



Stream graphs and link streams for the modeling of interactions over time

Matthieu Latapy¹ · Tiphaine Viard¹ · Clémence Magnien¹

Received: 13 March 2018 / Revised: 10 September 2018 / Accepted: 11 September 2018 / Published online: 3 October 2018
© Springer-Verlag GmbH Austria, part of Springer Nature 2018

Abstract

Graph theory provides a language for studying the structure of relations, and it is often used to study interactions over time too. However, it poorly captures the intrinsically temporal and structural nature of interactions, which calls for a dedicated formalism. In this paper, we generalize graph concepts to cope with both aspects in a consistent way. We start with elementary concepts like density, clusters, or paths, and derive from them more advanced concepts like cliques, degrees, clustering coefficients, or connected components. We obtain a language to directly deal with interactions over time, similar to the language provided by graphs to deal with relations. This formalism is self-consistent: usual relations between different concepts are preserved. It is also consistent with graph theory: graph concepts are special cases of the ones we introduce. This makes it easy to generalize higher level objects such as quotient graphs, line graphs, k -cores, and centralities. This paper also considers discrete versus continuous time assumptions, instantaneous links, and extensions to more complex cases.

Keywords Stream graphs · Link streams · Temporal networks · Time-varying graphs · Dynamic graphs · Dynamic networks · Longitudinal networks · Interactions · Time · Graphs · Networks

1 Introduction

Friendship, dependencies, similarities, or connections are typical examples of relations modeled by graphs or networks, i.e., sets of nodes and links: nodes represent individuals and two individuals are linked together if they are friends; nodes represent companies and they are linked together if they signed contracts with each other; nodes represent documents like web pages or articles, and they are linked together if they are similar; nodes represent computer devices and they are linked together if there is a wire between them; etc.

For decades, graph theory, social network analysis, and network science have developed a wide set of tools for the study of such graphs. In particular, they developed a *language* for describing networks, with elementary yet powerful concepts such as node degree (their number of links), paths (sequences of links going from one node to another one), density (the fraction of pairs of nodes actually linked

together), or cliques (sets of nodes all pairwise linked together). This language forms the basis of network studies, and there is a global consensus on a wide set of concepts that are used in the field; with few variations, all courses and reference books on graphs and networks start with them, see, for instance, Berge (1962), Bondy (1976), Wasserman and Faust (1994), West (2000), David and Jon (2010), Newman (2010), Diestel (2012), Barabási and Pósfai (2016), Scott (2017). Then, more advanced and specific concepts are defined on this common ground.

Contacts, shopping, travels, or traffic are typical examples of interactions that take place over time, i.e., streams of nodes and links active during specific periods of time: nodes are individuals linked together whenever they call each other; nodes are clients and products linked together when a client buys a product; nodes are places linked together when someone moves from one place to another; nodes are internet devices linked together when they exchange data; etc.

Such sequences of interactions play a key role in many areas, and they have been studied for a long time, see related work in Sect. 21. Although many variations exist, the most common approach is to model them by sequences of graphs (each graph then aggregates the interactions that occurred during a period of time), by labeled graphs (each link being

✉ Matthieu Latapy
matthieu.latapy@lip6.fr

¹ Sorbonne Universités, UPMC Univ Paris 06, CNRS UMR 7606, LIP6, 75005 Paris, France

labeled with its presence times), or other augmented graphs. This makes it possible to use graph theory to study these sequences of graphs, labeled graphs, and other variants. Other works deal directly with higher level methods for studying graphs, like stochastic block models, for instance, and extend them to cope with the dynamics. Finally, a few works define specific properties combining temporal and structural information, such as centrality measures, for instance.

In this paper, we propose a different approach: we develop a formalism to directly cope with interactions over time, in a way similar to what graph theory does for relations. This means that we do not transform interactions into graphs, but rather transform graph theory into a theory of interactions over time. We model them as *link streams* and *stream graphs* (depending on whether the dynamics is on links only, or on both nodes and links), so named to emphasize their streaming nature and the fact that they are *not* graphs or networks. Then, we start with the most elementary graph concepts and we define their equivalent for stream graphs and link streams. Finally, we elaborate on these basic concepts to extend more complex graph concepts. With the aim to make our formalism as intuitive as possible, we put much effort in proposing simple definitions, explaining them with different points of view (especially combinatorial and probabilistic ones), and to provide illustrations and detailed examples of all key concepts we introduce.

In addition to these subjective features, we also put much emphasis on two more objective features to ensure the relevance of our definitions. First, we want our formalism to be a generalization of graph theory in a very precise sense: when the stream has no dynamics, it is equivalent to a classical graph and its properties should be the same as those of this graph (see the end of Sect. 3). Second, we want the relations that exist between various graph properties (between density and degree, for instance) to still hold for stream properties. Similarly, if a graph concept is derived from another one (like clustering coefficient from density, for instance), we want the corresponding stream concept to be derived from the corresponding other stream concept. These features ensure both the self-consistency of our formalism and its consistency with graph theory.

After Sect. 2 that introduces a few notations needed in the whole paper, we present our framework from Sect. 3 to Sect. 17. Each of these sections is devoted to a key concept of graph theory that we redefine in the stream context. Therefore, they all have the same structure: first, we recall the relevant graph concepts and their key properties in italics; then, we introduce equivalent concepts for stream graphs with detailed examples and discuss their properties; we introduce additional related concepts specific to stream graphs; we discuss the case of link streams, i.e., when there is no dynamics on nodes; and we show that the newly introduced stream concepts are equivalent to the graph ones, whenever this makes sense. After these core sections, we show how our framework may

be used under either discrete and continuous modeling of time in Sect. 18; we show how it generalizes Δ -analysis and may be used with instantaneous links in Sect. 19; we show how it may be extended to bipartite streams and other particular cases in Sect. 20; and we present related work in Sect. 21. We discuss our contributions and future work in Sect. 22.

2 Preliminaries on set products and sizes

In this paper, we rely on a few notations that we introduce below.

Given two finite sets X and Y , one may consider the ordered pairs (x, y) with $x \in X$ and $y \in Y$. Then, $(x, y) \neq (y, x)$ and (x, x) exists if $x \in X$ and $x \in Y$. One may also consider unordered pairs xy with $x \in X$ and $y \in Y$, with $x \neq y$. Then, $xy = yx$ and xx do not exist. The *set of ordered pairs*, called cartesian product of X and Y , is denoted by $X \times Y$. One often uses this notation for the set of unordered pairs too. In this paper, however, we use both notions intensively and need to make a clear distinction between them. We, therefore, denote the *set of unordered pairs of distinct elements* by $X \otimes Y$.

Throughout this paper, we deal with *set sizes*, denoted by $|X|$ for a given set X , but the meaning of this notation depends on the type of X . If X is an interval $[\alpha, \omega]$ of \mathbb{R} , then $|X| = \omega - \alpha$. If it is an interval $[\alpha, \omega]$ of \mathbb{N} , then $|X| = \omega - \alpha + 1$. If X is the union of disjoint intervals of \mathbb{R} , then $|X|$ is the sum of these intervals' sizes. The same holds if it is the union of disjoint intervals of \mathbb{N} . If X is the product of sets of these types, then its size is the product of their sizes. Notice that, if X contains just one element then depending on the context it may be seen as a (degenerate) interval of \mathbb{R} or \mathbb{N} , thus having size 0 or 1, respectively. For instance, the union of the intervals $[1, 2]$ and $[3, 3]$ of \mathbb{R} has size 1, while the union of the same intervals of \mathbb{N} has size 3.

Notice that $|X \times Y| = |X| \cdot |Y|$, and therefore, $|X \times X| = n^2$ if $|X| = n$. This is different from $|X \otimes Y| = |(X \setminus Y) \times Y| + |(Y \setminus X) \times X| - |(X \setminus Y) \times (Y \setminus X)| + \frac{|X \cap Y|^2 - |X \cap Y|}{2}$, leading to $|X \otimes X| = \frac{n(n-1)}{2}$ if $|X| = n$, and $|X \otimes Y| = |X| \cdot |Y|$ if X and Y are disjoint.

3 Stream graphs and link streams

A (simple undirected¹) graph $G = (V, E)$ is defined by a finite set of nodes V and a set of links $E \subseteq V \otimes V$. Then, $uv \in E$ means that u and v are linked together in G .

¹ Unless explicitly specified, we always consider simple and undirected graphs and stream graphs; we discuss more general cases in Sect. 20.

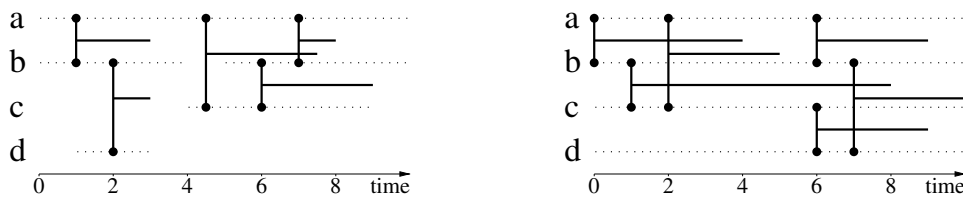


Fig. 1 Simple examples of stream graphs and link streams. Left: a stream graph $S = (T, V, W, E)$ with $T = [0, 10] \subseteq \mathbb{R}$, $V = \{a, b, c, d\}$, $W = [0, 10] \times \{a\} \cup ([0, 4] \cup [5, 10]) \times \{b\} \cup [4, 9] \times \{c\} \cup [1, 3] \times \{d\}$, and $E = ([1, 3] \cup [7, 8]) \times \{ab\} \cup [4.5, 7.5] \times \{ac\} \cup [6, 9] \times \{bc\} \cup [2, 3] \times \{bd\}$. In other words, $T_a = [0, 10]$, $T_b = [0, 4] \cup [5, 10]$, $T_c = [4, 9]$, $T_d = [1, 3]$, $T_{ab} = [1, 3] \cup [7, 8]$, $T_{ac} = [4.5, 7.5]$, $T_{bc} = [6, 9]$,

$T_{bd} = [2, 3]$, and $T_{ad} = T_{cd} = \emptyset$. Right: a link stream $L = (T, V, E)$ with $T = [0, 10] \subseteq \mathbb{R}$, $V = \{a, b, c, d\}$, and $E = ([0, 4] \cup [6, 9]) \times \{ab\} \cup [2, 5] \times \{ac\} \cup [1, 8] \times \{bc\} \cup [7, 10] \times \{bd\} \cup [6, 9] \times \{cd\}$. In other words, $T_a = T_b = T_c = T_d = T$ and $T_{ab} = [0, 4] \cup [6, 9]$, $T_{ac} = [2, 5]$, $T_{bc} = [1, 8]$, $T_{bd} = [7, 10]$ and $T_{cd} = [6, 9]$

Graphs model relations between nodes. For instance, nodes may represent individuals and links may represent friendship relations. Nodes may represent computers and links may represent physical connections between them. Examples are countless, making graphs the key formalism for studying network structures.

We define a (simple undirected ²) *stream graph* $S = (T, V, W, E)$ by a finite set of nodes V , a measurable set of time instants T , a set of temporal nodes $W \subseteq T \times V$, and a set of links $E \subseteq T \times V \otimes V$, such that $(t, uv) \in E$ implies $(t, u) \in W$ and $(t, v) \in W$. Then, $(t, v) \in W$ means that node v is present at time t in S , and $(t, uv) \in E$ means that nodes u and v are linked together at time t in S .

The set of time instants T may be continuous or discrete, which has little influence on the following, as we explain in Sect. 18. Until then, all the examples we give assume that T is an interval of \mathbb{R}^+ .

We define $v_t = 1$ if $(t, v) \in W$ and $v_t = 0$ otherwise, as well as $uv_t = 1$ if $(t, uv) \in E$ and $uv_t = 0$ otherwise. When $v_t = 1$ we say that node v is involved in S at time t or that v is present at time t , and when $uv_t = 1$ we say that nodes u and v are linked together at time t , or that link uv is present at time t . We denote by T_v the set of time instants at which v is present, by T_{uv} the set of time instants at which uv is present, by V_t the set of nodes present at time t , and by E_t the set of links present at time t : $T_v = \{t, v_t = 1\}$, $T_{uv} = \{t, uv_t = 1\}$, $V_t = \{v, v_t = 1\}$, and $E_t = \{uv, uv_t = 1\}$. Notice that $T_{uv} \subseteq T_u \cap T_v$.

If all nodes are present all the time, i.e., $T_v = T$ for all v or, equivalently, $V_t = V$ for all t , then we say that S is a *link stream* and we denote it by $L = (T, V, E)$ (with $W = T \times V$ implicitly). Indeed, there is no dynamics on nodes in this case, and S is fully defined by this triplet. Link streams play an important role in many situations, and therefore, we pay special attention to this case in all this paper.

We illustrate these definitions in Fig. 1 with *drawings* designed as follows. We display node names on a vertical axis on the left of the figure and time on a horizontal axis at the bottom of the figure. Each node presence times are

represented by a horizontal dotted line in front of its name, whenever the node is present. Each link presence times are represented by a horizontal solid line parallel to the two dotted lines of involved nodes and a vertical solid line joining these two dotted lines (marked with bullets) when the two nodes start interacting. In Fig. 1, for instance, in S (leftmost example), the node a arrives at time 0 and stays until time 10, and therefore, $[0, 10] \times \{a\} \subseteq W$, i.e., $T_a = [0, 10]$. This is represented by a dotted line from time 0 to 10 in front of a in the drawing. Likewise, b arrives at time 0, then leaves at time 4, joins again at time 5, and stays until time 10, and therefore, $([0, 4] \cup [5, 10]) \times \{b\} \subseteq W$, i.e., $T_b = [0, 4] \cup [5, 10]$. This is represented by a dotted line from time 0 to 4 and another one from time 5 to 10 in front of b . These two nodes interact from time 1 to time 3 and from time 7 to time 8, and therefore, $([1, 3] \cup [7, 8]) \times \{ab\} \subseteq E$, i.e., $T_{ab} = [1, 3] \cup [7, 8]$. This is represented by a solid line at time 1 between the dotted lines of a and b , with a horizontal line starting from its middle until time 3, and another such solid line at time 7 with a horizontal line until time 8.

Given a stream graph $S = (T, V, W, E)$, we define $G_t = (V_t, E_t)$, the *graph induced by S at time t* . In Fig. 1, for instance, we obtain for S at time 2 the graph $G_2 = (\{a, b, d\}, \{ab, bd\})$.

We also define $G(S) = (\{v, T_v \neq \emptyset\}, \{uv, T_{uv} \neq \emptyset\}) = (\bigcup_{t \in T} V_t, \bigcup_{t \in T} E_t)$ the *graph induced by S* : its nodes are those present in S and they are linked together in $G(S)$ if there exists a time instant in T such that they are linked together in S . In other words, it is the graph, where there is a link between two nodes if they interacted at least once. In Fig. 1, for instance, $G(S) = (\{a, b, c, d\}, \{ab, ac, bc, bd\})$ and $G(L) = (\{a, b, c, d\}, \{ab, ac, bc, bd, cd\})$. One may, in addition, associate with each node v or link uv a weight capturing a quantity of interest, like, for instance, their presence duration $|T_v|$ and $|T_{uv}|$.

Stream graphs model interactions between nodes over time, as well as the dynamics of nodes themselves. For instance, nodes may represent individuals present in a given building and links may represent contacts between them.

Nodes may represent online computers and links may represent data exchanges between them. Examples are countless, and we aim at making stream graphs the key formalism for studying jointly the dynamics and structure of interactions.

Since in a stream graph $S = (T, V, W, E)$ nodes are not present all the time in general, W may differ significantly from $T \times V$. To capture this, we define the *coverage* of S as follows:

$$\text{cov}(S) = \frac{|W|}{|T \times V|}.$$

For instance, in Fig. 1, the stream graph S has coverage $\text{cov}(S) = \frac{26}{40} = 0.65$.

Notice that $\text{cov}(S) = 1$ if and only if all nodes are present all the time, and therefore, it is equivalent to saying that S is a link stream.

If in addition for all u and v in V , $T_{uv} \in \{\emptyset, T\}$, i.e., all existing links are present all the time, then there is no significant distinction between S and $G(S)$, and we say that S is a *graph-equivalent stream*. This gives a formal ground to our wanted feature that stream graphs generalize graphs: we extend graph concepts to stream graphs in a way, such that if a stream graph S has a given stream graph property and happens to be a graph-equivalent stream, then its induced graph $G(S)$ has the corresponding graph property. In the following, we systematically check that this feature holds.

4 Size, duration, uniformity, and compactness

The number of nodes of a graph $G = (V, E)$ is denoted by $n = |V|$ and its number of links by $m = |E|$.

Given a stream graph $S = (T, V, W, E)$, we now define its number of nodes and links, as well as its duration. First notice that, unlike in graphs, some nodes may be present for much longer than others. To capture this, we define the *contribution of node v* as $n_v = \frac{|T_v|}{|T|}$, which may be seen as the notion of coverage restricted to a node v . We then define the *number of nodes* in S as follows:

$$n = \sum_{v \in V} n_v = \frac{|W|}{|T|}.$$

Then, each node contributes to the total number of nodes proportionally to its involvement in S : v in V accounts for 1 node only if it is present in S all the time.

We define similarly the contribution of a pair of nodes uv as $m_{uv} = \frac{|T_{uv}|}{|T|}$ and the *number of links* in S :

$$m = \sum_{uv \in V \otimes V} m_{uv} = \frac{|E|}{|T|}.$$

Like nodes, each link then contributes to m proportionally to its presence in S : uv in $V \otimes V$ accounts for 1 link only if it is present in S all the time.

Finally, we define the node and link contributions of a time instant t as $k_t = \frac{|V_t|}{|V|}$ and $l_t = \frac{|E_t|}{|V \otimes V|}$, leading to the following definition of the node duration k in S and the link duration l in S :

$$k = \int_{t \in T} k_t dt = \frac{|W|}{|V|} \quad \text{and} \quad l = \int_{t \in T} l_t dt = \frac{|E|}{|V \otimes V|}.$$

Like the number of nodes n and the number of links m , the node duration k may be seen as a duration of S , where each time contributes proportionally to the number of nodes present at this time, and the link duration l as a duration of S , where each time contributes proportionally to the number of links present at this time.

Notice that n is the expected value of $|V_t|$ when one takes a random time t in T . Likewise, m , k , and l are the expected value of $|E_t|$, $|T_v|$, and $|T_{uv}|$ when one takes a random time t in T , a random node v in V or a random pair of nodes in $V \otimes V$, respectively.

The following relation also hold: $\text{cov}(S) = \frac{|W|}{|T \times V|} = \frac{n}{|V|} = \frac{k}{|T|}$, $n \cdot |T| = k \cdot |V| = |W|$, and $m \cdot |T| = l \cdot |V \otimes V| = |E|$.

For the examples in Fig. 1, we obtain for S the following values: $n = \frac{|T_a|}{10} + \frac{|T_b|}{10} + \frac{|T_c|}{10} + \frac{|T_d|}{10} = 1 + 0.9 + 0.5 + 0.2 = 2.6$ nodes, $m = \frac{|T_{ab}|}{10} + \frac{|T_{ac}|}{10} + \frac{|T_{bc}|}{10} + \frac{|T_{bd}|}{10} = 0.3 + 0.3 + 0.3 + 0.1 = 1$ link, $k = \frac{26}{4} = 6.5$ time units, and $l = \frac{10}{6} = 1.66\dots$ time units. For L , we obtain $n = 4$ nodes, $m = 0.7 + 0.3 + 0.7 + 0.3 + 0.3 = 2.3$ links, $k = 10$ time units and $l = \frac{23}{6} = 3.833\dots$ time units.

In a link stream $L = (T, V, E)$, by definition $T_v = T$ for all v in V , and therefore, $n_v = 1$ and $n = |V|$. Likewise, for all t , $V_t = V$, and therefore, $k_t = 1$ and $k = |T|$. In a graph-equivalent stream, in addition $T_{uv} \in \{\emptyset, T\}$ for all uv in $V \otimes V$ and E_t is the same for all t . Then, the number of nodes and links in the stream are equal to the number of nodes and links in the corresponding graph.

Notice now that, in a given stream graph, for two nodes u and v , such that $|T_u| = |T_v|$ both $T_u = T_v$ or $T_u \cap T_v = \emptyset$ are possible, as well as all intermediary situations. This has a crucial influence on the possible existence of links between u and v , and so on the structure of S . To capture this, we define the *uniformity* of S as follows:

$$\Psi(S) = \frac{\sum_{uv \in V \otimes V} |T_u \cap T_v|}{\sum_{uv \in V \otimes V} |T_u \cup T_v|}.$$

If S has uniformity 1, then we say that it is uniform: for all u and v in V , $T_u = T_v$, i.e., all nodes are present at the same times.

We also define for any pair of nodes u and v in V the uniformity $\Psi(u, v) = \frac{|T_u \cap T_v|}{|T_u \cup T_v|}$. It measures the overlap between the presence times of u and v , thus their ability to be linked together.

Given a stream graph $S = (T, V, W, E)$, we define $S' = (T', V', W, E)$ such that $T' = [\min\{t, \exists(t, v) \in W\}, \max\{t, \exists(t, v) \in W\}]$ and $V' = \{v, \exists(t, v) \in W\}$. We then define the compactness of S as follows:

$$c(S) = \frac{|W|}{|T' \times V'|} = \text{cov}(S')$$

If S has a compactness of 1, then we say that it is compact: for all v in V , $T_v = [b, e] \subseteq T$, i.e., the presence times of all nodes is the same interval of T .

For the examples in Fig. 1, S has uniformity

$$\begin{aligned} \Psi(S) &= \frac{|T_a \cap T_b| + |T_a \cap T_c| + |T_a \cap T_d| + |T_b \cap T_c| + |T_b \cap T_d| + |T_c \cap T_d|}{|T_a \cup T_b| + |T_a \cup T_c| + |T_a \cup T_d| + |T_b \cup T_c| + |T_b \cup T_d| + |T_c \cup T_d|} \\ &= \frac{(4 + 5) + 5 + 2 + 4 + 2 + 0}{10 + 10 + 10 + 10 + (4 + 4) + (2 + 5)} = \frac{22}{55} = 0.4 \end{aligned}$$

and compactness $c(S) = \text{cov}(S) = \frac{26}{40}$, since on this particular case, $T' = T$ and $V' = V$, and therefore, $S' = S$.

If S is a link stream, then its uniformity and compactness are necessarily equal to 1, like L in Fig. 1.

5 Density

The density of graph $G = (V, E)$ is the probability when one takes a random element uv in $V \otimes V$ that there is a link between u and v in E : $\delta(G) = \frac{2m}{n(n-1)}$. If $n \in \{0, 1\}$ then $\delta(G)$ is defined to be 0.

We define the density of stream graph $S = (T, V, W, E)$ as the probability when one takes a random element (t, uv) of $T \times V \otimes V$ such that (t, u) and (t, v) are in W , that (t, uv) is in E :

$$\delta(S) = \frac{\sum_{uv \in V \otimes V} |T_{uv}|}{\sum_{uv \in V \otimes V} |T_u \cap T_v|} = \frac{\int_{t \in T} |E_t| dt}{\int_{t \in T} |V_t \otimes V_t| dt}$$

If $\sum_{uv \in V \otimes V} |T_u \cap T_v| = \int_{t \in T} |V_t \otimes V_t| dt = 0$ then we define $\delta(S)$ to be 0.

In other words, the density is the probability when one takes a random time and two random nodes such that a link may exist between them at this time that the link indeed exists. It is the fraction of possible links that do exist.

Notice that $\sum_{uv \in V \otimes V} |T_{uv}| = \int_{t \in T} |E_t| dt = |E|$. In addition, $\sum_{uv \in V \otimes V} |T_u \cap T_v| = \int_{t \in T} |V_t \otimes V_t| dt$ is related to the

uniformity $\Psi(S)$ of S , but it cannot be directly derived from $|T|$, $|V|$, $|W|$, and $|E|$.

For S defined in Fig. 1 (left), $\sum_{uv \in V \otimes V} |T_{uv}| = |T_{ab}| + |T_{ac}| + |T_{bc}| + |T_{bd}| = 3 + 3 + 3 + 1 = 10$, $\sum_{uv \in V \otimes V} |T_u \cap T_v| = |T_a \cap T_b| + |T_a \cap T_c| + |T_a \cap T_d| + |T_b \cap T_c| + |T_b \cap T_d| + |T_c \cap T_d| = 9 + 5 + 2 + 4 + 2 + 0 = 22$, and we obtain $\delta(S) = \frac{10}{22} \sim 0.45$. For L defined in this figure (right), $\sum_{uv \in V \otimes V} |T_{uv}| = 7 + 3 + 7 + 3 + 3 = 23$, $\sum_{uv \in V \otimes V} |T_u \cap T_v| = |V \otimes V| \cdot |T| = 60$ and we obtain $\delta(L) = \frac{23}{60} \sim 0.38$.

Notice that there is in general no relation between the density δ , the number of nodes n and the number of links m in a stream graph, see Fig. 2.

However, the classical graph relation $\delta = \frac{2m}{n(n-1)}$ holds for a link stream $L = (T, V, E)$. Indeed, we then have $T_u = T_v = |T|$ for all u and v , and $n = |V|$, which leads to

$$\delta(L) = \frac{\sum_{uv \in V \otimes V} |T_{uv}|}{\sum_{uv \in V \otimes V} |T|} = \frac{2 \cdot \sum_{uv \in V \otimes V} |T_{uv}|}{n \cdot (n-1) \cdot |T|} = \frac{2 \cdot m}{n \cdot (n-1)}$$

In addition, $\delta(L)$ is equal to the average density of G_t :

$$\begin{aligned} \frac{1}{|T|} \int_t \delta(G_t) dt &= \frac{1}{|T|} \int_t \frac{|E_t|}{|V_t \otimes V_t|} dt = \frac{1}{|T| \cdot |V \otimes V|} \int_t |E_t| dt \\ &= \frac{\int_t |E_t| dt}{\int_t |V_t \otimes V_t| dt} = \delta(L) \end{aligned}$$

, since, in L , $V_t = V$ for all t .

Finally, if we consider a graph-equivalent stream, then its density is equal to the density of the corresponding graph.

In addition to the global concept of density introduced above, we define the density of a pair of nodes uv in $V \otimes V$, the density of a node v in V , and the density at a time instant t in T , respectively, as follows:



Fig. 2 Two stream graphs with $n = 2$ nodes, $m = 1$ link, but with different densities: Left: $\delta = 0.75$. Right: $\delta = 1$

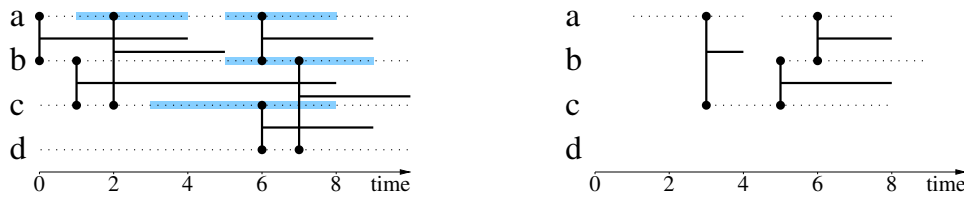


Fig. 3 Example of cluster with its induced sub-stream. Left: the cluster, displayed in blue, is $C = ([1, 4] \cup [5, 8]) \times \{a\} \cup [5, 9] \times \{b\} \cup [3, 8] \times \{c\}$. Right: the sub-stream induced by C is $S(C) = ([0, 10],$

$\{a, b, c, d\}, C, E(C))$ with $E(C) = [6, 8] \times \{ab\} \cup [3, 4] \times \{ac\} \cup [5, 8] \times \{bc\}$

$$\delta(uv) = \frac{|T_{uv}|}{|T_u \cap T_v|}, \quad \delta(v) = \frac{\sum_{u \in V, u \neq v} |T_{uv}|}{\sum_{u \in V, u \neq v} |T_u \cap T_v|} \quad \text{and}$$

$$\delta(t) = \frac{|E_t|}{|V_t \otimes V_t|}.$$

If $|T_u \cap T_v| = 0$, $\sum_{u \in V, u \neq v} |T_u \cap T_v| = 0$ or $|V_t \otimes V_t| = 0$, respectively, then we define $\delta(uv)$, $\delta(v)$ and $\delta(t)$ to be 0.

The density of uv is the probability that there is a link between u and v whenever this is possible, i.e., when they are both present. The density of v is the probability that a link between v and any other node exists whenever this is possible, and the density of t is equal to $\delta(G_t)$, the density of the graph G_t , i.e., the probability that a link exists between any two nodes present at time t .

For S defined in Fig. 1 (left), for instance, we obtain $\delta(ab) = \frac{|T_{ab}|}{|T_a \cap T_b|} = \frac{3}{9} = \frac{1}{3}$ and $\delta(bd) = \frac{|T_{bd}|}{|T_b \cap T_d|} = \frac{1}{2} = 0.5$. We also obtain $\delta(d) = \frac{|T_{da}| + |T_{db}| + |T_{dc}|}{|T_d \cap T_a| + |T_d \cap T_b| + |T_d \cap T_c|} = \frac{0+1+0}{2+2+0} = 0.25$ and $\delta(2) = \frac{|E_2|}{|V_2 \otimes V_2|} = \frac{2}{3 \cdot 2/2} = \frac{2}{3}$.

Notice that uv_t is strongly related to the concept of density: it is the probability that u and v are linked together at time t , which is equal to 1 or 0 depending on whether (t, uv) is in E or not. We then have

$$\delta(uv) = \frac{\int_{t \in T} uv_t dt}{\int_{t \in T} u_t \cdot v_t dt}, \quad \delta(v) = \frac{\sum_{u \in V} \int_{t \in T} uv_t dt}{\sum_{u \in V} \int_{t \in T} u_t \cdot v_t dt}, \quad \text{and} \quad \delta(t) = \frac{\sum_{uv \in V \otimes V} uv_t}{\sum_{uv \in V \otimes V} u_t \cdot v_t}.$$

Likewise, $\delta(S) = \frac{\sum_{uv \in V \otimes V} \int_{t \in T} uv_t dt}{\sum_{uv \in V \otimes V} \int_{t \in T} u_t \cdot v_t dt}$.

In a link stream $L = (T, V, E)$, $T_v = T$ for all v and $V_t = V$ for all t , and therefore, $\delta(uv) = \frac{|T_{uv}|}{|T|} = m_{uv}$, $\delta(t) = \frac{|E_t|}{|V \otimes V|} = l_t$, and as shown above, $\delta(L)$ is equal to the average of $\delta(t)$. In a graph-equivalent stream, $\delta(uv) \in \{0, 1\}$, and $\delta(t)$ is equal to the density of the induced graph.

The density $\delta(v)$ of node v is strongly related to its degree, that we introduce in Sect. 8.

6 Sub-streams and clusters

A graph $G' = (V', E')$ is a sub-graph of $G = (V, E)$ if $V' \subseteq V$ and $E' \subseteq E$. This is denoted by $G' \subseteq G$.

Given two graphs $G = (V, E)$ and $G' = (V', E')$, their intersection is the graph $G \cap G' = (V \cap V', E \cap E')$. It is their largest (with respect to inclusion) common sub-graph. Their union is $G \cup G' = (V \cup V', E \cup E')$; it is the smallest graph having both G and G' for sub-graphs.

A cluster C of $G = (V, E)$ is a subset of V . The set of links between nodes in C is $E(C) = \{uv \in E, u \in C \text{ and } v \in C\}$, and $G(C) = (C, E(C))$ denotes the sub-graph of G induced by C .

Given a cluster C , the properties of its induced sub-graph are said to be the properties of C ; for instance, $\delta(C)$ denotes $\delta(G(C))$.

We say that a stream $S' = (T', V', W', E')$ is a *sub-stream* of $S = (T, V, W, E)$ if $T' \subseteq T$, $V' \subseteq V$, $W' \subseteq W$, and $E' \subseteq E$. We denote this by $S' \subseteq S$.

Given two stream graphs $S = (T, V, W, E)$ and $S' = (T', V', W', E')$, their *intersection* is the stream graph $S \cap S' = (T \cap T', V \cap V', W \cap W', E \cap E')$. It is their largest (with respect to inclusion) common sub-stream. Their *union* is $S \cup S' = (T \cup T', V \cup V', W \cup W', E \cup E')$; it is the smallest stream graph having both S and S' for sub-streams.

We define a *cluster* C of $S = (T, V, W, E)$ as a subset of W . We define the set of links between nodes involved in C as $E(C) = \{(t, uv) \in E, (t, u) \in C \text{ and } (t, v) \in C\}$, and we denote by $S(C) = (T, V, C, E(C))$ the *sub-stream of S induced by C* , see Fig. 3.

Given a cluster C , we say that the properties of its induced sub-stream are the properties of C ; for instance, we denote $\delta(S(C))$ by $\delta(C)$. For any v in V , we also denote by T_v^C the set of times at which v is in C , and for any u and v in V , we denote by T_{uv}^C the set of time instants at which u and v are in C and are linked together. For any t in T , we denote by V_t^C the set of nodes present at time t in C and by E_t^C the set of links between nodes in C at time t .

In Fig. 3, for instance, $T_a^C = [1, 4] \cup [5, 8]$, $T_b^C = [5, 9]$, $T_c^C = [3, 8]$ and $T_d^C = \emptyset$; $T_{ab}^C = [6, 8]$, $T_{ac}^C = [3, 4] \cup \{5\}$, and $T_{bc}^C = [5, 8]$; $V_7^C = \{a, b, c\}$ and $E_7^C = \{ab, bc\}$.

Notice that the sub-streams of S induced by its clusters are defined over the same set of nodes V and the same time space T as S . We, therefore, define the sub-stream of S induced by a subset V' of V as the sub-stream induced by the node cluster

$(T \times V') \cap W$, i.e., $(T, V', (T \times V') \cap W, (T \times V' \otimes V') \cap E)$ of S . Likewise, we define the sub-stream of S induced by a subset T' of T as the sub-stream induced by $(T' \times V) \cap W$, i.e., $(T', V, (T' \times V) \cap W, (T' \times V \otimes V) \cap E)$ of S .

For the example in Fig. 3, for instance, the sub-stream induced by $\{a, b, c\}$ and $[6, 9]$ is $([6, 9], \{a, b, c\}, [6, 9] \times \{a, b, c\}, E')$ with $E' = [6, 9] \times \{ab\} \cup [6, 8] \times \{bc\}$.

7 Cliques

A clique of graph G is a cluster C of G of density 1. In other words, all pairs of nodes involved in C are linked together in G . A clique C is maximal if there is no other clique C' such that $C \subset C'$.

We define a *clique* of stream graph S as a cluster C of S of density 1. In other words, all pairs of nodes involved in C are linked in S whenever both are involved in C . A clique C is maximal if there is no other clique C' , such that $C \subset C'$.

We say that a clique is compact (resp. uniform) if its induced sub-stream is compact (resp. uniform). It is then fully defined by a set of nodes and a time interval (resp. a time set) meaning that all pairs of nodes are linked together at all these times.

For instance, in Fig. 4, the cluster $[0, 1] \times \{a, b\}$ is a compact clique. However, it is not maximal, as it is included in $[0, 4] \times \{a, b\}$, which is a maximal compact clique. This clique intersects another maximal compact clique, $[2, 4] \times \{a, b, c\}$. There is a unique other maximal compact clique involving three nodes, $[8, 9] \times \{b, c, d\}$. The

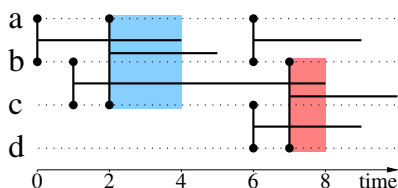


Fig. 4 Examples of maximal compact cliques. We display the two maximal compact cliques involving three nodes of the link stream L of Fig. 1 (right): $[2, 4] \times \{a, b, c\}$ and $[7, 8] \times \{b, c, d\}$. Its other maximal compact cliques are $[0, 4] \times \{a, b\}$, $[6, 9] \times \{a, b\}$, $[2, 5] \times \{a, c\}$, $[1, 8] \times \{b, c\}$, $[7, 10] \times \{b, d\}$, $[6, 9] \times \{c, d\}$ (involving two nodes each)

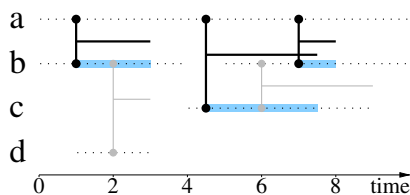


Fig. 5 Two examples of neighborhoods and degrees of nodes. We display in black the links involving the node under concern, and in grey the other links. Left: $N(a) = ([1, 3] \cup [7, 8]) \times \{b\} \cup [4.5, 7.5] \times \{c\}$ is

maximal compact clique $[0, 4] \times \{a, b\}$ is not a maximal clique, because it is, for instance, included in the clique $[0, 4] \times \{a, b\} \cup [6, 9] \times \{c, d\}$ (which is not compact). This clique is not maximal either, as it is, for instance, included in the clique $[0, 4] \times \{a, b\} \cup [6, 9] \times \{c, d\} \cup [5, 6] \times \{d\}$.

A clique in S does not in general induce a clique in $G(S)$: for instance, $[0, 1] \times \{a, b\} \cup [8, 9] \times \{c, d\}$ is a clique for the example in Fig. 4, but $\{a, b, c, d\}$ is not a clique in its induced graph. Instead, for any $[b, e] \subseteq T$ and $X \subseteq V$, if $[b, e] \times X$ is a compact clique in S , then X necessarily is a clique in $G(S)$. However, if $[b, e] \times X$ is maximal in S , then X is not necessarily maximal in $G(S)$, see, for instance, $[0, 4] \times \{a, b\}$ in Fig. 4 ($\{a, b\}$ is a clique in $G(S)$, but it is included in its other clique $\{a, b, c\}$). Conversely, if a cluster X of $G(S)$ is a clique, then, in general, there is no $[b, e]$, such that $[b, e] \times X$ is a compact clique in S , see Viard et al. (2016) for a more detailed discussion and practical evidence of the differences between maximal cliques in streams and their induced graphs. Finally, if one considers a graph-equivalent stream, then its maximal cliques are necessarily compact, and they correspond exactly to the maximal cliques of its induced graph.

8 Neighborhood and degree

In the graph $G = (V, E)$, the neighborhood $N(v)$ of $v \in V$ is the cluster $N(v) = \{u, uv \in E\}$, and the degree $d(v)$ of v is the number of nodes in this cluster, which is equal to the number of links involving v . We then have $\sum_{v \in V} d(v) = 2 \cdot m$.

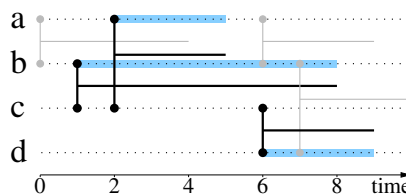
The average degree in G is $d(G) = \frac{1}{n} \cdot \sum_{v \in V} d(v)$, and the following relation between density and average degree holds: $\delta(G) = \frac{d(G)}{n-1}$.

In a stream graph $S = (T, V, W, E)$, we define the *neighborhood* of a node v as the following cluster:

$$N(v) = \{(t, u), (t, uv) \in E\}$$

and the *degree* $d(v)$ of v as the number of nodes in this cluster. As with graphs, this is equal to the number of links involving v :

$$d(v) = \frac{|N(v)|}{|T|} = \sum_{u \in V} \frac{|T_{uv}|}{|T|} = \sum_{u \in V} m_{uv}.$$



in blue, leading to $d(a) = \frac{3}{10} + \frac{3}{10} = 0.6$. Right: $N(c) = [2, 5] \times \{a\} \cup [1, 8] \times \{b\} \cup [6, 9] \times \{d\}$ is in blue, leading to $d(c) = \frac{13}{10} = 1.3$



Fig. 6 These two stream graphs have density 1 (all possible links exist), 2 nodes, and 1 link. However, the leftmost one has average node degree $d(V) = \frac{|T_a|}{|W|}d(a) + \frac{|T_b|}{|W|}d(b) + \frac{|T_c|}{|W|}d(c) = \frac{2}{8}0.5 + \frac{4}{8}1 + \frac{2}{8}0.5 = 0.75$ and the rightmost one has average node degree 1

With this definition, each node u contributes to the degree of v proportionally to the duration of its links with v , see Fig. 5 for an illustration.

As with graphs, the sum of the degree of all nodes in S is equal to twice the number of links in S : $\sum_{v \in V} d(v) = \sum_{v \in V} \sum_{u \in V} \frac{|T_{uv}|}{|T|} = 2 \cdot m$.

We now define the *average node degree of S* as follows:

$$d(V) = \frac{1}{n} \cdot \sum_{v \in V} n_v \cdot d(v) = \sum_{v \in V} \frac{|T_v|}{|W|} \cdot d(v).$$

In this definition, the contribution of each node v to the average node degree of S is weighted by its presence duration $|T_v|$.

As a consequence, there is no direct relation between the average node degree and the total number of links of S , as illustrated in Fig. 6. Likewise, the usual relation between average node degree and density does not hold in general.

Instead, in a link stream $L = (T, V, E)$, we have $n_v = 1$ for all v , and therefore, the following relation holds: $d(L) = \frac{1}{n} \cdot \sum_{v \in V} d(v) = \frac{2 \cdot m}{n}$. We have seen in Sect. 5 that $\delta(L) = \frac{2 \cdot m}{n \cdot (n-1)}$; therefore, the relation $\delta(L) = \frac{d(L)}{n-1}$ also holds.

Going further, we have $\delta(v) = \frac{\sum_{u \in V, u \neq v} |T_{uv}|}{\sum_{u \in V, u \neq v} |T|} = \frac{|N(v)|}{(|V|-1) \cdot |T|} = \frac{d(v)}{n-1}$.

Finally, if we consider a graph-equivalent stream, then the degree of any of its nodes is equal to the degree of this node in the corresponding graph, and the average node degree is preserved.

The definitions above generalize graph concepts to stream graphs. However, the temporal features of stream graphs make it natural to consider other generalizations that we now introduce.

Given a stream graph $S = (T, V, W, E)$, we define the instantaneous neighborhood of a node v at time t as $N_t(v) = \{u, (t, uv) \in E\}$, and the instantaneous degree of v at time t as the number of nodes in $N_t(v)$. If v is not involved in S at time t , then $N_t(v) = \emptyset$ and $d_t(v) = 0$. If v is involved in S at time t , then $N_t(v)$ and $d_t(v)$ are nothing, but the neighborhood and the degree of v in the graph G_t induced by S at time t .

The degree of v is exactly the average instantaneous degree of v at time t for all t in T : $d(v) = \int_t \frac{d_t(v)}{|T|} dt$. It is also natural to consider the average only for t in T_v , which is the expected instantaneous degree of v when it is involved in S ; we call it the expected degree of v and denote it by $\hat{d}(v) = \int_t \frac{d_t(v)}{|T_v|} dt$.

We also consider these two ways to average instantaneous degrees over nodes; either over all nodes in V , leading to $\sum_v \frac{d_t(v)}{|V|}$ which we call the degree at t and denote by $d(t)$, by analogy with $d(v) = \int_t \frac{d_t(v)}{|T|} dt$; or over nodes in V_t only, leading to $\hat{d}(t) = \sum_v \frac{1}{|V_t|} d_t(v)$, the expected degree at time t , which is exactly the average degree of G_t .

Let us now consider ways to average $d(v)$ and $d(t)$ over S as a whole.

The weighted average of $d(v)$, $\sum_{v \in V} \frac{|T_v|}{|W|} d(v) = \frac{1}{n} \sum_{v \in V} n_v \cdot d(v)$, is the average node degree of S , denoted by $d(V)$ and introduced above. Similarly, we introduce the weighted average of $d(t)$, $\int_t \frac{|V_t|}{|W|} d(t) dt = \frac{1}{k} \int_t k_t \cdot d(t) dt$, which we call the average time degree of S and denote by $d(T)$. Notice that, in general, $d(V) \neq d(T)$, as illustrated in Fig. 7.

For averages over all V and T , we obtain a unique quantity: $\sum_v \frac{1}{|V|} d(v) = \frac{2|E|}{|T \times V|} = \int_t \frac{1}{|T|} d(t) dt$, which is the average instantaneous degree of v at time t for a random (t, v) in $T \times V$; we call it the degree of S and denote it by $d(S)$.

Finally, it is also natural to consider the average instantaneous degree for (t, v) in W only: $\frac{\sum_v \int_t d_t(v) dt}{|W|} = \frac{\int_t \sum_v d_t(v) dt}{|W|} = \frac{2|E|}{|W|} = \frac{2m}{n}$. We call it the average expected degree of S and denote it by $\hat{d}(S)$.

In a link stream, we have $d(v) = \hat{d}(v)$, $d(t) = \hat{d}(t)$, and $d(V) = d(T) = d(S) = \hat{d}(S)$. In a graph-equivalent stream, we have in addition $d(t) = d(V)$, and, as already said, $d(V)$ is the average degree in the corresponding graph and $d(v)$ is the degree of v in this graph.

9 Clustering coefficient and transitivity ratio

In the graph $G = (V, E)$, the clustering coefficient of a given node v is the density of its neighborhood: $cc(v) = \delta(N(v))$. In other words, $cc(v)$ is the probability that two randomly chosen neighbors of v are linked together in G . By definition of the density, if $d(v) < 2$ then $cc(v) = 0$. The clustering coefficient of G as a whole is the average clustering coefficient of all its nodes: $cc(G) = \frac{1}{n} \cdot \sum_{v \in V} cc(v)$. It is the probability when one takes a random node v that this node has more than one neighbor and that two of its neighbors chosen at random are linked together.

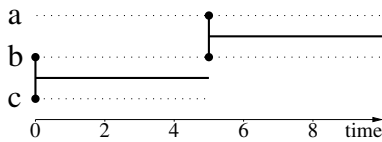


Fig. 7 Simple stream graph $S = (T, V, W, E)$ such that $d(V) \neq d(T)$. Indeed, we compute $d(V)$ with $n = 2.5$, $n_a = n_b = 1$, $n_c = 0.5$, $d(a) = 0.5$, $d(b) = 1$, and $d(c) = 0.5$, leading to $d(V) = \frac{1}{n} \sum_{v \in V} n_v \cdot d(v) = \frac{1}{2.5} (1 \cdot 0.5 + 1 \cdot 1 + 0.5 \cdot 0.5) = 0.7$, and we compute $d(T)$ with $k = \frac{25}{3}$, $k_t = 1$ for $t \in [0, 5]$, $k_t = \frac{2}{3}$ for $t \in [5, 10]$, and $d(t) = \frac{2}{3}$ for all t , leading to $d(T) = \frac{1}{k} \int_t k_t \cdot d(t) dt = \frac{3}{25} (\int_0^5 1 \cdot \frac{2}{3} dt + \int_5^{10} \frac{2}{3} \cdot \frac{2}{3} dt) = \frac{3}{25} (5 \cdot \frac{2}{3} + 5 \cdot \frac{4}{9}) = \frac{3}{25} \cdot \frac{50}{9} = \frac{2}{3}$

$$\begin{aligned} \vee = & [2, 4] \times \{(b, a, c), (c, a, b)\} \cup ([1, 4] \cup [6, 8]) \times \{(a, b, c), (c, b, a)\} \cup [7, 8] \times \{(c, b, d), (d, b, c)\} \cup [7, 9] \\ & \times \{(a, b, d), (d, b, a)\} \cup [2, 5] \times \{(a, c, b), (b, c, a)\} \cup [6, 8] \times \{(b, c, d), (d, c, b)\} \cup [7, 9] \times \{(b, d, c), (c, d, b)\} \end{aligned}$$

In G , the triplet (u, v, w) in $V \times V \times V$ with $u \neq v \neq w$ is a connected triplet if there is both a link between u and v and between v and w , i.e., $uv \in E$ and $vw \in E$. The set of all connected triplets of G is denoted by \vee . If in addition, there is a link between u and w , i.e., $uw \in E$, then (u, v, w) is a triangle and the set of all triangles of G is denoted by ∇ . The transitivity ratio of G is the probability, when one takes a random connected triplet, that it is a triangle: $tr(G) = \frac{|\nabla|}{|\vee|}$.

In a stream graph $S = (T, V, W, E)$, we define the *clustering coefficient of a given node v* as the density of its neighborhood:

$$cc(v) = \delta(N(v)) = \frac{\sum_{uw \in V \otimes V} |T_{vu} \cap T_{vw} \cap T_{uw}|}{\sum_{uw \in V \otimes V} |T_{vu} \cap T_{vw}|}$$

In other words, $cc(v)$ is the probability when one takes two random neighbors u and w of v at time t , i.e., a random (t, uw) in $T \times V \otimes V$, such that (t, vu) and (t, vw) are in E , that u is linked to w in S at time t , i.e., that (t, uw) is in E . By definition of density, if there is no such triplet then $cc(v) = 0$, see Fig. 8 for an illustration.

We define the *node clustering coefficient of S* as the average clustering coefficient of all its nodes, weighted by their presence in S :

$$cc(V) = \frac{1}{n} \cdot \sum_{v \in V} n_v \cdot cc(v) = \sum_{v \in V} \frac{|T_v|}{|W|} \cdot cc(v)$$

In S , we say that $(t, (u, v, w))$ in $T \times (V \times V \times V)$ with $u \neq v \neq w$ is a connected triplet if at time t there is both a link between u and v and between v and w , i.e., $(t, uv) \in E$

and $(t, vw) \in E$. We denote by \vee the set of all connected triplets of S . If in addition, there is a link between u and w at time t , i.e., $(t, uw) \in E$, then we say that $(t, (u, v, w))$ is a triangle and we denote the set of all triangles of S by ∇ . We define the *transitivity ratio $tr(S)$ of S* as the probability, when one takes a random connected triplet, that it is a triangle: $tr(S) = \frac{|\nabla|}{|\vee|}$.

In Fig. 8, for instance, the set \vee of all connected triplets contains $[2, 4] \times \{(b, a, c), (c, a, b)\}$, because for all t in $[2, 4]$, the links $(t, ba) = (t, ab)$ and $(t, ac) = (t, ca)$ are in E . The set ∇ of all triangles also contains $[2, 4] \times \{(b, a, c), (c, a, b)\}$, since for all t in $[2, 4]$, the link $(t, bc) = (t, cb)$ also is in E . This leads to

and

$$\begin{aligned} \nabla = & [2, 4] \times \{(b, a, c), (c, a, b), (a, b, c), (c, b, a), \\ & (a, c, b), (b, c, a)\} \cup [7, 8] \\ & \times \{(c, b, d), (d, b, c), (b, c, d), (d, c, b), (b, d, c), (c, d, b)\} \end{aligned}$$

$$\text{We thus obtain } tr(S) = \frac{2 \cdot 6 + 1 \cdot 6}{2 \cdot 2 + (3+2) \cdot 2 + 1 \cdot 2 + 2 \cdot 2 + 3 \cdot 2 + 2 \cdot 2 + 2 \cdot 2} = \frac{9}{17} \sim 0.52$$

In a link stream $L = (T, V, E)$, $n_v = 1$ for all v , and therefore, $cc(V) = \frac{1}{n} \sum_v cc(v)$. In a graph-equivalent stream, $cc(v)$ in the stream is equal to $cc(v)$ in the corresponding graph G , and $cc(V)$ is equal to $cc(G)$. Likewise, the transitivity ratio of a graph-equivalent stream is equal to the one of its corresponding graph.

Like with degrees in Sect. 8, the temporal features of stream graphs make it natural to consider other generalizations of clustering coefficient that we now introduce.

Given a stream graph $S = (T, V, W, E)$, we define the *instantaneous clustering coefficient of v at time t* as $cc_t(v) = \frac{\sum_{uw} vu_t \cdot vw_t \cdot uw_t}{\sum_{uw} vu_t \cdot vw_t}$. If v is not involved in S at time t , then $cc_t(v) = 0$. If v is involved in S at time t , then $cc_t(v)$ is exactly the clustering coefficient of v in G_t .

Like for degrees, it is natural to consider the following ways to average the instantaneous clustering coefficient:

$$\int_t \frac{cc_t(v)}{|T_v|} dt, \int_t \frac{cc_t(v)}{|T|} dt, \sum_v \frac{cc_t(v)}{|V_t|} = cc(G_t), \text{ and } \sum_v \frac{cc_t(v)}{|V|}$$

Notice that $cc(v) \neq \int_t \frac{cc_t(v)}{|T|} dt$, but $cc(v)$ is related to $cc_t(v)$ by: $cc(v) = \frac{\sum_{uw} |T_{vu} \cap T_{vw} \cap T_{uw}|}{\sum_{uw} |T_{vu} \cap T_{vw}|} = \frac{\sum_{uw} \int_t vu_t \cdot vw_t \cdot uw_t dt}{\sum_{uw} \int_t vu_t \cdot vw_t dt} = \frac{\int_t cc_t(v) \sum_{uw} vu_t \cdot vw_t dt}{\int_t \sum_{uw} vu_t \cdot vw_t dt}$.

It is then natural to define $cc(t)$ as such: $cc(t) = \frac{\sum_v cc_t(v) \sum_{uw} vu_t \cdot vw_t}{\sum_v \sum_{uw} vu_t \cdot vw_t}$, which is exactly $tr(G_t)$.

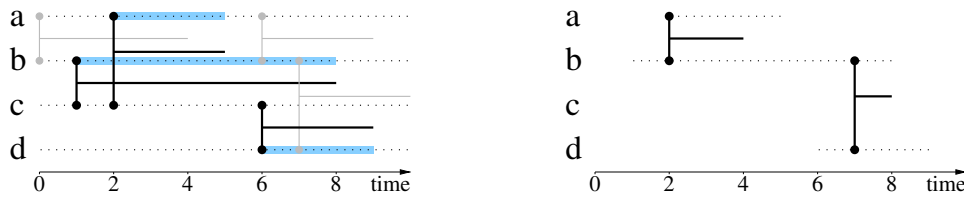


Fig. 8 Example of clustering coefficient. Left: we display in black the links involving node c in S , in grey the other links, and in blue the neighborhood of c , like in Fig. 5. Right: the sub-stream induced by

$N(c)$. The clustering coefficient of c in S , $cc(c)$, is the density of this sub-stream, $S(N(c))$. Here, we obtain $cc(c) = \delta(S(N(c))) = \frac{3}{5} = 0.6$

One may then consider the following ways to average $cc(v)$ and $cc(t)$: $\sum_v \frac{1}{|V|} cc(v)$, $\int_t \frac{1}{|T|} cc(t) dt$, $cc(V) = \sum_v \frac{|T_v|}{|W|} cc(v)$, $cc(T) = \int_t \frac{|V_t|}{|W|} cc(t) dt$, and $cc(S) = \int_t \frac{1}{|T|} \sum_v \frac{cc(v)}{|V|} dt = \sum_v \frac{1}{|V|} \int_t \frac{cc(v)}{|T|} dt = \frac{1}{|T \times V|} \sum_v \int_t cc_t(v) dt$, thus introducing the *time clustering coefficient* of S , $cc(T)$, and the *clustering coefficient of S* , $cc(S)$, by extending the definition of $cc(V)$, like we did for $d(T)$ and $d(S)$ from $d(V)$ in Sect. 8.

Finally, notice that $cc(t)$, $cc(v)$, and $tr(S)$ may be obtained from the definition of $cc_t(v) = \frac{\sum_{uw} vu_t \cdot vw_t \cdot uw_t}{\sum_{uw} vu_t \cdot vw_t}$ as follows: $\frac{\sum_v \sum_{uw} vu_t \cdot vw_t \cdot uw_t}{\sum_v \sum_{uw} vu_t \cdot vw_t} = cc(t)$; $\frac{\int_t \sum_{uw} vu_t \cdot vw_t \cdot uw_t dt}{\int_t \sum_{uw} vu_t \cdot vw_t dt} = cc(v)$; and $\frac{\int_t \sum_v \sum_{uw} vu_t \cdot vw_t \cdot uw_t dt}{\int_t \sum_v \sum_{uw} vu_t \cdot vw_t dt} = tr(S)$.

10 Neighborhoods and degrees in and of clusters

Given a graph $G = (V, E)$ and a cluster C of G , the internal neighborhood of v in C is $N_C(v) = N(v) \cap C = \{u \in C, uv \in E\}$ and its external neighborhood is $\overline{N}_C(v) = N(v) \setminus C = \{u \notin C, uv \in E\}$. The internal and external degrees of v in C , denoted, respectively, by $d_C(v)$ and $\overline{d}_C(v)$, are the number of nodes in $N_C(v)$ and $\overline{N}_C(v)$. The internal neighborhood and the internal degree of v in C are also its neighborhood and degree in $G(C)$.

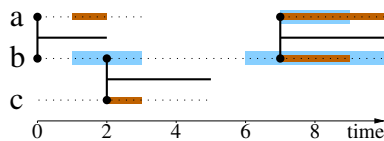


Fig. 9 Example of cluster (in blue) with its neighborhood (in red). $C = ([1, 3] \cup [6, 10]) \times \{b\} \cup [7, 9] \times \{a\}$. We then have $N_C(a) = [7, 9] \times \{b\}$, $\overline{N}_C(a) = \emptyset$, $N_C(b) = [7, 9] \times \{a\}$, $\overline{N}_C(b) = ([1, 2] \cup [9, 10]) \times \{a\} \cup [2, 3] \times \{c\}$, $N_C(c) = \overline{N}_C(c) = \emptyset$, and $N(C) = ([1, 2] \cup [7, 10]) \times \{a\} \cup [7, 9] \times \{b\} \cup [2, 3] \times \{c\}$. The intersection of $N(C)$ with C appears as overlaps between blue and red areas, leading to $d(C) = \frac{|[7,9] \times \{b\} \cup [7,9] \times \{a\}|}{10} = 0.4$ and $\overline{d}(C) = \frac{|([1,2] \cup [9,10]) \times \{a\} \cup [2,3] \times \{c\}|}{10} = 0.3$

The average degree in C , denoted by $d_C(C)$ or simply d_C , is the average degree of $G(C)$; it is equal to the average internal degree of nodes in C .

The neighborhood $N(C)$ of a cluster C is $N(C) = \cup_{v \in C} N(v)$. Notice that $N(C)$ may intersect C , but it is not included in C in general. The numbers of nodes in $N(C) \cap C$ and $N(C) \setminus C$ are often called the internal and external degrees of C , respectively, denoted by $d(C)$ and $\overline{d}(C)$.

Given a stream graph $S = (T, V, W, E)$ and a cluster C of S , we define the *internal neighborhood* of v involved in C as $N_C(v) = \cup_{(t,u) \in C} \{(t, u) \in C, (t, uv) \in E\}$ and the *external neighborhood* of v as $\overline{N}_C(v) = \cup_{(t,u) \in C} \{(t, u) \notin C, (t, uv) \in E\}$. Notice that, unlike for graphs, $N_C(v) \neq N(v) \cap C$ and $\overline{N}_C(v) \neq N(v) \setminus C$, and therefore, $N_C(v) \cup \overline{N}_C(v) \neq N(v)$ in general. Indeed, we take into account the neighbors of v only when v is involved in C , see Fig. 9 for an illustration.

We define the *internal and external degree* of v involved in C , denoted, respectively, by $d_C(v)$ and $\overline{d}_C(v)$, as the number of nodes in $N_C(v)$ and $\overline{N}_C(v)$. The internal neighborhood and the internal degree of v are its neighborhood and degree in $S(C)$.

We define the *average node degree* in C , denoted by d_C , as the average node degree of $S(C)$; it is the average internal degree of nodes involved in C , weighted by their presence in C : $d_C = \sum_v \frac{|T_v^C|}{|C|} d_C(v)$.

We define the *neighborhood* $N(C)$ of cluster C as $N(C) = \cup_{(t,v) \in C} \{(t, u), (t, uv) \in E\}$, see Fig. 9. Notice that $N(C)$ may intersect C , but it is not necessarily included in C . We call the numbers of nodes in $N(C) \cap C$ and $N(C) \setminus C$ the *internal and external degrees* of C , respectively, denoted by $d(C)$ and $\overline{d}(C)$.

In a graph-equivalent stream, any compact cluster $C = T_C \times V_C$ induces the cluster V_C in the corresponding graph, and the internal (resp. external) neighborhood of any node involved in C is equal to T_C times its internal (resp. external) neighborhood in V_C . Likewise, the neighborhood of C in the stream is equal to T_C times the neighborhood of V_C in the graph.

11 Relations between clusters and quotient stream

Let us consider a family $F = (C_1, C_2, \dots, C_k)$ of k clusters of $G = (V, E)$. The quotient graph induced by F is the graph $Q = (\{1, 2, \dots, k\}, E')$, where ij is in E' if $i \neq j$ and there is a u in C_i and a v in C_j , such that uv is in E .

Intuitively, the quotient graph captures relations between clusters: its nodes are the clusters of the original graph, and there is a link between two clusters if they contain nodes that are linked together in the original graph.

Notice that, if $F = (\{v\})_{v \in V}$, then Q is equivalent to G .

The intra-cluster density $\delta(F)$ of F is the probability when one takes a random pair of distinct nodes in a same cluster of F that there is a link between them in G :

$$\delta(F) = \frac{\sum_i |C_i \otimes C_i \cap E|}{\sum_i |C_i \otimes C_i|}$$

The inter-cluster density $\bar{\delta}(F)$ of F is the probability, when one takes a random pair of distinct nodes in two different clusters of F , that there is a link between

$$\bar{\delta}(F) = \frac{\sum_{i \neq j} |C_i \otimes C_j \cap E|}{\sum_{i \neq j} |C_i \otimes C_j|}$$

The density $\delta(C)$ of C is equal to the intra-cluster density of the family composed of C alone or the inter-cluster density of the family (C, C) . The external density of C , denoted by $\bar{\delta}(C)$, is defined as the inter-cluster density of the family $(C, V \setminus C)$. It is the probability when one takes a random node u in C and a random node v outside C that there is a link between them in G .

Given a family $F = (C_1, C_2, \dots, C_k)$ of k clusters of $S = (T, V, W, E)$, we define the *quotient stream induced by F* as the stream graph $Q = (T, \{1, 2, \dots, k\}, W', E')$, where (t, i) is in W' when there is a v , such that (t, v) is in C_i , and (t, ij) is in E' when $i \neq j$ and there is a (t, u) in C_i and (t, v) in C_j , such that (t, uv) is in E , see Fig. 10 for an illustration.

Intuitively, the quotient stream captures relations between clusters: its nodes are the clusters of the original stream, and there is a link between two clusters at a given time instant if they contain nodes that are linked together in the original stream at this time instant. In Fig. 10, for instance, there is a link between clusters A and B from time 9 to time 10 in the quotient stream, because during this time interval, a node of A and a node of B (b and d , respectively) are linked together in the original stream.

Notice that if $F = (T_v \times \{v\})_{v \in V}$, then Q is equivalent to S .

The *intra-cluster density* $\delta(F)$ of F is the probability, when one takes a random element (t, uv) of $T \times V \otimes V$, such that (t, u) and (t, v) are in a same cluster of F , that there is a link (t, uv) in S :

$$\delta(F) = \frac{\sum_i \sum_{u \neq v} |T_u^{C_i} \cap T_v^{C_i} \cap T_{uv}|}{\sum_i \sum_{u \neq v} |T_u^{C_i} \cap T_v^{C_i}|}$$

The *inter-cluster density* $\bar{\delta}(F)$ of F is the probability, when one takes a random element (t, uv) of $T \times V \otimes V$, such that (t, u) and (t, v) are in different clusters of F , that there is a link (t, uv) in S :

$$\bar{\delta}(F) = \frac{\sum_{i \neq j} \sum_{u \neq v} |T_u^{C_i} \cap T_v^{C_j} \cap T_{uv}|}{\sum_{i \neq j} \sum_{u \neq v} |T_u^{C_i} \cap T_v^{C_j}|}$$

As with graphs, the density $\delta(C)$ of C is equal to the intra-cluster density of the family composed of C alone, or the inter-cluster density of the family (C, C) . We define the *external density of C* , denoted by $\bar{\delta}(C)$, as the inter-cluster density of the family $(C, W \setminus C)$. It is the probability when one takes a random (t, u) in C and a random (t, v) in W but outside C that there is a link (t, uv) between them in S .

12 Line streams

The line graph \hat{G} of $G = (V, E)$ is the graph $\hat{G} = (E, \hat{E})$, where each node is a link of G and two nodes are linked together if they have an extremity in common: if $A = uv$ and $B = xy$ are two elements of E , then AB is in \hat{E} if $\{u, v\} \cap \{x, y\} \neq \emptyset$. In general, $\hat{G} \neq G$.

The set of links in G involving a given node v corresponds to a cluster in \hat{G} and this cluster has density 1. If, instead, we consider a set C of independent links (i.e., if uv and xy are in C , then $\{u, v\} \cap \{x, y\} = \emptyset$), then the corresponding cluster in \hat{G} has density 0. Finally, if we consider a clique of G of more than three nodes, then the cluster of \hat{G} corresponding to the links of this clique has density lower than 1, and it tends to 0 when the size of the clique grows.

We define the *line stream* \hat{S} of $S = (T, V, W, E)$ as the stream graph $\hat{S} = (T, \hat{V}, \hat{W}, \hat{E})$. The set $\hat{V} = \{uv, \exists(t, uv) \in E\}$ is the set of links in $G(S)$. The set \hat{W} is such that each node $A = uv$ is present in \hat{S} during the times at which the link uv is present in S , leading to $\hat{W} = E$. Finally, for all $A = uv$ and $B = xy$ in \hat{V} , there is a link (t, AB) in \hat{E} if $\{u, v\} \cap \{x, y\} \neq \emptyset$ and $\{(t, uv), (t, xy)\} \subseteq E$. In other words, A and B are linked together at time t if they have an extremity in common and are both present at time t , see Fig. 11 for an illustration. As with graphs, in general, $\hat{S} \neq S$.

The set of links in S involving a given node v corresponds to a cluster in \hat{S} , and this cluster has density 1. If, instead, we consider a set C of independent links (i.e., if (t, uv) and (s, xy) are in C , then $\{u, v\} \cap \{x, y\} = \emptyset$ or $t \neq s$), then the corresponding cluster in \hat{S} has density 0. As with graphs, the density of a cluster of \hat{S} corresponding to the links of a clique of S tends to 0.

For all t , the graph induced by \hat{S} at time t is the line graph of G_t . As a consequence, the line stream of a graph-equivalent stream is a graph-equivalent stream too, and its

corresponding graph is the line graph of the graph corresponding to the initial stream.

13 *k*-cores

The *k*-core of the graph $G = (V, E)$ is its largest cluster $C^k \subseteq V$, such that for all v in C^k , $d(v) \geq k$ in the sub-graph $G(C^k)$ of G induced by C^k . This cluster is unique for a given k and $C^{k+1} \subseteq C^k$ for all k . The *k*-core may be computed by iteratively removing from G all elements of V of degree lower than k . The 0-core of G is V , and the *k*-core contains all cliques of size $k + 1$ of G . The core number of v in V is the largest k , such that $v \in C^k$. The *k*-shell of G is $C^k \setminus C^{k+1}$.

We define the *k*-core of the stream graph $S = (T, V, W, E)$ as its largest cluster $C^k \subseteq W$, such that for all (t, v) in C^k , $d_t(v) \geq k$ in the sub-stream $S(C^k)$ of S induced by C^k , see Fig. 12 for an illustration.

This cluster is unique for a given k , and $C^{k+1} \subseteq C^k$ for all k . The *k*-core may be computed by iteratively removing from S all elements of W of instantaneous degree lower than k . The 0-core of S is W , and the *k*-core contains all compact cliques of S involving $k + 1$ nodes. We define the *core number* of (t, v) in W as the largest k , such that $(t, v) \in C^k$, and the *k*-shell of S as $C^k \setminus C^{k+1}$.

Notice that, for all t , the set of nodes v , such that $(t, v) \in C^k$ is exactly *k*-core of G_t . As a consequence, the *k*-core of a graph-equivalent stream is T times the *k*-core of the corresponding graph.

14 Paths and distances

In a graph $G = (V, E)$, a path P from $u \in V$ to $v \in V$ is a sequence $(u_0, v_0), (u_1, v_1), \dots, (u_k, v_k)$ of elements of $V \times V$, such that $u_0 = u, v_k = v$, and for all $i, u_i = v_{i-1}$ and $u_i v_i \in E$. The path P involves u, v , and v_i for all $i \in [1, k - 1]$, and the integer $k + 1$ is the length of P . If there exists a path from u to v in G , then v is reachable from u , which is denoted by $u \dashv\vdash v$. Reachability is symmetric ($u \dashv\vdash v$ implies $v \dashv\vdash u$) and transitive ($u \dashv\vdash v$ and $v \dashv\vdash w$ implies $u \dashv\vdash w$).

A subpath Q of P is a subsequence $(u_i, v_i), (u_{i+1}, v_{i+1}), \dots, (u_j, v_j)$ of the sequence defining P , with $j \geq i$. Then, Q is a path from u_i to v_j .

The path P is a cycle if $k > 0$ and $u = v$. In other words, it is a non-empty path from v to itself. If P has no subpath that is a cycle, then P is a simple path. If P is a cycle and has no subpath other than P itself that is a cycle, then P is a simple cycle. If there exists no simple cycle in G , then G is acyclic. If Q is a subpath of P and is a cycle from u_i to v_j (hence, $v_{i-1} = u_i = v_j = u_{j+1}$), then $P' = (u_0, v_0), \dots, (u_{i-1}, v_{i-1}), (u_{j+1}, v_{j+1}), \dots, (u_k, v_k)$ also is a path from u to v . If one iteratively removes the cycles of P in this way, one eventually obtains a simple path from u to v .

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 is 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 .

In a stream graph $S = (T, V, W, E)$, a path P from $(\alpha, u) \in W$ to $(\omega, v) \in W$ is a sequence $(t_0, u_0, v_0), (t_1, u_1, v_1), \dots, (t_k, u_k, v_k)$ of elements of $T \times V \times V$, such that $u_0 = u, v_k = v, t_0 \geq \alpha, t_k \leq \omega$, for all $i, t_i \leq t_{i+1}, v_i = u_{i+1}$, and $(t_i, u_i v_i) \in E, [\alpha, t_0] \times \{u\} \subseteq W, [t_k, \omega] \times \{v\} \subseteq W$, and for all $i, [t_i, t_{i+1}] \times \{v_i\} \subseteq W$.

We say that P involves $(t_0, u), (t_k, v)$, and (t, v_i) for all $i \in [1, k - 1]$ and $t \in [t_i, t_{i+1}]$. We say that path P starts at t_0 , arrives at t_k , has length $k + 1$ and duration $t_k - t_0$, see Fig. 13 for an illustration.

If there exists a path from (α, u) to (ω, v) in S , we say that (ω, v) is reachable from (α, u) , which we denote by $(\alpha, u) \dashv\vdash (\omega, v)$. Notice that reachability is not symmetric: if $(\alpha, u) \dashv\vdash (\omega, v)$, then in general, $(\omega, v) \not\dashv\vdash (\alpha, u)$ (in particular, this is always true if $\alpha \neq \omega$). We say that v is reachable from u if there exists α and ω , such that $(\alpha, u) \dashv\vdash (\omega, v)$, which we also denote by $u \dashv\vdash v$. Reachability is not symmetric in this case either: in Fig. 13, for instance, $d \dashv\vdash c$ (through P_1) but $c \not\dashv\vdash d$. Furthermore, unlike in graphs, reachability is not transitive: in Fig. 13, for instance, $c \dashv\vdash a$ and $a \dashv\vdash d$ but $c \not\dashv\vdash d$. We discuss reachability in more details and we give more complex examples in Sect. 15.



Fig. 10 Example of quotient stream induced by a family of clusters. Left: a stream graph and a family $F = (A, B, C)$ of clusters with $A = [0, 3] \times \{a\} \cup [7, 10] \times \{b\}$ (in red), $B = [2, 6] \times \{b\} \cup [8, 10] \times \{d\}$ (in blue), and $C = [3, 8] \times \{c\} \cup [0, 5] \times \{d\}$ (in green). Right: the

induced quotient stream. For instance, there is a link between A and C from time 7 to time 8, because there is a link between b and c at these times, and b is in A and c is on C at these times

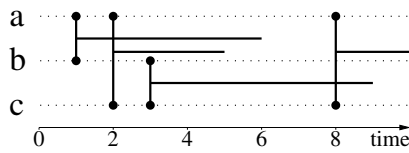
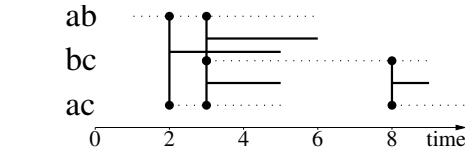


Fig. 11 Stream graph and its line stream. For instance, the node ab is present in the line stream from time 1 to time 6, because a and b are linked together from time 1 to time 6 in the original stream. There is



a link between nodes ab and bc in the line stream at time 4, because $\{a,b\} \cap \{b,c\} = \{b\} \neq \emptyset$ and $(4, ab)$ and $(4, bc)$ are both present in the original stream

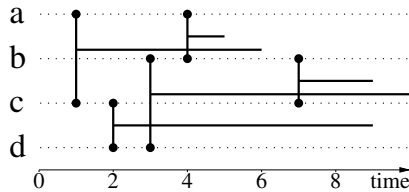
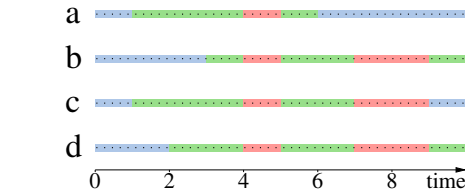


Fig. 12 Link stream L , its k -shells and its 2-core. Each color corresponds to a k -shell of L : its 0-shell in blue, its 1-shell in green, and its 2-shell in red. In this example, the 2-shell also is the 2-core of L . For instance, $(2, a)$ is not in the 2-core, since $d_2(a) = 1$ in L . As a



consequence, although $d_2(c) = 2$ in L , since $(2, c)$ is linked to $(2, a)$, it cannot have instantaneous degree 2 in the 2-core, and therefore, $(2, c)$ is not in the 2-core either

A *subpath* Q of path P is a subsequence $(t_i, u_i, v_i), (t_{i+1}, u_{i+1}, v_{i+1}), \dots, (t_j, u_j, v_j)$ of the sequence defining P , with $j \geq i$. Then, Q is a path from (t_i, u_i) to (t_j, v_j) . In Fig. 13, for instance, $Q_1 = (5, a, c)$, $Q_2 = (3, b, a), (7.5, a, b)$ and $Q_3 = (5, a, c), (6.5, c, b), (7.5, b, a)$ are subpaths of P_1, P_2 , and P_3 , respectively.

The path P is a *cycle* if $u = v$ and $[\alpha, \omega] \times \{v\} \subseteq W$. In other words, it is a path from v at time α to itself at time ω , such that v is present at all times from α to ω . This means that there is a path of length and duration 0 (i.e., the empty sequence) from (α, v) to (ω, v) in S , which makes stream cycles similar to graph cycles: they are non-empty paths equivalent to the empty path. For instance, Q_3 defined above is a cycle, but Q_2 is not, since b is not present from time 3 to time 7.5.

If P has no subpath that is a cycle, then we say that P is a *simple path*. If P is a cycle and has no subpath other than P itself that is a cycle, then P is a simple cycle. If there exists no simple cycle in S then S is *acyclic*.

If Q is a subpath of P and is a cycle from (t_i, u_i) to (t_j, v_j) (hence, $t_j \geq t_i$, $v_{i-1} = u_i = v_j = u_{j+1}$, and $[t_{i-1}, t_{j+1}] \times \{u_i\} \subseteq W$), then $P' = (t_0, u_0, v_0), \dots, (t_{i-1}, u_{i-1}, v_{i-1}), (t_{j+1}, u_{j+1}, v_{j+1}), \dots, (t_k, u_k, v_k)$ also is a path from (α, u) to (ω, v) . If one iteratively removes the cycles of P in this way, one eventually obtains a simple path from (α, u) to (ω, v) . In Fig. 13, for instance, P_1 and P_2 are simple paths, but P_3 is not. Instead, the path $(2, b, a)$ obtained by removing Q_3 from P_3 is simple path.

Paths in stream graphs are quite different from paths in graphs. First, as already said, their temporal nature makes them not symmetric: the existence of a path from u to v does

not imply the existence of a path from v to u . In addition, paths in stream graphs have a length like in graphs but also a duration. This leads to the following set of definitions that capture different notions for the cost of reaching a node from another one.

We say that P is a *shortest path* from (α, u) to (ω, v) if it has minimal length, and we call this length the *distance* from (α, u) to (ω, v) , denoted by $\partial((\alpha, u), (\omega, v))$. The distance $\partial(u, v)$ from u to v is the minimal such distance for all α and ω in T , and a shortest path from u to v is a path from u to v with length $\partial(u, v)$. For instance, in Fig. 13, the path P_1 is a shortest path from $(1, d)$ to $(9, c)$, but P_2 is not. It is impossible to reach c from d with a shorter path; therefore, P_1 also is a shortest path from d to c and $\partial(d, c) = 3$.

We say that P is a *fastest path* from (α, u) to (ω, v) if it has minimal duration, and we call this duration the *latency* from (α, u) to (ω, v) , denoted by $\ell((\alpha, u), (\omega, v))$. The latency $\ell(u, v)$ from u to v is the minimal such latency for all α and ω in T , and a fastest path from u to v is a path from u to v with duration $\ell(u, v)$. For instance, in Fig. 13, the path P_1 is not a fastest path from $(1, d)$ to $(9, c)$, since it has duration 3 and there is another path from $(1, d)$ to $(9, c)$ having duration 1.5, namely, $(3, d, b), (3, b, a), (4.5, a, c)$. This is a fastest path from $(1, d)$ to $(9, c)$ as no faster path exists. Since there is no other path from d to c with lower duration, it also is a fastest path from d to c and $\ell(d, c) = 1.5$.

We denote by $\mathcal{T}_\alpha(u, (t, v))$ the *time to reach* (t, v) from u at time α as follows: $\mathcal{T}_\alpha(u, (t, v)) = \omega - \alpha$, where $\omega \leq t$ is the minimal value, such that there is a path from (α, u) to (ω, v) in S and $[\omega, t] \subseteq T_v$. We call such a path a *foremost path* from (α, u) to (t, v) . For instance, in Fig. 13, the times

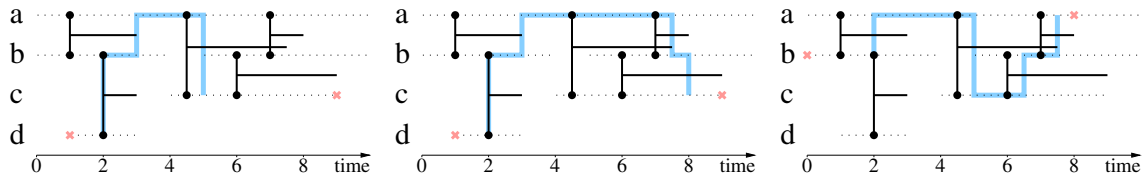


Fig. 13 Paths in a stream graph. Left: a path P_1 from $(1, d)$ to $(9, c)$: $P_1 = (2, d, b), (3, b, a), (5, a, c)$. This path has length 3 and duration 3. Center: another path P_2 from $(1, d)$ to $(9, c)$: $P_2 = (2, d, b), (3, b, a),$

$(7.5, a, b), (8, b, c)$. This path has length 4 and duration 6. Right: a path P_3 from $(0, b)$ to $(8, a)$: $P_3 = (2, b, a), (5, a, c), (6.5, c, b), (7.5, b, a)$. This path has length 4 and duration 5.5

to reach $(5, a), (3, b), (10, b)$, and $(5, c)$ from $(1, d)$ are 1, 1, 5, and 3.5, respectively. The corresponding foremost paths are $F_{5,a} = (2, d, b), (2, b, a), F_{3,b} = (2, d, b)$ in $\{(x, d, b), (y, b, a), (z, a, c), (6, c, b), x \in [2, 3], y \in [x, 3], z \in [4.5, 6]\}$, and $F_{5,c}$ in $\{(x, d, b), (y, b, a), (4.5, a, c), x \in [2, 3], y \in [x, 3]\}$.

In summary, the shortest paths are optimal regarding the number of hops, the fastest paths are optimal regarding the duration between starting and arrival times, and the foremost paths are optimal regarding arrival time. This captures the following intuition: if someone (at a given date) wants to go to another city by train (and arrive before a given date), this person may want to minimize the number of train changes (shortest path), the total time he or she spends traveling (fastest path), or the time at which he or she will arrive at the destination (foremost path).

If there is no path from (α, u) to (ω, v) , then we assert that $\partial((\alpha, u), (\omega, v)), \mathcal{L}((\alpha, u), (\omega, v))$, and $\mathcal{T}_\alpha(u, (\omega, v))$ are infinite. We, respectively, define the *diameter*, the *lapse*, and the *flood time* of S as the largest finite distance, the largest finite latency, and the largest finite time needed to reach an element of W from an element of W .

One may combine the notions above by considering, for instance, *fastest shortest paths* (the ones of minimum duration among those of minimal length) or *shortest fastest paths* (the ones of minimal length among those of minimal duration). For instance, in Fig. 13, the unique fastest shortest path from $(1, d)$ to $(9, c)$ is $(3, d, b), (3, b, a), (4.5, a, c)$. The fastest shortest paths from $(0, a)$ to $(9, c)$ are (x, a, c) for x in $[4.5, 7.5]$. The fastest shortest paths from $(7.6, a)$ to $(9, c)$ are $(x, a, b), (x, b, c)$ for x in $[7.6, 8]$. The fastest shortest paths from $(0, b)$ to $(6, b)$ are $(3, b, a), (x, a, c), (6, c, b)$ for x in $[4.5, 6]$. We discuss shortest fastest paths in more details and consider more complex examples in Sect. 17 for betweenness definitions.

Many extensions of the concept of path in streams make sense and have been considered in the literature (see Sect. 21 for references). We present two of the most common ones below.

First, one may capture the fact that transmission through a link has a cost, leading to the following notion: for a given γ , a γ -path P from $(\alpha, u) \in W$ to $(\omega, v) \in W$ is a

sequence $(t_0, u_0, v_0), (t_1, u_1, v_1), \dots, (t_k, u_k, v_k)$ of elements of $T \times V \times V$, such that $u_0 = u, v_k = v, t_0 \geq \alpha, t_k \leq \omega - \gamma$, for all $i, t_i \geq t_{i-1} + \gamma, u_i = v_{i-1}, [t_i, t_i + \gamma] \times \{u_i v_i\} \subseteq E$, and $[t_i, t_{i+1}] \times \{v_i\} \subseteq W$. The paths discussed, since the beginning of this section is equivalent to γ -paths with $\gamma = 0$, and concepts like reachability, cycles, distances, latencies, and others may easily be extended to this more general case. Notice also that γ may be a function of the links, involved nodes, time, and other complex features, thus capturing the fact that different links may induce different delays, that delay may vary over time, etc.

Another natural generalization consists in capturing the fact that nodes cannot forward information without delay. One then needs to add the constraint $t_{i+1} \geq t_i + \gamma'$ to the previous definition, where γ' captures the delay induced by node forwarding. Similarly, one may want to impose non-null delays on links and/or nodes but without bounds on these delays. The condition above then becomes $t_{i+1} > t_i$.

If $P = (t_0, u_0, v_0), \dots, (t_k, u_k, v_k)$ is a path of length k in S , then $(u_0, v_0), \dots, (u_k, v_k)$ is a path of length k in the induced graph $G(S)$. If it is a cycle in S , it is also a cycle in $G(S)$. However, the converse claims are false: paths in $G(S)$ do not correspond to paths in S , and in particular, a node may be reachable from another node in $G(S)$ but not in S . Notice also that the distance between two nodes in $G(S)$ is bounded by the size of V , whereas it is unbounded in S . For instance, if $T = [0, x]$ for a given integer $x, V = \{a, b\}, T_a = \bigcup_{i=0,1,\dots} [2i, 2i + 1], T_b = \bigcup_{i=0,1,\dots} [2i + 1, 2i + 2]$, and $T_{ab} = \{i, i = 1, \dots\}$, then the path $(0, a, b), (1, b, a), (2, a, b), \dots$ of length x is a shortest path from $(0, a)$ to (x, a) or (x, b) .

In a link stream $L = (T, V, E)$, since nodes are always present, the definition of path is much simpler: a path P from $(\alpha, u) \in T \times V$ to $(\omega, v) \in T \times V$ is a sequence $(t_0, u_0, v_0), (t_1, u_1, v_1), \dots, (t_k, u_k, v_k)$ of elements of $T \times V \times V$, such that $u_0 = u, v_k = v, t_0 \geq \alpha, t_k \leq \omega, t_i \geq t_{i-1}, u_i = v_{i-1}$, and $(t_i, u_i v_i) \in E$. In this case, as with graphs, the distance is bounded by the size of V . In addition, if $(\alpha, u) \dashrightarrow (\omega, v)$, then for all $\alpha' \leq \alpha$ and $\omega' \geq \omega, (\alpha', u) \dashrightarrow (\omega', v)$. However, the existence of a path between two given nodes in $G(L)$ still does not imply in general the existence of a path between them in L . For instance, if $T = [0, 1], V = \{a, b, c, d\}$, and

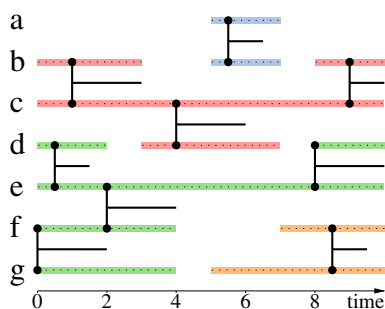


Fig. 14 Weakly connected components of a stream graph. This stream graph has four weakly connected components, each displayed with a different color: $[5, 7] \times \{a, b\}$ in blue, $([0, 3] \cup [8, 10]) \times \{b\} \cup [0, 10] \times \{c\} \cup [3, 7] \times \{d\}$ in pink, $([0, 2] \cup [8, 10]) \times \{d\} \cup [0, 10] \times \{e\} \cup [0, 4] \times \{f, g\}$ in green, and $[7, 10] \times \{f\} \cup [5, 10] \times \{g\}$ in orange

$E = \{(0, ab), (0, cd), (1, bc)\}$, then there is a path between a and d in $G(L)$ but not in L .

In a graph-equivalent stream, there is a path from a node to another one in the stream if and only if there is a path between them in the corresponding graph, and the shortest paths have the same length. As a consequence, the distance between two nodes is the same in the stream and its corresponding graph, and a path is a cycle in the stream if and only if the corresponding path is a cycle in the graph.

15 Connectedness and connected components

A graph $G = (V, E)$ is connected if for all u and v in V , there is a path between u and v in G . A cluster C is connected if $G(C)$ is connected, and it is a maximal connected cluster if it is included in no other connected cluster. These clusters are called the connected components of G , and they form a partition² of V . The reachability graph of G is the graph $R = (V, E')$, where $uv \in E'$ if $u \rightarrow v$ in G . The connected components of G are exactly but the cliques of R .

Given a stream graph $S = (T, V, W, E)$, we say that (ω, v) is weakly reachable from (α, u) , which we denote by $(\alpha, u) \dashrightarrow (\omega, v)$, if there is a sequence $(t_0, u_0, v_0), (t_1, u_1, v_1), \dots, (t_k, u_k, v_k)$ of elements of $T \times V \times V$, such that $u_0 = u, v_k = v$, for all $i, v_i = u_{i+1}$, and $(t_i, u_i v_i) \in E, [\alpha, t_0] \times \{u\} \subseteq W, [t_k, \omega] \times \{v\} \subseteq W$, and for all $i, [t_i, t_{i+1}] \times \{v_i\} \subseteq W$. This sequence is similar to a path from (α, u) to (ω, v) , except for time constraints: we do not necessarily have $t_0 \geq \alpha, t_{i+1} \geq t_i$, nor $\omega \geq t_k$. As a consequence, weak reachability is symmetric: if $(\alpha, u) \dashrightarrow (\omega, v)$ then $(\omega, v) \dashrightarrow (\alpha, u)$. In Fig. 14, for

instance, we have $(9, d) \dashrightarrow (3, g)$ through the sequence $(8, d, e), (3, e, f), (1, f, g)$.

We say that S is weakly connected if for all (α, u) and (ω, v) in $W, (\alpha, u) \dashrightarrow (\omega, v)$. We say that a cluster $C \subseteq W$ is weakly connected if its induced sub-stream $S(C)$ is weakly connected. It is a weakly connected component of S if it is a maximal weakly connected cluster of S . Intuitively, this corresponds to the disconnected parts of a drawing of S , see Fig. 14 for an illustration.

We say that $S = (T, V, W, E)$ is strongly connected if for all (α, u) and (ω, v) in W with $\alpha \leq \omega$, there is a path from (α, u) to (ω, v) in S . We say that a cluster C is strongly connected if $S(C)$ is strongly connected. We say that C is a maximal strongly connected cluster if it is included in no other strongly connected cluster, see Fig. 15 for an illustration. The examples in this figure show that the maximal connected clusters of S do not in general lead to a partition of W .

If S is strongly connected then there is a path between u and v in G_t for all (t, u) and (t, v) in W , i.e., G_t is a connected graph for all t . However, G_t may be connected for all t , even though S is not strongly connected. This happens, for instance, if $T = [0, 3], V = \{a, b\}, W = [0, 1] \times \{a\} \cup [2, 3] \times \{b\}$ and $E = \emptyset$.

If S is compact, though it is strongly connected if and only if G_t is connected for all t in T . Indeed, as already said, if S is strongly connected, then G_t necessarily is connected. Conversely, if G_t is connected for all t in T , then S necessarily is strongly connected: assume that there exist (α, v) and (ω, u) in W with $\omega \geq \alpha$ such that $(\alpha, v) \not\rightarrow (\omega, u)$; since S is compact, $(\alpha, v) \rightarrow (\omega, v)$, and therefore, this implies that $(\omega, v) \rightarrow (\omega, u)$, which contradicts the fact that G_ω is connected.

A cluster C is a maximal strongly connected compact cluster if it is compact, strongly connected, and included in no other strongly connected compact cluster. For instance, the link stream of Fig. 15 (left) has three maximal strongly connected compact clusters, namely $[0, 10] \times \{a, b, c\}, [0, 10] \times \{d, e\}$, and $[5, 10] \times \{a, b, c, d, e\}$. These clusters overlap, and therefore, maximal strongly connected compact clusters do not result in partition of W .

If $C = T_C \times V_C$ is a maximal strongly connected compact cluster, then even though V_C necessarily is a connected cluster of G_t , it is not in general a connected component of G_t . In Fig. 15 (left), for instance, $\{a, b, c\}$ is not a connected component of G_6 (it is included in the connected component $\{a, b, c, d, e\}$ of G_6), although $[0, 10] \times \{a, b, c\}$ is a maximal strongly connected compact cluster.

This leads to the following definition of strongly connected components of S : a strongly connected component C of S is a maximal compact cluster $C = T_C \times V_C$, such that V_C is a connected component of G_t for all t in T_C . This implies that C is a (not necessarily maximal) strongly connected compact cluster. For instance, the maximal strongly

² A partition of a set X into k parts is a family (P_1, P_2, \dots, P_k) of k subsets of X , such that $\cup_i P_i = X$ and $P_i \cap P_j = \emptyset$ for all $i \neq j$.

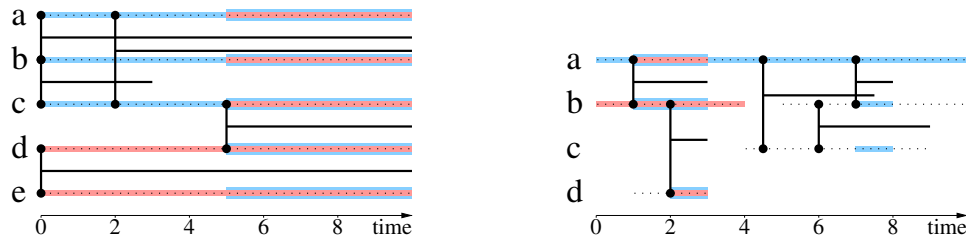


Fig. 15 Strongly connected clusters in a link stream (left) and a stream graph (right). Left: this link stream is not strongly connected, since $(0, a) \not\leftrightarrow (0, d)$, for instance. It has only two maximal strongly connected clusters, namely $[0, 10] \times \{a, b, c\} \cup [5, 10] \times \{d, e\}$ (in blue) and $[0, 10] \times \{d, e\} \cup [5, 10] \times \{a, b, c\}$ (in pink), which overlap. It also contains an infinity of strongly connected clusters which are not maximal and may have an intricate structure, like, for instance $[0, 4] \times \{a, b, c\} \cup [4, 5] \times \{c\} \cup [5, 9] \times \{a, b\} \cup [9, 9.5] \times \{c\} \cup [9, 10] \times \{d\}$. Right: this stream graph is not strongly connected, since $(0, a) \not\leftrightarrow (1, d)$, for instance. The cluster $[2, 3] \times \{a, b, d\}$ is strongly connected but not maximal as it is included in $[2, 3] \times \{a, b, d\} \cup [1, 2] \times \{a, b\}$, which is strongly connected too.

This cluster is not a maximal strongly connected cluster either, as it is included in $[2, 3] \times \{a, b, d\} \cup [1, 2] \times \{a, b\} \cup [0, 1] \times \{a\}$ and $[2, 3] \times \{a, b, d\} \cup [1, 2] \times \{a, b\} \cup [0, 1] \times \{b\}$ which are both strongly connected. Notice, however, that the union of these two clusters is not strongly connected, as $(0, a) \not\leftrightarrow (0, b)$, for instance. They are not maximal either, but they are included (among others), respectively, in $[2, 3] \times \{a, b, d\} \cup [1, 2] \times \{a, b\} \cup [0, 10] \times \{a\} \cup [7, 8] \times \{a, b, c\}$ (in blue) and $[2, 3] \times \{a, b, d\} \cup [1, 2] \times \{a, b\} \cup [0, 4] \times \{b\}$ (in pink) which both are maximal strongly connected clusters of this stream graph

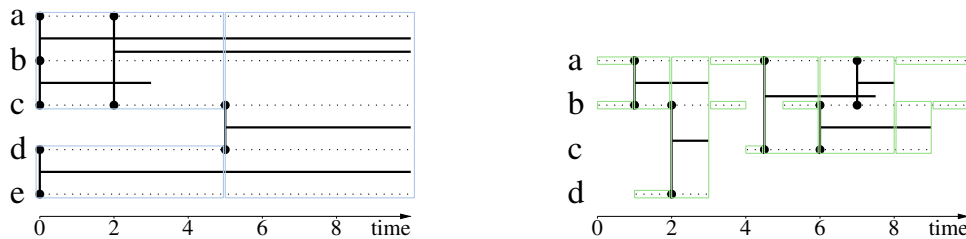


Fig. 16 Connected components in a link stream (left) and a stream graph (right). We indicate each component $C = T_C \times V_C$ with a rectangle. Left: in this link stream, the connected components are $[0, 5] \times \{a, b, c\}$, $[0, 5] \times \{d, e\}$, and $[5, 10] \times \{a, b, c, d, e\}$. Right: in this

stream graph, the connected components are $[0, 1] \times \{a\}$, $[0, 1] \times \{b\}$, $[1, 2] \times \{a, b\}$, $[1, 2] \times \{d\}$, $[2, 3] \times \{a, b, d\}$, $[3, 4] \times \{b\}$, $[3, 4.5] \times \{a\}$, $[4, 4.5] \times \{c\}$, $[4.5, 6] \times \{a, c\}$, $[5, 6] \times \{b\}$, $[6, 8] \times \{a, b, c\}$, $[8, 10] \times \{a\}$, $[8, 9] \times \{b, c\}$, and $[9, 10] \times \{b\}$

connected compact cluster $[0, 10] \times \{a, b, c\}$ of the link stream of Fig. 15 (left) is not a connected component, because $\{a, b, c\}$ is not a connected component of G_t . We display in Fig. 16 the connected components of our two examples.

The set of all strongly connected components of S is a partition of W . Indeed, each (t, v) in W clearly is in a connected component of S . Conversely, if (t, v) is in two distinct connected components $C = T_C \times V_C$ and $D = T_D \times V_D$ of S , then it means that V_C and V_D are two connected components of G_t to which v belongs, which implies that $V_C = V_D$. But then, $(T_C \cup T_D) \times V_C$ also is a strongly connected component, which contradicts the hypothesis.

Notice that the maximal clusters of S , such that for all t , the set of nodes involved in them at time t is a connected component of G_t (but are not necessarily compact) and do not lead to a partition of W . For instance, the two maximal strongly connected clusters of the link stream of Fig. 15 (left) both have these properties, but they overlap.

Given a stream graph $S = (T, V, W, E)$, we define its *reachability stream graph* $R = (T, V, W, E')$, where E' is the set of all (t, uv) in $T \times V \otimes V$, such that $v \dashv u$ in G_t . In other words, there is a link between u and v at time t in R if there is a path in S from u to v at time t . The strongly connected compact clusters of S are exactly the compact cliques of R .

In a link stream $L = (T, V, E)$, the weakly connected components of L are exactly the compact clusters $C = T \times V_C$, such that V_C is a connected component of $G(L)$. However, strong connectivity in link stream has only a few additional properties compared to strong connectivity in stream graphs in general, as illustrated in the figures of this section (the leftmost example is a link stream). Just notice that for all v in V , for all α and ω in T with $\omega \geq \alpha$, $(\alpha, v) \dashv (\omega, v)$ (thanks to an empty path). As a consequence, for all (α, u) and (ω, v) in $T \times V$, if $(\alpha, u) \dashv (\omega, v)$, then for all $\alpha' \leq \alpha$, $(\alpha', u) \dashv (\omega, v)$ and for all $\omega' \geq \omega$, $(\alpha, u) \dashv (\omega', v)$.

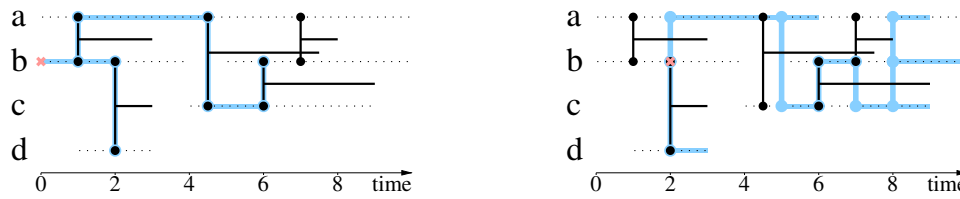


Fig. 17 Examples of trees. We consider a stream graph $S = (T, V, W, E)$ and display two of its sub-streams that are trees. Left: the tree $S_1 = (T, V, W_1, E_1)$ (in blue) of root $(0, b)$ (in pink), with $W_1 = [1, 4.5] \times \{a\} \cup ([0, 2] \cup \{6\}) \times \{b\} \cup [4.5, 6] \times \{c\} \cup \{(2, d)\}$ and $E_1 = \{(1, ab), (2, bd), (4.5, ac), (6, bc)\}$. Right: the tree $S_2 = (T, V,$

$W_2, E_2)$ (in blue) of root $(2, b)$ (in pink), with $W_2 = ([2, 6] \cup [8, 9]) \times W_2 = ([2, 6] \cup [8, 9]) \times \{a\} \cup ([6, 7] \cup [8, 10]) \times \{b\} \cup ([5, 6] \cup [7, 9]) \times \{c\} \cup [2, 3] \times \{d\}$ and $E_2 = \{(2, ab), (2, bd), (5, ac), (6, bc), (7, bc), (8, bc), (8, ab)\}$

In a graph-equivalent stream, the strongly connected components are equivalent to the connected component of the corresponding graph.

16 Trees and cascades

A graph $G = (V, E)$ is a tree of root r , with $r \in V$, if for all v in V , there is a unique simple path from r to v in G . Then, G is connected and acyclic, and any connected acyclic graph (with a distinguished node r) is a tree (of root r). This also implies that for all $v \neq r$ in V , there is a unique $u \neq v$, such that u is the last node before v on the simple path from r to v , called the predecessor of v and denoted by $p(v)$. In addition, the predecessor of the root is the root itself: $p(r) = r$.

Given a graph $G = (V, E)$, a sub-graph $G' = (V', E')$ of G is a shortest path tree of root r if it is a tree of root r and for all v in V' , the simple path from r to v in G' is a shortest path from r to v in G . A cascade is a maximal shortest path tree.

We say that a stream graph $S = (T, V, W, E)$ is a tree of root r , with $r \in W$, if for all (t, v) in W , there is a unique simple path from r to (t, v) in S . Then, S necessarily is weakly connected and acyclic, but the converse is not true. Notice also that a tree is not strongly connected in general, see Fig. 17 for an illustration.

If $S = (T, V, W, E)$ is a tree of root r , then for all $(t, v) \neq r$ in W either there is a unique last (t', u) with $u \neq v$ before (t, v) involved in the simple path from r to (t, v) , and we call it the predecessor of v at time t , or the simple path from r to (t, v) is the empty sequence, and we say that the predecessor of v at time t is r . We denote the predecessor by $p(t, v)$. In Fig. 17, for instance, $p(5, c) = (4.5, a)$, $p(6, b) = (6, c)$, and $p(1, b) = (0, b) = r$ in S_1 and $p(9, c) = (7, b)$ in S_2 .

If S is a tree of root r and α is the first time at which any node is involved in S , i.e., $\alpha = \min\{t, (t, v) \in W\}$, then necessarily r is in $(\{\alpha\} \times V) \cap W$. In other words, the root of S necessarily is one of the very first node occurrences in S . Moreover, it also is a tree of root r' for all r' in $(\{\alpha\} \times V) \cap W$. In Fig. 17 (right), for instance, S_2 also is a tree of root $(2, a)$ and a tree of root $(2, d)$.

Given a stream graph $S = (T, V, W, E)$, we say that a sub-stream $S' = (T, V, W', E')$ of S is a shortest path tree of root r if it is a tree of root r and for all (t, v) in W' , the simple path from r to (t, v) in S' is a shortest path from r to (t, v) in S . We define similarly fastest path trees and foremost-path trees. In Fig. 17, for instance, S_1 is a foremost-path tree of S .

For a given r in W , we denote by $R(r)$ the cluster of all elements of W reachable from r , and we call it the reachable cluster of r . We say that the sub-stream S' is a cascade of root r if it is a maximal foremost-path tree, in the sense that it is included in no other foremost-path tree with the same root.

If S is a graph-equivalent stream and $S' \subseteq S$ is a tree of root $r = (t, v)$, then its induced graph $G(S')$ is a tree of root v . If S' is a shortest path tree of S , then $G(S')$ is a shortest path tree of $G(S)$. The same holds for cascades.

17 Closeness and betweenness centralities

In a graph $G = (V, E)$, the closeness of a node v measures its proximity to other nodes: $C(v) = \sum_{u \neq v} \frac{1}{d(v, u)}$. The between-

ness of v measures how frequently v is involved in shortest paths in G : $B(v) = \sum_{u \in V, w \in V} \frac{\sigma(u, w, v)}{\sigma(u, w)}$, where $\frac{\sigma(u, w, v)}{\sigma(u, w)}$ is the frac-

tion of all the shortest paths from u to w that involve v if there is a path from u to w , 0 otherwise. In other words, the betweenness of v in V is the number of pairs of elements u and w of V , each counted with a weight equal to the fraction of shortest path between them that involve v . The betweenness of any cluster $X \subseteq V$ is $B(X) = \sum_{u \in V, w \in V} \frac{\sigma(u, w, X)}{\sigma(u, w)}$, where $\frac{\sigma(u, w, X)}{\sigma(u, w)}$ is the fraction of all the shortest paths from u to w that involve an element of X , if $u - w$, 0 otherwise. Notice that $B(v) = B(\{v\})$.

In the stream graph $S = (T, V, W, E)$, we define a general concept of closeness of a node v at a time instant t , with $(t, v) \in W$, as follows:

$$C_t(v) = \sum_{u \in V} \int_{(s,u) \neq (t,v)}^s \frac{1}{c_t(v, (s,u))} ds,$$

where $c_t(v, (s, u))$ represents the cost to reach (s, u) from v at time t .

The cost $c_t(v, (s, u))$ may be captured in various ways, the most basic being the time to reach (s, u) from v at time t : $c_t(v, (s, u)) = \mathcal{T}_t(v, (s, u))$. Notice, however, that we must have $c_t(v, (s, u)) \neq 0$ for all $(s, u) \neq (t, v)$. To ensure this, one may, for instance, define $c_t(v, (s, u))$ as the length of a non-empty shortest foremost path from (t, v) to (s, u) , i.e., $\partial((t, v), (t + \mathcal{T}_t(v, (s, u)), u))$ if it is different from 0. One may also combine both approaches by assuming that traversing a link has a cost γ , leading to the following cost function: $c_t(v, (s, u)) = \mathcal{T}_t(v, (s, u)) + \gamma \cdot \partial((t, v), (t + \mathcal{T}_t(v, (s, u)), u))$.

We now define the *betweenness* of a node $v \in V$ at a time instant $t \in T$, with $(t, v) \in W$, as follows:

$$B(t, v) = \sum_{u \in V, w \in V} \int_{i \in T_u, j \in T_w} \frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))} di dj,$$

where $\frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))}$ is the fraction of all shortest fastest paths from u at time i to w at time j that involve v at time t if there is a path from (i, u) to (j, w) , 0 otherwise. In other words, the betweenness of (t, v) in W is the number of pairs of elements (i, u) and (j, w) of W , each counted with a weight equal to the fraction of shortest fastest paths between them that involve (t, v) .

We extend the definition to any cluster $X \subseteq W$ as follows:

$$B(X) = \sum_{u \in V, w \in V} \int_{i \in T_u, j \in T_w} \frac{\sigma((i, u), (j, w), X)}{\sigma((i, u), (j, w))} di dj,$$

where $\frac{\sigma((i, u), (j, w), X)}{\sigma((i, u), (j, w))}$ is the fraction of all shortest fastest paths from (i, u) to (j, w) that involve at least an element of X if $(i, u) \rightarrow (j, w)$, 0 otherwise. Then, $B(t, v) = B(\{(t, v)\})$. We also use this approach to define the betweenness of node v as $B(v) = B(T_v \times \{v\})$, and the one of time t as $B(t) = B(\{t\} \times V_t)$.

Instead of shortest fastest paths, one may consider the fraction of fastest shortest paths, of shortest paths, of fastest simple paths, or other classes of paths. However, considering shortest fastest paths has the advantage of putting more emphasis on time than distance, and to avoid considering as equivalent fastest paths with very different lengths (in particular the non-simple ones).

In a graph-equivalent stream, the betweenness of any node v is equal to $\frac{|T|^2}{2}$ times its betweenness in the corresponding graph. Indeed, for any (i, u) and (j, w) with $j \geq i$, the fraction of paths involving $T \times \{v\}$ in a graph-equivalent

stream is the fraction of paths between u and w in the corresponding graph that involve v .

The rest of this section is devoted to detailed examples of betweenness centralities in various link streams, representative of what happens in stream graphs in general, to illustrate this concept in concrete situations.

Let us consider, for instance, the case of L_1 defined in Fig. 18 (left), and let us compute the betweenness $B(t, v)$ of (t, v) for all t . To do so, we consider successively all possible pairs of nodes.

Let us begin with u and w . There is a path from (i, u) to (j, w) only for i in $[0, 2]$ and j in $[2, 4]$. Then, there is a unique shortest fastest path, and it is $(2, u, v), (2, v, w)$. It involves v at time 2 and only at this time. Therefore, for all $i \in [0, 2]$ and $j \in [2, 4]$, the value of $\frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))}$ is 1 if $t = 2$, and 0 otherwise. These values are the same for paths from w to u .

For all times i and j , all shortest fastest paths from (i, u) to (j, w) , if any, are of the form (k, u, v) for $k \in [\max(1, i), \min(2, j)]$. For $i < j$, there is an infinity of such paths and at most one involves (t, v) , leading to $\frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))} = 0$. If $i = j$, then there is a unique shortest fastest

path, and it involves (t, v) only when $i = j = t$. Therefore, $\frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))}$ is different from 0 only for $i = j = t$, while

$i \in [0, 2]$ and $j \in [\max(1, i), 4]$, and therefore, the contribution to $B(t, v)$ of paths from u to v is 0. The same reasoning holds for paths from v to u , from w to v , and from v to w .

Finally, shortest fastest paths from v to v are empty sequences, and therefore, they do not involve (t, v) for any t .

This leads to $B(2, v) = 2 \cdot \int_0^2 \int_2^4 1 dj di = 8$ and for all $t \neq 2$, $B(t, v) = 0$.

Let us now consider the case of L_2 , defined in Fig. 18 (center), and let us first focus on the paths from u to w . For any i , if $j < 3$, then $(i, u) \not\rightarrow (j, w)$. For $i \in [0, 2]$ and $j \in [3, 10]$, the unique shortest fastest path from (i, u) to (j, w) is $(2, u, v), (3, v, w)$, in blue in the figure. For $i \in [2, 10]$ and $j \in [0, 8]$, $(i, u) \not\rightarrow (j, w)$. For $i \in [2, 6]$ and $j \in [8, 10]$, the unique shortest fastest path from (i, u) to (j, w) is $(6, u, v), (8, v, w)$, in green in the figure. Finally, for $i \in [6, 10]$ and any j , $(i, u) \not\rightarrow (j, w)$. Therefore, $\frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))}$ is different

from 0 only when $t \in [2, 3]$, $i \in [0, 2]$, $j \in [3, 10]$ and when $t \in [6, 8]$, $i \in [2, 6]$, $j \in [8, 10]$. It is then equal to 1.

Regarding paths from (i, w) to (j, u) , the unique shortest fastest path is $(4, w, v), (5, v, u)$ and it exists for $i \in [0, 4]$ and $j \in [5, 10]$. It involves (t, v) when $t \in [4, 5]$, leading to $\frac{\sigma((i, w), (j, u), (t, v))}{\sigma((i, w), (j, u))} = 1$ for $t \in [4, 5]$, $i \in [0, 4]$, $j \in [5, 10]$, and 0 otherwise.

Like for L_1 , the contribution of other pairs of nodes to $B(t, v)$ is 0, and therefore, we finally obtain $B(t, v) = \int_0^2 \int_3^{10} 1 dj di = 14$ for $t \in [2, 3]$, $B(t, v) = \int_2^6 \int_8^{10} 1 dj di = 8$ for $t \in [6, 8]$,

$B(t, v) = \int_0^4 \int_5^{10} 1dj di = 20$ for $t \in [4, 5]$, and $B(t, v) = 0$ otherwise.

In the case of L_3 defined in Fig. 18 (right), first notice that all shortest fastest paths from (i, u) to (j, w) and from (i, w) to (j, u) , if any, are of the form $(k, u, v), (k, v, w)$ or $(k, w, v), (k, v, u)$, respectively, with $k \in [2, 3], k \geq i$ and $k \leq j$. Therefore, $B(t, v) = 0$ if $t \notin [2, 3]$. Moreover, $\frac{\sigma((i,u),(j,w),(t,v))}{\sigma((i,u),(j,w))} = \frac{\sigma((i,w),(j,u),(t,v))}{\sigma((i,w),(j,u))}$.

If $t \in [2, 3]$, in the same way as for paths from u to v in L_1 , we are in one of two cases: either there is an infinity of shortest fastest paths from (i, u) to (j, w) and at most one of them involves (t, v) , or the fraction of values of i and j , such that $\frac{\sigma((i,u),(j,w),(t,v))}{\sigma((i,u),(j,w))} \neq 0$ is 0.

Since, like in the previous cases, the contribution of other pairs of nodes is 0, and therefore, we obtain $B(t, v) = 0$ in L_3 for all (t, v) .

However, let us consider the cluster $X = [2, 3] \times \{v\}$. Then, $\frac{\sigma((i,u),(j,w),X)}{\sigma((i,u),(j,w))} = \frac{\sigma((i,w),(j,u),X)}{\sigma((i,w),(j,u))}$ is equal to 1 for $i \in [0, 2]$ and $j \in [2, 5]$, for $i \in [2, 3]$ and $j \in [i, 5]$, and it is equal to 0 in all other cases. Moreover, $\frac{\sigma((i,u),(j,w),X)}{\sigma((i,u),(j,w))} = \frac{\sigma((i,w),(j,u),X)}{\sigma((i,w),(j,u))}$ is equal to 1 for $i \in [2, 3]$ and $j \in [i, 5]$, to 0.5 for $i \in [0, 1]$ and $j \in [3, 5]$, to $\frac{j-2}{j-1}$ for $i \in [0, 1]$ and $j \in [2, 3]$, to $\frac{j-2}{j-1}$ for $i \in [1, 2]$ and $j \in [2, 3]$, to $\frac{1}{3-i}$ for $i \in [1, 2]$ and $j \in [3, 5]$, and it is equal to 0 in all other cases. Likewise, $\frac{\sigma((i,v),(j,w),X)}{\sigma((i,v),(j,w))} = \frac{\sigma((i,v),(j,w),X)}{\sigma((i,v),(j,w))}$ is equal to 1 for $j \in [2, 3]$ and $i \in [0, j]$, to 0.5 for $i \in [0, 2]$ and $j \in [4, 5]$, to $\frac{1}{j-2}$ for $i \in [0, 2]$ and $j \in [3, 4]$, to $\frac{3-i}{j-i}$ for $i \in [2, 3]$ and $j \in [3, 4]$, to $\frac{3-i}{4-i}$ for $i \in [2, 3]$ and $j \in [4, 5]$, and it is equal to 0 in all other cases.

We, therefore, obtain

$$\begin{aligned}
 B(X) &= 2 \cdot \left(\int_0^2 \int_2^5 1dj di + \int_2^3 \int_i^5 1dj di \right) \\
 &+ 2 \cdot \left(\int_2^3 \int_i^5 1dj di + \int_0^1 \int_3^5 0.5dj di + \int_0^1 \int_2^3 \frac{j-2}{j-1} dj di \right) \\
 &+ \int_1^2 \int_2^3 \frac{j-2}{j-i} dj di + \int_1^2 \int_3^5 \frac{1}{3-i} dj di \\
 &+ 2 \cdot \left(\int_2^3 \int_0^j 1di dj + \int_0^2 \int_4^5 0.5dj di \right) \\
 &+ \int_0^2 \int_3^4 \frac{1}{j-2} dj di + \int_2^3 \int_3^4 \frac{3-i}{j-i} dj di \\
 &+ \int_2^3 \int_4^5 \frac{3-i}{4-i} dj di \Big) = 17 + (2 \ln(2) + 10) + (2 \ln(2) + 10) \sim 39.77
 \end{aligned}$$

Now, let us consider L_4 defined in Fig. 19 (left). We compute the contribution of u and w to the betweenness of $(3.5, v)$, displayed in red in the figure, i.e., $\frac{\sigma((i,u),(j,w),(3.5,v))}{\sigma((i,u),(j,w))}$ for

all i and j . There is a shortest fastest path from (i, u) to (j, w) only for $i \in [0, 1]$ and $j \in [6, 8]$, and it is always of the form $(1, u, x), (k, x, v), (l, v, y), (6, y, w)$ with $k \in [2, 4], l \in [3, 5]$, and $l \geq k$. Among them, the ones involving $(3.5, v)$ are exactly those such that $k \in [2, 3.5]$ and $l \in [3.5, 5]$. This leads to the fraction $\frac{|[2,3.5] \times [3.5,5]|}{|[2,3] \times [3,5]| + \frac{1}{2} |[3,4]|^2 + |[3,4] \times [4,5]|} \sim 0.64$.

Let us finally consider L_5 defined in Fig. 19 (right). We compute the contribution of u and w to the betweenness of (t, v) , for a t in $[b, c]$, like the one displayed in red in the figure. There are two families of shortest fastest paths from u to w in this link stream: $(2, u, x), (k, x, v), (l, v, y), (7, y, w)$ with $k \in [a, b]$ and $l \in [c, d]$, that we call the *blue* family; and $(10, u, x), (m, x, v), (n, v, y), (15, y, w)$ with $m \in [e, f]$ and $n \in [g, h]$, that we call the *green* family. Notice that (t, v) , for a t in $[b, c]$, is involved in all blue paths and in no green path. For $i \in [0, 2]$ and $j \in [7, 15[$ the shortest fastest paths from (i, u) to (j, w) are the blue ones (they all involve (t, v)); for $i \in [0, 2]$ and $j \in [15, 17]$ they are both the blue and green ones (a fraction $\frac{(b-a) \cdot (d-c)}{(b-a) \cdot (d-c) + (f-e) \cdot (h-g)}$ of them involve (t, v)); for $i \in [2, 10]$ and $j \in [15, 17]$, they are the green ones (none of them involve (t, v)); and for all other values i and j , there is no path from (i, u) to (j, w) . This leads to $\int_0^2 \int_7^{15} 1dj di + \int_0^2 \int_{15}^{17} \frac{(b-a) \cdot (d-c)}{(b-a) \cdot (d-c) + (f-e) \cdot (h-g)} dj di = 16 + 4 \cdot \frac{(b-a) \cdot (d-c)}{(b-a) \cdot (d-c) + (f-e) \cdot (h-g)}$.

In the computations above, we, however, assumed that $a \neq b, c \neq d, e \neq f$, and $g \neq h$ when we wrote that the

fraction of blue paths in the set of all green and blue paths is $\frac{(b-a) \cdot (d-c)}{(b-a) \cdot (d-c) + (f-e) \cdot (h-g)}$. If $a = b$ or $c = d$, but $e \neq f$ and $g \neq h$

this still holds, as there are infinitely less blue paths than green ones; the fraction of blue paths is 0. If $a = b$ and $e = f$, but $c \neq d$ and $g \neq h$, however, the fraction above is undefined and the fraction of blue paths becomes $\frac{(d-c)}{(d-c) + (h-g)}$.

Going further, if $a = b$, $c = d$, $e = f$, and $g = h$, then there is exactly one blue path and one green path, leading to a fraction of blue paths of $\frac{1}{2}$.

18 Discrete versus continuous time

All our examples and illustrations until now assumed that the set of time instants used in the definitions of stream graphs is an interval $[\alpha, \omega]$ of \mathbb{R} , thus bounded, infinite and continuous. When we defined stream graphs in Sect. 3, we, however, claimed that our formalism is much more general, and may be used with different kinds of time modeling: bounded or unbounded, finite or infinite, continuous or discrete, and all combinations.

Even with these various types of time sets, the formalism we developed in this paper applies directly (one just has to switch from integrals to sums in the case of discrete time). We illustrate this in this section by considering the situation, where the set of time instants is an interval of \mathbb{N} instead of \mathbb{R} , thus bounded, finite and discrete, see Fig. 20 for an illustration. We discuss more complex cases by the end of this section.

Let us consider $S = (T, V, W, E)$ with $T = [\alpha, \omega] \subseteq \mathbb{N}$. The definitions of $\text{cov}(S) = \frac{|W|}{|T \times V|}$, $n_v = \frac{|T_v|}{|T|}$, $n = \frac{|W|}{|T|}$, $m_{uv} = \frac{|T_{uv}|}{|T|}$, $m = \frac{|E|}{|T|}$, $k_t = \frac{|V_t|}{|V|}$, $k = \frac{|W_t|}{|V|}$, $l_t = \frac{|E_t|}{|V \otimes V|}$, $l = \frac{|E|}{|V \otimes V|}$, and $\Psi(S) = \frac{\sum_{m \in V \otimes V} |T_u \cap T_v|}{\sum_{m \in V \otimes V} |T_u \cup T_v|}$ are directly applicable. For the example in Fig. 20, we obtain, for instance, $\text{cov}(S) = \frac{37}{14 \cdot 4} \sim 0.66$, $n_a = 1$, $n_d = \frac{3}{14} \sim 0.21$, $n = \frac{37}{14} \sim 2.6$, $k_0 = 0.25$, $k_1 = 0.75$ and $l_{10} = \frac{2}{6} \sim 0.33$.

The definition of density also holds (the node-based definition is identical and in the time-based definition the integral just needs to be replaced by a sum): $\delta(S) = \frac{\sum_{m \in V \otimes V} |T_m|}{\sum_{m \in V \otimes V} |T_u \cap T_v|} = \frac{\sum_{t \in T} |E_t|}{\sum_{t \in T} |V_t \otimes V_t|}$. In our example in Fig. 20, $\delta(S) = \frac{0+1+2+2+1+0+1+1+2+3+2+1+1+0}{0+3+3+3+1+3+1+3+3+3+3+3+1} = 0.5$.

Going further, the definitions of sub-streams and clusters also apply. For instance, $C = \{(1, a), (1, b), (2, a), (2, b), (2, d)\}$ is a cluster of S defined in Fig. 20, and it induces the sub-stream $S' = (T, V, C, E')$ with $E' = \{(1, ab), (2, ab), (2, bd)\}$.

As a consequence, the concepts of cliques, neighborhoods, degrees, and clustering coefficients, which depend only on the concepts above (cluster, density, and number of nodes), are also directly applicable to this case. For instance,

$\{9\} \times \{a, b, c\}$ is a maximal compact clique of S defined in Fig. 20, as well as $\{1, 2, 3, 4\} \times \{a, b\}$, the neighborhood of d is $\{(2, b), (3, b)\}$, and therefore, d has degree $\frac{2}{14} \sim 0.14$.

The quotient stream and the line stream of a discrete stream graph are also discrete stream graphs, with unchanged definitions. Likewise, the definition of k -cores is unchanged.

Paths in S are now discrete objects, but this does not call for new definitions. For instance, in Fig. 20, $(7, a, c), (8, c, b)$ is a path from $(0, a)$ to $(9, b)$. It involves exactly $(7, a), (7, c), (8, c)$, and $(8, b)$. It has length 2 and duration 1. It is not a shortest path from $(0, a)$ to $(9, b)$, since path $(9, a, b)$ is shorter. It is not a fastest path either, because path $(9, a, b)$ is faster. However, the path $(8, a, c), (8, c, b)$ has length 2 and duration 0; therefore, it is a fastest path from $(0, a)$ to $(9, b)$ but not a shortest one.

Likewise, the definitions of connected clusters and components, as well as those of trees and cascades, that rely only on the concept of paths, directly translate to the discrete case. The concepts of closeness and betweenness also do, but the betweenness relies now on a counting of discrete sets of (discrete) paths. Replacing the integral in its definition by a sum, it becomes

$$B(t, v) = \sum_{(i,u) \in W, (j,w) \in W} \frac{\sigma((i, u), (j, w), (t, v))}{\sigma((i, u), (j, w))}$$

where $\sigma((i, u), (j, w))$ is now the (finite) number of shortest fastest paths from (i, u) to (j, w) , and $\sigma((i, u), (j, w), (t, v))$ is the number of these path that involve v at time t . Therefore, as before, $\frac{\sigma((i,u),(j,w),(t,v))}{\sigma((i,u),(j,w))}$ is the fraction of all shortest fastest paths from (i, u) to (j, w) that involve (t, v) .

In the case of Fig. 20, for instance, $\sigma((0, a), (3, d)) = |\{(2, a, b), (2, b, d), ((3, a, b), (3, b, d))\}| = 2$ paths. One of them involves $(2, b)$, and so $\frac{\sigma((0,a),(3,d),(2,b))}{\sigma((0,a),(3,d))} = 0.5$.

Finally, we have shown that considering a bounded discrete time interval does not call for new definitions and any change in our formalism: it directly applies to this case in a way very similar to the bounded continuous time case. This also holds for more complex time sets, including unbounded ones (then, a node v has to be present a finite fraction of this infinite time set to satisfy $n_v \neq 0$) and/or discontinuous ones (T may, for instance, be a collection of intervals of \mathbb{R} or \mathbb{N} , or even parts of \mathbb{Q}). As such cases have limited practical and theoretical interest, we do not give more details here.

19 Δ -analysis and instantaneous links

In some situations, directly studying the stream graph induced by a dataset makes little sense. If one considers phone calls or sexual contacts, for instance, nodes generally

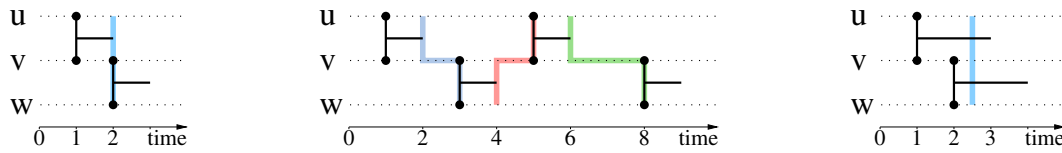


Fig. 18 Basic examples for betweenness centrality computations in link streams. Left: $L_1 = (T, V, E)$ with $T = [0, 4]$, $V = \{u, v, w\}$, and $E = [1, 2] \times \{uv\} \cup [2, 3] \times \{vw\}$. We display in blue the unique shortest fastest path from u to w . Center: $L_2 = (T, V, E)$ with $T = [0, 10]$, $V = \{u, v, w\}$, and $E = ([1, 2] \cup [5, 6]) \times \{uv\} \cup ([3, 4] \cup [8, 9]) \times \{vw\}$. We display in

blue and in green the two shortest fastest paths from u to w (for different starting and arrival times), and in red the unique shortest fastest path from w to u . Right: $L_3 = (T, V, E)$ with $T = [0, 6]$, $V = \{u, v, w\}$, and $E = [1, 3] \times \{uv\} \cup [2, 4] \times \{vw\}$. We display in blue an instance of shortest fastest path between u and w

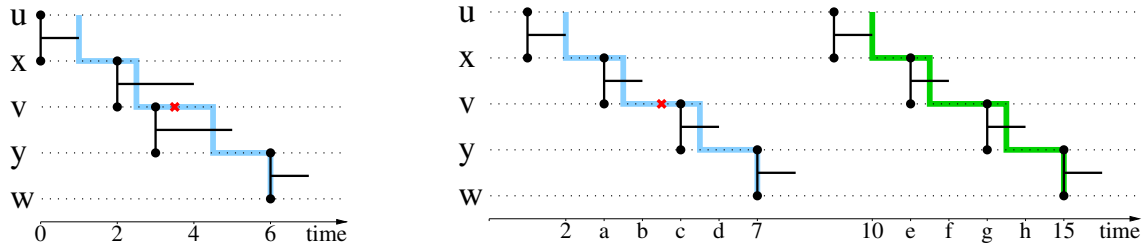


Fig. 19 More examples for betweenness centrality computations in link streams. Left: $L_4 = (T, V, E)$ with $T = [0, 8]$, $V = \{u, x, v, y, w\}$, and $E = [0, 1] \times \{ux\} \cup [2, 4] \times \{xv\} \cup [3, 5] \times \{vy\} \cup [6, 7] \times \{yw\}$. We display in blue an instance of shortest fastest path from u to w and in red the element $(3.5, v)$. Right: $L_5 = (T, V, E)$ with $T = [0, 17]$, $V = \{u, x, v, y, w\}$, and $E = ([1, 2] \cup [9, 10]) \times \{ux\} \cup ([a, b] \cup [e, f]) \times \{xv\} \cup ([c, d] \cup [g, h]) \times \{vy\} \cup ([7, 8] \cup [15, 16]) \times \{yw\}$, for given

values of $a, b, c, d, e, f, g,$ and h , such that $2 < a \leq b < c \leq d < 7$ and $10 < e \leq f < g \leq h < 15$. The shortest fastest paths from u to w all belong to two families: $(2, u, x), (k, x, v), (l, v, y), (7, y, w)$ with $k \in [a, b]$ and $l \in [c, d]$ (an instance is displayed in blue); and $(10, u, x), (m, x, v), (n, v, y), (15, y, w)$ with $m \in [e, f]$ and $n \in [g, h]$ (in green). We display in red an element (t, v) with $t \in [b, c]$

have only zero or one link at a time, leading to an instantaneous degree of 0 or 1. In the case of instant messaging or sensor-based measurements of proximity between individuals, links are instantaneous, leading to a density equal to 0.

In such cases and in many others, one is generally interested in the fact that nodes interact regularly, typically at least once every Δ units of time, for a given Δ . For instance, two individuals call each other at least once a day, two sensors detect each other at least once every ten seconds, etc.

Then, the usual approach consists in using this value of Δ as a parameter to define notions to describe the data, an approach that we call Δ analysis. For instance, one defines n_Δ and m_Δ as the expected number of nodes and links, respectively, present in a randomly chosen time interval of duration Δ in T . One defines the Δ -degree $d_\Delta(v)$ of a node v as its expected number of neighbors during a randomly chosen time interval of duration Δ in T . The Δ -density is defined as follows³. Assume that one takes a random time interval of duration Δ and two nodes involved in S at some time during this interval, i.e., a random triplet (I, u, v) with $I = [t - \frac{\Delta}{2}, t + \frac{\Delta}{2}] \subseteq T$, $u,$ and v in V , such that $T_u \cap I$ and

$T_v \cap I$ are non-empty. The Δ -density of S is the probability that these two nodes are linked together during this interval, i.e., that $T_{uv} \cap I$ is non-empty.

The formalism we developed in this paper provides a more general way to deal with such cases that we now present.

Given a stream graph $S = (T, V, W, E)$ with $T = [\alpha, \omega]$ and a value $\Delta \leq \omega - \alpha$, we define $S_\Delta = (T_\Delta, V, W_\Delta, E_\Delta)$ as the stream graph, such that $T_\Delta = [\alpha + \frac{\Delta}{2}, \omega - \frac{\Delta}{2}]$, $W_\Delta = (T_\Delta \times V) \cap \bigcup_{(t,v) \in W} [t - \frac{\Delta}{2}, t + \frac{\Delta}{2}] \times \{v\} = \{(t', v), t' \in T_\Delta, \exists (t, v) \in W \text{ s.t. } |t' - t| \leq \frac{\Delta}{2}\}$ and $E_\Delta = (T_\Delta \times V \otimes V) \cap \bigcup_{(t,uv) \in E} [t - \frac{\Delta}{2}, t + \frac{\Delta}{2}] \times \{uv\} = \{(t', uv), t' \in T_\Delta, \exists (t, uv) \in E \text{ s.t. } |t' - t| \leq \frac{\Delta}{2}\}$, see Fig. 21 for an illustration.

In other words, a node is present at time t' in S_Δ whenever it is present in S at a time t in $[t' - \frac{\Delta}{2}, t' + \frac{\Delta}{2}]$, i.e., $T_{\Delta v} = T_\Delta \cap \{t', \exists t \in T_v, |t' - t| \leq \frac{\Delta}{2}\}$. Likewise, any two nodes are linked together at time t' in S_Δ whenever they are linked together in S at a time t in $[t' - \frac{\Delta}{2}, t' + \frac{\Delta}{2}]$, i.e., $T_{\Delta uv} = T_\Delta \cap \{t', \exists t \in T_{uv}, |t' - t| \leq \frac{\Delta}{2}\}$.

We now show that the properties of S_Δ actually are equivalent to the Δ -properties of S , and therefore, one may conduct Δ analysis of S by transforming it into S_Δ first, and then using the formalism of this paper.

³ This is a generalization to stream graphs of the Δ -density introduced in Viard and Latapy (2014) for link streams.

Let us define instantaneous versions of the Δ -properties of S cited above: for all t in $[\alpha + \frac{\Delta}{2}, \omega - \frac{\Delta}{2}]$, $n_{\Delta t}$ is the number of distinct nodes present at some time in $[t - \frac{\Delta}{2}, t + \frac{\Delta}{2}]$: $n_{\Delta t} = |\{v \in V, \exists t', |t' - t| \leq \frac{\Delta}{2} \text{ and } (t', v) \in W\}|$. We define $m_{\Delta t}$ similarly, and $d_{\Delta t}(v)$ as the number of distinct nodes linked to v at some time in $[t - \frac{\Delta}{2}, t + \frac{\Delta}{2}]$.

First notice that $n_{\Delta t}$ in S is equal to $|V_t|$ in S_{Δ} . Indeed, $|V_t|$ in S_{Δ} is equal to $|\{v \in V, \exists t' \in T, (t', v) \in W \text{ and } |t' - t| \leq \frac{\Delta}{2}\}| = n_{\Delta t}$. Likewise, $m_{\Delta t}$ in S equals $|E_t|$ in S_{Δ} , and $d_{\Delta t}(v)$ in S equals $d_t(v)$ in S_{Δ} .

Notice now that $n_{\Delta} = \frac{1}{|T_{\Delta}|} \cdot \int_{t \in T_{\Delta}} n_{\Delta t} dt$ in S . Therefore, it is equal to $\frac{1}{|T_{\Delta}|} \cdot \int_{t \in T_{\Delta}} |V_t| dt$ in S_{Δ} , which is exactly n in S_{Δ} .

Similar reasoning lead to the facts that m_{Δ} in S is equal to m in S_{Δ} , and that $d_{\Delta}(v)$ in S is equal to $d(v)$ in S_{Δ} for all v .

Going further, we have $\delta_{\Delta}(S) = \delta(S_{\Delta})$. Indeed, $\delta(S_{\Delta})$ is the probability that a random (t, u, v) with $t \in T_{\Delta}$, $(t, u) \in W_{\Delta}$ and $(t, v) \in W_{\Delta}$ satisfies $(t, uv) \in E_{\Delta}$, and $\delta_{\Delta}(S)$ is the probability that a random (I, u, v) with I an interval of T of duration Δ , $u \in V$ and $v \in V$, such that $T_u \cap I \neq \emptyset$ and $T_v \cap I \neq \emptyset$ satisfies $T_{uv} \cap I \neq \emptyset$. A triplet (t, u, v) satisfies the first set of constraints if and only if the triplet (I, u, v) with $I = [t - \frac{\Delta}{2}, t + \frac{\Delta}{2}]$ fits the second set of constraints. In addition, it satisfies $(t, uv) \in E_{\Delta}$ if and only if (I, u, v) satisfies $T_{uv} \cap I \neq \emptyset$. Therefore, the two probabilities are equal.

Finally, our approach makes it easy to conduct the Δ analysis of a stream graph S : it is equivalent to analyzing the stream graph S_{Δ} with the general methods developed here, which go much further than the previously considered Δ properties.

This approach has another strength: one may use variable values of Δ , which may be a function of time, depend on the involved nodes or links, or any other property. One may, for instance, consider that two colleagues are in contact whenever they meet each other at least once a week, but for holidays, they remain in contact if they meet in the week before holidays and in the week after. It is easy to capture such modeling choice within our framework: they only change the way one builds S_{Δ} from S , and the analysis of S_{Δ} remains unchanged. Defining properties that would directly take into account such variations of Δ would be much more complex.

20 Bipartite streams and other generalizations

An important strength of the graph formalism is that it may easily be extended to encompass richer, more complex cases. For instance, one may consider directed links by defining directed graphs $G = (V, E)$ with $E \subseteq V \times V$ instead of $E \subseteq V \otimes V$. One may allow loops ($vv \in E$ is possible), and/or consider multigraphs (E is a multiset; thus, several links

between the two same nodes are possible). Going further, one may capture link strength or cost using weighted graphs, in which a weight is associated to each element of E and/or V . One may combine these extensions by considering, for instance, directed weighted multigraphs.

Dealing with such graph generalizations calls for an update of classical graph concepts. For instance, the density of directed graphs must take into account the fact that the number of possible links changed; it also leads to notions of in- and out-degrees (the number of links towards and from a given node); etc. Some properties are non-trivial to extend (like, for instance, the density for weighted graphs) but most just need to be patched, thus giving a great expressivity and wide areas of applications to graphs.

Bipartite graphs are a particularly pervasive graph extension, and this section details this case as an illustration: we show how a few key extensions of graph concepts to the bipartite case Latapy et al. (2008) lead to similar extensions for bipartite stream graphs.

A bipartite graph $G = (T, \perp, E)$ is defined by a set of top nodes T , a set of bottom nodes \perp , and a set of links $E \subseteq T \times \perp$: links may exist only between top and bottom nodes. This models data like client–product relations or affiliation networks: the considered nodes belong to two different sets and links may exist only between nodes in one set and nodes in the other set.

In G , $n_T = |T|$ and $n_{\perp} = |\perp|$ denote the number of top and bottom nodes. The definition of the number of links m is the same as in classical graphs. The (bipartite) density of G is then $\delta(G) = \frac{m}{n_T \cdot n_{\perp}}$: it is the probability when one takes two

nodes that may be linked together that they indeed are. Node neighborhoods and degrees are defined like in a classical (non-bipartite) graph. The average top and bottom degrees d_T and d_{\perp} of G are the average degrees of top and bottom nodes, respectively.

The top and bottom projections $G_T = (T, E_T)$ and $G_{\perp} = (\perp, E_{\perp})$ of G are defined by $E_T = \cup_{v \in \perp} N(v) \otimes N(v)$ and $E_{\perp} = \cup_{v \in T} N(v) \otimes N(v)$, respectively. In other words, in G_T , two (top) nodes are linked together if they have (at least) a (bottom) neighbor in common in G , and G_{\perp} is defined symmetrically. If $v \in T$ (resp. $v \in \perp$) then $N(v)$ always is a (not necessarily maximal) clique in G_{\perp} (resp. G_T).

Given a top node $v \in T$ (the case of bottom nodes is symmetrical), let us denote by $G \setminus v$ the (bipartite) graph obtained by removing node v and all its links from G : $G \setminus v = (T \setminus \{v\}, \perp, E \setminus (\{v\} \times \perp))$. The redundancy $rc(v)$ of $v \in T$ is the density of the sub-graph of $(G \setminus v)_{\perp}$ induced by its neighborhood $N(v)$ in G . In other words, it is the fraction of its pairs of neighbors that have (at least) another neighbor in common.

We define a bipartite stream graph $S = (T, \perp, W, E)$ from a set of top nodes T , a set of bottom nodes \perp , a time

span T , and two sets $W \subseteq T \times (T \cup \perp)$ and $E \subseteq T \times T \times \perp$, such that $(t, u, v) \in E$ implies $(t, u) \in W$ and $(t, v) \in W$, see Fig. 22 (left) for an illustration.

In S , we extend $n = \frac{|W|}{|T|}$ into $n_T = \frac{|W \cap (T \times T)|}{|T|}$ and $n_\perp = \frac{|W \cap (T \times \perp)|}{|T|}$, the numbers of top and bottom nodes, respectively. We extend $k = \frac{|W|}{|V|}$ into $k_T = \frac{|W \cap (T \times T)|}{|T|}$ and $k_\perp = \frac{|W \cap (T \times \perp)|}{|\perp|}$ similarly, and we define m and l like for classical (non-bipartite) stream graphs.

We define the (bipartite) density of S as $\delta(S) = \frac{\sum_{u \in T, v \in \perp} |T_{uv}|}{\sum_{u \in T, v \in \perp} |T_u \cap T_v|}$: it is the probability when one takes two

nodes when they may be linked together that they indeed are. We define node neighborhoods and degrees like in a classical (non-bipartite) graph. We define the average top and bottom degrees d_T and d_\perp of S as the average degrees of top and bottom nodes, respectively, weighted by their presence time in the stream.

We define the top and bottom projections $S_T = (T, T, W_T, E_T)$ and $S_\perp = (T, \perp, W_\perp, E_\perp)$ of S by $W_T = W \cap (T \times T)$, $W_\perp = W \cap (T \times \perp)$, $E_T = \cup_{(t,v) \in W_\perp} \{(t, uv) \text{ s.t. } (t, u, v) \in E \text{ and } (t, w, v) \in E\}$ and $E_\perp = \cup_{(t,v) \in W_T} \{(t, uv) \text{ s.t. } (t, v, u) \in E \text{ and } (t, v, w) \in E\}$, respectively. In other words, in S_T two (top) nodes are linked together at a given time instant if they have (at least) a (bottom) neighbor in common in S at this time, and S_\perp is defined symmetrically, see Fig. 22 for an illustration. Notice that, if $v \in T$ (resp. $v \in \perp$), then $N(v)$ always is a (not necessarily maximal) clique in S_\perp (resp. S_T).

Given a top node $v \in T$ (the case of bottom nodes is symmetrical), let us denote by $S \setminus v$ the (bipartite) stream graph obtained by removing node v and all its links from S : $S \setminus v = (T, T \setminus \{v\}, \perp, W \setminus (T \times \{v\}), E \setminus (T \times \{v\} \times \perp))$. The redundancy $rc(v)$ of $v \in T$ is the density of the sub-stream of $(S \setminus v)_\perp$ induced by its neighborhood $N(v)$ in S . In other words, it is the fraction of its pairs of neighbors and time instants that have (at least) another neighbor in common at this time.

If S is a graph-equivalent bipartite stream, then its corresponding graph also is bipartite. Moreover, the projections of S are also graph-equivalent streams, and their corresponding graphs are the projections of the graph corresponding to S . In addition, the bipartite properties of S are equivalent to the bipartite properties of its corresponding bipartite graph.

21 Related work

Studying interactions over time is crucial in a wide variety of contexts, leading to a huge number of papers dealing with various cases of interest. We cite, for instance, studies of phone calls (Kovanen et al. 2013; Blondel et al. 2015),

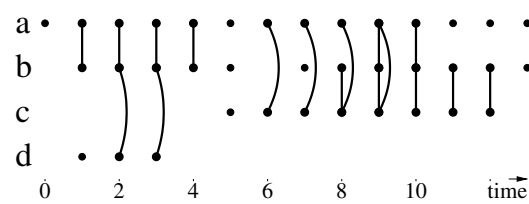


Fig. 20 Example $S = (T, V, W, E)$ of stream graph with discrete time. It is defined by $T = [0, 13] \subseteq \mathbb{N}$, i.e., $T = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13\}$, $V = \{a, b, c, d\}$, $T_a = T$, $T_b = \{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13\}$, $T_c = [5, 12] = \{5, 6, 7, 8, 9, 10, 11, 12\}$, $T_d = [1, 3] = \{1, 2, 3\}$, $T_{ab} = [1, 4] \cup [9, 10] = \{1, 2, 3, 4, 9, 10\}$, $T_{ac} = [6, 9] = \{6, 7, 8, 9\}$, $T_{bc} = [8, 12] = \{8, 9, 10, 11, 12\}$, $T_{bd} = [2, 3] = \{2, 3\}$, and $T_{ad} = T_{cd} = \emptyset$

contacts between individuals (Barrat and Cattuto 2013; Martinet et al. 2014), cattle exchanges (Dutta et al. 2014; Payen et al. 2017), messaging (Gomes et al. 2009; Gaumont et al. 2016b), or internet traffic (Harshaw et al. 2016), (Viard and Latapy 2014), but we could cite hundreds more. In each practical context, researchers and engineers face the challenge of analyzing the jointly temporal and structural nature of interactions, and they develop ad-hoc methods and tools to do so. Several surveys of these works are available from various perspectives (Masuda and Lambiotte 2016; Sizemore and Bassett 2017; Thompson et al. 2017; Snijders et al. 2010; Holme 2015; George and Kim 2013; Holme and Saramäki 2012; Doreian and Stokman 1997).

The most classical approach consists in splitting time into slices and then building a graph, often called snapshot, for each time slice: its nodes and links represent the interactions that occurred during this time slice. One obtains a sequence of snapshots (one for each slice), and may study the time evolution of their properties, see, for instance, Sikdar et al. (2015), Leskovec et al. (2007), Santoro et al. (2011), Gulyás et al. (2013), Blonder et al. (2012), Uddin et al. (2013), among many others. In Batagelj and Praprotnik (2016), the authors even design a general framework to combine and aggregate wide classes of temporal properties, thus providing a unified approach for snapshot sequence studies. However, these approaches need time slices large enough to ensure that each snapshot captures significant information. However, large slices lead to losses of temporal information, since all interactions within a same slice are merged. In addition, several or even varying slice durations may be relevant. As a consequence, choosing appropriate time slices is a research topic in itself (Léo et al. 2015; Ribeiro et al. 2013; Krings et al. 2012; Scholtes et al. 2016; Caceres and Berger-Wolf 2013). More importantly, key concepts like paths make little sense in this framework: paths within a slice do not respect the dynamics of interactions, and paths over several time slices are difficult to handle Léo et al. (2015).

To avoid these issues, several authors propose to encode the full information into various kinds of augmented graphs.

In Casteigts et al. (2012), Batagelj and Praprotnik (2016), Santoro et al. (2011), for instance, authors consider the graph of all nodes and links occurring within the data (i.e., the graph induced by the stream), and label each node and link with its presence times. In Wehmuth et al. (2015), Kostakos (2009), Michail (2015); Takaguchi et al. (2016), the authors duplicate each node into as many copies as its number of occurrences (they assume discrete time steps); then, an interaction between two nodes at a given time is encoded by a link between the copies of these nodes at this time, and each copy of a node is connected to its copy at the next time step. In Whitbeck et al. (2012), Nicosia et al. (2012) and others, the authors build reachability graphs: two nodes are linked together if they can reach each other in the stream. Other works transform the stream into multi-layer or multi-aspect graphs (Wehmuth et al. 2015, 2016; Kivelä et al. 2014). With such encodings, some key properties of the stream are equivalent to properties of the obtained graph, and therefore, studying this graph sheds light on the original data.

All these approaches have a clear advantage: once the data are transformed into one or several graphs, it is possible to use graph tools and concepts to study the interactions under concern. In the same spirit, various powerful methods for graph studies are extended to cope with the dynamics. This leads, for instance, to algebraic approaches for temporal network analysis (Batagelj and Praprotnik 2016; Praprotnik and Batagelj 2016), dynamic stochastic block models (Xu and Hero 2013; Matias and Miele 2016; Corneli et al. 2015, 2016), dynamic Markovian models (Stadtfeld and Block 2017; Stadtfeld et al. 2017; Snijders 2001; Snijders et al. 2010), signals on temporal networks (Hamon et al. 2015), adjacency tensors (Sun et al. 2006; Gauvin et al. 2014), temporal networks studies with walks (Starnini et al. 2012; Rocha and Masuda 2014; Saramäki and Holme 2015), dynamic graphlets (Hulovatyy et al. 2016; Harshaw et al. 2016) and temporal motif counting approaches (Kovanen et al. 2011; Paranjape et al. 2017). Clearly, these works extend higher level methods to the temporal setting, whereas we focus here on the most basic graph concepts, in the hope that they will form a unifying ground to such works.

Similar to what we do here, several works emphasize the importance of the streaming nature of interactions over time, then called contact sequences, temporal networks, or relational event sequences, see, for instance, Holme (2015), Holme and Saramäki (2012), Nicosia et al. (2013), Batagelj and Praprotnik (2016), Masuda and Lambiotte (2016), Butts (2008), Stadtfeld and Block (2017). Complementary to the approaches outlined above that extend methods, these works extend various graph concepts to deal with time.

In particular, path-related concepts received much attention because of their importance for spreading phenomena

and communication networks, see, for instance, Holme (2015), Whitbeck et al. (2012), Tang et al. (2010), Payen et al. (2017). Interestingly, although paths defined in these papers are similar to those we consider here, most derived concepts remain node-oriented. For instance, most authors define the centrality of a given node and connected components as sets of nodes (without time information) (Batagelj and Praprotnik 2016; Nicosia et al. 2013; Santoro et al. 2011; Whitbeck et al. 2012; Nicosia et al. 2012; Tang et al. 2010). In Chinellato et al. (2015), the authors introduce a centrality for time instants. Since the centrality of nodes may greatly change over time Magnien and Tarissan (2015), it is important to define centralities of each node at each time instant. Some authors did so for various kinds of centralities (Taylor et al. 2017; Flores and Romance 2017; Takaguchi et al. 2016; Tang et al. 2010; Sizemore and Bassett 2017), but, up to our knowledge, we are the first ones to consider paths from all nodes at all time instants to all other nodes at all other time instants, although a similar approach is proposed in Kivelä et al. (2017) for studying percolation. This approach has the advantage of fully capturing the dynamics of the data, in particular the fact that nodes are not always present.

Some works go beyond path-related notions and study dynamics of node and link presence, link repetitions, instantaneous degree, and triadic closure (Zignani et al. 2014; Hernández-Orallo et al. 2016; Stadtfeld and Block 2017; Conan et al. 2007; Tang et al. 2010; Perry and Wolfe 2013; Leskovec et al. 2008; Newman 2001; Uddin et al. 2016; Batagelj and Praprotnik 2016). However, up to our knowledge, there exists no previous generalization of density, neighborhood, or clustering coefficient that avoids time slicing. Interestingly, a notion of degree very close to the one we propose here was introduced in the context of medical studies (Uddin et al. 2014). A notion close to average degree is introduced in Rozenshtein et al. (2017) for dense dynamic sub-graphs searching. We also studied preliminary notions of density, cliques, quotient streams, and dense sub-streams in our own previous work (Viard et al. 2016; Gaumont et al. 2016a, b; Viard et al. 2015; Viard and Latapy 2014).

Finally, although there is a very rich body of works on temporal networks, dynamic graphs, longitudinal networks, time-varying graphs, relational event models, etc, none of these works aims at extending the basic graph theoretic language to the situation, where time and structure are equally important, like we try to do here. By defining such a basic language for streams, we expect to give a more formal, unified, and rigorous ground to the variety of works dealing with interactions over time.

22 Conclusion

In this paper, we introduce a formalism to deal directly with the intrinsically temporal and structural nature of interactions over time. We first define elementary concepts like numbers of nodes and links, density, clusters, and paths (Sects. 3–6 and Sect. 14). From them, we derive more advanced concepts like cliques, neighborhoods, degrees, clustering coefficients, and connected components (Sects. 7–10 and Sect. 15), and we show how to go further by introducing quotient streams, line streams, k -cores and centralities (Sects. 11–13 and Sect. 17). Our formalism is able to cope with both discrete and continuous time (Sect. 18), with both instantaneous links and links with durations (Sect. 19), and we also consider the case, where nodes have no dynamic, that we call link streams. Last but not least, our formalism may be extended to incorporate various features of the data, and we illustrate this with bipartite streams in Sect. 20.

The strength of our approach is to rely on very basic (but non-trivial) innovations like non-integer numbers of nodes and links, symmetric roles for time instants and nodes, a simple and intuitive concept of density, an elementary definition of clusters, and paths that connect a node at a given time to a node at a given time. These basic concepts make it easy to define more advanced objects: neighborhoods are clusters, degrees are fractional numbers of nodes in the neighborhoods, clustering coefficients are densities of neighborhoods, betweenness centralities are fractions of paths from any node at any time to any node at any time, etc. We demonstrate the strength of this approach by extending more advanced graph concepts such as quotient graph, trees, line graph, and k -cores, among others. Their definitions are mere retranscriptions of classical graph definitions into our formalism for stream graphs and link streams, and one may easily extend many other notions in this way.

In addition to this self-consistency, our formalism is consistent with graph theory in a very strong and precise way: if one considers a stream graph with no dynamics (nodes

are present all the time, and two nodes are either linked all the time or not at all), then the stream graph is equivalent to a graph and its stream properties are equivalent to the properties of the corresponding graph. As a consequence, our formalism is a generalization of graph theory, which provides a solid ground for generalizing other graph notions.

With our formalism, one is equipped with a wide set of concepts for describing data modeled as a stream graph or a link stream. It is natural to start with the description of how elementary metrics like k_t (the fraction of nodes present at time t) evolve over time, and of distributions of values of n_v (the fraction of time at which v is present) for all nodes. One may then study the instantaneous degree distribution, the degree distribution of nodes, and the time evolution of the time degree. More advanced metrics and properties, such as connectedness, clustering coefficient or centralities, give finer insight on the data. Finally, just like graph concepts do for relations, our formalism provides a language for describing interactions over time in an intuitive way, both at global and more local levels. Importantly, it does not require to choose a specific time scale for conducting such studies.

Data that would benefit from such an approach are countless, but we believe that analysis of network traffic, mobility traces, and financial transactions is among the most promising ones, and we are working on such applications. Indeed, modeling such data with (directed, weighted) stream graphs and link streams captures most of their features, and progress in these fields is currently limited by the lack of appropriate modeling. Some preliminary works show the interest of stream graphs in the contexts of internet traffic analysis Viard et al. (2018); Wilmet et al. (2018), contacts between individuals Viard et al. (2015, 2016), and recommender systems Viard and Fournier-S'niehotta (2018), but most remains to be done in this direction.

To conduct such real-world applications, it is crucial to design and implement convenient software able to efficiently compute the properties of large stream graphs. Work in this direction is in progress for the properties presented in this paper. However, it must be clear that some concepts raise serious algorithmic challenges. We worked, for instance,

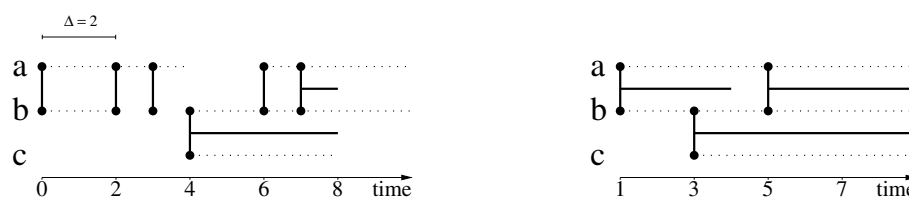


Fig. 21 Δ -analysis of a stream graph. We display a stream graph $S = (T, V, W, E)$ (left) and the stream graph $S_\Delta = (T_\Delta, V, W_\Delta, E_\Delta)$ (right) derived from it with $\Delta = 2$. Here, $T = [0, 10]$, $V = \{a, b, c\}$, $W = ([0, 4] \cup [6, 10]) \times \{a\} \cup ([0, 2] \cup [3, 4] \cup [4, 10]) \times \{b\} \cup [4, 8] \times$

$\{c\}$, and $E = (\{0, 2, 3, 6\} \cup [7, 8]) \times \{ab\} \cup [4, 8] \times \{bc\}$, leading to $T_\Delta = [1, 9]$, $W_\Delta = [1, 9] \times \{a, b\} \cup [3, 9] \times \{c\}$ and $E_\Delta = ([1, 4] \cup [5, 9]) \times \{ab\} \cup [3, 9] \times \{bc\}$. Notice that S contains instantaneous links, like, for instance, $(0, ab)$, and instantaneous nodes, like $(3, b)$

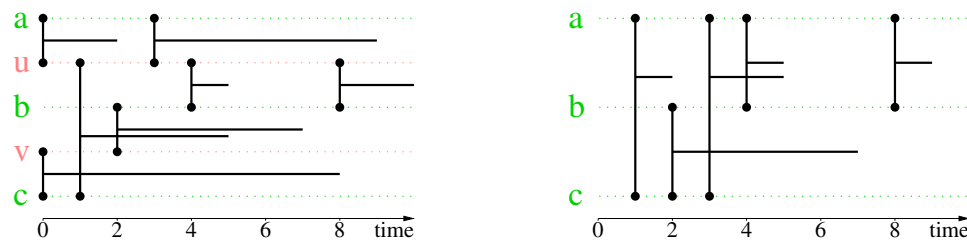


Fig. 22 Left: a bipartite link stream $L = (T, \top, \perp, E)$ with $T = [0, 10]$, $\top = \{u, v\}$, $\perp = \{a, b, c\}$, and $E = ([0, 2] \cup [3, 9]) \times \{(u, a)\} \cup ([4, 5] \cup [8, 10]) \times \{(u, b)\} \cup [1, 5] \times \{(u, c)\} \cup [2, 7] \times \{(v, b)\} \cup [0, 8] \times \{(v,$

$c)\}$. Right: its \perp projection L_{\perp} . For instance, a and c are linked together from time 3 to 5, because they both have u in their neighborhood for this time period in S

on clique and dense sub-stream computations (Viard et al. 2016; Gaumont et al. 2016b), and previous work exists on various problems, see, for instance, Wehmuth et al. (2016), Bui-Xuan et al. (2003), Casteigts et al. (2012, 2015), Bhadra and Ferreira (2012), Huanhuan et al. (2014). In particular, the authors of Casteigts et al. (2012) define a first complexity hierarchy for stream graphs. Still, most remains to be done in the design of efficient algorithms for stream graphs and the understanding of their complexity. This may lead to counter-intuitive results with important practical impact. For instance, although stream graphs are richer than usual graphs, because they include time information and their number of links is unbounded (in graphs, it is bounded by $\frac{n(n-1)}{2}$), computing their properties may be easier than computing the ones of induced graphs. Indeed, computation costs are often related to properties like maximum degree or maximum clique size, which are in general smaller in stream graphs than in their induced graphs. In addition, the temporal nature of stream graphs makes it easier to distribute storage and some computations by dividing the stream with respect to time windows.

Another important direction is the design of models of stream graphs and link streams, which play a crucial role for simulations and proofs. In particular, an important approach in graph studies consists in generating uniformly at random graphs that have a prescribed set of properties. For instance, the Erdős–Renyi model generates graphs with prescribed size and density, while the configuration model generates graphs with prescribed size and degree distribution. The definitions we introduce in this paper (in particular for density and degree) open the way to the definition of models for generating stream graphs with prescribed properties, and to a more unified understanding of already existing models, like the ones defined in Zhao et al. (2013), Laurent et al. (2015), Butts (2008), Snijders (2001), Leskovec et al. (2008), Snijders et al. (2010), Gulyás et al. (2013), Karsai et al. (2011), for instance.

In this paper, we extended classical concepts of graph theory to stream graphs, but it is clear that many other

concepts call for such generalizations. This includes, for instance, random walks and key concepts derived from them, like pagerank and eigenvector centralities, structural graph concepts like modular decomposition or treewidth, as well as more elaborate concepts like communities and modularity. Some of these directions were already explored in part (Starnini et al. 2012; Rocha and Masuda 2014; Saramäki and Holme 2015; Gaumont et al. 2016a), and we believe that stream graphs provide a promising framework to help explore them further.

In this direction, one may notice that stream graphs are not only generalizations of graphs. They actually lie at the crossroad of two very rich and powerful scientific areas: graph theory, as we have seen, and time series analysis. Indeed, if a stream graph has no dynamics, then it is equivalent to a graph; if it has no structure then it is equivalent to a time series. As a consequence, we consider that a very promising direction for future work is to *generalize time series concepts to stream graphs*, in a way similar to what we did with graph concepts in this paper.

Acknowledgements This work is funded in part by the European Commission H2020 FETPROACT 2016-2017 program under Grant 732942 (ODYCCEUS), by the ANR (French National Agency of Research) under Grants ANR-15-CE38-0001 (AlgoDiv) and ANR-13-CORD-0017-01 (CODDDE), by the French program “PIA - Usages, services et contenus innovants” under Grant O18062-44430 (REQUEST), and by the Ile-de-France program FUI21 under Grant 16010629 (iTRAC). We warmly thank the many colleagues and friends who read preliminary versions of this work and provided invaluable feedback.

References

- Barabási A-L, Pósfai M (2016) Network science. Cambridge University Press, Cambridge
- Barrat A, Cattuto C (2013) Temporal networks of face-to-face human interactions. Springer, Berlin, Heidelberg, pp 191–216
- Batagelj V, Praprotnik S (2016) An algebraic approach to temporal network analysis based on temporal quantities. Social Netw Anal Mining 6(1):28:1–28:22

- Berge C (1962) The theory of graphs and its applications. Wiley, New York
- Bhadra S, Ferreira A (2012) Computing multicast trees in dynamic networks and the complexity of connected components in evolving graphs. *J Internet Serv Appl* 3(3):269–275
- Blondel VD, Decuyper A, Krings G (2015) A survey of results on mobile phone datasets analysis. *EPJ Data Sci* 4(1):10
- Blonder B, Wey TW, Dornhaus A, James R, Sih A (2012) Temporal dynamics and network analysis. *Methods Ecol Evol* 3(6):958–972
- Bondy JA (1976) *Graph Theory Appl*. Elsevier Science Ltd., Oxford
- Bui-Xuan B-M, Ferreira A, Jarry A (2003) Computing shortest, fastest, and foremost journeys in dynamic networks. *Int J Found Comput Sci* 14(2):267–285
- Butts CT (2008) A relational event framework for social action. *Sociol Methodol* 38(1):155–200
- Caceres RS, Berger-Wolf T (2013) Temporal scale of dynamic networks. Springer, Berlin, Heidelberg, pp 65–94
- Casteigts A, Flocchini P, Mans B, Santoro N (2015) Shortest, fastest, and foremost broadcast in dynamic networks. *Int J Found Comput Sci* 26(4):499–522
- Casteigts A, Flocchini P, Quattrociocchi W, Santoro N (2012) Time-varying graphs and dynamic networks. *IJPEDES* 27(5):387–408
- Chinellato CE, Borges VA, Klaus W, Artur Z, da Silva APC (2015) Time centrality in dynamic complex networks. *Adv Complex Syst* 18(7–8):1550023
- Conan V, Leguay J, Friedman T (2007) Characterizing pairwise inter-contact patterns in delay tolerant networks. In: Proceedings of the 1st international conference on autonomic computing and communication systems, autonomics '07, pages 19:1–19:9, ICST, Brussels, Belgium, Belgium. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)
- Corneli M, Latouche P, Rossi F (2016) Block modelling in dynamic networks with non-homogeneous poisson processes and exact icl. *Social Netw Anal Mining* 6(1):55
- Corneli M, Latouche P, Rossi F (2015) Modelling time evolving interactions in networks through a non stationary extension of stochastic block models. In: ASONAM '15 Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2015, pp 1590–1591
- David E, Jon K (2010) *Networks, crowds, and markets: reasoning about a highly connected world*. Cambridge University Press, New York
- Diestel R (2012) *Graph theory*, 4th edn, volume 173 of Graduate texts in mathematics. Springer, New York
- Doreian P, Stokman F (1997) *Evolution of social networks*. The journal of mathematical sociology, vol 1. Gordon and Breach Publishers, Amsterdam
- Dutta BL, Ezanno P, Vergu E (2014) Characteristics of the spatio-temporal network of cattle movements in France over a 5-year period. *Prev Vet Med* 117(1):79–94
- Flores J, Romance M (2017) On eigenvector-like centralities for temporal networks: discrete vs. continuous time scales. *J Comput Appl Math* 330:1041–1051
- Gaumont N, Magnien C, Latapy M (2016a) Finding remarkably dense sequences of contacts in link streams. *Social Netw Anal Mining* 6(1):87:1–87:14
- Gaumont N, Viard T, Fournier-S'niehotta R, Wang Q, Latapy M (2016b) Analysis of the temporal and structural features of threads in a mailing-list. Springer International Publishing, Cham, pp 107–118
- Gauvin L, Panisson A, Cattuto C (2014) Detecting the community structure and activity patterns of temporal networks: a non-negative tensor factorization approach. *PLOS One* 9(1):1–13
- George B, Kim S (2013) *Spatio-temporal Networks: modeling and algorithms*. SpringerBriefs in Computer Science. Springer, New York
- Gomes LH, Almeida VAF, Almeida JM, Castro FDO, Bettencourt LA (2009) Quantifying social and opportunistic behavior in email networks. *Adv Complex Syst* 12(01):99–112
- Gulyás L, Kampis G, Legendi RO (2013) Elementary models of dynamic networks. *Eur Phys J Spec Top* 222(6):1311–1333
- Hamon R, Borgnat P, Flandrin P, Robardet C (2015) Duality between temporal networks and signals: Extraction of the temporal network structures. CoRR. <https://arxiv.org/abs/1505.03044>
- Harshaw CR, Bridges RA, Iannacone MD, Reed JW, Goodall JR (2016) Graphprints: towards a graph analytic method for network anomaly detection. In: Proceedings of the 11th annual cyber and information security research conference, CISRC '16, ACM, New York, NY, USA, pp 15:1–15:4
- Hernández-Orallo E, Cano JC, Calafate CT, Manzoni P (2016) New approaches for characterizing inter-contact times in opportunistic networks. *Ad Hoc Netw* 52:160–172 (**Modeling and Performance Evaluation of Wireless Ad Hoc Networks**)
- Holme P (2015) Modern temporal network theory: a colloquium. *Eur Phys J B* 88(9):234
- Holme P, Saramäki J (2012) Temporal networks. *Phys Rep* 519(3):97–125 (**Temporal Networks**)
- Huanhuan W, Cheng J, Huang S, Ke Y, Yi L, Yanyan X (2014) Path problems in temporal graphs. *Proc VLDB Endow* 7(9):721–732
- Hulovatyy Y, Chen H, Milenkovic T (2016) Exploring the structure and function of temporal networks with dynamic graphlets. *Bioinformatics* 32(15):2402
- Karsai M, Kivela M, Pan RK, Kaski K, Kertész J, Barabási A-L, Saramäki J (2011) Small but slow world: How network topology and burstiness slow down spreading. *Phys Rev E* 83:025102
- Kivela M, Arenas A, Barthelemy M, Gleeson JP, Moreno Y, Porter MA (2014) Multilayer networks. *J Complex Netw* 2(3):203–271
- Kivela M, Cambe J, Saramäki J, Karsai M (2018) Mapping temporal-network percolation to weighted, static event graphs. *Sci Rep* 8:12357. <https://doi.org/10.1038/s41598-018-29577-2>
- Kostakos V (2009) Temporal graphs. *Physica A Stat Mech Appl* 388(6):1007–1023
- Kovanen L, Karsai M, Kaski K, Kertész J, Saramäki J (2011) Temporal motifs in time-dependent networks. *J Stat Mech Theory Exp* 2011(11):P11005
- Kovanen L, Kaski K, Kertész J, Saramäki J (2013) Temporal motifs reveal homophily, gender-specific patterns, and group talk in call sequences. *PNAS* 110(45):18070–18075. <https://doi.org/10.1073/pnas.1307941110>
- Krings G, Karsai M, Bernhardsson S, Blondel VD, Saramäki J (2012) Effects of time window size and placement on the structure of an aggregated communication network. *EPJ Data Sci* 1(1):4
- Latapy M, Magnien C, Del Vecchio N (2008) Basic notions for the analysis of large two-mode networks. *Soc Netw* 30(1):31–48
- Laurent G, Saramäki J, Karsai M (2015) From calls to communities: a model for time-varying social networks. *Eur Phys J B* 88(11):301
- Leskovec J, Kleinberg JM, Faloutsos C (2007) Graph evolution: densification and shrinking diameters. *TKDD* 1(1):2
- Leskovec J, Backstrom L, Kumar R, Tomkins A (2008) Microscopic evolution of social networks. In: Ying L, Bing L, Sunita S (eds) Proceedings of the 14th ACM SIGKDD international conference on knowledge discovery and data mining, ACM, Las Vegas, Nevada, USA, Aug 24–27, pp 462–470
- Léo Y, Crespelle C, Fleury E (2015) Non-altering time scales for aggregation of dynamic networks into series of graphs. In Felipe Huici and Giuseppe Bianchi, editors, Proceedings of the 11th ACM conference on emerging networking experiments and technologies, CoNEXT 2015, ACM, Heidelberg, Germany, Dec 1–4, 2015, pp 29:1–29:7

- Magnien C, Tarissan F (2015) Time evolution of the importance of nodes in dynamic networks. In: ASONAM '15 Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2015. ACM, New York, pp 1200–1207
- Martinet L, Crespelle C, Fleury E (2014) Dynamic contact network analysis in hospital wards. Springer International Publishing, Cham, pp 241–249
- Masuda N, Lambiotte R (2016) A guide to temporal networks. Series on complexity science, vol 4. World Scientific, UK
- Matias C, Miele V (2017) Statistical clustering of temporal networks through a dynamic stochastic block model. *J R Stat Soc Ser B* 79:1119–1141
- Michail O (2015) An introduction to temporal graphs: an algorithmic perspective. Springer International Publishing, Cham, pp 308–343
- Newman MEJ (2001) Clustering and preferential attachment in growing networks. *Phys Rev E* 64:025102
- Newman M (2010) Networks: an introduction. Oxford University Press Inc, New York
- Nicosia V, Tang J, Musolesi M, Russo G, Mascolo C, Latora V (2012) Components in time-varying graphs. *Chaos: an Interdisciplinary. J Nonlinear Sci* 22(2):023101
- Nicosia V, Tang J, Mascolo C, Musolesi M, Russo G, Latora V (2013) Graph metrics for temporal networks. Springer Berlin Heidelberg, Berlin, Heidelberg, pp. 15–40
- Paranjape A, Benson AR, Leskovec J (2017) Motifs in temporal networks. In: de Rijke M, Shokouhi M, Tomkins A, Zhang M (eds) Proceedings of the Tenth ACM international conference on web search and data mining, WSDM 2017, Cambridge, United Kingdom, February 6–10, pp 601–610
- Payen A, Tabourier L, Latapy M (2017) Impact of temporal features of cattle exchanges on the size and speed of epidemic outbreaks. In: Osvaldo G, Beniamino M, Sanjay M, Giuseppe B, Carmelo Maria T, Ana Maria ACR, David T, Bernady OA, Elena NS, Alfredo C (eds) Computational science and its applications—ICCSA 2017—17th International Conference, Trieste, Italy, July 3–6, 2017, Proceedings, Part II, volume 10405 of Lecture Notes in Computer Science. Springer, New York, pp 84–97
- Pei J, Silvestri F, Tang J (eds) (2015) Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining, ASONAM 2015, ACM. Paris, France, August 25–28, 2015
- Perry PO, Wolfe PJ (2013) Point process modelling for directed interaction networks. *J R Stat Soc Ser B (Statistical Methodology)* 75(5):821–849
- Praprotnik S, Batagelj V (2016) Semirings for temporal network analysis. CoRR <https://arxiv.org/abs/1603.08261>
- Ribeiro B, Perra N, Baronchelli A (2013) Quantifying the effect of temporal resolution on time-varying networks. *Sci Rep* 3:3006
- Rocha LEC, Masuda N (2014) Random walk centrality for temporal networks. *N J Phys* 16(6):063023
- Rozenshtein P, Tatti N, Gionis A (2017) Finding dynamic dense subgraphs. *ACM Trans Knowl Discov Data* 11(3):271–2730
- Santoro N, Quattrociochi W, Flocchini P, Casteigts A, Amblard F (2011) Time-varying graphs and social network analysis: Temporal indicators and metrics. In: 3rd AISB social networks and multiagent systems symposium (SNAMAS), pp 32–38
- Saramäki J, Holme P (2015) Exploring temporal networks with greedy walks. *Eur Phys J B* 88(12):334
- Scholtes I, Wider N, Garas A (2016) Higher-order aggregate networks in the analysis of temporal networks: path structures and centralities. *Eur Phys J B* 89(3):61
- Scott J (2017) Social network analysis. SAGE Publications, Thousand Oaks
- Sikdar S, Ganguly N, Mukherjee A (2015) Time series analysis of temporal networks. CoRR <https://arxiv.org/abs/1512.01344>
- Sizemore AE, Bassett DS (2018) Dynamic graph metrics: tutorial, toolbox, and tale. *NeuroImage* 180:417–427
- Snijders TAB (2001) The statistical evaluation of social network dynamics. *Sociol Methodol* 31(1):361–395
- Snijders TAB, van de Bunt GG, Steglich CEG (2010) Introduction to stochastic actor-based models for network dynamics. *Soc Netw* 32(1):44–60 (**Dynamics of Social Networks**)
- Stadtfeld C, Block P (2017) Interactions, actors, and time: dynamic network actor models for relational events. *Sociol Sci* 4:318–352
- Stadtfeld C, Hollway J, Block P (2017) Dynamic network actor models: investigating coordination ties through time. *Sociol Methodol* 47(1):1–40
- Starnini M, Baronchelli A, Barrat A, Pastor-Satorras R (2012) Random walks on temporal networks. *Phys Rev E* 85:056115
- Sun J, Tao D, Faloutsos C (2006) Beyond streams and graphs: dynamic tensor analysis. In: Tina ER, Lyle HU, Mark C, Dimitrios G (eds) Proceedings of the Twelfth ACM SIGKDD international conference on knowledge discovery and data mining, ACM, Philadelphia, PA, USA, Aug 20–23, 2006, pp 374–383
- Takaguchi T, Yano Y, Yoshida Y (2016) Coverage centralities for temporal networks. *Eur Phys J B* 89(2):35
- Tang J, Musolesi M, Mascolo C, Latora V (2010) Characterising temporal distance and reachability in mobile and online social networks. *SIGCOMM Comput Commun Rev* 40(1):118–124
- Tang J, Scellato S, Musolesi M, Mascolo C, Latora V (2010) Small-world behavior in time-varying graphs. *Phys Rev E* 81:055101
- Tang J, Musolesi M, Mascolo C, Latora V, Nicosia V (2010) Analysing information flows and key mediators through temporal centrality metrics. In: Proceedings of the 3rd workshop on social network systems, SNS '10, ACM, New York, NY, USA, pp 3:1–3:6
- Taylor D, Myers SA, Clauset A, Porter MA, Mucha PJ (2017) Eigenvector-based centrality measures for temporal networks. *Multi-scale Model Simul* 15(1):537–574
- Thompson WH, Brantefors P, Fransson P (2017) From static to temporal network theory: applications to functional brain connectivity. *Netw Neurosci* 1(2):69–99
- Tiphaine V, Raphaël FS (2018) Movie rating prediction using content-based and link stream features. CoRR, [abs/1805.02893](https://arxiv.org/abs/1805.02893)
- Uddin S, Hossain L, Wigand RT (2014) New direction in degree centrality measure: towards a time-variant approach. *Int J Inform Technol Decis Making* 13(4):865
- Uddin S, Khan A, Piraveenan M (2016) A set of measures to quantify the dynamicity of longitudinal social networks. *Complexity* 21(6):309–320
- Uddin MS, Mahendra P, Chung KSK, Hossain L (2013) Topological analysis of longitudinal networks. In: 46th Hawaii International Conference on System Sciences, HICSS 2013, Wailea, HI, USA, 7–10 January, 2013. pp 3931–3940
- Viard T, Latapy M, Magnien C (2016) Computing maximal cliques in link streams. *Theor Comput Sci* 609:245–252
- Viard T, Fournier-S'niehotta R, Magnien C, Latapy M (2018) Discovering patterns of interest in IP traffic using cliques in bipartite link streams. In: Cornelius S, Coronges K, Gonçalves B, Sinatra R, Vespignani A (eds) Complex networks IX. CompleNet 2018. Springer proceedings in complexity. Springer, Cham
- Viard T, Latapy M (2014) Identifying roles in an IP network with temporal and structural density. In: 2014 Proceedings IEEE INFOCOM Workshops, IEEE, Toronto, ON, Canada, April 27–May 2, 2014, pp 801–806
- Viard T, Latapy M, Magnien C (2015) Revealing contact patterns among high-school students using maximal cliques in link streams. In: ASONAM '15 Proceedings of the 2015 IEEE/ACM international conference on advances in social networks analysis and mining 2015. ACM, New York, pp 1517–1522

- Wasserman S, Faust K (1994) *Social network analysis: Methods and applications*, vol 8. Cambridge University Press, Cambridge
- Wehmuth K, Fleury É, Ziviani A (2016) Multiaspect graphs: algebraic representation and algorithms. *Algorithms* 10(4):1
- Wehmuth K, Fleury É, Ziviani A (2016) On multiaspect graphs. *Theor Comput Sci* 651:50–61
- Wehmuth K, Ziviani A, Fleury E (2015) A unifying model for representing time-varying graphs. In: 2015 IEEE international conference on data science and advanced analytics, DSAA 2015, Campus des Cordeliers, IEEE, Paris, France, October 19–21, 2015, pp 1–10
- West DB (2000) *Introduction to graph theory*, 2nd edn. Prentice Hall, Prentice
- Whitbeck J, de Amorim MD, Conan V, Guillaume JL (2012) Temporal reachability graphs. In: Özgür BA, Eylem E, Lili Q, Alex CS (eds) *The 18th annual international conference on mobile computing and networking, Mobicom'12*, ACM, Istanbul, Turkey, Aug 22–26, 2012, pp 377–388
- Wilmet A, Viard T, Latapy M, Lamarche-PR (2018) Degree-based outliers detection within ip traffic modelled as a link stream. In: *Proceedings of the 2nd network traffic measurement and analysis conference (TMA)*
- Xu KS, Hero AO (2013) *Dynamic stochastic blockmodels: statistical models for time-evolving networks*. Springer, Berlin, Heidelberg, pp 201–210
- Zhao K, Karsai M, Bianconi G (2013) *Models, entropy and information of temporal social networks*. Springer, Berlin, Heidelberg, pp 95–117
- Zignani M, Gaito S, Rossi GP, Zhao X, Zheng H, Zhao BY (2014) Link and triadic closure delay: temporal metrics for social network dynamics. In Eytan A, Paul R, Munmun De C, Bernie H, Alice HH (eds) *Proceedings of the eighth international conference on weblogs and social media, ICWSM 2014*, The AAAI Press, Ann Arbor, Michigan, USA, June 1–4, 2014