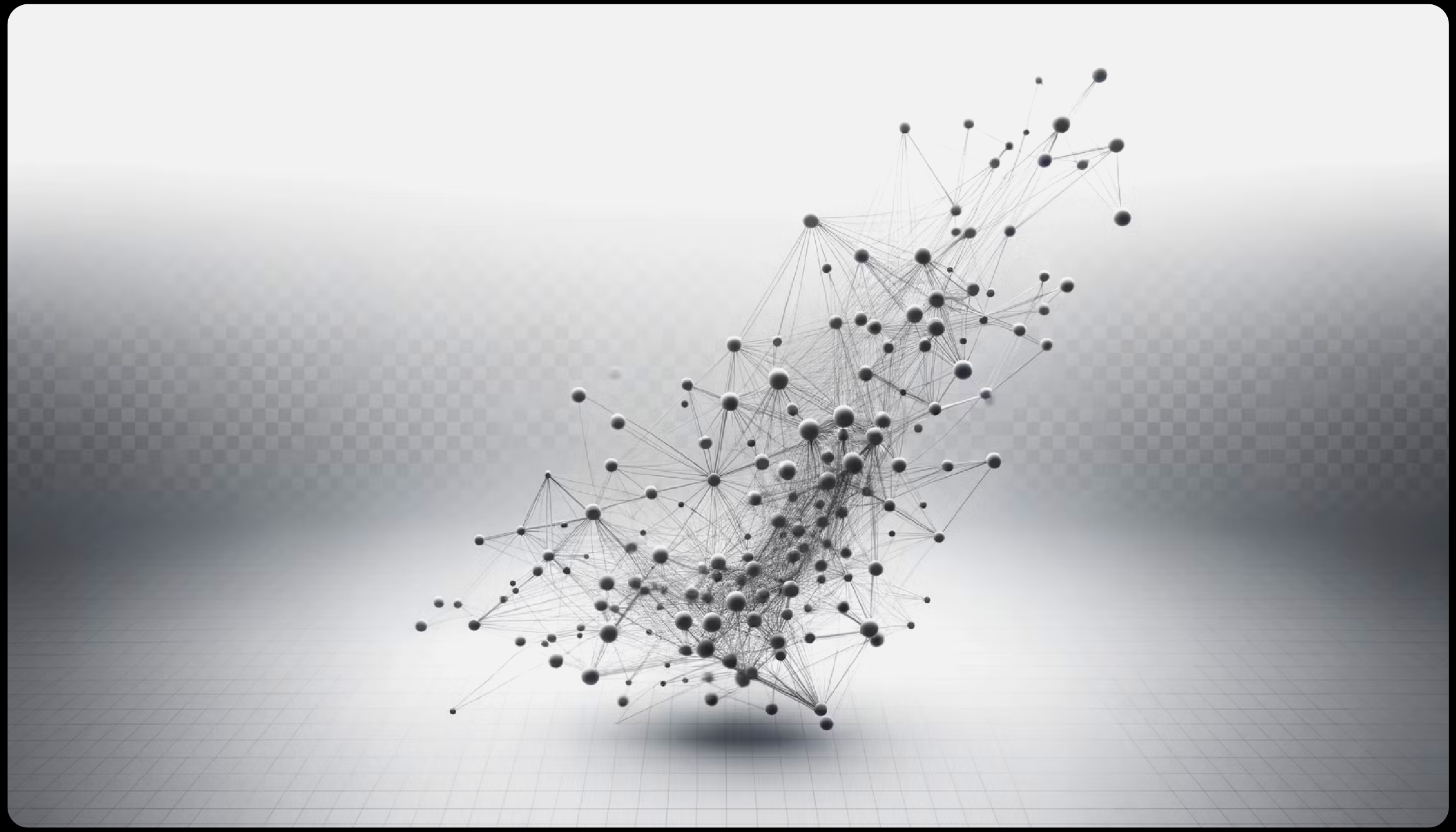


Vertex clustering in diverse dynamic networks

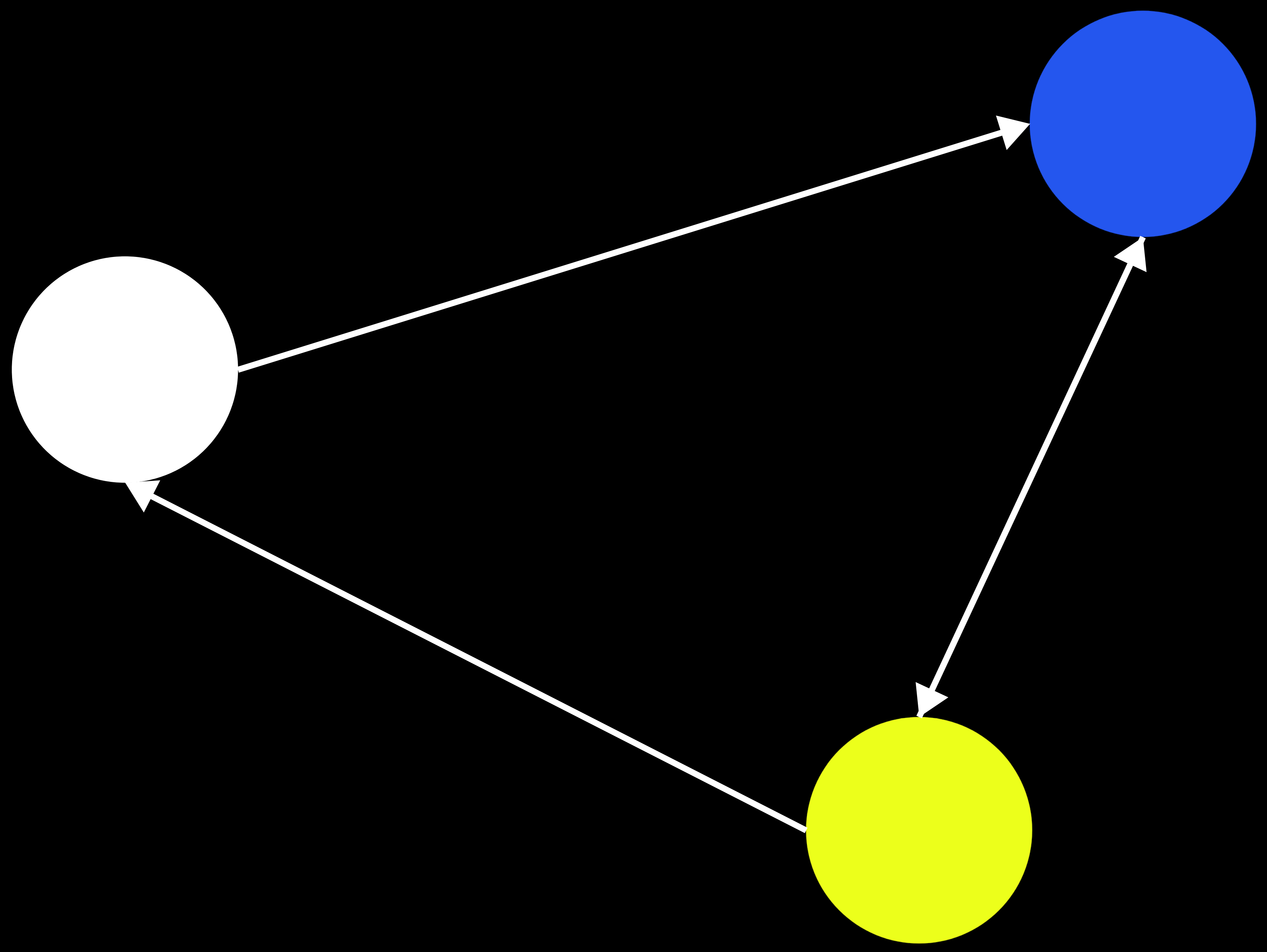
QCAM 2024 • ICERM
June 24, 2024 • 3:15 PM • Providence, RI

Dev Dabke
Level Ventures

Acknowledgements

Joint with Olga Dorabiala (University of Washington)

Setup



Motivation & Applications

- Animal herding: giraffes in Kenya
- Social networks, epidemiological concerns
- Economic agents: funds, companies, people
- Political actors and their voting patterns

Guiding Question

What is the relationship between the vertices as they evolve over time?

Previous Approaches

- Aggregation: convert dynamic graph to static one
- Community detection (heuristics)
- Evolutionary clustering
- Online algorithms
- Machine Learning (GNNs, GATs, etc.)

Our Approach: Spatiotemporal Graph *k*-means (STG*k*M)

1. Practical + Computable
2. Unsupervised with one parameter
3. Spatiotemporal smoothness
4. Theoretical guarantees
5. Experimental validation

The Method

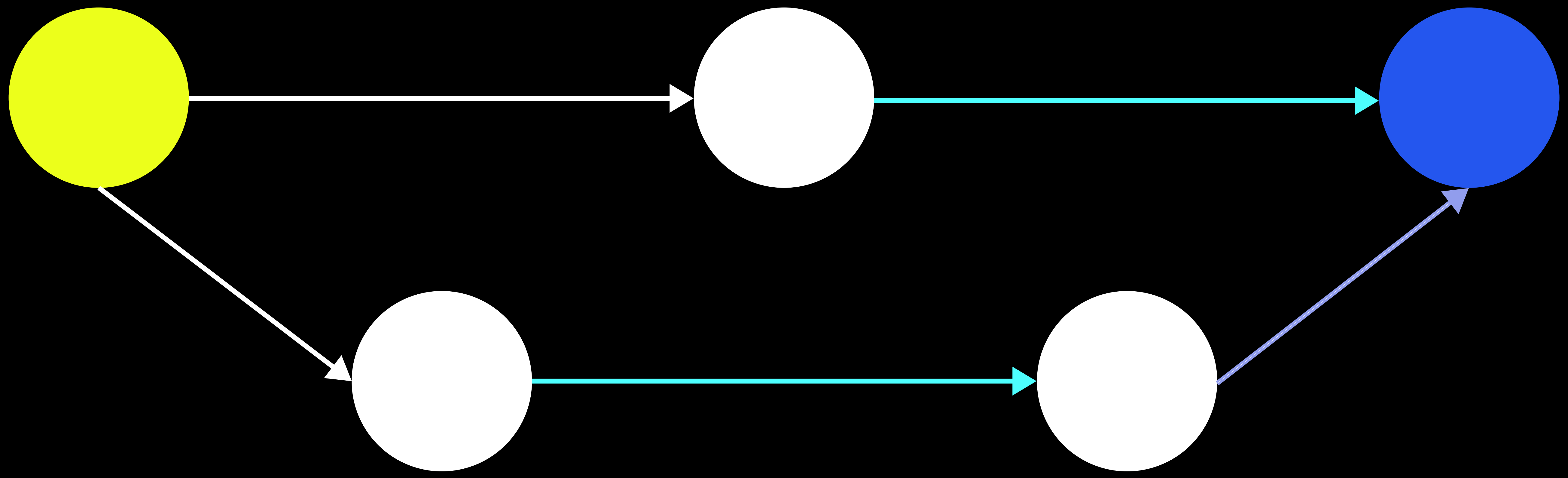
Mathematical Goal

Can we find a "good" partition of the vertices?

partition of k elements

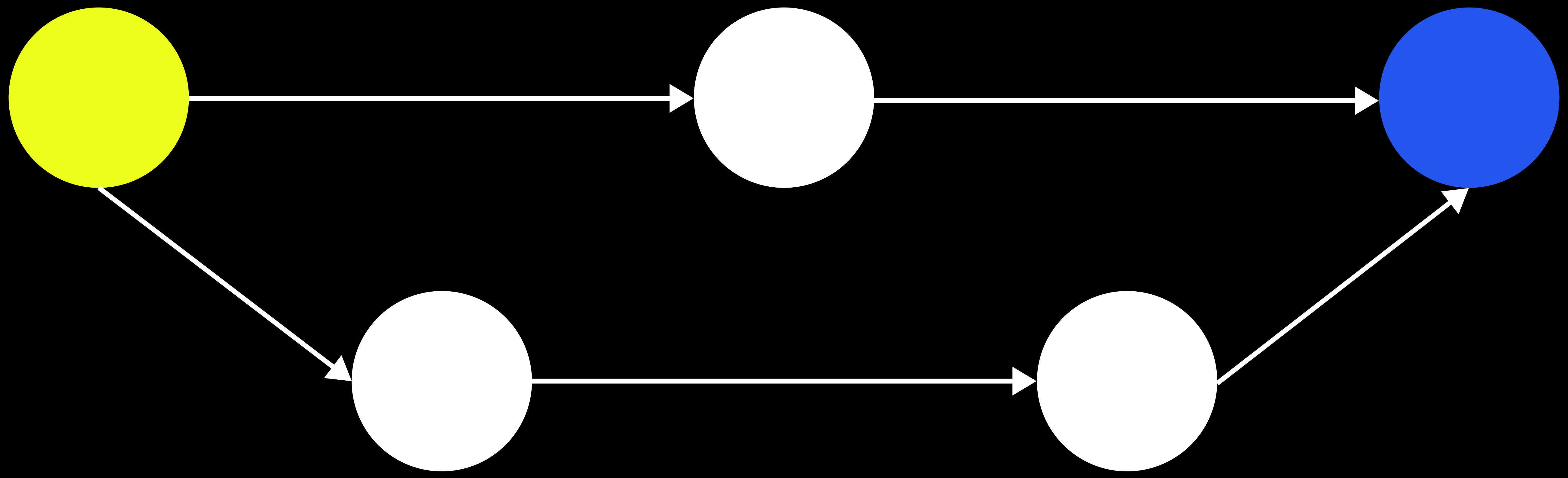
Primer: shortest journey

The shortest *dynamic* path between two vertices traversing one edge at a time.



Primer: shortest journey

The shortest *dynamic* path between two vertices traversing one edge at a time.



Mathematical Goal

Can we find a "good" partition of the vertices?

good: minimizes all shortest journeys

"*k*-means" Ideal Objective

"participation"
regularization matrix

distance based on shortest journey

$$\min_{c \in \mathcal{C}, W \in \mathcal{W}} \sum_{t \in \mathbb{T}} \sum_{u \in V} \sum_{j \in [k]} W_{u,j}^{t} \cdot \tilde{\delta}^{t}(u, c_j^t)$$

all possible
clusterings over
time & space

number of elements in our partition

set of vertices

timesteps

Relaxed Objective

$$\min_{c,W} \sum_{u \in V} \sum_{j \in [k]} W_{u,j}^{t} \cdot \delta^{t}(u, c_j^{t})$$

$$\text{such that } \delta^{t-q}(c_j^{t-1}, c_j^{t}) \leq \lambda, \text{ where } 1 \leq q \leq \gamma \text{ and } 1 \leq j \leq k$$

Algorithm Overview

1. Solve the relaxed objective (using updated versions of classical techniques)
2. Find cluster membership of each vertex at each timestep
3. Collect information over time for each vertex
4. Use agglomerative (or other) static clustering for each vertex based on cluster membership

Theoretical Results

"Standard" Dynamic Networks

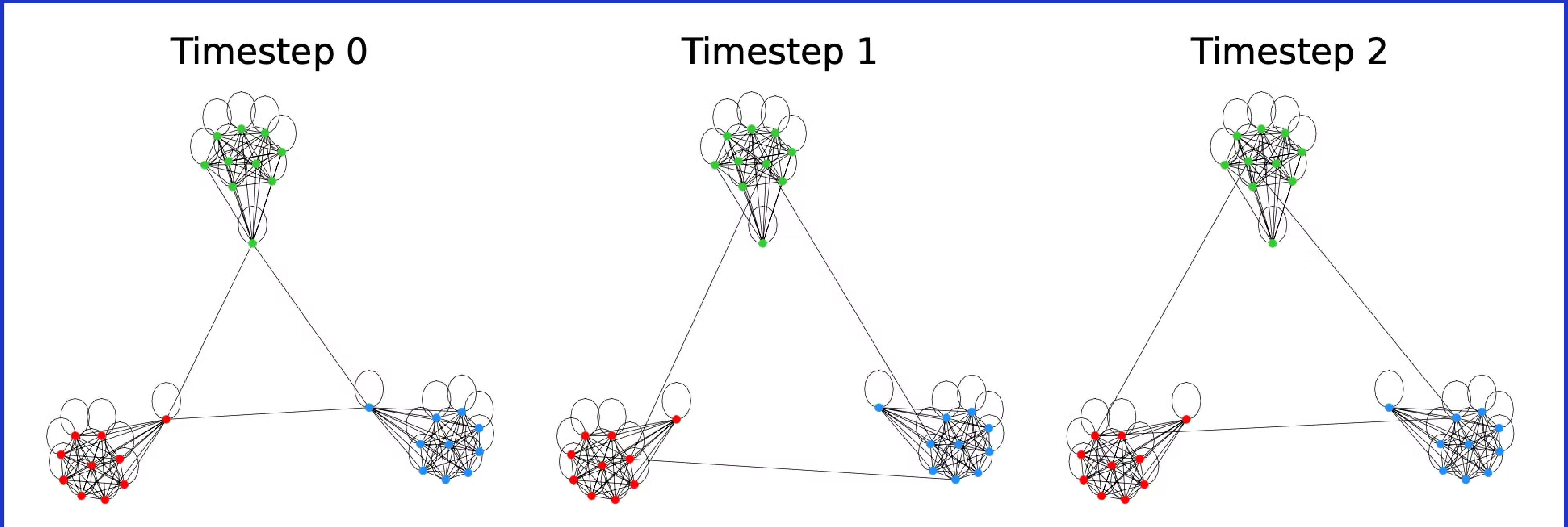
- A side quest: developing "standard" dynamic networks to test things with.
- Analogy with static graphs:
  - Cliques and friends: K5, K3,3; etc.
  - Paths
  - Cycles

Theorem 1. Connected Components

If a (non-stranding) dynamic network has self-loops, then using STG*k*M is just connected components.

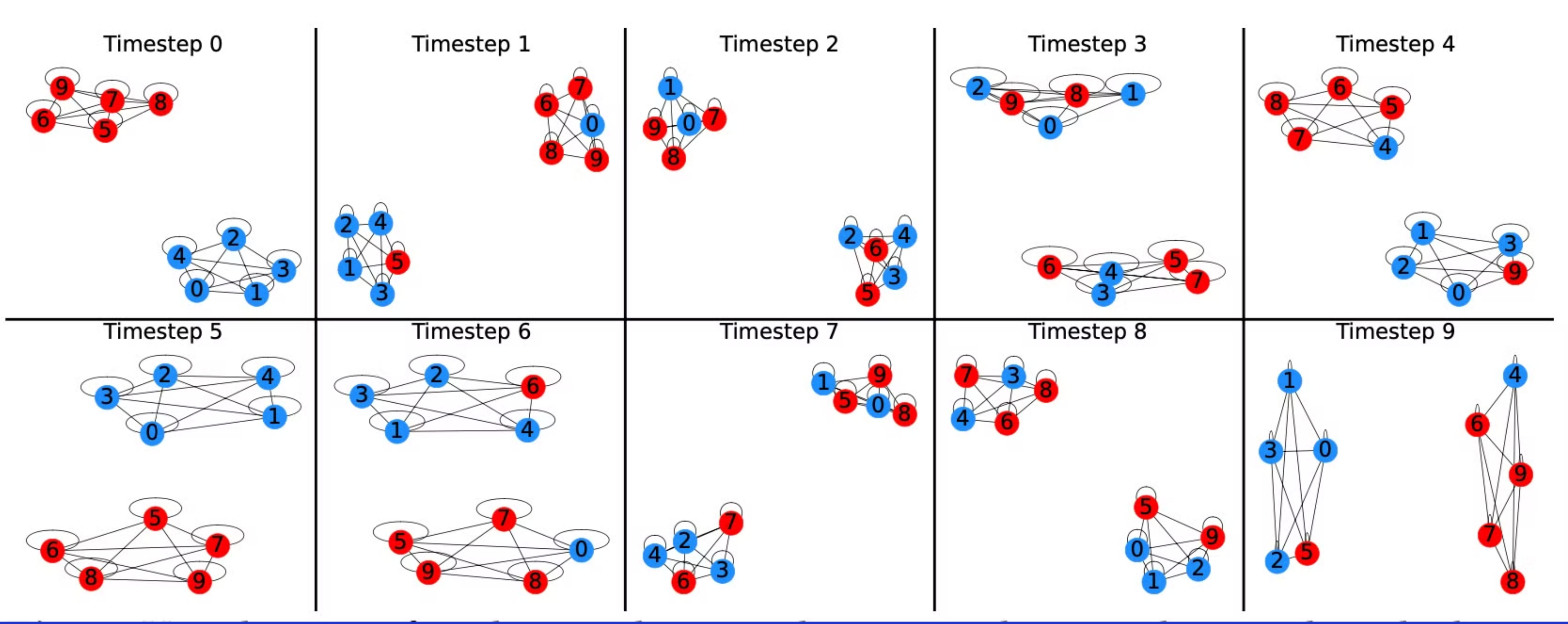
Theorem 2. Single Component

For certain connected graphs without self-loops, STGkM makes clusters that are more "correct" than connected components because connected components is a **strong** definition of a cluster.



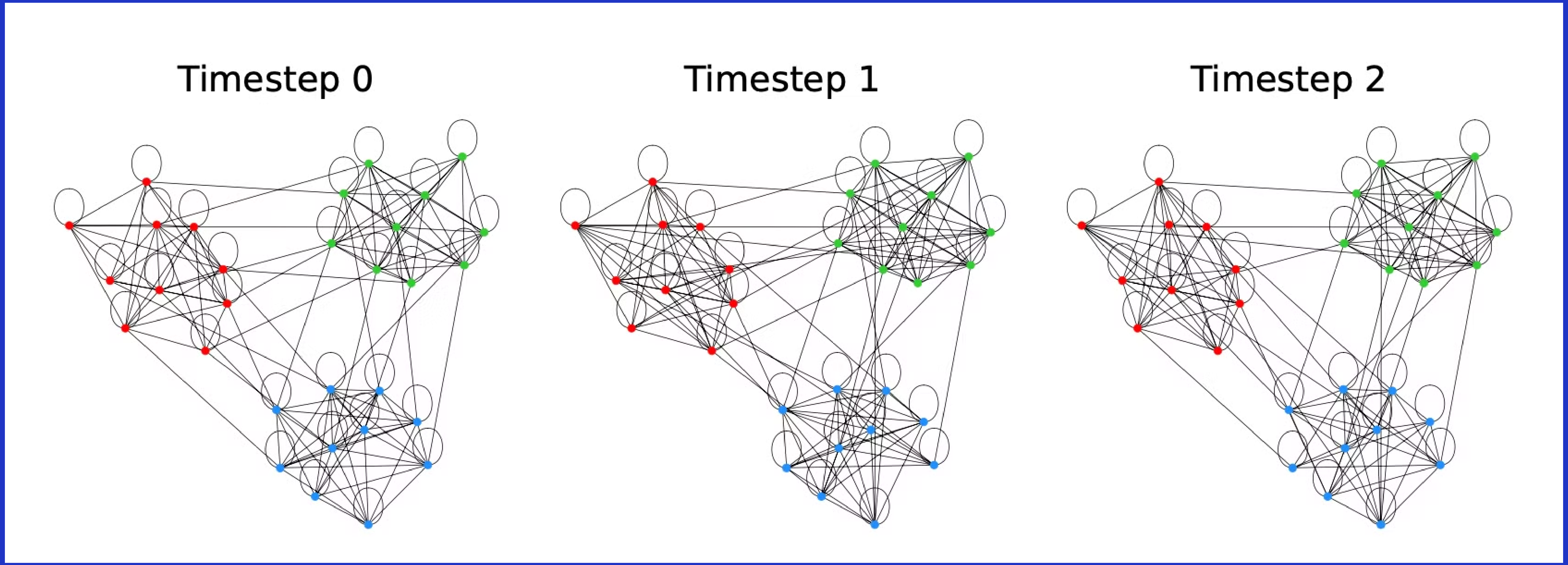
Theorem 3. Better than Aggregation

STGkM makes clusters that are more "correct" than simply counting the total number of edges between two vertices over time (because there are dynamic networks with a uniform number of total edges, but multiple obvious clusters).

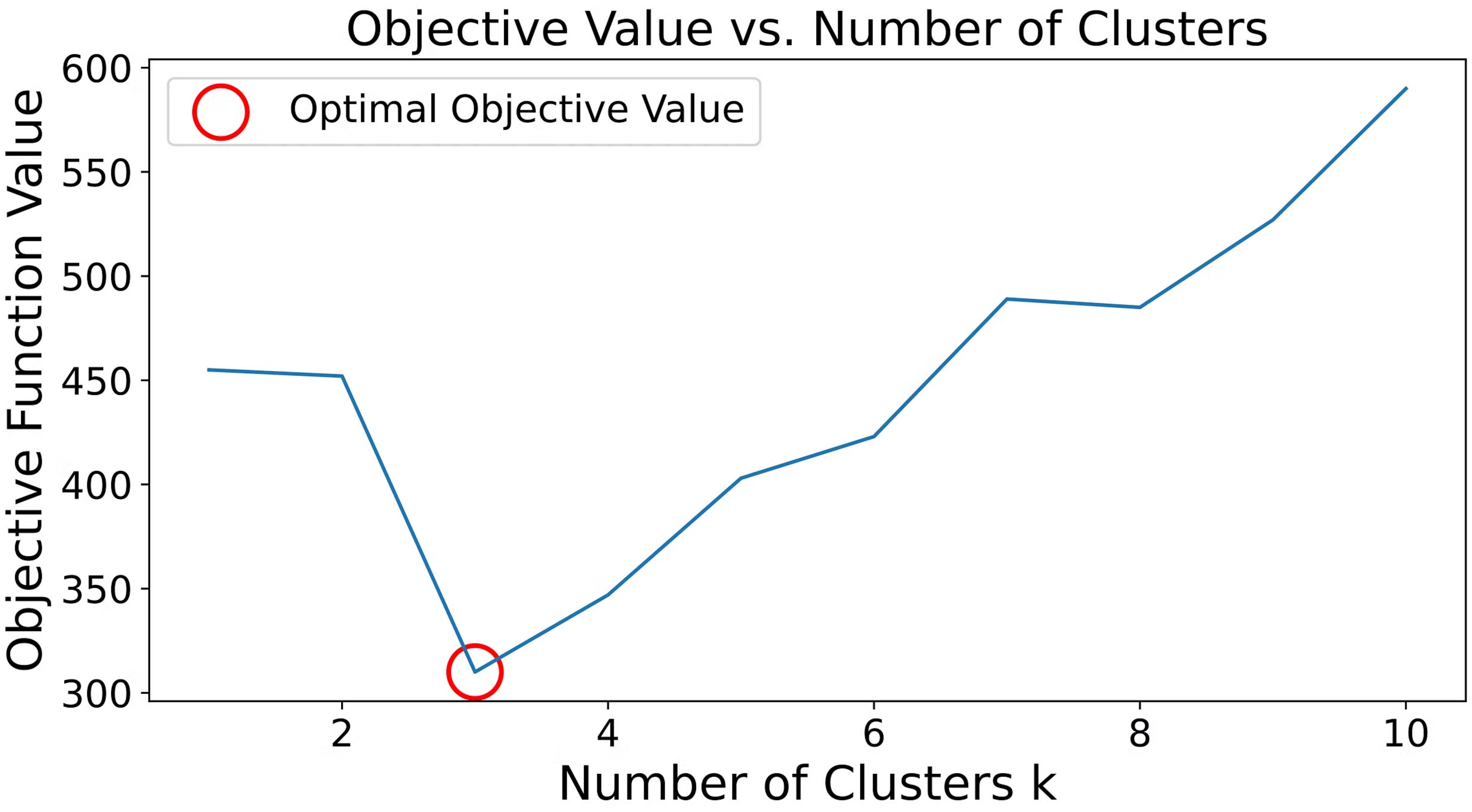


Theorem 4. Works in the Stochastic Setting

STGkM works in expectation.



Finding *k* with the Elbow Method



Experimental Results

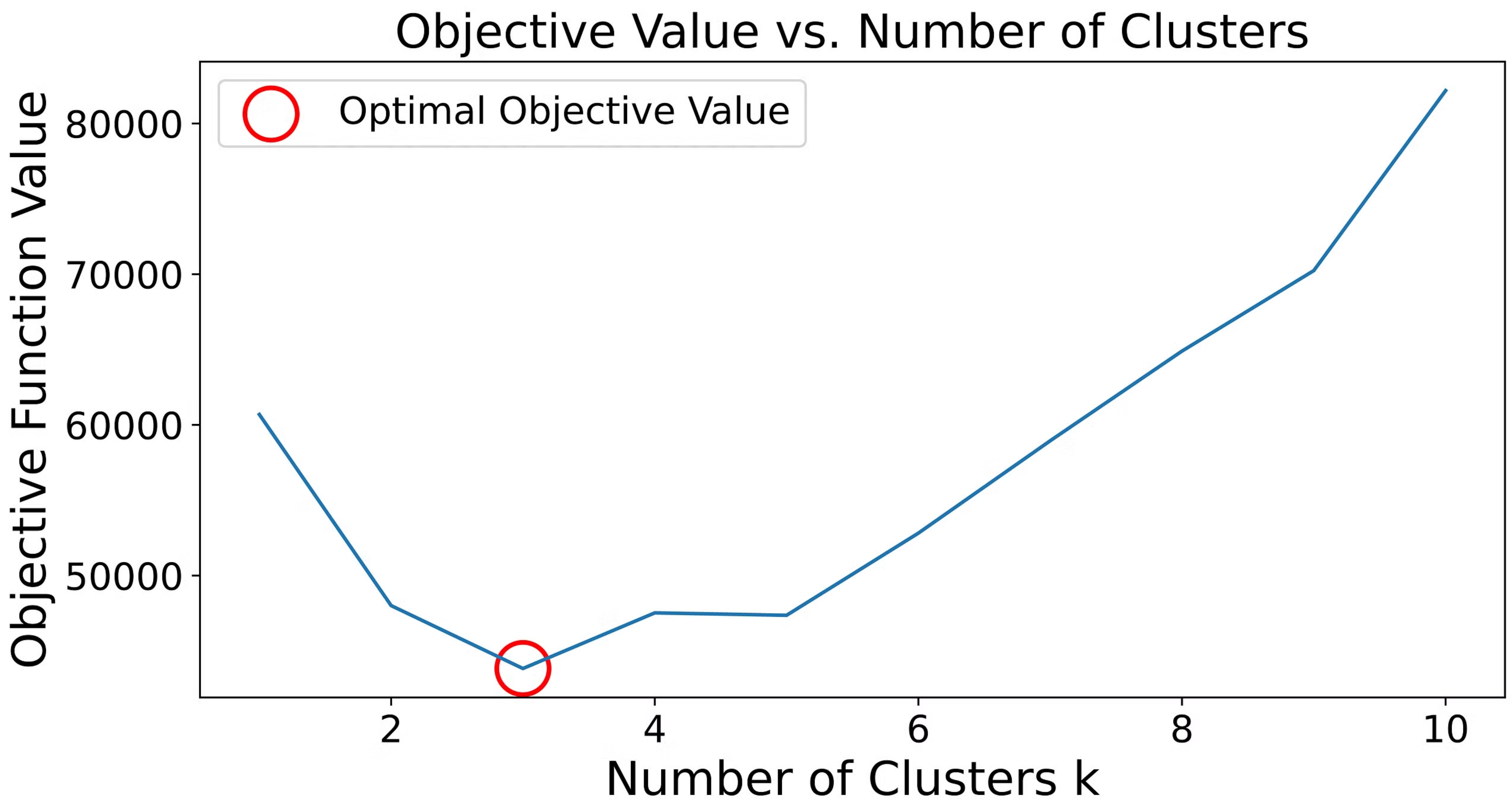
Synthetic Data

Dataset	**STGkM**	**CC**	**k-medoids**	**DCDID**
Clique-cross-Clique	**1.000**	0.019	**1.000**	0.019
Strong Random Clique-cross-Clique	**0.989**	0.032	0.932	0.240
Mixed Random Clique-cross Clique	**1.000**	**1.000**	**1.000**	**1.000**
Weak Random Clique-cross-Clique	0.920	**1.000**	0.971	0.983
Theseus Clique	**1.000**	**1.000**	0.541	**1.000**
Three Clusters	**1.000**	0.763	**1.000**	0.995

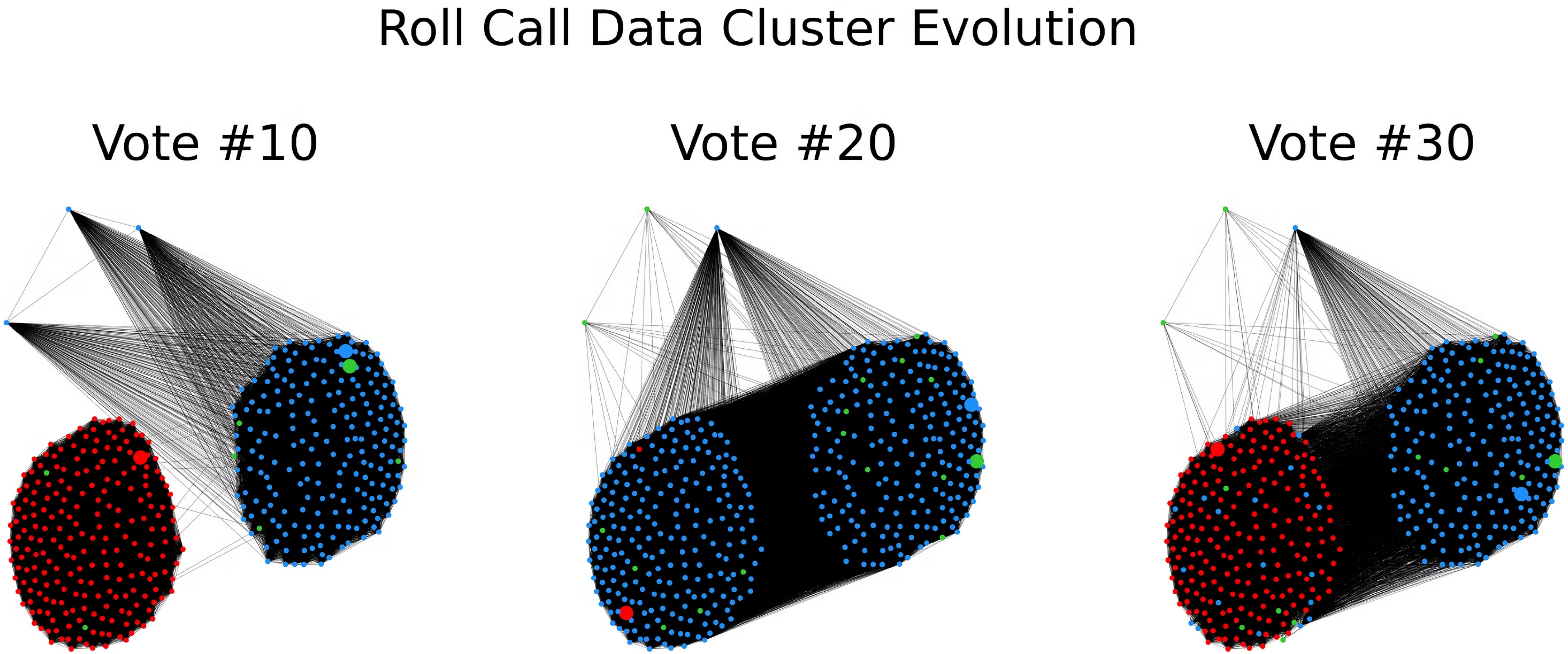
Rollcall

1. Vertices: member of US House of Representatives
2. Timestep: each rollcall vote ordered over time
3. Edges: two members are connected at a timestep iff they vote the same way

Rollcall: Number of Clusters



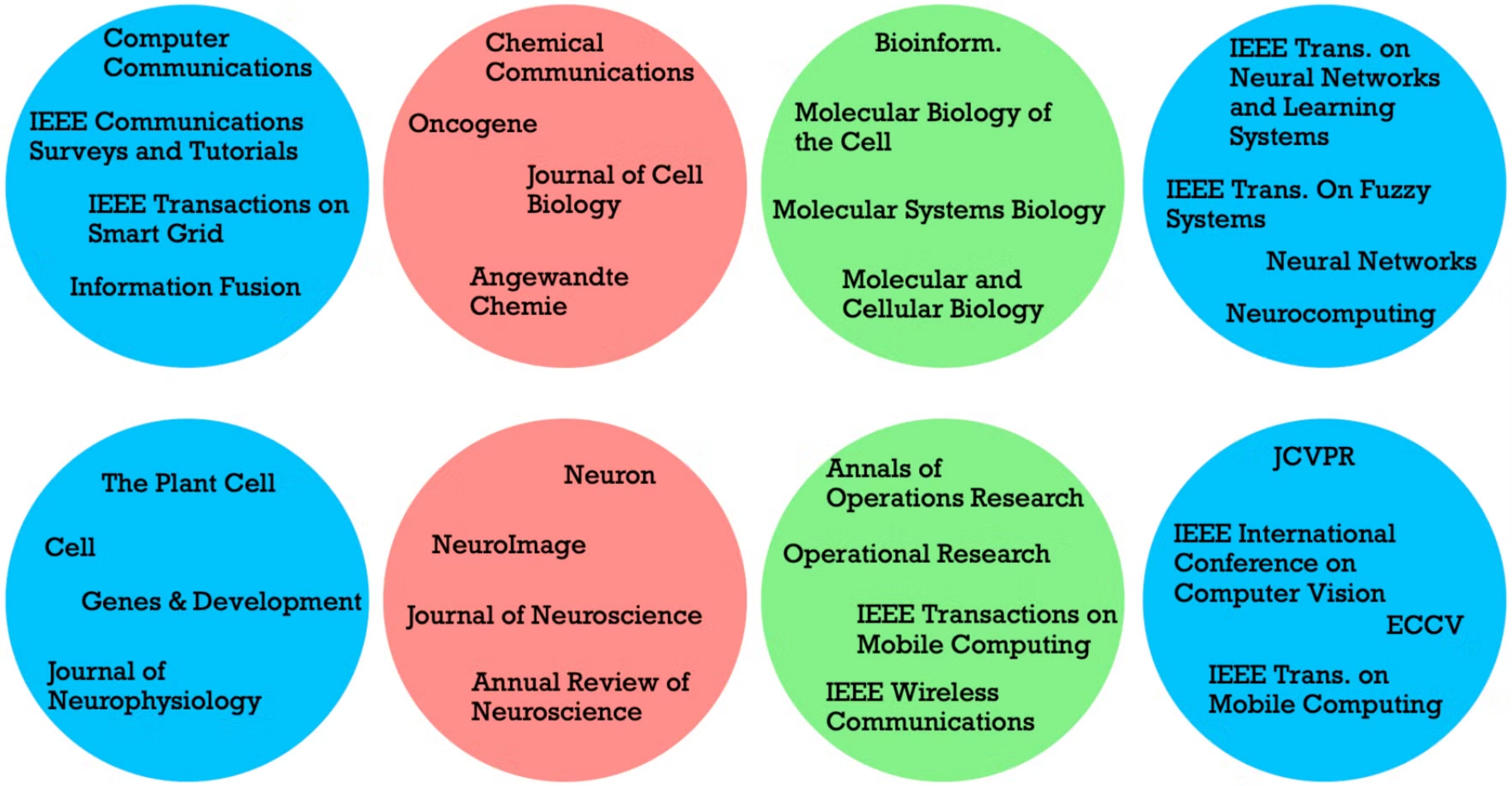
Rollcall: Swing Votes



Journal Communities

1. Vertices: journals
2. Timestep: year
3. Edges with weights: number of citations between journals

Journal Communities



Social Datasets

1. Facebook communities
2. Reddit communities

Conclusions & Future Work

Postscript: Industry Mathematics

Thank you