Highway Algorithm: Overlapping Network Community Detection at Scale Using Sparse Backbones

Two algorithms developed in this work: **Triad** and **Highway**

Zihe Zhou & Samin Aref

Department of Mechanical & Industrial Engineering
University of Toronto

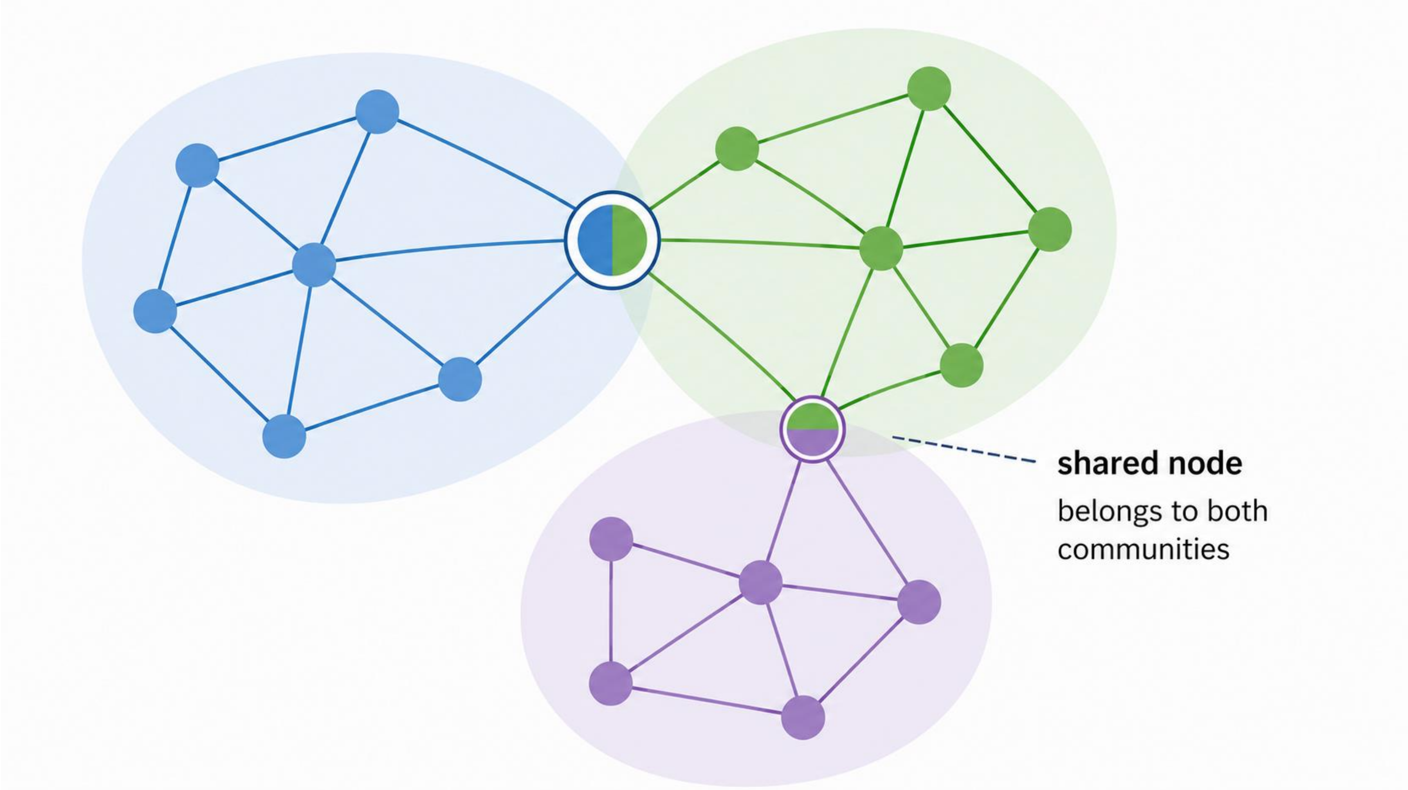
WAW 2026 CONFERENCE
June 18, 2026

Outline

Outline

What is Overlapping Community Detection?

A node may belong to more than one community at the same time.



Community

A group of nodes with stronger internal connectivity.

Overlap

Some nodes participate in multiple groups simultaneously.

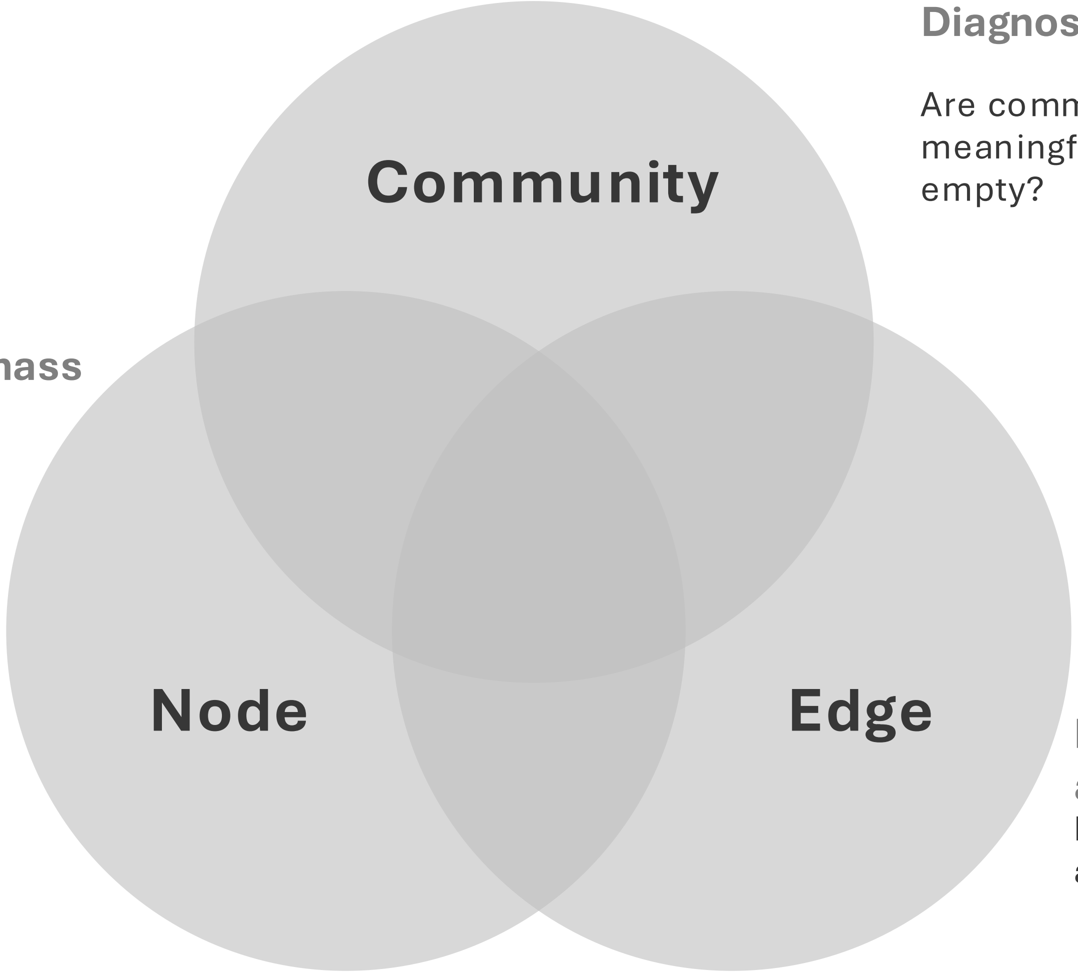
Goal of overlapping community detection

Infer multi-community memberships from network structure.

Overlapping community detection generalizes community detection beyond disjoint partitions.

How do We View Overlapping Community Detection?

Overlapping communities are shaped by node, edge, and community-level structures.



Diagnoses structural degeneracy

Are communities structurally meaningful rather than collapsed or empty?

Maximizes coherent community mass

Who belongs to which communities?

Penalizes irregular assignments

Do connected nodes support coherent assignments?

Outline

Triad: A 3-dimensional Optimization Framework

A quadratically constrained programming (QCP) model for accurate and structurally reliable overlapping assignments.

1

Node membership variables

- Soft assignment of node i to community j

2

Edge-based coherence

- Reward shared community mass along edges
- Penalize irregular boundary transitions

3

Community-level regularization

- Prevent mass collapse, empty groups, and fragmented communities

Triad's Bottleneck

! QCP optimization is computationally expensive.

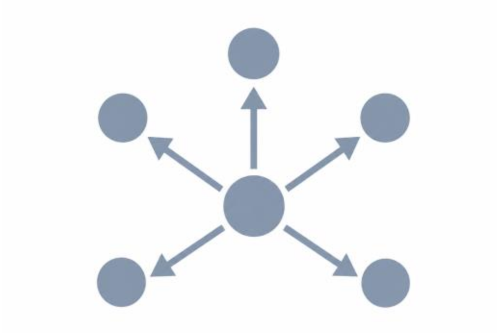
! Runtime increases quickly as network size grows.

! This makes Triad difficult to scale to large networks.

This motivates us to develop Highway: Preserve structural reliability while improving scalability.

Existing Overlapping Community Detection Methods

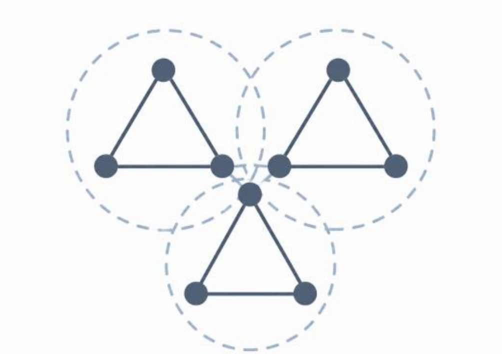
Current approaches offer different trade-offs between accuracy, interpretability, and scalability.



Label Propagation

- Fast and scalable
- Simple local update
- May be unstable or noisy

- COPRA (Gregory, 2010)
- SLPA (Xie *et al.*, 2011)



Clique Percolation

- Overlap via shared cliques
- Clear structural intuition
- Sensitive to dense patterns

- Kclique (Palla *et al.*, 2008)
- LAIS2 (Baumes *et al.*, 2005)
- MultiCom (Hollocou *et al.*, 2018)

Optimization-based

- Strong structural modeling
- High-quality assignment
- Computationally expensive

- BIGCLAM (Yan *et al.*, 2013)

Outline

The Main Challenge

Existing methods often struggle to achieve high performance and high scalability.

High performance

Optimized-based / dense inference

- Higher computational cost
- Strong structural modeling
- Triad uses node-, edge-, community-level constraints to reduce degenerate overlap assignments

Hard to optimize both simultaneously

High scalability

Local / simple methods

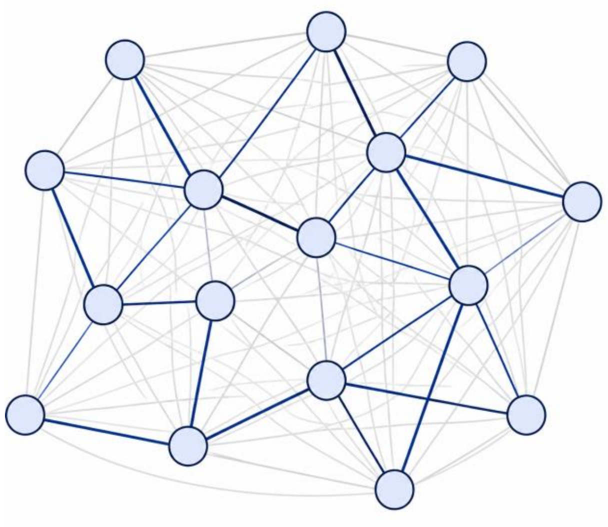
- Efficient on large networks
- Often weaker structural modeling

Highway - From Full Graph to Sparse Backbone

Do we need the full graph, or only the informative edges?

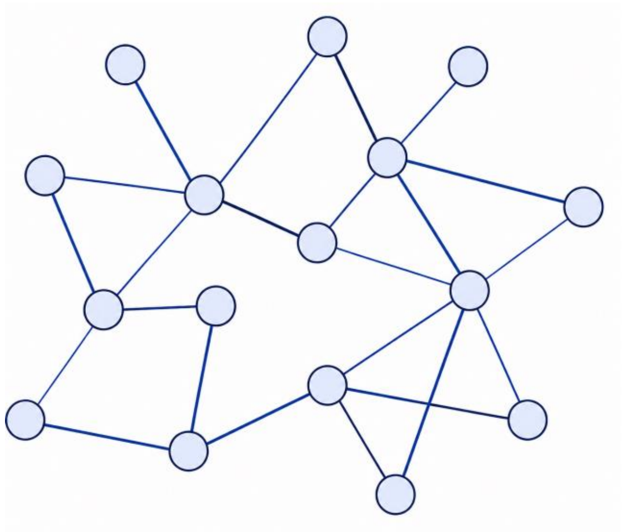
Full Graph

Many edges, mixed signals



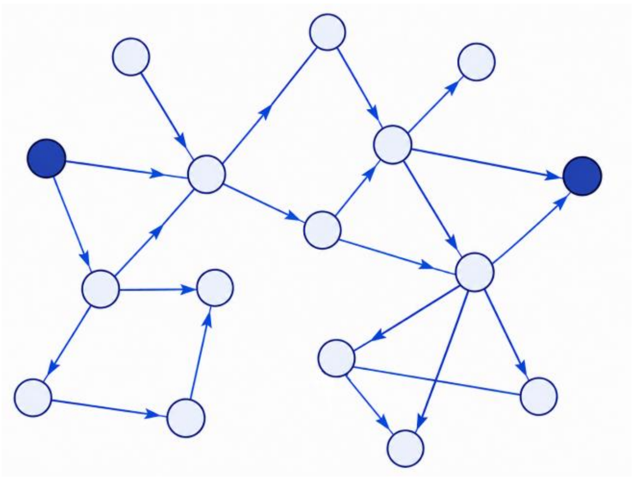
Sparse Backbone

Retain informative edges



Efficient Inference

Backbone propagation



We need to preserve the edges that carry community signals.

Outline

Sparse Backbone Construction

A hybrid edge score preserves structural signals before propagation.

■ **Hybrid Edge Score: combine global and local signals.**

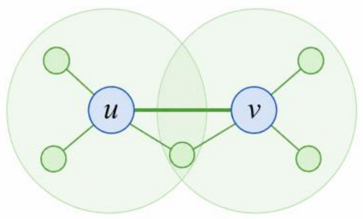
Modularity-inspired score

Favor edges stronger than random expectation

$$s_{mod}(u,v) = 1 - \frac{d_u d_v}{2m}$$

Jaccard overlap score

Favor edges with similar neighborhoods



$$s_{\mathrm{jac}}(u,v) = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

Hybrid score

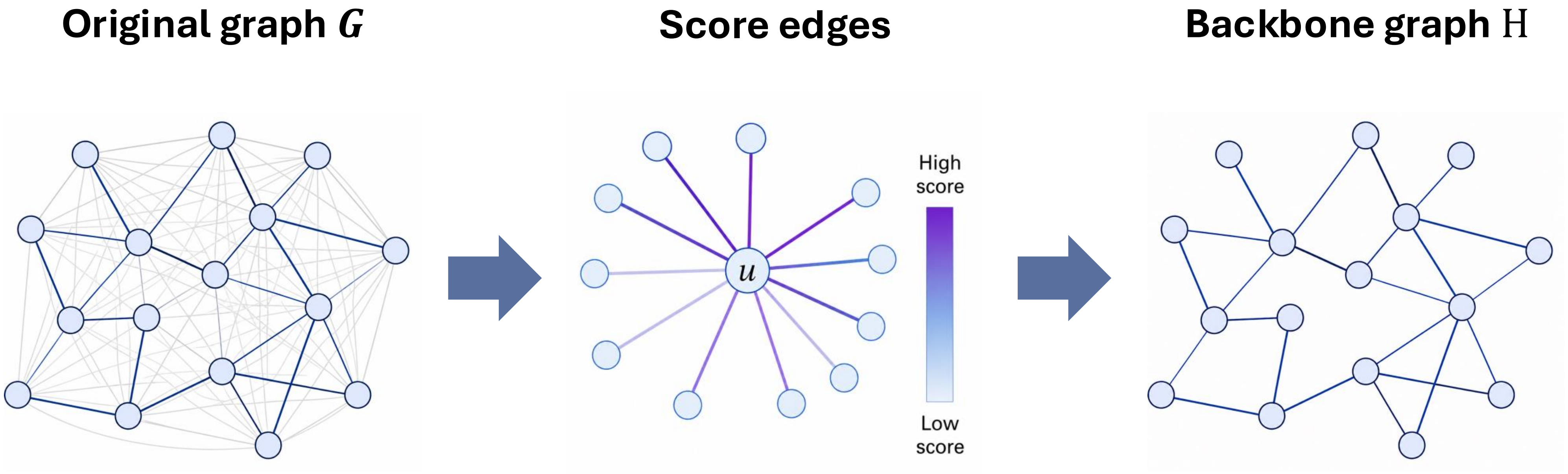
Balance global importance and local similarity

$$s(u,v) = \omega * s_{mod}(u,v) + (1-\omega) * s_{\mathrm{jac}}(u,v)$$

Sparse Backbone Construction

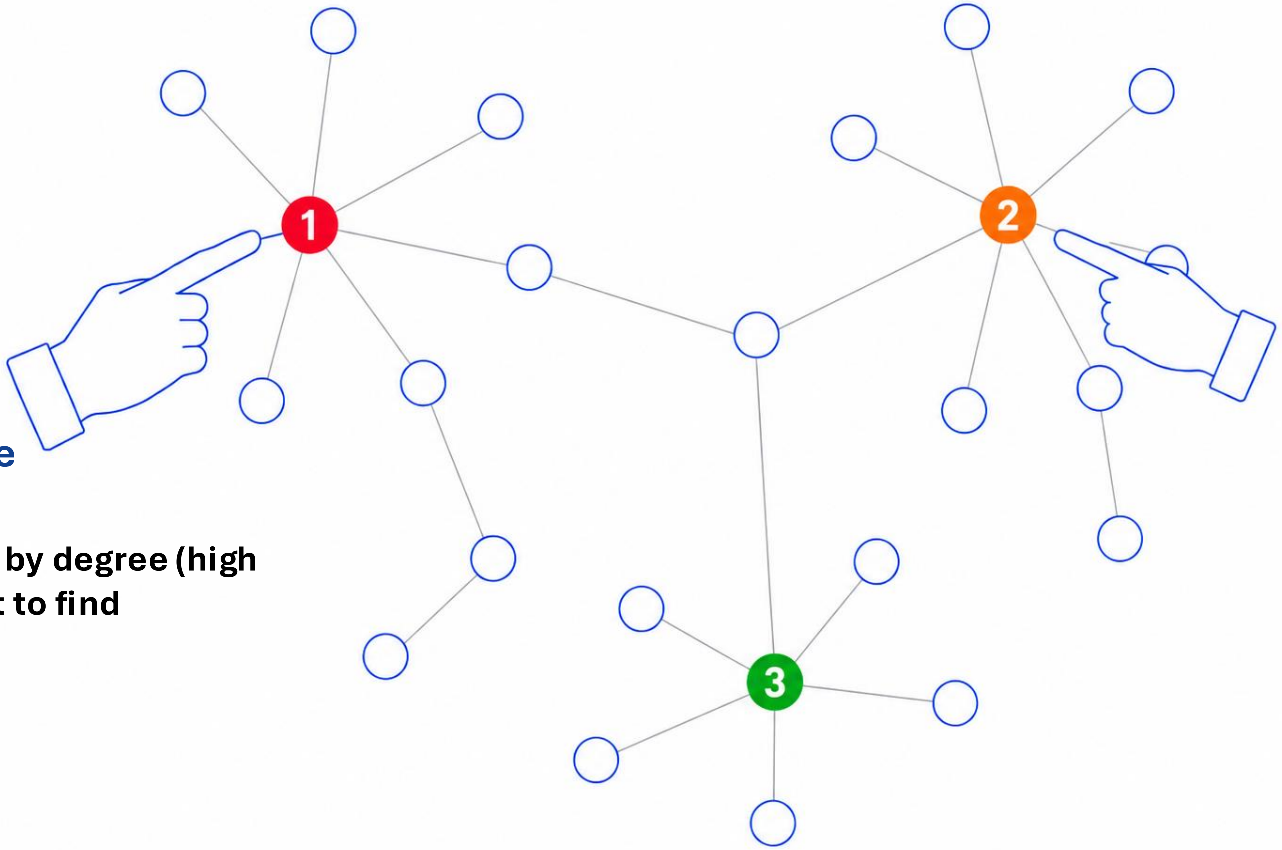
A hybrid edge score preserves structural signals before propagation.

- **Construct Backbone: retain the strongest edges around each node based on their edge scores.**



Selected Distributed Anchor Nodes

High-degree nodes are selected greedily, while nearby nodes are covered to avoid redundant anchors.



1 **Sort by degree**

Order all nodes in G by degree (high to low). Scan the list to find uncovered nodes.

2 **Greedy cover**

Select the first uncovered node as an anchor a_c. Cover a_c and its neighbors. Repeat until all nodes are covered.

For each anchor a_c

$$\alpha_{a_c}(c) = 1$$

$$\alpha_{a_c}(j) = 0, \forall j \neq c$$

Neighbor-only Propagation

Nodes receive anchor signals from neighbors, not from themselves.

1 Propagate on backbone

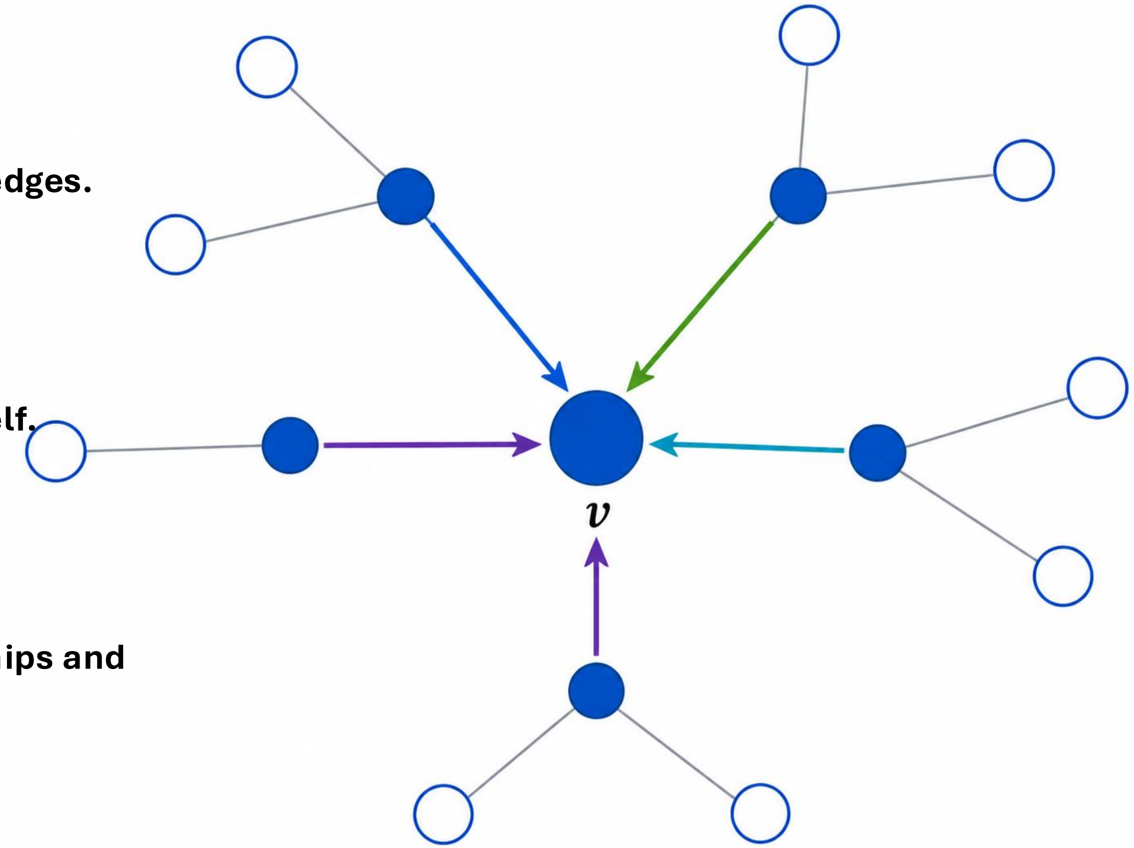
Use only sparse backbone edges.

2 Neighbor-only update

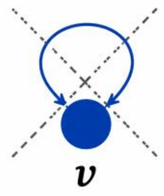
A node cannot reinforce itself.

3 Keep top-r_p signals

Retain strongest memberships and normalize.



No self-reinforcement



Self-loop is not allowed.

$$a_v(c) = \sum_{u \in \mathcal{N}_H(v)} w_{uv} a_u(c)$$

$$w_{uv} = \frac{1}{\sqrt{d_u d_v}}$$

From Propagated Signals to Communities

Group similar anchor patterns and calibrate memberships into overlapping communities.

1 Extract patterns

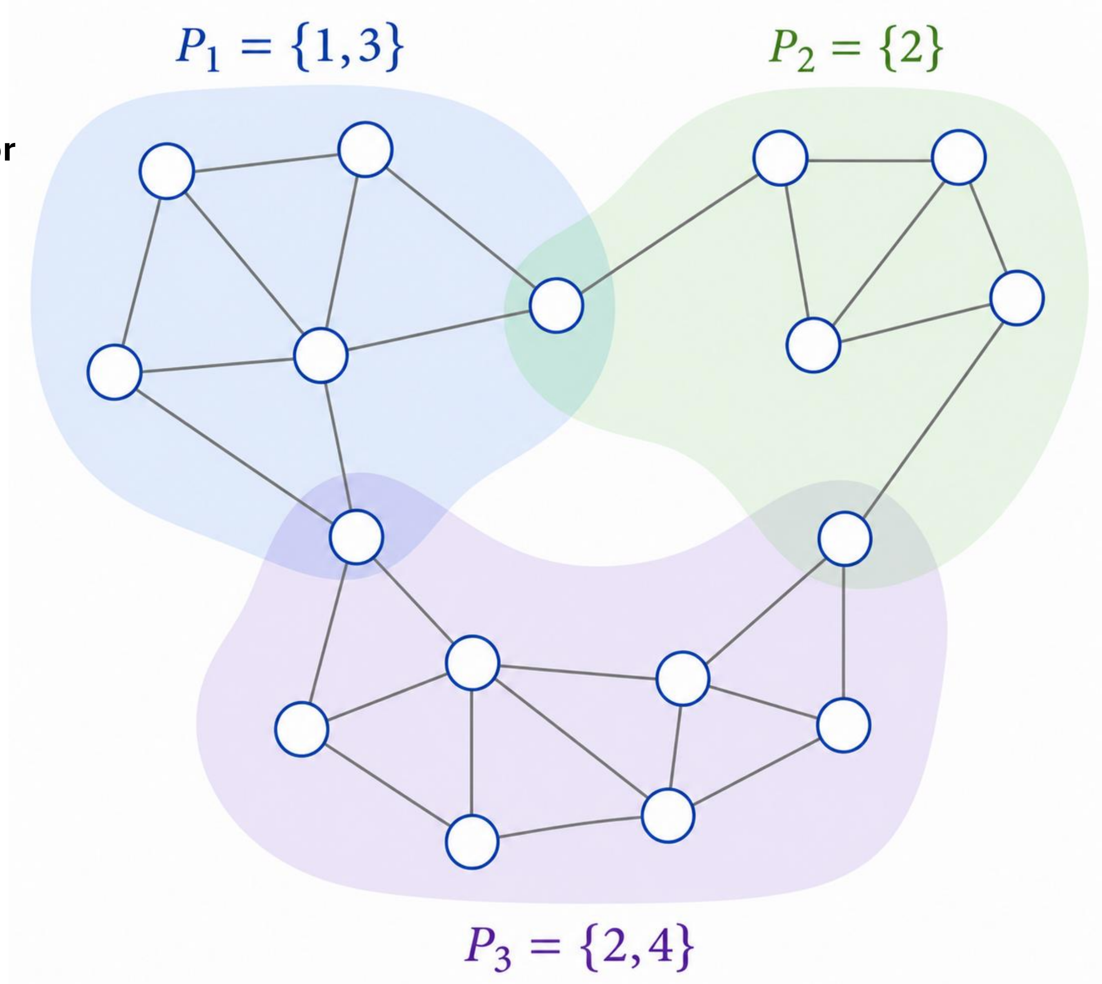
Nodes with the same active anchor set from a pattern.

2 Evaluate reliability

Reliable patterns have strong internal support and structured boundaries

3 Calibrate memberships

Combine pattern-level and neighbor-level consensus.



Calibration rule

Pattern consensus
+
Neighbor consensus

Outline

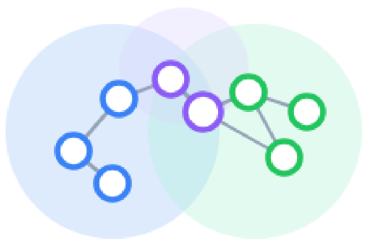
Experimental Setup

Evaluating Highway on synthetic overlapping network benchmarks.

Benchmark

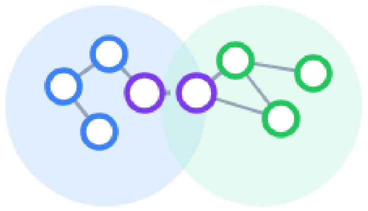
LFR – Lancichinetti *et al.* (2008)

Classic benchmark for overlapping community detection.



ABCD+o^2 – Jordan *et al.* (2025)

Overlapping benchmark with overlap and noise controls.



Range of mixing parameter values

LFR mixing parameter (noise) μ_w	0.1	0.2	0.3	0.4	0.5	0.6	0.7
ABCD+o^2 mixing parameter (noise) ξ	0.05	0.15	0.25	0.35	0.45	0.55	0.65

Difficulty increases →

Experimental Setup

Evaluating Highway on synthetic overlapping network benchmarks.

Compared Methods

10 existing algorithms + HighwayFull (ablation)

Label Propagation

SLPA, COPRA

Optimization-based

BigClam

Clique / Local Expansion

Kclique, LAIS2, MultiCom

Ablation

HighwayFull (without backbone graph)

Evaluation Metrics

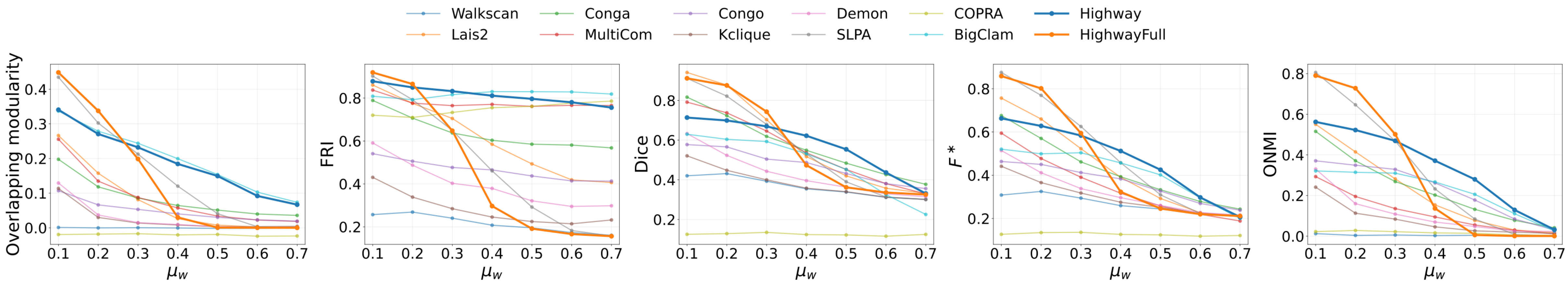
Higher values indicate better performance.

- **Q** — Overlapping Modularity
- **FRI** — Fuzzy Rand Index (FRI)
- **Dice** — Dice Coefficient
- **F*** — F-score
- **ONMI** — Overlapping Normalized Mutual Information

Performance on LFR Benchmark

Highway maintains competitive detection quality under increasing mixing and noise.

LFR benchmark: increasing mixing parameter (noise) μ_w

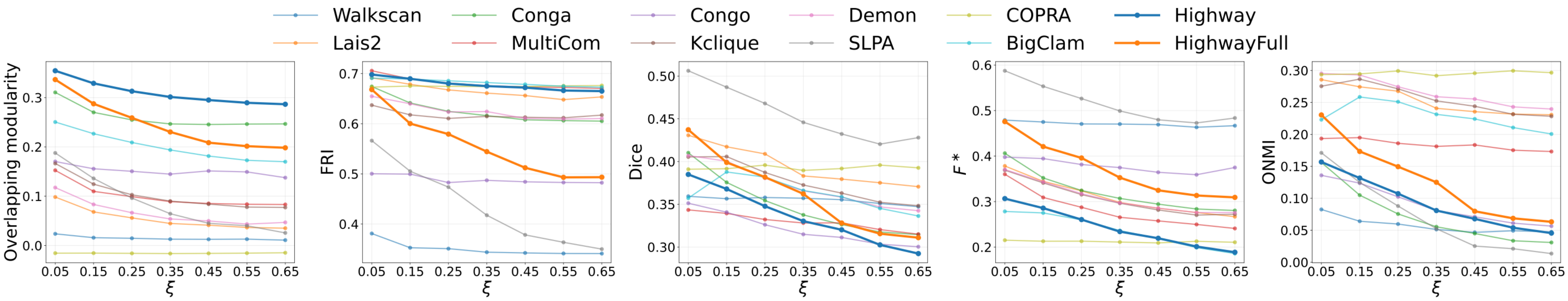


Highway maintains competitive detection quality under increasing mixing, especially in structurally challenging settings.

Performance on ABCD+o^2 Benchmarks

Highway maintains competitive detection quality under increasing mixing and noise.

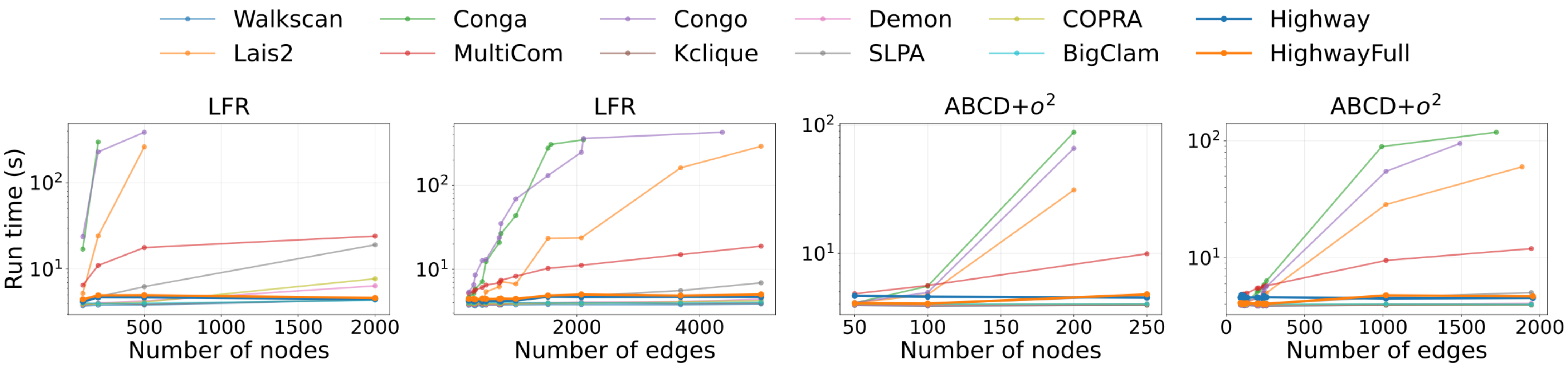
ABCD+o^2 benchmark: increasing mixing parameter (noise) ξ



The purpose of Highway is not winning every metric, but to provide a stable and scalable solution with strong quality.

Scalability Without Sacrificing Quality

Highway achieves competitive detection quality while maintaining stable runtime.



Highway provides a favorable accuracy–efficiency trade-off, with runtime remaining stable as instance size grows.

Outline

Takeaways & Conclusions

Highway shifts overlapping community detection from full-graph inference to backbone-guided signal propagation.

- Highway uses a sparse backbone for scalable overlapping community detection.
- Compared with optimization-based alternatives, Highway avoids expensive dense full-graph inference.
- Compared with local/simple methods, Highway preserves stronger structural signals before propagation.
- Its core pipeline combines sparse backbone construction, anchor propagation, and pattern calibration.
- Highway will be open-source and publicly available for future research and applications.

Highway provides a favorable performance-scalability trade-off for large-scale overlapping community detection.

Availability & Contact

We will make Highway widely available to support reproducible research.

Highway Preprint on arXiv

The preprint of our paper will be available on arXiv after peer-review.

Algorithm in CDlib

Highway will be included in CDlib, the python library for community detection.

Code on GitHub

The source code and examples will be open-source on GitHub.

zhouchen.zhou@mail.utoronto.ca

We welcome your questions and feedback!

References

1. Gregory, S. (2010). Finding overlapping communities in networks by label propagation.

2. Xie, J., Szymanski, B. K., & Liu, X. (2011). SLPA: Uncovering overlapping communities in social networks via a speaker-listener interaction dynamic process.

3. Palla, G., Derényi, I., Farkas, I., & Vicsek, T. (2005). Uncovering the overlapping community structure of complex networks.

4. Baumes, J., Goldberg, M., Krishnamoorthy, M., Magdon-Ismail, M., & Preston, N. (2005). Efficient identification of overlapping communities.

5. Hollocou, A., Bonald, T., & Lelarge, M. (2018). Multiple local community detection.

6. Yang, J., & Leskovec, J. (2013). Overlapping community detection at scale: A nonnegative matrix factorization approach.

7. Lancichinetti, A., Fortunato, S., & Radicchi, F. (2008). Benchmark graphs for testing community detection algorithms.

8. Jordan, D., Prałat, P., & Théberge, F. (2025). The ABCD+o^2 benchmark for overlapping community detection.

9. Nicosia, V., Mangioni, G., Carchiolo, V., & Malgeri, M. (2009). Extending modularity definition for directed graphs with overlapping communities.

References

10. Hüllermeier, E., & Rifqi, M. (2009). A fuzzy variant of the Rand index for comparing clustering structures.

11. Lancichinetti, A., Fortunato, S., & Kertész, J. (2009). Detecting the overlapping and hierarchical community structure of complex networks.

12. DeWolfe, R., Prałat, P., & Théberge, F. (2026). A pragmatic method for comparing clusterings with overlaps and outliers.

13. Rossetti, G., Milli, L., & Cazabet, R. (2019). CDlib: A Python library to extract, compare and evaluate communities from complex networks.

Thank You!

Questions & Discussion

I am currently looking for PhD positions.

B.C.S., Computer Science Major, AI Specialization, Computational Mathematics Minor, University of Waterloo

M.Eng., Mechanical and Industrial Engineering, University of Toronto, expected August 2026