

The Trials and Tribulations of Tractably Tabulating Triangles

C. Seshadhri

(Sandia National Labs, Livermore)

Joint work with Madhav Jha and Ali
Pinar (Sandia National Labs, Livermore)

Theory and practice



Theory

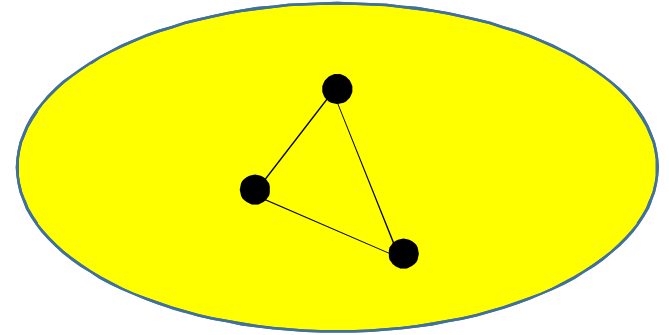


Practice



There and back again

We love triangles



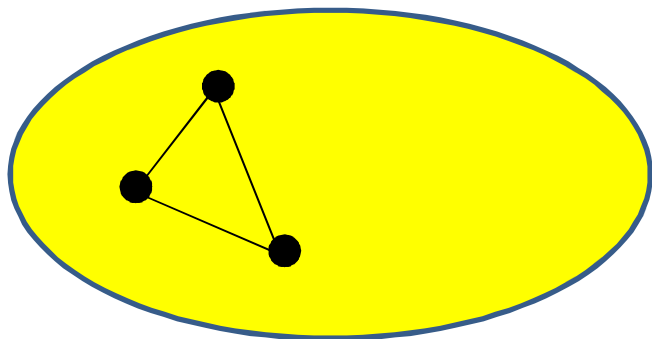
- The building block of communities. The sign of social networks. The transitivity of relationships
- One subgraph to rule them all

Social sciences: [Holland-Leinhardt70] [Coleman88] [Skvoretz90]
[Portes98] [Burt04] [Welles etal10][Faust10] [Szell-Thurner10]

Physical sciences: [Watts-Strogatz98] [Eckmann-Moses02]
[Fagiolo07] [Milo etal10] [Son etal10] [Leskovec etal10] [Winkler-Reichardt13]

Algorithmics: [Becchetti etal08][Berry etal11] [Gleich-S12] [Rohe-Qin13]

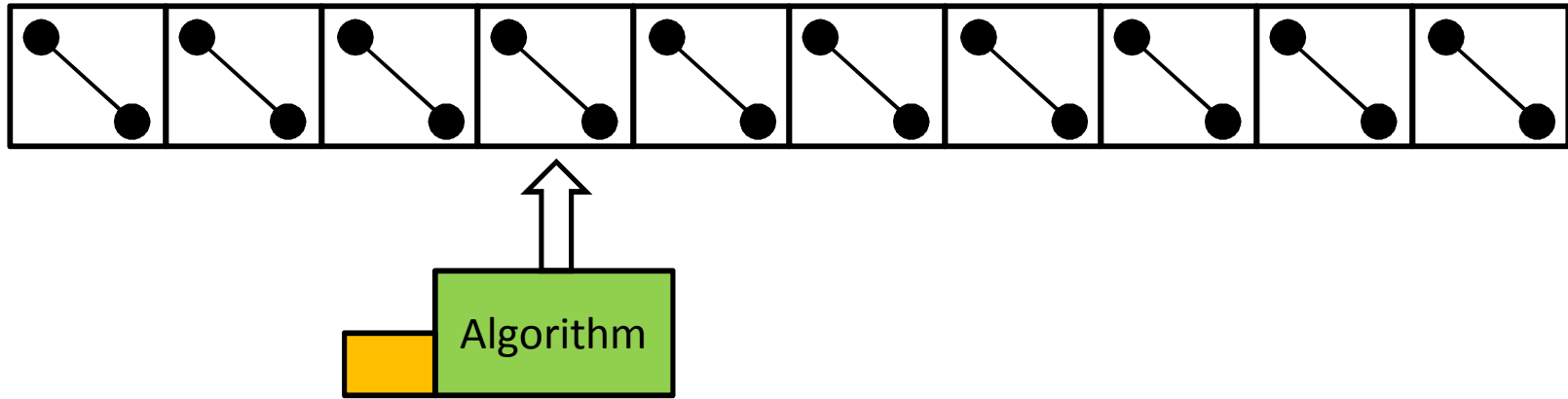
Counting triangles



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

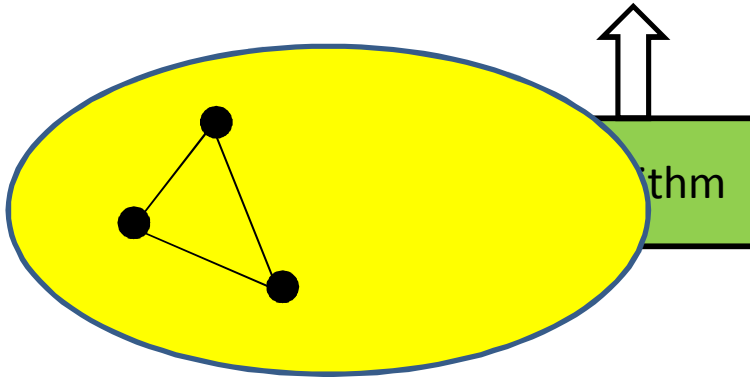
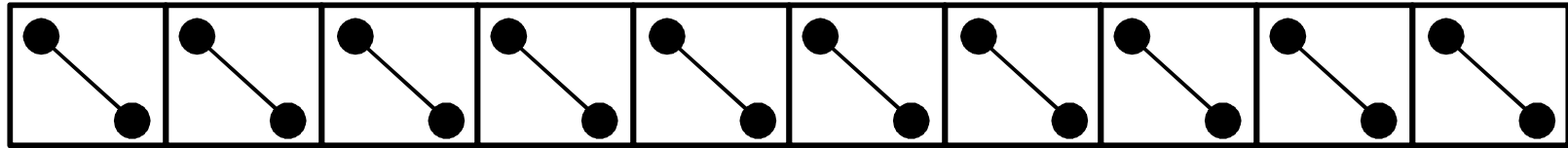
- How to accurately count/estimate T, or related “triangle measures”?
- We’re looking for efficient, scalable algorithms
- Even clever enumeration won’t work

The streaming setting



- Crucial for real-time analytics of temporal graph
 - Memory orders of magnitude smaller than stream
- **No assumption on graph or ordering of edges**
- Single pass over data. If algorithm forgets something, too bad

The streaming setting



- Algorithm should output (estimate) number of triangles at the end
- Output with high probability (> 0.99) over coin flips, NOT over the input

History



[Bar-Yossef etal02]

[Jowhari-Ghodsia05]

[Ahn-Guha-McGregor12]

[Pagh-Tsourakakis12]

[Kane etal12]

[Braverman-Ostrovsky-Vilenchik13]

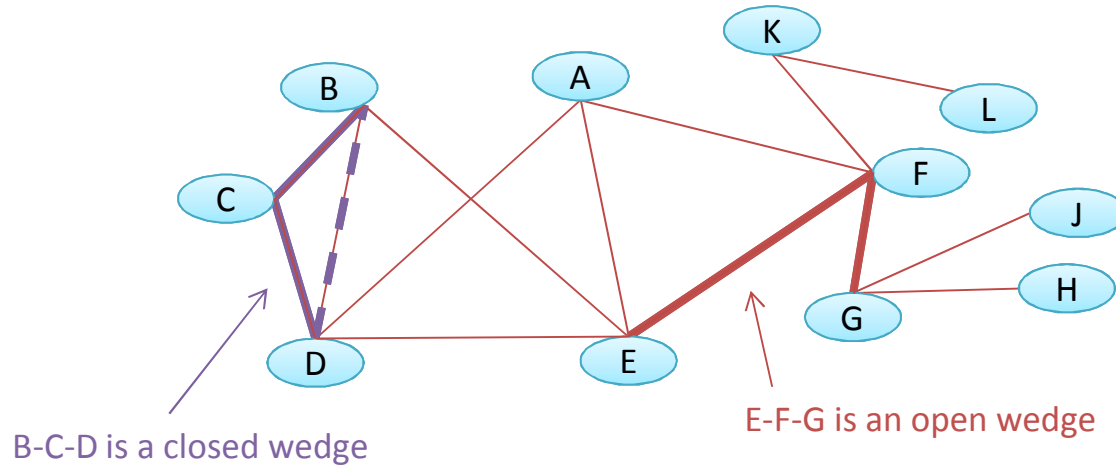


[Buriol etal06]

(Breaks down at million edge graph.)



Notation



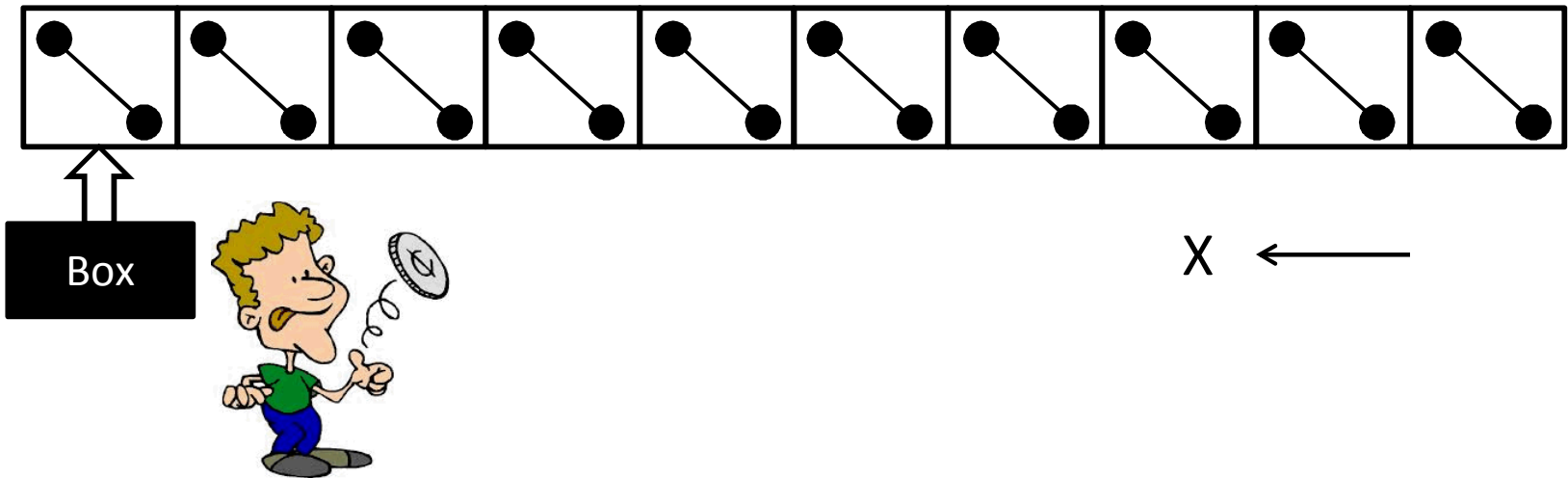
- n = no. of vertices
- m = no. of edges
- W = no. of wedges (paths of length 2)
 - “Center” of wedge is middle vertex
- T = no. of triangles
- Transitivity = $\tau = 3T/W$ = fraction of closed wedges
- Amen

Result

- [\[Jha-S-Pinar13\]](#) Fix $\epsilon > 0$. Algorithm's features:
 - $O(\sqrt{n})$ space
 - Output τ' such that $|\tau - \tau'| < \epsilon$ w.h.p.
 - Works terrible in practice
- Heuristic “fix”
 - Somewhat principled, I think
 - Works well in practice
 - Estimates τ to within 0.01 error with 40K edges, for graph with $> 200M$ edges
- [\[Pavan-Tangwongsan-Tirthapura-Wu13\]](#) Alternate approach, works well in practice

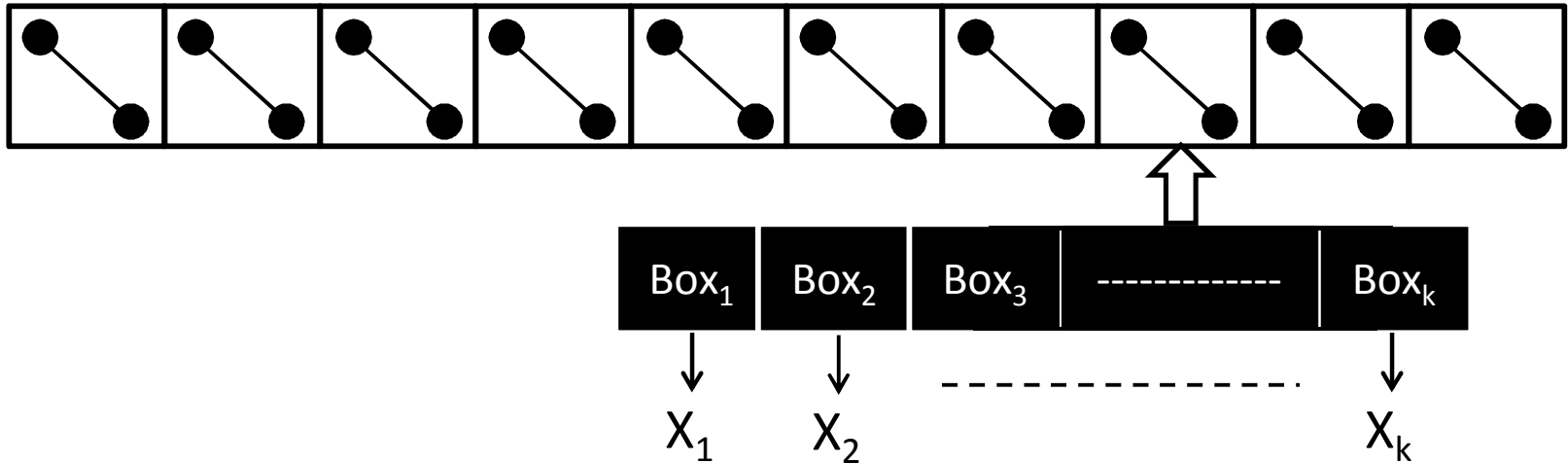


The scheme



- Fix the graph stream
- Construct small space box that outputs Bernoulli r.v. X such that $E[X] = \tau$

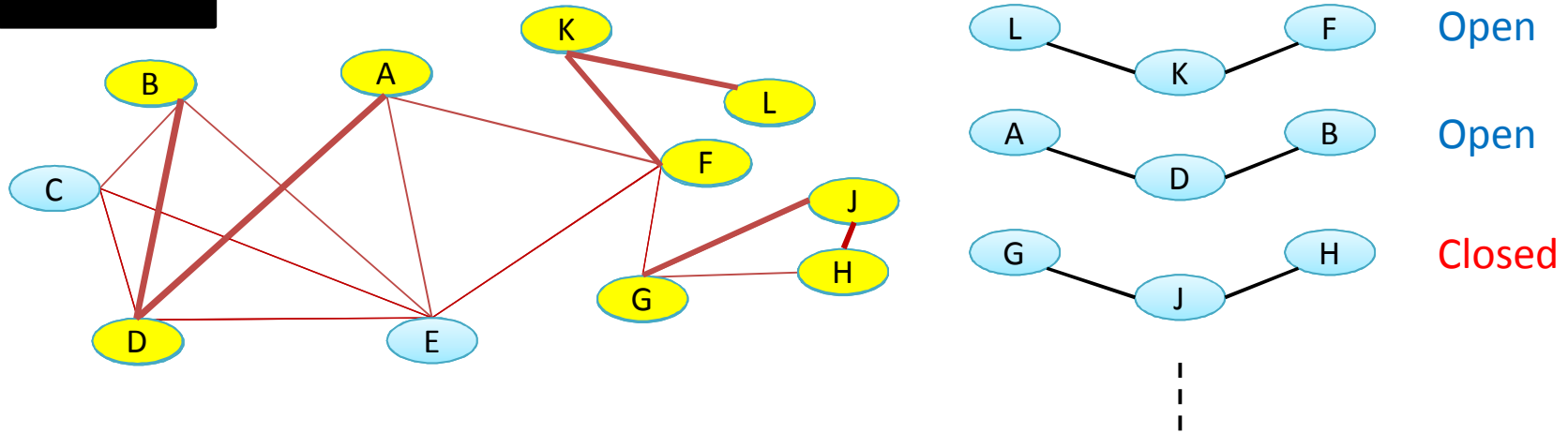
The scheme



- Generate i.i.d. X_1, X_2, \dots, X_k
 - Output fraction of these that are 1 as τ'
- [Chernoff] If $k \sim 1/\epsilon^2$, $|\tau - \tau'| < \epsilon$ w.h.p.
- Total space = $|\text{Box}|/\epsilon^2$

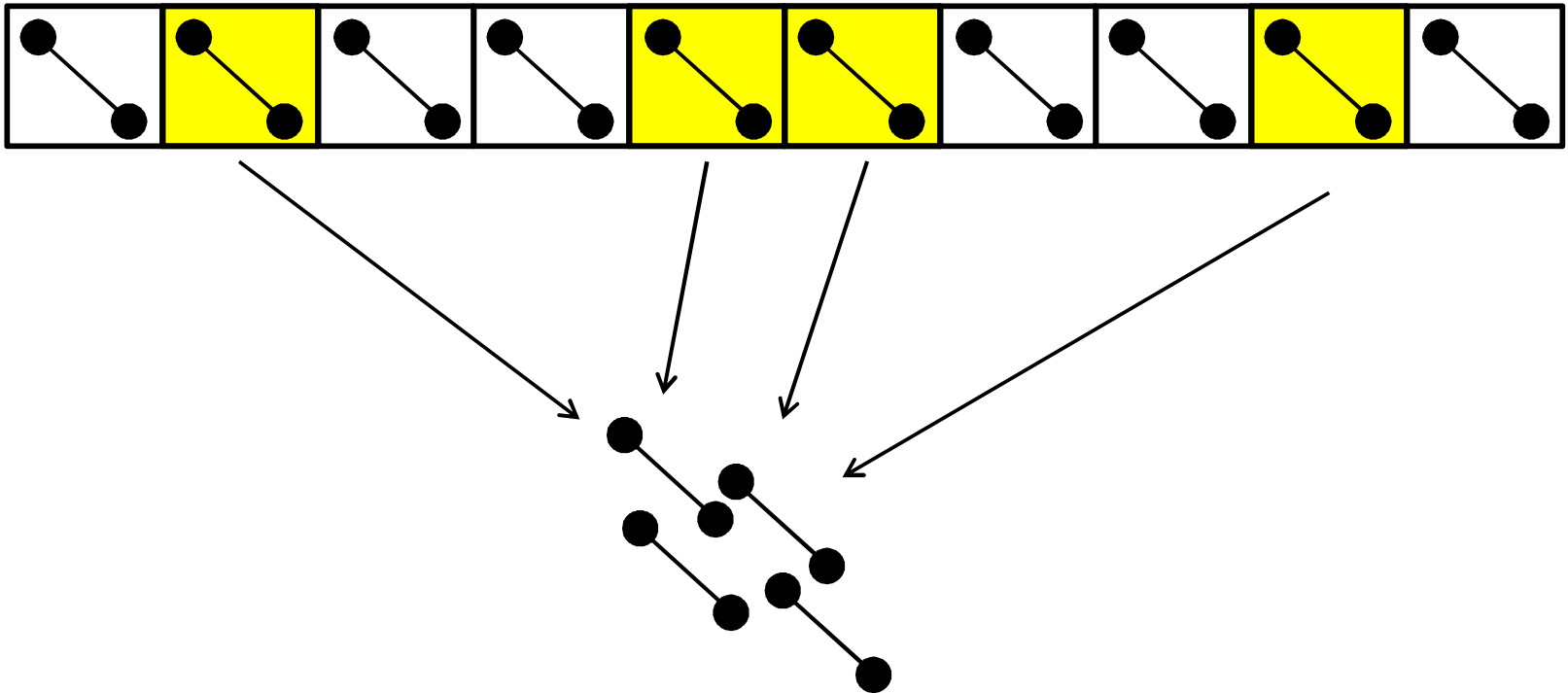
What has it got in its pockets?

Box?



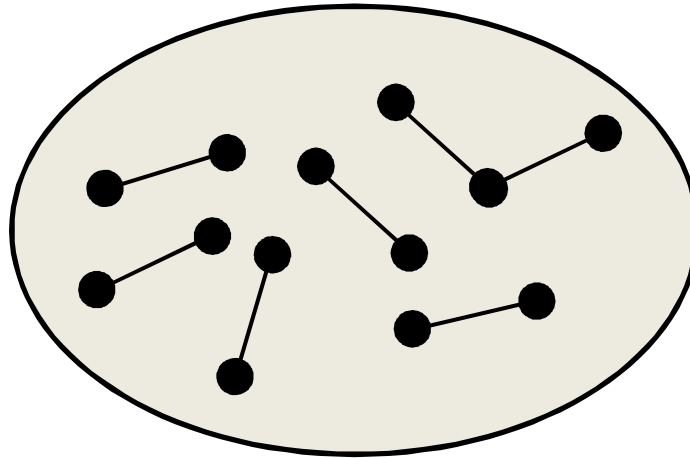
- $\tau = 3T/W =$ fraction of closed wedges
- Sample uniform at random (u.a.r.) wedge.
 $X = 1$ if closed and $X = 0$ otherwise
- How to sample u.a.r. wedge from edge stream?

Getting a wedge



- Picking random edges from stream is easy
 - [Vitter85] Reservoir sampling
- How many to pick before you get a wedge?

Getting a wedge



1

2

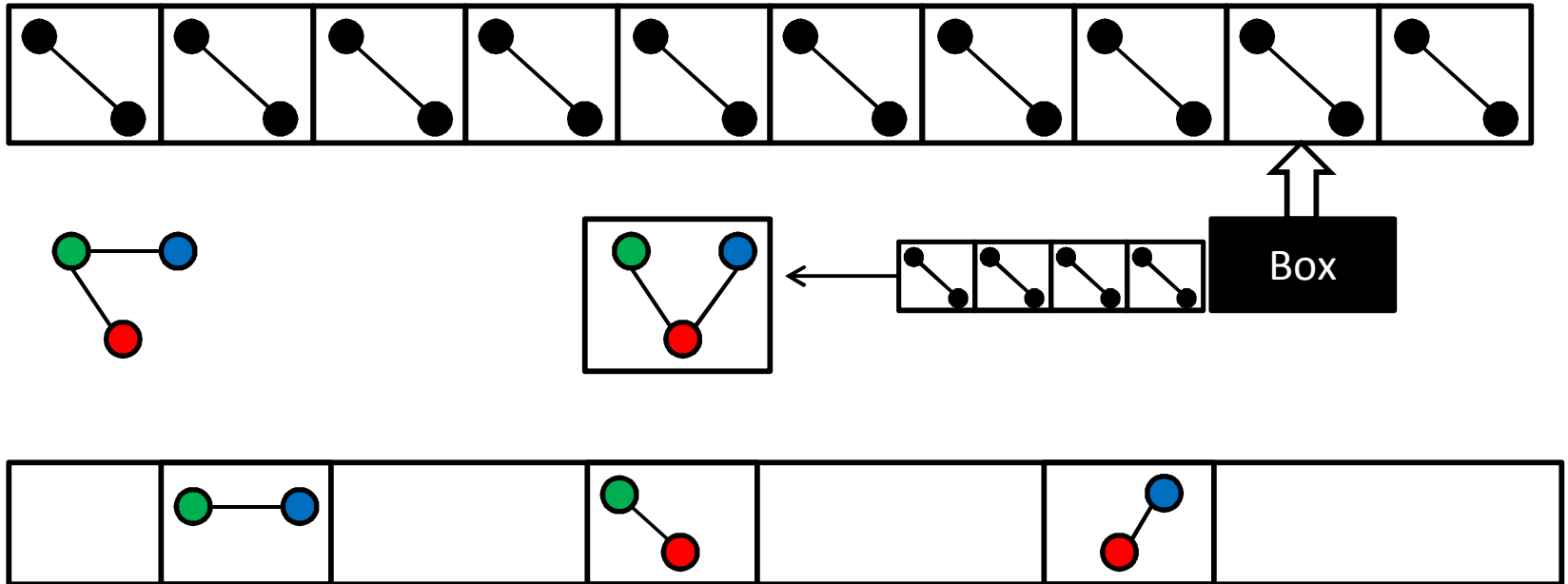
3

n

- Birthday paradox calculation!
- $O\left(\frac{m}{\sqrt{W}}\right) = O(\sqrt{n})$ u.a.r. edges suffice to get wedge w.h.p.

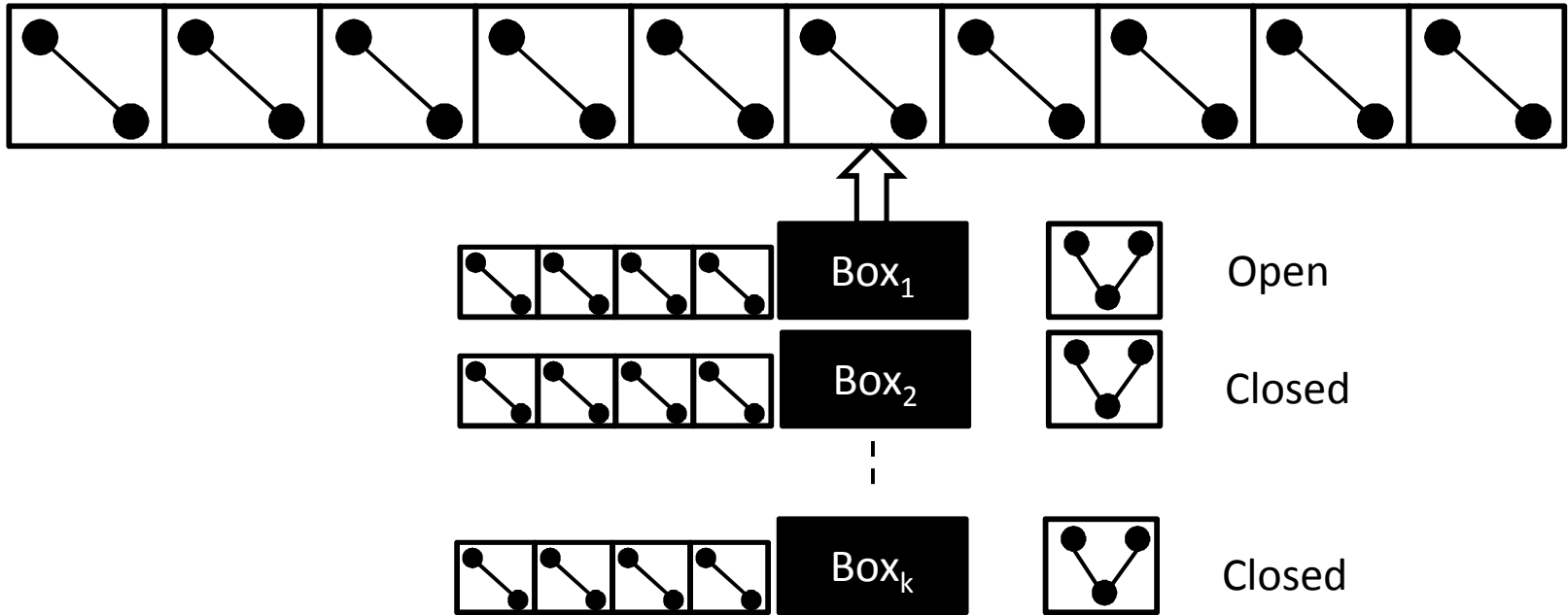


The box



- Each triangle has one future-closed wedge
- Fraction of future-closed wedge = $\tau/3$
- We can estimate this fraction instead, scale by 3

All together



- $O(\sqrt{n})$ space for constant error ϵ
- Result looks nice, cute arguments, we're done!



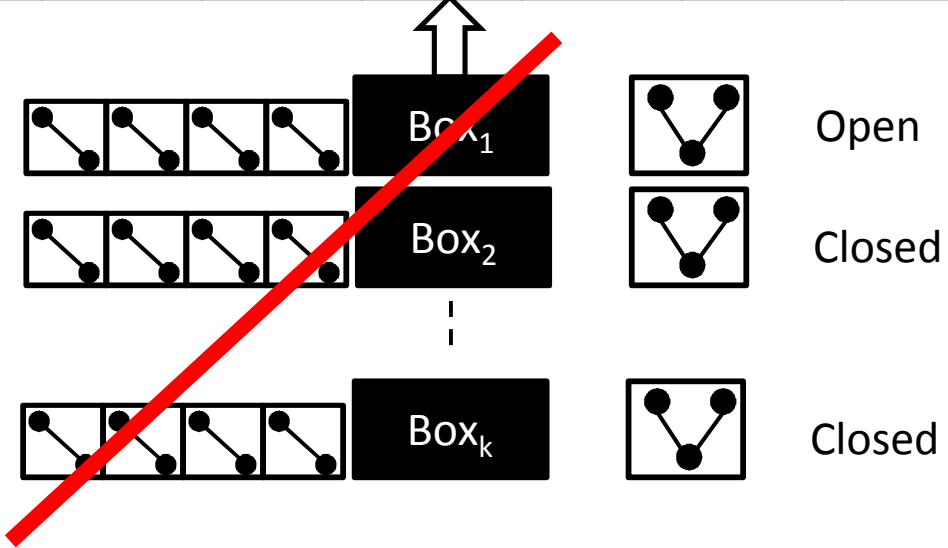
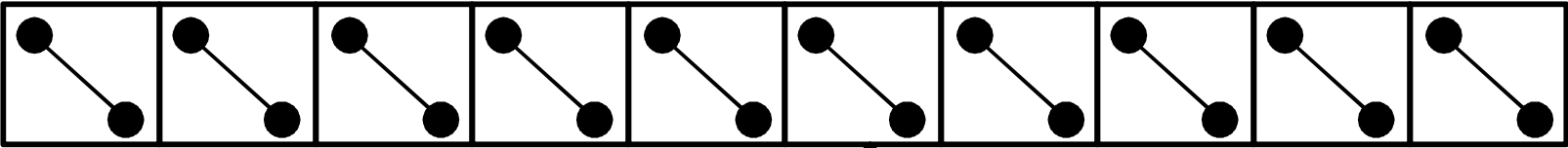
It's awful in practice



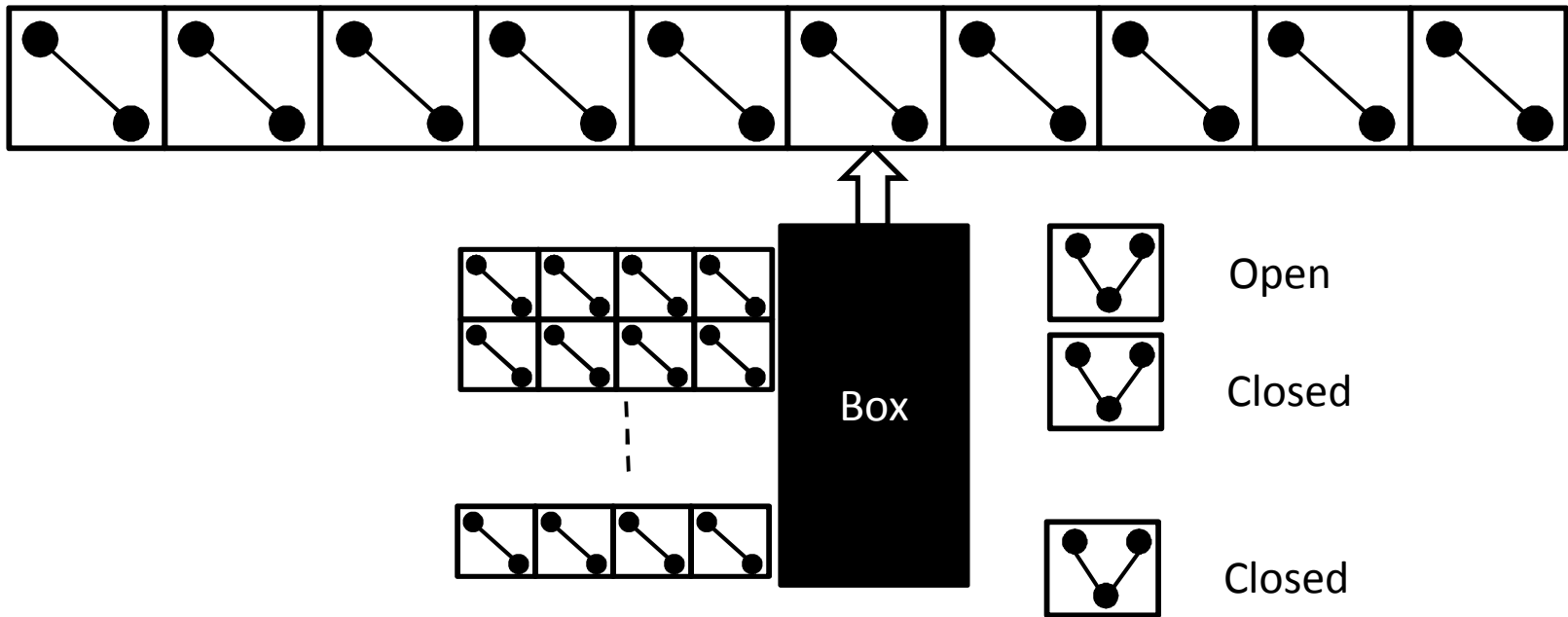
$$\frac{\sqrt{n}}{\epsilon^2} \approx \frac{\sqrt{10^6}}{(0.01)^2} = 10^7$$

Graph with million vertices, desired error of 0.01

Heuristica

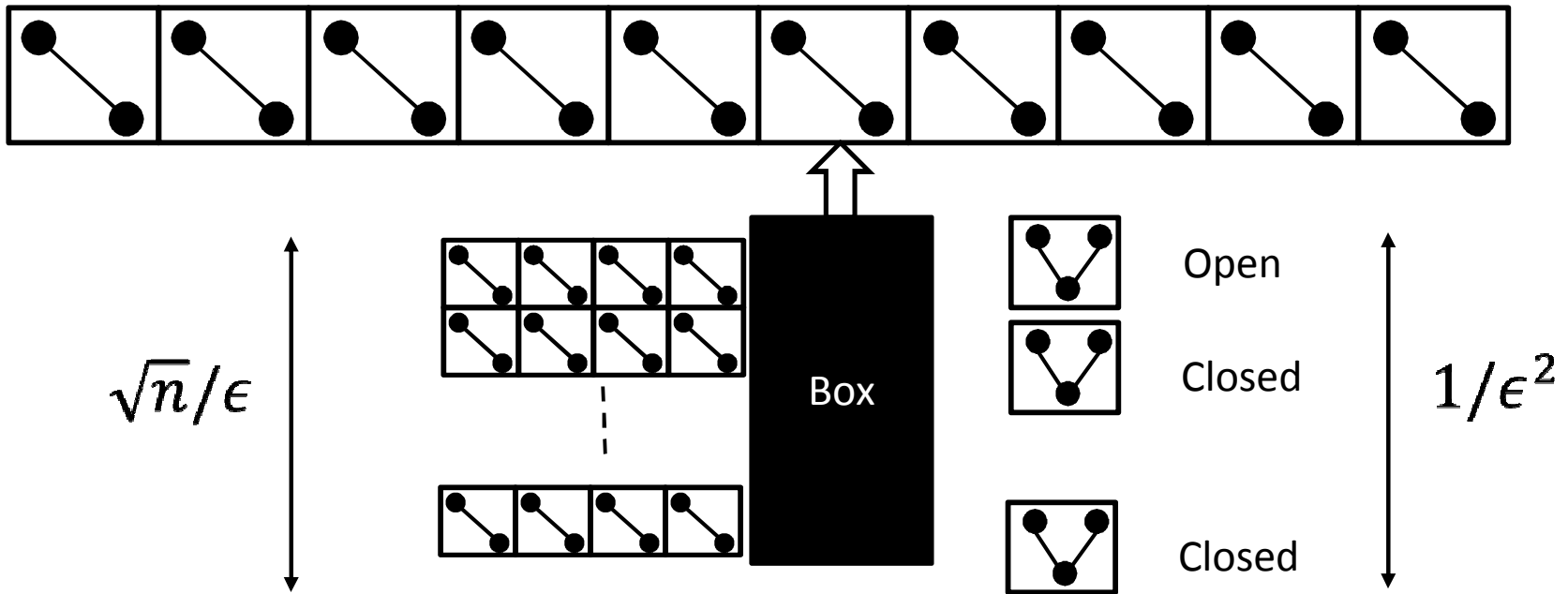


Heuristica

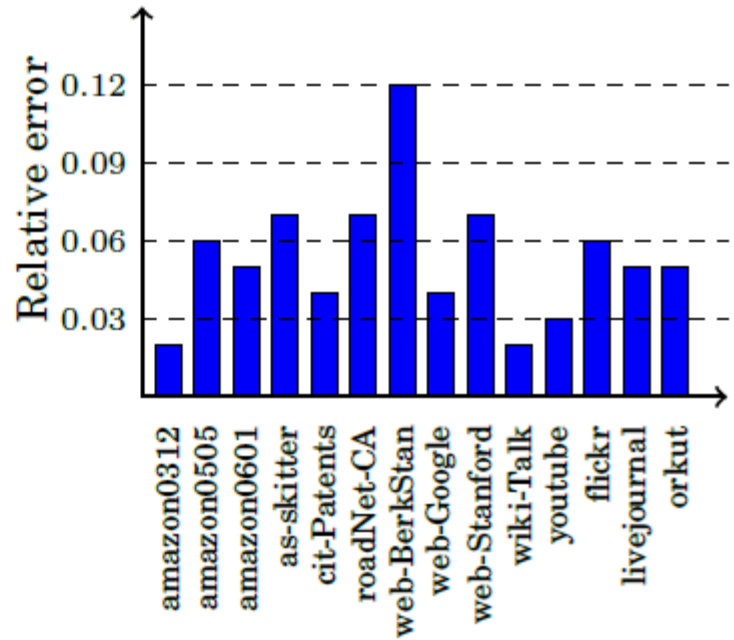
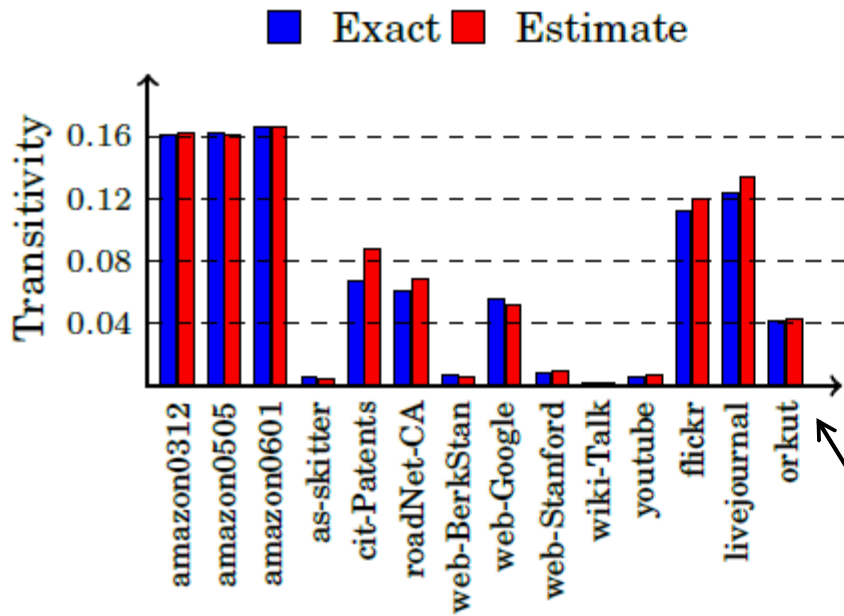
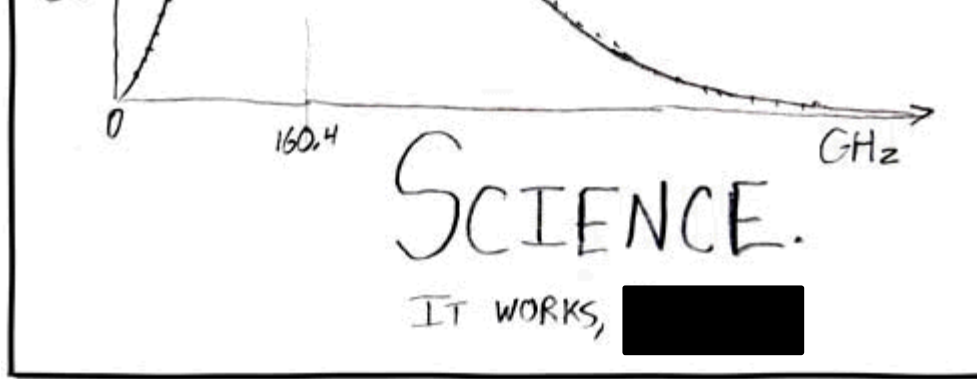


- Chernoff: “To get ϵ error, need $1/\epsilon^2$ wedges”
- Birthday paradox: “If you select $k\sqrt{n}$ u.a.r. edges, you get k^2 u.a.r. wedges w.h.p”
- Chernoff needs independent wedges, birthday gives dependent wedges

Crossing fingers



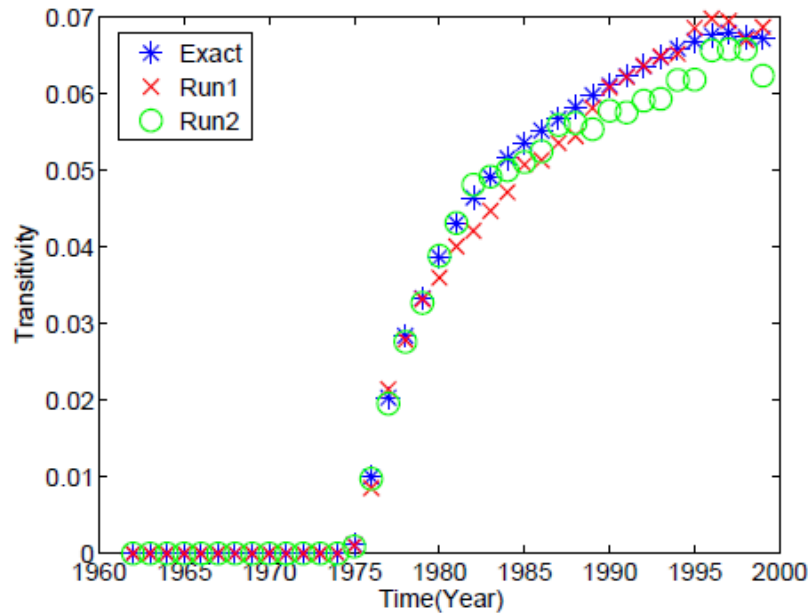
- Assume wedges are sufficiently de-correlated
- \sqrt{n}/ϵ edges suffice. Huge savings!



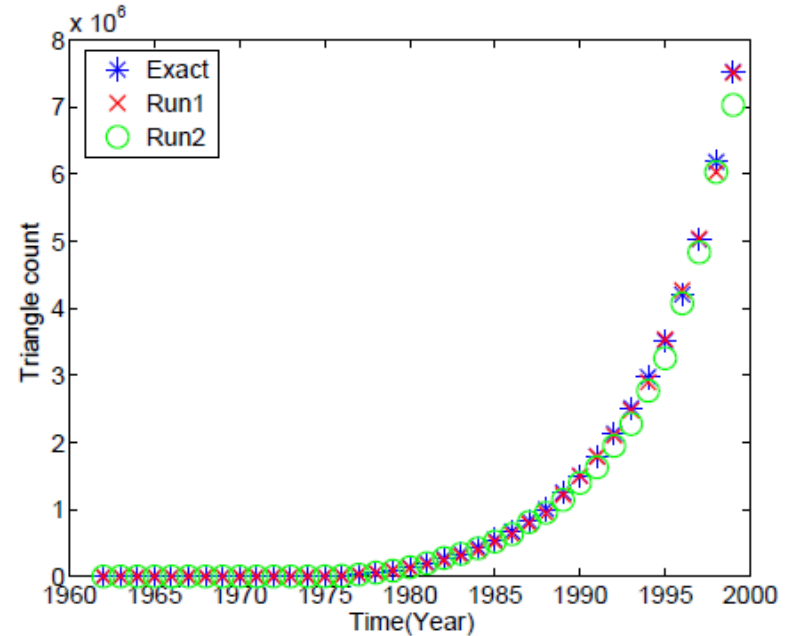
200M edges

Total edge storage = 40K

Something that isn't boring



(a) Transitivity



(b) Triangle count

cit-Patents: 16M edges, 100K edge storage

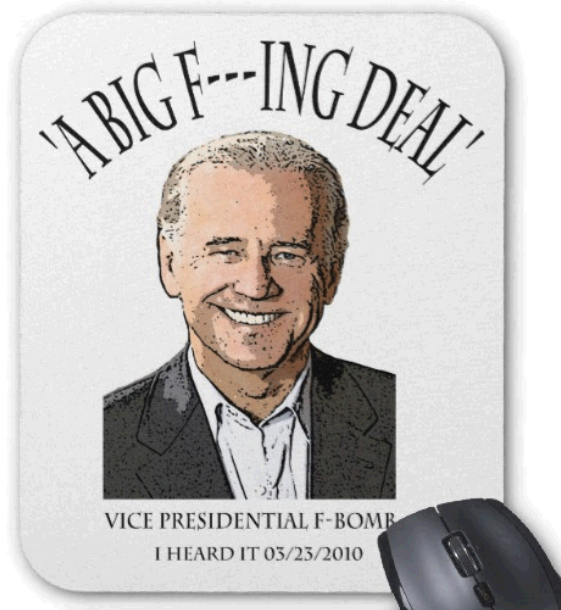


A dialogue



This is really cool!
Some theory and it works!

Darn.



Real graph streams are multi-graphs. And repeated edges will kill your algorithm. And removing repeated edges from a stream is expensive.

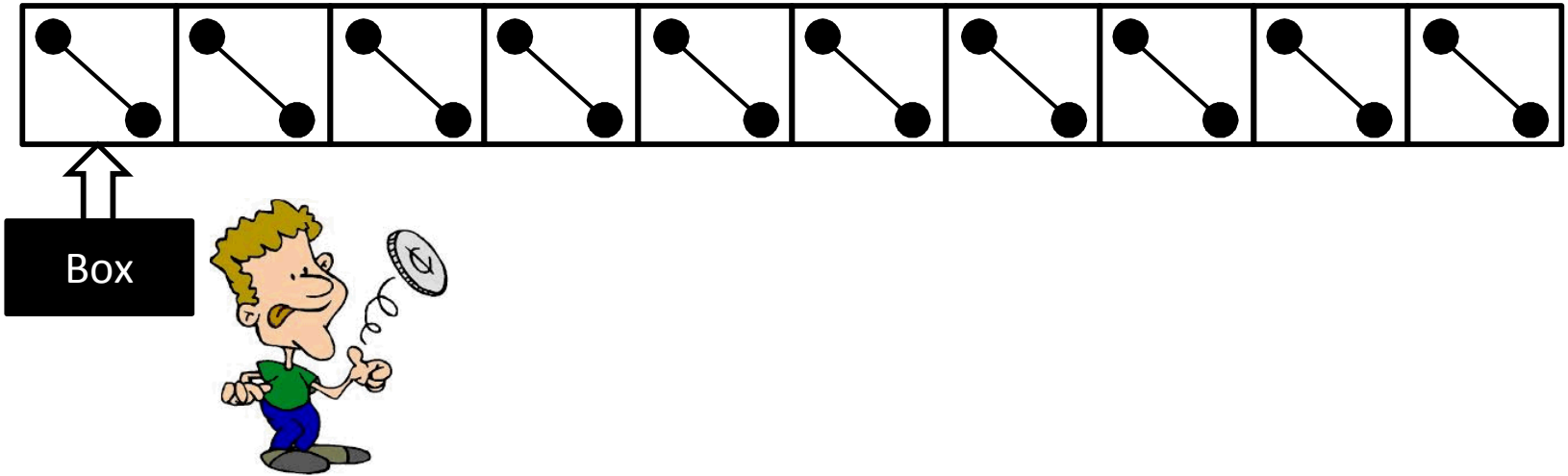
- [\[Jha-S-Pinar13\]](#) Extension to multi-graphs

Who cares?



- Nobody.
No one really “cares”
- We don’t want simply the transitivity, or the number of triangles

More complex quantities



- Estimate T_v , for high degree v ?
- Estimate T_d , triangles incident to deg. d ?
- Sudden jumps in clustering coefficient?
- More vertex/edge based triangle numbers, not global count

Get stronger results

- What we discussed was baby stuff for streaming algorithms theory
- How to use more powerful tools?
 - Andrew's talk...
- [Alon-Matias-Szegedy] type results?
- ℓ_p -sampling anyone?



Bigger pictures?

- [\[Ahmed-Neville-Kompella13\]](#) How to get a representative subsample for streamed graph?
 - That preserves “most structure”
- Our sample only counts triangles. Can you do more with it?
- Our sample has flavor of induced edge sampling...?

Thanks!

