The Complexity of Finding Small Separators in Temporal Graphs

Philipp Zschoche
Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
zschoche@tu-berlin.de

Till Fluschnik[1]
Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
till.fluschnik@tu-berlin.de

Hendrik Molter[2]
Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
h.molter@tu-berlin.de

Rolf Niedermeier
Institut für Softwaretechnik und Theoretische Informatik, TU Berlin, Germany
rolf.niedermeier@tu-berlin.de

Abstract

Temporal graphs are graphs with time-stamped edges. We study the problem of finding a small vertex set (the separator) with respect to two designated terminal vertices such that the removal of the set eliminates all temporal paths connecting one terminal to the other. Herein, we consider two models of temporal paths: paths that pass through arbitrarily many edges per time step (non-strict) and paths that pass through at most one edge per time step (strict). Regarding the number of time steps of a temporal graph, we show a complexity dichotomy (NP-hardness versus polynomial-time solvability) for both problem variants. Moreover we prove both problem variants to be NP-complete even on temporal graphs whose underlying graph is planar. We further show that, on temporal graphs with planar underlying graph, if additionally the number of time steps is constant, then the problem variant for strict paths is solvable in quasi-linear time. Finally, we introduce and motivate the notion of a temporal core (vertices whose incident edges change over time). We prove that the non-strict variant is fixed-parameter tractable when parameterized by the size of the temporal core, while the strict variant remains NP-complete, even for constant-size temporal cores.

2012 ACM Subject Classification Mathematics of computing → Paths and connectivity problems, Theory of computation → Fixed parameter tractability, Theory of computation → Problems, reductions and completeness

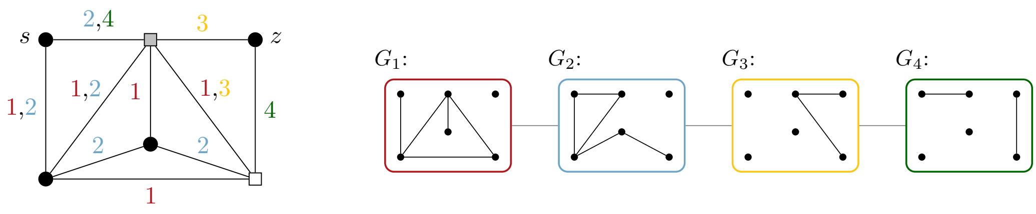
Keywords and phrases (non-)strict temporal paths, temporal core, single-source shortest paths, node multiway cut, length-bounded cuts, parameterized complexity

Digital Object Identifier 10.4230/LIPIcs.MFCS.2018.45

Related Version A full version of the paper is available at `https://arxiv.org/abs/1711.00963`.

Acknowledgements We thank anonymous reviewers for their constructive feedback which helped us to improve the presentation of this work.

[1] Supported by the DFG, project DAMM (NI 369/13) and project TORE (NI 369/18).
[2] Partially supported by the DFG, project MATE (NI 369/17).



(a) A temporal graph $\boldsymbol{G}$.

(b) Layers of $\boldsymbol{G}$.

Figure 1 Subfigure (a) shows a temporal graph $\boldsymbol{G}$ and subfigure (b) shows its four layers $G_1, \dots, G_4$. The gray squared vertex forms a strict temporal (s, z)-separator, but no temporal (s, z)-separator. The two squared vertices form a temporal (s, z)-separator.

1 Introduction

In complex network analysis, it is nowadays very common to have access to and process graph data where the interactions among the vertices are time-stamped. When using static graphs as a mathematical model, the dynamics of interactions are not reflected and important information of the data might not be captured. *Temporal graphs* address this issue. A temporal graph is, informally speaking, a graph where the edge set may change over a discrete time interval, while the vertex set remains unchanged. Having the dynamics of interactions represented in the model, it is essential to adapt definitions such as connectivity and paths to respect temporal features. This directly affects the notion of *separators* in the temporal setting. Vertex separators are a fundamental primitive in static network analysis and it is well-known that they can be computed in polynomial time (see, e.g., proof of [1, Theorem 6.8]). In contrast to the static case, Kempe et al. [25] showed that in temporal graphs it is NP-hard to compute minimum separators.

Temporal graphs are well-established in the literature and are also referred to as time-varying [27] and evolving [15] graphs, temporal networks [24, 25, 30], multidimensional networks [8], link streams [26, 36], and edge-scheduled networks [7]. In this work, we use the well-established model in which each edge has a time stamp [8, 24, 3, 22, 25, 30]. Assuming discrete time steps, this is equivalent to a sequence of static graphs over a fixed set of vertices [31]. Formally, we define a temporal graph as follows.

▶ **Definition 1.1** (Temporal Graph). An (undirected) *temporal graph* $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$ is an ordered triple consisting of a set V of vertices, a set $\boldsymbol{E} \subseteq \binom{V}{2} \times \{1, \dots, \tau\}$ of *time-edges*, and a maximal time label $\tau \in \mathbb{N}$.

See Figure 1 for an example with $\tau = 4$, that is, a temporal graph with four time steps, also referred to as *layers*. The static graph obtained from a temporal graph $\boldsymbol{G}$ by removing the time stamps from all time-edges we call the *underlying graph* of $\boldsymbol{G}$.

Many real-world applications have temporal graphs as underlying mathematical model. For instance, it is natural to model connections in public transportation networks with temporal graphs. Other examples include information spreading in social networks, communication in social networks, biological pathways, or spread of diseases [24].

A fundamental question in temporal graphs, addressing issues such as connectivity [5, 30], survivability [27], and robustness [34], is whether there is a "time-respecting" path from a distinguished start vertex s to a distinguished target vertex z.[3] We provide a thorough study of the computational complexity of separating s from z in a given temporal graph.

Moreover, we study two natural restrictions of temporal graphs:

(i) planar temporal graphs and

(ii) temporal graphs with a bounded number of vertices incident to edges that are not permanently existing – these vertices form the so-called *temporal core*.

Both restrictions are naturally motivated by settings e.g. occurring in (hierarchical) traffic networks. We also consider two very similar but still significantly differing temporal path models (both used in the literature), leading to two corresponding models of temporal separation.

Two path models. We start with the introduction of the "non-strict" path model [25]. Given a temporal graph $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$ with two distinct vertices $s, z \in V$, a *temporal* (s, z)*-path* of length ℓ in $\boldsymbol{G}$ is a sequence $P = ((\{s = v_0, v_1\}, t_1), (\{v_1, v_2\}, t_2), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell))$ of time-edges in $\boldsymbol{E}$, where $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and $t_i \leq t_{i+1}$ for all $i \in \{1, \ldots, \ell - 1\}$. A vertex set S with $S \cap \{s, z\} = \emptyset$ is a *temporal* (s, z)*-separator* if there is no temporal (s, z)-path in $\boldsymbol{G} - S := (V \setminus S, \{(\{v, w\}, t) \in \boldsymbol{E} \mid v, w \in V \setminus S\}, \tau)$. We are ready to state the central problem of our paper.

Temporal (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$, two distinct vertices $s, z \in V$, and $k \in \mathbb{N}$.

Question: Does $\boldsymbol{G}$ admit a temporal (s, z)-separator of size at most k?

Our second path model is the "strict" variant. A temporal (s, z)-path P is called *strict* if $t_i < t_{i+1}$ for all $i \in \{1, \ldots, \ell - 1\}$. In the literature, strict temporal paths are also known as journeys [3, 2, 31, 30].[4] A vertex set S is a *strict temporal* (s, z)*-separator* if there is no strict temporal (s, z)-path in $\boldsymbol{G} - S$. Thus, our second main problem, Strict Temporal (s, z)-Separation, is defined in complete analogy to Temporal (s, z)-Separation, just replacing (non-strict) temporal separators by strict ones.

While the strict version of temporal separation immediately appears as natural, the non-strict variant can be viewed as a more conservative version of the problem. For instance, in a disease-spreading scenario the spreading speed might be unclear. To ensure containment of the spreading by separating patient zero (s) from a certain target (z), a temporal (s, z)-separator might be the safer choice.

Main results. Table 1 provides an overview on our results.[5]

A central contribution is to prove that both Temporal (s, z)-Separation and Strict Temporal (s, z)-Separation are NP-complete for all $\tau \geq 2$ and $\tau \geq 5$, respectively, strengthening a result by Kempe et al. [25] (they show NP-hardness of both variants for all $\tau \geq 12$). For Temporal (s, z)-Separation, our hardness result is already tight.[6] For the strict variant, we identify a dichotomy in the computational complexity by proving

[3] In the literature the sink is usually denoted by t. To be consistent with Michail [31] we use z instead as we reserve t to refer to points in time.

[4] We also refer to Himmel [21] for a thorough discussion and comparison of temporal path concepts.

[5] Due to the space constraints, several details and proofs (marked with $\star$) are deferred to a long version of this paper, see e.g. `https://arxiv.org/abs/1711.00963`.

[6] Temporal (s, z)-Separation with $\tau = 1$ is equivalent to (s, z)-Separation on static graphs.

Table 1 Overview on our results. Herein, NP-c. abbreviates NP-complete, n and $\boldsymbol{m}$ denote the number of vertices and time-edges, respectively, $G_\downarrow$ refers to the underlying graph of an input temporal graph. [a] (Thm. 3.1; W[1]-hard wrt. k) [b] (Thm. 3.2) [c] (Cor. 4.3) [d] (Prop. 4.4) [e] (Thm. 5.2)

	General (Section 3)		Planar $G_\downarrow$ (Section 4)		Temporal core (Section 5)
(s,z)-Separation	$2 \leq \tau \leq 4$	$5 \leq \tau$	τ unbounded	τ constant	constant size
Temporal	NP-complete[a]		NP-c.[c]	*open*	$n^{O(1)} + O(\boldsymbol{m} \log \boldsymbol{m})$ [e]
Strict Temporal	$\mathcal{O}(k \cdot \boldsymbol{m})$ [b]	NP-c.[a]	NP-c.[c]	$\mathcal{O}(\boldsymbol{m} \log \boldsymbol{m})$ [d]	NP-complete[a]

polynomial-time solvability of Strict Temporal (s,z)-Separation for $\tau \leq 4$. Moreover, we prove that both problems remain NP-complete on temporal graphs that have an underlying graph that is planar.

We introduce the notion of temporal cores in temporal graphs. Informally, the temporal core of a temporal graph is the set of vertices whose edge-incidences change over time. We prove that Temporal (s,z)-Separation is fixed-parameter tractable (FPT) when parameterized by the size of the temporal core, while Strict Temporal (s,z)-Separation remains NP-complete even if the temporal core is empty.

A particular aspect of our results is that they demonstrate that the choice of the model (strict versus non-strict) for a problem can have a crucial impact on the computational complexity of said problem. This contrasts with wide parts of the literature where both models were used without discussing the subtle but crucial differences in computational complexity.

Technical contributions. To show the polynomial-time solvability of Strict Temporal (s,z)-Separation for $\tau \leq 4$, we prove that a classic separator result of Lovász et al. [28] translates to the strict temporal setting. This is surprising since many other results about separators in the static case do not apply in the temporal case. In this context, we also develop a linear-time algorithm for Single-Source Shortest Strict Temporal Paths, improving the running time of the best known algorithm due to Wu et al. [37] by a logarithmic factor.

We settle the complexity of Length-Bounded (s,z)-Separation on planar graphs by showing its NP-hardness, which was left unanswered by Fluschnik et al. [17] and promises to be a valuable intermediate problem for proving hardness results. In the hardness reduction for Length-Bounded (s,z)-Separation we introduce a grid-like, planarity-preserving vertex gadget that is generally useful to replace "twin" vertices which in many cases are not planarity-preserving and which are often used to model weights.

While showing that Temporal (s,z)-Separation is fixed-parameter tractable when parameterized by the size of the temporal core, we employ a case distinction on the size of the temporal core, and show that in the non-trivial case we can reduce the problem to Node Multiway Cut. We identify an "above lower bound parameter" for Node Multiway Cut that is suitable to lower-bound the size of the temporal core, thereby making it possible to exploit a fixed-parameter tractability result due to Cygan et al. [12].

Related work. Our most important reference is the work of Kempe et al. [25] who proved that Temporal (s,z)-Separation is NP-hard. In contrast, Berman [7] proved that computing temporal (s,z)-cuts (edge deletion instead of vertex deletion) is polynomial-time solvable. In the context of survivability of temporal graphs, Liang and Modiano [27] studied cuts where an edge deletion only lasts for δ consecutive time stamps. Moreover, they studied a temporal maximum flow defined as the maximum number of sets of journeys where each two journeys in a set do not use a temporal edge within some δ time steps. A different notion of temporal flows on temporal graphs was introduced by Akrida et al. [2]. They showed how to compute in polynomial time the maximum amount of flow passing from a source vertex s to a sink vertex z until a given point in time.

The vertex-variant of Menger's Theorem [29] states that the maximum number of vertex-disjoint paths from s to z equals the size of a minimum-cardinality (s, z)-separator. In static graphs, Menger's Theorem allows for finding a minimum-cardinality (s, z)-separator via maximum flow computations. However, Berman [7] proved that the vertex-variant of an analogue to Menger's Theorem for temporal graphs, asking for the maximum number of (strict) temporal paths instead, does not hold. Kempe et al. [25] proved that the vertex-variant of the former analogue to Menger's Theorem holds true if the underlying graph excludes a fixed minor. Mertzios et al. [30] proved another analogue of Menger's Theorem: the maximum number of strict temporal (s, z)-path which never leave the same vertex at the same time equals the minimum number of node departure times needed to separate s from z, where a node departure time (v, t) is the vertex v at time point t.

Michail and Spirakis [32] introduced the time-analogue of the famous Traveling Salesperson problem and studied the problem on temporal graphs of dynamic diameter $d \in \mathbb{N}$, that is, informally speaking, on temporal graphs where every two vertices can reach each other in at most d time steps at any time. Erlebach et al. [14] studied the same problem on temporal graphs where the underlying graph has bounded degree, bounded treewidth, or is planar. Additionally, they introduced a class of temporal graphs with regularly present edges, that is, temporal graphs where each edge is associated with two integers upper- and lower-bounding consecutive time steps of edge absence. Axiotis and Fotakis [5] studied the problem of finding the smallest temporal subgraph of a temporal graph such that single-source temporal connectivity is preserved on temporal graphs where the underlying graph has bounded treewidth. In companion work, we recently studied the computational complexity of (non-strict) temporal separation on several other restricted temporal graphs [18].

2 Preliminaries

Let $\mathbb{N}$ denote the natural numbers without zero. For $n \in \mathbb{N}$, we use $[n] := [1, n] = \{1, \ldots, n\}$.

Static graphs. We use basic notations from (static) graph theory [13]. Let $G = (V, E)$ be an *undirected, simple graph*. We use $V(G)$ and $E(G)$ to denote the set of vertices and set of edges of G, respectively. We denote by $G - V' := (V \setminus V', \{\{v, w\} \in E \mid v, w \in V \setminus V'\})$ the graph G without the vertices in $V' \subseteq V$. For $V' \subseteq V$, $G[V'] := G - (V \setminus V')$ denotes the *induced subgraph* of G by V'. A *path* of length ℓ is sequence of edges $P = (\{v_1, v_2\}, \{v_2, v_3\}, \ldots, \{v_\ell, v_{\ell+1}\})$ where $v_i \neq v_j$ for all $i, j \in [\ell + 1]$ with $i \neq j$. We set $V(P) = \{v_1, v_2, \ldots, v_{\ell+1}\}$. Path P is an (s, z)*-path* if $s = v_1$ and $z = v_{\ell+1}$. A set $S \subseteq V \setminus \{s, z\}$ of vertices is an (s, z)*-separator* if there is no (s, z)-path in $G - S$.

Temporal graphs. Let $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$ be a temporal graph. The graph $G_i(\boldsymbol{G}) = (V, E_i(\boldsymbol{G}))$ is called *layer* i of the temporal graph $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$ where $\{v, w\} \in E_i(\boldsymbol{G}) \Leftrightarrow (\{v, w\}, i) \in \boldsymbol{E}$. The *underlying graph* $G_\downarrow(\boldsymbol{G})$ of a temporal graph $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$ is defined as $G_\downarrow(\boldsymbol{G}) := (V, E_\downarrow(\boldsymbol{G}))$, where $E_\downarrow(\boldsymbol{G}) = \{e \mid (e, t) \in \boldsymbol{E}\}$. (We write G_i, E_i, $G_\downarrow$, and $E_\downarrow$ for short if $\boldsymbol{G}$ is clear from the context.) For $X \subseteq V$ we define the *induced temporal subgraph* of X by $\boldsymbol{G}[X] := (X, \{(\{v,w\},t) \in \boldsymbol{E} \mid v,w \in X\}, \tau)$. We say that $\boldsymbol{G}$ is *connected* if its underlying graph $G_{\downarrow}$ is connected. For surveys concerning temporal graphs we refer to [9, 31, 24, 26, 23].

Strict and non-strict temporal separators. Throughout the paper we assume that the underlying graph of the temporal input graph $\boldsymbol{G}$ is connected and that there is no time-edge between s and z. Furthermore, in accordance with Wu et al. [37] we assume that the time-edge set $\boldsymbol{E}$ is ordered by ascending time stamps. Moreover, we can assume that the number of layers is at most the number of time-edges:

▶ **Lemma 2.1** ($\star$). *Let $\mathcal{I} = (\boldsymbol{G} = (V, \boldsymbol{E}, \tau), s, z, k)$ be an instance of* (Strict) Temporal (s,z)-Separation. *There is an algorithm which computes in $\mathcal{O}(|\boldsymbol{E}|)$ time an instance $\mathcal{I}' = (\boldsymbol{G}' = (V, \boldsymbol{E}', \tau'), s, z, k)$ of* (Strict) Temporal (s,z)-Separation *which is equivalent to $\mathcal{I}$, where $\tau' \leq |\boldsymbol{E}'|$.*

Regarding our two models, we have the following connection:

▶ **Lemma 2.2** ($\star$). *There is a linear-time computable many-one reduction from* Strict Temporal (s,z)-Separation *to* Temporal (s,z)-Separation *that maps any instance $(\boldsymbol{G} = (V, \boldsymbol{E}, \tau), s, z, k)$ to an instance $(\boldsymbol{G}' = (V', \boldsymbol{E}', \tau'), s, z, k')$ with $k' = k$ and $\tau' = 2 \cdot \tau$.*

3 Hardness Dichotomy Regarding the Number of Layers

In this section we settle the complexity dichotomy of both Temporal (s,z)-Separation and Strict Temporal (s,z)-Separation regarding the number τ of time steps. We observe that both problems are strongly related to the following NP-complete [10, 35] problem:

Length-Bounded (s,z)-Separation (LBS)
Input: An undirected graph $G = (V, E)$, distinct vertices $s, z \in V$, and $k, \ell \in \mathbb{N}$.
Question: Is there a subset $S \subseteq V \setminus \{s,z\}$ such that $|S| \leq k$ and there is no (s,z)-path in $G - S$ of length at most ℓ?

Length-Bounded (s,z)-Separation is NP-complete even if the lower bound ℓ for the path length is five [6] and W[1]-hard with respect to the postulated separator size [20]. We obtain the following, improving a result by Kempe et al. [25] who showed NP-completeness of Temporal (s,z)-Separation and Strict Temporal (s,z)-Separation for all $\tau \geq 12$.

▶ **Theorem 3.1** ($\star$). Temporal (s,z)-Separation *is* NP*-complete for every maximum label $\tau \geq 2$ and* Strict Temporal (s,z)-Separation *is* NP*-complete for every $\tau \geq 5$. Moreover, both problems are* W[1]*-hard when parameterized by the solution size k.*

We only present the construction of the NP-hardness reduction for Temporal (s,z)-Separation, which is inspired by Baier et al. [6], and postpone the rest to the long version.

Proof (Construction). To show NP-completeness of Temporal (s,z)-Separation for $\tau = 2$ we present a reduction from the Vertex Cover problem where, given a graph $G = (V, E)$ and an integer k, the task is to determine whether there exists a set $V' \subseteq V$ of size at most k such that $G - V'$ does not contain any edge. Let $(G = (V, E), k)$ be an instance of Vertex Cover. We say that $V' \subseteq V$ is a *vertex cover* in G of size k if $|V'| = k$ and V' is a solution to $(G = (V, E), k)$. We refine the gadget of Baier et al. [6, Theorem 3.9] and reduce from Vertex Cover to Temporal (s,z)-Separation. Let $\mathcal{I} := (G = (V, E), k)$ be a Vertex Cover instance and $n := |V|$. We construct a Temporal (s,z)-Separation

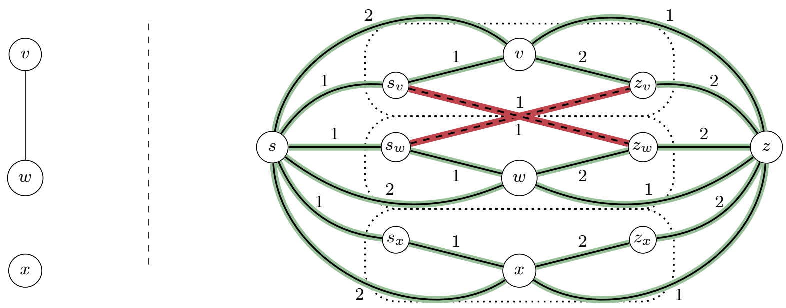


Figure 2 The Vertex Cover instance $(G, 1)$ (left) and the corresponding Temporal (s, z)-Separation instance from the reduction of Theorem 3.1 (right). The edge-edges are dashed (red), the vertex-edges are solid (green), and the vertex gadgets are in dotted boxes.

instance $\mathcal{I}' := (\mathbf{G}' = (V', \mathbf{E}', 2), s, z, n + k)$, where $V' := V \cup \{s_v, t_v \mid v \in V\} \cup \{s, z\}$ are the vertices and the time-edges are defined as

$$\mathbf{E}' := \overbrace{\{(\{s, s_v\}, 1), (\{s_v, v\}, 1), (\{v, z_v\}, 2), (\{z_v, z\}, 2), (\{s, v\}, 2), (\{v, z\}, 1) \mid v \in V\}}^{\text{vertex-edges}} \cup \underbrace{\{(\{s_v, z_w\}, 1), (\{s_w, z_v\}, 1) \mid \{v, w\} \in E\}}_{\text{edge-edges}}.$$

Note that $|V'| = 3 \cdot n + 2$, $|\mathbf{E}'| = 6 \cdot |V'| + 2 \cdot |E|$, and $\mathcal{I}'$ can be computed in polynomial time. For each vertex $v \in V$ there is a *vertex gadget* which consists of three vertices s_v, v, z_v and six vertex-edges. In addition, for each edge $\{v, w\} \in E$ there is an *edge gadget* which consists of two edge-edges $\{s_v, z_w\}$ and $\{z_v, s_w\}$. See Figure 2 for an example. ◀

In the remainder of this section we prove that the bound on τ is tight in the strict case (for the non-strict case the tightness is obvious). This is the first case where we can observe a significant difference between the strict and the non-strict variant of our separation problem.

▶ **Theorem 3.2.** Strict Temporal (s, z)-Separation *for maximum label* $\tau \leq 4$ *can be solved in* $\mathcal{O}(k \cdot |\mathbf{E}|)$ *time, where* k *is the solution size.*

As a subroutine hidden in several of our algorithms, we need to solve the Single-Source Shortest Strict Temporal Paths problem on temporal graphs: find shortest strict paths from a source vertex s to all other vertices in the temporal graph. Herein, we say that a strict temporal (s, z)-path is *shortest* if there is no strict temporal (s, z)-path of length $\ell' < \ell$. Indeed, we provide a linear-time algorithm for this. We believe this to be of independent interest; it improves (with few adaptations to the model; for details we refer to the long version) previous results by Wu et al. [37], but in contrast to the algorithm of Wu et al. [37] our subroutine cannot be adjusted to the non-strict case.

▶ **Proposition 3.3** ($\star$). Single-Source Shortest Strict Temporal Paths *is solvable in* $\Theta(|\mathbf{E}|)$ *time.*

Our algorithm behind Theorem 3.2 executes the following steps:

1. As a preprocessing step, remove unnecessary time-edges and vertices from the graph.
2. Compute an auxiliary graph called *directed path cover graph* of the temporal graph.
3. Compute a separator for the directed path cover graph.

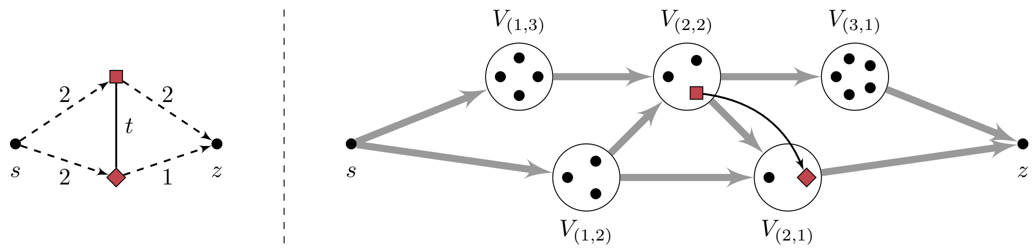


Figure 3 The left side depicts an excerpt of a reduced temporal graph with maximum time-edge label $\tau = 4$. Dashed arcs labeled with a number x indicate a shortest strict temporal path of length x. The right side depicts the directed path cover graph D from s to z of the reduced temporal graph. A gray arc from vertex set $V_{(i,j)}$ to vertex set $V_{(i',j')}$ denotes that for two vertices $v \in V_{(i,j)}$ and $w \in V_{(i',j')}$ there can be an arc from v to w in D. Take as an example the square-shaped vertex in $V_{(2,2)}$ and the diamond-shaped vertex in $V_{(2,1)}$.

In the following, we explain each of the steps in more detail.

The preprocessing *reduces* the temporal graph such that it has the following properties. A temporal graph $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$ with two distinct vertices $s, z \in V$ is *reduced* if

(i) the underlying graph $\boldsymbol{G}_\downarrow$ is connected,
(ii) for each time-edge $e \in \boldsymbol{E}$ there is a strict temporal (s, z)-path which contains e, and
(iii) there is no strict temporal (s, z)-path of length at most two in $\boldsymbol{G}$.

This preprocessing step can be performed in polynomial time:

▶ **Lemma 3.4** ($\star$). *Let $\mathcal{I} = (\boldsymbol{G} = (V, \boldsymbol{E}, \tau), s, z, k)$ be an instance of* Strict Temporal (s, z)-Separation*. In $\mathcal{O}(k \cdot |\boldsymbol{E}|)$ time, one can either decide $\mathcal{I}$ or construct an instance $\mathcal{I}' = (\boldsymbol{G}' = (V', \boldsymbol{E}', \tau), s, z, k')$ of* Strict Temporal (s, z)-Separation *such that $\mathcal{I}'$ is equivalent to $\mathcal{I}$, $\boldsymbol{G}'$ is reduced, $|V'| \le |V|$, $|\boldsymbol{E}'| \le |\boldsymbol{E}|$, and $k' \le k$.*

Lovász et al. [28] showed that the minimum size of an (s, z)-separator for paths of length at most four in a graph is equal to the number of vertex-disjoint (s, z)-paths of length at most four in a graph. We adjust their idea to strict temporal paths on temporal graphs. The proof of Lovász et al. [28] implicitly relies on the transitivity of connectivity in static graphs. This does not hold for temporal graphs; hence, we have to extend their result to the temporal case. To this end, we define a directed auxiliary graph.

▶ **Definition 3.5** (Directed Path Cover Graph)**.** Let $\boldsymbol{G} = (V, \boldsymbol{E}, \tau = 4)$ be a reduced temporal graph with $s, z \in V$. The *directed path cover graph* from s to z of $\boldsymbol{G}$ is a directed graph $D = (V, \vec{E})$ such that $(v, w) \in \vec{E}$ if and only if

(i) $v, w \in V$,
(ii) $(\{v, w\}, t) \in \boldsymbol{E}$ for some $t \in [\tau]$, and
(iii) $v \in V_{(i,j)}$ and $w \in V_{(i',j')}$ such that $i < i'$, $v \in V_{(2,2)}$ and $w \in V_{(2,1)}$, $v = s$ and $w \in V_{(1,j)}$, or $w = z$ and $v \in V_{(i,1)}$ for some $i, j \in \{2, 3\}$.

Herein, a vertex $x \in V$ is in the set $V_{(i,j)}$ if the shortest strict temporal (s, x)-path is of length i and the shortest strict temporal (x, z)-path is of length j.

Figure 3 depicts a generic directed path cover graph of a reduced temporal graph with $\tau = 4$. Note that due to the definition of reduced temporal graphs, one can prove that the set $V_{(1,1)}$ is always empty, and hence not depicted in Figure 3. This is a crucial property that allows us to prove the following.

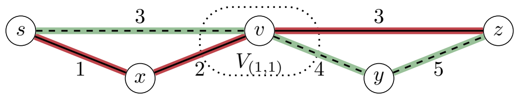


Figure 4 A reduced temporal graph with maximum label $\tau = 5$ where the vertex set $V_{(1,1)}$ of the directed path cover graph is not empty. The solid (red) and dashed (green) edges are strict temporal paths and show that edges $(\{s, v\}, 3)$ and $(\{v, z\}, 3)$ are not removed when the graph is reduced. Furthermore, v is not removed since $((\{s, v\}, 3), (\{v, z\}, 3))$ is not a strict temporal path.

▶ **Lemma 3.6** ($\star$). *Let $\boldsymbol{G} = (V, \boldsymbol{E}, \tau = 4)$ be a reduced temporal graph with $s, z \in V$. Then the directed path cover graph D from s to z of $\boldsymbol{G}$ can be computed in $\mathcal{O}(|\boldsymbol{E}|)$ time and $S \subseteq V \setminus \{s, z\}$ is a strict temporal (s, z)-separator in $\boldsymbol{G}$ if and only if S is an (s, z)-separator in D.*

Figure 4 shows that if $\tau = 5$, then we can construct a reduced temporal graph where the set $V_{(1,1)}$ is not empty. This indicates why our algorithm fails for $\tau = 5$.

Finally, with Lemmata 3.4 and 3.6 we can prove Theorem 3.2.

Proof of Theorem 3.2. Let $\mathcal{I} := (\boldsymbol{G} = (V, \boldsymbol{E}, \tau = 4), s, z, k)$ be an instance of Strict Temporal (s, z)-Separation. First, apply Lemma 3.4 in $\mathcal{O}(k \cdot |\boldsymbol{E}|)$ time to either decide $\mathcal{I}$ or to obtain an instance $\mathcal{I}' = (\boldsymbol{G}' = (V', \boldsymbol{E}', \tau), s, z, k')$ of Strict Temporal (s, z)-Separation. In the second case, compute the directed path cover graph D of $\boldsymbol{G}'$ from s to z in $\mathcal{O}(|\boldsymbol{E}'|)$ time (by Lemma 3.6). Next, check whether D has an (s, z)-separator of size at most k' in $\mathcal{O}(k' \cdot |\boldsymbol{E}'|)$ time by a folklore result [19]. By Lemma 3.6, D has an (s, z)-separator of size k' if and only if $\boldsymbol{G}'$ has a strict temporal (s, z)-separator of size k'. Since by Lemma 3.4 we have that $\boldsymbol{G}'$ is reduced, $|V'| \leq |V|$, $|\boldsymbol{E}'| \leq |\boldsymbol{E}|$, and $k' \leq k$, the overall running time is $\mathcal{O}(k \cdot |\boldsymbol{E}|)$. ◀

4 On Temporal Graphs with Planar Underlying Graph

In this section, we study our problems on *planar temporal graphs*, that is, temporal graphs that have a planar underlying graph. We show that both Temporal (s, z)-Separation and Strict Temporal (s, z)-Separation remain NP-complete on planar temporal graphs. On the positive side, we show that on planar temporal graphs with a constant number of layers, Strict Temporal (s, z)-Separation can be solved in $\mathcal{O}(|\boldsymbol{E}| \cdot \log |\boldsymbol{E}|)$ time.

In order to prove our hardness results, we first prove NP-hardness for Length-Bounded (s, z)-Separation on planar graphs – a result which we consider to be of independent interest; note that NP-completeness on planar graphs was only known for the edge-deletion variant of Length-Bounded (s, z)-Separation on undirected graphs [17] and weighted directed graphs [33].

▶ **Theorem 4.1.** Length-Bounded (s, z)-Separation *on planar graphs is* NP*-hard.*

Proof. We give a many-one reduction from the NP-complete [17] edge-weighted variant of Length-Bounded (s, z)-Cut, referred to as Planar Length-Bounded (s, z)-Cut, where the input graph $G = (V, E)$ is planar, has edge costs $c : E \to \{1, k + 1\}$, has maximum degree $\Delta = 6$, the degree of s and z is three, and s and z are incident to the outer face. Since the maximum degree is constant, one can replace a vertex with a planar grid-like gadget.

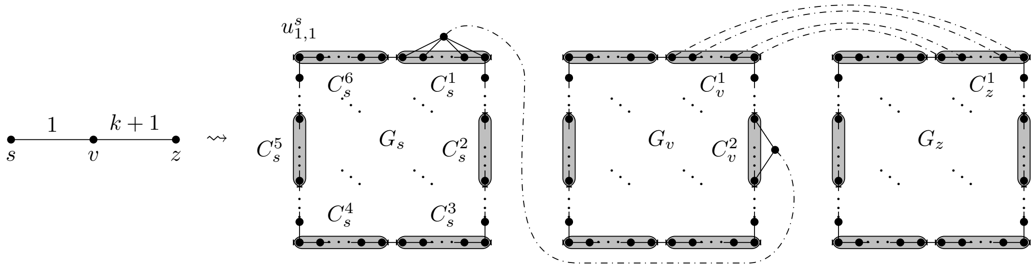


Figure 5 A simple planar graph G (left) with edge costs (above edges) and the obtained graph G' in the reduction from Theorem 4.1. The connector sets are highlighted in gray. The edge-gadgets are indicated by dash-dotted lines.

Let $\mathcal{I} := (G = (V, E, c), s, z, \ell, k)$ be an instance of Planar Length-Bounded (s, z)-Cut, and we assume k to be even[7]. We construct an instance $\mathcal{I}' := (G', s', z', \ell', k)$ of Length-Bounded (s, z)-Separation as follows (refer to Figure 5 for an illustration).

Construction. For each vertex $v \in V$, we introduce a *vertex-gadget* G_v which is a grid of size $(2k + 2) \times (2k + 2)$, that is, a graph with vertex set $\{u^v_{i,j} \mid i, j \in [2k + 2]\}$ and edge set $\{\{u^v_{i,j}, u^v_{i',j'}\} \mid |i - i'| + |j - j'| = 1\}$. There are six pairwise disjoint subsets $C^1_v, \ldots, C^6_v \subseteq V(G_v)$ of size $k + 1$ that we refer to as *connector sets*. As we fix an orientation of G_v such that $u^v_{1,1}$ is in the top-left, there are two connector sets on the top of G_v, two on the bottom of G_v, one on the left of G_v, and one on the right of G_v. Formally, $C^1_v = \{u^v_{1,k+2}, \ldots, u^v_{1,2k+2}\}$, $C^2_v = \{u^v_{k/2,2k+2}, \ldots, u^v_{3k/2,2k+2}\}$, $C^3_v = \{u^v_{2k+2,k+2}, \ldots, u^v_{2k+2,2k+2}\}$, $C^4_v = \{u^v_{2k+2,1}, \ldots, u^v_{2k+2,k+1}\}$, $C^5_v = \{u^v_{k/2,1}, \ldots, u^v_{3k/2,1}\}$, and $C^6_v = \{u^v_{1,1}, \ldots, u^v_{1,k+1}\}$.

Note that all (x, y)-paths are of length at most $k' := (2k + 2)^2 - 1$, for all $x, y \in V(G_v)$, because there are only $(2k + 2)^2$ vertices in $V(G_v)$.

Let $\phi(G)$ be a plane embedding of G. We say that an edge e incident with vertex $v \in V$ is *at position i on v* if e is the ith edge incident with v when counted clockwise with respect to $\phi(G)$.

For each edge $e = \{v, w\}$, we introduce an *edge-gadget* G_e that differs on the weight of e, as follows. Let e be at position $i \in \{1, \ldots, 6\}$ on v and at position $j \in \{1, \ldots, 6\}$ on w.

If $c(e) = 1$, then G_e is constructed as follows. Add a path consisting of $(\ell + 1) \cdot k' - 1$ vertices and connect one endpoint with each vertex in C^i_v by an edge and connect the other endpoint with each vertex in C^j_w by an edge.

If $c(e) = k + 1$, then G_e is constructed as follows. We introduce a planar matching between the vertices in C^i_v and C^j_w. That is, for instance, we connect vertex $u^v_{1,k+2+p}$ with vertex $u^w_{1,2k+2-p}$ for each $p \in \{0, \ldots, k\}$, if $i = j = 1$, or we connect vertex $u^v_{1,1+p}$ with vertex $u^w_{2k+2,3k/2-p}$ for each $p \in \{0, \ldots, k\}$, if $i = 6$ and $j = 2$ (we omit the remaining cases). Then, replace each edge by a path of length at least $(\ell + 1) \cdot k' + 1$ where its endpoints are identified with the endpoints of the replaced edge. Hence, a path between two vertex-gadgets has length at least $(\ell + 1) \cdot k' + 1$.

Next, we choose connector sets $C^{i'}_s$ and $C^{j'}_z$ such that no vertex $v \in C^{i'}_s \cup C^{j'}_z$ is adjacent to a vertex from an edge-gadget. Such i' and j' always exist because the degrees of s and z are both three. Now, we add two special vertices s' and z' and edges between s' and each vertex in $C^{i'}_s$, as well as between z' and each vertex in $C^{j'}_z$.

[7] If k is odd, since s and z are incident to the outer face, then we can add a path of length $\ell - 1$ with endpoints s and z and set the budget for edge deletions to $k + 1$.

Finally, we set $\ell' := 2 + (\ell + 1) \cdot k' + \ell\,((\ell + 1) \cdot k' + 1)$. Note that G' can be computed in polynomial time. Moreover, one can observe that G' is planar by obtaining an embedding from ϕ. This concludes the description of the construction.

Correctness. We claim that $\mathcal{I}$ is a yes-instance if and only if $\mathcal{I}'$ is a yes-instance.

$\Rightarrow$: Let $\mathcal{I}$ be a yes-instance. Thus, there is a solution $C \subset E$ with $c(C) \leq k$ such that there is no (s, z)-path of length at most ℓ in $G - C$. We construct a set $S \subset V(G')$ of size at most k by taking for each $\{v, w\} \in C$ one arbitrary vertex from the edge-gadget $G_{\{v,w\}}$ into S. Note that since $c(C) \leq k$, each edge in C is of cost one.

Assume towards a contradiction that there is a shortest (s', z')-path P' of length at most ℓ' in $G' - S$. Since a path between two vertex-gadgets has length at least $(\ell+1) \cdot k' + 1$, we know that P' goes through at most ℓ edge-gadgets. Otherwise P' would be of length at least $2 + (\ell + 1) \cdot [(\ell + 1) \cdot k' + 1] = 2 + (\ell + 1) \cdot k' + \ell \cdot [(\ell + 1) \cdot k' + 1] + 1 = \ell' + 1$. Now, we reconstruct an (s, z)-path P in G corresponding to P' by taking an edge $e \in E$ into P if P' goes through the edge-gadget G_e. Hence, the length of P is at most ℓ. This contradicts that there is no (s, z)-path of length at most ℓ in $G - C$. Consequently, there is no (s', z')-path of length at most ℓ' in $G' - S$ and $\mathcal{I}'$ is a yes-instance.

$\Leftarrow$: Let $\mathcal{I}'$ be a yes-instance. Thus, there is a solution $S \subseteq V(G')$ of minimum size (at most k) such that there is no (s', z')-path of length at most ℓ' in $G' - S$. Since S is of minimum size, it follows from the following claim that $V(G_v) \cap S = \emptyset$ for all $v \in V$.

▶ **Claim 4.2.** *Let G_v be a vertex-gadget and $i, j \in \{1, \ldots, 6\}$ with $i \neq j$. Then, for each vertex set $S \subseteq V(G_v)$ of size at most k it holds that there are $v_1 \in C_v^i \setminus S$ and $v_2 \in C_v^j \setminus S$ such that there is a (v_1, v_2)-path of length at most k' in $G_v - S$.*

Proof of Claim 4.2. Let C_v^i, C_v^j two connector sets of a vertex-gadget G_v, where $i, j \in \{1, \ldots, 6\}$ and $i \neq j$. We add vertices a and b and edges $\{a, a'\}$ and $\{b, b'\}$ to G_v, where $a' \in C_v^i$ and $b' \in C_v^j$. There are $\binom{6}{2}$ different cases in which $i \neq j$. It is not difficult to see that in each case there are $k + 1$ vertex-disjoint (a, b)-paths. The claim then follows by Menger's Theorem [29]. ◀

Note that by minimality of S, it holds that $V(G_e) \cap S = \emptyset$ for all $e \in E$ with $c(e) = k + 1$. We construct an edge set $C \subseteq E$ of cost at most k by taking $\{v, w\} \in E$ into C if there is a $y \in V(G_{\{v,w\}}) \cap S$.

Assume towards a contradiction that there is a shortest (s, z)-path P of length at most ℓ in $G - C$. We reconstruct an (s', z')-path P' in G' which corresponds to P as follows. First, we take an edge $\{s', v\} \in E(G')$ such that $v \in C_s^{i'} \setminus S$. Such a v always exists, because $|C_s^{i'}| = k + 1$ and $|S| \leq k$. Let $\{s, w\} \in E$ be the first edge of P and at position i on w. Then we add a (v, v')-path P_s in $G_s - S$, such that $v' \in C_s^i \setminus S$. Due to Claim 4.2, such a (v, v')-path P_s always exists in $G_s - S$ and is of length at most k'.

We take an edge-gadget G_e into P' if e is in P. Recall, that an edge-gadget is a path of length $(\ell+1) \cdot k' + 1$. Due to Claim 4.2, we can connect the edge-gadgets $G_{\{v_1,v_2\}}, G_{\{v_2,v_3\}}$ of two consecutive edges $\{v_1, v_2\}, \{v_2, v_3\}$ in P by a path of length at most k' in G_{v_2}. Let $\{v_p, z\}$ be the last edge in P, be at position j on z, $v \in C_z^j$, and $v' \in C_z^{j'}$. We add a (v, v')-path of length k' in $G_z - S$ (Claim 4.2). Note that P' visits at most $\ell + 1$ vertex-gadgets and ℓ edge-gadgets. The length of P' is at most $2 + (\ell + 1) \cdot k' + \ell\,[(\ell + 1) \cdot k' + 1] = \ell$. This contradicts that S forms a solution for $\mathcal{I}'$.

It follows that there is no (s, z)-path of length at most ℓ in $G - C$ and $\mathcal{I}$ is a yes-instance. ◀

From the proofs of Theorem 3.1 and Lemma 2.2 (planarity-preserving reductions for the underlying graph), together with Theorem 4.1 we get the following:

▶ **Corollary 4.3.** *Both* Temporal (s,z)-Separation *and* Strict Temporal (s,z)-Separation *on planar temporal graphs are* NP*-complete.*

In contrast to the case of general temporal graphs, Strict Temporal (s,z)-Separation on planar temporal graphs is efficiently solvable if the maximum label τ is any constant. To this end, we employ the optimization variant of Courcelle's Theorem [4, 11].

▶ **Proposition 4.4** ($\star$)**.** Strict Temporal (s,z)-Separation *on planar temporal graphs can be solved in $\mathcal{O}(|\boldsymbol{E}| \cdot \log |\boldsymbol{E}|)$ time, if the maximum label τ is constant.*

Due to space constrains, we only sketch how one can develop MSO formulas over temporal graphs and postpone the full proof to the long version.

Proof (Sketch). Let $\mathcal{I} = (\boldsymbol{G} = (V, \boldsymbol{E}, \tau), s, z, k)$ be an instance of Strict Temporal (s,z)-Separation, where the underlying graph $G_\downarrow$ of $\boldsymbol{G}$ is planar. We define the edge-labeled graph $L(\boldsymbol{G})$ to be $G_\downarrow$ with the edge-labeling $\omega : E(G_\downarrow) \to [2^\tau - 1]$ with $\omega(\{v,w\}) = \sum_{i=1}^{\tau} \mathbb{1}_{\{v,w\} \in E_i} \cdot 2^{i-1}$, where $\mathbb{1}_{\{v,w\} \in E_i} = 1$ if and only if $(\{v,w\}, i) \in \boldsymbol{E}$, and 0 otherwise. Observe that in binary representation, the i-th bit of $\omega(\{v,w\})$ is 1 if and only if $\{v,w\}$ exists at time point i.

We define the optimization variant of Strict Temporal (s,z)-Separation in MSO on $L(\boldsymbol{G})$. First, the MSO formula $\text{layer}(e,t) := \bigvee_{i=1}^{\tau} \bigvee_{j \in \sigma(i, 2^\tau - 1)} \big(t = i \ \wedge \omega(e) = j\big)$ checks whether an edge e is present in the layer t, where $\sigma(i,j) := \{x \in [j] \mid i\text{-th bit of } x \text{ is } 1\}$. Second, we can write an MSO formula $\text{tempadj}(v,w,t) := \exists_{e \in E} \big(\text{inc}(e,v) \wedge \text{inc}(e,w) \wedge \text{layer}(e,t)\big)$ to determine whether two vertices v and w are adjacent at time point t. Third, there is an MSO formula

$$\text{path}(S) := \exists_{x_1,\ldots,x_{\tau+1} \in V \setminus S} \Big(x_1 = s \wedge x_{\tau+1} = z \wedge \textstyle\bigwedge_{i=1}^{\tau} \big(x_i = x_{i+1} \vee \text{tempadj}(x_i, x_{i+1}, i)\big)\Big)$$

to check whether there is a strict temporal (s,z)-path which does not visit any vertex in S. Note that the length of $\text{layer}(e,t)$, and hence the length of $\text{path}(S)$, is upper-bounded by some function in $2^{\mathcal{O}(\tau)}$. The facts that the length of a strict temporal (s,z)-path is at most τ and the treewidth of a planar graph can be bounded in its diameter (see Flum and Grohe [16]), together with an application of Courcelle's Theorem (optimization variant, see long version) on the MSO formula $\phi(S) := S \subseteq (V \setminus \{s,z\}) \wedge \neg\text{path}(S)$ complete the proof. ◀

5 On Temporal Graphs with Small Temporal Cores

In this section, we investigate the complexity of deciding (Strict) Temporal (s,z)-Separation on temporal graphs where the number of vertices whose incident edges change over time is small. We call the set of such vertices the *temporal core* of the temporal graph.

▶ **Definition 5.1** (Temporal core)**.** For a temporal graph $\boldsymbol{G} = (V, \boldsymbol{E}, \tau)$, the vertex set $W = \{v \in V \mid \exists \{v,w\} \in (\bigcup_{i=1}^{\tau} E_i) \setminus (\bigcap_{i=1}^{\tau} E_i)\} \subseteq V$ is called the *temporal core.*

A temporal graph is often composed of a public transport system and an ordinary street network. Here, the temporal core consists of vertices involved in the public transport system.

For Strict Temporal (s,z)-Separation, we can observe that the hardness reduction described in the proof of Theorem 3.1 produces an instance with an empty temporal core. In stark contrast, we show that Temporal (s,z)-Separation is fixed-parameter tractable when parameterized by the size of the temporal core[8]. We reduce an instance to Node

[8] Note that we can compute the temporal core in $\mathcal{O}(|\boldsymbol{E}| \log |\boldsymbol{E}|)$ time.

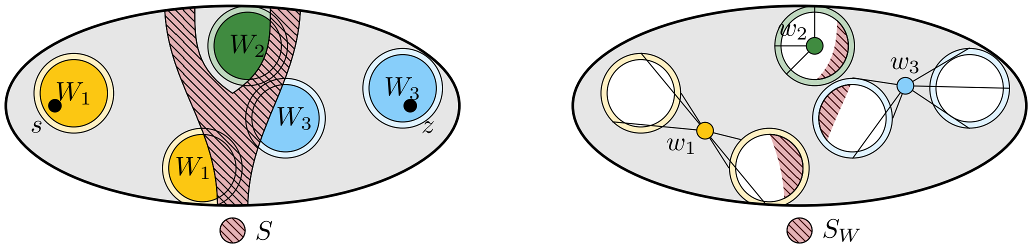


Figure 6 Illustration of the idea behind the proof of Theorem 5.2. *Left-hand side*: Sketch of a temporal graph $\boldsymbol{G}$ (enclosed by the ellipse) with temporal (s, z) separator S (red hatched) and induced partition $\{S_W, W_1, W_2, W_3\}$ of the temporal core W, where $S_W = W \cap S$. The outer rings of W_1, W_2, W_3 contain the open neighborhood of the sets. *Right-hand side*: Sketch of the constructed graph G' (enclosed by the ellipse). The partition $\{S_W, W_1, W_2, W_3\}$ is guessed in steps (1) and (2). The vertices w_1, w_2, w_3 with edges to the neighborhood of W_1, W_2, W_3, respectively, are created in step (3).

Multiway Cut (NWC) in such a way that we can use an above lower bound FPT-algorithm due to Cygan et al. [12] for NWC as a subprocedure in our algorithm for Temporal (s, z)-Separation. Note that the above lower bound parameterization is crucial to obtain the desired FPT-running time bound. Recall the definition of NWC:

Node Multiway Cut (NWC)
Input: An undirected graph $G = (V, E)$, a set of terminals $T \subseteq V$, and an integer k.
Question: Is there a set $S \subseteq (V \setminus T)$ of size at most k such there is no (t_1, t_2)-path for every distinct $t_1, t_2 \in T$?

We remark that Cygan et al.'s algorithm can be modified to obtain a solution S. Formally, we show the following.

▶ **Theorem 5.2.** Temporal (s, z)-Separation *can be solved in* $2^{|W|\cdot(\log|W|+2)} \cdot |V|^{\mathcal{O}(1)} + \mathcal{O}(|\boldsymbol{E}|\log|\boldsymbol{E}|)$ *time, where* W *denotes the temporal core of the input graph.*

Proof. Let instance $\mathcal{I} = (\boldsymbol{G} = (V, \boldsymbol{E}, \tau), s, z, k)$ of Temporal (s, z)-Separation with temporal core $W \subseteq V$ be given. Without loss of generality, we can assume that $s, z \in W$, as otherwise we add two vertices one being incident only with s and the other being incident only with z, both only in layer one. Furthermore, we need the notion of a *maximal static subgraph* $\widehat{G}$ of a temporal graph $\boldsymbol{G} = (V, \boldsymbol{E})$: It contains all edges that appear in every layer, more specifically $\widehat{G} = (V, \widehat{E})$ with $\widehat{E} = \bigcap_{i\in[\tau]} E_i$. Our algorithm works as follows.

(1) Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
(2) Guess a number r and a partition $\{W_1, \ldots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same W_i, for some $i \in [r]$.
(3) Construct the graph G' by copying $\widehat{G} - W$ and adding a vertex w_i for each part W_i. Add edge sets $\{\{v, w_i\} \mid v \in N_{\widehat{G}}(W_i) \setminus W\}$ for all $i \in [r]$ and for all $i, j \in [r]$ add an edge $\{w_i, w_j\}$ if $N_{\widehat{G}}(W_i) \cap W_j \neq \emptyset$.
(4) Solve the NWC instance $\mathcal{I}' = (G', \{w_1, \ldots, w_r\}, k - |S_W|)$.
(5) If a solution S' is found for $\mathcal{I}'$ and $S' \cup S_W$ is a solution for $\mathcal{I}$, then output yes.
(6) If all possible guesses in (1) and (2) are considered without finding a solution for $\mathcal{I}$, then output no.

See Figure 6 for a visualization of the constructed graph G'. Since we do a sanity check in step (5) it suffices to show that if $\boldsymbol{G}$ has a temporal (s, z)-separator of size at most k, then there is a partition $\{S_W, W_1, \ldots, W_r\}$ of W where s and z are in different parts such that

(i) the NWC instance $\mathcal{I}'$ has a solution of size at most $k - |S_W|$, and
(ii) if S' is a solution to $\mathcal{I}'$, then $S_W \cup S'$ is a temporal (s,z)-separator in $\boldsymbol{G}$.

Let S be a temporal (s,z)-separator of size at most k in $\boldsymbol{G}$. First, we set $S_W = S \cap W$. Let $C_1, \ldots, C_r$ be the connected components of $\widehat{G} - S$ with $C_i \cap W \neq \emptyset$ for all $i \in [r]$. Now we construct a partition $\{S_W, W_1, \ldots, W_r\}$ of W such that $W_i = W \cap C_i$ for all $i \in [r]$. It is easy to see that s and z are in different parts of this partition. Observe that for $i, j \in [r]$ with $i \neq j$ the vertices $v \in W_i$ and $u \in W_j$ are in different connected components of $\widehat{G} - S$. Hence, $w_1, \ldots, w_r$ are in different connected components of $G' - (S \setminus S_W)$. Thus $S \setminus S_W$ is a solution of size at most $k - |S_W|$ of the NWC instance $\mathcal{I}' = (G', \{w_1, \ldots, w_r\}, k - |S_W|)$, proving (i).

For the correctness, it remains to prove (ii). Let S' be a solution of size at most $k-|S_W|$ of the NWC instance $\mathcal{I}'$. We need to prove that $S' \cup S_W$ forms a temporal (s,z)-separator in $\boldsymbol{G}$. Clearly, if $S' = S \setminus S_W$, we are done by the arguments before. Thus, assume $S' \neq S \setminus S_W$. Since S' is a solution to $\mathcal{I}'$, we know that $w_1, \ldots, w_r$ are in different connected components of $G' - S'$. Hence, for $i, j \in [r]$ with $i \neq j$ the vertices $v \in W_i$, $u \in W_j$ are in different connected components of $\widehat{G} - (S' \cup S_W)$.

Now assume towards a contradiction that there is a temporal (s,z)-path P in $\boldsymbol{G} - (S' \cup S_W)$. Observe that $\{s, z\} \subseteq V(P) \cap W$. Hence, we have two different vertices $u_1, u_2 \in V(P) \cap W$ such that there is no temporal (u_1, u_2)-path in $\boldsymbol{G} - S$ and all vertices that are visited by P between u_1 and u_2 are contained in $V \setminus W$: Take the furthest vertex in P that is also contained in W and is reachable by a temporal path from s in $\boldsymbol{G} - S$ as u_1, and take the next vertex (after u_1) in P that is also contained in W as u_2. Note that u_1 and u_2 are disconnected in $\widehat{G} - S$, and hence there are $i, j \in [r]$ with $i \neq j$ such that $u_1 \in W_i$ and $u_2 \in W_j$. Since P does not visit any vertices in $(S' \cup S_W)$ we can conclude that u_1 and u_2 are connected in $\widehat{G} - (S' \cup S_W)$, and hence w_i and w_j are connected in $G' - S'$. This contradicts the fact that S' is a solution for $\mathcal{I}'$.

Running time: It remains to show that the our algorithm runs in the proposed time. For the guess in step (1) there are at most $2^{|W|}$ many possibilities. For the guess in step (2) there are at most $B_{|W|} \leq 2^{|W| \cdot \log(|W|)}$ many possibilities, where B_n is the n-th Bell number. Step (3) and the sanity check in step (5) can clearly be done in polynomial time.

Let L be a minimum (s,z)-separator in $\widehat{G} - (W \setminus \{s,z\})$. If $k \geq |W \setminus \{s,z\}| + |L|$, then $(W \setminus \{s,z\}) \cup L$ is a temporal (s,z)-separator of size at most k for $\boldsymbol{G}$. Otherwise, we have that $k - |L| < |W|$. Cygan et al. [12] showed that NWC can be solved in $2^{k-b} \cdot |V|^{\mathcal{O}(1)}$ time , where $b := \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$. Since s and z are not in the same W_i for any $i \in [r]$, we know that $|L| \leq b$. Hence, $k - b \leq k - |L| < |W|$ and step (4) can be done in $2^{|W|} \cdot |V|^{\mathcal{O}(1)}$ time. Thus we have an overall running time of $2^{|W| \cdot (\log |W| + 2)} \cdot |V|^{\mathcal{O}(1)} + \mathcal{O}(|\boldsymbol{E}| \log |\boldsymbol{E}|)$. ◀

We conclude that the strict and the non-strict variant of TEMPORAL (s,z)-SEPARATION behave very differently on temporal graphs with a constant-size temporal core. While the strict version stays NP-complete, the non-strict version becomes polynomial-time solvable.

6 Conclusion

The temporal path model strongly matters when assessing the computational complexity of finding small separators in temporal graphs. This phenomenon has so far been neglected in the literature. We settled the complexity dichotomy of TEMPORAL (s,z)-SEPARATION and STRICT TEMPORAL (s,z)-SEPARATION by proving NP-hardness on temporal graphs with $\tau \geq 2$ and $\tau \geq 5$, respectively, and polynomial-time solvability if the number of layers is below the respective constant. The mentioned hardness results also imply that both problem variants are W[1]-hard when parameterized by the solution size k. When considering the parameter combination $k + \tau$, it is easy to see that Strict Temporal (s, z)-Separation is fixed-parameter tractable [38]: There is a straightforward search-tree algorithm that branches on all vertices of a strict temporal (s, z)-path which has length at most τ. Whether the non-strict variant is fixed-parameter tractable regarding the same parameter combination remains open.

We showed that (Strict) Temporal (s, z)-Separation on temporal graphs with planar underlying graphs remains NP-complete. However, for the planar case we proved that if additionally the number τ of layers is a constant, then Strict Temporal (s, z)-Separation is solvable in $\mathcal{O}(|\boldsymbol{E}| \cdot \log |\boldsymbol{E}|)$ time. We leave open whether Temporal (s, z)-Separation admits a similar result. Finally, we introduced the notion of a temporal core as a temporal graph parameter. We proved that on temporal graphs with constant-size temporal core, while Strict Temporal (s, z)-Separation remains NP-hard, Temporal (s, z)-Separation is solvable in polynomial time.

References

1 Ravindra K. Ahuja, Thomas L. Magnanti, and James B. Orlin. *Network Flows: Theory, Algorithms and Applications*. Prentice Hall, 1993.

2 Eleni C Akrida, Jurek Czyzowicz, Leszek Gąsieniec, Łukasz Kuszner, and Paul G Spirakis. Temporal flows in temporal networks. In *Proceedings of the 10th International Conference on Algorithms and Complexity (CIAC '17)*, pages 43–54. Springer, 2017.

3 Eleni C Akrida, Leszek Gąsieniec, George B Mertzios, and Paul G Spirakis. On temporally connected graphs of small cost. In *Proceedings of the 13th International Workshop on Approximation and Online Algorithms (WAOA '15)*, pages 84–96. Springer, 2015.

4 Stefan Arnborg, Jens Lagergren, and Detlef Seese. Easy problems for tree-decomposable graphs. *Journal of Algorithms*, 12(2):308–340, 1991.

5 Kyriakos Axiotis and Dimitris Fotakis. On the size and the approximability of minimum temporally connected subgraphs. In *Proceedings of the 43rd International Colloquium on Automata, Languages, and Programming (ICALP '16)*, pages 149:1–149:14. Schloss Dagstuhl - Leibniz-Zentrum fuer Informatik, 2016.

6 Georg Baier, Thomas Erlebach, Alexander Hall, Ekkehard Köhler, Petr Kolman, Ondřej Pangrác, Heiko Schilling, and Martin Skutella. Length-bounded cuts and flows. *ACM Transactions on Algorithms*, 7(1):4:1–4:27, 2010.

7 Kenneth A Berman. Vulnerability of scheduled networks and a generalization of Menger's Theorem. *Networks*, 28(3):125–134, 1996.

8 Stefano Boccaletti, Ginestra Bianconi, Regino Criado, Charo I Del Genio, Jesús Gómez-Gardenes, Miguel Romance, Irene Sendina-Nadal, Zhen Wang, and Massimiliano Zanin. The structure and dynamics of multilayer networks. *Physics Reports*, 544(1):1–122, 2014.

9 Arnaud Casteigts, Paola Flocchini, Walter Quattrociocchi, and Nicola Santoro. Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408, 2012.

10 H.W Corley and David Y Sha. Most vital links and nodes in weighted networks. *Operations Research Letters*, 1(4):157–160, 1982.

11 Bruno Courcelle and Joost Engelfriet. *Graph Structure and Monadic Second-order Logic: a Language-theoretic Approach*. Cambridge University Press, 2012.

12 Marek Cygan, Marcin Pilipczuk, Michał Pilipczuk, and Jakub Onufry Wojtaszczyk. On multiway cut parameterized above lower bounds. *ACM Transactions on Computation Theory*, 5(1):3:1–3:11, 2013.

13 Reinhard Diestel. *Graph Theory, 5th Edition*, volume 173 of *Graduate Texts in Mathematics*. Springer, 2016.

14 Thomas Erlebach, Michael Hoffmann, and Frank Kammer. On temporal graph exploration. In *Proceedings of the 42nd International Colloquium on Automata, Languages, and Programming (ICALP '15)*, pages 444–455. Springer, 2015.

15 Afonso Ferreira. Building a reference combinatorial model for MANETs. *IEEE Network*, 18(5):24–29, 2004.

16 Jörg Flum and Martin Grohe. *Parameterized Complexity Theory*. Springer-Verlag, Berlin, 2006.

17 Till Fluschnik, Danny Hermelin, André Nichterlein, and Rolf Niedermeier. Fractals for kernelization lower bounds. *SIAM Journal on Discrete Mathematics*, 32(1):656–681, 2018.

18 Till Fluschnik, Hendrik Molter, Rolf Niedermeier, and Philipp Zschoche. Temporal graph classes: A view through temporal separators. *arXiv preprint arXiv:1803.00882*, 2018. To appear in *Proceedings 44th International Workshop on Graph-Theoretic Concepts in Computer Science (WG '18)*.

19 Lester R Ford and Delbert R Fulkerson. Maximal flow through a network. *Canadian Journal of Mathematics*, 8(3):399–404, 1956.

20 Petr A. Golovach and Dimitrios M. Thilikos. Paths of bounded length and their cuts: Parameterized complexity and algorithms. *Discrete Optimization*, 8(1):72–86, 2011.

21 Anne-Sophie Himmel. Algorithmic investigations into temporal paths. Masterthesis, TU Berlin, April 2018. URL: `http://fpt.akt.tu-berlin.de/publications/theses/MA-anne-sophie-himmel.pdf`.

22 Anne-Sophie Himmel, Hendrik Molter, Rolf Niedermeier, and Manuel Sorge. Adapting the Bron-Kerbosch algorithm for enumerating maximal cliques in temporal graphs. *Social Network Analysis and Mining*, 7(1):35:1–35:16, 2017.

23 Petter Holme. Modern temporal network theory: a colloquium. *European Physical Journal B*, 88(9):234, 2015.

24 Petter Holme and Jari Saramäki. Temporal networks. *Physics Reports*, 519(3):97–125, 2012.

25 David Kempe, Jon Kleinberg, and Amit Kumar. Connectivity and inference problems for temporal networks. *Journal of Computer and System Sciences*, 64(4):820–842, 2002.

26 Matthieu Latapy, Tiphaine Viard, and Clémence Magnien. Stream graphs and link streams for the modeling of interactions over time. *arXiv preprint arXiv:1710.04073*, 2017.

27 Qingkai Liang and Eytan Modiano. Survivability in time-varying networks. *IEEE Transactions on Mobile Computing*, 16(9):2668–2681, 2017.

28 László Lovász, Víctor Neumann-Lara, and Michael Plummer. Mengerian theorems for paths of bounded length. *Periodica Mathematica Hungarica*, 9(4):269–276, 1978.

29 Karl Menger. Zur allgemeinen Kurventheorie. *Fundamenta Mathematicae*, 10(1):96–115, 1927.

30 George B Mertzios, Othon Michail, Ioannis Chatzigiannakis, and Paul G Spirakis. Temporal network optimization subject to connectivity constraints. In *Proceedings of the 40th International Colloquium on Automata, Languages, and Programming (ICALP '13)*, pages 657–668. Springer, 2013.

31 Othon Michail. An introduction to temporal graphs: An algorithmic perspective. *Internet Mathematics*, 12(4):239–280, 2016.

32 Othon Michail and Paul G. Spirakis. Traveling salesman problems in temporal graphs. *Theoretical Computer Science*, 634:1–23, 2016.

33 Feng Pan and Aaron Schild. Interdiction problems on planar graphs. *Discrete Applied Mathematics*, 198:215–231, 2016.

34 Salvatore Scellato, Ilias Leontiadis, Cecilia Mascolo, Prithwish Basu, and Murtaza Zafer. Evaluating temporal robustness of mobile networks. *IEEE Transactions on Mobile Computing*, 12(1):105–117, 2013.

35 Baruch Schieber, Amotz Bar-Noy, and Samir Khuller. The complexity of finding most vital arcs and nodes. Technical report, University of Maryland at College Park, College Park, MD, USA, 1995.

36 Tiphaine Viard, Matthieu Latapy, and Clémence Magnien. Computing maximal cliques in link streams. *Theoretical Computer Science*, 609:245–252, 2016.

37 Huanhuan Wu, James Cheng, Yiping Ke, Silu Huang, Yuzhen Huang, and Hejun Wu. Efficient algorithms for temporal path computation. *IEEE Transactions on Knowledge and Data Engineering*, 28(11):2927–2942, 2016.

38 Philipp Zschoche. On finding separators in temporal graphs. Masterthesis, TU Berlin, August 2017. URL: `http://fpt.akt.tu-berlin.de/publications/theses/MA-philipp-zschoche.pdf`.