Chapter 12

Dynamic Community Detection

Summary

- Communities in dynamic networks
- Evaluation & Benchmarking
- Visualization

Reading

- *"Challenges in community discovery on temporal networks."* Cazabet & Rossetti

Community Detection in Dynamic Networks

Time flies like an arrow; fruit flies like a banana

Communities
In Dynamic Networks

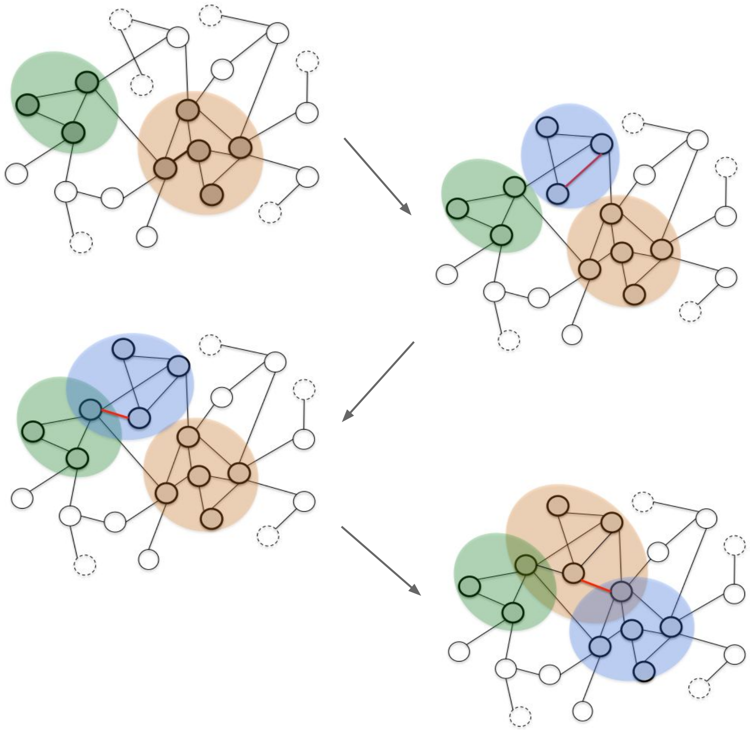
Networks change with time…

- Nodes appear and vanish
- Edges appear and vanish

…communities must change too!

DCD:

identify/track changes in community structure



Cazabet, Remy, and Giulio Rossetti. "Challenges in community discovery on temporal networks." Temporal Network Theory. Springer, Cham, 2019. 181-197.

A Novel Problem:

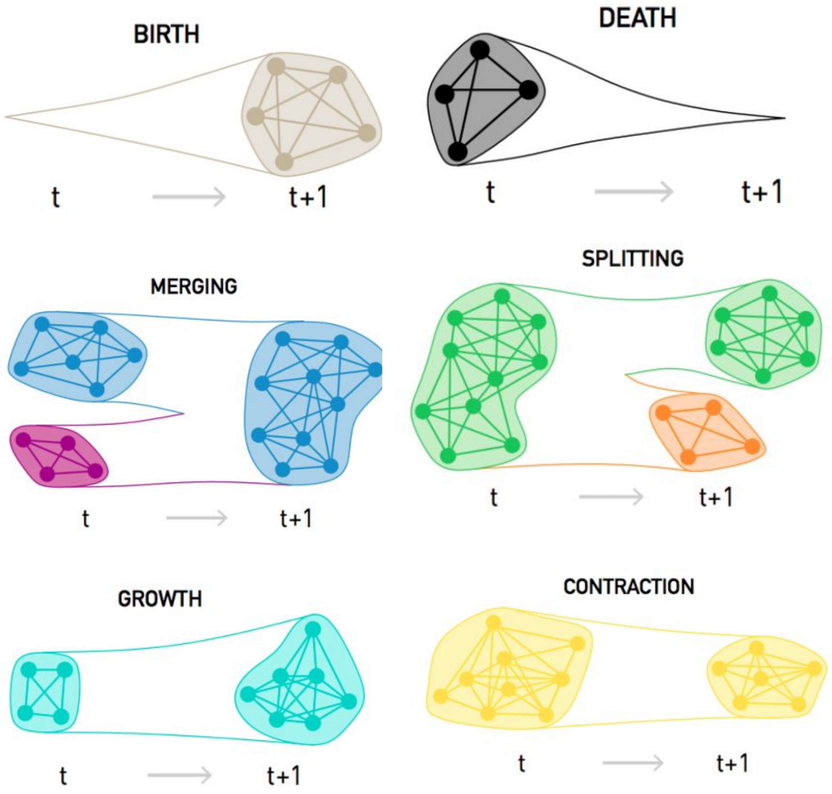
Community life-cycle tracking

As time goes by the rising of novel nodes and edges (as well as the vanishing of old ones) led to network perturbations

Communities can be deeply affected by such changes

Three main strategies:

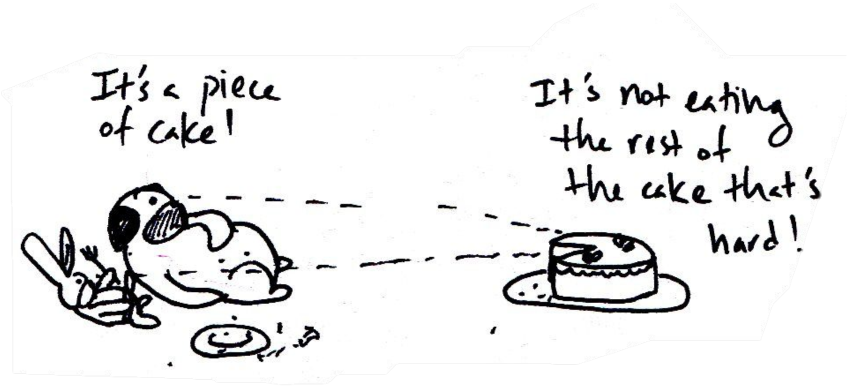
- Identify & Match
- Informed Iterative algorithms
- Stable Identification



The Optimist:

“Ok, It’s a piece of cake!”

1. Find communities at each network observation (using a static algorithm)
2. Match communities across consecutive network observations
3. Observe differences



Two major issues:

- Community Smoothing
- Theseus’ Ship Paradox

Community Smoothness

Communities are arbitrarily defined
(same issue of static CD)

Most “efficient” algorithms are stochastic

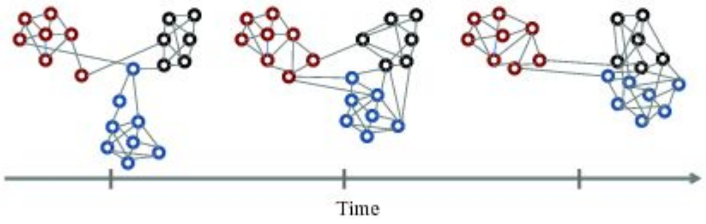
- Change in communities might be due to structural changes OR to arbitrary choices of the algorithm
- The same algorithm ran twice on the same graph *might yield* different results

Desiderata:

- a “simple” (parsimonious) model
- a trade-off between quality and simplicity (smoothness)

No Smoothness:
Partition at each t should be the same as found by a static algorithm

Smoothness:
Partition at t is a trade-off between “good” communities for the graph at t and similarity with partitions at different times



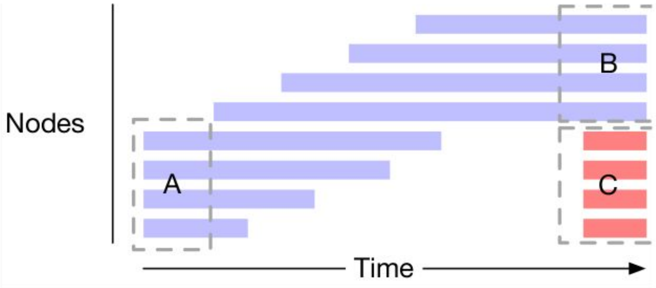
Theseus' Ship Paradox

I. Theseus killed the Minotaur in Crete and came back to Athens on his boat

II. His boat was conserved as memory during a very long time

III. The boat was deteriorating, so pieces of it were gradually replaced.

IV. Until one day, all original parts were replaced



Plutarch 75

Theseus' Ship Paradox

A. Is this ship still the same as Theseus boat ?

B. If another boat was built using all pieces of the original boat, which one would be the "real" Theseus boat ?

Community evolution/identity is an <u>arbitrary</u> concept

Fig. A - Ship of Theseus - Original

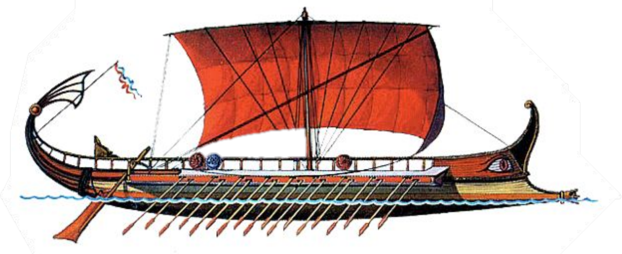
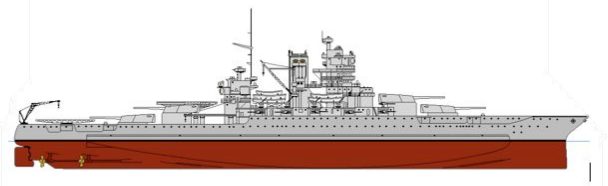
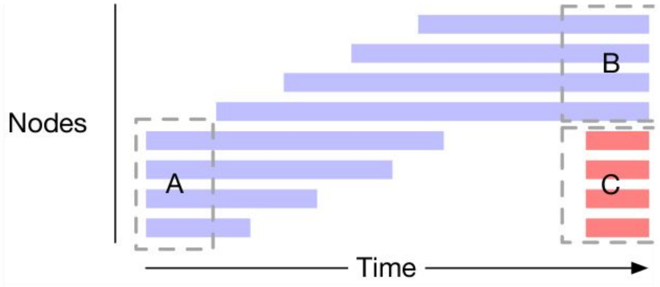


Fig. B - Ship of Theseus - Reconstructed



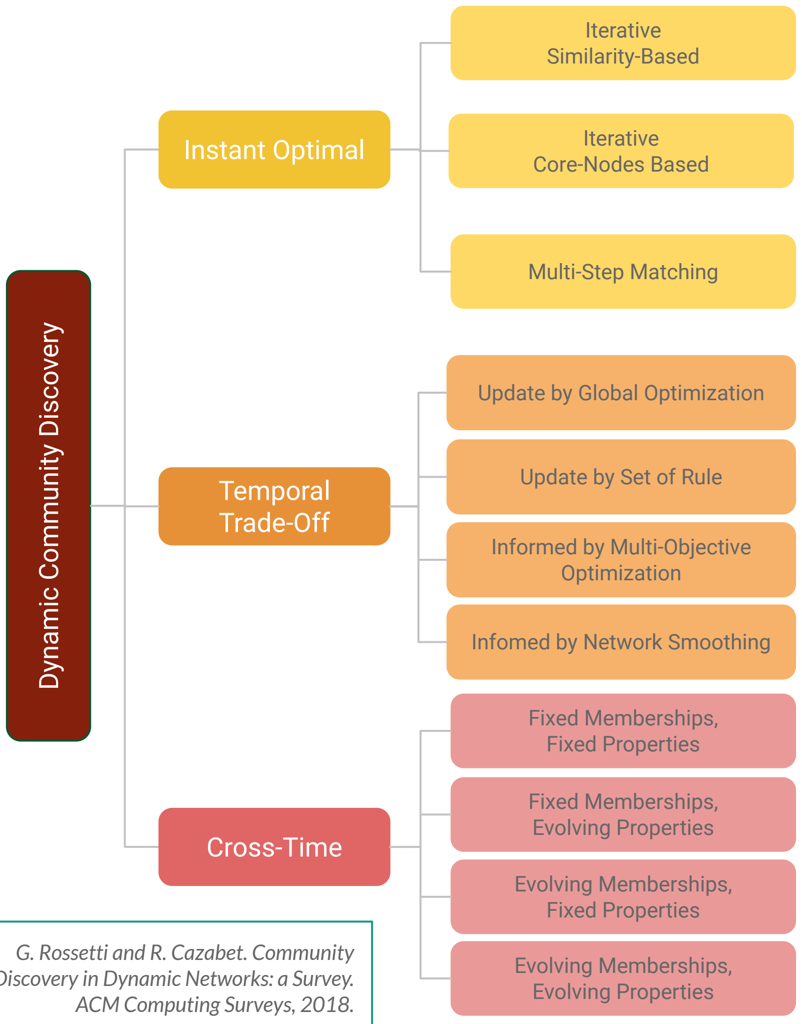


DCD Algorithms Taxonomy

Hierarchical categorization

First Level:
Increasing degree of smoothness *(none -> complete)*

Second Level:
Algorithmic Approach *(how to deal with Theseus)*



G. Rossetti and R. Cazabet. Community Discovery in Dynamic Networks: a Survey. ACM Computing Surveys, 2018.

Taxonomy

Instant Optimal

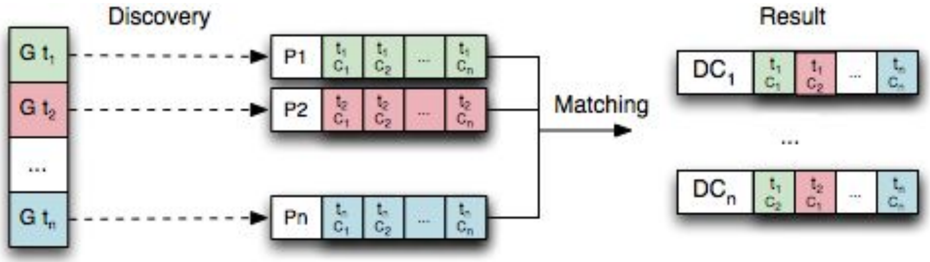
"Communities found at time t are optimal for the network at time t"

Strengths

Definition consistent with static CD, parallelisation

Drawbacks

Lack of smoothness, only Snapshot Network repr.



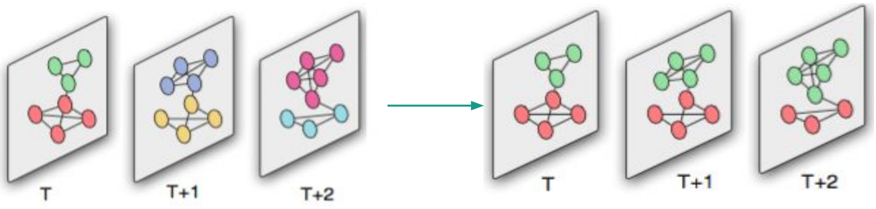
Taxonomy

Two-Step

1. Communities are detected at every step using a static algorithm (e.g. Louvain Algorithm)
2. Similarities are computed between communities in consecutive steps (at t and t+1 (e.g., Jaccard index))

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|}$$

3. Most similar communities are matched between t and t+1



Advantages:

- Easy to model, can extend smoothly static approaches

Drawbacks:

- The reduction to static scenarios trough temporal discretization is not always a good idea
  - How to choose the temporal threshold?
  - To what extent can we trust the obtained results?

Greene, et al. "Tracking the evolution of communities in dynamic social networks." 2010 international conference on advances in social networks analysis and mining. IEEE, 2010

Taxonomy

Temporal Trade-Off

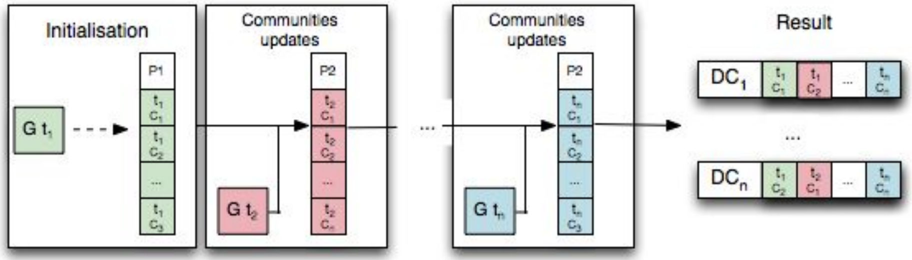
“Communities found at time t represent a trade-off between the graph at t and its previous states”

Strengths

Online, incremental, natural smoothness

Drawback

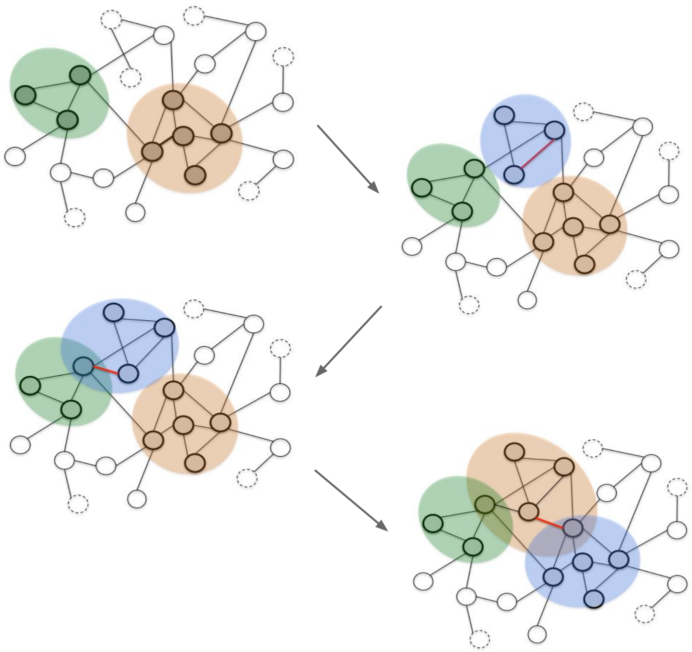
Iterative, risk of avalanche effect



Taxonomy

Tiles

1. Social Interactions define the communities a user belongs to
2. Dynamic graphs as *edge streams*
3. Online updates of communities as nodes/edges appear/vanish



Advantages:

- Punctual updates of the community structure
- Low computational complexity

Drawbacks:

- Ad-Hoc model

Rossetti, et al. "Tiles: an online algorithm for community discovery in dynamic social networks." Machine Learning 106.8 (2017): 1213-1241.

Taxonomy

Cross-Time

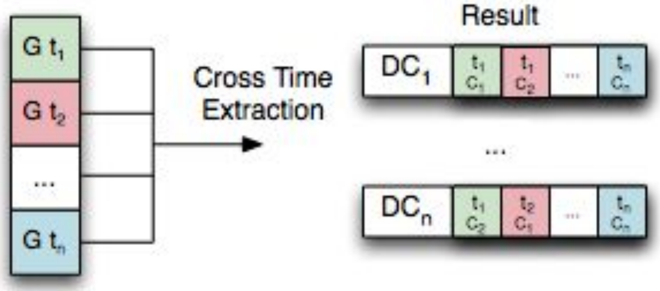
“Communities at t are defined relatively to all other steps”

Strengths

Perfectly smoothed, stable, solution

Drawback

Non online, batch computation, lacks incrementality

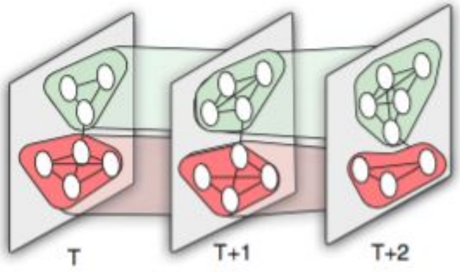


Taxonomy

Transversal Network

1. A transversal network is built: nodes are couples (nodes, time), edges link the same node in adjacent snapshots
2. A community detection algorithm is run on this transversal network

   (Note: modified Modularity to avoid overestimating expected edges between nodes in different time steps, i.e., custom random graph)



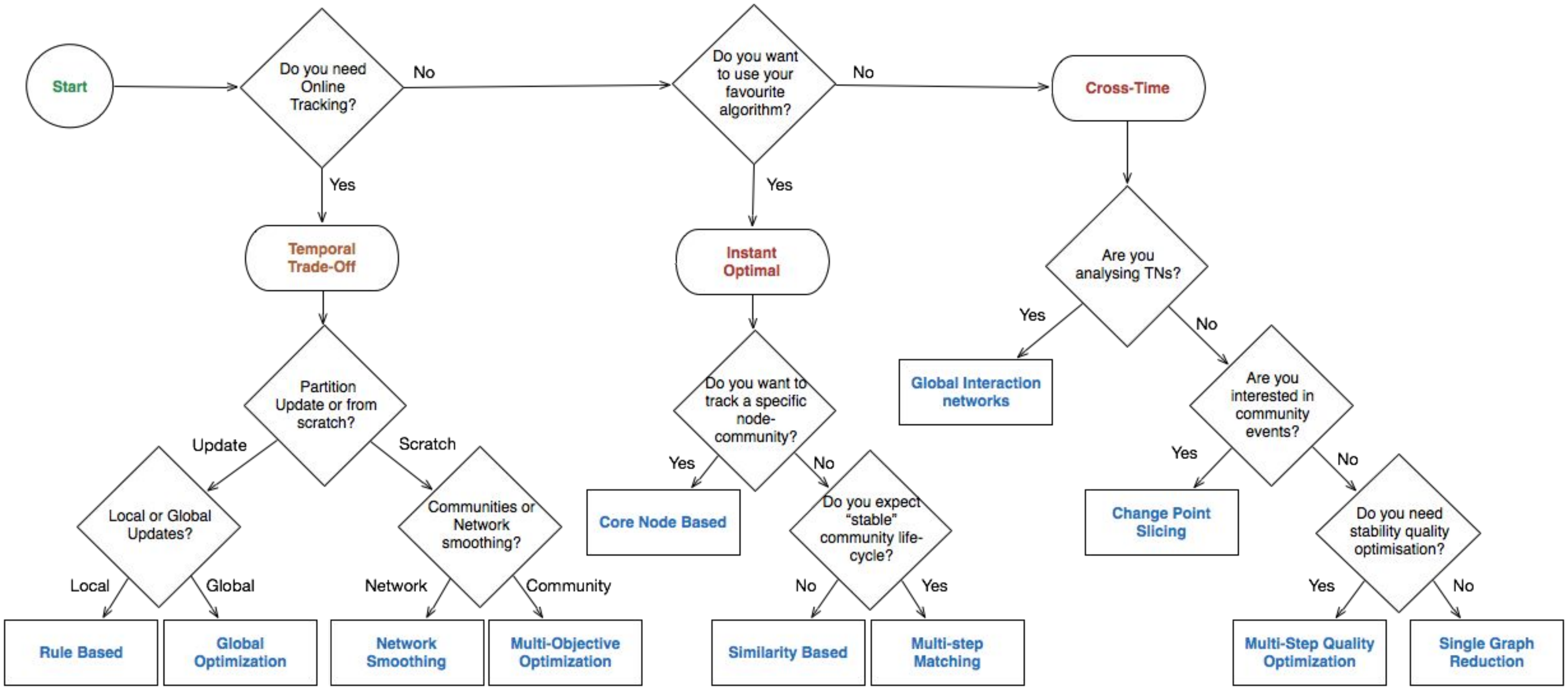
Advantages:

- Maximal smoothing and stability

Drawbacks:

- No Community Events are detected
- All the network history needs to be known in advance

Mucha, Peter J., et al. "Community structure in time-dependent, multiscale, and multiplex networks." science 328.5980 (2010): 876-878



Choosing the correct approach: one of many possible roadmaps…

Community Discovery in Dynamic Networks

Evaluation Strategies

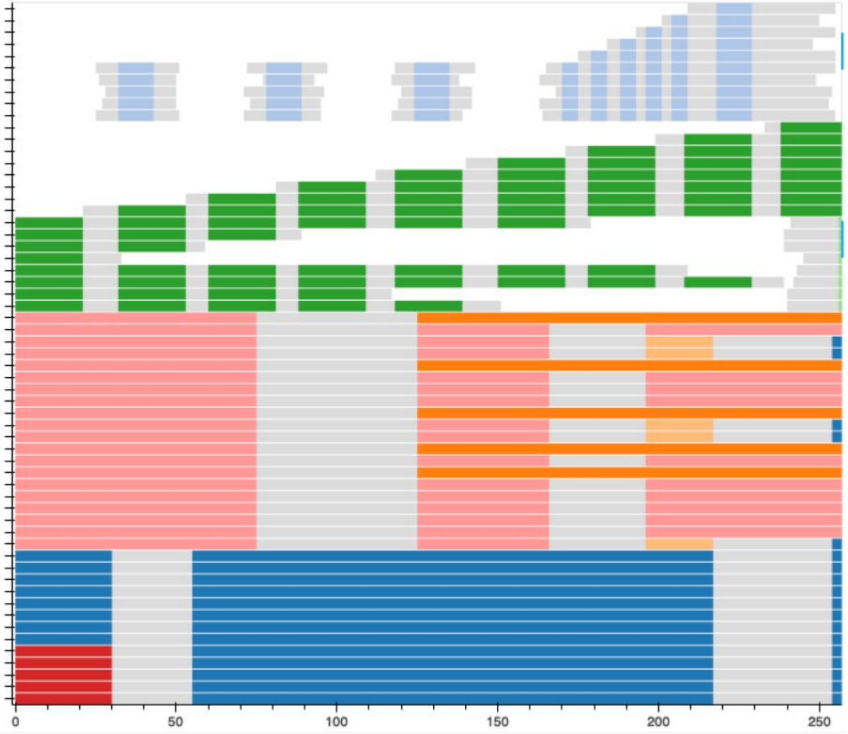
Strategies

Internal Evaluation

- Partition quality function
  (i.e., modularity, conductance, density…)
- Community characterization
  (i.e., size distribution, overlap distribution…)
- Execution time and Complexity

External Evaluation

- Ground truth testing
  (or partitions comparison)



Ground truth testing: Issues

- Few real world datasets with annotated ground truth partition are available
  (mostly static networks)
- Reliability of partition labelling
  (semantic partitions not always reflect topological ones)
- Scarcity of network generators handling community dynamics
  (i.e. birth, death, merge, split)

"I think we're past the point where rebooting will help."

Benchmark

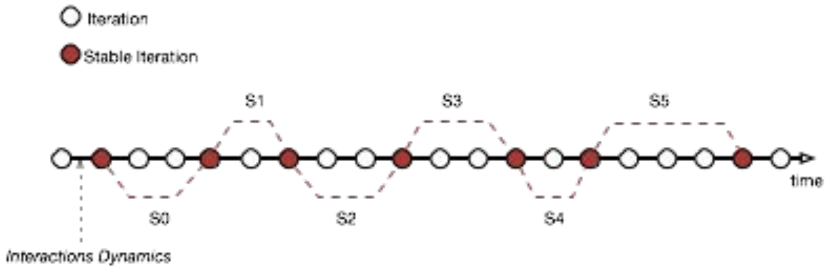
RDyn

Dynamic network generator that

- Guarantee power law degree distribution
- Guarantee small-world effect
- Exploit planted communities
  (having power law size distribution)
- Handle node/edge rise/fall
- Handle Community Dynamics (merge/split)
- Generate tunable-quality time-aware network partitions
  (i.e. conductance/modularity/density)

Expected outputs:

- Synthetic graph (TN/SN)
- Temporal communities with planted events



Events are handled by:

1. “Semantically” planting updated communities;
2. Converging to the final stable state by leveraging intra-community edge probability side effects

A state (iteration) is called stable when a minimum partition quality is reached (i.e. modularity/conductance/density)

G. Rossetti, RDyn: graph benchmark handling community dynamics. Journal of Complex Networks (2017)

Summarizing

Mesoscale Evolutions

Node/edge local dynamics affect community structures

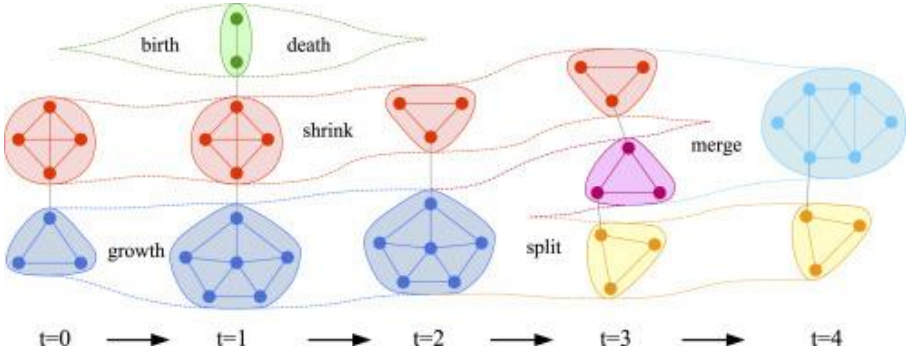
- Communities are subject to events/operations
- Life-cycles can be identified and studied

The complexity behind such ill posed problem grows

- Stability/Persistence
- Smoothness

Every family of approaches depend on

- Specific analytical needs
- Dynamic Network Representation adopted



Chapter 12

Conclusion

Take Away Messages

1. As topology evolve, community do too
2. Smoothness and stability are key issues
    a. Theseus ship paradox
3. Communities are subject to events
    a. Life-cycle tracking

Suggested Readings

- *"Challenges in community discovery on temporal networks."* Cazabet & Rossetti
- *"Community Discovery in Dynamic Networks: a Survey."* Rossetti & Cazabet

What's Next

Chapter 13:
Diffusion: Decision based models