Lecture Notes for Comp 480 - Winter 2023

Arthur Ayestas Hilgert

January 9, 2023

Here are lecture summaries for the sections of Probability on Trees and Networks that I will present.

Contents

Course Introduction

If you lose my notes please consult my web page:
`http://arthurayestas.xyz/comp480.html`

Introduction to graphs... mainly notation

Definition 1 (Graph)**.** A **graph** $G = (V, E)$ is an ordered pair containing a set of vertices V and a set of edges $E \subseteq V \times V$.

Remark 2. We may also write the vertex and edge set of G as $V(G)$ and $E(G)$ respectively.

Definition 3 (Edge)**.** An **edge** is a member of E. We write such an edge as: $\langle x, y \rangle$ with $x, y \in V$.

Definition 4 (Oriented Edge)**.** An **(oriented) edge** is an edge that is direction agnostic. We write such an edge as: $[x, y]$.

Theorem 5.

$$[x, y] \iff \langle x, y \rangle \wedge \langle y, x \rangle \iff (x, y), (y, x) \in E$$

Remark 6. In these notes I will add the notation E_s to denote a symmetric edge set.

Definition 7 (Vertex Adjacency)**.** Vertices x and y are **adjacent** if $(x, y) \in E \vee (y, x) \in E$. We write this as $x \sim y$. Vertex adjacency is **symmetric**.

Definition 8 (Edge Adjacency)**.** Edges $e_1 = (x, y)$ and $e_2 = (w, z)$ are **adjacent** if $y = z$. We write this as $e_1 \sim e_2$. Edge adjacency is **symmetric**.

Definition 9 (Incident)**.** Vertex x and edge $e = (y, z)$ are **incident** if $x = z$.

Definition 10 (Degree)**.** The **degree** of x is the number vertecies that are adjacent to x. We write this quantity as: $\deg(x)$.

Definition 11 (Locally Finite)**.** The graph $G = (V, E)$ is locally finite if $deg(x) < \infty$ for all $x \in V$.

Definition 12 (Regularity)**.** The graph $G = (V, E)$ is d-**regular** if $deg(x) = d$ for all $x \in V$.

Definition 13 (Cartesian Product)**.** The **Cartesian product** of graphs G_1 & G_2 is written as: $G_1 \square G_2$, with the following vertex and edge sets:

$$V(G_1 \square G_2) := V(G_1) \times V(G_2)$$

$$E(G_1 \square G_2) := \{((x_1, x_2), (y_1, y_2)) : (x_1 = y_1 \wedge (x_2, y_2) \in E(G_2)) \vee (x_2 = y_2 \wedge (x_1, y_1) \in E(G_2))\} \subseteq V(G_1 \square G_2)$$

Definition 14 (Tensor Product)**.** The **Tensor product** of graphs G_1 & G_2 is written as: $G_1 \times G_2$, with the following vertex and edge sets:

$$V(G_1 \times G_2) := V(G_1) \times V(G_2)$$

$$E(G_1 \times G_2) := \{((x_1, x_2), (y_1, y_2)) : (x_1, y_1) \in E(G_1) \wedge (x_2, y_2) \in E(G_2)\} \subseteq V(G_1 \times G_2)$$

Definition 15 (Strong Product)**.** The **Strong product** of graphs G_1 & G_2 is written as: $G_1 \boxtimes G_2$, with the following vertex and edge sets:

$$V(G_1 \boxtimes G_2) := V(G_1) \times V(G_2)$$

$$E(G_1 \boxtimes G_2) := E(G_1 \square G_2) \cup E(G_1 \times G_2)$$

Definition 16 (Vertex Simple)**.** A path that does not visit a vertex more than once is **vertex simple**.

Definition 17 (Edge Simple)**.** A path that does not traverse an edge more than once is **edge simple**.

Definition 18 (Distance)**.** The **distance** between two vertices x & y is the number of edges of the shortest path between x and y, denoted $d(x, y)$.

Definition 19 (Network)**.** A **network** is a graph G, along with weight function $c : E(G) \to \mathbb{R}$.

Definition 20 (Induced Subnetwork)**.** Given $K \subseteq V(G)$, the **induced subnetwork** $G \restriction K$, is the subnetwork with vertex set K, edges $(K \times K) \cap E$, and weight function $c : (K \times K) \cap E \to \mathbb{R}$.

Definition 21 (Multigraph)**.** A **multigraph** is a pair of sets V, E, and a tail function $(\cdot)^- : E \to V$ and a and head function $(\cdot)^+ : E \to V$.

Definition 22 (Identifying)**.** The **multigraph** G/K obtained by identifying K to vertex $z \notin V(G)$ has vertex set $(V \setminus K) \cup z$, edge set $E(G)$, and inherit head and tail functions from G altered such that mappings to $x \in K$ now map to z.

Definition 23 (Contraction)**.** Given $e \in E(G)$, the **contraction** G/e is obtained by removing e from $E(G)$, then identifying both e^+ and e^-.

Definition 24 (Multigraph Homomorphism)**.** A **multigraph homomorphism** $\phi : G_1 \to G_2$ preserving incidence and edge orientation.

Definition 25 (Network Homomorphism)**.** A **network homomorphism** is a graph homomorphism satisfying $c_1(e) = c_2(\phi(e))$ for all $e \in E(G_1)$.

Branching Number

Definition 26 (nth level)**.** The **nth level** of tree T is the number of vertices of distance n from the root o of T. We write this quantity as: T_n.

Definition 27 (Lower Growth Rate)**.** The **lower growth rate** of a locally finite infinite tree is defined as:

$$\underline{\mathrm{gr}} T := \varliminf_{n \to \infty} \sqrt[n]{|T_n|}$$

Definition 28 (Upper Growth Rate)**.** The **upper growth rate** of a locally finite infinite tree is defined as:

$$\overline{\mathrm{gr}} T := \varlimsup_{n \to \infty} \sqrt[n]{|T_n|}$$

Definition 29 (Growth Rate)**.** The **growth rate** of a locally finite infinite tree, is defined as:

$$\mathrm{gr} T := \lim_{n \to \infty} \sqrt[n]{|T_n|}$$

Definition 30 (Flow)**.** A **flow** function is a nonnegative function θ respecting the property that for all vertices x with parent z and children $y_1, ..., y_d$, $\theta((z,x)) = \sum_{i=1}^{d} \theta((x, y_i))$ holds.

Definition 31 (Branching Number)**.** Let T be a locally finite infinite tree, and let θ be a flow function. We define the **branching number** as follows:

$$\text{br}T := \sup\{\lambda \geq 1 \in \mathbb{R} | \exists \theta \forall n \in \mathbb{N} \forall x \in T_n : \theta((x_{\text{parent}}, x)) \leq \lambda^{-n}\}$$

Theorem 32. *$brT \leq \underline{gr}T$*

Proof. By induction, it is clear that $\sum_{x \in T_n} \theta(e(x))$, $\forall n \in \mathbb{N}$.

Therefore, given the flow constraint and choosing $\lambda > \underline{\text{gr}}T$, we have:

$$\begin{aligned} \varliminf_{n\to\infty} \sum_{x \in T_n} \theta(e(x)) &\leq \varliminf_{n\to\infty} \sum_{x \in T_n} \lambda^{-n} \\ &= \varliminf_{n\to\infty} |T_n| \lambda^{-n} \\ &= 0 \end{aligned}$$

Therefore, there is no flow from the root to infinity. $\square$

Electric Current

Definition 33 (Conductance)**.** A **conductance** function is a function $c : E \to \mathbb{R}^+$. We call $c(e) \in c(E)$ **conductances**.

Definition 34 (Energy)**.** The **energy** of flow θ, given conductances $c(E)$ is defined as follows (I have added notation **E**):

$$\mathbf{E} := \sum_e \frac{\theta(e)^2}{c(e)}$$

Theorem 35. *If $\lambda < brT$, then **current flows**. If $\lambda > brT$, then **current does not flow**.*

Random Walks

Definition 36 (Voltage Function)**.** A **voltage function** on vertices a_0 & a_1, given conductance function c, is a function $v : V \to \mathbb{R}_0^+$ satisfying the following conditions:

$$v(a_0) = 0 \tag{1}$$
$$v(a_1) = 1 \tag{2}$$
$$v(x)\sum_{i=1}^{d} c_i = \sum_{i=1}^{d} c_i v(y_i) \tag{3}$$

with $y_1, ..., y_d$ being the children of x.

Theorem 37 (Voltage as Probability)**.** *Let $x\alpha_1...\alpha_n$ be a random walk starting at x, such that $a_0 = \alpha_i$ & $a_1 = \alpha_j$ have been reached exactly once.*
$v(x) = \mathbb{P}\{j < i\}$

Definition 38 (Transient)**.** A transient random walk visits its starting vertex infinitely often.

Definition 39 (Recurrent)**.** A transient random walk visits its starting vertex infinitely often.

Remark 40. A random walk is either transient or recurrent.

Theorem 41. *If $\lambda < brT$, then RW_λ is **transient**. If $\lambda > brT$, then RW_λ is **recurrent**.*

Percolation

Theorem 42. *For any tree T, $p_c(T) = (brT)^{-1}$*