex.

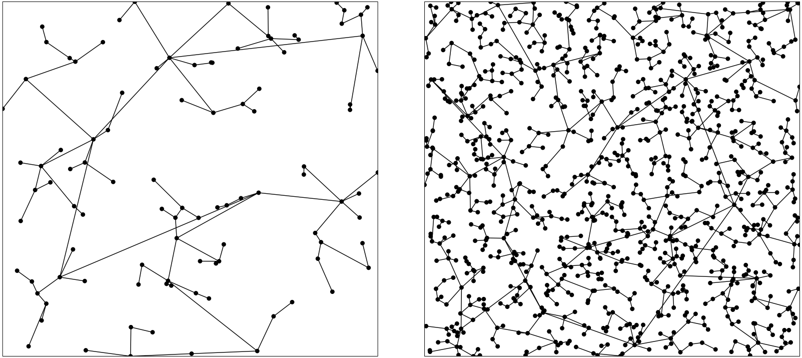
On-line nearest-neighbour graph

Vertices $1, \ldots, n$ arrive sequentially, with independent, uniformly random locations $U_1, \ldots, U_n$ in the unit d-cube $[0,1]^d$. Each vertex after the first is joined by an edge to its nearest predecessor (Euclidean distance). Here is a picture for $d = 2$:



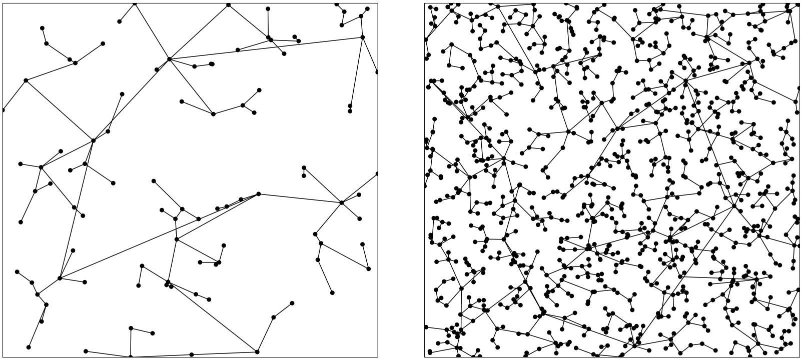
NB. Graph is a tree; different order will give different graph.

On-line nearest-neighbour graph



Simulations of on-line nearest-neighbour graph (ONG) in $[0,1]^2$, on $n = 100$ points (*left*) and $n = 1000$ points (*right*).

On-line nearest-neighbour graph



Simulations of on-line nearest-neighbour graph (ONG) in $[0,1]^2$, on $n = 100$ points (*left*) and $n = 1000$ points (*right*).

Note the ONG is more inhomogeneous than the ordinary NNG: Old vertices tend to be more highly connected; old edges tend to be longer.

On-line nearest-neighbour graph

Although it is a natural model of a time-evolving spatial network, the earliest appearance of the on-line nearest-neighbour graph (ONG) that I am aware of is in Steele (1989). It also appears as a limiting case of network models of Fabrikant *et al.* (2002), Manna & Sen (2002) and Flaxman *et al.* (2006). The name "on-line nearest-neighbour graph" is due to Penrose (2005).

On-line nearest-neighbour graph

Although it is a natural model of a time-evolving spatial network, the earliest appearance of the on-line nearest-neighbour graph (ONG) that I am aware of is in Steele (1989). It also appears as a limiting case of network models of Fabrikant *et al.* (2002), Manna & Sen (2002) and Flaxman *et al.* (2006). The name "on-line nearest-neighbour graph" is due to Penrose (2005).

Some graph-theoretic properties of the ONG were studied by Berger *et al.* (2007) and Jordan & Wade (2015).

On-line nearest-neighbour graph

Although it is a natural model of a time-evolving spatial network, the earliest appearance of the on-line nearest-neighbour graph (ONG) that I am aware of is in Steele (1989). It also appears as a limiting case of network models of Fabrikant *et al.* (2002), Manna & Sen (2002) and Flaxman *et al.* (2006). The name "on-line nearest-neighbour graph" is due to Penrose (2005).

Some graph-theoretic properties of the ONG were studied by Berger *et al.* (2007) and Jordan & Wade (2015).

We consider the total power-weighted edge length

$$\mathcal{O}_n^{d,\alpha} := \sum_{i=2}^{n} \min_{1 \le j < i} \| U_j - U_i \|^{\alpha},$$

as studied by Penrose (2005), Penrose & Wade (2008), and Wade (2007–9).

On-line nearest-neighbour graph

Consider the total power-weighted edge length

$$\mathcal{O}_n^{d,\alpha} := \sum_{i=2}^{n} \min_{1 \le j < i} \| U_j - U_i \|^\alpha = \sum_{i=2}^{n} D_{i,i}^\alpha,$$

where, as before, $D_{i,i}$ = distance from U_i to nearest $\{U_1, \ldots, U_{i-1}\}$.

On-line nearest-neighbour graph

Consider the total power-weighted edge length

$$\mathcal{O}_n^{d,\alpha} := \sum_{i=2}^{n} \min_{1 \le j < i} \|U_j - U_i\|^\alpha = \sum_{i=2}^{n} D_{i,i}^\alpha,$$

where, as before, $D_{i,i}$ = distance from U_i to nearest $\{U_1, \ldots, U_{i-1}\}$.

As in the heuristic for the nearest-neighbour graph, $\mathbb{E}\, D_{i,i}^\alpha \asymp i^{-\alpha/d}$ and $\mathbb{V}\mathrm{ar}\, D_{i,i}^\alpha \asymp i^{-2\alpha/d}$, so

$$\mathbb{E}\, \mathcal{O}_n^{d,\alpha} \begin{cases} \to \text{const.} & \text{if } \alpha > d, \\ \asymp \log n & \text{if } \alpha = d, \\ \asymp n^{1-(\alpha/d)} & \text{if } \alpha < d. \end{cases}$$

Moreover, assuming local dependence, one might guess

$$\mathbb{V}\mathrm{ar}\, \mathcal{O}_n^{d,\alpha} \begin{cases} \to \text{const.} & \text{if } \alpha > d/2, \\ \asymp \log n & \text{if } \alpha = d/2, \\ \asymp n^{1-(2\alpha/d)} & \text{if } \alpha < d/2. \end{cases}$$

On-line nearest-neighbour graph: Law of large numbers

Theorem (Wade 2007, Penrose & Wade 2008)

Let $d \in \mathbb{N}$ and $\alpha > 0$. Then, as $n \to \infty$,

$$n^{(\alpha/d)-1}\mathcal{O}_n^{d,\alpha} \to c'_{d,\alpha}, \text{ in } L^1, \text{ if } 0 < \alpha < d.$$

On the other hand for $\alpha = d$ we have $\mathbb{E}\,\mathcal{O}_n^{d,d} \sim \pi^{-d/2}\Gamma(1+(d/2))\log n$, and, for $\alpha > d$ we have $\mathcal{O}_n^{d,\alpha} \to W^{d,\alpha}$ in L^2, where $W^{d,\alpha}$ is an $\mathbb{R}_+$-valued random variable with finite variance (not Gaussian).

Here

$$c'_{d,\alpha} := \frac{d}{d-\alpha} c_{d,\alpha} \text{ for } \alpha \in (0,d).$$

For example, $c'_{2,1} = 2c_{2,1} = 1$.

On-line nearest-neighbour graph: Law of large numbers

Theorem (Wade 2007, Penrose & Wade 2008)

Let $d \in \mathbb{N}$ and $\alpha > 0$. Then, as $n \to \infty$,

$$n^{(\alpha/d)-1}\mathcal{O}_n^{d,\alpha} \to c'_{d,\alpha}, \text{ in } L^1, \text{ if } 0 < \alpha < d.$$

On the other hand for $\alpha = d$ we have $\mathbb{E}\,\mathcal{O}_n^{d,d} \sim \pi^{-d/2}\Gamma(1+(d/2))\log n$, and, for $\alpha > d$ we have $\mathcal{O}_n^{d,\alpha} \to W^{d,\alpha}$ in L^2, where $W^{d,\alpha}$ is an $\mathbb{R}_+$-valued random variable with finite variance (not Gaussian).

Intuition: Increasing α increases the relative importance of long ($\approx$ early) edges. For example, if $\alpha > d/2$ then the very first edge contributes a positive fraction of the total variance (order 1).

On-line nearest-neighbour graph: Large α

Theorem (Penrose & Wade 2008, Wade 2009)

Let $d \in \mathbb{N}$ and $\alpha > d/2$. Then, as $n \to \infty$,

$$\mathcal{O}_n^{d,\alpha} - \mathbb{E}\,\mathcal{O}_n^{d,\alpha} \to Q_{d,\alpha}, \text{ in } L^2,$$

where $Q_{d,\alpha}$ is a mean-0 random variable.

On-line nearest-neighbour graph: Large α

Theorem (Penrose & Wade 2008, Wade 2009)

Let $d \in \mathbb{N}$ and $\alpha > d/2$. Then, as $n \to \infty$,

$$\mathcal{O}_n^{d,\alpha} - \mathbb{E}\,\mathcal{O}_n^{d,\alpha} \to Q_{d,\alpha}, \text{ in } L^2,$$

where $Q_{d,\alpha}$ is a mean-0 random variable.

- Note if $\alpha \in (d/2, d]$ one has $\mathbb{E}\,\mathcal{O}_n^{d,\alpha} \to \infty$, so the centering is non-trivial.

On-line nearest-neighbour graph: Large α

Theorem (Penrose & Wade 2008, Wade 2009)

Let $d \in \mathbb{N}$ and $\alpha > d/2$. Then, as $n \to \infty$,

$$\mathcal{O}_n^{d,\alpha} - \mathbb{E}\,\mathcal{O}_n^{d,\alpha} \to Q_{d,\alpha}, \text{ in } L^2,$$

where $Q_{d,\alpha}$ is a mean-0 random variable.

- Note if $\alpha \in (d/2, d]$ one has $\mathbb{E}\,\mathcal{O}_n^{d,\alpha} \to \infty$, so the centering is non-trivial.
- We know $Q_{d,\alpha}$ is non-Gaussian for $d = 1$, $\alpha > 1/2$ (when it is characterized by a distributional fixed-point equation), and when $d \geq 2$, $\alpha > d$ (when $Q_{d,\alpha} = W_{d,\alpha} - \mathbb{E}\,W_{d,\alpha}$ from above). Conjecture is that $Q_{d,\alpha}$ is non-Gaussian for all $\alpha > d/2$.

On-line nearest-neighbour graph: Small α

Conjecture (Penrose 2005, Wade 2009)

Let $d \in \mathbb{N}$ *and* $\alpha \in (0, d/2)$. *Then, as* $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{O}_n^{d,\alpha} - \mathbb{E}\, \mathcal{O}_n^{d,\alpha} \right) \xrightarrow{d} \text{non-degenerate normal.}$$

On-line nearest-neighbour graph: Small α

Conjecture (Penrose 2005, Wade 2009)

Let $d \in \mathbb{N}$ and $\alpha \in (0, d/2)$. Then, as $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{O}_n^{d,\alpha} - \mathbb{E}\, \mathcal{O}_n^{d,\alpha} \right) \xrightarrow{d} \text{non-degenerate normal.}$$

Theorem (Penrose 2005)

This conjecture holds for $\alpha \in (0, d/4)$.

On-line nearest-neighbour graph: Small α

Conjecture (Penrose 2005, Wade 2009)

Let $d \in \mathbb{N}$ and $\alpha \in (0, d/2)$. Then, as $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{O}_n^{d,\alpha} - \mathbb{E}\, \mathcal{O}_n^{d,\alpha} \right) \xrightarrow{d} \text{non-degenerate normal.}$$

Theorem (Penrose 2005)

This conjecture holds for $\alpha \in (0, d/4)$.

Example. For the case of the total length ($\alpha = 1$) we have that the central limit theorem holds for $d \geq 5$; it is conjectured for $d \geq 3$. When $d = 1$, we have $\mathcal{O}_n^{1,1} - \mathbb{E}\, \mathcal{O}_n^{1,1} \to Q_{1,1}$, non-Gaussian distribution characterized by a fixed point.

On-line nearest-neighbour graph: Small α

Conjecture (Penrose 2005, Wade 2009)

Let $d \in \mathbb{N}$ and $\alpha \in (0, d/2)$. Then, as $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{O}_n^{d,\alpha} - \mathbb{E}\, \mathcal{O}_n^{d,\alpha} \right) \xrightarrow{d} \text{non-degenerate normal.}$$

Theorem (Penrose 2005)

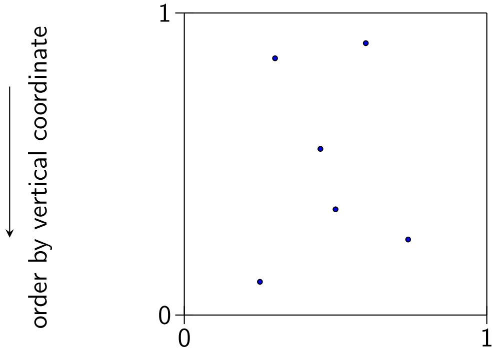
This conjecture holds for $\alpha \in (0, d/4)$.

Example. For the case of the total length ($\alpha = 1$) we have that the central limit theorem holds for $d \geq 5$; it is conjectured for $d \geq 3$. When $d = 1$, we have $\mathcal{O}_n^{1,1} - \mathbb{E}\, \mathcal{O}_n^{1,1} \to Q_{1,1}$, non-Gaussian distribution characterized by a fixed point. In the critical case $\alpha = d/2$ one also anticipates a central limit theorem, with a logarithmic variance scaling.

Outline

Minimal directed spanning tree

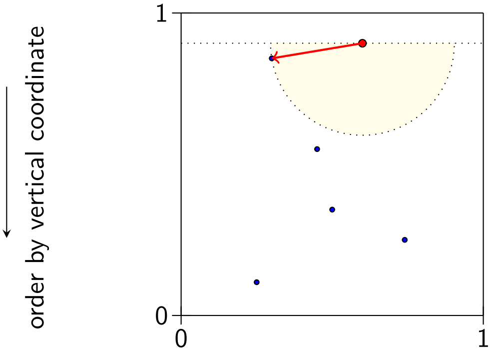
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0, 1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Points are distributed uniformly at random.

Minimal directed spanning tree

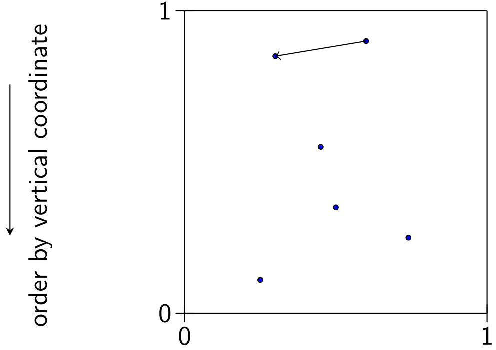
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

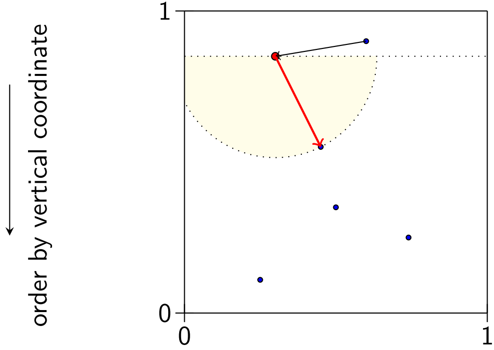
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

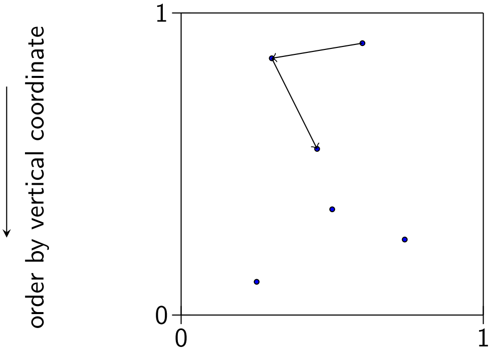
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

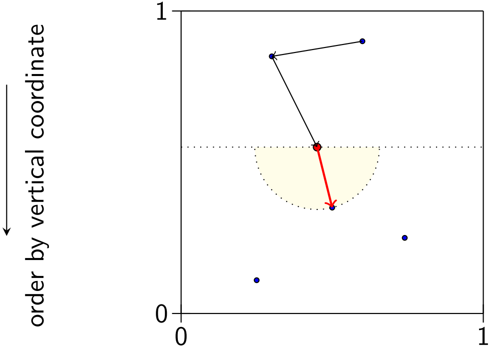
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

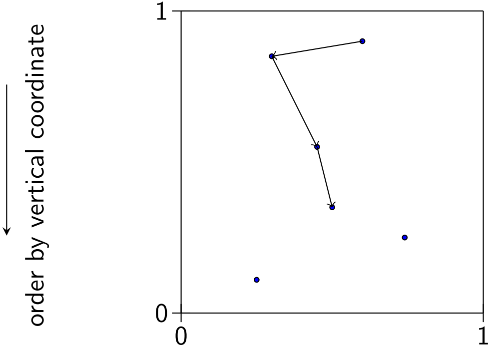
Vertices $1, \dots, n$ have uniform random locations $U_1, \dots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

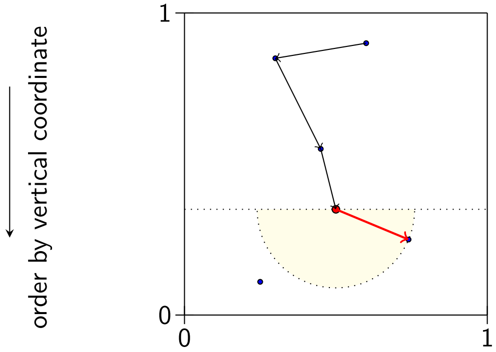
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

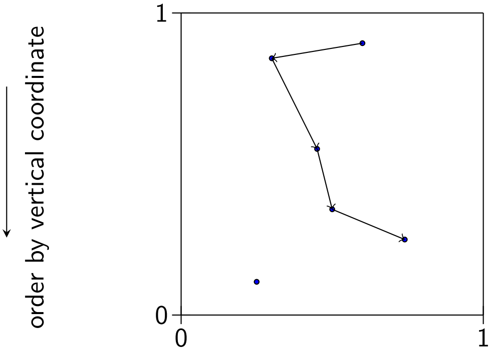
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

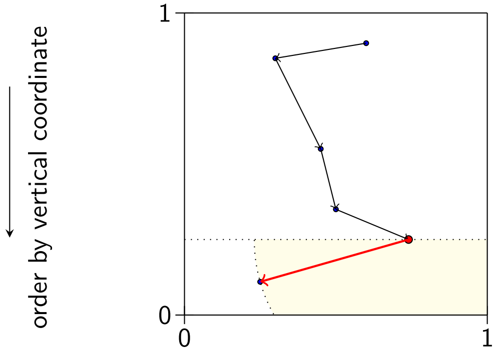
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

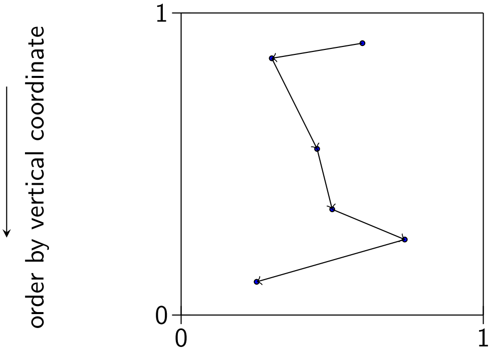
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

Minimal directed spanning tree

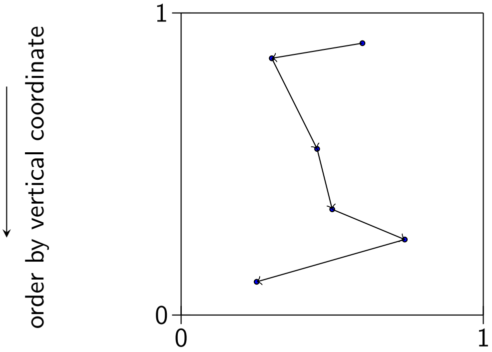
Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



Join each point to its nearest neighbour below.

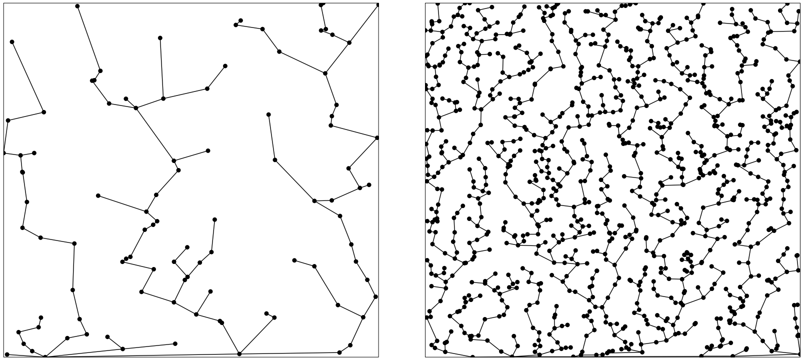
Minimal directed spanning tree

Vertices $1, \ldots, n$ have uniform random locations $U_1, \ldots, U_n$ in $[0,1]^d$. For every vertex, insert a (directed) edge from that vertex to its directed nearest neighbour, i.e., the closest vertex with strictly smaller dth coordinate. (The vertex with minimal dth coordinate does not emit an edge.) Here is a picture for $d = 2$, $n = 6$:



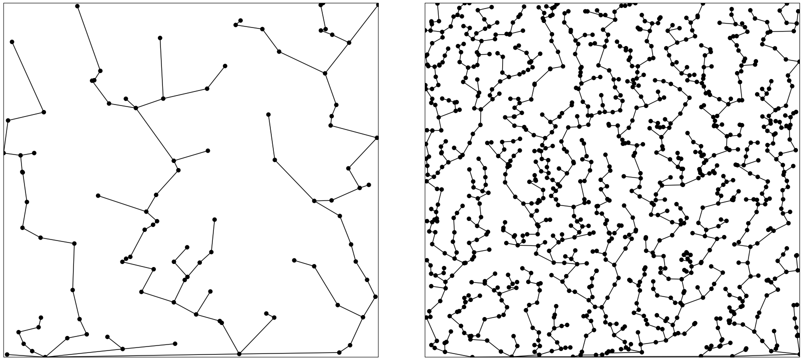
NB. The graph is tree.

Minimal directed spanning tree



Simulations of minimal directed spanning tree (MDST) in $[0,1]^2$, on $n = 100$ points (*left*) and $n = 1000$ points (*right*).

Minimal directed spanning tree



Simulations of minimal directed spanning tree (MDST) in $[0,1]^2$, on $n = 100$ points (*left*) and $n = 1000$ points (*right*).

Note again some inhomogeneity, this time due to spatial boundary effects near the base of the d-cube, where edges are typically much longer than the edges in the bulk.

Minimal directed spanning tree

Model is a variant of the minimal directed spanning tree introduced by Bhatt & Roy (2004); also a limiting case of Nandi & Manna (2007). Motivation is from the modelling of drainage or river networks.

Minimal directed spanning tree

Model is a variant of the minimal directed spanning tree introduced by Bhatt & Roy (2004); also a limiting case of Nandi & Manna (2007). Motivation is from the modelling of drainage or river networks.

We consider the total power-weighted edge length

$$\mathcal{M}_n^{d,\alpha} := \sum_{i=1}^{n} \min_{1 \leq j \leq n,\, j \neq i} \| U_j - U_i \|^{\alpha},$$

where $\alpha > 0$ (e.g., $\alpha = 1$ is the total edge length).

The limit theory of $\mathcal{M}_n^{d,\alpha}$ was considered by Penrose & Wade (2010).

Minimal directed spanning tree

Model is a variant of the minimal directed spanning tree introduced by Bhatt & Roy (2004); also a limiting case of Nandi & Manna (2007). Motivation is from the modelling of drainage or river networks.

We consider the total power-weighted edge length

$$\mathcal{M}_n^{d,\alpha} := \sum_{i=1}^{n} \min_{1 \leq j \leq n,\, j \neq i} \|U_j - U_i\|^{\alpha},$$

where $\alpha > 0$ (e.g., $\alpha = 1$ is the total edge length).

The limit theory of $\mathcal{M}_n^{d,\alpha}$ was considered by Penrose & Wade (2010).

Distinguishing feature: Unlike the ordinary nearest-neighbour graph, one gets some long edges near to the bottom boundary. These boundary effects may disrupt the Gaussian limit law, and can be described in terms of the on-line nearest-neighbour graph in one lower dimension.

Minimal directed spanning tree

Model is a variant of the minimal directed spanning tree introduced by BHATT & ROY (2004); also a limiting case of NANDI & MANNA (2007). Motivation is from the modelling of drainage or river networks.

We consider the total power-weighted edge length

$$\mathcal{M}_n^{d,\alpha} := \sum_{i=1}^{n} \min_{1 \leq j \leq n, j \neq i} \|U_j - U_i\|^{\alpha},$$

where $\alpha > 0$ (e.g., $\alpha = 1$ is the total edge length).

The limit theory of $\mathcal{M}_n^{d,\alpha}$ was considered by PENROSE & WADE (2010).

Distinguishing feature: Unlike the ordinary nearest-neighbour graph, one gets some long edges near to the bottom boundary. These boundary effects may disrupt the Gaussian limit law, and can be described in terms of the on-line nearest-neighbour graph in one lower dimension.

Increasing α increases relative importance of long edges, and leads to a Gaussian/non-Gaussian phase transition. Small α: only bulk contributes (Gaussian), large α only boundary contributes (non-Gaussian).

Minimal directed spanning tree: Limit theory

Theorem (Penrose & Wade* 2010)

- *If* $0 < \alpha < d/2$, *then, as* $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{M}_n^{d,\alpha} - \mathbb{E}\,\mathcal{M}_n^{d,\alpha} \right) \xrightarrow{d} Z,$$

where Z *is non-degenerate normal.*

Minimal directed spanning tree: Limit theory

Theorem (Penrose & Wade* 2010)

- *If $0 < \alpha < d/2$, then, as $n \to \infty$,*

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{M}_n^{d,\alpha} - \mathbb{E}\,\mathcal{M}_n^{d,\alpha} \right) \xrightarrow{d} Z,$$

  where Z is non-degenerate normal.

- *If $\alpha > d/2$, then, as $n \to \infty$,*

$$\mathcal{M}_n^{d,\alpha} - \mathbb{E}\,\mathcal{M}_n^{d,\alpha} \xrightarrow{d} Q_{d-1,\alpha},$$

  where $Q_{d-1,\alpha}$ is the limit law for the α-weighted length of the $(d-1)$-dimensional on-line nearest-neighbour graph.

Minimal directed spanning tree: Limit theory

Theorem (Penrose & Wade* 2010)

- *If* $0 < \alpha < d/2$, *then, as* $n \to \infty$,

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{M}_n^{d,\alpha} - \mathbb{E}\, \mathcal{M}_n^{d,\alpha} \right) \xrightarrow{d} Z,$$

  where Z *is non-degenerate normal.*

- *If* $\alpha > d/2$, *then, as* $n \to \infty$,

$$\mathcal{M}_n^{d,\alpha} - \mathbb{E}\, \mathcal{M}_n^{d,\alpha} \xrightarrow{d} Q_{d-1,\alpha},$$

  where $Q_{d-1,\alpha}$ *is the limit law for the* α*-weighted length of the* $(d-1)$*-dimensional on-line nearest-neighbour graph.*

- *If* $\alpha = d/2$, *then, as* $n \to \infty$,

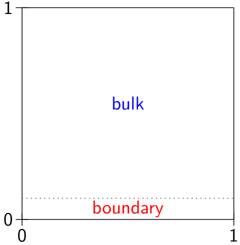
$$\mathcal{M}_n^{d,d/2} - \mathbb{E}\, \mathcal{M}_n^{d,d/2} \xrightarrow{d} Z + Q_{d-1,d/2},$$

  where Z *is non-degenerate normal, independent of* $Q_{d-1,d/2}$.

* Caveat: Penrose & Wade actually prove this for the Poissonized version.

Minimal directed spanning tree: Heuristics

Define $\varepsilon_n = n^{-\beta}$ for $\beta \approx 1/2$.

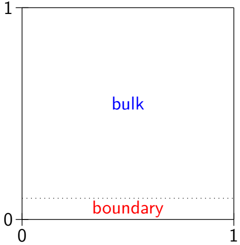


The normal component (when present) in the limit theorem arises from the bulk of the domain, vertices in $[0,1]^{d-1} \times [\varepsilon_n, 1]$.

The ONG component (when present) arises from edges very close to the boundary, i.e., vertices in $[0,1]^{d-1} \times [0, \varepsilon_n]$.

Minimal directed spanning tree: Heuristics

Define $\varepsilon_n = n^{-\beta}$ for $\beta \approx 1/2$.



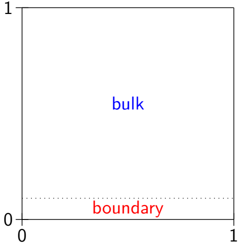
The normal component (when present) in the limit theorem arises from the bulk of the domain, vertices in $[0,1]^{d-1} \times [\varepsilon_n, 1]$.

The ONG component (when present) arises from edges very close to the boundary, i.e., vertices in $[0,1]^{d-1} \times [0, \varepsilon_n]$.

$$\mathcal{M}_n^{d,\alpha} = \mathcal{M}_n^{d,\alpha:\text{bulk}} + \mathcal{M}_n^{d,\alpha:\text{bndry}}.$$

Minimal directed spanning tree: Heuristics

Define $\varepsilon_n = n^{-\beta}$ for $\beta \approx 1/2$.



The normal component (when present) in the limit theorem arises from the bulk of the domain, vertices in $[0,1]^{d-1} \times [\varepsilon_n, 1]$.

The ONG component (when present) arises from edges very close to the boundary, i.e., vertices in $[0,1]^{d-1} \times [0, \varepsilon_n]$.

$$\mathcal{M}_n^{d,\alpha} = \mathcal{M}_n^{d,\alpha:\text{bulk}} + \mathcal{M}_n^{d,\alpha:\text{bndry}}.$$

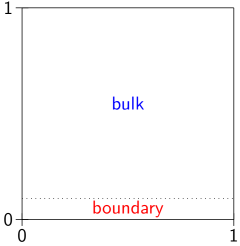
Similarly to the ordinary nearest-neighbour graph, local dependence and roughly homogeneous edge lengths imply

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{M}_n^{d,\alpha:\text{bulk}} - \mathbb{E}\,\mathcal{M}_n^{d,\alpha:\text{bulk}} \right) \xrightarrow{d} Z.$$

In particular, if $\alpha = d/2$, there's no scaling, while if $\alpha > d/2$, then $\mathcal{M}_n^{d,\alpha:\text{bulk}} - \mathbb{E}\,\mathcal{M}_n^{d,\alpha:\text{bulk}} \to 0$ in probability.

Minimal directed spanning tree: Heuristics

Define $\varepsilon_n = n^{-\beta}$ for $\beta \approx 1/2$.



The normal component (when present) in the limit theorem arises from the bulk of the domain, vertices in $[0,1]^{d-1} \times [\varepsilon_n, 1]$.

The ONG component (when present) arises from edges very close to the boundary, i.e., vertices in $[0,1]^{d-1} \times [0, \varepsilon_n]$.

$$\mathcal{M}_n^{d,\alpha} = \mathcal{M}_n^{d,\alpha:\text{bulk}} + \mathcal{M}_n^{d,\alpha:\text{bndry}}.$$

On the other hand, we claim $\mathcal{M}_n^{d,\alpha:\text{bndry}} \approx \mathcal{O}_{N_n}^{d-1,\alpha}$ for $N_n \to \infty$. If $\alpha \geq d/2$, then $\alpha > (d-1)/2$, and, by limit theorem for the ONG, $\mathcal{M}_n^{d,\alpha:\text{bndry}} - \mathbb{E}\,\mathcal{M}^{d,\alpha:\text{bndry}} \xrightarrow{d} Q_{d-1,\alpha}$. If $\alpha < d/2$, then

$$n^{(\alpha/d)-(1/2)} \left(\mathcal{M}_n^{d,\alpha:\text{bndry}} - \mathbb{E}\,\mathcal{M}_n^{d,\alpha:\text{bndry}}\right) \to 0,$$

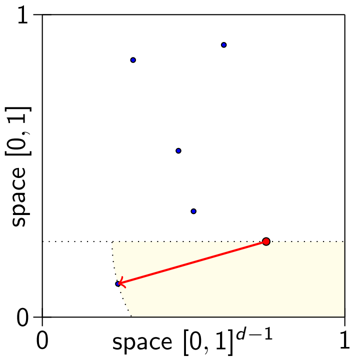
in probability.

Minimal directed spanning tree: Heuristics

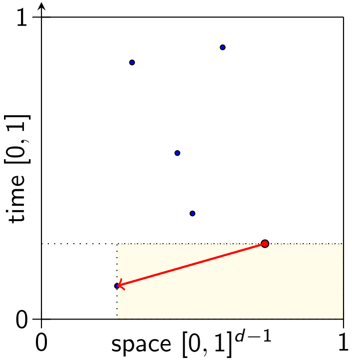
Why the approximation by the ONG near the boundary?

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.



MDST on $[0,1]^d$

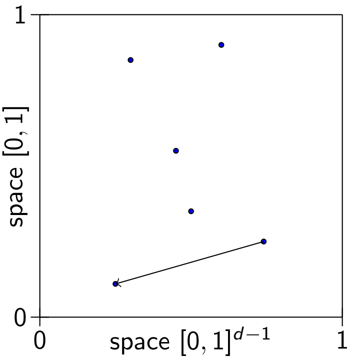


ONG on $[0,1]^{d-1}$

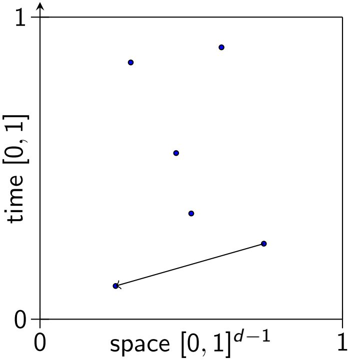
Distance on $[0,1]^d$ in MDST but distance on $[0,1]^{d-1}$ in ONG.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.



MDST on $[0,1]^d$

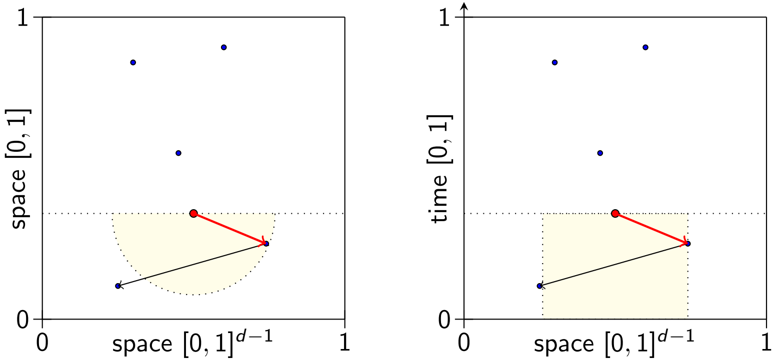


ONG on $[0,1]^{d-1}$

Edge is the “same” but length will be different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

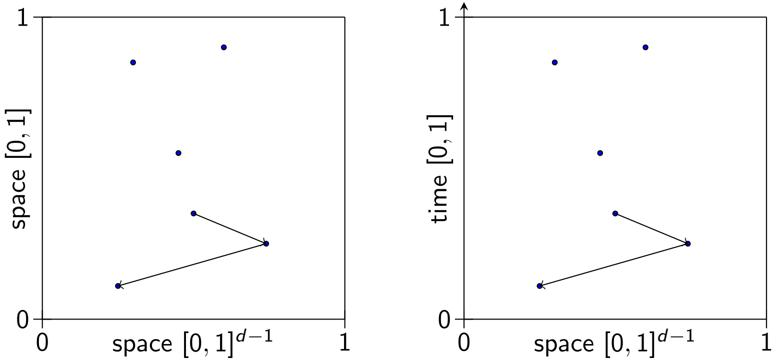


MDST on $[0,1]^d$ ONG on $[0,1]^{d-1}$

Again, edge is the "same" but length is different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

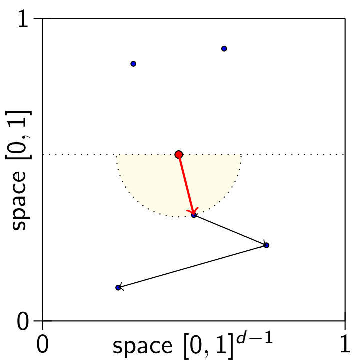


MDST on $[0,1]^d$ ONG on $[0,1]^{d-1}$

Again, edge is the “same” but length is different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

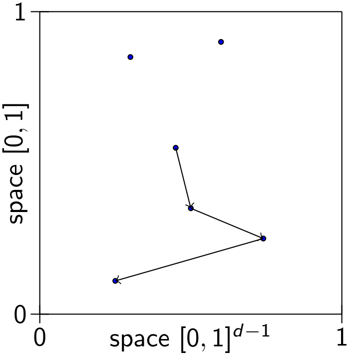


MDST on $[0,1]^d$ ONG on $[0,1]^{d-1}$

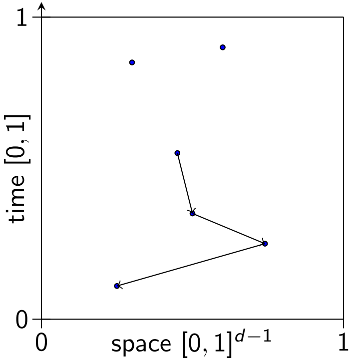
Again, edge is the "same" but length is different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.



MDST on $[0,1]^d$

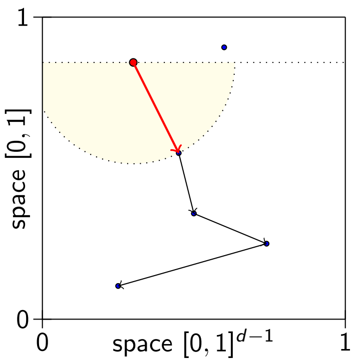


ONG on $[0,1]^{d-1}$

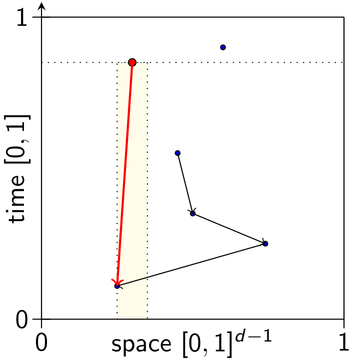
Again, edge is the "same" but length is different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.



MDST on $[0,1]^d$

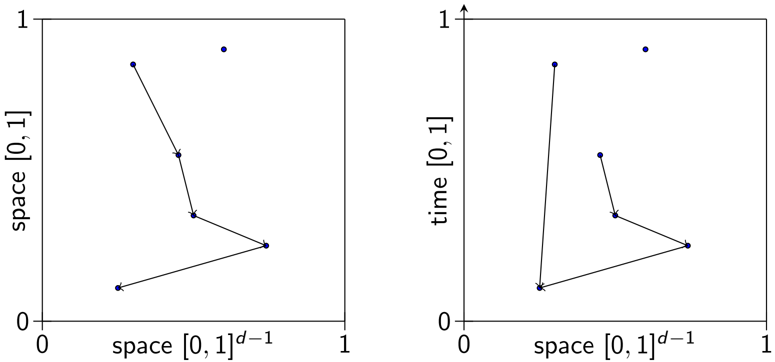


ONG on $[0,1]^{d-1}$

Not just distances, but nearest neighbours are different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

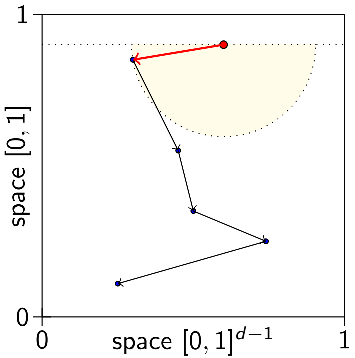


MDST on $[0,1]^d$ ONG on $[0,1]^{d-1}$

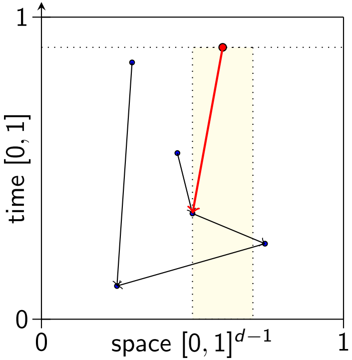
Not just distances, but nearest neighbours are different.

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.



MDST on $[0,1]^d$

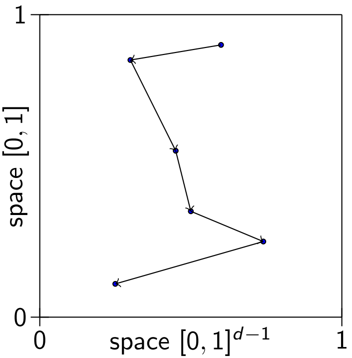


ONG on $[0,1]^{d-1}$

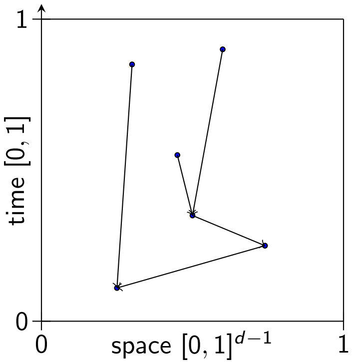
Another coupling "failure".

Minimal directed spanning tree: Heuristics

Why the approximation by the ONG near the boundary? Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.



MDST on $[0,1]^d$



ONG on $[0,1]^{d-1}$

Another coupling "failure".

Minimal directed spanning tree: Heuristics

Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

Minimal directed spanning tree: Heuristics

Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

Errors are of two types: (i) "same" edge present in both graphs, but different lengths; (ii) different edges in different graphs.

Minimal directed spanning tree: Heuristics

Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

Errors are of two types: (i) "same" edge present in both graphs, but different lengths; (ii) different edges in different graphs.

In general both types of error can occur and can be significant (as in previous picture).

Minimal directed spanning tree: Heuristics

Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

Errors are of two types: (i) "same" edge present in both graphs, but different lengths; (ii) different edges in different graphs.

In general both types of error can occur and can be significant (as in previous picture).

But, if we consider only the boundary zone, all vertical distances are very small, so errors of type (i) are minimized in magnitude. A similar argument shows that errors of type (ii) either rarely occur, or, if they do, give only a small discrepancy in the two edge lenghts.

Minimal directed spanning tree: Heuristics

Couple MDST on $[0,1]^d$ to an ONG on $[0,1]^{d-1}$ by projecting points onto the first $d-1$ coordinates, and use dth coordinate order for the time in the ONG.

Errors are of two types: (i) "same" edge present in both graphs, but different lengths; (ii) different edges in different graphs.

In general both types of error can occur and can be significant (as in previous picture).

But, if we consider only the boundary zone, all vertical distances are very small, so errors of type (i) are minimized in magnitude. A similar argument shows that errors of type (ii) either rarely occur, or, if they do, give only a small discrepancy in the two edge lenghts.

Thus one obtains that the MDST length in the boundary zone is well approximated by the ONG in $(d-1)$ dimensions.

Outline

Closing remarks

Some other directions:

- Rates of convergence: Avram, Bertsimas, Penrose, Yukich, Schulte, Eichelsbacher, Thäle, Lachièze-Rey, Peccati, Yang, Bhattacharjee (2022), ...

Closing remarks

Some other directions:

- Rates of convergence: Avram, Bertsimas, Penrose, Yukich, Schulte, Eichelsbacher, Thäle, Lachièze-Rey, Peccati, Yang, Bhattacharjee (2022), ...
- MDST can be defined with other directional relations, e.g., in terms of cones: Georgiou, Tawhari (talk later today).

Closing remarks

Some other directions:

- Rates of convergence: Avram, Bertsimas, Penrose, Yukich, Schulte, Eichelsbacher, Thäle, Lachièze-Rey, Peccati, Yang, Bhattacharjee (2022), ...
- MDST can be defined with other directional relations, e.g., in terms of cones: Georgiou, Tawhari (talk later today).
- Can combine geometric connectivity rule with some version of preferential attachment. Various hybrid models of this type, e.g., Manna & Sen (2002), Fabrikant *et al.* (2002), Flaxman *et al.* (2006), Jordan & Wade (2015), ...

Closing remarks

Some other directions:

- Rates of convergence: Avram, Bertsimas, Penrose, Yukich, Schulte, Eichelsbacher, Thäle, Lachièze-Rey, Peccati, Yang, Bhattacharjee (2022), ...
- MDST can be defined with other directional relations, e.g., in terms of cones: Georgiou, Tawhari (talk later today).
- Can combine geometric connectivity rule with some version of preferential attachment. Various hybrid models of this type, e.g., Manna & Sen (2002), Fabrikant *et al.* (2002), Flaxman *et al.* (2006), Jordan & Wade (2015), ...

Thank you for listening!

References

- F. Avram & D. Bertsimas, On central limit theorems in geometrical probability. *Ann. Appl. Probab.* **3** (1993) 1033–1046.
- N. Berger, B. Bollobás, C. Borgs, J. Chayes, & O. Riordan, Degree distribution of the FKP network model. *Theoret. Comput. Sci.* **379** (2007) 306–316.
- A.G. Bhatt & R. Roy, On a random directed spanning tree. *Adv. in Appl. Probab.* **36** (2004) 19–42.
- C. Bhattacharjee, Gaussian approximation for rooted edges in a random minimal directed spanning tree. *Random Structures Algorithms* (2022).
- P.J. Bickel & L. Breiman, Sums of functions of nearest neighbor distances, moment bounds, limit theorems and a goodness of fit test. *Ann. Probab.* **11** (1983) 185–214.
- A. Fabrikant, E. Koutsoupias, & C.H. Papadimitriou, Heuristically optimized trade-offs: A new paradigm for power laws in the internet. pp. 110–122 in *ICALP 2002*, (Widmayer *et al.*, eds.), Springer, 2002.
- A.D. Flaxman, A.M. Frieze, & J. Vera, A geometric preferential attachment model of networks. *Internet Math.* **3** (2006) 187–205.
- J. Jordan, Degree sequences of geometric preferential attachment graphs. *Adv. Appl. Probab.* **42** (2010) 319–330.
- J. Jordan & A.R. Wade, Phase transitions for random geometric preferential attachment graphs. *Adv. Appl. Probab.* **47** (2015) 565–588.
- S.S. Manna, G. Mukherjee, & P. Sen, Scale-free network on a vertical plane. *Phys. Rev. E* **69** (2004) 017102.

References

- S.S. Manna & P. Sen, Modulated scale-free network in Euclidean space. *Phys. Rev. E* **66** (2002) 066114.
- A.K. Nandi & S.S. Manna, A transition from river networks to scale-free networks. *New J. Phys.* **9** (2007) 30.
- M.D. Penrose, Multivariate spatial central limit theorems with applications to percolation and spatial graphs. *Ann. Probab.* **33** (2005) 1945–1991.
- M.D. Penrose & A.R. Wade, On the total length of the random minimal directed spanning tree. *Adv. in Appl. Probab.* **38** (2006) 336–372.
- M.D. Penrose & A.R. Wade, Limit theory for the random on-line nearest-neighbor graph. *Random Structures Algorithms* **32** (2008) 125–156.
- M.D. Penrose & A.R. Wade, Random directed and on-line networks. Section II.7, pp. 248–274 in *New Perspectives in Stochastic Geometry* (Kendall & Molchanov, eds.) Oxford University Press, 2009.
- M.D. Penrose & A.R. Wade, Limit theorems for random spatial drainage networks. *Adv. in Appl. Probab.* **42** (2010) 659–688.
- M.D. Penrose & J.E. Yukich, Central limit theorems for some graphs in computational geometry. *Ann. Appl. Probab.* **11** (2001) 1005–1041.
- J.M. Steele, Cost of sequential connection for points in space. *Oper. Res. Lett.* **8** (1989) 137–142.
- A.R. Wade, Explicit laws of large numbers for random nearest-neighbour-type graphs. *Adv. in Appl. Probab.* **39** (2007) 326–342.
- A.R. Wade, Asymptotic theory for the multidimensional random on-line nearest-neighbour graph. *Stochastic Process. Appl.* **119** (2009) 1889–1911.