V-Combiner: Speeding-up Iterative Graph Processing on a Shared-memory Platform with Vertex Merging

Azin Heidarshenas[†], Serif Yesil[†], Dimitrios Skarlatos[†], Sasa Misailovic[†], Adam Morrison*, Josep Torrellas[†]

University of Illinois Urbana-Champaign[†]

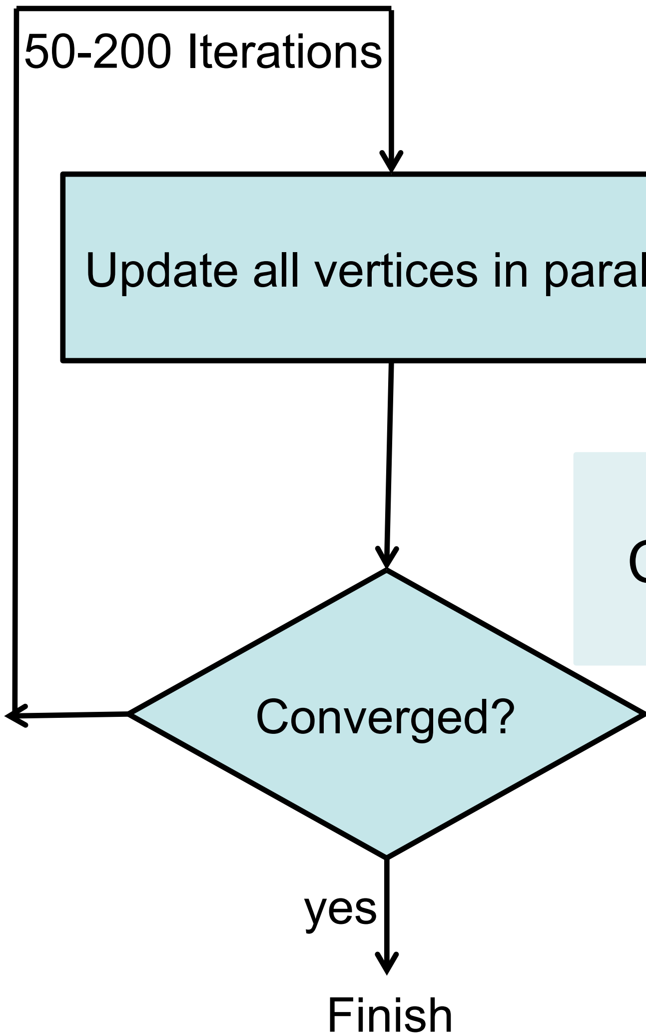
Tel-Aviv University*

International Conference on Supercomputing (ICS), June 2020

ILLINOIS

TEL AVIV UNIVERSITY אוניברסיטת תל אביב

Iterative graph processing




```
parallel for v in vertices
    for u in v.neighbors
        ...
    // update v
```

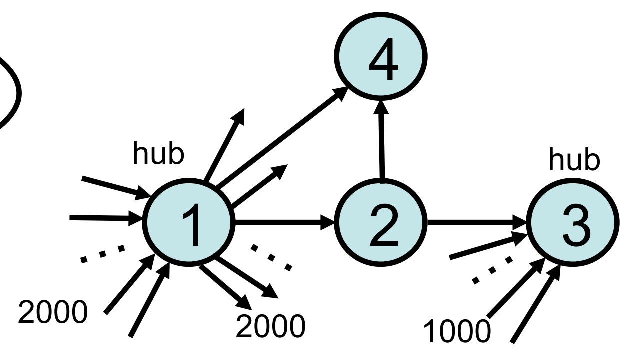
- Page Rank
- Community Detection
- HITS
- Belief Propagation

Computational complexity ∝ #Iterations

Graph processing can be approximate

Example: CEO of Company X wants to invest **only** on the most influential customers in their network

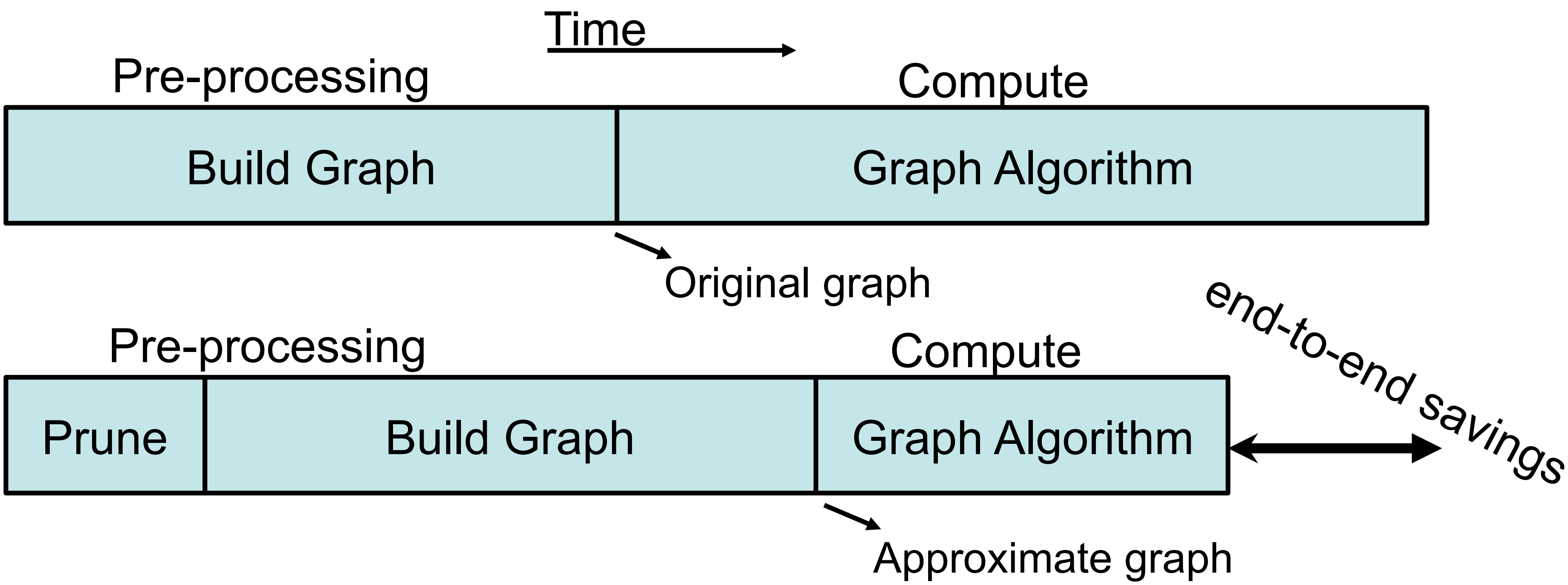
Vertex	Page Rank
1	0.0510103
3	0.0255164
4	7.3626e-05
2	5.16674e-05



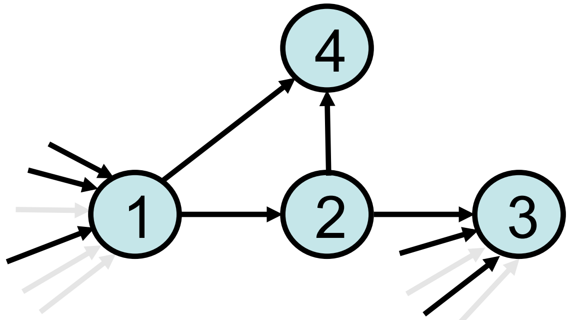
Computing Page Ranks of Vertices 2 and 4 is useless.

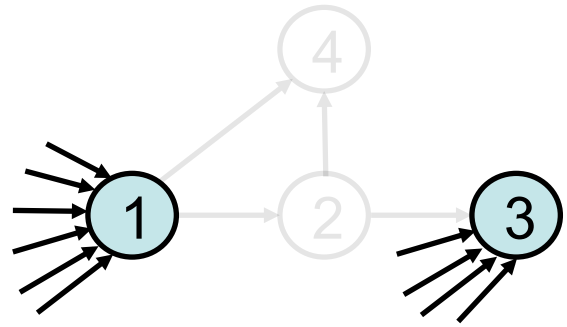
Pruning graphs can be effective

Removing useless computation ⟺ Removing certain vertices / edges (pruning)



Overview of Sparsification and K-core





Sparsification[1]
Prunes **only edges, probabilistically from dense regions**

K-core[2]
Prunes **vertices (along with their edges), until the remaining vertices have a degree of at least K**

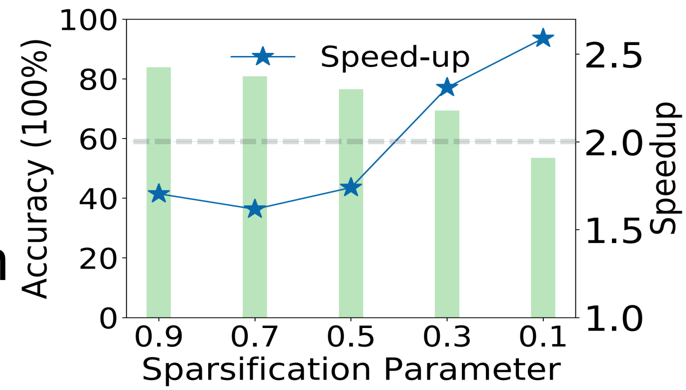
[1] Spectral sparsification of graphs: theory and algorithms. Commun. ACM 56, 2013
[2] K-core decomposition of large networks on a single PC, VLDB, 2015

Limitations of Sparsification and K-core

Desirable speedup > 2x

Accuracy is the ratio of vertices found in the top ranking.

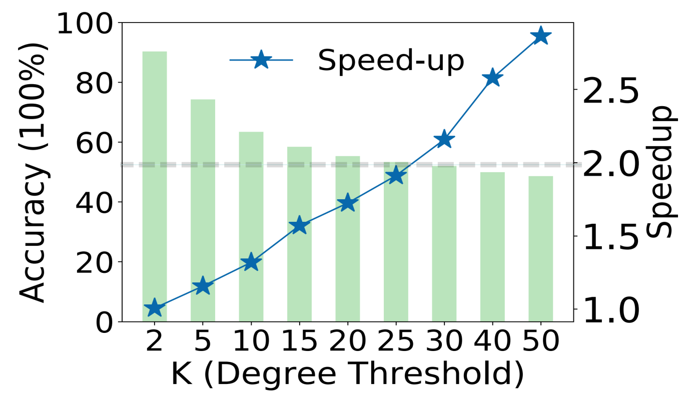
At the highest accuracy (~80%), Sparsification achieves 1.6x for Page Rank.



Degree of pruning

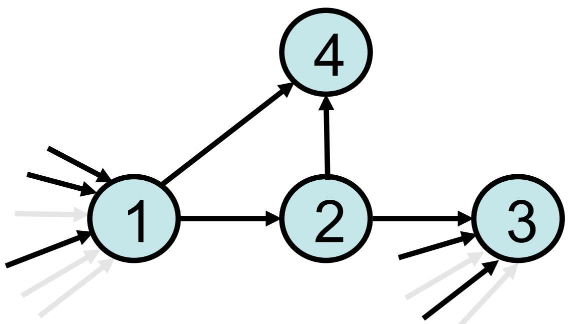
Accuracy is the ratio of vertices with correct communities.

High speedup is achieved only at low Accuracy (<60%) for Community Detection.

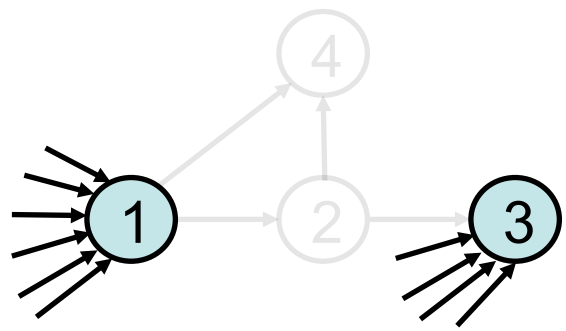


Degree of pruning

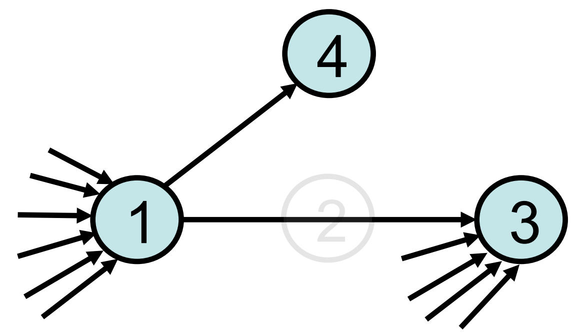
Addressing the Limitations



Sparsification[1]
Prunes **only edges, probabilistically from dense regions**



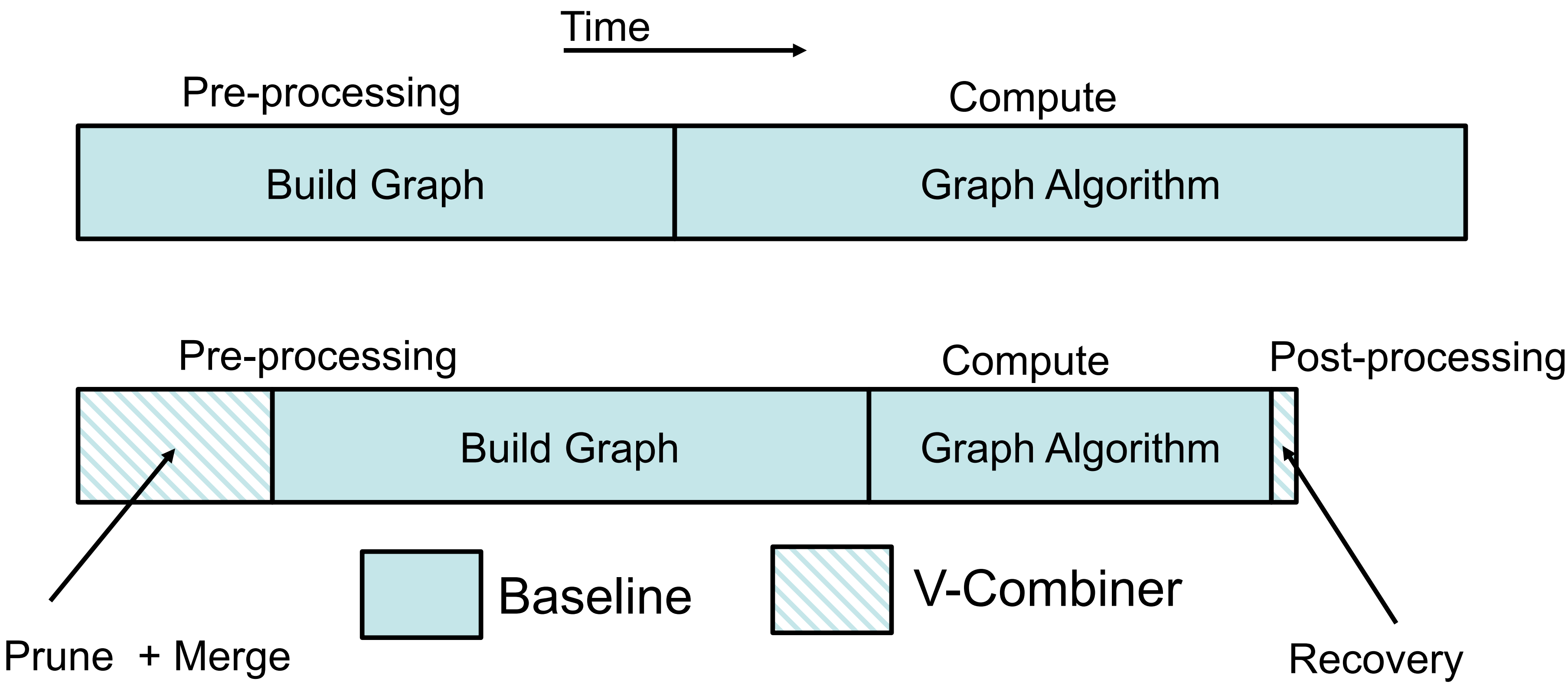
K-core[2]
Prunes **vertices (along with their edges), until the remaining vertices have a degree of at least K**



V-Combiner
Prunes **and merges** certain vertices into hubs (**in the direction of information flow**), so that hubs stay connected to the rest of the graph

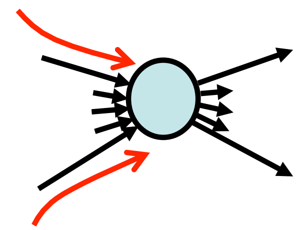
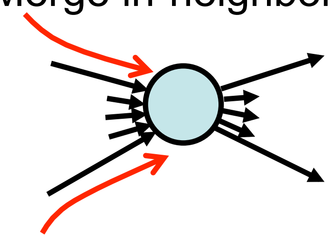
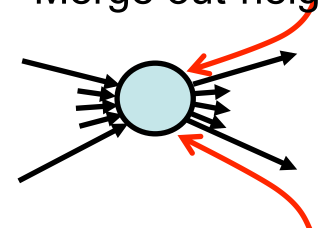
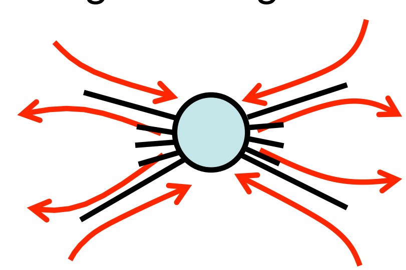
[1] Spectral sparsification of graphs: theory and algorithms. Commun. ACM 56, 2013
[2] K-core decomposition of large networks on a single PC, VLDB, 2015

Overview of V-Combiner



More merging vs. pre-processing time vs. performance savings

Different Vertex Merging Scenarios

Example App.	Edges	Information flow	
Page Rank, Comm. Detection	Directed	One-way	Merge in-neighbors 
HITS	Directed	Two-way	Merge in-neighbors  Merge out-neighbors 
Belief Propagation	Undirected	Two-way	Merge all neighbors 

Classification of Vertices in V-Combiner

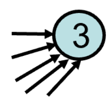
Supernode: Large in-degree (but not too large)

> Large in-degree for supernode→ More mergings per supernode

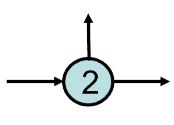
Subnode: Small in- and out-degree, at least one supernode in its out-neighborhood

> Small in- and out-degree for subnode → Less distortion after pruning

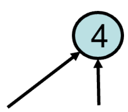
Regular: Neither a supernode nor a subnode



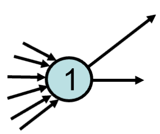
Supernode



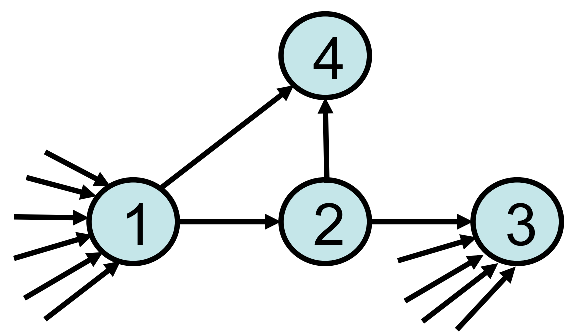
Subnode



Regular



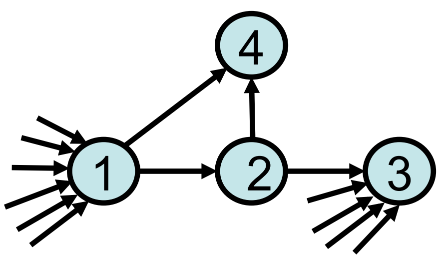
Regular



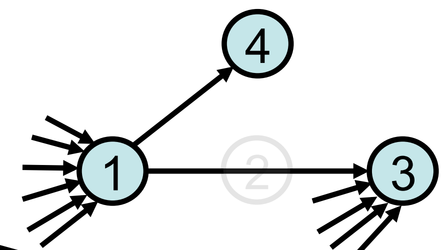
Prune + Merge in V-Combiner

for e in edges
    //MERGE
    if e.dst is a subnode and e.src is NOT a subnode **then**
        // Increment in-degree of the supernode by one

    //PRUNE
    if e.src is a subnode and e.dst is NOT a subnode **then**
        // Decrement in-degree of the e.dst by one



Vertex	Old in-degree	New in-degree
1	6	6
2	1	0
3	5	**5**
4	2	1



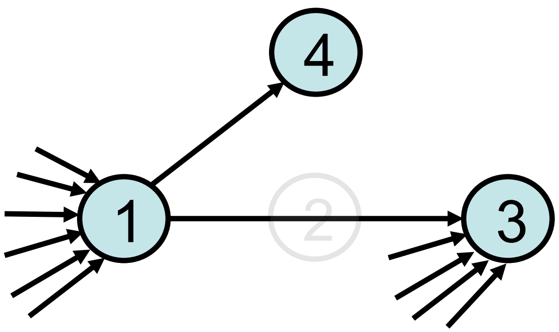
One increment and one decrement cancel out.

Recovery in V-Combiner

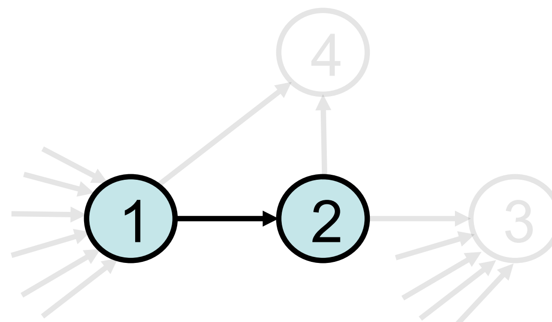
No subnodes in the approximate graph

Recover using the in-neighbors' values and the graph algorithm operator

- More efficient using Delta graph
- As if an extra iteration of the algorithm is run, but only for the subnodes



Approximate graph



Delta graph

For Page Rank: Pr[2] = 0.85 Pr[1] / 2 + 0.15

ILLINOIS

TEL AVIV UNIVERSITY אוניברסיטת תל אביב

Evaluation Setup

End-to-end speedup measured.

44 Intel Xeon cores, no hyper-threading and DVFS

4 graph applications:
- Page Rank (PR)
- Community Detection (CD)
- Hyperlink-Induced Topic Search (HITS)
- Belief Propagation (BP)

5 graph inputs
- Friendster social network (FS)
- Twitter social network (TW)
- Page-Level Domain graph (PLD)
- Arabic domain network (AR)
- Dbpedia network (DB)

Accuracy Metrics

Top-K Accuracy:
The ratio of vertices in the top ranking of the exact result that are also in the top ranking of the approximate result

- Page Rank
- HITS
- Belief Propagation

Classification Accuracy:
The ratio of vertices that have been correctly assigned to their communities

- Community Detection

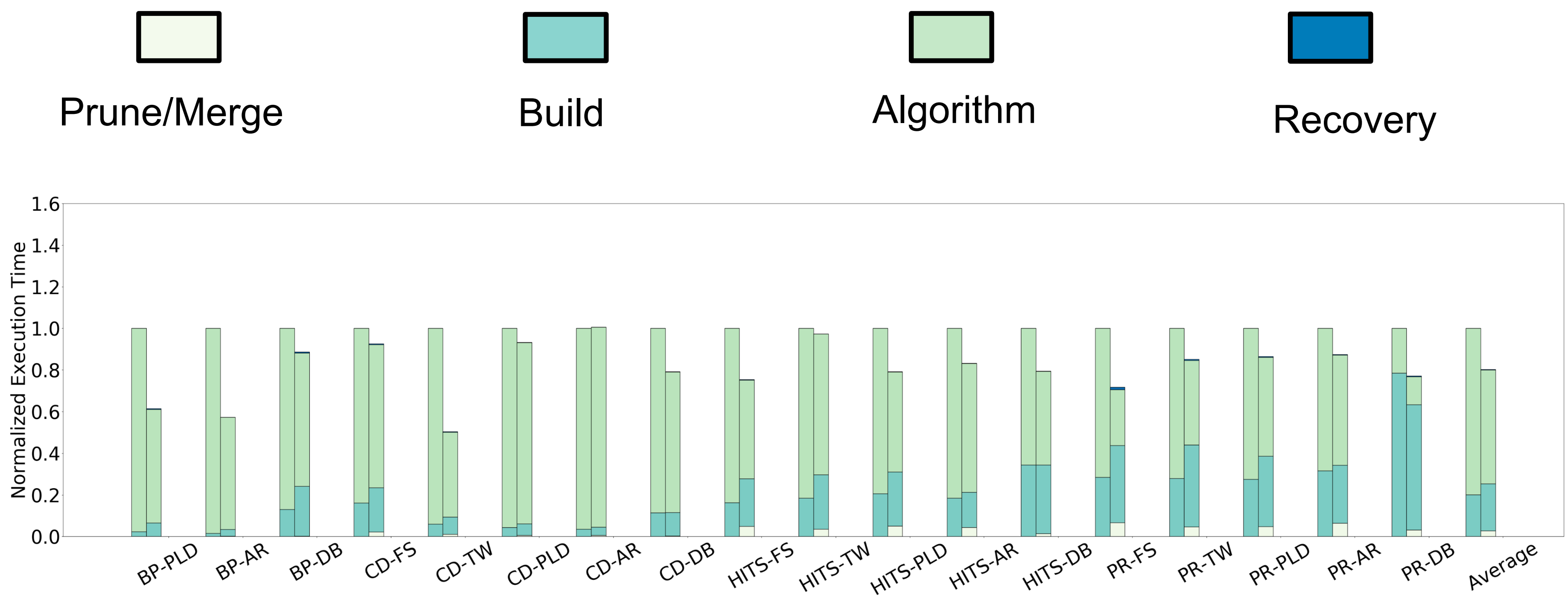
Accuracy threshold of 90%.

End-to-End Performance

ILLINOIS

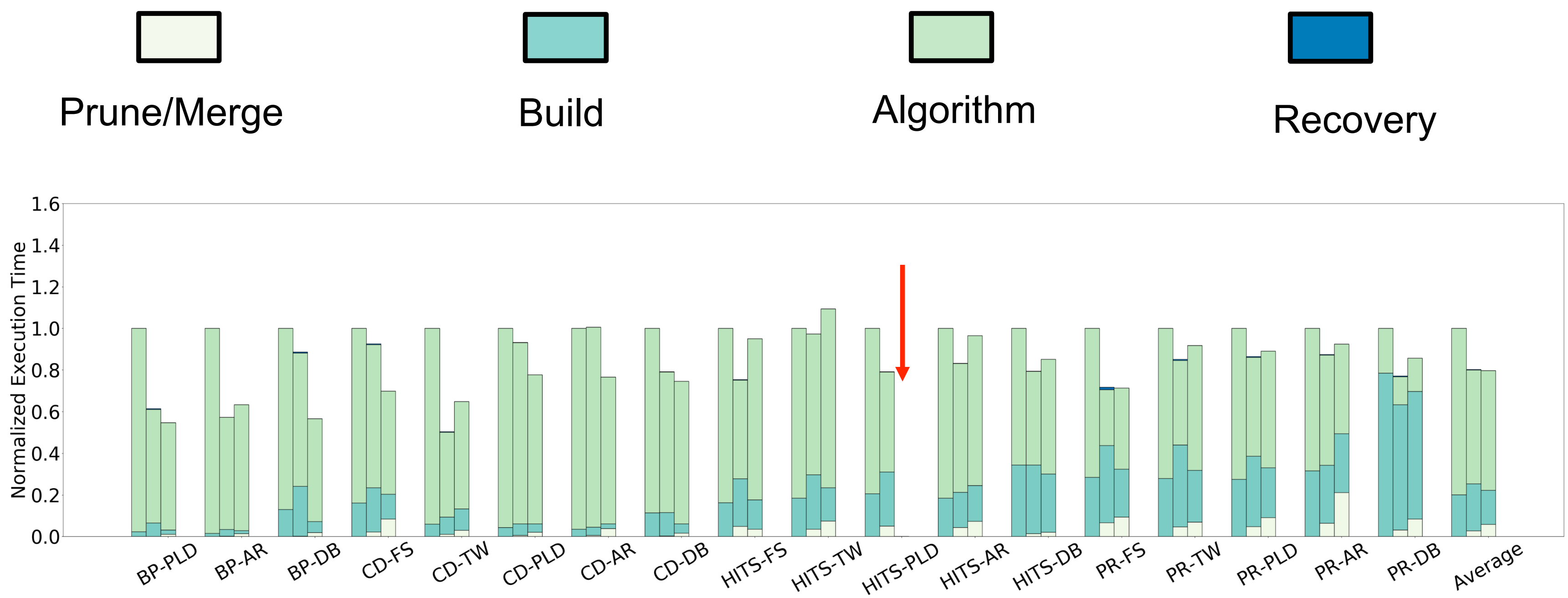
TEL AVIV UNIVERSITY אוניברסיטת תל אביב

End-to-End Performance: V-Combiner



1.25 end-to-end speedup at mean accuracy of 91.8%

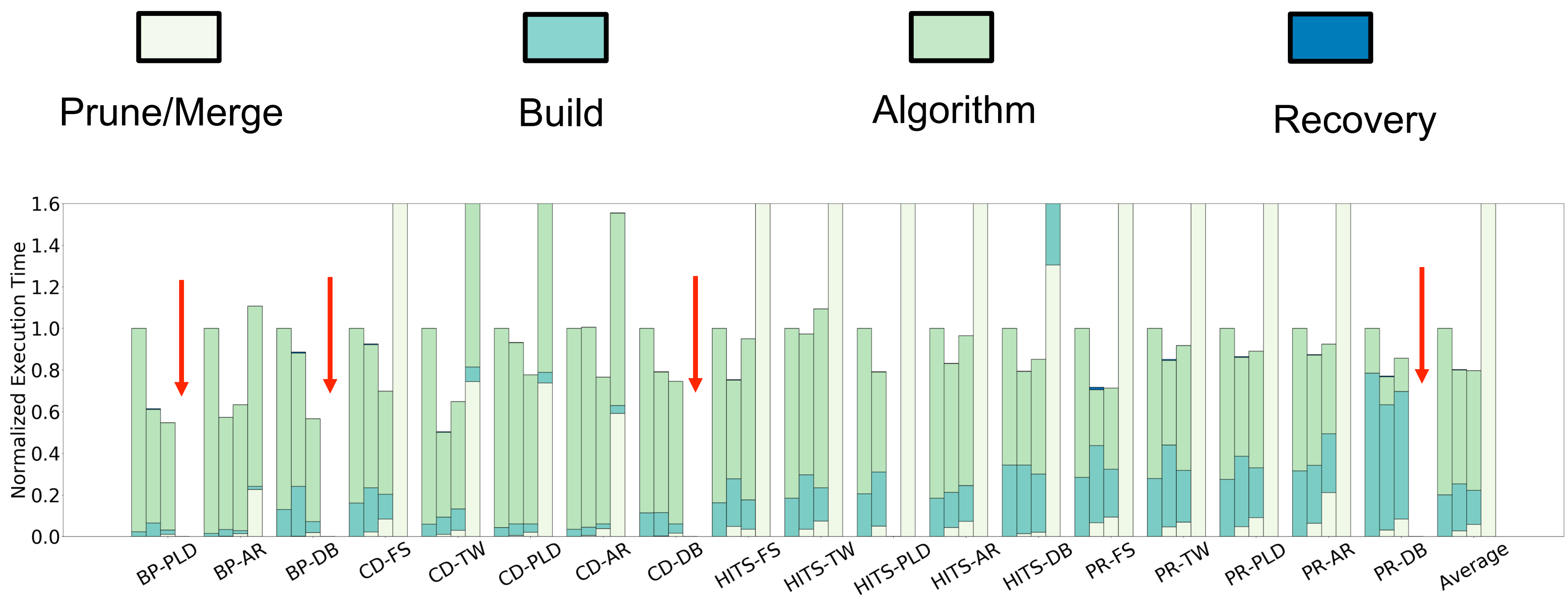
End-to-End Performance: Sparsification



Sparsification fails to meet accuracy threshold in 1 benchmark

ILLINOIS TEL AVIV UNIVERSITY אוניברסיטת תל אביב

End-to-End Performance: K-core



K-core fails to meet accuracy threshold in 4 benchmarks

More in the Paper

- Details of other scenarios of the merging
- Choosing the merging parameters
- Algorithm performance and accuracy analysis
- Analysis of connectivity
- Analysis of the average length of the paths
- Analysis of pruning/merging parameters
- …

ILLINOIS TEL AVIV UNIVERSITY אוניברסיטת תל אביב

Take-away

- Iterative graph processing is computationally expensive and can be approximate.
- V-Combiner is a pruning + merging + recovery technique
- It has the following advantages over the state-of-the-art pruning techniques:
  - Preserving average length of the paths
  - Maintaining connectivity
  - Improving load balancing
  - Modest pre-processing overhead

V-Combiner: Speeding-up Iterative Graph Processing on a Shared-memory Platform with Vertex Merging

Azin Heidarshenas†, Serif Yesil†, Dimitrios Skarlatos†, Sasa Misailovic†, Adam Morrison*, Josep Torrellas†

University of Illinois Urbana-Champaign†

Tel-Aviv University*