technische universität
dortmund

Complexity and Efficient Algorithms Group /
Department of Computer Science

Testing the Cluster Structure of Graphs

Christian Sohler

Very Large Networks

Examples

- Social networks
- The World Wide Web
- Cocitation graphs
- Coauthorship graphs

Data size

- GigaByte upto TeraByte (only the graph)
- Additional data can be in the Peta-Byte range

Information in the Network Structure

Social network

- Edge: Two persons are „friends“
- Well-connected subgraph: A social group

Cocitation graphs

- Edge: Two papers deal with a similar subject
- Well-connected subgraph: Papers in a scientific area

Coauthor graphs

- Edge: Two persons have worked together
- Well-connected subgraph: Scientific community

How can we extract this information?

Objective

- Identify the well-connected subgraphs (*clusters*) of a huge graph

Problem

- Classical algorithms require at least linear time
- Might be too large for huge networks

Our approach

- Decide, if the graph has a *cluster structure* or is far away from it
- If yes, get a representative vertex from each (sufficiently big) cluster
- Running time sublinear in the input size

Formalizing the Problem – The Input

Input Model

- Undirected graph G=(V,E) with vertex set {1,…,n}
- Max. degree bounded by constant d
- Graph is stored in adjacency lists
- We can query for the i-th edge incident to vertex j in O(1) time

Formalizing the Problem – Cluster Structure

Definition

- The *conductance* $\Phi(C,V\text{-}C)$ is defined as $\frac{|\{(u,v) \in E : u \in C \text{ and } v \in V - C\}|}{d\,|C|}$
- The *conductance* $\Phi_G(G)$ of G is $\min_{C:|C| \leq |V|/2} \Phi(C,V\text{-}C)$

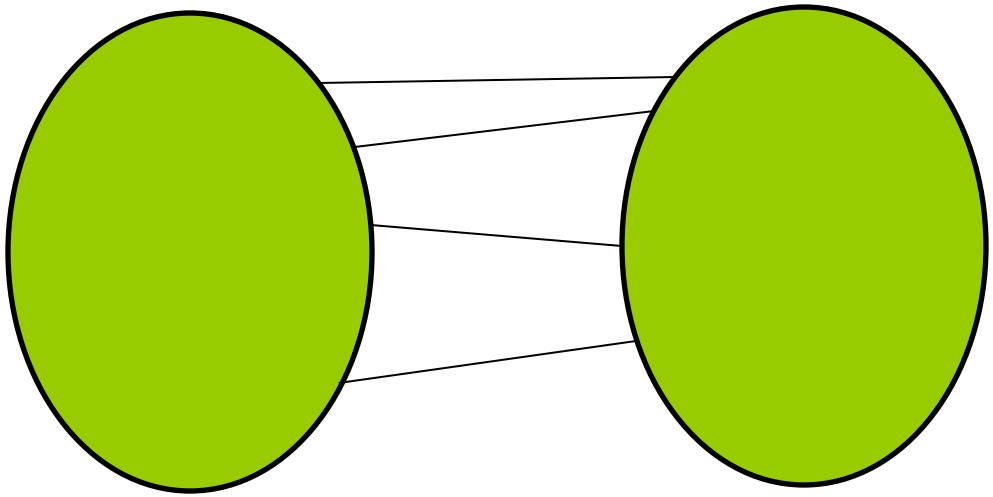
Definition

- A subset $C \subseteq V$ is called *(Φ_{in}, Φ_{out})-cluster*, if
- $\Phi_G(G[C]) \geq \Phi_{in}$
- $\Phi(C, V\text{-}C) \leq \Phi_{out}$

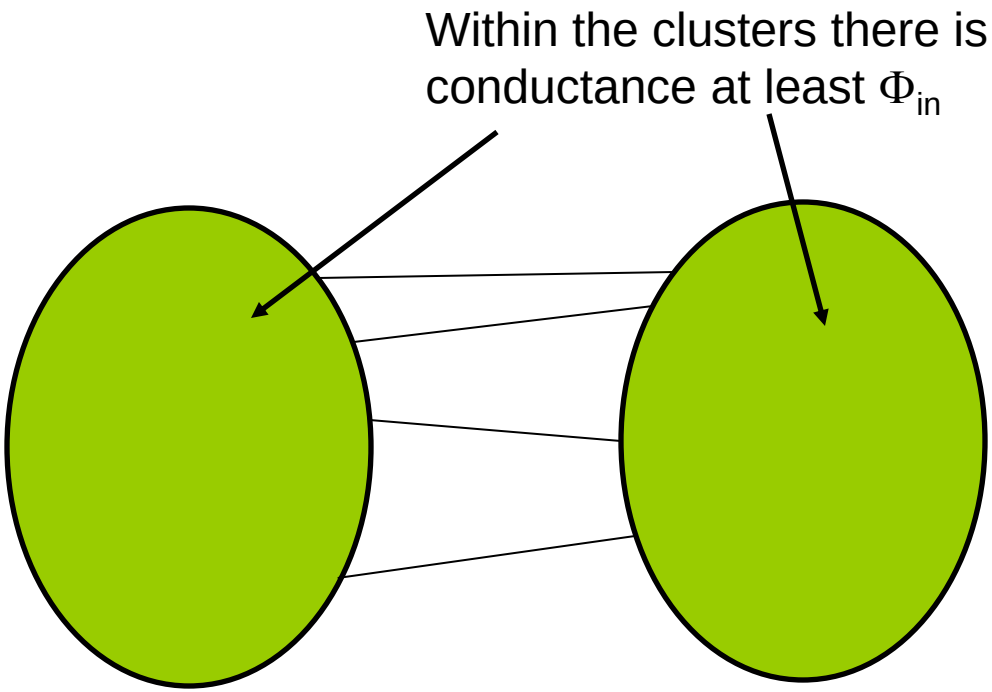
Definition

- A partition of V into at most k (Φ_{in}, Φ_{out})-clusters is called *$(k, \Phi_{in}, \Phi_{out})$-clustering*

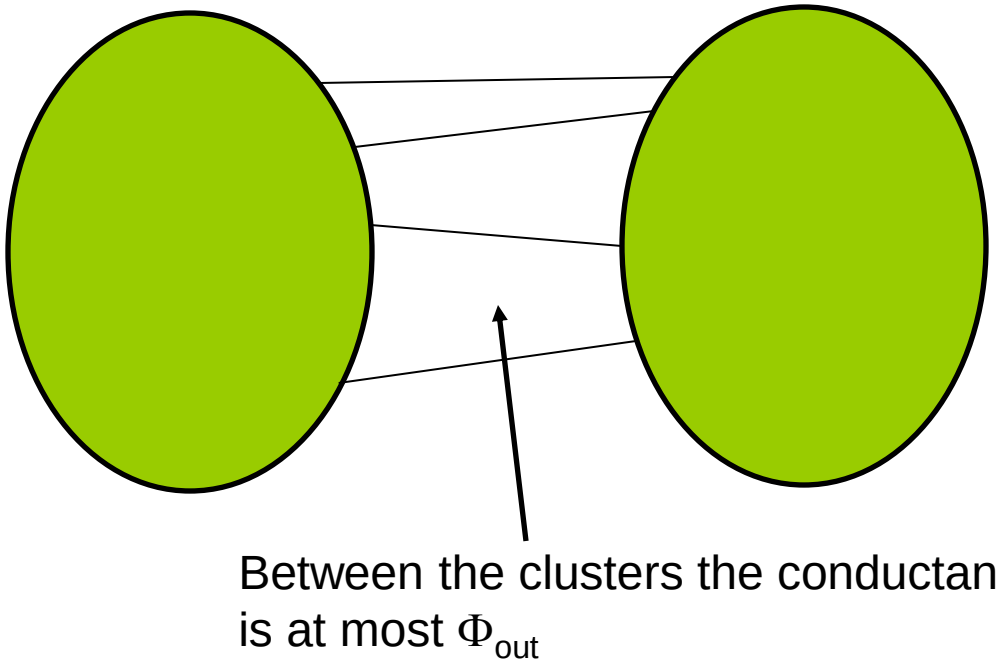
$(2, \Phi_{in}, \Phi_{out})$-Clustering – An Example



$(2, \Phi_{in}, \Phi_{out})$-Clustering – An Example



$(2, \Phi_{in}, \Phi_{out})$-Clustering – An Example



Formalizing the Problem – Property Testing

Definition [Goldreich, Ron, 2002]

- A graph with degree bound d is *ε-far from a property P*, if differs from any graph with maximum degree d that has property P in more than εd|V| edges.

- An algorithm is *property tester* in the bounded degree graph model, if given oracle access to a degree bounded graph G
  (a) it accepts every graph from P with probability at least 2/3
  (b) it rejects every graph that is ε-far from P with probability at least 2/3

- The *query complexity* of the algorithm is the number of queries made to the oracle

Formalizing the Problem

Our Objective

- Develop a property tester that

  (a) accepts with probability at least 2/3, when the input graph is a *$(k, \Phi_{in}, \Phi_{out})$-clustering*

  (b) rejects with probability at least 2/3, if the input graph is ε-far from every *$(k, \Phi_{in}^*, \Phi_{out}^*)$-clustering*

- The query complexity (and running time) of the tester should be as small as possible

Previous Work

k=1: Testing Expansion

- Conjectured an algorithm based on collision-statistics of random walks
- The algorithm is supposed to accept in $O^*(\sqrt{n})$ running time every Φ-expander and reject every expander, which is ε-far from a Φ^*-expander [Goldreich, Ron, 2000]
- First proof with a polylogarithmic gap between Φ and Φ^* [Czumaj, Sohler, 2010]
- Improvement of parameters to constant gap (with running time $O^*(n^{1/2+\delta})$) [Nachmias, Shapria, 2010; Kale, Seshadri 2011]

- O* assumes all input parameters except n to be constant and supresses logarithmic factors

Previous Work

TestingExpansion(G,ε)

- Sample Θ(1/ε) vertices uniformly at random
- **For each** sample vertex **do**
  - Perform O*(√n) random walk of length Θ*(log n) from each vertex
  - **if** the number of collisions among end points is too high **then reject**
- **accept**

Analysis

- If G is an expander, then a random walk converges to the *uniform distribution*
- Let p(v) be the distribution of the end points of a random walk starting at v
- $||p(v)||^2$ is the expected number of collisions
- The uniform distribution minimizes $||p(v)||^2$

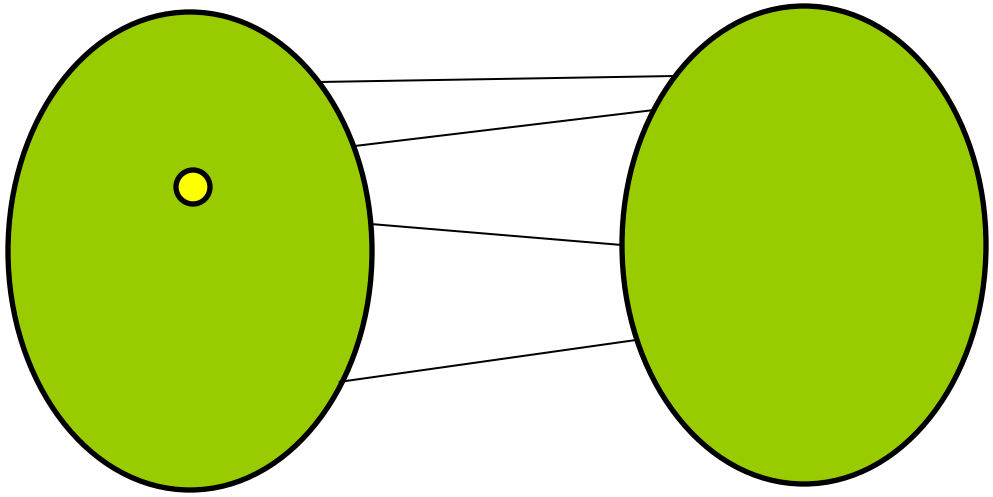
Testing k-Clusterings

Main Idea

- Two random walks starting from the same cluster should eventually have almost the same distribution (and this is almost uniform on the cluster)
- Two random walks starting in different cluster should have different distribution

Obstacles

- We cannot test against the uniform distribution since we don‘t know the clusters
- We do not compare stationary distributions



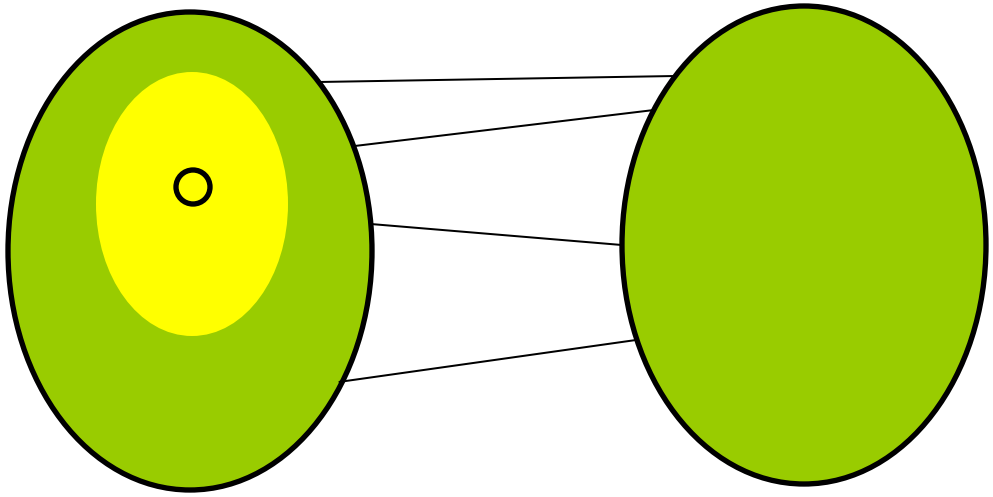
Testing k-Clusterings

Main Idea

- Two random walks starting from the same cluster should eventually have almost the same distribution (and this is almost uniform on the cluster)
- Two random walks starting in different cluster should have different distribution

Obstacles

- We cannot test against the uniform distribution since we don‘t know the clusters
- We do not compare stationary distributions



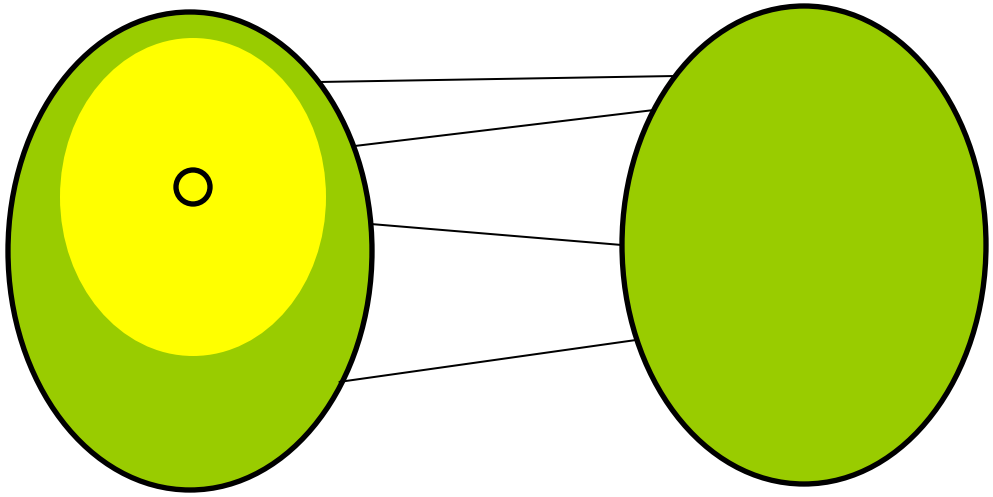
Testing k-Clusterings

Main Idea

- Two random walks starting from the same cluster should eventually have almost the same distribution (and this is almost uniform on the cluster)
- Two random walks starting in different cluster should have different distribution

Obstacles

- We cannot test against the uniform distribution since we don‘t know the clusters
- We do not compare stationary distributions



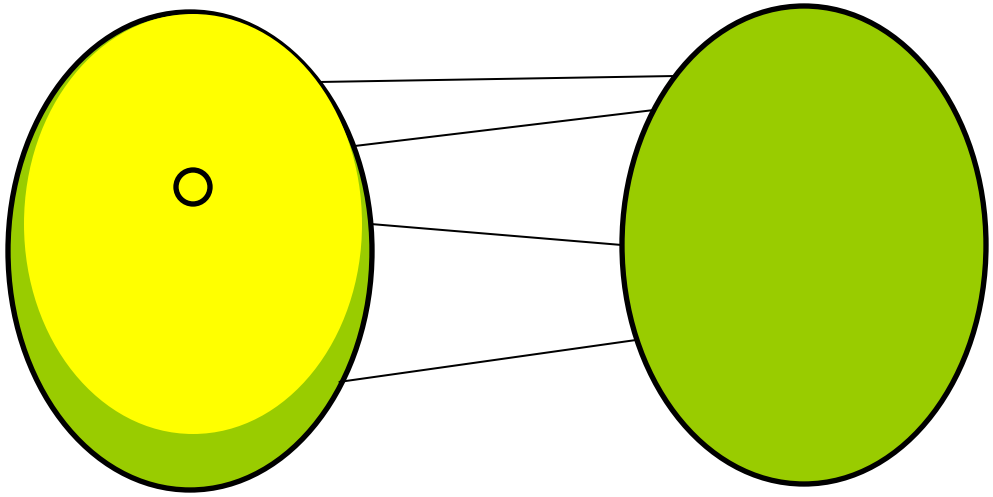
Testing k-Clusterings

Main Idea

- Two random walks starting from the same cluster should eventually have almost the same distribution (and this is almost uniform on the cluster)
- Two random walks starting in different cluster should have different distribution

Obstacles

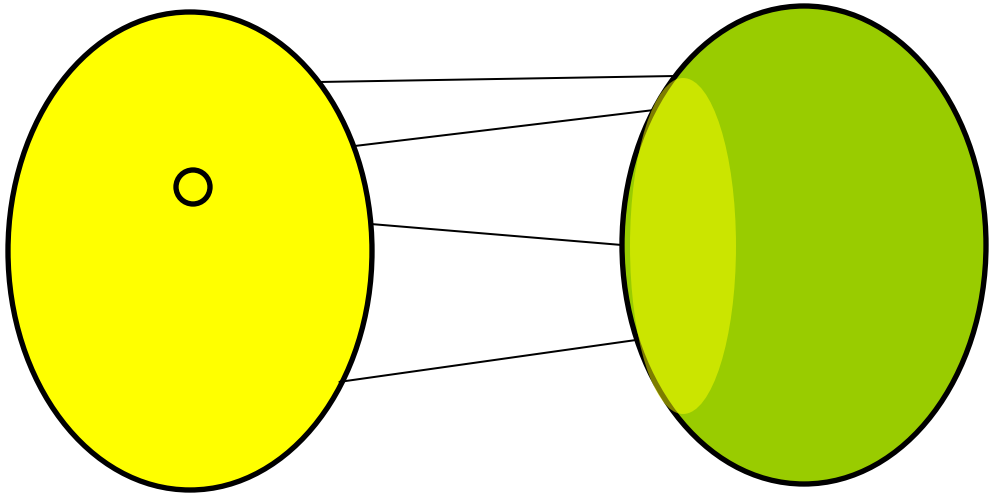
- We cannot test against the uniform distribution since we don‘t know the clusters
- We do not compare stationary distributions



Testing k-Clusterings

Main Idea

- Two random walks starting from the same cluster should eventually have almost the same distribution (and this is almost uniform on the cluster)
- Two random walks starting in different cluster should have different distribution



Obstacles

- We cannot test against the uniform distribution since we don‘t know the clusters
- We do not compare stationary distributions

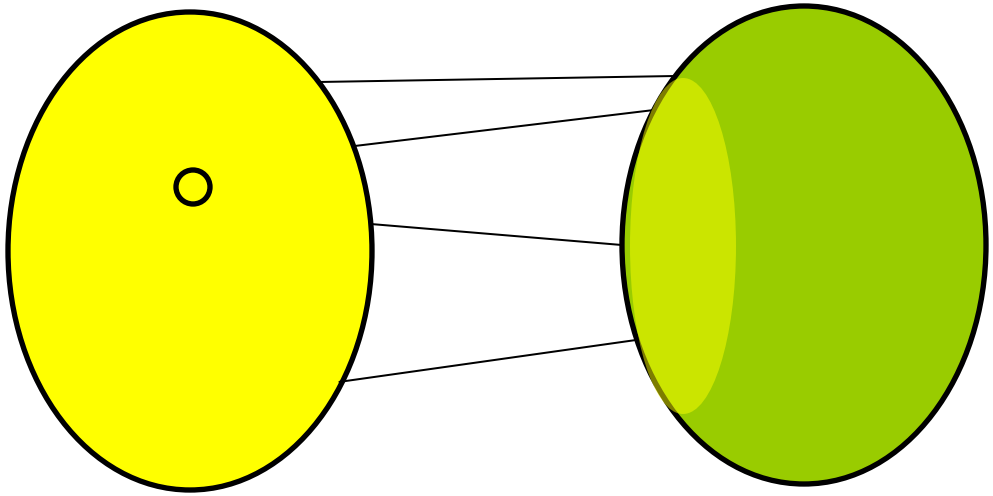
Testing k-Clusterings

Main Idea

- Two random walks starting from the same cluster should eventually have almost the same distribution (and this is almost uniform on the cluster)
- Two random walks starting in different cluster should have different distribution

Obstacles

- We cannot test against the uniform distribution since we don‘t know the clusters
- We do not compare stationary distributions



The Algorithm

ClusteringTest

- Sample set S of s vertices uniformly at random
- For any $v \in S$ let D(v) be the distribution of end points of a random walk of length $\Theta^*(\log n)$ starting at v
- **for each** pair $u,v \in S$ **do**
- **if** D(u) and D(v) are close **then** add an edge (u,v) to the „cluster graph“ on vertex set S
- **accept**, if and only if the cluster graph is a collection of at most k cliques

Completeness

Lemma (informal)

- Let p(v) denote the distribution of the end point of a random walk of given length. For our choice of parameters,

  (a) for most pairs u,v are from the same cluster C, $||p(v)-p(u)||^2 \leq 1/(4n)$,

  (b) for most pairs u,v are from different clusters, $||p(v)-p(u)||^2 > 1/n$.

Completeness

Lemma (informal)

- Let p(v) denote the distribution of the end point of a random walk of given length. For our choice of parameters,

  (a) for most pairs u,v are from the same cluster C, $||p(v)-p(u)||^2 \leq 1/(4n)$,

  (b) for most pairs u,v are from different clusters, $||p(v)-p(u)||^2 > 1/n$.

Consequence

- If we can estimate the distance of two distribution in sublinear time up to an l_2^2-error of 1/(4n), then ClusteringTest accepts any $(k, \Phi_{in}, \Phi_{out})$-clustering.

Completeness

Lemma (informal)

- Let p(v) denote the distribution of the end point of a random walk of given length. For our choice of parameters,

  (a) for most pairs u,v are from the same cluster C, $||p(v)-p(u)||^2 \leq 1/(4n)$,

  (b) for most pairs u,v are from different clusters, $||p(v)-p(u)||^2 > 1/n$.

Consequence

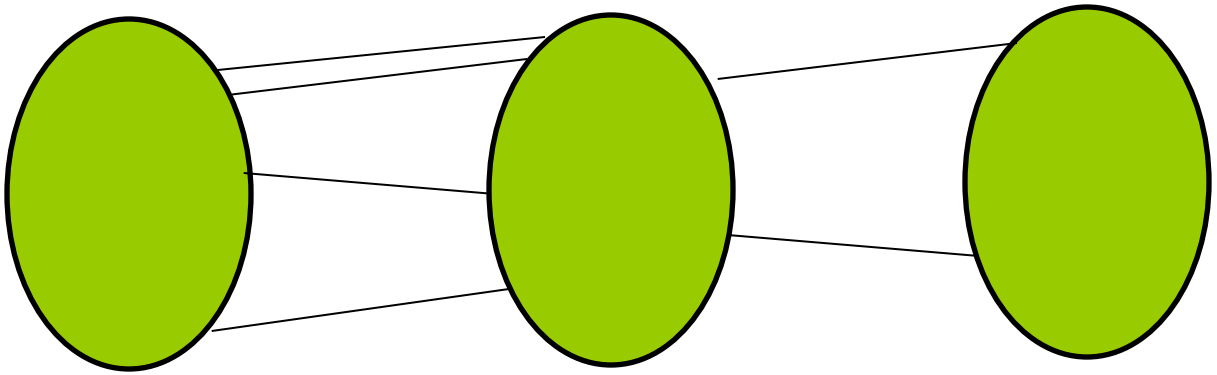
- If we can estimate the distance of two distribution in sublinear time up to an l_2^2-error of 1/(4n), then ClusteringTest accepts any $(k, \Phi_{in}, \Phi_{out})$-clustering.
- Can be done using previous work of [Batu et al.,2013] or [Chan, Diakonikolas, Valiant, Valiant, 2014]

Soundness

Lemma (informal)

- If G is ε-far from a (k, Φ_{in},*, Φ_{out} *)-clustering then one can partition V into k+1 subsets $C_1,\ldots,C_{k+1}$ of size $\Omega(\varepsilon n)$ such that $\Phi(C_i, V\text{-}C_i)$ is small for all i.

Example: ε-far from (2, Φ_{in},*, Φ_{out} *)-clustering



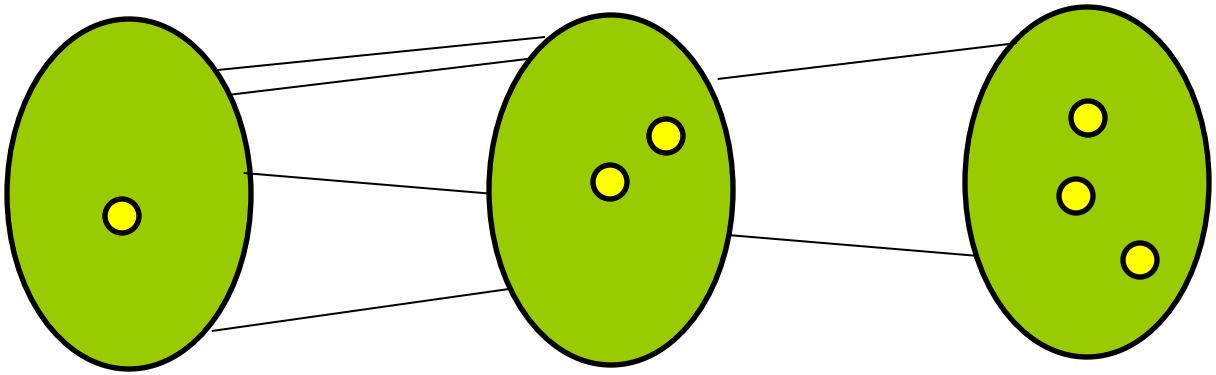
Soundness

Lemma (informal)

- If G is ε-far from a (k, Φ_{in},*, Φ_{out} *)-clustering then one can partition V into k+1 subsets $C_1,\ldots,C_{k+1}$ of size $\Omega(\varepsilon n)$ such that $\Phi(C_i, V\text{-}C_i)$ is small for all i.

Example: ε-far from (2, Φ_{in},*, Φ_{out} *)-clustering

Sample will hit all k1 subsets



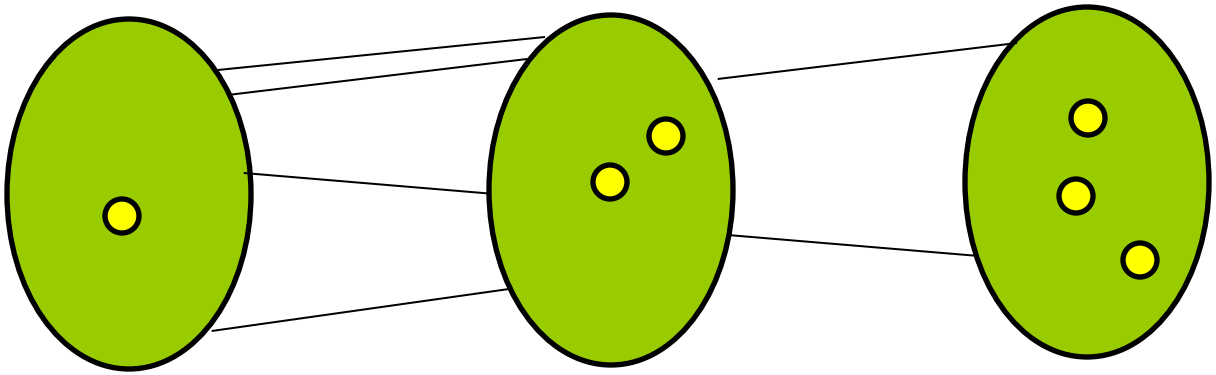
Soundness

Lemma (informal)

- If G is ε-far from a (k, Φ_{in},*, Φ_{out} *)-clustering then one can partition V into k+1 subsets $C_1,\ldots,C_{k+1}$ of size $\Omega(\varepsilon n)$ such that $\Phi(C_i, V\text{-}C_i)$ is small for all i.

Example: ε-far from (2, Φ_{in},*, Φ_{out} *)-clustering

Distance between vertices from different clusters is big



Summary

Theorem

- Algorithm ClusteringTester accepts every $(k, \Phi_{in}, \Phi_{out})$-clustering with probability at least 2/3 and rejects every graph that is ε-far from every $(k, \Phi_{in}^*, \Phi_{out}^*)$-clustering with probability at least 2/3, where $\Phi_{out} = O(\varepsilon^4 \Phi_{in}/\log^2 n)$ and $\Phi_{in}^* = \Theta(\varepsilon^4 \Phi_{in}/\log^2 n)$.
- The running time of the algorithm is $O^*(\sqrt{n})$.