SAND2013-3750C

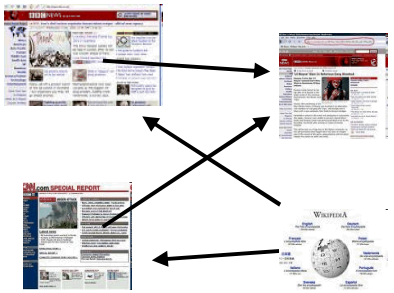
Triadic Measures on Graphs: The Power of Wedge Sampling

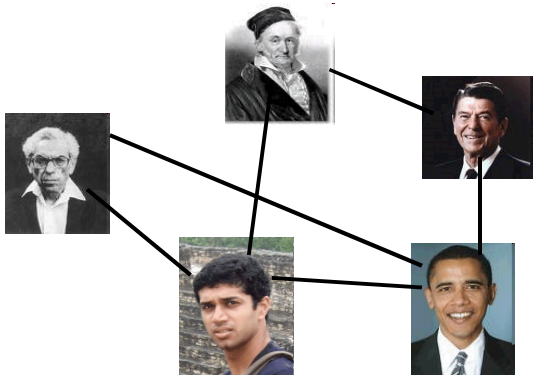
C. Seshadhri, Ali Pinar,
Tamara G. Kolda

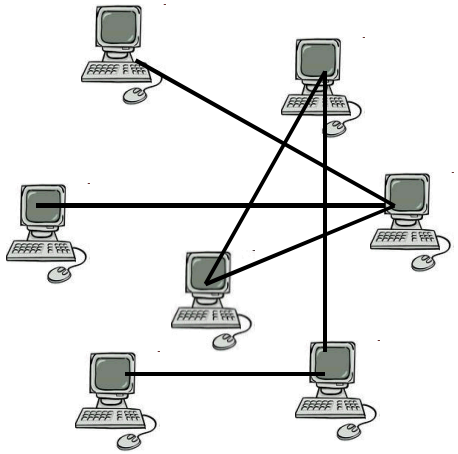
Sandia National Labs, CA

U.S. Department of Defense
Defense Advanced Research Projects Agency

Real graphs are huge

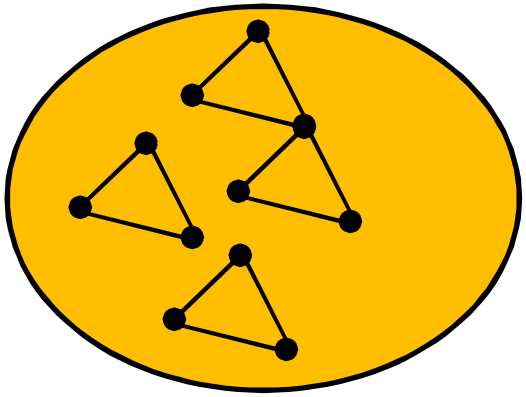






- n = no. of vertices >> millions
- m = no. of edges >> 10s to 100s millions
- But we'd like to process them fast
  - Get some sense of "what's in the data"

Searching for sub-patterns



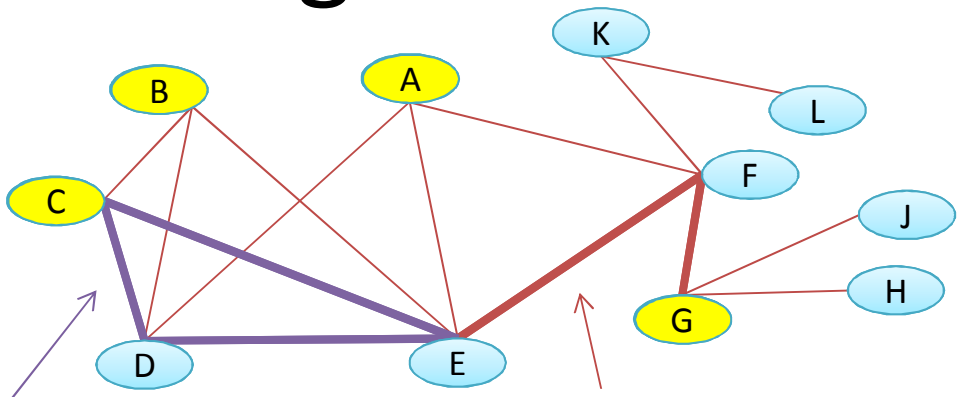
Graph	n	m	T
web-BerkStan	700K	6.6M	64M
flickr	1.8M	15M	550M
livejournal	5.2M	48M	300M

- Common analysis method is to find sub-structures of interest
  - Motifs in bioinformatics, graphlets, structural signatures in sociology
  - (Think of graph as simple and undirected)

- The triangle: a clique of size 3
  - When is “friend of friend” also a friend
  - Well-known to be abundant in real-world (esp. social) graphs
  - Rich history of analyzing triangles in massive graphs

Triangle information

$c_A = 2/3$

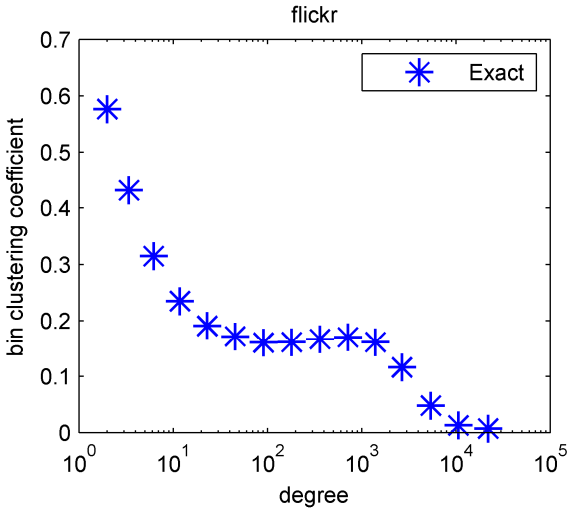


c_3
$= ¼(2/3 + 2/3 + 2/3 + 0)$
$= ½$

- W = no. of wedges (paths of length 2)
  - "Center" of wedge is middle vertex
- T = no. of triangles
- [Wasserman-Faust] $\tau = 3T/W$ = fraction of closed wedges
- [Watts-Strogatz] $cc_v = Tv/Wv$
- $cc = \sum_v cc_v/n$
- Degree-wise: cc_d = avg. of cc_v over v of degree d

Real triangle information

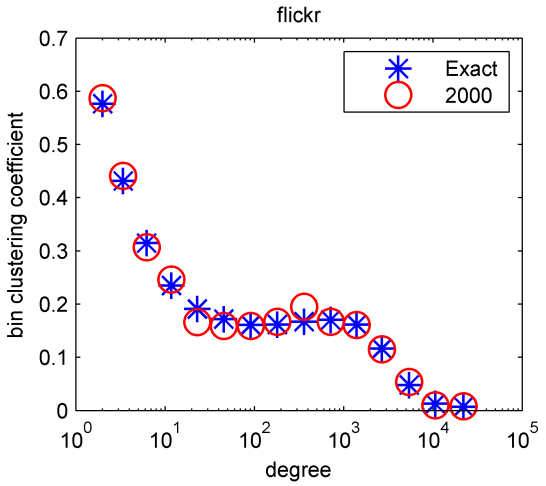
Name	n	m	T	τ	cc
flickr	1.8M	15M	550M	0.11	0.38



- Very common method of presenting triangle information
  - You'll see this in most results/papers/talks on large graphs
- Computing this computationally painful
  - Scalability major issue. W and T grows superlinear in m
  - Especially degree-wise plots. Currently, requires full enumeration of triangles

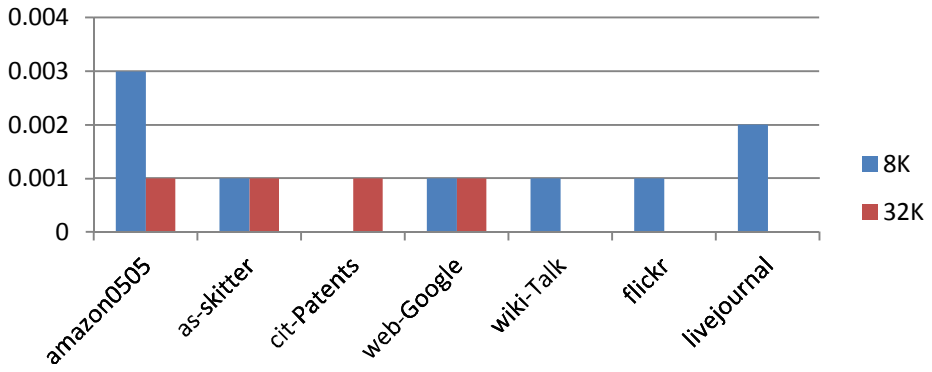
Our contribution

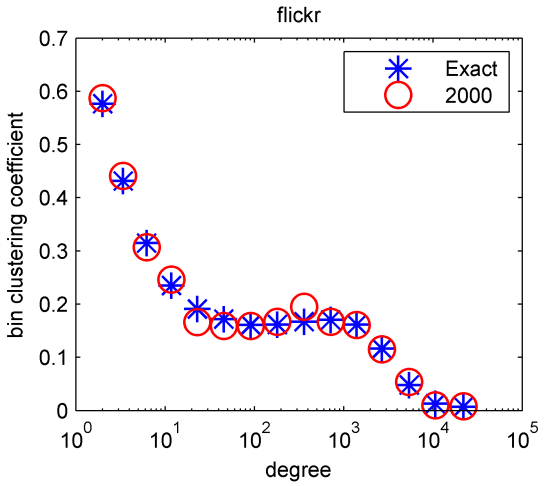
Name	n	m	T	τ	cc
flickr	1.8M	15M	548M	0.112	0.375
			542M	0.111	0.371



- Wedge sampling: a versatile method to estimate all this triangle information
  - Provable theoretical guarantees on approximation-accuracy-time tradeoff, based on fairly simple math
  - Previous fast methods only estimate T, τ, maybe cc
  - Extensive experiments on variety of large graphs
- Speedup of 100X on large graphs over enumeration
  - Also faster and more accurate than comparable methods that only estimate T

Our contribution



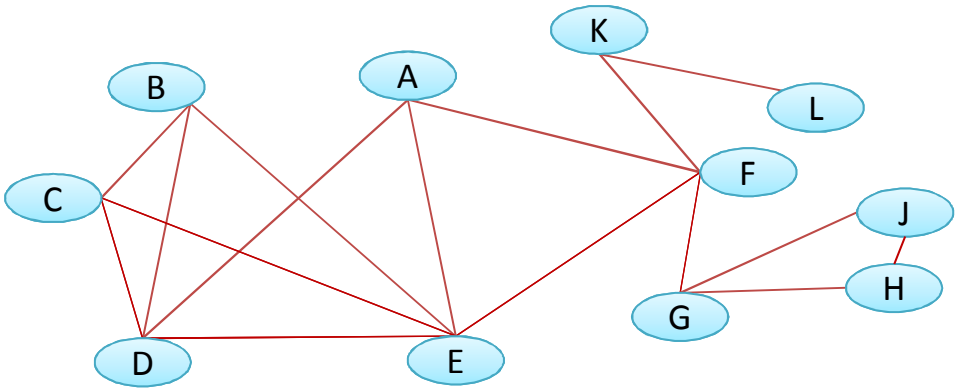


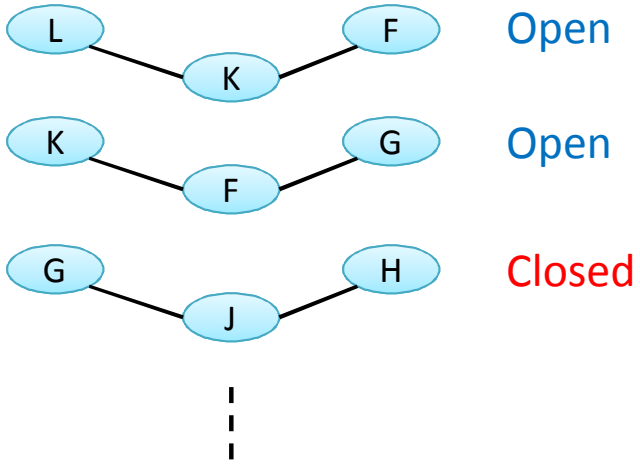
- Wedge sampling: a versatile method to estimate all this triangle information
- Speedup of 100X on large graphs over enumeration
- [Schank Wagner 05] Older independent discovery of wedge sampling in more theoretical context
  - Do not estimate degree-wise coefficients, or perform large-scale experiments
  - Please cite them!

Previous data mining work

- [Cohen09, Suri-Vassilvitskii11, Chu Cheng11] Clever enumeration schemes
  - We compare our times with these
- [Tsourakakis08, Avron10] Eigenvalue methods for estimating T
  - Scalability is an issue
- [Tsourakakis-Kang-Miller-Faloutsos09, Tsourakakis-Kolountzakis-Miller11] Doulion: a sampling sparsification for estimating T
  - Nice memory properties and quite fast in practice
  - Wedge sampling is faster and more accurate. Strong theoretical guarantee
- GraphLab, Map-Reduce methods, etc. for faster enumerations

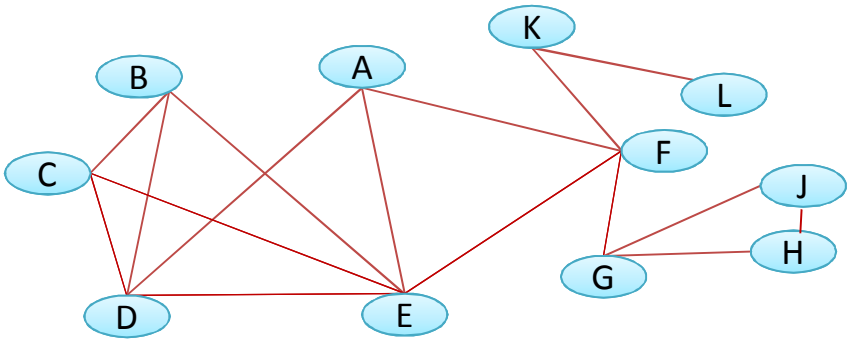
Wedge sampling for τ

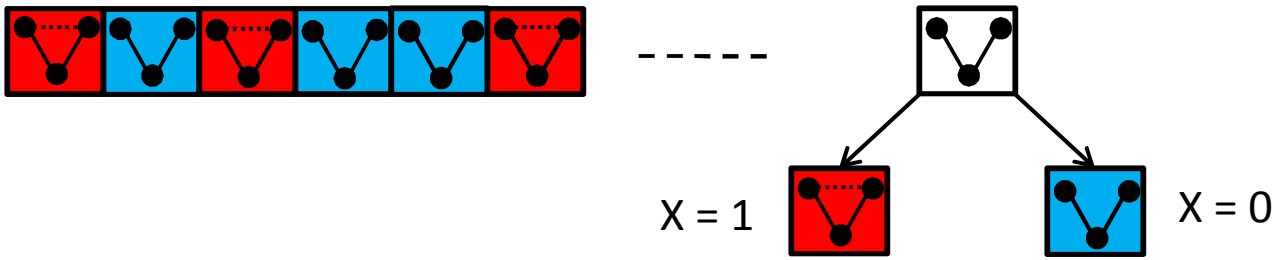




- τ = 3T/W = fraction of closed wedges
- Consider list of all wedges, indexed with open/closed

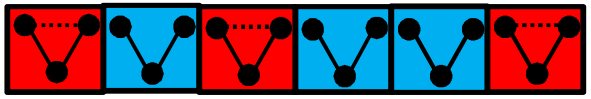
Wedge sampling for τ

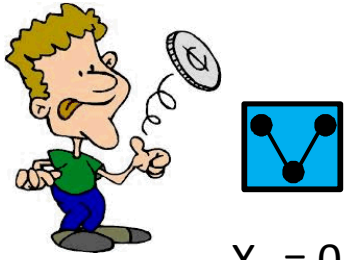




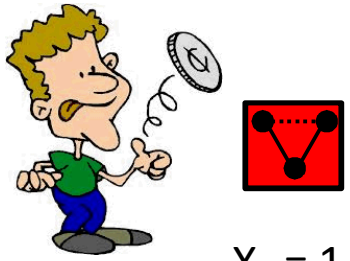
- τ = 3T/W = fraction of closed wedges
- Consider list of all wedges, indexed with open/closed
- Pick a uniform random wedge. X = 1 if wedge is closed. Else X = 0
- X is Bernoulli random variable, with E[X] = τ

Repeat, repeat, repeat

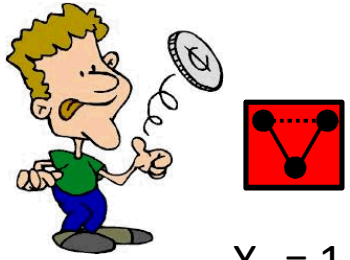




$X_1 = 0$



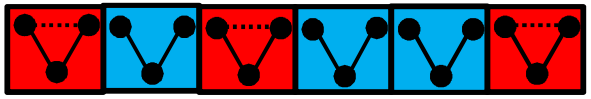
$X_1 = 1$

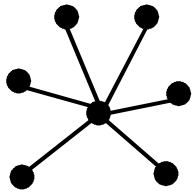


$X_2 = 1$

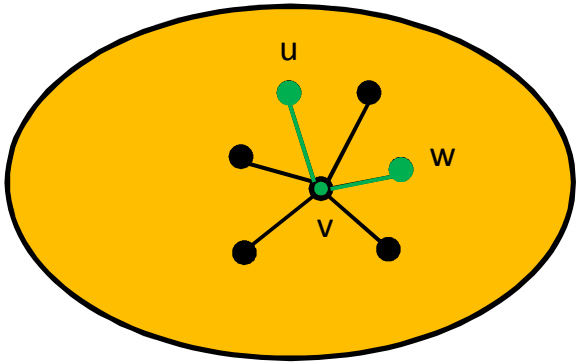
- Perform k independent experiments. Let Y = (1/k) sum Xi
  - Y is fraction of closed wedges in sample
  - E[Y] = tau. Y converges to tau as k grows
- [Chernoff-Hoeffding]: Set k = $(2\varepsilon)^{-2}\log(2/\delta)$. Then $\Pr[|Y - \tau| > \varepsilon] < \delta$
  - With prob > 1- δ, estimate is accurate within ε
  - With 38K samples, error < 0.01 with prob > 0.999
  - Number of samples independent of graph size

Don’t generate list



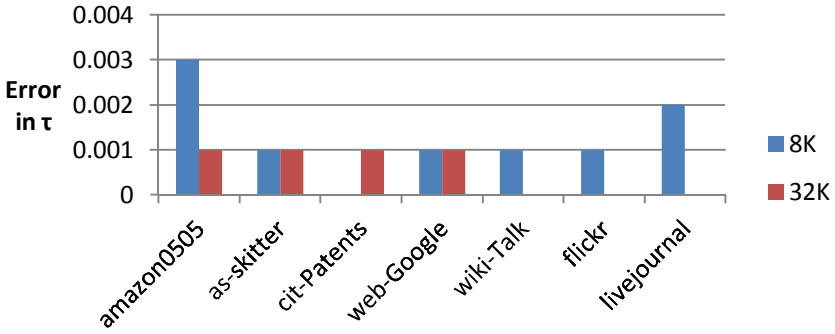


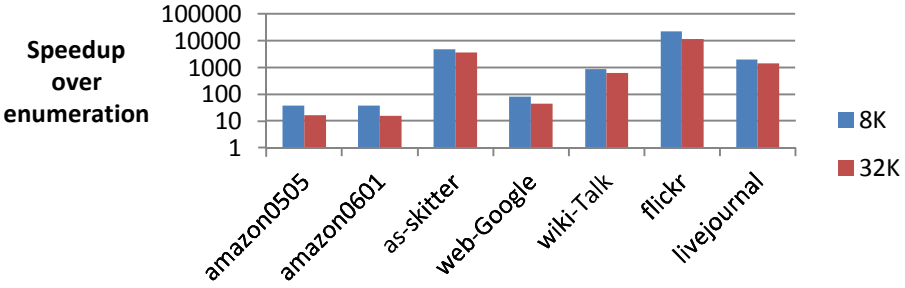
$$W_v = \binom{d_v}{2}$$

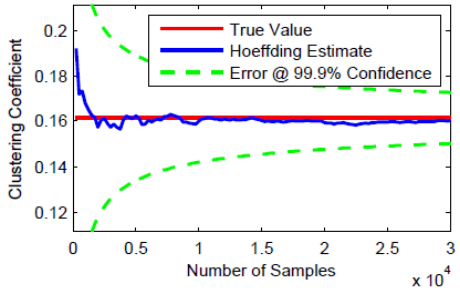


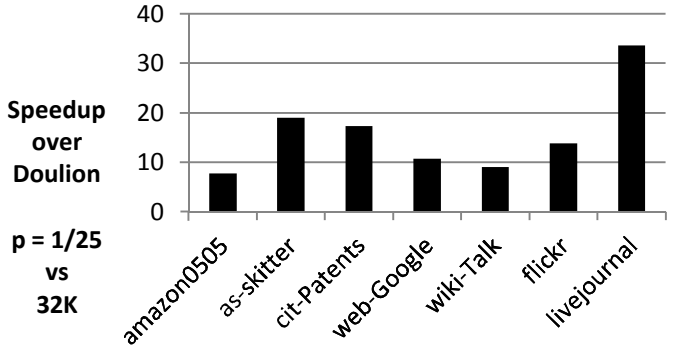
- But list of wedges not possible to generate. So how to get random wedge?
- Pick vertex v with probability W_v/W
- Pick two uniform random neighbors of v to get wedge (u,v,w)
  - This is a uniform random wedge
- So simply repeat this many times to get a set of wedges. Output fraction of closed wedges as estimate for τ

It works



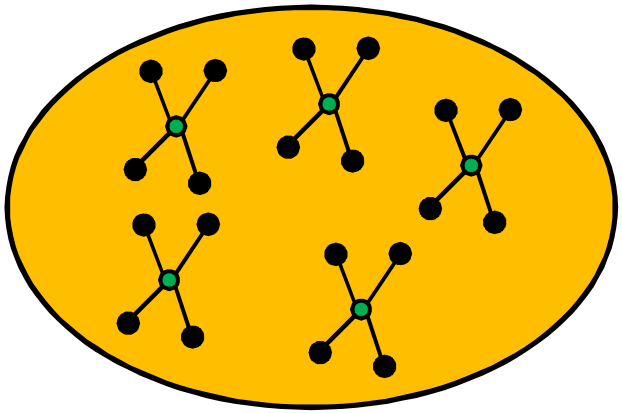


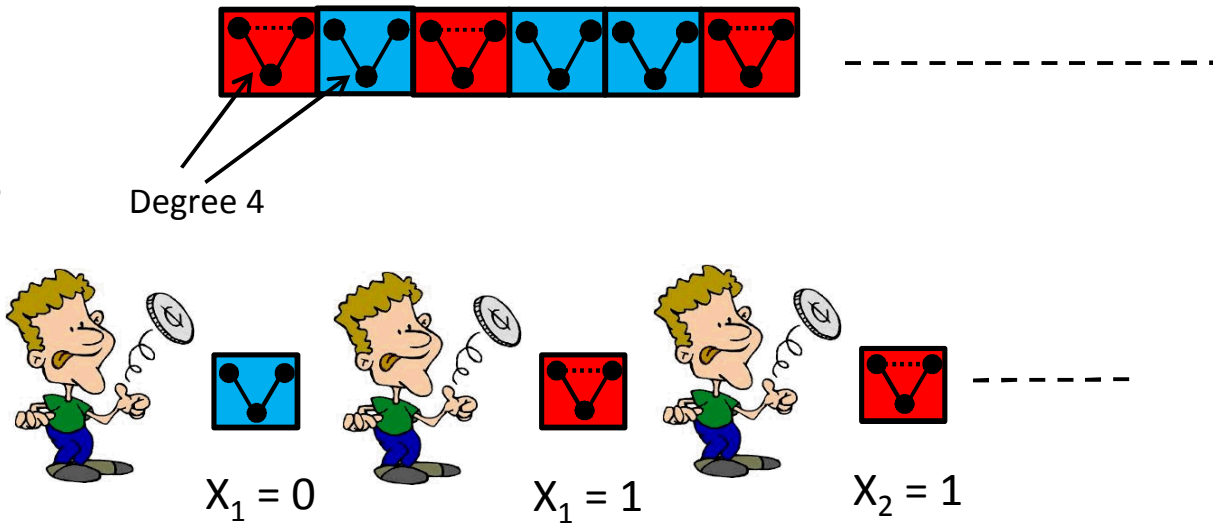




- Convergence is fast
- Same number of samples for all graphs. (Setting up the sampling requires linear time.)
- Caveat: if very few triangles in graph, overall estimate for number of triangles can be off

What about cc?

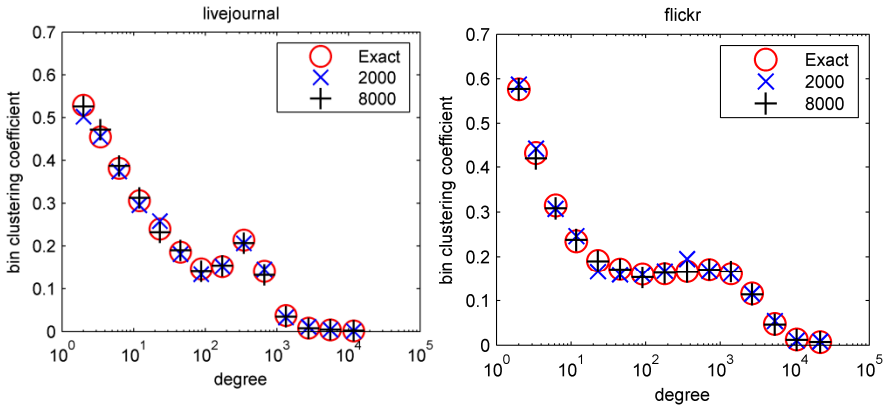


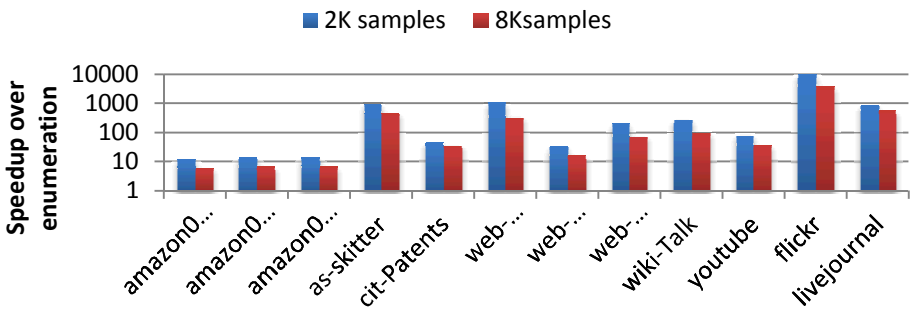


- Say we want cc_4
- Sample exclusively from wedges centered at degree 4 vertices

- By tweaking this, one can also count number of triangles involving degree 4 vertices

It works



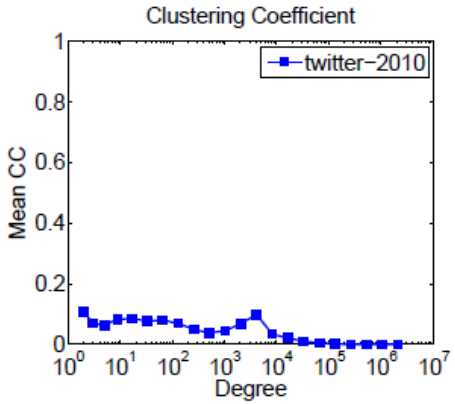


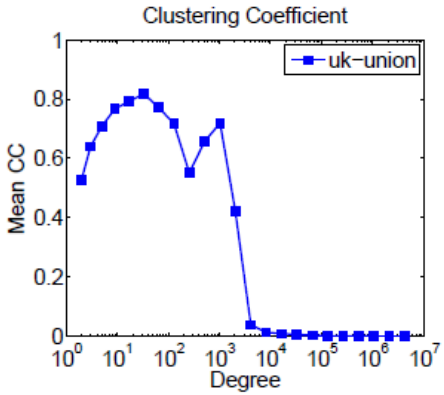
- 2K samples per bin gives good estimate
- Doing sampling with binning is little more work, but straightforward
  - Bins are at powers of 2

- If you make bins smaller (so plot is finer), it becomes more expensive

Scaling it for BIG graphs

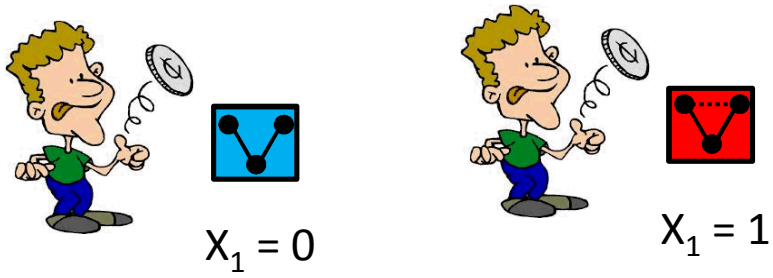
Graph	n	m	T	τ
Twitter-2010	42M	1.2B	34B	0.0008
uk-union	132M	4.6B	447B	0.007





- [Kolda Pinar Plantenga S Task 13] Wedge sampling in Map-Reduce

If I have a minute left…



- Contribution #1: Introducing data mining world to sampling methods
- Contribution #2: Showing that it actually works on real graphs, and can give good results

- Try out wedge sampling!

Thank you!

scomand@sandia.gov

http://arxiv.org/abs/1202.5230