DYNAMIC NETWORKS (Dynamic of networks)

• Most real world networks are dynamic

- ‣ Facebook friendship
	- People joining/leaving
	- Friend/Unfriend
- ‣ Twitter mention network
	- Each mention has a timestamp
	- Aggregated every day/month/year => still dynamic
- ‣ World Wide Web
- ‣ Urban network
- \mathbf{P} …

- Most real world networks are dynamic
	- ‣ Nodes can appear/disappear
	- ‣ Edges can appear/disappear
	- ‣ Nature of relations can change
- How to represent those changes?
- How to manipulate dynamic networks?

Dynamic Network Properties

Independently of the studied data, dynamic networks can have various properties:

- **Edge** presence can be **punctual** or **with duration**
- \cdot **Node** presence can be unspecified, punctual or contin**uous**
- If **time is continuous**, it can be **bounded** on a period of analysis or **ubounded**
- If **nodes** have attributes, they can be **constant** or **timedependent**
- If **edges** have weights, they can be **constant** or **timedependent**

SEVERAL FORMALISMS

TEMPORAL NETWORK

Collected dataset, for instance in (t,u,v) format

Examples: -SocioPatterns -Enron

 $-$ ……

TEMPORAL NETWORK

Snapshots Link Stream Interval Graph

Figure 1: Simple examples of stream graphs and link streams. Left: a stream

Vocabulary

Many different names have been used to for networks changing with time, but there is no broad consensus in the literature on the meaning of those terms, unless they are used with an explicit reference to a paper defining those terms. Here is a list of the most popular:

- **Dynamic Networks** and **Dynamic Graphs**
- **Longitudinal Networks**
- **Evolving Graphs**
- **Link Streams** & **Stream Graphs** (Latapy, Viard, and Magnien 2018)
- **Temporal Networks**, **Contact Sequences** and **Interval Graphs** (Holme and Saramäki 2012)
- **Time Varying Graphs** (Casteigts et al. 2012)

ANALYZING DYNAMIC NETWORKS

ANALYZING DYNAMIC NETWORKS

- Few snapshots
- Slowly Evolving Networks (SEN)
- Degenerate/Unstable temporal networks

FEW SNAPSHOTS

FEW SNAPSHOTS

- The evolution is represented as a series of *a few* snapshots.
- Many changes between snapshots
	- ‣ Cannot be visualized as a "movie"

FEW SNAPSHOTS

- Each snapshot can be studied as a static graph
- The evolution of the properties can be studied "manually"
- "Node X had low centrality in snapshot t and high centrality in snapshot t+n"

SLOWLY EVOLVING NETWORKS (SEN)

- Edges change (relatively) slowly
- The network is well defined at any t
	- ‣ Nodes/edges described by (long lasting) intervals
	- ‣ Enough snapshots to track nodes
- A static analysis at every (relevant) t gives a dynamic vision
- No formal distinction with previous case (higher observation frequency)

- Visualization
	- ‣ Problem of stability of node positions

Leskovec, Jure, Jon Kleinberg, and Christos Faloutsos. "Graph evolution: Densification and shrinking diameters." *ACM Transactions on Knowledge Discovery from Data (TKDD)* 1.1 (2007): 2.

• Centralities

TIME SERIES ANALYSIS

- TS analysis is a large field of research
- Time series: evolution of a value over time
	- ‣ Stock market, temperatures…
- "Killer app":
	- ‣ Detection of periodic patterns
	- ‣ Detection of anomalies
	- ‣ Identification of global trends
	- ‣ Evaluation of auto-correlation
	- ‣ Prediction of future values

• e.g. ARIMA (Autoregressive integrated moving average)

https://en.wikipedia.org/wiki/Autoregressive_integrated_moving_average

UNSTABLE/DEGENERATE TEMPORAL NETWORKS

Matthieu Latapy, Tiphaine Viard, and Clémence Magnien. "Stream graphs and link streams for the modeling of interactions over time". In: *Social Network Analysis and Mining* 8.1 (2018), p. 61.

UNSTABLE TEMPORAL NETWORK

- The network at a given *t* is not meaningful
- How to analyze such a network?

UNSTABLE TEMPORAL NETWORK

UNSTABLE TEMPORAL NETWORK \Box

- Common solution: transform into SEN using aggregation/ sliding windows with the consensus in the sensus user is a consensus in the literature on the meaning of those terms, unless they are used with an explicit reference to a paper de to a paper de ning those terms. Here is a list of those terms. Here is a list of the most
	- **Information loss**
- ‣ How to chose a proper aggregation window size? Static networks representing dynamic information can be ob-• **Dynamic Networks** and **Dynamic Graphs**
	- New theoretical tools developed to deal with such networks • **Longitudinal Networks** • **Evolving Graphs**
		- **Link Streams** & **Stream Graphs** (Latapy, Viard, and Magnien 2018)
		- **Temporal Networks**, **Contact Sequences** and **Interval Graphs** (Holme and Saramäki 2012)
		- **Time Varying Graphs** (Casteigts et al. 2012)

CENTRALITIES α NETWORK PROPERTIES IN STREAM GRAPHS

STREAM GRAPHS

Stream Graph (SG)- Definition

Stream Graphs have been proposed in*^a* as a generic formalism – it can represent any type of dynamic networks, continuous, discrete, with or without duration, with the objective or redefining typical notions of graphs on dynamic networks, including degenerate ones.

Let's define a Stream Graph

$$
S = (T, V, W, E)
$$

- *T* **Set of Possible times** (Discrete or Time intervals)
- *V* **Set of Nodes**
- *W* **Vertices presence time** $V \times T$
Edges presence time $V \times V \times T$
	- *Edges presence time* $V \times V \times T$

*^a*Latapy, Viard, and Magnien 2018.

STREAM GRAPHS

SG - Time-Entity designation

It is useful to work with Stream Graphs to introduce some new notions mixing entities (nodes, edges) and time:

STREAM GRAPHS time is continuous and edges are discrete, we take by convention

|T| = 1, i.e., we simply count nodes/edges presence time. N_u **Node presence**: The fraction of the total time during which u is present in the network $\frac{|T_u|}{|T|}$ *|T | L*_{uv} | Edge presence: The fraction of the total time during which (u,v) is present in the network $\frac{|T_{uv}|}{|T|}$ $|T|$

$STREAM GRAPHS$ *|T |*

SG - Rede!ning Graph notions

The general idea of redefining static network properties on Stream Graphs is that if the network stays unchanged along time, then properties computed on the stream graph should yield the same values as the same property computed on the aggregated graph.

STREAM GRAPHS

SG - *N* **&** *L*

The number/quantity of nodes in a stream graph is defined as the total presence time of nodes divided by the dataset duration. In general, it isn't an integer.

More formally:

$$
N = \sum_{v \in V} N_v = \frac{|W|}{|T|}
$$

For instance, $N=2$ if there are 4 nodes present half the time, or two nodes present all the time. Hulte unity, ^{or} **Two strea** $$ $\overline{}$

> In addition, $\delta(L)$ is eq 1 $|T|$ · $|V$ ⊗ $V|$ $\sqrt{ }$ $\int_t |E_t| dt =$ \int_{t} \int_t |*V* Finally, if we consid of the corresponding gra

STREAM GRAPHS

di'erent densities, respectively ¹ $N = 2$

For instance,
$$
L = 2
$$
 if there are 4 edges L is a *involved* in C and C is a *involved* in C .

\nFor instance, $L = 2$ if there are 4 edges L is a *involved* in C .

\nFor instance, $L = 2$ if there are 4 edges L is a *involved* in C are L .

\nFor instance, $L = 2$ if there are 4 edges L is a *involved* in C are *linked* and *involved* in C

 ${ac}$ }
 \hat{f} W_{α} define a clique of β

STREAM GRAPHS

 $L = 1$

STREAM & Eppeaning $\mathbf{v} \bullet \mathbf{v}$ $(resp. uniform).$ It is then $\mathbf{f}^{\text{ref}}_{\text{eff}}$ In Figure 3, for instance, *T ^C* $=\left[\mathbf{\hat{r}},\mathbf{\hat{s}}\right]$ \overline{r} \overline{c} \overline{a} W is ω ω clique ω compact (resp. if ω set) meaning that all pairs of

The density in static networks can be understood as the fraction of existing edges among all possible edges,

$$
d = \frac{L}{L_{\text{max}}}.
$$

 $L_{\rm b}$ the following we will ${\rm E}[{\bm{v}}]$ in $A \cdot {\rm E}[{\bm{x}}$ and ${\rm E}[{\bm{c}}]$ In the following, we will useignt, as 4: Leasternples of \bf{m} a understood as the probability of pact cliques involving three and $[7, 8] \times \{b, c, d\}$. Its oth is *d* = *N*2, because *N*² is the only way to de"ne *L*max in static networks, unlike in Stream Graphs. Unlike in Stream Graphs. Unlike in Stream Graphs. Unlike in Stream Graphs. U $[2, 5] \times \{a, c\}, [1, 8] \times \{b, c\},$

> a For instance, in Figure 4

STREAM GRAPHS SIRLAM GRAPHS Note that a common way to density in the density in static networks in $N-2$ $I-1$ $n = 2$ $N = 2$ $L = 1$

STREAM GRAPHS

SG - Clusters & Substreams

In static networks, a cluster is a set of nodes, and we have defined an (induced) subgraph of this cluster as a graph composed of the nodes of the cluster and the edges existing between those nodes. In Stream Graphs, a clusters *C* is as subset of *W*, and the corresponding (induced) substream $S(C) = (T, V, C, E(C))$, with $E(C) = \{(t, (u, v)) \in E, (t, u), (t, v) \in C\}.$

Example of subgraph (red,left) and induced substream (right).

STREAM GRAPHS

SG - Cliques

Having defined substreams and density, we can now naturally define a clique by analogy with static networks as a substream of density 1. A clique is said to be a **maximal clique** if it is not included in any other clique.

Red and Grey are the two maximal cliques of size three in this Stream Graph.

 $\{a, c\}, [1, 8] \times \{b, C\}$ \mathbb{Z} \mathbb{Z} $\mathbf{1}$. This decay be ques involvir ques involving three nodes of the link stream *L* of Figure 1 (right): $[2, 4] \times \{$ $8] \times \{b, c, d\}$. Its other maximal compact cliques are $[0, 4] \times \{a, b\}$, $[6, 9] \times$ \mathbf{F} Examples of maximal compact changeles. We display the two maximal $\{a, c\}$, $[1, 8] \times \{b, c\}$, $[7, 10] \times \{b, d\}$, $[6, 9]$ $\{c, d\}$ (involving two nodes each

a instrance, in Figure 5G - Neighborhood $N(u)$ to clique. However, it is a compact clique. $ximal$, as it is include degliberhood y α of node y is defined as the duster compact clique. The neighborhood $N(u)$ of node u is defined as the cluster com- intersects may not her posed of node-times such as an edge-time exists between it and There is a unique of u , i.e., D and $\{u, v, v, c\}$. ${\rm net}$ sect $\overline{{\rm sect}}$ nother a node-time of u , i.e., \mathbb{R}^1 requirement, the different department of $N(u) = \{v_t, (u, v)_t \in E\}$ \times {b, c, d}. The maximal contribution Examples of graphs with *N* = 2 nodes, *L* = 1 link, but with Figure 2: Two stream graphs with *n* = 2 nodes, *m* = 1 link, but with dierent $(0, 4] \times \{a, b\}$ is not a maximal clique because it is for instance included in the $\{a, b\} \cup [6, 9] \times \{c\}$ **SG** - Degree $k(u)$ and maximal either, as not maximal eit not maximal eit $G_{\text{in}}\frac{1}{\pi}\int_{0}^{1} \delta(G_{\text{in}}) \mathrm{d}t \mathrm{d}\pi$ $\oint \hat{d} \hat{d} \hat{d} \left(\hat{G}^{\dagger} \hat{d} \right) d\hat{d} \hat{d} \overline{d}$ l
W $\widetilde{J}_{1} = \frac{1}{2}$ *|T| |T| t* **length**. \bar{L} , $\frac{V}{V}$ for all \bar{L} the Neighborhood of node *u*, i.e. $\lim_{k\to\infty} \lim_{n\to\infty} S$ does not in general induce a clique in $G(S)$: for instance, $[0,1]\times\{0\}$ • **Fastest Paths** are paths of **minimal duration**. ${c, d}$ is a clique for the example $f(x)$ Figure 4 but ${a, b, c, d}$ is not a clique *k*(*u*) = *|N*(*u*)*|* • **Foremost Paths** are paths **arriving "rst**. $\texttt{diag}[\texttt{above},\texttt{Index}]$ for any $\texttt{diag}[\texttt{loop}]$ in $\texttt{diag}[\texttt{loop}]$, $e]\times X$ is a compact \mathbf{F} computed that \mathbf{F} **node** v in V , and the density $\int_{\frac{9}{6}^2}^{3}$ 12 $\frac{1}{|V_t|}$ and $\delta(t) = \frac{|E_t|}{|V_t \otimes V_t|}$ *.* time Example, the neighborhood of node 2 is highlighted in grey. $k(c) = \frac{5+2.5+5}{10} = 1.25.$ 0, respectively, then we define $\delta(uv)$,

*^u*2*V,u*=*^v |Tuv|*

13 **SG - Clustering coe!cient**

The clustering coefficient $C(u)$ of node u is defined as the density

of the ego-netwo<u>rk of u_{η} </u>, e. \vec{v} eventually obtains a $\vec{G}(\vec{w})$ \vec{v} = \vec{p} $d\vec{k}$ \vec{f} \vec{r} \vec{v} \vec{v} \vec{v} \vec{v} . $\mathcal{F}(\mathcal{H})$ *also is a path from u to v. If one iteratively removes the cycles of* P *in this way,*

The path P *is a shortest path from u to v if there is no path in* G of length lower than *k.* Then, *k is called the distance between u and v and it is denoted by* $\partial(u, v)$ *. If there no path between u* and *v* then their distance is infinite. The diameter of G is the largest *finite distance between two nodes in V .* Figure 14: Weakly components of \mathbb{R}^n μ e μ

PATHS AND DISTANCES IN STREAM GRAPHS

PATHS

SG - Paths

In a Stream Graph S=(T,V,W,E), a **path** P from node-time x_α to node-time y_{ω} is a sequence (t_0, x, v_0) , (t_1, v_0, v_1) , ..., (t_k, v_k, y) of elements of $T \times V \times V$ such that $t_0 \geq \alpha, t_k \leq \omega$, $((t_i, u_i, v_i)) \in E$. We say that P starts at t_0 , arrives at t_k , has length $k+1$ and **duration** $t_k - t_0$.

Examples of two paths from (node 0, t=0.5) to (node 3, t=1). The left one starts at 3, arrives at 5, has length 3 and duration 2. The right one starts at 1, arrives at 4.5, has length 3 and duration 3.5.

sentations introduced m and its corresponding graphy, and more athrits a charin restricted class sponding path is a cycle \inf_{nonloc} eraph. of shuffling methods that randomize specific temporal topological aspects of a network using the two level reresentations introduced in Section II A 2 above.

he \int_{Ω} $\mathbb{H}\mathbb{R}$ SG - Shortest FFastenshufflings furthermore • **Shortest Paths**, as in static networks, are paths of **minimal** is , connected $\mathbf{ig}(\mathbf{g}(\mathbf{h}))$ is continuated, connecting is which randomize the static grap no other connected: Paths transmittent at a trace alled the increasing • **Foremost Paths** are paths **arriving "rst**. Furthermore, one cange and the set afeabolic of in instruction for instantaneous stance: **Fastest shortest paths (paths of minimum this use to maniss with a ke**of minimal length) **Shortest fastest paths** (paths of minimal length among those of **IMAINE DEAL AND RAINER** 15 Connectedness and connected components *1. Link and timeline shuings* duration *t*max *t*min and the number of events (*C* or $\mathcal{L}(V,E)$ is connected if for a digital intermediate representation (in detail subset of $\mathcal{L}(V,E)$ $\log f$ *orm a partition* 3 *of V* sibility graph is the prophetic graph of $\log f$ $\log f$ $\log f$ $\log f$ E' *if* $u = v$ *in* G *. The connected components of G, are exactly but the components of* G *are exactly but* stream Fastest sfortest path Wesha of tension that sation amission that a head c \ln we denote by (*a*, *u*) - distributing the timelines (*x*), \ln engage (*t*), \sin the new limit v_k) of elements of a V to pological as beging a network using the transition of $E, [\alpha, t_0] \times \{u\} \subseteq W, [t_k$ resentations introduced in rection \mathcal{H}^A , \mathcal{H}^B_i , \mathcal{H}^C_i and \mathcal{H}^C_i is similar to a path from (*e*, u_{σ}) (of the are individual unionidate in the constraint of the constraints: we do not do the constraint of the co $\min_{\alpha} \max_{i=1}^N \sum_{j=1}^N \sigma_{ij} = 1$, $t_{i+1} \geq t_{i+1}$, t_{i+1} is the time to t_{i+1} and t_{i+1} and t_{i+1} and t_{i+1} and t_{i+1} are t_{i+1} and t_{i+1} and t_{i+1} are t_{i+1} and t_{i+1} and t_{i+1} a of a network but not the individual timelines, and *tim* stateg Popogeti Rift loat hand is the state of the content of the time left is the time of the time of the time of the time of the doubliset of *Literal Calc*, day, the call as learned we ${\mathcal{L}}$ vent sinks in G^{stat}). In practice they are implemented **by peha of religing the line several more restricted class** distributing the timelines (*i,j) distribution*, inn
distributing the timelines (*i,j) i,j) on t*he new, inn
of shuilling methods that dapped be socitio temporal. Without replacement. Processive shuings, on the other hand, constrain the $\cos\theta$ the individual timeling $\mathcal{B}_{(i,j)}^{\mathbf{H}}$ ($\mathcal{C}_{\mathcal{L}}^{\mathcal{L}}$ in $\mathcal{C}_{\mathcal{L}}^{\mathcal{L}}$) $\mathcal{C}_{\mathcal{L}}^{\mathcal{L}}$ are implemented by $\mathcal{C}_{\mathcal{L}}^{\mathcal{L}}$ models as they all conserve the nodes V , the temporal *E*). Event shuillings furthermore conserve the multiset $\lim_{R\to\infty} \frac{1}{R}$ durations, $p(\tau) = \left[\tau_q\right]^C$ Different shuffling methods and the construction of the methods and detailed the reserved of the second of the second of the second
These molecular constrains construction of the methods of the second of the second of the second of the second but the dink sultime at the ast dinglated the ratice ${\bf s}$ ibl $\bar{\bf r}$ h $\bm k$ b $\bm p$ (at $\bm q$) $\bm k$ ni $\bm p$, bhe contant k on $\bm s$ of $\bm v$ only condut durations and otherwise redistributes the meting letely. random, while the most standard the electron and the most while We furthermore define several more restricted class of shuil in the still exist that deal we have the world. to repeated to to and to in to in to in to in to to resentations introduced in Section II A 2 above. *1. Link and timeline shuings* Check experience

events.

(*, u*), which we denote by (*, u*)---(*, v*), if there is a sequence (*t*0*, u*0*, v*0), (*t*1*, u*1*, v*1),

random sequence shuing, P[*p^T* ()], conserves all the

graphs individually and independently using any shuf-

In a Stream Graph S=(T,V,W,E), a **path** *P* from node-time *x*↵ to *v^k* = *v*, *t*⁰ , *t^k* , for all *i*, *tⁱ ti*+1, *vⁱ* = *ui*+1, and (*ti, uivi*) *E*, [*, t*0] *{u} W*,

^N(*a*) = ([1*,* 3] [7*,* 8]) *{b}* [4*.*5*,* ⁷*.*5] *{c}* is in blue, leading to *^d*(*a*) = ³

RANDOM MODELS FOR DYNAMIC NETWORKS

Laetitia Gauvin et al. "Randomized reference models for temporal networks". In: *SIAM Review* 64.4 (Nov. 2022)

RANDOM MODELS

- In many cases, in network analysis, useful to compare a network to a randomized version of it
	- ‣ Clustering coefficient, assortativity, modularity, …
- In a static graph, 2 main choices:
	- ‣ Keep only the number of edges (ER model)
	- ‣ Keep the number of edges and the degree of nodes (Configuration model)
- In dynamic networks, it is more complex...

Snapshot Shu#ing SG - Component Community

Snapshot Shuffling keeps the order of snapshots, randomize \parallel edges inside snapshots. Any random model for static network can | be used, such as ER random graphs or a degree preserving randomization.
2018) for details and the simplest one is the simplest of the simplest one is the simplest of the simplest one is the simplest one

Sequence Shu#ing

Sequence Shuffling keeps each snapshot identical, switch randomly their order.

$RANDOMMODELS$

Link Shuffling Link Shu!ing keeps activation time per node pairs, randomize

Link Shuffling keeps activation time per node pairs, randomize the aggregated graph. For instance, a simple way to achieve this is to pick two node pairs at random (connected or not) of the aggregated graph, and to exchange activation time of these node

Timeline Shu!ing Timeline Shu!ing

edges activation time. For instance, a simple way to achieve this is to redistribute randomly activation period among all edges, e.g.: **Timeline Shuffling** keeps the aggregated graph, randomize

RANDOM MODELS

More constrained Shuffling

Variants of these shufflings with additional constraints have been proposed, for instance the Local timeline shuffling, randomizing **they are e.g. not guaranteed to sample the same of the same** events time edge by edge, or the **Weight constrained timeline shuffling**, randomizing events while conserving the number of observations for each edge. See (Gauvin et al. 2018) for details. not all compositions generate a microcanonical RRM \sim $r = \frac{1}{2}$ $\frac{1}{\sigma}$ is the shuing shuing shuing and time- $\sum_{i=1}^{n}$ shot shuings always result in a MRRM \sim

FIG. II.7: **Illustration of intersections between**

 $t \sqrt{D}$ random between the existing links. Laetitia Gauvin et al. "Randomized reference models for temporal networks". In: *SIAM Review* 64.4 (Nov. 2022)

characterize them in Section V C).

compositions have been used in the literature to produce

MRRMs that randomize both topological and temporal

aspects of a network at the same time (we describe and

DYNAMIC COMMUNITY DETECTION

Rossetti, G., & Cazabet, R. (2018). Community discovery in dynamic networks: a survey. *ACM Computing Surveys (CSUR)*, *51*(2), 1-37.

Cazabet, R., Boudebza, S., & Rossetti, G. (2020). Evaluating community detection algorithms for progressively evolving graphs. *Journal Of Complex Networks*

COMMUNITY DETECTION structural nature of these exchanges. We use a typical dataset obtained from a public mailing-list archive to il-

Static networks

Dynamic Networks L yi raillic in convolts

Clusters: Sets of nodes Clusters: Sets of time-nodes, i.e., pairs (node,time) ing email exchanges. Indeed, we expect the exchanges of tural and temporal point of views. This is illustrated in

A.2 Dataset (2016). Analysis of the temporal and structural features of threads in a Gaumont, N., Viard, T., Fournier-S'Niehotta, R., Wang, Q., & Latapy, M. mailing-list. In *Complex Networks VII*

COMMUNITY DETECTION

Static networks Dynamic Networks

Clusters: Sets of nodes

Clusters: Sets of time-nodes, i.e., pairs (node,time)

APPROACHES TO DCD

DYNAMIC COMMUNITIES ?

More than 50 methods published, broad categories

Rossetti, G., & Cazabet, R. (2018). Community discovery in dynamic networks: a survey. *ACM Computing Surveys (CSUR)*, *51*(2), 1-37. 54

CATEGORIES

- Instant optimal:
	- ‣ Allows reusing static algorithms
	- ‣ No partition smoothing
	- ‣ Labels can be smoothed
	- ‣ Simple to parallelize

CATEGORIES

• Temporal trade-off

- ‣ Cannot be parallelized (iterative)
- \rightarrow \Rightarrow Best suited for real-time analysis / tasks

• Cross-Time

- ‣ Requires to know the whole evolution in advance
- \rightarrow \Rightarrow Not suited for real-time analysis, potentially the best smoothed (a posteriori interpretation)

WHAT MAKES DCD INTERESTING

NARRATIVES ?

SMOOTHNESS / STABILITY

- No Smoothness: Partition at **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

COMMUNITY EVENTS

PROGRESSIVE EVOLUTION

2 communities 1 community ?? Intermediate state

How to *track* communities, giving a *coherent* dynamic structure ?

IDENTITY PRESERVATION

Ship of Theseus [Plutarch., 75]

2 problems: 2 problems: posed of the same nodes as the other community at its start. Which cluster (B or C) has the same 1) Find node clusters at each t *•* Instant Optimal approaches are the best choice when the final goal is to provide 2)Assign labels between same communities at \neq t

 c_1 computer that are as c_1 that are possible as c_1 the evolution of the evol u 197. UI.
rk:Thoor *•* Cross-Time approaches are the best choice when the final goal is to provide com-Cazabet, R., & Rossetti, G. (2019). Challenges in community discovery on temporal networks. In *Temporal Network Theory* (pp. 181-197). Springer, Cham.

EMPIRICAL EVALUATION

Cazabet, R., Boudebza, S., & Rossetti, G. (2020). Evaluating community detection algorithms for progressively evolving graphs. *Journal Of Complex Networks*

SETTING

- Choose methods based on the same definition of a static community: Modularity (most widespread), but different approaches to dynamics
- Generate dynamic networks with planted dynamic community structure

SETTING

(b) The static graph at time $t=0$, version *sharp* $(\alpha = 0.9, \beta = 0.05, \beta_r = 0.01)$ $\begin{array}{c} \hline \text{E} & \text{E} \end{array}$ (b) The static graph at time t=0, ver munities. Grey areas represent and \mathbf{S} represent and \mathbf{S} represent and \mathbf{S} represents are identified and \mathbf{S} represents and \mathbf{S} represents are identified and \mathbf{S} represents and \mathbf{S} repre

(c) The static graph at time $t=0$, version *blurred* $(\alpha = 0.8, \beta = 0.25, \beta_r = 0.01)$

METHODS

- Instant Optimal
	- ‣ No smoothing
		- Louvain at each step, match with Jaccard
- Temporal trade-off
	- ‣ Implicit Global
		- Louvain at each step in initialized by the previous partition (same local maximum), +Jaccard
	- ‣ DYNAMO
		- Update partition only based on edge changes to keep modularity high
	- ‣ Smoothed-graph
		- Each snapshot is modified to artificially raise the probability to obtain similar partition as previous step, then Louvain+Jaccard
- Cross-Time
	- ‣ Transversal Network
		- Create a single graph by adding edges between same nodes in successive snapshots (Mucha et al.), then (modified) Modularity optimization
	- ‣ Label-Smoothing
		- Create a "Community Survival graphs": nodes are static communities(Louvain), edges weighted by Jaccard Similarity. Apply Louvain on it. 66

 $t=1$

(b) The static graph at time $t=0$, version *sharp* $(\alpha = 0.9, \beta = 0.05, \beta_r = 0.01)$

67

(a) No-Smoothing (b) Label-Smoothing

(c) The static graph at time $t=0$, version *blurred*

 $(\alpha = 0.8, \beta = 0.25, \beta_r = 0.01)$

Ground truth

 $\mathbf 0$

 $_{68}^{\circ}$ (e) Implicit-Global

 \mathbb{H}^+ 150 50 100 200 250

(c) DYNAMO (d) Transversal-Network

(f) Smoothed-Graph

MEASURING DC QUALITY?

- Evaluation at each step (No smoothness)
	- ‣ Average Mutual Information (similarity at each step)
	- ‣ Average Modularity
- Evaluation of Smoothness
	- ‣ **SM-P**artitions: Average Mutual Information between successive partitions
		- Label independent, insensitive to glitches, Identity loss
	- ‣ **SM-N**odes: Inverse of number of affiliation change
		- Sensitive to glitches
	- ‣ **SM-L**abels: Inverse of Shannon entropy of nodes labels
		- **-** Sensitive to Identity Loss
- **•** Longitudinal Score
	- ‣ Modified mutual information of time-node (u,t)

MEASURING DC QUALITY?

TO SUM UP ON DYNAMIC GRAPHS

TO SUM UP

- Currently, most practitioners still use the snapshot approaches
	- ‣ No widespread framework
	- No widespread coding libraries (pathpy, tnetwork, tacoma=>limited usage)
	- ‣ Datasets still relatively limited
- But considered an important topic to work on
	- ‣ Dynamic is everywhere
	- Dynamic changes many things in many cases