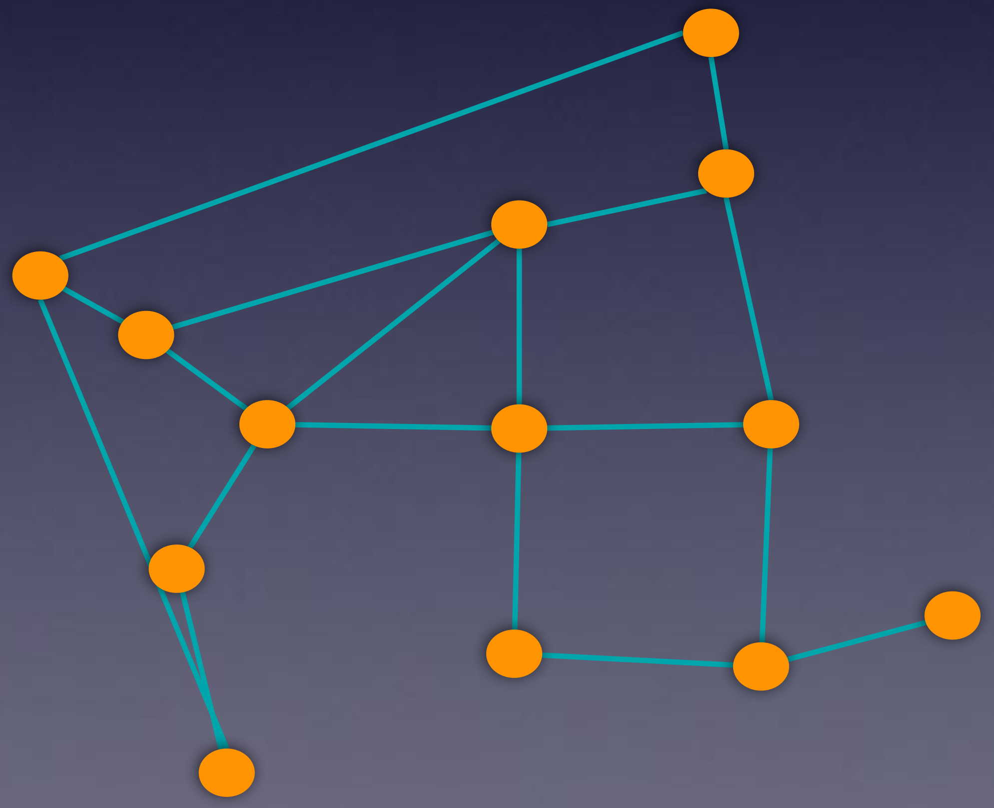
Testable Bounded Degree Graph Properties Are Random Order Streamable

Morteza Monemizadeh
Rutgers , Amazon

Joint work with: S. Muthukrishnan, Pan Peng, Christian Sohler

Sparse Graphs

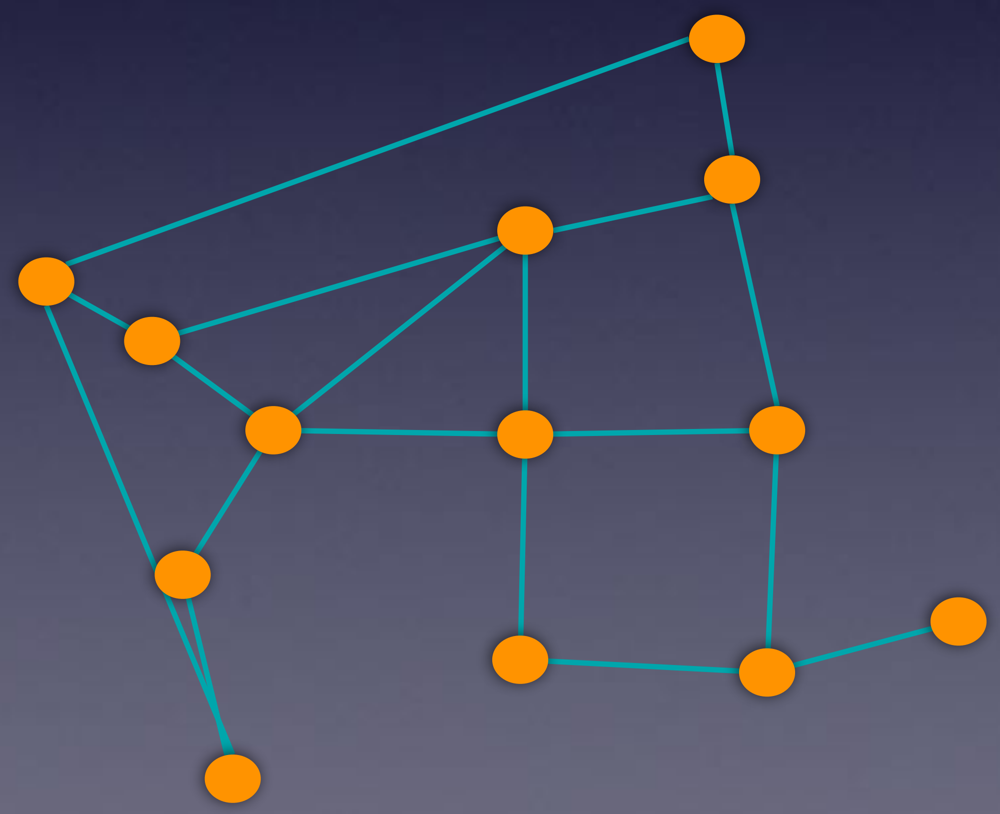
Given a graph $G(V,E)$ where $n=|V|$ and $m = |E| = O(n)$.



Examples

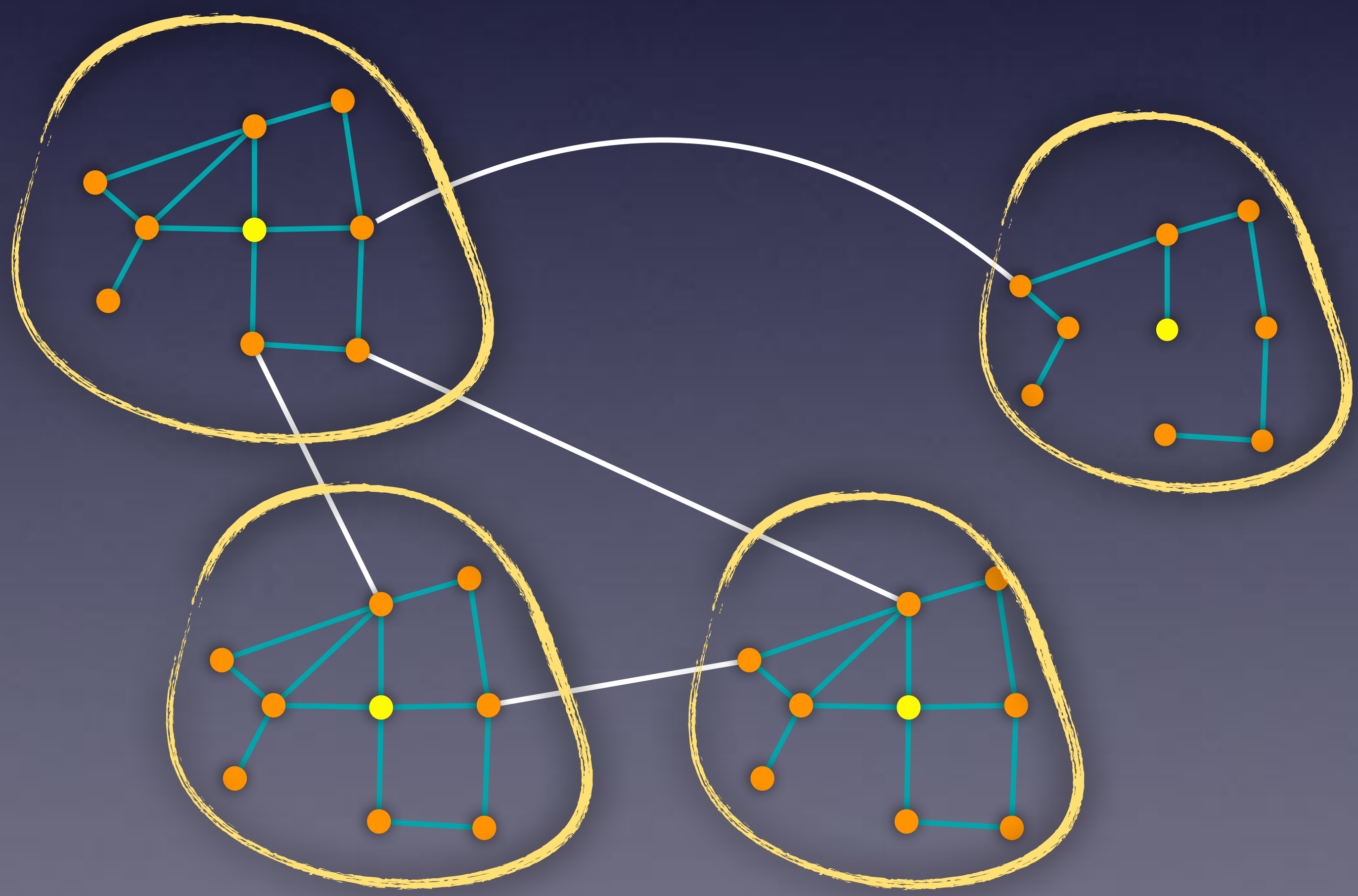
d-Bounded Degree Graphs

Given a graph G(V,E) whose maximum degree d is constant, where $n=|V|$ and $0 \leq m = |E| \leq nd$.



(Ɛ,k,d)-Hyperfinite G(V,E)

G is (ε,k,d)-hyperfinite graph if we remove a set of at most εdn edges of G s.t. the remaining graph has connected components of size at most k.



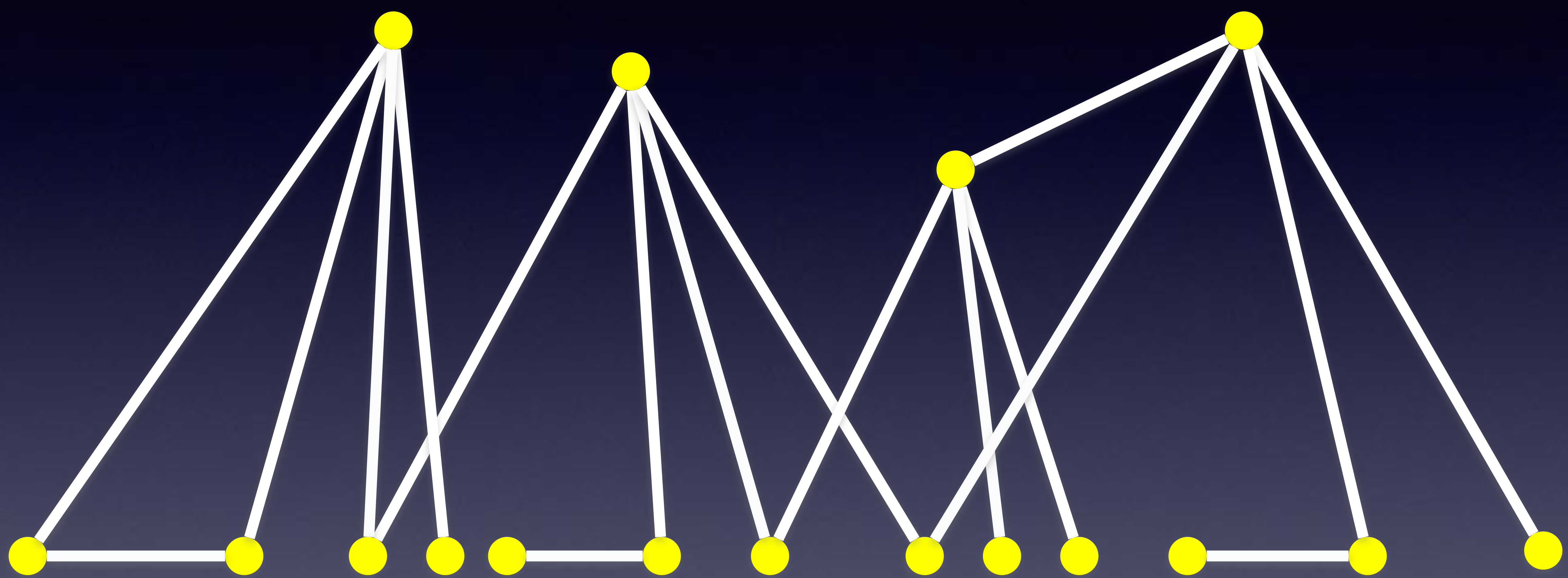
Arboricity

- ☛ Is a way to quantify the density of a graph G(V,E).
- ☛ $c = \max_U \{|E(U)|/(|U|-1)\}$ where U is a subset of V.
- ☛ G can be partitioned into at most c forests.
- ☛ Planar graphs have arboricity c = 3.

Maximum Matching

Given a graph G(V,E), find a set of pairwise non-adjacent of maximum size, i.e., no two edges share a common edge.

Example



Example

Maximum Matching

- 30-years-old algorithm due to Micali and Vazirani with running time $m\sqrt{n}$.
- Greedy algorithm returns maximal matching (2-approximation of maximum matching).

Maximum Matching

- 30-years-old algorithm due to Micali and Vazirani with running time $m\sqrt{n}$.
- Greedy algorithm returns maximal matching (2-approximation of maximum matching).

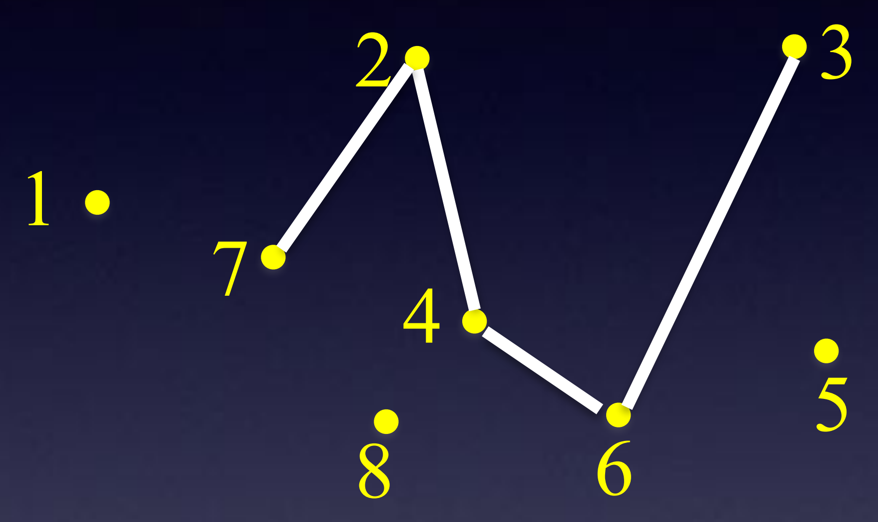
Big Data Models for Graphs

- ☞ Data Streams: Graph Streams
- ☞ Property Testing: Testing Graph Properties.
- ☞ Sublinear Time Approximation Algorithms

Streaming Model

Stream S =

Streaming Model



Stream S = (2,4) , (3,6) , (2,7) , (4,6), ...

Graph Streams

- ☞ Adverserial or Random Order Model
  - ☞ O(c)-approximate the size of matching in c-bounded arboricity graphs using $O(c\log^2 n)$ space in adversarial model.
  - ☞ O(polylog n)-approximate the size of matching in general graphs using O(polylog n) space in random order model.

Graph Streams

☞ Adverserial or Random Order Model

In general, it is not clear which graph problems can be solved with much smaller space in the random order stream than in the adversary order stream.

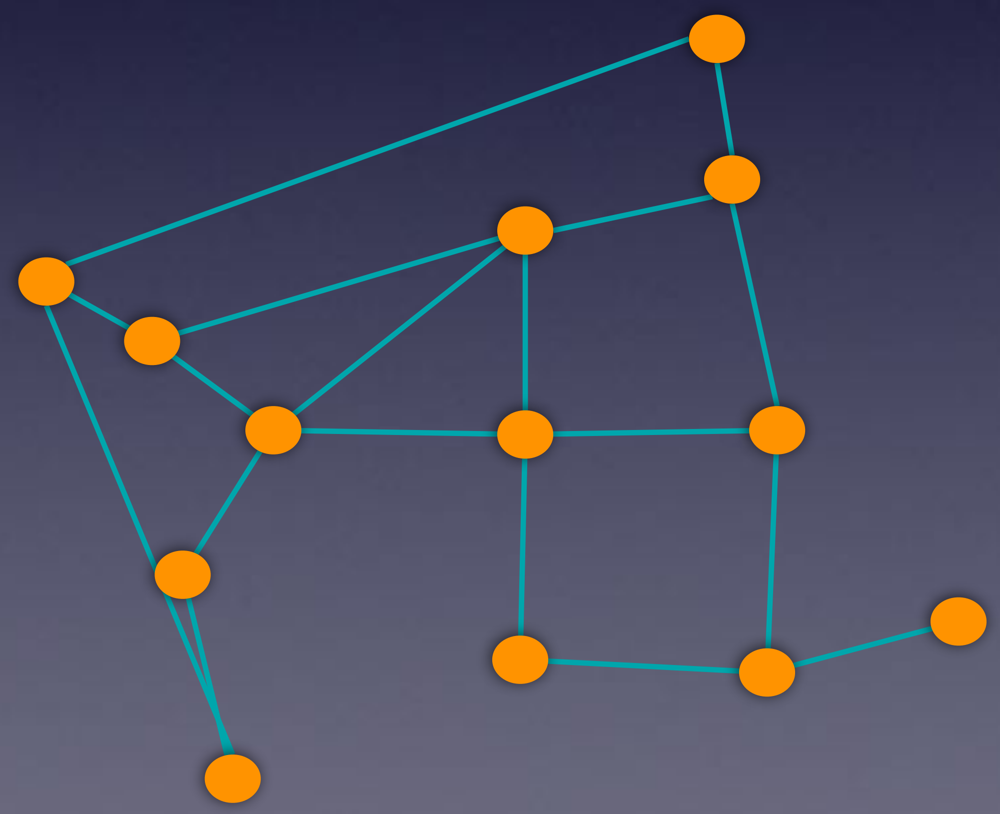
Graph Streams

- ☞ Semi-Streaming Model: O(n polylog(n)) space
  - ☞ Sparse Graphs: m=O(n)
    - ☞ O(polylog(n)) or even better O(1) space

Constant Query Property Testing

d-Bounded Graph

Given a graph $G(V,E)$ whose maximum degree d is constant, where $n=|V|$ and $0 \leq m = |E| \leq nd$.



Adjacency List Model

Query access to the adjacency list of G:

For any vertex v and index i one can query the i-th neighbor (if exists) of v in constant time.

Property Testing

A property $\prod_n$ for d-bounded n-vertex graphs is testable with query complexity q, if for every $\boldsymbol{\varepsilon}$, d and n, there exists an algorithm that performs q(n,d, $\boldsymbol{\varepsilon}$) queries to the adjacency list of the graph and with probability 2/3

- Accepts any n-vertex d-bounded graph G satisfying $\prod_n$,
- Rejects any n-vertex d-bounded graph G that is $\boldsymbol{\varepsilon}$-far from satisfying $\prod_n$,
- If q(d, $\boldsymbol{\varepsilon}$) is independent of n, we call $\prod_n$ constant query testable.

Property Testing

Theorem: Any d-bounded graph property that is constant-query testable in the adjacency list model can be tested in random order streaming model with constant space.

Examples

Adversary Order Model:

Testing k-edge connectivity, k-vertex connectivity and cycle-freeness of d-bounded degree graphs needs $\mathbf{\Omega}(n^{1-O(\varepsilon)})$ space.

Dynamic graph stream algorithms in o(n) space.
Huang and Peng, ICALP 2016.

Examples

Random Order Model:

k-edge connectivity, k-vertex connectivity and cycle-freeness of d-bounded degree graphs are testable in constant space in the random order stream model, since they are constant-query testable in the adjacency list model.

Property testing in bounded degree graphs.

Oded Goldreich and Dana Ron, Algorithmica 2002

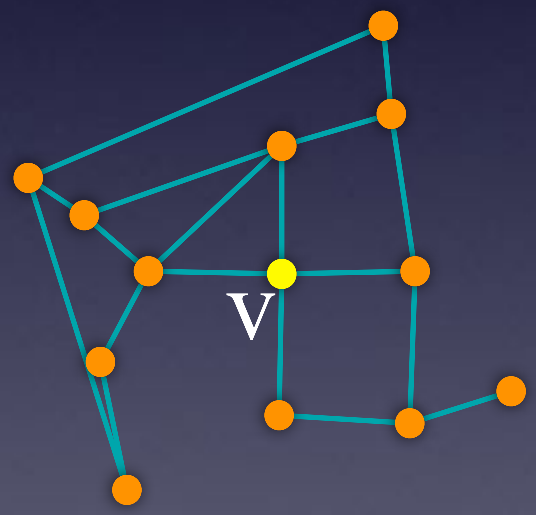
Property Testing

Proof (sketch): Every constant query property tester

- Samples a constant number of vertices
- Explores the k-discs of these vertices?
- Makes deterministic decisions based on the explored graph.

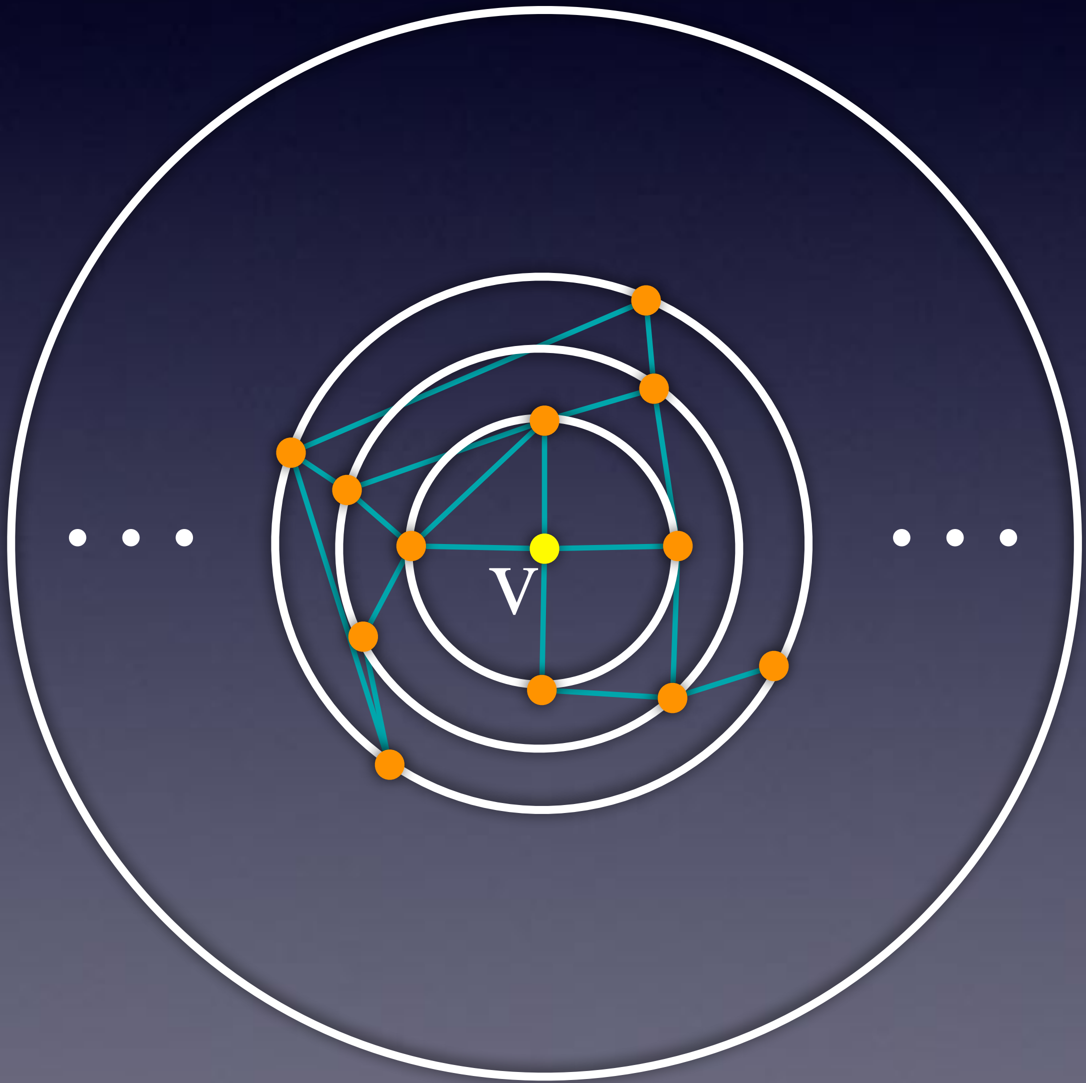
k-disc

The local neighborhood of depth k of a vertex is the subgraph induced by all vertices of distance at most k.



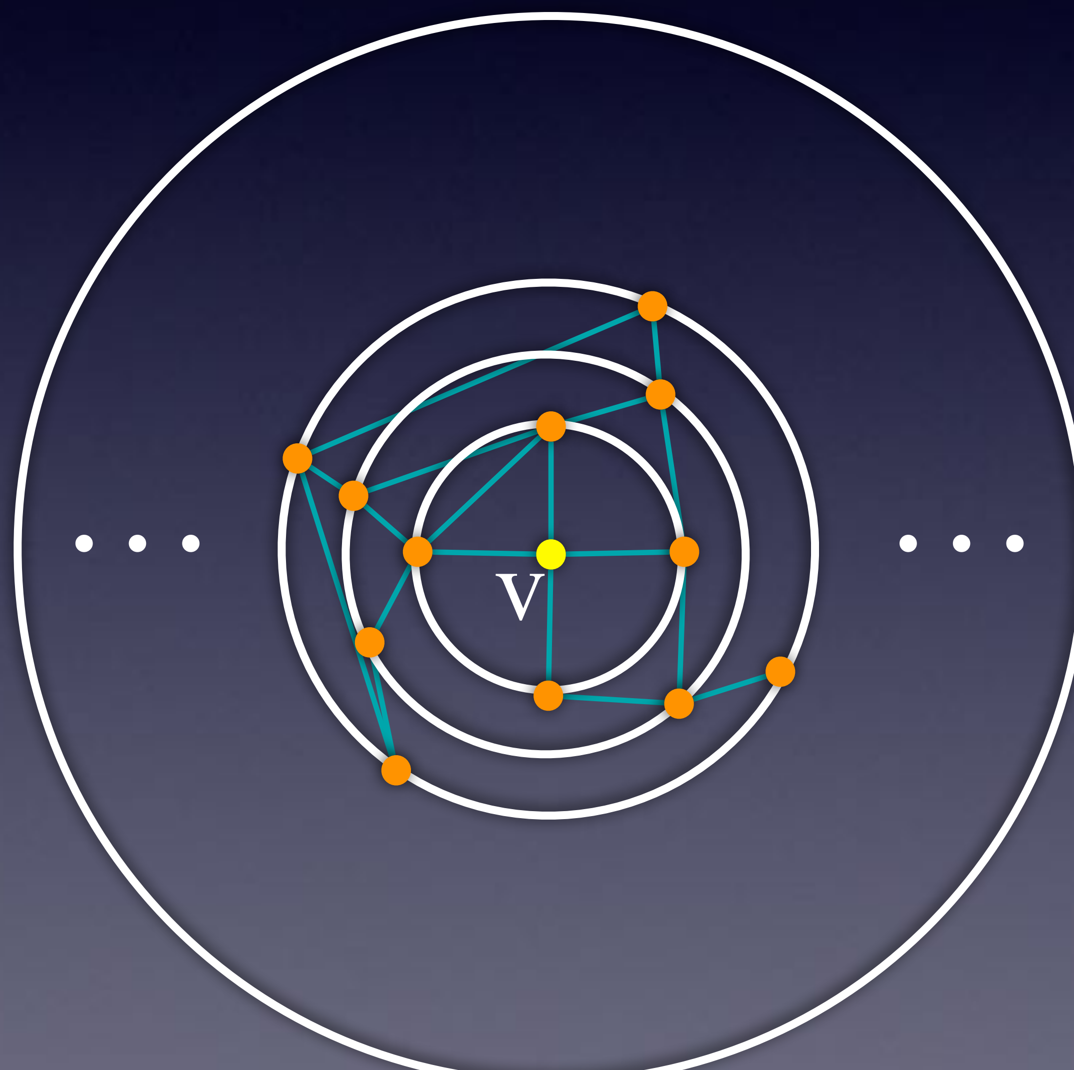
k-disc

The local neighborhood of depth k of a vertex is the subgraph induced by all vertices of distance at most k.



k-disc

The local neighborhood of depth k of a vertex is the subgraph induced by all vertices of distance at most k.



A k-disc has at most d^{k+1} vertices and d^{k+2} edges.

Constant-Time Approximation Algorithms

Adjacency List Model

Query access to the adjacency list of G:

For any vertex v index i one can query the the i-th neighbor (if exists) of v in constant time.

(x,y)-Approximation

We call a value t an (x,y)-approximation for the problem P, if for any instance I, we have

$$OPT(I) \leq t \leq x \cdot OPT(I) + y$$

For a minimization optimization problem P and an instance I, we let OPT(I) denote the value of some optimal solution of I.

O(1)-time Approximation Algorithm

Theorem: There exists an algorithm that uses constant space in the random order model, and with probability 2/3, $(1,\varepsilon n)$-approximates the size of a maximal matching.

Based on Locality Lemma due to Nguyen and Onak, FOCS'08

O(1)-time Approximation Algorithm

Similar result holds for minimum vertex cover, maximum matching, the number of connected components.

O(1)-time Approximation Algorithm

Theorem: There exists an algorithm that uses constant space in the random order model, and with probability 2/3, $(1\pm\varepsilon)$-approximates the size of a maximal matching.

O(1)-time Approximation Algorithm

Greedy Matching

Stream : e_1 e_2 e_3 e_4 … e_i …

Current Matching

M

O(1)-time Approximation Algorithm

Greedy Matching

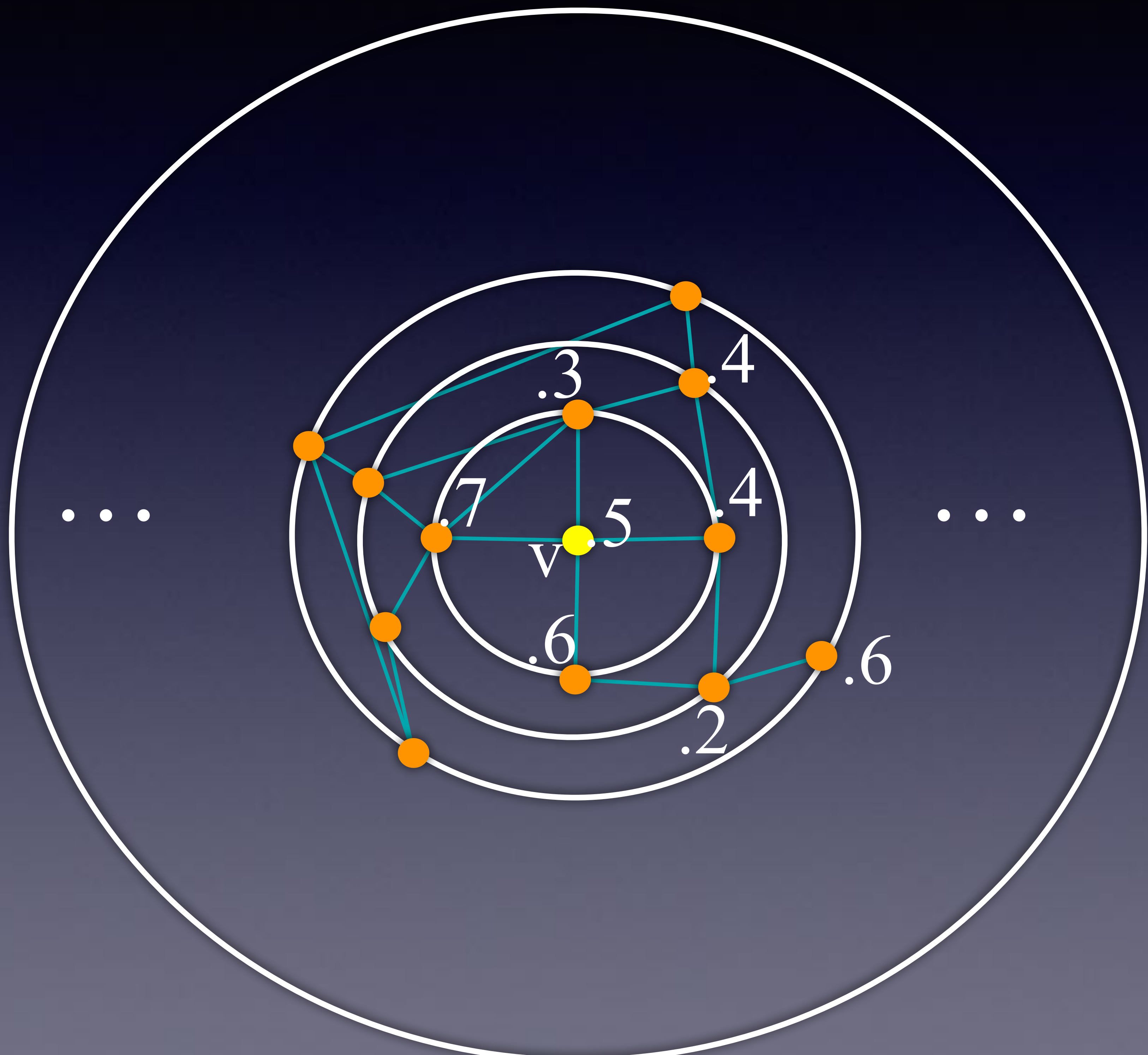
…

Stream : e_1 e_2 e_3 e_4 … e_i …

Is e_i in M?

Current Matching
M

O(1)-time Approximation Algorithm



k-Disc Primitive in Data Streams

k-disc Primitive

Given a random order stream S of edges of an underlying d-bounded degree graph G(V,E).

Sample the full k-disc of a vertex v (almost) uniformly at random.

2-Pass Streaming Algorithm

2-Pass Algorithm

First Pass:

Sample a set S of $(d^{k+2})!$ vertices and collect their observed k-discs in S.

In expectation, there exists at least one vertex in S whose full k-disc is observed.

2-Pass Algorithm

Second Pass:

Find the degree of vertices in (partially explored) k-discs of the vertices in S.

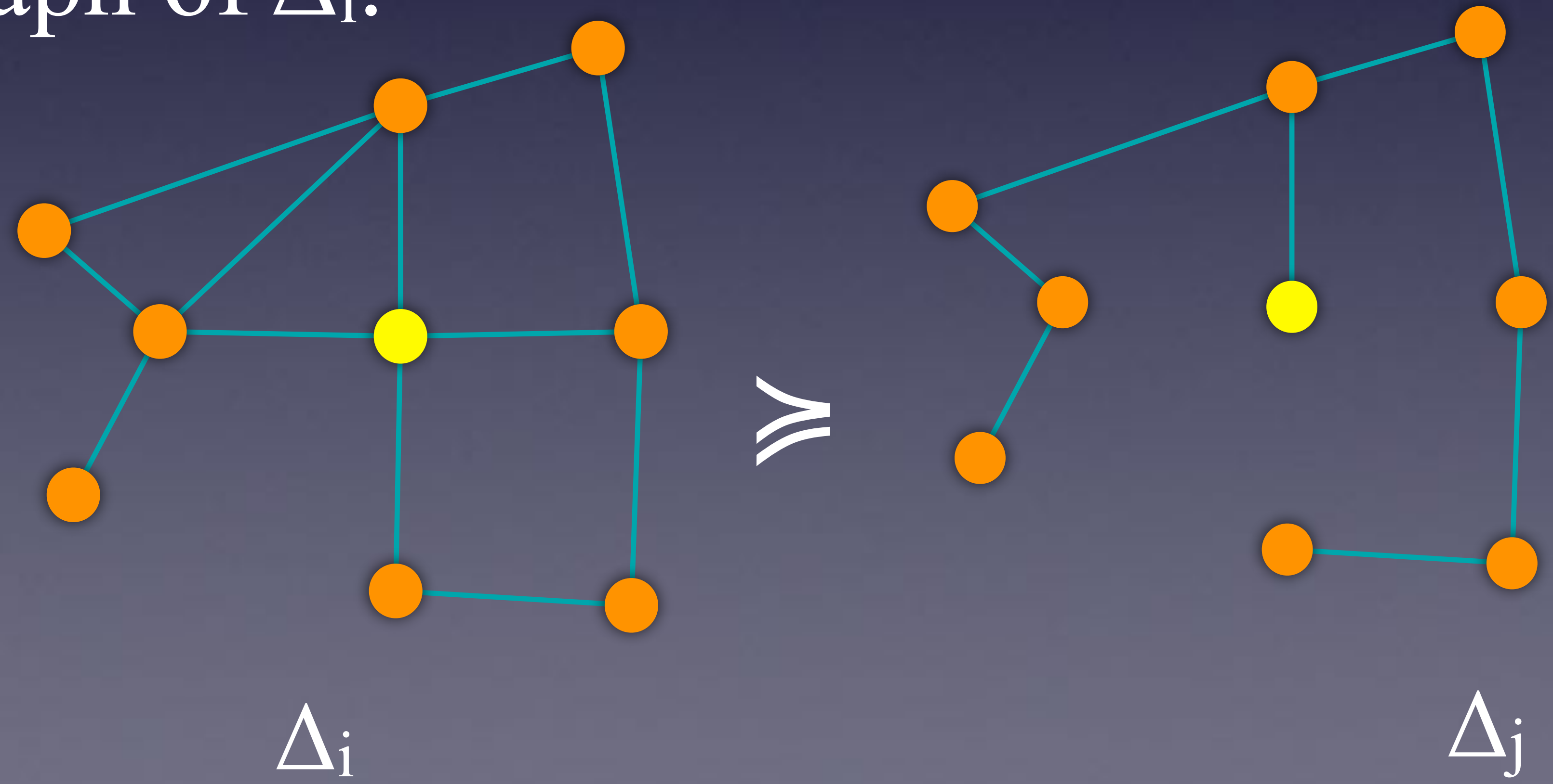
Report the k-disc of a vertex in S that is fully explored.

1-Pass Streaming Algorithm

Partial Order

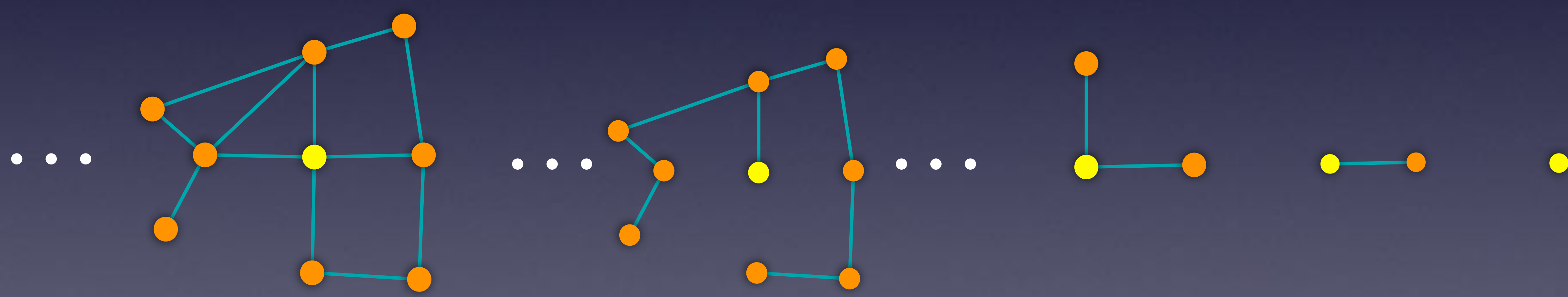
$H_{d,k}=\{\Delta_1,\ldots,\Delta_x\}$: The set of all k-disc isomorphism types.

$\Delta_i \geqslant \Delta_j$: Δ_j is root-preserving isomorphic to some subgraph of Δ_i.



Ordering

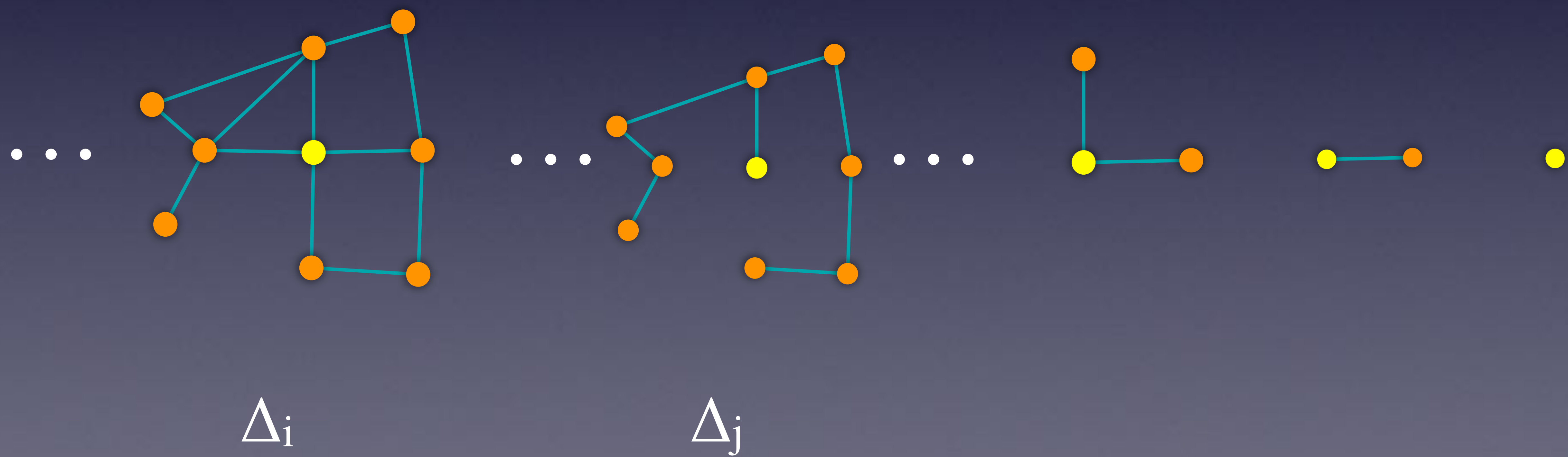
Order all the k-disc types $\Delta_1, \ldots, \Delta_x$ such that if $\Delta_i \succcurlyeq \Delta_j$, then $i \leq j$.



Δ_i Δ_j

Ordering

Order all the k-disc types $\Delta_1, \ldots, \Delta_x$ such that if $\Delta_i \geqslant \Delta_j$, then $i \leq j$.

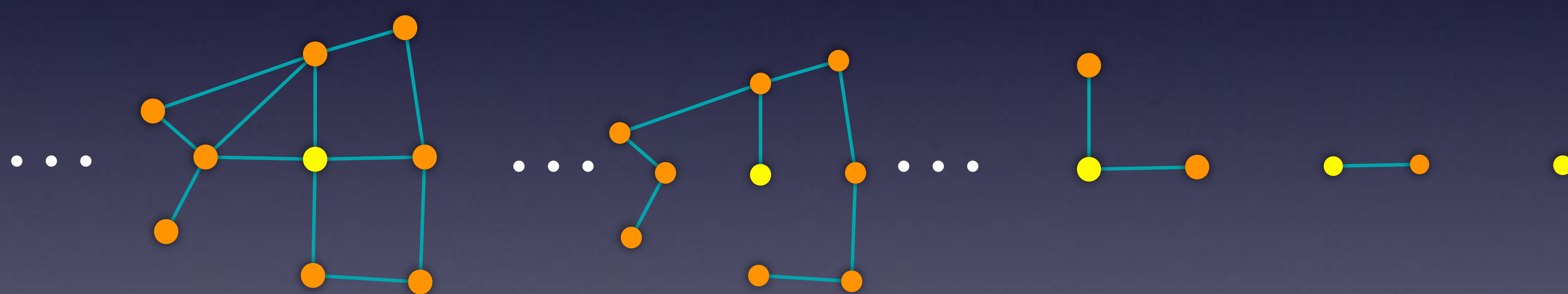


$\mathcal{G}(j)$: All the indices i, except j itself, such that $\Delta_i \geqslant \Delta_j$.

Frequency Vector F(G,d)

V_i: The set of vertices with k-disc isomorphic to Δ_i ,

$$V_i=\{v \in V: \mathrm{disc}_{k,G}(v) \cong \Delta_i\}$$



	Δ_i		Δ_j				
	10010		10		2324	8273	9744

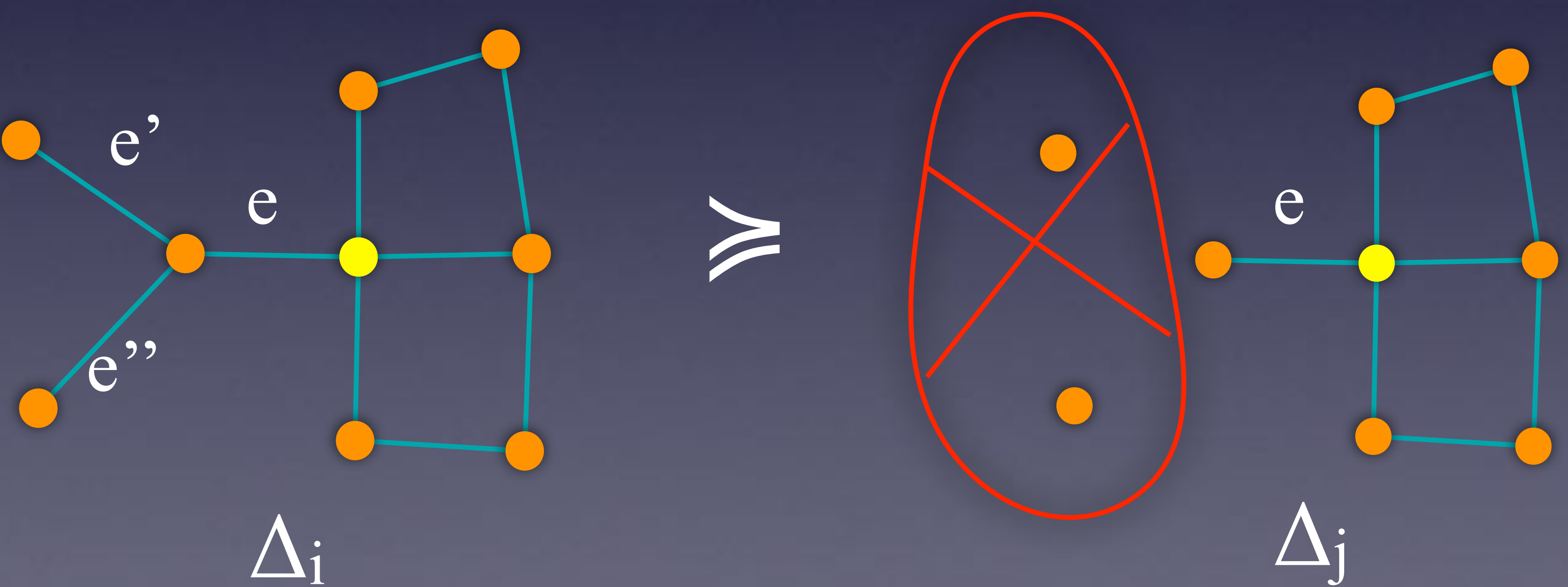
$|V_i|$ $f_i=|V_i|/n$

Marginal Probability

Let S be a random order Stream.

Let v be a vertex with k-disc isomorphic to Δ_i.

Marginal Probability: The probability $\lambda(\Delta_j|\Delta_i)$ that the observed k-disc of v in S is $disc_k(v,S) \cong \Delta_j$ for any j such that $\Delta_i \succcurlyeq \Delta_j$.



Stream S: …, e',…, e'', …, e,…

1-Pass Algorithm

Preprocssing:

Sample a set T of $O(2^{(d^{k+2})}! / \varepsilon^2)$ vertices.

1-Pass Algorithm

Preprocssing:

Sample a set T of $O(2^{(d^{k+2})!}/\varepsilon^2)$ vertices.

Streaming:

For each vertex $v \in T$:

☞ Collect the observed k-disc $disc_k(v,S)$ from the stream S.

☞ Let H_v be $disc_k(v,S)$.

1-Pass Algorithm

Postprocessing:

Let $H = \cup_{v \in T} H_v$.

1-Pass Algorithm

Postprocessing:

Let $H = \cup_{v \in T} H_v$.

For i =1 to x where $x = |F(G,d)|$

1-Pass Algorithm

Postprocessing:

Let $H = \cup_{v \in T} H_v$.

For i =1 to x where $x = |F(G,d)|$

☞ $Y_i = |\{v \in T : disc_{k,H}(v) \cong \Delta_i\}| / |T|$

1-Pass Algorithm

Postprocessing:

Let $H = \cup_{v \in T} H_v$.

For i =1 to x where $x = |F(G,d)|$

- ☞ $Y_i = |\{v \in T: disc_{k,H}(v) \cong \Delta_i\}|/|T|$
- ☞ $X_i = (Y_i - \sum_{j \in \mathcal{G}(i)} X_j \cdot \lambda(\Delta_i|\Delta_j)) \cdot \lambda^{-1}(\Delta_i|\Delta_i)$

$\mathcal{G}(i)$: All the indices j, except i itself, such that $\Delta_j \geqslant \Delta_i$.

1-Pass Algorithm

Postprocessing:

Let $H = \cup_{v \in T} H_v$.

For i =1 to x where $x = |F(G,d)|$

- ☞ $Y_i = |\{v \in T: disc_{k,H}(v) \cong \Delta_i\}|/|T|$
- ☞ $X_i = (Y_i - \sum_{j \in \mathcal{G}(i)} X_j \cdot \lambda(\Delta_i|\Delta_j)) \cdot \lambda^{-1}(\Delta_i|\Delta_i)$

Return $X_1, \ldots, X_x$.

$\mathcal{G}(i)$: All the indices j, except i itself, such that $\Delta_j \geqslant \Delta_i$.

Open Problems

☞ In general, it is not clear which graph problems can be solved with much smaller space in the random order stream than in the adversary order stream.

☞ What can we say about testing graph properties of unbounded planar (or minor-free) graphs in data streams?

Thank You

(Almost) Isomorphic Graphs

Benjamini, Shapira, and Schramm, STOC'08
Newman and Sohler, STOC'11

G_1 and G_2 : ($\mathcal{E}$,k,d)-hyperfinite graphs.

If $|F(G_1,d)-F(G_2,d)|_1 \leq \mathcal{E}dn$, then G_1 and G_2 are $\mathcal{E}$-close.

G_1 and G_2 are $\mathcal{E}$-close:

If we insert/delete at most $\mathcal{E}dn$ edges from G_1, then G_1 and G_2 becomes isomorphic.