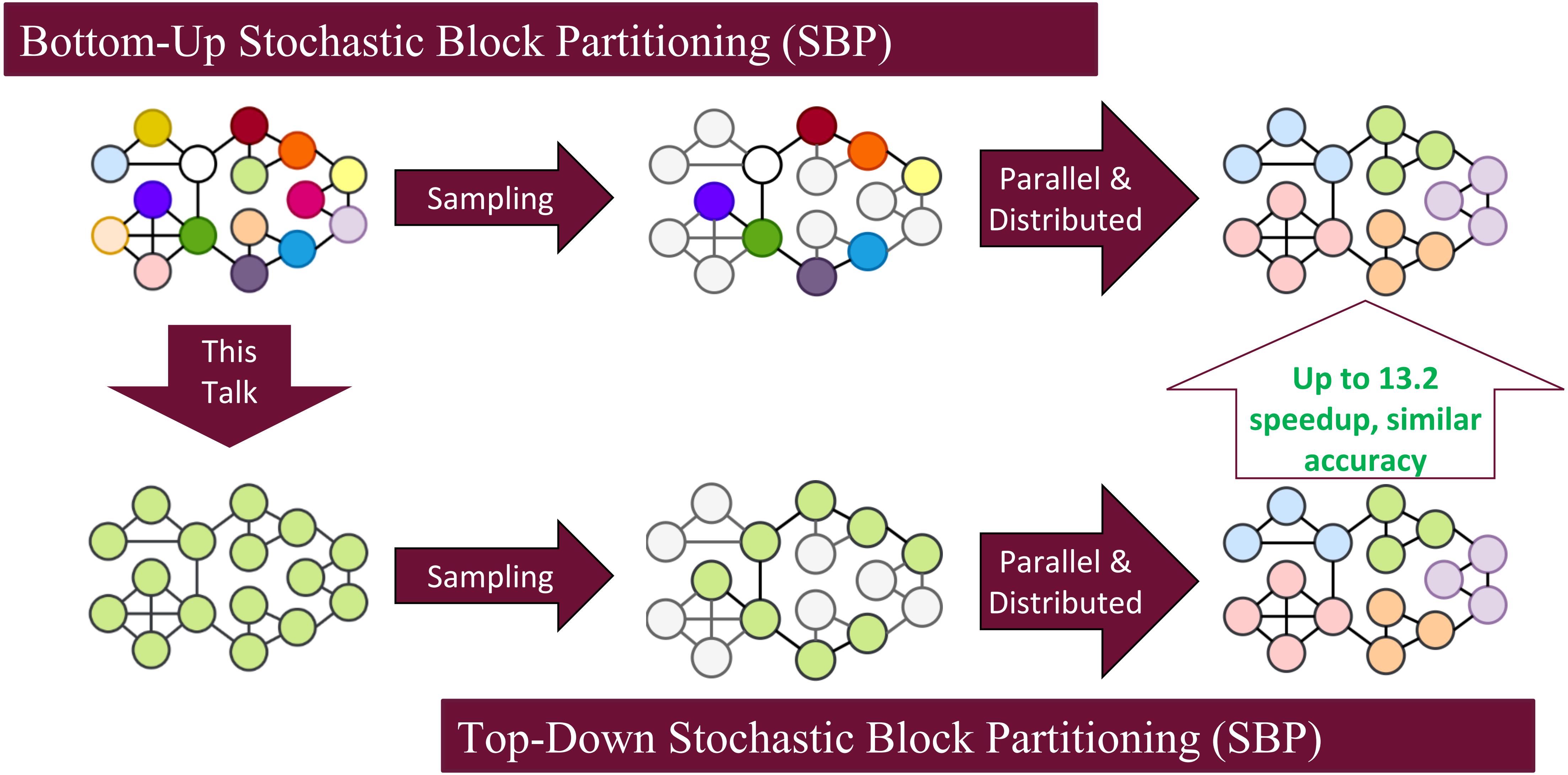
Top-Down SBP: Turning Graph Clustering Upside Down

VIRGINIA TECH → AMD FRANK WANYE
MIT LINCOLN LAB VITALIY GLEYZER
MIT LINCOLN LAB EDWARD KAO
VIRGINIA TECH WU-CHUN FENG

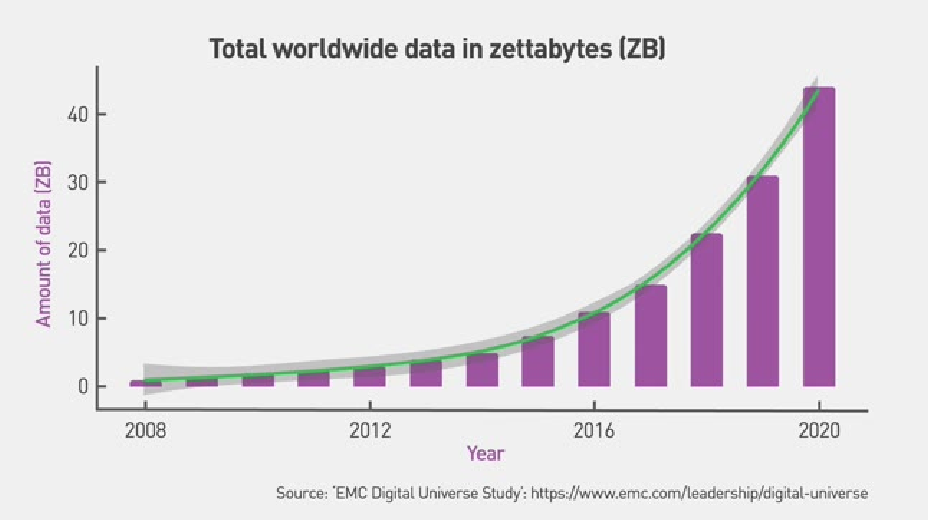
JULY 22, 2025
34TH ACM INTERNATIONAL SYMPOSIUM ON HIGH-PERFORMANCE PARALLEL AND DISTRIBUTED COMPUTING (HPDC)

synergy.cs.vt.edu nsf-shrec.org www.stonybrook.edu

Forecast


Bottom-Up Stochastic Block Partitioning (SBP)
Sampling
Parallel & Distributed
This Talk
Up to 13.2 speedup, similar accuracy
Sampling
Parallel & Distributed
Top-Down Stochastic Block Partitioning (SBP)

Introduction



- Worldwide data collected doubling every 2 years
- Much of this data is relational → graph representation
- Groups of strongly connected vertices correlate to functional groups within data
- Graph clustering: process of finding such groups

Motivation

Applications across many domains

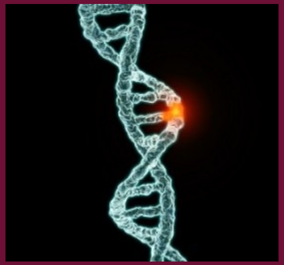
Networking

Intrusion Detection

Finance

Fraud Detection

Bioinformatics



Drug Discovery

Social Media

Recommendation Systems

- *Accurate* graph clustering is difficult
- Difficulty highlighted by collaborative efforts, e.g., Graph Challenge [1] , sponsored by IEEE/Amazon/MIT

[1] Edward Kao, Vijay Gadepally, Michael Hurley, Michael Jones, Jeremy Kepner, Sanjeev Mohindra, Paul Monticciolo, Albert Reuther, Siddharth Samsi, William Song, Diane Staheli, and Steven Smith. Streaming graph challenge: Stochastic block partition. In Proceedings of the 2017 IEEE High Performance Extreme Computing Conference (HPEC), 2017.

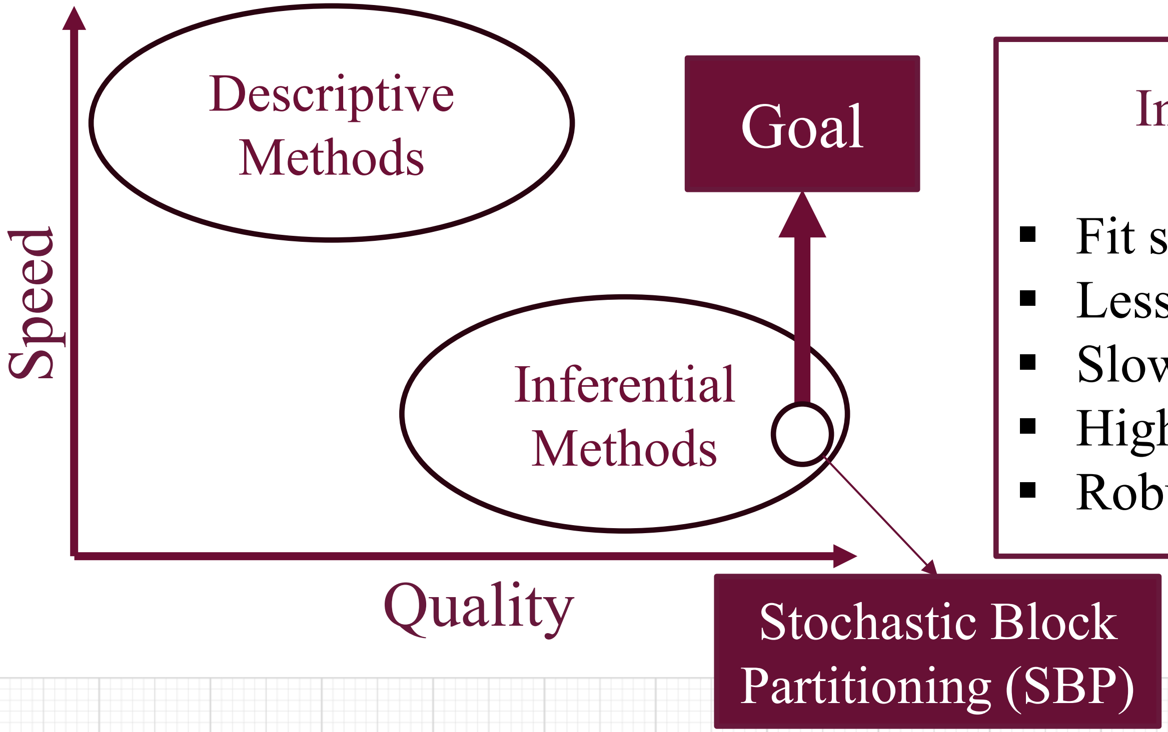
Fast vs. Accurate Graph Clustering

- Optimal graph clustering is NP-hard → solved via heuristics
- Two classes of heuristics: descriptive and inferential[1]

Enable <u>accurate</u> graph clustering in large graphs by accelerating inferential graph clustering methods

Descriptive Methods

- Encompass many commonly used heuristics
- Fast
- Lower quality solutions
- Prone to overfitting



Inferential Methods

- Fit statistical models to data
- Less commonly used
- Slow
- Higher quality solutions
- Robust against overfitting

[1] Tiago P. Peixoto. Descriptive vs. inferential community detection in networks: pitfalls, myths, and half-truths. Cambridge University Press, 2023.

OVERVIEW

1. Introduction to Graph Clustering
2. **Background & Contributions**
3. Approach
   a) Top-Down SBP
   b) Accelerated Top-Down SBP
4. Results on Real-World Graphs
5. Summary and Future Work

SBP Algorithm Overview

Statistical Model	Quality Function	Optimization Algorithm

Stochastic Blockmodel (SBM)[1]

- Generative model
- Models the graph in relation to connectivity between clusters (blocks)

Cluster

Cluster	0	1	2	3
0	12	2	3	1
1	1	20	2	4
2	3	1	13	5
3	2	$\boxed{4}$	1	17

Number of edges from cluster 3 to cluster 1

Description Length (H)[2]

- Quality function for inference over **SBM**s
- Number of bits needed to encode **SBM**
- $H = f(graph\ size, blockmodel\ parameters)$
- Lower ***H*** =
  - Better compression
  - Lower entropy → more stability
  - Better quality of clusters

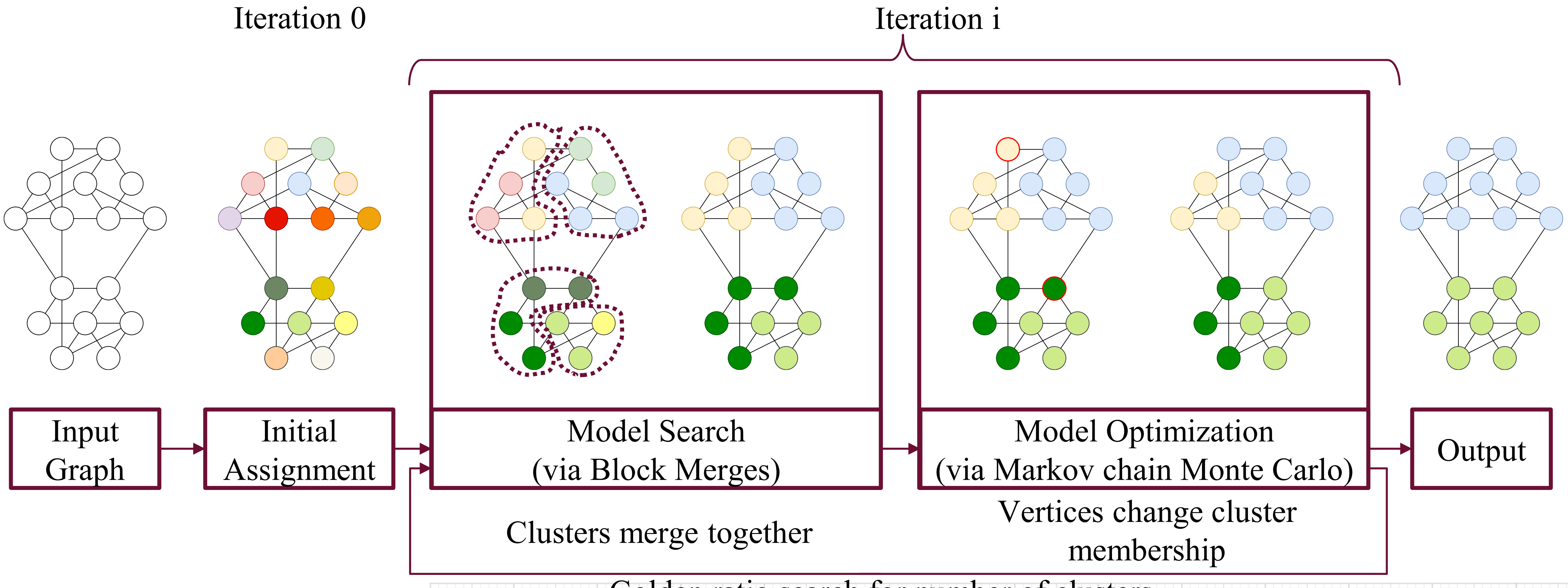
[1] Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. In Physical Review E, vol 83, no 1, 2011.
[2] Tiago P. Peixoto. Parsimonious Module Inference in Large Networks. In Physical Review Letters, vol 110, no 14, 2013.

SBP Algorithm Overview

Statistical Model	Quality Function	Optimization Algorithm

Iterative, agglomerative, Markov chain Monte-Carlo (MCMC) based optimization of description length ***H*** [1,2]

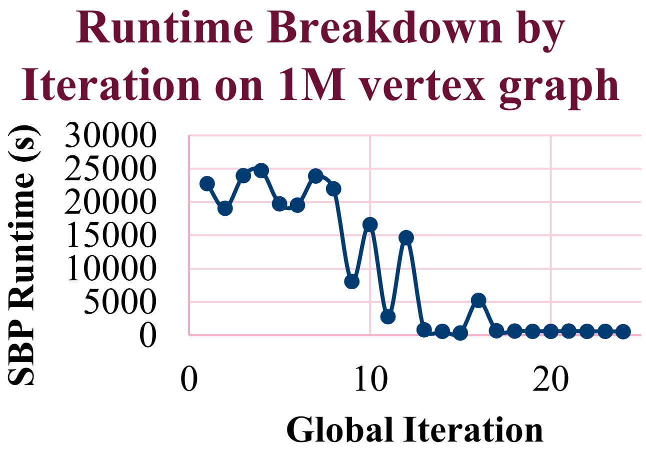
$O(E \log^2 E)$
E: number of graph edges



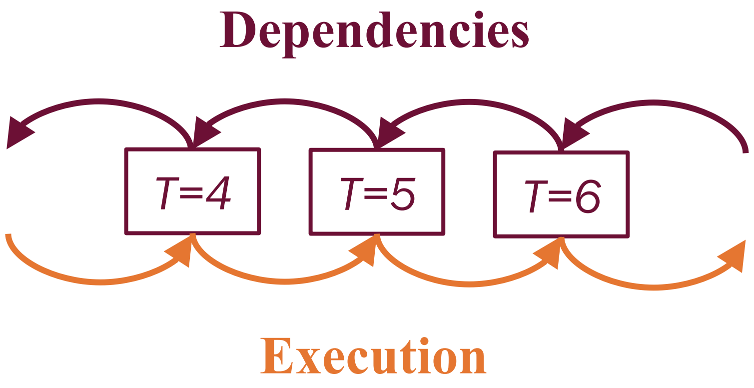
[1] Tiago P. Peixoto. Parsimonious Module Inference in Large Networks. In Physical Review Letters, vol 110, no 14, 2013.
[2] Tiago P. Peixoto. Efficient Monte Carlo and greedy heuristic for the inference of stochastic block models. In Physical Review E, vol 89, no 1, 2014.

Contributions

Computational Profile



MCMC Computation



Challenges

- Random memory access patterns
- Row ***and*** column-wise indexing
- Front-heavy computation

- Inherently sequential optimization technique[1]
- State at time T depends on ***all*** previous timesteps

Contributions

- Top-Down computation approach
  - 7.7X speedup over Bottom-Up
  - 4X lower memory usage

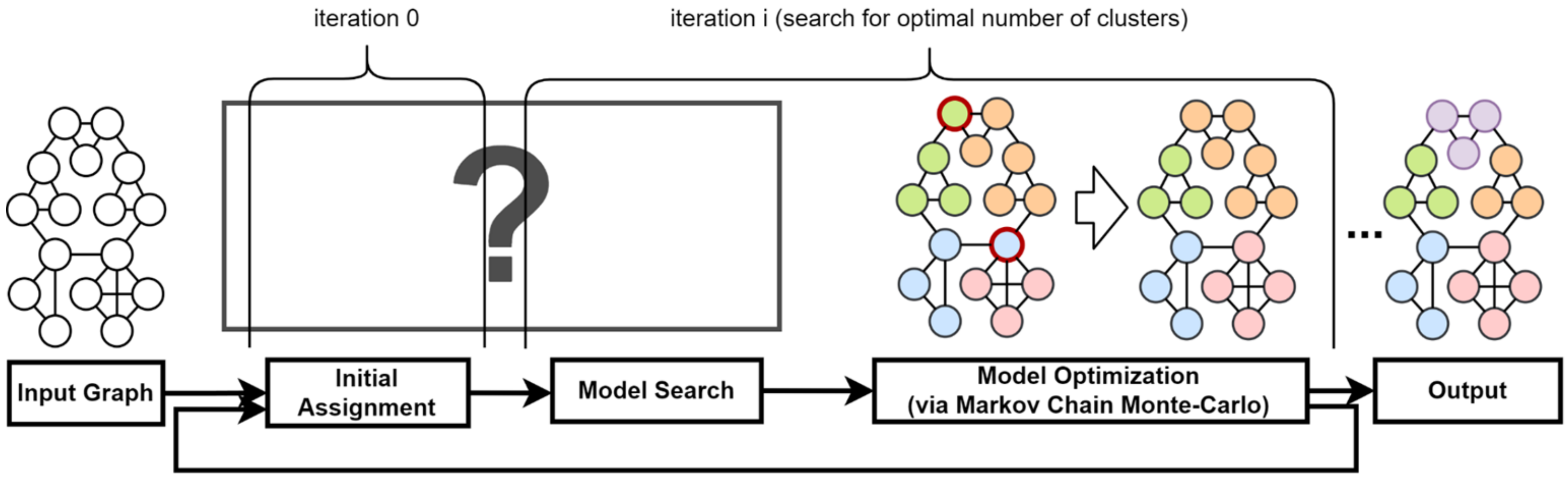
- Integration of Top-Down approach with prior SBP parallelization efforts
  - 13.2X speedup over accelerated Bottom-Up

[1] Darren J Wilkinson. Parallel Bayesian Computation. In Erricos John Kontoghiorghes, editor, Handbook of Parallel Computing and Statistics, 2005.

OVERVIEW

1. Introduction to Graph Clustering
2. Background & Contributions
3. Approach
    a) Top-Down SBP
    b) Accelerated Top-Down SBP
4. Results on Real-World Graphs
5. Summary and Future Work

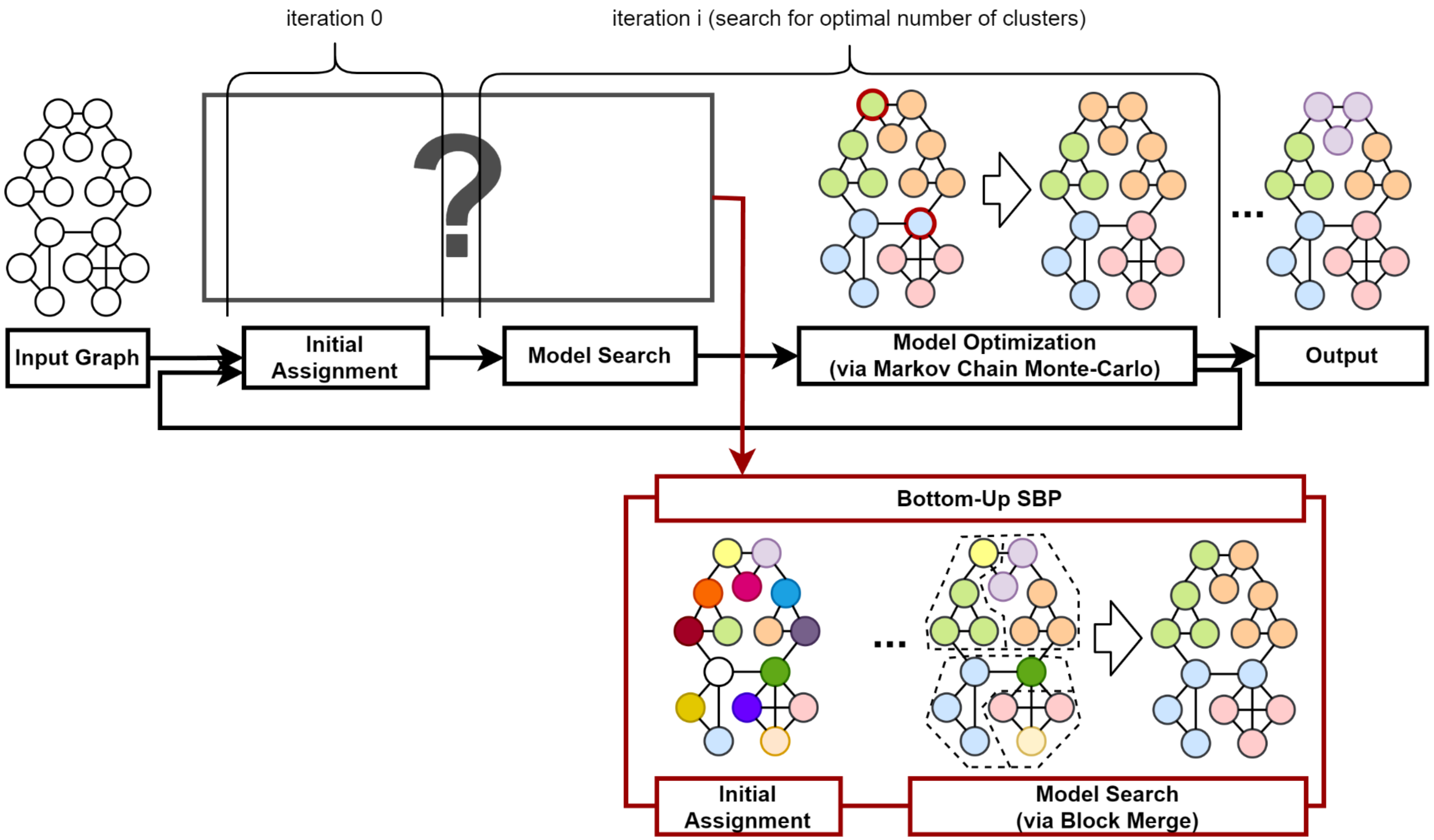
Top-Down SBP: Overview



Approach

- Replicate overall algorithm structure of SBP
- Block merges replaced with block splits
- Splits accepted/rejected based on change in SBM description length
- Same algorithmic complexity: $O(E \log^2 E)$

Top-Down SBP: Overview

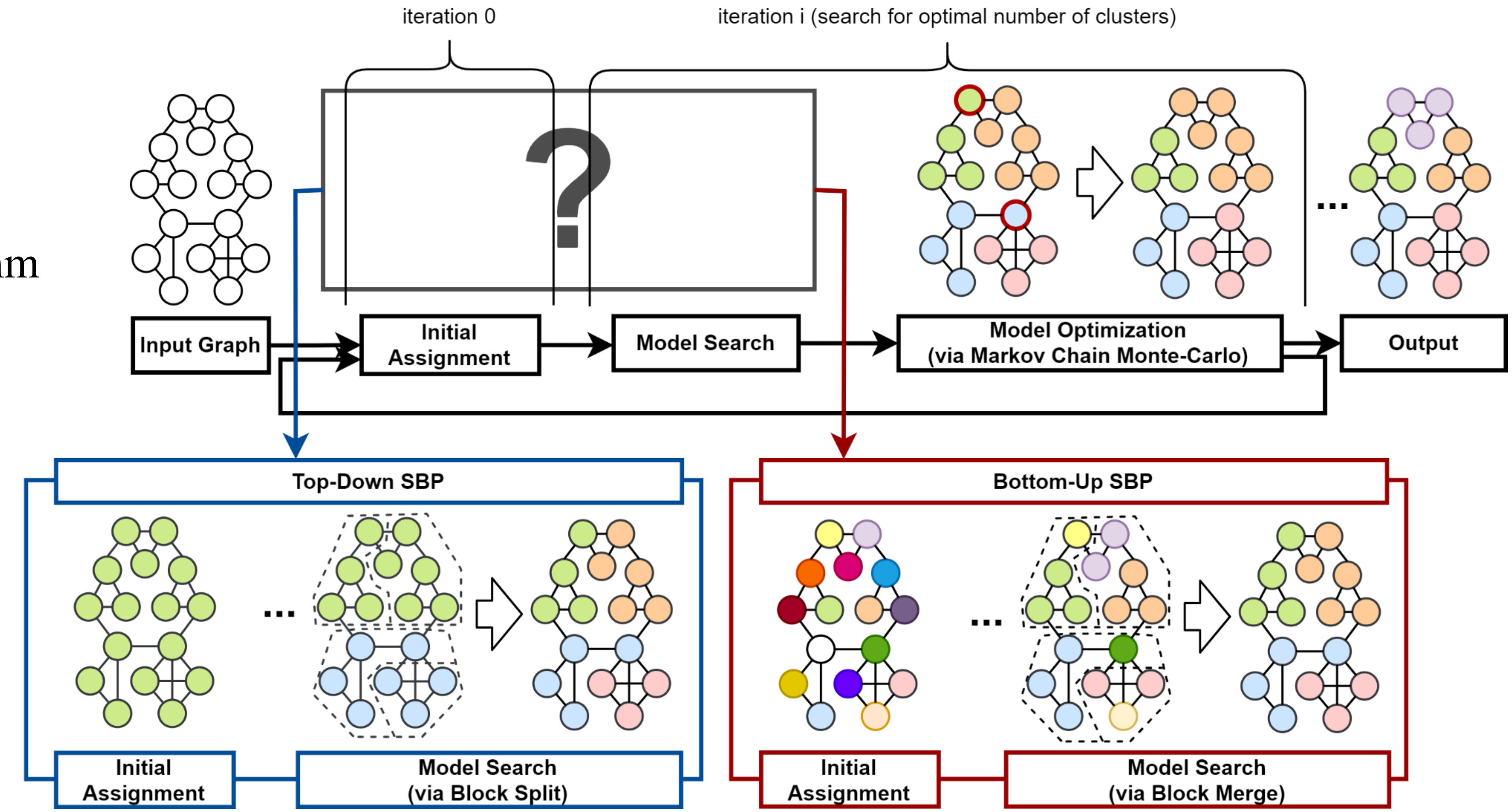


Approach

- Replicate overall algorithm structure of SBP
- Block merges replaced with block splits
- Splits accepted/rejected based on change in SBM description length
- Same algorithmic complexity: $O(E \log^2 E)$

Top-Down SBP: Overview

Approach
- Replicate overall algorithm structure of SBP
- Block merges replaced with block splits
- Splits accepted/rejected based on change in SBM description length
- Same algorithmic complexity: $O(E \log^2 E)$



Block-Splitting Heuristic

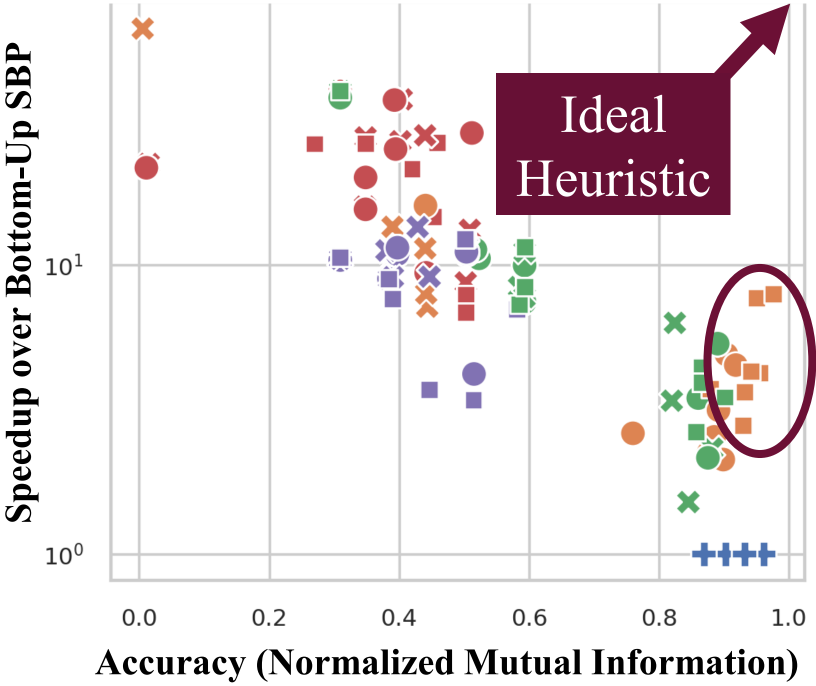
Splitting Heuristics

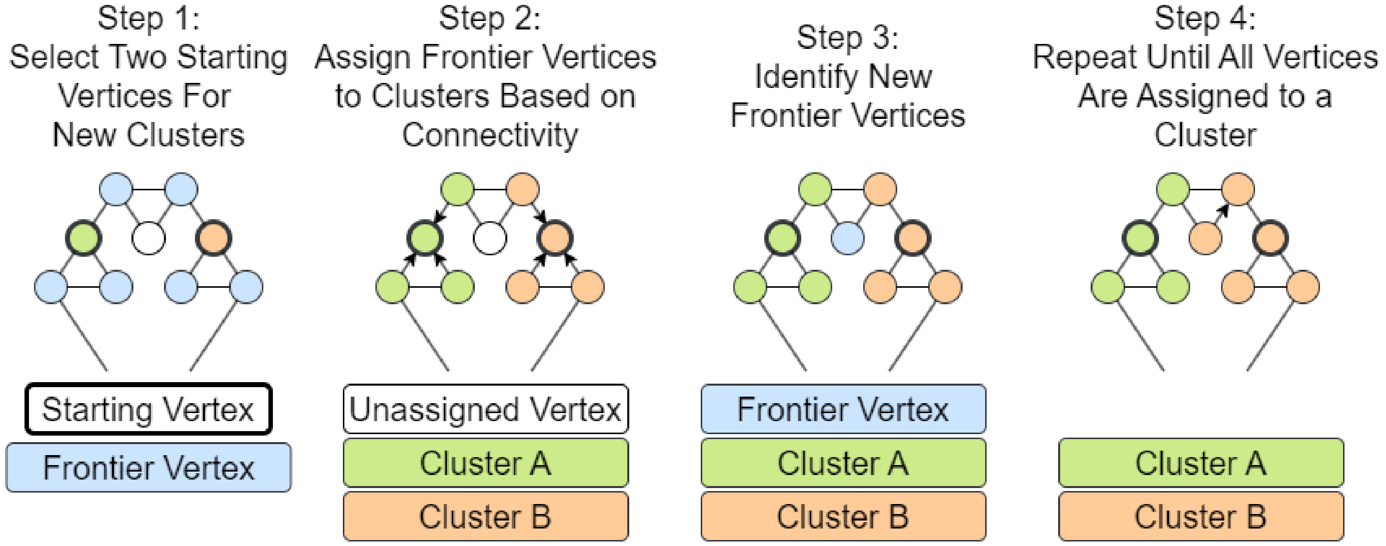
- Uniform random
- Two competing snowball samples
- One snowball sample
- Snowball sampling based on connectivity

Split Initializations

- ■ Uniform random
- ● Degree-weighted random initialization
- × Selecting the two highest-degree vertices

Exploring Splitting Heuristics





Best Heuristic: Connectivity snowball + random initialization

- Idea: clusters should be split based on vertex locality
- Two vertices are randomly chosen to initialize the new clusters

Single-Threaded Results

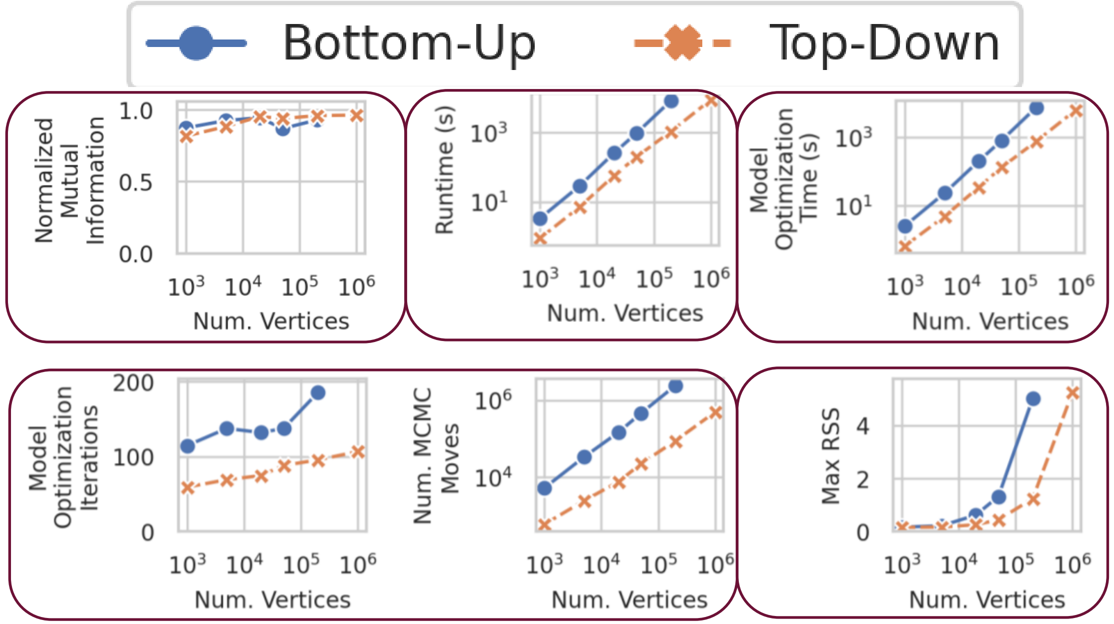
Graphs

- Official IEEE/Amazon/MIT Graph Challenge synthetic datasets

Num. Vertices	Num. Edges	Num. Clusters
1,000	8,032	11
5,000	51,157	19
20,000	473,329	32
50,000	1,187,682	44
200,000	4,754,406	71
1,000,000	23,772,977	125

Experiments

- Running single-threaded Bottom-Up SBP and Top-Down SBP
- Ookami cluster
  - 32 GB memory, Fujitsu A64FX CPUs



Little-to-no difference in accuracy

Up to 7.7X speedup over Bottom-Up SBP

Fewer MCMC iterations and fewer MCMC vertex moves → faster model optimization time

Up to 4.1X lower memory usage

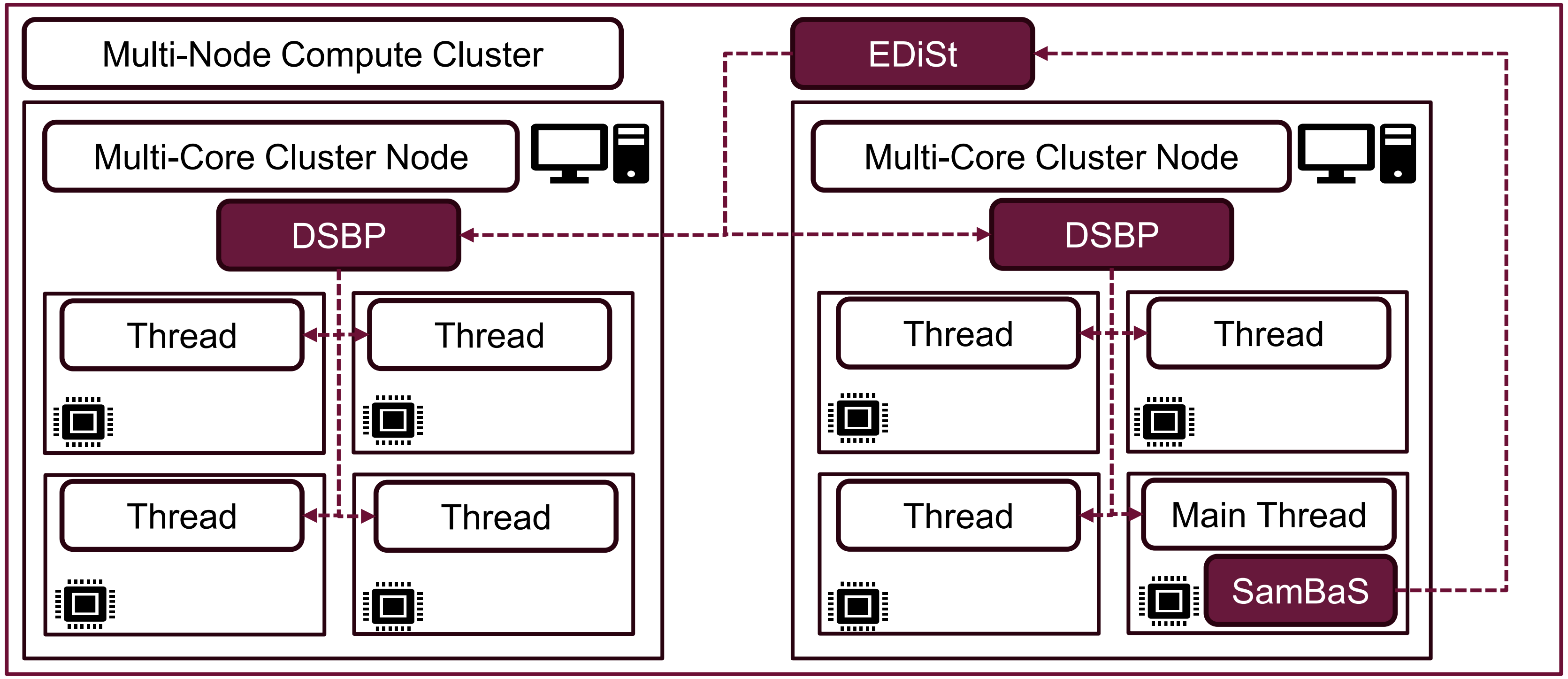
OVERVIEW

1. Introduction to Graph Clustering
2. Background & Contributions
3. Approach
   a) Top-Down SBP
   b) Accelerated Top-Down SBP
4. Results on Real-World Graphs
5. Summary and Future Work

Accelerated Top-Down SBP: Overview

- Bottom-Up SBP has been successfully accelerated using a combination of the following[1][2]:

Shared-Memory Parallelization (**DSBP**)[2] + Multi-Node Parallelization (**EDiSt**)[3] + Sampling (**SamBaS**)[4]



- We adapt these approaches from Bottom-Up SBP to Top-Down SBP

[1] Frank Wanye, Vitaliy Gleyzer, Edward Kao, Wu-chun Feng. An Integrated Approach for Accelerating Stochastic Block Partitioning. In Proceedings of the 27th IEEE High Performance Extreme Computing Conference (HPEC), 2023.
[2] Ahsen Uppal, Thomas Rolinger, Howie Huang. Decontentioned Stochastic Block Partition. In Proceedings of the 27th IEEE High Performance Extreme Computing Conference (HPEC), 2023.
[3] Frank Wanye, Vitaliy Gleyzer, Edward Kao, Wu-chun Feng. Exact Distributed Stochastic Block Partitioning. In Proceedings of the 25th IEEE International Conference on Cluster Computing (CLUSTER), 2023.
[4] Frank Wanye, Vitaliy Gleyzer, Edward Kao, Wu-chun Feng. SamBaS: Sampling-Based Stochastic Block Partitioning. In IEEE Transactions on Network Science and Engineering

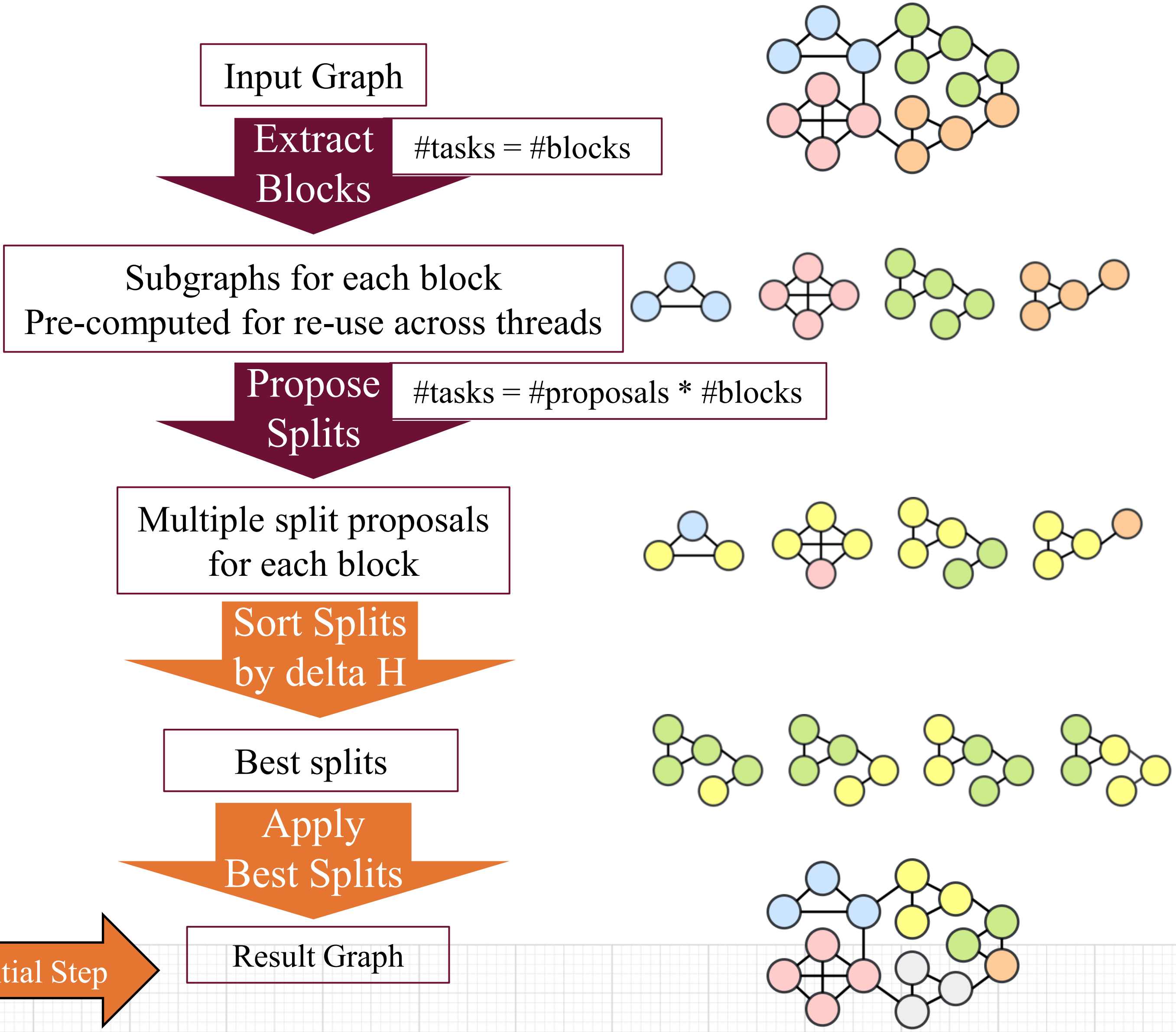
Shared-Memory Parallelism

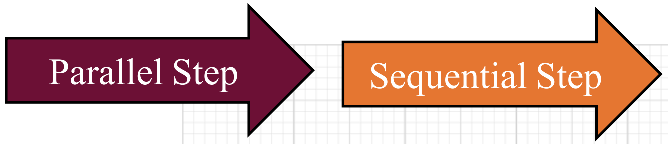
Model Optimization Phase

- Batched asynchronous Gibbs method
    - Embarrassingly parallel within each batch

Model Search Phase

- Requires pre-computing subgraphs to reduce memory usage





Multi-Node Parallelism

EDiSt: <u>E</u>xact <u>Di</u>stributed <u>St</u>ochastic Block Partitioning

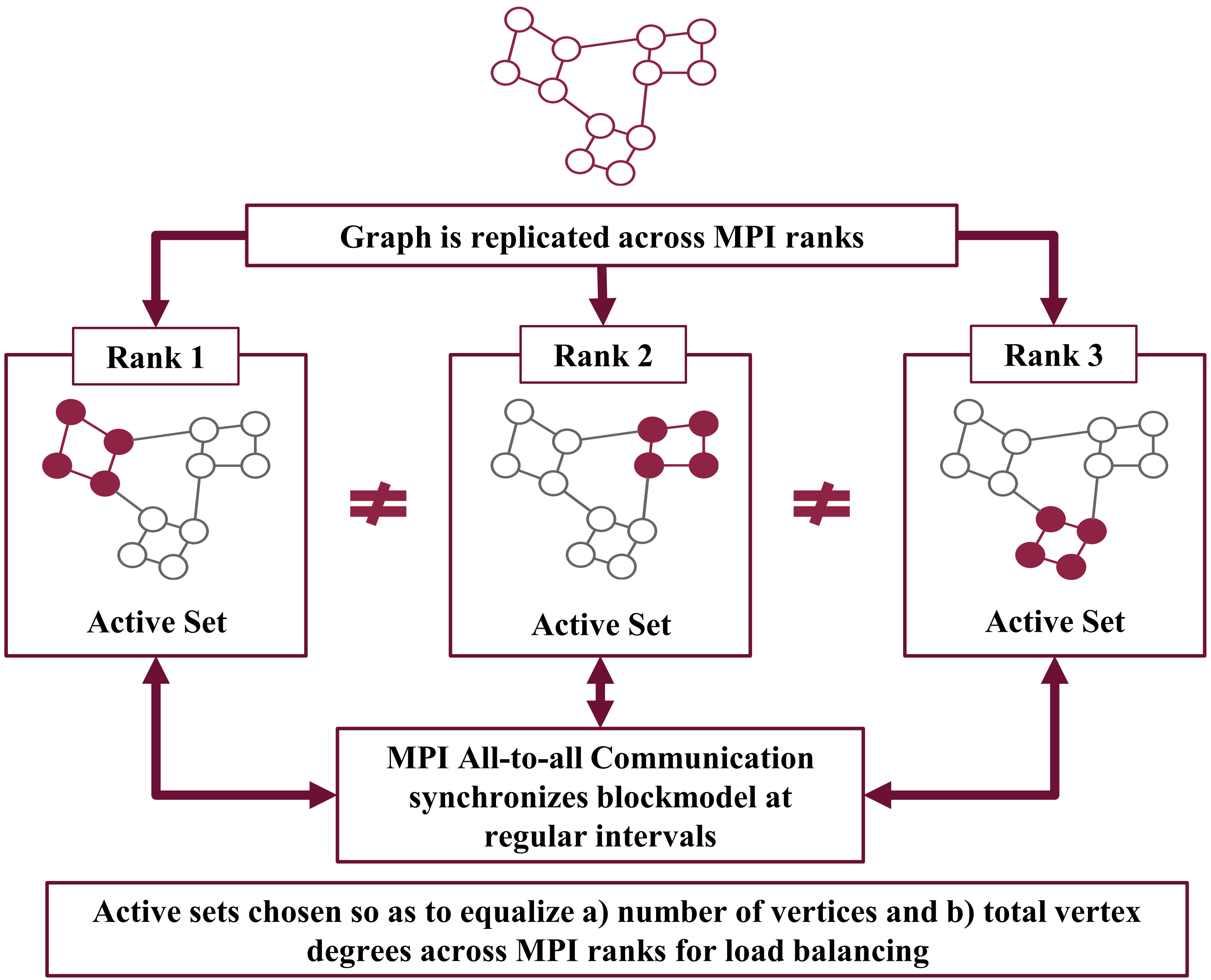
- Data replication → minimize broken dependencies → helps retain accuracy
- Difference between Bottom-Up and Top-Down SBP: amount of communication in model search phase

Bottom-Up SBP

- Block-level operation: O(#blocks) data transferred

Top-Down SBP

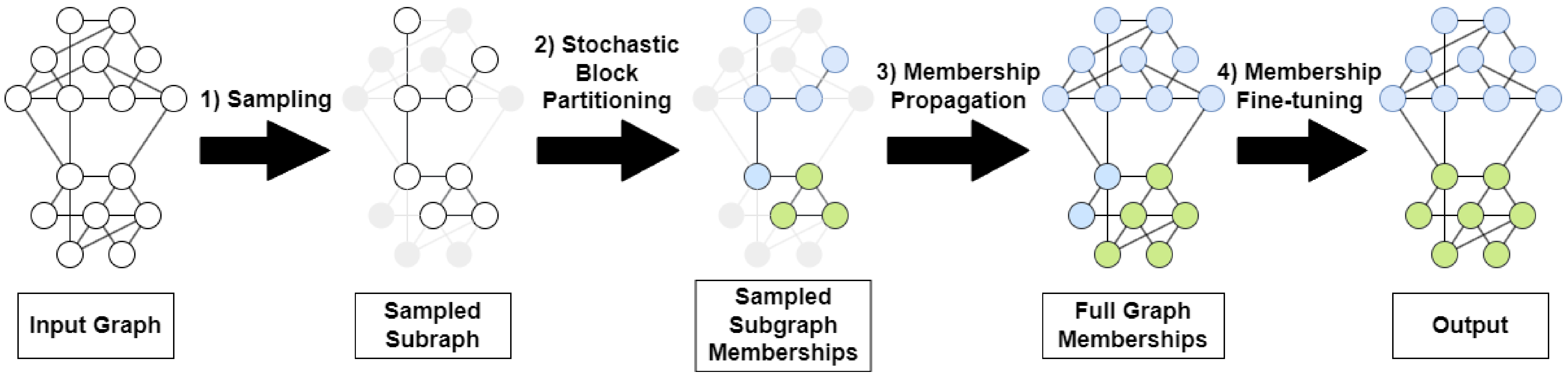
- Vertex-level operation: O(#vertices) data transferred



MPI = Message Passing Algorithm)

Sampling

SamBaS: 4 step sampling approach to accelerating SBP
Integration: Run Top-Down SBP in step 2



✓ Reduces memory and compute cost of initial iterations

✓ Flexible → SBP & fine-tuning can be replaced with accelerated variants

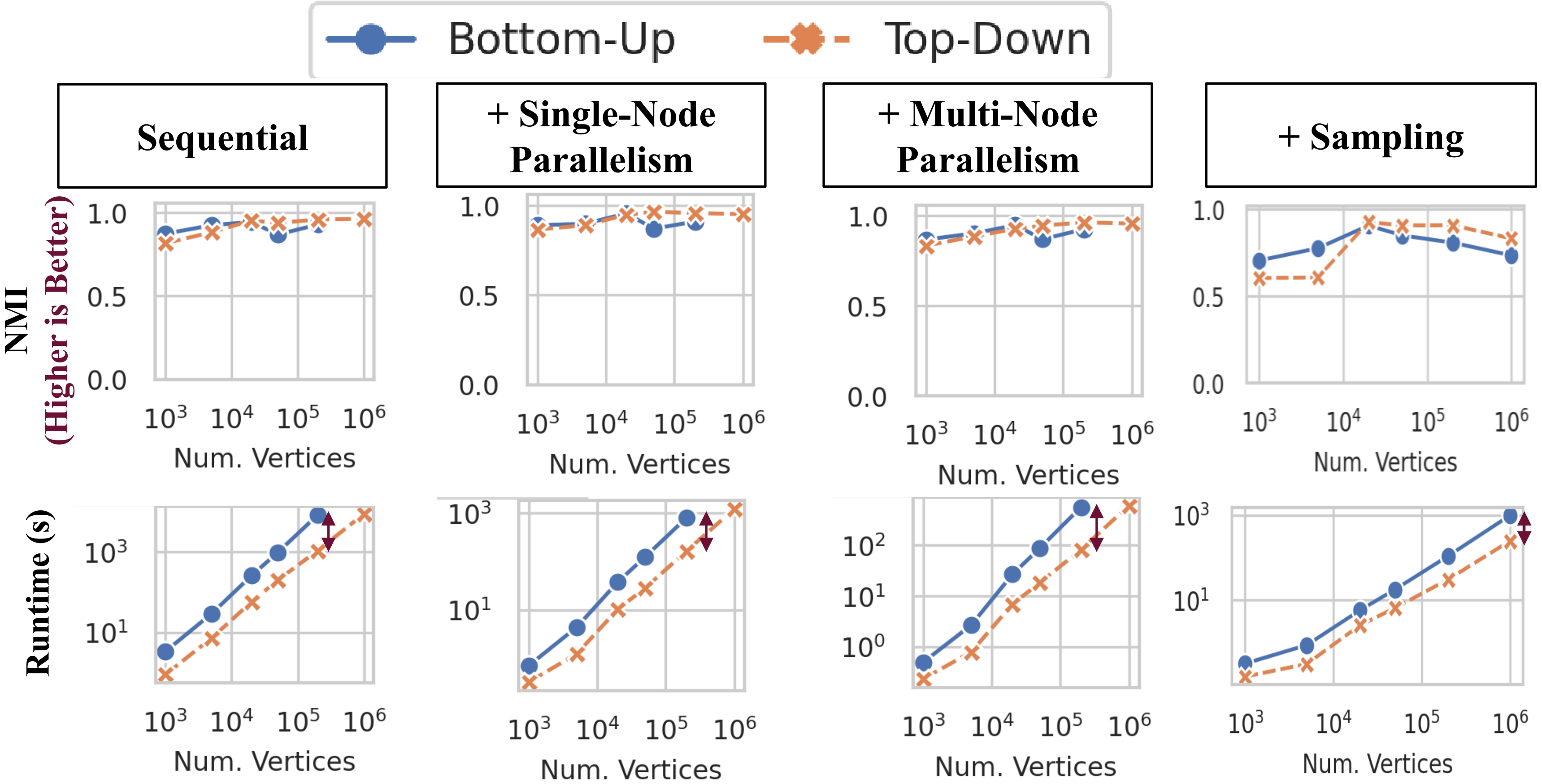
NMI = Normalized Mutual Information

Accelerated Top-Down SBP Results

Graphs
Official IEEE/Amazon/MIT Graph Challenge synthetic datasets

Num. Vertices	Num. Edges	Num. Clusters
1,000	8,032	11
5,000	51,157	19
20,000	473,329	32
50,000	1,187,682	44
200,000	4,754,406	71
1,000,000	23,772,977	125

Hardware
Ookami cluster: 1-4 nodes, 48 cores per node, 32 GB memory, Fujitsu A64FX CPUs



7.7X speedup

4.9X speedup

6.9X speedup

4.0X speedup

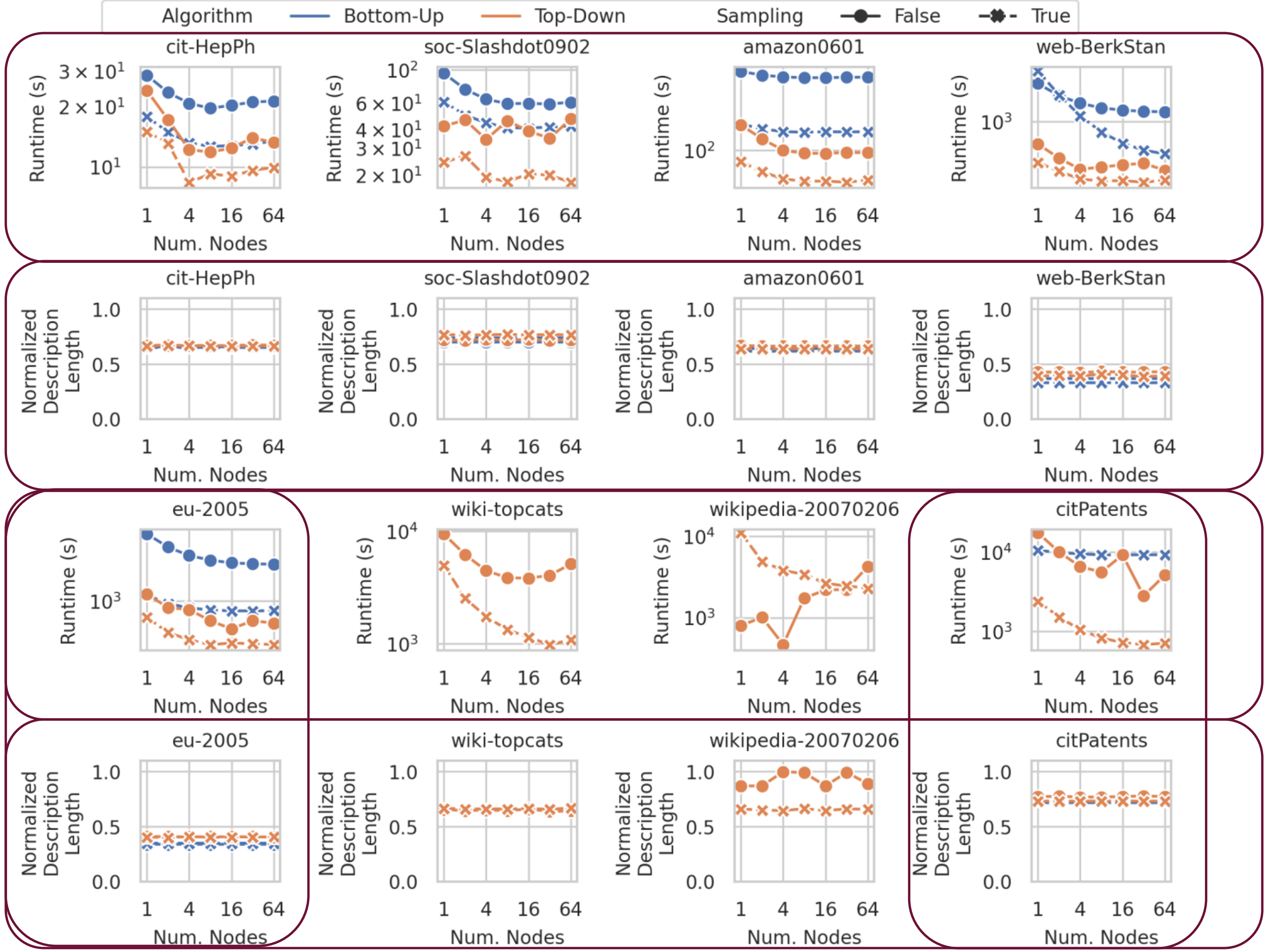
Overall speedup over sequential Bottom-Up SBP: 220X

OVERVIEW

1. Introduction to Graph Clustering
2. Background & Contributions
3. Approach
   a) Top-Down SBP
   b) Accelerated Top-Down SBP
4. **Results on Real-World Graphs**
5. Summary and Future Work

Top-Down SBP: Real-World Graph Results

Dataset Name	Num. Vertices	Num. Edges
cit-HepPh	34,546	421,534
soc-Slashdot0902	82,168	870,161
web-BerkStan	685,230	7,600,595
amazon0601	403,394	10,162,164
citPatents	3,774,768	16,518,947
eu-2005	862,664	18,733,713
wiki-topcats	1,791,489	28,508,141
wikipedia-20070206	3,515,067	45,013,315



Hardware

- Ookami (1-64 nodes, 48 cores per node, 32 GB memory, Fujitsu A64FX CPUs)

Little-to-no difference in accuracy

Accelerated Top-Down SBP is up to 13.2X faster than equivalent Bottom-Up SBP, and up to 403X faster than sequential Bottom-Up SBP

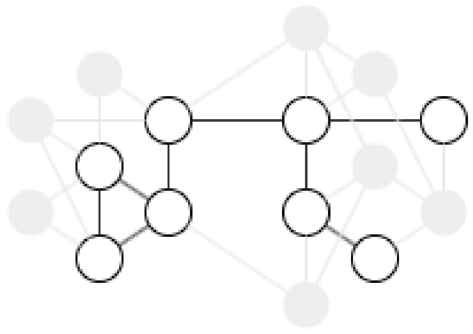
Process graphs up to 4.4X larger on same hardware

OVERVIEW

1. Introduction to Graph Clustering
2. Background & Contributions
3. Approach
   a) Top-Down SBP
   b) Accelerated Top-Down SBP
4. Results on Real-World Graphs
5. **Summary and Future Work**

Future Work

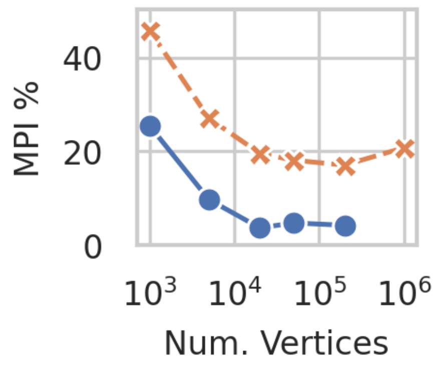
Sampling



Limitation: Large (30% - 50%) sample sizes needed to maintain accuracy

Potential Solution: Alternative data reduction methods like coresets could help reduce required sample size

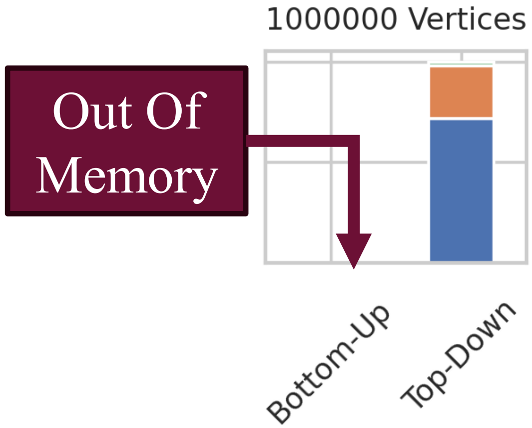
Parallelism



Limitation: Poor parallel efficiency

Potential Solution: MPI all-to-all primitives → MPI single-sided primitives
GPU acceleration

Memory Usage



Limitation: High memory usage limits graph size

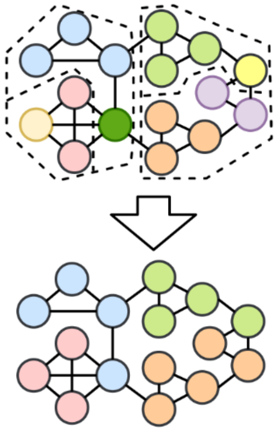
Potential Solution: Data distribution in multi-node implementation could alleviate memory bottlenecks

Bottom-Up SBP | Top-Down SBP

Summary

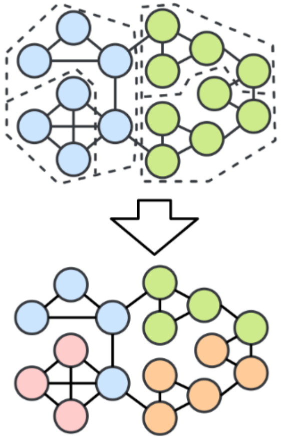
Bottom-Up Clustering

- Traditional approach to SBP
- Clusters merge over time
- × High memory requirements
- × Many MCMC moves
- × Slow compute

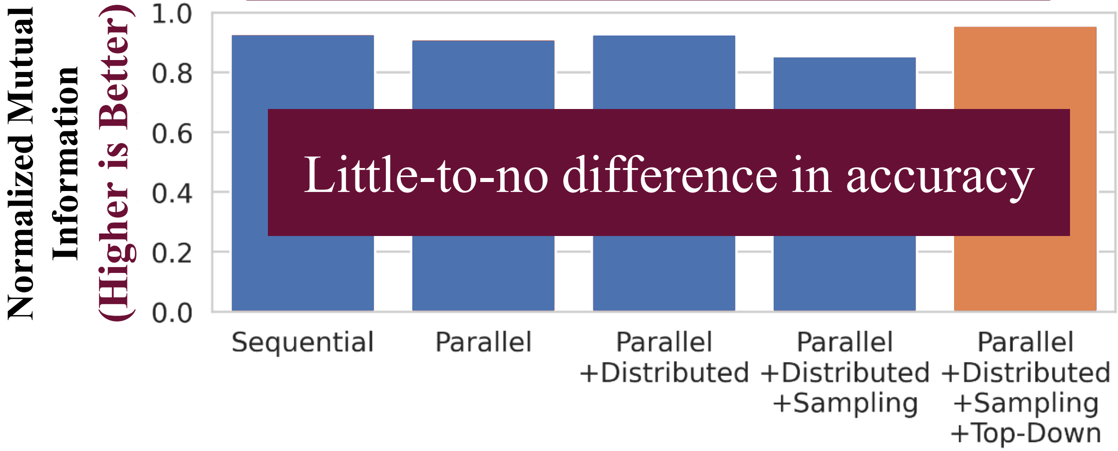


Top-Down Clustering

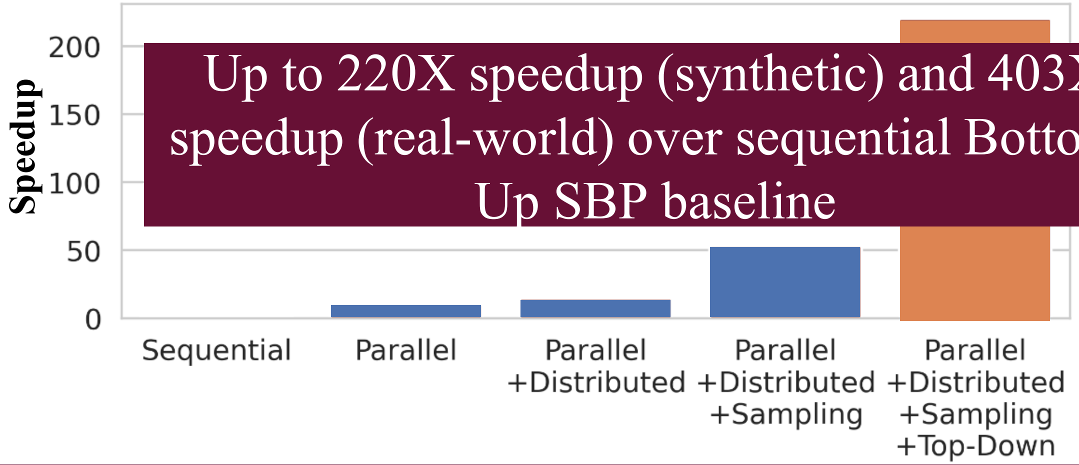
- Novel approach
- Clusters split over time
- ✓ Lower memory requirements
- ✓ Fewer MCMC moves
- ✓ Faster compute
- ✓ Accelerated with sampling + parallel and distributed computing



200,000 vertices



Little-to-no difference in accuracy



Up to 220X speedup (synthetic) and 403X speedup (real-world) over sequential Bottom-Up SBP baseline

Process graphs up to 4.1X larger on same hardware

Questions?