BERKELEY LAB
Bringing Science Solutions to the World

U.S. DEPARTMENT OF ENERGY

Approximately and Efficiently Estimating Dynamic Point-to-Point Shortest Path

Berkeley UNIVERSITY OF CALIFORNIA

Georgia Tech

Alok Tripathy, Oded Green

University of California, Berkeley / Lawrence Berkeley National Lab

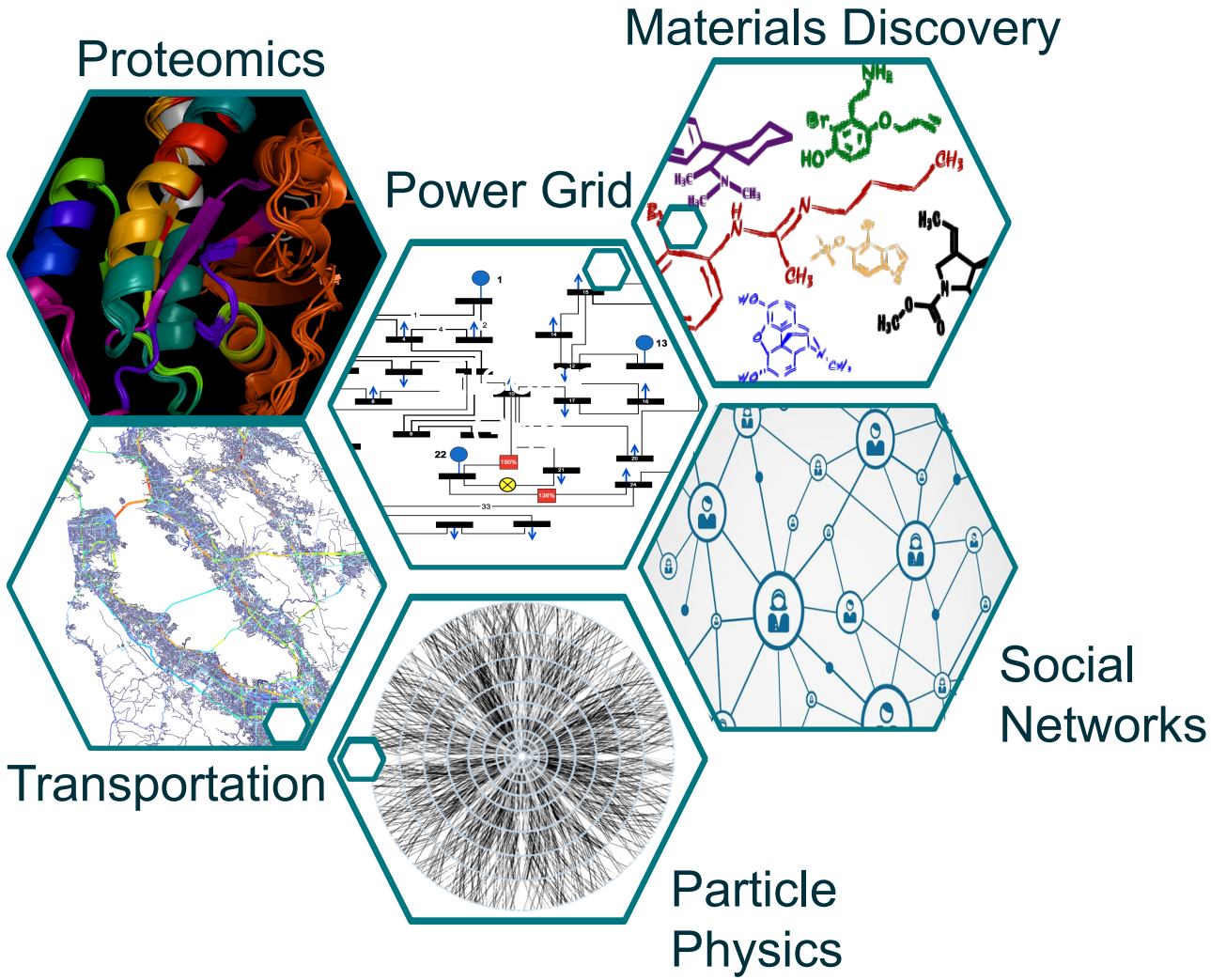
NVIDIA / Georgia Institute of Technology

NVIDIA

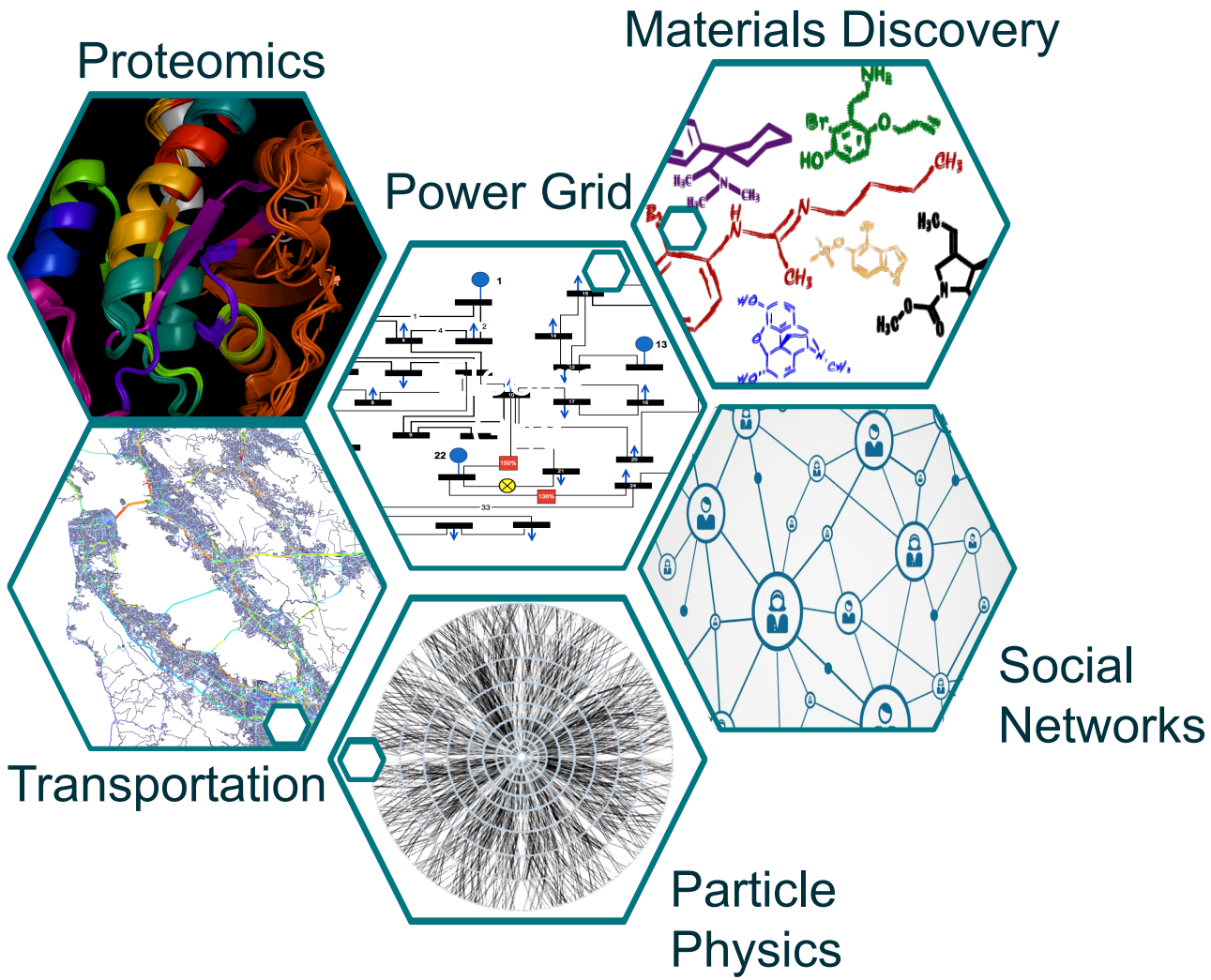
NSF

December 10th, 2020

Why focus on graphs?

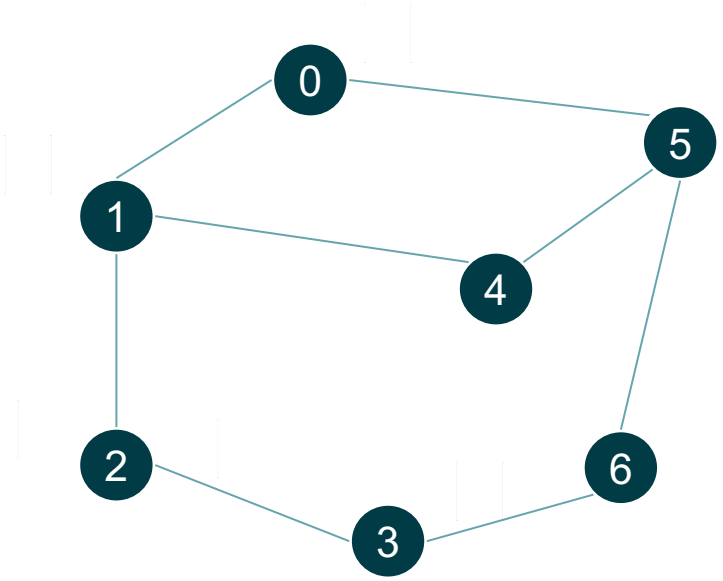


Analytics on graphs



- Single-Source Shortest Path
- All-Pairs Shortest Path
- **Point-to-Point Shortest Path**

Point-to-Point Shortest Path (PPSP)



Input Queries

$q(1,6)$	$q(1,0)$	$q(1,5)$	...

Output Paths

3	1	2	...

- Series of $s - t$ shortest path queries

PPSP Issues

- Static graphs
  - » Exact case can be expensive – approximations might suffice
  - » Opportunity to preprocess graph
- Dynamic graphs
  - » Shortest paths can change wildly with new edges
  - » Need to efficiently handle graph updates

What do we do for PPSP?

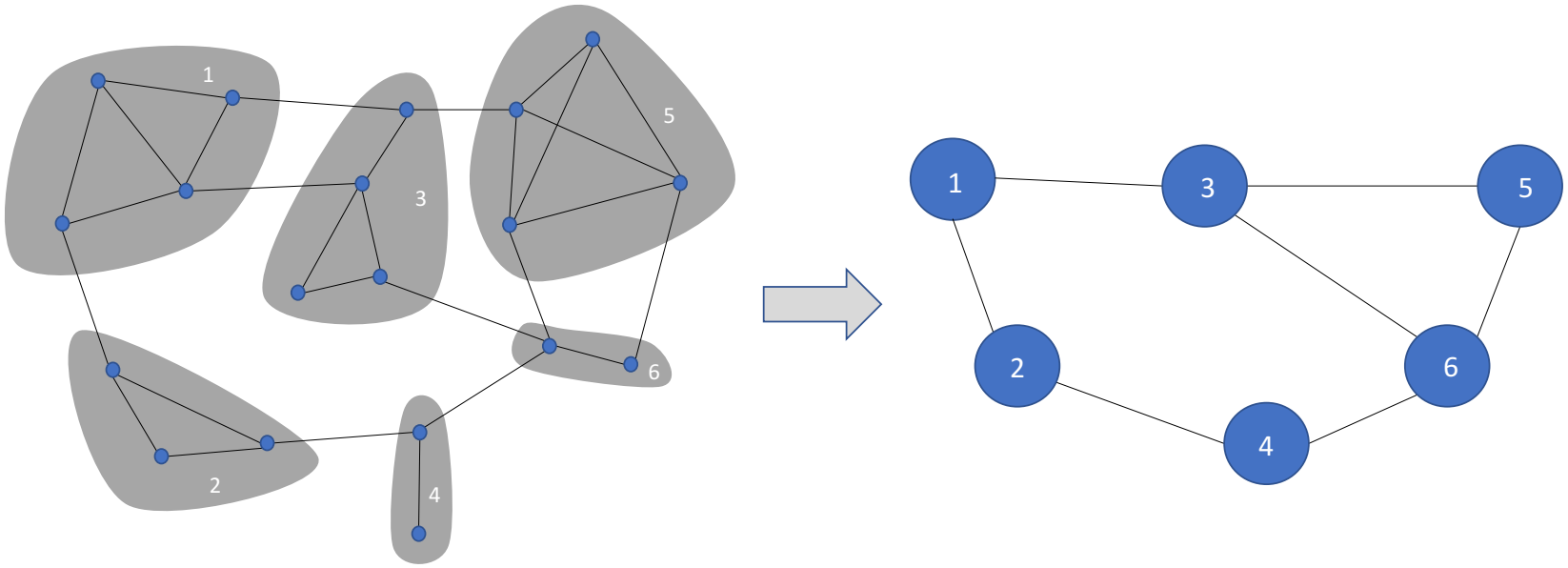
- Two algorithms

- Static graph algorithm
  - » Solves approximate PPSP
  - » Leverages opportunity to preprocess graph

- Dynamic graph algorithm
  - » Extends static graph alg. to dynamic graphs

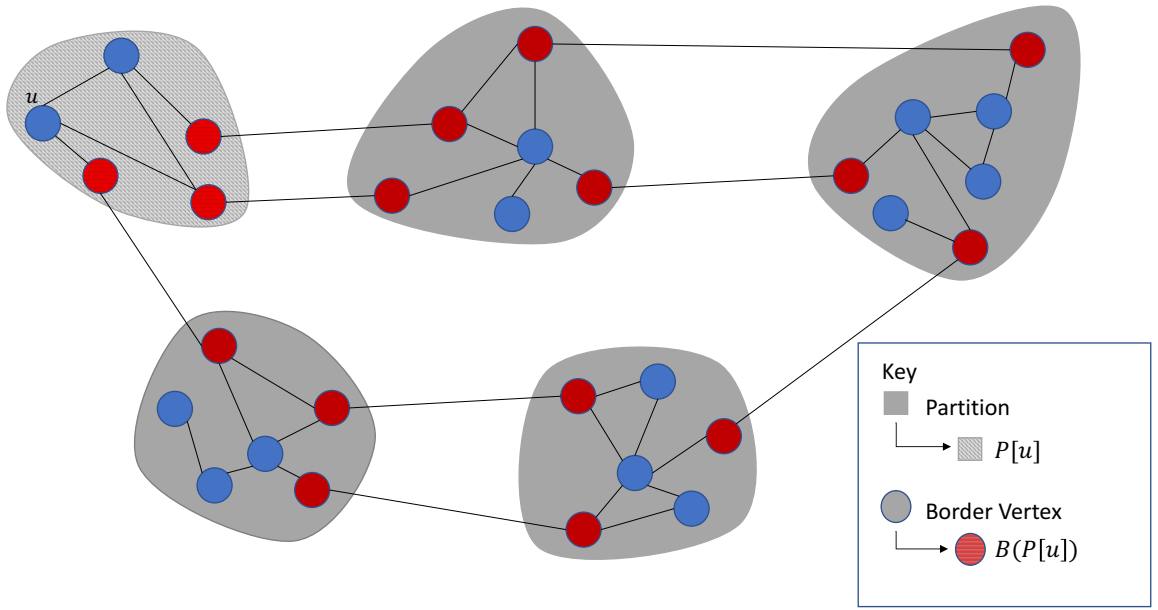
Static Graph PPSP Algorithm

Static PPSP Algorithm: Preprocessing



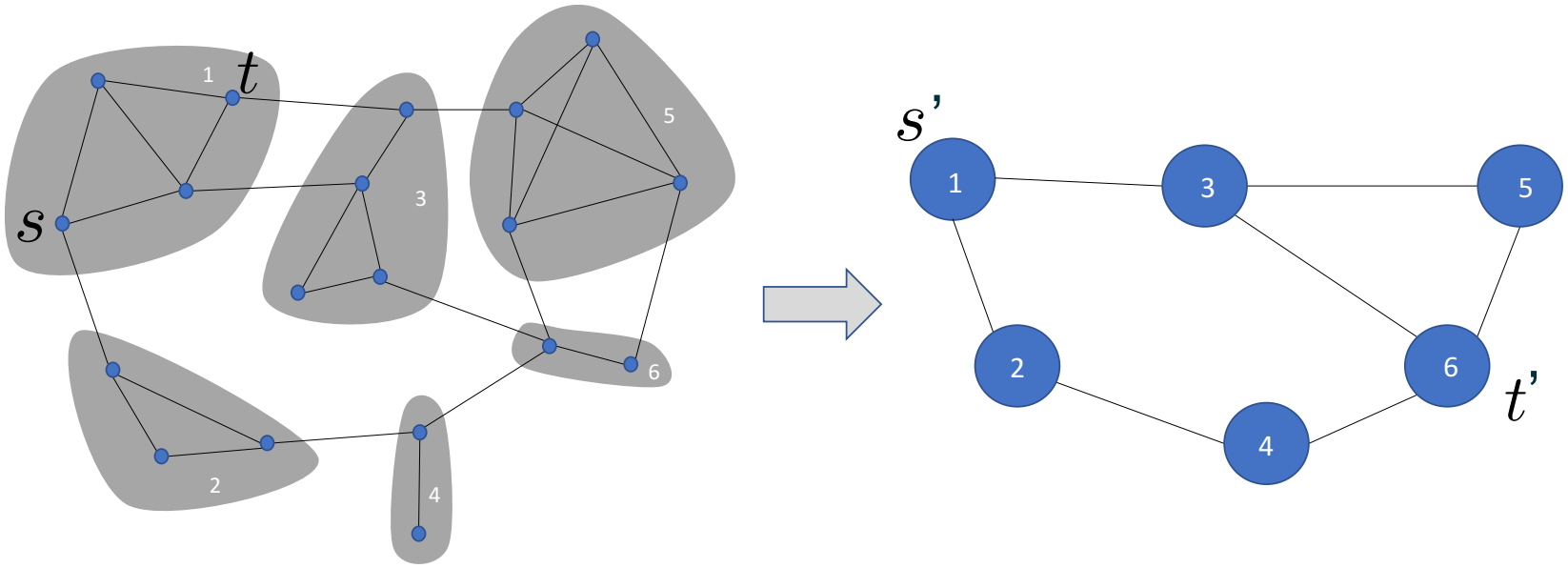
- Create supergraph from graph partitioning
  - » Partitioning time amortized by queries
- Operate either at partition-level or supergraph-level

Static PPSP Algorithm: Preprocessing distances



- Partition-level: Precompute border vertex distances within partition
  - » Parallelizeable
  - » Could do APSP instead
- Supergraph-level: APSP

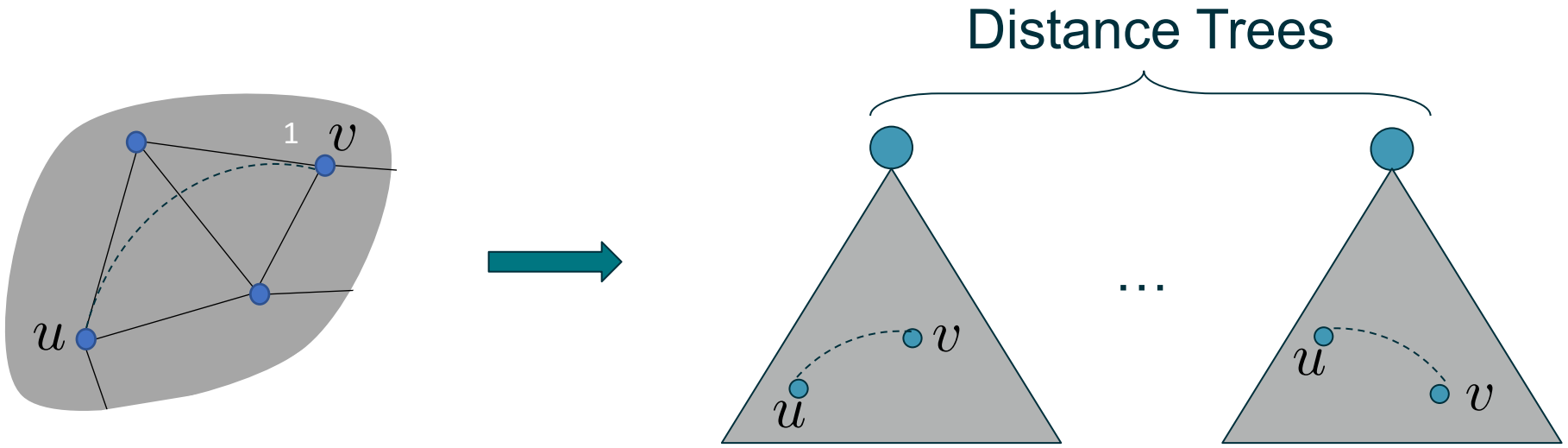
Static PPSP Algorithm: Querying



- Two simple cases for $q(s,t)$
  - » Intra-partition – leverage precomputed border vertex distances
  - » Inter-partition – traverse supergraph
- Every query is handled at the partition-level or supergraph-level

Dynamic Graph PPSP Algorithm

Dynamic PPSP Algorithm: Updates



- Two simple cases for inserting (u, v)
  - » Intra-partition – update distances with dynamic BC alg on partition
  - » Inter-partition – same as above, but on supergraph
- Every update is handled at the partition-level or supergraph-level

O. Green, R. McColl, and D. Bader. A Fast Algorithm For Streaming Betweenness Centrality. In *4th ASE/IEEE International Conference on Social Computing (SocialCom)*, 2012.

Performance of Static and Dynamic PPSP Algorithms

Implementation Details

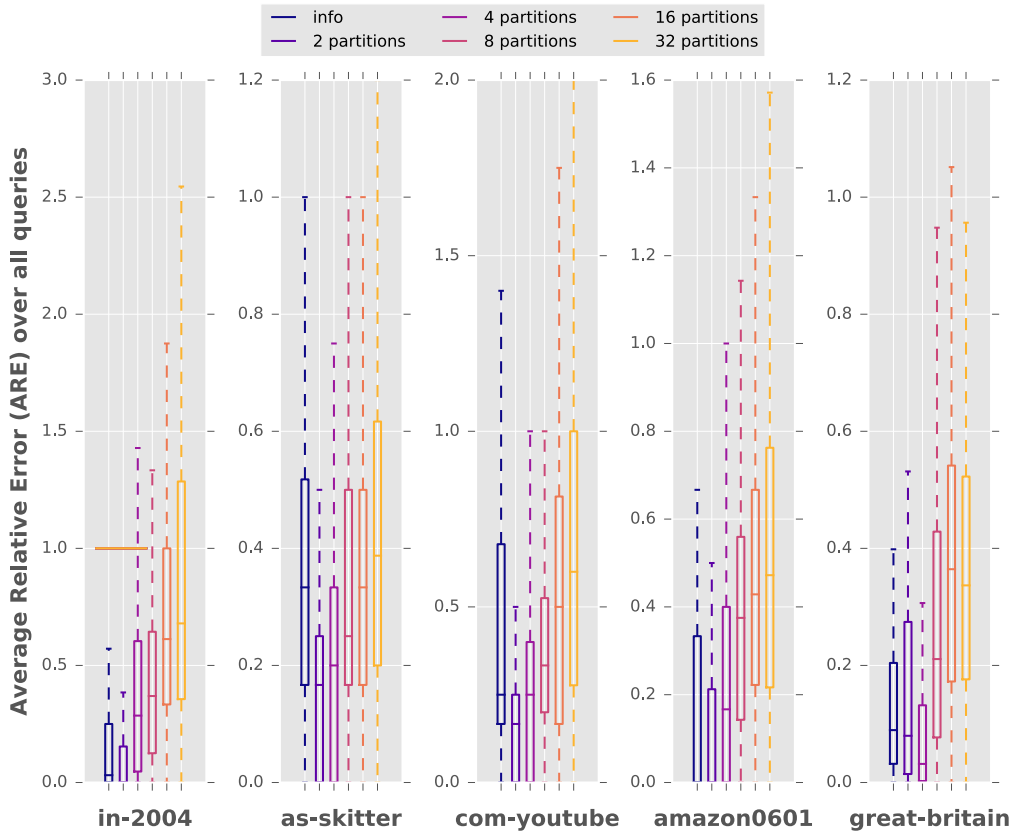
- OpenMP shared-memory implementation
- System:
  - » 56 core Intel Xeon E7-4850v3 @ 2.80GHz
  - » 2TB of DDR4 RAM

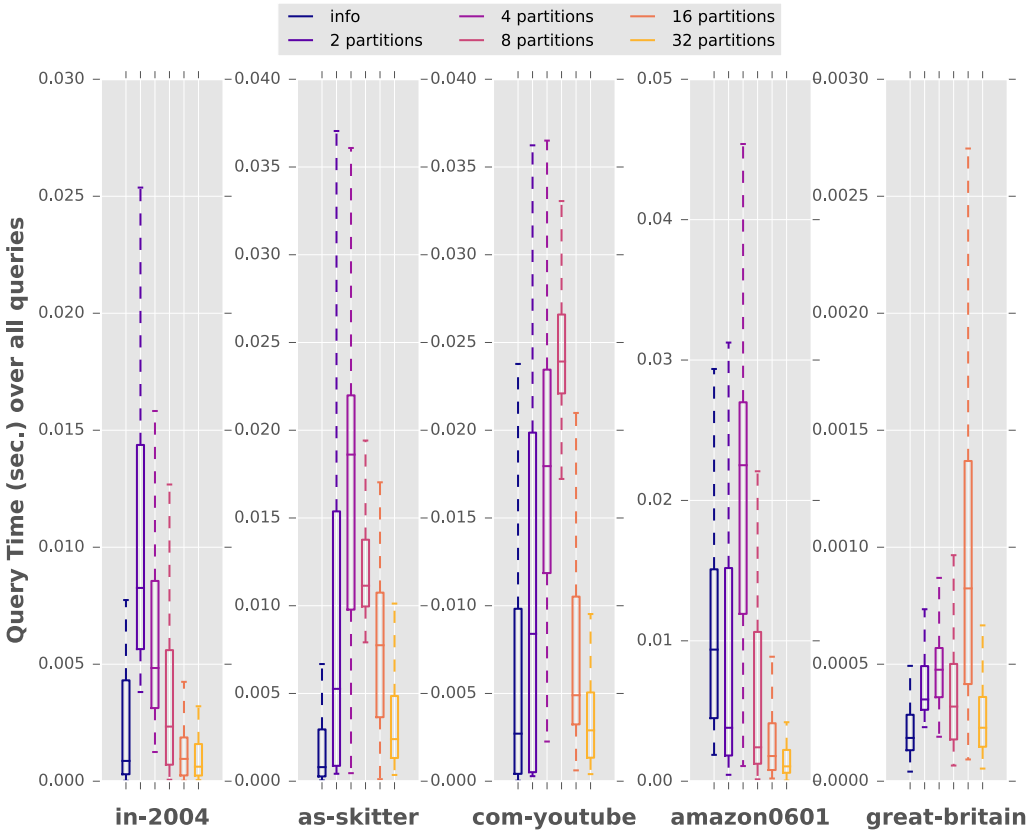
- Partitioners: METIS (2-32 partitions), Infomap
- Datasets:

Table 3: List of Networks Used

Networks for Experiments				
Graph	Type	$\|V\|$	$\|E\|$	Infomap Partitions
in-2004	Internet	1.3M	13.5M	53
as-skitter	Internet	1.6M	11.0M	182
com-youtube	Social	1.1M	3.0M	951
amazon0601	Internet	0.40M	3.4M	3
great-britain	Road	7.7M	8.2M	8

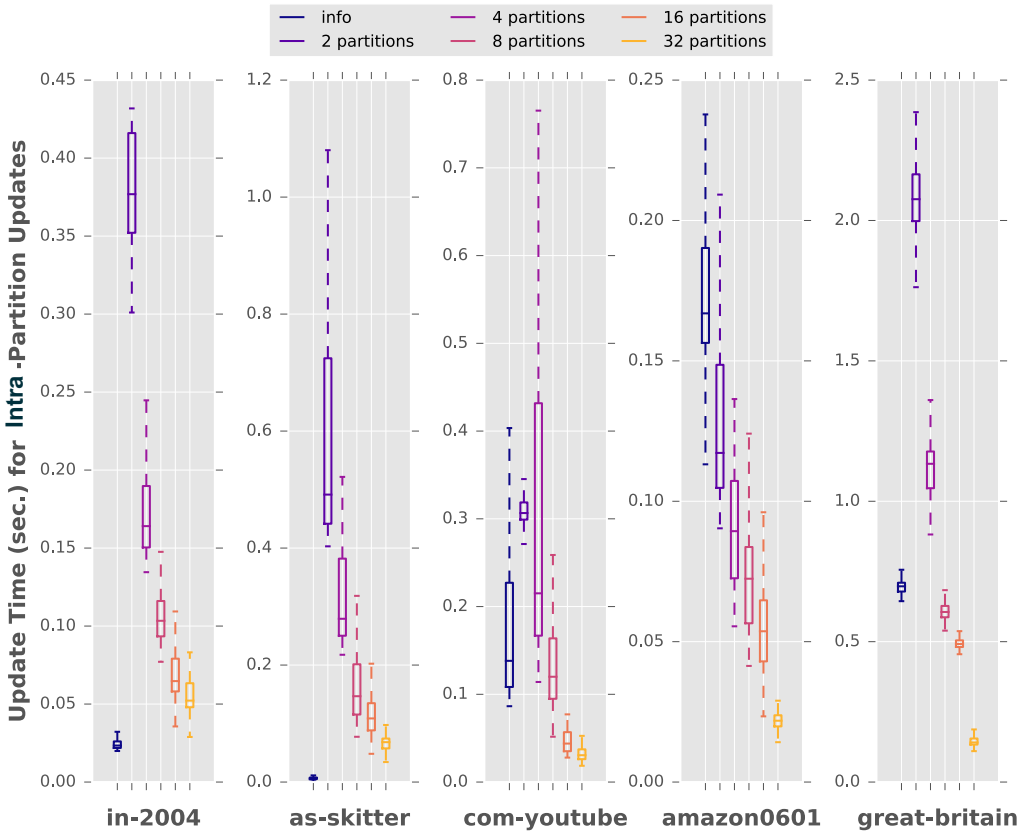
Static Graph Algorithm

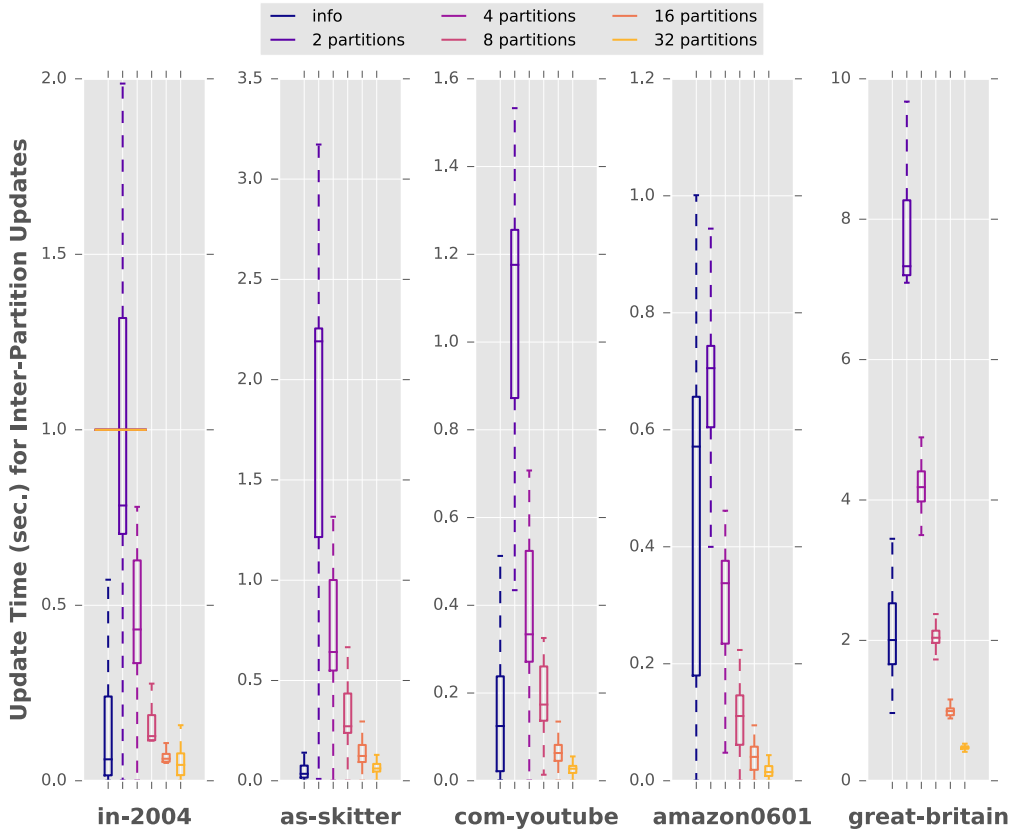




- 100 random queries
- Error goes up with number of partitions, Time goes up then down

Dynamic Graph Algorithm





- 100 randomly reinserted edges
- Update time goes down with number of partitions

Conclusions

- Graphs are important
  - » Shortest paths in graphs are important
- PPSP can come up in both static and dynamic graphs
  - » Need to minimize work done for queries and updates
- Our work
  - » Uses graph partitioning to solve approximate PPSP on static graphs
  - » Extends static-graph algorithm to dynamic graphs
  - » Every query and update operates on small number of nodes