Blockmodeling dynamic networks: a Monte Carlo simulation study

Marjan Cugmas & Aleš Žiberna

Faculty of Social Sciences, University of Ljubljana

APPLIED STATISTICS 2021

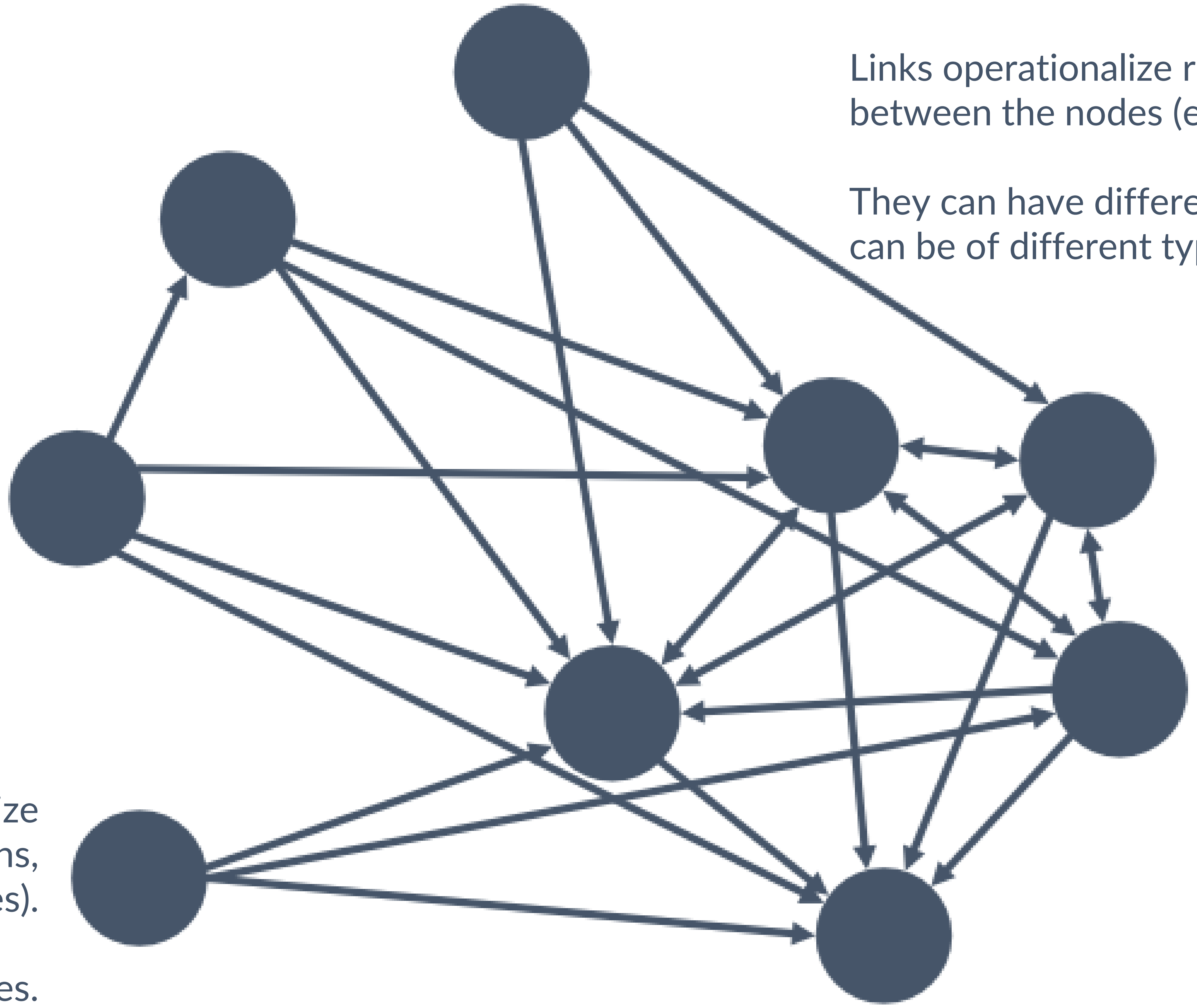
University *of Ljubljana*
Faculty *of Social Sciences*

Network

Relationships between units can be operationalized by a network.

Links operationalize relationships between the nodes (e.g., contact).

They can have different weights or can be of different types.

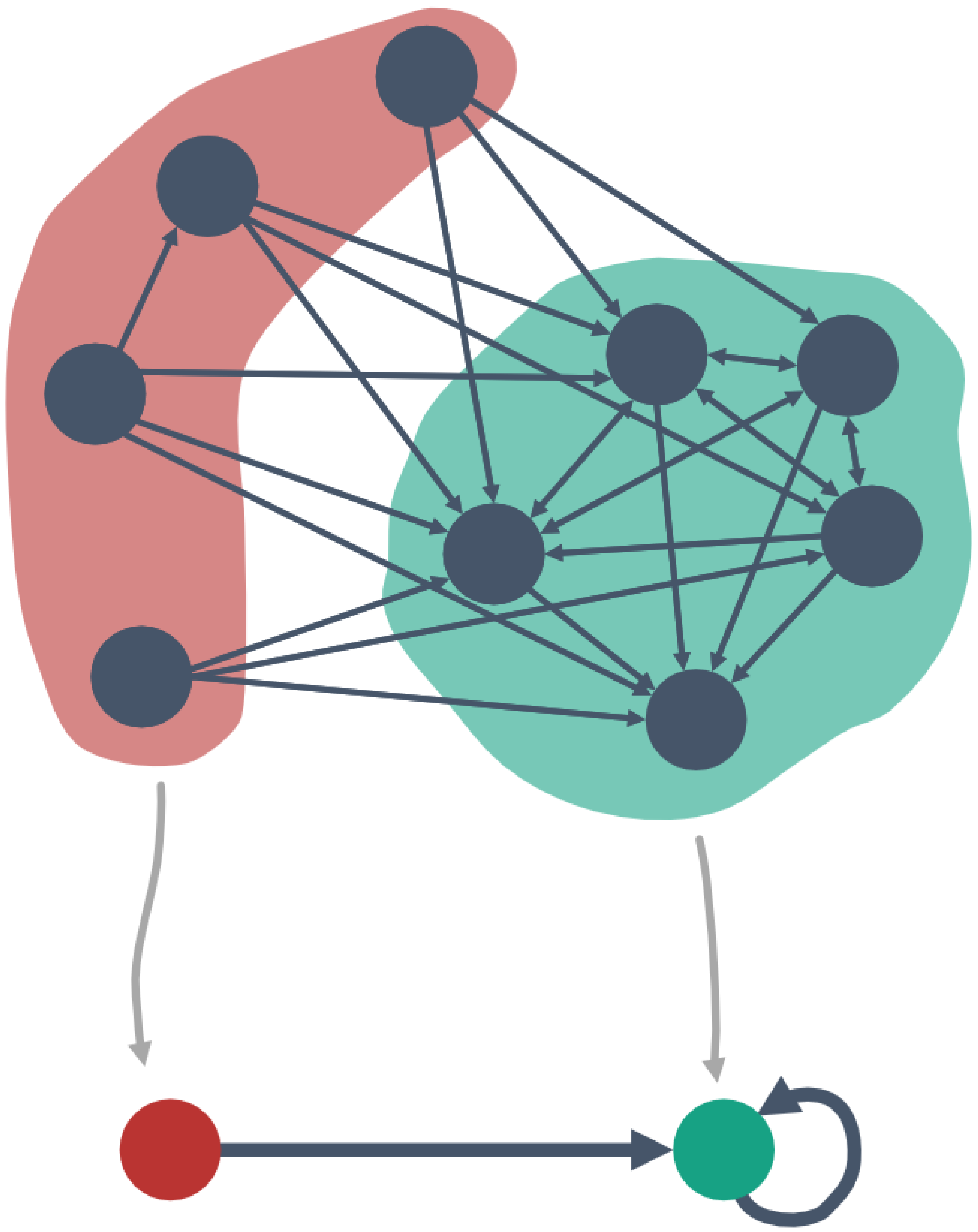


Nodes in a network operationalize units (e.g., individuals, organizations, countries).

They can have different properties.

Blockmodeling

With blockmodeling we can study the relationships between the units.



Blockmodeling is clustering approach for reducing large, potentially incoherent network to a smaller, comprehensible structure that is easier to interpret.

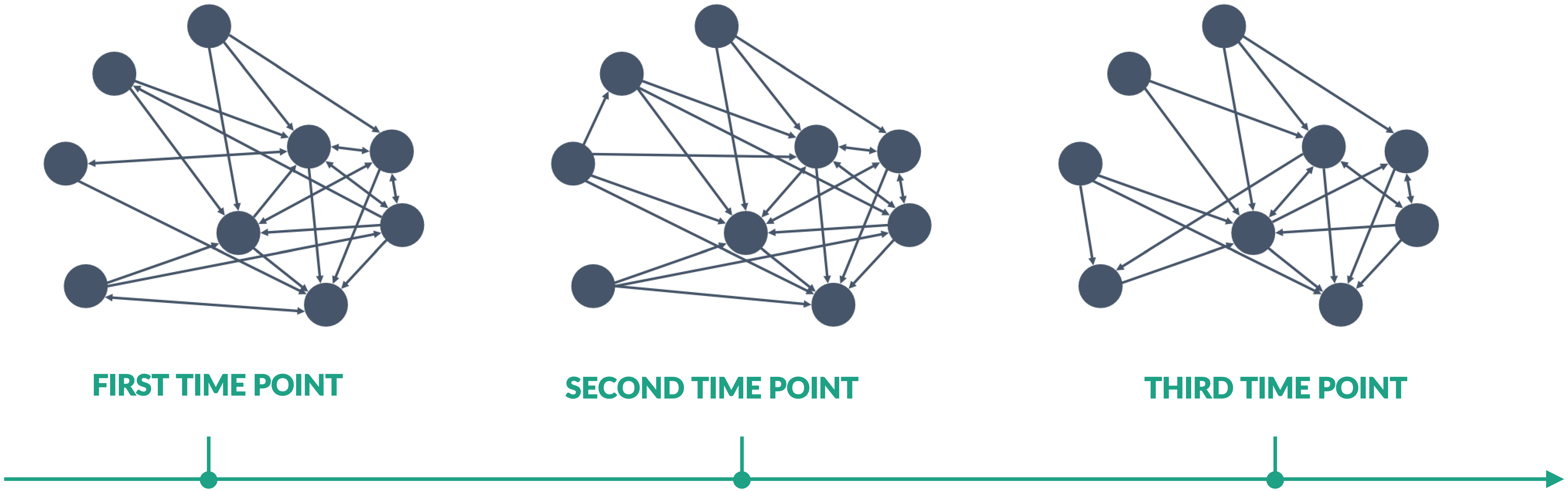
The result of blockmodeling is a partition of equivalent (according to their links in the network) nodes and an image matrix representing the links between and within the obtained clusters.

The term block refers to the links between two clusters and within one cluster.

Dynamic networks

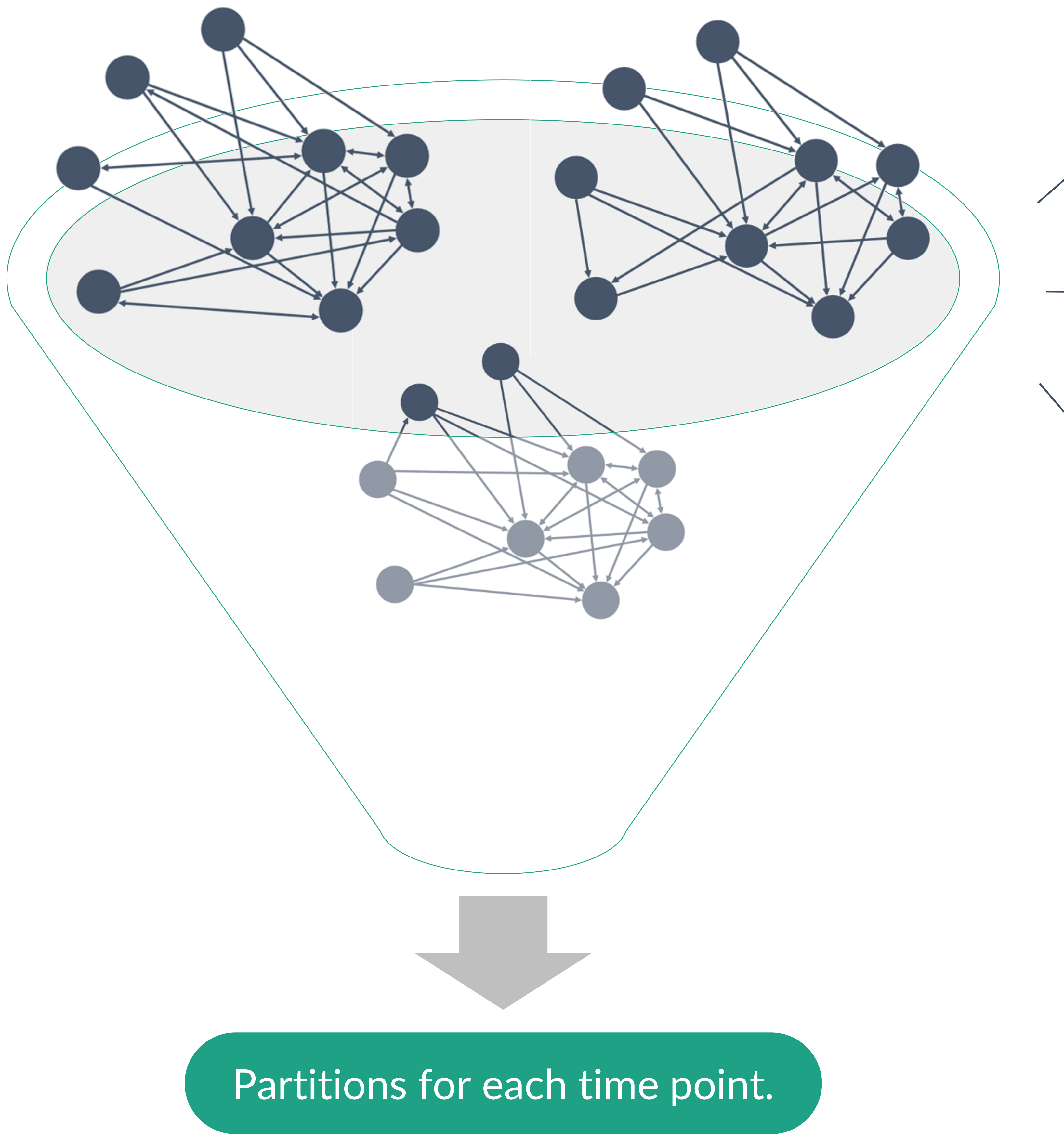
Several types of dynamic networks exists. Here is a focus on networks, measured at multiple points in time.

- Snapshot networks: most of nodes are present at all time points and the same relations are measured.
- Example: a survey of friendships among high school students in February, March and April.



Blockmodeling of dynamic networks

The idea is to take advantage of the fact that consecutively observed networks are dependent.



Validity

Considering dependency between the ties from different time points can increase validity of the results.

The aim

To identify equivalent groups and ties among the identified groups for each time point and to study how they change in time.

Recent development

Most approaches for blockmodeling dynamic networks were developed in recent years.

Stochastic BM of dynamic networks

Selection of blockmodeling approaches is limited to those implemented in R.

SBMfDN

Matias & Miele (2016)

Statistical clustering of temporal networks through a dynamic stochastic block model

ESBMfDN

Bartolucci & Pandolfi (2020)

An exact algorithm for time-dependent variational inference for the dynamic stochastic block model

SBMfMPN

Bar-Hen et al. (2020)

Block models for generalized multipartite networks

SBMfLN

Škulj & Žiberna (2021)

Stochastic blockmodeling for linked networks

KBMfLN

Žiberna (2020)

K-means-based algorithm for blockmodeling linked networks

Stochastic blockmodeling: assume an underlying statistical model and estimate it by maximizing some likelihood-based measure. A model enables statistical inference.

Deterministic blockmodeling: iterative algorithm search for homogenous blocks in term of tie values.

Conditional cluster probabilities: cluster probabilities in a current time point depend on cluster membership in a previous time point(s).

Linked and multipartite networks: a collection of at least two one-mode networks and one two-mode network linking these one-mode networks. In the context of dynamic networks, the two-mode networks "link" the same units from different time points. Such network is blockmodeled as a single network (with the restriction that nodes from different one-mode networks can not mix).

Within group ties probabilities are fixed in time.

Exact version of SBMfDN. Blockmodel type is fixed in time (as currently implemented in R).

Like SBMfMPM expect they enable weighting different parts (e.g., one-mode and two -mode) of a network and the estimation approach is slightly different.

The aim

Addressed by Monte Carlo simulations.

Empirically compare blockmodeling approaches.

Evaluate sensitivity to the basic network characteristics.

Propose guidelines for choosing blockmodeling approaches.

1

NETWORKS WITH DIFFERENT PROPERTIES

Different network characteristics are considered, such as network size, blockmodel type, etc.

2

NETWORKS LOOK LIKE REAL WORD NETWORKS

The networks are generated by considering local network mechanisms which makes them closer to the real-world networks.

3

KNOWN BLOCKMODEL TYPES AND PARTITIONS

The networks are generated such that blockmodel types and partitions are known. Both can change in time.

Considered factors

Detailed descriptions follow on the next slides.

NETWORK SIZE

Small (48 nodes) and
large (96 nodes) networks.

BLOCK DENSITIES

Low and high differences between null and complete block densities.

GROUPS' STABILITY

Nodes can change group membership.

MECHANISMS

Inconsistencies are generated randomly or by local mechanisms.

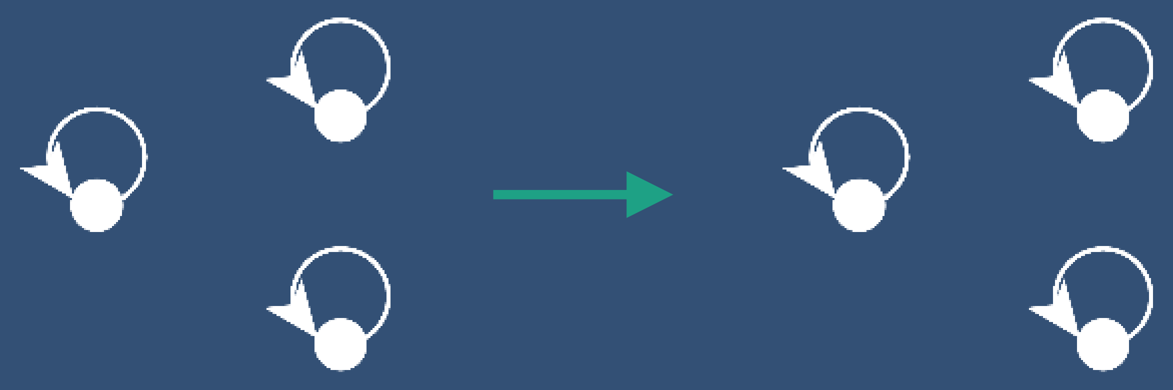
BLOCKMODEL TYPES

They remain the same or change in time.

Three groups are in all generated networks.

BLOCKMODEL TYPES

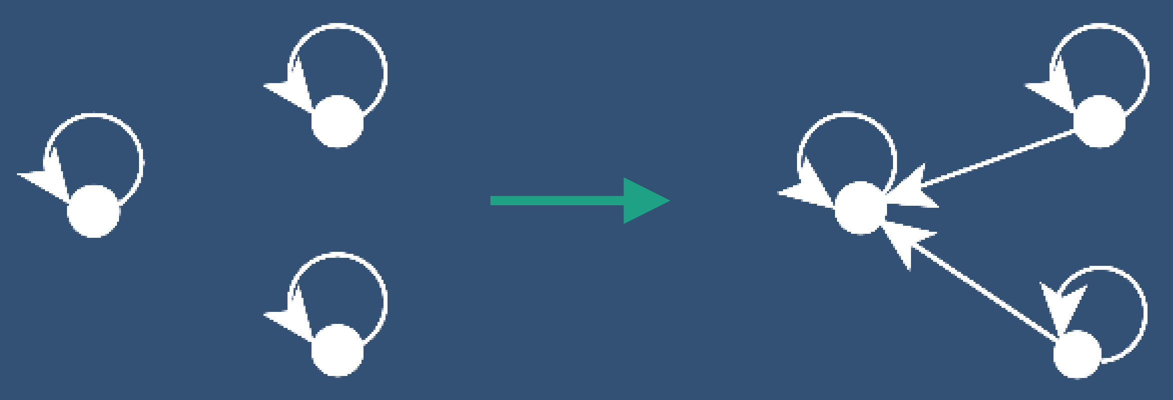
The three most essential blockmodel types and three types of transitions between them are assumed. Some transitions imply a minor change in the global network structure while some imply a major change.



NO CHANGE

The cohesive blockmodel type remains at both time points.

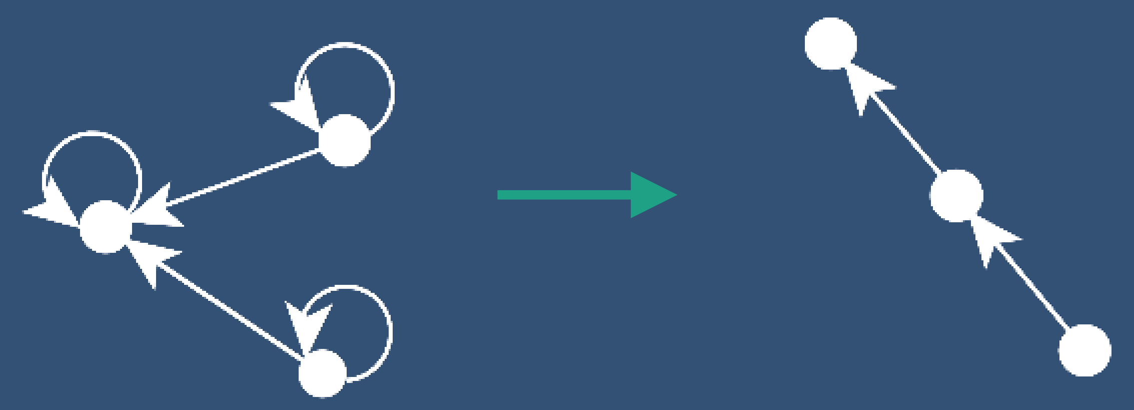
cohesive → cohesive



MINOR CHANGE

The nodes in one group establish links to all the other nodes.

cohesive → core-cohesive



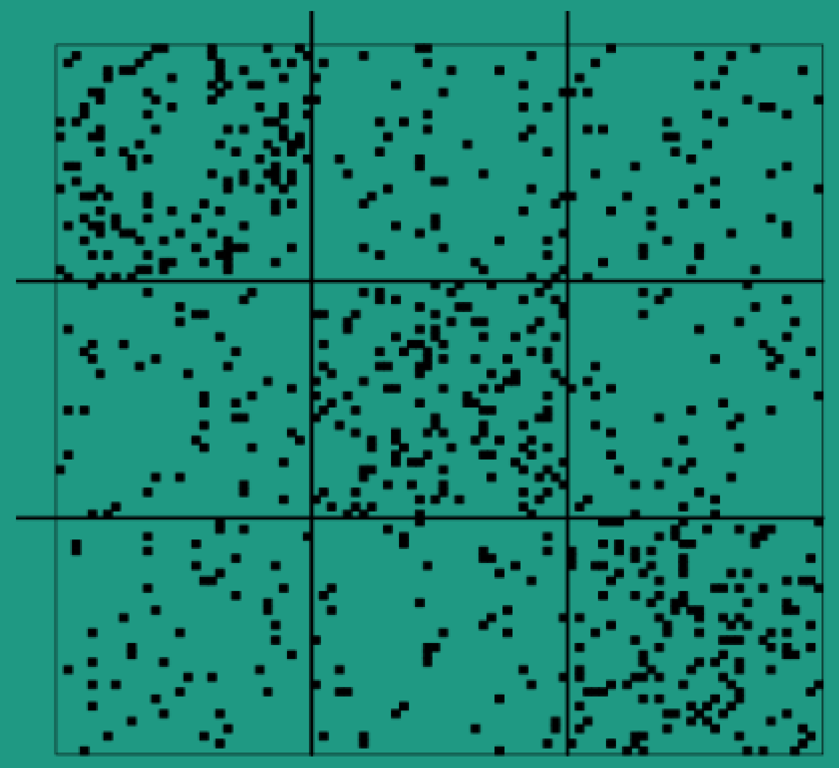
MAJOR CHANGE

The links within clusters dissolve, hierarchical structure emerges.

core-cohesive → hierarchical

BLOCK DENSITIES

Densities in null blocks are set to 0.05 for all generated networks.
Densities in complete blocks are set to 0.15 in some and 0.20 in other generated networks.



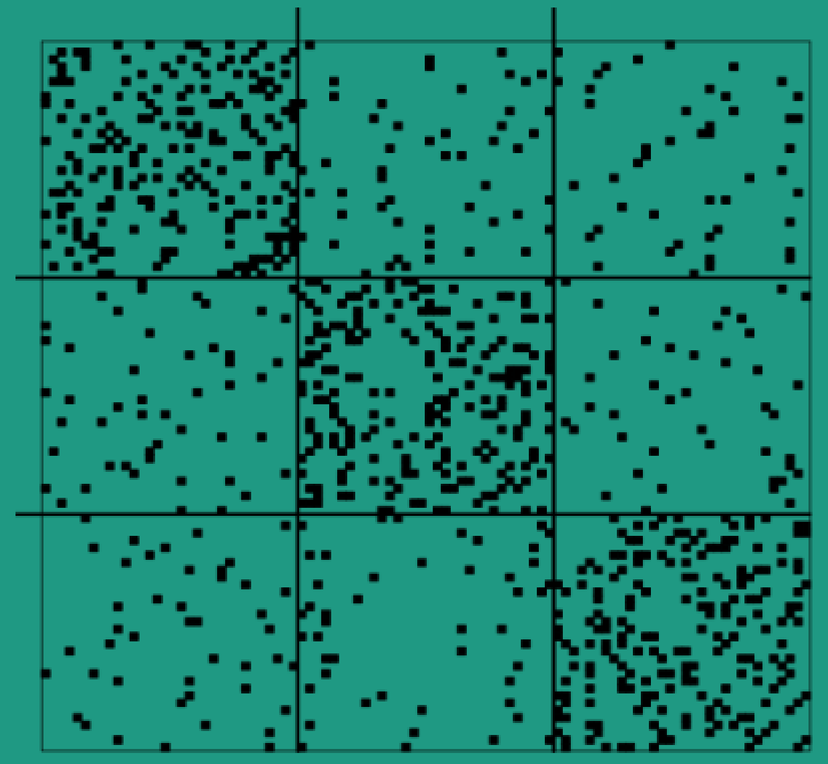
EXAMPLE OF NETWORK WITH COHESIVE BLOCKMODEL

Low difference between the density of null and complete blocks

0.05 (null)
0.15 (complete)

High difference between the density of null and complete blocks

0.05 (null)
0.20 (complete)



EXAMPLE OF NETWORK WITH COHESIVE BLOCKMODEL

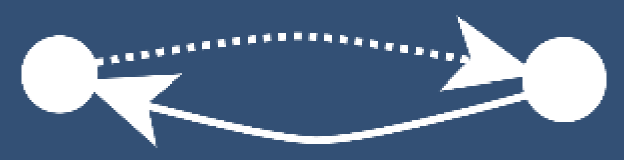
GROUPS' STABILITY

The selected number of pairs are relocated between the clusters at each time period.
This does not affect cluster sizes.

Groups' stability	Percentage of relocated pairs between the cluster			Adjusted Rand Index
	TP 1 vs TP 2	TP 1 vs TP 3	TP 1 vs TP 4	TP 1 vs TP 4
Constant	0	0	0	1.00
Stable	3	7	10	0.72
Unstable	7	13	20	0.51
Random	33	66	100	0.00

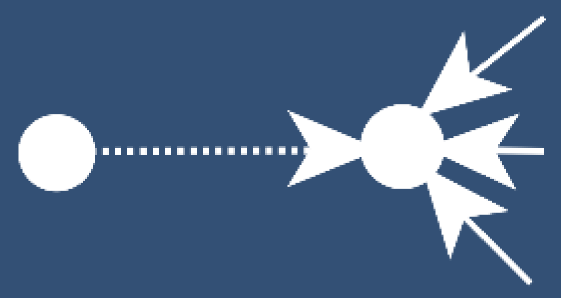
MECHANISMS

The links within blocks can be generated completely at random or based on the selected local network mechanisms (all mechanisms are assumed to have similar strengths reflected by the vector θ).



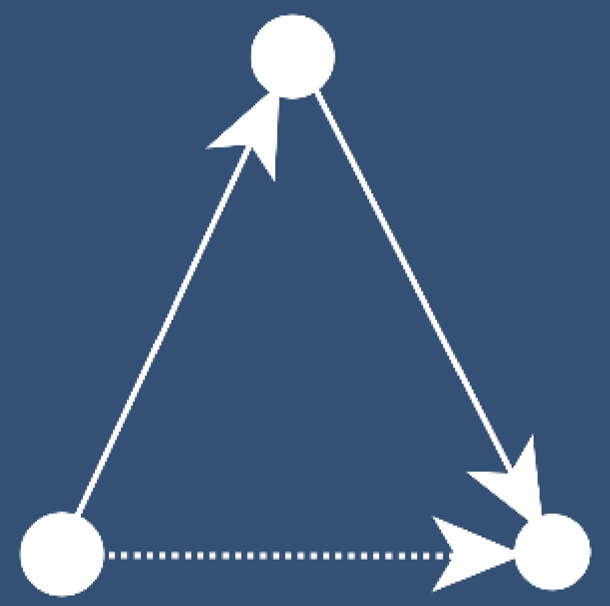
MUTUALITY

Tendency to reciprocate links.



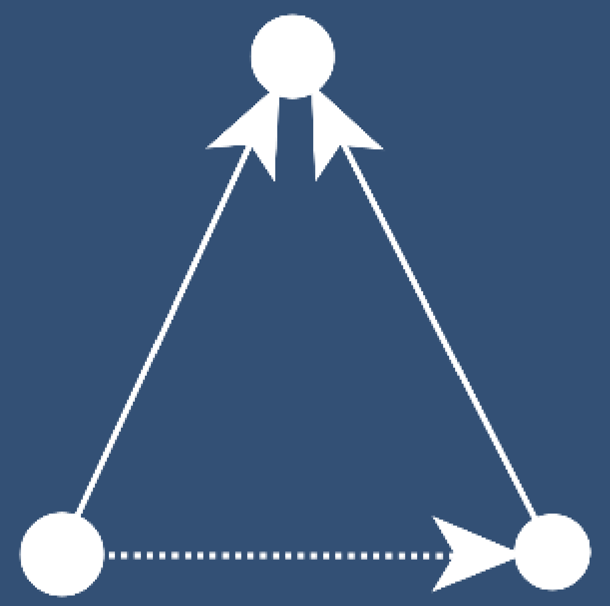
POPULARITY

Tendency to create links to those with the highest in-degree.



TRANSITIVITY

Tendency to create links to those who are “liked by a friend”.

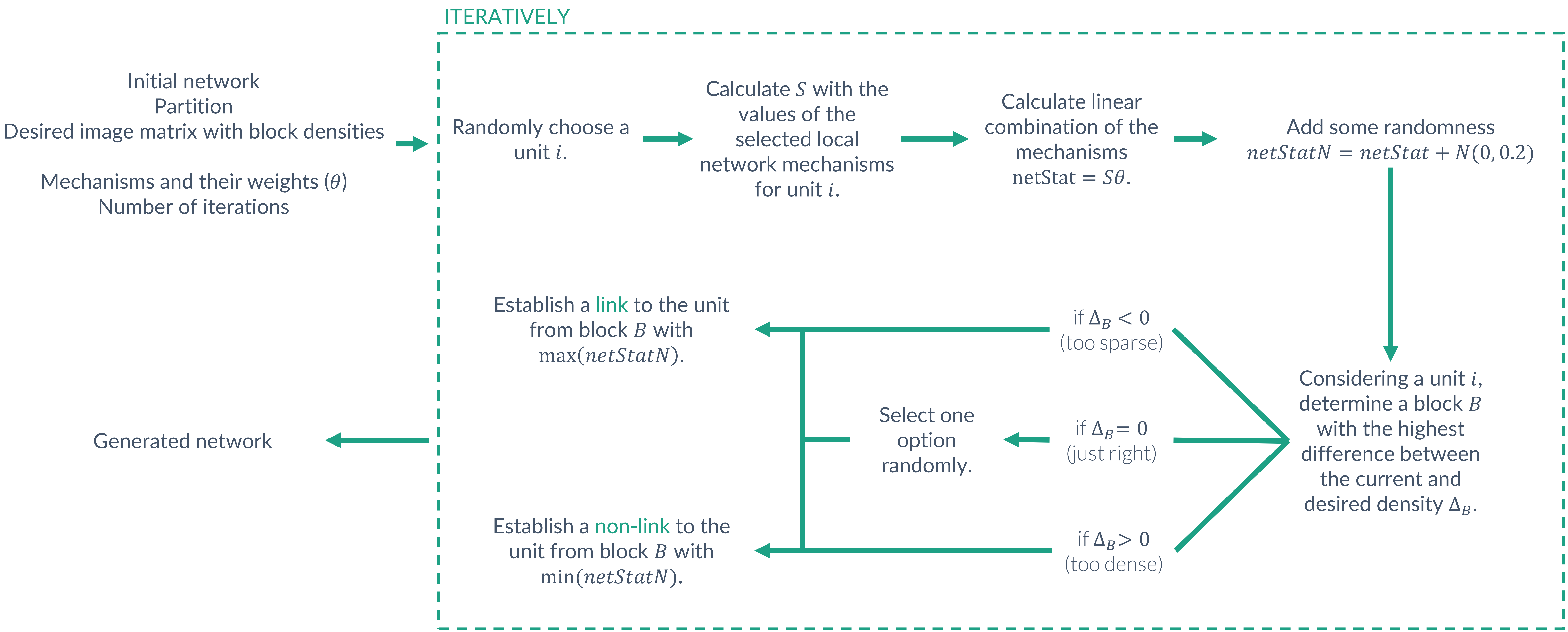


OUTGOING-SHARED PARTNER

Tendency to create links to those who “like the same others”.

Generating networks

The 2,500 iterations were used.



Generating temporal networks

The algorithm for generating networks was used forth times for each temporal network.

EMPTY NETWORK

↓ APPLY ALGORITHM FOR GENERATING NETWORKS ↓

FIRST TIME NETWORK → APPLY ALGORITHM FOR GENERATING NETWORKS → SECOND TIME NETWORK → APPLY ALGORITHM FOR GENERATING NETWORKS → THIRD TIME NETWORK → APPLY ALGORITHM FOR GENERATING NETWORKS → FORTH TIME NETWORK

FIRST OBSERVATION

The first blockmodel type with the pre-specified block densities.

INTERMEDIATE OBSERVATIONS

The intermediate block densities are calculated and used. Linear change is assumed.

Example of block densities from the first to the last time point:
0.20 → 0.15 → 0.10 → **0.05**

LAST OBSERVATION

The second blockmodel type with the pre-specified block densities.

Separate blockmodeling approaches

Networks from each time points are blockmodeled separately.

STOCHASTIC	KMEANS	SBMfLN*
Mariadassou et al. (2010)	Žiberna (2020)	Škulj & Žiberna (2021)
Bernoulli stochastic blockmodeling	**K-means based blockmodeling**	**Stochastic blockmodeling for linked networks**
`BM_Bernoulli` (`blockmodels`)	`kmBlockORPC` (`kmBlockTest`)	`stochBlockORP` (`StochBlockTest`)
`explore_min = 10` `explore_max = Inf`	`rep = 1000`	`rep = 1000`

Temporal blockmodeling approaches

Default and manual initial partitions are considered.

SBMfDN	ESBMfDN	SBMfMPN	SBMfLN	KBMfLN
Matias & Miele (2016)	Bartolucci & Pandolfi (2020)	Bar-Hen et al. (2020)	Škulj & Žiberna (2021)	Žiberna (2020)
Statistical clustering of temporal networks through a dynamic stochastic block model	**An exact algorithm for time-dependent variational inference for the dynamic stochastic block model**	**Block models for generalized multipartite networks**	**Stochastic blockmodeling for linked networks**	**K-means-based algorithm for blockmodeling linked networks**
`select.dynsbm` `estimate.dysbm` `(dynsbm)`	`est_var_dyn_exact`	`multipartiteBMFixedModel` `(GREMLINS)`	`stochBlockORP` `(StochBlockTest)`	`kmBlockORPC` `(kmBlockTest)`
`iter.max = 20` `nstart = 25`	`maxit = 1000` `start = 0`	`maxiterVE = 1000` `maxiterVEM = 1000`	`rep = 1000`	`rep = 1000`
+ SBMfLN* 1. initial partition + SBMfLN* 2. initial partition	+ SBMfLN* initial partition	+ SBMfLN* initial partition	+ SBMfLN* initial partition	+ KMEANS initial partition

Evaluating results

Partitions are compared with the Adjusted Rand Index.

1

SELECT THE RESULT (DEFAULT VS. MANUAL INITIAL PARTITION)

Each approach produced two sets of results (for default initial partition and for manual initial partition). The one with the best (minimum or maximum) value of the optimized criterion are further analyzed.

2

EVALUATE THE OBTAINED PARTITIONS

The obtained partitions are compared to the true partitions with the Adjusted Rand index.

The mean ARI for all time points is interpreted.

Adjusted Rand Index is defined as the proportion of all possible pairs that are in the same cluster and all possible pairs in different clusters in both partitions (time points).

COMPARABILITY

ARI is comparable among the networks of different sizes and number of clusters.

PERFECT FIT

The ARI value equals 1 when the estimated partition and the true partition are the same.

RANDOM PARTITIONS

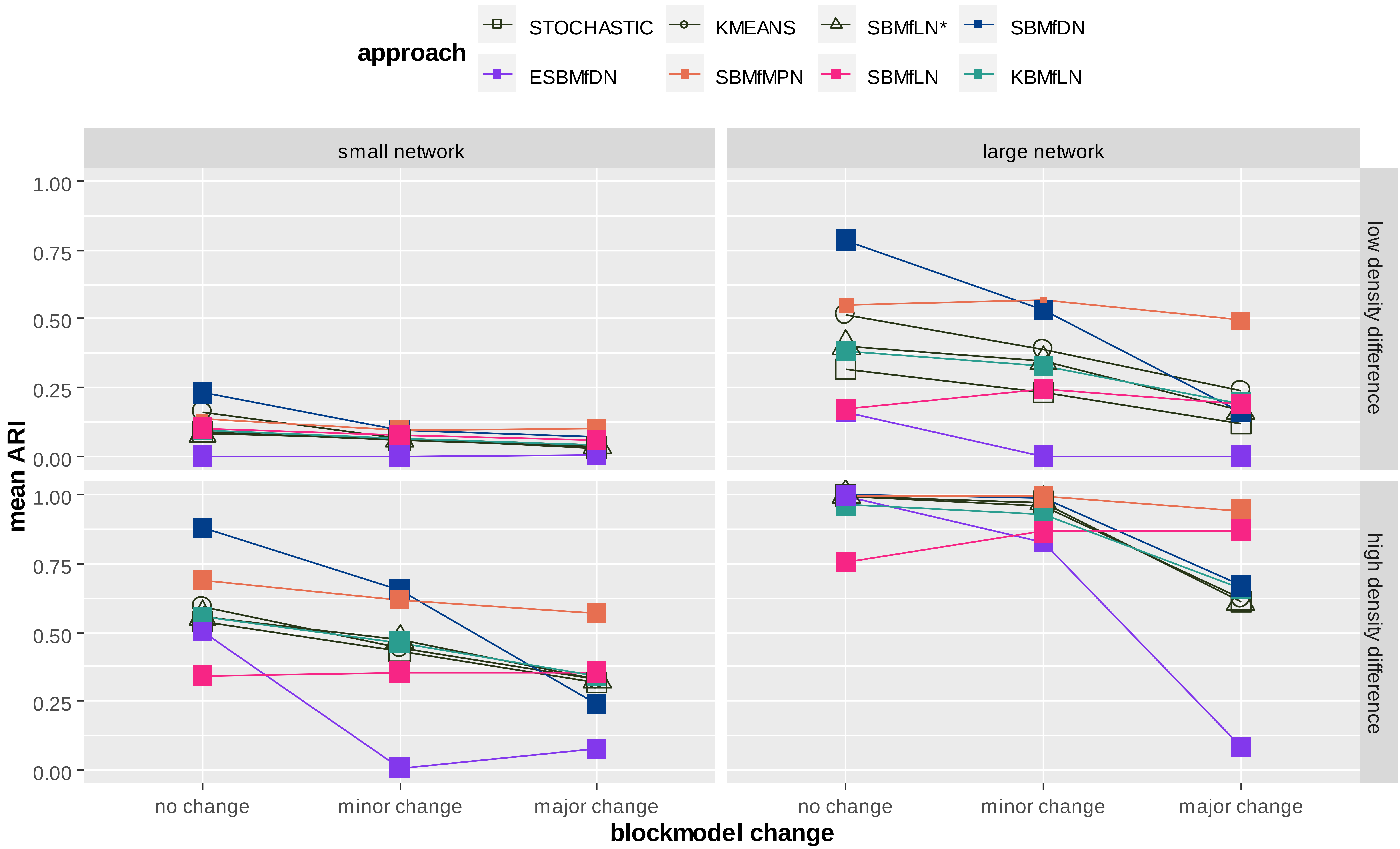
In the case of two random partitions, the expected value of ARI equals 0.

Results

The bellow summary is obtained over all simulation factors (i.e., also network size and mechanisms).

The problem is too hard for all blockmodeling approaches when the networks are small, and the density differences are low.

SBMfDN and SBMfMPN seems the most efficient.

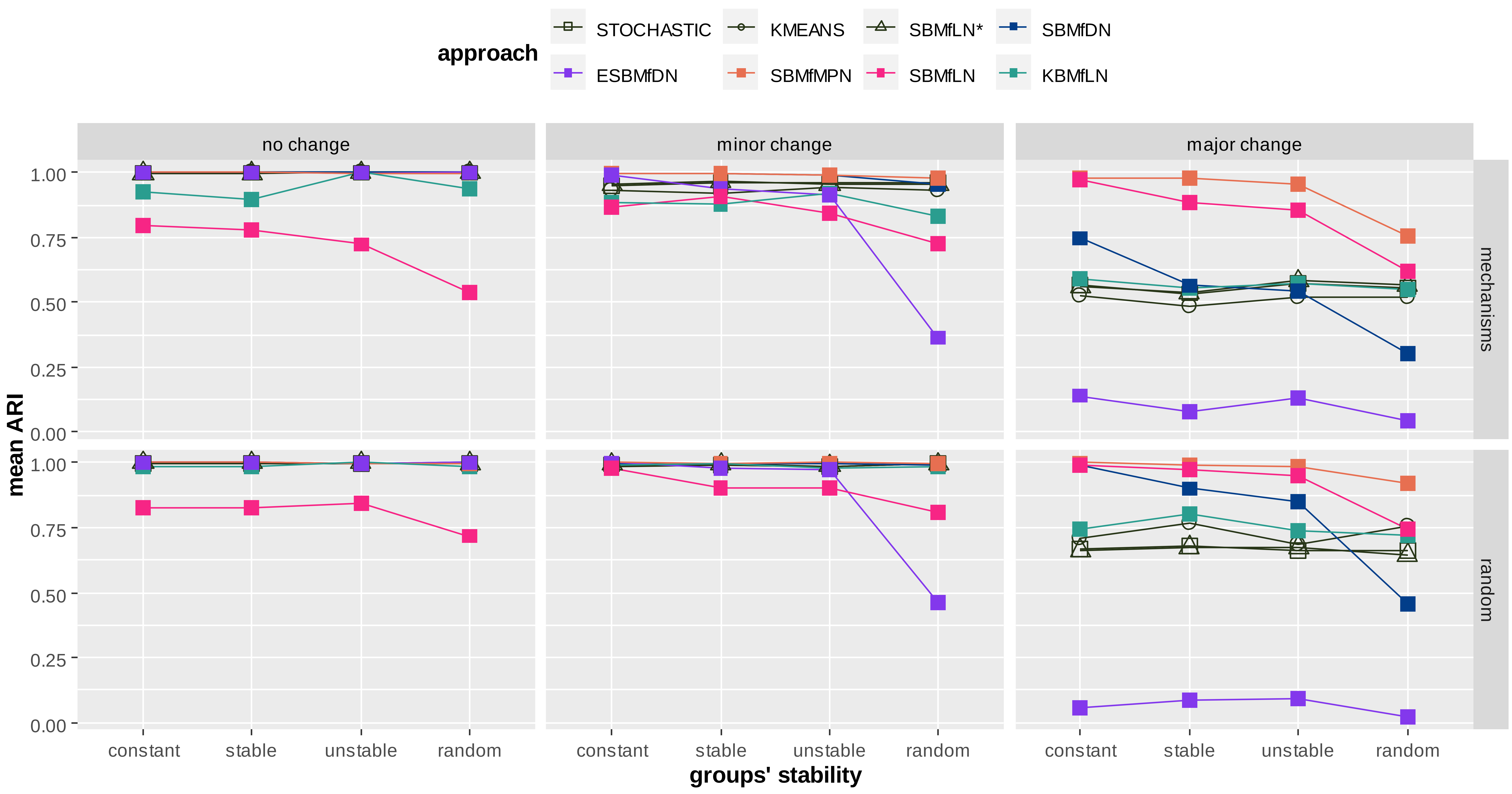


SBMfDN and SBMfMPN seems the most efficient. SBMfMPN does not converge in some cases.

The problem is relatively easy when the networks are large, and the density differences are high.

Large networks & high density difference

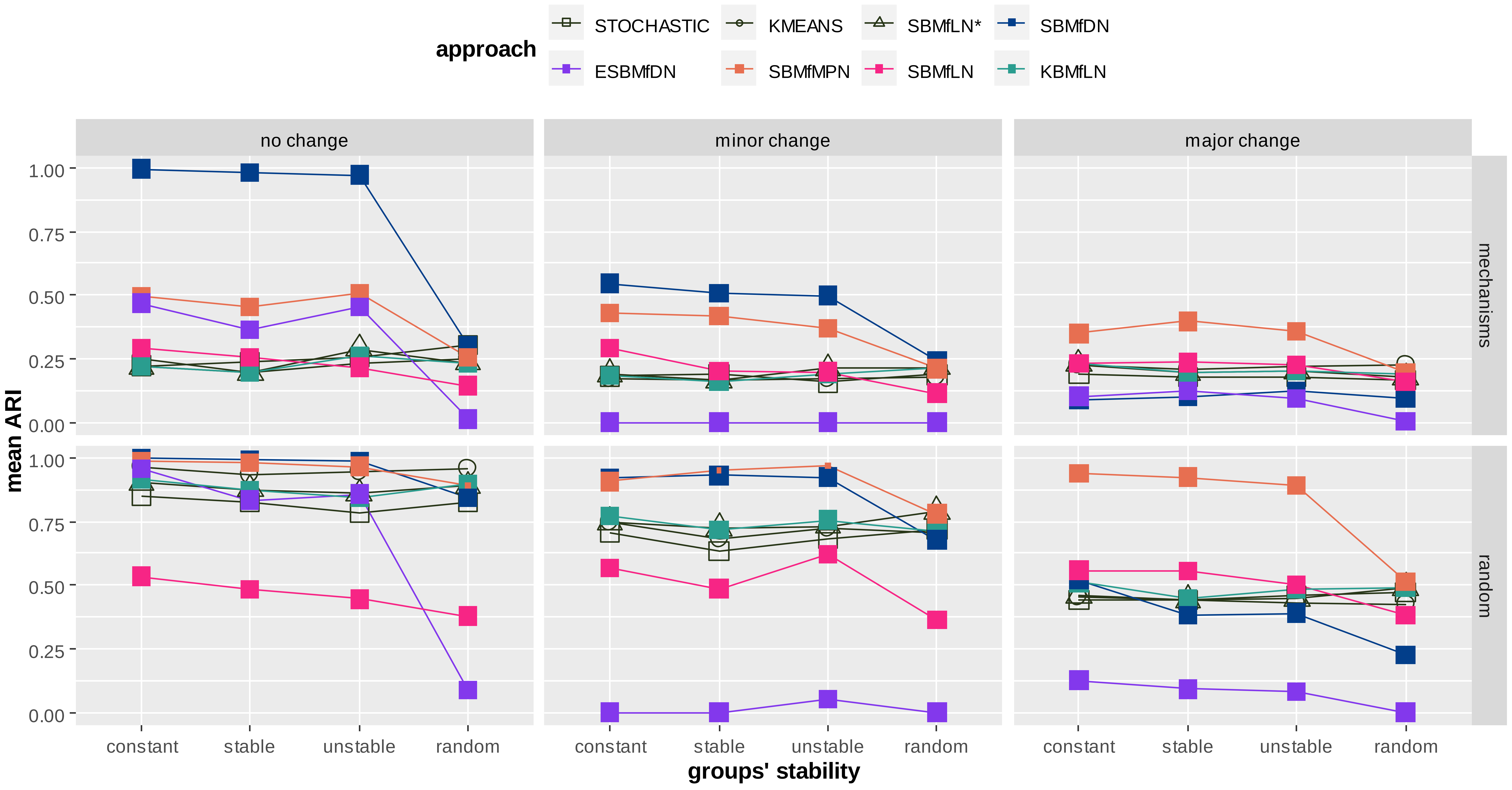
An easy problem for most approaches if the change in a blockmodel type is not major.



- All approaches provides fairly good results in the case of no or minor change of a blockmodel.
- The exception is SBMfLN which in overall gives not so good results, but it is less sensitive to the change of blockmodel type.
- SBMfMPN is the safest way to go.

Small network & high density difference

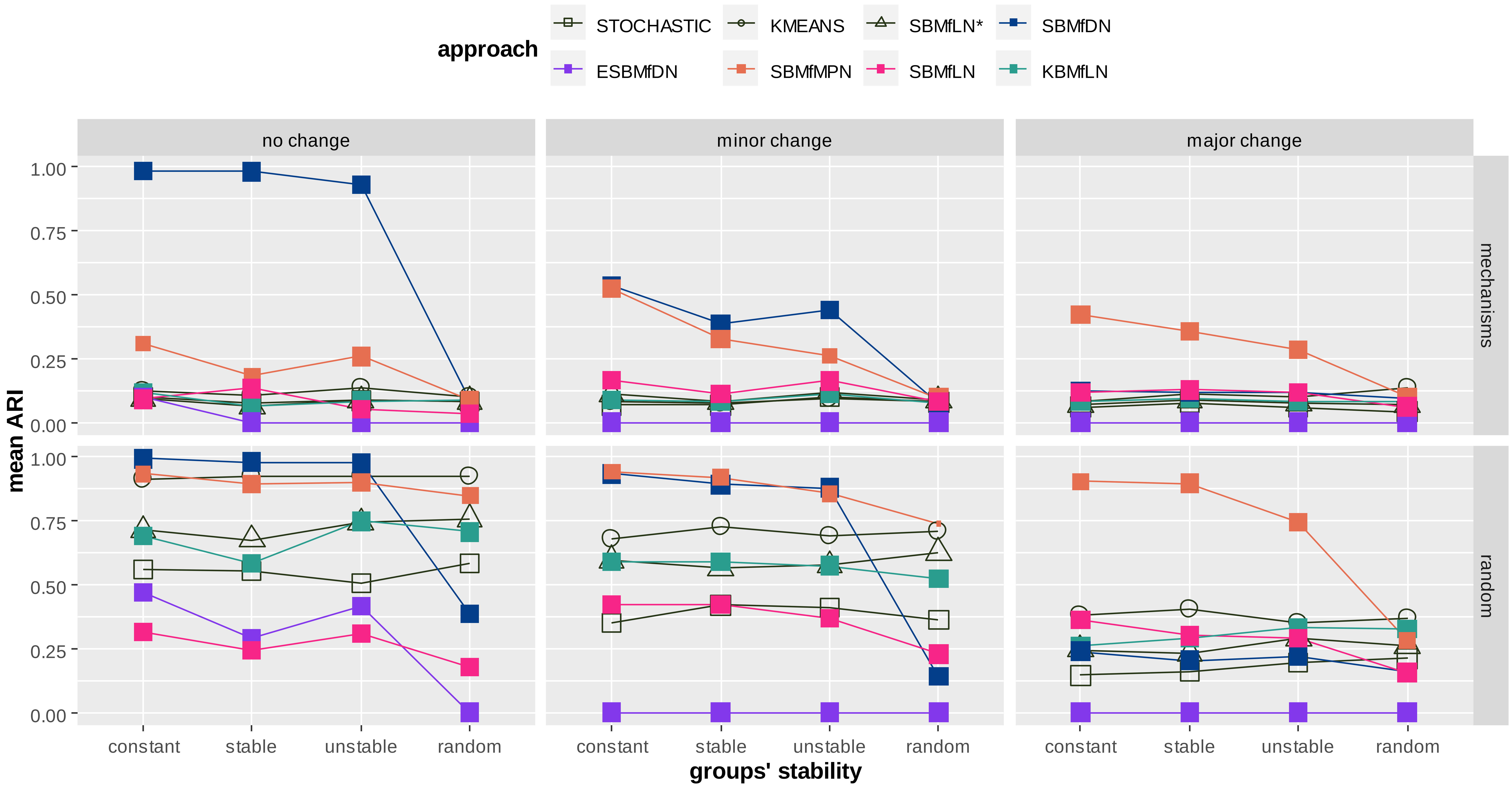
Several factors affect the efficiency of the methods.



- Approaches works better when the links within blocks are randomly generated.
- The change of a blockmodel type worsen the results.
- The stability of partitions affect all approaches which consider all time point simultaneously.
- SBMfDN and SBMfMPN generally produces the best results.

Large network & low density difference

Similar results as in the case of small networks and high density difference.



- SBMfDN produces the best results when there is no change in the blockmodel type (especially on the diagonal).
- The results of separate blockmodeling of networks for each time point are less sensitive to the stability of partitions.

  Yet, blockmodeling these networks simultaneously can bring benefits (especially) when there is not a lot of changes in a network.

CONCLUSION

This study attempt to compare the efficiency of different blockmodeling approaches. Overall, several factors (network size, blocks' densities, local network mechanisms, etc.) affect efficiency of blockmodeling approaches. Approaches that were not primarily developed for analyzing temporal networks works well in many cases.

01

PRIOR KNOWLEDGE & SEPARATE ANALYSES

Start with separate preliminary analyses of obtained networks to confirm your knowledge about the network. Various factors can affect the efficiency of blockmodeling approaches.

02

TRY WITH DIFFERENT INITIAL PARTITIONS

Use different initial partitions (e.g., from separate analysis) and keep the solution with the best criterion value.

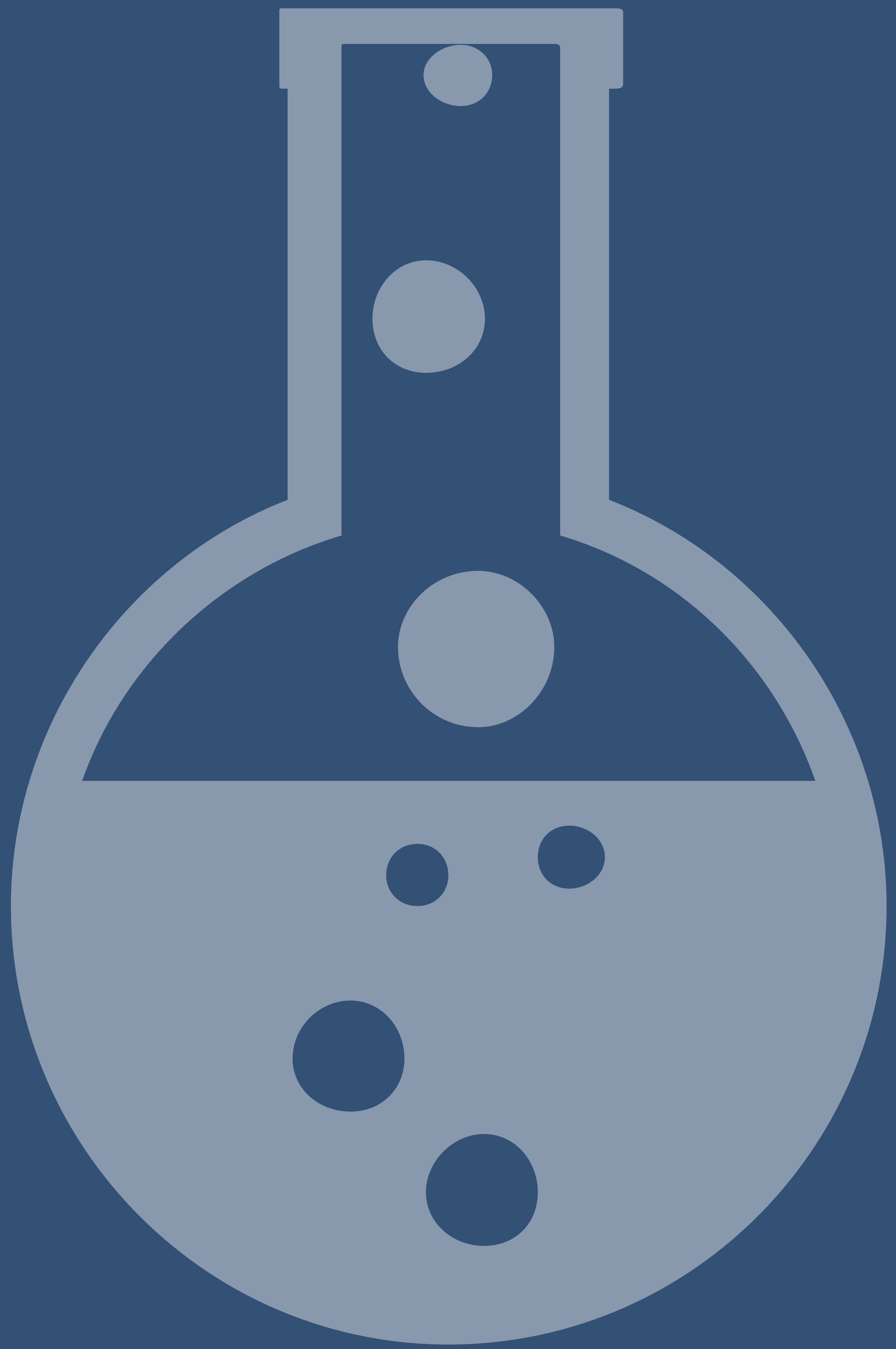
03

DON'T FORGET ON SBMfMPN (Bar-Hen et al.) AND SBMfDN (Matias & Miele)

The SBMfMPN with provide the best results if a major change of a blockmodel type is expected. SBMfDN is preferred in other cases.

Future work

The presented study will be extended.



ADDITIONAL FACTORS

New and departure nodes, different approaches to generating temporal networks (e.g, intermediate observations vs. additional observations), etc.

ADDITIONAL APPROACHES

Additional approaches and different initial partitions.

REAL NETWORKS

Comparison of different blockmodeling approaches on the real empirical networks.