

# Sparse Temporal Spanners with Low Stretch

**Davide Bilò**  

Department of Information Engineering, Computer Science and Mathematics,  
University of L'Aquila, Italy

**Gianlorenzo D'Angelo**  

Gran Sasso Science Institute, L'Aquila, Italy

**Luciano Gualà**  

Department of Enterprise Engineering, University of Rome "Tor Vergata", Italy

**Stefano Leucci**  

Department of Information Engineering, Computer Science and Mathematics,  
University of L'Aquila, Italy

**Mirko Rossi**  

Gran Sasso Science Institute, L'Aquila, Italy

---

## Abstract

A *temporal graph* is an undirected graph  $G = (V, E)$  along with a function  $\lambda : E \rightarrow \mathbb{N}^+$  that assigns a time-label to each edge in  $E$ . A path in  $G$  such that the traversed time-labels are non-decreasing is called a *temporal path*. Accordingly, the distance from  $u$  to  $v$  is the minimum length (i.e., the number of edges) of a temporal path from  $u$  to  $v$ . A temporal  $\alpha$ -spanner of  $G$  is a (temporal) subgraph  $H$  that preserves the distances between any pair of vertices in  $V$ , up to a *multiplicative* stretch factor of  $\alpha$ . The *size* of  $H$  is measured as the number of its edges.

In this work, we study the size-stretch trade-offs of temporal spanners. In particular we show that temporal cliques always admit a temporal  $(2k - 1)$ -spanner with  $\tilde{O}(kn^{1+\frac{1}{k}})$  edges, where  $k > 1$  is an integer parameter of choice. Choosing  $k = \lfloor \log n \rfloor$ , we obtain a temporal  $O(\log n)$ -spanner with  $\tilde{O}(n)$  edges that has almost the same size (up to logarithmic factors) as the temporal spanner given in [Casteigts et al., JCSS 2021] which only preserves temporal connectivity.

We then turn our attention to *general* temporal graphs. Since  $\Omega(n^2)$  edges might be needed by any connectivity-preserving temporal subgraph [Axiotis et al., ICALP'16], we focus on approximating distances from a *single source*. We show that  $\tilde{O}(n/\log(1 + \varepsilon))$  edges suffice to obtain a stretch of  $(1 + \varepsilon)$ , for any small  $\varepsilon > 0$ . This result is essentially tight in the following sense: there are temporal graphs  $G$  for which any temporal subgraph preserving exact distances from a single-source must use  $\Omega(n^2)$  edges. Interestingly enough, our analysis can be extended to the case of *additive* stretch for which we prove an upper bound of  $\tilde{O}(n^2/\beta)$  on the size of any temporal  $\beta$ -additive spanner, which we show to be tight up to polylogarithmic factors.

Finally, we investigate how the *lifetime* of  $G$ , i.e., the number of its distinct time-labels, affects the trade-off between the size and the stretch of a temporal spanner.

**2012 ACM Subject Classification** Theory of computation  $\rightarrow$  Sparsification and spanners

**Keywords and phrases** temporal spanners, temporal graphs, graph sparsification, approximate distances

**Digital Object Identifier** 10.4230/LIPIcs.ESA.2022.19

**Related Version** *Full Version:* <https://arxiv.org/abs/2206.11113>

**Funding** *Davide Bilò:* Partially supported by the research project "Spanners and Oracles in Static and Temporal Networks" (04ATE2022.AVVIO.Bilò) funded by the University of L'Aquila.

*Gianlorenzo D'Angelo:* Partially supported by the Italian MIUR PRIN 2017 Project ALGADIMAR "Algorithms, Games, and Digital Markets".

*Stefano Leucci:* Partially supported by the research project "Spanners and Oracles in Static and Temporal Networks" (04ATE2022.AVVIO.Bilò) funded by the University of L'Aquila.



© Davide Bilò, Gianlorenzo D'Angelo, Luciano Gualà, Stefano Leucci, and Mirko Rossi;  
licensed under Creative Commons License CC-BY 4.0

30th Annual European Symposium on Algorithms (ESA 2022).

Editors: Shiri Chechik, Gonzalo Navarro, Eva Rotenberg, and Grzegorz Herman; Article No. 19; pp. 19:1–19:16

Leibniz International Proceedings in Informatics



LIPICs Schloss Dagstuhl – Leibniz-Zentrum für Informatik, Dagstuhl Publishing, Germany

## 1 Introduction

A *temporal graph* is a graph  $G = (V, E)$  in which each edge can be used only in certain time instants. This recurrent idea of time-evolving graphs has been formalized in multiple ways, and a simple widely-adopted model is the one of Kempe, Kleinberg, and Kumar [11], in which each edge  $e \in E$  has an assigned time-label  $\lambda(e)$  representing the instant in which  $e$  can be used. A path from a vertex to another in  $G$  is said to be a *temporal path* if the time-labels of the traversed edges are non-decreasing. Accordingly, a graph is *temporally connected* if there exists a temporal path from  $u$  to  $v$ , for every two vertices  $u, v \in V$ .

Notice that, unlike paths in *static* graphs, the existence of temporal paths is neither symmetric nor transitive.<sup>1</sup> For this reason, temporal graphs exhibit a different combinatorial structure compared to static graphs, and even problems that admit easy solutions on static graphs become more challenging in their temporal counterpart. Indeed, one of the main problems introduced in the seminal paper of Kempe, Kleinberg, and Kumar [11] is that of finding a sparse temporally connected subgraph  $H$  of an input temporal graph  $G$ . Such a subgraph  $H$  is sometimes referred to as a *temporal spanner* of  $G$ . While any spanning-tree is trivially a connectivity-preserving subgraph of a *static* graph, not all temporal graphs  $G$  admit a temporal spanner having  $O(n)$  edges [11]. In particular, [11] exhibits a class of temporal graphs that contain  $\Theta(n \log n)$  edges and cannot be further sparsified. Later, [4] provided a stronger negative result showing that there are temporal graphs  $G$  such that any temporal spanner of  $G$  must use  $\Theta(n^2)$  edges. These strong lower bounds on general graphs motivated [7] to focus on *temporal cliques* instead. Here the situation improves significantly, as only  $O(n \log n)$  edges are sufficient to guarantee temporal connectivity. This gives rise to the following natural question, which is exactly the focus of our paper: *can one design a temporal spanner that also guarantees short temporal paths between any pair of vertices?*

To address this question, we measure the *length* of a temporal path as the number of its edges,<sup>2</sup> and we introduce the notion of temporal  $\alpha$ -spanner of a temporal graph  $G$ , i.e., a subgraph  $H$  of  $G$  such that  $d_H(u, v) \leq \alpha \cdot d_G(u, v)$  for every pair of vertices  $u, v \in V$ , where  $d_H(u, v)$  (resp.  $d_G(u, v)$ ) denotes the length of a shortest temporal path from  $u$  to  $v$  in  $H$  (resp.  $G$ ). Our main question then becomes that of understanding which trade-offs can be achieved between the *size*, i.e., the number of edges, of  $H$  and the value of its *stretch-factor*  $\alpha$ . This same question received considerable attention on static graphs and gave rise to a significant amount of work (see, e.g., [1]), hence we deem investigating its temporal counterpart as a very interesting research direction.

To the best of our knowledge, the only temporal  $\alpha$ -spanner currently known is actually the connectivity-preserving subgraph of [7] having size  $O(n \log n)$ . However, a closer inspection of its construction shows that the resulting  $\alpha$ -spanner can have stretch  $\alpha = \Theta(n)$ . In particular, even the problem of achieving stretch  $o(n)$  using  $o(n^2)$  edges remains open.

In this paper we investigate which size-stretch trade-offs can be attained by selecting subgraphs of temporal graphs, as detailed in the following.

<sup>1</sup> Indeed, a temporal path from  $u$  to  $v$  is not necessarily a temporal path from  $v$  to  $u$ , even when  $G$  is undirected. Moreover, the existence of a temporal path from  $u$  to  $v$ , and of a temporal path from  $v$  to  $w$ , does not imply the existence of a temporal path from  $u$  to  $w$ .

<sup>2</sup> Alternative definitions for the length of a temporal path are also natural, e.g., the *arrival time*, *departure time*, *duration*, or *travel time* as we briefly discuss in the conclusions.

## 1.1 Our results

**Temporal cliques.** Following [7], we start by considering temporal cliques (see Section 3). Our main result is the following: given a temporal clique  $G$  and an integer  $k \geq 2$ , we can construct, in polynomial time, a temporal  $(2k-1)$ -spanner of  $G$  having size  $O(kn^{1+1/k} \log^{1-1/k} n)$ . Interestingly, the special case  $k = \lfloor \log n \rfloor$  shows that  $O(n \log^2 n)$  edges suffice to ensure that a temporal path of length  $O(\log n)$  exists between any pair of vertices. For this choice of  $k$ , the size of our spanner is only a logarithmic factor away from the size the temporal spanner of [7] that uses  $O(n \log n)$  edges and only preserves connectivity. We obtain our results by constructing hierarchical clustering of the vertices that guides the constructions of temporal paths.

We also show that there are temporal cliques for which any temporal spanner with stretch smaller than 3 must have  $\Omega(n^2)$  edges.

**Single-source temporal spanners on general graphs.** Next, in Section 4, we move our attention from temporal cliques to general temporal graphs. As already pointed out, there are temporal graphs that do not admit any connectivity-preserving subgraph with  $o(n^2)$  edges [4]. Hence, we consider the special case in which we have a single source  $s$ . One can observe that any temporal graph  $G$  admits a temporal subgraph containing  $O(n)$  edges and preserving the connectivity from  $s$  (see also [11]). However, to the best of our knowledge, no non-trivial result is known on the size of subgraphs preserving approximate distances from  $s$ .

We formalize this problem by introducing the notion of *single-source temporal  $\alpha$ -spanner* of  $G = (V, E)$  w.r.t. a source  $s \in V$ , which we define as a subgraph  $H$  of  $G$  such that  $d_H(s, v) \leq \alpha \cdot d_G(s, v)$  for every  $v \in V$ . Our main contribution for the single-source case is the following: given any temporal graph  $G$ , we can compute in polynomial time a single-source temporal  $(1 + \varepsilon)$ -spanner having size  $O(\frac{n \log^4 n}{\log(1+\varepsilon)})$ , where  $\varepsilon > 0$  is a parameter of choice.

Furthermore, we show that any single-source temporal 1-spanner (i.e., a subgraph preserving *exact* distances from  $s$ ) must have  $\Omega(n^2)$  edges in general. Our construction can be generalized to provide a lower bound of  $\Omega(\frac{n^2}{\beta})$  on the size of any single-source temporal  $\beta$ -additive spanner, namely a subgraph  $H$  that preserves single-source distances up to an *additive* term of at most  $\beta \geq 1$  (i.e., we require  $d_H(s, v) \leq d_G(s, v) + \beta$  for all  $v \in V$ ).

Interestingly, the same techniques used to obtain our single-source temporal  $(1 + \varepsilon)$ -spanner can be also applied to build a single-source temporal  $\beta$ -additive spanner of size  $O(\frac{n^2 \log^4 n}{\beta})$ , which essentially matches our aforementioned lower bound.

**The role of lifetime.** An important parameter that measures how time-dependent is a temporal graph  $G = (V, E)$  is its *lifetime*, i.e., the number  $L$  of distinct time-labels associated with the edges of  $G$ . Indeed, a temporal graph with lifetime  $L = 1$  is just a *static* graph, while any temporal graph trivially satisfies  $L = O(n^2)$ . It is not surprising that the lifetime plays a crucial role in determining the number of edges required by temporal spanners. For example, the lower bound of  $\Omega(n^2)$  on the size of any connectivity-preserving temporal subgraph requires  $L = \Omega(n)$  [4]. In this paper, we also present a collection of results with the goal of shedding some light on the lifetime-size trade-off of temporal spanners. In particular, our results provide the following lifetime-dependant upper bounds on the size of temporal  $\alpha$ -spanners

- As far as temporal cliques are concerned, we show how to build, in polynomial time, a temporal 3-spanner with  $O(2^L n \log n)$  edges. This implies that, when  $L = O(1)$ , we can achieve stretch 3 with  $\tilde{O}(n)$  edges.<sup>3</sup>

<sup>3</sup> The notation  $\tilde{O}(f(n))$  is a synonym for  $O(f(n) \cdot \text{polylog } f(n))$ .

- If  $L = 2$ , we can find (in polynomial time) a temporal 2-spanner of a temporal clique having size  $O(n \log n)$ . We deem this result interesting since, as soon as  $L > 2$ , our lower bound of  $\Omega(n^2)$  on the size of any temporal 2-spanner still applies.
- We show that, when  $L$  is small, *general* temporal graphs can be sparsified by exploiting known size-stretch trade-offs for spanners of static graphs. In particular, we show that if it is possible to compute, in polynomial time, an  $\alpha$ -spanner of a static graph having size  $f(n)$ , then one can also build a temporal  $\alpha$ -spanner of size  $O(Lf(n))$ . This yields, e.g., a temporal  $\lfloor \log n \rfloor$ -spanner of size  $o(n^2)$  on general temporal graphs with  $L = o(n)$ .

Due to space limitations, these results and some of the proofs are omitted and can be found in the full version of the paper.

## 1.2 Related work

The definitions of temporal graphs and temporal paths given in the literature sometimes differ from the ones we adopt here. We now discuss how our results relate to some of the most common variants. A first difference concerns the notion of temporal paths: some authors consider *strict temporal paths* [2, 7, 11], i.e., temporal paths in which edge labels must be strictly increasing (rather than non-decreasing). As observed by [11], if we adopt strict temporal paths then there are dense graphs that cannot be sparsified, indeed no edge can be removed from a temporal clique in which all edges have the same time-label. As observed in [7], one can get rid of these problematic instances by assuming that time-labels are *locally distinct*, namely that all the time-labels of the edges incident to any single vertex are distinct. In this case all temporal paths are also strict temporal paths and hence they focus on temporal paths as defined in our paper. A second difference concerns whether edges are allowed to have multiple time-labels, as in [2, 12]. In this case, each edge  $e$  is associated to a non-empty set of time instants  $\lambda(e) \subseteq \mathbb{N}^+$  in which  $e$  is available. We observe that any algorithm that sparsifies a temporal clique with single time-labels can be directly used on the case of multiple time-labels by selecting an arbitrary time-label for each edge (see also the discussion in [7]). This is no longer true when we consider general temporal graphs, since removing edge labels might affect distances. However, all our algorithms work also in the case of multiple labels and, since our lower bounds are given for single labels, they also apply to the case of multiple labels.

Another research line concerns random temporal graphs. In particular, temporal cliques in which each edge has a single time-label chosen u.a.r. from the set  $\{1, \dots, \alpha\}$ , where  $\alpha \geq 4$ , admit temporal spanners with  $O(n \log n)$  edges w.h.p. [2]. In [8], the authors study connectivity properties of random temporal graphs defined as an Erdős-Rényi graph  $G_{n,p}$  in which each edge  $e$  has time-label chosen as the rank of  $e$  in a random permutation of the graph's edges. They show that  $p = \frac{\log n}{n}$ ,  $p = \frac{2 \log n}{n}$ ,  $p = \frac{3 \log n}{n}$ , and  $p = \frac{4 \log n}{n}$  are sharp thresholds to guarantee that the resulting temporal graph  $G$  satisfies the following respective conditions asymptotically almost surely: a fixed pair of vertices can reach each other via temporal paths in  $G$ , there is some vertex  $s$  which can reach all other vertices in  $G$  via temporal paths,  $G$  is temporally connected,  $G$  and admits a temporal spanner with  $2n - 4$  edges (which is tight when time-labels are locally distinct).

Besides temporal graphs, other models to represent graphs or paths that evolve over time have been considered in the literature, we refer the interested reader to [9] for a survey.

Finally, as we already mentioned, there is a large body of literature concerning spanners on static graphs (see [1] for a survey on the topic) and clustering techniques similar to the ones we employ on temporal cliques have proven to be a useful tool to design sparse spanner also in this setting (see, e.g., [5, 6]).

A reader that is already familiar with the area might notice that our upper bound of  $\tilde{O}(n^{1+\frac{1}{k}})$  on the size of a temporal  $(2k-1)$ -spanner of a *temporal clique*, happens to resemble the classical upper bound of  $O(n^{1+\frac{1}{k}})$  on the size of a  $(2k-1)$ -spanner of a *general static graph* [3]. Nevertheless, the first result only applies to complete (temporal) graphs and requires different technical arguments to handle temporal paths.

## 2 Model and preliminaries

Let  $G = (V, E)$  be an undirected *temporal graph* with  $n$  vertices, and a labeling function  $\lambda : E \rightarrow \mathbb{N}^+$  that assigns a *time-label*  $\lambda(e)$  to each edge  $e$ . If  $G$  is complete we will say that it is a *temporal clique*. A temporal path  $\pi$  from vertex  $u$  to vertex  $v$  is a path in  $G$  from  $u$  to  $v$  such that the sequence  $e_1, e_2, \dots, e_k$  of edges traversed by  $\pi$  satisfies  $\lambda(e_i) \leq \lambda(e_{i+1})$  for all  $i = 1, \dots, k-1$ . We denote with  $|\pi|$  the length of  $\pi$ , i.e., the number of its edges. A shortest temporal path from vertex  $u$  to vertex  $v$  is a temporal path from  $u$  to  $v$  with minimum length. We denote with  $d_G(u, v)$ , the length of a shortest temporal path from  $u$  to  $v$  in  $G$ . Given a generic graph  $H$ , we denote by  $V(H)$  its vertex-set and by  $E(H)$  its edge-set.

For  $\alpha \geq 1$  and  $\beta \geq 0$ , a temporal  $(\alpha, \beta)$ -spanner of  $G$  is a (temporal) subgraph  $H$  of  $G$  such that  $V(H) = V$  and  $d_H(u, v) \leq \alpha \cdot d_G(u, v) + \beta$ , for each  $u, v \in V$ . We call a temporal  $(\alpha, \beta)$ -spanner: (i) temporal  $\alpha$ -spanner if  $\beta = 0$ , (ii) temporal  $\beta$ -additive spanner if  $\alpha = 1$ , (iii) temporal preserver if  $\alpha = 1$  and  $\beta = 0$ . We say that  $H$  is a *single-source* temporal  $(\alpha, \beta)$ -spanner w.r.t. a vertex  $s \in V$ , if  $d_H(s, v) \leq \alpha \cdot d_G(s, v) + \beta$ , for each  $v \in V$ . The *size* of a temporal spanner is the number of its edges.

We define the *lifetime*  $L$  of  $G$  as the number of distinct time-labels of its edges. Furthermore, we assume w.l.o.g. that each time instant in  $\{1, \dots, L\}$  is used by at least one time-label (since otherwise we can replace each time-label with its rank in the set  $\{\lambda(e) \mid e \in E\}$ ), so that  $L = \max_{e \in E} \lambda(e)$ .

We will make use of the following well-known result:

► **Lemma 1.** *Given a collection  $\mathcal{S}$  of subsets of  $\{1, \dots, n\}$ , where each subset has size at least  $\ell$  and  $|\mathcal{S}|$  is polynomially bounded in  $n$ , we can find in polynomial time a subset  $R \subseteq \{1, \dots, n\}$  of size  $O((n/\ell) \log n)$  that hits all subsets in the collection, i.e.,  $R \cap S \neq \emptyset$  for all  $S \in \mathcal{S}$ .*

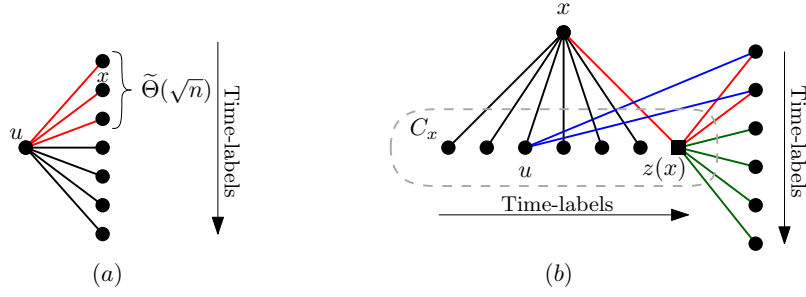
## 3 Spanners for temporal cliques

In this section, we design an algorithm such that, given a temporal clique  $G$ , returns a temporal  $(2k-1)$ -spanner  $H$  of  $G$  with size  $\tilde{O}(n^{1+\frac{1}{k}})$ , for any integer  $k > 1$ . We also provide a temporal clique  $G$  for which any temporal 2-spanner of  $G$  has size  $\Omega(n^2)$ .

Before describing the algorithm for constructing temporal  $(2k-1)$ -spanners, we show as a warm up how to construct a temporal 3-spanner and a temporal 5-spanner of size  $\tilde{O}(n^{1+\frac{1}{2}})$  and  $\tilde{O}(n^{1+\frac{1}{3}})$ , respectively.

### 3.1 Our temporal 3-spanner

Given a temporal clique  $G$ , we construct a temporal 3-spanner  $H$  of  $G$  via a clustering technique. For each  $u \in V$ , we select a set  $E_u$  containing the  $\Theta(\sqrt{n \log n})$  edges incident to  $u$  having the smallest labels (ties are broken arbitrarily). We define  $S_u = \{v \in V \mid (u, v) \in E_u\}$ . Next, we find a hitting set  $R \subseteq V$  of the collection  $\{S_u\}_{u \in V}$ . Thanks to Lemma 1, we can deterministically compute a hitting set of size  $|R| = O(\sqrt{n \log n})$ .



■ **Figure 1** (a) A vertex  $u$  and its neighbours in  $G$ , the edges are sorted top-down by increasing time-label. The red edges are those belonging to  $E_u$  while the black edges belong to  $E(G) \setminus E_u$ . (b) An example of a cluster  $C_x$  where  $x \in R$ . The edges incident to  $x$  are sorted from left to right by increasing-time label. The black and red edges are added to  $E(H)$  during the initialization phase, in particular the red edges are those belonging to  $E_{z(x)}$ . The blue edges are added w.r.t.  $u$  to  $E(H)$  during the first augmentation. The green edges belong to  $E(G) \setminus E_{z(x)}$  and are added to  $E(H)$  during the second augmentation.

We partition the vertices of  $V$  into  $|R|$  clusters. More precisely, we create a cluster  $C_x \subseteq V$  for each vertex  $x \in R$ . Each vertex  $u \in V$  belongs to exactly one arbitrarily chosen cluster  $C_x$  that satisfies  $x \in S_u$ , i.e.,  $x$  hits  $S_u$ . We call  $x$  the center of cluster  $C_x$ .<sup>4</sup> Moreover, we choose the special vertex of cluster  $C_x$  as a vertex  $z(x)$  in  $C_x$  that maximizes the label of the edge  $(x, z(x))$ .

Notice that, for every  $x \in R$  and  $u \in C_x$ ,  $u$  can reach  $z(x)$  via a temporal path of length at most 2 in  $G$  by using the edges  $(u, x)$  and  $(x, z(x))$  since, by definition of  $z(x)$  and  $S_u$ , we have  $\lambda(u, x) \leq \lambda(x, z(x))$ .

We now build our temporal spanner  $H$  of  $G$ . The set of edges  $E(H)$  is constructed in three phases (See Figure 1 for an example of the whole construction):

**Initialization:** For each  $u \in V$ , we add the edges in  $E_u$  to  $E(H)$ ;

**First Augmentation:** For every  $u \in V$ , we add the edges in  $E_{u, z(x)} = \{u\} \times S_{z(x)}$  to  $E(H)$ , where  $x$  is the center of the cluster containing  $u$ ;

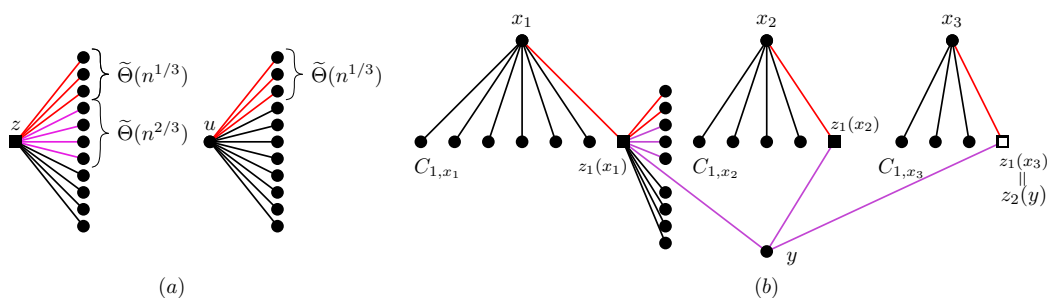
**Second Augmentation:** For each  $x \in R$ , we add the edges in  $E_{z(x), V} = \{z(x)\} \times V$  to  $E(H)$ . It is easy to see that  $H$  contains  $O(n\sqrt{n \log n})$  edges. We now show that for any  $u, v \in V$  there is a temporal path from  $u$  to  $v$  of length at most 3 in  $H$ . Indeed, let  $x \in R$  be the center of the cluster  $C_x$  containing  $u$ . If  $v = z(x)$  then, since  $u \in C_x$ , the initialization phase ensures that  $(u, x) \in E(H)$  and  $(x, z(x)) \in E(H)$ , which form a temporal path as we already discussed above. We hence assume that  $v \neq z(x)$ . If  $(z(x), v) \in E_{z(x)}$  then the first augmentation phase added  $(u, v) \in E_{u, z(x)}$  to  $E(H)$ , which is a temporal path of length one from  $u$  to  $v$ . Otherwise  $(z(x), v) \in E(G) \setminus E_{z(x)}$  and, the second augmentation phase added edge  $(z(x), v)$  to  $E(H)$ . Moreover, since  $(z(x), v) \notin E_{z(x)}$ ,  $(z(x), v)$  is not among the  $\Theta(\sqrt{n \log n})$  edges incident to  $z(x)$  with lowest labels. As a consequence, since  $(x, z(x)) \in E_{z(x)}$ , we have  $\lambda(x, z(x)) \leq \lambda(z(x), v)$ . Hence, the edges  $(u, x)$ ,  $(x, z(x))$ , and  $(z(x), v)$  form a temporal path of length 3 from  $u$  to  $v$  in  $H$ .

### 3.2 Our temporal 5-spanner

We show how to modify the construction of a temporal 3-spanner given in previous section in order to obtain a temporal 5-spanner of size  $\tilde{O}(n^{4/3})$ . The idea is to replace the single-level clustering of Section 3.1 with a two-level clustering, where the second-level clustering

<sup>4</sup> Here and throughout the paper, the center of a cluster is not required to belong to the cluster itself.





■ **Figure 2** (a) Two vertices  $u$  and  $z$  of  $G$ , where  $z \in Z_1$  and the red edges belong to  $E_{1,u}$  and  $E_{1,z}$ , respectively. For vertex  $z$  the purple edges belong to  $E_{2,z}$ . (b) A two level clustering. The level one consists of three cluster  $C_{1,x_1}$ ,  $C_{1,x_2}$ ,  $C_{2,x_3}$ . The level-two cluster  $C_{2,y}$ , with  $y \in R_2$ , contains vertices  $z_1(x_1)$ ,  $z_1(x_2)$  and  $z_1(x_3)$ , where  $z_2(y) = z_1(x_3)$ .

partitions the special vertices of the first level clustering and the number of selected clusters decreases as we move from the first level to the second one.

The level-one clustering is built similarly to the one used in our temporal 3-spanner. For each vertex  $u \in V$  we define sets  $E_{1,u}$  and  $S_{1,u}$  where  $E_{1,u}$  consists of the  $\Theta(n^{1/3} \log^{2/3} n)$  edges with the smallest label among those incident to  $u$  (ties are broken arbitrarily) and  $S_{1,u} = \{v \in V \mid (u, v) \in E_{1,u}\}$ . We compute a hitting set  $R_1$  of the collection  $\{S_{1,u}\}_{u \in V}$ , where  $R_1$  has size  $O(n^{2/3} \log^{1/3} n)$  thanks to Lemma 1. We partition the vertices of  $V$  into  $|R_1|$  clusters  $C_{1,x}$ , for each  $x \in R_1$ , as before, and let  $z_1(x)$  the vertex in  $C_{1,x}$  that maximizes the label of the edge  $(x, z_1(x))$ .

The level-two clustering is built on top of the vertices  $Z_1 = \{z_1(x) \mid x \in R_1\}$ . For each  $u \in Z_1$ , we define  $E_{2,u}$  as a set of  $\Theta(n^{2/3} \log^{1/3} n)$  edges with the smallest label among those that are incident to  $u$  but do not belong to  $E_{1,u}$ . We also define a corresponding set  $S_{2,u} = \{v \in V \mid (u, v) \in E_{2,u}\}$ . We once again invoke Lemma 1 to compute a hitting set  $R_2$  of size  $O(n^{1/3} \log^{2/3} n)$  of the collection  $\{S_{2,u}\}_{u \in Z_1}$ . Based on  $R_2$ , we partition the special vertices in  $Z_1$  by associating each  $u \in Z_1$  to an arbitrary cluster  $C_{2,y}$  centered in  $y \in R_2$  such that  $y \in S_{2,u}$ . Each cluster  $C_{2,y}$  has an associated special vertex  $z_2(y) \in C_{2,y}$  chosen among the ones that maximize the label of the edge  $(y, z_2(y))$ , see Figure 2.

We are now ready to build our temporal 5-spanner  $H$ . As before, the set of edges  $E(H)$  is constructed in three phases:

**Initialization:** For each  $u \in V$ , we add the edges in  $E_{1,u}$  to  $E(H)$  and, for each  $u \in Z_1$ , we add the edges in  $E_{2,u}$  to  $E(H)$ ;

**First Augmentation:** For every  $u \in V$ , we add the edges in  $E_{u,z_1(x)} = \{u\} \times S_{1,z_1(x)}$  to  $E(H)$ , where  $x$  is the center of the cluster containing  $u$ . Moreover, for each  $z \in Z_1$  we add the edges in  $\{z\} \times (S_{1,z_2(y)} \cup S_{2,z_2(y)})$  to  $E(H)$ , where  $y$  is the center of the level-two cluster  $C_{2,y}$  containing  $z$ ;

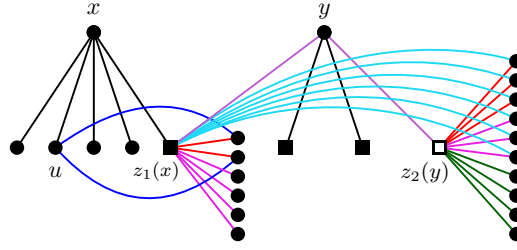
**Second Augmentation:** For each  $y \in R_2$ , we add the set  $\{z_2(y)\} \times V$  to  $E(H)$ .

See Figure 3 for an example of the whole construction. We now show that  $H$  is a 5-spanner of size  $O(n^{4/3} \log^{2/3} n)$ .

► **Lemma 2.** *Let  $u, v \in V$ . There is a temporal path from  $u$  to  $v$  of length at most 5 in  $H$ .*

**Proof.** Let  $x \in R_1$  be the center of the level-one cluster  $C_{1,x}$  containing  $u$  and  $y \in R_2$  be the center of the level-two cluster  $C_{2,y}$  containing  $z_1(x)$ .

We first show that in  $H$  there exists a temporal path  $\pi$  of length 4 from  $u$  to  $z_2(y)$  consisting of the sequence of edges  $(u, x)$ ,  $(x, z_1(x))$ ,  $(z_1(x), y)$ ,  $(y, z_2(y))$ . Notice that, the edges  $(u, x)$ ,  $(x, z_1(x))$ ,  $(z_1(x), y)$ ,  $(y, z_2(y))$  belong to  $E_{1,u}$ ,  $E_{1,z_1(x)}$ ,  $E_{2,z_1(x)}$ , and  $E_{2,z_2(y)}$ ,



■ **Figure 3** An example of a two-level cluster and of the edges added to  $E(H)$  during the spanner construction. The black, red and purple edges are added during initialization phase. In particular, for every  $u \in V$ , the red edges are those in  $E_{1,u}$  and, for every  $z \in Z_1$ , the purple edges are those in  $E_{2,z}$ . The dark blue and light blue edges are those added to  $E(H)$  during the first augmentation phase. The green edges are the edges added to  $E(H)$  during the second augmentation phase.

respectively. Moreover, the initialization phase ensures that they all belong to  $E(H)$ . Then, by definition of  $z_1(x)$ , we have  $\lambda(u, x) \leq \lambda(x, z_1(x))$ . Moreover, since  $(x, z_1(x)) \in E_{1,z_1(x)}$  and  $(z_1(x), y) \in E_{2,z_1(x)}$ , then  $\lambda(x, z_1(x)) \leq \lambda(z_1(x), y)$ . Finally,  $(y, z_2(y)) \in E_{2,z_2(y)}$  and, by definition of  $z_2(y)$ , we have  $\lambda(z_1(x), y) \leq \lambda(y, z_2(y))$ .

If  $v = z_2(y)$ , then  $u$  can reach  $v$  via a temporal path of length 4 in  $H$ , by using  $\pi$ . Moreover, if  $v = z_1(x)$  then  $u$  can reach  $v$  via a temporal path of length 2 by using the subpath  $\pi_1$  of  $\pi$  consisting of the edges  $(u, x)$  and  $(x, z_1(x))$ . Otherwise, we can build a temporal path to  $v$  by considering one of following three cases (to be checked in order):

- If  $(z_1(x), v) \in E_{1,z_1(x)}$ , then  $v \in S_{1,z_1(x)}$  and, due to the first augmentation phase, we have that  $(u, v) \in E(H)$ .
- If  $(z_2(y), v) \in E_{1,z_2(y)} \cup E_{2,z_2(y)}$ , then vertex  $v \in S_{1,z_2(y)} \cup S_{2,z_2(y)}$  and the first augmentation phase ensures that  $(z_1(x), v) \in E(H)$ . Moreover, since  $(z_1(x), v) \notin E_{1,z_1(x)}$ , we have that  $\lambda(x, z_1(x)) \leq \lambda(z_1(x), v)$ . Hence the concatenation of  $\pi_1$  with the edge  $(z_1(x), v)$  yields a temporal path of length 3 from  $u$  to  $v$  in  $H$ .
- If  $(z_2(y), v) \notin (E_{1,z_2(y)} \cup E_{2,z_2(y)})$ , the second augmentation phase ensures that  $(z_2(y), v) \in E(H)$ . Moreover,  $\lambda(y, z_2(y)) \leq \lambda(z_2(y), v)$ . Therefore the concatenation of  $\pi$  with the edge  $(z_2(y), v)$  yields a temporal path of length 5 from  $u$  to  $v$  in  $H$ . ◀

▶ **Lemma 3.** *The size of  $H$  is  $O(n^{4/3} \log^{2/3} n)$ .*

### 3.3 Our temporal $(2k - 1)$ -spanner

In this section, we describe an algorithm that, given an integer  $k \geq 2$  and a temporal clique  $G$  of  $n$  vertices, returns a temporal  $(2k - 1)$ -spanner of  $G$  with size  $O(k \cdot n^{1+\frac{1}{k}} \log^{\frac{k-1}{k}} n)$ .

The idea is to define a hierarchical clustering of  $G$ , where a generic level- $i$  clustering partitions the special vertices of the level- $(i - 1)$  clustering and determines the special vertices of level  $i$ . As we move from one clustering level to the next, the number of clusters decreases by a factor of roughly  $n^{\frac{1}{k}}$ , thus allowing us to add an increasing number of edges incident to the special vertices into the spanner.

We ensure that each vertex  $u \in V$  can reach some special vertex by moving upwards in the clustering hierarchy. These special vertices work as hubs, i.e., each of them allows to directly reach a subset of vertices of  $V$ , and some special vertex of higher level (via a temporal path of length at most 2). Then  $u$  can reach any vertex in  $v \in V$  by first reaching a suitable special vertex  $z$  in the hierarchy, and then following the edge  $(z, v)$ .



■ **Algorithm 1** Computes a temporal  $(2k - 1)$ -spanner.

---

```

Input : A temporal clique  $G$ ;
Output : A temporal  $(2k - 1)$ -spanner of  $G$ ;

1  $Z_0 \leftarrow V$ ;
2 foreach  $u \in V$  do  $E(u) \leftarrow \{(u, v) \mid v \in V\}$ ;
3 for  $i = 1, \dots, k - 1$  do
4   foreach  $u \in Z_{i-1}$  do
5      $E_{i,u} \leftarrow$  set of the first min-time label  $n^{\frac{i}{k}} \log n$  edges of  $E(u)$ ;
6      $E(u) \leftarrow E(u) \setminus E_{i,u}$ ;
7      $S_{i,u} \leftarrow \{v \in V : (u, v) \in E_{i,u}\}$ ;
8    $R_i \leftarrow$  hitting set of  $\{S_{i,u}\}_{u \in Z_{i-1}}$  computed as in Lemma 1;
9    $C \leftarrow \emptyset$ ; // Set of vertices in  $Z_{i-1}$  that are already clustered
10  foreach  $x \in R_i$  do
11     $C_{i,x} = \{u \in Z_i \setminus C : x \in S_{i,u}\}$ ;
12     $z_i(x) \leftarrow \arg \max_{u \in C_{i,x}} \{\lambda(u, x)\}$ ;
13     $C \leftarrow C \cup C_{i,x}$ ;
14   $Z_i \leftarrow \{z_i(x) \in Z_{i-1} : x \in R_i\}$ ;
15  $H \leftarrow (V, \emptyset)$  for  $i = 1$  to  $k - 1$  do // Initialization
16   foreach  $u \in Z_{i-1}$  do  $E(H) \leftarrow E(H) \cup E_{i,u}$ ;
17 for  $i = 1, \dots, k - 1$  do // First augmentation
18   foreach  $u \in Z_{i-1}$  do
19     Let  $x \in R_i$  such that  $u \in C_{i,x}$ ;
20      $E(H) \leftarrow E(H) \cup \{u\} \times \{\bigcup_{j=1}^i S_{z_i(x),j}\}$ ;
21 foreach  $z \in Z_{k-1}$  do  $E(H) \leftarrow E(H) \cup \{z\} \times V$ ; // Second Augmentation
22 return  $H$ ;
```

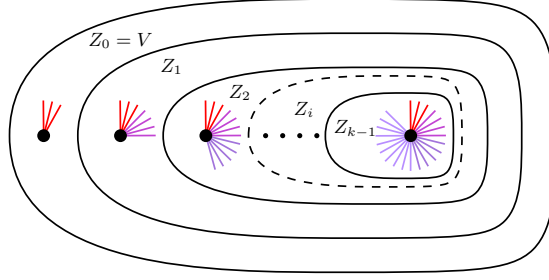
---

We build our clustering in  $k - 1$  rounds indexed from 1 to  $k - 1$  (a detailed pseudocode is given in Algorithm 1), where the generic  $i$ -th round defines a set  $Z_i$  of level- $i$  special vertices. Initially,  $Z_0 = V$ , i.e., all vertices are special vertices of level 0. During the  $i$ -th round, the level- $i$  clustering is computed from the set of vertices in  $Z_{i-1}$  defined at the previous round as follows.

For each  $u \in Z_{i-1}$ , we let  $E_{i,u}$  be a set of  $\delta_i = \Theta(n^{\frac{i}{k}} \log^{\frac{k-i}{k}} n)$  edges with the smallest label among those that are incident to  $u$  but do not belong to  $\bigcup_{j=1}^{i-1} E_{u,j}$ , and we denote by  $S_{i,u} = \{v \in V : (u, v) \in E_{i,u}\}$  the set containing the endvertices of the edges incident to  $u$  in  $E_{i,u}$ . We now compute a hitting set  $R_i \subseteq V$  of the collection  $\{S_{i,u} \mid u \in Z_{i-1}\}$  having size at most  $O(\frac{n}{\delta_i} \log n)$ . Lemma 1 guarantees that  $R_i$  always exists. Notice that, as  $i$  increases, the time labels of the edges in  $E_{i,u}$  became larger,  $\delta_i$  increases, and  $|R_i|$  decreases.

We now partition the vertices in  $Z_{i-1}$  into  $|R_i|$  clusters  $C_{i,x}$ , one for each  $x \in R_i$ . We do so by adding each vertex  $u \in Z_{i-1}$  into an arbitrary cluster  $C_{i,x}$  such that  $x \in S_{i,u}$ . We call  $x$  the center of the cluster  $C_{i,x}$ . Moreover, for each cluster  $C_{i,x}$ , we choose a special vertex  $z_i(x) \in C_{i,x}$  as a vertex that maximizes the label of edge  $(x, z_i(x))$ .

Once the hierarchical clustering is built, our algorithm proceeds to construct a temporal  $(2k - 1)$ -spanner  $H$  of  $G$ . At the beginning  $H = (V, \emptyset)$ , then edges are added to  $H$  in the following three phases:



■ **Figure 4** The set of edges selected for each vertex during the initialization phase.

**Initialization:** For each  $u \in V$ , we add to  $E(H)$  all the edges in the sets  $E_{i,u}$  for  $i = 1, \dots, j+1$ , where  $j$  is the largest integer between 0 and  $k-2$  for which  $u \in Z_j$ , see Figure 4.

**First Augmentation:** For each  $i = 1, \dots, k-1$  and each  $u \in Z_{i-1}$ , we consider the center  $x \in R_i$  of the level- $i$  cluster  $C_{i,x}$  containing  $u$ , and we add to  $E(H)$  all the edges  $(u, v)$  with  $v \in \bigcup_{j=1}^i S_{j,z_i(x)}$ .

**Second Augmentation:** We add to  $E(H)$  all edges incident to some vertex in  $Z_{k-1}$ .

We now show that all vertices are at distance at most  $2k-1$  in  $H$ , and that the size of  $H$  is  $O(k \cdot n^{1+\frac{1}{k}} \log^{\frac{k-1}{k}} n)$ .

► **Lemma 4.** *For every  $u, v \in V(G)$ ,  $d_H(u, v) \leq (2k-1)d_G(u, v)$ .*

**Proof.** Let  $z_0 = u$  and, for  $i = 1, \dots, k-1$ , let  $z_i = z_i(x_i)$  where  $x_i \in R_i$  is the center of the cluster  $C_{i,x_i}$  containing  $z_{i-1}$ . The initialization phase ensures that, for any  $i$ , there exists a temporal path from  $z_0$  to  $z_i$  in  $H$  of length  $2i$  entering  $z_i$  with the edge  $(x_i, z_i) \in E_{i,z_i}$ .<sup>5</sup> Indeed,  $\pi_i$  can be chosen as the path that traverses edge  $(z_{i-1}, x_i) \in E_{i,z_{i-1}}$  and edge  $(x_i, z_i) \in E_{i,z_i}$ , in this order. Notice that, by definition of  $z_i$ ,  $\lambda(z_{i-1}, x_i) \leq \lambda(x_i, z_i)$ . Moreover, if  $i < k-1$ ,  $\lambda(x_i, z_i) \leq \lambda(z_i, x_{i+1})$  since  $(x_i, z_i) \in E_{i,z_i}$  while  $(z_i, x_{i+1}) \in E_{i+1,z_i}$ . See Figure 5.

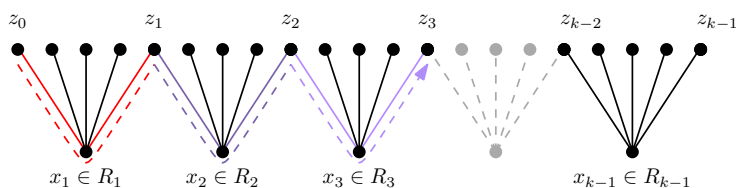
If  $v = z_i$  for some  $i = 1, \dots, k-1$  then, from the discussion above, we know that  $\pi_i$  is a temporal path from  $u$  to  $v$  in  $H$  of length  $2i < 2k-1$ . Otherwise, we distinguish two cases depending on whether there exists some  $i = 1, \dots, k-1$  such that  $(z_i, v) \in \bigcup_{j=1}^i E_{j,z_i}$ .

Suppose that the above condition is met, and let  $i > 0$  be the minimum index for which  $(z_i, v) \in \bigcup_{j=1}^i E_{j,z_i}$ . If  $\lambda(z_i, v) \geq \lambda(x_i, z_i)$ , then  $\pi_i$  followed by edge  $(z_i, v)$ , is a temporal path from  $u$  to  $v$  of length  $2i+1 \leq 2k-1$ . If  $\lambda(z_i, v) < \lambda(x_i, z_i)$  then, since  $(z_i, v) \in \bigcup_{j=1}^i E_{j,z_i}$ , we have  $v \in \bigcup_{j=1}^i S_{j,z_i}$  and the first augmentation phase adds  $(z_{i-1}, v)$  to  $E(H)$ . By hypothesis we have  $(z_{i-1}, v) \notin \bigcup_{j=1}^{i-1} E_{j,z_{i-1}}$  and hence  $\lambda(z_{i-1}, v) \geq \lambda(x_{i-1}, z_{i-1})$ . This shows that  $\pi_{i-1}$  followed by  $(z_{i-1}, v)$  is a temporal path from  $u$  to  $v$  in  $H$  of length  $2i-1 \leq 2k-1$ .

It only remains to handle the case in which, for every  $i$ , we have  $(z_i, v) \notin \bigcup_{j=1}^i E_{j,z_i}$ . In this case, the algorithm adds  $(z_{k-1}, v)$  to  $E(H)$  during the second augmentation phase. Moreover, since  $\lambda(z_{k-1}, v) \geq \lambda(x_{k-1}, z_{k-1})$ , the path  $\pi_{k-1}$  followed by edge  $(z_{k-1}, v)$  is a temporal path from  $u$  to  $v$  in  $H$  of length  $2k-1$ . ◀

► **Theorem 5.** *Given a temporal clique  $G$ , for any  $k \geq 1$ , the above algorithm computes a temporal  $(2k-1)$ -spanner  $H$  of size  $O(k \cdot n^{1+\frac{1}{k}} \log^{\frac{k-1}{k}} n)$ .*

<sup>5</sup> This path is not necessarily a simple path (e.g., when  $z_i = z_{i+1}$ ). The existence of a non-simple temporal path of length  $\ell$  implies the existence of simple temporal path of length at most  $\ell$ .



■ **Figure 5** The hierarchy of clusters for vertex  $z_0$ , where  $z_i = z_i(x_i)$  where  $x_i \in R_i$  is such that  $z_{i-1} \in C_{i,x_i}$ . The dashed line is the temporal path  $\pi_3$  that goes from  $z_0$  to  $z_3$ .

We conclude this section with a simple lower bound on the size of any temporal 2-spanners of a temporal clique.

► **Theorem 6.** *There exists a temporal clique  $G$  of  $n$  vertices such that any temporal 2-spanner of  $G$  has size  $\Omega(n^2)$ .*

## 4 Single-source spanners for general temporal graphs

In the first part of this section we design an algorithm that, for every  $0 < \varepsilon < n$ , builds a single-source temporal  $(1 + \varepsilon)$ -spanner of  $G$  w.r.t.  $s$  of size  $O\left(\frac{n \log^4 n}{\log(1+\varepsilon)}\right)$ . We observe that, for constant values of  $\varepsilon$ , the size of the computed spanner is almost linear, i.e., linear up to polylogarithmic factors. The algorithm can be extended so as, for every  $1 \leq \beta < n$ , it builds a single-source temporal  $\beta$ -additive spanner of  $G$  w.r.t.  $s$  of size  $O\left(\frac{n^2 \log^4 n}{\beta}\right)$ .<sup>6</sup>

Our upper bounds leave open the problem of deciding whether a temporal graph  $G$  admits a single-source temporal preserver w.r.t.  $s$  of size  $\tilde{O}(n)$ . We answer to this question negatively in the second part of this section. More precisely, we show a temporal graph  $G$  of size  $\Theta(n^2)$  and a source vertex  $s$  for which no edge can be removed if we want to keep a shortest temporal path from  $s$  to every other vertex  $u$ . The construction can be extended to show a lower bound of  $\Omega(n^2/\beta)$  on the size of single-source temporal  $\beta$ -additive spanners, for every  $\beta \geq 1$ . This implies that our upper bound on the size of single-source temporal additive spanners is asymptotically optimal, up to polylogarithmic factors.

### 4.1 Our upper bound

In this section we present an algorithm that, for every  $0 < \varepsilon < n$ , computes a single-source temporal  $(1 + \varepsilon)$ -spanner of  $G$  w.r.t.  $s$  of size  $O\left(\frac{n \log^4 n}{\log(1+\varepsilon)}\right)$  in polynomial time.<sup>7</sup>

In the following we say that a temporal path is  $\tau$ -restricted if it uses edges of time-label of at most  $\tau$ . Our algorithm computes a spanner that, for every  $\tau = 1, \dots, L$ , contains  $(1 + \varepsilon)$ -approximate  $\tau$ -restricted temporal paths from  $s$  to any vertex  $v$  (recall that  $L$  is the lifetime of  $G$ ). More formally, for two vertices  $u$  and  $v$  of  $G$ , we denote by  $d_G^{\leq \tau}(u, v)$  the length of a shortest  $\tau$ -restricted temporal path from  $u$  to  $v$  in  $G$ . We assume  $d_G^{\leq \tau}(u, v) = +\infty$  when  $G$  does not contain a  $\tau$ -restricted temporal path from  $u$  to  $v$ . The single-source temporal  $(1 + \varepsilon)$ -spanner  $H$  of  $G$  w.r.t.  $s$  computed by our algorithm is such that, for every  $v \in V$ , and for every  $\tau = 1, \dots, L$ ,  $d_H^{\leq \tau}(s, v) \leq (1 + \varepsilon)d_G^{\leq \tau}(s, v)$ .

<sup>6</sup> We refer the interested reader to the full version of this paper for details.

<sup>7</sup> Our algorithm also works in the case of directed temporal graphs and/or multiple time-labels. Both the algorithm and the stretch analysis require no modification. Regarding the running time, we only need to observe that  $\tau$ -restricted shortest paths can be computed in polynomial time even in directed/multiple-label temporal graphs.

## 19:12 Sparse Temporal Spanners with Low Stretch

■ **Algorithm 2** Computes a set  $\Pi_v$  of temporal paths from  $s$  to  $v$  in  $G$  that provides a good approximation of any shortest  $\tau$ -restricted temporal path from  $s$  to  $v$  in  $G$ .

---

**Input** : A temporal graph  $G$  (with lifetime  $L$ ), a source vertex  $s \in V$ , and a vertex  $v \in V$ .  
**Output** : A set  $\Pi_v$  of temporal paths from  $s$  to  $v$  in  $G$  such that, for every  $\tau = 1, \dots, L$ , there exists a  $\tau$ -restricted temporal path  $\pi \in \Pi_v$  such that  $|\pi| \leq (1 + \delta)d_G^{\leq \tau}(s, v)$ .

---

```

1  $\Pi_v \leftarrow \emptyset$ ;  $t \leftarrow +\infty$ ;
2 for  $\tau = 1$  to  $L$  do
3   if  $d_G^{\leq \tau}(s, v) \neq +\infty$  and  $d_G^{\leq \tau}(s, v) < \frac{t}{1+\delta}$  then
4     Let  $\pi$  be a shortest  $\tau$ -restricted temporal path from  $s$  to  $v$  in  $G$ ;
5      $\Pi_v \leftarrow \Pi_v \cup \{\pi\}$ ;
6      $t \leftarrow |\pi|$ ;
7 return  $\Pi_v$ ;
```

---

For technical convenience, in the following we design an algorithm that, for any  $0 < \delta < n$  and any positive integer  $k$ , builds a single-source temporal  $(1 + \delta)^k$ -spanner of  $G$  w.r.t.  $s$  of size  $O\left(\frac{kn^{1+\frac{1}{k}} \log^{2-\frac{1}{k}} n}{\log(1+\delta)}\right)$ . The desired bound of  $O\left(\frac{n \log^4 n}{\log(1+\varepsilon)}\right)$  on the size of the single-source temporal  $(1 + \varepsilon)$ -spanner is obtained by choosing  $k = \lceil \log n \rceil$  and  $\delta = (1 + \varepsilon)^{\frac{1}{\log n}} - 1$ .

Our algorithm uses a subroutine that, for a given vertex  $v$  of  $G$ , computes a set  $\Pi_v$  of  $O\left(\frac{\log n}{\log(1+\delta)}\right)$  temporal paths from  $s$  to  $v$  of  $G$  such that, for every  $\tau = 1, \dots, L$ ,  $\Pi_v$  contains a  $\tau$ -restricted temporal path  $\pi$  satisfying  $|\pi| \leq (1 + \delta)d_G^{\leq \tau}(s, v)$ .

The subroutine (see Algorithm 2 for the pseudocode) builds  $\Pi_v$  iteratively by adding a subset of shortest  $\tau$ -restricted temporal paths from  $s$  to  $v$  in  $G$ , where  $\tau = 1, \dots, L$ . We do so by scanning shortest  $\tau$ -restricted temporal paths from  $s$  to  $v$  in increasing order of values of  $\tau$ . The scanned path  $\pi$  is added to  $\Pi_v$  if no other path already contained in  $\Pi_v$  has a length of at most  $(1 + \delta)|\pi|$ . The next lemma shows the correctness of our subroutine and bounds the number of paths contained in  $\Pi_v$ .

► **Lemma 7.** *For every  $\tau = 1, \dots, L$ , there is a  $\tau$ -restricted temporal path  $\pi$  in  $\Pi_v$  such that  $|\pi| \leq (1 + \delta)d_G^{\leq \tau}(s, v)$ . Moreover,  $|\Pi_v| = O\left(\frac{\log n}{\log(1+\delta)}\right)$ .*

In the rest of this section, for any given temporal path  $\pi$ , we denote by  $\pi(\ell)$  the subpath of  $\pi$  containing the last  $\min\{\ell, |\pi|\}$  edges of  $\pi$ . We observe that  $\pi(\ell) = \pi$  when  $|\pi| \leq \ell$ . Moreover, for two vertices  $u$  and  $v$  of a temporal path  $\pi$  that visits  $u$  before  $v$ , we denote by  $\pi[u, v]$  the temporal subpath of  $\pi$  from  $u$  to  $v$ .

Before diving into the technical details, we describe the main idea of our algorithm and show how we can use it to build a single-source temporal  $(1 + \delta)^2$ -spanner of  $G$  w.r.t.  $s$  of size  $O\left(\frac{n^{3/2} \log^{3/2} n}{\log(1+\delta)}\right)$ .

For technical convenience, let  $R_0 = V$  and  $\mathcal{P}_0 = \bigcup_{v \in R_0} \Pi_v$ . In principle, we could build our single-source temporal  $(1 + \delta)$ -spanner of  $G$  w.r.t.  $s$  by simply setting its edge set to  $\bigcup_{\pi \in \mathcal{P}_0} E(\pi)$ . Unfortunately, Lemma 7 alone is insufficient to provide a subquadratic upper bound on the size of this spanner. Therefore, to obtain a spanner of truly subquadratic size, we compute a single-source temporal  $(1 + \delta)^2$ -spanner  $H$  of  $G$  w.r.t.  $s$  instead.

We build  $H$  by adding all the *short* temporal paths in  $\mathcal{P}_0$ , i.e., all paths with at most  $\ell_0$  edges for a suitable choice of  $\ell_0$ , and by replacing each *long* temporal path  $\pi \in \mathcal{P}_0$  from  $s$  to some vertex  $v$  with the shortest temporal path from  $s$  to  $x$  in  $\Pi_x$ , for some vertex  $x$  that *hits*  $\pi(\ell_0)$ , combined with  $\pi[x, v]$ .

■ **Algorithm 3** Single-source temporal spanner of a temporal graph  $G$ .

---

**Input** : A temporal graph  $G = (V, E)$  of  $n$  vertices and a source vertex  $s \in V$ .

**Output** : A single-source temporal spanner  $H$  of  $G$  w.r.t.  $s$ .

```

1 for  $i = 0, \dots, k-1$  do  $\ell_i \leftarrow n^{\frac{i+1}{k}} \log^{1-\frac{i+1}{k}} n$ ;
2 foreach  $v \in V$  do use Algorithm 2 to compute  $\Pi_v$ ;
3  $R_0 \leftarrow V$ ;  $\mathcal{P}_0 \leftarrow \{\pi \in \Pi_v \mid v \in R_0\}$ ;
4 for  $i = 1, \dots, k-1$  do
5    $\mathcal{P}_{i-1}^{long} = \{\pi \in \mathcal{P}_{i-1} \mid |\pi| > \ell_i\}$ ;
6    $R_i \leftarrow$  hitting set of  $\{\pi(\ell_{i-1}) \mid \pi \in \mathcal{P}_{i-1}^{long}\}$  computed as in Lemma 1;
7    $\mathcal{P}_i \leftarrow \bigcup_{v \in R_i} \Pi_v$ ;
8 return  $H = \left( V, \bigcup_{i=0}^{k-1} \bigcup_{\pi \in \mathcal{P}_i} E(\pi(\ell_i)) \right)$ ;

```

---

In more details, we define  $\ell_0 = \sqrt{n \log n}$  and we introduce a new parameter  $\ell_1 = n$ . We say that a temporal path  $\pi \in \mathcal{P}_0$  is *short* if  $|\pi| \leq \ell_0$ ; it is *long* otherwise. Let  $\mathcal{P}_0^{long} = \{\pi \in \mathcal{P}_0 \mid |\pi| > \ell_0\}$  be the subset of long temporal paths in  $\mathcal{P}_0$ . We compute a set  $R_1$  that hits  $\{\pi(\ell_0) \mid \pi \in \mathcal{P}_0^{long}\}$  using Lemma 1, and we then use this set to define a new collection of temporal paths  $\mathcal{P}_1 = \bigcup_{v \in R_1} \Pi_v$ .<sup>8</sup> The edge set of  $H$  is defined as  $E(H) = \bigcup_{i \in \{0,1\}} \bigcup_{\pi \in \mathcal{P}_i} E(\pi(\ell_i))$ . The next lemma shows that this simple algorithm already computes a single-source temporal  $(1 + \delta)^2$ -spanner of  $G$  w.r.t.  $s$  of truly subquadratic size.

► **Lemma 8.** *For every  $\tau = 1, \dots, L$  and every  $v \in V$ ,  $d_H^{\leq \tau}(s, v) \leq (1 + \delta)^2 d_G^{\leq \tau}(s, v)$ . Moreover, the size of  $H$  is  $O\left(\frac{n^{3/2} \log^{3/2} n}{\log(1+\delta)}\right)$ .*

The technique we used to replace each of the temporal paths in  $\mathcal{P}_0^{long}$  with a temporal path that is longer by a factor of at most  $(1 + \delta)$  can be applied recursively on the set  $\mathcal{P}_1$ , for a suitable choice of  $\ell_1$ , to obtain an even sparser spanner. As we show now,  $k-1$  levels of recursion allow us to compute a single-source temporal  $(1 + \delta)^k$ -spanner  $H$  of  $G$  w.r.t.  $s$  of size  $O\left(\frac{kn^{1+\frac{1}{k}} \log^{2-\frac{1}{k}} n}{\log(1+\delta)}\right)$ .

In the following we provide the technical details (see Algorithm 3 for the pseudocode). For every  $i = 0, \dots, k-1$ , let  $\ell_i = n^{\frac{i+1}{k}} \log^{1-\frac{i+1}{k}} n$ . As before, let  $R_0 = V$  and  $\mathcal{P}_0 = \bigcup_{v \in R_0} \Pi_v$ . During the  $i$ -th iteration, the algorithm computes a set  $R_i$  that hits  $\{\pi(\ell_{i-1}) \mid \pi \in \mathcal{P}_{i-1}^{long}\}$ , where  $\mathcal{P}_{i-1}^{long} = \{\pi \in \mathcal{P}_{i-1} \mid |\pi| > \ell_{i-1}\}$  is the set of *long* temporal paths of  $\mathcal{P}_{i-1}$ . The  $i$ -th iteration ends by computing the set  $\mathcal{P}_i = \bigcup_{v \in R_i} \Pi_v$  that is used in the next iteration. The edge set of the graph  $H$  that is returned by the algorithm is  $E(H) := \bigcup_{i=0}^{k-1} \bigcup_{\pi \in \mathcal{P}_i} E(\pi(\ell_i))$ .

► **Theorem 9.** *For every  $\tau = 1, \dots, L$ , for every  $i = 1, \dots, k$ , and for every  $v \in R_{k-i}$ , we have that  $d_H^{\leq \tau}(s, v) \leq (1 + \delta)^i d_G^{\leq \tau}(s, v)$ . Moreover, the size of  $H$  is  $O\left(\frac{kn^{1+\frac{1}{k}} \log^{2-\frac{1}{k}} n}{\log(1+\delta)}\right)$ .*

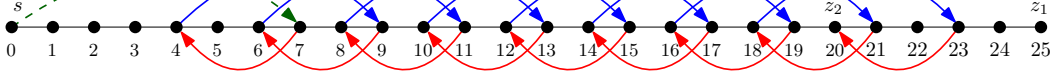
**Proof.** We start proving the first part of the theorem statement. The proof is by induction on  $i$ . Fix a vertex  $v \in R_{k-i}$  such that  $d_G^{\leq \tau}(s, v)$  is finite.

For the base case  $i = 1$ , we observe that  $\Pi_v$  is entirely contained in  $H$  by construction. Therefore, by Lemma 7,  $d_H^{\leq \tau}(s, v) \leq (1 + \delta) d_G^{\leq \tau}(s, v)$  and the claim follows.

We now prove the inductive case. We assume that the claim holds for  $i-1$  and we prove it for  $i$ . Let  $\pi \in \Pi_v$  be a shortest  $\tau$ -restricted temporal path from  $s$  to  $v$  among those in  $\Pi_v$ . By Lemma 7,  $|\pi| \leq (1 + \delta) d_G^{\leq \tau}(s, v)$ . Moreover, by definition,  $\pi \in \mathcal{P}_{k-i}$ . If  $\pi$  is short, i.e.,

---

<sup>8</sup> With a little abuse of notation,  $R_1$  hits a temporal path  $\pi$  if  $R_1$  hits  $V(\pi)$ .



■ **Figure 6** Example of the lower bound with  $h = 2$  and  $\beta = 0$ . Path  $\pi_1$  consists of the black edges. Path  $\pi_2$  is built on top of  $\pi_1$ , whose vertices have been numbered in order of a traversal from  $s$ , and consists of: the first hop from  $s$  to  $\mu$  (in green), the forward hops (in blue), and the backward hops (in red). The arrows on the edges of  $\pi_2$  are directed away from  $s$  along  $\pi_2$ .

$|\pi| \leq \ell_{k-i}$ , then  $\pi$  is entirely contained in  $H$  and therefore  $d_H^{\leq \tau}(s, v) \leq (1 + \delta)d_G^{\leq \tau}(s, v) \leq (1 + \delta)^{k-i}d_G^{\leq \tau}(s, v)$ . So, in the following we assume that  $\pi$  is long. Let  $x \in R_{k-i+1}$  be a vertex that hits  $\pi(\ell_{k-i})$ . By construction, the path  $\pi[x, v]$ , being a subpath of  $\pi(\ell_{k-i})$ , is entirely contained in  $H$ . Let  $\tau'$  be the label of the edge incident to  $x$  in  $\pi[x, v]$ . Clearly,  $\tau' \leq \tau$ . Moreover, by inductive hypothesis,  $d_H^{\leq \tau'}(s, x) \leq (1 + \delta)^{i-1} \cdot |\pi[s, x]|$ .

As a consequence,  $d_H^{\leq \tau}(s, v) \leq d_H^{\leq \tau'}(s, x) + |\pi[x, v]| \leq (1 + \delta)^{i-1} \cdot |\pi[s, x]| + |\pi[x, v]| \leq (1 + \delta)^{i-1} \cdot |\pi| \leq (1 + \delta)^i d_G^{\leq \tau}(s, v)$ .

To bound the size of  $H$ , we first observe that, for each  $v \in V$ ,  $|\Pi_v| = O\left(\frac{\log n}{\log(1+\delta)}\right)$  by Lemma 7. Next, using Lemma 1, we observe that each  $R_i$ , with  $i \geq 1$ , has size  $|R_i| = O\left(\frac{n \log n}{\ell_{i-1}}\right) = O\left(n^{1-\frac{i}{k}} \log^{\frac{i}{k}} n\right)$ . Furthermore, also  $|R_0| = n = n^{1-\frac{0}{k}} \log^{\frac{0}{k}} n$ . Therefore, for every  $i = 0, \dots, k-1$ , we have  $|R_i| \ell_i = O\left(n^{1+\frac{1}{k}} \log^{1-\frac{1}{k}} n\right)$ . As a consequence,

$$\sum_{\pi \in \mathcal{P}_i} |\pi(\ell_i)| = \sum_{v \in R_i} \sum_{\pi \in \Pi_v} |\pi(\ell_i)| = O\left(|R_i| \ell_i \frac{\log n}{\log(1+\delta)}\right) = O\left(\frac{n^{1+\frac{1}{k}} \log^{2-\frac{1}{k}}}{\log(1+\delta)}\right).$$

Hence,  $|E(H)| = \sum_{i=0}^{k-1} \sum_{\pi \in \mathcal{P}_i} |\pi(\ell_i)| = O\left(\frac{kn^{1+\frac{1}{k}} \log^{2+\frac{1}{k}}}{\log(1+\delta)}\right)$ . ◀

The following corollary follows by choosing  $\tau = L$  and  $i = k$  (so that  $R_{k-i} = R_0 = V$ ):

► **Corollary 10.** *Let  $G$  be a temporal graph with  $n$  vertices and let  $s$  be a vertex of  $G$ . The graph  $H$  returned by Algorithm 3 is a single-source temporal  $(1 + \delta)^k$ -spanner of  $G$  w.r.t.  $s$  of size  $O\left(\frac{kn^{1+\frac{1}{k}} \log^{2+\frac{1}{k}}}{\log(1+\delta)}\right)$ .*

## 4.2 Our lower bound

In this section we show that, for every  $\beta \geq 0$ , there is a temporal graph  $G$  of  $n$  vertices for which the size of any single-source temporal  $\beta$ -additive spanner of  $G$  w.r.t.  $s$  is  $\Omega\left(\frac{n^2}{1+\beta}\right)$ . This gives a lower bound of  $\Omega(n^2)$  for the size of a single-source temporal preserver.

The temporal graph  $G$  has  $n = (13 + \beta)h$  vertices, where  $h$  is an integer, and is formed by the union of  $h$  pairwise edge-disjoint temporal paths  $\pi_1, \dots, \pi_h$ . Each path  $\pi_i$  goes from  $s$  to a vertex  $z_i$  and has length  $\Omega(n - i(1 + \beta))$ . The construction guarantees that the unique temporal path of  $G$  from  $s$  to  $z_i$  of length of at most  $d_G(s, z_i) + \beta$  is  $\pi_i$ . This implies that the size of  $G$  is  $\Omega\left(\frac{n^2}{1+\beta}\right)$ , as desired.

The temporal path  $\pi_1$  is a Hamiltonian path that spans all the  $n$  vertices of  $G$  and goes from  $s$  to  $z_1$ . All edges of  $\pi_1$  have time-label 1. The remaining temporal paths are defined recursively. More precisely, for each  $i = 2, \dots, h$ , the temporal path  $\pi_i$  is defined on top of the temporal path  $\pi_{i-1}$  as follows. Let us number the vertices visited in a traversal of  $\pi_{i-1}$  from  $s$  to  $z_{i-1}$  in order from 0 to  $|\pi_{i-1}| - 1$ . The temporal path  $\pi_i$  is defined as a sequence of hops over the vertices of  $\pi_{i-1}$ . We call *offset* a value  $\mu$  that is equal to  $\beta + 7$  for even values of  $\beta$ , and to  $\beta + 8$  for odd values of  $\beta$ . The first hop is the one from  $s$  to vertex  $\mu$ ,



if it exists. The rest of the path is given by a maximal alternating sequence of *backward* and *forward* hops that do not visit  $z_{i-1}$ . A generic backward hop goes from vertex  $j$ , with  $j$  odd, to vertex  $j - 3$ , while a generic forward hop goes from vertex  $j$ , with  $j$  even, to vertex  $j + 5$ . All the edges of  $\pi_i$  have time-label  $i$ . A pictorial example of the definition of  $\pi_i$  is given in Figure 6. The choice of odd values for the offset is a necessary condition to have pairwise edge-disjoint paths, while the dependency of the offset on  $\beta$  guarantees that  $\pi_i$  is the unique temporal path from  $s$  to  $z_i$  in  $G$  such that  $|\pi_i| \leq d_G(s, z_i) + \beta$ . Finally, the alternating sequence of backward and forward hops guarantees that  $|\pi_i| = \Omega(n - i(1 + \beta))$ . The above discussion yields the following theorem, and a corollary for the case  $\beta = 0$ .

► **Theorem 11.** *For every positive integer  $n$  and every  $\beta \geq 0$ , there is a temporal graph  $G$  of  $n$  vertices and a source vertex  $s$  of  $G$  such that any single-source temporal  $\beta$ -additive spanner of  $G$  w.r.t.  $s$  has size  $\Omega\left(\frac{n^2}{1+\beta}\right)$ .*

► **Corollary 12.** *For every positive integer  $n$ , there is a temporal graph  $G$  of  $n$  vertices such that any single-source temporal preserver of  $G$  w.r.t.  $s$  has size  $\Theta(n^2)$ .*

## 5 Conclusions

In this paper we addressed the size-stretch trade-offs for temporal spanners. We showed that a temporal clique admits a temporal  $(2k - 1)$ -spanner of size  $\tilde{O}(n^{1+\frac{1}{k}})$ , which implies a spanner having size  $\tilde{O}(n)$  and stretch  $O(\log n)$ . The previous best-known result was the temporal-spanner of [7] which only preserves temporal connectivity between vertices. Our construction guarantees  $O(\log n)$ -approximate distances at the cost of only an additional  $O(\log n)$  multiplicative factor on the size. We also considered the single-source case for *general* temporal graphs, where we provided almost-tight size-stretch trade-offs, along with the special case of temporal graphs with bounded lifetime.

The main problem that remains open is understanding whether better trade-offs are achievable for temporal cliques. In particular, no superlinear lower bounds are known even for the case of 3-spanners.

Finally, as we already mentioned, temporal graphs admit other natural notions of distances between vertices (which have been used, e.g., in [10, 12, 13]). The most commonly used distances are the *earliest arrival time*, the *latest departure time*, the *fastest time* (i.e., the smallest difference between the arrival and departure time of a temporal path from  $u$  to  $v$ ), and – if each edge has an associated travel time – the *shortest time* distance (i.e., the minimum sum of the travel times of the edges of a temporal path from  $u$  to  $v$ ). One can wonder whether sparse temporal spanners with low stretch are attainable also in the case of the above distances. Unfortunately the answer is negative and strong lower bounds on the size of temporal  $\alpha$ -spanners for temporal cliques can be shown even for large values of  $\alpha$ , as we discuss in the full version of the paper.

---

## References

- 1 Abu Reyan Ahmed, Greg Bodwin, Faryad Darabi Sahneh, Keaton Hamm, Mohammad Javad Latifi Jebelli, Stephen G. Kobourov, and Richard Spence. Graph spanners: A tutorial review. *Comput. Sci. Rev.*, 37:100253, 2020. doi:10.1016/j.cosrev.2020.100253.
- 2 Eleni C. Akrida, Leszek Gasieniec, George B. Mertzios, and Paul G. Spirakis. The complexity of optimal design of temporally connected graphs. *Theory Comput. Syst.*, 61(3):907–944, 2017. doi:10.1007/s00224-017-9757-x.
- 3 Ingo Althöfer, Gautam Das, David P. Dobkin, Deborah Joseph, and José Soares. On sparse spanners of weighted graphs. *Discret. Comput. Geom.*, 9:81–100, 1993. doi:10.1007/BF02189308.

- 4 Kyriakos Axiotis and Dimitris Fotakis. On the size and the approximability of minimum temporally connected subgraphs. In Ioannis Chatzigiannakis, Michael Mitzenmacher, Yuval Rabani, and Davide Sangiorgi, editors, *43rd International Colloquium on Automata, Languages, and Programming, ICALP 2016, July 11-15, 2016, Rome, Italy*, volume 55 of *LIPICs*, pages 149:1–149:14. Schloss Dagstuhl – Leibniz-Zentrum für Informatik, 2016. doi:10.4230/LIPICs.ICALP.2016.149.
- 5 Surender Baswana and Sandeep Sen. A simple and linear time randomized algorithm for computing sparse spanners in weighted graphs. *Random Struct. Algorithms*, 30(4):532–563, 2007. doi:10.1002/rsa.20130.
- 6 Davide Bilò, Fabrizio Grandoni, Luciano Gualà, Stefano Leucci, and Guido Proietti. Improved purely additive fault-tolerant spanners. In Nikhil Bansal and Irene Finocchi, editors, *Algorithms - ESA 2015 - 23rd Annual European Symposium, Patras, Greece, September 14-16, 2015, Proceedings*, volume 9294 of *Lecture Notes in Computer Science*, pages 167–178. Springer, 2015. doi:10.1007/978-3-662-48350-3\_15.
- 7 Arnaud Casteigts, Joseph G. Peters, and Jason Schoeters. Temporal cliques admit sparse spanners. *J. Comput. Syst. Sci.*, 121:1–17, 2021. doi:10.1016/j.jcss.2021.04.004.
- 8 Arnaud Casteigts, Michael Raskin, Malte Renken, and Viktor Zamaraev. Sharp thresholds in random simple temporal graphs. In *Proceedings of the 62nd Annual IEEE Symposium on Foundations of Computer Science (FOCS)*. IEEE, 2022. Full version at arXiv:2011.03738.
- 9 Petter Holme. Temporal networks. In Reda Alhajj and Jon G. Rokne, editors, *Encyclopedia of Social Network Analysis and Mining, 2nd Edition*. Springer, 2018. doi:10.1007/978-1-4939-7131-2\_42.
- 10 Silu Huang, Ada Wai-Chee Fu, and Ruifeng Liu. Minimum spanning trees in temporal graphs. In Timos K. Sellis, Susan B. Davidson, and Zachary G. Ives, editors, *Proceedings of the 2015 ACM SIGMOD International Conference on Management of Data, Melbourne, Victoria, Australia, May 31 – June 4, 2015*, pages 419–430. ACM, 2015. doi:10.1145/2723372.2723717.
- 11 David Kempe, Jon M. Kleinberg, and Amit Kumar. Connectivity and inference problems for temporal networks. *J. Comput. Syst. Sci.*, 64(4):820–842, 2002. doi:10.1006/jcss.2002.1829.
- 12 George B. Mertzios, Othon Michail, and Paul G. Spirakis. Temporal network optimization subject to connectivity constraints. *Algorithmica*, 81(4):1416–1449, 2019. doi:10.1007/s00453-018-0478-6.
- 13 Huanhuan Wu, James Cheng, Silu Huang, Yiping Ke, Yi Lu, and Yanyan Xu. Path problems in temporal graphs. *Proc. VLDB Endow.*, 7(9):721–732, 2014. doi:10.14778/2732939.2732945.