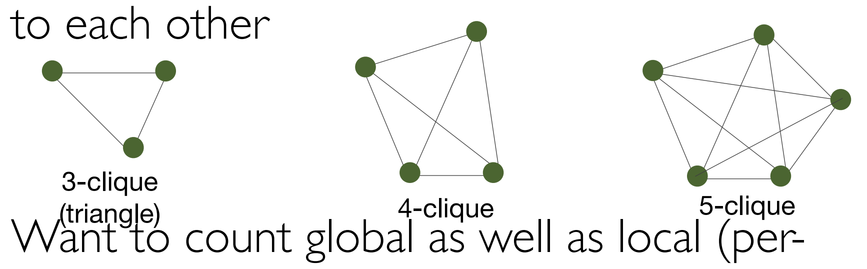
Shweta Jain
University of Illinois, Urbana-Champaign

Counting cliques in real-world graphs

C. Seshadhri
University of California, Santa Cruz

Problem Definition

- Given an undirected, simple graph G, we want to **count the number of k-cliques in G, for all k**
- A k-clique is a set of k vertices, all connected to each other



- Want to count global as well as local (per-vertex and per-edge) k-cliques
- As k increases, number of k-cliques increases exponentially. Enumeration not feasible. Existing methods only work for k <= 5
- **Applications:** Dense subgraph discovery, graph analysis and modeling, community detection, graph visualization

Our Contribution

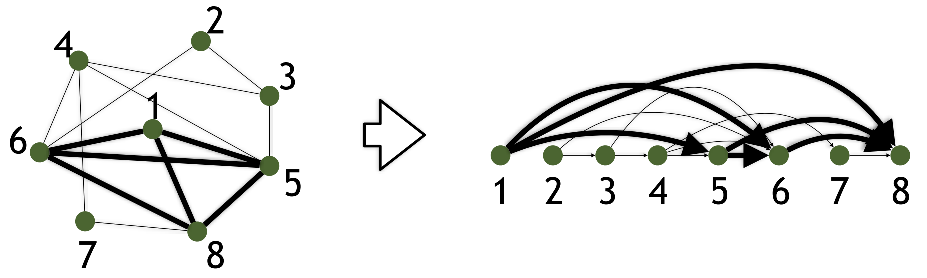
- We present two algorithms, **TuránShadow (Best Paper, WWW 2017)** and **Pivoter (Best Paper, WSDM 2020)** for counting k-cliques in large, real-world graphs
- **TuránShadow**:
  - Sequential, randomized, approximation algorithm that gives global counts
  - Works very well for **k upto 10**
  - **>100x** speedup with excellent accuracy
- **Pivoter:**
  - Sequential, **exact** clique counting algorithm
  - Works for **all** k
  - Gives **global** as well as **local** counts
  - Extremely fast (**>1000x** speedup over all other methods, including **parallel** algorithms)
  - Improved runtime of clique counting from $O(2^n)$ to $O(n3^{n/3})$ (first improvement in decades!)

Related work

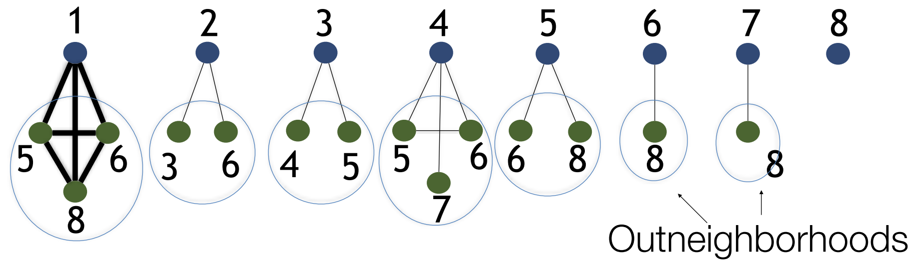
- Clique Enumeration, GRAFT, Edge Sampling, Color Coding, Path Sampling, ESCAPE (most built on clique enumeration)
- Typically, **only work for k<=5**

Background

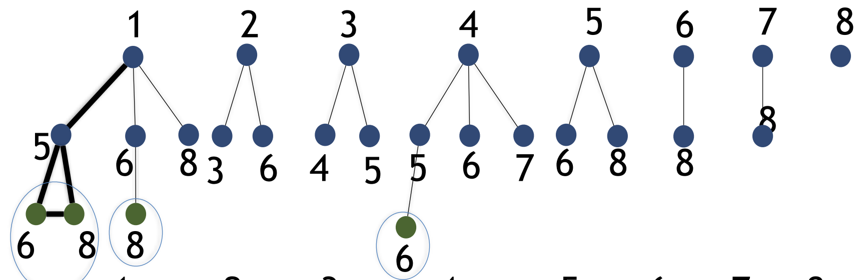
- **Clique Enumeration** [Chiba, Nishizeki, 85]
  - Choose an ordering and convert graph to directed acyclic graph (DAG) - edges go from lower to higher vertices

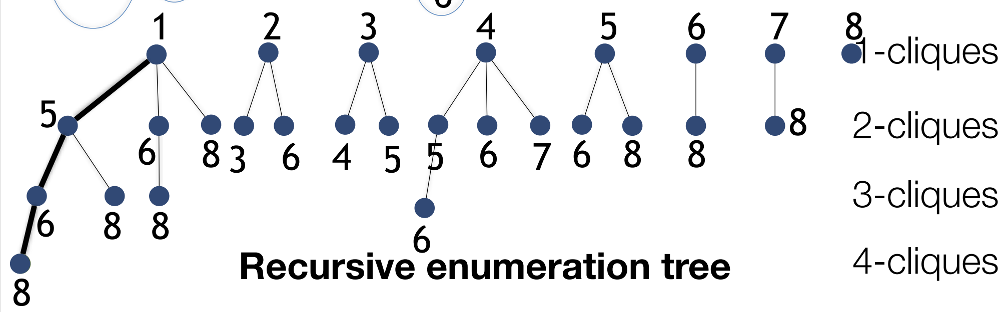


Counting k-cliques in G = counting k-1-cliques in **out-neighborhood** of each vertex.



- Same problem as before. Recurse!





Recursive enumeration tree

- Every blue node represents a recursive call and path from root gives a unique clique in graph
- Total recursive calls = total cliques in graph
- Infeasible for real-world graphs (10^{40} k-cliques!)
- Can we count without enumerating?

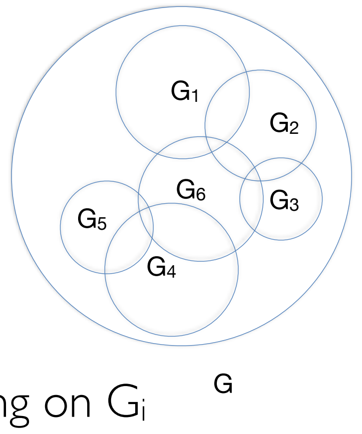
TuránShadow

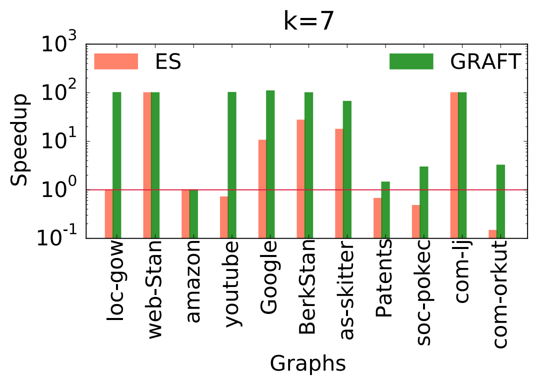
Best Paper, WWW 2017

- Use sampling to overcome combinatorial explosion
- **Random sampling:**
  - Sample multiple k-vertex sets, check what fraction of sets are cliques, scale to total number of k-vertex sets
  - Gives unbiased estimate of global number of k-cliques
  - Fails because it requires prohibitively large number of samples for sparse graphs

Turán's theorem: If a graph on n vertices has edge density > $(1-(k-1)^{-1})$, it has at least $n^{(k-2)}$ k-cliques.

- If graph G is Turán-dense (i.e. has edge density > Turán density for given k), random sampling works. Can bound number of samples required and guarantee accuracy
- But real-world graphs not dense enough for large k
- **TuránShadow**:
  - Decompose graph into smaller (possibly overlapping) subgraphs with following properties:
  - For each subgraph G_i,
  - there is an integer k_i
  - Each G_i **is**
  - Turán-dense for k_i
  - There is a **bijection**
  - between all k_i cliques
  - in G_i and k-cliques in G
  - Perform random sampling on G_i



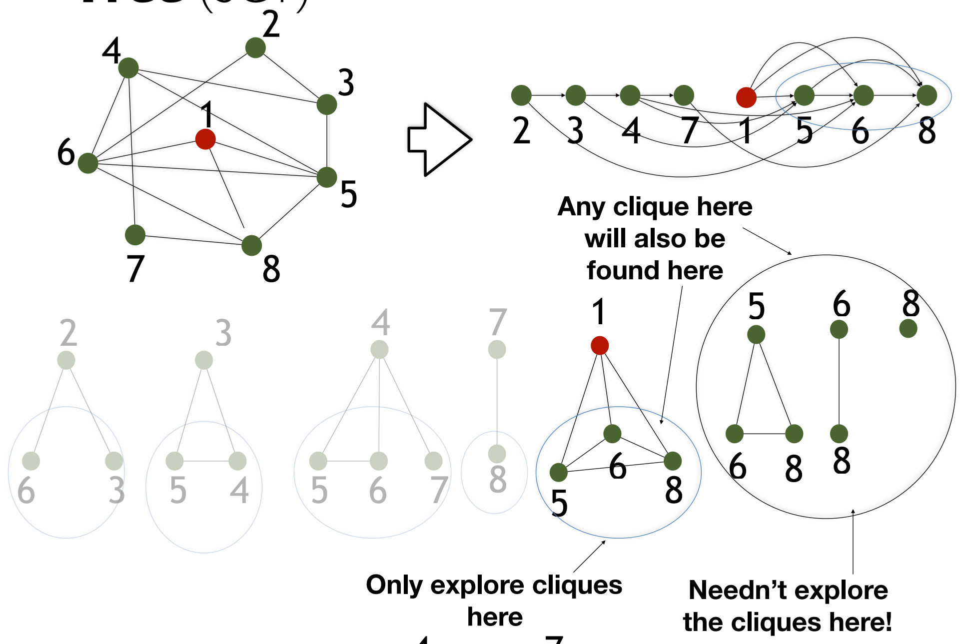


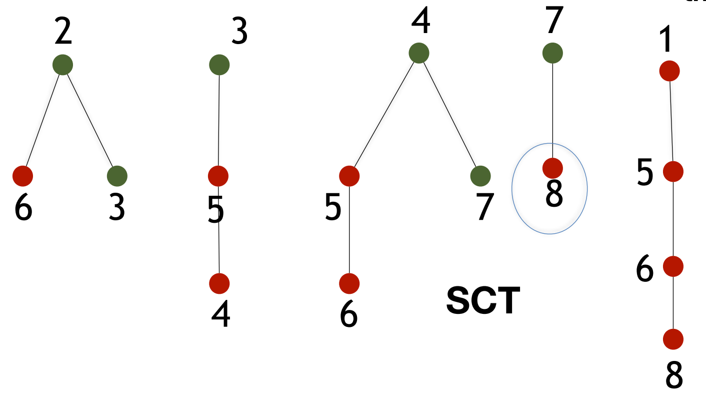
- Red line indicates speedup of one
- >100x speedup even just for k=7

Pivoter

Best Paper, WSDM 2020

- Uses an idea called **pivoting** (first used in maximal clique enumeration in [Bron, Kerbosch, 73])
- Defines a structure called **Succinct Clique Tree** (SCT)





- **Size of SCT << size of enumeration tree.**

Cliques encoded in tree <=> cliques in graph

Pivoter can count all cliques in $O(n3^{n/3})$.

- Once the tree is constructed, simply use **formulas** to get the cliques contributed by all the root to leaf paths
- >**1000x speedup** even over parallel/ approximate methods

Graph	Vertices	Edges	Max k	k=13, TS	k=13, kClist	all k, Pivoter
Stanford	0.2M	2M	61	230	12600	5
Berk-Stan	0.6M	6M	201	1198	> 172500	25
as-skitter	2M	11M	67	798	12480	120
orkut	3M	110M	51	> 28800	> 172500	5174

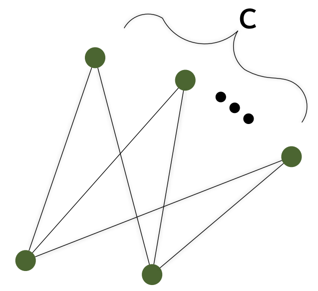
Shweta Jain and C. Seshadhri acknowledge the support of NSF Awards CCF-1740850, CCF-1813165, and ARO Award W911NF1910294.

Fixed parameter tractability

- Running time of Pivoter is exponential in n
- Yet, we are able to count cliques in graphs with millions of nodes and edges.
- What explains the performance of Pivoter?

c-closed graphs

- **c-closed graphs**: A graph is c-closed for an integer c if non-adjacent vertices have at most c common neighbors.



- Inspired by the fact that when two members in a social network have many common acquaintances, it is unlikely that they will not know each other.
- c for real-world graphs is small
- Pivoter and Clique Enumeration are fixed parameter tractable with respect to c.

When the vertices are ordered by degeneracy, outneighborhoods are either small or dense.

Running time of Pivoter as well as Clique Enumeration is $O(c!n^2)$

- Which other problems are fixed parameter tractable on c-closed graphs?

sjain12@ucsc.edu