



# Temporal networks in biology and medicine: a survey on models, algorithms, and tools

Mohammad Mehdi Hosseinzadeh<sup>1</sup> · Mario Cannataro<sup>2</sup> · Pietro Hiram Guzzi<sup>2</sup> · Riccardo Dondi<sup>1</sup>

Received: 3 November 2022 / Revised: 16 December 2022 / Accepted: 17 December 2022 / Published online: 31 December 2022  
© The Author(s), under exclusive licence to Springer-Verlag GmbH Austria, part of Springer Nature 2022

## Abstract

The use of static graphs for modelling and analysis of biological and biomedical data plays a key role in biomedical research. However, many real-world scenarios present dynamic behaviours resulting in both node and edges modification as well as feature evolution. Consequently, ad-hoc models for capturing these evolutions along the time have been introduced, also referred to as dynamic, temporal, time-varying graphs. Here, we focus on temporal graphs, i.e., graphs whose evolution is represented by a sequence of time-ordered snapshots. Each snapshot represents a graph active in a particular timestamp. We survey temporal graph models and related algorithms, presenting fundamentals aspects and the recent advances. We formally define temporal graphs, focusing on the problem setting and we present their main applications in biology and medicine. We also present temporal graph embedding and the application to recent problems such as epidemic modelling. Finally, we further state some promising research directions in the area. Main results of this study include a systematic review of fundamental temporal network problems and their algorithmic solutions considered in the literature, in particular those having application in computational biology and medicine. We also include the main software developed in this context.

## 1 Introduction

Networks are commonly used for modelling association among the building blocks of complex systems in biology and medicine Cannataro et al. (2010). Models are often built from available data; thus, the availability of static data has lead to the introduction of simple static models (Aittokallio and Schwikowski 2006; Parthasarathy et al. 2010; Dondi et al. 2021a). Due to technological advances, the production of more sophisticated data has shed light into quantitative and time-varying relationships among biological entities in a given organism (Murugan et al. 2021; Mattsson and Takes 2021; Shawn et al. 2022). Therefore, researchers have introduced more complex models able to gather such information and to reuse them to produce relevant knowledge. Examples of these models are: temporal networks modelling cellular components, or spreading processes, like the spread

of information and disease (Russell et al. 2020; Petrizzelli et al. 2022; Cinaglia and Cannataro 2022).

We here focus on temporal (or time-varying) networks. A temporal network is a network whose edges are present only at certain points (called timestamps) in a discrete time domain, so it may be seen as a sequence of static networks, one for each timestamp, as represented in Fig. 1. Each edge may also be weighted or labelled, to carry more information. Temporal networks are a special kind of dynamic networks, which are graphs that from an initial state may change due to different events, not only to edges, but also to nodes (node addition and deletion, node splitting, and node merging) in a continuous evolution Kazemi et al. (2020).

After the introduction of these models, it has been clear that novel ad-hoc algorithms developed for the analysis of such networks were necessary. First approaches were based on the reductionist approach involving the aggregation of different dimensions (or timestamps) into a single network discarding all the information related to time. This approach enabled the reuse of the large plethora of existing methods but caused the loss of all the important information related to temporal evolution.

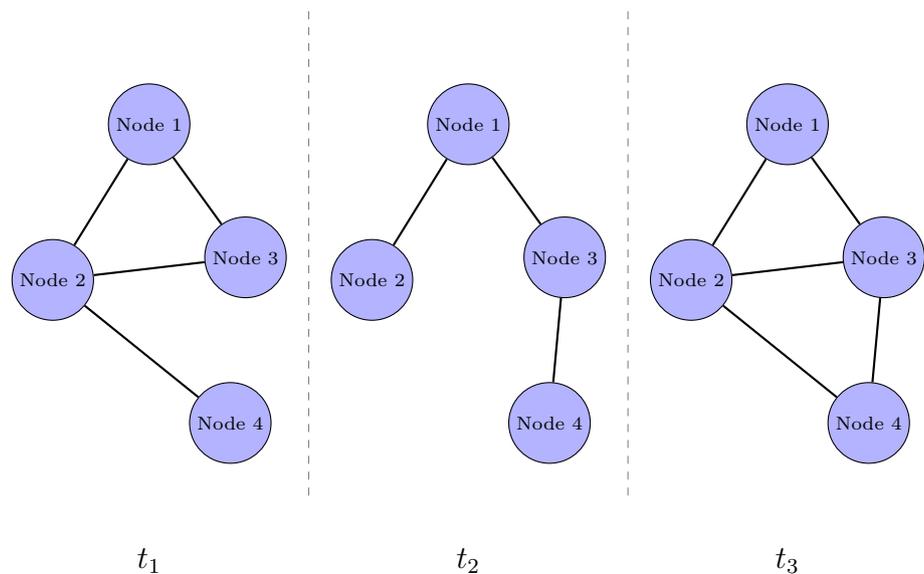
Consequently, to overcome this limitation, novel methods have been introduced. Many of these methods were based on the analysis of each single timestamp independently via

✉ Pietro Hiram Guzzi  
hguzzi@gmail.com

<sup>1</sup> Department of Literature, Philosophy, Communication Studies, University of Bergamo, Bergamo, Italy

<sup>2</sup> Department of Surgical and Medical Sciences and Data Analytics Research Center, University Magna Graecia of Catanzaro, Catanzaro, Italy

**Fig. 1** Figure depicts three temporal snapshots of a temporal network



the existing methods for static network analysis Kumar Das et al. (2021). Then, results were aggregated by means of time-series analysis. Nevertheless, it was clear that the best option was the analysis of the network using an holistic approach integrating the different snapshots.

Natural and social systems show a behaviour that can be described as temporal networks, where interactions are observed only intermittently Li et al. (2017). This happens for example in cellular metabolic networks Almaas et al. (2004), protein structure networks Ortuso et al. (2021) or in brain networks William and Peter (2016).

Holme and Saramaki presented in Holme and Saramäki (2012) the first review paper related to temporal graph in 2012. Main contribution of this seminal work was to define clearly the temporal graph data structure and main problems (Holme and Saramäki 2012, 2019; Masuda and Lambiotte 2016)

The survey in Michail (2016) focuses on algorithm design. It highlights several issues related to path problems, Menger's Theorem, dissemination, local and global properties, and network design.

In this paper, we present a systematic literature review of the main temporal network problems and their algorithmic solutions. The main aims of this paper are to introduce basic concepts of temporal networks and to review main algorithms for their analysis, as well as main applications in biology and medicine. We also include the main software developed in this context, with the goal of highlighting also the tools available to analyse temporal networks.

Some of these problems (for example, dense subgraph and colored motif discovering, graph covering) have not been analysed by other surveys and this represents a novel contributions of our survey. A second contribution is the focus on biological and medicine applications, which is not

deepen in the previous surveys. Finally, a third novel contribution is the presentation of software tools for the analysis of temporal networks, while other survey are mainly theoretical.

The rest of the paper is organised as follows. In Sect. 2, we define the main concepts related to temporal networks. In Sect. 3, we introduce some network measures that have been introduced to evaluate the relevance of nodes/edges in a temporal networks or general properties of a temporal network. In Sect. 4, we consider the main graph problems and the main computational results, including methods for temporal networks traversing, such as path finding and connectivity, and for covering and finding dense subgraphs in temporal networks.

In Sect. 5, we discuss motif search in temporal networks, that is the core operation in several network analysis methods, such as network alignment and network embedding.

In Sect. 6, we recall the main software tools for temporal networks analysis, that are key elements for the application of fundamental methods to solve concrete problems.

In Sect. 7, we review the main problems in biology and medicine that have been modelled with temporal graphs, such as brain connectome, protein–protein interaction networks, and epidemiology.

In Sect. 8, we report a summary of the main contributions of the paper.

Finally, in Sect. 9, we discuss open problems and future research directions.

## 1.1 Methodology

The reporting of this systematic review was guided by the standards of the Preferred Reporting Items for Systematic

Review and Meta-Analysis (PRISMA) Statement Sarkis-Onofre et al. (2021).

## 2 Basic concepts of temporal networks

A graph  $G = (V, E)$  is a pair of sets, where  $V$  is a set of nodes or vertices, and  $E$  is a set of edges (directed or undirected). A multigraph  $G = (V, E, D)$  consists of three sets, where  $D$  is a class of dimension sets, that allows to define properties to edges of the multigraph. A temporal graph is a special case of multigraph, where  $D$  is a set  $T$  of discrete and ordered timestamps. We present next the formal definition of temporal graph (see Fig. 1).

**Definition 1** A temporal graph  $G = (V, E, T)$  is defined over three sets:

1.  $V$ , a set of nodes
2.  $T$ , an ordered set of timestamps  $t_1 < t_2 \dots < t_M$
3.  $E$ , a set of temporal edges, where each temporal edge is a triplet  $(u, v, t)$ , with  $u, v \in V$  and  $t \in T$ .

Notice that in a temporal graph, edges may change from one timestamp to the other, while the set of nodes is not changing. In the following, we denote by  $n$  the number of nodes, by  $m$  the number of temporal edges, and by  $t_M$  the number of timestamps.

We denote by  $G(t) = (V, E(t))$ , where  $t \in T$ , the static graph induced by the active edges  $E(t)$  defined at timestamp  $t$ .  $G(t)$  is called the snapshot of  $G$  at timestamp  $t$ . The temporal graph  $G$  can be seen as a sequence of snapshots  $(G(1), G(2), \dots, G(t_M))$ , where  $t_M$  is the maximum timestamp in  $T$ . In Fig. 1, we have presented an example of a temporal graph consisting of three snapshots.

We denote by  $A(t)$ , where  $t \in T$ , the adjacency matrix (also called connectivity matrix) of a snapshot  $G(t)$  at timestamp  $t$ .  $A(t)$  is a  $V \times V$ -matrix, where  $A(t)_{u,v} = 1$  if  $(u, v, t) \in E(t)$  and  $u \neq v$  (there is an edge between nodes  $u$  and  $v$  at timestamp  $t$ ); otherwise,  $A(t)_{u,v} = 0$ . Notice that the diagonal entries of  $A(t)$  are equal to 0, that is,  $A(t)_{v,v} = 0$  for any  $v \in V$ .

A temporal graph can be seen also as an edge-labelled graph (see Fig. 2), where labels associated with an edge represent the timestamps where that edge is available (or active).

Next, we present the definition of underlying graph of a temporal graph. Informally, the underlying graph represents the relations and the structure of a temporal graph, no matter in which timestamp, and can be seen as the union of the snapshots (see the right part of Fig. 2) where the edges are not labelled by timestamps.

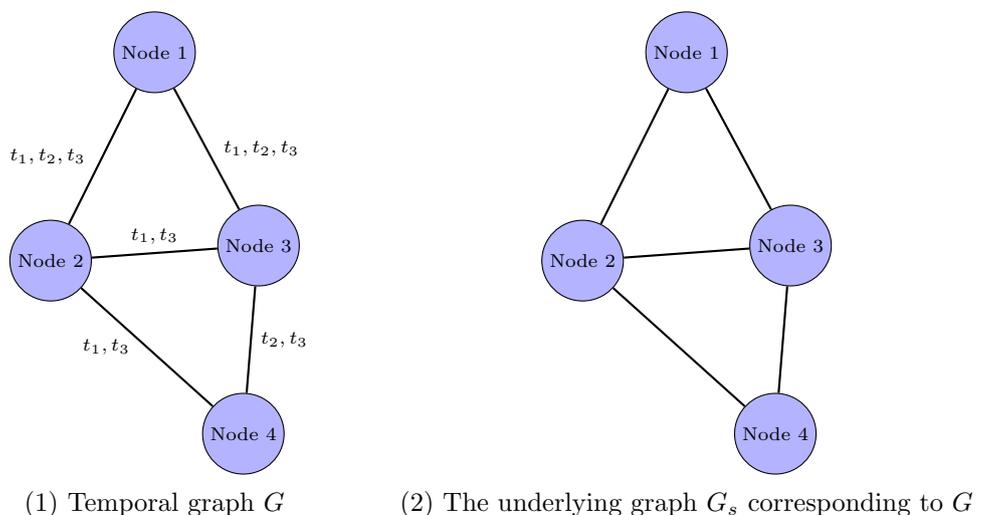
**Definition 2** Given a temporal graph  $G = (V, E, T)$ , the underlying static graph  $G_s = (V, E_s)$  of  $G$  is defined on the same node set  $V$ , while the set  $E_s$  of edges is defined as follows:

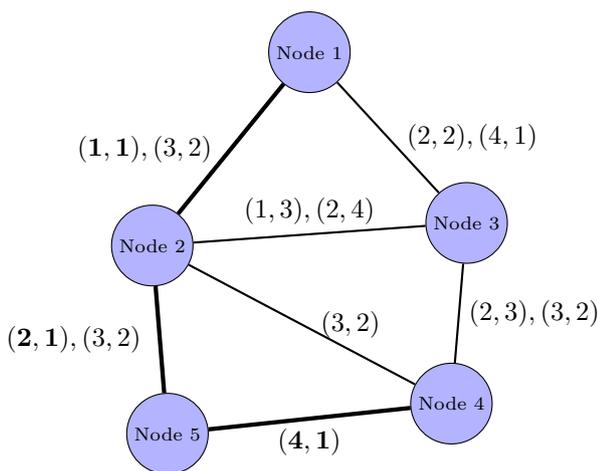
$$E_s = \{(u, v) : (u, v, t) \in E, \text{ for some } t \in T\}.$$

Notice that while a snapshot is a static graph that contains only the edges defined in a specific timestamp, the underlying graph contains all the edges defined in some timestamp of the time domain.

An extension of the definition of temporal graph has been given in Huanhuan et al. (2014), called extended temporal graph, where each temporal edge is denoted by a quadruple  $(u, v, t, \lambda)$ , with  $u, v$  are two nodes,  $t$  is a timestamp, and  $\lambda$  is the traversal time to go from  $u$  to  $v$  (see Fig. 3). Essentially, it represents a weighted version of the definition of temporal graphs, where if  $(u, v, t, \lambda)$  is a temporal edge, then starting

**Fig. 2** (1) A representation of the temporal graph of Fig. 1 as a labelled graph and (2) the corresponding underlying graph  $G_s$





**Fig. 3** An extended temporal graph, where edges are labelled by pair  $(t, \lambda)$ , where  $t$  is a timestamp and  $\lambda$  the time required to traverse the labelled edge at time  $t$ . In bold, a temporal path from Node 1 to Node 4 that starts at a time 1 and arrives at time 5

from node  $u$  at time  $t$ , it is possible to traverse the edge from  $u$  to  $v$  and arrive at time  $t + \lambda$  in  $v$ . Notice that in some applications, the weight/traversal time of an edge may change over time, and possibly, there could be different traversal time to go from  $u$  to  $v$  at the same time  $t$ .

One of main concept that has been extended from static graphs to temporal graphs is that of traversing a graph with a walk or a path. We start to give the definition of temporal walk (see the example of Fig. 4).

**Definition 3** Given a temporal graph  $G = (V, E, T)$ , a temporal walk in  $G$  is a sequence  $w = \langle v_1 e_1 v_2 e_2 \dots e_{q-1} v_q \rangle$  of nodes  $v_i, 1 \leq i \leq q$ , and temporal edges  $e_i, 1 \leq i \leq q - 1$ , such that the following properties hold:

- For each  $i$ , with  $1 \leq i \leq q - 1$ ,  $e_i = (v_i, v_{i+1}, t_i)$ , for some timestamps  $t_i$  of  $T$
- For each  $i$ , with  $1 \leq i \leq q - 1$ , consider the temporal edges  $e_i = (v_i, v_{i+1}, t_i)$  and  $e_{i+1} = (v_{i+1}, v_{i+2}, t_{i+1})$ , then  $t_i < t_{i+1}$  (or in some case  $t_i \leq t_{i+1}$ ).

The starting time of  $w$  is  $t_1$ , then ending time is  $t_{q-1}$ .

Now, we can define the related concept of temporal path, trail, and circuit (see Fig. 5 for an example of a temporal path in a temporal path and Fig. 3 for an example of a temporal path in an extended temporal graph).

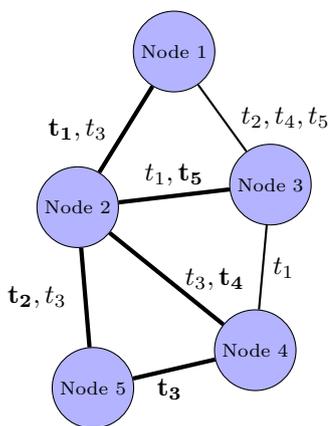
**Definition 4** Given a temporal graph  $G = (V, E, T)$ .

- A temporal path is a temporal walk where all the vertices are distinct.
- A temporal trail  $r = \langle v_1 e_1 v_2 e_2 \dots e_{q-1} v_q \rangle$  is a walk, such that each edge is traversed at most once, that is, for each  $e_i = (v_i, v_{i+1}, t_i)$  and  $e_j = (v_j, v_{j+1}, t_{j+1})$ , then  $(v_i, v_{i+1})$  is different from  $(v_j, v_{j+1})$ .
- A temporal circuit is a temporal walk that starts and ends from a same node.

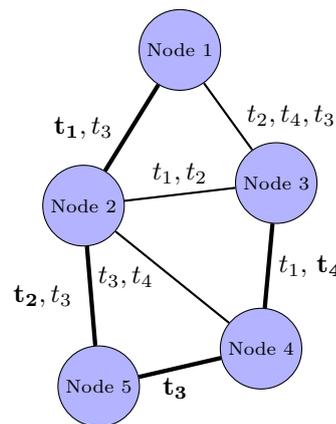
Another fundamental concept of static graphs that has been considered in temporal graphs is the node degree. Next, we introduce the definition of temporal node degree.

**Definition 5** Given a temporal graph  $G = (V, E, T)$ , a temporal degree of node  $v$  over time interval  $[t_i, t_j]$  in  $G$ , where  $t_i < t_j$  is given by

$$deg_v(t_i, t_j) = \sum_{u \in V \setminus \{v\}} \sum_{t=t_i}^{t_j} \phi(v, u, t), \tag{1}$$



**Fig. 4** An example of temporal walk in a temporal graph from Node 1 to Node 3. The thick edges belong to the walk; notice that the timestamps in bold satisfy the time constraint



**Fig. 5** An example of temporal path in a temporal graph. The thick edges belong to the path and each node on the path is traversed once; notice also in this case that the timestamps in bold satisfies the time constraint

where  $u \in V \setminus v$ , and  $\phi(v, u, t)$  is equal to 1 if there is a temporal edge between nodes  $v$  and  $u$  at timestamp  $t$ ; otherwise,  $\phi(v, u, t)$  is equal to 0.

Notice that  $deg_v(t)$  is the temporal degree of  $v$  in timestamp  $t$ , which is defined as in Eq. 1 where  $t_i = t_j = t$ .

Notice that we can define the function  $\phi(v, u)$  for the underlying static graph as follows. If there is at least one edge between nodes  $v$  and  $u$  (where  $u \in V \setminus v$ ) in a timestamp  $t$ , with  $t = 1, 2, \dots, t_M$ ,  $\phi(v, u)$  is equal to 1, otherwise  $\phi(v, u)$  is equal to 0. Formally,  $\phi(v, u)$  is defined as follows:

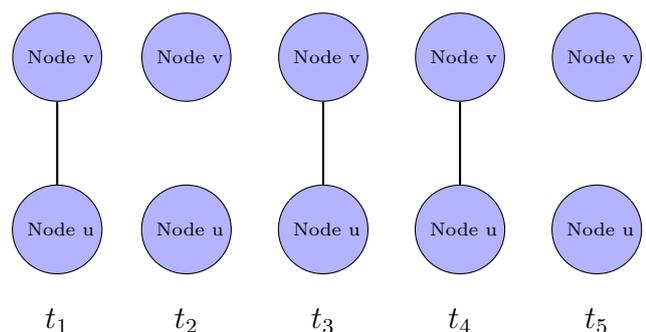
$$\phi(v, u) = \begin{cases} 1 & \text{if } \sum_{t=1}^{t_M} \phi(v, u, t) > 0 \\ 0 & \text{if } \sum_{t=1}^{t_M} \phi(v, u, t) = 0. \end{cases} \tag{2}$$

Now, we define inter-contact time between two nodes. It measures the difference between two consecutive timestamps where a certain temporal edge appears. Given two nodes connected by at least two temporal edges, the definition of inter-contact time is defined as follows.

**Definition 6** Given two nodes  $v$  and  $u$  connected by  $z \geq 2$  temporal edges, the inter-contact time  $\tau_{v,u}$  between  $v$  and  $u$  is a list of size  $z - 1$ , where each value of the list is defined as follows. Assume that the  $i$ -th temporal edge connecting  $v$  and  $u$ ,  $1 \leq i \leq z - 1$ , is in timestamp  $t_a$ , the  $i + 1$ -th temporal edge connecting  $v$  and  $u$  is in timestamp  $t_b > t_a$ , then the  $i$ -th value of  $\tau_{v,u}$  is defined as  $t_b - t_a$ .

Consider the example of Fig. 6, the inter-contact time between nodes  $v$  and  $u$  is the list [2, 1].

Next, we define the concept of optimal path and optimal distance in a temporal graph. Unlike in static graphs, in



**Fig. 6** A simple example of a temporal network with two nodes  $v$  and  $u$  and five timestamps that illustrates the concept of the inter-contact time. The inter-contact time  $\tau_{v,u}$  between nodes  $v$  and  $u$  is a list of size 2, since there exist three temporal edges connecting  $v$  and  $u$ .  $\tau_{v,u}$  is equal to [2,1], where the first value is the difference between timestamps of second and first temporal edge connecting  $v$  and  $u$ , that is timestamps 3 and 1; the second value of the vector is the difference between timestamps 4 and 3 (the timestamps of the third and second temporal edge connecting  $v, u$ )

temporal graphs, an optimal temporal path can be defined in several ways, based on different criteria such as arrival time, overall traversal time, and number of traversed edges. Therefore, here, we define optimal temporal path between two nodes  $u$  and  $v$  as a temporal path that starts in  $u$ , arrives in  $v$ , where  $u$  and  $v$  are distinct, and that minimizes one of the measures defined in Sect. 4.1. The measure can be the arrival time (leading to foremost temporal path), the overall traversal time (leading to fastest temporal path), or the number of traversed edges (leading to shortest temporal path). The optimal distance between  $u$  and  $v$ , denoted by  $\alpha_{u,v}$ , is the measure of an optimal temporal path in a temporal graph  $G$ . We denote by  $\alpha_{u,v}^I$ , the *optimal temporal distance* in the temporal graph defined over time interval  $I = [t_i, t_j]$ , where  $t_i \leq t_j$ . Notice that, if  $t = t_i = t_j$ ,  $\alpha_{u,v}^I$  is the *optimal temporal distance* in  $G(t)$ .

### 3 Temporal network measures

Temporal network measures have been widely applied to represent local properties of nodes or global properties of the entire network.

#### 3.1 Centrality measures

Centrality measures defined in static graphs have been extended to temporal graphs. In graph analysis, centrality measures are important concepts and have been widely used to identify important nodes in a graph. Centrality has been defined with different metrics and each of them shows the importance of a node from a different perspective. Here, we define three important centrality measures: degree centrality, closeness centrality, and betweenness centrality in a temporal graphs.

Degree centrality of a node shows the importance of the node based on the number of its interactions. A node with higher number of interactions can be a good candidate to pass information to large part of the nodes in a network. Degree centrality in a temporal graph is calculated, similarly to static case, with an additional sum across timestamps. More precisely, temporal degree centrality of node  $v$  in a temporal graph is calculated as the sum of the degrees of  $v$  in each snapshot. Following Thompson et al. (2017), Temporal Degree centrality  $TD(v)$  of node  $v$  is given by

$$TD(v) = \sum_{t=1}^{t_M} deg_v(t),$$

where recall that  $deg_v(t)$  is the degree of node  $v$  in snapshot  $G(t)$ .

The second centrality measure of a node that we consider is the temporal closeness centrality which shows the

importance of a node by measuring how fast other nodes are reachable from it in a temporal graph Pan and Saramäki (2011). Temporal closeness centrality of a node  $v$  is obtained by averaging the sum of the inverses of the optimal distances between  $v$  and other nodes of the temporal graph. Indeed, temporal closeness centrality  $TC(v)$  of node  $v$  is defined by

$$TC(v) = \frac{1}{n-1} \sum_{u \in V \setminus v} \frac{1}{\alpha_{v,u}}$$

where recall that  $\alpha_{v,u}$  is the optimal distance for any  $v, u \in V$  where  $v \neq u$ . Therefore, the node with higher closeness centrality is an ideal node to transmit quickly an information to all other nodes in a network.

Betweenness centrality measures the importance of a node based on the number of times that the node appears in an optimal temporal path between any pair of nodes. Nodes with high betweenness centrality are crucial for the commutation in a network, since they connect with optimal paths many nodes with each other. Betweenness centrality has been used in many applications such as social network analysis Tang et al. (2009), machine learning Şimşek and Barto (2008), and neuroscience van den Heuvel et al. (2010). Temporal betweenness centrality  $TB(v)$  of node  $v$  in a temporal graph is defined as

$$TB(v) = \sum_{s \neq v \neq u \in V} \frac{\gamma_{s,u}(v)}{\gamma_{s,u}}$$

where  $\gamma_{s,u}$  is the number of optimal paths between nodes  $s$  and  $u$  in the temporal graph and  $\gamma_{s,u}(v)$  is the number of optimal paths between nodes  $s$  and  $u$  which pass through node  $v$  in the temporal graph.

### 3.2 Other temporal network measures

In network mining, burstiness shows the distribution of the temporal structure of connectivity Thompson et al. (2017). Therefore, in temporal network analysis, burstiness measure is an important property of many processes (Barabási 2010; Vazquez et al. 2007; Vázquez et al. 2006; Barabasi 2005; Min et al. 2011). Burstiness measure in temporal networks has been used in several applications, including communications (Barabasi 2005; Eckmann et al. 2004; Jo et al. 2012), spreading of diseases Vazquez (2013), epidemics Takaguchi et al. (2013), and neuroscience William and Peter (2016). Burstiness measure is introduced by Goh and Barabási (2008) and defined in Holme and Saramäki (2012) as follows:

$$B_{v,u} = \frac{\sigma(\tau_{v,u}) - \mu(\tau_{v,u})}{\sigma(\tau_{v,u}) + \mu(\tau_{v,u})}$$

where  $\tau_{v,u}$  is a vector of inter-contact times between nodes  $v$  and  $u$  such that  $u \in V \setminus v$  (see Definition 6), and  $\sigma$  and  $\mu$  are, respectively, standard deviation and mean of  $\tau_{v,u}$ . *Burstiness* belongs to the range  $[-1, 1]$ , where  $B_{v,u} = 1$  indicates completely bursty sequence,  $B_{v,u} = 0$  corresponds to a natural sequence, and  $B_{v,u} = -1$  shows a completely periodic sequence. Bursts can be defined either for nodes or for edges. Here, the number of bursts are calculated per edge and can be extended to a nodal measure by summing over the bursty coefficients across all edges for a given node.

Fluctuability is an important measure which provides information about the global properties of a temporal network. Fluctuability measure quantifies the temporal variability of connectivity Thompson et al. (2017) and it is defined as follows ( $\phi$  is defined in Sect. 2):

$$F = \frac{\sum_v \sum_u \phi(v, u)}{\sum_v \sum_u \sum_t \phi(v, u, t)}$$

where  $v$  and  $u$  are two distinct nodes in  $V$ . *Fluctuability Measure* can be at most 1, when every edge is defined in a single timestamp of the time domain.

Volatility Measure is a *global* measure of a temporal network which indicates how many changes, on average, are observed between two consecutive timestamps. Following Thompson et al. (2017), volatility is defined as:

$$Vol = \frac{1}{t_M - 1} \sum_{t=1}^{t_M-1} D(A(t), A(t+1)),$$

where recall that  $A(t)$  is the connectivity matrix of the snapshot  $G$  at time  $t$ ,  $t_M$  is the overall number of timestamps in the time domain, and  $D$  is the Hamming distance function which quantifies the difference between connectivity matrix at times  $t$  and  $t+1$  (notice that in undirected graphs, each edge is considered only once).

Reachability Latency Measure is another global measure of a temporal network. Reachability measures the time needs to reach nodes in a temporal network. Reachability Latency Measure quantifies the optimal distance to reach a fraction  $r$  of the nodes in a network.

Temporal Efficiency Measure is a global measure of a temporal network. Similar to reachability, Efficiency estimates temporal properties based on optimal paths. Following Thompson et al. (2017), to obtain the Temporal Efficiency Measure first, we need to calculate at each timestamp the inverse of the optimal distance for all nodes. Then, the inverse optimal distance are averaged across timestamps. Formally, Temporal Efficiency Measure is defined as

$$H = \frac{1}{t_M(n^2 - n)} \sum_{v,u,t} \frac{1}{\alpha_{v,u}^t}, v \neq u.$$

Notice that the global temporal efficiency measure could be computed at a nodal level as well.

## 4 Fundamental problems in temporal networks

In this section, we present fundamental problems related to temporal graphs that have been considered in the literature. We start with problems related to paths, walks, and connectivity, which have been the focus on many research works, and then, we consider other problems, the identification of dense subgraphs, and graph covering.

### 4.1 Path, walk, and circuit problems

Finding walks or paths is a topic that have been extensively studied in computer science, with several applications in biology (for example finding chain of activation of proteins) or epidemiology (finding a chain of contagions). The topic has been extensively studied from the introduction of temporal networks, since the temporal constraint adds in some cases a level of difficulty in algorithm design.

Given two vertices  $u$  and  $v$ , a *foremost temporal path* (or earliest-arrival time path)  $p$  starting from timestamp  $t$  is a temporal path that (1) starts from  $u$  and arrives in  $v$ ; (2) the starting time of  $p$  is in a timestamp  $t'$  greater or equal than  $t$ ; (3) the arrival time is minimized.

A foremost temporal Path from a starting vertex  $u$  to any other node of a temporal graph can be computed in  $O(nt_M + m)$  time Mertzios et al. (2019). The algorithm for this problem (FTPA) given in Mertzios et al. (2019), similarly to the classic breadth-first search algorithm (Cormen et al. 2009; Dondi et al. 2019) for static graphs, explores the temporal edges in increasing order of timestamps.

Other basic temporal path problems from a node  $u$  to a node  $v$  have been considered in Huanhuan et al. (2014), given two timestamps  $t_1$  and  $t_2$  in  $T$ , with  $t_1 \leq t_2$ :

1. *Latest-departure path*: a temporal path from  $u$  to  $v$  that has the largest departure time after  $t_1$  and that has ending time before  $t_2$
2. *Fastest path*: a temporal path from  $u$  to  $v$  that starts after  $t_1$  and ends before  $t_2$  and that requires the minimum time, defined as the difference between the arrival time and the starting time
3. *Shortest path*: a temporal path from  $u$  to  $v$  that starts after  $t_1$  and ends before  $t_2$  and that consists of the minimum number of traversed edge (or has minimum weight for temporal weighted graphs).

Latest Departure path can be solved in  $O(n + mt_M)$  time Huanhuan et al. (2014), using an algorithm that applies the

same strategy of the FTPA algorithm, but considering temporal edges in reverse order of timestamps.

Fastest path can be solved in  $O(|S|(n + mt_M))$ , where  $S$  is the set of outgoing edges from the starting node having a timestamp between  $t_1$  and  $t_2$  Huanhuan et al. (2014). The Fastest Path can be solved using the FTPA algorithm, iterating the algorithm for each starting time  $t$  in  $S$  and observing that, among the fastest paths from  $u$  to  $v$  that start at time  $t$ , a fastest path reaching node  $v$  is a foremost path. This algorithm can be improved to  $O(\min n|S|, n + mt_M)$  time by pruning unnecessary paths.

A similar approach can be considered for the Shortest path problem, observing that a prefix of a shortest path must be a shortest path to an intermediate node within a given timestamp; notice that the property does not hold if the given timestamp is not considered. Then, by applying a strategy inspired by Dijkstra's Algorithm Cormen et al. (2009) and by pruning some unnecessary paths, shortest path problem can be solved in  $O(n + M \log d_{max})$  time, where  $d_{max}$  is the largest degree of a node in  $G$  Huanhuan et al. (2014).

Another path problem has been introduced in Casteigts et al. (2021), which adds constraints that the path has to respect. In particular, the problem looks for a *restless* temporal path from  $u$  to  $v$ , which is defined as a temporal path from  $u$  to  $v$  if each pair of consecutive edges in the path are traversed within a time  $\Delta$ .

The restless path problem is a decision problem that asks whether there exists a restless path from a node  $u$  to a node  $v$  of a temporal graph. The complexity of the problem has been analysed for many temporal graph classes (Casteigts et al. 2021; Zschoche 2022). The Restless Path problem is NP-complete Casteigts et al. (2021), even if it is restricted cases: (1) for all finite  $\Delta \geq 1$  when the time domain consists of  $\Delta + 1$  timestamps, even when each edge is active in one timestamp; (2) for temporal graphs whose underlying graphs have all but one edge; (3) for temporal graphs whose underlying graphs have degree bounded by six.

The problem has been considered also in the parameterized complexity framework and it is known to be W[1]-hard when parameterized by distance to disjoint paths, for all  $\Delta \geq 1$ , even if each temporal edge is active in at most one time stamp. The Restless Path problem admits a fixed-parameter algorithm when parameterized by a parameter introduced in Casteigts et al. (2021), called timed feedback vertex number. Other complexity results have been given for the problem considering properties of the underlying graph Casteigts et al. (2021).

A variant of the restless path problem introduced in Casteigts et al. (2021) is the Short Restless Path problem, that looks for a restless path of length at most  $k$  between two nodes. The Short Restless Path problem is NP-hard Casteigts et al. (2021), but it admits a fixed-parameter tractable probabilistic algorithm of time complexity

$O(2^{k t_M}(|V| + |E|))$  based on the multilinear monomial detection technique. The algorithm can be derandomized obtaining a deterministic algorithm of time complexity  $2^{O(k)} O(t_M(|V| + |E|))$ . The problem admits also a fixed-parameter randomized algorithm of time complexity  $O^*(4^{k-p_{min}})$  where  $p_{min}$  is the minimum length of a temporal path between nodes  $u$  and  $v$  Zschoche (2022).

A classical (static graph) problem is computing whether a graph is Eulerian, that is, it contains a circuit or a walk where each edge is traversed exactly once. A temporal Eulerian circuit in a temporal graph is a temporal circuit (that is, it respects the constraint that timestamps of consecutive edges are increasing or non-decreasing) and it is an Euler circuit in the associated underlying static graph.

While finding whether a static graph contains an Eulerian circuit is solvable in polynomial time, finding whether a temporal graph contains a temporal Eulerian circuit is NP-complete even when each temporal edge is active in at most three timestamps Bumpus and Meeks (2021). The result is proved by showing an interesting relation of finding an Eulerian circuit to the Travelling Salesman problem on temporal stars.

The problem of finding an Eulerian trail is NP-complete even when the time domain consists of two timestamps Marino and Silva (2021). The result is proven with a complex reduction from the Not All Equal 3-SAT problem.

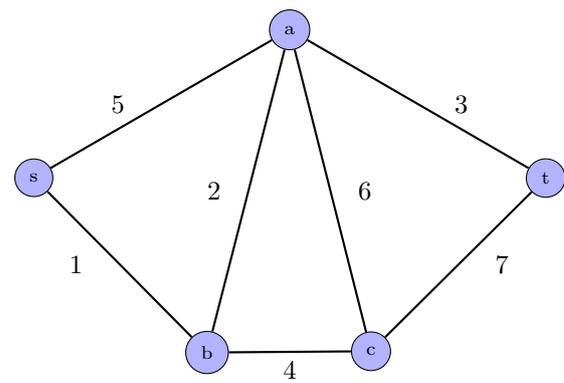
A number of variants of path and walk problems have been considered in the literature. A notable example is the robustness of the network to delays Fuchsle et al. (2022), where the delays may be known in advance or online.

## 4.2 Connectivity and Menger's theorem

Menger's Theorem is a well-known result in graph theory. Given two nodes  $u$  and  $v$  of a static graph, the theorem states that the maximum number of node disjoint paths from  $u$  to  $v$  is equal to the minimum number of nodes that have to be removed to separate  $u$  from  $v$ . Menger's Theorem does not hold for temporal graphs, in particular even for single-labelled temporal graphs, where each edge is associated with a single timestamp (Berman 1996; Kempe et al. 2002) (see the example in Fig. 7).

The results is not only of theoretical relevance, but it has consequences on path computation problem. In particular, computing the maximum number of node disjoint paths from  $u$  to  $v$ , which can be solved in polynomial time on static graphs, is an NP-hard problem for temporal graphs (Kempe et al. 2002; Ibiapina et al. 2022).

Although the Menger's theorem cannot be directly extended to temporal graphs, two modifications of the theorem have been proved:



**Fig. 7** The example give in Kempe et al. (2002) that shows that Menger's theorem does not hold for temporal graphs. There exists only one temporal vertex disjoint path from  $s$  to  $t$ , while the removal of one of the vertices in  $\{a, b, c\}$  does not disconnect  $s$  from  $t$

- For single-labelled temporal graphs, replacing in Menger's Theorem node disjointness with edge disjointness. This leads to the property that the maximum number of edge disjoint temporal paths from  $u$  to  $v$  is equal to the minimum number of temporal edges that has to be removed to separate  $u$  from  $v$  Kempe et al. (2002).
- Replacing in Menger's Theorem node disjointness with departure time disjointness and node removal with time removal. Two temporal paths are *time disjoint* when they leave each node at different timestamps. A node-time-removal consists of removing all the temporal edges incident in a node  $x$  which are active in a given timestamp  $t$  (notice that for an undirected graphs, an edge  $(u, v)$  has to be replaced by two antiparallel arcs and the node-time-removal causes the deletion only of one the two parallel arcs). Mertzios et al. (2019) proved that the maximum number of departure disjoint temporal paths from  $u$  to  $v$  is equal to the minimum number of node-time-removal needed to separate  $u$  from  $v$ .

If, instead of departure time disjointness, we consider temporal node disjointness (two paths/walks cannot traverse a node at the same time), then the maximum number of temporal node disjoint temporal walks from  $u$  to  $v$  is equal to the minimum number of node-time-removal needed to separate  $u$  from  $v$  Mertzios et al. (2019). The same property does not hold if we consider temporal paths instead of temporal walks Ibiapina et al. (2022). Notice that in this case, Menger's Theorem holds only for one temporal path and a node separator of size 1 Ibiapina et al. (2022). In particular, the result in Ibiapina et al. (2022) shows that computing of the maximum number  $k$  of temporal node disjoint paths can be computed in polynomial time when  $k = 2$  and NP-hard for  $k \geq 3$ .

A connectivity problem that has recently been applied to PPI network Jimmy et al. (2019) is a temporal extension of the Steiner Tree problem. Given a weighted temporal graph and a set of  $k$  demands, where each demand is a protein pair or a set of proteins, the problem looks for a subgraph of minimum weight, such that the nodes of each demand are connected in some timestamp. Different versions of the problem have been considered, varying the definition of demands, that can be connections between pairs of nodes or between all pairs of a given set. The variants of the problems are not only NP-hard, but also NP-hard to approximate within a nontrivial approximation factor (that is solving in each timestamp independently) Jimmy et al. (2019). On the other hand, when the input graph grows monotonically on the time domain ( $E(i), 1 \leq i \leq t_M - 1$ , is contained in  $E(i + 1)$ ), a better approximation factor can be achieved. Moreover, several variants of the problem can be solved with Integer Linear Programming, an approach that is validated also on real PPI networks Jimmy et al. (2019).

Other connectivity properties of temporal graphs have been considered in the literature, including connected components of a temporal graphs and Edmond’s Theorem (the maximum number of arborescences in a graph is equal to the minimum cut).

### 4.3 Dense subgraphs in temporal networks

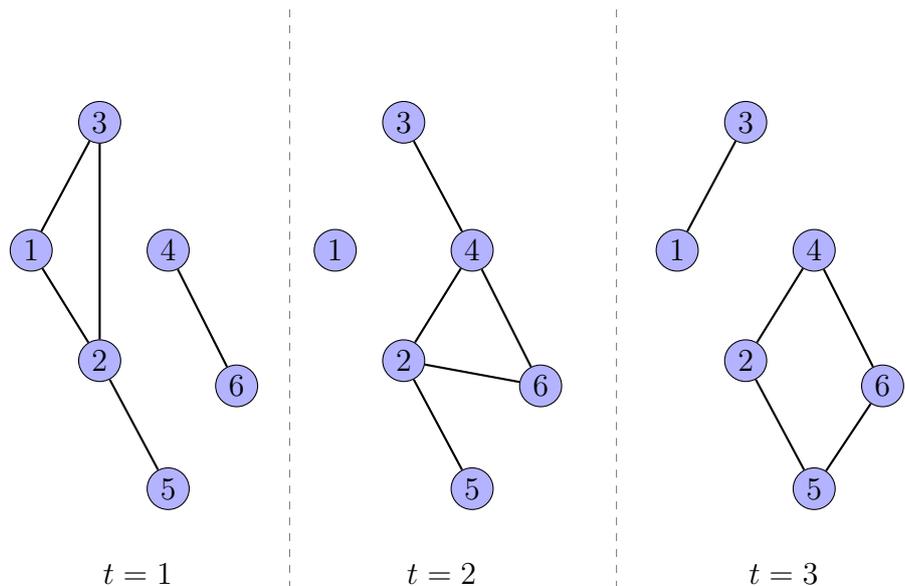
Detection of cohesive subgraphs, such as communities, is one of the fundamental problems in network mining. The community detection problem has been used in different biological analysis, such as gene expression and drug interaction analysis (Fratkin et al. 2006; Saha et al. 2010). Moreover, many different definitions of cohesive subgraphs

have been considered in the literature, such as cliques and  $s$ -plexes. One of the most applied definition of cohesivity is based on density (ratio between number of edges and number of nodes). Given an input graph, the dense subgraphs problem looks for a subgraph of maximum density. The densest subgraph problem is solvable in polynomial time using Goldberg’s algorithm Goldberg (1984) and can be approximated in linear time with Charikar greedy algorithm, which has a  $\frac{1}{2}$ -approximate factor (Asahiro et al. 2000; Charikar 2000). The densest subgraph problem has been recently extended to find a set of densest subgraphs, for example the K-Densest Subgraphs approach (Galbrun et al. 2016; Dondi et al. 2021b), which asks for a collection of  $k$  densest, distinct subgraphs.

Dense subgraph have been studied also for temporal networks (Coscia et al. 2011; Fortunato 2010; Rozenstein et al. 2020; Riccardo and Mehdi 2021a; Zhang et al. 2022). In particular, a dense subgraph problem in a temporal network, called k-Densest-Episodes problem, has been introduced in Rozenstein and Gionis (2019); Rozenstein et al. (2020). The k-Densest-Episodes problem considers a temporal graph  $G = (V, E, T)$ , with a positive integer  $k \geq 1$ , and asks for a set  $S$  of  $k$  temporal subgraphs  $S = \{G[T_1, V_1], \dots, G[T_k, V_k]\}$ , which belong to disjoint time intervals  $(T_1, \dots, T_k)$  (see the example in Fig. 8, for  $k = 1$ ). Notice that two sets  $V_i, V_j, 1 \leq i, j \leq k$  and  $i \neq j$ , may be overlapping. The objective function of the problem aims to maximize the sum of the densities of the  $k$  subgraphs in  $S$ .

The problem is solvable with polynomial-time algorithm (Rozenstein and Gionis 2019; Rozenstein et al. 2020). Rozenstein and Gionis (2019) combines (1) a dynamic programming to find a segmentation into not overlapping time intervals, and (2) Goldberg’s algorithm to compute a densest

**Fig. 8** An example of a temporal graph with a densest subgraph induced by nodes  $\{1, 2, 3, 4, 6\}$  of density  $\frac{7}{5}$  in interval  $[1, 2]$



subgraph in the graph defined in each interval. Due to time complexity of the dynamic programming and Goldberg's algorithm, the algorithm is not suitable for large networks. Hence, the work in Rozenshtein et al. (2020) introduced a heuristic, called KGAPPROX, based on approximate dynamic programming (ApproxD) and an approximate algorithm for the densest subgraph problem (ApprDens). The KGAPPROX has a running time of  $O(\frac{k^2}{\epsilon_1 \epsilon_2} |t_M| m \log^2 n)$ , where  $\epsilon_1$  and  $\epsilon_2$  are, respectively, approximation parameters of ApproxD and ApprDens. Moreover, the work in Rozenshtein et al. (2020) proposed a post-processing step which possibly shrinks intervals of the solution when the density is not decreased. The authors of Rozenshtein et al. (2020) introduced another problem called k-Densest-Episodes-EC with the goal of obtaining larger covering of nodes. The objective function of k-Densest-Episodes problem includes the sum of densities and a function of the fraction of covered nodes. Unlike the k-Densest-Episodes problem, the k-Densest-Episodes-EC is NP-hard Rozenshtein et al. (2020).

Another attempt to solve k-Densest-Episodes problem is given by Riccardo and Mehdi (2021a), where a fast heuristic called Local-search Temporal Densest Subgraphs (LSTDS) is presented. LSTDS start with an initial segmentation computed considering the distribution of active edges in time domain. Then, a local search approach is applied to improve the density of the solution by modifying the intervals of the segmentation. The work in Castelli et al. (2020) introduced a different approach, called Temporal Densest Genetic Algorithm (TDGA), for the k-Densest-Episodes problem. TDGA considers candidate solutions obtained by changing the endpoints of intervals and by applying the Charikar's greedy algorithm to compute a dense subgraph in each interval. In Dondi et al. (2022), it is proposed a different approach that, given a solution of k-Densest-Episodes, aims at identifying, for each subgraph  $G_i$ ,  $1 \leq i \leq k$ , a new subgraph  $G'_i$  with density close to that of  $G_i$  and defined over a subinterval of  $I_i$ .

#### 4.4 Graph covering

Covering a static graph with a minimum cardinality set of nodes is a well-known and fundamental problem, called Node Cover or Vertex Cover, in computer science. Vertex Cover has been considered by a number of papers, due to its relevance in practice and due to its importance in graph theory, theoretical computer science, and algorithm design. Recently, this problem has been considered for temporal graphs, where different formulations have been proposed.

We recall that, in a static graph  $H = (U, A)$ , a set  $C$  of nodes covers  $H$  if, for each edge  $(u, v) \in A$ ,  $u \in C$  or  $v \in C$ .

The first temporal variant of Vertex Cover has been introduced in Akrida et al. (2020). Given a temporal graph  $G = (V, E, T)$ , a (non-temporal edge)  $e = (u, v)$  is temporally

covered by a pair  $(a, t)$ , with  $a \in V$  and  $t \in T$ , if  $a$  is one of the endpoints of  $e$  (that is  $a = u$  or  $a = v$ ) and  $e$  is active in timestamp  $t$ . In a temporal graph  $G = (V, E, T)$ , a temporal vertex cover is a subset  $C \subseteq V \times T$ , such that edge  $(u, v)$  is temporally covered by some pairs  $(a, t) \in C$ . The Temporal Vertex Cover problem asks for a temporal vertex cover of minimum cardinality.

The sliding window temporal vertex cover problem proposed in Akrida et al. (2020) is a variant that imposes the covering of each edge in intervals (called sliding windows) where the edge is active. The Sliding Window Temporal Vertex Cover problem asks for a subset  $C \subseteq V \times T$ , such that each edge  $(u, v)$  active in a sliding window is covered by a pair  $(a, t)$ , where  $t$  is a timestamp of the sliding window and  $a$  is equal to  $u$  or  $v$ .

Temporal vertex cover is known to be NP-hard and also hard to approximate with factor  $(1 - \epsilon)$  unless NP has a  $n^{O(\log \log n)}$ -time deterministic algorithm Akrida et al. (2020). On the positive side, the problem can be approximated via a reduction to the Hitting Set problem, achieving an approximation factor of  $H_{n-1} - \frac{1}{2}$ , where  $H_{n-1}$ , where  $H_n = \sum_{i=1}^n \frac{1}{i} \approx \ln n$  Akrida et al. (2020).

Sliding Window Temporal Vertex Cover is also NP-hard Akrida et al. (2020) and it can be solved via dynamic programming in  $O(t_M \Delta (n + m) \cdot 2^{n\Delta})$  time Akrida et al. (2020), where  $\Delta$  is the size of the sliding window. Furthermore, even for  $\Delta = 2$ , with maximum degree of the underlying graph  $G$  at most 3, every connected component of every snapshot containing at most seven vertices, the problem is APX-hard Akrida et al. (2020).

Sliding window temporal vertex cover admits approximation algorithms, with respect to various parameters. More precisely, it admits approximation factors: (1)  $\ln n + \ln \Delta + \frac{1}{2}$  Akrida et al. (2020; p. 2)  $2k$ , where  $k$  is the maximum number of times that each edge can appear in a sliding window Akrida et al. (2020; p. 3)  $d - 1$ , with  $d$  the maximum vertex degree at every snapshot Hamm et al. (2022). Furthermore, complexity lower bounds for the two problems have been proved in Akrida et al. (2020).

The problems have been considered also in sparse graphs Hamm et al. (2022). More precisely, Sliding Window Temporal Vertex Cover is NP-hard even when the underlying graph is a cycle or a path, for every  $\Delta \geq 2$ , while Temporal Vertex Cover can be solved in polynomial time in these cases Hamm et al. (2022). Sliding Window Temporal Vertex Cover admits a PTAS on temporal paths and cycles Hamm et al. (2022). Different exponential algorithms have been considered for Sliding Window Temporal Vertex Cover, in particular a fixed-parameter algorithm where the parameter is the size of a temporal vertex cover Akrida et al. (2020).

A different formulation of temporal graph cover has been proposed in Rozenshtein et al. (2021) for summarizing event timelines in temporal networks. The problem aims

at defining the activity interval of each node of the graph, so that, for each temporal edge  $(u, v, t)$ , at least one of  $u$  or  $v$  is active in an interval that includes  $t$ . In this case, we say that the defined activity intervals cover  $E$ .

Four variants of this problem have been proposed in Rozenstein et al. (2021) and they differ for the interval activity definition (it can be a single interval or a sequence of more non overlapping intervals) and the objective function (the minimization of the sum of node activities or the minimization of the maximum activity of a node). The length of an activity interval  $[t_1, t_2]$  is  $t_2 - t_1$ .

In the first variant, called *MinTimeLineCover*, given a temporal graph  $G = (V, E, T)$ , the goal is to find activity intervals for the nodes in  $V$ , so that they cover  $E$  and the sum of node activity lengths is minimized. A second variant, called *MinMaxTimeLine*, given a temporal graph  $G = (V, E, T)$ , aims at finding activity intervals for the nodes in  $V$ , so that they cover  $E$  and the objective function is the minimization of the maximum length of an activity interval. In Fig. 9, we give an example of covering a temporal graph. Notice that the solution represented in Fig. 9 is a solution both of *MinTimeLineCover* and *MinMaxTimeLine*. The objective function of *MinTimeLineCover* has value 2 (two nodes have an activity interval of length 1), while the objective function of *MinMaxTimeLine* has value 1 (the maximum length of an activity interval is 1).

*MinTimeLineCoverK*, given a temporal graph  $G = (V, E, T)$  and an integer  $K \geq 2$ , asks for the definition of activity intervals, at most  $K$  for each node in  $V$ , so that they cover  $E$  and the sum of node activity interval lengths is minimized. *MinMaxTimeLineK*, given a temporal graph  $G = (V, E, T)$  and an integer  $K \geq 2$ , asks for the definition

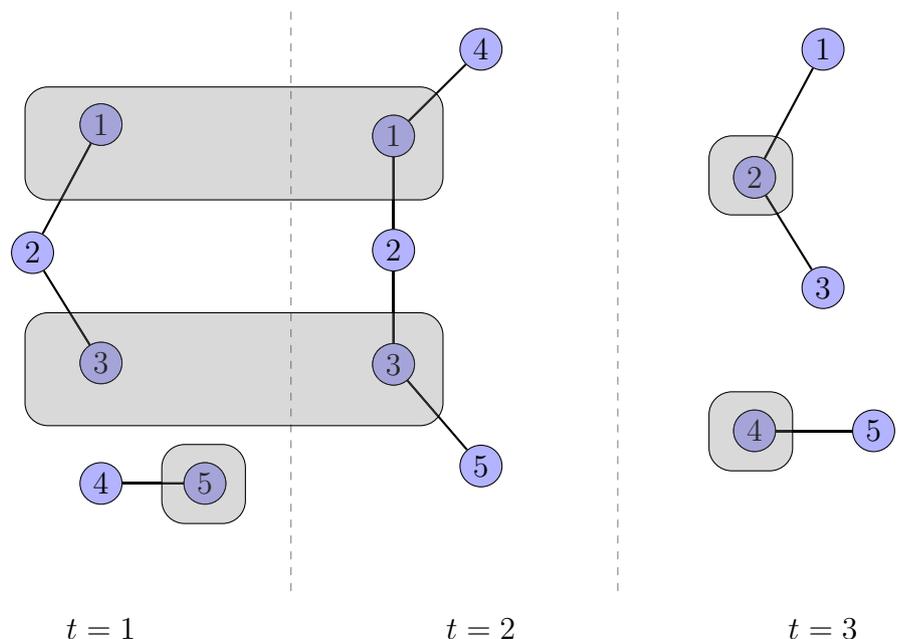
of activity intervals, at most  $K$  for each node in  $V$ , so that they cover  $E$  and the maximum span of a node (defined as the sum, for a node, of the lengths of its activity intervals) is minimized. Notice that some of the intervals may be empty, when they are not needed to solve the problems.

*MinTimeLineCover*, *MinTimeLineCoverK*, and *MinMaxTimeLineK* are all NP-hard Rozenstein et al. (2021), even if the time domain consists of exactly two timestamps Froese et al. (2022) (for one timestamp, all the problem variants are trivial, since all the solutions have a cost equal to 0). Surprisingly, *MinMaxTimeLine* can be solved by a polynomial time algorithm, more precisely in  $O(n \log n)$  via binary search and via a polynomial reduction to the 2-SAT problem Rozenstein et al. (2021). Moreover, *MinTimeLineCoverK* and *MinMaxTimeLineK* cannot be approximated within any factor, unless  $P = NP$ , since deciding if these two problems admits a solution of cost 0 is an NP-complete problem Rozenstein et al. (2021).

The computational and parameterized complexity of *MinTimeLineCover*, *MinTimeLineCoverK* and *MinMaxTimeLineK* has been deeply analysed (Froese et al. 2022; Dondi 2022). *MinTimeLineCover* is NP-hard even in very restricted cases Dondi (2022): (1) when at most one temporal edge is defined in each timestamp; (2) when each vertex is incident in at most two temporal edges, for each timestamp, and the time domain is defined over three timestamps.

Fixed-parameter algorithms have been defined for the problem *MinTimeLineCover*. When the time domain consists of two timestamps, *MinTimeLineCover* is fixed-parameter tractable for parameter cost of the solution, via a parameterized reduction to the Min 2-SAT problem Froese et al. (2022). We recall that the Min 2-SAT problem, given

**Fig. 9** An example of a cover of a temporal graph, as defined by *MinTimeLineCover* and *MinMaxTimeLine*, where the grey areas represent the interval activities of nodes. Nodes 1 and 3 have an activity interval of length of 1, since they are active in timestamp 1 and 2, and the remaining vertices have an activity interval of length 0, since they are active in a single timestamp



a Boolean formula in conjunctive normal form, asks for the removal of the minimum number of clauses that lead to a satisfiable formula. A fixed-parameter tractable algorithm for the problem is given for parameters (1) the maximum number of nodes with temporal edges in a timestamp and (2) the maximum length of an interval where a node has incident temporal edges. The algorithm is based on a dynamic programming approach that considers, for each timestamp  $t$ , the possible activity intervals of nodes having active edges in timestamp  $t$ .

MinTimeLineCoverK and MinMaxTimeLineK, for  $K \geq 2$ , are NP-hard if we are seeking for a solution of cost 0, even for temporal graphs containing three identical layers Froese et al. (2022). For cost equal to 0, MinTimeLineCoverK and MinMaxTimeLineK are both fixed-parameter tractable when parameterized by  $n$  using Integer Linear Programming Froese et al. (2022). MinMaxTimeLineK is  $W[1]$ -hard when parameterized by  $n$  for cost equal to 1, while MinTimeLineCoverK parameterized by  $n$  plus the cost of a solution admits a fixed-parameter algorithm based on dynamic programming Froese et al. (2022). This latter result is obtained by splitting the activity of nodes into two sets, having cost 0 (solvable for the case cost equal to 0) and having cost at least 1. Interestingly, it is shown that this latter set induces an interval graph, and the number of possible interval graphs is upper bounded by a function of  $n$  and the cost of a solution. Finally, MinTimeLineCoverK and MinMaxTimeLineK are fixed-parameter tractable when parameterized by  $n + K$ . For MinTimeLineCoverK, the result is achieved by dynamic programming, while for MinMaxTimeLineK using a bounded search tree algorithm Froese et al. (2022).

Efficient heuristics based on solving restricted variants of MinTimeLineCoverK and MinMaxTimeLineK have been designed and evaluated experimentally on synthetic and real datasets (extracted from Twitter) Rozenshtein et al. (2021).

## 5 Searching motifs in temporal networks

Detecting interesting subgraphs with a given property or with a statistically significance is a fundamental problem in graph mining. Here, we consider some approaches that have been considered in temporal graphs.

### 5.1 Graph motifs

A graph (or a network) motif is defined as a subgraph of limited size that appears with a statistically significance in the input graph. Finding motifs is fundamental to understand network structural properties. Identifying triangle (cliques of size three) is used for example to compute clustering coefficients.

Different definitions of temporal motifs have been introduced in the literature, based on different temporal properties:

1. A first definition defines a motif as an ordered set of temporal edges, so that the static graph induced by the edges is connected and (a) two consecutive (defined by the order) temporal edges occur at a time bounded by parameter  $\Delta_C$ ; (b) the temporal edges incident on a node of the motif must be consecutive temporal edges in the motif Kovanen et al. (2011)
2. A second approach Song et al. (2014) defines a motif as an ordered set of temporal edges, so that the first and the last temporal edge of the motif are at a temporal distance at most  $\Delta_W$
3. A third definition Hulovatyy et al. (2015) relaxes property (b) of the first definition and asks only for induced subgraphs (that is, all the temporal edges of nodes connecting nodes of the subgraph have to be considered in the motif)
4. A fourth definition Paranjape et al. (2017) relaxes property (b) of the first definition and asks only for induced subgraphs (as the third definition) and adds the condition that the first and the last temporal edge of the motif are at a temporal distance at most  $\Delta_W$ .

One of the most applied definition, proposed in Liu et al. (2021) and Grasso et al. (2021), states that a temporal motif is a connected subgraph whose edges' temporal edges  $e_1 = (v_1, v_2, t_1), \dots, e_z = (v_z, v_{z+1}, t_z)$  can be ordered, so that  $t_i < t_{i+1} \leq t_i + 1$  (or for  $t_i \leq t_{i+1}$  simultaneous events).

An occurrence of a temporal motif  $M$  in a temporal graph  $T$  is a subgraph  $H$  of  $T$ , such that there exists a mapping  $f$  between the nodes of  $M$  and the nodes of  $H$  which is bijective and such that (1) if there exists a temporal edge between two nodes  $u, v$  in  $M$ , in some timestamp  $t$  of  $T$ , then there exists a temporal edge between two nodes  $u, v$  in  $H$ , in some timestamp  $t'$  of  $T$ ; (2) the chronological order between edges of the motif is respected in the occurrence of the motif; (3) the temporal edges in  $H$  are active in a time interval of length  $\Delta$ .

### 5.2 Colored motif

A well-known problem in computational biology and, more generally, in computer science, is the quest for colored motifs inside a network (Lacroix et al. 2006; Bruckner et al. 2010; Betzler et al. 2011). The problem has application in metabolic networks and PPI networks (Lacroix et al. 2006; Bruckner et al. 2010), where the goal is to identify functional motifs. Given a multi-set of colors, called motif, representing functionalities, the goal is to identify a connected component in a graph whose vertices match the motif. The

problem has been recently considered also for temporal networks, mainly adding the constraint that the connected component should be a temporal path Thejaswi et al. (2020). Several variants of the problem have been considered, showing that they are NP-hard (except for the case when a specific ordering on the colors is defined in the motif), but admit fixed-parameter algorithms based on multilinear monomial detection technique Thejaswi et al. (2020). The approximation complexity of an optimization variant of the problem that asks for the maximum number of colors included in a solution (so the subgraph may contain only some colors of the motif) has also been considered, showing that it cannot be approximated within factor  $O(n^{\frac{1}{2}-\epsilon})$ , for any  $\epsilon > 0$ , unless  $P = NP$  Riccardo and Mehdi (2021b). For the same variant of the problem, a heuristic based on local search has been designed Riccardo and Mehdi (2021b).

### 5.3 Temporal network comparison

Network comparison aims at the identification of correspondence among two or more networks. For static networks, the alignment can be defined as follows Guzzi and Milenković (2018). Given two input networks  $G_1$  and  $G_2$ , the problem of finding an alignment between the two networks may be translated into the search for a mapping between nodes of  $G_1$  and nodes of  $G_2$ , with the aim of the maximization of a quality score. The computational hardness of this problem arises from the NP-completeness of the underlying subgraph isomorphism problem Cannataro et al. (2010). When considering temporal networks with  $k$  timestamps, the problem becomes more complicated, since we have to deal with  $k$  temporal subgraphs.

The subgraph isomorphism problem in this context becomes the Temporal Subgraph Isomorphism problem Liu et al. (2021) (TSI). From a practical point of view, TSI is the search from meaningful time-ordered patterns in a sequence of timestamps. These patterns can be expressed as the search of correspondences of a query graph  $G_q$  in a temporal graph  $G_t$ .

As evidenced in Redmond and Cunningham (2016, 2013), there exist three approaches for solving the problem. A first approach (*extract and test*) is based on the extensive extraction of all the subgraphs with some specific temporal properties (as specified in the query), and then performing isomorphism test on each subgraph. A second approach (*test and extract*) first performs subgraph isomorphisms and then filters results discarding those that do not have the temporal properties. A third hybrid approach leverages temporal information during the isomorphism test.

More recently, Liang et al. (2021) proposed a framework that transforms a dynamic network into a set of snapshots. This approach is a trade-off between the approaches that do not consider the dynamic evolution of the networks and the

previous algorithms that focus on the separate evolution of two networks. In this approach, the dynamic of the two considered network is considered as two separate sequence of snapshot. Then, the authors propose a framework for aligning each pair of corresponding snapshots. An extension of the previous approach is applied to multirelational temporal network in Chen and Liang (2022).

An approach to the alignment of temporal networks has been proposed in Elhesha et al. (2019). Given two temporal graphs, the goal is the definition of an alignment function that maps all the nodes of the first graph in a subset of the nodes of the second graph, such that it maximizes an objective function that includes (1) A similarity between the corresponding pair of nodes minus (2) the sum number of connected components in each snapshot.

### 5.4 Representation learning for temporal networks

Network representation learning (or network embedding) Guzzi and Zitnik (2022) is a common way for applying machine learning on graph data, to get benefit from both. Classical embedding algorithms focused on static networks and they transformed a network into a low-dimensional vector space to analyse the network itself (Nelson et al. 2019; Shawn et al. 2022; Guzzi et al. 2022; Veltri et al. 2007). Currently, there exist many algorithms and many classification attempts that are categorised and described in some previous surveys (Cui et al. 2018; Chang et al. 2020; Nelson et al. 2019; Goyal and Ferrara 2018; Hamilton et al. 2017; Cannataro et al. 2013).

These algorithms receive as input an unweighted graph  $G = (V, E)$  represented by its associated adjacency matrix  $A$  and a real-valued matrix  $X$  containing node attributes  $X \in \mathbb{R}^{n \times |V|}$ . The goal of each algorithm is to map each node into a vector  $z \in \mathbb{R}^d$  where  $d < |V|$ .

Some recent surveys (Cui et al. 2018; Chang et al. 2020; Nelson et al. 2019; Goyal and Ferrara 2018; Hamilton et al. 2017; Cannataro et al. 2013) present an overview of existing methods that may be categorised in shallow embedding methods and graph neural methods. Algorithms belonging to the first class encode each node ( $v_i \in G$ ) into a single vector through the use of a simple encoding function defined as

$$ENC(v_i) = Mv_i, \quad (3)$$

where  $M$  is a matrix containing the embedding vectors and  $v_i$  is a vector used for selecting the column. Algorithms belonging to the second class learn a graph neural network for representing nodes and presents the main advantage of being inductive, i.e., they can easily learn the representation of a novel node without processing the whole graph Guzzi et al. (2022).

More recently, the rising of temporal network has generated the need for the introduction of novel embedding methods for temporal networks.

**Definition 7** Given a temporal network graph  $G = (V, E, T)$ , the task of time-varying network embedding is to learn a mapping function  $f : V \rightarrow K \subseteq \mathbb{R}^d, d \ll \|V\|$  which maps each node of the graph into a low-dimensional space by preserving both node proximity and temporal constraints. Two nodes are close in the vector space if they are topologically close and there exists a temporal proximity between them.

In Mohan and Pramod (2022), authors designed a random surfing model based on an adaption of Markov chains to represent node similarity. Then, a deep auto-encoder is used to perform non-linear dimensionality reduction. Generated embeddings are then evaluated using a temporal link prediction benchmark.

The work by Goyal et al. (2020) has a similar goal. Given a temporal graph represented by the temporal snapshot, a method for learning a representation of nodes at each time step is presented. Then, the obtained representation is used for predicting future links among nodes. The algorithm called *dyngraph2vec* is based on multiple multilayered non-linear networks to analyse each network. Then, recurrent networks are used to learn the structure of the temporal transitions.

## 6 Software tools for temporal networks

As we introduced, temporal network analysis is currently an hot topic in research. Consequently, as we presented in this survey, many different problems and algorithms to solve them have been introduced. Here, we first present software tools used for specific problems for temporal networks, which are implementations of different algorithms proposed to solve the problems that have been discussed in the previous sections. Moreover, more recently, even

comprehensive libraries are available. In Table 2, we presented some softwares and their functions. The last row of the table contains the link of a page of the GitHub portal showing the whole set of softwares of GitHub tagged as `temporal networks`.

### 6.1 Software tools for specific problems

Table 1 shows software tools that have been used for specific temporal network problems.

In particular, Elhessa et al. (2019) implemented tempo algorithm for temporal network alignment, Aparício et al. (2019); Nassa et al. (2011) provided user-friendly interface and implementation source for GoT-WAVE algorithm for temporal global pairwise network alignment problem, Rozenshtein et al. (2021) implemented algorithms for vertex cover problems (k-Inner, k-Budget, and k-Baseline), Grasso et al. (2021) implemented MODIT as a method for counting motifs (of any size) in a temporal network for motif discovery problem, Thejaswi et al. (2020); Chow et al. (2021); Ren et al. (2021) implemented variants of *node colored motif discovery*, constrained to be paths, in temporal graphs (Temporal-patterns-mk2), Rozenshtein et al. (2020) applied KGAPPROX algorithm for k-Densest-Episode problem, and Riccardo and Mehdi (2021b) implemented CTPLS algorithm for maximum colorful path in a temporal graph (Max CPTG) problem.

### 6.2 General purpose software tools

Here, we present some general purpose software tools that address general problems in network analysis. In Table 2, we presented some softwares and their functions. The last row of the table contains the link to a page of the GitHub portal showing the whole set of softwares of GitHub tagged as `temporal networks`.

**Table 1** Software tools used for specific problems for temporal networks

Problem	Language	URL
Alignment problem	C++	<a href="https://www.cise.ufl.edu/rehessa/temporal.zip">https://www.cise.ufl.edu/rehessa/temporal.zip</a> – url not available
Global pairwise network alignment	C++/Python	<a href="https://www.dcc.fc.up.pt/got-wave/">https://www.dcc.fc.up.pt/got-wave/</a>
Vertex Cover problems	Python	<a href="https://github.com/polinapolina/the-network-untangling-problem">https://github.com/polinapolina/the-network-untangling-problem</a>
Motif discovery problem	Java	<a href="https://github.com/RobertoGrasso96/MODIT">https://github.com/RobertoGrasso96/MODIT</a>
Node colored motif discovery problem	C	<a href="https://github.com/suhastheju/temporal-patterns-mk2">https://github.com/suhastheju/temporal-patterns-mk2</a>
k-Densest-Episode problem	Python	<a href="https://github.com/polinapolina/segmentation-meets-densest-subgraph">https://github.com/polinapolina/segmentation-meets-densest-subgraph</a>
Maximum Colorful Path in a Temporal Graph problem	Python	<a href="https://github.com/mehdihosseinzadeh/ColorPath_TG">https://github.com/mehdihosseinzadeh/ColorPath_TG</a>
Data Structures	Python	<a href="https://github.com/hilsabeckt/hybridtempstruct">https://github.com/hilsabeckt/hybridtempstruct</a>

**Table 2** A synopsis of some selected softwares.

Software	Language	Analysis	Visualization
Temporal network analysis <sup>a</sup>	Python/C++	IO-S-P	Y
Teneto - Temporal centrality network Measure <sup>b</sup>	Python	IO-S	Y
TGLib <sup>c</sup>	R	IO-S-P	N
NetworkX Hagberg et al. (2008) <sup>d</sup>	Python	IO-NS-CA-C-ME-EM	Y
TimeNexus <sup>e</sup>	Cytoscape App	IO-PA	Y
Py3Plex Skrlj et al. (2019) <sup>f</sup>	Python	IO-NS-CA-C-PA	Y
General collection of packages on GitHub <sup>g</sup>	Python	N	N

In Visualization column: Y and N, respectively, indicate if the software supports visualization

In analysis column: *IO* Network Input Output, *S* Network Statistics, *P* PathAnalysis, *C* Clustering, *ME* Module Extraction, *EM* Network Embedding.

<sup>a</sup><https://gitlab.com/tgpublic/tglib>

<sup>b</sup><https://github.com/wiheto/teneto>

<sup>c</sup><https://programminghistorian.org/en/lessons/temporal-network-analysis-with-r>

<sup>d</sup><https://networkx.org>

<sup>e</sup><https://apps.cytoscape.org/apps/timenexus>

<sup>f</sup><https://github.com/SkBlaz/py3plex>

<sup>g</sup><https://github.com/topics/temporal-networks>

## 7 Applications of temporal networks in biology and medicine

Temporal networks have been applied to different biological and medical problems. We will focus on four main applications: analysis of brain connectome, protein–protein interaction networks, and epidemics modelling applications to single cell technologies.

### 7.1 Brain connectome

Several studies have analysed the human brain using temporal graph theory, where neural units are nodes extracted, for example, from MRI (Magnetic Resonance Imaging) data, and interactions between nodes change over time Thompson et al. (2017). An edge between two nodes is defined when there is an anatomical connectivity or a synchronized functional activity is observed. The identification of neural units that are structurally connected in clusters or modules led to better understand neurophysiological processes.

Structural and functional connectome properties have been studied in several work that applies the temporal network framework. Node centrality measures provide information on how the relevance of nodes inside the brain network. In Thompson et al. (2017), centrality measure has been applied to analyse which nodes and correspondent brain regions are central in the brain network dynamics, using temporal degree centrality and closeness centrality, for example illustrating the relevance of visual region, frontoparietal attention, dorsal attention, and default mode.

The contribution given in Thompson et al. (2017) includes also insights into the burstiness of inter-contact

time and its relation with specific tasks, the application of fluctuability and volatility, and reachability latency (related to a given task).

Another application of temporal centrality measures in connectome is due to van den Heuvel et al. (2010), where it is explored the communication between different regions of the brain. In particular, van den Heuvel et al. (2010) showed that schizophrenia impacts global network connectivity of frontal and temporal brain regions. They found significant reduction of betweenness centrality of frontal regions hubs of patients showing a less central hub role of these regions in the brain network. A similar approach has been proposed in Fransson and Thompson (2020), where betweenness is applied to identify hubs in brain networks.

Connectivity is a fundamental property of networks and in particular of brain network. The functional connectivity of brain network and the learning of brain network structure have been considered in Kim et al. (2021); Kong et al. (2021).

### 7.2 Protein–protein interaction networks

Protein–Protein Interaction Networks (PPIN) are undirected graphs, whose nodes represent proteins, more precisely a gene coding protein; edges represent interaction between two proteins, mainly co-expressed proteins. Since proteins can become active or inactive, their interactions usually change over time, and thus, a natural extension to PPIN is to consider a temporal representation Przytycka et al. (2010), that is a temporal graph.

A well-known problem that has been considered for PPI networks is the identification of paths representing processes

that, starting with the activation of a protein, leads to the activation of another protein (Cannataro et al. 2010; Jimmy et al. 2019). These paths are associated with the maximization of an objective function which is the probability of the chain of interactions represented by the path. An extension of this problem looks for a collection of paths associated with chains of activation interactions, instead of a single path Jimmy et al. (2019).

A similar problem for PPI networks is the inference of the way receptor signaling propagates and how it affects transcription factors Khodaverdian and Yosef (2022). This problem is modeled as finding a walk in a temporal graph (where vertices can be present only in some timestamp, while edges are present only when both endpoints are present) from a source node to a target node that starts in the first timestamp and ends in the last. Furthermore, the walk must satisfy a constraint that one of its edges connects two nodes in the same snapshot or in consecutive snapshots (timestamps  $i$  and  $i + 1$ ). The problem of finding a shortest walk can be solved in polynomial time when the time domain consists of two timestamps Khodaverdian and Yosef (2022), while in general is NP-hard and cannot be approximated within factor  $\frac{|t_M|}{2}$ , when there at least  $t_M \geq 3$  timestamps Khodaverdian and Yosef (2022).

Another relevant problem of PPIN is the identification of groups of co-expressed protein as they represent sets of protein related to a same functionality (Han et al. 2004; Cannataro et al. 2010; Ou-Yang et al. 2014). This may lead to clustering problems (partition of networks into groups of cohesive subgraphs) or to the identification of the top  $k$  cohesive subgraphs (possibly overlapping and with some uncovered nodes). The identification of relevant groups of protein may lead also to subgraph query, that is, the identification of whether a subgraph belongs to a larger graph. In some cases, graphs can be colored, where colors represent functionalities and the goal is to look if there is a group of elements that represent a given multi-set of functionalities.

The identification of protein complexes (sets of proteins related to some biological functions) with temporal networks is a problem addressed in Li et al. (2020). First, a temporal network is built by taking into account spatial information and gene expression data. Then, a clustering algorithm (Markov Clustering) is applied on the temporal network constructed to identify protein complexes.

Temporal centrality measures have been applied also for the analysis of protein–protein interaction networks. In Meng et al. (2022), hubs, which are fundamental to analyse protein–protein interaction networks, are identified by degree centrality.

Analysing of the yeast protein–protein interaction network regulated both in time and in space is reported in Han et al. (2004). In particular, Han et al. (2004) considers a significant proportion of proteins as ‘hubs’, in which hubs

are classified into two types based on expression with their partners on same/different time or space.

The work in Lebre et al. (2010) addresses the time-varying networks of gene regulation in systems biology. Given time course gene expression data, authors analyse gene by gene and find the topology of regulatory network and the changing over time.

### 7.3 Temporal graphs for epidemics modelling

Spreading of diseases may be trivially modelled using a temporal network. Nodes of the network are the individuals, while edges their contacts. A set of node labels is used to model the status (i.e., Healthy, Suscepted, Infected, Recovered) (Masuda et al. 2021; Guzzi et al. 2022), while the temporal evolution of the contacts is represented by the temporal snapshot of the network itself. The rationale behind this representation is that the structure (and in particular its dynamic counterpart) of these networks may model and predict the epidemic propagation Guzzi et al. (2022). Temporal networks have been introduced some years ago in the literature, and their use is increased during the COVID-19 pandemic. Common applications of temporal networks cover a broad range, from epidemic modelling, to evaluation of measures for epidemic controlling.

The use of networks in epidemiology help researchers to describe the dynamics of spreading focusing on interactions between individuals or groups, where the spreading process is driven by contacts in the network Balcan et al. (2009). Main limitation of the use of network is the requiring of mobility and contact data to feed network models.

Humphries et al. (2021) introduced a framework for modelling and analysing spreading. The framework enables the analysis of spreading considering both an individual and a pair-based perspective of modelling. For the contagion process, they consider the susceptible-infected-recovered (SIR) model, built on top of a network with time-varying edges. Petrizzelli et al. presented the benefits of a topology-aware versus an age-based vaccination strategy to mitigate the spreading of the virus Petrizzelli et al. (2022).

A fundamental problem of the SIR model is the identification of chain of interactions via the identification of restless paths in temporal graphs (Casteigts et al. 2021; Zschoche 2022) (see Sect. 4). In this approach, nodes represent individuals and a contact between two individuals at time  $t$  is represented with a temporal edge at timestamp  $t$  between the corresponding nodes. In this case, a chain of infections involving different individuals is a temporal path. To represent the time when an individual can spread the infection (hence, it is not recovered), a time constrained is added to the path, that is, a contact must be observed in a time window.

The importance of temporal networks for epidemics has been considered in Enright and Kao (2018). When considering the spreading process, the concept of temporal path is useful to define nodes that can be reached by the infection. Consider a time interval. A network/graph is temporally strongly connected when each pair of nodes is connected by a temporal path in the considered interval. A network is temporally connected if, for every pair of nodes, at least one is reachable by a temporal path starting from the other in the considered interval. A network is temporal contagion connected when there exists at least one node, such that there exist temporal paths starting from that node and reaching all the other nodes of the graph in the considered interval. These concepts are introduced to define a spreading that starts from some node of the networks and reaches/infected the other nodes in some time. When considering the spreading of a disease, setting, some variants of the previous concepts can be defined, considering for example when a node can possibly spread a contagion.

Temporal Subgraph Isomorphism between temporal networks has also been considered for comparing the spreading process of two diseases Enright and Kao (2018).

#### 7.4 Temporal graphs and single cell technologies

Due to the fast growing variety of single cell technologies and datasets accessible, numerous single cell approaches utilizing temporal networks have been proposed. We report here two examples of applications of temporal graphs to Single Cell data. The first example is the Temporal Network Flow Entropy (TNFE) method introduced in Gao et al. (2022) to detect the critical states during a disease. The work constructs a temporal network on two timestamps for each sample (individual) considered based on gene expression matrix. Then, via a comparison of the differences between the snapshots of the two timestamps, they construct a static network (called differential network). They calculate the Network Flow Entropy (NFE) on the differential network at each timestamp. This measure is used to characterize the network fluctuation of molecules from each individual sample against a given reference sample to identify the crucial stages of disease progression on an individual basis. The experimental work of Gao et al. (2022) is applied on a simulated dataset and six datasets associated with real diseases taken from the NCBI GEO and The Cancer Genome Atlas (TCGA) databases.

Another contribution is presented in Wang et al. (2022), where the network of mitochondria in a cell is analysed. This network undergoes quick alterations through fission, fusion, and motility. Fusion and fission are structural processes that disperse mitochondria over the cell and create densely linked networks. The work of Wang et al. (2022) proposes a software for Mitochondrial Temporal Network

Tracking (MitoTNT), where the temporal network is built considering live-cell fluorescence microscopy data in four dimensions (space dimensions and time). Wang et al. (2022) applies MitoTNT for three tasks: high-resolution investigation of the dynamics of mitochondrial network, node-level study of mitochondrial fission/fusion, and analysis of the mitochondrial temporal network.

## 8 Discussion

As discussed so far, temporal network analysis is currently a hot topic in research. In particular, in the biological and biomedical scenario, the possibility to model the intrinsic variability of the biological systems has increased the interest of researchers. This has led to the definitions of several problems, which show, from a computational point of view, similarities and differences with respect to the static cases.

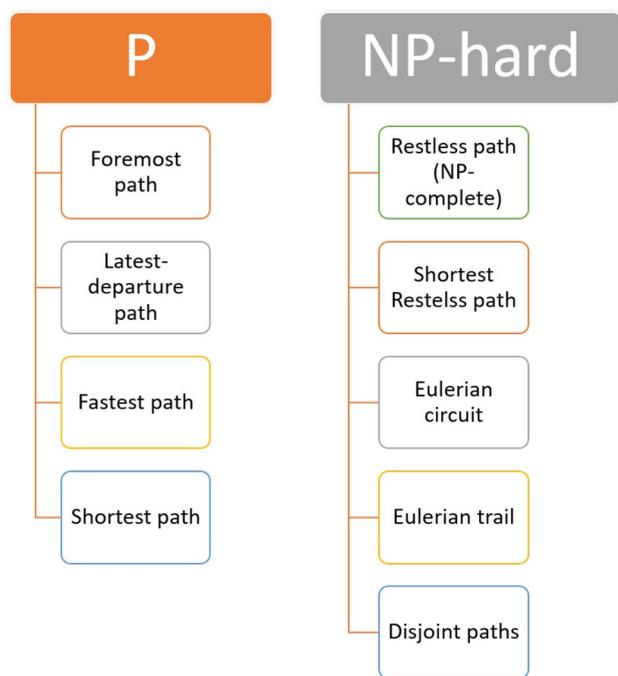
The interest of research of the algorithmic community has initially focused on finding paths or walks. In fact, as discussed in Sect. 4, the temporal evolution leads to several combinatorial problems; for example, it has to be pointed out the difference between fastest and shortest path. The interest on paths and walks problem is motivated also by the several applications of these problems, from transportation to epidemics (see Sect. 7), and for the definitions of centrality measures (see Sect. 3).

Main fundamental problems in temporal networks show complexity ranging from polynomial (P) to non-polynomial (NP-hard) complexity; thus, in Fig. 10, we present the classification of the complexity of the fundamental problems related to paths, walks, circuits, and trails. Notice that all the problems which are NP-hard for temporal graphs are in P for static graphs (when they can be defined in this latter model).

In this survey, we consider also other other fundamental problems for temporal graphs (connectivity, identification of dense subgraphs, covering temporal edges, motif, and colored motif identification), due to their practical relevance and to their interesting properties, different from static cases. Some of these problems have already found application in biology and medicine, others, for example covering temporal edges, are promising approaches that may find future applications, due also to their relevance of the corresponding problem on static graphs.

Notice that although we have considered fundamental problems for temporal graphs, several other problems have been considered in the literature. For example, Feedback Edge Set Haag et al. (2022), Maximum Matching (Baste et al. 2020; Mertzios et al. 2020), and Coloring Marino and Silva (2022), have been considered also for temporal graphs.

Another example is the identification of cohesive subgraphs in a temporal graph. We have considered in Sect. 4 a model of cohesive subgraph based on density (called dense subgraph),



**Fig. 10** Classification of the main problems related to paths, walks, circuits, and trails in temporal graphs, according to their complexity. Notice that Restless path is a decision problem (and it is NP-complete), while the others are all optimization problems

since it is a fundamental primitive in graph mining with several applications Bahmani et al. (2012). Other models of cohesive subgraphs have been considered for temporal graphs, like cliques Himmel et al. (2017),  $k$ -plexes Bentert et al. (2019), and  $k$ -cores (Wu et al. 2015; Liu et al. 2022).

Another direction that has been considered in the literature is the network design. Given a static graph which represents an underlying graph, network design aims at defining timestamps where edges are active, so that we obtain a temporal graph that respects some connectivity property and that minimizes some objective function, like the maximum number of timestamps defined for an edge, or the overall number of timestamps assigned Mertzios et al. (2019).

Another aspects of temporal graphs that have been recently considered is the definition of models, similarly to what has been done for static graphs. In particular, the work in Casteigts et al. (2021) has introduced a random temporal graph model similar to the Erdos–Renyi random graph model, studying the connectivity properties of the resulting temporal graphs.

## 9 Conclusions and future research

In this section, we summarize the main contribution of our work and we point out some future directions as well as some open problems in temporal networks.

After introducing basic concepts, definitions, and measures of temporal networks, we recalled fundamental problems of temporal networks, such as those regarding paths, walks, circuits, and connectivity, with special focus on finding dense subgraphs and on graph covering. Then, we reviewed main algorithms regarding the search of motifs in temporal networks and temporal network comparison (network alignment). An introduction of the role on representation learning (network embedding) has been also provided.

Moving from an abstract layer (problem formulation and basic algorithms), to a concrete layer (software and applications), the paper discusses main software tools and applications for temporal networks. In particular, some of the discussed algorithms that have been implemented in open source software tools are briefly described in a dedicated section. Emerging applications of temporal networks in medicine (brain connectome), biology (protein–protein interaction networks), and public health (epidemics modeling) are also described.

Several contributions have been given on the algorithmic/complexity analysis of problems in temporal networks. Comparing with the case of static graphs, there is a need for the implementations/development of novel algorithms/software tools that are able to solve many problems on temporal networks (for example for the many variant of temporal paths considered in literature). A first contribution in this direction is Teneto Thompson et al. (2017), an open-source library for temporal network analysis, supporting centrality and connectivity measures and temporal network generation, and the TGlib library Oettershagen and Mutzel (2022), an open-source library which includes the computation of temporal centrality measures and statistics, and the implementation of temporal path algorithms.

Considering the problems on temporal networks, although many variants of the path problems have been considered due to their application in epidemiology and transportation, the problem could be further investigated for practical applications, for example by considering a minimum contact time for the transmission of virus.

From an algorithmic point of view, one of the most interesting problem is the parameterized complexity of MinTimeLineCover, when parameterized by the cost of the solution (that is the number of vertices that are active in more than a single timestamp). MinTimeLineCover admits a fixed-parameter algorithm for the parameter cost of the solution, when the time domain consists of exactly two timestamps Froese et al. (2022), but it is not known whether it is a fixed-parameter for larger time domains. Finally, it should be noted that the introduction of parallel and high performance algorithms may significantly impact the field Guzzi et al. (2013).

## Declarations

**Conflict of interest** Authors declare that they have no conflict of interest.

## References

- Aittokallio T, Schwikowski B (2006) Graph-based methods for analysing networks in cell biology. *Brief Bioinform* 7(3):243–255
- Akrida EC, Mertzios GB, Spirakis PG, Zamaraev V (2020) Temporal vertex cover with a sliding time window. *J Comput Syst Sci* 107:108–123
- Albert-László B (2010) Bursts: the hidden patterns behind everything we do from your e-mail to bloody crusades. Penguin
- Alexei V (2013) Spreading dynamics following bursty activity patterns. *Temporal networks*. Springer, pp 161–174
- Almaas E, Kovacs B, Vicsek T, Oltvai ZN, Barabási A-L (2004) Global organization of metabolic fluxes in the bacterium *escherichia coli*. *Nature* 427(6977):839–843
- Aming L, Cornelius Sean P, Liu Y-Y, Long W, Barabási A-L (2017) The fundamental advantages of temporal networks. *Science* 358(6366):1042–1046
- Andrea M, Ana S (2021) Königsberg sightseeing: Eulerian walks in temporal graphs. In: Flocchini Paola, Moura Lucia (eds) *Combinatorial Algorithms - 32nd International Workshop, IWOCA 2021*, Ottawa, ON, Canada, July 5–7, 2021, Proceedings, vol 12757. *Lecture notes in computer science*. Springer, pp 485–500
- Aparício D, Ribeiro P, Milenković T, Silva F (2019) Temporal network alignment via got-wave. *Bioinformatics* 35(18):3527–3529
- Arnaud C, Michael R, Malte R, Viktor Z (2021) Sharp thresholds in random simple temporal graphs. 62nd IEEE Annual Symposium on Foundations of Computer Science, FOCS 2021, Denver, CO, USA, February 7–10, 2022. IEEE, pp 319–326
- Asahiro Y, Iwama K, Tamaki H, Tokuyama T (2000) Greedily finding a dense subgraph. *J Algorithms* 34(2):203–221
- Bahmani B, Kumar R, Vassilvitskii S (2012) Densest subgraph in streaming and mapreduce. *PVLDB* 5(5):454–465
- Balcan D, Colizza V, Gonçalves B, Hao H, Ramasco JJ, Vespignani A (2009) Multiscale mobility networks and the spatial spreading of infectious diseases. *Proc Natl Acad Sci* 106(51):21484–21489
- Barabasi A-L (2005) The origin of bursts and heavy tails in human dynamics. *Nature* 435(7039):207–211
- Barna S, Allison H, Samir K, Louiqa R, Xiao-Ning Z (2010) Dense subgraphs with restrictions and applications to gene annotation graphs. *Annual International Conference on Research in Computational Molecular Biology*. Springer, pp 456–472
- Baste J, Bui-Xuan B-M, Roux A (2020) Temporal matching. *Theor Comput Sci* 806:184–196
- Bentert M, Himmel A-S, Molter H, Morik M, Niedermeier R, Saitenmacher R (2019) Listing all maximal k-plexes in temporal graphs. *ACM J Exp Algorithmics* 24(1):1.13:1-1.13:27
- Berman KA (1996) Vulnerability of scheduled networks and a generalization of Menger's theorem. *Networks* 28(3):125–134
- Betzler N, van Bevern R, Fellows MR, Komusiewicz C, Niedermeier R (2011) Parameterized algorithmics for finding connected motifs in biological networks. *IEEE ACM Trans Comput Biol Bioinform* 8(5):1296–1308
- Bruckner S, Hüffner F, Karp RM, Shamir R, Sharan R (2010) Topology-free querying of protein interaction networks. *J Comput Biol* 17(3):237–252
- Byungjoon Min, Goh K-I, Alexei Vazquez (2011) Spreading dynamics following bursty human activity patterns. *Physical Review E* 83(3):036102
- Cannataro M, Guzzi PH, Veltri P (2010) Protein-to-protein interactions: technologies, databases, and algorithms. *ACM Computing Surv (CSUR)* 43(1):1–36
- Cannataro M, Guzzi PH, Sarica A (2013) Data mining and life sciences applications on the grid. *Wiley Interdiscip Rev* 3(3):216–238
- Casteigts A, Himmel A-S, Molter H, Zschoche P (2021) Finding temporal paths under waiting time constraints. *Algorithmica* 83(9):2754–2802
- Castelli M, Dondi R, Hosseinzadeh MM (2020) Genetic algorithms for finding episodes in temporal networks. *Procedia Comput Sci* 176:215–224
- Chang S, Tong J, Zhu Y, Cui P, Wang F (2020) Network embedding in biomedical data science. *Brief Bioinform* 21(1):182–197
- Charikar M (2000) Greedy approximation algorithms for finding dense components in a graph. In: *Approximation algorithms for combinatorial optimization, Third International Workshop, APPROX 2000*, Proceedings, pages 84–95
- Chow K, Sarkar A, Elheshha R, Cinaglia P, Ay A, Kahveci T (2021) ANCA: alignment-based network construction algorithm. *IEEE/ACM Trans Comput Biol Bioinform* 18(2):512–524
- Cinaglia P, Cannataro M (2022) Network alignment and motif discovery in dynamic networks. *Netw Model Anal Health Inform Bioinform* 11:10
- Cormen TH, Leiserson CE, Rivest RL, Stein C (2009) *Introduction to algorithms*, 3rd edn. MIT Press
- Coscia M, Giannotti F, Pedreschi D (2011) A classification for community discovery methods in complex networks. *Stat Data Mining* 4(5):512–546
- Cui P, Wang X, Pei J, Zhu W (2018) A survey on network embedding. *IEEE Trans Knowl Data Eng* 31(5):833–852
- Dondi R (2022) Insights into the complexity of disentangling temporal graphs. In Ugo Dal Lago and Daniele Gorla, editors, *Proceedings of the 23rd Italian Conference on Theoretical Computer Science, ICTCS 2022*, Rome, Italy, September 7-9, 2022, volume 3284 of *CEUR Workshop Proceedings*, pages 1–13. CEUR-WS.org
- Dondi R, Hosseinzadeh MM, Guzzi PH (2021a) A novel algorithm for finding top-k weighted overlapping densest connected subgraphs in dual networks. *Appl Netw Sci* 6(1):1–17
- Dondi R, Hosseinzadeh MM, Mauri G, Zoppis I (2021b) Top-k overlapping densest subgraphs: approximation algorithms and computational complexity. *J Comb Optim* 41(1):80–104
- Eckmann J-P, Moses E, Sergi D (2004) Entropy of dialogues creates coherent structures in e-mail traffic. *Proc Natl Acad Sci* 101(40):14333–14337
- Elheshha R, Sarkar A, Boucher C, Kahveci T (2019) Identification of co-evolving temporal networks. *BMC Genom* 20(6):1–16
- Enright J, Kao RR (2018) Epidemics on dynamic networks. *Epidemics* 24:88–97
- Fortunato S (2010) Community detection in graphs. *Phys Rep* 486(3–5):75–174
- Francesco Ortuso, Daniele Mercatelli, Hiram Guzzi Pietro, Manuel Giorgi Federico (2021) Structural genetics of circulating variants affecting the sars-cov-2 spike/human ace2 complex. *J Biomol Struct Dyn* 40:1–11
- Francesco P, Hiram GP, Tommaso M (2022) Beyond covid-19 pandemic: topology-aware optimization of vaccination strategy for minimizing virus spreading. *Comput Struct Biotechnol J* 20:2664
- Fransson P, Thompson WH (2020) Temporal flow of hubs and connectivity in the human brain. *Neuroimage* 223:117348
- Fratkin E, Naughton BT, Brutlag DL, Batzoglu S (2006) Motifcut: regulatory motifs finding with maximum density subgraphs. *Bioinformatics* 22(14):e150–e157
- Froese V, Kunz P, Zschoche P (2022) Disentangling the computational complexity of network untangling. *CoRR*, [arXiv:abs/2204.02668](https://arxiv.org/abs/2204.02668)
- Füchle E, Molter H, Niedermeier R, Renken M (2022) Temporal connectivity: Coping with foreseen and unforeseen delays. In: James

- Aspnes and Othon Michail, (eds.), 1st Symposium on Algorithmic Foundations of Dynamic Networks, SAND 2022, March 28–30, 2022, Virtual Conference, volume 221 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, pages 17:1–17:17
- Galbrun E, Gionis A, Tatti N (2016) Top-k overlapping densest subgraphs. *Data Min Knowl Disc* 30(5):1134–1165
- Gao R, Yan J, Li P, Chen L (2022) Detecting the critical states during disease development based on temporal network flow entropy. *Brief Bioinform*
- Goh K-I, Barabási A-L (2008) Burstiness and memory in complex systems. *Europhys Lett* 81(4):48002
- Goldberg Andrew V (1984) Finding a maximum density subgraph. Technical report, Berkeley
- Goyal P, Ferrara E (2018) Graph embedding techniques, applications, and performance: A survey. *Knowl-Based Syst* 151:78–94
- Goyal P, Chhetri SR, Canedo A (2020) *dyngraph2vec*: capturing network dynamics using dynamic graph representation learning. *Knowl-Based Syst* 187:104816
- Guzzi PH, Agapito G, Cannataro M (2013) *coresnp*: parallel processing of microarray data. *IEEE Trans Comput* 63(12):2961–2974
- Guzzi PH, Petrizzelli F, Mazza T (2022b) Disease spreading modeling and analysis: a survey. *Brief Bioinform* 23(4):bbac230
- Haag R, Molter H, Niedermeier R, Renken M (2022) Feedback edge sets in temporal graphs. *Discret Appl Math* 307:65–78
- Hagberg A, Swart P, Chult DS (2008) Exploring network structure dynamics and function using *networkx*. Technical report. Los Alamos National Lab, Los Alamos
- Hamilton W, Ying R, Leskovec J (2017) Representation learning on graphs: Methods and applications. *arXiv preprint arXiv:1709.05584*
- Hamm T, Klobas N, Mertzios GB, Spirakis PG (2022) The complexity of temporal vertex cover in small-degree graphs. *CoRR, arXiv:abs/2204.04832*
- Han JJD, Nicolas B, Tong H, Goldberg DS, Berriz GF, Zhang LV, Denis D, Walhout Albertha JM, Cusick Michael E, Roth Frederick P et al (2004) Evidence for dynamically organized modularity in the yeast protein-protein interaction network. *Nature* 430(6995):88–93
- Himmel A-S, Molter H, Niedermeier R, Sorge M (2017) Adapting the bron-kerbosch algorithm for enumerating maximal cliques in temporal graphs. *Soc Netw Anal Min* 7(1):35:1–35:16
- Hiram GP, Luisa DP, Alessandro G, Pierangelo V (2022) *Pcn-miner*: an open-source extensible tool for the analysis of protein contact networks. *Bioinformatics* 38:4235–4237
- Hiram GP, Marinka Z (2022a) Editorial deep learning and graph embeddings for network biology. *IEEE/ACM Trans Comput Biol Bioinform* 19(2):653–654
- Hiram GP, Tijana M (2018) Survey of local and global biological network alignment: the need to reconcile the two sides of the same coin. *Brief Bioinform* 19(3):472–481
- Holme P, Saramki J (2012) Temporal networks. *Phys Rep* 519(3):97–125
- 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
- Huanhuan W, Cheng J, Yi L, Yiping K, Yuzhen H, Da Yan W, Hejun W (2015) Core decomposition in large temporal graphs. 2015 IEEE International Conference on Big Data (IEEE BigData 2015), Santa Clara, CA, USA, October 29 - November 1, 2015. IEEE Computer Society, pp 649–658
- Hulovatyy Y, Chen H, Milenković T (2015) Exploring the structure and function of temporal networks with dynamic graphlets. *Bioinformatics* 31(12):i171–i180
- Humphries R, Mulchrone K, Tratalos J, More SJ, Hövel P (2021) A systematic framework of modelling epidemics on temporal networks. *Appl Netw Sci* 6(1):1–19
- Ibiapina A, Lopes R, Marino A, Silva A (2022) Menger’s theorem for temporal paths (not walks). *CoRR, arXiv:abs/2206.15251*
- Jayanta KD, Giuseppe T, Pierangelo V, Guzzi Pietro H, Swarup R (2021) Data science in unveiling COVID-19 pathogenesis and diagnosis: evolutionary origin to drug repurposing. *Brief Bioinform* 22(2):855–872
- Jimmy W, Khodaverdian A, Weitz B, Yosef N (2019) Connectivity problems on heterogeneous graphs. *Algorithms Mol Biol* 14(1):5:1–5:17
- Jo H-H, Karsai M, Kertész J, Kaski K et al (2012) Circadian pattern and burstiness in human communication activity. *New J Phys* 14(1):013055
- Kazemi SM, Goel R, Jain K, Kobzyev I, Sethi A, Forsyth P, Poupart P (2020) Representation learning for dynamic graphs: a survey. *J Mach Learn Res* 21(70):1–73
- Kempe D, Kleinberg JM, Kumar A (2002) Connectivity and inference problems for temporal networks. *J Comput Syst Sci* 64(4):820–842
- Khodaverdian A, Yosef N (2022) Directed shortest walk on temporal graphs. *bioRxiv*
- Kim B-H, Ye JC, Kim J-J (2021) Learning dynamic graph representation of brain connectome with spatio-temporal attention. *Adv Neural Inf Process Syst* 34:4314–4327
- Kong Y, Gao S, Yue Y, Hou Z, Shu H, Xie C, Zhang Z, Yuan Y (2021) Spatio-temporal graph convolutional network for diagnosis and treatment response prediction of major depressive disorder from functional connectivity. *Hum Brain Mapp* 42(12):3922–3933
- Kovanen L, Karsai M, Kaski K, Kertész J, Saramäki J (2011) Temporal motifs in time-dependent networks. *J Stat Mech* 2011(11):P11005
- Kumar PR, Jari S (2011) Path lengths, correlations, and centrality in temporal networks. *Phys Rev E* 84(1):016105
- Lacroix V, Fernandes CG, Sagot M-F (2006) Motif search in graphs: Application to metabolic networks. *IEEE ACM Trans Comput Biol Bioinform* 3(4):360–368
- Lebre S, Becq J, Devaux F, Stumpf MPH, Lelandaïs G (2010) Statistical inference of the time-varying structure of gene-regulation networks. *BMC Syst Biol* 4(1):1–16
- Li M, Meng X, Zheng R, Fang-Xiang W, Li Y, Pan Y, Wang J (2020) Identification of protein complexes by using a spatial and temporal active protein interaction network. *IEEE ACM Trans Comput Biol Bioinform* 17(3):817–827
- Liang S, Shaowei T, Zaiqiao M, Qiang Z (2021) Cross-temporal snapshot alignment for dynamic networks. *IEEE Trans Knowledge Data Eng*
- Liu J, Chang X, Yin C, Weiqiang W, Song Y (2022) K-core based temporal graph convolutional network for dynamic graphs. *IEEE Trans Knowl Data Eng* 34(8):3841–3853
- Lvjia C, Shangsong L (2022) Cross-temporal snapshot alignment for dynamic multi-relational networks. *Journal of physics conference. IOP Publishing, Bristol*, p 012038
- Marino A, Silva A (2022) Coloring temporal graphs. *J Comput Syst Sci* 123:171–185
- Masuda N, Lambiotte R (2016) A guide to temporal networks. *World Scientific*
- Masuda N, Miller JC, Holme P (2021) Concurrency measures in the era of temporal network epidemiology: A review. *J R Soc Interface* 18(179):20210019
- Mattsson CES, Takes FW (2021) Trajectories through temporal networks. *Appl Netw Sci* 6(1):1–31
- Merlin BB, Kitty M (2021) Edge exploration of temporal graphs. In: *Flocchini Paola, Moura Lucia (eds) Combinatorial Algorithms - 32nd International Workshop, IWOCA 2021, Ottawa, ON, Canada, July 5–7, 2021, Proceedings, vol 12757. Lecture notes in computer science. Springer, Berlin*, pp 107–121

- Mertzios GB, Michail O, Spirakis PG (2019) Temporal network optimization subject to connectivity constraints. *Algorithmica* 81(4):1416–1449
- Mertzios G, Molter H, Niedermeier R, Zamaraev V, Zschoche P (2020) Computing maximum matchings in temporal graphs. In: Christophe Paul and Markus Bläser, (eds), 37th International Symposium on Theoretical Aspects of Computer Science, STACS 2020, March 10–13, 2020, Montpellier, France, volume 154 of LIPIcs, Schloss Dagstuhl - Leibniz-Zentrum für Informatik, 27:1–27:14
- Michail O (2016) An introduction to temporal graphs: an algorithmic perspective. *Internet Math* 12(4):239–280
- Mohan A, Pramod KV (2022) Representation learning for temporal networks using temporal random walk and deep autoencoder. *Discrete Appl Math*. <https://doi.org/10.1016/j.dam.2022.01.017>
- Murugan A, Husain K, Rust MJ, Hepler C, Bass J, Pietsch JMM, Swain PS, Jena SG, Toettcher JE, Chakraborty AK et al (2021) Roadmap on biology in time varying environments. *Phys Biol* 18(4):041502
- Nassa G, Tarallo R, Guzzi PH, Ferraro L, Cirillo F, Ravo M, Nola E, Baumann M, Nyman TA, Cannataro M et al (2011) Comparative analysis of nuclear estrogen receptor alpha and beta interactomes in breast cancer cells. *Mol BioSyst* 7(3):667–676
- Oetersshagen L, Mutzel P (2022) Tglib: an open-source library for temporal graph analysis. *CoRR*, [arXiv:abs/2209.12587](https://arxiv.org/abs/2209.12587)
- Ou-Yang L, Dai D-Q, Li X-L, Min W, Zhang X-F, Yang P (2014) Detecting temporal protein complexes from dynamic protein-protein interaction networks. *BMC Bioinform* 15(1):1–14
- Paranjape A, Benson A, Leskovec J (2017) Motifs in temporal networks. In: Proceedings of the tenth ACM international conference on web search and data mining, pp 601–610
- Penghang L, Valerio G, Erdem SA (2021) Temporal network motifs Models, limitations, evaluation. *IEEE Trans Knowl Data Eng* 4:945–957
- Petter H, Jari S (2019) A map of approaches to temporal networks. *Temporal network theory*. Springer, pp 1–24
- Przytycka TM, Singh M, Slonim DK (2010) Toward the dynamic interactome: it's about time. *Briefings Bioinform* 11(1):15–29
- Redmond U, Cunningham P (2016) Subgraph isomorphism in temporal networks. *arXiv preprint* [arXiv:1605.02174](https://arxiv.org/abs/1605.02174)
- Ren Y, Sarkar A, Veltri P, Ay A, Dobra A, Kahveci T (2021) Pattern discovery in multilayer networks. *IEEE/ACM Trans Comput Biol Bioinform* 19(2):741–752
- Riccardo D, Mehdi HM (2021a) Dense sub-networks discovery in temporal networks. *SN Comput Sci* 2(3):1–11
- Riccardo D, Mehdi HM (2021b) Finding colorful paths in temporal graphs. *International Conference on Complex Networks and Their Applications*. Springer, pp 553–565
- Riccardo D, Giancarlo M, Italo Z (2019) Graph algorithms. In: Ranganathan Shoba, Gribskov Michael, Nakai Kenta, Schönbach Christian (eds) *Encyclopedia of bioinformatics and computational biology*, vol 1. Elsevier, pp 940–949
- Riccardo D, Mehdi HM, Italo Z (2022) Dense temporal subgraphs in protein-protein interaction networks. *International Conference on Computational Science*. Springer, pp 469–480
- Roberto G, Giovanni M, Alfredo F, Alfredo P (2021) Modit: motif discovery in temporal networks. *Front Big Data* 4:806014
- Rozenstein P, Gionis A (2019) Mining temporal networks. In: Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, ACM, pp 3225–3226
- Rozenstein P, Bonchi F, Gionis A, Sozio M, Tatti N (2020) Finding events in temporal networks: segmentation meets densest subgraph discovery. *Knowl Inf Syst* 62(4):1611–1639
- Rozenstein P, Tatti N, Gionis A (2021) The network-untangling problem: from interactions to activity timelines. *Data Min Knowl Discov* 35(1):213–247
- Russell Peter J, Hertz Paul E, Beverly McMillan, Joel Benington (2020) *Biology: the dynamic science*. Cengage Learning, Omen
- Sarkis-Onofre R, Catalá-López F, Aromataris E, Lockwood C (2021) How to properly use the prisma statement. *SystRev* 10(1):1–3
- Shawn G, Jiang M, Guzzi PH, Milenković T (2022) Modeling multi-scale data via a network of networks. *Bioinformatics* 38(9):2544–2553
- Şimşek Ö, Barto A (2008) Skill characterization based on betweenness. *Adv Neural Inform Processing Syst*, 21
- Skrlj B, Kralj J, Lavrac N (2019) Py3plex toolkit for visualization and analysis of multilayer networks. *Appl Netw Sci* 4(1):94
- Song C, Ge T, Chen C, Wang J (2014) Event pattern matching over graph streams. *Proc VLDB Endowment* 8(4):413–424
- Srinivasan P, Shirish T, Duygu U (2010) A survey of graph mining techniques for biological datasets. *Managing and mining graph data*. Springer, pp 547–580
- Takaguchi T, Masuda N, Holme P (2013) Bursty communication patterns facilitate spreading in a threshold-based epidemic dynamics. *PLoS ONE* 8(7):e68629
- Tang J, Musolesi M, Mascolo C, Latora V (2009) Temporal distance metrics for social network analysis. In: Proceedings of the 2nd ACM workshop on Online social networks, pp 31–36
- Thejaswi S, Gionis A, Lauri J (2020) Finding path motifs in large temporal graphs using algebraic fingerprints. *Big Data* 8(5):335–362
- Thompson WH, Fransson P (2016) Bursty properties revealed in large-scale brain networks with a point-based method for dynamic functional connectivity. *Sci Rep* 6(1):1–13
- Thompson WH, Brantefors P, Fransson P (2017) From static to temporal network theory: applications to functional brain connectivity. *Network Neurosci* 1(2):69–99
- Ursula R, Pádraig C (2013) Temporal subgraph isomorphism. 2013 *IEEE/ACM International Conference on advances in social networks analysis and mining (ASONAM 2013)*. IEEE, pp 1451–1452
- van den Heuvel MP, Mandl RCW, Stam CJ, Kahn RS, Pol HEH (2010) Aberrant frontal and temporal complex network structure in schizophrenia: a graph theoretical analysis. *J Neurosci* 30(47):15915–15926
- Vázquez A, Oliveira JG, Dezsö Z, Goh K-I, Kondor I, Barabási A-L (2006) Modeling bursts and heavy tails in human dynamics. *Phys Rev E* 73(3):036127
- Vazquez A, Racz B, Lukacs A, Barabasi A-L (2007) Impact of non-poissonian activity patterns on spreading processes. *Phys Rev Lett* 98(15):158702
- Veltri P, Cannataro M, Tradigo G (2007) Sharing mass spectrometry data in a grid-based distributed proteomics laboratory. *Inform Processing Manag* 43(3):577–591
- Walter N, Marinka Z, Bo W, Jure L, Anna G, Roded S (2019) To embed or not: network embedding as a paradigm in computational biology. *Front Genetics* 10:381
- Wang Z, Natekar P, Tea C, Tamir S, Hakozaki H, Schoeneberg J (2022) Mitotnt: mitochondrial temporal network tracking for 4d live-cell fluorescence microscopy data. *bioRxiv*
- Xiangmao M, Wenkai L, Xiang J, Dino BH, Wang Wenkang W, Fang-Xiang LM (2022) Temporal-spatial analysis of the essentiality of hub proteins in protein-protein interaction networks. *IEEE Trans Netw Sci Eng* 9(5):3504–3514
- Zhang Y, Lin L, Yuan P, Jin H (2022) Significant engagement community search on temporal networks: Concepts and algorithms. *arXiv preprint* [arXiv:2206.06350](https://arxiv.org/abs/2206.06350)

Zschoche P (2022) Restless temporal path parameterized above lower bounds. CoRR, [arXiv:abs/2203.15862](https://arxiv.org/abs/2203.15862)

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.