SAND2014-16937PE

Sandia National Laboratories

DARPA

Fast Algorithms for Evolving Graphs via Assays, Sampling & Theory (FEAST)

Team: Tamara G. Kolda (PI), Ali Pinar, C. "Sesh" Seshadhri, Rick McCormick (PM), Madhav Jha, Jon Berry, Cindy Phillips, Todd Plantenga

U.S. DEPARTMENT OF ENERGY

Sandia National Laboratories is a multi-program laboratory managed and operated by Sandia Corporation, a wholly owned subsidiary of Lockheed Martin Corporation, for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

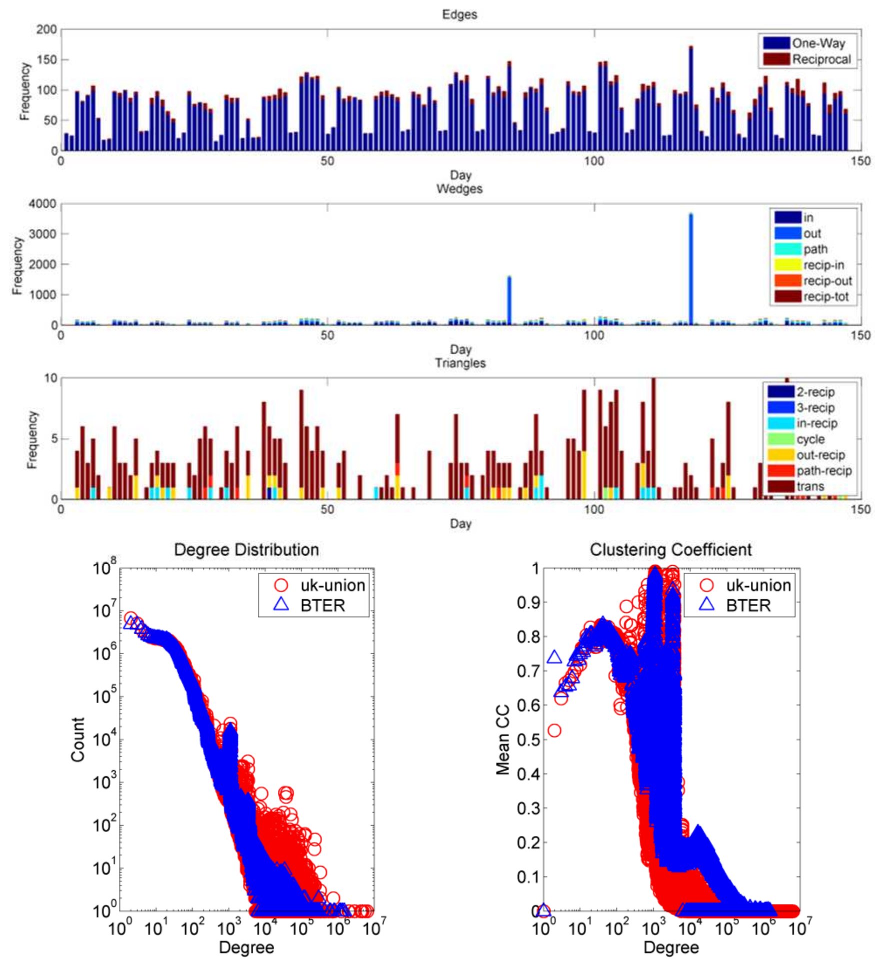
FEAST Objectives

<u>Objective</u>: Measure, reproduce, and exploit quantifiable characteristics of real-world graphs

1. Graph Assays: Sampling-based techniques to compute frequency of common patterns in large-scale graphs

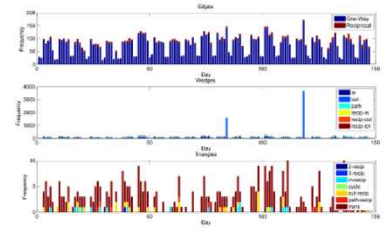
2. Generative Models: Reproduce characteristics shown in the assays

3. Efficient Algorithms: Take advantage of structure of social networks



FEAST's edge is Full Circle Approach

Sandia National Laboratories



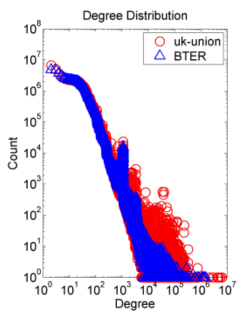
Capability

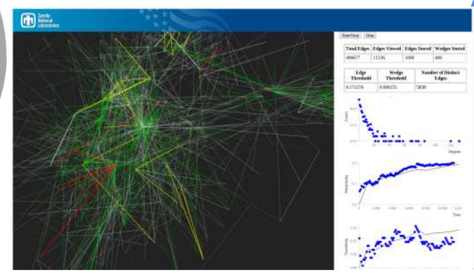
Theory

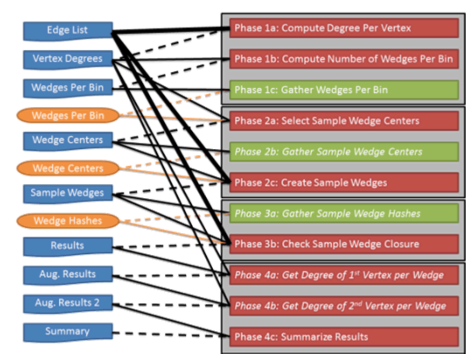
Algorithm

Implemen-
tation

Validation





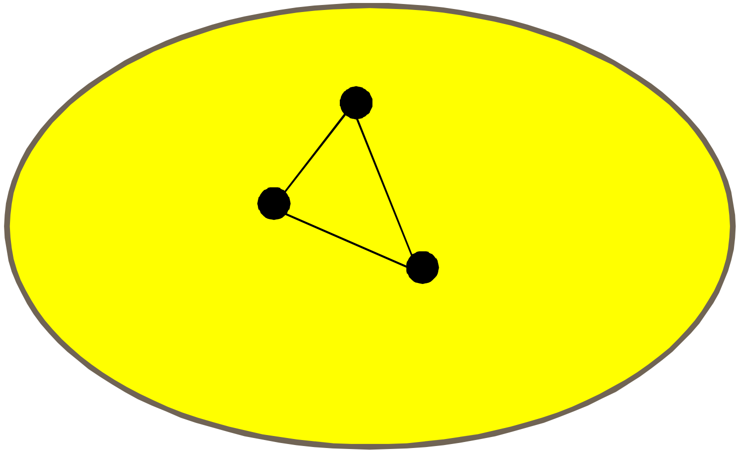


Dealing with real-world graph streams

M. Jha, A. Pinar, and C. Seshadhri***, Path Sampling: A Fast and Provable Method for Estimating 4-Vertex Subgraph Counts,*** Submitted for conference publication***.***

One ring to rule them all 

Sandia National Laboratories

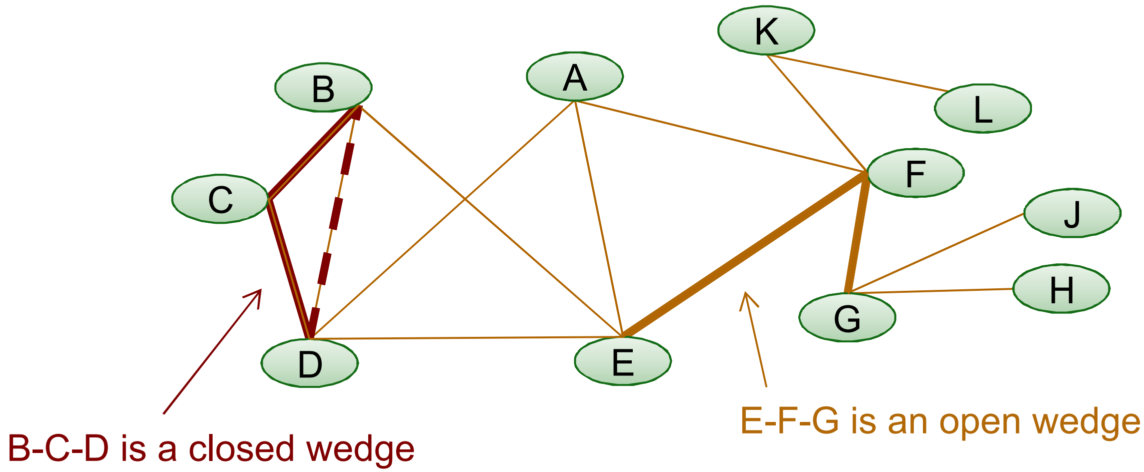


Graph [SNAP, LAW]	n	m	T
web-BerkStan	700K	6.6M	64M
flickr	1.8M	15M	550M
livejournal	5.2M	48M	300M
uk-union	132M	4B	450B

The smallest "community". Friends of friends are friends. Two of your friends are often friends.

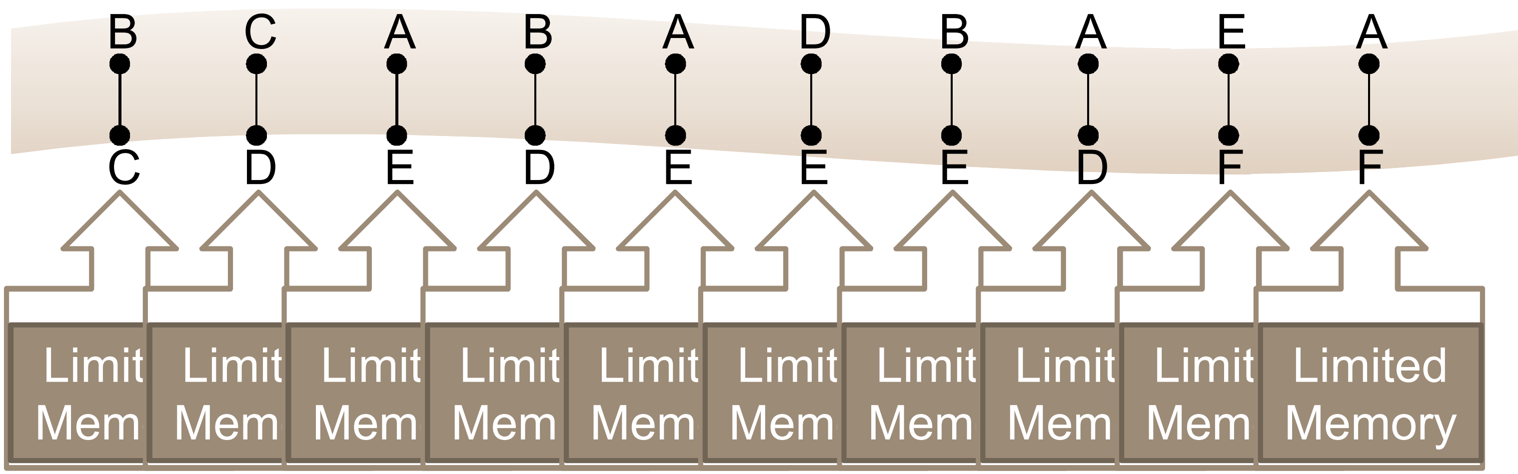
Why you should care about triangles in your graph: [Holland-Leinhardt70] [Coleman88] [Skvoretz90] [Portes98] [Watts-Strogatz98] [Eckmann-Moses02] [Burt04] [Fagiolo07] [Welles etal10] [Faust10] [Szell-Thurner10] [Milo etal10] [Son etal10] [Leskovec etal10] [Winkler-Reichardt13]

Triangle information

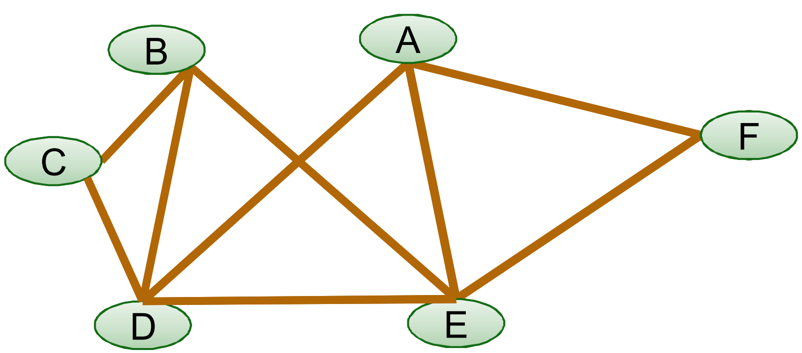


- W = no. of wedges (paths of length 2)
  - "Center" of wedge is middle vertex
- T = no. of triangles
- Transitivity = τ = 3T/W = fraction of closed wedges

Streaming Triangle Counting

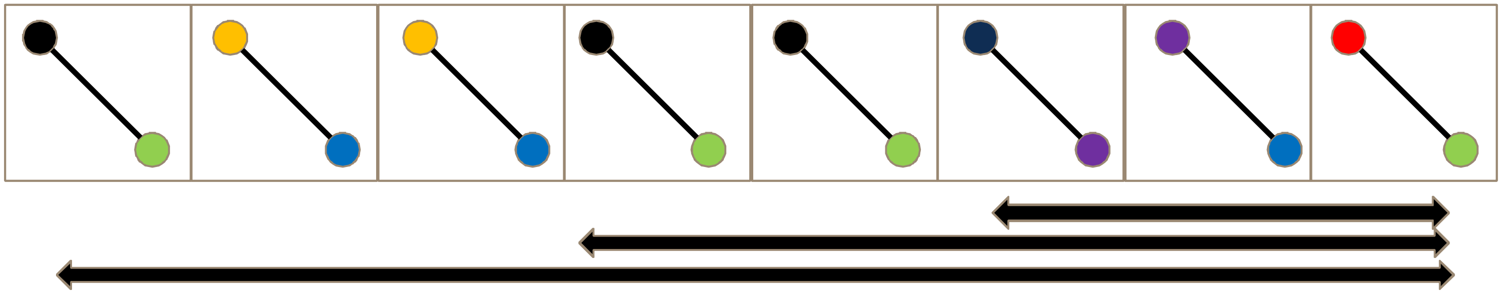


Triangles so far: 4
Graph seen so far:



- Data streams important for situational awareness
  - Streaming algorithms also useful for large data sets
- Algorithmically
  - See each edge only once
  - Either take action or lose that piece of information forever

Real-world messiness

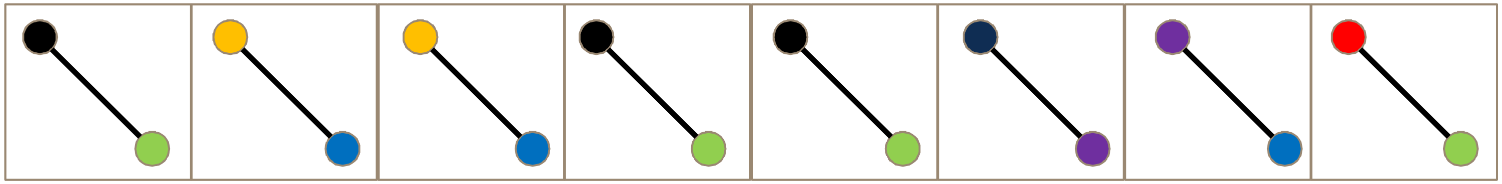


- Real-world streams are multigraphs: edges can be repeated
  - Consider communication network. Obvious repeats
- There is no true “graph”. It depends on how you aggregate
  - Different time intervals give different graphs

Standard approaches

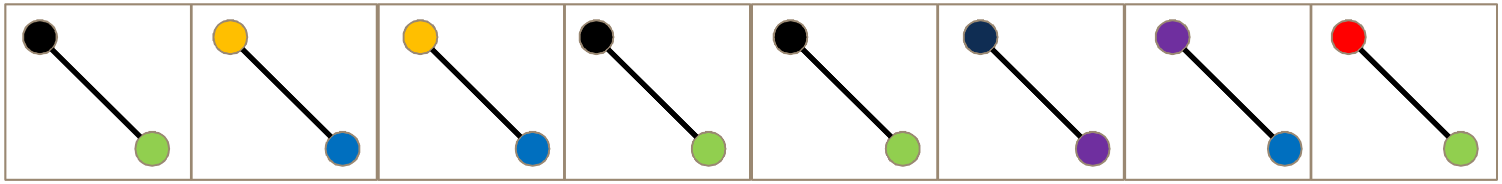
- There are no repeats. Assume graph is simple
- Aggregate every edge seen. The “window” is all of history

Drawbacks of ignoring repeats



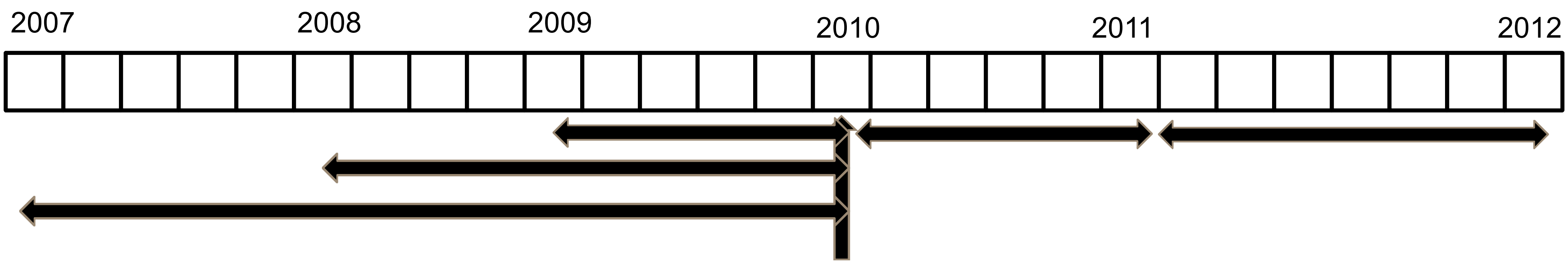
- Assumptions useful for algorithmic progress, but avoids real-world complexities
  - Algorithms cannot be deployed in “wild”
- Removing repeated edges requires extra pass over edges
  - Assumption of no repeats is expensive to enforce
- Not clear how to store information of various time-windows simultaneously

Our results



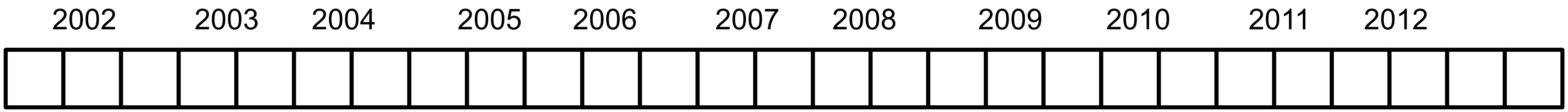
- Algorithm for approximating triangle counts and transitivity in graph stream with repeated edges
  - No preprocessing. Works with raw stream
- Maintain information on multiple time windows with same data structures
- Provable bounds on accuracy, excellent empirical behavior
- Based on methods in [Jha-Seshadhri-Pinar13], but needs new ideas to overcome issues

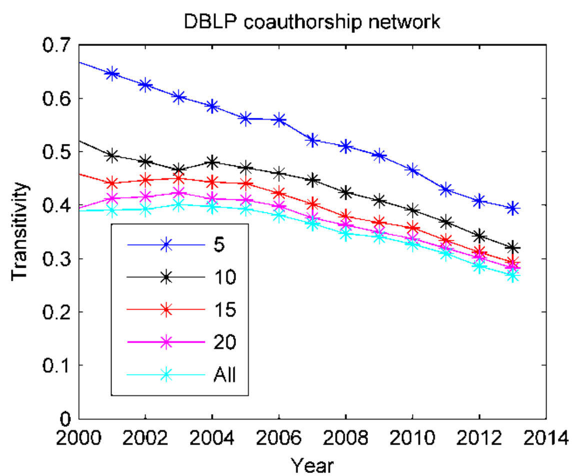
Case study: DBLP graph

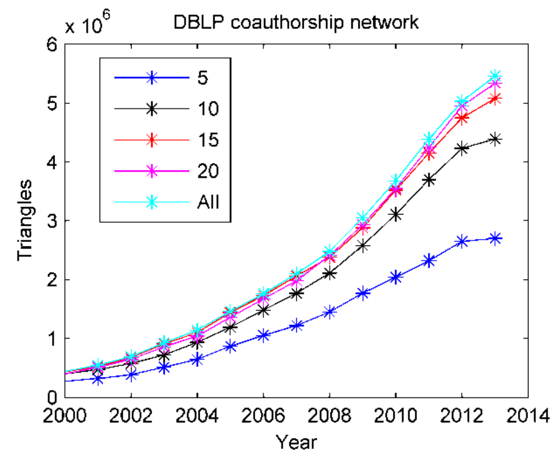


- DBLP co-authorship graph: all paper records over 50 years gives graph stream
  - Naturally repeated edges. Colleagues work together for many papers
  - Size = 3600K, non-repeated edges = 254K
- For graph G[t:t+Δt], there is associated transitivity and triangle count
  - How does this vary with t and Δt?

Triangle trends in DBLP graph

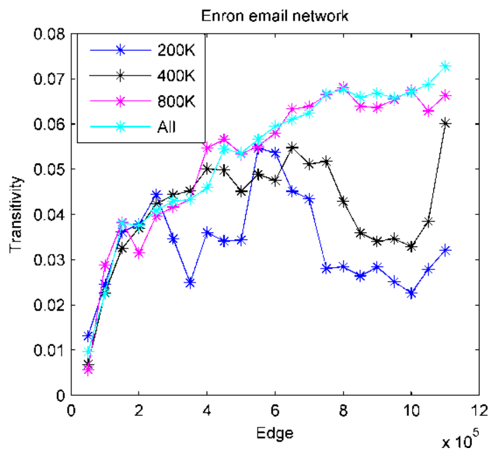


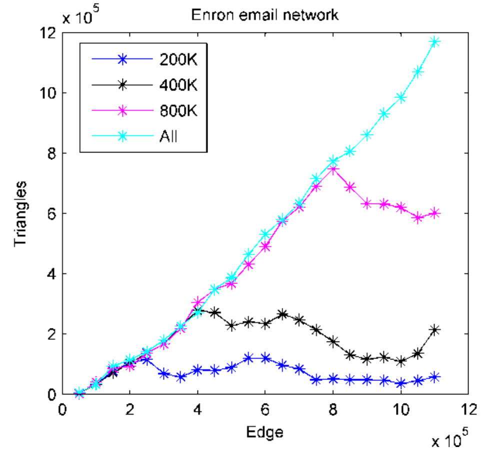




- Size = 3600K, non-repeated edges = 254K
- Results obtained with storing 30K edges

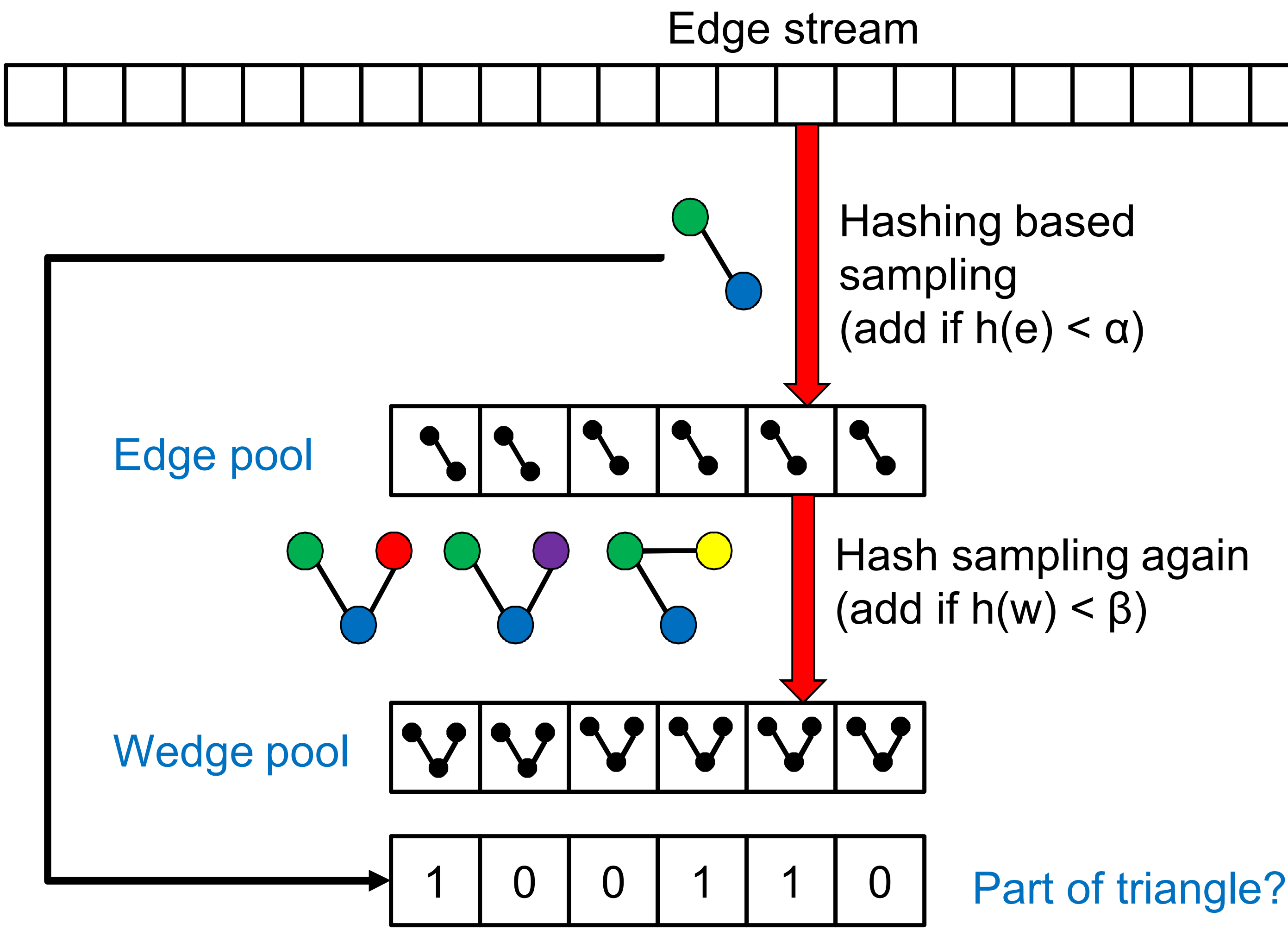
Triangle trends in Enron graph





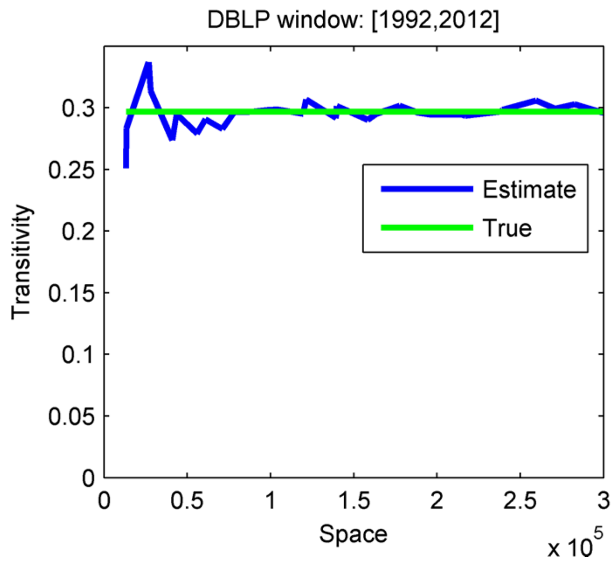
- Enron email network: stream size 1100K, non-repeated 300K
- Storage used = 8K
- Trends "opposite" to DBLP graph

Algorithm Sketch



Streaming Algorithm Features

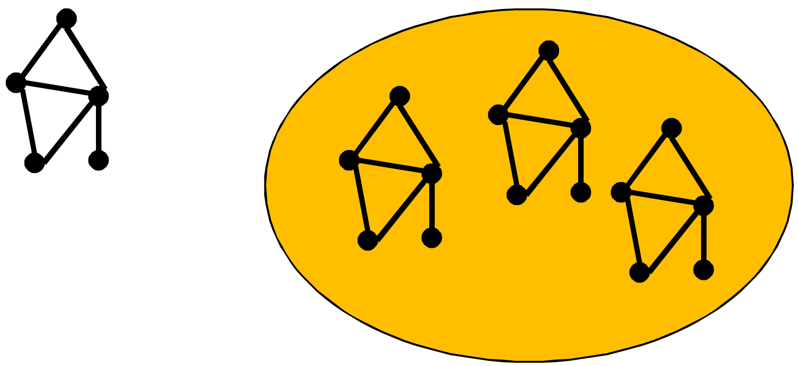
- Only two parameters α, β
  - No knowledge of graph required
- Provable guarantee on expectation
  - Provable variance bound (though not useful in practice)
- Space around 1% of total stream
- Accuracy always within 5%



Pushing the boundaries of pattern counting

M. Jha, A. Pinar, and C. Seshadhri***, Path Sampling: A Fast and Provable Method for Estimating 4-Vertex Subgraph Counts,***
Submitted for conference publication

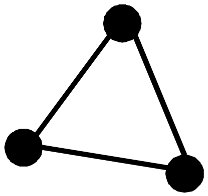
The importance of pattern counting

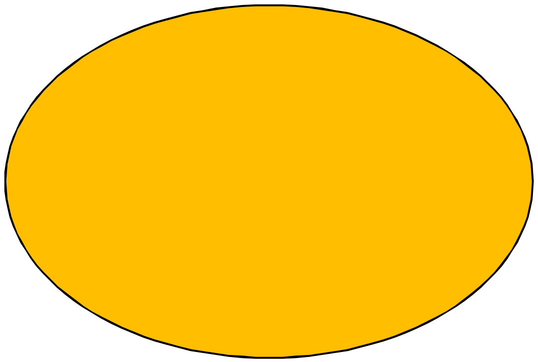


- Find the count of some subgraph in G
- Quantifiable method to "measure" graphs, compare them, categorize them
- Understand global properties of network from local measurements
- Extensively used in bioinformatics, social networks, cybersecurity

- Called graphlets, motifs, subgraph isomorphism (SGI), pattern counting

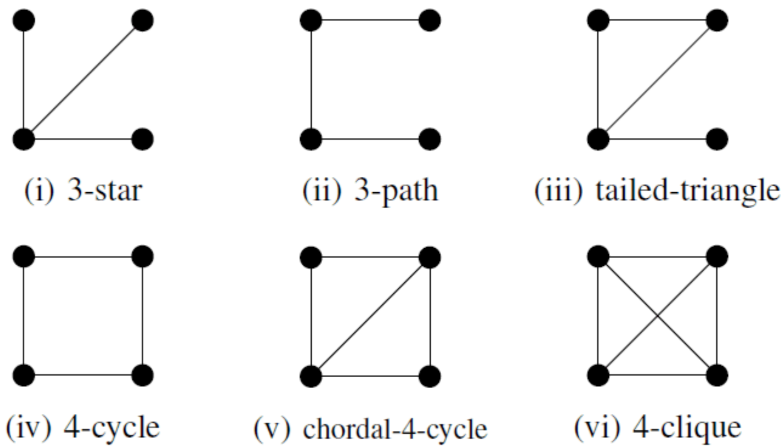
Triangle counting





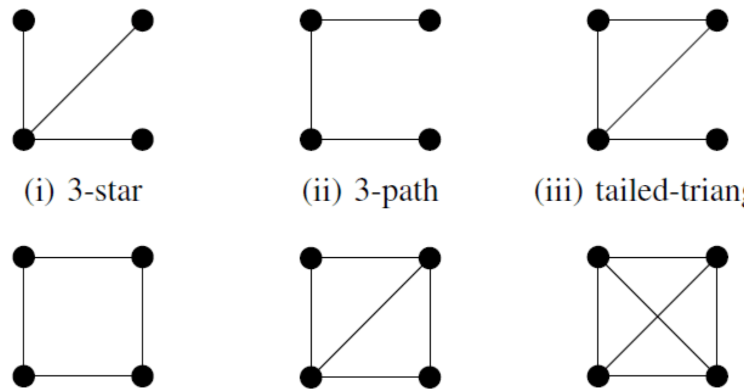
- The standard, classic, well-studied problem
  - Past work too long to list here ☺
- Summary: tractable, good provable methods that work in practice
- Can scale to billions of edges (with right hardware)
- Haven't found a graph where we can't count triangles

Beyond 3 vertices: how about 4?



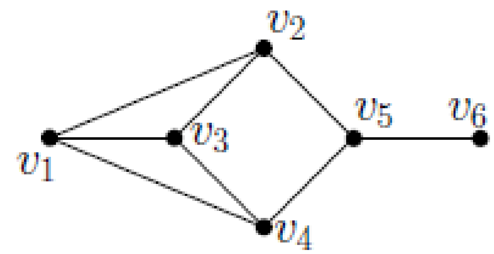
- Much richer set of (connected) patterns
- Are there any trends in pattern counts?
  - What does it signify?

Induced vs non-induced



(i) 3-star (ii) 3-path (iii) tailed-triangle

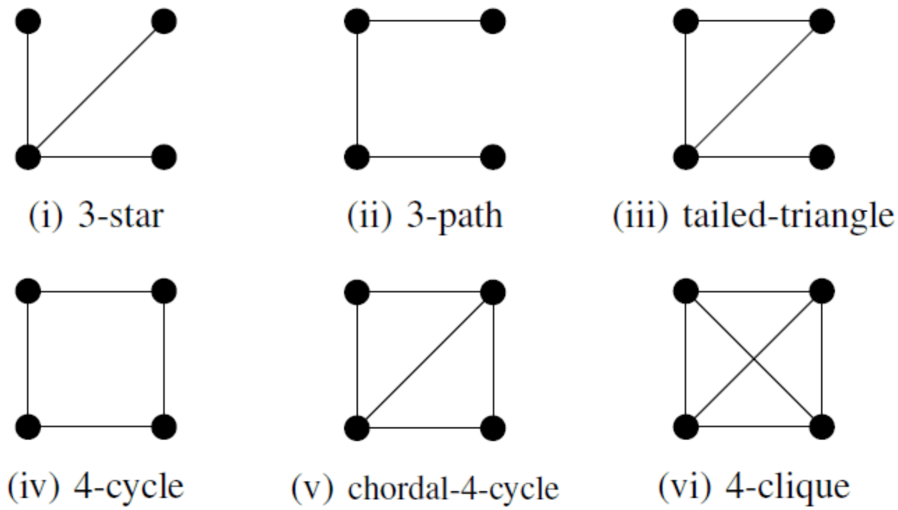
(iv) 4-cycle (v) chordal-4-cycle (vi) 4-clique



$$\begin{pmatrix} 1 & 0 & 1 & 0 & 2 & 4 \\ 0 & 1 & 2 & 4 & 6 & 12 \\ 0 & 0 & 1 & 0 & 4 & 12 \\ 0 & 0 & 0 & 1 & 1 & 3 \\ 0 & 0 & 0 & 0 & 1 & 6 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} C_1 \\ C_2 \\ C_3 \\ C_4 \\ C_5 \\ C_6 \end{pmatrix} = \begin{pmatrix} N_1 \\ N_2 \\ N_3 \\ N_4 \\ N_5 \\ N_6 \end{pmatrix}$$

- (Vanilla) subgraph: take subset of edges
- Induced subgraph: take subset of vertices, take all edges in them
- Let C_i is induced count of pattern i
  - Getting vanilla counts not hard

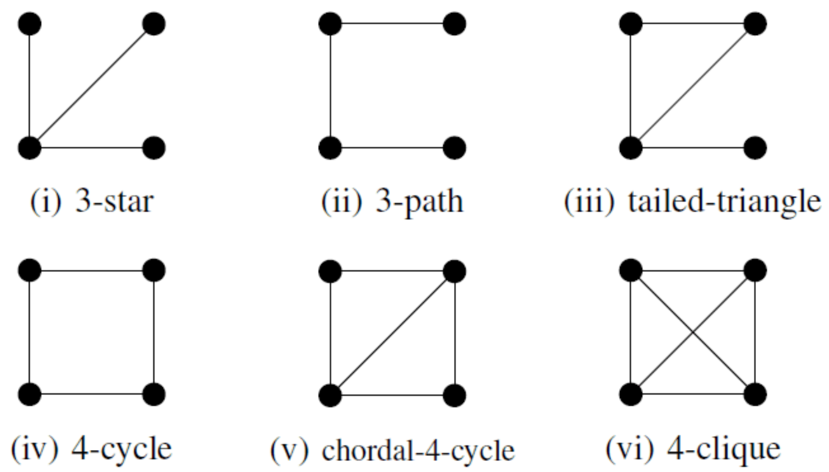
Do not enumerate



(i) 3-star (ii) 3-path (iii) tailed-triangle
(iv) 4-cycle (v) chordal-4-cycle (vi) 4-clique

Graph	n	m	3-path	Tail-tri	4-cycle	4-clique	Time
Web-Berk	600K	6M	10B	400B	20B	1B	2 hrs
Flickr	1M	15M	7T	100M	100B	25B	60 hrs
Orkut	3M	200M	10T	1T	70B	3B	19 hrs

Past art

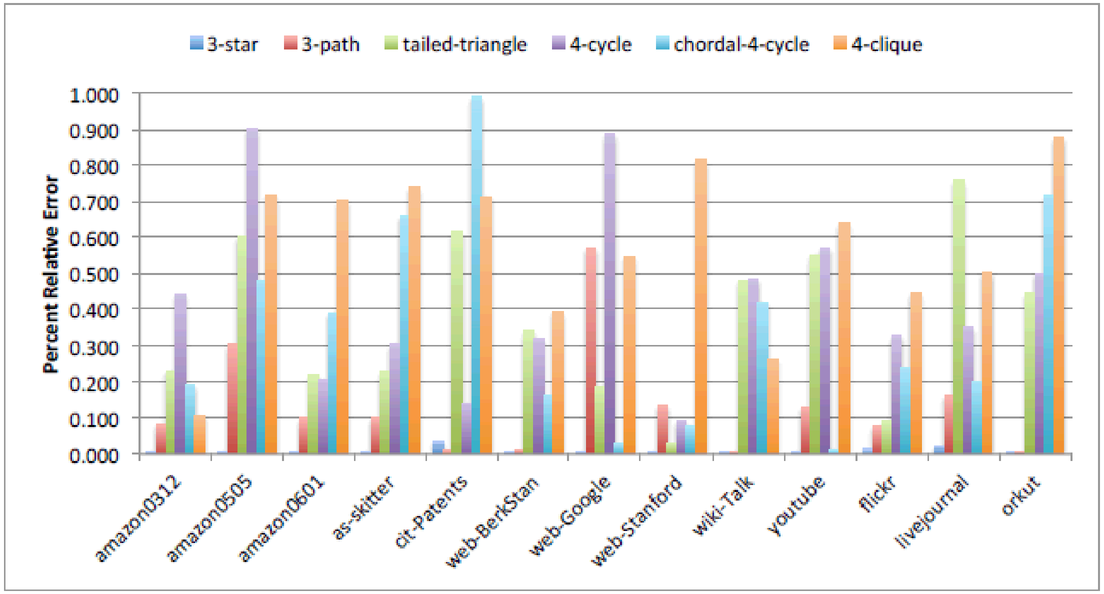


- MCMC methods, color coding, graph sparsification
- No provable methods, accuracies at best ~10%, often need clusters
- Nothing tailored for 4 vertices
- No results for (say) 100M edges

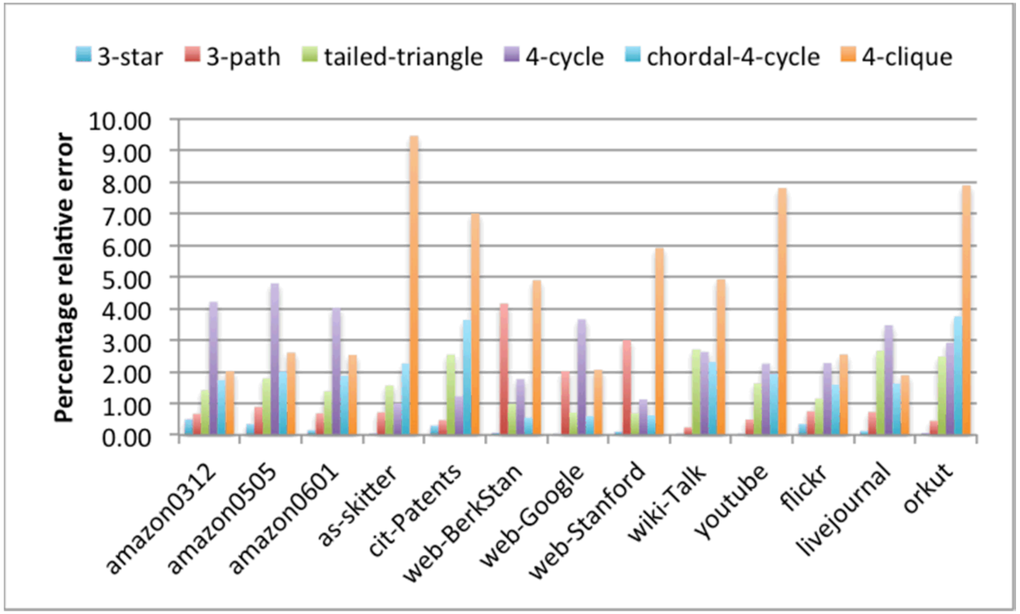
Our results

Graph	Time (enum)	Path-sampling
Web-Berk	2 hrs	3 sec
Flickr	60 hrs	2 sec
Orkut	19 hrs	16 sec

- New algorithm, based on 3-path sampling, for estimating all 4-vertex patterns
- Fast and accurate

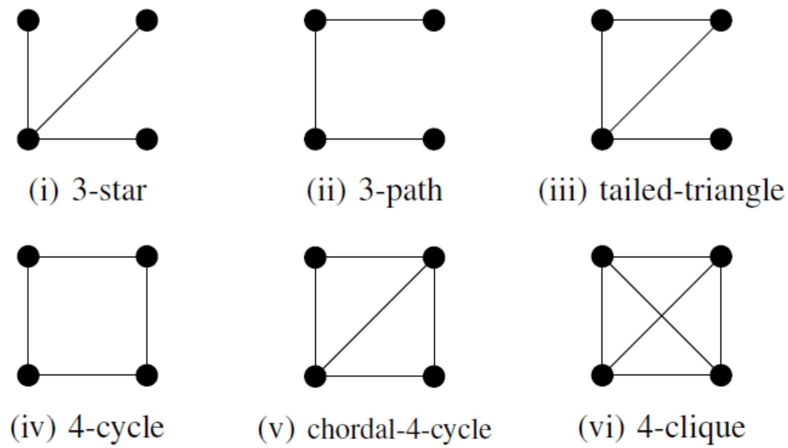


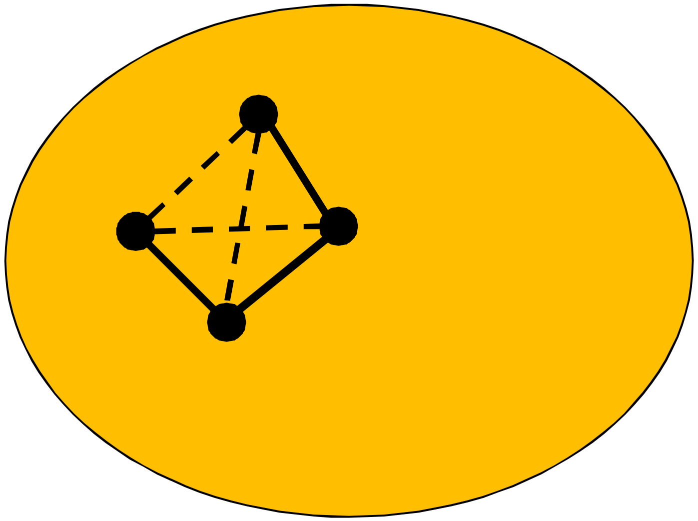
Mathematical basis



- Provable bounds on accuracy
- Algorithm outputs hard mathematical error bounds for any desired confidence
- "With confidence > 99.9%, the output is within 3% of true answer."

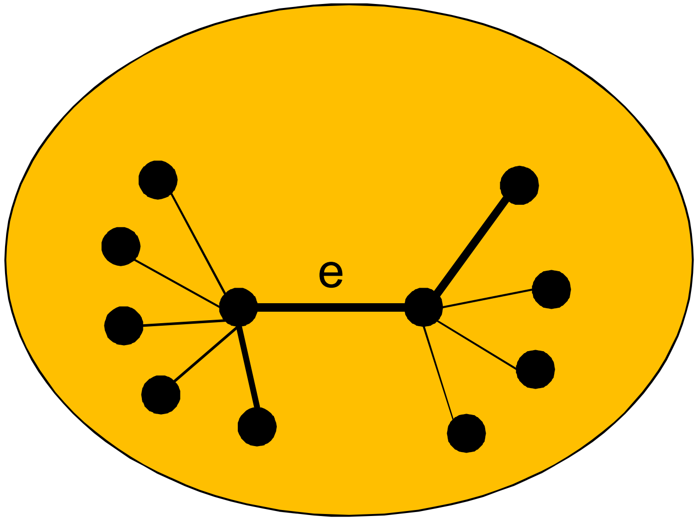
The method





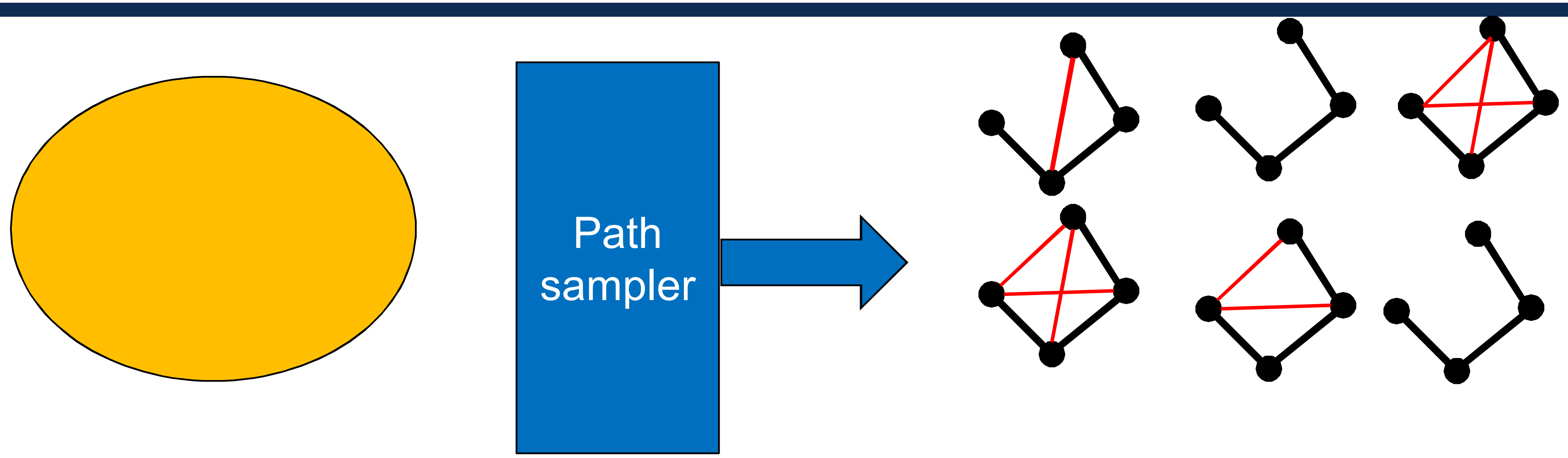
- Except for 3-stars, each pattern contains a 3-path
- Sample set of uniform random 3-path, check the vertices to see what pattern is induced
- Extrapolate these counts to get estimates

Sampling random 3-paths



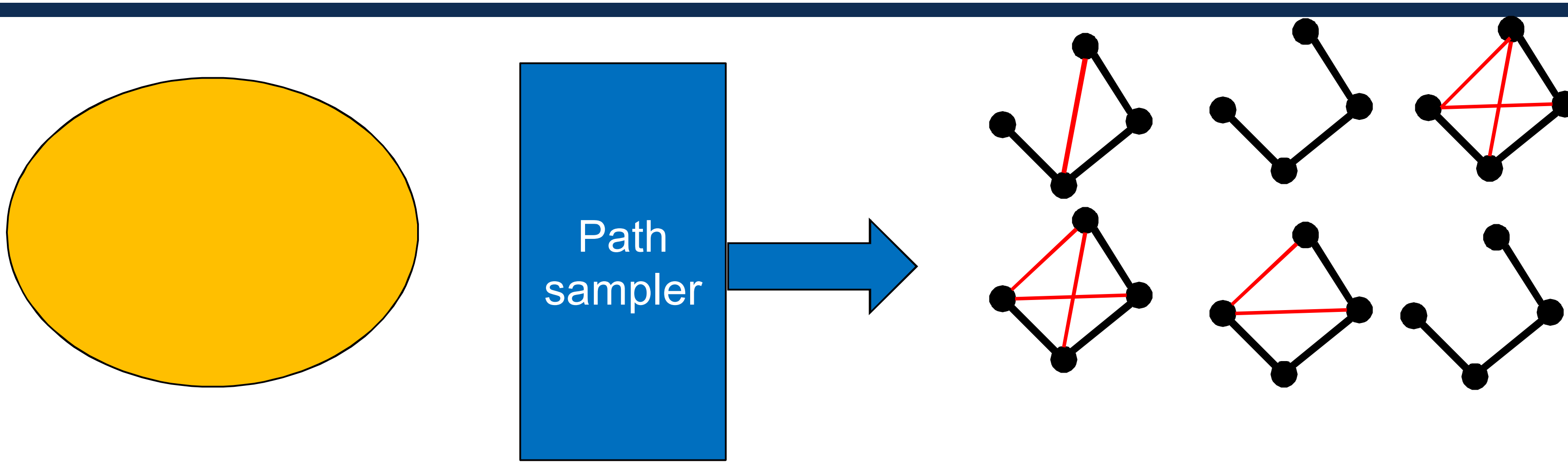
- First set for all edges, $W_{u,v} = (d_u - 1)(d_v - 1)$.
- Pick edge e = (u,v) with probability prop. to $W_{u,v}$
- Pick uniform random neighbor of u and of v
- If output is 3-path, guaranteed to be uniform random

The big picture



- Use pattern counts from samples to estimate true count
- Not hard to argue that our output is unbiased estimator of true count
- No assumption on graph, probability over randomness of algorithm
- How many samples needed to get accurate estimates?

The devilish details



- Works, but (provable) accuracy is not great
- Design methods to reduce samples
- Can give provable bounds: “for s samples, with 99.9% confidence, the true count is within 1% of answer”

Brain networks

Graph	n	m	3-star	3-path	Tail-tri	4-cycle	Chord-cycle	4-clique
KKI-21_KKI2009-01	800K	65M	1.2E+13	7.8E+12	4.8E+12	1.3E+11	7.8E+11	2.0E+11
MRN114_M87147006	661K	54M	4.6E+12	6.6E+12	3.6E+12	2.1E+11	6.9E+11	1.9E+11
NKI-TRT_4176156_2	466K	30M	4.8E+11	1.1E+12	8.0E+11	2.1E+10	1.8E+11	7.0E+10

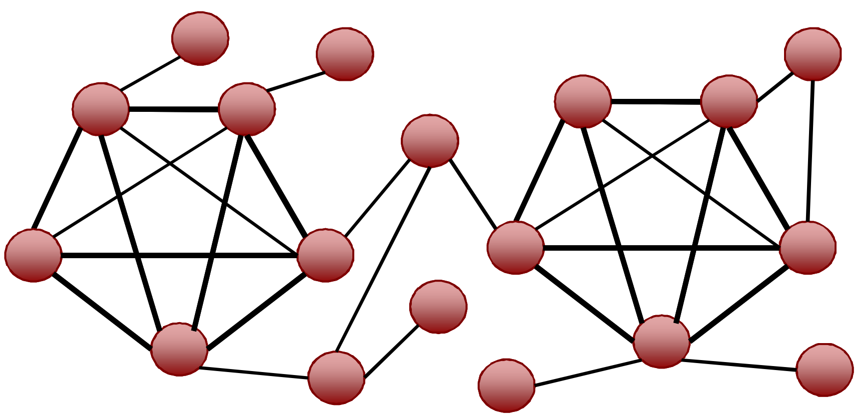
- Each run took less than 25 sec
  - Enumeration took hours
- Provable accuracies all within 3% with 99.9% confidence

Making community detection faster

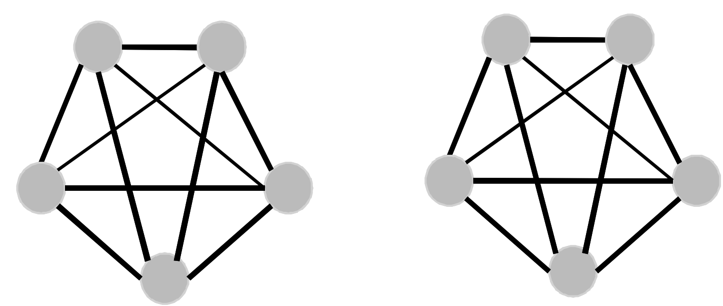
C. Peng, T. Kolda, and A. Pinar***, Accelerating Community Detection by Using k-core subgraphs,*** submitted for conference publication

Dense structures anchor communities

- Key Observations:
  - Low degree vertices do not have a significant impact on community structure.
  - Communities are built around dense regions.
  - Once these dense regions are identified, it is much easier to build the communities around these regions.
- Idea: reduce the size of the graph, so that dense cores are preserved, and then find the communities.
- Approach: work on the k-core of a graph.



Original Graph



3-core subgraph

How the proposed approach works

Original Graph

K-core of the graph

Linear-time computation

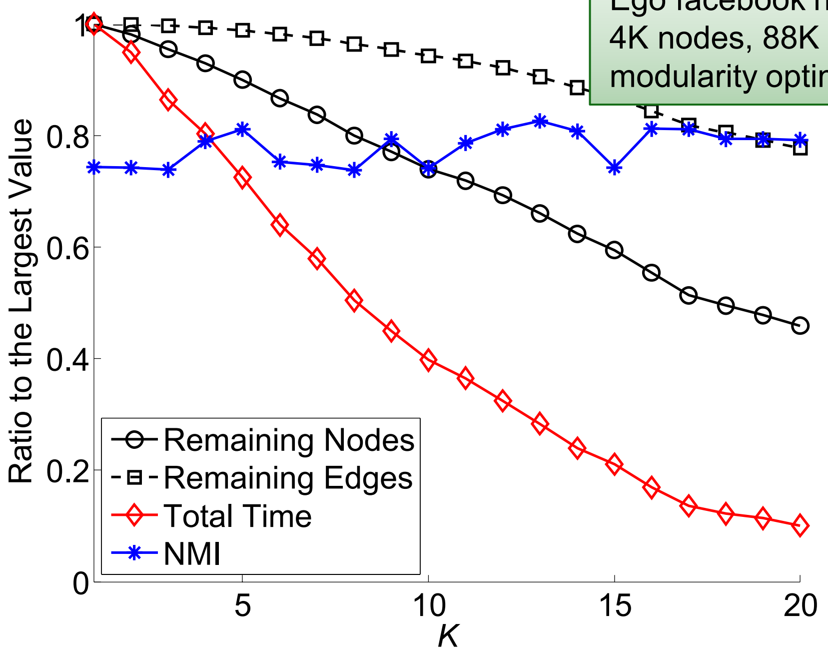
Favorite community finding algorithm

Communities of the k-core graph

Greedy,
Label propagation

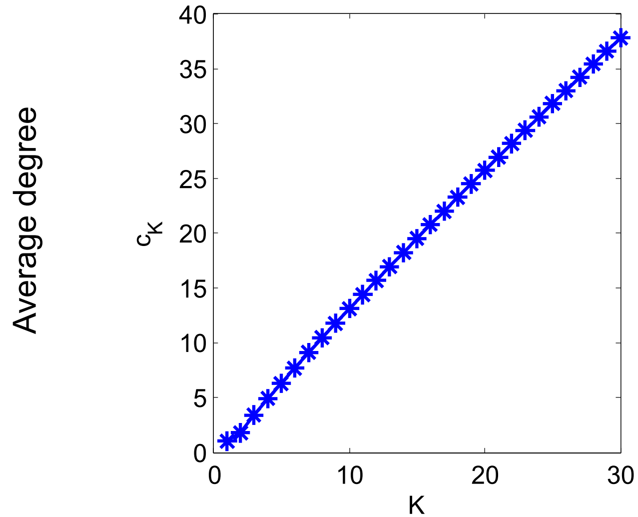
Communities of the original graph

Works well in practice, because ..



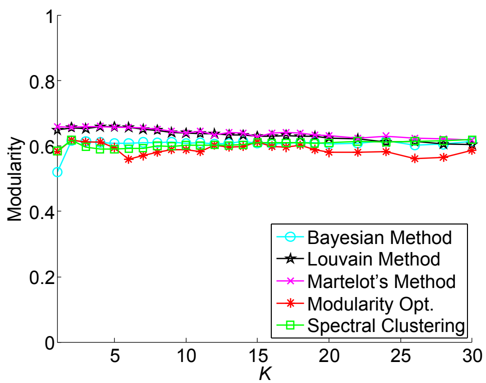
For the Erdős-Rényi random graph $G(n,m)$ on n vertices with m edges, and for $k \geq 3$, with high probability, a giant K-core appears suddenly when m reaches $c_K n/2$; here $c_K = \min_{\lambda>0} \lambda/\pi_K(\lambda)$ and $\pi_K(\lambda) = P\{\text{Poisson}(\lambda) \geq K-1\}$.

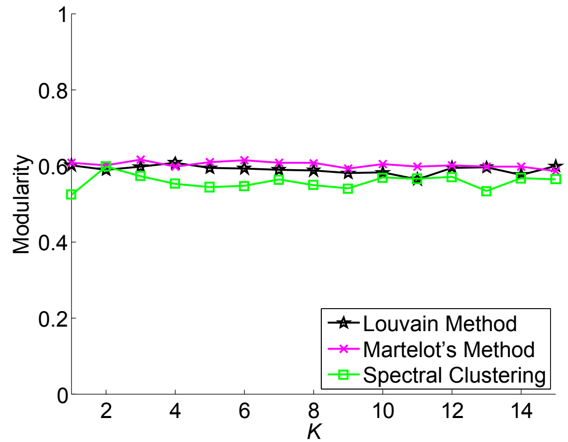
Pittel, Spencer, and Wormald
J. Comb. Theory Ser. B, 1996

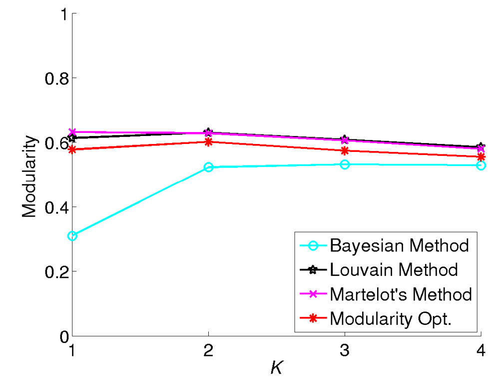


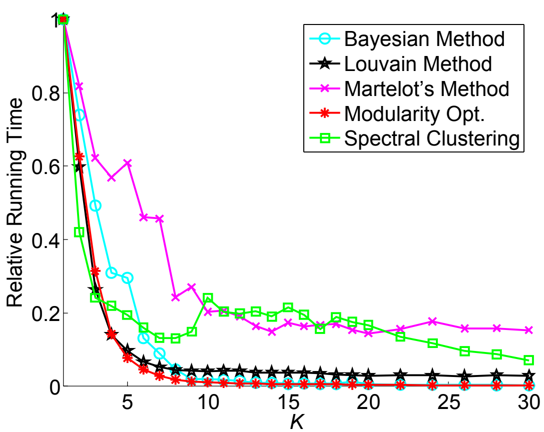
- Works well (y metric)
  - Quality of communities is preserved, even improved.
  - Runtime decreases dramatically.
- Because significantly dense structures are preserved after k-core reduction.
  - Significance depends on the average degree.
  - Corollary: the method will find bigger clusters, but will miss small cliques.

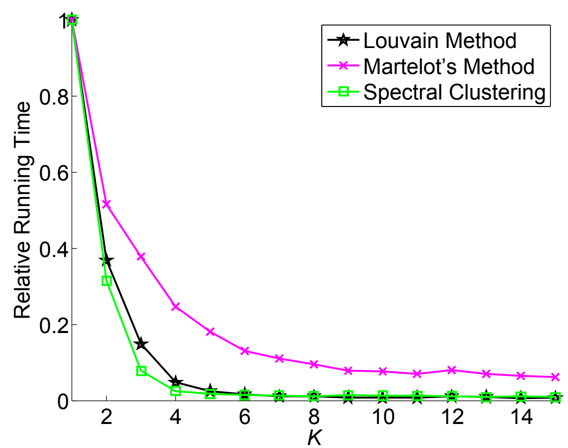
Good results on all graphs and all algorithms

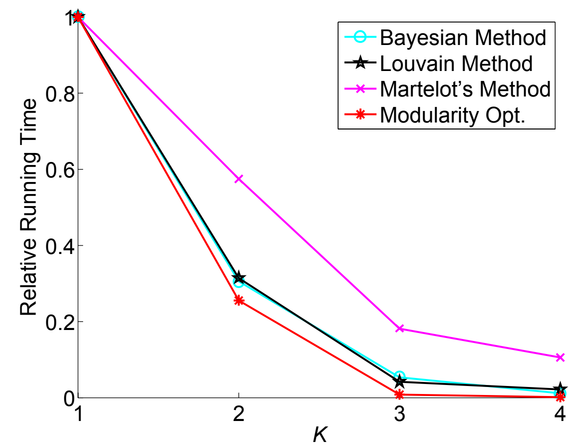












Ca-HepPh

Email-Enron

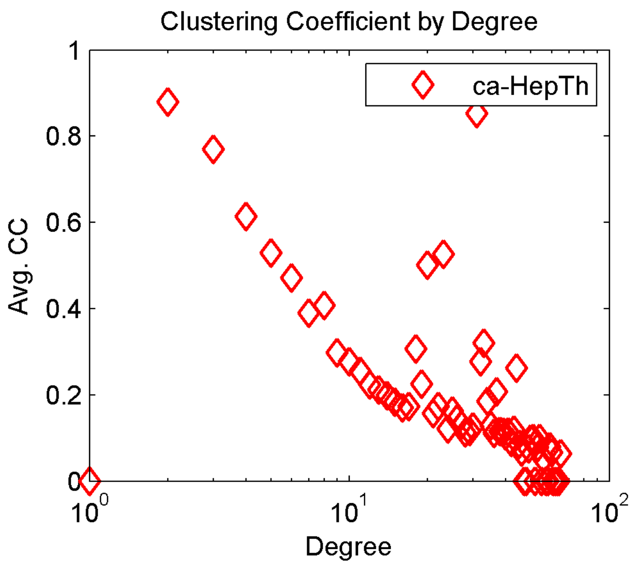
Oregon1

Implications of triangle density

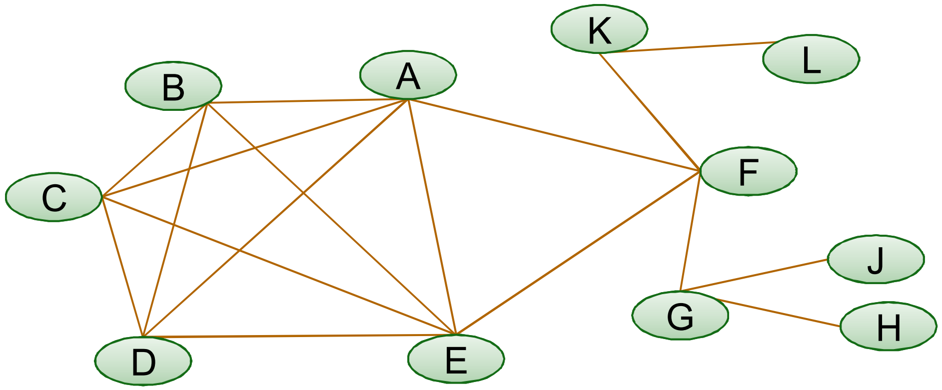
R. Gupta, T. Roughgarden, C. Seshadhri, ***Decompositions of Triangle-Dense Graphs***, Innovations in Theoretical Computer Science (ITCS)

J. Wang, R. Gupta, T. Roughgarden, C. Seshadhri, ***Counting Small Cliques in Social Networks via Triangle-preserving Decompositions,*** Preprint

Social graphs have many triangles



Social networks have high clustering coefficients!!!
Why? Groups of cohorts that are all friends with each other.

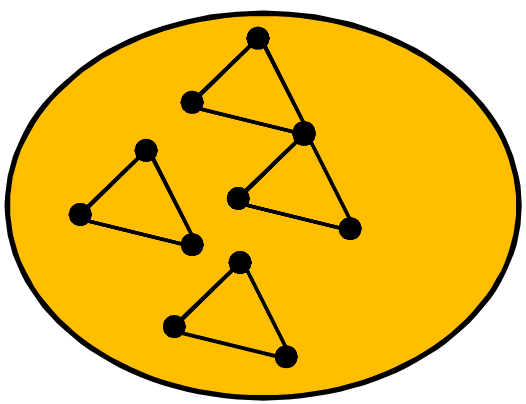


What does this imply about global structure of the graph?

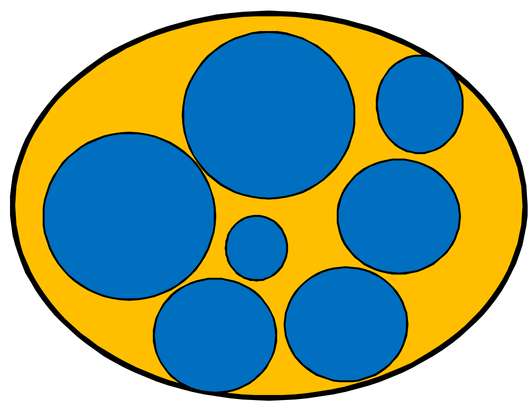
Intuition says…

Graph is triangle-dense

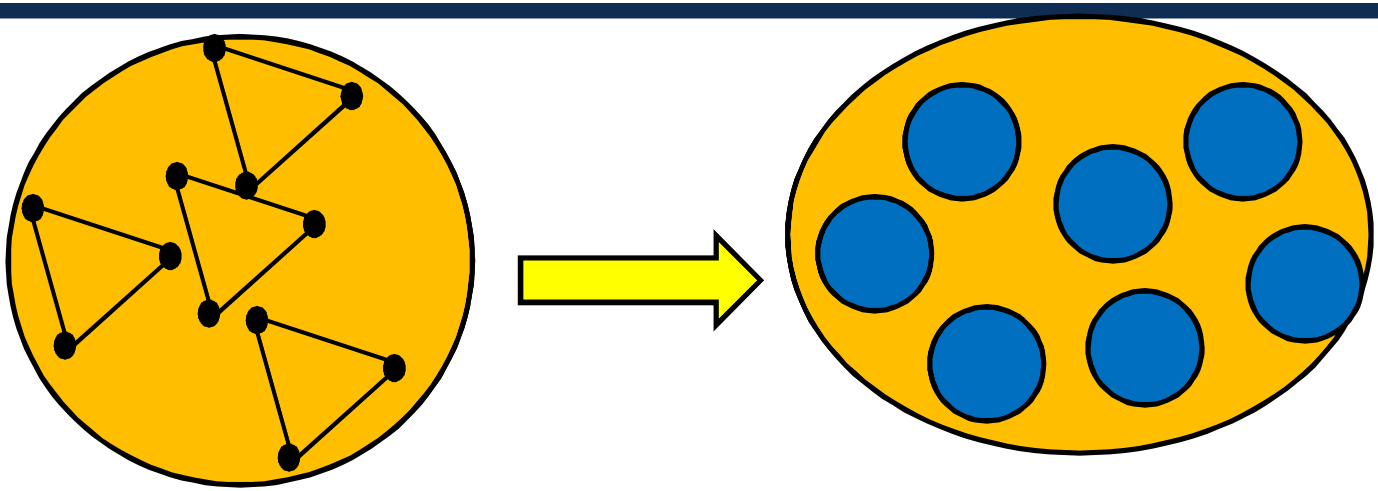
There exists "community structure"/clusters in the network



suggests

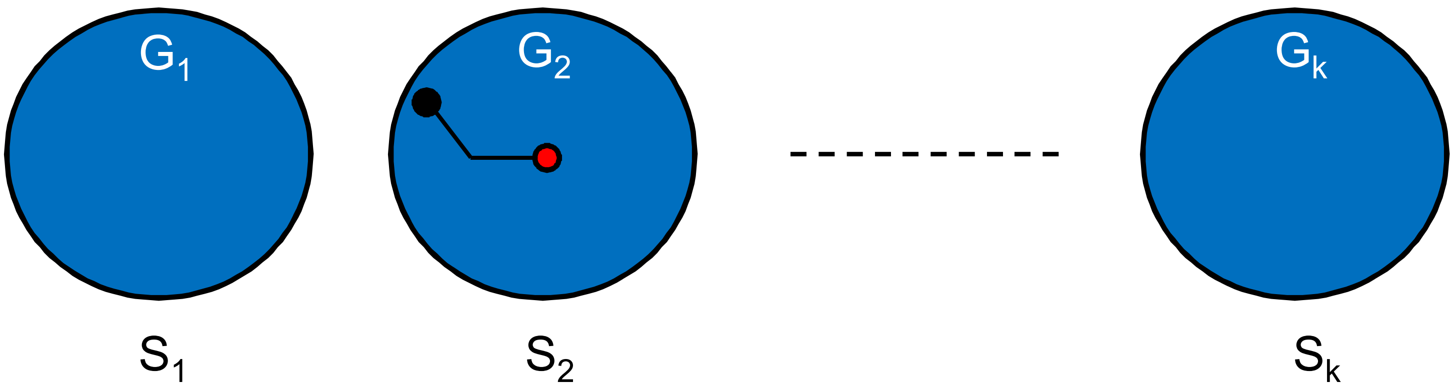


Going back to the start



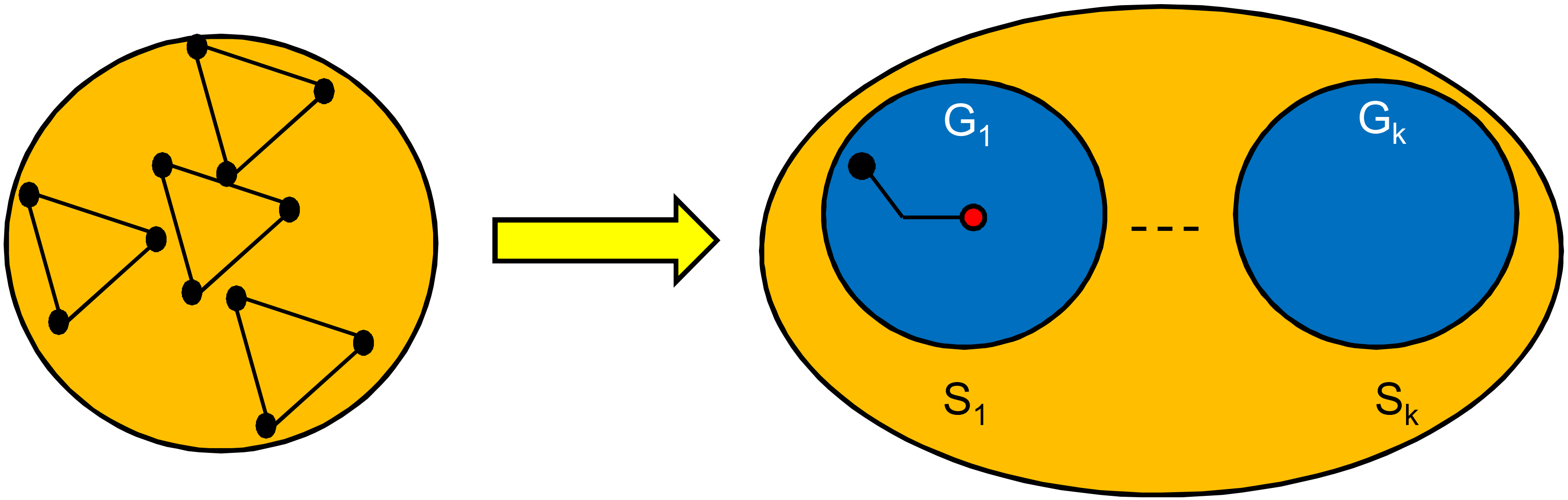
- If graph has many triangles, then graph is "special"
- Can we prove something mathematically? Try to give an algorithm with some hard guarantee?
  - How to even formalize such a question?
- Also helps in modeling

Tightly-knit family



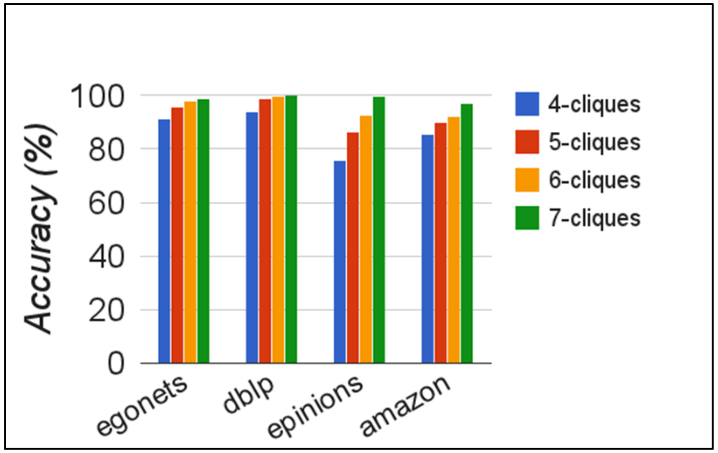
- All sets S_i are disjoint
- Each G_i is dense
- Each G_i has radius 2
    - "Central vertex" with shortest path of 2 to everyone else
- Our "approximation" to disjoint union of cliques

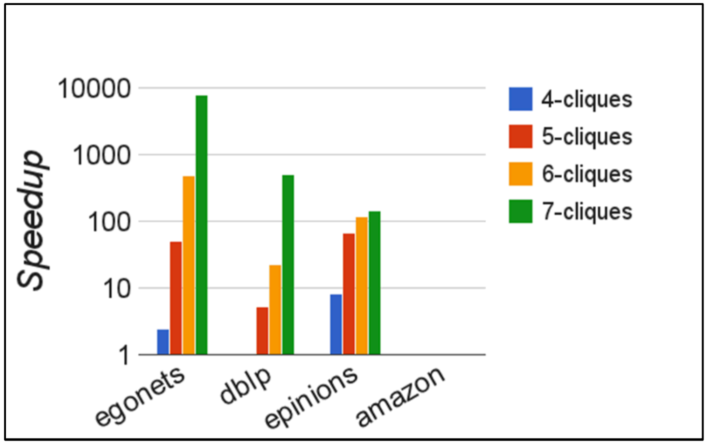
Main theorem



- Suppose G is (constant) triangle-dense
- Then G contains a tightly-knit family containing a constant fraction of triangles of G
- There is efficient algorithm to construct decomposition

How is this useful?



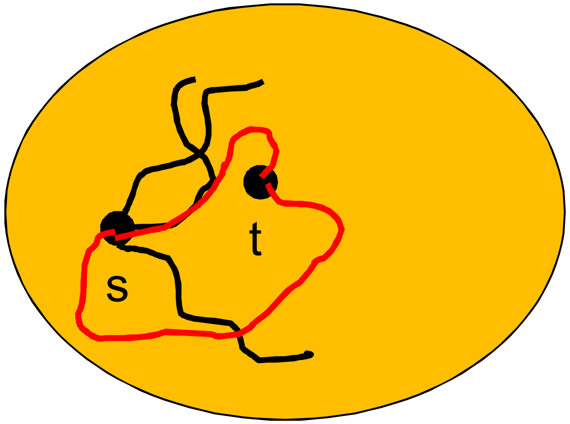


- Clique counting: fundamental problem in network analysis
- We can use the decomposition to count cliques

Faster algorithms for Personalized PageRank

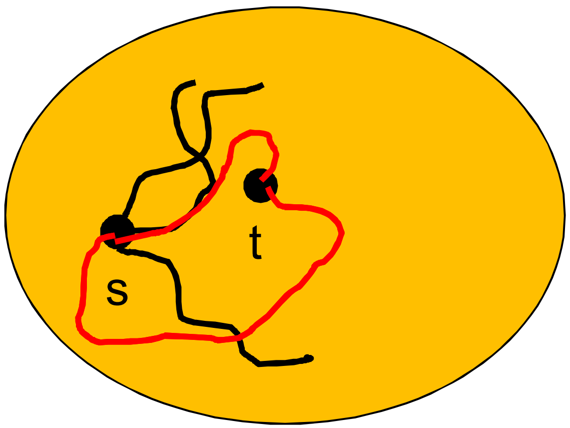
P. Lofgren, S. Banerjee, A. Goel, C. Seshadhri, ***FAST-PPR: Scaling Personalized PageRank Estimation for Large Graphs***, SIGKDD Conference on Knowledge Discovery and Data Mining 2014

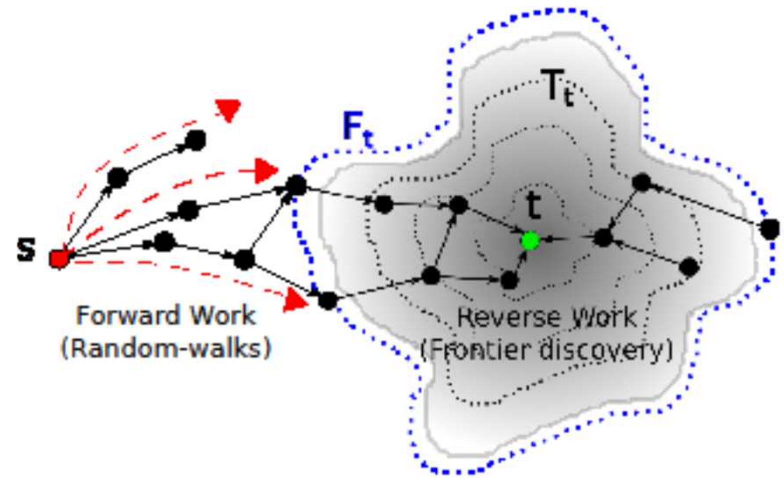
Personalized PageRank



- Start with vertex s.
- PPR_s is a score assigned to every vertex t, a measure of "importance" w.r.t s
  - Number in (0,1). Can be expressed as a probability
- How to compute $PPR_s(t)$ quickly?

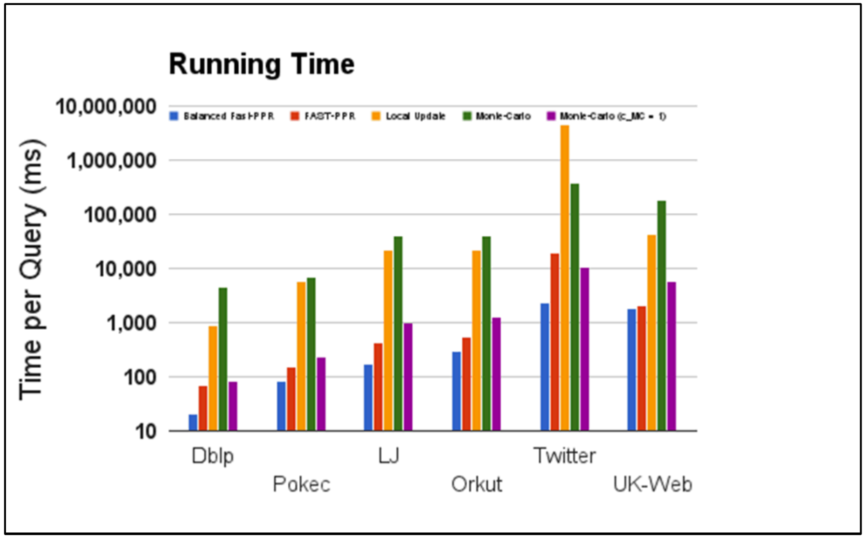
Our algorithm

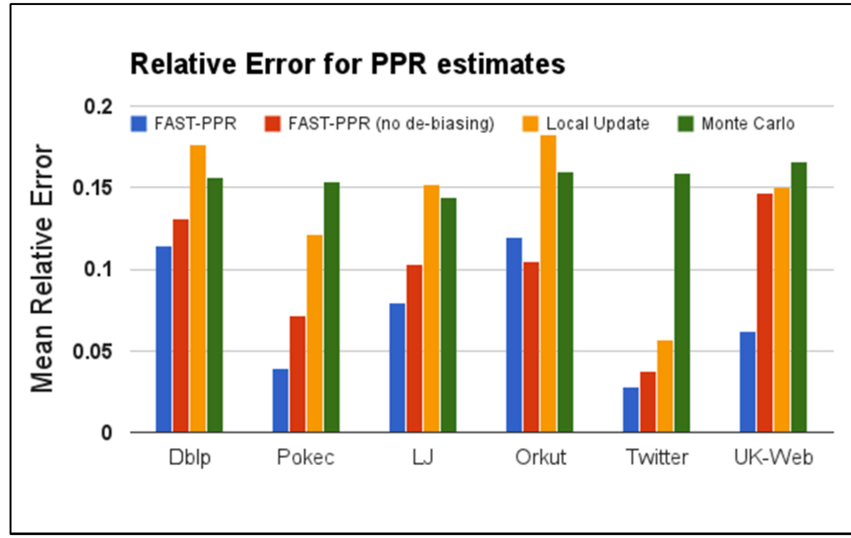




- Previous methods required 1/PPRs(t) time to get good estimate of PPRs(t)
- We design new bidirectional estimate that only requires [1/PPRs(t)]1/2
- Typically PPRs(t) is like 1/n, where n = 1B
  - This is major improvement

You get the idea





- FastPPR is many orders of magnitude faster with comparable accuracy

Phase II Publications (1/2)

- Journal Papers
  - T. G. Kolda, A. Pinar, T. Plantenga and C. Seshadhri, ***A Scalable Generative Graph Model with Community Structure***to appear in SIAM J. Scientific Computing,(Preprint: arXiv:1302.6636)
  - C. Seshadhri, A. Pinar and T. G. Kolda, ***Wedge Sampling for Computing Clustering Coefficients and Triangle Counts on Large Graphs***, Statistical Analysis and Data Mining, Vol. 7, No. 4, pp. 294-307, August 2014, doi:10.1002/sam.11224
  - T. G. Kolda, A. Pinar, T. Plantenga, C. Seshadhri and C. Task, ***Counting Triangles in Massive Graphs with MapReduce***, to appear in SIAM J. Scientific Computing (Preprint:arXiv:1301.5887)
- Conference Papers
  - P. Lofgren, S. Banerjee, A. Goel, C. Seshadhri, ***FAST-PPR: Scaling Personalized PageRank Estimation for Large Graphs***, KDD'14: Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, August 2014. arXiv:1404.3181
  - C. Klymko, D. F. Gleich and T. G. Kolda, ***Using Triangles to Improve Community Detection in Directed Networks***, The Second ASE International Conference on Big Data Science and Computing, Big Data Science, May 2014, arXiv:1404.5874
  - Jonathan W. Berry, Luke K. Fostvedt, Daniel J. Nordman, Cynthia A. Phillips, C. Seshadhri, Alyson G. Wilson, ***Why do simple algorithms for triangle enumeration work in the real world?*** SIGACT Innovations in Theoretical Computer Science, January 2014.
  - R. Gupta, T. Roughgarden, C. Seshadhri, ***Decompositions of Triangle-Dense Graphs***, SIGACT Innovations in Theoretical Computer Science (ITCS), Jan 2014 arXiv:1309.7440 doi:10.1145/2554797.2554840

Phase II Publications (2/2)

- Preprints
  - C. Seshadhri, A. Pinar, N. Durak and T. G. Kolda, ***Directed Closure Measures for Networks with Reciprocity***, arXiv:1302.6220, – Submitted for journal publication
  - M. Jha, C. Seshadhri, and A. Pinar, ***A space efficient streaming algorithm for estimating transitivity and triangle counts using the birthday paradox***, – Submitted for journal publication
  - J. Ray, A. Pinar, and C. Seshadhri***, A stopping criterion for Markov chains when generating independent random graphs,*** arXiv:1210.8184 – Submitted for journal publication
  - M. Jha, A. Pinar, and C. Seshadhri***, Path Sampling: A Fast and Provable Method for Estimating 4-Vertex Subgraph Counts,*** Submitted for conference publication***.***
  - M. Jha, C. Seshadhri, and A. Pinar, ***Counting Triangles in Real-World Graph Streams: Dealing with Repeated Edges and Time Windows,*** arXiv:1310:7665 **--** Submitted for conference publication
  - C. Peng, T. Kolda, and A. Pinar***, Accelerating Community Detection by Using k-core subgraphs,*** submitted for conference publication***.*** arXiv:1403.2226
  - J. Wang, R. Gupta, T. Roughgarden, C. Seshadhri, ***Counting Small Cliques in Social Networks via Triangle-preserving Decompositions,*** February 2014. Preprint

Phase II Presentations and Service

Presentations

- Feb 2014: T. Plantenga, ***Generating Large Graphs with Desired Community Structure***, SIAM Parallel Processing in Scientific Computing, Portland, OR, Feb 18-21, 2014
- Feb 2014: A. Pinar, ***Generating Large Graphs for Benchmarking***, SIAM Parallel Processing in Scientific Computing, Portland, OR, Feb 18-21, 2014
- April 2014: A. Pinar, ***Counting Small Patterns in Large Graphs***, 2014 SIAM Data Mining: Workshop on Mining Networks and Graphs: A Big Data Analytic Challenge, Philadelphia, PA, April 26, 2014.
- May 2014: T. Kolda, ***Sandia Software for Networks from DARPA GRAPHS Program***, DARPA GRAPHS Special Projects Meeting, Arlington, VA, May 13-14 2014
- July 2014: A. Pinar, ***Accelerating Community Detection by Using K-core Subgraphs***," SIAM Workshop on Network Science, Chicago, IL, July 6-7, 2014.
- July 2014: T. Kolda, **Analytical and Algorithmic Challenges in Network Analysis,** Keynote presentation, 6th SIAM Workshop on Combinatorial Scientific Computing, Lyon, France, July 21-24, 2014

Service

- Workshop Organization
  - Nov 2014: Dagstuhl Seminar on Seminar on "High-performance Graph Algorithms and Applications in Computational Science," Saarbrucken, Germany – Co-organized by Pinar
  - Aug 2014: MIT Lincoln Labs Graph Exploitation Symposium, Pinar on OC
  - Jul 2014: SIAM Workshop on Network Science, San Diego, CA – Pinar SC chair.
  - Apr 2014: SDM Workshop on Mining Networks and Graphs, Philadelphia, PA – co-organized by Pinar
- Program Committees
  - CIKM14 – Pinar on PC
  - WWW14 – Pinar on PC
  - SDM14 – Kolda on SPC, Berry, Pinar, Jha on PC
  - NetSciCon14 – Phillips general chair
- Editorial Board
  - SIAM Journal on Scientific Computing (Kolda, Pinar)
  - Journal of Complex Networks (Kolda, Pinar)

Phase 2 Progress and Changes

Milestone themes and status

- Streaming methods for evolving graphs
  - Algorithms for multi-graphs; ability to observe multiple windows
- Community detection
  - Directed communities; speeding up existing methods
- Higher-order patterns
  - Exact and sampling algorithms for 4-vertex patterns
- Models for evolving graphs
  - More emphasis on cyber data and benchmarking; expect rapid progress with the new FY.
- Anomaly detection on evolving networks
  - Ongoing effort
- Better models for degree distributions
  - Priority after the PI meeting
- Hyperbolic embeddings
  - On hold so far

Future plans

- In good shape with milestones
- Increased emphasis on impact
- Reprioritize goals based on feedback
  - Will continue to dialog with partners
- Need to make room for brain networks
- Evolving graph models will have more focus on cyber data
- FEASTPACK software
  - Release code for new algorithms
  - Alternatives to the Matlab interface

PI: Tamara G. Kolda
Sandia National Labs
tgkolda@sandia.gov
925-294-4769

PM: Rick McCormick
Sandia National Labs
fbmccor@sandia.gov
505-284-1209

Moving forward

- In good shape with our milestones
  - 6 journal (3 pending), 8 conference (4 pending) papers, many invited/plenary talks
- Increased emphasis on impact
  - Reprioritize goals based on feedback
  - Will continue to dialog with partners
- Always open to collaborations
  - Currently have 1 postdoc, 4 interns, and 1 visitor
- Need to make room for brain networks
- FEASTPACK software is freely available
  - Plan to release code for new algorithms
  - Will provide lternatives to the Matlab interface

PI: Tamara G. Kolda
Sandia National Labs
tgkolda@sandia.gov
925-294-4769

Ali Pinar
Sandia National Labs
apinar@sandia.gov
925-294-4683

C. Seshadhri
Sandia National Labs
scomand@sandia.gov
925-294-3696