A Novel Method for Vertex Clustering in Dynamic Networks

Joint Mathematics Meetings 2024 • Spectra Special Session
January 5, 2024 • 11:30 AM • San Francisco, CA

Devavrat Vivek Dabke
Level Ventures

Olga Dorabiala
University of Washington

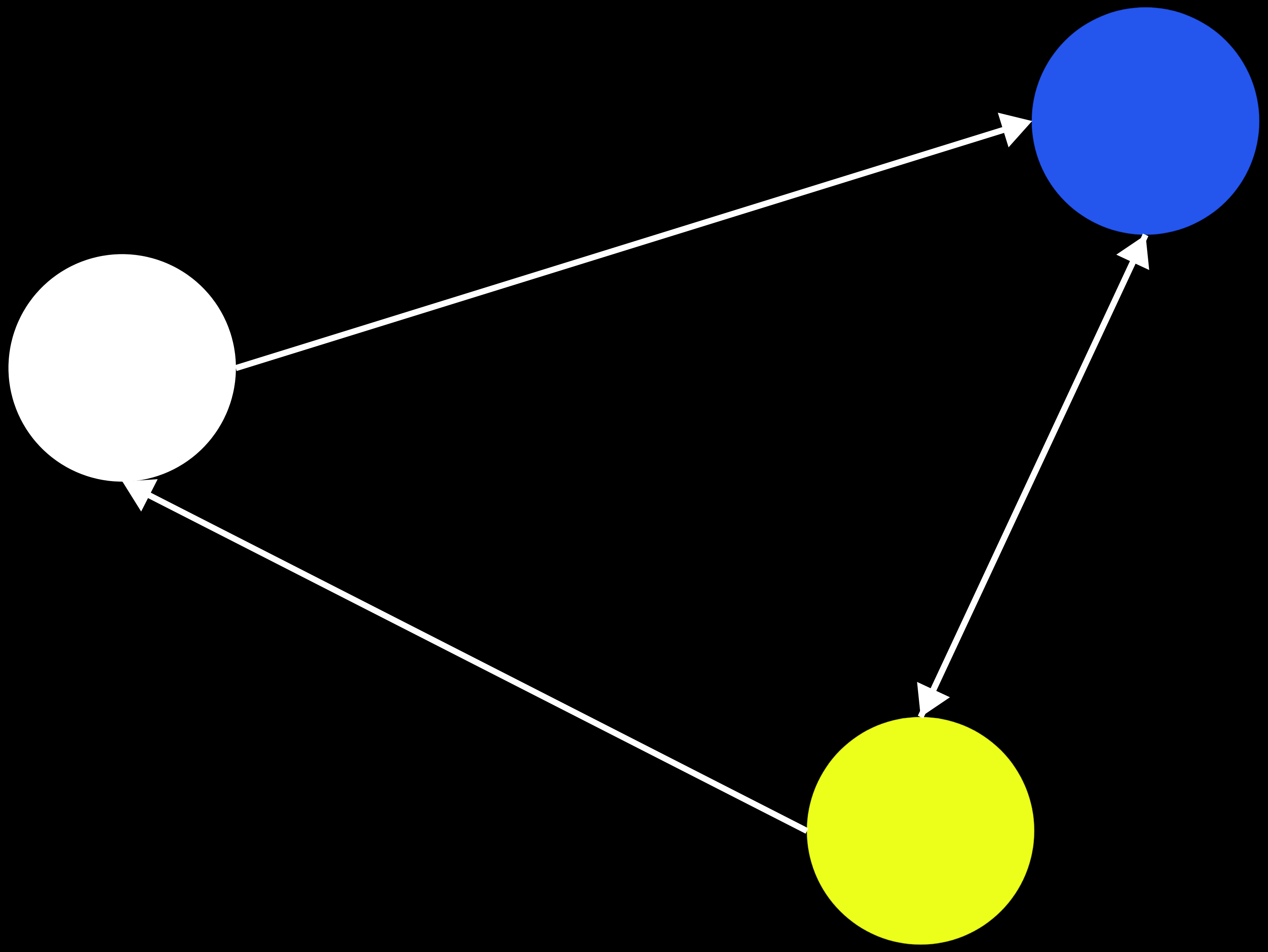
Acknowledgements

This work is based on a publication with the same name published in **2023** in *12th International Conference on* **Complex Networks** *and their Applications*.

Part of this work was conducted with support from

- the *University of Washington*
- *Princeton University*
- the *National Science Foundation*

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 (ML)

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

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

The Method

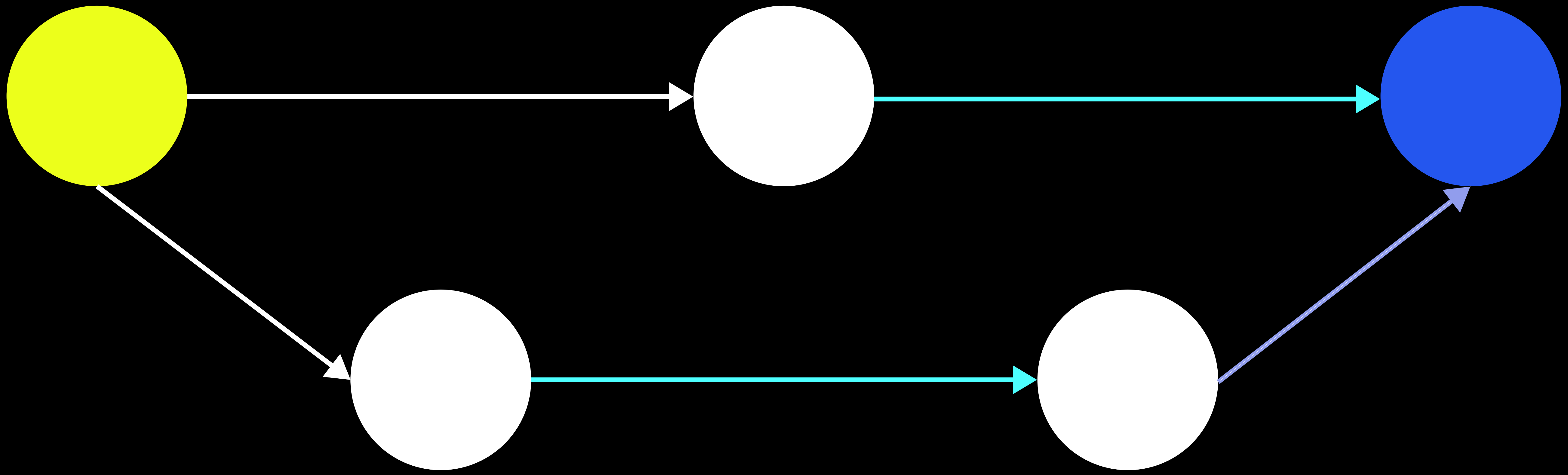
Mathematical Goal

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

partition of k elements

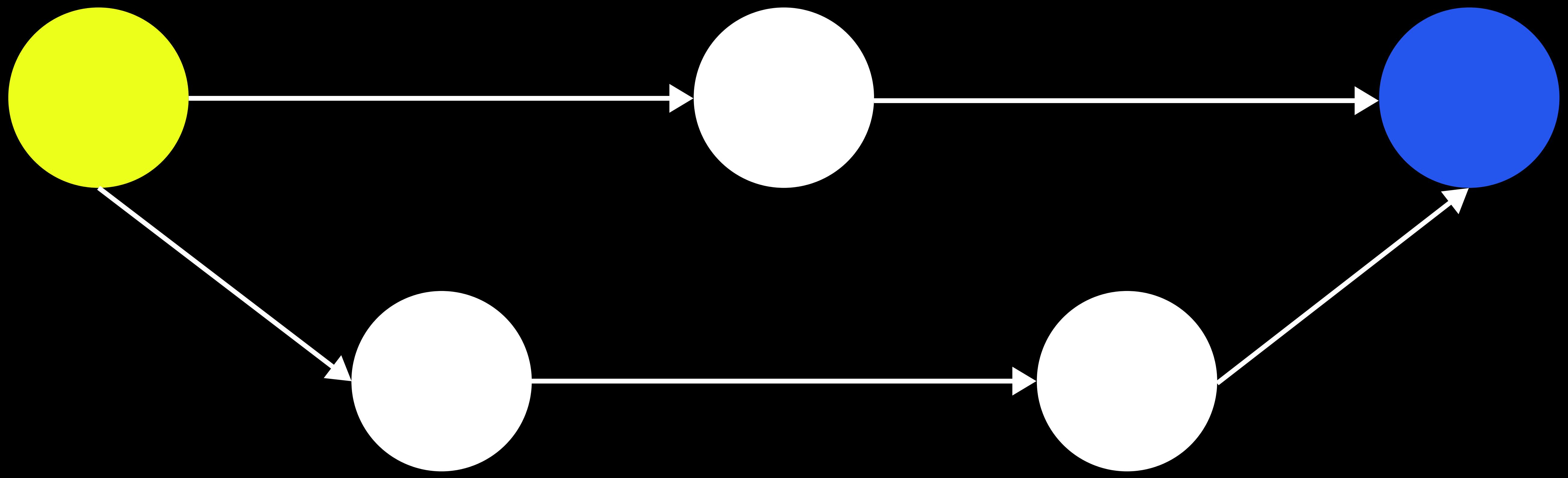
Primer: shortest journey

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.

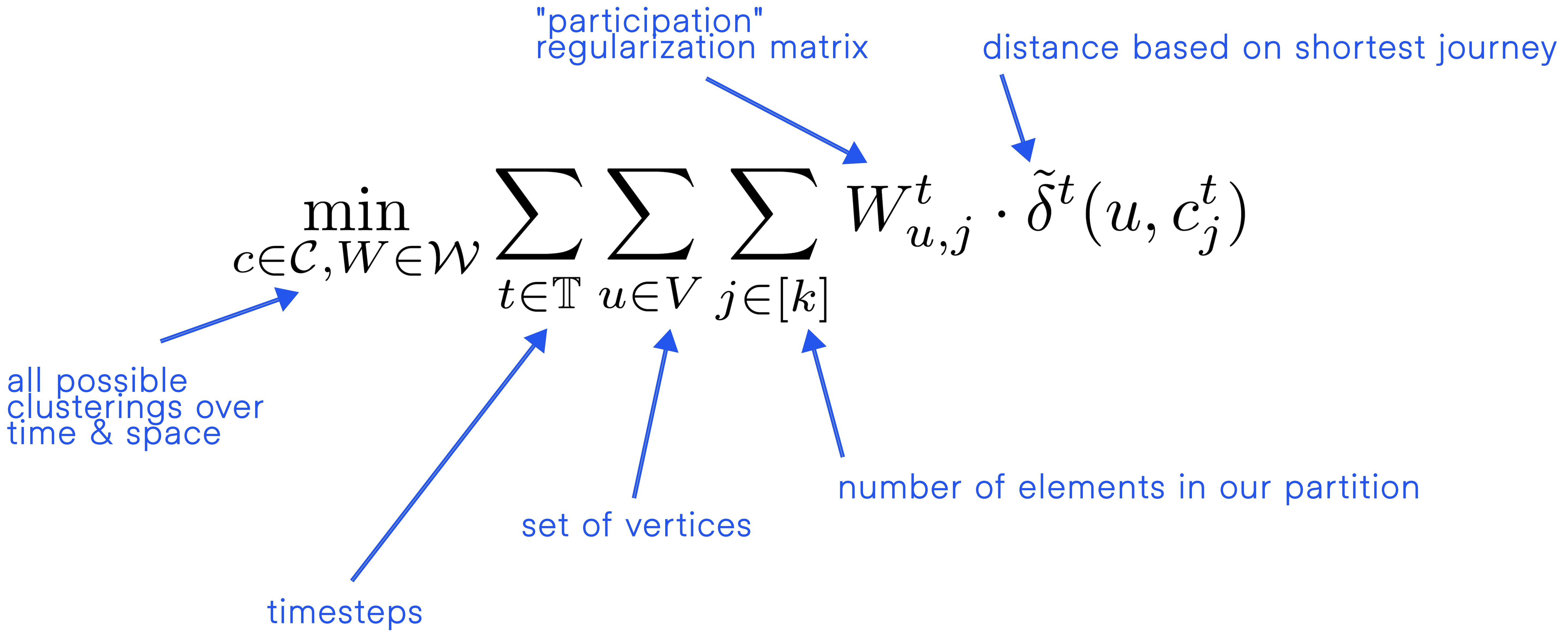


Mathematical Goal

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

good: minimizes all shortest journeys

"*k*-means" Ideal Objective



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

"participation"
regularization matrix

distance based on shortest journey

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

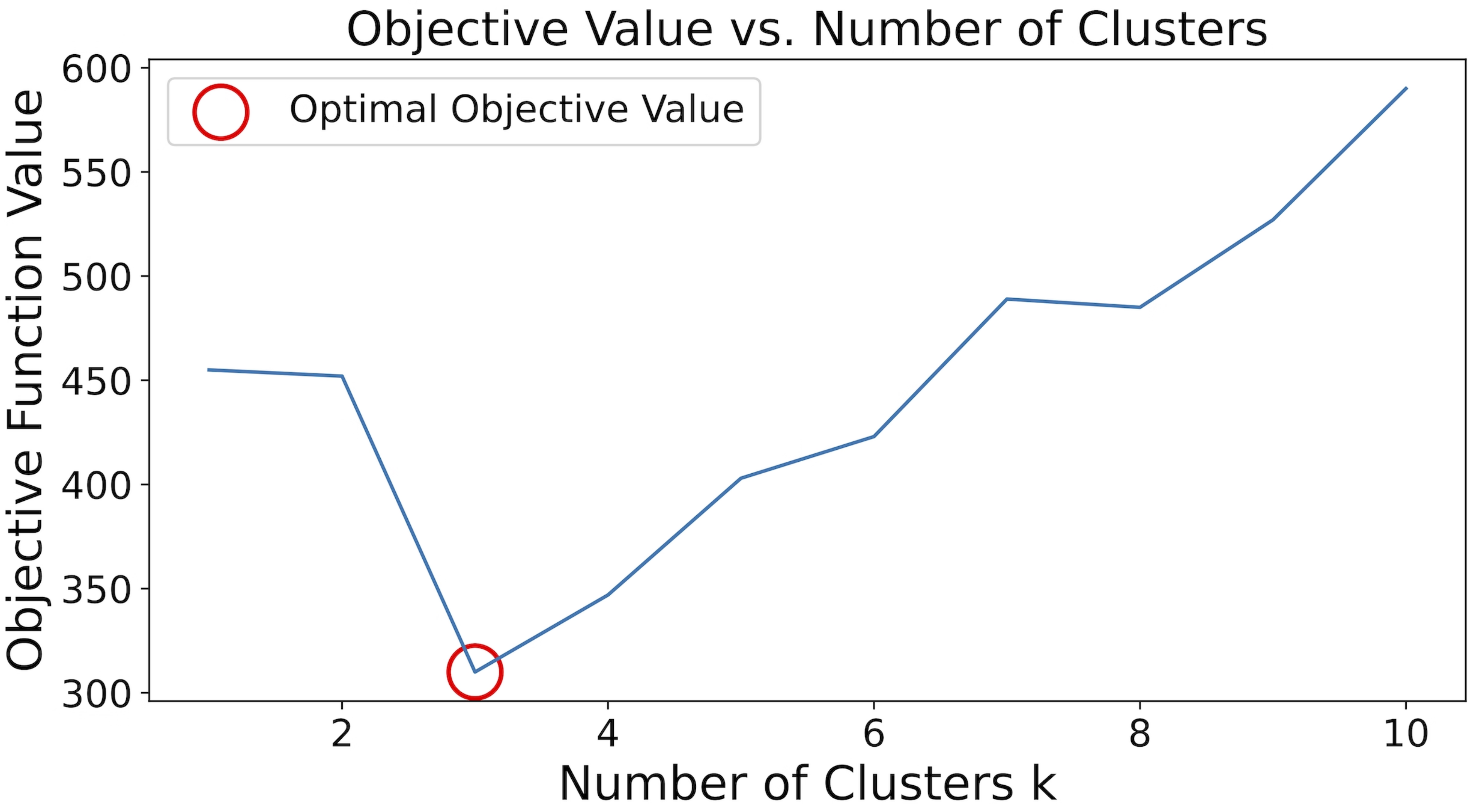
1. Solve the relaxed objective (using updated versions of classical techniques)
2. Find cluster membership of each vertex at each timestep
3. Collect information over time for each vertex
4. Use agglomerative (or other) static clustering for each vertex based on cluster membership

Results

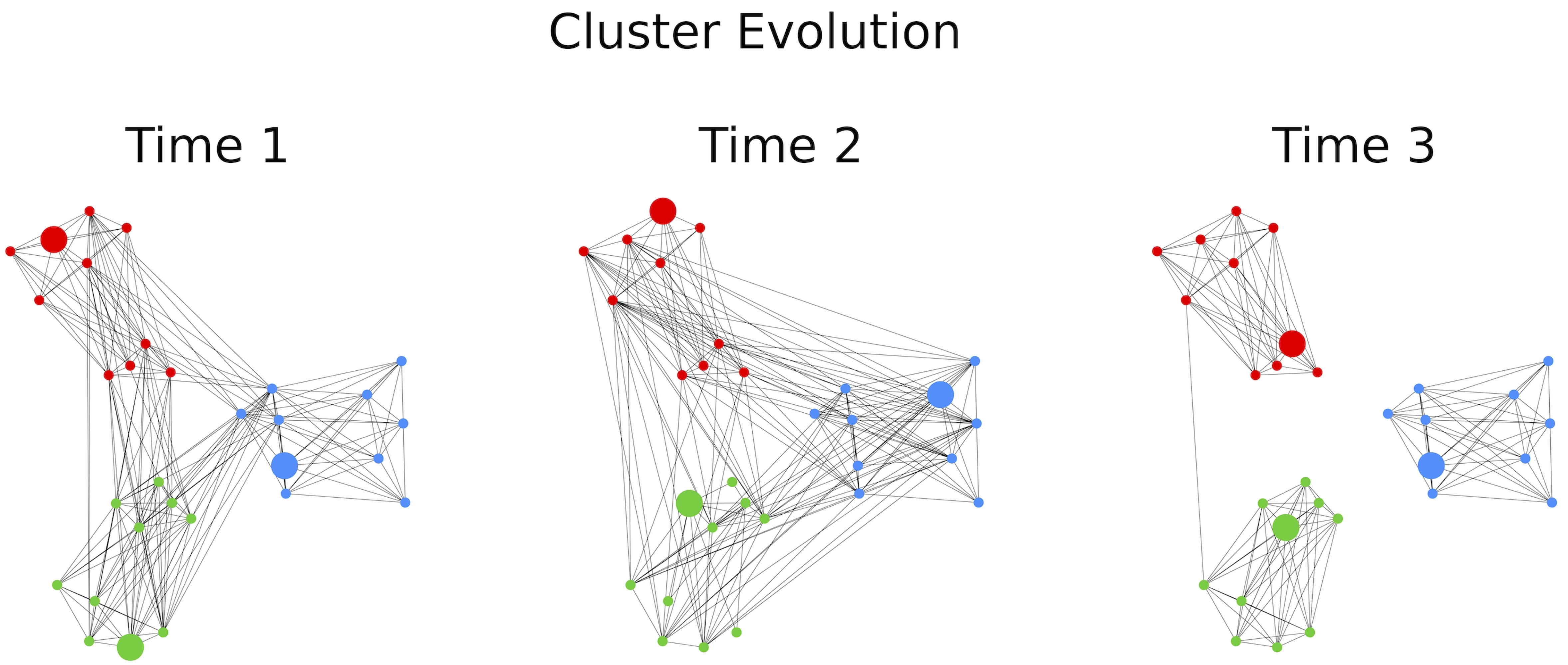
Theorem

For a holding, non-stranding dynamic graph with k connected components, the partition induced by STGkM are exactly the connected components given sufficient time.

Finding *k* with the Elbow Method



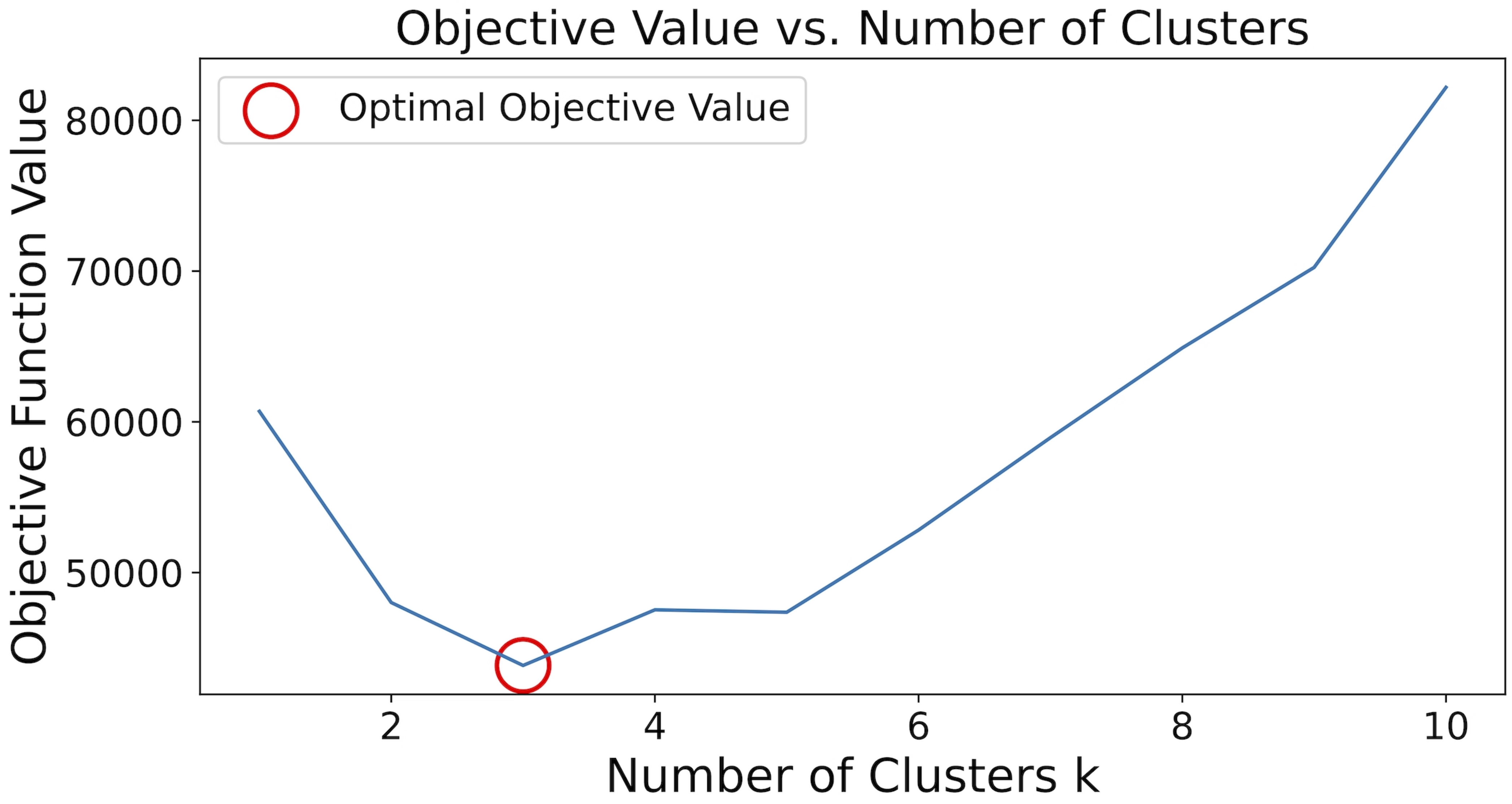
Finding *k* with the Elbow Method



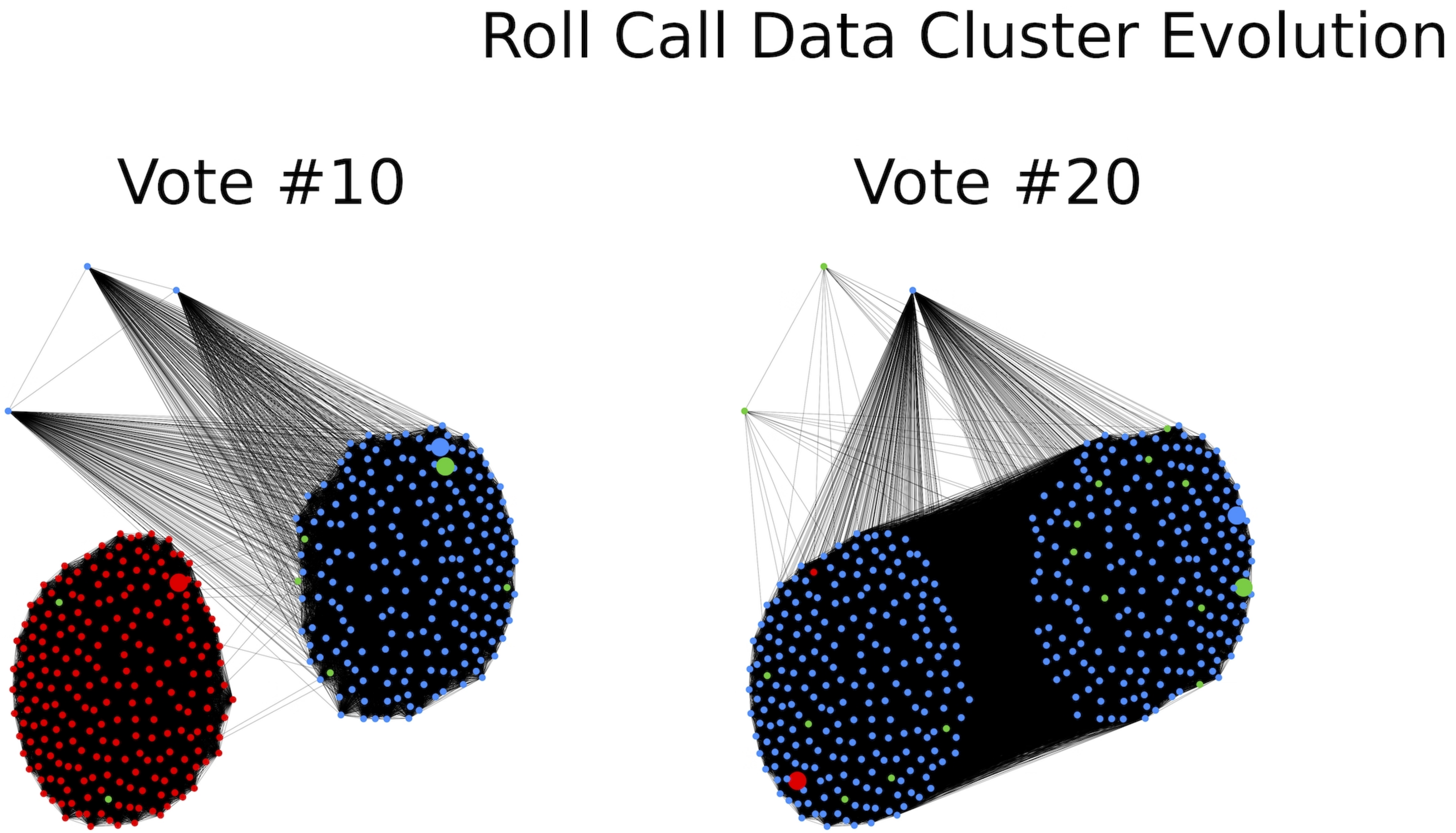
Experimental: Rollcall Votes

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

Experimental: Number of Clusters



Experimental: the Swing Votes



Conclusions & Future Work

Postscript: Industry Mathematics

Thank you