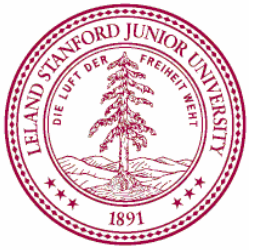
What I Did Last Summer

LINGOs, GPUs, and Monitoring Vertex

Imran Haque
Department of Computer Science
Pande Lab, Stanford University

http://cs.stanford.edu/people/ihaque
http://folding.stanford.edu
ihaque@cs.stanford.edu



Folding@home distributed computing

CUP XI
10 March 2010

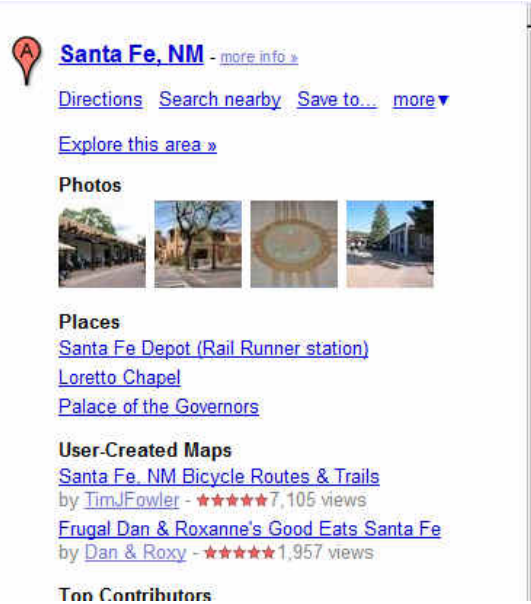
A Dead White Guy

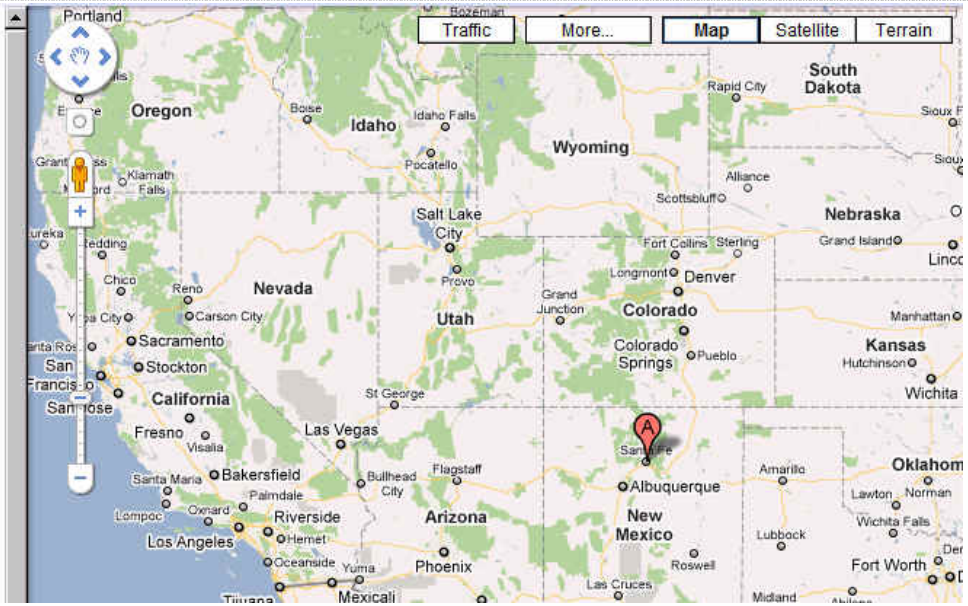
Ronnie Van Zant,
Lead singer, Lynyrd Skynyrd

Won't you give me three steps,
Gimme three steps mister,
Gimme three steps towards the door?
Gimme three steps
Gimme three steps mister,
And you'll never see me no more.

"Gimme Three Steps"

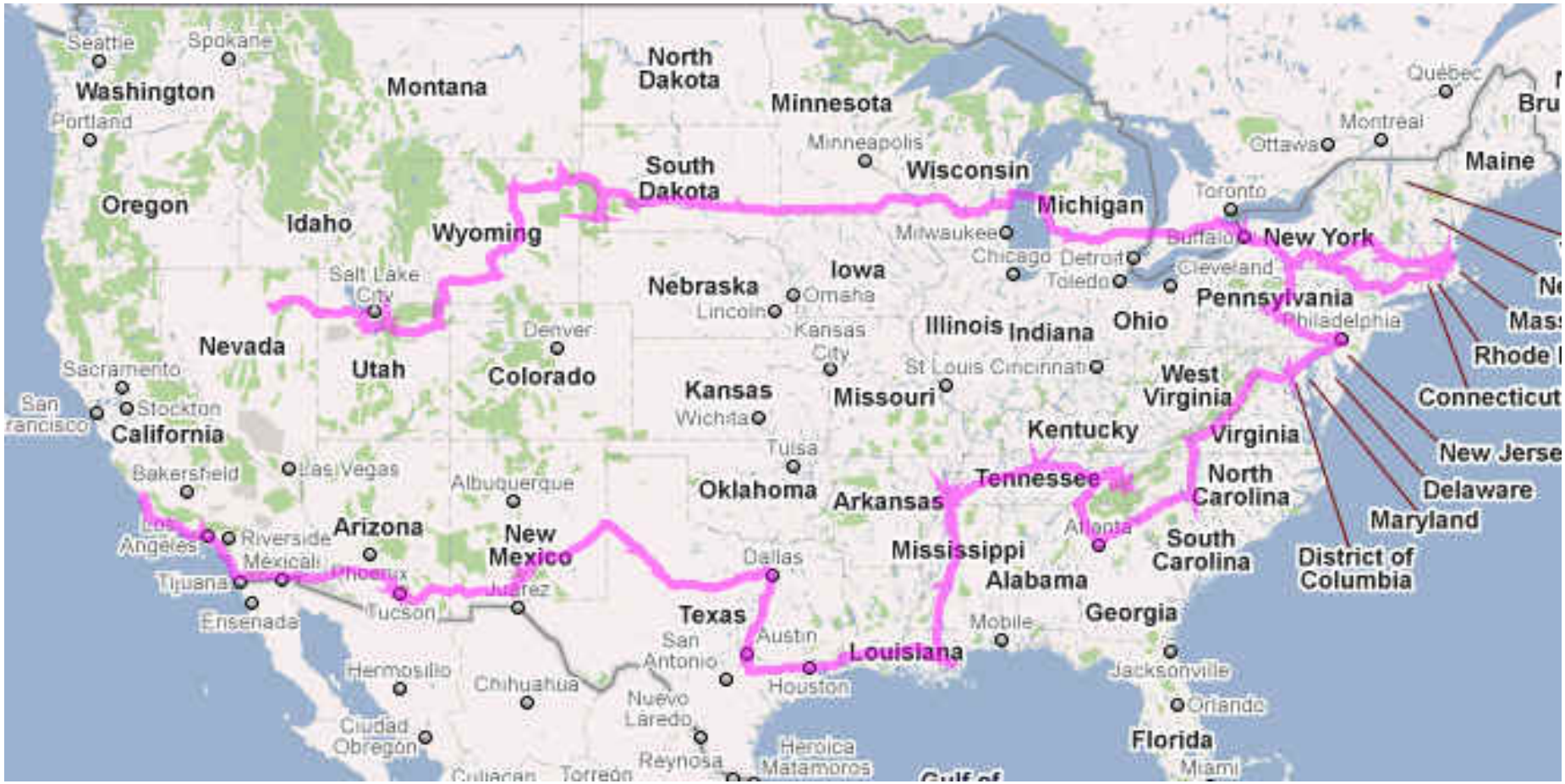
It Began in Santa Fe…





LINGO's kind of slow…
think you could put it on a GPU?

…and continued in Cambridge



A Survey of GPU-enabled Comp Chem

Package Name	Application	Authors/First Release	Speedup
VMD	Ion placement	Stone et al., 2007	100x
TeraChem	Two-electron integral (quantum chem)	Ufimtsev and Martinez, 2008	130x
Folding@home/ OpenMM	Molecular dynamics	Friedrichs et al., 2009	735x
PAPER	3-D chemical similarity	Haque and Pande, 2009	30x
	Poisson-Boltzmann solvation	Narumi et al., 2009	40x
VMD	MO Display	Stone et al., 2009	125x
GPU ROCS	3-D chemical similarity	OpenEye, 2010	120x
SIML	**1-D chemical similarity**	**Haque, Pande, and Walters, 2010**	**83x**

Introduction to LINGO

- "1-D" similarity method comparing canonical SMILES strings of molecules by fragmentation:

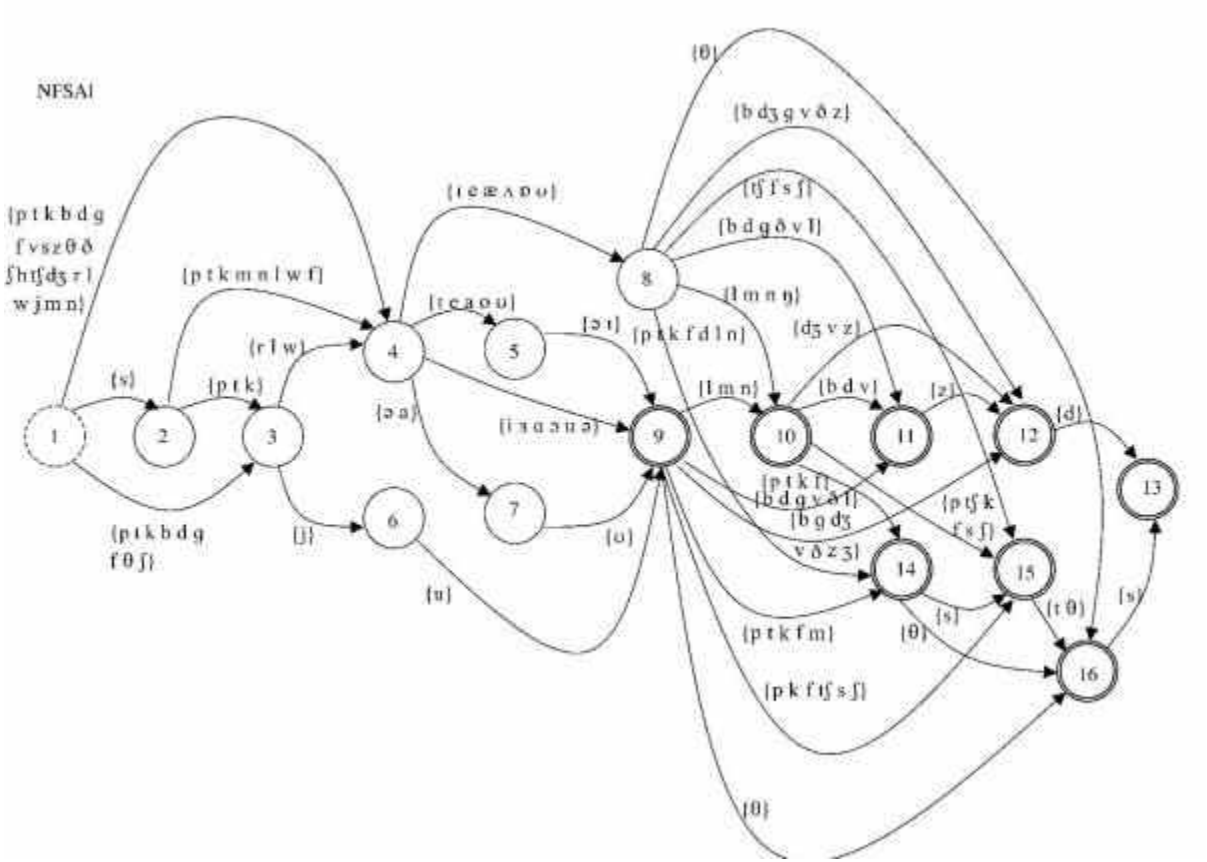
  Benzene -> c1ccccc1 -> [c1cc, ccc1, 1ccc, cccc, cccc]

  Pyridine -> n1ccccc1 -> [n1cc, ccc1, 1ccc, cccc, cccc]

$$T_{A,B} = \frac{|A \cap B|}{|A \cup B|} \qquad T_{A,B} = \frac{\sum_{i=1}^{\ell} \left(1 - \frac{|N_{A,i} - N_{B,i}|}{N_{A,i} + N_{B,i}}\right)}{\ell}$$

- Efficient implementation by Grant et al. (2006): build DFA from reference string (O(N)), run query strings through automaton to calculate Tanimoto (also O(N))

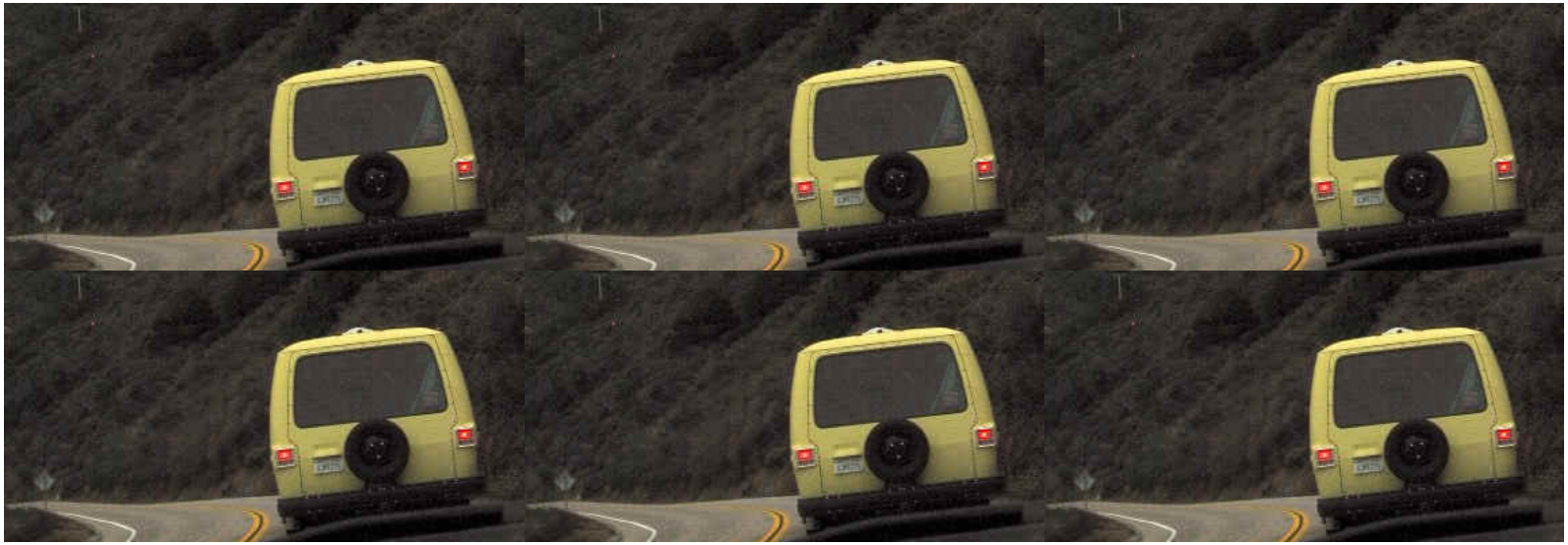
DFAs, Big Yellow Vans, and GPUs



DFAs, Big Yellow Vans, and GPUs

DFAs, Big Yellow Vans, and GPUs

DFAs, Big Yellow Vans, and GPUs



How to get me to work on LINGOs

A New LINGO algorithm

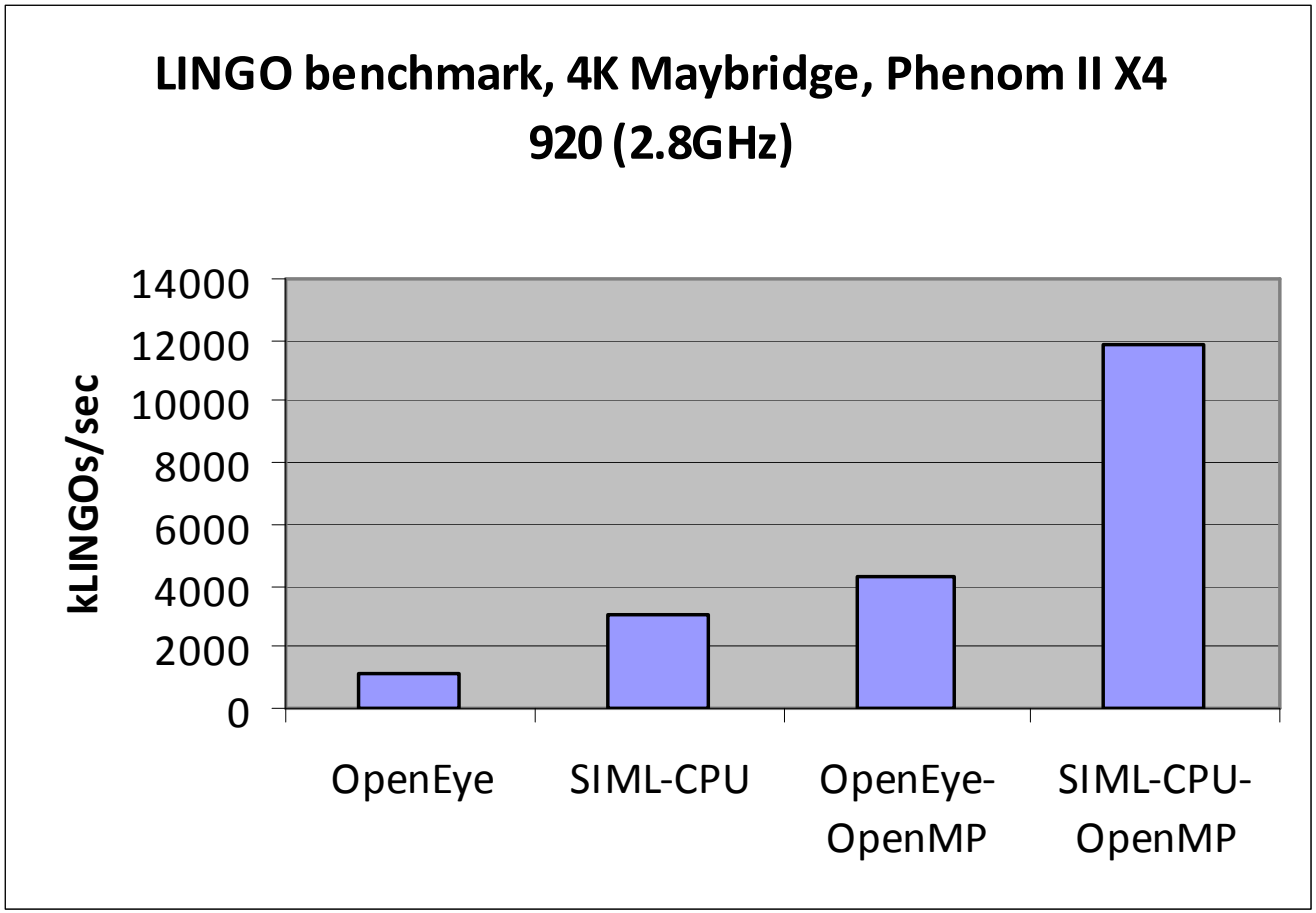
- Pack LINGO substrings into 32-bit integers
- Molecules → sorted lists of integers (and counts)

$$T_{AB} = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

- Calculate intersection by algorithm similar to merging sorted lists: simple control logic, cache friendly

Haque IH, Pande VS, Walters WP. *JCIM* ***2010***

CPU Performance



SIML-CPU 2.75x the speed of OELingoSim…

but this is no good for GPUs

Mo’ monitors, mo’ faster

GPU Memory Optimizations

Mol 1	@H](	C(C)	C=C)	NH0+	[NH0	COc0	ccc0	0CC=
Mol 2	](C(	@H](	ccc(	cc0)	(=O)	O-])	)[O-	(cc0
Mol 3	COC(	@H](	ccc(	cc0)	(=O)	](c0	(cc0	ccc0
Mol 4	NNC(	(=O)	O)c0	ccn0	CC(=	NC(=	)NNC	=O)N

Row-major layout is fine for the (non-vectorized) CPU because we can rely on cache to bring in partial rows for each core…

… but *kills* GPU performance

Haque IH, Pande VS, Walters WP. *JCIM* **2010**

GPU Memory Optimizations

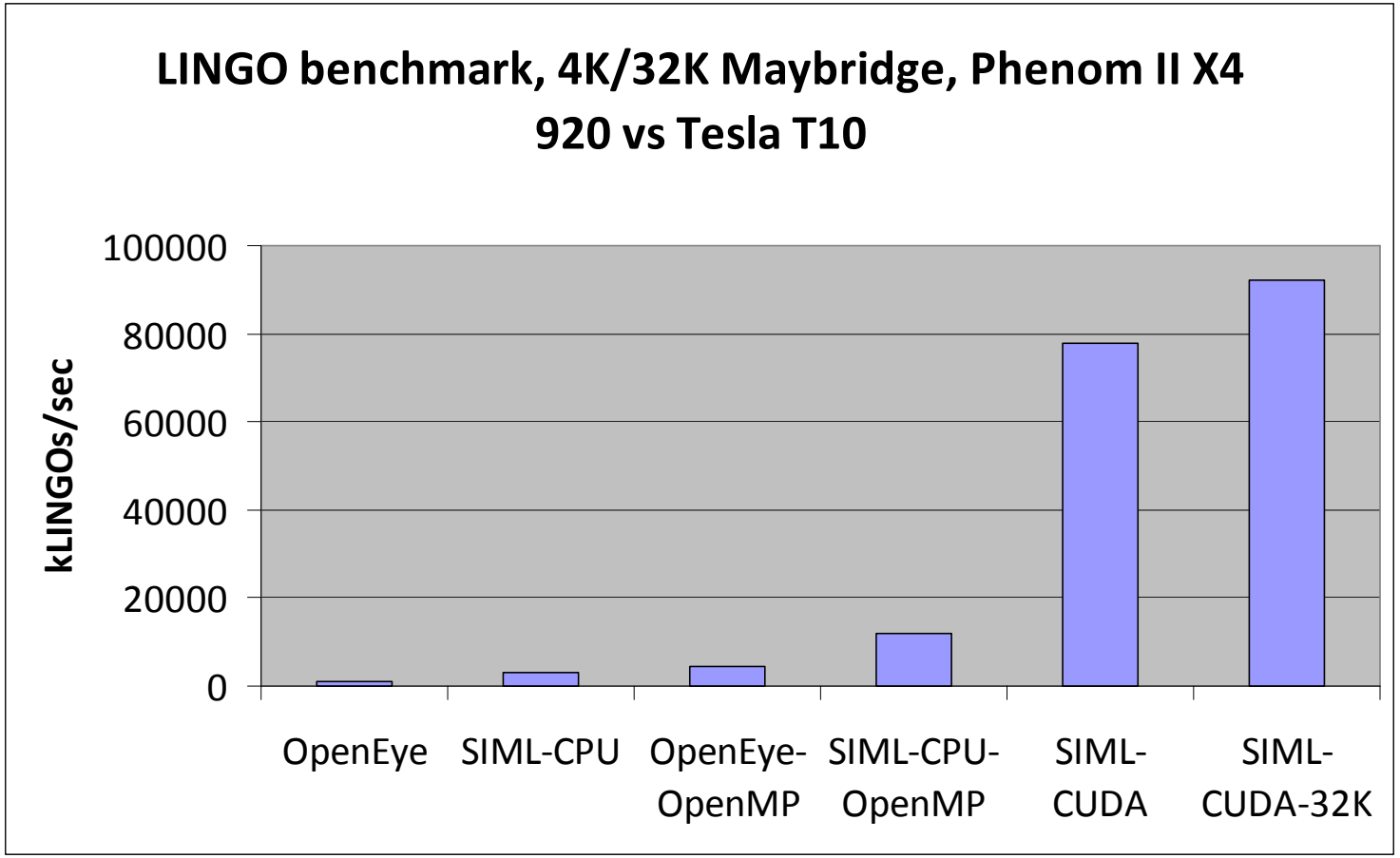
Mol 1	Mol 2	Mol 3	Mol 4
@H](	](C(	COC(	NNC(
C(C)	@H](	@H](	(=O)
C=C)	ccc(	ccc(	O)c0
NH0+	cc0)	cc0)	ccn0
[NH0	(=O)	(=O)	CC(=
COc0	O-])	](c0	NC(=
ccc0	)[O-	(cc0	)NNC
0CC=	(cc0	ccc0	=O)N

Transposing molecule layout to column-major maximizes spatial locality among threads.

Can barrier on each row to guarantee coalescing, or use a 2D texture (if available on hardware) for more speed.

Haque IH, Pande VS, Walters WP. *JCIM* **2010**

GPU Performance



SIML-CUDA 83x the speed of OELingoSim…more like it!

(alpha version of SIML-OpenCL is *only* about 22x speedup)

Can I Have a DVD Now Please, Ant?

Who needs something this fast?

- 1K-10K molecules passes for a “large” data set in lit
- **Actually** large chemical databases:

  PubChem: 31M
  ZINC: 34M
  GDB-13: 970M

- Assay data exists for hundreds of thousands of mols

- I’m interested in doing **very** large-scale data integration

Example Application: PubChem

- All-vs-all, ~19.5M compounds, OE Isomeric SMILES
  380 x 10^{12} Tanimotos = 0.63 nmol

- Get neighbors at 4σ to define neighbor graph
- Histogram full matrix to choose significance cutoff
- Interesting graph properties?

```
Starting GPU computation on Compound_02800001_02825000.ism.gz
Took 91.01 ms to initialize GPULingo object with query SMILES matrices
Starting Tanimoto computation...
    Took 826.22 ms to compile 22851 query SMILES (for next frame)
    Took 447.43 ms to sync last tile (Compound_02775001_02800000.ism.gz)
Took 4844.04 ms to compute and histogram 24949 x 24436 Tanimotos on GPU
(125856 kLINGO/sec)
```

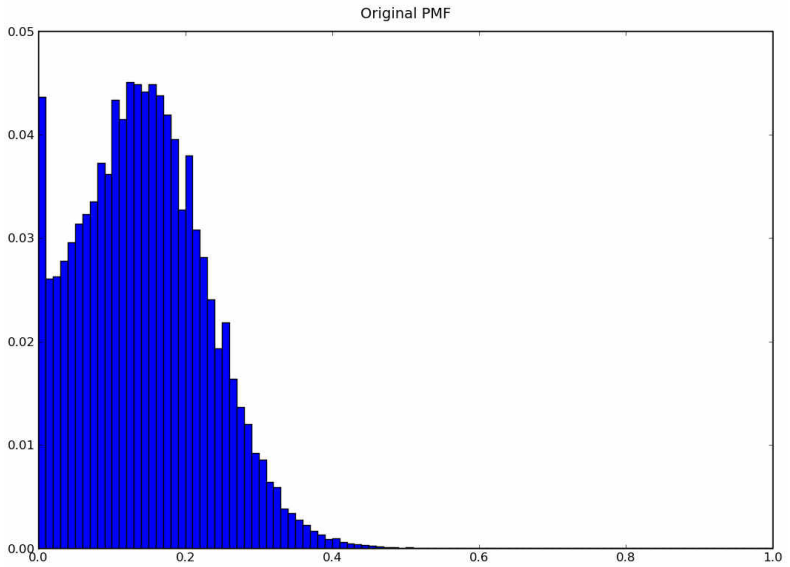
Similarity Properties of PubChem

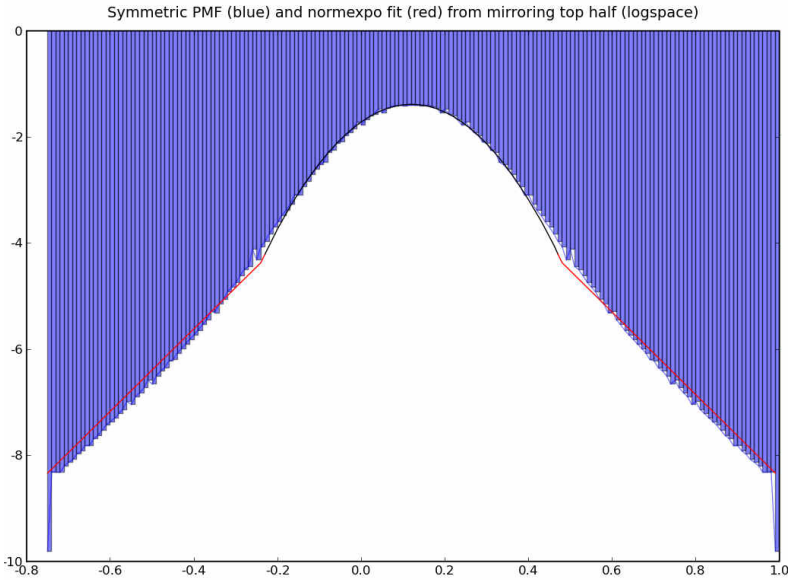
- Tanimoto distribution

  4σ-equivalent at T=0.56

  24.5 x 10^9 edges in graph
  (380 x 10^{12} possible)

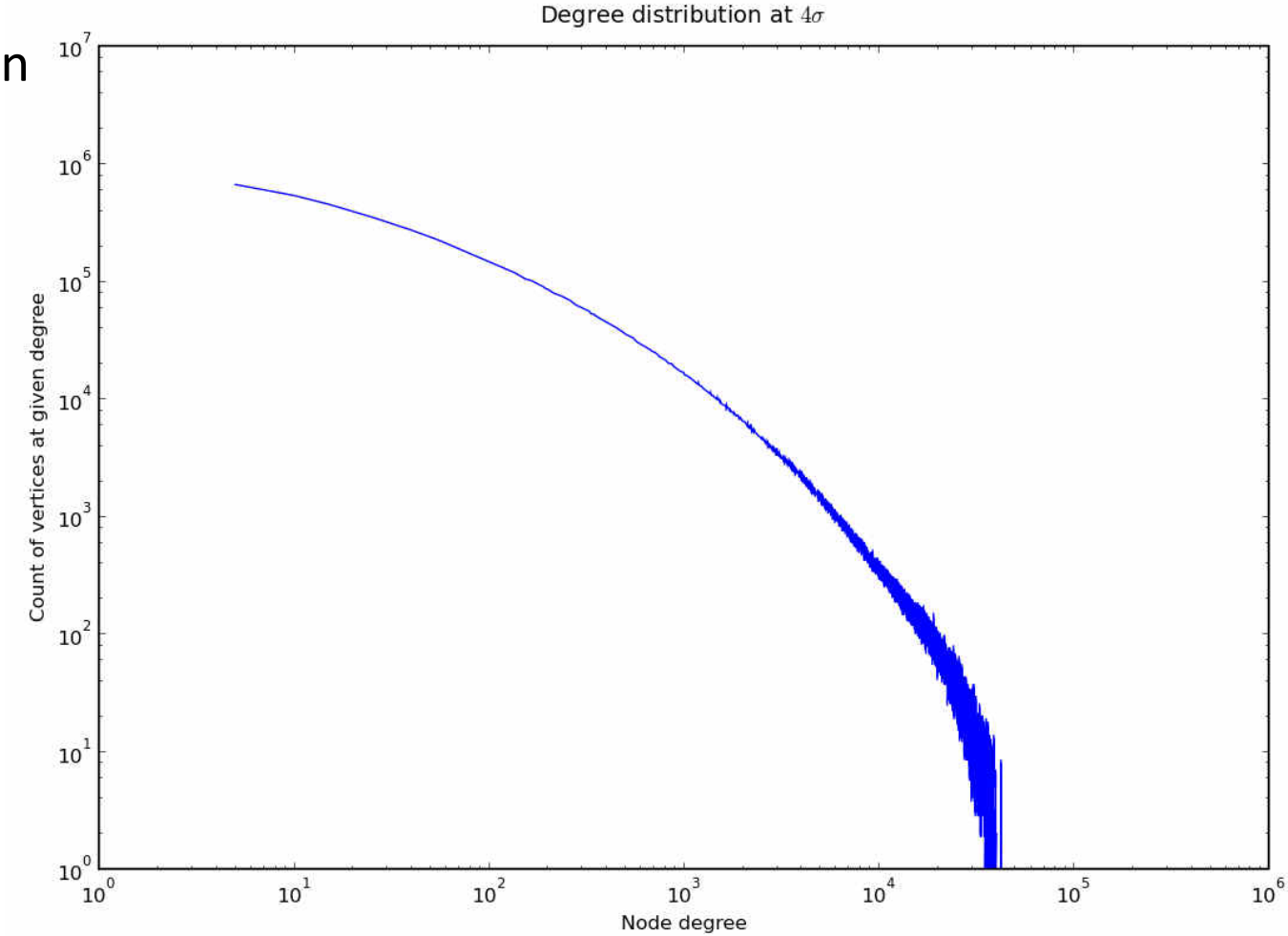
- Distribution is *not* normal in the positive tail





Graph Properties of PubChem – History?

- Degree distribution

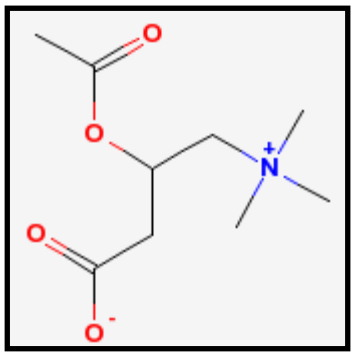


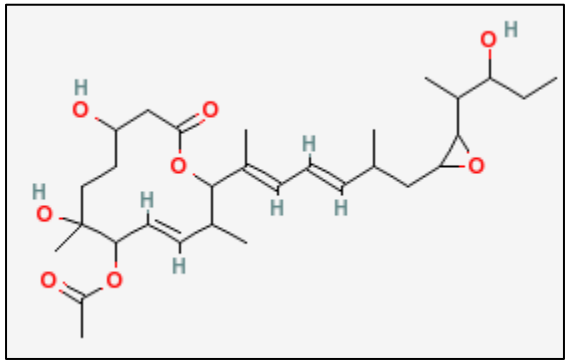
Not power-law (scale-free, preferential attachment process)

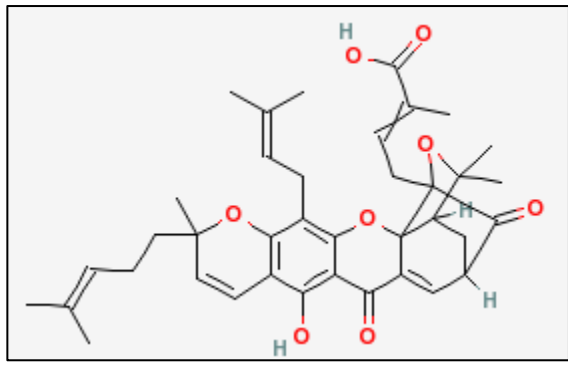
Not Poisson (random)

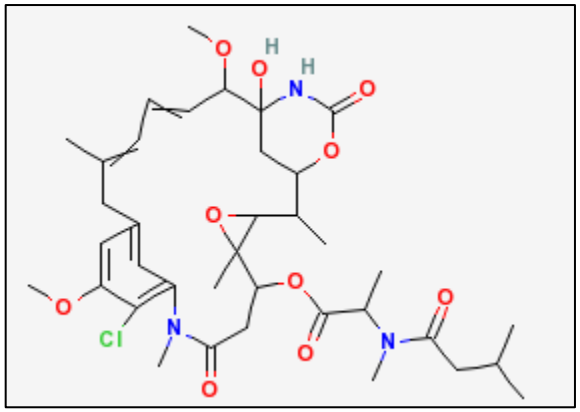
Connected Components of PubChem

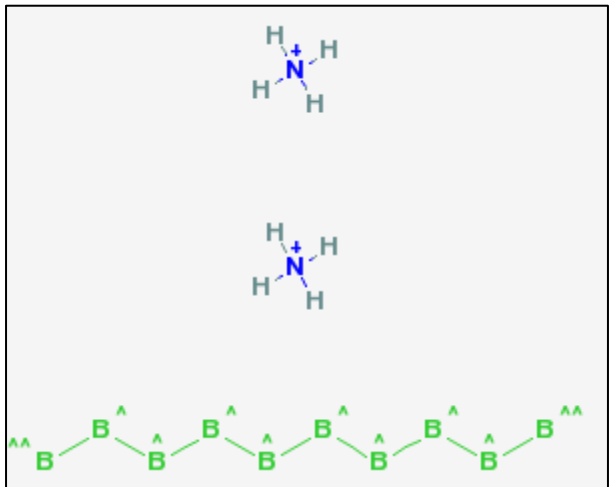
- Not all molecules are reachable from each other
- 1st CC: 19,113,304 molecules; 2nd CC: 692





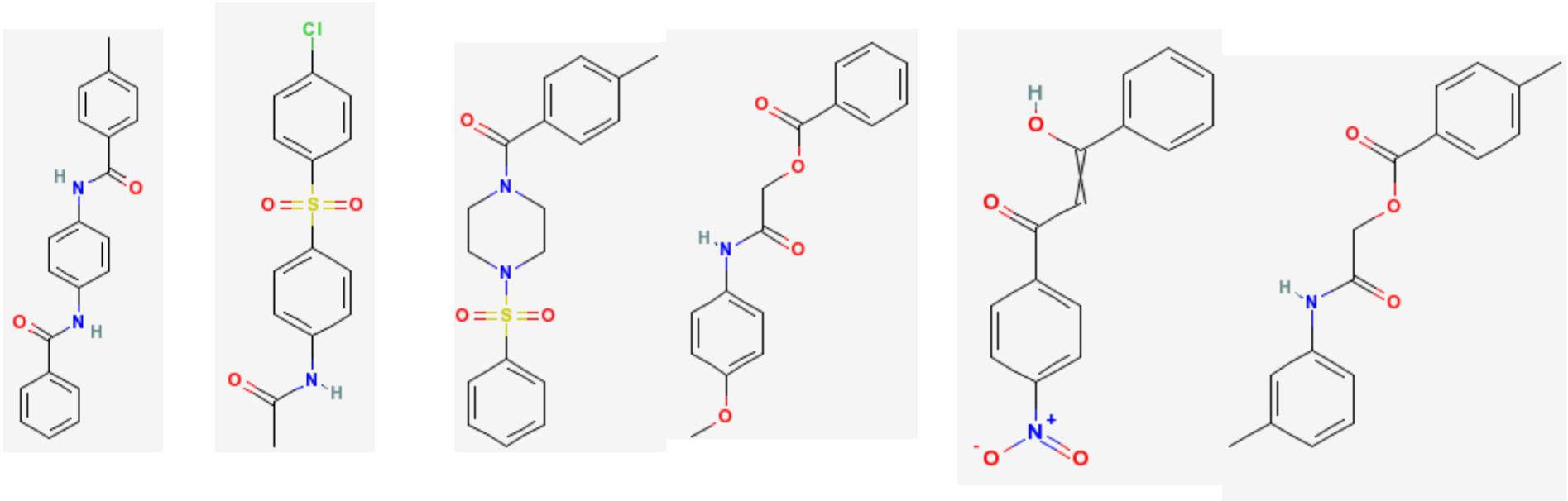






Graph Clustering of PubChem

- Clustered using dbclus algorithm
- Cluster sizes: 104973, 69338, 61069, 51944, …
- Limitations of LINGO



Acknowledgments

Stanford

- Vijay Pande (PI)
- Paul Novick
- Del Lucent
- Peter Eastman
- Mark Friedrichs
- Randy Radmer

Collaborators

- Pat Walters
- Kim Branson
- Brian Goldman
- Richard Dixon
- John Chodera
- Michael Houston
- Anthony Nicholls
- Roger Sayle

Conclusion

- Got a large-scale LINGO problem? Try SIML. BSD (permissive open-source) license, Python bindings:

https://simtk.org/home/siml

- There's (hopefully?) interesting information about chemical space and chemical biology to be learned from large-scale cheminformatics

- **Any and all monitor donations will be accepted.**