Journal of Computer and System Sciences 154 (2025) 103679

Contents lists available at ScienceDirect

Journal of Computer and System Sciences

journal homepage: www.elsevier.com/locate/jcss

An FPT algorithm for timeline cover [☆]

Riccardo Dondi [a,*], Manuel Lafond [b,1,2]

[a] Dipartimento di Lettere, Filosofia, Comunicazione, Università degli Studi di Bergamo, Bergamo, Italy
[b] Department of Computer Science, Université de Sherbrooke, Street, Sherbrooke, Canada

ARTICLE INFO

Article history:
Received 16 August 2024
Received in revised form 13 January 2025
Accepted 12 May 2025
Available online 22 May 2025

Keywords:
Temporal graphs
Temporal vertex cover
Graph algorithms
Parameterized complexity

ABSTRACT

One of the most studied problem in theoretical computer science, VERTEX COVER, has been recently considered in the temporal graph framework. Here we study a VERTEX COVER variant, called k-TIMELINECOVER. Given a temporal graph k-TIMELINECOVER asks to define an interval for each vertex so that for every temporal edge existing in a timestamp t, at least one of the endpoints has an interval that includes t. The goal is to decide whether it is possible to cover every temporal edge while using vertex intervals of total span at most k. k-TIMELINECOVER has been shown to be NP-hard, but its parameterized complexity has not been fully understood when parameterizing by the span of the solution. We settle this open problem by giving an FPT algorithm that combines two techniques, a modified form of iterative compression and a reduction to DIGRAPH PAIR CUT.

1. Introduction

Temporal graphs are emerging as one of the main models to describe the dynamics of complex networks. They describe how relations (edges) change in a discrete time domain [2,3], while the vertex set is not changing. The main focus of the algorithmic analysis of temporal graphs has been on finding paths or walks that respect some time constraints and on analyzing how graph connectivity changes when taking into account these temporal constraints [2,4–12]. However, many other classical problems in computer science have been recently extended to temporal graphs. A fundamental problem in graph theory and theoretical computer science, VERTEX COVER, has been considered on temporal graphs [13–15]. The first variant of VERTEX COVER in temporal graphs has been introduced in [13] and, given a temporal graph, asks for the minimum number of timestamps where vertices are defined to be active, such that each (non-temporal) edge $e = uv$ is temporally covered, that is, there exists a timestamp t where e is defined and one of (u, t) and (v, t) belongs to the cover. A second variant asks for each temporal edge to be temporally covered at least once for every interval of a given length. The two variants are both NP-hard, also in very restricted cases [13]. Results on the problem variants, including their approximability, have been given in [13,14].

Here we study a third variant of VERTEX COVER on temporal graph, introduced in [15] as NETWORK UNTANGLING. NETWORK UNTANGLING has applications in discovering event timelines and summarizing temporal networks. It considers a sequence of temporal interactions between entities (e.g. discussions between users in a social network) and aims to explain the observed

---

[☆] A preliminary version of this paper appeared in [1].
[*] Corresponding author.
E-mail addresses: riccardo.dondi@unibg.it (R. Dondi), manuel.lafond@usherbrooke.ca (M. Lafond).
[1] Contributing author.
[2] These authors contributed equally to this work.

https://doi.org/10.1016/j.jcss.2025.103679

interactions with few (and short) *activity intervals* of entities, such that each interaction is covered by at least one of the two entities involved (i.e. at least one of the two entities is active when an interaction between them is observed).

Network Untangling can be seen as a variant of Vertex Cover, where we search for a minimum cover of the interactions, called temporal edges. The size of this temporal vertex cover is based on the definition of the *span* of a vertex, that is the length of vertex activity. In particular, the span of a vertex is defined as the difference between the maximum and minimum timestamp where the vertex is active. Hence, if a vertex is active in exactly one timestamp, it has a span equal to 0. This models the idea that each vertex is present in the network because we know that they interacted at least once, but that sustained periods of interaction are relatively rare. This assumption is motivated in [15] by analyzing interactions in social media. A specific topic, like an event hashtag, may be very active around the time interval when the event occurs, but it may be discussed and referenced, less frequently, outside of this interval.

Four combinatorial formulations of Network Untangling have been defined in [15], varying the definition of vertex activity (a single interval or $h \geq 2$ intervals) and the objective function (minimization of the sum of vertex spans or minimization of the maximum vertex span). Here we consider the formulation, denoted by k-TimelineCover, where vertex activity is defined as a single interval and the objective function is the minimization of the sum of vertex spans. Hence, given a temporal graph, k-TimelineCover asks for a cover of the temporal edges that has minimum span and such that each vertex is active in one time interval.

We focus on this specific problem, since it is not known to be FPT or not, while the variant of the problem where vertex activity is defined as two intervals is known to be NP-hard when the span is equal to 0 [16]. Hence it is unlikely that this problem variant admits an FPT algorithm for parameter the span. The k-TimelineCover problem is known to be NP-hard [15], also in very restricted cases, when each timestamp contains at most one temporal edge [17], when each vertex has at most two incident temporal edges in each timestamp and the temporal graph is defined over three timestamps [17], and when the temporal graph is defined over two timestamps [16]. k-TimelineCover is also known to be approximable within factor $O(T \log n)$, where n is the number of vertices and T is the number of timestamps of the temporal graph [18]. Note that, since the span of a vertex activity in exactly one timestamp is equal to 0, k-TimelineCover is trivially in P when the temporal graph is defined on a single timestamp, since in this case any solution of the problem has span 0. Furthermore, deciding whether there exists a solution of k-TimelineCover that has span equal to 0 can be decided in polynomial time via a reduction to 2-SAT [15].

k-TimelineCover has been considered also in the parameterized complexity framework. The definition of span leads to a problem where the algorithmic approaches applied to Vertex Cover cannot be easily extended for the parameter span of the solution. Indeed, in Vertex Cover for each edge we are sure that at least one of the endpoints must be included in the solution, thus at least one of the vertices contributes to the cost of the solution. This leads to the textbook FPT algorithm of branching over the endpoints of any edge. For k-TimelineCover, a vertex with span 0 may cover a temporal edge, as the vertex can be active only in the timestamp where the temporal edge is defined. This makes it more challenging to design FPT algorithms when the parameter is the span of the solution. In this case, k-TimelineCover is known to admit a parameterized algorithm only when the input temporal graph is defined over two timestamps [16], with a parameterized reduction to the Almost 2-SAT problem. However, the parameterized complexity of k-TimelineCover for the span parameter on general instances has been left open [16,17]. The authors of [16] have also analyzed the parameterized complexity of the variants of Network Untangling proposed in [15], considering other parameters in addition to the span of the solution: the number of vertices of the temporal graph, the length of the time domain, and the number of intervals of vertex activity.

Our contributions. We solve the open question on the parameterized complexity of k-TimelineCover by showing that the problem is FPT in parameter k, the span of a solution, even if the number of timestamps is unbounded. Our algorithm takes time $O^*(2^{5k \log k})$, where the O^* notation hides polynomial factors. Our algorithm is divided into two phases, each using a different technique. First, given a temporal graph G, we use a variant of iterative compression, where we start from a solution $\mathcal{S}^*$ of span at most k on a subgraph of G induced by a subset of vertices (taken across all timestamps), and then try to maintain such a solution after adding a new vertex of G to the graph under consideration. This requires us to reorganize which vertices involved in $\mathcal{S}^*$ should be in the solution or not, and in which timestamps. One challenge is that since the number of such timestamps is unbounded, there are too many ways to choose how to include or not include the vertices that are involved in $\mathcal{S}^*$. We introduce the notion of a *feasible assignment*, which is a partial guess of a solution that reorganizes $\mathcal{S}^*$ (see Definition 4 for the formal definition). There are only $2^{O(k \log k)}$ ways of reorganizing the vertices in $\mathcal{S}^*$. We try each such feasible assignment $\mathcal{X}$, and we must then find a temporal cover of the whole graph G that "agrees" with $\mathcal{X}$.

This leads to the second phase of the algorithm, which decides if such an agreement cover exists through a reduction to a variant of a problem called Digraph Pair Cut. In this problem, we receive a directed graph and forbidden pairs of vertices, and we must delete at most k arcs so that a specified source vertex does not reach both vertices from a forbidden pair. It is known that the problem can be solved in time $O^*(2^k)$. In this work, we need a version where the input specifies a set of deletable and undeletable arcs, which we call Constrained Digraph Pair Cut. The Digraph Pair Cut problem and its variants have played an important role in devising randomized kernels using matroids [19] and, more recently, in establishing a dichotomy in the complexity landscape of constraint satisfaction problems [20,21]. The vertex-deletion variant also admits a randomized polynomial kernel, and other FPT results are known for weighted arc-deletion variants [22]. Here, the problem is useful since it can model the implications of including a vertex in the solution or not and, in a more challenging way, allows implementing the notion of cost using our definition of span. We hope that the techniques developed for this reduction can be useful for other variants of temporal graph cover.

2. Preliminaries

We introduce some notation for graphs and temporal graphs. For an integer n, we denote $[n] = \{1, \dots, n\}$ and for two integers i, j, we denote $[i, j] = \{i, i+1, \dots, j-1, j\}$ (which is empty if $i > j$). Temporal graphs are defined over a discrete time domain, which is a sequence $1, 2 \dots, T$ of timestamps. A temporal graph is also defined over a set of *vertices*, that do not change in the time domain and are defined in all timestamps, and are associated with *temporal vertices*, which are vertices defined in specific timestamps. For a vertex v and $t \in [T]$, we use (v, t) to denote the temporal vertex associated with v at timestamp t. A *temporal edge* $(u, t)(v, t)$ connects two temporal vertices (u, t) and (v, t), that belong to the same timestamp t and that are associated with distinct vertices u, v, respectively (edges are not directed).

Definition 1. A temporal graph $G = (V, E, T)$ consists of:

1. A time domain $\{1, 2 \dots, T\}$;
2. A set $V = V(G)$ of *vertices*; V has a corresponding set $\mathcal{V}_T$ of *temporal vertices*, which consists of vertices in specific timestamps, defined as follows:

$$\mathcal{V}_T = \{(v, t) : v \in V \wedge t \in [T]\}.$$

3. A set $E = E(G)$ of *temporal edges*, which satisfies:

$$E \subseteq \{(u, t)(v, t) : u, v \in V \wedge t \in [T] \wedge u \neq v\}.$$

In the notation, we usually use standard lettering for sets of vertices and calligraphic letters for sets of temporal vertices, e.g., V versus $\mathcal{V}_T$. As for edges, there is no notion of non-temporal edges and so no such distinction is needed.

For a directed (static) graph H, we denote by (u, v) an arc from vertex u to vertex v (we consider only directed static graphs, but not directed temporal graphs).

Given a temporal graph $G = (V, E, T)$ and a set of vertices $B \subseteq V$, we define the set $\tau(B)$ of all temporal vertices of B across all times:

$$\tau(B) = \{(v, t) : v \in B \wedge t \in [T]\}.$$

If $B = \{v\}$, we may write $\tau(v)$ instead of $\tau(\{v\})$. For intuition, one may refer to Fig. 1 and think of a temporal graph as a matrix with $|V|$ rows and T columns, where rows are vertices and the t-th column has the vertices at time t arranged vertically. Then, $\tau(B)$ represents the rows corresponding to the vertices in B.

In a similar manner, we may refer to the columns of G. For a subset $I \subseteq [T]$, we denote:

$$\pi(I) = \{(v, t) : v \in V \wedge t \in I\}.$$

For a subset $W \subseteq V$ of vertices, we denote by $G[W]$ the temporal subgraph induced by $\tau(W)$, that is, the temporal graph whose vertex set is W, time domain is $[T]$ and whose edge set is

$$\{(u, t)(v, t) \in E : (u, t), (v, t) \in \tau(W)\}.$$

We also use the notation $G - W = G[V \setminus W]$. Observe that $G[W]$ and $G - W$ are temporal graphs over the same time domain as G.

In order to define the problem we are interested in, we need to define the *assignment* of a set of vertices.

Definition 2. Consider a temporal graph $G = (V, E, T)$ and a set $W \subseteq V$ of vertices. An *assignment of* W is a subset $\mathcal{X} \subseteq \tau(W)$ such that if $(u, p) \in \mathcal{X}$ and $(u, q) \in \mathcal{X}$, with $p, q \in [T]$ and $p \leq q$, then $(u, t) \in \mathcal{X}$, for each $t \in [p, q]$.

For example in Fig. 1, the highlighted temporal vertices form an assignment of $W = \{v, u, z, w\}$. If W is clear from the context or not relevant, then we may say that $\mathcal{X}$ is an assignment, without specifying W. Given an assignment $\mathcal{X}$, to refer to the set of assigned temporal vertices restricted to a subset of vertices $W \subseteq V$, we use the notation

$$\mathcal{X}[W] := \mathcal{X} \cap \tau(W) = \{(v, t) : (v, t) \in \mathcal{X} \wedge v \in W\}.$$

If $W = \{v\}$ has a single vertex, we write $\mathcal{X}[v]$ instead of $\mathcal{X}[\{v\}]$. Note that in this case $\mathcal{X}[v]$ contains temporal vertices that belong to a contiguous interval of timestamps. Consider a set $I \subseteq [T]$ of timestamps. An assignment $\mathcal{X}$ *intersects* I if there exists $(v, t) \in \mathcal{X}$ such that $t \in I$ (in other words, $\mathcal{X} \cap \pi(I) \neq \emptyset$).

Now, we give the definition of *temporal cover*.

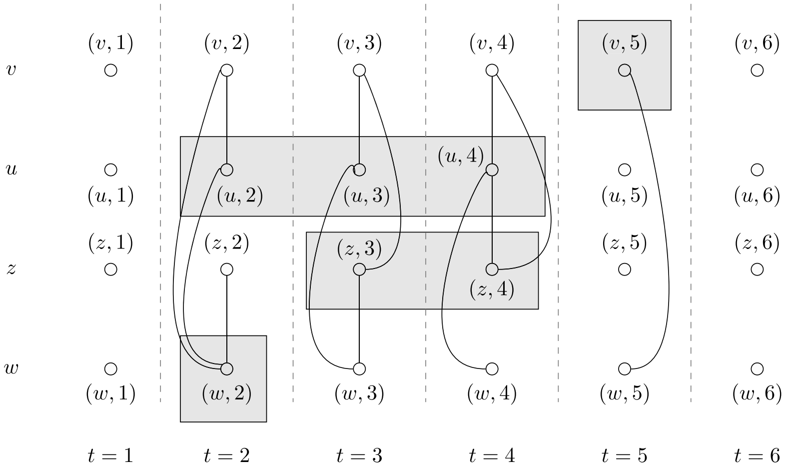

Fig. 1. An example of k-TimelineCover on a temporal graph G consisting of four vertices and six timestamps. For each timestamp, we draw the temporal edges of G, for example for $t=2$, the temporal edges are $(v,2)(u,2)$, $(v,2)(w,2)$, $(u,2)(w,2)$, $(z,2)(w,2)$. Also note that in $t=1$ and $t=6$ no temporal edge is defined. A temporal cover $\mathcal{X}=\{(v,5),(u,2),(u,3),(u,4),(z,3),(z,4),(w,2)\}$ is represented with gray rectangles. Note that $sp(\mathcal{X})=3$.

Definition 3. Given a temporal graph $G=(V,E,T)$ a *temporal cover* of G is an assignment $\mathcal{X}\subseteq\mathcal{V}_T$ of V such that the following properties hold:

1. For each $v\in V$, $\mathcal{X}[v]$ is non-empty;
2. For each $(u,t)(v,t)\in E$, with $t\in[T]$, at least one of (u,t), (v,t) is in $\mathcal{X}$.

For a temporal cover $\mathcal{X}$ of G, the *span* of v in $\mathcal{X}$ is defined as:

$$sp(v,\mathcal{X})=\max\{t_2-t_1:(v,t_1)\in\mathcal{X},(v,t_2)\in\mathcal{X}\}.$$

Note that if a temporal cover $\mathcal{X}$ contains, for a vertex $v\in V$, a single temporal vertex (v,t), then $sp(v,\mathcal{X})=0$. The span of $\mathcal{X}$, denoted by $sp(\mathcal{X})$, is then defined as:

$$sp(\mathcal{X})=\sum_{v\in V}sp(v,\mathcal{X}).$$

The definition of temporal cover requires that for each vertex at least one of its associated temporal vertices belongs to the cover. This is not strictly necessary, since it might be possible to cover every temporal edge without this condition. However, this condition simplifies some of the definitions and proofs below. Note that if an assignment of a vertex is not needed to cover temporal edges, we can assign the temporal vertex to some timestamp without increasing the span.

Now, we are able to define k-TimelineCover (an example is presented in Fig. 1).

Problem 1. (k-TimelineCover)
Input: A temporal graph $G=(V,E,T)$, an integer k.
Question: Does there exist a temporal cover of G of span at most k?

A temporal cover $\mathcal{S}^*\subseteq\mathcal{V}_T$ of span at most k will sometimes be called a *solution*. Our goal is to determine whether k-TimelineCover is FPT in parameter k.

3. An FPT algorithm

In this section we present our FPT algorithm, which consists of two parts:

1. The iterative compression technique.
2. A reduction to the Constrained Digraph Pair Cut problem.

Before presenting the details of our algorithm, we present the main idea and some definitions. Recall that our parameter, that is the span of a solution of k-TimelineCover, is denoted by k.

Consider a temporal graph G and assume we have a temporal cover $\mathcal{S}^*$ of span at most k of the subgraph $G-\{w\}$, for some vertex $w\in V$. The idea of the iterative compression step is, starting from $\mathcal{S}^*$, to show how to decide in

FPT time whether there exists a solution of k-TIMELINECOVER for G. This is done by solving a subproblem, called RESTRICTED TIMELINE COVER, where we must modify $\mathcal{S}^*$ to consider also w. A solution to this subproblem is computed by branching on the assignments of vertices having a positive span in $\mathcal{S}^*$ and on w, and then reducing the problem to CONSTRAINED DIGRAPH PAIR CUT. RESTRICTED TIMELINE COVER is defined as follows.

Problem 2. (RESTRICTED TIMELINE COVER)
Input: A temporal graph $G = (V, E, T)$, a vertex $w \in V$, an integer k, a temporal cover $\mathcal{S}^*$ of $G - \{w\}$ of span at most k.
Output: Does there exist a temporal cover of G of span at most k?

For technical reasons that will become apparent later, we will assume that the temporal graph contains no edge at timestamps 1 and T, i.e. for every $(u,t)(v,t) \in E$, we have $t \in [2, T-1]$ (as in Fig. 1). In particular, this avoids considering different gadget definitions in the reduction to CONSTRAINED DIGRAPH PAIR CUT, as the cases where a vertex is assigned the first or the last of its associated temporal vertex behaves somehow differently. It is easy to see that if this is not already the case, we can add two such “dummy” timestamps without any temporal edges.

Informally, if we are able to solve RESTRICTED TIMELINE COVER in FPT time, then we can obtain an FPT algorithm for k-TIMELINECOVER as well. Indeed, we can first compute a temporal cover on a small subset of vertices (for example a single vertex), and then we can add, one at a time, the other vertices of the graph. This requires at most $|V|$ iterations, and each time a vertex is added, we compute a solution of RESTRICTED TIMELINE COVER to check whether it is possible to find a temporal cover of span at most k after the addition of a vertex.

Iterative compression

We now present our approach based on iterative compression to solve the RESTRICTED TIMELINE COVER problem. Given a solution $\mathcal{S}^*$ for $G - \{w\}$, we focus on the vertices of V that have a positive span in $\mathcal{S}^*$ and vertex w. An example of our approach, that illustrates the sets of vertices and temporal vertices used by the algorithm, is presented in Fig. 2, along with a smaller and more focused example in Fig. 3.

While our approach is inspired by iterative compression, there are some specific properties we point out here. When we solve RESTRICTED TIMELINE COVER, given a solution $\mathcal{S}^*$ we have to consider three sets of vertices (and associated temporal vertices). Indeed, first we can distinguish between vertices that have positive span in $\mathcal{S}^*$ and vertices that have span zero in $\mathcal{S}^*$. As in iterative compression approach, we can enumerate in FPT time every subset of vertices having positive span in $\mathcal{S}^*$, but we cannot enumerate every assignment of these vertices, since T is not a function of k (and so the number of temporal vertices). Thus, our approach “guesses” the subset of vertices having positive span in $\mathcal{S}^*$ that will be assigned to an interval that includes a timestamp where some positive span vertex is assigned in $\mathcal{S}^*$, since the number of these timestamps is bounded by $2k$. For the vertices in this set we can enumerate the possible assignments. For the remaining vertices having positive span in $\mathcal{S}^*$, we cannot enumerate their assignments, but they have some specific properties, that will help us in computing a solution of RESTRICTED TIMELINE COVER, for example they will force some assignments of other vertices.

Consider the input of RESTRICTED TIMELINE COVER that consists of a temporal graph $G = (V, E, T)$, a vertex $w \in V$, an integer k and a temporal cover $\mathcal{S}^* \subseteq \mathcal{V}_T \setminus \tau(w)$ of $G - \{w\}$ of span at most k. Define the following sets associated with $\mathcal{S}^*$:

$$\mathcal{S} = \{(v,t) \in \mathcal{S}^* : sp(v, \mathcal{S}^*) \geq 1\} \cup \tau(w)$$

$$V_{\mathcal{S}} = \{v \in V : sp(v, \mathcal{S}^*) \geq 1\} \cup \{w\}.$$

The set $V_{\mathcal{S}}$ is defined as the set of vertices having span greater than 0 in $\mathcal{S}^*$, plus the vertex w. Then, $\mathcal{S}$ contains (1) the actual temporal vertices of $\mathcal{S}^*$ that contribute to a positive span, plus (2) all the temporal vertices corresponding to the vertex w in every timestamp.

In a similar vein, we define the sets of vertices of $\mathcal{S}^*$ having a span equal to 0 (see Fig. 2):

$$Z_{\mathcal{S}} = V \setminus V_{\mathcal{S}}.$$

Note that the Z in $Z_{\mathcal{S}}$ stands for “zero”. Next, define the following set $I_{\mathcal{S}}$ of timestamps associated with $V_{\mathcal{S}} \setminus \{w\}$:

$$I_{\mathcal{S}} = \{t \in [T] : (u,t) \in \mathcal{S} \text{ for some } u \in V_{\mathcal{S}} \setminus \{w\}\}.$$

Essentially, $I_{\mathcal{S}}$ contains those timestamps where the vertices of $V_{\mathcal{S}} \setminus \{w\}$, that is of span greater than zero, have associated temporal vertices in $\mathcal{S}$. These timestamps are essential for computing a solution of RESTRICTED TIMELINE COVER, that is to compute whether there exists a temporal cover of G of span at most k starting from $\mathcal{S}$. First, we show two easy properties of $\mathcal{S}^*$ and $I_{\mathcal{S}}$ on the temporal graph $G - \{w\}$.

Lemma 1. *Let $\mathcal{S}^*$ be a solution of k-TIMELINECOVER on instance $G - \{w\}$ and let $I_{\mathcal{S}}$ be the associated set of timestamps. Then $|I_{\mathcal{S}}| \leq 2k$.*

Proof. The result follows from the fact that, since $|V_{\mathcal{S}} \setminus \{w\}| \leq k$ and $V_{\mathcal{S}} \setminus \{w\}$ contains only vertices of positive span, $2k$ timestamps contribute at least k to the span of $\mathcal{S}^*$. □

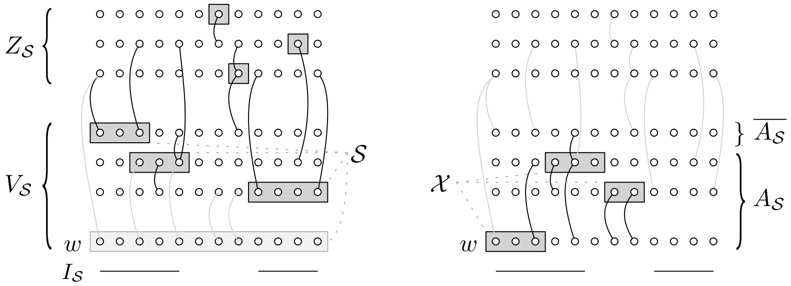

Fig. 2. An illustration of the main concepts used in the iterative compression step. Each row represents a vertex and each column a timestamp. The first and last columns, which are assumed to contain no edges, are not shown. We assume $k = 7$. Left: the temporal vertices of a temporal vertex cover $\mathcal{S}^*$ of $G - \{w\}$ are highlighted in dark gray rectangles (the w temporal vertices are not in $\mathcal{S}^*$, although they are included in $\mathcal{S}$). The gray rectangles in the $V_{\mathcal{S}}$ part represent $\mathcal{S}$ (where now $\tau(w)$ is included), and the gray rectangles in the $Z_{\mathcal{S}}$ part represent the representatives (to be defined in the reduction to Constrained Digraph Pair Cut). Right: a feasible assignment $\mathcal{X}$, the temporal vertices in the gray boxes, of span 5 (edges with an endpoint in $\tau(Z_{\mathcal{S}})$ are grayed because they are not relevant for feasible assignments).

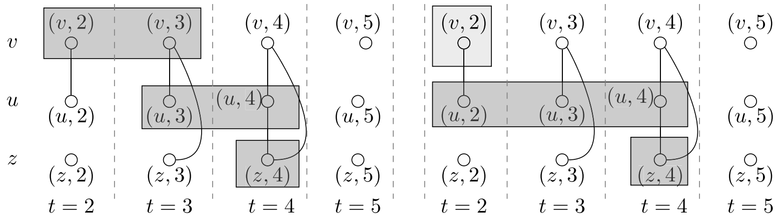

Fig. 3. An example of application of iterative compression (timestamps 1 and 6 are not shown as they are edgeless, also temporal vertex w is not shown, its assignment is defined as in Fig. 1). In the left part, we represent solution $\mathcal{S}^* = \{(v,2),(v,3),(u,3),(u,4),(z,4)\}$, where the vertices in $\mathcal{S}^*$ are highlighted with gray rectangles. Note that $I_{\mathcal{S}} = \{2,3,4\}$, $V_{\mathcal{S}} = \{v,u\}$, $\mathcal{S} \setminus \tau(w) = \{(v,2),(v,3),(u,3),(u,4)\}$, $Z_{\mathcal{S}} = \{z\}$. In the right part, we represent in gray a feasible assignment $\mathcal{X}$ associated with $\mathcal{S}$, containing temporal vertices $(u,2)$, $(u,3)$, $(u,4)$; in light gray we highlight the temporal vertex corresponding to $\overline{A_{\mathcal{S}}} = \{v\}$. The reduction to Constrained Digraph Pair Cut eventually leads to the solution of k-TimelineCover represented in Fig. 1.

We next argue that if a temporal edge is not covered by an element of $\mathcal{S}$, then we need temporal vertices associated with $Z_{\mathcal{S}}$ to cover it. Recall that we use the notation $\mathcal{S}^*[Z_{\mathcal{S}}] = \mathcal{S}^* \cap \tau(Z_{\mathcal{S}})$.

Lemma 2. *Let $\mathcal{S}^*$ be a solution of k-TimelineCover on instance $G - \{w\}$. Then, for each vertex $v \in Z_{\mathcal{S}}$, $sp(v, \mathcal{S}^*) = 0$. Moreover, $\mathcal{S}^*[Z_{\mathcal{S}}]$ covers each temporal edge of $G - \{w\}$ not covered by $\mathcal{S} \setminus \tau(w)$.*

Proof. Since $\mathcal{S}^*$ is a cover of $G - \{w\}$ and $\mathcal{S} \setminus \tau(w)$ contains all temporal vertices of $\mathcal{S}^*$ associated with $V_{\mathcal{S}} \setminus \{w\}$, it follows that the temporal edges not covered by $\mathcal{S} \setminus \tau(w)$ must be covered by the temporal vertices of $\mathcal{S}^*$ associated with vertices of $Z_{\mathcal{S}}$, that is $\mathcal{S}^*[Z_{\mathcal{S}}]$. Furthermore, by the definition of $Z_{\mathcal{S}}$, it follows that for each vertex $v \in Z_{\mathcal{S}}$, $sp(v, \mathcal{S}^*) = 0$. □

Now, we introduce the concept of feasible assignment, which is used to "guess" how $\mathcal{S}^*$ is rearranged in a solution of Restricted Timeline Cover.

Definition 4 *(Feasible assignment).* Consider an instance of Restricted Timeline Cover that consists of a temporal graph $G = (V, E, T)$, a vertex $w \in V$, an integer k, a temporal cover $\mathcal{S}^*$ of $G - \{w\}$ of span at most k, and sets $\mathcal{S}$, $V_{\mathcal{S}}$, and $I_{\mathcal{S}}$ associated with $\mathcal{S}^*$.

We say that an assignment $\mathcal{X}$ of $V_{\mathcal{S}}$ is a *feasible assignment* (with respect to G and $\mathcal{S}^*$) if all of the following conditions hold:

1. the span of $\mathcal{X}$ is at most k;
2. every edge of $G[V_{\mathcal{S}}]$ is covered by $\mathcal{X}$;
3. $\mathcal{X}[w]$ is non-empty;
4. for every $v \in V_{\mathcal{S}} \setminus \{w\}$, at least one of the following holds:
    (a) $\mathcal{X}[v]$ is empty;
    (b) $\mathcal{X}[v]$ intersects with $I_{\mathcal{S}}$; or
    (c) $\mathcal{X}[v]$ contains a temporal vertex (v,t) such that $(v,t)(w,t) \in E$ and $(w,t) \notin \mathcal{X}$.

Given a feasible assignment $\mathcal{X}$, we denote

$$A_{\mathcal{S}}(\mathcal{X}) = \{v \in V_{\mathcal{S}} : X[v] \neq \emptyset\} \qquad \overline{A_{\mathcal{S}}}(\mathcal{X}) = \{v \in V_{\mathcal{S}} : X[v] = \emptyset\}.$$

One can check that Fig. 2 on the right exhibits a feasible assignment with respect to G and $\mathcal{S}^*$ (cases 4.(a), 4.(b), and 4.(c) occur on the first, second, and third rows of $V_{\mathcal{S}}$, respectively). Informally, a feasible assignment $\mathcal{X}$ specifies how a desired solution should intersect with $\tau(w)$ (point 3) and how other temporal vertex intervals should intersect or not with $I_{\mathcal{S}}$ (point 4). Point 4 considers the possible cases, which are not mutually exclusive, for a feasible assignment of the temporal vertices of a vertex $v \in V_{\mathcal{S}} \setminus \{w\}$:

- None of the associated temporal vertices in $I_{\mathcal{S}}$ belongs to the computed solution (Case 4.(a)).
- Some of its associated temporal vertices in $I_{\mathcal{S}}$ belongs to the solution (Case 4.(b))
- Some of the (v,t) temporal vertices are forced, since they belong to an edge $(v,t)(w,t)$ with $t \in I_{\mathcal{S}}$, that we know is not covered by (w,t) (Case 4.(c))

The set $A_{\mathcal{S}}(\mathcal{X})$ refers to those vertices whose interval is chosen, thus "assigned" (hence the A), and shall not be questioned from now on. The complementary set $\overline{A_{\mathcal{S}}}(\mathcal{X})$ of vertices $v \in V_{\mathcal{S}}$ such that $\mathcal{X}[v] = \emptyset$, which result from Case 4.(a), are interpreted as having vertices whose time is in $I_{\mathcal{S}}$ being "not assigned" yet (hence the $\overline{A}$), and indicate that the desired solution should contain unassigned vertices not in a time in $I_{\mathcal{S}}$. We need to allow those cases to bound the number of feasible assignments to enumerate.

Since G and $\mathcal{S}^*$ are fixed in the remainder, we assume that all feasible assignments are with respect to G and $\mathcal{S}^*$ without explicit mention. We now relate feasible assignments to temporal covers.

Definition 5. Let $\mathcal{X}^*$ be a temporal cover of G and let $\mathcal{X}$ be a feasible assignment. We say that $\mathcal{X}^*$ *agrees* with $\mathcal{X}$ if:

- for each $v \in A_{\mathcal{S}}(\mathcal{X})$, $\mathcal{X}^*[v] = \mathcal{X}[v]$;
- for each $v \in \overline{A_{\mathcal{S}}}(\mathcal{X})$ and each $t \in I_{\mathcal{S}}$, $\mathcal{X}^*$ contains every neighbor (u,t) of (v,t) such that $(u,t) \in \tau(Z_{\mathcal{S}})$.

The intuition of $\mathcal{X}^*$ agreeing with $\mathcal{X}$ is as follows. For $v \in A_{\mathcal{S}}(\mathcal{X})$, $\mathcal{X}$ "knows" which temporal vertices of $\tau(v)$ should be in the solution, and for agreement we require $\mathcal{X}^*$ to contain exactly those. For $v \in \overline{A_{\mathcal{S}}}(\mathcal{X})$, we interpret that $\mathcal{X}$ does not want any temporal vertex (v,t) with $t \in I_{\mathcal{S}}$. Thus, to cover the edges incident to (v,t) that go outside of $V_{\mathcal{S}}$, we require $\mathcal{X}^*$ to contain the other endpoint (u,t). Note an important subtlety: we act "as if" $\mathcal{X}^*$ should not contain (v,t) or other temporal vertices of $\overline{A_{\mathcal{S}}}(\mathcal{X})$ with timestamp in $I_{\mathcal{S}}$, but the definition does not forbid it. Hence, $\mathcal{X}^*$ *can* contain a temporal vertex of $\overline{A_{\mathcal{S}}}(\mathcal{X})$ in some timestamps of $I_{\mathcal{S}}$, as long as $\mathcal{X}^*$ contains also its neighbors (at time $I_{\mathcal{S}}$) outside $V_{\mathcal{S}}$.

The main purpose of feasible assignments and agreement is as follows.

Lemma 3. *Let $\mathcal{X}^*$ be a temporal cover of G of span at most k. Then there exists a feasible assignment $\mathcal{X}$ such that $\mathcal{X}^*$ agrees with $\mathcal{X}$.*

Proof. Construct $\mathcal{X} \subseteq \mathcal{X}^*$ as follows: first add $\mathcal{X}^*[w]$ to $\mathcal{X}$, and then for $v \in V_{\mathcal{S}} \setminus \{w\}$, add $\mathcal{X}^*[v]$ to $\mathcal{X}$ if and only if $\mathcal{X}^*[v]$ intersects with the set $I_{\mathcal{S}}$, or if it contains a temporal vertex (v,t) incident to an edge $(v,t)(w,t) \in E$ such that $(w,t) \notin \mathcal{X}^*[w]$. Note that since $\mathcal{X}^*$ is an assignment of V, $\mathcal{X}$ is an assignment of $V_{\mathcal{S}}$ (there may be some $v \in V_{\mathcal{S}}$ such that $\mathcal{X}[v]$ is empty, but recall that this is allowed by the definition of an assignment).

We first focus on arguing that $\mathcal{X}$ satisfies each condition of a feasible assignment (Definition 4). For Condition 1, since $\mathcal{X}^*$ has span at most k and $\mathcal{X} \subseteq \mathcal{X}^*$, it is clear that $\mathcal{X}$ also has span at most k. For Condition 3, $\mathcal{X}^*[w]$ is non-empty by the definition of a temporal cover, and we added all of $\mathcal{X}^*[w]$ to $\mathcal{X}$. For Condition 4, we explicitly require in our construction of $\mathcal{X}$ that for each $v \in V_{\mathcal{S}} \setminus \{w\}$, if $\mathcal{X}[v]$ is non-empty, then it is equal to $\mathcal{X}^*[v]$ and it either intersects with $I_{\mathcal{S}}$ or covers an edge not covered by $\mathcal{X}^*[w] = \mathcal{X}[w]$.

Let us focus on Condition 2, which asks to cover edge of $G[V_{\mathcal{S}}]$. Let $(u,t)(v,t) \in E(G[V_{\mathcal{S}}])$. If $u = w$, then if we did not add (w,t) to $\mathcal{X}$, then $\mathcal{X}^*$ must contain (v,t) and we added $\mathcal{X}^*[v]$ to $\mathcal{X}$, thereby covering the edge. The same holds if $v = w$. Assume $u \neq w, v \neq w$, and suppose without loss of generality that $\mathcal{X}^*$ contains (u,t) to cover the edge. Since both u and v belong to $V_{\mathcal{S}}$, and since at least one of (u,t) or (v,t) must be in $\mathcal{S}^*$, then at least one of those is also in $\mathcal{S}$. It follows that $t \in I_{\mathcal{S}}$ and that $\mathcal{X}^*[u]$ intersects with $I_{\mathcal{S}}$, and by our construction of $\mathcal{X}$, it holds that $(u,t) \in \mathcal{X}$.

We deduce that $\mathcal{X}$ covers every edge of $G[V_{\mathcal{S}}]$. Therefore, $\mathcal{X}$ is a feasible assignment.

It remains to show that $\mathcal{X}^*$ agrees with $\mathcal{X}$. For $v \in A_{\mathcal{S}}(\mathcal{X})$, $\mathcal{X}^*[v] = \mathcal{X}[v]$ by the construction of $\mathcal{X}$. For $v \in \overline{A_{\mathcal{S}}}(\mathcal{X})$, there is no $(v,t) \in \mathcal{X}^*$ with $t \in I_{\mathcal{S}}$, as otherwise we would have added $\mathcal{X}^*[v]$ to $\mathcal{X}$. For every such (v,t), $\mathcal{X}^*$ must contain all of its neighbors in $\tau(Z_{\mathcal{S}})$ to cover the edges, as required by the definition of agreement. □

Assuming that a solution $\mathcal{X}^*$ exists, Lemma 3 only argues the existence of a feasible assignment $\mathcal{X}$ that agrees with it, but does not say how to find such an $\mathcal{X}$. To achieve this, we will simply enumerate *all* feasible assignments, and for each of them check if it agrees with some solution.

It therefore remains to show that the number of feasible assignments is bounded by a function of k, and can be enumerated in FPT time. We first show that the latter can be achieved through the following steps. Start with $\mathcal{X}$ as an empty set and then apply the following steps:

(1) *Choose the w interval:* branch into every non-empty assignment $\mathcal{X}_w$ of $\{w\}$ of span at most k. In each branch, add the chosen subset $\mathcal{X}_w$ to $\mathcal{X}$;
(2) *Include forced temporal vertices:* for every edge $(v,t)(w,t) \in E(G[V_{\mathcal{S}}])$ such that $(w,t) \notin \mathcal{X}_w$, add (v,t) to $\mathcal{X}$;
(3) *Choose starting points:* for every $v \in V_{\mathcal{S}} \setminus \{w\}$, such that $\mathcal{X}[v] = \emptyset$ at this moment, branch into $|I_{\mathcal{S}}|+1$ options: either add no temporal vertex of $\tau(v)$ to $\mathcal{X}$, or choose a temporal vertex (v,t) and add it to $\mathcal{X}$, where $t \in I_{\mathcal{S}}$;
(4) *Extend current intervals:* for every $v \in V_{\mathcal{S}} \setminus \{w\}$ such that $\mathcal{X}[v] \neq \emptyset$ at this moment, branch into every assignment $\mathcal{X}_v$ of $\{v\}$ of span at most k that contains all temporal vertices in $\mathcal{X}[v]$ (note that there could be several such vertices because of step 2 above; if no such assignment exists, abort the current branch). For each such branch, add every temporal vertex of $\mathcal{X}_v \setminus \mathcal{X}$ to $\mathcal{X}$.
At the end of each branching, we obtain a set $\mathcal{X}$ of temporal vertices. We check whether $\mathcal{X}$ satisfies all conditions of a feasible assignment: if it does, we add $\mathcal{X}$ to the set of enumerated assignments, and otherwise we do not.

We now bound the time required to obtain every feasible assignment. Note that because these assignments only intersect with $\tau(V_{\mathcal{S}})$, whose size is $O(Tk)$, the running time only depends on T and k.

Theorem 1. *The above steps enumerate every feasible assignment in time* $O(2^{4k\log k}T^3k^4)$, *where* $n = |V|$.

Proof. We first argue that every feasible assignment is enumerated. Consider a feasible assignment $\mathcal{X}$. First, consider the set $\mathcal{X}_w = \mathcal{X}[w]$, which is non-empty by Condition 3 of feasible assignments. Since in Step (1) we branch into every non-empty assignment of $\{w\}$, then we eventually enumerate $\mathcal{X}_w$. In what follows, we assume that we are in the branch where $\mathcal{X}_w$ is added in the first step.

Now, consider a vertex $v \in V_{\mathcal{S}} \setminus \{w\}$ and consider the set $\mathcal{X}[v]$. If no temporal vertex of $\tau(v)$ belongs to $\mathcal{X}$ (hence $\mathcal{X}[v]$ is empty), then Step (3) enumerates this case (note that Step (2) does not add a temporal vertex of $\tau(v)$ either: $\mathcal{X}[v]$ empty implies that there is no edge of the form $(v,t)(w,t)$ with $(w,t) \notin \mathcal{X}$, since $\mathcal{X}$ must cover this edge). Assume that $\mathcal{X}[v]$ is non-empty. According to Conditions 4.(b) and 4.(c) of a feasible assignment, there are only two ways for $\mathcal{X}[v]$ to be non-empty: either some (v,t) is in $\mathcal{X}[v]$ to cover an edge incident to a temporal vertex of $\tau(w)$ not covered by $\mathcal{X}_w$, or some (v,t) is in a timestamp of $I_{\mathcal{S}}$. In the former case, (v,t) is added in Step (2), and in the latter case, one of the branches of Step (3) will add (v,t) to the set under construction. Since the span of $\mathcal{X}$ and hence of $\mathcal{X}[v]$ is at most k, it follows that $\mathcal{X}[v]$ is an assignment of $\{v\}$ of span at most k, and Step (4) will branch into a case where it adds $\mathcal{X}[v]$. Since this is true for every v, it follows that $\mathcal{X}$ will be enumerated at some point.

Now, we discuss the number of feasible assignments enumerated by Steps (1) – (4). Step (1) is computed in $O(Tk)$ time, as there are $O(Tk)$ possible non-empty intervals $\mathcal{X}_w$ of span at most k (to see this, note that there are $O(T)$ choices for the starting point of the interval, and once this is chosen, there are $O(k)$ possibilities for the finishing point of the interval). For each branch defined at Step (1), Step (2) can be computed in Tk time as the temporal vertices of w can have at most Tk neighbors in $G[V_{\mathcal{S}}]$. Note that because each vertex of $V_{\mathcal{S}}$ has a positive span, we have $|V_{\mathcal{S}} \setminus \{w\}| \leq k$. This implies that there are at most $2k$ forced temporal vertices, as otherwise the overall span is greater than k. Also observe that Step (2) does not perform any branching.

Step (3), for each vertex $v \in V_{\mathcal{S}} \setminus \{w\}$, branches into at most $2k+1$ cases, as from Lemma 1 we have that $|I_{\mathcal{S}}| \leq 2k$. There are at most k vertices in $V_{\mathcal{S}} \setminus \{w\}$, since each such vertex contributes at least one to the span of $\mathcal{S}^*$, and it follows that the number of branches explored in Step (3) is at most $(2k+1)^k$.

As for Step (4), for each branch defined in Step (3) for $v \in V_{\mathcal{S}} \setminus \{w\}$ such that $\mathcal{X}[v] \neq \emptyset$, it branches over $O(k^2)$ possible choices of timestamps a, b that are endpoints of $\mathcal{X}[v]$. Again since $|V_{\mathcal{S}}| \leq k$, the number of branches explored in Step (4) is at most $O(k^{2k})$.

Thus the overall time to enumerate all considered sets with Steps (1) – (4) is $O(Tk \cdot (Tk + (2k+1)^k k^{2k}))$, where the addition of the Tk term after the parenthesis is because of the time to compute Step (2), and the other terms and factors are due to branching. To simplify, we upper bound this complexity with $O(2^{4k\log k}T^2k^2)$.

Lastly, for each such set, we must check whether it is indeed a feasible assignment. Each of the four conditions can be checked, in the worst case, by traversing the whole $G[V_{\mathcal{S}}]$ subgraph and checking the condition. The subgraph has $O(k)$ vertices, $O(Tk)$ temporal vertices, and $O(Tk^2)$ temporal edges, so we upper bound the required time for checks by $O(Tk^2)$. Multiplying the number of enumerated sets by Tk^2 yields the final complexity. □

Reducing to CONSTRAINED DIGRAPH PAIR CUT

Our strategy is now, given one feasible assignment (that can be enumerated thanks to Theorem 1) to verify whether there is a temporal cover that agrees with it.

More specifically, consider a feasible assignment $\mathcal{X} \subseteq \tau(V_{\mathcal{S}})$, enumerated at some point from the given solution $\mathcal{S}^*$ to $G - \{w\}$. Our goal is to decide whether G admits a temporal cover $\mathcal{X}^*$ of span at most k that agrees with $\mathcal{X}$. Since we branch over every possible feasible assignment $\mathcal{X}$, if there is a temporal cover $\mathcal{X}^*$ of G of span at most k, then by Theorem 1 our enumeration will eventually consider an $\mathcal{X}$ that $\mathcal{X}^*$ agrees with, and hence we will be able to decide of the existence of $\mathcal{X}^*$.

We show that finding $\mathcal{X}^*$ reduces to the CONSTRAINED DIGRAPH PAIR CUT problem, as we define below. For a directed graph H, we denote its set of arcs by $A(H)$ (to avoid confusion with $E(G)$, which is used for the edges of an undirected graph G). For $F \subseteq A(H)$, we write $H - F$ for the directed graph with vertex set $V(H)$ and arc set $A(H) \setminus F$. Recall that in our setting, digraphs are standard directed graphs and are not temporal.

Problem 3. (CONSTRAINED DIGRAPH PAIR CUT)
Input: A directed graph $H = (V(H), A(H))$, a source vertex $s \in V(H)$, a set of vertex pairs $P \subseteq \binom{V(H)}{2}$ called *forbidden pairs*, a subset of arcs $D \subseteq A(H)$ called *deletable arcs*, and an integer k'.
Output: Does there exist a set of arcs $F \subseteq D$ of H such that $|F| \leq k'$ and such that, for each $\{u, v\} \in P$, at least one of u or v is not reachable from s in $H - F$?

It is known that CONSTRAINED DIGRAPH PAIR CUT can be solved in time $O^*(2^{k'})$ [19], but a few remarks are needed before proceeding. In [19], the authors only provide an algorithm for the *vertex-deletion* variant, and do not consider deletable/undeletable arcs. We show at the end of the section that we can make an arc undeletable by adding enough parallel paths between the two endpoints, and that our formulation of CONSTRAINED DIGRAPH PAIR CUT reduces to the simple vertex-deletion variant.

So let us fix a feasible assignment $\mathcal{X}$ for the remainder of the section. We will denote

$$A_{\mathcal{S}} = A_{\mathcal{S}}(\mathcal{X}) \qquad \overline{A_{\mathcal{S}}} = \overline{A_{\mathcal{S}}}(\mathcal{X}),$$

recalling that $\overline{A_{\mathcal{S}}}$ denotes vertices v of $V_{\mathcal{S}}$ that do not have an assigned set of vertices, so that $\mathcal{X}[v] = \emptyset$ and, in particular, do not intersect with $I_{\mathcal{S}}$ (while $A_{\mathcal{S}}$ denotes those v with $\mathcal{X}[v]$ non-empty).

We first find the main intuitions of the reduction. We know how $\mathcal{X}^*$ must intersect with temporal vertices in $\tau(A_{\mathcal{S}})$, by the notion of agreement. It remains to decide how $\mathcal{X}^*$ should intersect with the temporal vertices of $\overline{A_{\mathcal{S}}} \cup Z_{\mathcal{S}}$. Consider $v \in Z_{\mathcal{S}}$ first. Recall that there is a unique $i \in [T]$ with $(v, i) \in \mathcal{S}^*$. We use (v, i) as a "*representative*" of v, noting that the set of representatives of all the $v \in Z_{\mathcal{S}}$ vertices cover all the edges in $G[Z_{\mathcal{S}}]$. For $v \in \overline{A_{\mathcal{S}}}$, we just put $(v, 2)$ as an arbitrary representative.[3] We then construct gadgets for the following task: given the enforced temporal vertices from $A_{\mathcal{S}}$, how can we modify the representatives of the $Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$ vertices, possibly choosing longer spans instead, to find a temporal cover that agrees with $\mathcal{X}$?

To construct H, for each vertex $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$ whose representative is (v, i), we introduce in H the set of vertices $v_1^+, \ldots, v_{i-1}^+, v_i^-, v_{i+1}^+, \ldots, v_T^+$, where v_i^- has a "special" sign because it is a representative. A source s is also added to H, with the key intuition being that in a solution $H - F$ to our CONSTRAINED DIGRAPH PAIR CUT instance, we aim to have the following (see Claim 1):

- If s reaches some v_t^+ vertex in $H - F$, then the corresponding temporal vertex (v, t) must be in the solution $\mathcal{X}^*$ (and otherwise (v, t) should not be in);
- If s reaches some v_t^-, then the corresponding temporal vertex (v, t) must *not* be in the solution (and otherwise (v, t) should be in).

The set $\mathcal{X}^*$ obtained from this idea should cover all the edges not already covered by $\mathcal{X}$. Using the forbidden pairs, such a construction is not too hard to achieve. For example, if $(u, i)(v, i) \in E(G)$ with $(u, i), (v, i)$ that are both representatives, u_i^-, v_i^- exist. To force the coverage of the edge, we add the forbidden pair $\{u_i^-, v_i^-\}$. In this manner, s is forced to *not* reach one of the two, which corresponds to adding either (u, i) or (v, i) to cover the edge $(u, i)(v, i)$ (for other cases, see details below). The most technical challenge we need to deal with is that the span of $\mathcal{X}^*$ should correspond to the number of arcs to delete from H, as will be detailed below.

In what follows, given a vertex $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$, we denote by $r(v)$ the temporal vertex which is a *representative* of v, that is, for $v \in Z_{\mathcal{S}}$

$$r(v) = (v, t), \text{ with } (v, t) \in \mathcal{S}^*,$$

and for $v \in \overline{A_{\mathcal{S}}}$

$$r(v) = (v, 2).$$

Note that for any edge $(u, t)(v, t)$ with the endpoint (u, t) in $\tau(A_{\mathcal{S}}) \setminus \mathcal{X}$ or in $\tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$, we must enforce (v, t) in our cover (recall that $\pi(I_{\mathcal{S}})$ has all temporal vertices whose time is in $I_{\mathcal{S}}$).

The following observation will be useful for our reduction to CONSTRAINED DIGRAPH PAIR CUT.

---

[3] Note that representatives for $Z_{\mathcal{S}}$ are important because they cover the edges in $G[Z_{\mathcal{S}}]$ – for $\overline{A_{\mathcal{S}}}$ these representatives are meaningless, but choosing one allows us to have a uniform construction for both the $\overline{A_{\mathcal{S}}}$ and $Z_{\mathcal{S}}$ vertices.

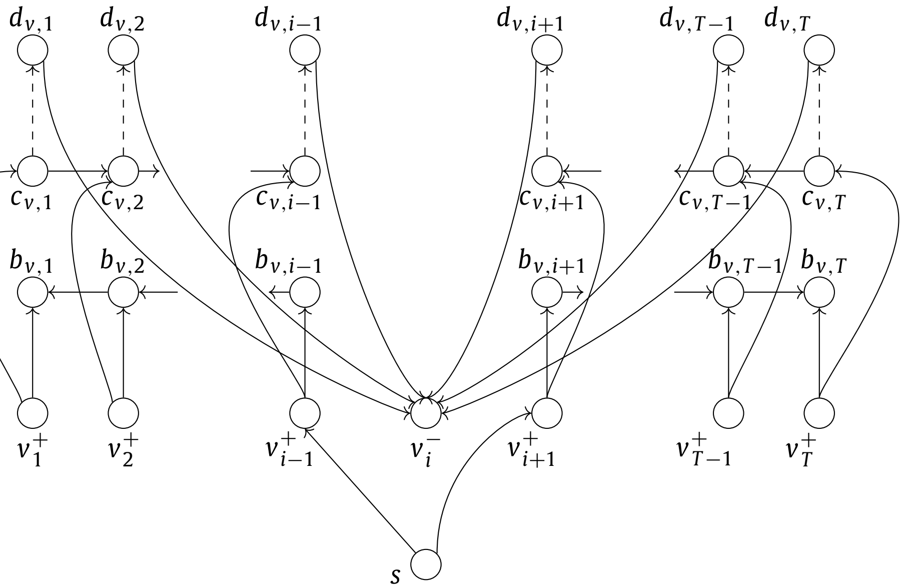

Fig. 4. Gadget for $v \in Z_{\mathcal{S}} \cup U_{\mathcal{S}}$, with $r(v) = i$, where $i \in [2, T-1]$. We assume that there exist temporal edges $(u,t)(v,t) \in E(G)$, where $t \in \{i-1, i+1\}$, such that $(u,t) \in \tau(A_{\mathcal{S}}) \setminus \mathcal{X}) \cup (\tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}}))$ and (v,t) in $\tau(Z_{\mathcal{S}})$, thus arcs from s to v_t^+ are added. The dashed arcs represent deletable arcs.

Observation 1. Let $(u,t)(v,t) \in E(G)$ such that $u \in \overline{A_{\mathcal{S}}}$ and $v \notin A_{\mathcal{S}}$. Then $v \in Z_{\mathcal{S}}$. Moreover, if $(u,t) \in \tau(\overline{A_{\mathcal{S}}}) \setminus \pi(I_{\mathcal{S}})$, then we have $r(v) = (v,t)$.

Proof. There cannot be a temporal edge $(u,t)(v,t)$ between two temporal vertices of $\tau(\overline{A_{\mathcal{S}}})$, since $\mathcal{X}$ must cover every temporal edge in $G[V_{\mathcal{S}}]$ and contains no temporal vertices of $\tau(\overline{A_{\mathcal{S}}})$ (since is defined $\tau(\overline{A_{\mathcal{S}}})$ to have an empty intersection with $\mathcal{X}$, see Definition 4). Thus $v \notin \overline{A_{\mathcal{S}}}$, and since $v \notin A_{\mathcal{S}}$, we have $v \in Z_{\mathcal{S}}$. Next suppose that (u,t) is not in $\pi(I_{\mathcal{S}})$. Then $t \notin I_{\mathcal{S}}$, implying that $(u,t) \notin \mathcal{S}^*$. Hence, the temporal edge must be covered by (v,t), which implies $r(v) = (v,t)$ by our definition of representatives. $\square$

Now, from the feasible assignment $\mathcal{X} \subseteq \tau(V_{\mathcal{S}})$, sets $\overline{A_{\mathcal{S}}}$, $A_{\mathcal{S}}$, $Z_{\mathcal{S}}$, and their representatives, we present our reduction to the Constrained Digraph Pair Cut problem. We construct an instance of this problem that consists of the directed graph $H = (V(H), A(H))$, the set of forbidden pairs $P \subseteq \binom{V(H)}{2}$, and the deletable arcs $D \subseteq A(H)$ by applying the following steps. Step 5 is the most involved and is shown in Fig. 4. The intuition of these steps is provided afterwards. Recall that for each $v \in \overline{A_{\mathcal{S}}}$, we require that a temporal edge $(v,t)(u,t)$, with $t \in I_{\mathcal{S}}$ and $u \in A_{\mathcal{S}}$, must be covered by (u,t).

1. Add to H the source vertex s;
2. For each $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$, let $r(v) = (v,i)$ be the representative. Add to H the vertices $v_1^+, \dots, v_{i-1}^+, v_i^-, v_{i+1}^+, \dots, v_T^+$.
3. *(Forced temporal vertices.)* For each temporal edge $(u,t)(v,t) \in E(G)$ with one endpoint (u,t) in $(\tau(A_{\mathcal{S}}) \setminus \mathcal{X}) \cup (\tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}}))$ and the other endpoint (v,t) in $\tau(Z_{\mathcal{S}})$, there are two cases:
   (a) if $(v,t) \neq r(v)$, add the undeletable arc (s, v_t^+) to H;
   (b) if $(v,t) = r(v)$, add the forbidden pair $\{s, v_t^-\}$ to P.
4. *(Unforced temporal vertices.)* For each temporal edge $(u,t)(v,t) \in E(G)$ with both endpoints $(u,t),(v,t)$ in $\tau(Z_{\mathcal{S}}) \cup (\tau(\overline{A_{\mathcal{S}}}) \setminus \pi(I_{\mathcal{S}}))$, there are three cases. First, note that at least one of (u,t), (v,t) is a representative of a vertex in $Z_{\mathcal{S}}$. Indeed, if $u, v \in Z_{\mathcal{S}}$, this is because one of $r(u)$ or $r(v)$ must cover the temporal edge, and if $u \in \overline{A_{\mathcal{S}}}$, then $r(v) = (v,t)$ and $v \in Z_{\mathcal{S}}$ by Observation 1 (or if $v \in \overline{A_{\mathcal{S}}}$, $r(u) = (u,t)$ and $u \in Z_{\mathcal{S}}$). The subcases are then:
   (a) if $r(u) = (u,t)$ and $r(v) = (v,t)$, add the pair $\{u_t^-, v_t^-\}$ to P;
   (b) if $r(u) = (u,t)$, $(v,t) \neq r(v)$, add the undeletable arc (u_t^-, v_t^+) to H;
   (c) if $r(v) = (v,t)$, $(u,t) \neq r(v)$, add the undeletable arc (v_t^-, u_t^+) to H;
5. *(Span gadgets.)* For $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$, create a gadget for v as follows. Add the vertices $b_{v,j}, c_{v,j}, d_{v,j}$, for $j \in [T] \setminus \{i\}$, and the set of arcs shown in Fig. 4, that is there are arcs $(v_j^+, b_{v,j})$, $(v_j^+, c_{v,j})$, $(c_{v,j}, d_{v,j})$, $(d_{v,j}, v_i^-)$, for each $j \in [T] \setminus \{i\}$ and four directed paths
   (a) from $b_{v,i-1}$ to $b_{v,1}$
   (b) from $c_{v,1}$ to $c_{v,i-1}$
   (c) from $b_{v,i+1}$ to $b_{v,T}$
   (d) from $c_{v,T}$ to $c_{v,i+1}$.

Add to D the set of deletable arcs $(c_{v,j}, d_{v,j})$, for $j \in [T] \setminus \{i\}$.
Then add the following pairs to P:
(a) $\{d_{v,h}, b_{v,j}\}$, with $1 \leq h < j \leq i-1$;
(b) $\{d_{v,h}, b_{v,j}\}$, with $i+1 \leq j < h \leq T$;
(c) $\{c_{v,h}, d_{v,j}\}$, with $1 \leq h \leq i-1$ and $i+1 \leq j \leq T$;
(d) $\{c_{v,h}, d_{v,j}\}$, with $1 \leq j \leq i-1$ and $i+1 \leq h \leq T$.
Note that, for each $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$, we have created $T + 3(T-1) = 4T - 3$ vertices in H in this step. The subgraph of H induced by these vertices will be called the *gadget corresponding to* v.

Define $k' = k - sp(\mathcal{X})$. This concludes the construction. Note that the only deletable arcs in D are the arcs $(c_{v,j}, d_{v,j})$ introduced in the last step.

As mentioned in our previous intuition, from here, the interpretation of H is that if we delete arc set F, then

(p1) For $(v,t) \neq r(v)$ we include (v,t) in $\mathcal{X}^*$ if and only if s reaches v_t^+ in $H - F$;
(p2) For $(v,t) = r(v)$ we include (v,t) in $\mathcal{X}^*$ if and only if s does *not* reach v_t^- in $H - F$.

Step 3 describes an initial set of vertices s is forced to reach and an initial set of vertices that s is forced not to reach, which correspond to temporal vertices that are forced in $\mathcal{X}^*$. First consider an edge $(u,t)(v,t)$ with endpoint (u,t) in $\tau(A_{\mathcal{S}}) \setminus \mathcal{X}$. If $(v,t) \in \tau(V_{\mathcal{S}})$, then by feasibility the edge is already covered by $\mathcal{X}$, so we only bother when $(v,t) \in \tau(Z_{\mathcal{S}})$. Since $(u,t) \notin \mathcal{X}$, (v,t) is forced in $\mathcal{X}^*$. If $(v,t) \neq r(v)$, then v_t^+ exists and adding (s, v_t^+) forces s to reach v_t^+, thereby forcing (v,t) into $\mathcal{X}^*$ to cover the edge. If $(v,t) = r(v)$, v_t^- exists and forbidding the pair $\{s, v_t^-\}$ forces s to *not* reach v_t^-, again forcing (v,t) into $\mathcal{X}^*$. A similar situation occurs when $(u,t) \in \tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$: those are assumed to not be in $\mathcal{X}^*$, so (v,t) is enforced into $\mathcal{X}^*$ in the same manner.

Step 4 describes a mechanism to cover edges $(u,t)(v,t)$ when neither endpoint is forced. This happens when one endpoint is in $\tau(Z_{\mathcal{S}})$ and the other is in the same set, or is from an unassigned vertex but not in a time in $I_{\mathcal{S}}$, i.e., in $\tau(\overline{A_{\mathcal{S}}}) \setminus \pi(I_{\mathcal{S}})$. Crucially, one of the two endpoints must be a representative of a vertex in $Z_{\mathcal{S}}$. The following cases can happen:

- When both $(u,t), (v,t)$ are representative, the forbidden pair $\{u_t^-, v_t^-\}$ in P requires that s does not reach at least one of the two, i.e., that we include at least one in $\mathcal{X}^*$.
- When only (u,t) is representative, the undeletable arc (u_t^-, v_t^+) enforces that if s reaches u_t^- (i.e. $(u,t) \notin \mathcal{X}^*$), then s reaches v_t^+ (i.e. $(v,t) \in \mathcal{X}^*$).

The relevance of representative vertices $r(v)$, with $v \in Z_{\mathcal{S}}$, is that if some temporal edge did not have an endpoint in another representative temporal vertex, an additional case would occur and we could not deal with it in our reduction.

Finally, Step 5 enforces the number of deleted arcs to correspond to the span of a solution. That is, it ensures that if we want to add to $\mathcal{X}^*$ a set of h consecutive temporal vertices of vertex $v \in Z_{\mathcal{S}}$ to our solution of Restricted Timeline Cover (so with a span equal to $h-1$), then we have to delete $h-1$ deletable arcs of the corresponding gadget of H in order to obtain a solution to Constrained Digraph Pair Cut (and vice-versa). Indeed, consider the gadget in Fig. 4. Let $v \in Z_{\mathcal{S}}$ with $r(v) = (v,i)$ and suppose for instance that s reaches v_l^+ and v_r^+, with $l < i < r$, as in Fig. 5. This corresponds to adding $(v,l), \ldots, (v,r)$ to $\mathcal{X}^*$. In this setting, s can reach all the vertices $c_{v,l}, \ldots, c_{v,r}$ and $d_{v,l}, \ldots, d_{v,r}$ (excluding $c_{v,i}, d_{v,i}$ which do not exist). The forbidden pairs $\{c_{v,l}, d_{v,h}\}$ with $h \in [i+1, r]$ enforce the deletion of $(c_{v,h}, d_{v,h})$ for each such h. Similarly, the pairs $\{c_{v,r}, d_{v,h}\}$ with $h \in [l, i-1]$ enforce the deletion of the $(c_{v,h}, d_{v,h})$ arcs. This sums up to $r - l$ deletions, which corresponds to the span of $(v,l), \ldots, (v,r)$.

As another case, suppose that v reaches v_l^+, v_r^+ with $l < r < i$, as in Fig. 6. Then s reaches $b_{v,l}, \ldots, b_{v,r}$ as well as $c_{v,l}, \ldots, c_{v,r}$, and then in turn $d_{v,l}, \ldots, d_{v,r}$. Then the forbidden pairs $\{b_{v,r}, d_{v,h}\}$ for $h \in [l, r-1]$ enforce the deletion of $(c_{v,h}, d_{v,h})$ for each $h \in [l, r-1]$. Again, this requires $r - l$ deletions, which corresponds to the desired span.

The case $l = i$ ($r = i$, respectively) is similar to the previous ones, the deleted arcs are $(c_{v,h}, d_{v,h})$, with $h \in [i+1, r]$ ($h \in [l, i-1]$, respectively).

Note that Step 5 is the reason we added dummy timestamps 1 and T. If $(v,1)$ or (v,T) were allowed to be representative, we would need a different gadget for these cases.

We can now argue that our construction is correct.

Lemma 4. *Suppose that there exists a solution $\mathcal{X}^*$ of* Restricted Timeline Cover *that agrees with $\mathcal{X}$. Then there is a set of arcs $F \subseteq D$ with $|F| \leq k'$ such that s does not reach a forbidden pair in $H - F$, where $k' = k - sp(\mathcal{X})$.*

Proof. Let $\mathcal{X}^*$ be a solution of Restricted Timeline Cover that agrees with $\mathcal{X}$. By definition of Restricted Timeline Cover $\mathcal{X}^*$ has span at most k. Note that for $v \in A_{\mathcal{S}}$, agreement requires that $\mathcal{X}^*[v] = \mathcal{X}[v]$, and so the span of v in $\mathcal{X}^*$ is the same as the span of v in $\mathcal{X}$. Recalling that $\{Z_{\mathcal{S}}, \overline{A_{\mathcal{S}}}, A_{\mathcal{S}}\}$ form a partition of V, we get

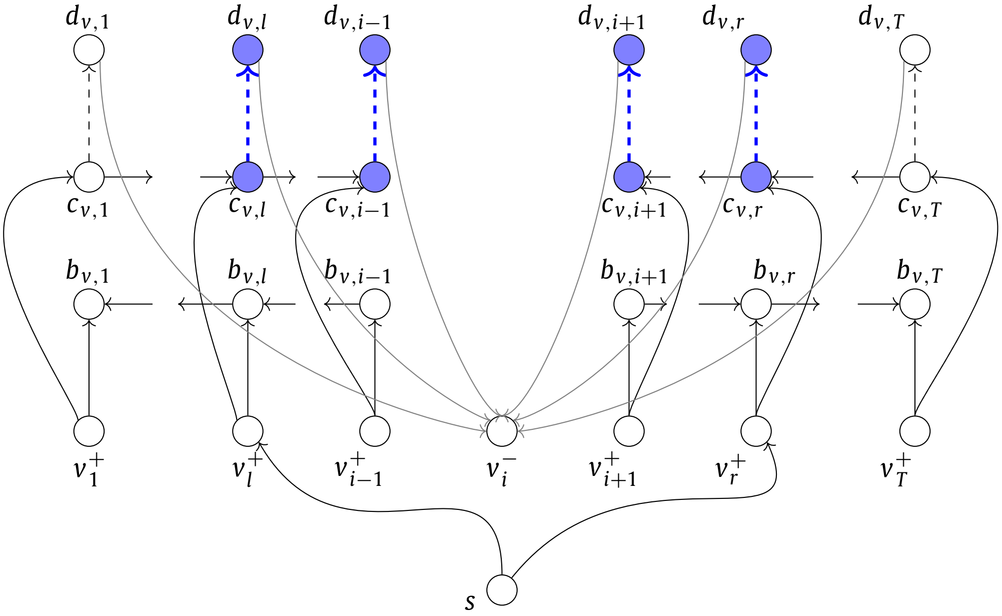

Fig. 5. An illustration of the case where s reaches v_l^+ and v_r^+, with $l<i<r$. Some of the vertices involved in forbidden pairs, in particular in forbidden pairs $\{c_{v,l},d_{v,i+1}\}$, $\{c_{v,l},d_{v,r}\}$, $\{c_{v,i-1},d_{v,i+1}\}$, $\{c_{v,i-1},d_{v,i+1}\}$, $\{c_{v,i+1},d_{v,l}\}$, $\{c_{v,i+1},d_{v,i-1}\}$, $\{c_{v,r},d_{v,l}\}$, $\{c_{v,r},d_{v,i-1}\}$, are highlighted; the thick dashed arcs are removed by a solution of Constrained Digraph Pair Cut, so that s cannot reach $d_{v,l}$, $d_{v,i-1}$, $d_{v,i+1}$ and $d_{v,r}$, thus s cannot reach both vertices in the pair $p\in P$ for each $p\in\{\{c_{v,l},d_{v,i+1}\},\{c_{v,r},d_{v,i-1}\}\}$.

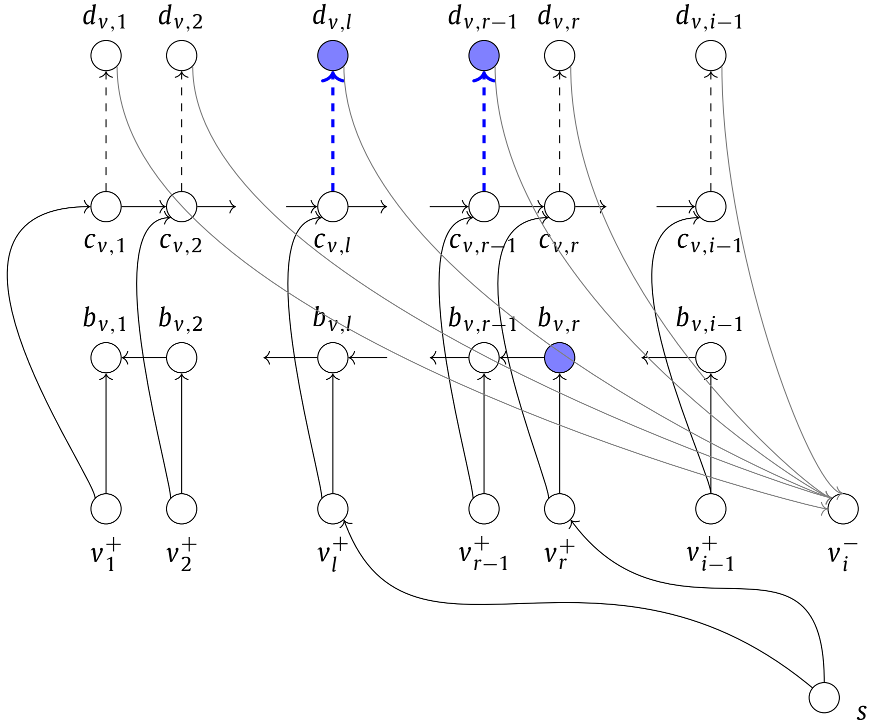

Fig. 6. An illustration of the case where s reaches v_l^+ and v_r^+, with $l<r<i$. Some of the forbidden pairs, in particular $\{b_{v,r},d_{v,l}\}$ and $\{b_{v,r},d_{v,r-1}\}$ are highlighted; the thick dashed arcs are removed by a solution of Constrained Digraph Pair Cut, so that s cannot reach $d_{v,l}$ and $d_{v,r-1}$.

$$\sum_{v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}} sp(v, \mathcal{X}^*) \leq k - sp(\mathcal{X}) = k'.$$

Moving on to the construction of F, we may assume that for every $v \in V$, at least one of $(v,2), \ldots, (v, T-1)$ is in $\mathcal{X}^*$, as otherwise we add one arbitrarily without affecting the span (if only $(v,1)$ or (v,T) is in $\mathcal{X}^*$, remove it first). For each $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$, consider the gadget corresponding to v in H and delete some of its dashed arcs as follows (we recommend referring to Fig. 4, Fig. 5 Fig. 6).

First, if only one of $\tau(v)$ is in $\mathcal{X}^*$, no action is required on the gadget. So assume that $\mathcal{X}^*[v]$ has at least two temporal vertices; in the following we denote by (v,l) and (v,r) the temporal vertices associated with v having minimum and maximum timestamp, respectively, contained in $\mathcal{X}^*$ (note, r stands for "right" and is not necessarily related to the representative $r(v)$). We assume that $l, r \in [2, T-1]$ and $l < r$. Note that

$$\mathcal{X}^*[v] = \{(v,l), (v, l+1), \ldots, (v,r)\}.$$

Let $r(v) = (v,i)$ be the representative of v, where $i \in [2, T-1]$. Then

- Suppose that $l, r \in [2, i-1]$, then: delete every arc $(c_{v,q}, d_{v,q})$, with $l \leq q \leq r-1$
- Suppose that $l, r \in [i+1, T-1]$, then: delete every arc $(c_{v,q}, d_{v,q})$, with $l+1 \leq q \leq r$
- Suppose that $l \in [2, i]$ and $r \in [i, T-1]$, then (recall $l < r$): delete every arc $(c_{v,q}, d_{v,q})$, with $l \leq q \leq i-1$ (assuming $l \leq i-1$), and delete every arc $(c_{v,q}, d_{v,q})$, with $i+1 \leq q \leq r$ (assuming $r \geq i+1$).

We see that by construction for all $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$, the number of arcs deleted in the gadget corresponding to v is equal to the number of temporal vertices in $\mathcal{X}^*[v]$ minus one, that is the span of v in $\mathcal{X}^*$. Since these vertices have span at most k', it follows that we deleted at most k' arcs from H. Denote by H' the graph obtained after deleting the aforementioned arcs. We argue that in H', s does not reach a forbidden pair. To this end, we claim the following.

Claim 1. *For $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$ and $t \in [T]$, if s reaches v_t^+ in H', then $(v,t) \in \mathcal{X}^*$, and if s reaches v_t^- in H', then $(v,t) \notin \mathcal{X}^*$.*

Proof. The proof is by induction on the distance between s and the vertex in H'. As a base case, consider the out-neighbors of s in H'. Suppose that v_t^+ is such that $(s, v_t^+) \in A(H')$. By inspecting the steps of the construction, we see that this arc was added to H by Step 3 because G contains a temporal edge $(u,t)(v,t)$ with $(u,t) \in (\tau(A_{\mathcal{S}}) \setminus \mathcal{X}) \cup (\tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}}))$ and $(v,t) \in \tau(Z_{\mathcal{S}})$. If $(u,t) \in \tau(A_{\mathcal{S}}) \setminus \mathcal{X}$, then $(u,t) \notin \mathcal{X}$ and $(u,t) \notin \mathcal{X}^*$ either, since it agrees with $\mathcal{X}$. In this case we must have $(v,t) \in \mathcal{X}^*$ to cover the edge, as desired. If instead $(u,t) \in \tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$, then $(v,t) \in \mathcal{X}^*$ holds by the second part of the definition of agreement. Again by inspecting the construction of H, we see that s does not have an out-neighbor of the form v_t^-, and so this suffices for the base case.

Now consider a vertex of the form v_t^+ or v_t^- at distance greater than 1 from s in H', and assume by induction that the claim holds for vertices of this form at a smaller distance. Suppose that this vertex is v_t^+. By inspecting every step of the construction, including Step 5, we see that the only possible in-neighbors of v_t^+ are either s, or some u_t^-. The s case was handled as a base case, and so we assume the latter. Thus any shortest path from s to v_t^+ ends with an arc (u_t^-, v_t^+). Since s reaches u_t^- with a shorter path, we know by induction that $(u,t) \notin \mathcal{X}^*$. Moreover, the arc (u_t^-, v_t^+) could only have been created on Step 4, and so $(u,t)(v,t) \in E(G)$. Therefore, $(v,t) \in \mathcal{X}^*$ must hold to cover $(u,t)(v,t)$, as desired.

So consider instead a vertex of the form v_t^- that s reaches in H'. Assume for contradiction that $(v,t) \in \mathcal{X}^*$. By inspecting the construction, we see that the only in-neighbors of v_t^- belong to the gadget corresponding to v. Moreover, the only way to reach v_t^- from s is to go through some v_j^+, where $j \in [T] \setminus \{t\}$, and then through some other vertices of the gadget. Consider a shortest path from s to v_t^- in H', and let v_j^+ be the first vertex of the gadget corresponding to v in this path. By induction, we may assume that $(v,j) \in \mathcal{X}^*$, and hence the span of v is at least one, since we are currently assuming that $(v,t) \in \mathcal{X}^*$.

Now, the existence of v_t^- implies that $(v,t) = r(v)$. Consider $\{(v,l), (v,l+1), \ldots, (v,r)\} = \mathcal{X}^*[v]$. Then we have $l \in [2,t]$, $r \in [t, T-1]$, and $j \in [l,r] \setminus \{t\}$. In this case, we have removed all the arcs $(c_{v,y}, d_{v,y})$, with $y \in [l,r] \setminus \{t\}$. Moreover, by inspecting the steps of the construction, we see that the only out-neighbors of v_j^+ are $b_{v,j}$ and $c_{v,j}$. Thus v_j^+ can only reach v_t^- through a $c_{v,y}$ and then a $d_{v,y}$ vertex. However, because of our deletions, v_j^+ cannot reach any vertex $d_{v,y}$, thus it cannot reach v_t^-, leading to a contradiction (one can check that this holds whether v_j^+ is to the left or to the right of v_t^-). We deduce that $(v,t) \notin \mathcal{X}^*$, which concludes the proof of the claim. □

Now, armed with the above claim, assume for contradiction that in H', s reaches both vertices of a forbidden pair $q \in P$. If q was created on Step 3, then $q = \{s, v_t^-\}$, where there is an edge $(u,t)(v,t) \in E(G)$ such that $(u,t) \in (\tau(A_{\mathcal{S}}) \setminus \mathcal{X}) \cup (\tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}}))$ and $(v,t) \in \tau(Z_{\mathcal{S}})$. Under the assumption that s reaches both elements of q, we get that s reaches v_t^-. But then, Claim 1 implies that $(v,t) \notin \mathcal{X}^*$. If $(u,t) \in \tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$, this contradicts the fact that $\mathcal{X}^*$ agrees with $\mathcal{X}$, since (v,t) should be in $\mathcal{X}^*$. If $(u,t) \in \tau(A_{\mathcal{S}}) \setminus \mathcal{X}$, then $(u,t) \notin \mathcal{X}$ implies that $(u,t) \notin \mathcal{X}^*$, reaching a contradiction since the edge $(u,t)(v,t)$ is not covered.

If q was created on Step 4, then $q = \{u_t^-, v_t^-\}$, where $(u,t)(v,t)$ is an edge of G. By Claim 1, this implies that $(u,t),(v,t) \notin \mathcal{X}^*$, a contradiction since $\mathcal{X}^*$ would not cover the edge.

We may thus assume that q was created on Step 5. Let $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$ be the vertex for which the corresponding gadget contains the two vertices of q. Let (v,i) be the representative of v, where $i \in [2, T-1]$. Suppose first that q is $\{d_{v,h}, b_{v,j}\}$ with $h < j \leq i-1$. Then for s to be able to reach both $d_{v,h}$ and $b_{v,j}$, the path from s to $d_{v,h}$ must pass through some v_l^+, $c_{v,l}$, $c_{v,h}$ (possibly $h = l$) and $d_{v,h}$, with $l \leq h$. On the other hand, since s reaches $b_{v,j}$, we have that s must reach v_r^+ and $b_{v,r}$, for some $r \in [j, i-1]$. By Claim 1, $(v,l),(v,r) \in \mathcal{X}^*$, in which case we deleted the deletable arc $(c_{v,h}, d_{v,h})$ (since $h < j$, and either the first or third situation arises in our list of three cases of arc deletions). Thus s cannot reach $d_{v,h}$.

Likewise, assume that q is $\{d_{v,h}, b_{v,j}\}$ with $i+1 \leq j < h$. Then s must reach $c_{v,r}$, $c_{v,h}$ (possibly identical to $c_{v,r}$) and $d_{v,h}$, with $r \geq h$. On the other hand, s must reach v_l^+, with $l \leq j < h$, since s reaches $b_{v,j}$. By Claim 1, $(v,l),(v,r) \in \mathcal{X}^*$, in which case we removed the deletable arc $(c_{v,h}, d_{v,h})$. Thus s cannot reach $d_{v,h}$.

Assume that q is $\{c_{v,h}, d_{v,j}\}$ or $\{c_{v,j}, d_{v,h}\}$ with $h < i < j$. Then s must reach v_l^+, $c_{v,l}$, $c_{v,h}$, possibly identical to $c_{v,l}$ and $d_{v,h}$, if $(c_{v,h}, d_{v,h})$ is not deleted. s must reach $c_{v,r}$, $c_{v,j}$, possibly identical to $c_{v,r}$, and $d_{v,j}$, if $(c_{v,i}, d_{v,j})$ is not deleted. By Claim 1, $(v,l),(v,r) \in \mathcal{X}^*$, in which case we deleted the deletable arcs $(c_{v,h}, d_{v,h})$ and $(c_{v,j}, d_{v,j})$. Thus s cannot reach $d_{v,h}$ and $d_{v,j}$.

Having handled every forbidden pair, we deduce that we can remove at most k' edges from H so that s does not reach any of them. □

Lemma 5. *Suppose that there is a set of arcs $F \subseteq D$ with $|F| \leq k' = k - sp(\mathcal{X})$ such that s does not reach a forbidden pair in $H - F$. Then there is a solution $\mathcal{X}^*$ to* Restricted Timeline Cover *that agrees with $\mathcal{X}$. Moreover, $\mathcal{X}^*$ can be computed in polynomial time from F.*

Proof. Suppose that there is a set $F \subseteq D$ with at most k' arcs such that s does not reach a forbidden pair in $H - F$. Denote $H' = H - F$. We construct $\mathcal{X}^*$ from F, which will also show that it can be reconstructed from F in polynomial time. Define $\mathcal{X}^*$ as follows:

- for each $v \in A_{\mathcal{S}}$, add every element of $\mathcal{X}[v]$ to $\mathcal{X}^*$;
- for each $(v,t) \in V(G) \setminus \tau(A_{\mathcal{S}})$, we add (v,t) to $\mathcal{X}^*$ if and only if one of the following holds: (1) $v_t^+ \in V(H)$ and s reaches v_t^+ in H'; or (2) $v_t^- \in V(H)$, and s does *not* reach v_t^- in H';
- for each $(v,j),(v,h) \in \mathcal{X}^*$ with $j < h$, add (v,t) to $\mathcal{X}^*$ for each $t \in [j+1, h-1]$.

We note that $\mathcal{X} \subseteq \mathcal{X}^*$. To see this, recall that by definition, $\mathcal{X}$ is an assignment of $V_{\mathcal{S}}$ and thus it only intersects with $\tau(V_{\mathcal{S}})$. Then by the definition of $\overline{A_{\mathcal{S}}}$ and $A_{\mathcal{S}}$, we have that $\mathcal{X}$ only intersects with $\tau(A_{\mathcal{S}})$. In the construction of $\mathcal{X}^*$, for every $v \in A_{\mathcal{S}}$ we add all of $\mathcal{X}[v] = \mathcal{X} \cap \tau(v)$ to $\mathcal{X}^*$. Thus it follows that we add all of $\mathcal{X}$ to $\mathcal{X}^*$.

We next argue that $\mathcal{X}^*$ agrees with $\mathcal{X}$. Indeed, for $v \in A_{\mathcal{S}}$, there is no gadget corresponding to v in the construction and thus we only add $\mathcal{X}[v]$ to $\mathcal{X}^*$. This satisfies the first requirement $\mathcal{X}^*[v] = \mathcal{X}[v]$ of agreement. For $u \in \overline{A_{\mathcal{S}}}$, consider $(u,t) \in \tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$ and a neighbor (v,t) of (u,t) in $\tau(Z_{\mathcal{S}})$. If $(v,t) \neq r(v)$, Step 3 adds an undeletable arc from s to v_t^+, hence s reaches that vertex and we put (v,t) in $\mathcal{X}^*$. If $(v,t) = r(v)$, Step 3 adds $\{s, v_t^-\}$ to P, and thus s does not reach v_t^- in H', and again we add (v,t) to $\mathcal{X}^*$. Therefore, we add all the $\tau(Z_{\mathcal{S}})$ neighbors of (u,t) to $\mathcal{X}^*$, and so it agrees with $\mathcal{X}$.

We next claim that $\mathcal{X}^*$ covers every temporal edge of G. Let $(u,t)(v,t) \in E(G)$. We list every possible case for the sets that contain (u,t) and (v,t):

1. $(u,t) \in \tau(A_{\mathcal{S}})$ and $(v,t) \in \tau(V_{\mathcal{S}})$. Then $(u,t)(v,t)$ is covered by $\mathcal{X}$ because $\mathcal{X}$ covers $G[V_{\mathcal{S}}]$. The edge is also covered by $\mathcal{X}^*$ since $\mathcal{X} \subseteq \mathcal{X}^*$.
2. $(u,t) \in \tau(A_{\mathcal{S}})$ and $(v,t) \in \tau(Z_{\mathcal{S}})$. If $(u,t) \in \mathcal{X}$, the edge is covered by $\mathcal{X} \subseteq \mathcal{X}^*$. If $(u,t) \notin \mathcal{X}$, then $(u,t) \in \tau(A_{\mathcal{S}}) \setminus \mathcal{X}$. If $(v,t) \neq r(v)$, Step 3 adds an undeletable arc from s to v_t^+, which implies that this arc is in H'. Thus s reaches v_t^+ and $(v,t) \in \mathcal{X}^*$ by construction, and $(u,t)(v,t)$ is covered. If $(v,t) = r(v)$, then $\{s, v_t^-\}$ is in P owing to Step 3, and thus s does not reach v_t^- in H'. Again by construction, $(v,t) \in \mathcal{X}^*$.
3. $(u,t) \in \tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$. If $(v,t) \in \tau(V_{\mathcal{S}})$, then the edge is in $G[V_{\mathcal{S}}]$ and $\mathcal{X} \subseteq \mathcal{X}^*$ must cover it. So assume $(v,t) \in \tau(Z_{\mathcal{S}})$. In that case, Step 3 is applied in the same manner as the previous case, that is, either (s, v_t^+) is an undeletable arc or $\{s, v_t^-\}$ is forbidden, both of which ensure that $(v,t) \in \mathcal{X}^*$.
4. $(u,t) \in \tau(Z_{\mathcal{S}}) \cup (\tau(\overline{A_{\mathcal{S}}}) \setminus \pi(I_{\mathcal{S}}))$. If $(v,t) \in \tau(A_{\mathcal{S}})$ or $(v,t) \in \tau(\overline{A_{\mathcal{S}}}) \cap \pi(I_{\mathcal{S}})$, then we can swap the role of (v,t) and (u,t) in the previous cases to argue that the edge is covered. Thus, we assume that $(v,t) \in \tau(Z_{\mathcal{S}}) \cup (\tau(\overline{A_{\mathcal{S}}}) \setminus \pi(I_{\mathcal{S}}))$ as well.
   As argued in Step 4, one of (u,t) or (v,t) must be a representative. To see this, note that if $(u,t),(v,t) \in \tau(Z_{\mathcal{S}})$, then this holds because representatives cover every edge of $G[Z_{\mathcal{S}}]$. If $(u,t) \notin \tau(Z_{\mathcal{S}})$, then $r(v) = (v,t)$ by Observation 1. A symetric argument applies if $(v,t) \notin \tau(Z_{\mathcal{S}})$ instead.
   So assume without loss of generality that $(u,t) = r(u)$. If $(v,t) = r(v)$, Step 4 adds $\{u_t^-, v_t^-\}$ to P. Thus s does not reach one of the two, implying that at least one of (u,t) or (v,t) is in $\mathcal{X}^*$ to cover the edge. If $(v,t) \neq r(v)$, then (u_t^-, v_t^+) is an undeletable arc of H. If s does not reach u_t^-, we add (u,t) to $\mathcal{X}^*$ and we cover the edge. Otherwise, s reaches u_t^- and in turn v_t^+, in which case we add (v,t) to $\mathcal{X}^*$.

Because $A_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}} \cup Z_{\mathcal{S}}$ is a partition of V, the above cases handle every possible situation, and thus $\mathcal{X}^*$ covers every edge.

We next claim that $sp(X^*) \leq k$. Since $\mathcal{X}^*[A_{\mathcal{S}}] = \mathcal{X}$, the vertices in $A_{\mathcal{S}}$ have span equal to $sp(\mathcal{X})$. We must argue that the vertices of $V \setminus A_{\mathcal{S}} = Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$ have a span of at most

$$k' = k - sp(\mathcal{X}).$$

Consider a vertex $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$ that has span $sp(v, \mathcal{X}^*)$ more than 0 in $\mathcal{X}^*$. We want to show that $sp(v, \mathcal{X}^*)$ edges of H were deleted in the gadget corresponding to v.

In the following we denote by (v, l) and (v, r), with $l, r \in [2, T-1]$ the temporal vertices of minimum and maximum timestamp, respectively, such that $(v, l) \in \mathcal{X}^*$ and $(v, r) \in \mathcal{X}^*$.

Let $(v, i) = r(v)$, with $v \in Z_{\mathcal{S}} \cup \overline{A_{\mathcal{S}}}$. Suppose first that $r < i$. Then by the construction of $\mathcal{X}^*$, s reaches v_l^+ and v_r^+, hence s reaches:

1. $c_{v,l}$ and thus $c_{v,j}$, for each $j \in [l, i-1]$
2. $b_{v,r}$ and thus $b_{v,j}$, for each $j \in [1, r]$.

Thus the arcs $(c_{v,j}, d_{v,j})$, with $j \in [l, r-1]$, have to be deleted due the forbidden pairs $\{d_{v,j}, b_{v,y}\}$, with $l \leq j < y \leq r$. This amounts to $r - l$ deletions, which is the span of v in $\mathcal{X}^*$.

Suppose instead that with $i < l$. Similarly to the previous case, by the construction, s reaches v_l^+ and v_r^+, hence s reaches:

(1) $c_{v,r}$ and thus $c_{v,j}$, for each $j \in [i+1, r]$
(2) $b_{v,l}$ and thus $b_{v,j}$, for each j with $j \in [l, T]$.

Thus the arcs $(c_{v,j}, d_{v,j})$, for each $j \in [l+1, r]$, have to be deleted due the forbidden pairs $\{d_{v,j}, b_{v,y}\}$, with $l \leq y < j \leq r$. Again, this amounts to $r - l$ deletions, which is the span of v in $\mathcal{X}^*$.

Finally, suppose that $l \leq i \leq r$. We have three cases depending on the fact that $l = i$, $r = i$ or $l < i < r$. Consider the first case $l = i < r$. Thus by the construction of $\mathcal{X}^*$, s reaches v_r^+ but does not reach v_i^-. Moreover, s reaches $c_{v,r}$ thus s reaches $c_{v,j}$, for each $j \in [i+1, r]$. Thus arcs $(c_{v,j}, d_{v,j})$, for each $j \in [i+1, r]$, have to be deleted in order to make v_i^- not reachable from s. This amounts to $r - i = r - l$ deletions, which is the span of v in $\mathcal{X}^*$.

Consider the second case $l < i = r$. Similarly to the previous case, s reaches v_l^+ but does not reach v_i^-. Moreover, s reaches $c_{v,j}$, for each j with $j \in [l, i-1]$. Thus arcs $(c_{v,j}, d_{v,j})$, for each $j \in [l, i-1]$ have to be deleted in order to make v_i^- not reachable from s. This amounts to $i - j = r - l$ deletions, which is the span of v in $\mathcal{X}^*$.

Finally, consider the third case $l < i < r$. Then arcs $(c_{v,j}, d_{v,j})$, with $j \in [l, i-1]$ and $(c_{v,j}, d_{v,j})$, with $j \in [i+1, r]$, have to be deleted due to forbidden pairs $\{c_{v,j}, d_{v,z}\}$, with $j < i < z$ and forbidden pairs $\{c_{v,z}, d_{v,j}\}$, with $z < i < j$. This requires $i - l + r - i = r - l$ deletions, which is the span of v in $\mathcal{X}^*$.

We thus see that each vertex v of $V \setminus A_{\mathcal{S}}$ has a span that is at most the number of arcs deleted in the gadget of H corresponding to v. Therefore, $\mathcal{X}^*$ is a temporal cover of span at most $sp(\mathcal{X}) + k' \leq k$, thus completing the proof. □

Wrapping up

Before concluding, we must show that we are able to use the results of [19] to get an FPT algorithm for Constrained Digraph Pair Cut, as we have presented it. As we mentioned, the FPT algorithm in [19] studied the vertex-deletion variant and does not consider undeletable elements, but this is mostly a technicality, as we show in the following lemma.

Lemma 6. *The* Constrained Digraph Pair Cut *problem can be solved in time $O^*(2^k)$, where k is the number of arcs to delete.*

Proof. We call Vertex-Deletion Digraph Pair Cut the problem in which, given a directed graph H, a source $s \in V(H)$, pairs $P \subseteq \binom{V}{2}$, and integer k, we must decide whether there is $R \subseteq V(H) \setminus \{s\}$ with $|R| \leq k$ such that in $H - R$, s does not reach both u and v for every $\{u, v\} \in P$ (note that $H - R$ removes vertices here, not arcs). In [19, Theorem 6.1], this problem was shown to be solvable in time $O^*(2^k)$. We show that Constrained Digraph Pair Cut as we defined it reduces to Vertex-Deletion Digraph Pair Cut, with the same parameter value k.

Suppose that we have an instance of Constrained Digraph Pair Cut, with directed graph H, source s, pairs P, deletable arcs D, and integer k. From this instance, obtain an instance of Vertex-Deletion Digraph Pair Cut with directed graph H', source s', pairs P', and integer k as follows (note that k is unchanged). First for each $u \in V(H)$, add $k+1$ copies $u^1, \ldots, u^{k+1}$ of u to $V(H')$. Note that this also applies to s, which has corresponding copies $s^1, \ldots, s^{k+1}$ in H'. Also add a new vertex s' to $V(H')$, which serves as the source for the modified instance. Add to $A(H')$ the set of arcs $(s', s^1), \ldots, (s', s^{k+1})$. Then for each deletable arc $(u, v) \in D$, add to H' a new vertex $q_{u,v,1}$, and the set of arcs

$$\{(u^i, q_{u,v,1}) : i \in [k+1]\} \cup \{(q_{u,v,1}, v^j : j \in [k+1]\}$$

Finally, for each undeletable arc $(u,v) \in A(H) \setminus D$, add to H' the $k+1$ new vertices $q_{u,v,1}, q_{u,v,2}, \dots, q_{u,v,k+1}$, and then add to $A(H')$ the set of arcs

$$\{(u^i, q_{u,v,l}) : i \in [k+1], l \in [k+1]\} \cup \{(q_{u,v,l}, v^j) : l \in [k+1], j \in [k+1]\}$$

In other words, each vertex of H has $k+1$ corresponding vertices in H', making the latter pointless to delete. For $(u,v) \in D$, deleting the arc corresponds to deleting $q_{u,v,1}$ since it removes the path of length 2 from every u^i to every v^j. For $(u,v) \in A(H) \setminus D$, there are too many $q_{u,v,l}$ copies, making them pointless to delete.

Finally, for each $\{u,v\} \in P$, we add to P' all the pairs $\{u^i, v^j\}$ for every $i, j \in [k+1]$.

Assume that there is $F \subseteq D$ with $|F| \leq k$ such that s reaches no pair of P in $H - F$. In H', we delete the set of vertices $R = \{q_{u,v,1} : (u,v) \in F\}$. Note that $|R| = |F| \leq k$. Moreover, because every arc in $(u,v) \in F$ is deletable, there are no $q_{u,v,l}$ vertices for $l \geq 2$, and thus the paths of length 2 from the u^i's to the v^j's are effectively removed in $H' - R$. This means that, for vertices x^i, y^j of $H' - R$, if there is still a path of length 2 from x^i to y^j, then the arc (x,y) is present in $H - F$.

Suppose for contradiction that in $H' - R$, s' reaches both $\{u^i, v^j\} \in P'$. Because aside from the arcs incident to s', every arc of H' is incident to a vertex of the form $q_{u,v,l}$, in H', the path from s' to u^i in $H' - R$ has the form

$$s' \to s^{b_0} \to q_{s,x_1,a_1} \to x_1^{b_1} \to q_{x_1,x_2,a_2} \to x_2^{b_2} \to q_{x_2,x_3,a_3} \to \dots \to q_{x_l,u,a_{l+1}} \to u^i$$

for some vertices $x_1, \dots, x_l \in V(H)$ and indices $b_0, a_1, b_1, \dots, a_{l+1}$. By our argument on the paths of length 2, this means that in $H - F$, all the arcs $(s, x_1), (x_1, x_2), \dots, (x_l, u)$ are present, and that s reaches u in $H - F$. By the same logic, s also reaches v in $H - F$, a contradiction since $\{u^i, v^j\} \in P'$ implies that $\{u,v\} \in P$. Thus R is a solution for H'.

Conversely, assume that there is $R \subseteq V(H') \setminus \{s'\}$ with $|R| \leq k$ such that s reaches no pair of P' in $H' - R$. We may assume that R does not contain a vertex u^i with $u \in V(H)$, since R cannot contain every copy of u. Likewise, for undeletable $(u,v) \in A(H) \setminus D$, we may assume that R does not contain a vertex $q_{u,v,l}$ since R cannot contain every copy. Therefore, we may assume that R only contains vertices of the form $q_{u,v,1}$, where $(u,v) \in D$. Define $F = \{(u,v) : q_{u,v,1} \in R\}$. Note that F has at most $|R| \leq k$ arcs, and they are all deletable. Now suppose for contradiction that s reaches both u, v for $\{u,v\} \in P$ in $H - F$. Hence in $H - F$ there are paths P_u, P_v from s to u and v, respectively. Note that for every arc (x,y) of P_u, the vertex $q_{x,y,1}$ is still present in $H' - R$. Thus by replacing every (x,y) with the subpath $x^1, q_{x,y,1}, y^1$, we can obtain a path from s' to u^1 in $H' - R$. Likewise, there is a path from s' to v^1 in $H' - R$. This is a contradiction since $\{u,v\} \in P$ implies that $\{u^i, v^j\} \in P'$. Hence F is a valid solution for H.

To conclude, constructing H' can clearly be done in time polynomial in $|V(H)| + |A(H)|$. It follows that we can solve the instance H in time $O^*(2^k)$ by constructing H' and solving the vertex-deletion variant on it. □

We are able now to prove the main result of our contribution.

Theorem 2. *k-TimelineCover on a temporal graph $G = (V, E, \mathcal{T})$ can be solved in time $O^*(2^{5k \log k})$.*

Proof. First, we discuss the correctness of the algorithm we presented. Assume that we have an ordering on the vertices of G and that v is the first vertex of this ordering. A solution $\mathcal{S}^*$ of k-TimelineCover on $G[\{v\}]$ is equal to $\mathcal{S}^* = \emptyset$.

Then for i, with $i \in [2, |V|]$, let G_i be the temporal graph induced by the first i vertices and let w be the $i+1$-th vertex. Given a solution $\mathcal{S}^*$ of k-TimelineCover on instance (G_i, k) of span at most k, we can decide whether there exists a solution of k-TimelineCover on instance (G_{i+1}, k) by computing whether there exists a solution $\mathcal{X}^*$ of the Restricted Timeline Cover problem on instance G_i, w, k, $\mathcal{S}^*$. By Lemma 3 and by Theorem 1 if there exists such an $\mathcal{X}^*$, then there exists a feasible assignment $\mathcal{X}$ such that $\mathcal{X}^*$ agrees with $\mathcal{X}$. Because we enumerate all feasible assignments, we will eventually consider such an $\mathcal{X}$. By Lemmas 4 and 5, via the reduction to Constrained Digraph Pair Cut, we will correctly find a positive answer to the existence of a solution $\mathcal{X}^*$ for Restricted Timeline Cover on instance G_i, w, k and $\mathcal{S}^*$. Conversely, suppose that no solution exists for that instance. Then for any feasible assignment $\mathcal{X}$, there cannot be a temporal cover $\mathcal{X}^*$ of span at most k that agrees with $\mathcal{X}$ (because we assume that no such solution exists). Then by Lemmas 4 and 5, our reduction to Constrained Digraph Pair Cut will correctly return a negative answer on every feasible assignment $\mathcal{X}$. Thus the Restricted Timeline Cover subproblems are solved correctly, and once it is solved on $G_{|V|}$, we have a solution to k-TimelineCover.

Now, we discuss the complexity of the algorithm. We must solve Restricted Timeline Cover $|V|$ times. For each iteration, by Theorem 1 we can enumerate the feasible assignments in $O(2^{4k \log k} T^3 n)$ time. For each such assignment, the reduction from Restricted Timeline Cover to Constrained Digraph Pair Cut requires polynomial time, and each generated instance can be solved in time $O^*(2^k)$. The time dependency on k is thus $O^*(2^{4k \log k} \cdot 2^k)$, which we simplify to $O^*(2^{5k \log k})$. □

4. Conclusion

We have presented an FPT algorithm for the k-TimelineCover problem, a variant of Vertex Cover on temporal graphs recently considered for timeline activities summarizations. We point out some relevant future directions on this topic: (1) to improve, if possible, the time complexity of k-TimelineCover by obtaining a single exponential time algorithm (of the form $O^*(c^k)$); (2) to establish whether k-TimelineCover admits a polynomial kernel, possibly randomized (which it might, since Digraph Pair Cut famously admits a randomized polynomial kernel); and (3) to study other problems related to covering temporal graphs based on the definition of assignment.

CRediT authorship contribution statement

Riccardo Dondi: Writing – review & editing, Writing – original draft, Project administration, Methodology, Investigation, Formal analysis, Conceptualization. **Manuel Lafond:** Writing – review & editing, Writing – original draft, Project administration, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Ethical approval

This article does not contain any studies with human participants or animals performed by any of the authors.

Funding

ML acknowledges financial support from the Natural Sciences and Engineering Research Council of Canada (grant no RGPIN-2019-05817).

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Manuel Lafond reports financial support was provided by Natural Sciences and Engineering Research Council of Canada. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

We would like to thank the reviewers for their useful comments that helped us improve the presentation of the paper.

Data availability

No data was used for the research described in the article.

References

[1] R. Dondi, M. Lafond, An FPT algorithm for temporal graph untangling, in: 18th International Symposium on Parameterized and Exact Computation, IPEC 2023, September 6-8, 2023, Amsterdam, the Netherlands, in: LIPIcs, vol. 285, pp. 12:1–12:16, https://doi.org/10.4230/LIPICS.IPEC.2023.12.

[2] D. Kempe, J.M. Kleinberg, A. Kumar, Connectivity and inference problems for temporal networks, J. Comput. Syst. Sci. 64 (4) (2002) 820–842, https://doi.org/10.1006/jcss.2002.1829.

[3] P. Holme, Modern temporal network theory: a colloquium, Eur. Phys. J. B 88 (9) (2015) 234.

[4] H. Wu, J. Cheng, S. Huang, Y. Ke, Y. Lu, Y. Xu, Path problems in temporal graphs, Proc. VLDB Endow. 7 (9) (2014) 721–732, https://doi.org/10.14778/2732939.2732945.

[5] H. Wu, J. Cheng, Y. Ke, S. Huang, Y. Huang, H. Wu, Efficient algorithms for temporal path computation, IEEE Trans. Knowl. Data Eng. 28 (11) (2016) 2927–2942, https://doi.org/10.1109/TKDE.2016.2594065.

[6] T. Erlebach, M. Hoffmann, F. Kammer, On temporal graph exploration, J. Comput. Syst. Sci. 119 (2021) 1–18, https://doi.org/10.1016/j.jcss.2021.01.005.

[7] P. Zschoche, T. Fluschnik, H. Molter, R. Niedermeier, The complexity of finding small separators in temporal graphs, J. Comput. Syst. Sci. 107 (2020) 72–92, https://doi.org/10.1016/j.jcss.2019.07.006.

[8] T. Fluschnik, H. Molter, R. Niedermeier, M. Renken, P. Zschoche, Temporal graph classes: a view through temporal separators, Theor. Comput. Sci. 806 (2020) 197–218, https://doi.org/10.1016/j.tcs.2019.03.031.

[9] B.M. Bumpus, K. Meeks, Edge exploration of temporal graphs, in: Combinatorial Algorithms - 32nd International Workshop, IWOCA 2021, Ottawa, ON, Canada, July 5-7, 2021, Proceedings, 2021, pp. 107–121.

[10] A. Marino, A. Silva, Königsberg sightseeing: Eulerian walks in temporal graphs, in: Combinatorial Algorithms - 32nd International Workshop, IWOCA 2021, Ottawa, ON, Canada, July 5-7, 2021, Proceedings, 2021, pp. 485–500.

[11] E.C. Akrida, G.B. Mertzios, P.G. Spirakis, C.L. Raptopoulos, The temporal explorer who returns to the base, J. Comput. Syst. Sci. 120 (2021) 179–193, https://doi.org/10.1016/j.jcss.2021.04.001.

[12] R. Dondi, M.M. Hosseinzadeh, Colorful path detection in vertex-colored temporal, Netw. Sci. 11 (4) (2023) 615–631, https://doi.org/10.1017/NWS.2023.17.

[13] E.C. Akrida, G.B. Mertzios, P.G. Spirakis, V. Zamaraev, Temporal vertex cover with a sliding time window, J. Comput. Syst. Sci. 107 (2020) 108–123, https://doi.org/10.1016/j.jcss.2019.08.002.

[14] T. Hamm, N. Klobas, G.B. Mertzios, P.G. Spirakis, The complexity of temporal vertex cover in small-degree graphs, in: Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, the Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022, 2022, pp. 10193–10201.

[15] P. Rozenshtein, N. Tatti, A. Gionis, The network-untangling problem: from interactions to activity timelines, Data Min. Knowl. Discov. 35 (1) (2021) 213–247, https://doi.org/10.1007/s10618-020-00717-5.

[16] V. Froese, P. Kunz, P. Zschoche, Disentangling the computational complexity of network untangling, Theory Comput. Syst. 68 (1) (2024) 103–121, https://doi.org/10.1007/S00224-023-10150-Y.

[17] R. Dondi, Untangling temporal graphs of bounded degree, Theor. Comput. Sci. 969 (2023) 114040, https://doi.org/10.1016/j.tcs.2023.114040.

[18] R. Dondi, A. Popa, Timeline cover in temporal graphs: exact and approximation algorithms, in: Combinatorial Algorithms - 34th International Workshop, IWOCA 2023, Tainan, Taiwan, June 7-10, 2023, Proceedings, 2023, pp. 173–184.

[19] S. Kratsch, M. Wahlström, Representative sets and irrelevant vertices: new tools for kernelization, J. ACM 67 (3) (2020) 16–11650, https://doi.org/10.1145/3390887.

[20] E.J. Kim, S. Kratsch, M. Pilipczuk, M. Wahlström, Directed flow-augmentation, in: Proceedings of the 54th Annual ACM SIGACT Symposium on Theory of Computing, 2022, pp. 938–947.
[21] E.J. Kim, S. Kratsch, M. Pilipczuk, M. Wahlström, Flow-augmentation iii: complexity dichotomy for Boolean csps parameterized by the number of unsatisfied constraints, in: Proceedings of the 2023 Annual ACM-SIAM Symposium on Discrete Algorithms (SODA), SIAM, 2023, pp. 3218–3228.
[22] E.J. Kim, S. Kratsch, M. Pilipczuk, M. Wahlström, Directed flow-augmentation, in: STOC '22: 54th Annual ACM SIGACT Symposium on Theory of Computing, Rome, Italy, June 20 - 24, 2022, 2022, pp. 938–947.