Insights into the Complexity of Disentangling Temporal Graphs

Riccardo Dondi[1]

[1]Università degli Studi di Bergamo, Italy

Abstract

We consider a variant of vertex cover in temporal graphs recently introduced to summarize timeline activities in social networks. Since the problem is already NP-hard when the time domain considered consists of two timestamps, we analyse the complexity of the problem in other restricted cases. We prove that the problem is NP-hard when (1) there exists exactly one interaction in each timestamp and (2) when each vertex has at most two interactions in each timestamp and the time domain consists of three timestamps. Moreover, we prove that the problem is fixed parameter tractable when the parameter is the size of the time window activity, which bounds the number of vertices with active temporal edges in a timestamp and the length of the interval where a vertex has incident temporal edges.

Keywords

Temporal Graphs, Graph Algorithms, Computational Complexity, Vertex Cover

1. Introduction

The analysis of networks is recently considering representations that extend the classic graph model. In this context, *temporal graphs* describe dynamics of edge activity in a discrete time domain [1, 2]. Many works on temporal graphs have been focused on finding paths and analyzing their connectivity [1, 3, 4, 5, 6, 7, 8, 9] and to find cohesive subgraphs [10, 11]. Recently, one of the most prominent problem in graph theory and theoretical computer science, Vertex Cover, has been considered also for temporal graphs [12, 13].

In this paper we consider a variant of Vertex Cover, called Network Untangling, proposed in [13] and motivated by discovering event timelines and summarizing temporal networks. Given sequences of interactions, the problem looks for an explanation of the observed interactions with few (and short) *activity intervals*, such that each interaction is covered by at least one of the two entities involved (at least one of the two entities is active when the interaction is observed). This can be seen as a variant of Vertex Cover, where we have to cover temporal edges with the minimum activity of vertices, called span. The *span* of a vertex is the difference between the starting and ending timestamps of the interval where it is defined to be active.

Four formulations of the problem have been considered in [13], depending on the fact that each vertex/entity can be active in one or $k \geq 2$ intervals and that the goal is to minimize the sum of spans of vertex activities or the maximum activity span of a vertex. Here we focus on the

Proceedings of the 23rd Italian Conference on Theoretical Computer Science, Rome, Italy, September 7-9, 2022
riccardo.dondi@unibg.it (R. Dondi)

formulation where each vertex can be active in exactly one interval and the objective function is the minimization of the sum of the span of vertex activities, a problem called MinTimelineCover.

Other two variants of Vertex Cover in temporal graphs have been considered in [12]. A first variant asks for the minimum number of vertex activities such that each non-temporal edge $e = \{u, v\}$ is temporally covered, that is there exists a timestamp t where e is active and one of (u, t) and (v, t) belongs to the cover. A second variant asks each temporal edge to be temporally covered at least once for every interval of a given length Δ. The two variants are shown NP-hard, also in very restricted cases [12]. Further results on the problem variants, including their approximability, have been given in [12, 14].

The MinTimelineCover problem is known to be NP-hard also in very restricted cases, in particular when the time domain consists of two timestamps [15] (notice that when the time domain consists of a single timestamp, the problem is trivially in P, since any solution of the problem has span 0). MinTimelineCover is fixed-parameter tractable, when parameterized by the span of the solution and the time domain consists of exactly two timestamps [15]. Furthermore, in [15] the parameterized complexity of the variants of Network Untangling proposed in [13] has been explored, considering as parameters the number of vertices of the temporal graph, the length of the time domain, the number of intervals of vertex activity and the span of a solution.

In this paper, we further analyze the complexity of the MinTimelineCover problem by considering restrictions of the temporal input graph. We prove in Section 3 that the problem is NP-hard even when there exists at most one interaction in each timestamp. In Section 4 we consider a bound on the degree of the input temporal graph and on the length of the time domain and we show that MinTimelineCover NP-hard when each vertex has at most two interactions in each timestamp and the time domain consists of three timestamps. Finally, in Section 5 we prove that the problem is fixed parameter tractable when the parameter is the size of the time window activity that bounds (1) the number of vertices with temporal edges in a timestamp and (2) the interval length where a vertex has incident temporal edges. We conclude the paper with some open problems in Section 6. In Section 2 we present some definitions and we formally define the MinTimelineCover problem. Some of the proofs are omitted due to space constraints.

2. Preliminaries

We start this section by introducing the definition of discrete time domain over which is defined a temporal graph.

Definition 1. *A discrete time domain $\mathcal{T}$ is a sequence of timestamp t_i, $1 \leq i \leq |\mathcal{T}|$, where each t_i is an integer and $t_i < t_{i+1}$. An interval $T = [t_i, t_j]$ over $\mathcal{T}$, where $t_i, t_j \in \mathcal{T}$ and $t_i \leq t_j$, is the sequence of timestamps with value between t_i and t_j.*

Two intervals $T_1 = [t_{a,1}, t_{b,1}]$, $T_2 = [t_{a,2}, t_{b,2}]$ are disjoint if they do not share any timestamp, that is $t_{a,1} \leq t_{b,1} < t_{a,2} \leq t_{b,2}$ or $t_{a,2} \leq t_{b,2} < t_{a,1} \leq t_{b,1}$. The concatenation of the disjoint intervals T_1 and T_2 is an interval $T_1 \cdot T_2$ obtained by merging the two time intervals T_1 and T_2, that is, assuming without loss of generality that $t_{a,1} \leq t_{b,1} < t_{a,2} \leq t_{b,2}$, it is defined as follows:

$$T_1 \cdot T_2 = [t_{a,1}, t_{b,2}].$$

Given a set of pairwise disjoint intervals $T_1 = [t_{a,1}, t_{b,1}], T_2 = [t_{a,2}, t_{b,2}], \dots, T_q = [t_{a,q}, t_{b,q}]$, where $t_{a,1} \leq t_{b,1} < t_{a,2} \leq t_{b,2} < \dots < t_{a,q} \leq t_{b,q}$, we can define the concatenation of these intervals:

$$T_1 \cdot T_2 \cdot \dots T_q = [t_{a,1}, t_{q,2}].$$

We present now the definition of temporal graph. We consider a model where the vertex set is not changing in the time domain.

Definition 2. *A temporal graph $G_t = (V_t, E_t, \mathcal{T})$ consists of*

1. *A set V_t of vertices*
2. *A time domain $\mathcal{T}$*
3. *A set $E_t \subseteq V_t \times V_t \times \mathcal{T}$ of temporal edges, where a temporal edge of G_t is a triple $\{u, v, t\}$, with $u, v \in V_t$ and $t \in \mathcal{T}$.*

Given an interval I of $\mathcal{T}$, $E_t(I)$ denotes the set of active edges in the timestamps of I, that is:

$$E_t(I) = \{\{u, v, t\} | \{u, v, t\} \in E_t \wedge t \in I\}$$

$E_t(s)$ denotes the set of active edges in timestamp s.

Given a vertex $v \in V_t$, an activity interval of v is defined as an interval $I_v = [l_v, r_v]$ of the time domain where v is considered active, while in any timestamp not in I_v, v is considered inactive. Informally, I_v defines the time interval where v is considered active. Notice that if $I_v = [l_v, r_v]$ is an activity interval of v, there may exist temporal edges $\{u, v, t\}$, with $t < l_v$ or $t > r_v$ (see the example in Fig. 1). An activity timeline $\mathcal{A}$ is a set of activity intervals, defined as:

$$\mathcal{A} = \{I_v : v \in V_t\}$$

Given a temporal graph $G_t = (V_t, E_t, \mathcal{T})$, a timeline $\mathcal{A}$ covers $G_t = (V_t, E_t, \mathcal{T})$ if for each temporal edge $\{u, v, t\} \in E_t$, t belongs to I_u or to I_v, where I_u (I_v, respectively) is the activity interval of u (of v, respectively) defined by $\mathcal{A}$.

The span of an interval $I_v = [l_v, r_v]$, for some $v \in V$, is defined as follows:

$$s(I_v) = |r_v - l_v|.$$

Notice that for an interval $I_v = [l_v, r_v]$ consisting of a single timestamp, that is where $l_v = r_v$, it holds that $s(I_v) = 0$. The overall span of an activity timeline $\mathcal{A}$ is equal to

$$s(\mathcal{A}) = \sum_{I_v \in \mathcal{A}} s(I_v).$$

Now, we are ready to define the problem we are interested into (see the example of Fig. 1).

Problem 1. *(*MinTimelineCover*)*
Input: *A temporal graph $G_t = (V_t, \mathcal{T}, E_t)$.*
Output: *An activity timeline of minimum span that covers G_t.*

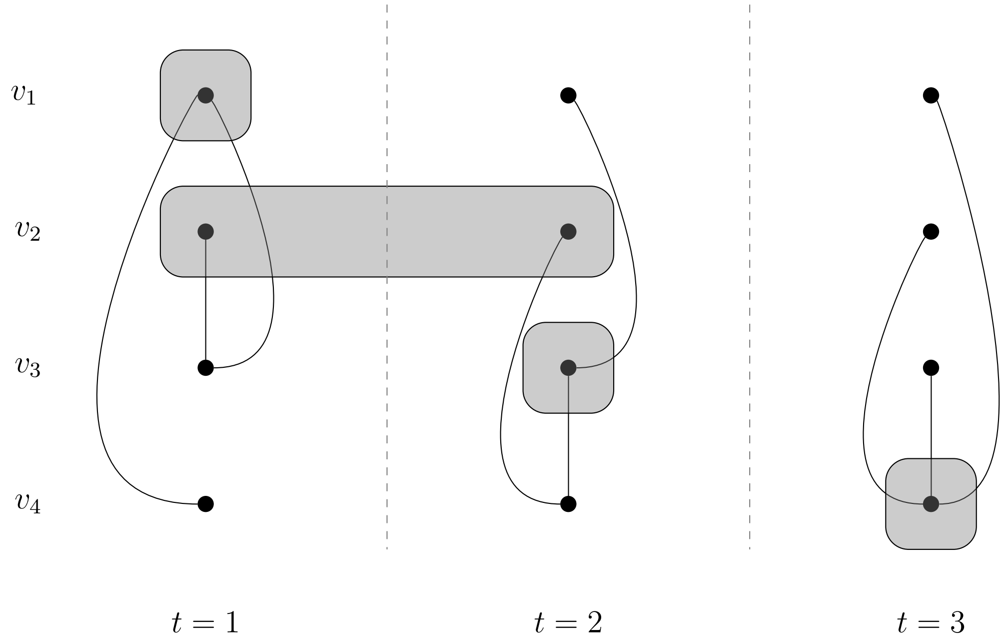

Figure 1: An example of MinTimelineCover on a temporal graph G_t consisting of four vertices (v_1, v_2, v_3, v_4) and three timestamps $(1, 2, 3)$; for each timestamp, we present the active temporal edges of G_t. For example for $t = 1$, the active edges are $\{v_1, v_3, 1\}$, $\{v_1, v_4, 1\}$, $\{v_2, v_3, 1\}$. The intervals in gray represent the activity intervals of each vertex; for example vertex v_1 is active in timestamp 1, with span equal to 0, vertex v_2 is active in interval $[1, 2]$ and it has a span equal to 1. Notice that the activity timeline defined by the vertex activity intervals covers the temporal graph with total span equal to 1.

Given a temporal graph $G_t = (V_t, E_t, \mathcal{T})$ and a vertex $v \in V_t$, the local degree of v in a timestamp t, denoted by $\Delta_L(v, t)$, is the number of temporal edges $\{v, u, t\}$, with $u \in V_t$. The local degree Δ_L of G_t is the maximum over v and t of $\Delta_L(v, t)$.

Given an interval I of the time domain $\mathcal{T}$, the time window associated with I (denoted by $w(I)$) is defined as

$$w(I) = \{(v, t) : \{v, u, t\} \in E_t \wedge t \text{ in } I\}$$

that is the set of pairs consisting of vertices of V_t and timestamps of I, such that there exists a temporal edge incident in v in some timestamp of I. We define a temporal graph to be (w, h)-window-constrained if (1) the temporal edges incident in each vertex belong to an interval of length at most w and (2) at most h vertices are active in each timestamp.

3. Hardness of MinTimelineCover for Bounded Activity

In this section we consider the MinTimelineCover problem when each timestamp contains at most a single active edge, and thus the local degree Δ_L of G_T is also bounded by 1. We denote this restriction by 1-MinTimelineCover. We prove that 1-MinTimelineCover is NP-hard by giving a reduction from the Vertex Cover problem. First, we recall the definition of Vertex Cover:

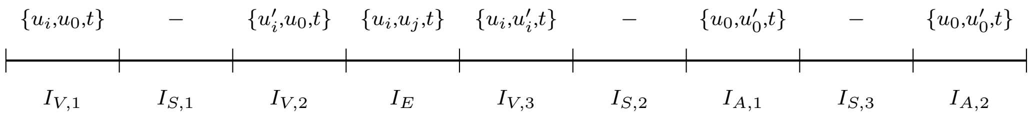

Figure 2: A sketch of the time domain $\mathcal{T}$ built by the reduction. For each of the nine disjoint intervals introduced to define $\mathcal{T}$, we report in the upper part a temporal edge defined in that interval; a symbol – is used when no temporal edge is defined in the interval.

Problem 2. *(*Vertex Cover*)*
Input: *A graph $G = (V, E)$.*
Output: *A minimum cardinality vertex cover V' of G , that is for each $\{u, v\} \in E$, $u \in V'$ or $v \in V'$.*

Given an instance $G = (V, E)$ of Vertex Cover, where $|V| = n$ and $|E| = m$, we build a corresponding temporal graph $G_t = (V_t, E_t, \mathcal{T})$, instance of 1-MinTimelineCover, as follows (a sketch of the temporal graph is given in Fig. 2). First, we define the time domain $\mathcal{T}$, which consists of $2m^4 + m^2 + m + n + 3$ timestamps and is defined starting from the following disjoint intervals:

$$I_{V,1} = [1, n] \ \ (n \text{ timestamps}) \qquad I_{S,1} = [n + 1, m^2 + n] \ \ (m^2 \text{ timestamps})$$

$$I_{V,2} = [m^2+n+1, m^2+2n] \ \ (n \text{ timestamps}) \quad I_E = [m^2+2n+1, m^2+m+2n] \ \ (m \text{ timestamps})$$

$$I_{V,3} = [m^2 + m + 2n + 1, m^2 + m + 3n] \ \ (n \text{ timestamps})$$
$$I_{S,2} = [m^2 + m + 3n + 1, m^4 + m^2 + m + 3n] \ \ (m^4 \text{ timestamps})$$

$$I_{A,1} = [m^4 + m^2 + m + 3n + 1, m^4 + m^2 + m + 3n + 1] \ \ (1 \text{ timestamp})$$

$$I_{S,3} = [m^4 + m^2 + m + 3n + 2, 2m^4 + m^2 + m + 3n + 2] \ \ (m^4 \text{ timestamps})$$

$$I_{A,2} = [2m^4 + m^2 + m + 3n + 3, 2m^4 + m^2 + m + 3n + 3] \ \ (1 \text{ timestamp})$$

Notice that $I_{A,1}$ and $I_{A,2}$ consists of a single timestamp. The time domain $\mathcal{T}$ is the concatenation of the intervals defined previously:

$$\mathcal{T} = I_{V,1} \cdot I_{S,1} \cdot I_{V,2} \cdot I_E \cdot I_{V,3} \cdot I_{S,2} \cdot I_{A,1} \cdot I_{S,3} \cdot I_{A,2}$$

The set V_t is defined as follows:

$$V_t = \{u_i, u'_i : v_i \in V, 1 \leq i \leq n\} \cup \{u_0\} \cup \{u'_0\}$$

Now, we define the set E_t of temporal edges in each interval of the time domain $\mathcal{T}$. In each time interval $I_{S,x}$, $x \in \{1, 2, 3\}$, no temporal edge is active. Furthermore, we assume that the edges of G are ordered based on the lexicographic order, and we refer to the p-th edge of G as the edge in position p based on this order. Recall that $E_t(I)$ denotes the set of temporal edges active in interval I. Next, we define the sets of temporal edges in $I_{V,1}$, $I_{V,2}$, I_E, $I_{V,3}$, $I_{A,1}$ and $I_{A,2}$:

$$E_t(I_{V,1}) = \{\{u_i, u_0, i\} : 1 \leq i \leq n\}$$

$$E_t(I_{V,2}) = \{\{u_i', u_0, t\} : t = m^2 + n + i, 1 \leq i \leq n\}$$

$$E_t(I_E) = \{\{u_i, u_j, t\} : t = m^2 + 2n + p, 1 \leq p \leq m, \{v_i, v_j\} \text{ is the } p\text{-edges of } E\}$$

$$E_t(I_{V,3}) = \{\{u_i, u_i', t\} : t = m^2 + m + 2n + i, 1 \leq i \leq n\}$$

$$E_t(I_{A,1}) = \{\{u_0, u_0', t\} : t = m^4 + m^2 + m + 3n + 1\}$$

$$E_t(I_{A,2}) = \{\{u_0, u_0', t\} : t = 2m^4 + m^2 + m + 3n + 3\}.$$

We start by proving some properties of G_t. First, notice that by construction in each timestamp there exists at most one active temporal edge, thus G_t is an instance of 1-MinTimelineCover. Now, we present a property of vertices u_0 and u_0'.

Lemma 1. *Given an instance G of* Vertex Cover*, let G_t be the corresponding instance of* 1-MinTimelineCover*. Consider a solution $\mathcal{A}$ of* 1-MinTimelineCover *on instance G_t of span at most $n(m^2 + m + 2n)$, then each of the vertices u_0 and u_0' is active in exactly one of the timestamps of intervals $I_{A,1}$ and $I_{A,2}$.*

We show next how to relate a vertex cover of G and a solution of MinTimelineCover on G_t.

Lemma 2. *Given an instance G of* Vertex Cover*, let G_t be the corresponding instance of* 1-MinTimelineCover*. Given a vertex cover of G consisting of k vertices, there exists a solution of* 1-MinTimelineCover *on instance G_t of span at most $k(m^2 + m + 2n) + (n - k)(n + m)$.*

Proof. Consider a vertex cover $V' \subseteq V$ of G, with $|V'| = k$. We define a solution $\mathcal{A}$ of 1-MinTimelineCover as follows:

- Vertex u_0 (u_0', respectively), is active in the unique timestamp of interval $I_{A,1}$ (of interval $I_{A,2}$, respectively); u_0 and u_0' have a span equal to 0
- For each vertex $v_i \in V \setminus V'$, vertex u_i is active in timestamp i and has a span equal to 0, u_i' is active in the interval between timestamps $m^2 + n + i$ and $m^2 + m + 2n + i$, and has a span equal to $m + n$.

- For each vertex $v_i \in V'$, u_i is active in the interval between timestamp i and timestamp $m^2 + m + 2n + i$, and has a span equal to $m^2 + m + 2n$; vertex u'_i is active in timestamp $m^2 + n + i$, and has a span equal to 0.

First, we show that $\mathcal{A}$ is a solution of 1-MinTimelineCover, that is it covers every temporal edge of G_t. Indeed, the temporal edges of $I_{A,1}$ and $I_{A,2}$ are covered by u_0 and u'_0. The temporal edges in $I_{V,1}$ are covered by vertices u_i, $1 \leq i \leq n$, since u_i is active in timestamp i. The temporal edges of $I_{V,2}$ are covered by u'_i, $1 \leq i \leq n$, since u'_i is active in timestamp $m^2 + n + i$. Since V' is a vertex cover of G, by construction u_i is active in interval $[i, m^2 + m + n + i]$ or u_j is active in interval $[j, m^2 + m + 2n + j]$; both intervals include I_E where temporal edge $\{u_i, u_j, t\}$ is defined. Finally, the edges of $I_{V,3}$ are covered either by u_i, if $v_i \in V'$, since u_i in this case is active in interval $[i, m^2 + m + 2n + i]$, or by u'_i, if $v_i \in V \setminus V'$, since u'_i in this case is active in interval $[m^2 + n + i, m^2 + m + 2n + i]$. It follows that all the temporal edges are covered, thus $\mathcal{A}$ covers G_t.

Now, consider the span of $\mathcal{A}$. Since k vertices u_i (with $v_i \in V'$) have a span of $m^2 + m + 2n$ and $n - k$ vertices u'_i (with $v_i \in V \setminus V'$) have a span of $n + m$, the overall span of $\mathcal{A}$ is $k(m^2 + m + 2n) + (n - k)(n + m)$.

□

Based on Lemma 1, we can prove the following result.

Lemma 3. *Given an instance G of* Vertex Cover*, let G_t be the corresponding instance of* 1-MinTimelineCover*. Given a solution $\mathcal{A}$ of* 1-MinTimelineCover *on instance G_t having span $k(m^2+m+2n)+(n-k)(n+m)$, there exists a vertex cover of G consisting of at most k vertices.*

Now, we are able to prove the main result of this section.

Theorem 1. 1-MinTimelineCover *is NP-hard.*

Proof. It follows from Lemma 2 and Lemma 3, that we have designed a polynomial-time reduction from Vertex Cover to 1-MinTimelineCover. Since Vertex Cover is NP-hard [16], it follows that also 1-MinTimelineCover is NP-hard.

□

4. MinTimelineCover **for Bounded Degree and Time Domain**

In this section we consider another restriction of the MinTimelineCover problem, when the input temporal graph has bounded local degree and bounded time domain. We prove that MinTimelineCover is NP-hard even when the time domain consists of three timestamps and the local degree $\Delta_L \leq 2$, by giving a reduction from Vertex Cover on cubic graphs (a variant of Vertex Cover denoted by Cubic Vertex Cover). We recall that a graph is cubic when each of its vertex has degree three.

Given an instance $G = (V, E)$ of Cubic Vertex Cover, we build a corresponding temporal graph $G_t = (V_t, E_t, \mathcal{T})$ as follows (a sketch of G_t is given in Fig. 3). First, we define the time domain $\mathcal{T} = [1, 2, 3]$.

The vertex set V_t is defined as follows. Let U_i, with $v_i \in V$ and $1 \leq i \leq |V|$, be the following set of vertices:

$$U_i = \{u_{i,1}, u_{i,2}, u_{i,3}, w_{i,a}, w_{i,b}, z_i : v_i \in V\}.$$

The set V_t is then defined as follows:

$$V_t = \bigcup_{i=1}^{|V|} U_i.$$

The set E_t consists of three subsets $E_{t,1}$, $E_{t,2}$, $E_{t,3}$, representing edges active in timestamp $1, 2, 3$, respectively. As in the reduction of Section 3, we assume that the edges of G are ordered based on lexicographic order. Since G is cubic, based on this order we refer to the edges incident on a vertex $v \in V$ as the first (second, third, respectively) edge incident in v. The set $E_{t,1}$, $E_{t,2}$, $E_{t,3}$ are defined as follows:

$$E_{t,1} = \{\{w_{i,a}, z_i, 1\}, \{w_{i,b}, z_i, 1\} : v_i \in V\}$$

$$E_{t,2} = \{\{u_{i,1}, w_{i,a}, 2\}, \{u_{i,2}, w_{i,a}, 2\}, \{u_{i,2}, w_{i,b}, 2\}, \{u_{i,3}, w_{i,b}, 2\} : v_i \in V\}$$

$$E_{t,3} = \{\{w_{i,a}, z_i, 3\}, \{w_{i,b}, z_i, 3\} : v_i \in V\} \cup$$
$$\{\{u_{i,x}, u_{j,y}, 3\} : \{v_i, v_j\} \in E \text{ and } \{v_i, v_j\} \text{ is the } x\text{-th edge of } v_i$$
$$\text{and the } y\text{-th edge of } v_j \ 1 \leq x, y \leq 3\}$$

We start by proving some properties of the temporal graph G_t.

Lemma 4. *Given an instance G of* Cubic Vertex Cover*, let G_t be the corresponding instance of* MinTimelineCover*. Then the local degree Δ_L of G_T is* 2.

Next, we prove that we can restrict ourselves to solutions where each vertex z_i, $1 \leq i \leq |V|$, is active only in timestamp 3.

Lemma 5. *Given an instance G of* Cubic Vertex Cover*, let G_t be the corresponding instance of* MinTimelineCover*. Then, given a solution $\mathcal{A}$ of* MinTimelineCover *on instance G_t, we can compute in polynomial time a solution $\mathcal{A}'$ of* MinTimelineCover *on instance G_t such that each vertex z_i is active only in timestamp* 3 *and the span of $\mathcal{A}'$ is at most the span of $\mathcal{A}$.*

Now, we show how to relate and a solution of MinTimelineCover on G_t.

Lemma 6. *Given an instance G of* Cubic Vertex Cover*, let G_t be the corresponding instance of* MinTimelineCover*. Then, given a solution V' of* Cubic Vertex Cover*, with $|V'| = k$, we can compute in polynomial time a solution of* MinTimelineCover *on instance G_t of span at most $|E| + k - |V|$.*

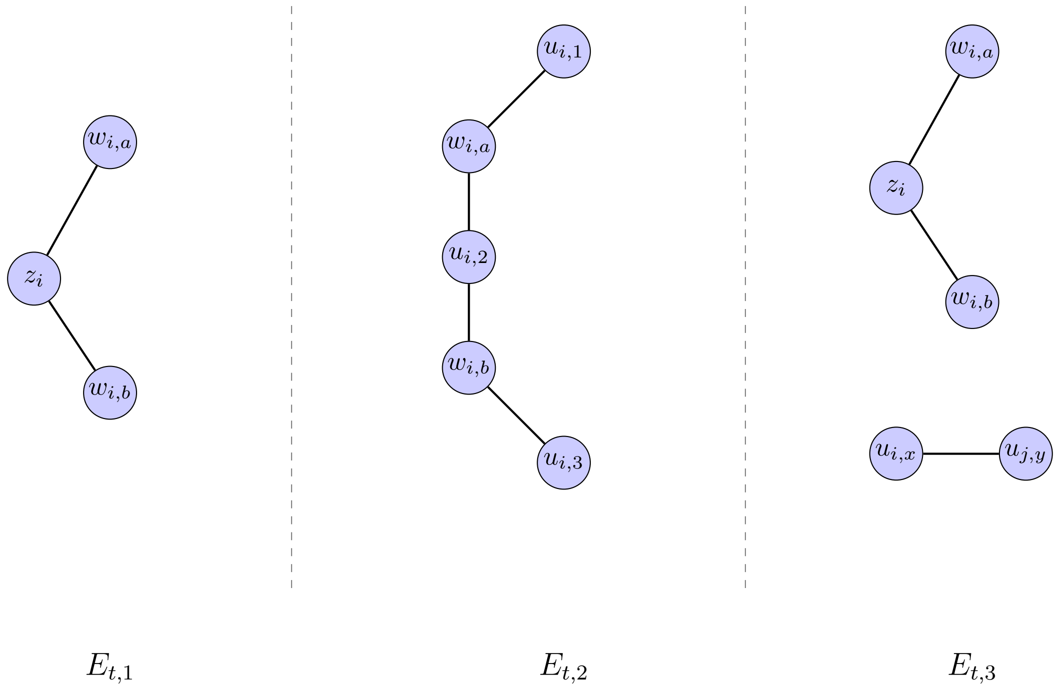

Figure 3: The structure of the temporal graph G_t; $E_{t,1}$, $E_{t,2}$, $E_{t,3}$ are the sets of temporal edges defined in the three timestamps 1, 2 and 3, respectively.

Proof. Consider a solution V' of Cubic Vertex Cover on instance G, we define a solution $\mathcal{A}$ of MinTimelineCover on instance G_t as follows. For each set U_i, $1 \leq i \leq |V|$, associated with $v_i \in V \setminus V'$, $\mathcal{A}$ is defined as follows:

- $w_{i,a}$, $w_{i,b}$ are active in interval $[1, 2]$, each one with span 1
- For each $\{v_i, v_j\} \in E$, which is the p-th edge of v_i and the q-th edge of v_j, $u_{i,p}$ is active in timestamp 3 with span 0
- Vertex z_i is active in timestamp 3, with span 0.

For each set U_i, $1 \leq i \leq |V|$, associated with $v_i \in V'$, $\mathcal{A}$ is defined as follows:

- Vertices $w_{i,a}$, $w_{i,b}$ are active in timestamp 1, each one with span 0
- Each vertex $u_{i,p}$, $1 \leq p \leq 3$, is active in timestamp 2 (the span of u_i depends on the next point)
- For each $\{v_i, v_j\} \in E$, with $v_i, v_j \in V'$, which is the p-th edge of v_i and the q-th edge of v_j: if $i < j$ then $u_{i,p}$ is active in interval $[2, 3]$ (with span 1), else $u_{j,q}$ is active in interval $[2, 3]$ (with span 1).
- Vertex z_i is active in timestamp 3.

By construction, $\mathcal{A}$ covers each temporal edge of G_t. Indeed, the temporal edges in $E_{t,1}$ are covered by $w_{i,a}, w_{i,b}$ for each i with $1 \leq i \leq |V|$. The temporal edges in $E_{t,2}$ are either covered by $w_{i,a}, w_{i,b}$, if $v_i \in V \setminus V'$, or by $u_{i,1}$, $u_{i,2}$, $u_{i,3}$, if $v_i \in V'$. The temporal edges $\{w_{i,a}, z_i, 3\}$ and $\{w_{i,b}, z_i, 3\}$ of $E_{t,3}$ are covered by z_i; the temporal edges $\{u_{i,x}, u_{j,y}, 3\}$ are covered by one of $u_{i,x}$, $u_{j,y}$.

Now, the span of $\mathcal{A}$ is 1 for each $\{v_i, v_j\} \in E$, where $v_i, v_j \in V'$. Consider now an edge $\{v_i, v_j\} \in E$, where either v_i or v_j in $V \setminus V'$, assume without loss of generality that $v_i \in V'$; then $\mathcal{A}$ has a span of 2 for the set U_i (both $w_{i,a}$ and $w_{i,b}$ have a span 1) for the three edges incident in v_i in G. Hence the overall span of $\mathcal{A}$ is $|E| - |V \setminus V'| = |E| + k - |V|$, thus concluding the proof.

□

Based on Lemma 5, we can prove the following result.

Lemma 7. *Given an instance G of* Cubic Vertex Cover *let G_t be the corresponding instance of* MinTimelineCover. *Then, given a solution* MinTimelineCover *on instance G_t of span $|E|+k-|V|$, we can compute in polynomial time a solution of* Cubic Vertex Cover *of size at most k.*

Now, we can prove the main result of this section.

Theorem 2. MinTimelineCover *is NP-hard even when the time domain consists of three timestamps and the local degree $\Delta_L \leq 2$.*

Proof. By Lemma 4, Lemma 6 and Lemma 7 it follows that that we have designed a polynomial-time reduction from Cubic Vertex Cover to MinTimelineCover when the time domain consists of three timestamps and the local degree $\Delta_L \leq 2$. Since Cubic Vertex Cover is NP-hard [17], it follows that also MinTimelineCover is NP-hard when the time domain consists of three timestamps and the local degree $\Delta_L \leq 2$.

□

5. Bounding the Time Window

In this section we consider the parameterized complexity of MinTimelineCover for (w, h)-window constrained temporal graphs, when the parameters are w, h (the size of the time window). Notice that when one of w, h is the parameter, the problem is not in the class XP (if $h = 2$ for the result in Section 3, if $w = 2$ for the hardness of MinTimelineCover on a time domain of two timestamps [15]).

We define $W_{j,w} = w([j-w+1, j])$, that is a time window of length w that ends in timestamp j and that consists of the set of pairs (v, t) such that v has an active edge in timestamp t, with $j - w + 1 \leq t \leq j$.

An *activity assignment* $F_{j,w}$ for a time window $W_{j,w}$ is a function that establishes, for each pair (v, t) of $W_{j,w}$, if v is active in timestamp t. Formally, an *activity assignment* $F_{j,w}$ is a function

$$F_{j,w} : W_{j,w} \to \{0, 1\}$$

such that for each pair (v, t) it holds that:

- v is active in timestamp t if and only if $F_{j,w}(v, t) = 1$ (thus v is not active in timestamp t if and only if $F_{j,w}(v, t) = 0$)

- if $F_{j,w}(v, t_1) = 1$ and $F_{j,w}(v, t_2) = 1$, with $j - w + 1 \leq t_1 \leq t_2 \leq j$, then $F_{j,w}(v, t) = 1$ for each t with $t_1 \leq t \leq t_2$.

The span of $F_{j,w}$, denoted by $s(F_{j,w})$, is the span induced by the activity assignment $F_{j,w}$.

Consider two time windows $W_{j,w}$ and $W_{i,w}$, with $1 \leq j - w + 1 \leq i < j \leq |\mathcal{T}|$, and two assignment functions $F_{j,w}$ and $F_{i,w}$. $F_{j,w}$ and $F_{i,w}$ are in *agreement* if, for each $(v, t) \in W_{j,w} \cap W_{i,w}$ it holds that $F_{j,w}(v, t) = F_{i,w}(v, t)$. An activity timeline $\mathcal{A}$ is in agreement with an activity assignment $F_{j,w}$ if the activity defined by $\mathcal{A}$ for the pairs in $W_{j,w}$ is identical to $F_{j,w}$.

Next, we describe a dynamic programming algorithm to compute a solution of MinTimelineCover parameterized by w and h. Given two assignment functions $F_{j,w}$ and $F_{j-1,w}$ that are in agreement, define the value $D(F_{j,w}, F_{j-1,w})$ as the span added by $F_{j,w}$ (in timestamp j) with respect to $F_{j-1,w}$. Formally, $D(F_{j,w}, F_{j-1,w})$ is defined as follows:

$$D(F_{j,w}, F_{j-1,w}) = |\{v : (v, j-1) \in W_{j-1,w} \wedge (v, j) \in W_{j,w} \wedge F_{j,w}(v, j) = F_{j-1,w}(v, j) = 1\}|$$

Given an activity assignment $F_{j,w}$ of an active window $W_{j,w}$, define the function $C[F_{j,w}]$ as the minimum span of an activity timeline of the temporal graph G_t on interval $[1, j]$, such that:

1. The activity of vertices in the time window $W_{j,w}$ is defined by $F_{j,w}$
2. Each temporal edge $\{u, v, t\}$ of G_t, with $1 \leq t \leq j$, is covered
3. Each vertex active in timestamp j is not active in interval $[1, j - w]$ (since G_t is (w, h)-window constrained)

Now, $C[F_{j,w}]$ is computed with the following recurrence:

- If $j > w$, then $C[F_{j,w}]$ is the minimum, over $F_{j-1,w}$ in agreement with $F_{j,w}$, of

$$C[F_{j-1,w}] + D(F_{j,w}, F_{j-1,w})$$

- If $j = w$, $C[F_{w,w}] = s(F_{w,w})$, that is the span of $F_{w,w}$.

Next, we show the correctness of the recurrence.

Lemma 8. *$C[F_{j,w}] = q$, if and only if there exists an activity timeline of G_t in interval $[1, j]$ of span q.*

Based on Lemma 8, we can prove the main result of this section.

Theorem 3. *A solution of* MinTimelineCover *on instance G_t can be computed in $O(2^{h(w+1)} h|\mathcal{T}|)$ time.*

Proof. The correctness of the recurrence follows from Lemma 8, thus the span of an optimal solution of MinTimelineCover on instance G_t is computed in entry $C[F_{|\mathcal{T}|,w}]$.

Next, we consider the time complexity of the algorithm. The base case $C[F_{w,w}]$ can be computed in $O(2^{hw})$ time, as there exist at most h vertices with active temporal edges in each timestamp of interval $[1, w]$. Now, we consider the case $j > w$. First, notice that each entry $C[F_{j,w}]$ can have at most 2^{hw} values and there are $O(\mathcal{T})$ of such entries. Now, $C[F_{j,w}]$

is computed starting from the values $C[F_{j-1,w}]$ (where $F_{j,w}$ and $F_{j-1,w}$ are in agreement) This requires the definition of the activity timeline of vertices active in timestamp j (at most h), in $O(2^h)$ time and, for each of them, the computation of $D(F_{j,w}, F_{j-1,w})$, which requires $O(h)$ time. Hence the overall time complexity of the dynamic programming algorithm is $O(2^{h(w+1)}h|\mathcal{T}|)$.

□

6. Conclusion

We have considered a variant of Vertex Cover, called MinTimelineCover, on temporal graphs. We have shown that the problem is NP-hard even in restricted cases (one temporal edge active in each timestamp, local degree two for temporal graphs on a time domain of three timestamps). Moreover, we have shown that MinTimelineCover is fixed-parameter tractable when the size of the activity time window is the parameter.

There are several interesting future directions related to MinTimelineCover. First, the parameterized complexity of MinTimelineCover is open when the problem is parameterized by the span of the solution (recall that in this case the problem is fixed-parameter tractable when the time domain consists of two timestamps [15]). Moreover, it would be interesting to analyze the approximation complexity of the problem.

References

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

[2] P. Holme, Modern temporal network theory: a colloquium, The European Physical Journal B 88 (2015) 234.

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

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

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

[6] 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. doi:10.1016/j.jcss.2019.07.006.

[7] 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. doi:10.1016/j.tcs.2019.03.031.

[8] B. M. Bumpus, K. Meeks, Edge exploration of temporal graphs, in: P. Flocchini, L. Moura (Eds.), Combinatorial Algorithms - 32nd International Workshop, IWOCA 2021, Ottawa, ON, Canada, July 5-7, 2021, Proceedings, volume 12757 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 107–121. doi:10.1007/978-3-030-79987-8_8.

[9] A. Marino, A. Silva, Königsberg sightseeing: Eulerian walks in temporal graphs, in: P. Flocchini, L. Moura (Eds.), Combinatorial Algorithms - 32nd International Workshop, IWOCA 2021, Ottawa, ON, Canada, July 5-7, 2021, Proceedings, volume 12757 of *Lecture Notes in Computer Science*, Springer, 2021, pp. 485–500. doi:10.1007/978-3-030-79987-8_34.

[10] P. Rozenshtein, F. Bonchi, A. Gionis, M. Sozio, N. Tatti, Finding events in temporal networks: segmentation meets densest subgraph discovery, Knowl. Inf. Syst. 62 (2020) 1611–1639. doi:10.1007/s10115-019-01403-9.

[11] R. Dondi, M. M. Hosseinzadeh, Dense sub-networks discovery in temporal networks, SN Comput. Sci. 2 (2021) 158. doi:10.1007/s42979-021-00593-w.

[12] 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. doi:10.1016/j.jcss.2021.04.001.

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

[14] T. Hamm, N. Klobas, G. B. Mertzios, P. G. Spirakis, The complexity of temporal vertex cover in small-degree graphs, CoRR abs/2204.04832 (2022). doi:10.48550/arXiv.2204.04832.

[15] V. Froese, P. Kunz, P. Zschoche, Disentangling the computational complexity of network untangling, in: L. D. Raedt (Ed.), Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI 2022, Vienna, Austria, 23-29 July 2022, ijcai.org, 2022, pp. 2037–2043. doi:10.24963/ijcai.2022/283.

[16] R. M. Karp, Reducibility among combinatorial problems, in: R. E. Miller, J. W. Thatcher (Eds.), Proceedings of a symposium on the Complexity of Computer Computations, held March 20-22, 1972, at the IBM Thomas J. Watson Research Center, Yorktown Heights, New York., The IBM Research Symposia Series, Plenum Press, New York, 1972, pp. 85–103.

[17] P. Alimonti, V. Kann, Some APX-completeness results for cubic graphs, Theor. Comput. Sci. 237 (2000) 123–134.