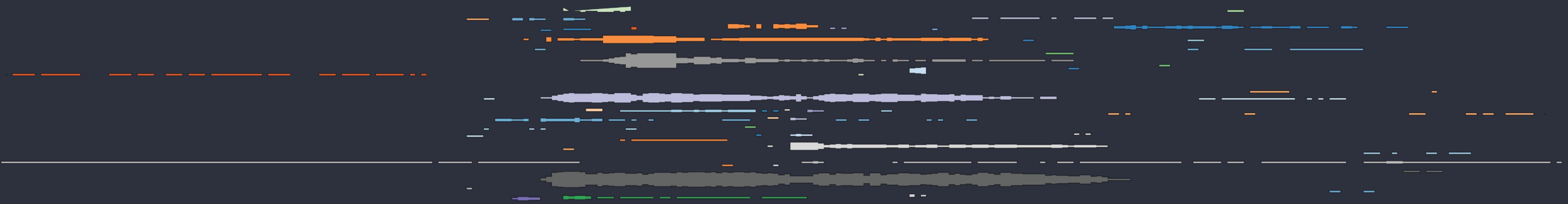


Constrained information flows in temporal networks reveal intermittent communities

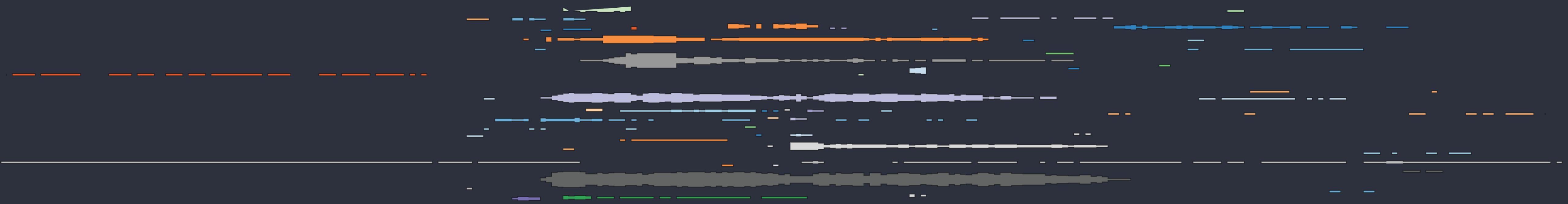


Ulf Aslak, University of Copenhagen, SODAS

Martin Rosvall, Umeå University, Department of Physics

Sune Lehmann, Technical University of Denmark, DTU Compute

A **method** for finding intermittent communities in temporal networks

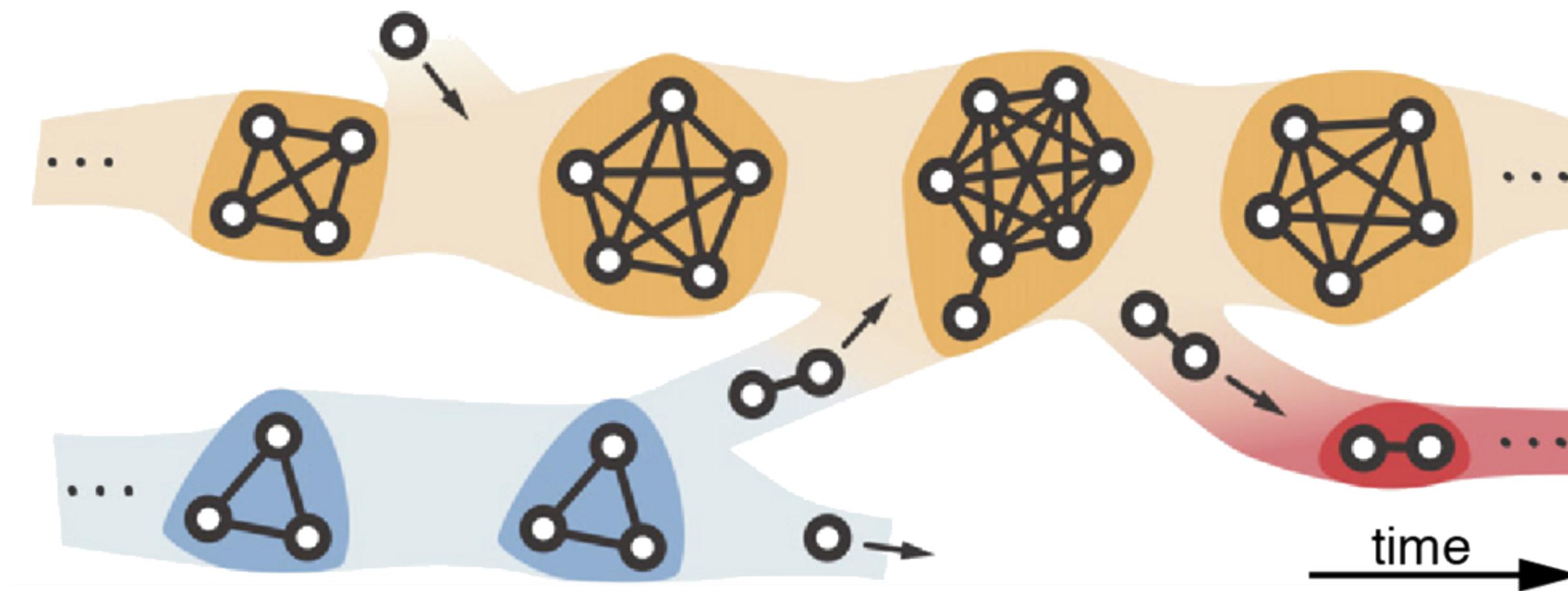


Ulf Aslak, University of Copenhagen, SODAS

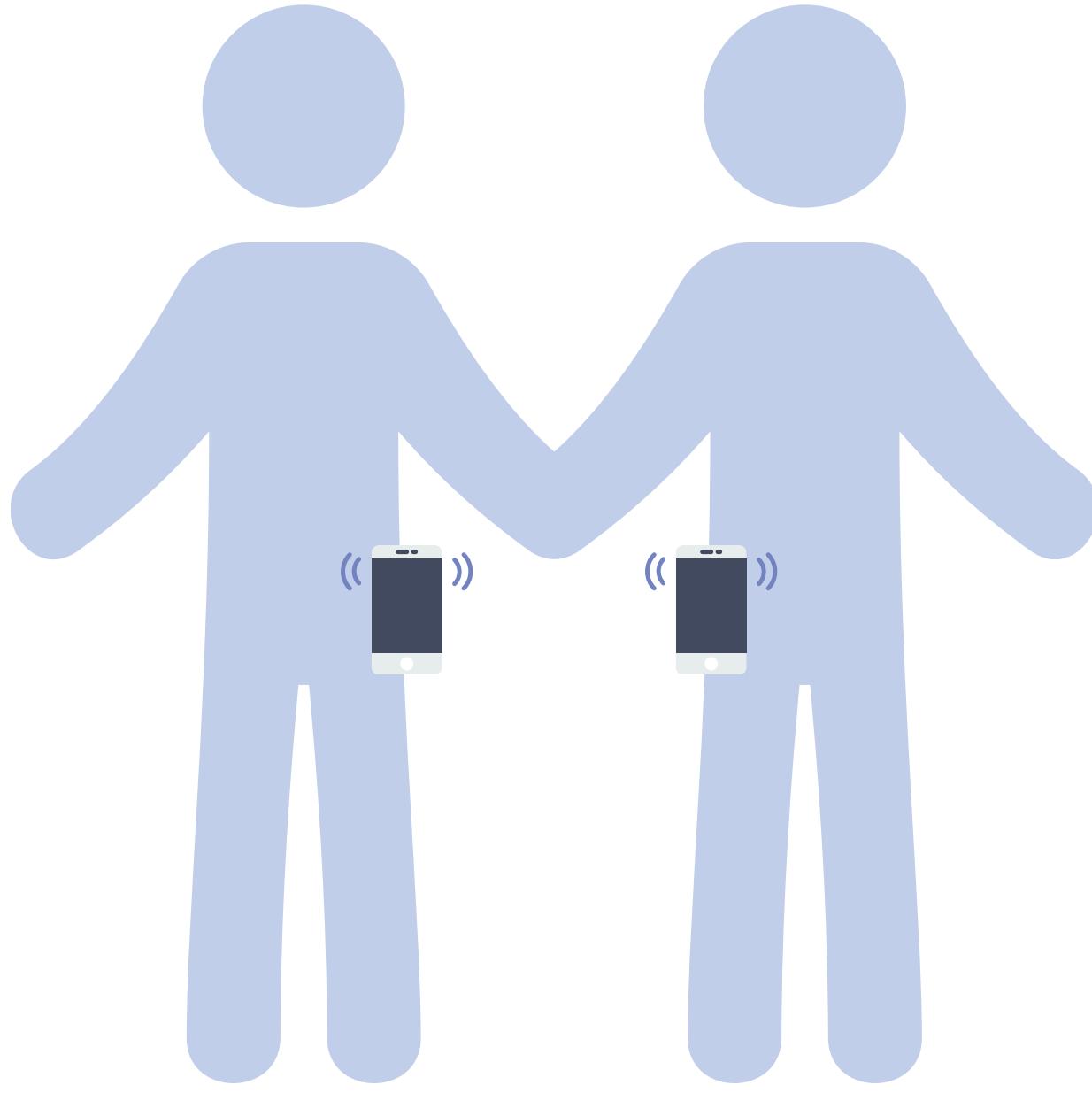
Martin Rosvall, Umeå University, Department of Physics

Sune Lehmann, Technical University of Denmark, DTU Compute

We want to understand social communities and how they change in time



To measure temporal proximity networks we can use smartphones



2 years

1000 individuals

bluetooth (5 min)

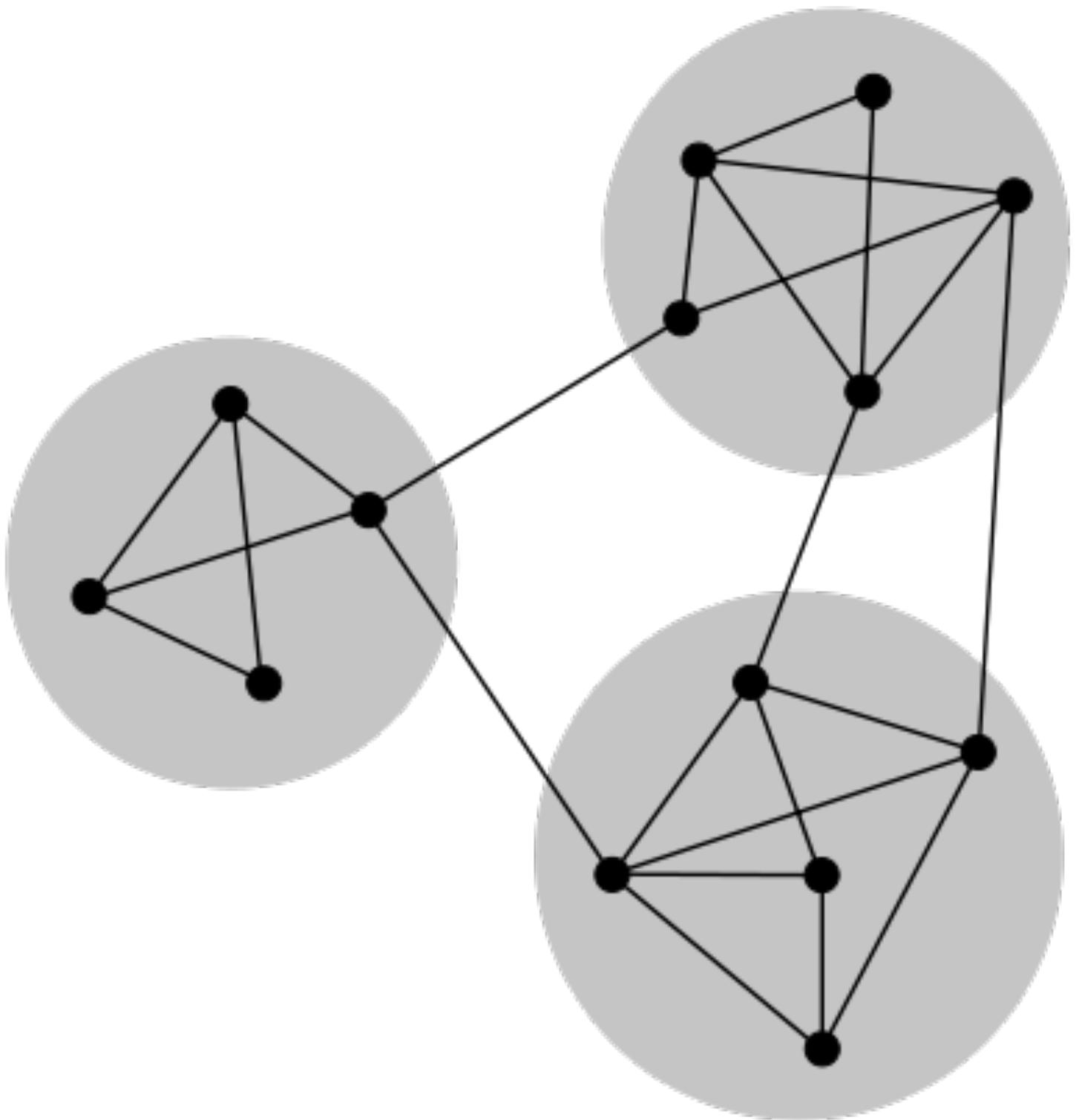
**High accuracy
Frequent sampling**

**“Not too bad” population
Dense connections**



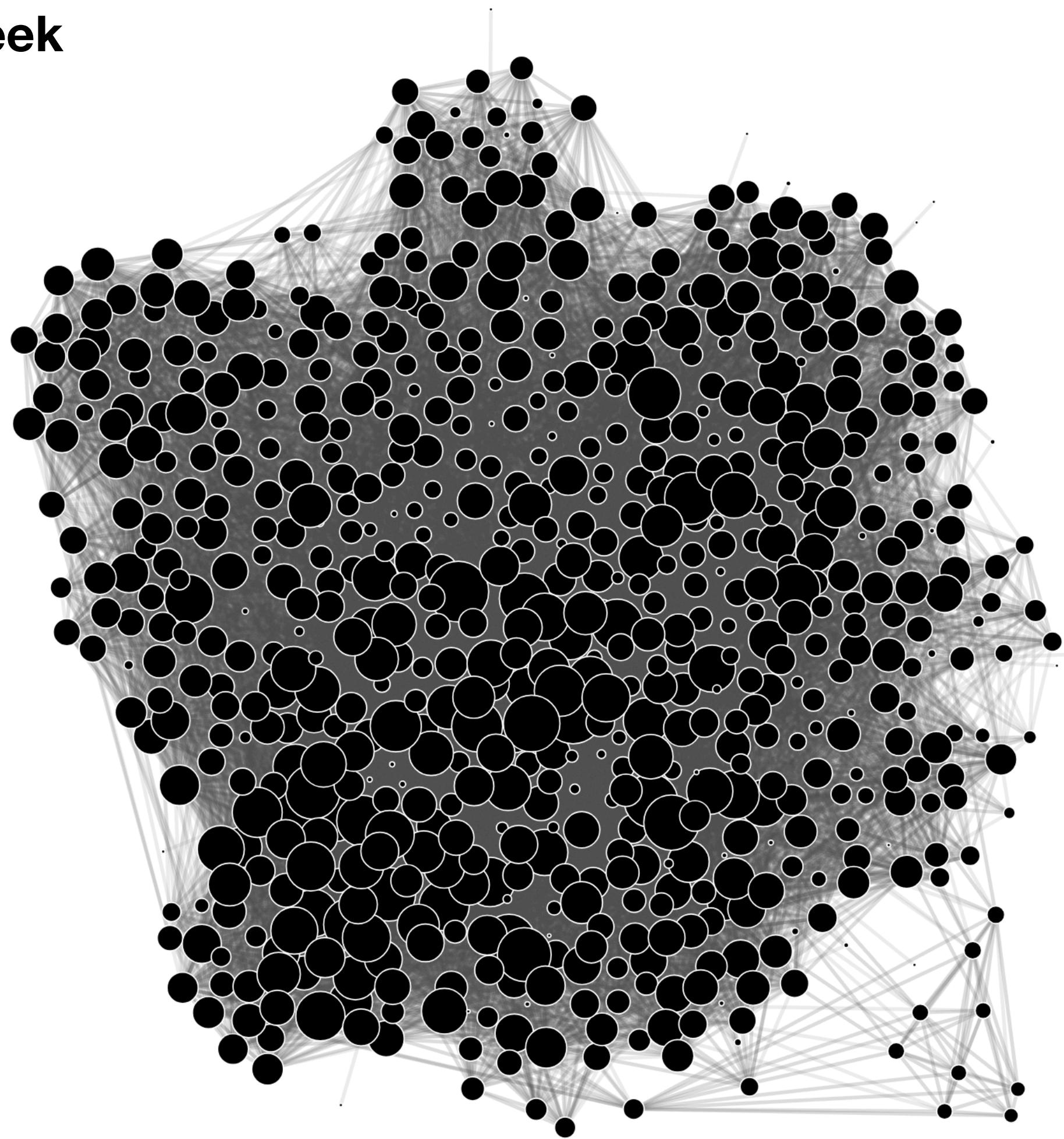


Can we expect beautiful community structure to emerge?



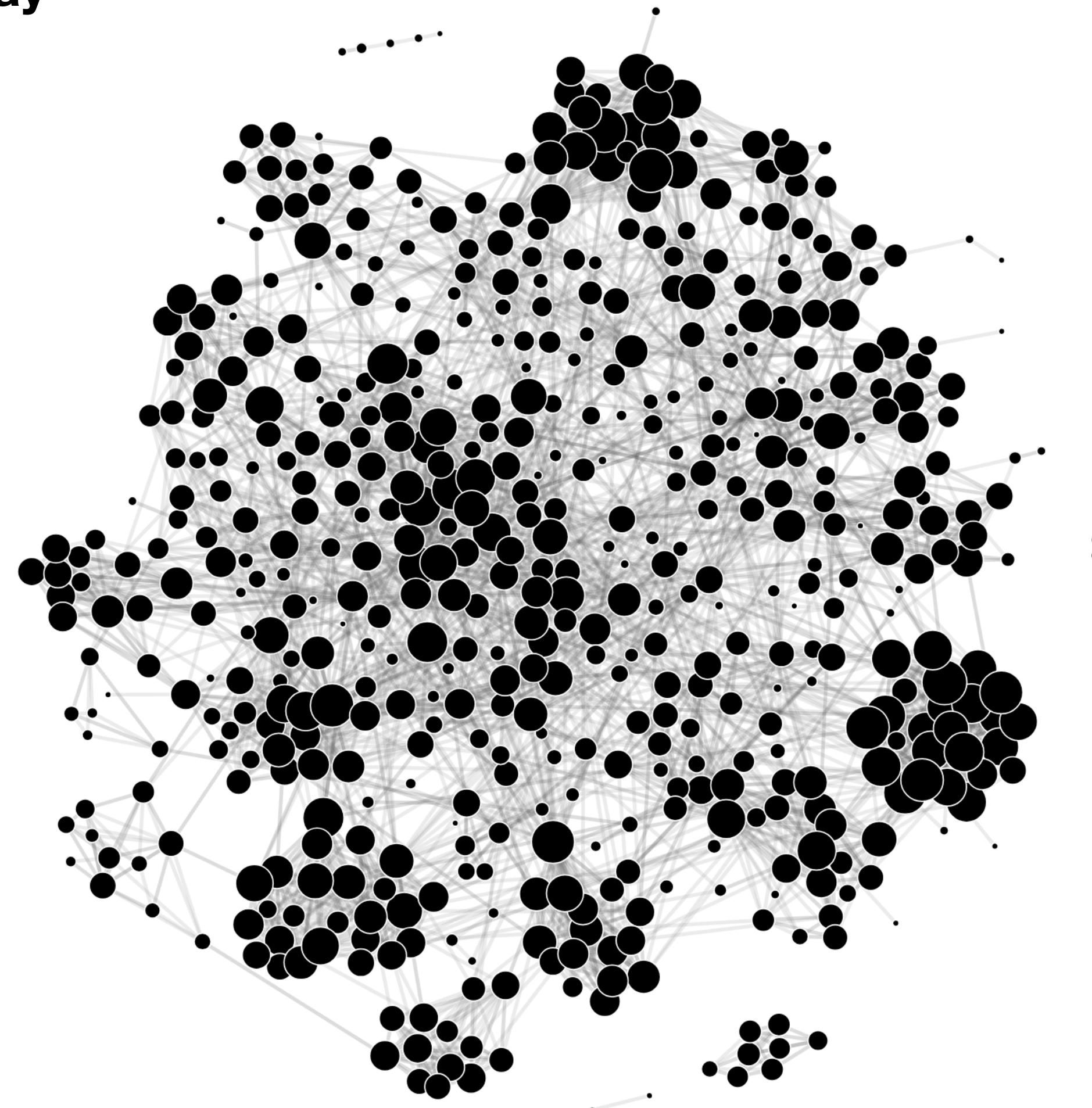
Not immediately: aggregating over time, we just get a big hairball

One week



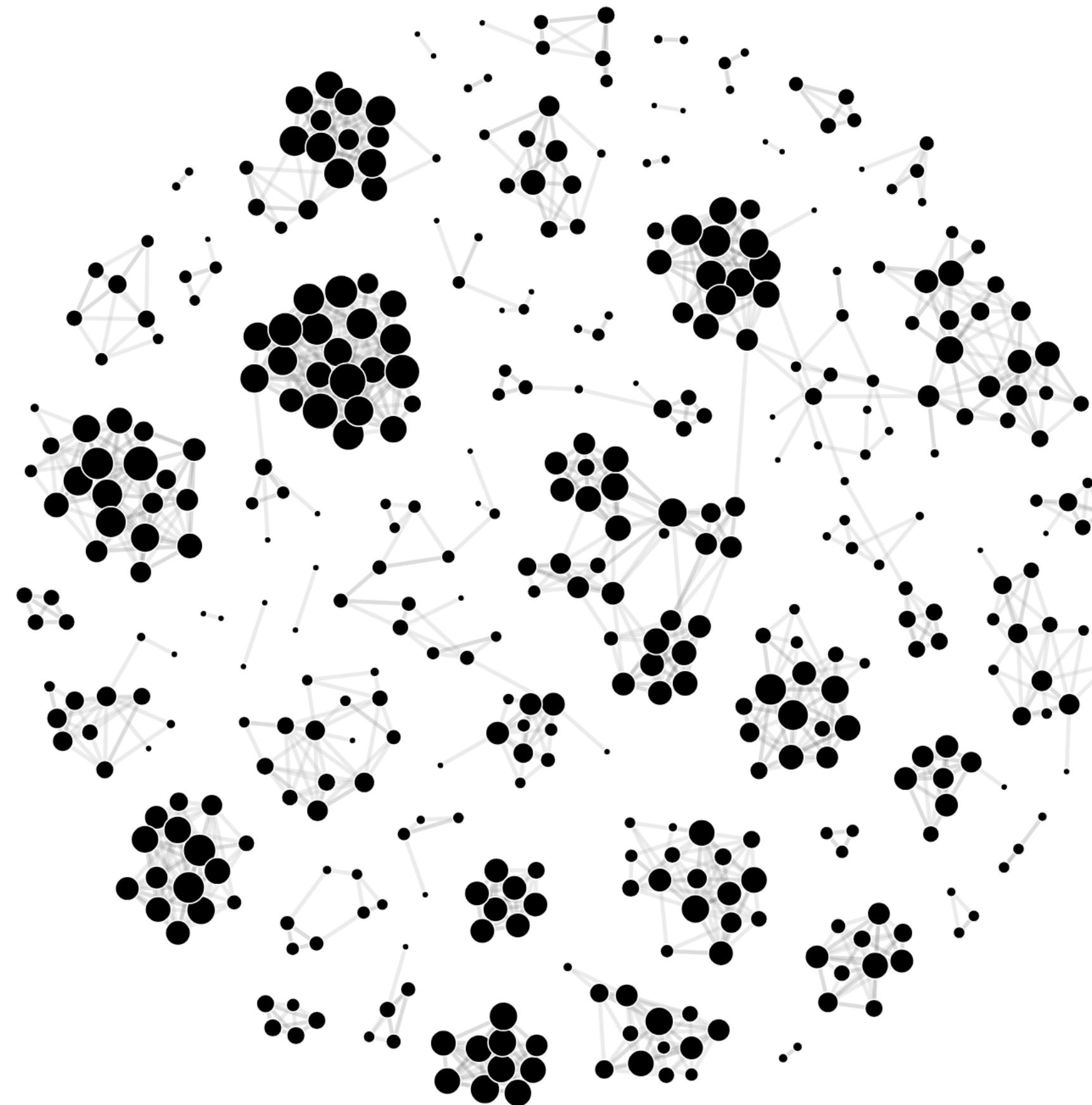
Shorter aggregates reveal more modularity

One day



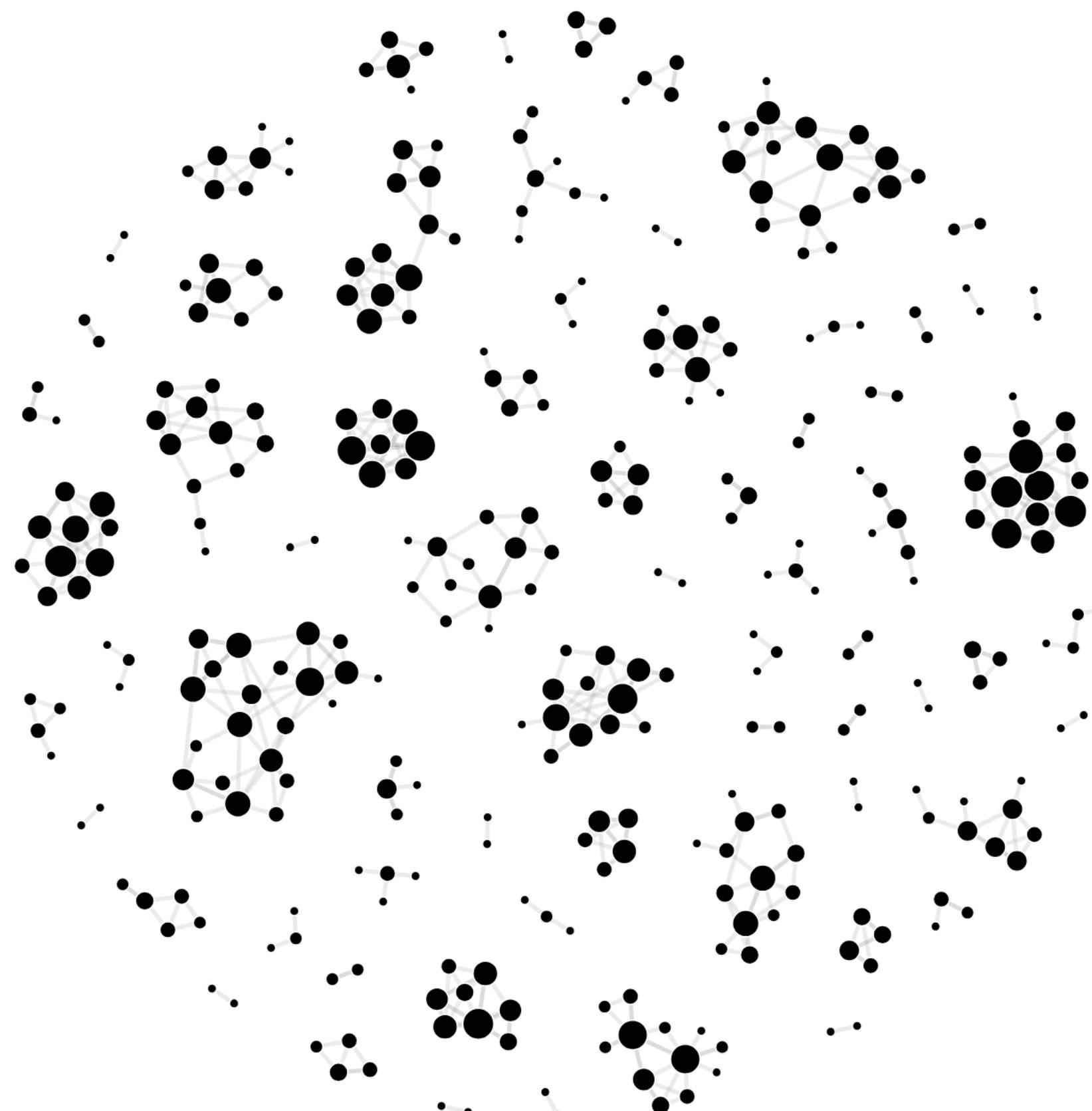
Shorter aggregates reveal more modularity

One hour

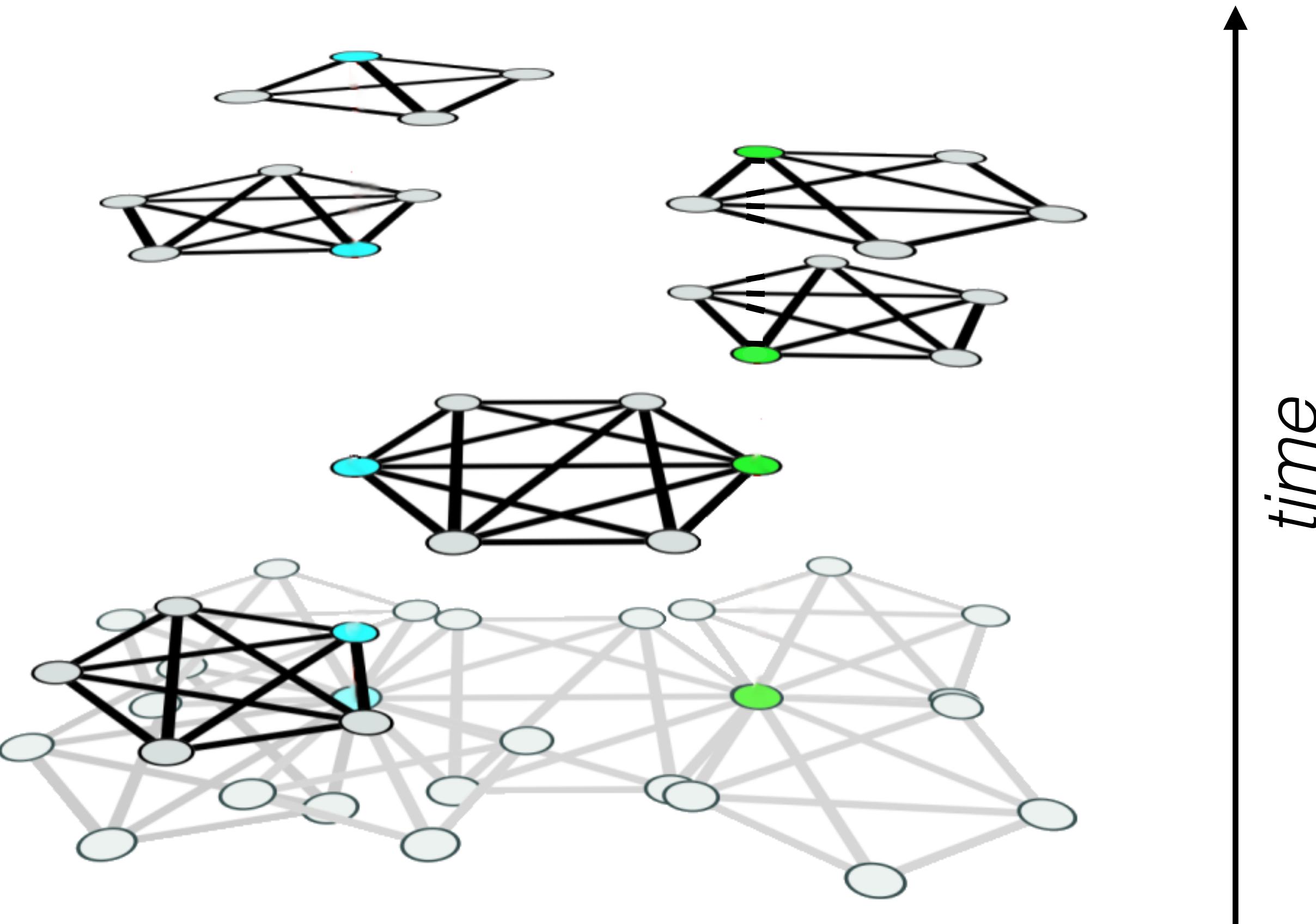


Shorter aggregates reveal more modularity

5 minutes

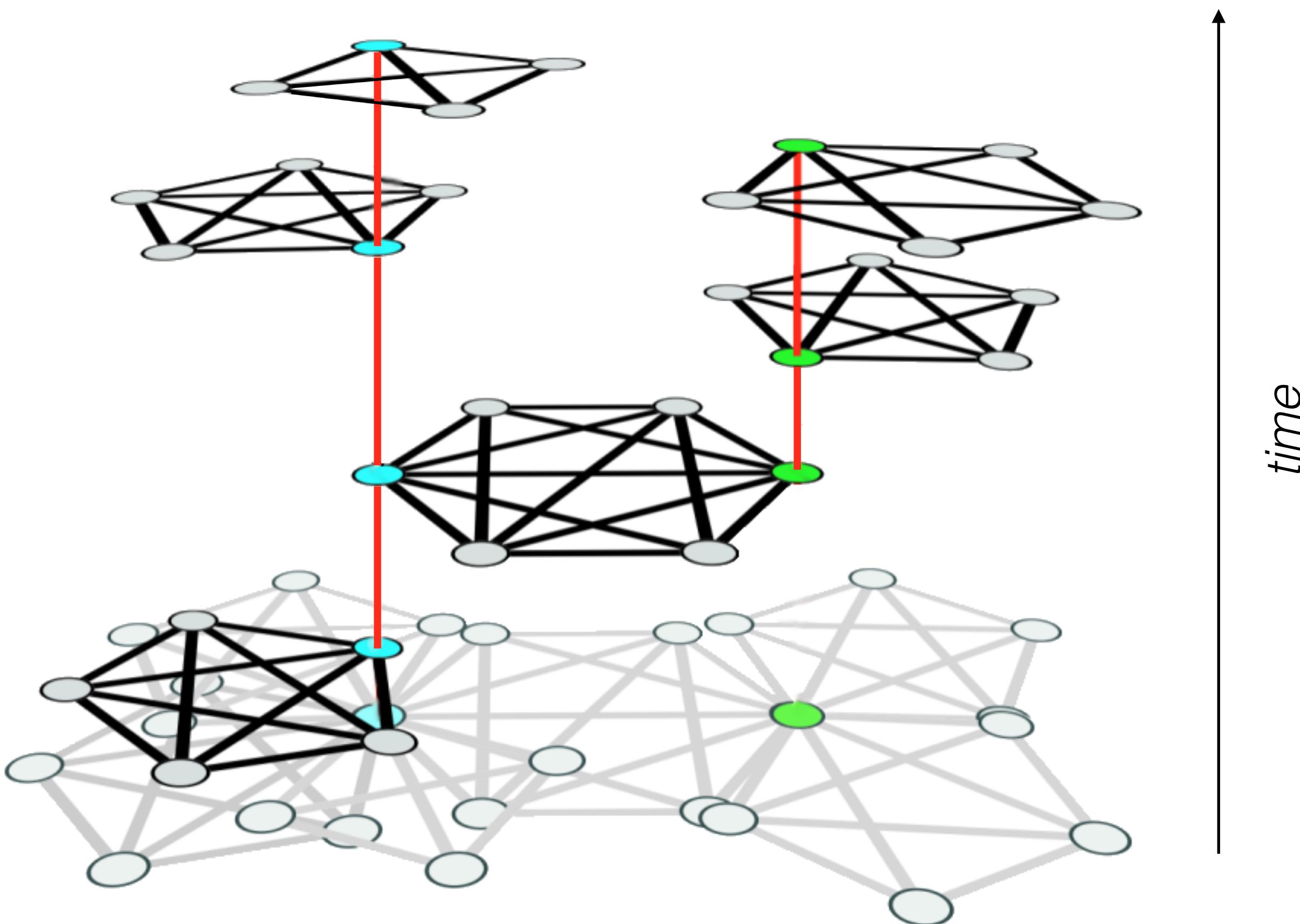


There are many overlapping communities which are intermittent in time

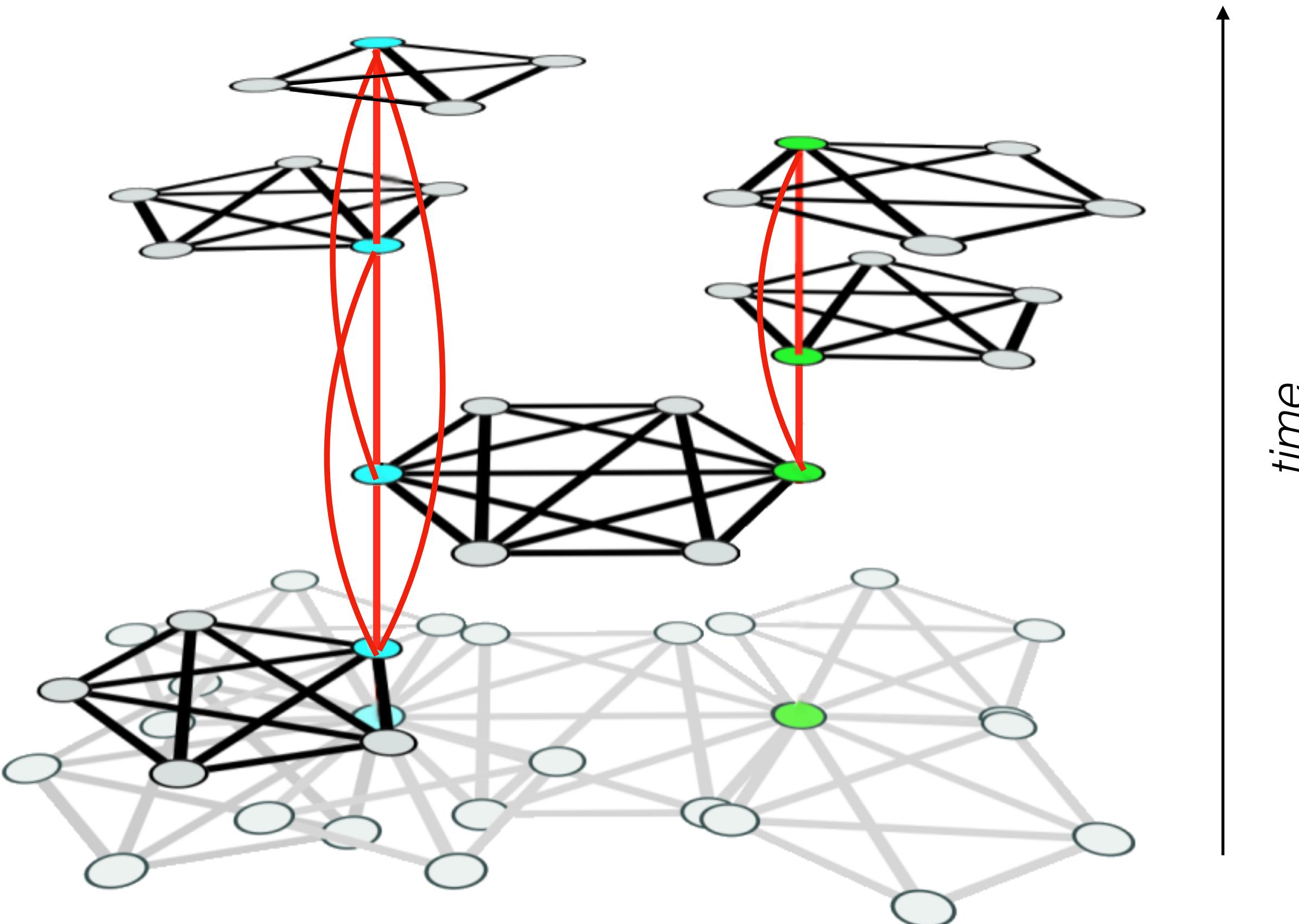


We need a way to find **overlapping intermittent** communities

Approach: Self-connect nodes through time



There are many ways to create temporal dependencies



Holme, P. (2015). Modern temporal network theory: a colloquium. *The European Physical Journal B*, 88(9), 234.

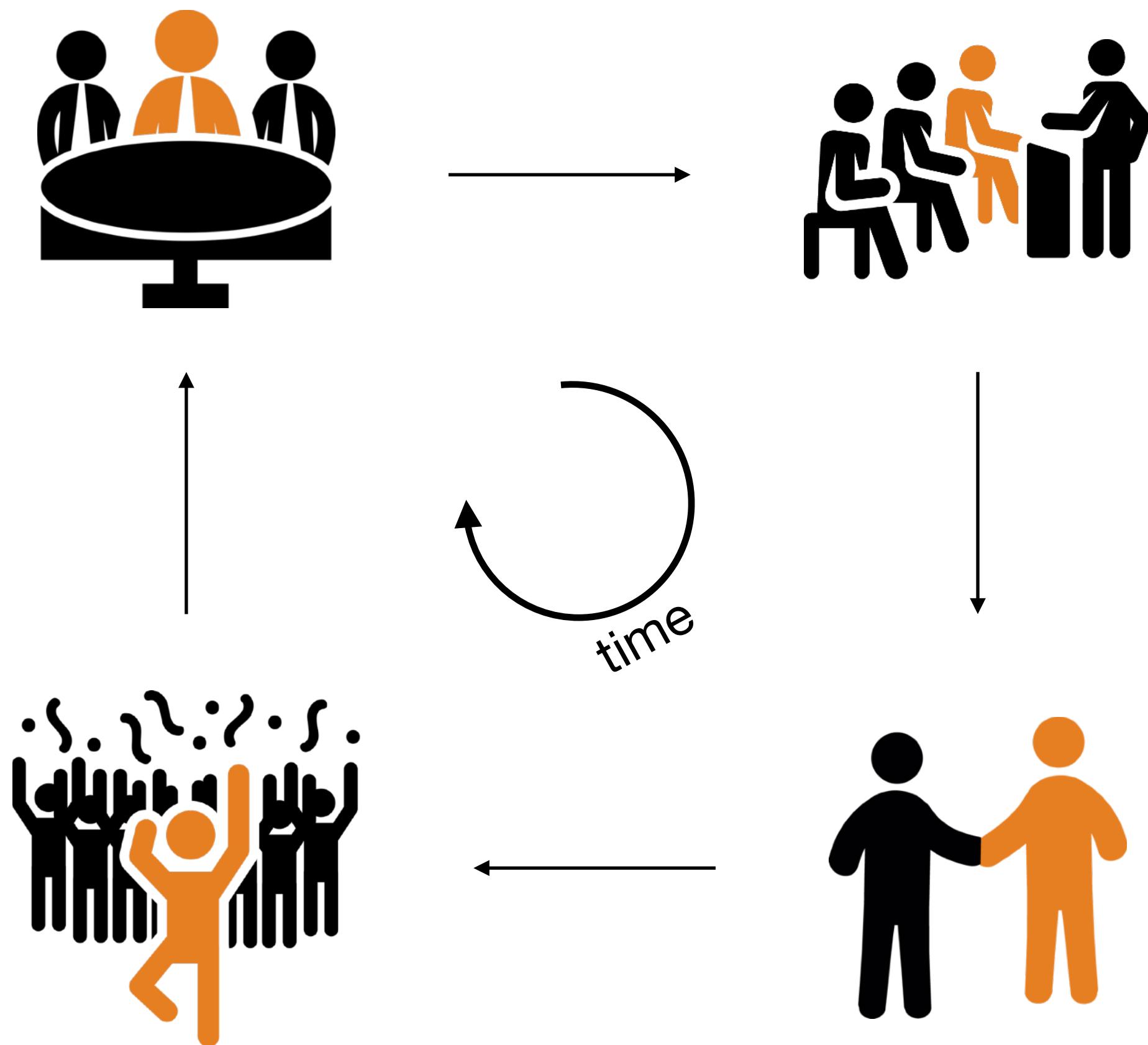
Mucha, P. J., Richardson, T., Macon, K., Porter, M. A., & Onnela, J. P. (2010). Community structure in time-dependent, multiscale, and multiplex networks. *science*, 328(5980), 876-878.

Petri, G., & Expert, P. (2014). Temporal stability of network partitions. *Physical Review E*, 90(2), 022813.

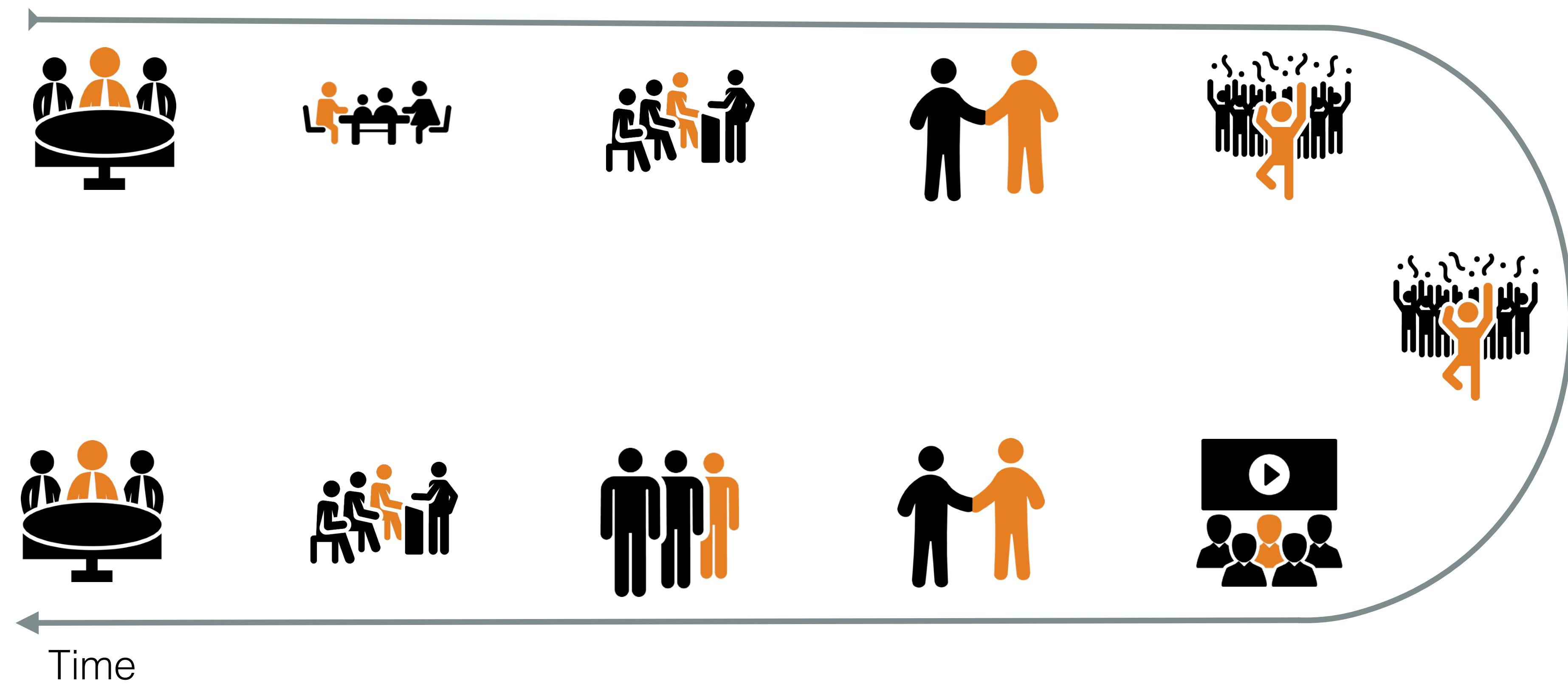
De Bacco, C., Power, E. A., Larremore, D. B., & Moore, C. (2017). Community detection, link prediction, and layer interdependence in multilayer networks. *Physical Review E*, 95(4), 042317.

We want to create inter-temporal dependencies that are **meaningful** and **interpretable**

We suggest a principle for defining meaningful and interpretable temporal dependencies

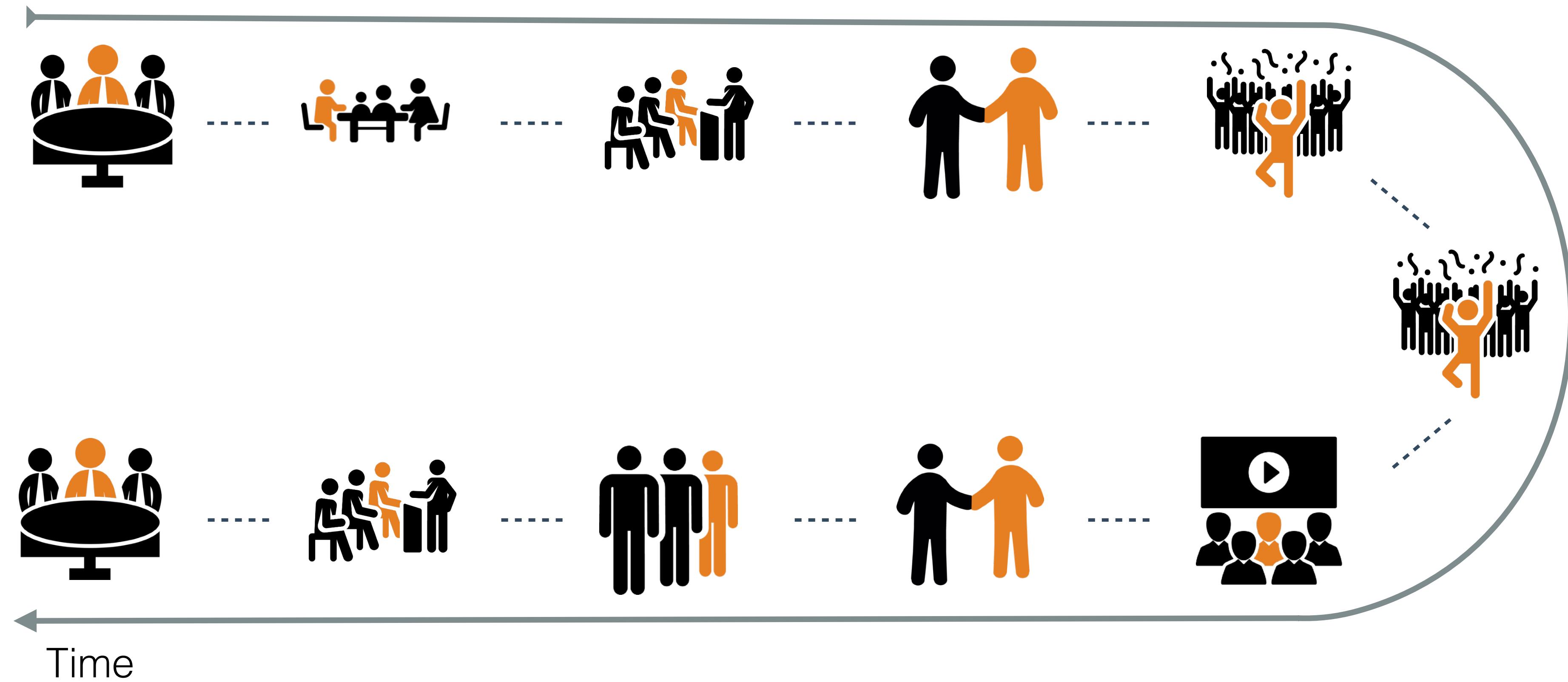


We suggest a principle for defining meaningful and interpretable temporal dependencies



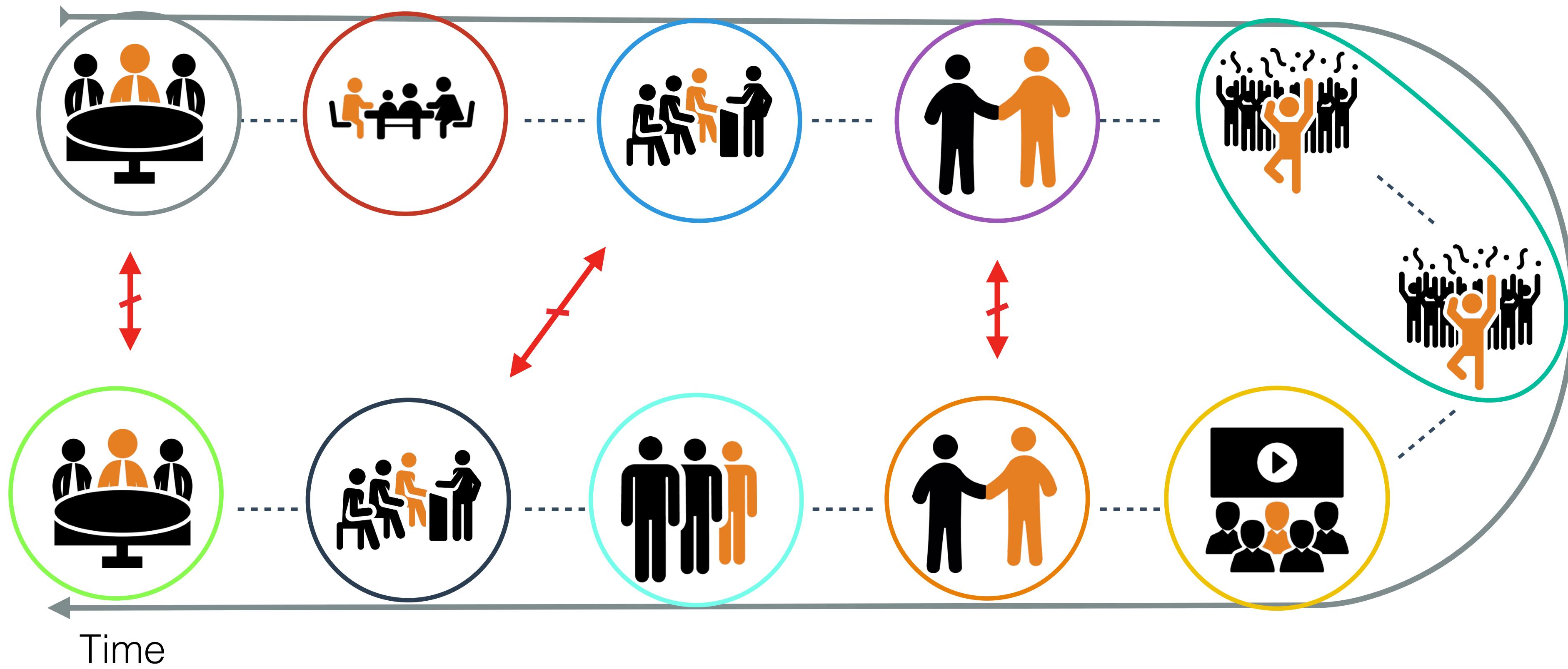
We suggest a principle for defining meaningful and interpretable temporal dependencies

- **Adjacent coupling** results in temporally continuous communities



We suggest a principle for defining meaningful and interpretable temporal dependencies

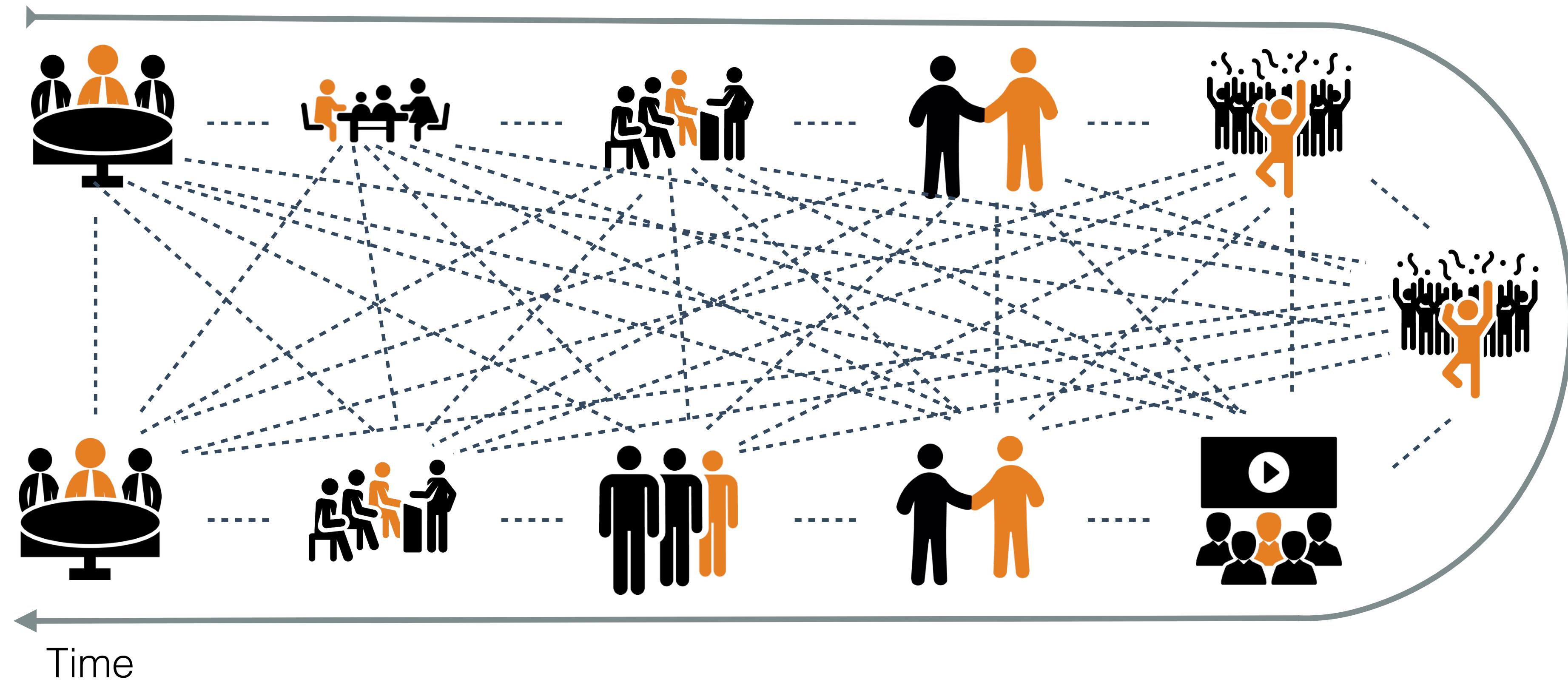
- **Adjacent coupling** results in temporally continuous communities



Intermittence not captured

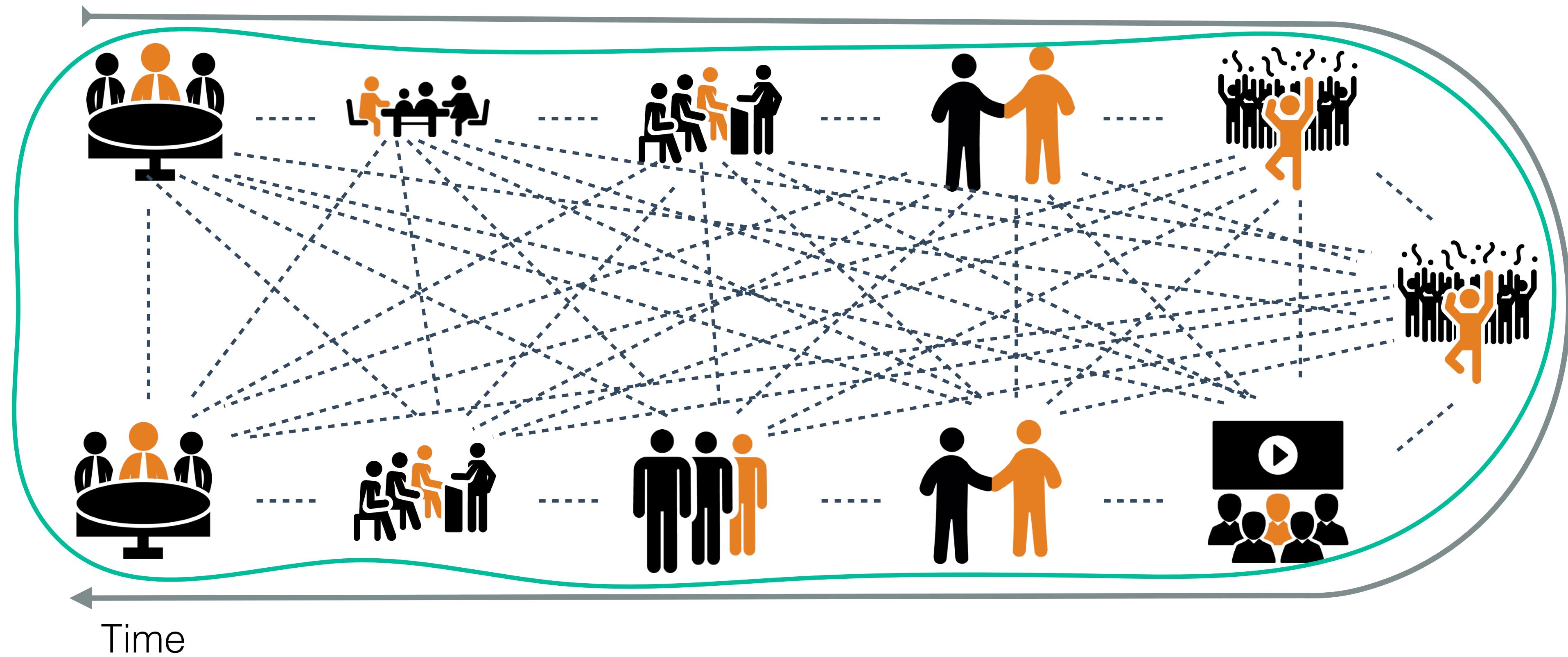
We suggest a principle for defining meaningful and interpretable temporal dependencies

- **Full coupling** merges overlapping communities



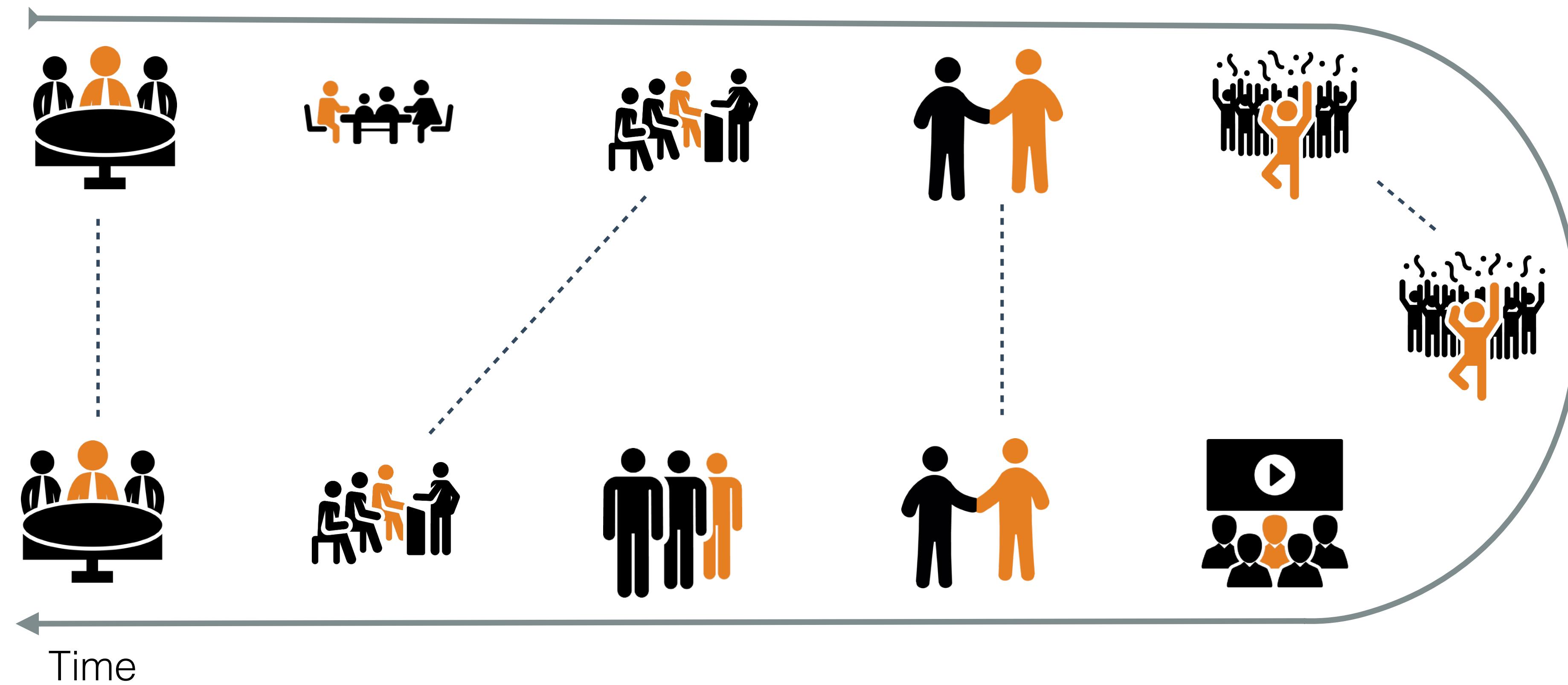
We suggest a principle for defining meaningful and interpretable temporal dependencies

- **Full coupling** merges overlapping communities



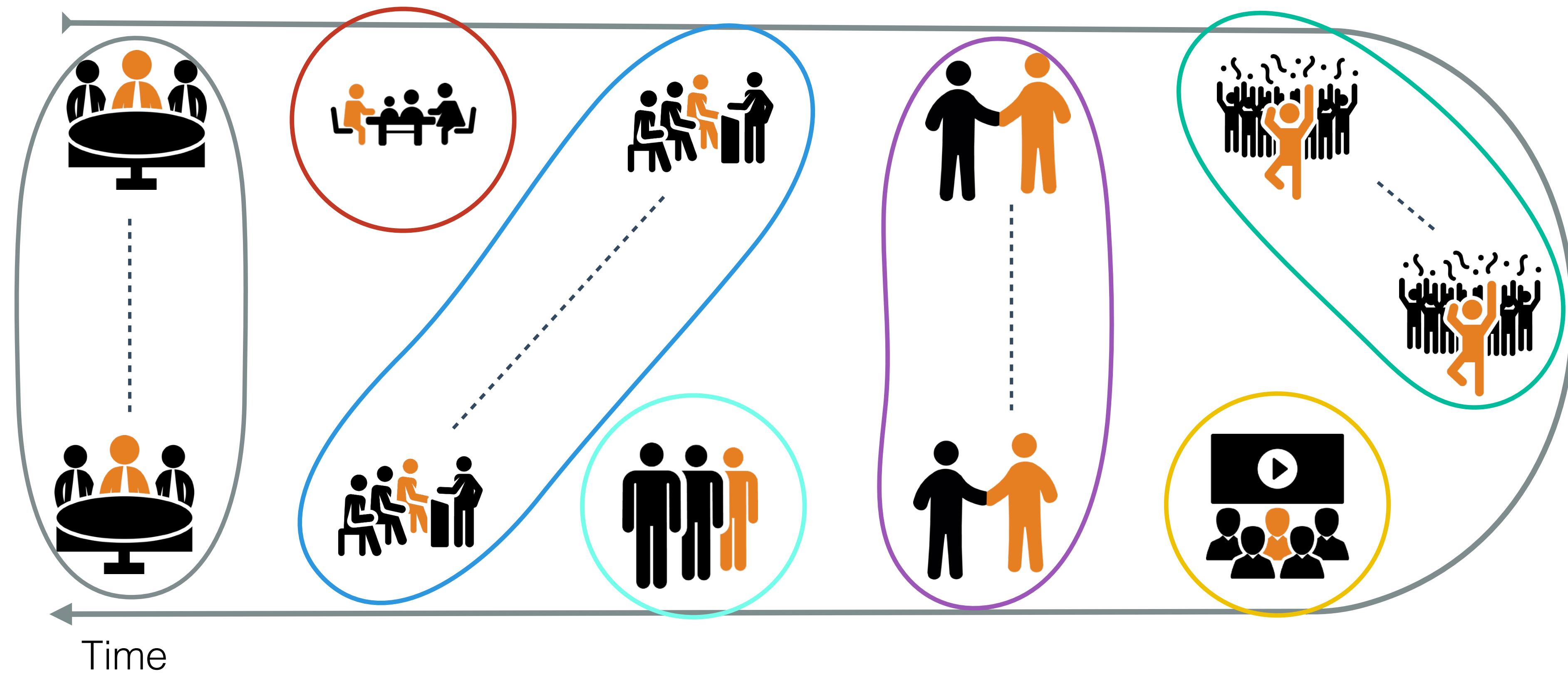
We suggest a principle for defining meaningful and interpretable temporal dependencies

- We suggest: **neighborhood flow coupling**. Finds intermittent and overlapping communities

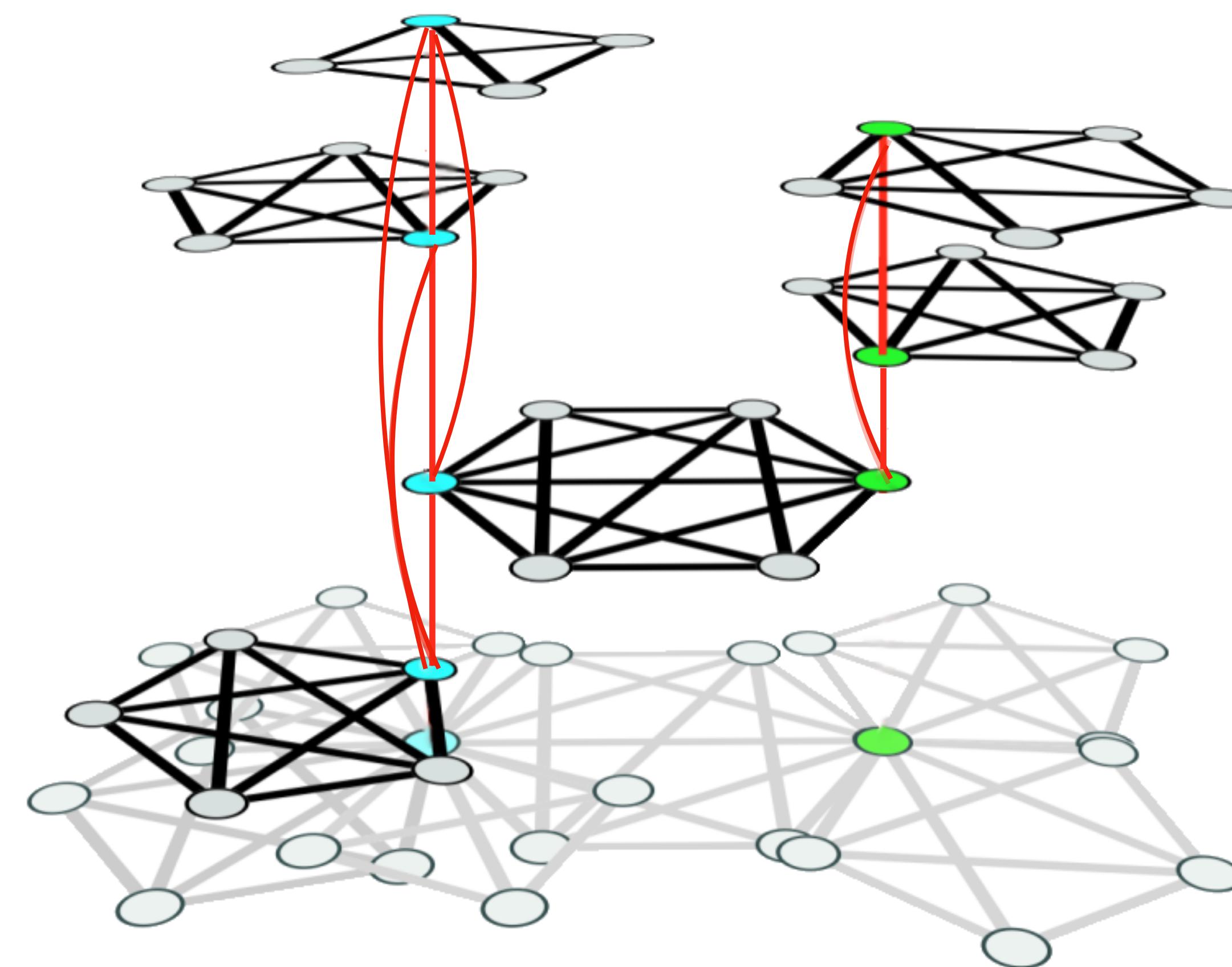


We suggest a principle for defining meaningful and interpretable temporal dependencies

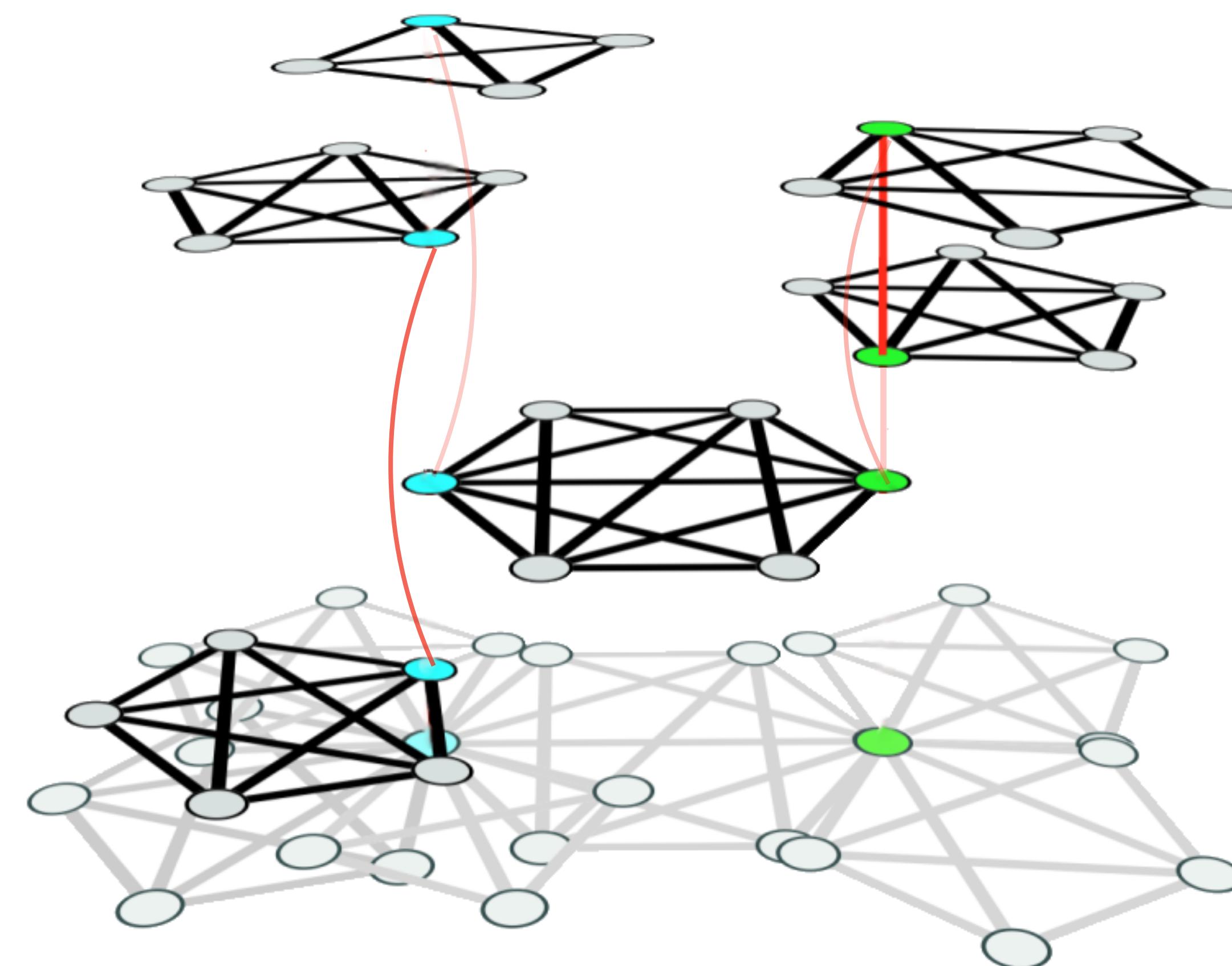
- We suggest: **neighborhood flow coupling**. Finds intermittent and overlapping communities



Neighborhood flow coupling: Couple two states with a strength that is proportional to how similar their surrounding network flows are



Neighborhood flow coupling: Couple two states with a strength that is proportional to how similar their surrounding network flows are

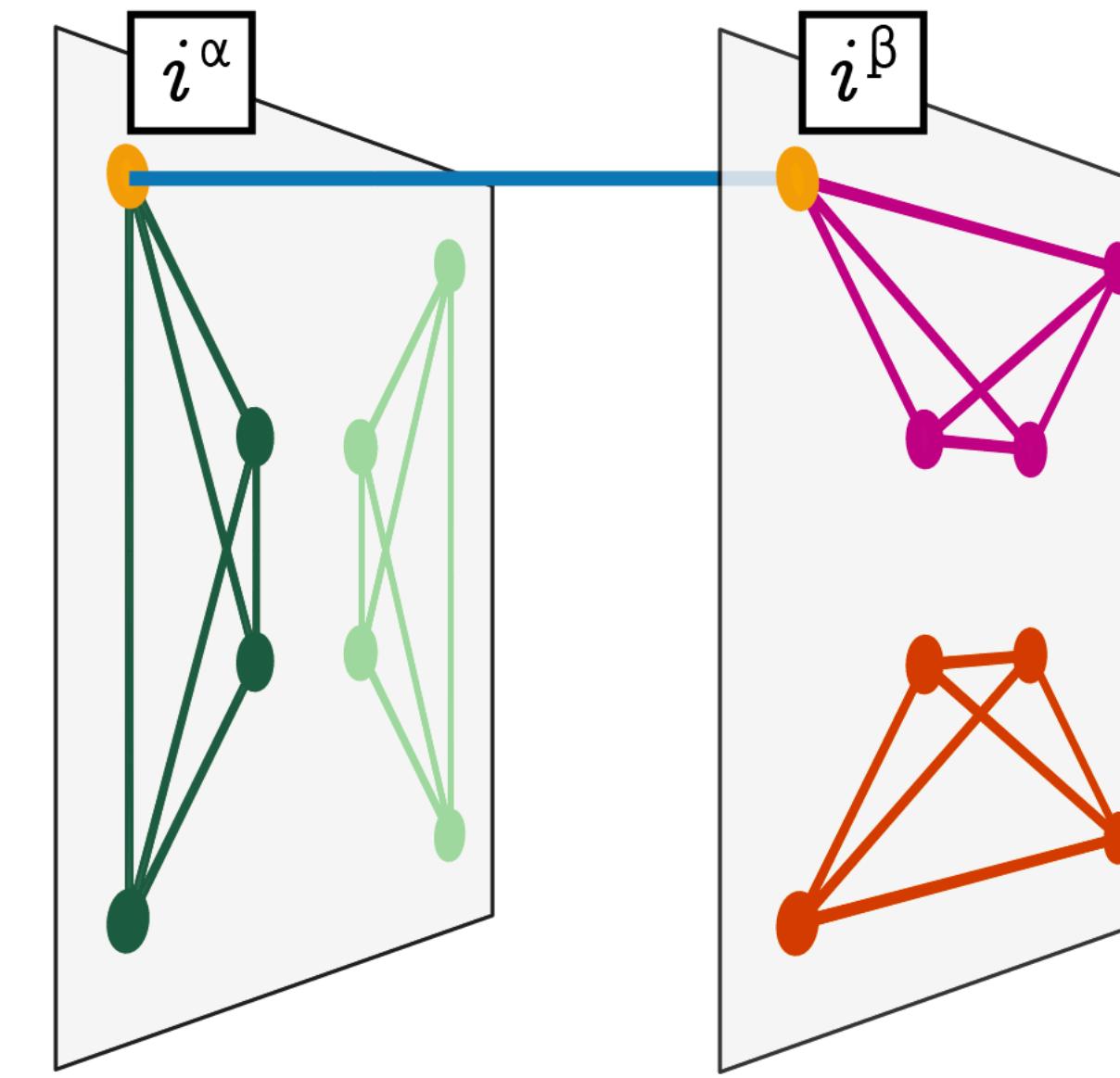


We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD}(\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$

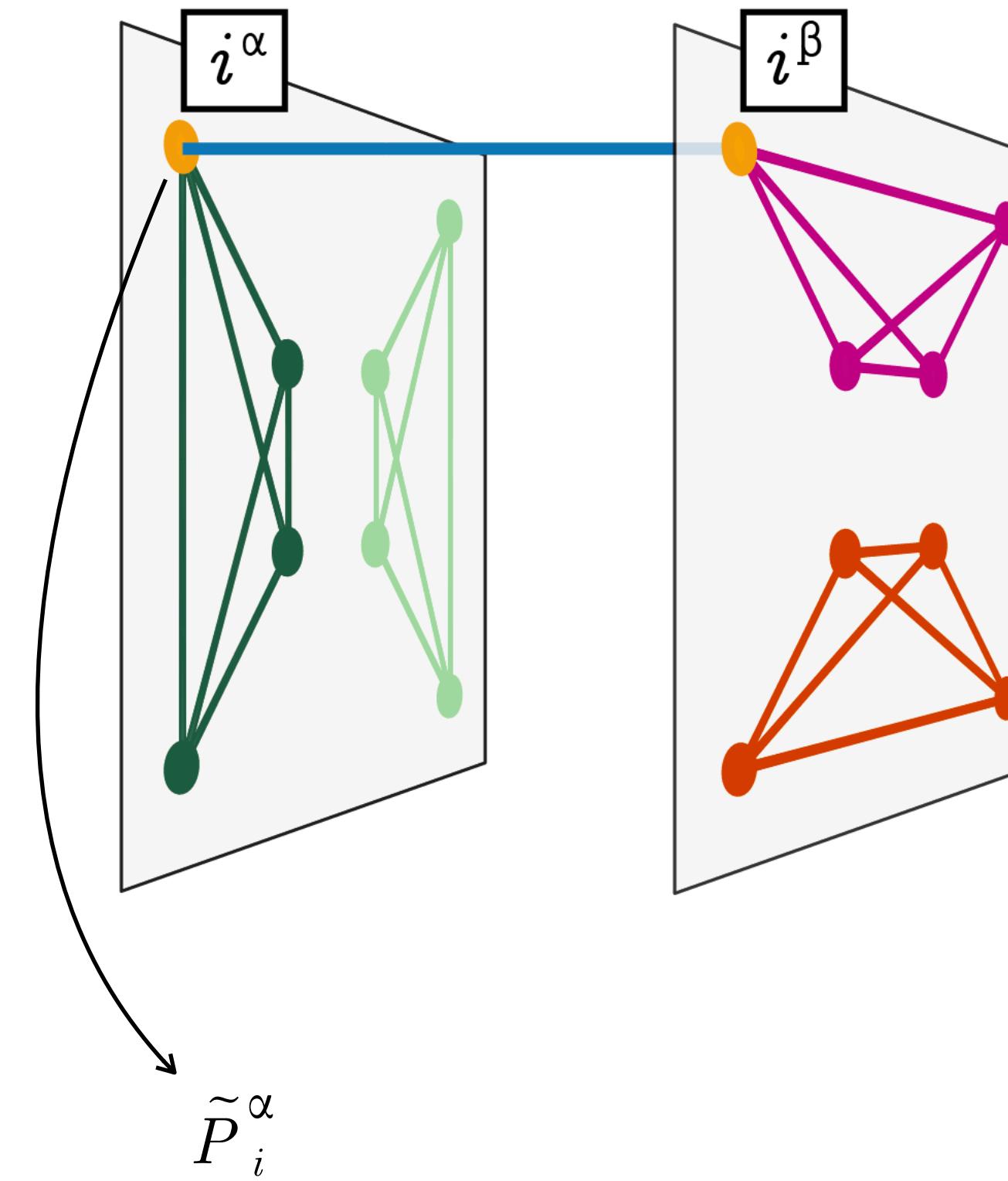
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



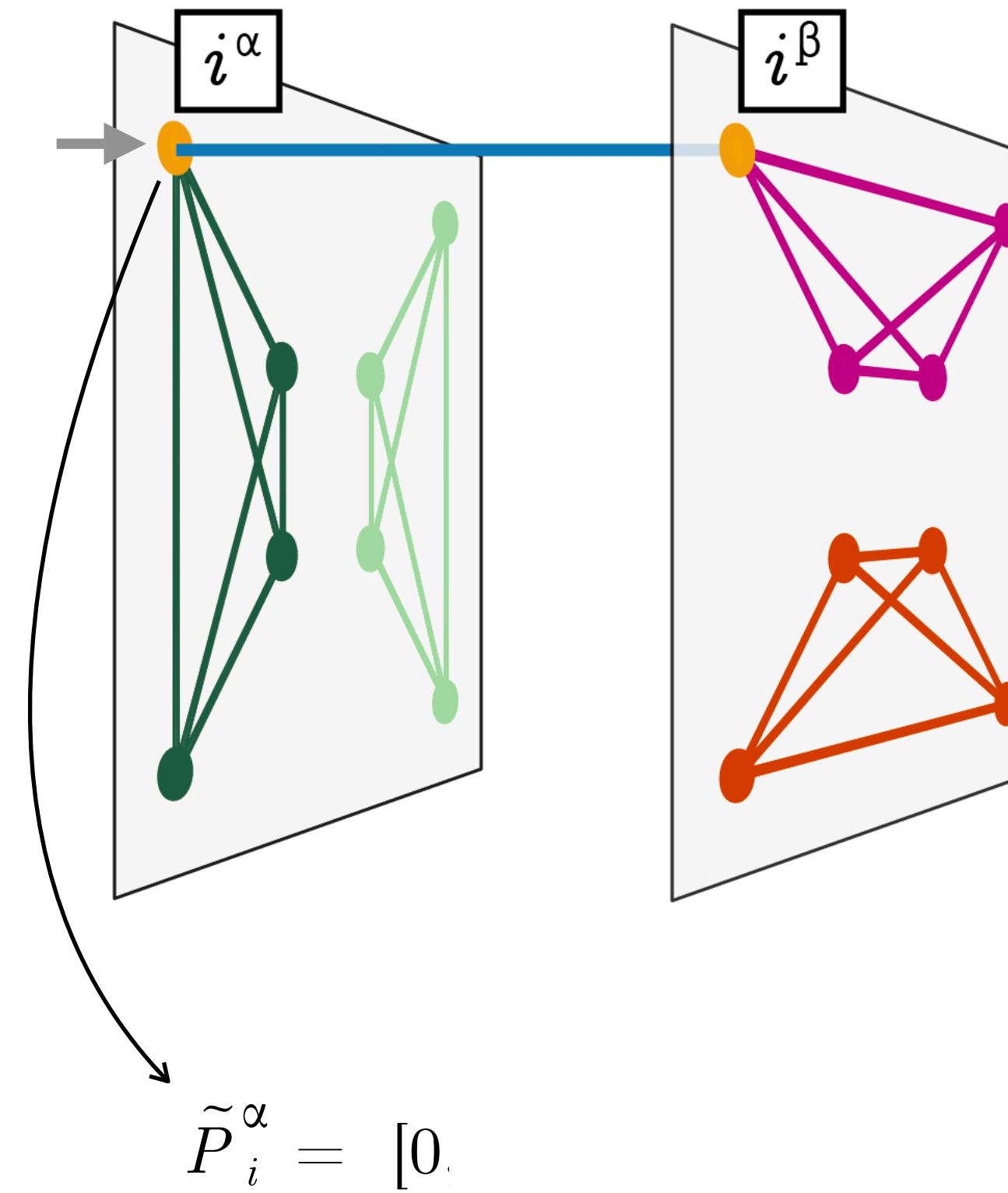
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



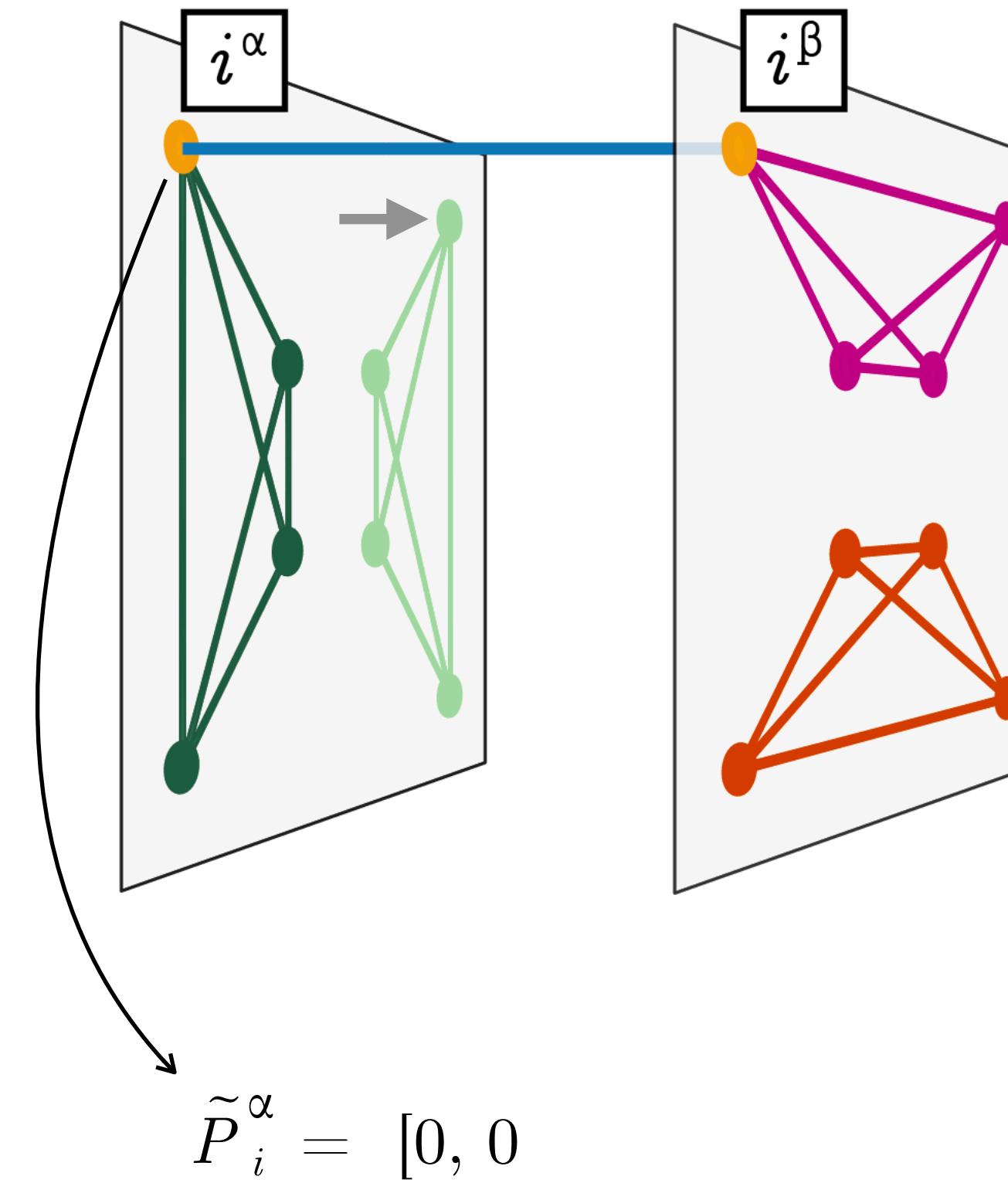
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



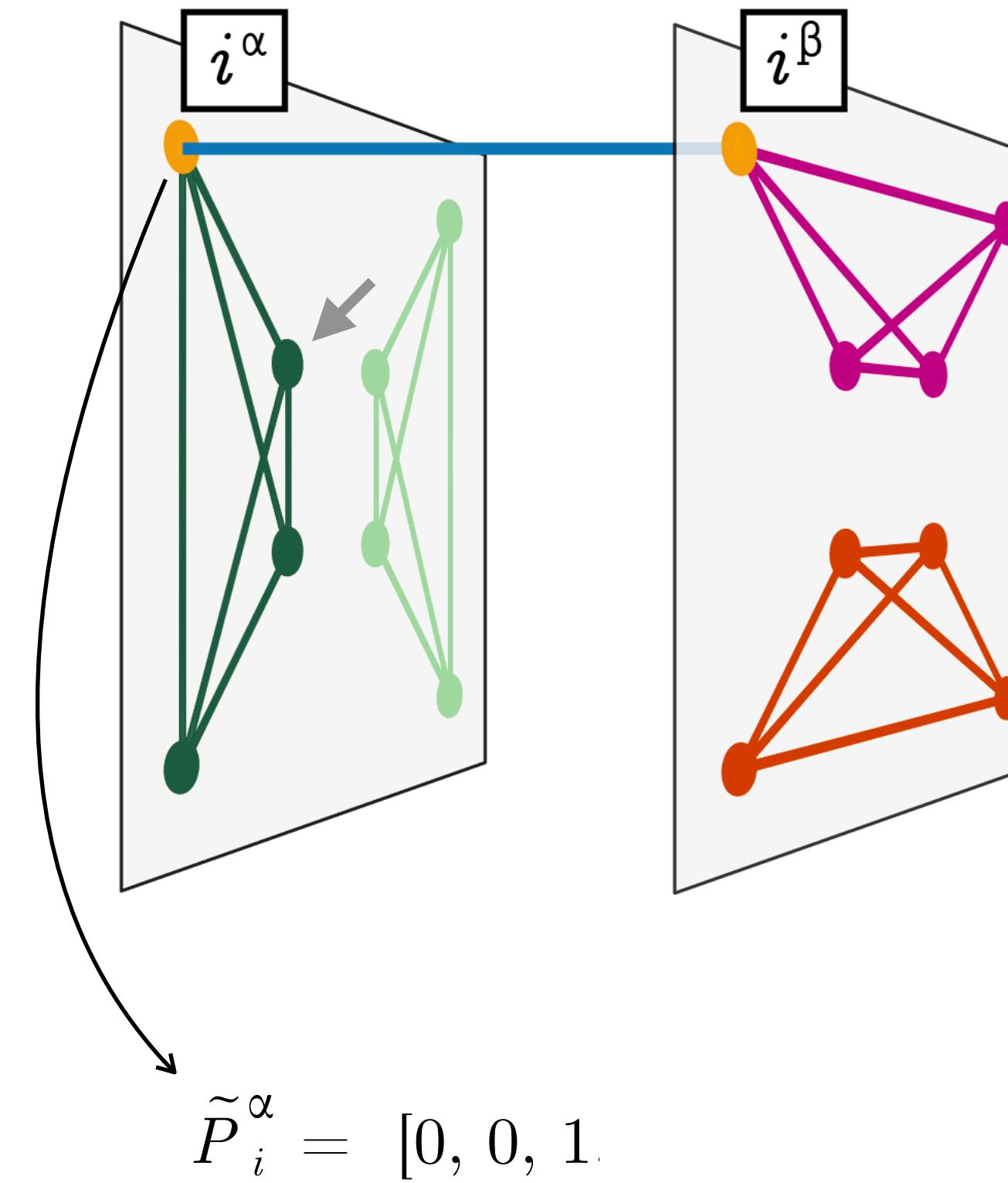
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



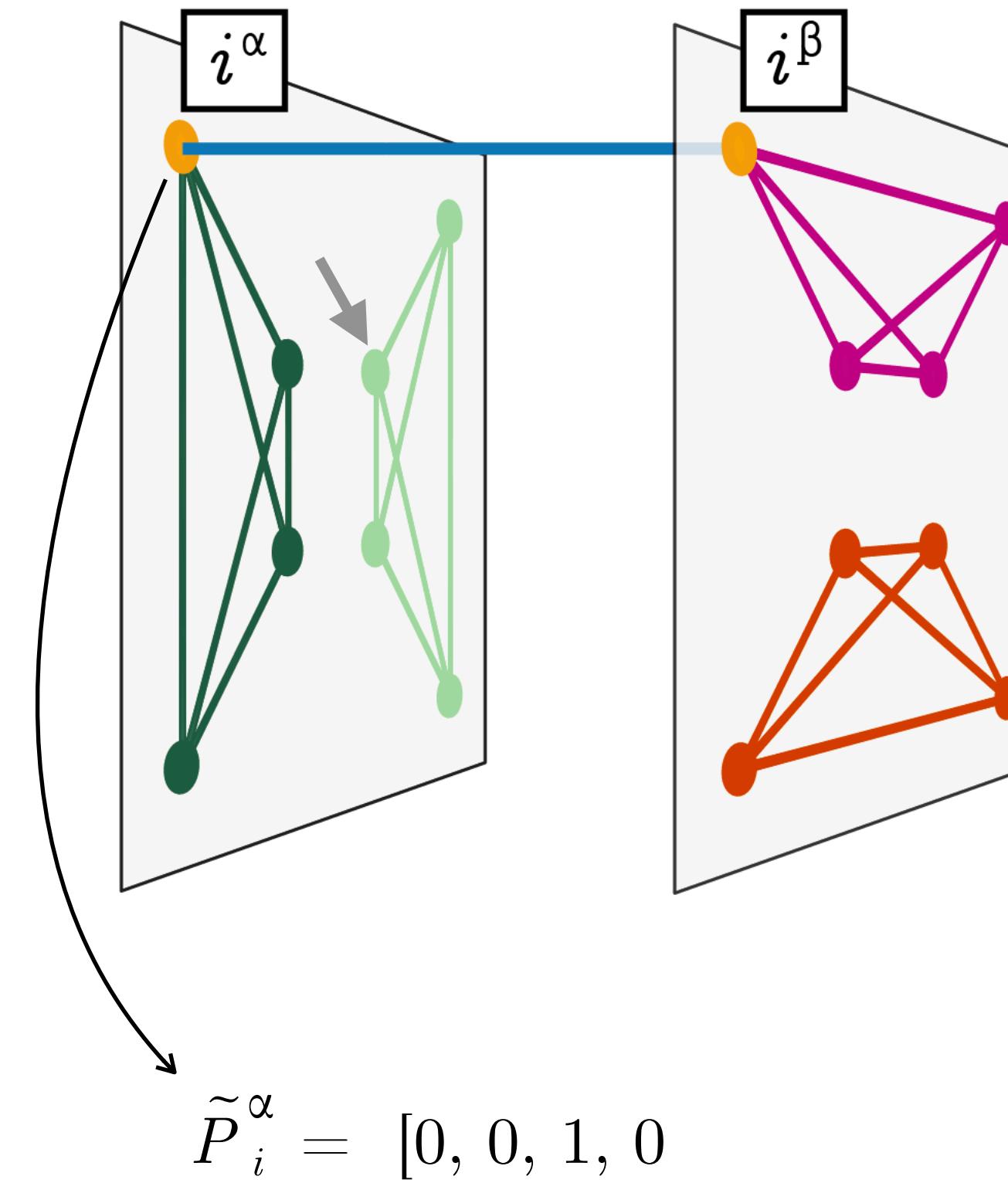
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



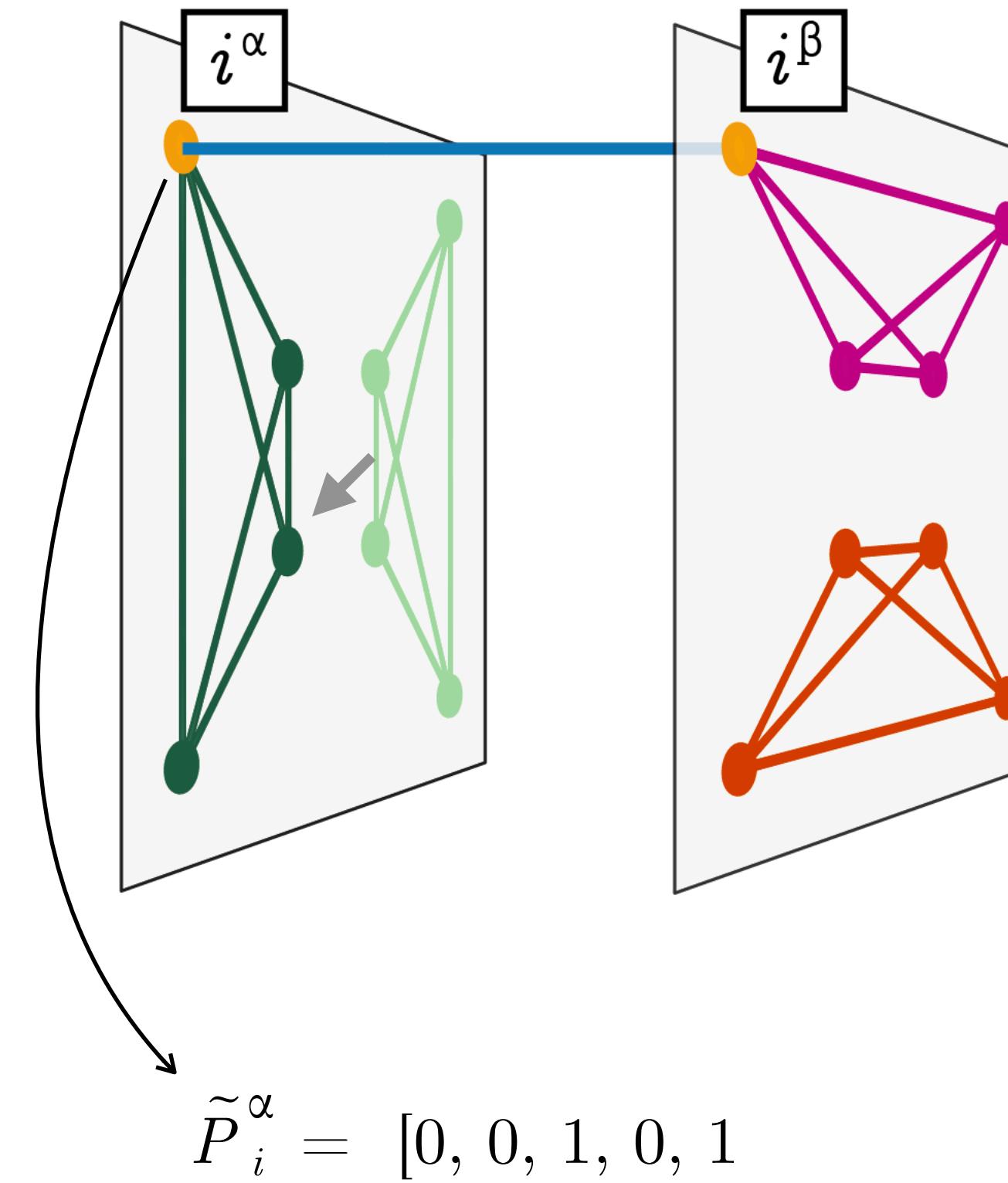
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



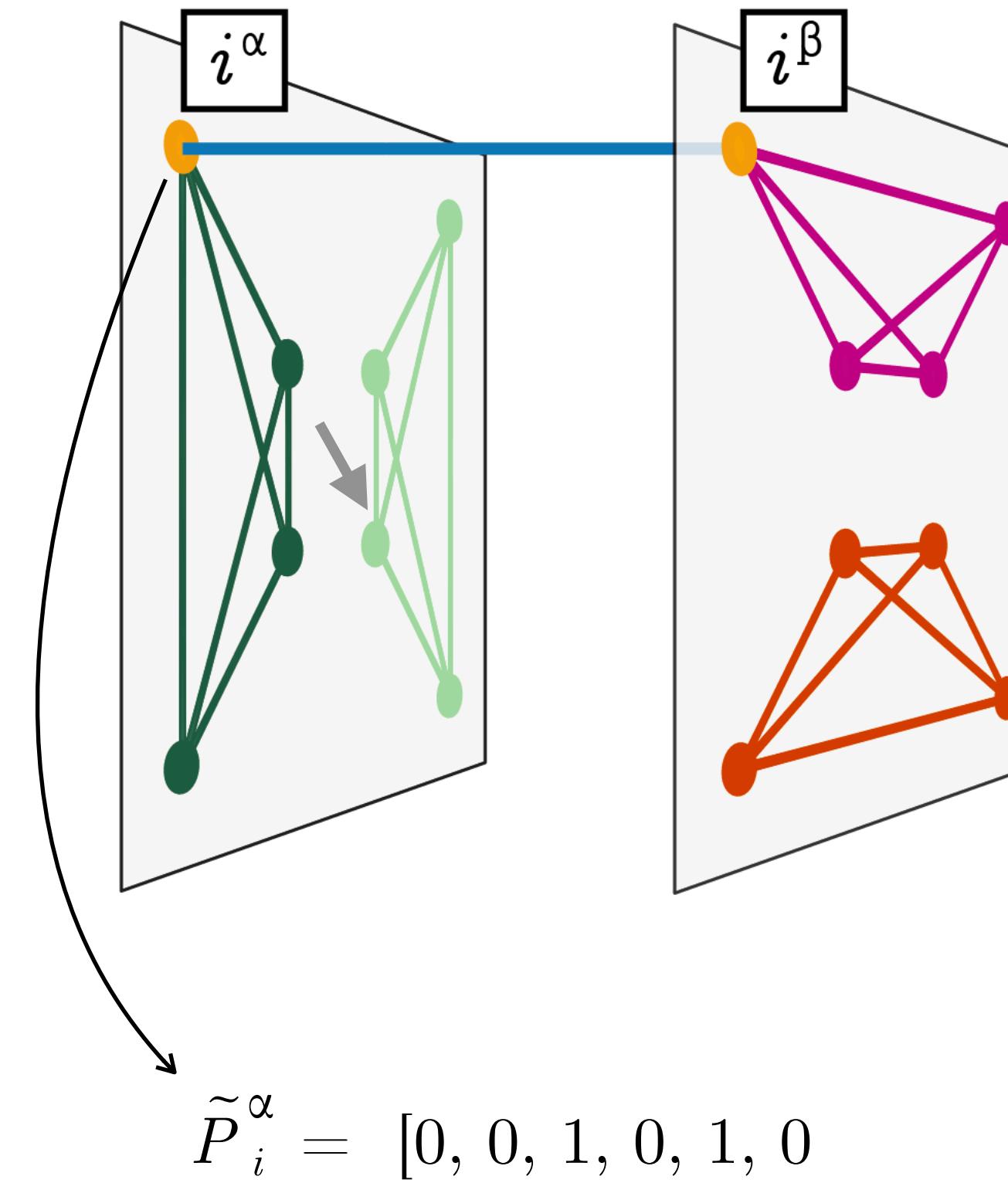
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



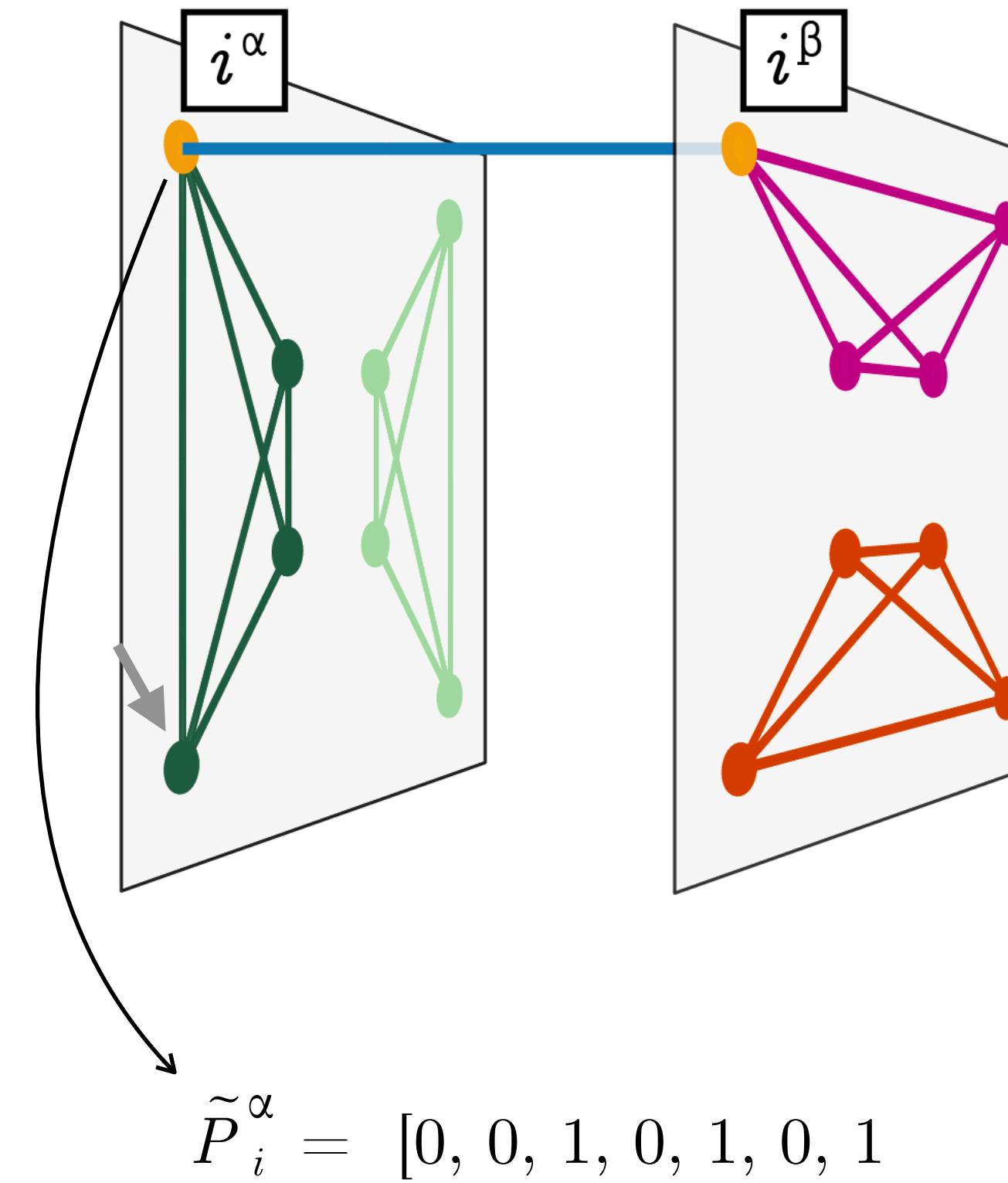
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



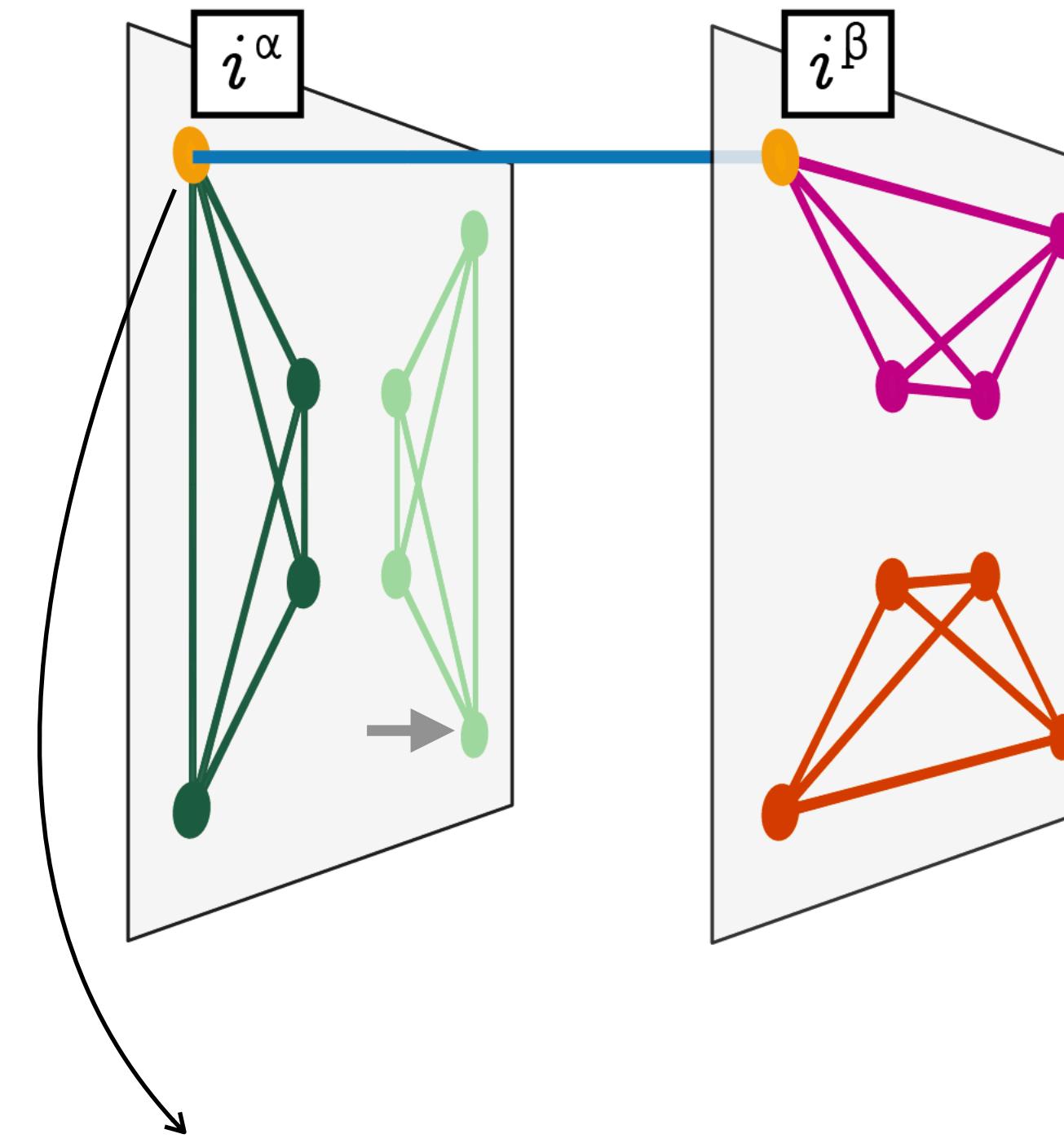
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



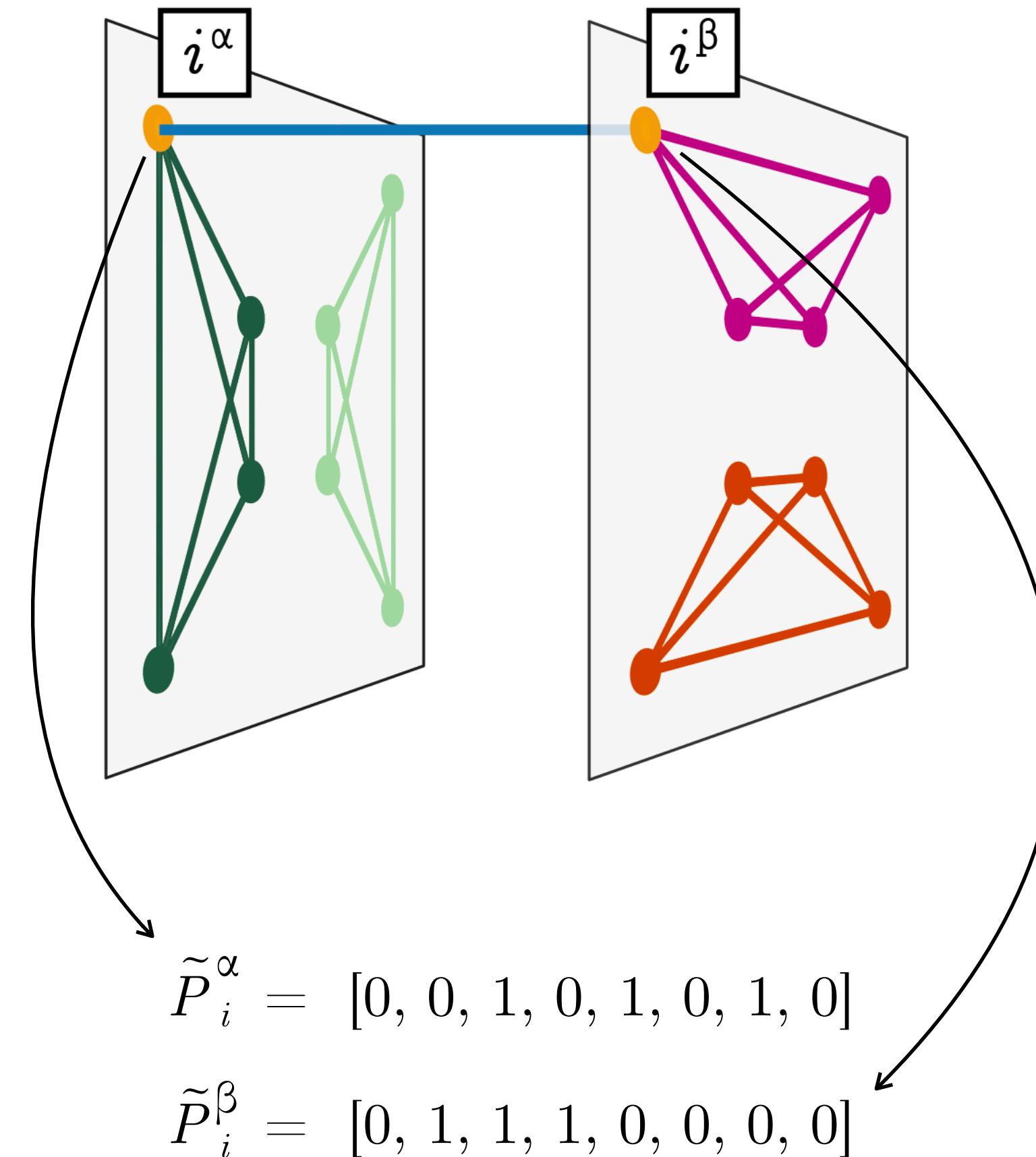
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



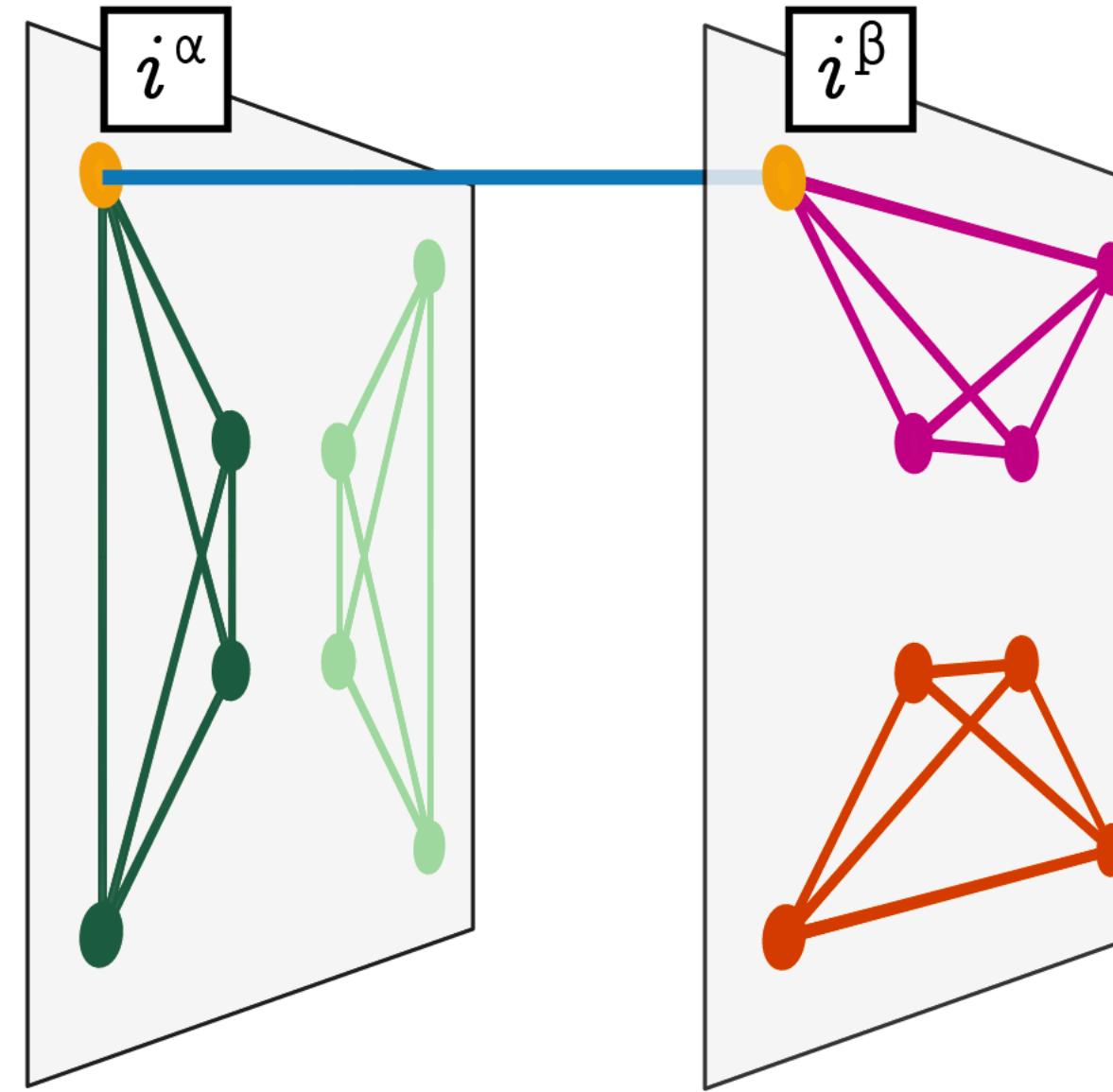
We use the Jensen-Shannon similarity to estimate temporal dependencies

$$D_i^{\alpha\beta} = 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta)$$



We use the Jensen-Shannon similarity to estimate temporal dependencies

$$\begin{aligned} D_i^{\alpha\beta} &= 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta) \\ &= 1 - \text{JSD}(\\ &\quad [0, 0, 1, 0, 1, 0, 1, 0] / 3, \\ &\quad [0, 1, 1, 1, 0, 0, 0, 0] / 3 \\ &) \end{aligned}$$

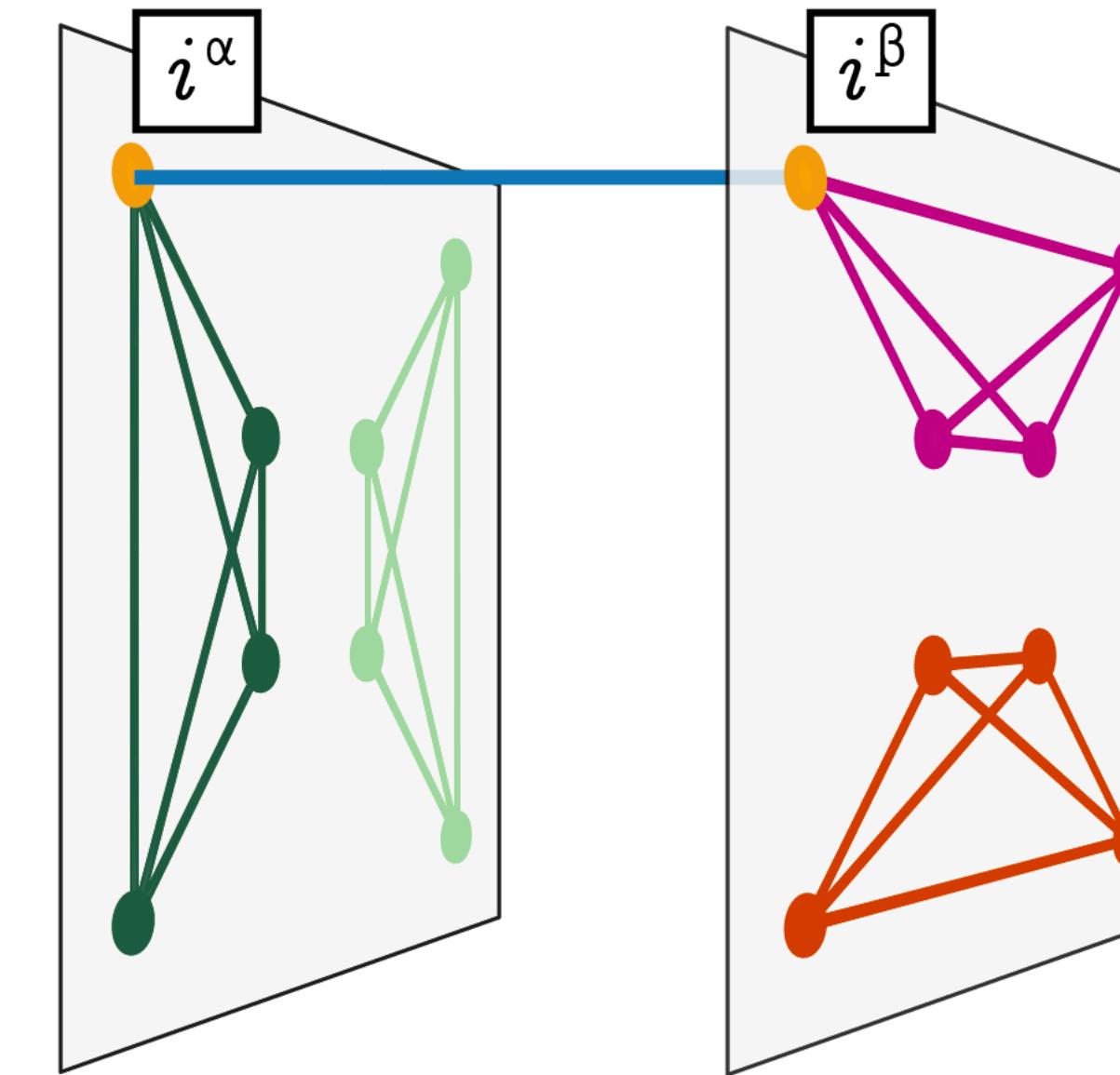


$$\tilde{P}_i^\alpha = [0, 0, 1, 0, 1, 0, 1, 0]$$

$$\tilde{P}_i^\beta = [0, 1, 1, 1, 0, 0, 0, 0]$$

Meaning: Jensen-Shannon divergence measures lost information when two states are merged

$$\begin{aligned} D_i^{\alpha\beta} &= 1 - \text{JSD} (\mathbf{P}_i^\alpha, \mathbf{P}_i^\beta) \\ &= 1 - \text{JSD}(\\ &\quad [0, 0, 1, 0, 1, 0, 1, 0] / 3, \\ &\quad [0, 1, 1, 1, 0, 0, 0, 0] / 3 \\ &\quad) \\ &= 0.333... \end{aligned}$$



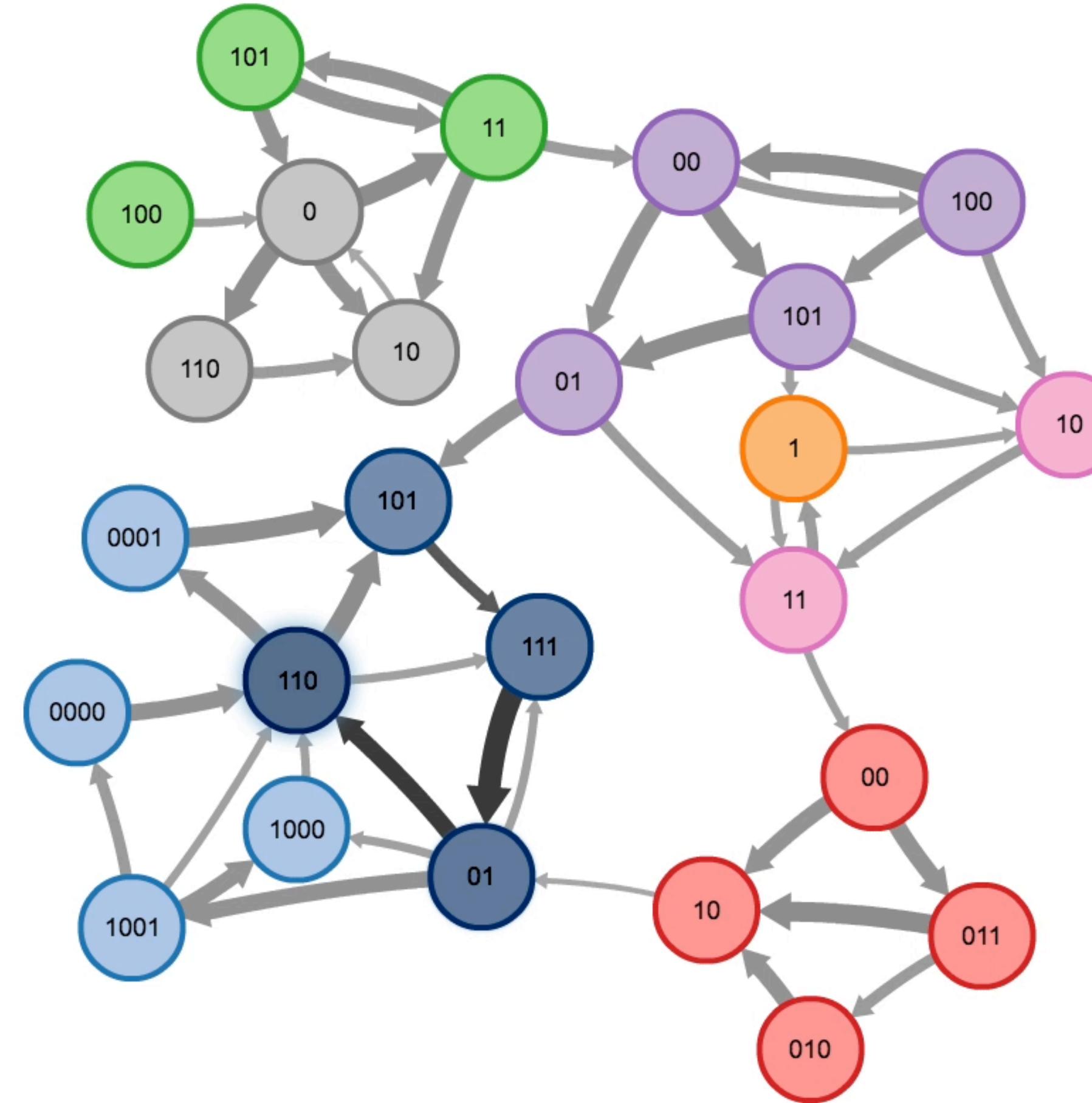
$$\tilde{P}_i^\alpha = [0, 0, 1, 0, 1, 0, 1, 0]$$

$$\tilde{P}_i^\beta = [0, 1, 1, 1, 0, 0, 0, 0]$$

Putting this principle into a **flow-based** modeling framework (and Infomap)

Flow-based community detection minimizes description length of random walk on a network

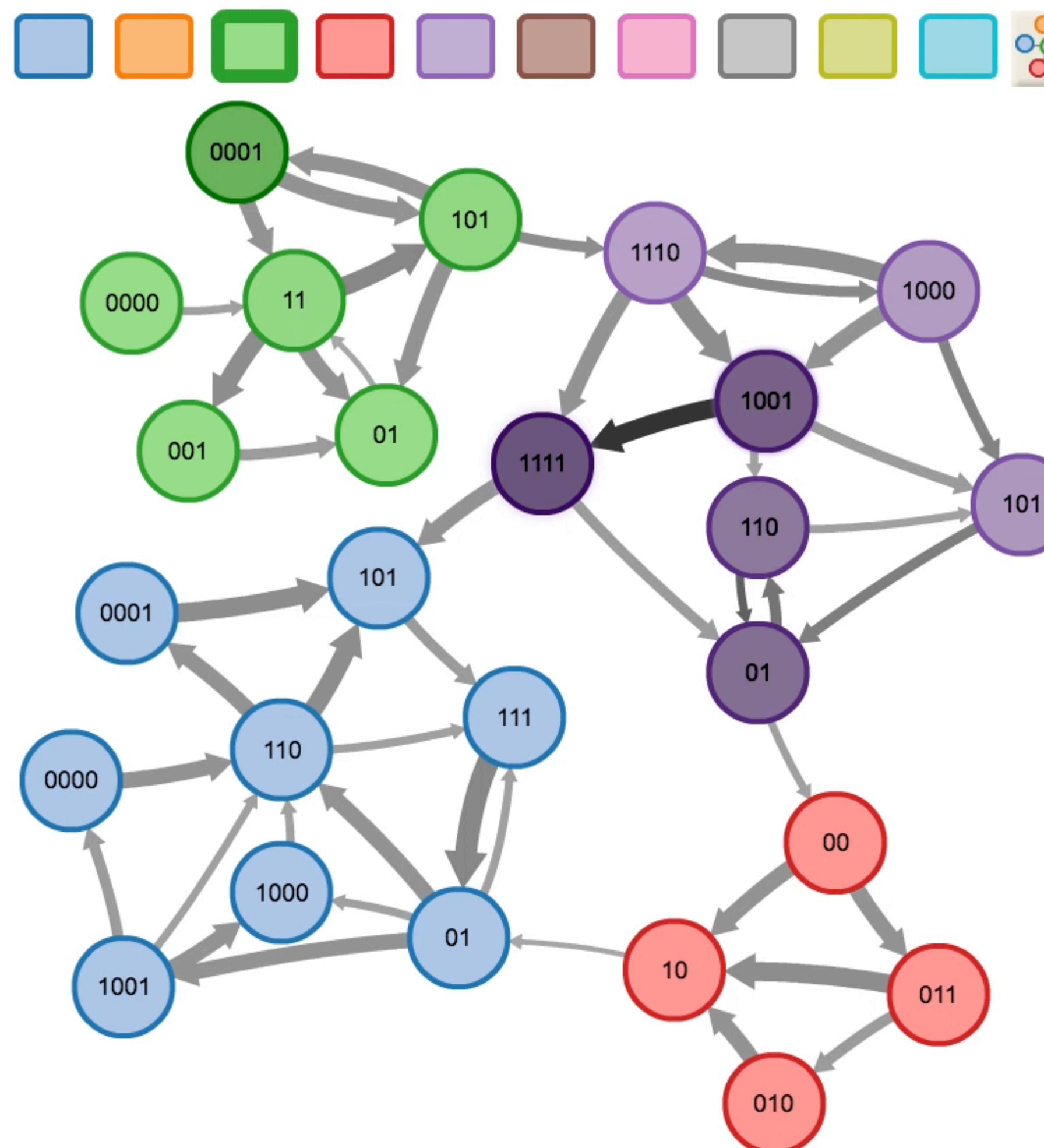
your cost function
↓
 $L(M)$



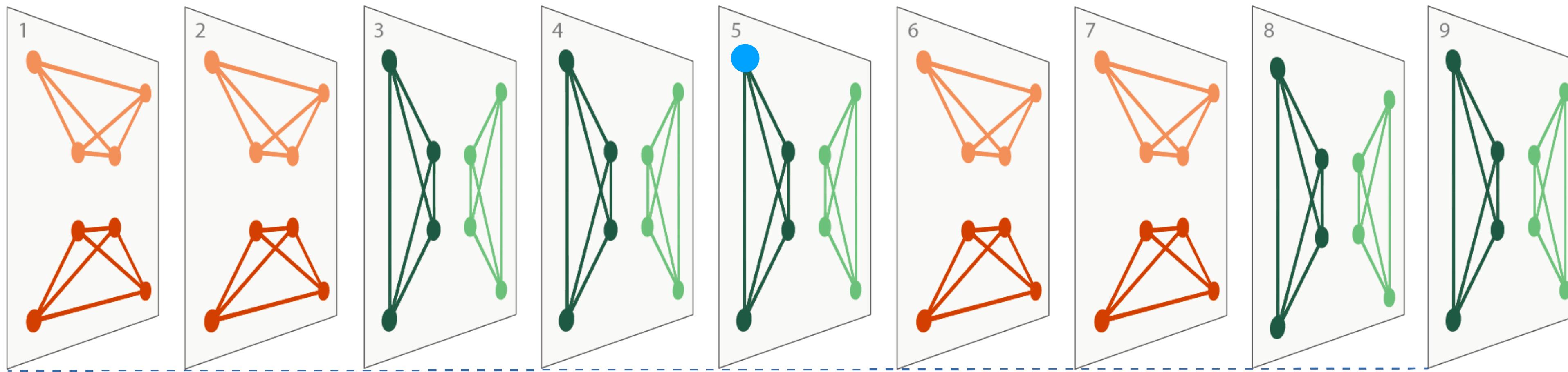
Flow-based community detection minimizes description length of random walk on a network

your cost function

$$L(M) = q \cdot H(Q) + \sum_{j=1}^m p_j \cdot H(P_j)$$



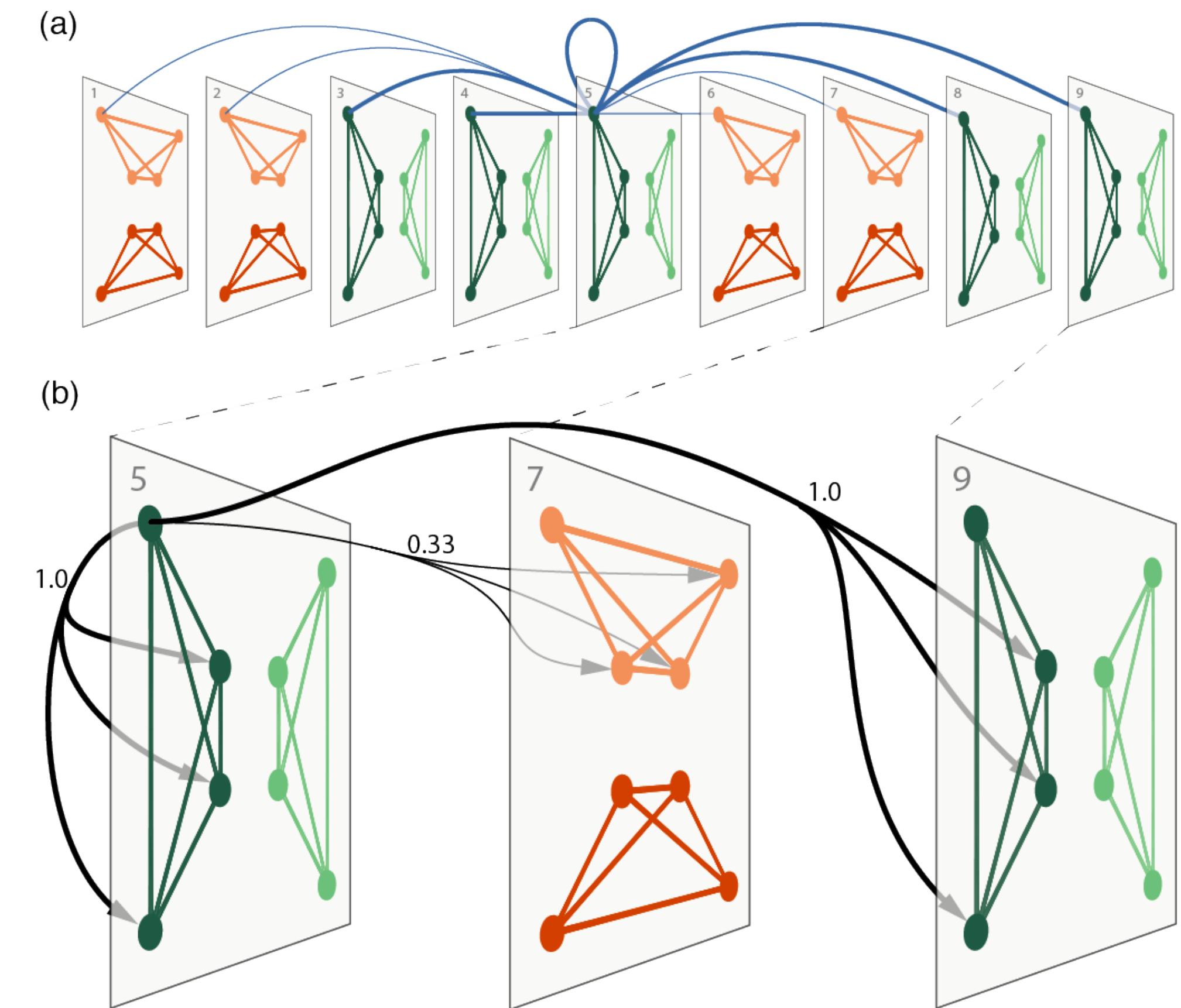
A random walker on a multilayer network changes layer with probability r



Create “new network” where weights represent random walker transition probabilities

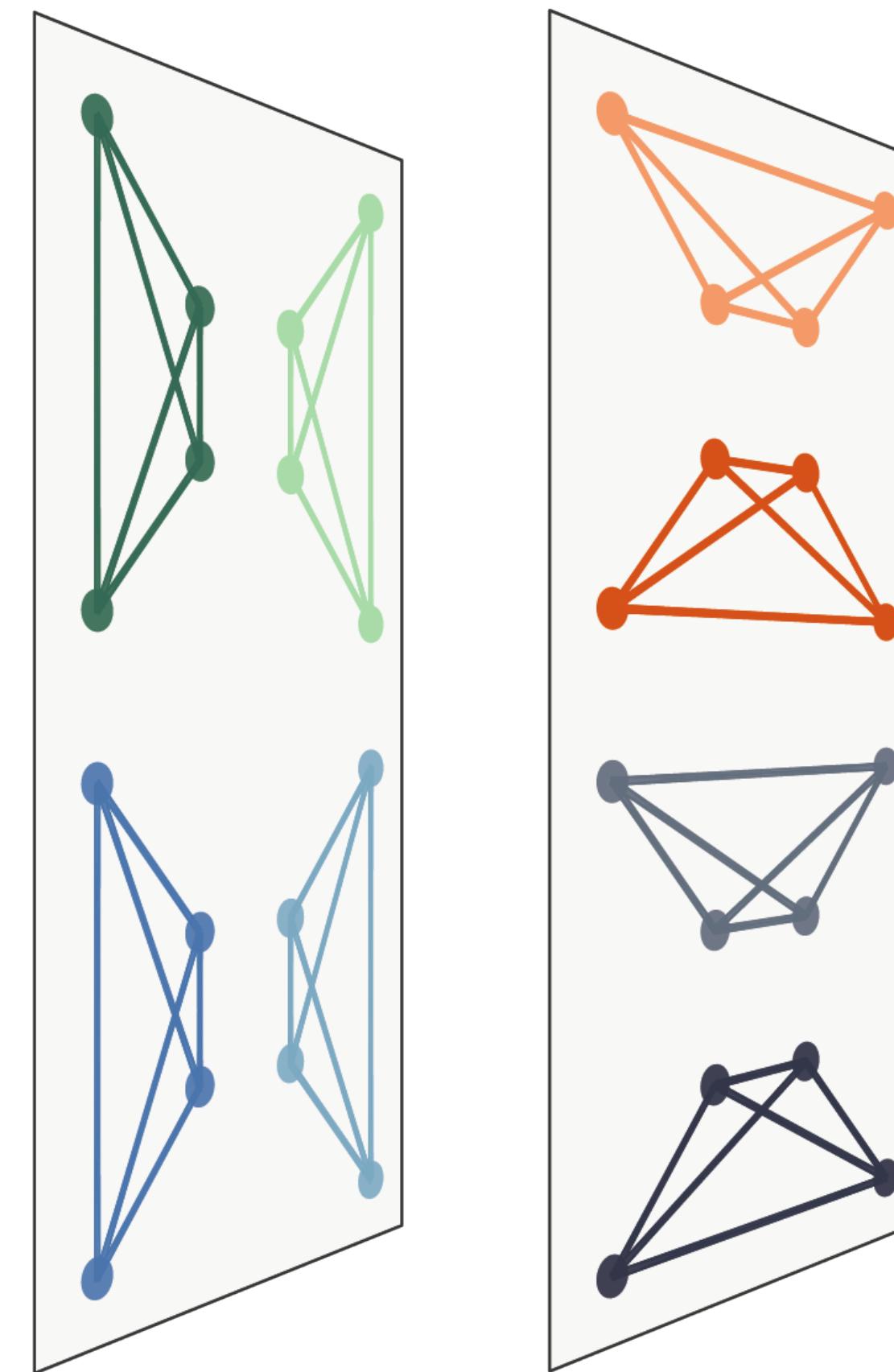
$$P_{ij}^{\alpha\beta}(r) = (1 - r) \frac{W_{ij}^\beta}{s_i^\beta} \delta_{\alpha\beta} + r \frac{D_i^{\alpha\beta}}{S_i^\alpha} \frac{W_{ij}^\beta}{s_i^\beta}$$

- (a) Use neighborhood flow coupling to get D_i
- (b) Re-weight network with $P_{ij}^{\alpha\beta}$
- > Minimize the Map Equation

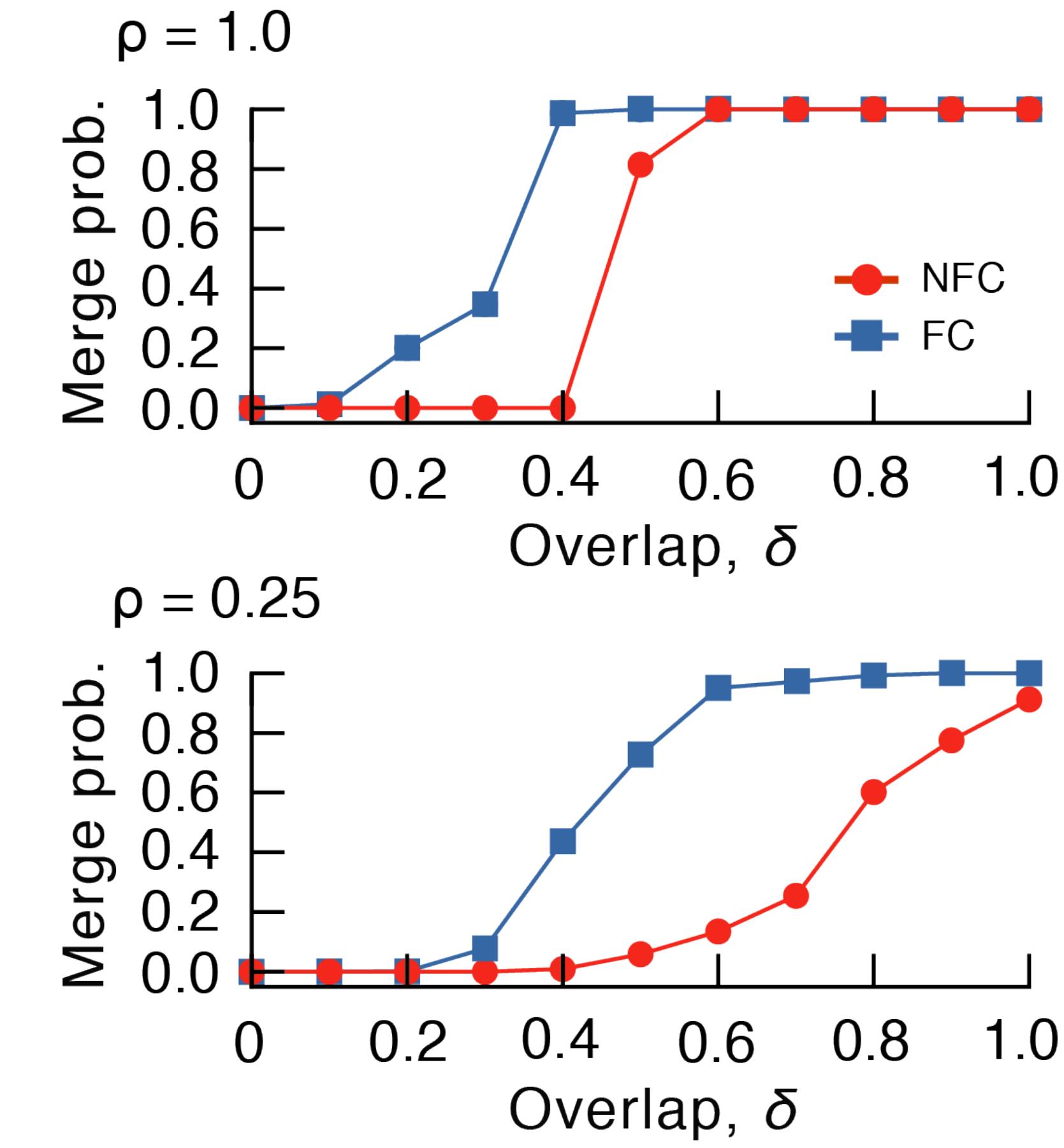
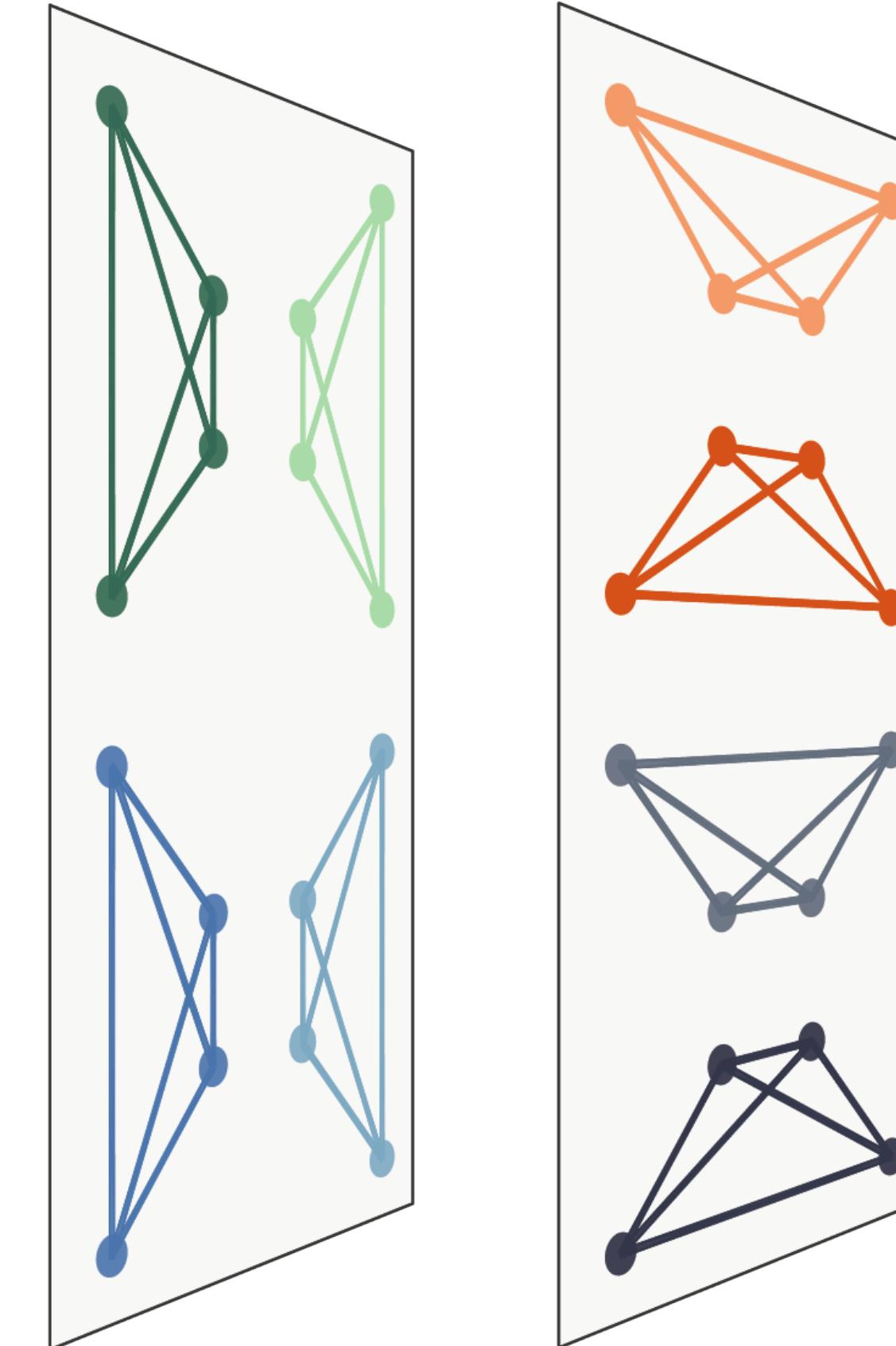


How much better is this for community detection in temporal networks?

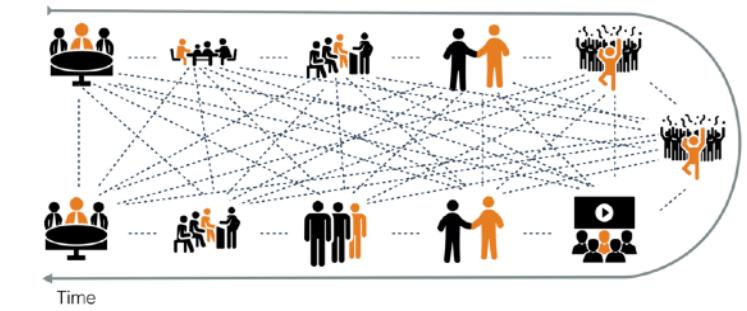
Neighborhood flow coupling enables detection of **overlapping** communities



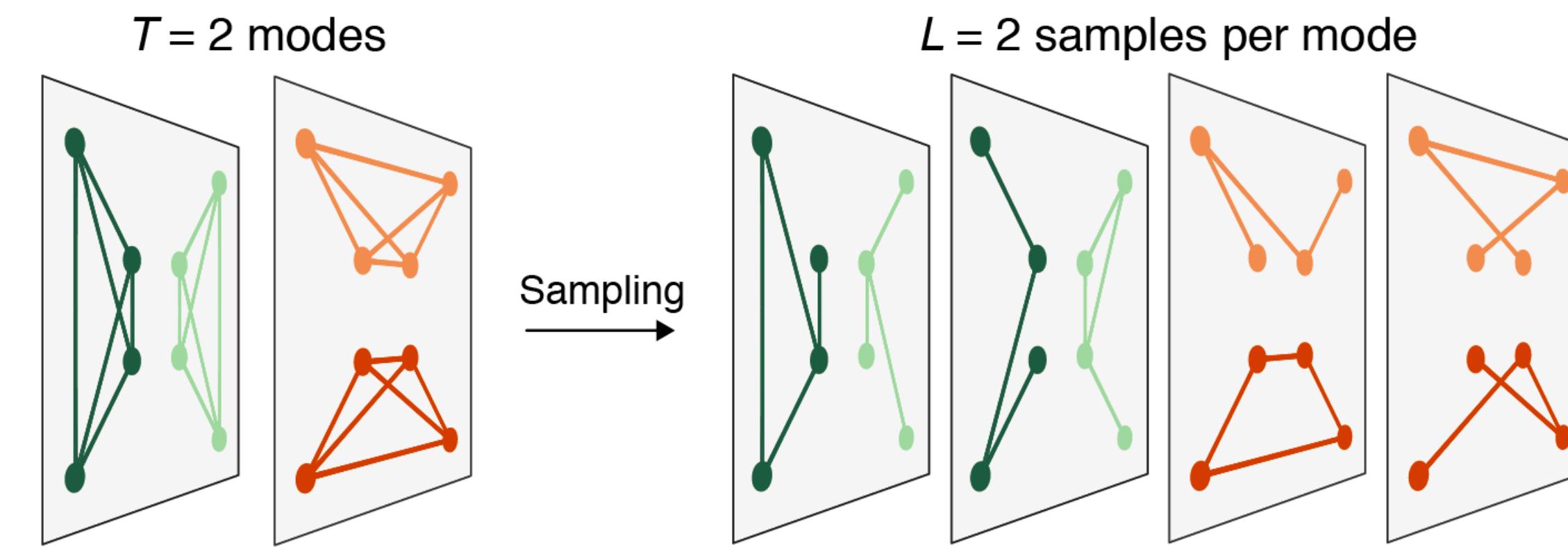
Neighborhood flow coupling enables detection of **overlapping** communities



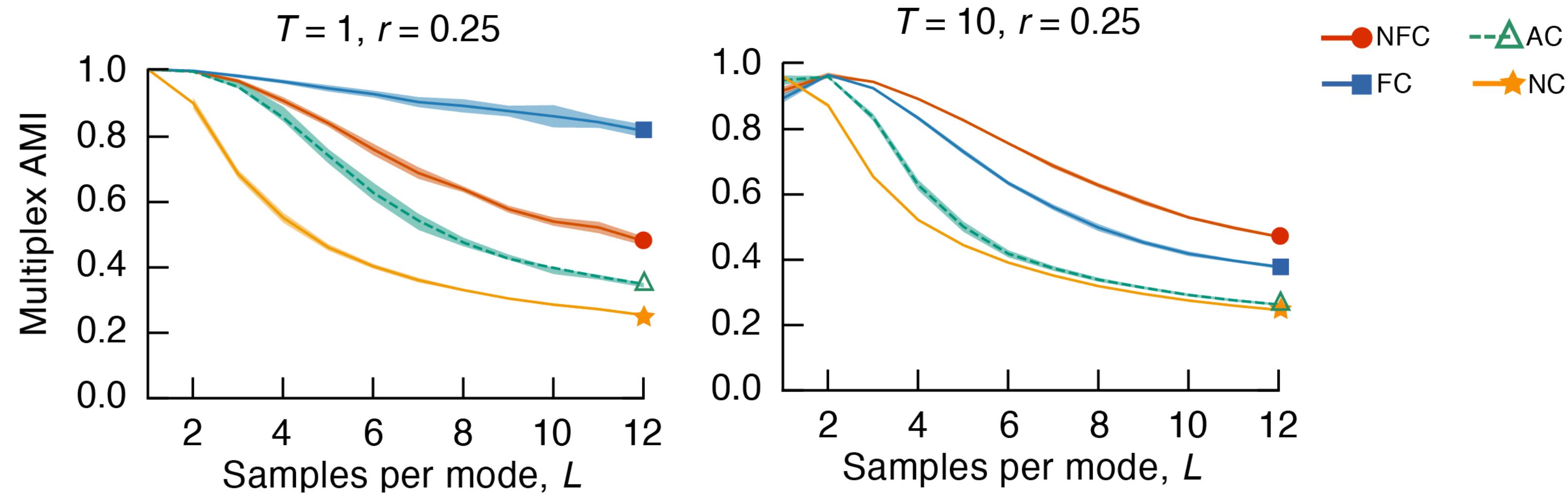
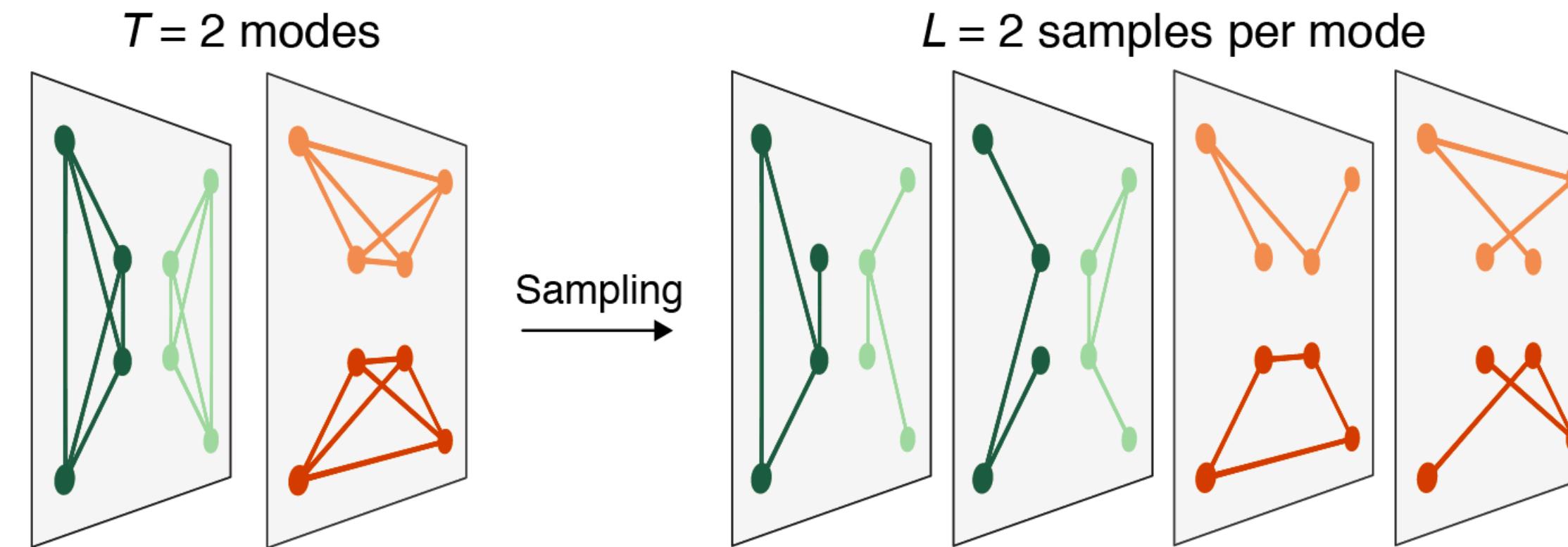
Full coupling (FC):



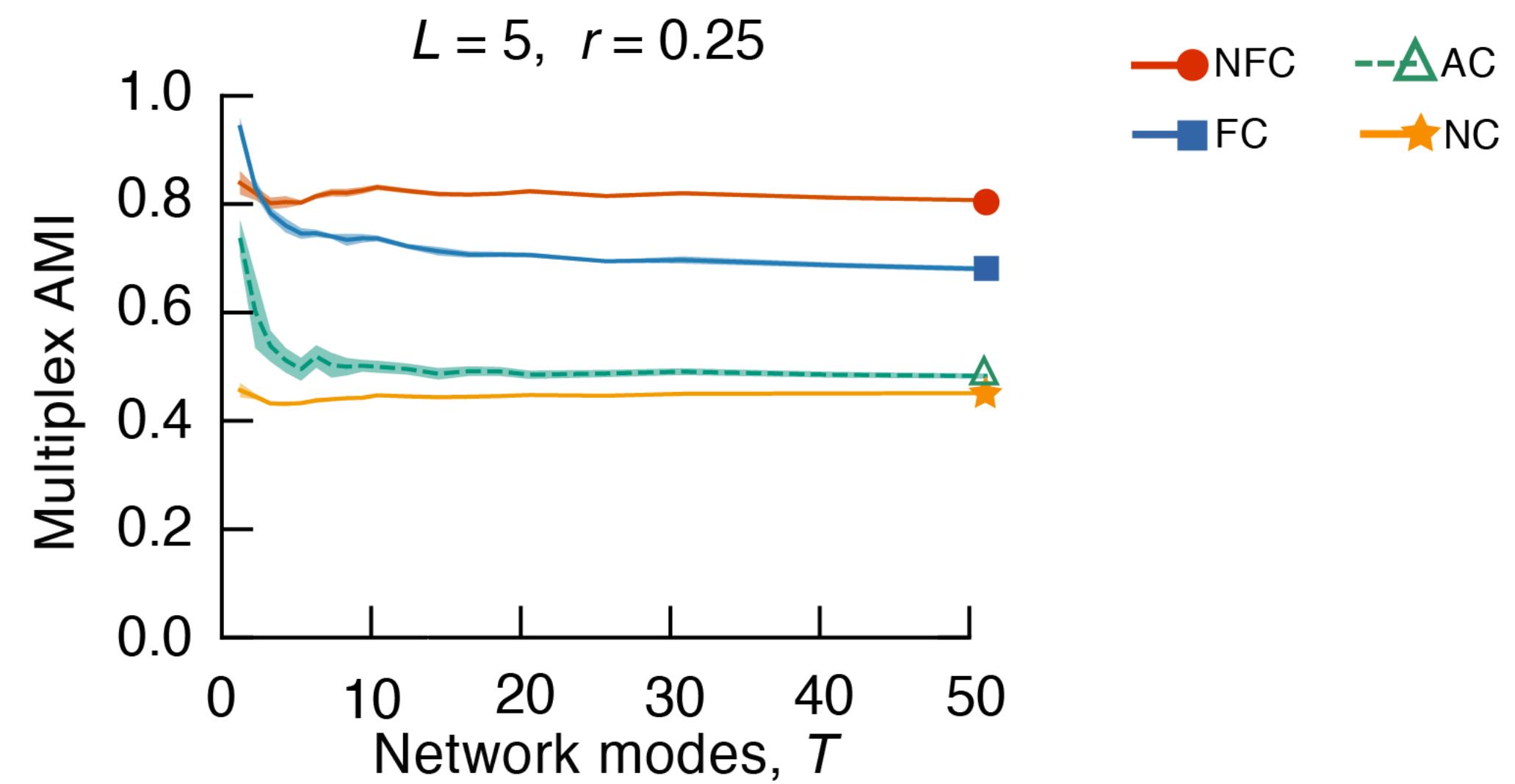
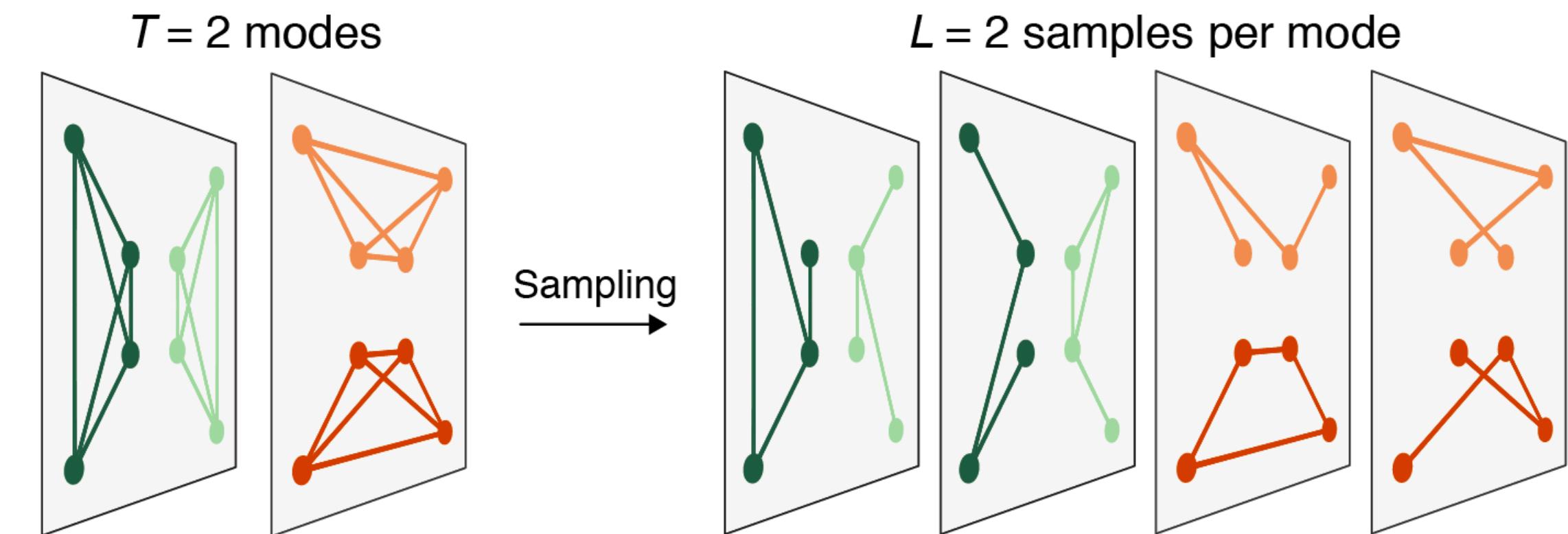
Neighborhood flow coupling enables detection of **intermittent** communities



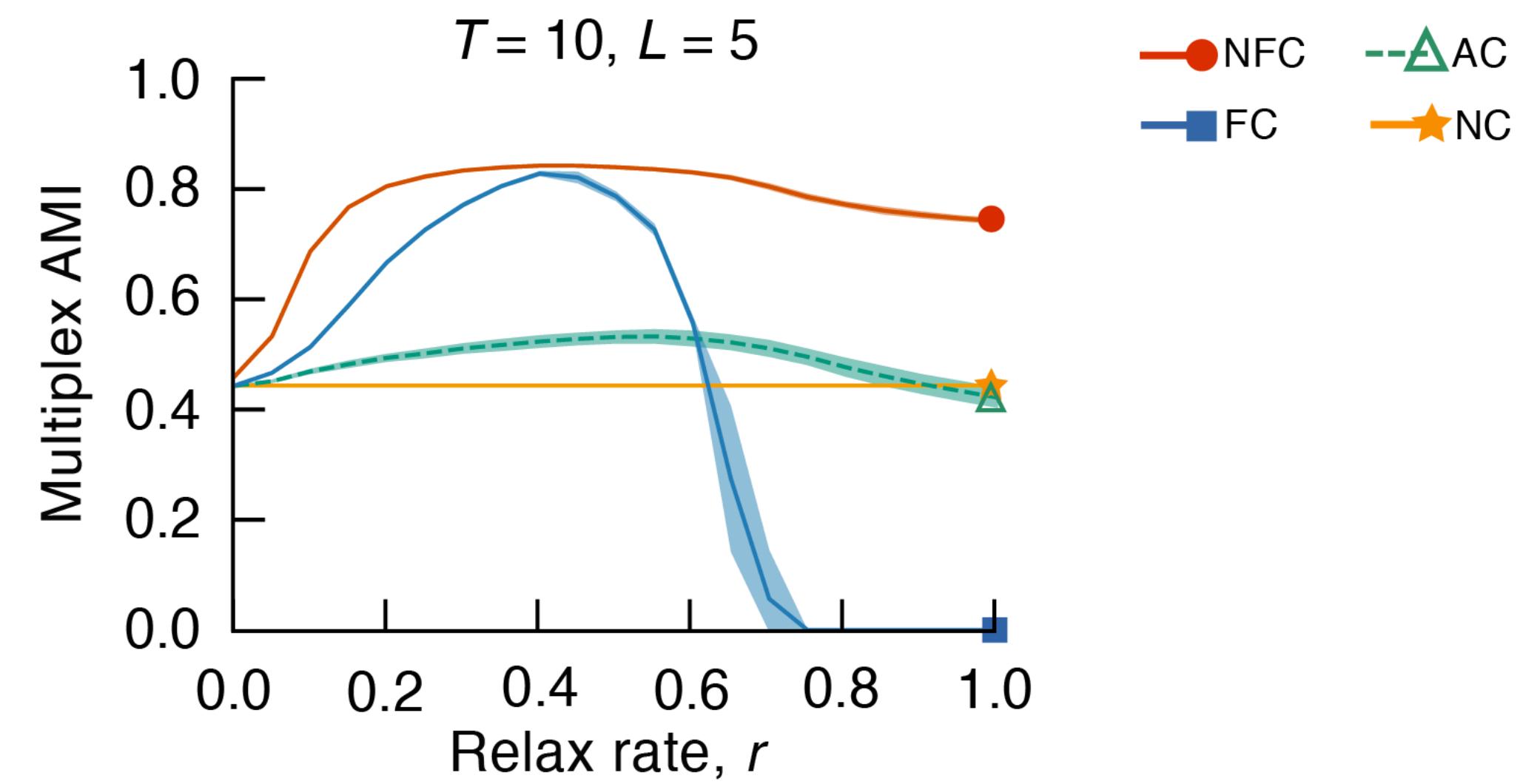
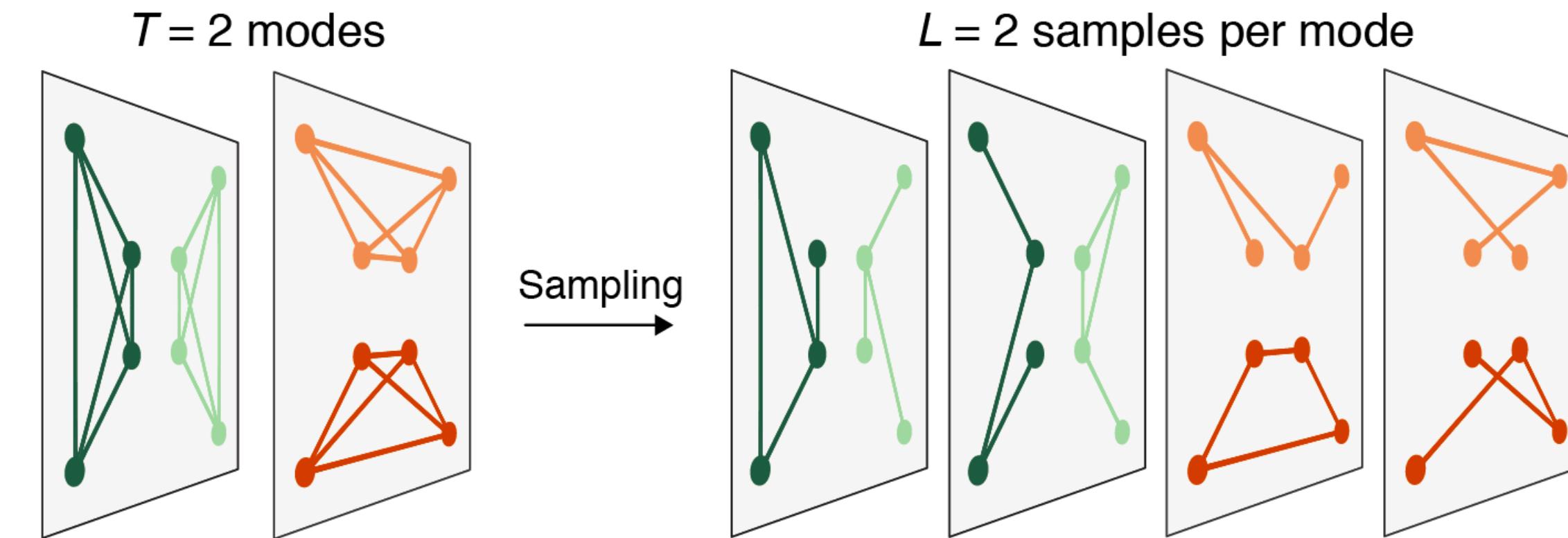
Neighborhood flow coupling enables detection of **intermittent** communities



Neighborhood flow coupling enables detection of **intermittent** communities



Neighborhood flow coupling is parameter insensitive



This repository Search

Pull requests Issues Marketplace Explore

mapequation / infomap

Code Issues 2 Pull requests 1 Projects 0 Wiki Insights

Multi-level network clustering based on the Map equation

network-analysis clustering-algorithm infomap map-equation information-theory

680 commits 5 branches 0 releases

Branch: master New pull request

danieledler Fix normalize flow to nan if zero link out weight

examples Fix R igraph example for new igraph version

interfaces Fix python tree iterator interface

src Fix normalize flow to nan if zero link out weight

.gitignore Display per iteration statistics

.hgignore Added js example running in b

CHANGES.txt Increase version

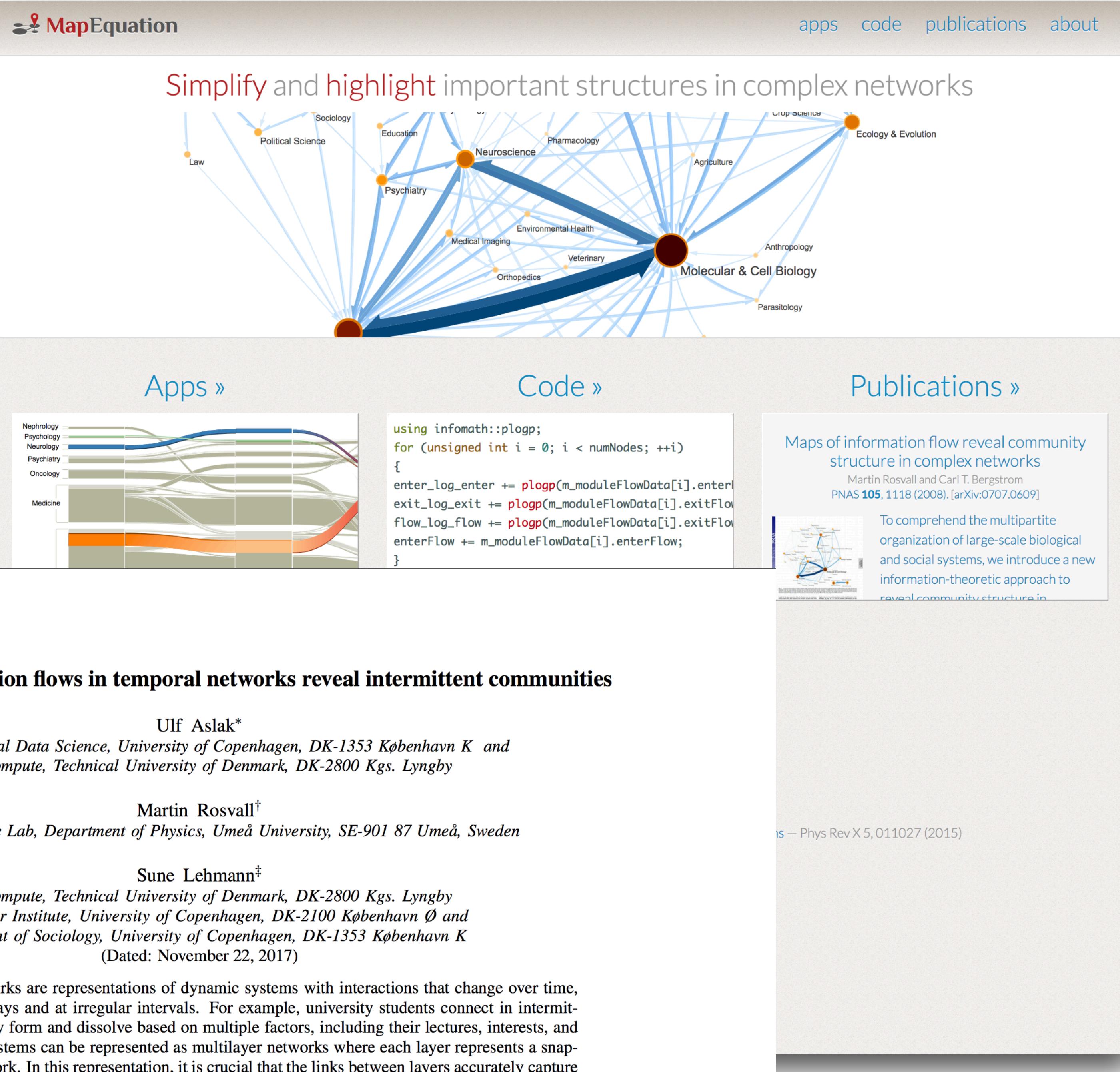
LICENSE GPLv3.txt * Added README.txt and update

Makefile Use python3 to setup python3

README.md Fix #9 broken link in readme

ninetriangles.net Added example network ninet

README.md



Infomap Software Package

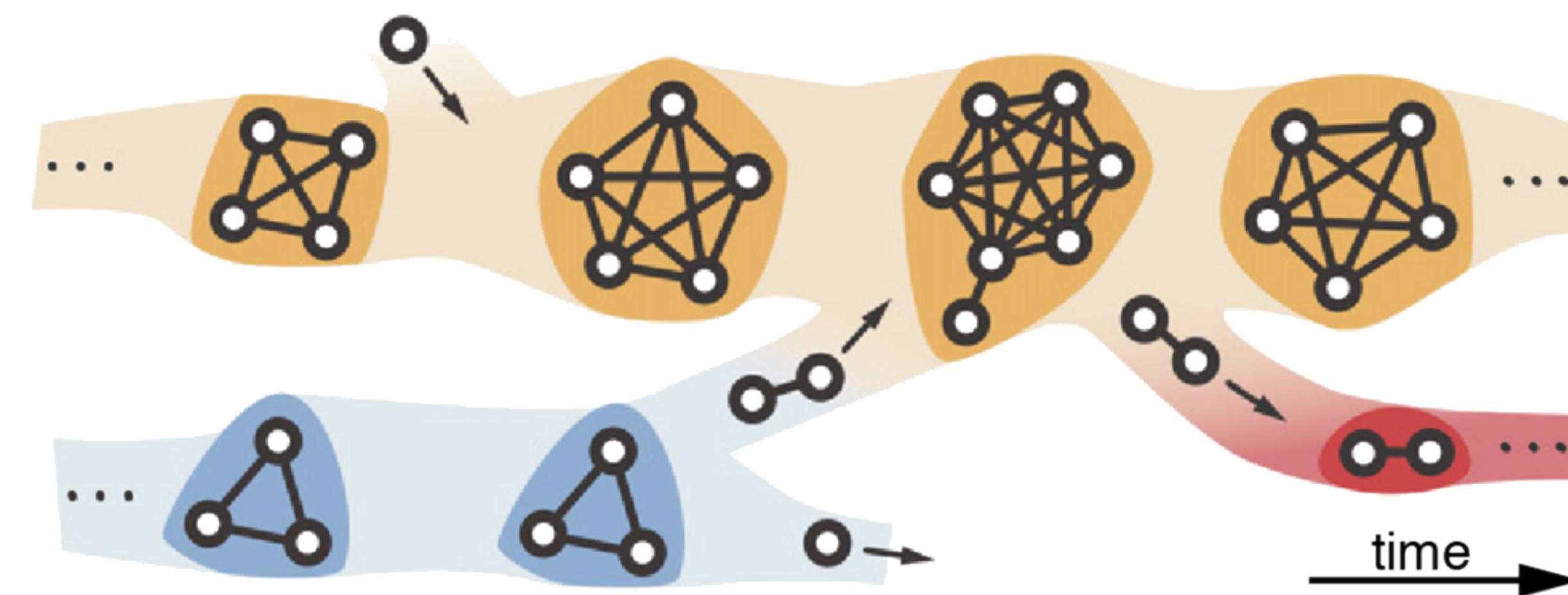
Infomap is a network clustering algorithm based on the Map equation.

For more info, see www.mapequation.org.

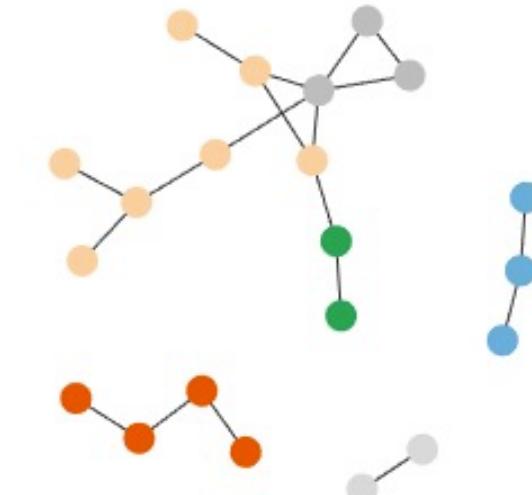
Getting started:

21 Nov 2017

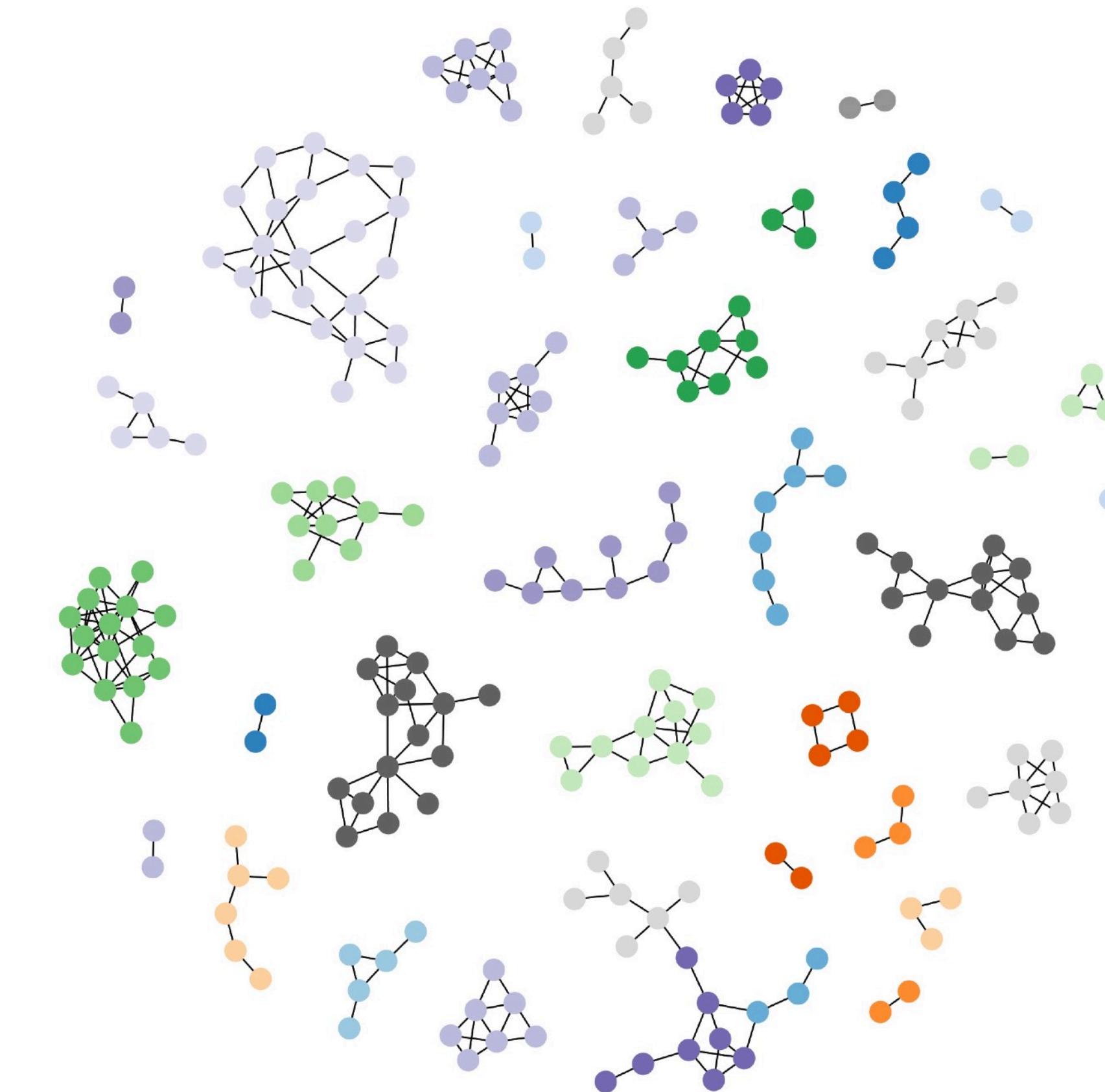
How does this method help us understand social contact networks?



We use two fundamentally different temporal contact networks

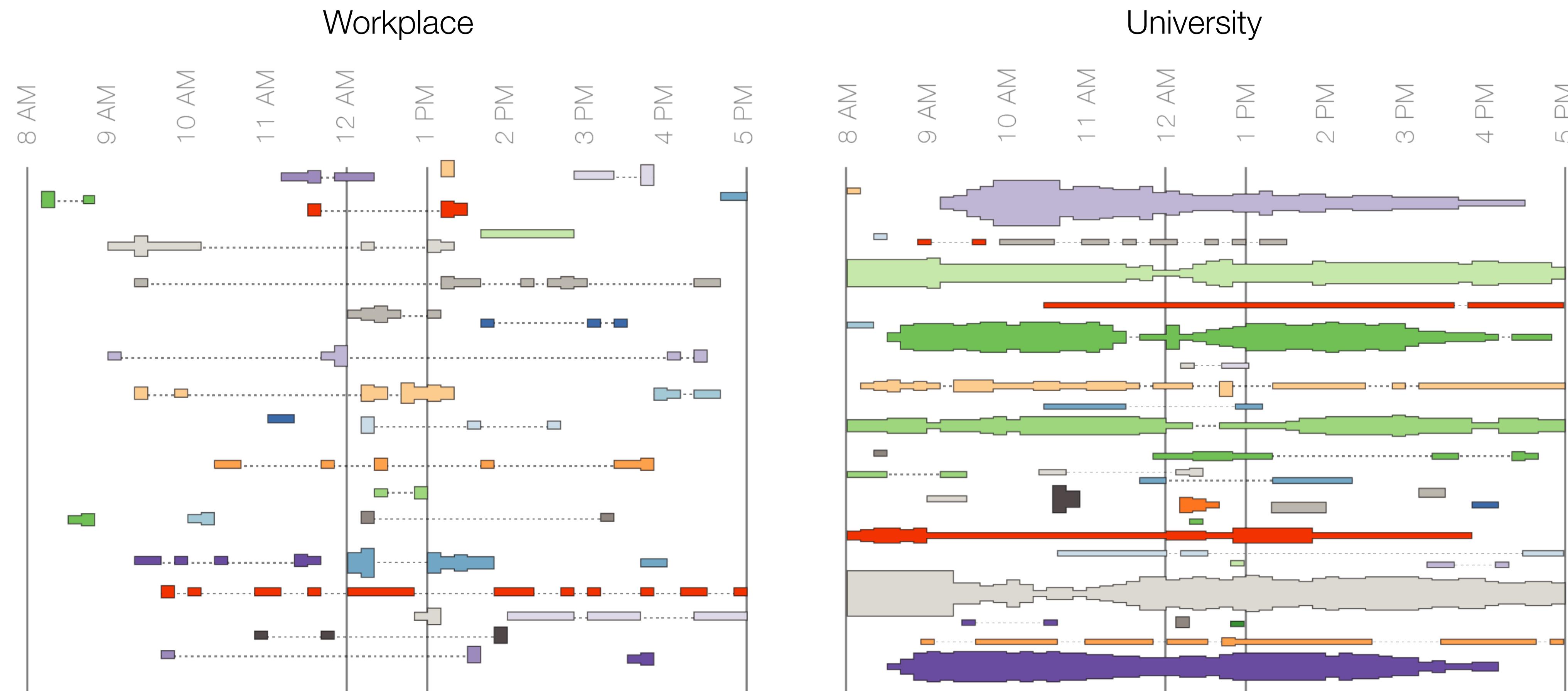


Workplace ($N = 145$)



University ($N = 665$)

Visualization gives us an intuitive understanding of how our method behaves



- We introduce ***neighborhood flow coupling***: a principle for inferring intertemporal dependencies in networks
 - We provide an open source implementation
 - NFC makes flow based community detection less parameter sensitive
 - **NFC highlights interesting structure in social networks!**