Assigning times to minimise reachability in temporal graphs

Jessica Enright
Global Academy of Agriculture and Food Security, University of Edinburgh, Edinburgh, UK
jessica.enright@ed.ac.uk

Kitty Meeks
School of Computing Science, University of Glasgow, Glasgow, UK
kitty.meeks@glasgow.ac.uk

Fiona Skerman
Department of Mathematics, Uppsala University, Uppsala, Sweden
fiona.skerman@math.uu.se

Abstract

Temporal graphs (in which edges are active only at specified time steps) are an increasingly important and popular model for a wide variety of natural and social phenomena. They are of particular relevance when considering spreading processes on graphs, for example the spread of a disease or the dissemination of sensitive information.

A rich topic for research into spreading processes on graphs has been the use of graph modifications to limit the number of vertices that can be reached from any given starting vertex. In this work we introduce a new type of modification for temporal graphs, in which edges cannot be deleted but we can control the times at which they are active; we investigate the problem of determining an assignment of times to the edges so as to minimise the maximum number of vertices reachable from any single starting vertex.

We study two versions of this problem, both of which we show to be NP-hard (the more general is hard even on trees and directed acyclic graphs), and identify a few cases in which the problem can be solved or approximated efficiently.

2012 ACM Subject Classification G.2.2 Graph Algorithms

Keywords and phrases Temporal graphs; graph modification; graph contagion; influence spreading

1 Introduction

Temporal (or dynamic) graphs have emerged recently as a useful structure for representing real-world situations, and as a rich source of new algorithmic problems [1,10,11,13,15,16,17]. A temporal graph changes over time: each edge in the graph is only active at certain timesteps. Some notions from the study of static graphs transfer immediately to the temporal setting, but others – including the very basic notions of connectivity and reachability – become much more complex in the temporal setting.

Temporal graphs are therefore of particular interest when considering the dynamics of spreading processes on graphs, for example the spread of a disease or the dissemination of sensitive information. The maximum number of vertices that can be reached from a single
Assigning times to minimise reachability in temporal graphs

starting vertex $v$, known as the reachability of $v$, has emerged as an important measure in both epidemiology \cite{6,7,14} and the study of network vulnerability \cite{5,9}; note that in the case of static, undirected graphs the reachability of $v$ is equal to the number of vertices in the connected component containing $v$.

Previous work on this topic has addressed questions of the following form: given a set of rules for how an input graph $G$ can be modified, how small can we make the maximum reachability taken over all vertices of $G$? The answer to these questions can be interpreted as the effectiveness of an optimal intervention to control a spreading process on the graph, or the worst-case impact of an attack on a graph’s reliability. Modification rules considered in the literature include vertex deletions, edge deletions, and the deletion of edges at some budgeted number of times. The related optimisation problems of deleting edge-times to minimise various cost measures while ensuring that the set of vertices reached from every vertex remains unchanged have also been studied \cite{1,16} (the notion of “assignment” of times to edges introduced in \cite{16} can be regarded as an edge-time deletion problem when all edges are initially active at all times).

Graphs of livestock trades present us with an example in which it is desirable to reduce the maximum reachability of a graph, as discussed in \cite{6,7}. Here, vertices represent farms, and the sale of an animal from one farm to another (together with the associated risk of disease transmission) is naturally represented with a (directed) edge. If a disease incursion starts at the farm represented by $v$, then the farms at risk of infection are precisely those in the reachability set of $v$, and it is natural to try to minimise the worst case number of farms that might be infected.

It is not clear, however, how one might realistically remove edges from such a graph – forbidding trade is likely impossible. A more realistic hope is that the set of edges (e.g. the trades that take place) is fixed, but we can intervene to alter the relative times at which the edges are active (i.e. reorder the trades, via alterations in sale and auction dates).

The focus of this paper is on precisely this kind of temporal graph modification: we cannot remove edges, but we can choose the order in which they are active. (In contrast with the model of \cite{16}, the number of edge-times to be assigned to each edge is fixed.) In the simplest version of the problem, we can reorder edges with complete freedom, but for many applications there are likely to be additional constraints which stop us rescheduling edges independently. Specifically, we may require that particular subsets of edges are all active simultaneously: this corresponds, for example, to a set of contacts that will take place at a particular conference, or the trades that will be made at a specific named livestock market event (e.g. the “Spring Bull Sale”), whenever the event is scheduled. Indeed, scenarios of this kind where the timing of contacts can be controlled by an organisation responsible for scheduling events (e.g. an auctioneer) perhaps represent the most likely real-world application for this rescheduling approach. We therefore consider a more general version of the problem involving assigning times to classes of simultaneously occurring edges. We mostly see that both versions of the problem are intractable, but we are able to identify a small number of special cases which admit polynomial-time exact or approximation algorithms.

The rest of the paper is organised as follows. We describe our model in more detail in Section \ref{sec:model}, give formal definitions of the problems we consider in Section \ref{sec:definitions} and summarise our results in Section \ref{sec:results}. Section \ref{sec:indep} is devoted to the setting in which edges can be reordered independently, and in Section \ref{sec:classes} we consider the generalisation of the problem involving classes of simultaneously active edges. In an appendix, we give some results on a variant of the problem in which the goal is instead to maximise the size of the smallest reachability set.
1.1 Our model, notation, and some simple observations on reachability

A vertex \( v \) is reachable from vertex \( u \) in a graph (digraph) if there is path (directed path) from \( u \) to \( v \) in that graph (digraph). The reachability set of \( v \) is the set of all vertices reachable from \( v \). A directed acyclic graph (DAG) is a digraph that does not contain any directed cycle. In any graph, we refer to a vertex of degree one as a leaf, and any edge incident with a leaf is a leaf edge. For any natural number \( n \), we write \([n]\) as shorthand for the set \( \{1, \ldots, n\} \).

In its most general form, a temporal graph is a static (directed) graph \( G = (V, E) \), which we refer to as the underlying graph, together with a function \( T \) that maps each edge to a list of time steps at which that edge is active. Note that these time steps need not give a continuous interval for any particular edge. A strict temporal path (sometimes called a time-respecting path in the literature) from \( u \) to \( v \) in a temporal graph \( (G = (V, E), T) \) is a (directed) path from \( u \) to \( v \) composed of edges \( e_0, e_1, \ldots, e_k \) such that each edge \( e_i \) is assigned a time \( t(e_i) \) from its image in \( T \) where \( t(e_i) < t(e_{i+1}) \) for \( 0 \leq i < k \). We say a vertex \( v \) is temporally reachable from \( u \) in a temporal graph \( (G = (V, E), T) \), if there is a strict temporal path from \( u \) to \( v \); we adopt the convention that every vertex \( v \) is temporally reachable from itself. A vertex \( v \) is temporally reachable from \( u \) after time \( \tau \) if there is a temporal path from \( u \) to \( v \) in which every edge is assigned a time strictly greater than \( \tau \). A weak temporal path, and the concept of weak temporal reachability are defined analogously, where we replace the requirement \( t(e_i) < t(e_{i+1}) \) with a weak inequality. Both notions have been considered in the literature on temporal graphs (see, for example [3, 10, 17, 8]); in this paper we use the strict notion of temporal reachability (and, unless otherwise stated, all temporal paths should be assumed to be strict), although many of our results can also be adapted to the weak case.

We call the set of vertices that are temporally reachable from vertex \( u \) in a temporal graph the temporal reachability set of \( u \). Note that the temporal reachability set of any vertex \( u \) in the temporal graph \( (G, T) \) is a subset of its reachability set in the static underlying graph \( G \). The maximum temporal reachability of a temporal graph is the maximum cardinality of the temporal reachability set of any vertex in the graph. Note that the temporal reachability set of any vertex can be computed using a breadth-first search, and so the maximum reachability can be computed in polynomial time by considering each vertex in turn.

As we are only interested in reachability in the sense defined above, we do not care about the absolute times at which edges are active, simply the order in which they are active. Moreover, we will assume that it is determined in advance which subsets of edges will be active simultaneously, and the number of timesteps at which each such subset is active: this can be represented by a multiset \( \mathcal{E} = \{E_1, \ldots, E_h\} \) of subsets of the edge-set of \( G \), where each element of \( \mathcal{E} \) is a subset of edges which is active simultaneously, and the multiplicity of each element in \( \mathcal{E} \) is equal to the number of timesteps at which the subset is active. This model is appropriate when each element of \( \mathcal{E} \) corresponds to the connections that will be active when a particular event (e.g. a livestock market, a conference, or a flight) takes place: we can potentially change the relative timing of the various events, but each event will cause a fixed subset of edges to be active.

We further restrict the way in which we may alter timing by requiring that each element of \( \mathcal{E} \) is assigned a distinct timestep. Some restriction of this kind is required to avoid the problem becoming trivial in the setting where we require strict temporal reachability: we could always minimise the maximum reachability by assigning all edge subsets the same timestep. (Note that, in the case of weak reachability, we will always be able to find a solution that does not reuse timesteps and is at least as good as any that does.) With these restrictions, the timesteps at which each edge is active can be specified uniquely by...
the multiset $E$ together with a bijection $t: E \to |E|$: the edge $e$ is active at time $s$ if and only if $t(E_i) = s$ for some $E_i \in E$ with $e \in E_i$. We will write $(G, E, t)$ for a temporal graph represented in this way, and $\text{reach}_{G, E, t}(u)$ for the temporal reachability set of $u$ in $(G, E, t)$.

We now make some simple observations about reachability sets.

Proposition 1. Let $G$ be an undirected graph, $v$ a vertex of degree one in $G$, and $u$ the unique neighbour of $v$. Then $\text{reach}_{G, E, t}(v) \subseteq \text{reach}_{G, E, t}(u)$.

Proposition 2. Let $uv$ be an edge in the undirected graph $G$, and assume that $uv$ is first active at time $t_1$, and no other edge incident with $v$ is active at any time before or equal to $t_1$. Then $\text{reach}_{G, E, t}(v) \subseteq \text{reach}_{G, E, t}(u)$.

For the next observation, we need one more piece of notation: we write $N_G(v)$ for the set of vertices in $G$ that are adjacent to $v$.

Proposition 3. For any undirected temporal graph $(G, E, t)$ and $v \in V(G)$, we have $\text{reach}_{G, E, t}(v) \supseteq N_G(v) \cup \{v\}$.

In a directed graph $G$, we write $N_G^{\text{out}}(v)$ for the set of vertices $\{u: \overrightarrow{vu} \in E(G)\}$. We can now make an analogous observation for the directed case.

Proposition 4. For any directed temporal graph $(G, E, t)$ and $v \in V(G)$, we have $\text{reach}_{G, E, t}(v) \supseteq N_G^{\text{out}}(v) \cup \{v\}$.

Finally, we define the edge-class interaction graph of $(G, E)$ (where $G$ is an undirected graph) to be the graph with vertex-set $[h]$ in which vertices $i$ and $j$ are adjacent if and only if there exist $e_i \in E_i$ and $e_j \in E_j$ such that $e_i$ and $e_j$ are incident.

Proposition 5. Suppose that $i$ and $j$ are non-adjacent vertices in the edge-class interaction graph of $(G, E)$, and that the sets $E_i$ and $E_j$ are active at consecutive timesteps. Then swapping the timesteps assigned to $E_i$ and $E_j$ does not change the reachability set of any vertex in $G$.

An analogous result holds for directed graphs if we define the edge-class interaction graph by making $i$ and $j$ adjacent if and only if there is some $v \in V(G)$ such that $v$ has an in-edge in $E_i$ and an out-edge in $E_j$ (or vice versa).

1.2 Problems considered

The main problem we consider in this paper is the following.

**Min-Max Reachability Temporal Ordering**

**Input:** A graph $G = (V, E)$, a list $E = \{E_1, \ldots, E_h\}$ of subsets of $E$, and a positive integer $k$

**Question:** Is there a bijective function $t: E \to [h]$ such that maximum temporal reachability of $(G, E, t)$ is at most $k$?

The simplest case is when $E$ consists of pairwise disjoint singleton sets (so that all edges can be reordered independently; this is analogous to the notion of single-edge single-label temporal graphs in [9]); we refer to this special case as **Singleton Min-Max Reachability Temporal Ordering**. In this singleton setting, we will sometimes abuse notation and
consider $t$ to be an ordering of the edges of $G$, and describe a temporal graph as $(G, E, t)$ where $E$ is the edge-set of $G$.

Observe that changing the order in which edges are active can have a huge impact on the reachability of vertices in the graph. For example, let $G$ consist of a clique on $r$ vertices, where each vertex in the clique has $s$ pendant leaves. If all edges in the clique are active (in an arbitrary order) before all edges incident with the leaves, then each vertex in the clique will reach the entire graph, a total of $r(s + 1)$ vertices. On the other hand, if edges incident with leaves are active first (in some order), then no vertex reaches more than $r + s$ vertices (including itself).

We conclude this section with two simple observations about situations in which both problems admit efficient algorithms; $\Delta(G)$ denotes the maximum degree of the graph $G$.

**Proposition 6.** **Min-Max Reachability Temporal Ordering** and is solvable in polynomial-time when $k \leq \Delta(G)$.

**Proof.** By Proposition 3, we know that the maximum reachability set of $(G, E, t)$ is at least $\Delta(G) + 1 > k$, so we must have a no-instance.

**Proposition 7.** **Min-Max Reachability Temporal Ordering** belongs to FPT when parameterised by the number $h$ of timesteps.

**Proof.** Given any instance of Min-Max Reachability Temporal Ordering, there are at most $h!$ bijective functions $t$ from $E_1, \ldots, E_h$ to $h$. Having fixed such an ordering, we can compute the reachability of each vertex of the corresponding temporal graph in linear time using a breadth-first search, and hence we can find the maximum temporal reachability of the whole temporal graph in time $O(n(n + m))$. It follows that we can solve Min-Max Reachability Temporal Ordering in time $O(h! \cdot n(n + m))$ by considering each of the $h!$ orderings in turn.

### 1.3 Summary of results

We begin by considering the special case in which $E$ consists of pairwise disjoint, singleton edge sets. We show that Singleton Min-Max Reachability Temporal Ordering is NP-complete on general graphs, but can be solved efficiently in some special cases: Singleton Min-Max Reachability Temporal Ordering is polynomial-time solvable on DAGs, and admits an FPT algorithm parameterised by either (1) vertex cover number, or (2) the maximum permitted reachability $k$, when restricted to trees. We also give a polynomial-time algorithm to compute a constant-factor approximation to the optimisation version of the problem on graphs of bounded maximum degree.

In the more general case of Min-Max Reachability Temporal Ordering, we show that the problem remains NP-complete even on DAGs, and is W[1]-hard on trees when parameterised by the vertex cover number. In this setting we can obtain a constant-factor approximation to the optimisation problem on graphs of bounded maximum degree provided that the edge-class interaction graph is bipartite or has bounded degree.

## 2 The case of singleton edge classes

In this section we consider the restricted version of the problem in which we can reschedule each edge independently. In Section 2.1 we show that even this special case of the problem is NP-complete on general graphs, before giving efficient exact algorithms for some special cases.
Assigning times to minimise reachability in temporal graphs

in Section 2.2 and deriving some general bounds on the achievable maximum reachability which lead to an approximation algorithm for bounded degree graphs in Section 2.3.

Several results in this section will make use of the following simple lemma.

\textbf{Lemma 8.} Given any yes-instance \((G, E, k)\) of Singleton Min-Max Reachability in Temporal Ordering, there is an assignment \(t : E \rightarrow |E|\) of times to edges such that the maximum reachability of \((G, E, t)\) is at most \(k\) and all leaf-edges are assigned times strictly before all other edges. Moreover, in such an assignment, the relative order of the leaf-edges does not change the maximum reachability.

\textbf{Proof.} Suppose that \(t : E \rightarrow |E|\) is an assignment of times to edges such that the maximum reachability of \((G, E, t)\) is at most \(k\), and suppose that \(v\) is a leaf with neighbour \(u\). We claim that, if \(t'\) is the assignment of times to edges obtained from \(t\) by moving \(uv\) to the start and leaving the relative order of edges otherwise unchanged, then no vertex in \(G\) has reachability set of size more than \(k\) in \((G, E, t')\). To see this, first note that, for any vertex \(w \in \{u, v\}\) with \(u \in \text{reach}_{(G, E, t)}(w)\), we have \(u \notin \text{reach}_{(G, E, t')}(w)\), and so \(|\text{reach}_{(G, E, t')}(w)| < |\text{reach}_{(G, E, t)}(w)|\). Next, observe that the reachability set of \(u\) is unchanged as it includes \(v\) under any ordering, and so has size at most \(k\). Finally we observe that, although the reachability set of \(v\) could increase in size, we must have \(|\text{reach}_{(G, E, t')}(v)| \leq |\text{reach}_{(G, E, t')}(u)|\) and so, by the reasoning above, \(|\text{reach}_{(G, E, t')}(v)| \leq k\).

Finally, to see that the relative ordering of leaf-edges does not change the maximum reachability, we consider two cases. First, if the leaf-edges \(e_1\) and \(e_2\) are not incident, then we know by Proposition 5 that swapping the times assigned to \(e_1\) and \(e_2\) has no effect on the reachability of any vertex. Secondly, if \(e_1\) and \(e_2\) are incident, then swapping their assigned times results in an isomorphic temporal graph (i.e. the vertices can be relabelled to give an identical temporal graph) and so in particular the maximum reachability does not change. 

\textbf{2.1 NP-completeness}

In this section we prove that Singleton Min-Max Reachability in Temporal Ordering is NP-complete. Due to space constraints, full details of the proof (including figures illustrating the construction) are postponed to the appendix.

\textbf{Theorem 9.} Singleton Min-Max Reachability in Temporal Ordering is NP-complete.

\textbf{Proof.} We give a reduction from the following problem, shown to be NP-complete in [2]; here \(e(A, B)\) denotes the number of edges with one endpoint in \(A\) and the other in \(B\).

\begin{center}
\begin{talign}
\text{Minimum Bisection for Cubic Graphs} \\
\text{Input:} \text{ A 3-regular graph } G = (V, E), \text{ and a positive integer } c. \\
\text{Question:} \text{ Is there a partition of } V \text{ into two sets } A \text{ and } B \text{ with } |A| = |B| \text{ such that } e(A, B) \leq c? \\
\end{talign}
\end{center}

Let \((G = (V, E), c)\) be an instance of Minimum Bisection for Cubic Graphs. We will construct an instance of Singleton Min-Max Reachability in Temporal Ordering which is a yes-instance if and only if \((G, c)\) is a yes-instance for Minimum Bisection for Cubic Graphs. In defining our instance of Singleton Min-Max Reachability in Temporal Ordering we will make use of two substructures, which we now define.
For positive integers \( r \) and \( s \) and a vertex \( v \), an \((r, s)\)-decoration consists of a clique on \( r \) vertices, each of which has an additional \( s \) neighbours of degree one. We say that an \((r, s)\)-decoration is \textit{rooted at} \( v \) if there is an edge from \( v \) to each vertex in the clique. We make the following observation about \((r, s)\)-decorations.

\[ \text{Observation 1.} \]

Let \( F \) be a graph which contains an \((r, s)\)-decoration rooted at \( v \). Then, for any ordering of the edges within the decoration such that all leaf-edges come first, the maximum reachability set of any vertex in the decoration (except perhaps \( v \)) has cardinality precisely \( r + s + p \), where \( p \) is the number of vertices outside the decoration reachable from \( v \) (including \( v \) itself) strictly after the first edge in the decoration incident with \( v \) is active.

For any two vertices \( u \) and \( v \) and an integer \( h \geq 4 \), a \( u-v \) \( h \)-gadget consists of a triangle \( abc \) where \( a \) is adjacent to \( u \), \( b \) is adjacent to \( v \), \( a \) and \( b \) each have \( h - 4 \) additional pendant leaves and \( c \) has \( h - 3 \) additional pendant leaves. (see Figure 1).

\[ \text{Observation 2.} \]

Let \( F \) be a graph which contains a \( u-v \) \( h \)-gadget. There is an ordering of the edges of \( F \), in which all leaf-edges come first, such that no vertex in the gadget, except perhaps \( u \) and \( v \), has a reachability set of cardinality greater than \( h \). In any such ordering:

1. \( ua \) is active strictly after any other edge incident with \( u \), and \( vb \) is active strictly after any other edge incident with \( v \);
2. \( ua \) and \( vb \) are both active strictly before \( ab \);
3. \( ua \) is active strictly before \( ac \), and \( vb \) is active strictly before \( bc \);
4. \( u \) and \( v \) both reach \( a, b \) and \( c \).

Moreover, there exists such an ordering in which \( u \) and \( v \) both reach only \( a \), \( b \) and \( c \) within the gadget, and no vertex in the gadget other than \( u \) and \( v \) reaches any vertex outside the gadget.

We now define the graph \( H \) in our instance of \textsc{Singleton Min-Max Reachability Temporal Ordering}. For each vertex \( v_i \in V(G) = \{v_1, \ldots, v_n\} \), the graph \( H \) contains a three-vertex path \( v_i^a w_i v_i^b \); there is an \((r, h - r - 10)\)-decoration rooted at each of \( v_i^a \) and \( v_i^b \), where \( r \) and \( h \) are values to be determined later. For every edge \( v_i v_j \in E(G) \), \( H \) contains a \( v_i^a - v_j^b \) \( h \)-gadget and a \( v_i^b - v_j^a \) \( h \)-gadget. Finally, \( H \) contains two further vertices \( x_a \) and \( x_b \); \( x_a \) is adjacent to \( v_1^a, \ldots, v_n^a \) and \( x_b \) is adjacent to \( v_1^b, \ldots, v_n^b \). The construction of \( H \) is illustrated in Figure 2. Finally, we set the maximum permitted reachability to be \( h \); to complete the
definition of the instance of Singleton Min-Max Reachability Temporal Ordering it remains to fix the values of \( r \) and \( h \), which we will do after some analysis of the reachability sets in \( H \).

Suppose that the ordering \( t \) of the edges gives maximum reachability at most \( h \). We may assume by Lemma \( 3 \) that all edges incident with a leaf come at the start of \( t \); in particular this means that the leaves incident with \( w_i \) are only in the reachability set of \( w_i \) for each \( i \). Using Observations \( 1 \) and \( 2 \) we can deduce that, in order for the maximum size of the reachability set of any vertex \( v \) that both \( v^a \) (respectively \( v^b \)) and at most \( r \) edges between them.

Thus, by Observation \( 1 \), in order for the maximum size of the reachability set of any vertex \( v \) to be at most \( h \), all edges incident with \( v^a \) (respectively \( v^b \)) that do not belong to either an incident gadget or decoration must be active strictly before all those that do. By Observation \( 2 \), we may assume without loss of generality that, for each edge incident with \( v_i \) in \( G \), in \( H \) the vertex \( v^a_i \) (respectively \( v^b_i \)) reaches precisely those vertices in the corresponding gadget that belong to the central triangle. It further follows from Observation \( 2 \) that \( v^a_i \) (respectively \( v^b_i \)) reaches these vertices strictly after any other edge incident with \( v^a_i \) (respectively \( v^b_i \)) is active. Thus, by Observation \( 4 \), in order for the maximum size of the reachability set of any vertex in the decoration pendant at \( i \) to be at most \( h \), all edges incident with \( v^a_i \) (respectively \( v^b_i \)) that do not belong to either an incident gadget or decoration must be active strictly before all those that do.

Note that, in addition to its pendant leaves, each vertex \( w_i \) must reach \( v^a_i \) and \( v^b_i \), together with the \( r \) neighbours each of these two vertices has in a pendant decoration, and the nine vertices of gadgets that each reaches. Thus the reachability set of each vertex \( w_i \) will contain at least \( 1 + (h − 2r − 21) + 2 + 2r + 18 = h \) vertices; to avoid reaching more than \( h \) vertices, it must be that \( x_a v^a_i \) (respectively \( x_b v^b_i \)) is active strictly before any other edge incident with \( v^a_i \) (respectively \( v^b_i \)). Based on this assumption, we observe that the reachability set of any vertex \( v^a_i \) is a subset of the reachability set of \( x_a \) (and similarly for \( v^b_i \) and \( x_b \)). Thus the only vertices whose reachability sets we still need to consider are \( x_a \) and \( x_b \). If we define \( A \) to be the set of vertices \( v_i \) such that \( v^a_i w_i \) is active strictly before \( v^b_i w_i \), and \( B = V \setminus A \), it is straightforward to verify that

\[
1 + n + n + rn + 3 \cdot \frac{3}{2} n + |A|(1 + r) + 3|\{ e \in E(G) : e \ has\ at\ least\ one\ endpoint\ in\ A\}| = 1 + n \left( \frac{13}{2} + r \right) + |A|(1 + r) + 3 \cdot \frac{3}{2} |A| + 3 \frac{3}{2} e(A, B),
\]

\[
= 1 + n \left( \frac{13}{2} + r \right) + |A| \left( \frac{11}{2} + r \right) + 3 \frac{3}{2} e(A, B).
\]

Similarly, we see that \( x_b \) has a reachability set of size

\[
1 + n \left( \frac{13}{2} + r \right) |B| \left( \frac{11}{2} + r \right) + 3 \frac{3}{2} e(A, B).
\]

We now set \( r = 3n \) and

\[
h = 1 + n \left( \frac{13}{2} + 3n \right) + n \left( \frac{11}{2} + 3n \right) + \frac{3n}{2}.
\]

If \((G, \alpha)\) is a yes-instance, it is clear that we also have a yes-instance for Singleton Min-Max Reachability Temporal Ordering, as we can choose the relative order of each pair of edges \((v^a_i w_i, v^b_i w_i)\) so that the resulting sets \( A \) and \( B \) have the same cardinality and at most \( \alpha \) edges between them. Conversely, we need to argue that \((G, \alpha)\) is a yes-instance whenever we can order edges so that both \( x_a \) and \( x_b \) have reachability sets of size at most \( 1 + n \left( \frac{13}{2} + 7n \right) + \frac{3}{2} \left( \frac{11}{2} + 7n \right) + \frac{3n}{2} \).
Figure 2 The construction of the graph $H$ in the instance of **Singleton Min-Max Reachability Temporal Ordering.**
We may assume without loss of generality that \( \alpha \leq \frac{3}{4}n \) (the total number of edges in \( G \)) and so \( \frac{3}{4} \alpha \leq \frac{3}{4}n < 3n \). Thus, even if \( e(A, B) = 0 \), we cannot have
\[
1 + n \left( \frac{13}{2} + r \right) + \max \{|A|, |B|\} \left( \frac{11}{2} + r \right) + \frac{3}{2} e(A, B) \leq 1 + n \left( \frac{13}{2} + 3n \right) + n \left( \frac{11}{2} + 3n \right) + \frac{3\alpha}{2}
\]
unless \( |A| = |B| = n/2 \). In this case, it is clear that the inequality only holds if \( e(A, B) \leq \alpha \).

Thus, if the maximum reachability is at most \( h \), there must be a partition of \( V(G) \) into two sets \( A \) and \( B \) with \( |A| = |B| \) such that \( e(A, B) \leq \alpha \), as required.

### 2.2 Exact algorithms for special cases

In this section, we demonstrate that certain restrictions on the input graph lead to efficient exact algorithms in the singleton edge-class setting. We begin by giving a simple necessary and sufficient condition for a yes-instance when the input graph is a DAG, implying that the problem is solvable in polynomial time in this case.

**Theorem 10.** Let \((G, \mathcal{E}, k)\) be an instance of Singleton Min-Max Reachability Temporal Ordering where \( G = (V, \overrightarrow{E}) \) is a DAG. Then \((G, \mathcal{E}, k)\) is a yes-instance if and only if \( k \geq \Delta^{\text{out}} + 1 \), where \( \Delta^{\text{out}} \) is the maximum out-degree of \( G \).

**Proof.** We know by Proposition 4 that, for any bijection \( t: \mathcal{E} \to [|E|] \), the maximum reachability of \((G, \mathcal{E}, t)\) will be at least \( \Delta^{\text{out}} + 1 \), so if \( k \leq \Delta^{\text{out}} \) we will certainly have a no-instance.

Conversely, we argue that there is an ordering \( t: \mathcal{E} \to [|E|] \) such that the maximum reachability of \((G, \mathcal{E}, t)\) is at most \( \Delta^{\text{out}} + 1 \). Fix a topological ordering \( \{v_1, \ldots, v_m\} \) of the vertices of \( G \). We now define a related ordering of the edges: fix any ordering \( \{e_1, \ldots, e_m\} \) of \( E(G) \) such that \( \overrightarrow{v_i \cdot w} \) precedes \( \overrightarrow{v_j \cdot w} \) whenever \( v_i \) precedes \( v_j \) in the topological ordering. If \( E_i = \{v_i\} \) for each \( i \), we now define \( t(e_i) = m + 1 - i \in [m] \).

We claim that there is no strict temporal path on more than one edge in \((G, \mathcal{E}, t)\). Suppose, for a contradiction that \( u, v, w \) is a strict temporal path in \((G, \mathcal{E}, t)\). In this case we must have \( t(\overrightarrow{v \cdot w}) > t(\overrightarrow{u \cdot w}) \), which by definition of \( t \) means that \( \overrightarrow{v \cdot w} \) precedes \( \overrightarrow{u \cdot w} \) in our edge ordering; this implies that \( v \) precedes \( u \) in the topological ordering of the vertices, but as \( v \) is reachable from \( u \) this is not possible. Thus we can conclude that the longest temporal path in \((G, \mathcal{E}, t)\) consists of a single edge, so the reachability set of any vertex \( v \) is contained in \( \{v\} \cup N^{\text{out}}_G(v) \), as required.

Next, we consider the situation in which the input graph is a tree. We first give a simple argument showing that, when restricted to trees, Singleton Min-Max Reachability Temporal Ordering is in FPT parameterised by the vertex cover number of the tree; the vertex cover number of a graph is the cardinality of the smallest vertex set \( U \) such that every edge has at least one endpoint in \( U \). The key observation is that we can bound the number of non-leaf edges in terms of the vertex cover number, and then exploit Lemma 5.

**Theorem 11.** Let \( T = (V_T, E_T) \) be a tree with \( n \) vertices and vertex cover number \( c \). Then we can solve Singleton Min-Max Reachability Temporal Ordering on \( T \) in time \((2c - 2)!n^2\).

**Proof.** We begin by arguing that \( T \) has at most \( 2c - 1 \) non-leaf vertices. Let \( U \) be a vertex cover for \( T \) of size \( c \), and let \( U_1, \ldots, U_r \) be the connected components of \( T[U] \), the subgraph of \( T \) induced by \( U \). Recall that \( W = V_T \setminus U \) must be an independents set, and note that any \( w \in W \) has at most one neighbour in each connected component of \( T[U] \), otherwise we
would have a cycle. Moreover, there are at most \( r - 1 \) vertices in \( W \) with more than one neighbour in \( U \), as the existence of at least \( r \) such vertices would again imply the existence of a cycle in \( T \). Thus we can conclude that the number of vertices of degree at least two in \( T \) is at most \( |U| + r - 1 \leq 2r - 1 \).

By Lemma 5, it suffices to consider orderings of the edges in which all leaf-edges are assigned times strictly before all other edges; moreover, the relative ordering of leaf edges does not matter. Thus, it remains only to consider all possible orderings of edges whose endpoints both have degree at least two. Since we have argued that there are at most \( 2r - 1 \) vertices of degree at least two and \( T \) is acyclic, we can conclude that there are at most \( 2r - 2 \) such edges, and so we need to consider at most \( (2r - 2)! \) possible edge orderings. For any given ordering we can compute the maximum reachability in time \( \mathcal{O}(n^2) \) using a breadth first search (as the number of edges is linear in the number of vertices), giving a total time complexity of \( \mathcal{O}((2r - 2)! n^2) \).

We now give a more general result, showing that we can solve Singleton List Min-Max Reachability Temporal Ordering in polynomial time whenever both the maximum permitted reachability and the maximum length of any list are bounded by constants; in fact we give an FPT algorithm with respect to this dual parameterisation. Intuitively, our approach records states that capture the information needed from all possible orderings of edges in subtrees of a rooted version of our tree, computing these states using dynamic programming from the leaves upward.

\begin{theorem}
When the underlying graph \( G \) is a tree, Singleton Min-Max Reachability Temporal Ordering can be solved in time \( n \cdot k^{\mathcal{O}(k)} \).
\end{theorem}

\textbf{Proof.} Let \( (T = (V_T, E_T), \mathcal{E} = \{e : e \in E_T\}, k) \) be the input to our instance of Singleton Min-Max Reachability Temporal Ordering. We fix an arbitrary root vertex \( v_r \in V_T \), and for each \( v \in V_T \) we denote by \( T_v \) the subtree of \( T \) rooted at \( v \). For every vertex \( v \in V_T \) other than the root, we refer to the first edge on the path from \( v \) to \( v_r \) as \( e_v \). By Proposition 8 we may assume that the maximum degree of \( T \) is at most \( k - 1 \).

For each vertex \( v \in V_T \), we record a set of states; a state is a pair \((\alpha, \beta)\) where \( \alpha, \beta \in [k] \).

For vertex \( v \), we say a state \((\alpha, \beta)\) is realisable if there is an ordering of the edges incident at \( v \) in \( T_v \) such that:
1. \( e_v \) is assigned time \( \tau_i \),
2. the largest reachability set within \( T_v \) that reaches \( v \) before \( e_v \) is of size \( \alpha \),
3. the number of vertices (including \( v \)) reachable from \( v \) after \( e_v \) is exactly \( \beta \), and
4. there is no reachability set within \( T_v \) under this temporal ordering of size more than \( k \).

Observe that there are at most \( k^2 \) possible states for any vertex.

It is straightforward to generate all realisable states at a leaf \( v \); the realisable states here are precisely the states \((1, 1)\). Now consider a non-leaf vertex \( v \). Assume that we have a list of all realisable states for each child of \( v \). We argue that we can find all realisable states of \( v \) from the realisable states of its children in time bounded by a function of \( k \).

We will reason about a joint state \( \{(\alpha_1, \beta_1), \ldots, (\alpha_d, \beta_d)\} \) of the children \( v_1, \ldots, v_d \) of \( v \), along with an ordering \( \Pi \) of the edges incident at \( v \). A state \((\alpha, \beta)\) of a non-root vertex \( v \) is realisable if and only if there is a joint state \( \{(\alpha_1, \beta_1), \ldots, (\alpha_d, \beta_d)\} \) of the children of \( v \) and an ordering \( \Pi \) of the edges incident at \( v \) such that:

\begin{itemize}
    \item \( \alpha = \max \{\Pi(u v_i) \mid \Pi(e_v) \leq \Pi(u v_i) \} (\sum_{j \in \Pi(u v_i) \mid \Pi(w v_i) \leq \Pi(u v_i)} \beta_j) + \alpha_i + 1 \)
    \item \( \beta = \sum_{i \in \Pi(u v_i) \mid \Pi(w v_i) \leq \Pi(u v_i)} (\beta_i) + 1 \), and
    \item for all \( 1 \leq i \leq d \), \( \sum_{j \in \Pi(u v_i) \mid \Pi(w v_i) \leq \Pi(u v_i)} \beta_j \) + \( \alpha_i + 1 \leq k \).
\end{itemize}
Assigning times to minimise reachability in temporal graphs

At the root vertex $v_r$ we follow a similar procedure, but without reference to the edge to a parent: a given state $(\alpha, \beta)$ of $v_r$ is realisable if and only if there exists some joint state \{$(\alpha_1, \beta_1), \ldots, (\alpha_d, \beta_d)$\} of the children of the root, along with an ordering $\Pi$ of the edges to those children such that:

\[
\alpha = \max \sum_{j, \Pi(vv_j) > \Pi(vv_i)} (\beta_j) + \alpha_i + 1
\]

\[
\beta = \sum_{i, \Pi(vv_i) > r} (\beta_i) + 1, \text{ and}
\]

\[
\text{for all } 1 \leq i \leq d, \sum_{j, \Pi(vv_j) > \Pi(vv_i)} (\beta_j) + \alpha_i + 1 \leq k.
\]

Thus, in order to compute the list of realisable states for a vertex $v$ with $d$ children in the rooted tree, we consider all possible joint child states. A joint child state consists of an ordering of the edges incident at a vertex, for which there are at most $d!$ possibilities, together with a choice of one of the realisable states for each child (where each child has at most $k^2$ realisable states). Therefore the number of possible joint child state and ordering combinations to consider is at most $d!k^{2d}$. For each joint child state and ordering, we can determine the corresponding states of the parent in time $O(d^2)$. Thus the total time required at each vertex is $O(d!k^{2d}d^2)$. By our initial assumption that the maximum degree is at most $k - 1$, it follows that the time required at any vertex is $O(k^2k^2) = k^{O(k)}$. It follows that we can solve Singleton Min-Max Reachability Temporal Ordering in time $n \cdot k^{O(k)}$.

Note that the procedure described can easily be adapted to output a suitable ordering of the edges, if one exists: at each vertex $v$, we concatenate the orderings of edges in the subtrees rooted at its children in such a way that the relative ordering $\Pi$ of edges incident with $v$ is preserved.

We further observe that this approach can be extended to give a polynomial-time algorithm to solve Singleton Min-Max Reachability Temporal Ordering on graphs of bounded treewidth. However, in the simplest approach we have been able to devise, the states that are required for accounting at each bag of the tree decomposition and the details of the calculation of parent states from child states are sufficiently complex that they are unlikely to yield a practically implementable algorithm. We give an outline of the state we believe is sufficient here, and leave the development of a more practical approach as future work.

\textbf{Theorem 13.} Singleton Min-Max Reachability Temporal Ordering is solvable in polynomial time on graphs of bounded treewidth, provided the maximum permitted reachability in the instance is bounded by a constant.

\textbf{Sketch proof.} We assume that we are provided with a nice tree decomposition of $G$ along with the input; for the definition of a nice tree decomposition we refer the reader to \[3\]. The techniques used here are standard, so we only describe the possible states we record for each node in the decomposition.

We adapt the state used in the proof of Theorem 12 but this becomes much more complicated when dealing with tree decompositions. Let $B$ be a bag in a nice tree decomposition, and $E_B$ the set of all edges incident at members of $B$. Let $\mathcal{P}(B)$ be the powerset of $B$, and $\mathcal{D}$ be the set of all functions with domains that are members of $\mathcal{P}(B)$ and ranges that are subsets of $|E_B|$. Then a state for bag $B$ consists of:

- $\Pi$: an assignment of times from $1...|E_B|$ to the members of $E_B$, essentially an ordering of these edges
- $\mathcal{Y}^T$: a function from the cross product $\mathcal{D} \times \mathcal{D}$ to a natural number between 1 and $k$

A state $(\Pi, \mathcal{Y}^T)$ for bag $B$ is realisable if there exists an ordering of the edges incident at vertices present either in $B$ or its descendants such that:
the ordering is consistent with $\Pi$, and

- under that edge ordering, for every $(U^T, W^T) \in \mathcal{P}(B) \times \mathcal{P}(B)$, $V^T((U^T, W^T))$ is the size of the union of
  - the largest reachability set of a vertex in $B$ or one of its descendants that reaches exactly vertices in the domain of $U^T$ by their times assigned by $U^T$ and no other vertices in $B$, and
  - the size of the joint reachability set of vertices in the domain of $W^T$ after their times assigned by $W^T$.

The need for the complexity of this state comes from one complicating case: essentially the vertex added at an introduce node may be reached by some existing large reachability set and itself also reach a large reachability set, both of which contain vertices that are present only in the descendants of the node, but not at the introduce node’s bag itself. To perform the necessary accounting, we must know the size of the union of these two sets - they may have many vertices in common. This motivates the requirement for the function $\mathcal{V}^T$, and the consideration of all pairs of subsets of the bags, with times assigned to their members.

\section{General bounds and an approximation algorithm}

In this section we show that, in the singleton case, it is always possible to find an ordering of the edges so that the maximum reachability is bounded by a function of the edge chromatic number of the input graph, and that this bound is in fact tight for certain graphs. As a consequence, we obtain a constant-factor approximation algorithm for bounded degree graphs. We begin with a general result showing that we can bound the maximum reachability by considering the maximum reachability of subgraphs that partition the edge-set.

\begin{lemma}
Let $G = (V, E)$ be a graph, $\mathcal{E}$ a collection of subsets of $E$, and let $\mathcal{P} = \{\mathcal{E}_1, \ldots, \mathcal{E}_s\}$ be a partition of $\mathcal{E}$. Let $G_i = (V, \bigcup \mathcal{E}_i)$ for each $1 \leq i \leq s$, and suppose that for each $i$ there is an assignment $t_i$ of times to elements of $\mathcal{E}_i$ such that the maximum reachability of $(G_i, \mathcal{E}_i, t_i)$ is at most $r_i$. Then there is an assignment $t$ of times to elements of $\mathcal{E}$ such that the maximum reachability of $(G, \mathcal{E} = E, t)$ is at most $\prod_{i=1}^s r_i$.
\end{lemma}

\begin{proof}
We proceed by induction on $s$. The base case, for $s = 1$, is trivially true, so suppose that $s > 1$ and that the result holds for any $s' < s$.

Let $\mathcal{E}' = \bigcup_{i=1}^{s-1} \mathcal{E}_i$, and $G' = (V, \bigcup \mathcal{E}')$. We can then apply the inductive hypothesis to $G'$, $\mathcal{E}'$ and $\mathcal{P}' = \{\mathcal{E}_1, \ldots, \mathcal{E}_{s-1}\}$ to see that there exists an assignment $t'$ of times to elements of $\mathcal{E}'$ such that the maximum reachability of $(G', \mathcal{E}', t')$ is at most $\prod_{i=1}^{s-1} r_i$.

Now consider an assignment $t$ of times to elements of $\mathcal{E}$, such that $t(E_j) = t'(E_j)$ if $E_j \in \mathcal{E}'$, and otherwise $t(E_j) = t_s(E_j) + |\mathcal{E}'|$. Note that, for any $v \in V$, the set of vertices that $v$ reaches in $(G, \mathcal{E}, t)$ after time $|\mathcal{E}'|$ is precisely reach$_{G, \mathcal{E}, t_s}(v)$.

We claim that the maximum reachability of $(G, \mathcal{E}, t)$ is at most $\prod_{i=1}^s r_i$, as required. To see that this is true, we fix an arbitrary vertex $v \in V$, and argue that $|\text{reach}_{G, \mathcal{E}, t}(v)| \leq \prod_{i=1}^s r_i$. Set $U = \text{reach}_{G', \mathcal{E}', t'}(v)$ and recall that $|U| \leq \prod_{i=1}^{s-1} r_i$. Now let $w \in \text{reach}_{G, \mathcal{E}, t} \setminus U$, and observe that there must be a strict temporal path from some $u \in U$ to $w$ that uses only edges of $\mathcal{E}_s$; equivalently, $w \in \text{reach}_{G, \mathcal{E}_s, t_s}(u)$. We therefore see that

$$|\text{reach}_{G, \mathcal{E}, t}(v)| = \bigcup_{u \in U} \text{reach}_{G, \mathcal{E}_s, t_s}(u) \leq |U| \cdot r_s \leq \prod_{i=1}^s r_i,$$

completing the proof.
\end{proof}
We use this result to obtain an upper bound on the maximum reachability that can be achieved in the singleton case, in terms of the edge chromatic number of $G$. The edge chromatic number of $G = (V, E)$, written $\chi'(G)$, is the smallest $c \in \mathbb{N}$ such that there is a colouring $f : E \rightarrow [c]$ such that $f(e_1) \neq f(e_2)$ whenever $e_1$ and $e_2$ are incident.

**Theorem 15.** Given any graph $G = (V, E)$, there is assignment $t : E(G) \rightarrow |E(G)|$ of times to edges such that the maximum reachability of $(G, E, t)$ is at most $2^{|E(G)|}$.

**Proof.** Fix a proper edge colouring $c : E(G) \rightarrow \chi'(G)$ of $G$, and suppose that $E_i$ is the set of edges receiving colour $i$ under $c$. Since $G_i = (V, E_i)$ consists of disjoint edges and perhaps isolated vertices, the largest connected component of $G_i$ contains at most two vertices, and hence for any assignment $t_i$ of times to edges in $E_i$ we have that the maximum reachability of $(G_i, E_i, t_i)$ is at most two. The result now follows immediately from Lemma 14.

We now argue that the bound in Theorem 15 is tight for both paths and binary trees; note that the edge chromatic number of any tree is equal to the maximum degree, so Theorem 15 gives upper bounds of four and eight respectively for these graphs.

**Proposition 16.** Let $P = (V_P, E_P)$ be a path on at least five vertices. Then there is no assignment $t$ of times to edges of $P$ such that the maximum reachability of $(V_P, E_P, t)$ is strictly less than four.

**Proof.** Let $e_1 = uv$ and $e_2 = vw$ be two incident, non-leaf edges; we may assume without loss of generality that $t(e_1) < t(e_2)$. Recall from Proposition 3 that $u$ necessarily reaches all of its neighbours, so $|\text{reach}_{V_P, E_P, t}(u) \setminus \{w\}| \geq 3$; however, the fact that $t(e_1) < t(e_2)$ means that there is a strict temporal path from $u$ to $w$ and so $w \in \text{reach}_{V_P, E_P, t}(u)$, implying that $|\text{reach}_{V_P, E_P, t}(u)| \geq 4$.

**Proposition 17.** Let $B = (V_B, E_B)$ be a binary tree, and let $t$ be any assignment of times to edges of $B$. Then the maximum reachability of $(B, E_B, t)$ is at least 8.

**Proof.** We will assume, for a contradiction, that no vertex has a reachability set of size greater than seven in $(B, E_B, t)$.

We claim that $B$ must contain some edge $e_1$ such that:

1. $t(e_1) < t(e')$ for every edge $e'$ incident with $e_1$,
2. both endpoints of $e_1$ are at distance at least four from any leaves, and
3. both endpoints of $e_1$ are at distance at least four from the root.

To find such an edge, we start at an arbitrary vertex $s$ at distance three from the root (i.e. at depth four); we will construct a path starting at $s$ which leads away from the root. Each time we reach a new vertex, we choose our next edge to be the edge (out of the two possibilities) which is assigned the earlier time. We stop when we reach a vertex $t$ at distance three from a leaf, and call the resulting path $P$.

We say that an edge $e$ on $P$, not incident $s$ or $t$, is a minimum edge if both other edges of $P$ incident with $e$ are assigned times strictly later than that assigned to $e$. If $P$ does not contain a minimum edge, then it consists of a (possibly empty) segment on which the assigned times of edges increase along the path, followed by a (possibly empty) segment on which the assigned times of edges decrease along the path. If either segment contains more than six edges, we would have a reachability set of size at least eight within this subpath, so we may assume that both segments contain at most 6 edges. Since such a path has precisely $d - 5$ edges, this is only possible if $d - 5 \leq 12$, that is, if $d \leq 17$. We may therefore assume that $P$ contains a minimum edge, $e_1$. 

23:14 Assigning times to minimise reachability in temporal graphs
We now argue that $e_1$ has the required properties. It is clear from the construction of $P$ and the fact that $e_1$ is not incident with either endpoint of $P$ that conditions (2) and (3) are satisfied. Condition (1) follows from the definition of a minimum edge together with the construction of $P$: since $P$ is a minimum edge, it must be assigned a time before the earlier of the two incident edges below $e_1$ (as this is the edge incident with $e_1$ in $P$; the definition of a minimum edge means that $e_1$ is assigned a time later than the edge above it (the other edge incident with $e_1$ in $P$); if the remaining edge incident with $e_1$ was assigned a time earlier than $e_1$ we would have chosen $P$ to include this edge instead.

We continue the argument using this choice of edge $e_1$; we will reason about a subtree including $e_1$, which is illustrated in Figure 3. Let $v$ be the endpoint of $e_1$ furthest from the root. It is clear that all vertices adjacent to both endpoints of $e_1$, together with the endpoints themselves, will belong to the reachability set of $v$ (see highlighted vertices in Figure 3). Thus, if the reachability set of $v$ has size at most seven, it can contain at most one more vertex; we may therefore assume without loss of generality that no descendant of $v$, other than the neighbours of $v$, is reachable from $v$.

Let $e_2 = vu$ and $e_3 = vw$ be the other two edges incident with $v$, and assume that $t(e_2) < t(e_3)$. Further let $e_4 = ux$ and $e_5 = uy$ be the other two edges incident with $u$, and assume that $t(e_4) < t(e_5)$. As we are assuming that $v$ does not reach any of its descendants other than $u$ and $w$, we conclude that $t(e_4), t(e_5) < t(e_2)$. Finally, let $e_6 = yz$ and $e_7 = yz'$ be the other two edges incident with $y$, where $t(e_6) < t(e_7)$.

Suppose first that $t(e_6) < t(e_3)$; we will argue that in this case the reachability set of $z$ will have size at least eight. To see this, note that $z$ reaches itself, both its children, $z'$, $y$, $u$, $v$ and $w$. Thus we may assume from now on that $t(e_6) > t(e_3)$ (and hence also $t(e_7) > t(e_5)$).

Now consider the reachability set of $x$. We see that $x$ reaches both of its children, $u$, $y$, $z$, $z'$, $v$ and $w$, giving $x$ a reachability set of size at least nine, which is a contradiction. Therefore we conclude that the maximum reachability of $(B, E_B, t)$ is at least 8, as required.

We conclude this section by using Theorem 15 (combined with Proposition 3) to derive a linear-time approximation algorithm, whose optimisation ratio depends only on the maximum degree of the input graph.
Assigning times to minimise reachability in temporal graphs

Theorem 18. Given any graph $G$, we can compute a $2^{\Delta(G)+1}$-approximation to the optimisation version of Singleton Min-Max Reachability Temporal Ordering in linear time, where $\Delta(G)$ denotes the maximum degree of $G$. Moreover, we can also compute an assignment of times to edges which achieves this approximation ratio in polynomial time.

Proof. We claim that it suffices to compute the maximum degree $\Delta(G)$ and to return $2^{\Delta(G)+1}$. To see that this satisfies the requirements, let $\operatorname{opt}(G)$ denote the smallest value of the maximum reachability of $(G, E, t)$ taken over all assignments $t$ of times to edges. We know from Theorem 15 that $\operatorname{opt}(G) \leq 2^{\Delta(G)+1}$; conversely by Proposition 3 we know that $\operatorname{opt}(G) \geq \Delta(G) + 1$. Hence

$$\operatorname{opt}(G) \leq 2^{\Delta(G)+1} \leq \frac{2^{\Delta(G)+1}}{\Delta(G) + 1} \operatorname{opt}(G),$$

as required. To see that we can compute a suitable assignment of times to edges in polynomial time, we observe that a $(\Delta + 1)$-edge colouring of any graph can be constructed in polynomial time [18]; given such a colouring we follow the method of Theorem 15 to construct a suitable assignment of times.

Corollary 19. Given any graph $G$ of bounded degree, a constant-factor approximation to the optimisation version of Singleton Min-Max Reachability Temporal Ordering can be computed in linear time.

The general problem

In this section we see that the tractable cases we identified in Section 2 do not extend to the more general setting where edge classes of cardinality greater than one are allowed. We begin by complementing Theorem 10 with a hardness result for DAGs.

Theorem 20. Min-Max Reachability Temporal Ordering is NP-complete, even if $G$ is a DAG with bounded degree, $k$ is at most 9, and $|E_i| \leq 3$ for each $E_i \in \mathcal{E}$.

Proof. We provide a reduction from $(3,4)$-SAT. Let $\Phi = C_1 \wedge \cdots \wedge C_m$ be our instance of $(3,4)$-SAT, and suppose that the variables in $\Phi$ are $x_1, \ldots, x_n$. We construct an instance $(G, \mathcal{E}, k)$ (with the properties in the statement of the theorem) which is a yes-instance if and only if $\Phi$ is satisfiable.

The vertex-set of $G$ consists of two sets, $V_{\text{clause}} = \{c_j : 1 \leq j \leq m\}$, and $V_{\text{var}} = \{v_{x_i,1}, v_{x_i,2}, v_{x_i,3}, v_{\neg x_i,1}, v_{\neg x_i,2}, v_{\neg x_i,3} : 1 \leq i \leq n\}$. $G$ contains directed edges $\overrightarrow{v_{x_i,1}v_{x_i,2}}$, $\overrightarrow{v_{x_i,2}v_{x_i,3}}$, $\overrightarrow{v_{\neg x_i,1}v_{\neg x_i,2}}$, and $\overrightarrow{v_{\neg x_i,2}v_{\neg x_i,3}}$ for each $1 \leq i \leq n$; for each $1 \leq j \leq m$, if $C_j = (\ell_1 \lor \ell_2 \lor \ell_3)$, we also have edges $\overrightarrow{c_jv_{\ell_1,1}}$, $\overrightarrow{c_jv_{\ell_2,1}}$, and $\overrightarrow{c_jv_{\ell_3,1}}$.

We now define the set $\mathcal{E}$ of edge-classes. For each clause $C_j$ and literal $\ell$ appearing in $C_j$, we have four sets in $\mathcal{E}$:

- two copies of the set $\{c_j, \overrightarrow{v_{\ell,1}v_{\ell,2}} \}$, denoted $E_{C_j,\ell}^{(1)}$ and $E_{C_j,\ell}^{(2)}$, and
- two copies of the set $\{c_j, \overrightarrow{v_{\ell,1}v_{\ell,2}} \}$, denoted $E_{C_j,\neg\ell}^{(1)}$ and $E_{C_j,\neg\ell}^{(2)}$.

We complete the construction of our instance of Min-Max Reachability Temporal Ordering by setting $k = 9$. It is straightforward to verify that $G$ is a DAG with bounded degree.

Note that the only vertices with reachability set of cardinality greater than 3 in the static graph $G$ are those corresponding to clauses, so it suffices to argue that there is a function
Suppose now that $\Phi$ has a satisfying assignment $B : \{x_1, \ldots, x_n\} \rightarrow \{\text{TRUE}, \text{FALSE}\}$. Let $t$ be any bijection $\mathcal{E} \rightarrow [4m]$ such that $t(\mathcal{E}_{C_\ell, t}) \leq 2m$ whenever $B(\ell)$ evaluates to TRUE, and $t(E^{(i)}_{C_\ell, t}) \geq 2m + 1$ whenever $B(\ell)$ evaluates to FALSE. Fix an arbitrary clause $C_j$. Since $B$ is a satisfying assignment for $\Phi$, we know that there is some literal $\ell$ appearing in $C_j$ which evaluates to TRUE under $B$. We claim that $v_{t, 3} \notin \mathcal{E}_{C_\ell, t}(c_j)$. To see that this is true, observe that the edge $v_{t, 1}v_{t, 3}$ appears only in the sets $E^{(1)}_{C_\ell, t}$ and $E^{(2)}_{C_\ell, t}$ where either $\ell$ or $\neg\ell$ appears in $C_i$; since we are assuming that $B(\ell)$ evaluates to TRUE, it follows from the definition that each such set is active only during the first $2m$ timesteps. On the other hand, $v_{t, 1}v_{t, 3}$ appears only in the sets $E^{(1)}_{C_\ell, t}$ and $E^{(2)}_{C_\ell, t}$ where either $\ell$ or $\neg\ell$ appears in $C_r$, and so is only at timesteps greater than or equal to $2m + 1$. Since the only directed path from $c_j$ to $v_{t, 3}$ uses the edges $v_{t, 1}v_{t, 3}$ and $v_{t, 2}v_{t, 3}$ in this order, we see that there cannot be a strict temporal path from $c_j$ to $v_{t, 3}$ in $(\mathcal{E}, t)$. Hence $|\mathcal{E}_{C_\ell, t}(c_j)| \leq 9$, as required.

Conversely, suppose that there is a bijection $t : \mathcal{E} \rightarrow [4m]$ such that the maximum reachability of $(\mathcal{E}, t)$ is at most 9. We define $\text{maxtime}_t(x_i)$ to be the latest timestep assigned by $t$ to any edge-set of the form $E_{c_j, t}^{(r)}$ where $r \in \{1, 2\}$ and $\ell \in \{x_i, \neg x_i\}$. We now define a truth assignment as follows:

$$B(\ell) = \begin{cases} \text{TRUE} & \text{if } t^{-1}(\text{maxtime}_t(x_i)) \text{ is of the form } E_{c_j, t}^{(r)}; \\ \text{FALSE} & \text{if } t^{-1}(\text{maxtime}_t(x_i)) \text{ is of the form } E_{c_j, t}^{(r)}, \end{cases}$$

Now fix an arbitrary clause $C_j$ and suppose that the literal $\ell \in \{x_i, \neg x_i\}$ appears in $C_j$. We claim that, if $B(\ell)$ evaluates to FALSE, we have $v_{t, 1}, v_{t, 2}, v_{t, 3} \in \mathcal{E}_{C_\ell, t}(c_j)$. By construction of $B$, we know that $t^{-1}(\text{maxtime}_t(x_i))$ is of the form $E_{c_j, t}^{(r)}$ (for some clause $C_r$ which involves the variable $x_i$) and so includes the edge $v_{t, 2}v_{t, 3}$. By definition of $\text{maxtime}_t(x_i)$, this means there exist distinct timesteps $s_1, s_2 < \text{maxtime}_t(x_i)$ such that $t(E_{c_j, t}^{(r)}) = s_r$ for $r \in \{1, 2\}$; without loss of generality we may assume that $s_1 < s_2$. Since $c_j, v_{t, 3} \in E_{c_j, t}^{(1)} = t^{-1}(s_1)$ and $v_{t, 1}v_{t, 3} \in E_{c_j, t}^{(2)} = t^{-1}(s_2)$, we have a strict temporal path $c_j, v_{t, 1}, v_{t, 2}, v_{t, 3}$, so we do indeed have $v_{t, 1}, v_{t, 2}, v_{t, 3} \in \mathcal{E}_{C_\ell, t}(c_j)$. Hence, if every literal in $C_j$ evaluates to FALSE under $B$, we would have $|\mathcal{E}_{C_\ell, t}(c_j)| = 10$, contradicting our assumption that the maximum reachability of $(\mathcal{E}, t)$ is at most 9. Thus we can conclude that every clause contains at least one literal which evaluates to TRUE under $B$, and so $B$ is a satisfying assignment for $\Phi$. □

Next we show that, in contrast with Theorem 11, the general version of the problem is $\text{W}[1]$-hard when parameterised by the vertex cover number of the input graph, even when the input graph is a tree.

**Theorem 21.** **Min-Max Reachability Temporal Ordering** is $\text{W}[1]$-hard parameterised by the vertex cover number of $G$, even if we require $G$ to be a tree.

**Proof.** We prove this result by means of a reduction from the following problem, shown to be $\text{W}[1]$-hard in 4.
Let \((G, k)\) be the input to an instance of \textbf{p-Clique}, and suppose that \(V(G) = \{v_1, \ldots, v_n\}\) and \(E(G) = \{e_1, \ldots, e_m\}\). We will construct an instance \((G', \{E_1, \ldots, E_h\}, k')\) of MIN-MAX REACHABILITY TEMPORAL ORDERING, such that \((G', \{E_1, \ldots, E_h\}, k')\) is a yes-instance for MIN-MAX REACHABILITY TEMPORAL ORDERING if and only if \((G, k)\) is a yes-instance for \textbf{p-Clique}.

We construct \(G'\) as follows. Let \(P\) be a path on \(k + 1\) vertices, whose endpoints are denoted \(s\) and \(r\) respectively. We obtain \(G'\) from \(P\) by adding \(n \left(\frac{k}{2}\right) + 1\) new leaves \(\{u_i^j : 1 \leq i \leq n, 1 \leq j \leq \left(\frac{k}{2}\right) + 1\}\) adjacent to \(s\) and \(m\) new leaves \(\{w_j : e_j \text{ incident with } v_i\}\) adjacent to \(r\). Note that \(G'\) has \(k + 1 + m + n \left(\frac{k}{2}\right) + 1\) = \(O(m + k^2n)\) vertices.

We now define the edge subsets \(E = \{E_1, \ldots, E_n\}\): we have one subset corresponding to each vertex of \(G\). We set

\[
E_i = \{e \in E(P)\} \cup \{su_i^j : 1 \leq j \leq \left(\frac{k}{2}\right) + 1\} \cup \{rw_j : e_j \text{ incident with } v_i\}.
\]

To complete the construction of our instance of MIN-MAX REACHABILITY TEMPORAL ORDERING, we set \(k' = |G'| - \left(\frac{k}{2}\right)\). It is clear that we can construct \((G', E, k')\) from \((G, k)\) in polynomial time.

We begin by arguing that \(s\) is the only vertex in \(G'\) whose reachability set can contain more than \(k'\) vertices, regardless of the choice of ordering.

\begin{claim}
Fix an arbitrary bijective function \(t : E \rightarrow [n]\). Then, for any vertex \(x \in V(G') \setminus \{s\}\), \(|\text{reach}_{G', E, t}(x)| \leq k'\).
\end{claim}

\begin{proof}
Fix \(x \in V(G') \setminus \{s\}\). It suffices to demonstrate that we can find \(\left(\frac{k}{2}\right)\) vertices in \(V(G')\) that are not in \(\text{reach}_{G', E, t}(x)\).

First suppose that \(x = u_i^j\) for some \(i\) and \(j\), and set \(U = \{u_i^\ell : \ell \neq j\}\). Note that \(|U| = \left(\frac{k}{2}\right)\). Since both edges on the unique path from \(x\) to any \(u \in U\) belong only to the edge subset \(E_i\), there cannot be a strict temporal path from \(x\) to \(u\). Thus \(U\) is a set of \(\left(\frac{k}{2}\right)\) vertices which are not contained in the temporal reachability set of \(x\).

Now suppose that \(x \neq u_i^j\) for any \(i, j\). Fix \(i\) such that \(E_i = t^{-1}(1)\). We claim that \(U = \{u_i^j : 1 \leq j \leq \left(\frac{k}{2}\right) + 1\}\) does not lie in the temporal reachability set of \(x\). To see that this is the case, note that edges incident with vertices in \(U\) are only active at timestep 1, and so if any such edge belongs to a strict temporal path it must be the first edge on such a path; hence there can only be a strict temporal path from some vertex \(y\) to a vertex \(u \in U\) if \(y\) is adjacent to \(u\). However, by choice of \(x\) (which is neither \(u_i^j\) for any \(i, j, or s\)) we know that \(x\) is not adjacent to any vertex in \(U\), and hence no vertex in \(U\) is in the reachability set of \(x\). Since \(|U| = \left(\frac{k}{2}\right) + 1\), this completes the proof of the claim.
\end{proof}

We will say that the bijective function \(t : E \rightarrow [n]\) is \textit{good for} \(s\) if \(|\text{reach}_{G', E, t}(s)| \leq k'\). It follows from Claim 22 that \((G', E, k')\) is a yes-instance if and only if some function \(t\) is good for \(s\). It therefore remains to show that there is a function \(t\) which is good for \(s\) if and only if \(G\) contains a clique of on \(k\) vertices.

To show that this is true, we first give a characterisation of the temporal reachability set of \(s\).
Claim 23. Fix an arbitrary bijective function $t: E \to [n]$. Then the only vertices of $G'$ that do not belong to $\text{reach}_{G',E,i}(s)$ are vertices $w_i$ such that $e_i = v_jv_k$ and $t(E_j), t(E_k) \leq k$.

Proof. First observe that, for any choice of $t$, $\text{reach}_{G',E,i}(s)$ contains
1. every vertex $w_i$ (with $1 \leq i \leq n$ and $1 \leq j \leq (\frac{k}{2}) + 1$), and
2. every vertex of $P$.

Now consider a vertex $w_i$; by definition, $w_i$ is in $\text{reach}_{G',E,i}(s)$ if and only if there is a strict temporal path from $s$ to $w_i$. There is only one possible choice of path, and the first $k$ edges on this path are active at every timestep. Thus we have a strict temporal path from $s$ to $w_i$ if and only if the edge $vw_i$ is active at some timestep after the first $k$. Since $vw_i$ is active only at $t(E_j)$ and $t(E_k)$, where $e_i = w_jw_k$, this means that $w_i$ is in the temporal reachability set of $s$ if and only if at least one of $t(E_j)$ and $t(E_k)$ is strictly greater than $k$. Conversely, the only vertices of $G'$ that are not in $\text{reach}_{G',E,i}(s)$ are the vertices $w_i$ such that $e_i = v_jv_k$ and $t(E_j), t(E_k) \leq k$, as required. □ (Claim)

Now suppose that $G$ contains a clique induced by the vertices $\{v_{i_1}, \ldots, v_{i_k}\}$. We claim that any function $t$ which maps $\{E_{i_1}, \ldots, E_{i_k}\}$ to $[k]$ is good for $s$. By Claim 23, we see that the vertices of $G'$ that are not in $\text{reach}_{G',E,i}(s)$ are the vertices $w_i$ such that $e_i = v_jv_k$ and $E_j, E_k \in t^{-1}([k]) = \{E_{i_1}, \ldots, E_{i_k}\}$. In other words, the vertices not in the temporal reachability set correspond to edges in $G$ which have both endpoints in the set $\{v_{i_1}, \ldots, v_{i_k}\}$. Since, by assumption, this set of vertices induces a clique, we know that there are precisely $\binom{k}{2}$ such vertices, so the reachability set misses $\binom{k}{2}$ vertices and $t$ is indeed good for $s$.

Conversely, suppose that the function $t$ is good for $s$, and set $C = \{i : t(E_i) \leq k\}$. We claim that $\{v_i : i \in C\}$ induces a clique in $G$. We know by Claim 23 that the only vertices that do not belong to $\text{reach}_{G',E,i}(s)$ are vertices $w_i$ such that $e_i = v_jv_k$ and $j, \ell \in C$. We therefore know, since $t$ is good for $s$, that there must be $\binom{k}{2}$ unordered pairs $\{j, \ell\} \subset C$ such that $G$ contains an edge $v_jv_\ell$. Since the total number of unordered pairs from $C$ is equal to $\binom{k}{2}$, it follows that there is an edge between every pair of vertices in the set $\{v_i : i \in C\}$, implying that this set of $k$ vertices does indeed induce a clique in $G$, as required.

Finally, we note that the vertices $s$ and $t$ together with every second internal vertex on the path $P$ form a vertex cover for $G'$, meaning that the vertex cover number of $G'$ is at most $k/2 + 1$.

The expressive power of Min-Max Reachability Temporal Ordering on highly restricted graph classes comes from the fact that, with edge-classes of size two or more, the decisions made at one location can have an effect on distant parts of the graph. It is therefore natural to ask whether we can regain some tractability in this setting by placing structural restrictions on the edge-class interaction graph: note that, in the proof of Theorem 24, although the graph $G$ we construct is a tree, the edge-class interaction graph is a clique.

We now build on the results of Section 2.3 to show that suitable restrictions on the edge-class interaction graph can allow the design of efficient approximation algorithms; it remains open whether there exist efficient algorithms to solve the problem exactly when the edge-class interaction graph is sufficiently highly structured. We begin by using Lemma 14 to give an analogous bound to that of Theorem 15 in the general (non-singleton) case.

Theorem 24. Let $(G,E,k)$ be an instance of Min-Max Reachability Temporal Ordering, let $H$ be the edge-class interaction graph of $(G,E)$, and let $d = \max_{E' \in E} \Delta((V,E'))$ be the maximum number of edges from any one element of $E$ that are incident with any single vertex of $G$. In this case there is an assignment $t : E \to |E|$ of times to edge classes such that the maximum reachability of $(G,E,t)$ is at most $(d+1)\chi(H)$, where $\chi(H)$ is the chromatic number of $H$. ▶
Proof. Fix a proper vertex colouring \( c : \mathcal{E} \rightarrow [\chi(H)] \) of \( H \), and for each \( 1 \leq i \leq s = \chi(H) \) let \( \mathcal{E}_i \) denote the subset of \( \mathcal{E} \) consisting of those elements that receive colour \( i \) under \( c \). By Lemma \([14]\) it suffices to argue that for each \( i \) there is an assignment \( t_i \) of times to elements of \( \mathcal{E} \), such that the maximum reachability of \( (G_i = (V \cup \mathcal{E}_i), \mathcal{E}_i, t_i) \) is at most \( d + 1 \).

We note that, by definition of the edge-class interaction graph and a proper colouring, any pair of incident edges in \( G_i \) must belong to the same element of \( \mathcal{E} \). Thus we see that \( G_i \) is a star forest in which all edges of every connected component belong to the same element of \( \mathcal{E} \). By our assumption on maximum number of edges from any element of \( \mathcal{E} \) that are incident with a single vertex, we see that no component of \( G_i \) contains more than \( d + 1 \) vertices, completing the proof.

Since it is possible to construct a proper 2-colouring of any bipartite graph in linear time, Theorem \([24]\) immediately gives rise to an efficient constant factor approximation algorithm for \textsc{Min-Max Reachability Temporal Ordering}, whenever it is possible to compute the chromatic number of the edge-class interaction graph (or a constant-factor approximation to this value) efficiently, provided that the maximum degree of any edge class is bounded. Note that this second condition will certainly be satisfied if the maximum degree of the input graph is bounded. We obtain the following corollary by recalling that (1) every graph \( H \) admits a proper \((\Delta(H) + 1)\)-colouring, which can be constructed greedily, and (2) it is possible to decide in polynomial time whether a given graph is bipartite.

\begin{corollary}
Let \((G = (V,E),k)\) be an instance of \textsc{Min-Max Reachability Temporal Ordering}, suppose that \( \max_{E' \in \mathcal{E}} \Delta_{E'} \leq d \), where \( \Delta_{E'} \) denotes the maximum degree of the graph \((V(G),E')\), and let \( H \) be the edge-class interaction graph of \( G \). Then we can compute, in polynomial time, an approximation to the optimisation version of \textsc{Min-Max Reachability Temporal Ordering} with approximation ratio \( d \) if \( H \) is bipartite, and approximation ratio \((d + 1)^{\Delta(H)+1}\) otherwise.
\end{corollary}

\section{Conclusions and future work}

We have shown \textsc{Min-Max Reachability Temporal Ordering} is extremely difficult to solve exactly: it remains intractable even in the special case of pairwise disjoint singleton edge-classes, and the two highly restricted cases (DAGs, and trees parameterised by vertex cover number) which are almost trivial in this singleton case become intractable as soon as larger edge-classes are allowed. Given the strength of these hardness results, it is natural to seek approximation algorithms for the optimisation version of the problem, and we show that there exist polynomial-time algorithms to compute constant-factor approximations on graphs of bounded degree either in the singleton case or when the edge-class interaction graph is bipartite. A natural direction for further work is to try to generalise these approximation algorithms and/or improve the approximation factor, as well as to seek further special cases which admit efficient exact algorithms.

In order to better model real-world reordering problems of this form, it would be interesting to consider a generalisation of \textsc{Min-Max Reachability Temporal Ordering} in which each edge-class has a list of permitted times. This would correspond to practical restrictions on when each individual event can be scheduled: for example, a particular event might have to be schedule for a specific day of the week or at a certain time of year (while the Spring Bull Sale can perhaps be rescheduled within a window of a several weeks, it is probably not acceptable to move it to October). We believe that our FPT algorithm for trees (parameterised by the maximum permitted reachability) can be adapted to deal with a
list version in the singleton case, but it is not even clear whether the singleton version can be solved efficiently on DAGs when arbitrary lists of permitted times are allowed. A further generalisation would be to associate different costs with the possible times for each edge class (to model the different costs that might be associated with rescheduling events to different times), and seek to minimise the maximum reachability subject to a given budget constraint.

A number of other variations would also be of practical interest. For certain applications, it might also be relevant to consider the problem of minimising the average cardinality of the reachability set over all vertices of the graph, or indeed the expected size of the reachability set (perhaps given some distribution over starting vertices) in a probabilistic model where each edge has an associated transmission probability. Additionally, in [7], the authors introduce a notion of $(\alpha, \beta)$-reachability, in which the timesteps at which consecutive edges in a temporal path are active must differ by at least $\alpha$ and at most $\beta$; this is a more realistic model for the spread of a disease, as individuals are not instantaneously infectious when infected, and do not remain infectious indefinitely. It would be very interesting to investigate this problem in the reordering context introduced here.

References

1. Eleni C. Akrida, Leszek Gąsieniec, George B. Mertzios, and Paul G. Spirakis. The complexity of optimal design of temporally connected graphs. *Theory of Computing Systems*, 61(3):907–944, Oct 2017. URL: [https://doi.org/10.1007/s00224-017-9757-x](https://doi.org/10.1007/s00224-017-9757-x).
2. T. N. Bui, S. Chaudhuri, F. T. Leighton, and M. Sipser. Graph bisection algorithms with good average case behavior. *Combinatorica*, 7(2):171–191, Jun 1987. URL: [https://doi.org/10.1007/BF02579448](https://doi.org/10.1007/BF02579448).
3. M. Cygan, F.V. Fomin, L. Kowalik, D. Lokshtanov, D. Marx, M. Pilipczuk, M. Pilipczuk, and S. Saurabh. *Parameterized Algorithms*. Springer International Publishing, 2015.
4. Rod G. Downey and Michael R. Fellows. Fixed-parameter tractability and completeness ii: On completeness for w[1]. *Theoretical Computer Science*, 141(1):109 – 131, 1995. URL: [http://www.sciencedirect.com/science/article/pii/0304397594000973](http://www.sciencedirect.com/science/article/pii/0304397594000973).
5. Pál Gronás Drange, Markus Sortland Dregi, and Pim van ’t Hof. On the computational complexity of vertex integrity and component order connectivity. In Hee-Kap Ahn and Chan-Su Shin, editors, *Algorithms and Computation*, volume 8889 of *Lecture Notes in Computer Science*, pages 285–297. Springer International Publishing, 2014. doi:10.1007/978-3-319-13075-0_23.
6. Jessica Enright and Kitty Meeks. Deleting edges to restrict the size of an epidemic: A new application for treewidth. *Algorithmica*, 80(6):1857–1889, Jun 2018. URL: [https://doi.org/10.1007/s00453-017-0311-7](https://doi.org/10.1007/s00453-017-0311-7).
7. Jessica Enright, Kitty Meeks, George B. Mertzios, and Viktor Zamaraev. Deleting edges to restrict the size of an epidemic in temporal networks. arXiv:1805.06836 [cs.DS]; accepted to MFCS 2019.
8. Till Fluschnik, Hendrik Molter, Rolf Niedermeier, and Philipp Zschoche. Temporal graph classes: A view through temporal separators. In *44th International Workshop on Graph-Theoretic Concepts in Computer Science (WG ’18)*, pages 216–227. Springer International Publishing, 2018.
9. Daniel Gross, Monika Heinig, Lakshmi Iswara, L. William Kazmierczak, Kristi Luttrel, John T. Saccoman, and Charles Suffel. A survey of component order connectivity models of graph theoretic networks. *SWEAS Trans. Math.*, 12(9), 2013.
10. A. Himmel, H. Molter, R. Niedermeier, and M. Sorge. Adapting the Bron–Kerbosch algorithm for enumerating maximal cliques in temporal graphs. *Social Network Analysis and Mining*, 7(1):35, Jul 2017. doi:10.1007/s13278-017-0455-0.
Assigning times to minimise reachability in temporal graphs

11. P. Holme and J. Saramäki. Temporal networks. *Physics Reports*, 519(3):97 – 125, 2012. doi:10.1016/j.physrep.2012.03.001

12. R.M. Karp. *Reducibility among Combinatorial Problems*, pages 85–103. Springer US, Boston, MA, 1972. doi:10.1007/978-1-4684-2001-2_9

13. D. Kempe, J. Kleinberg, and A. Kumar. Connectivity and inference problems for temporal networks. *Journal of Computer and System Sciences*, 64(4):820 – 842, 2002. doi:10.1006/jcss.2002.1829

14. Angsheng Li and Linqing Tang. The complexity and approximability of minimum contamination problems. In Mitsunori Ogihara and Jun Tarui, editors, *Theory and Applications of Models of Computation*, volume 6648 of *Lecture Notes in Computer Science*, pages 298–307. Springer Berlin Heidelberg, 2011. doi:10.1007/978-3-642-20877-5_30

15. Q. Liang and E. Modiano. Survivability in time-varying networks. *IEEE Transactions on Mobile Computing*, 16(9):2668–2681, Sept 2017. doi:10.1109/TMC.2016.2636152

16. G.B. Mertzios, O. Michail, I. Chatzigiannakis, and P.G. Spirakis. Temporal network optimization subject to connectivity constraints. In F.V. Fomin, R. Freivalds, M. Kwiatkowska, and D. Peleg, editors, *Automata, Languages, and Programming*, pages 657–668, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

17. O. Michail. An introduction to temporal graphs: An algorithmic perspective. *Internet Mathematics*, 12(4):239–280, 2016. doi:10.1080/15427951.2016.1177801

18. Jayadev Misra and David Gries. A constructive proof of vizing’s theorem. *Inf. Process. Lett.*, 41(3):131–133, 1992. URL: https://doi.org/10.1016/0020-0190(92)90041-S doi:10.1016/0020-0190(92)90041-S
## A List version of the problem

In applications, it seems natural that we might want to restrict the possible timesteps at which each edge (class) is active; this gives rise to a list version of our problem.

### List Min-Max Reachability Temporal Ordering

**Input:** A graph $G = (V, E)$, a list $E = \{E_1, \ldots, E_h\}$ of subsets of $E$, subsets $L_1, \ldots, L_h$ of $N$, and a positive integer $k$

**Question:** Is there a bijective function $t : E \rightarrow \mathbb{N}$ such that $t(E_i) \in L_i$ for each $i$ and the maximum temporal reachability of $(G, E, t)$ is at most $k$?

We now show that this version is extremely hard: even restricting the edge-class interaction graph to be a star does not help.

**Theorem 26.** List Min-Max Reachability Temporal Ordering is NP-complete, even if the edge-class interaction graph of the input is a star, and the underlying graph $G$ is a disjoint union of five-vertex paths.

**Proof.** We give a reduction from the NP-complete problem Vertex Cover. Suppose that $(G, k)$ is the input to our instance of Vertex Cover. We construct an instance $(H, E = \{E_1, \ldots, E_h\}, \{L_1, \ldots, L_h\}, t)$ of List Min-Max Reachability Temporal Ordering which is a yes-instance if and only if $(G, k)$ is a yes-instance for Vertex Cover.

$H$ consists of $|E(G)|$ disjoint paths on 5 vertices, which are in one-to-one correspondence with the edges of $G$; we refer to the path corresponding to $e \in E(G)$ as $P_e$. If $e = uv$, we refer to the endpoints of $P_e$ as $u[e]$ and $v[e]$, and their unique neighbours on $P_e$ as $u'[e]$ and $v'[e]$ respectively. We call the edges of $P_e$ that are not incident with either $u[e]$ or $v[e]$ the middle edges of $P_e$, and we refer to the midpoint of $P_e$ (the vertex not adjacent to either $u[e]$ or $v[e]$) as $x_e$.

Suppose that $V(G) = \{v_1, \ldots, v_n\}$. We set $h = n + 1$, for each $1 \leq i \leq n$ we set $E_i = \{v_i[e]v_i'[e] : e \in E(G)\}$, and we set $E_{n+1}$ to be the set of all edges that are middle edges of some path. For $1 \leq i \leq n$, we set $L_i = [n+1] \setminus \{k+1\}$, and we set $L_{n+1} = \{k+1\}$. We complete the construction of our instance of List Min-Max Reachability Temporal Ordering by setting $\ell = 4$.

Suppose first that $G$ contains a vertex cover $X \subset V(G)$ with $|X| = k$. Let $t : E \rightarrow [n+1]$ be any function such that $t(E_i) \leq k$ for all $v_i \in X$, and $t(E_{n+1}) = k$. We claim that the maximum temporal reachability of $(H, E, t)$ is at most 4. First observe that, for any $t$, the maximum temporal reachability of $(H, E, t)$ cannot be more than 5, since every connected component of $H$ contains exactly 5 vertices. Moreover, if any vertex has reachability set of size 5, it must be $x_e$ for some edge $e \in E(G)$: as both edges incident with $x_e$ occur simultaneously, there is no strict temporal path between the two connected components of $P_e \setminus \{x_e\}$, so no vertex other than $x_e$ can possibly reach all other vertices in its component. Suppose now that reach$_{H, E, t}(x_e) = 5$ for some $e = v_iv_j \in E(G)$. In this case, we must have $t(E_i), t(E_j) > k + 1$; by construction of $t$, this only happens if neither $v_i$ nor $v_j$ belongs to $X$. However, as $v_iv_j$ is an edge, this contradicts the assumption that $U$ is a vertex cover. Hence the maximum reachability of $(H, E, t)$ is at most 4, as required.

Conversely, suppose that we have a function $t : E \rightarrow [n+1]$, with $t(E_i) \in L_i$ for all $i$, such that the maximum reachability of $(H, E, t)$ is at most 4. We claim that $G$ has a vertex cover of size at most $k$. Set $X = \{v_i \in V(G) : t(E_i) \leq k\}$. It is clear that $X \subset V(G)$ and $|X| = k$; it remains to demonstrate that $X$ is a vertex cover. To do this, fix an arbitrary edge
Assigning times to minimise reachability in temporal graphs

If neither \( v_i \) nor \( v_j \) belongs to \( X \), we would have \( \text{reach}_{H,E,t}(x_v) = V(P_e) \), since both middle edges of \( P_e \) are active strictly before either of the remaining two edges. Thus the maximum reachability of \((H,E,t)\) would be at least 5, giving a contradiction. As our choice of \( e \) was arbitrary, we may conclude that \( X \) is indeed a vertex cover for \( G \). ▶

**B A maximisation version**

In some settings, we might wish to maximise rather than minimise the reach of a spreading process on a temporal graph: consider, for example, the case of viral marketing. With this in mind, it is natural to define the following maximisation version of our problem.

Max-Min Reachability Temporal Ordering

**Input:** A graph \( G = (V,E) \), a multiset \( E = \{E_1,\ldots,E_h\} \) of subsets of \( E \), and a positive integer \( k \)

**Question:** Is there a bijective function \( t : E \rightarrow [h] \) such that minimum temporal reachability of \((G,E,t)\) is at least \( k \)?

Singleton and list variants are defined in the same way as for Min-Max Reachability Temporal Ordering.

We note that a number of our results on Min-Max Reachability Temporal Ordering can be extended naturally to Max-Min Reachability Temporal Ordering; we state and prove these results here.

**Theorem 27.** Let \((G,E,k)\) be the input to an instance of Max-Min Reachability Temporal Ordering, where \( E \) is a set of pairwise disjoint singleton sets, and \( G = (V,E) \) is a DAG. Then \((G,E,k)\) is a yes-instance if and only if the maximum reachability of the static graph \( G \) is at least \( k \).

**Proof.** It is clear that the maximum reachability of the temporal graph can never exceed that of the static graph, so it remains to show that there is some function \( t \) such that these two quantities are equal. As in the proof of Theorem 10 we fix an ordering \( e_1,\ldots,e_m \) of the edges of \( G \) which corresponds to a topological ordering of the vertices. If \( E_i = \{e_i\} \) for each \( i \), we now define \( t(e_i) = i \) (so we have reversed the ordering from Theorem 10). Now consider an arbitrary pair of vertices \( u \) and \( v \), such that \( v \) is in the static reachability set of \( u \); it remains to show that \( v \) is in \( \text{reach}_{G,E,t}(u) \). Since \( v \) is in the static reachability set of \( u \), there is a directed path \( u = w_0, w_1,\ldots,w_r = v \) in \( G \) such that \( \bar{w}_{i-1}w_i \in E(G) \) for \( 1 \leq i \leq r \). It follows from the definition of a topological ordering that \( u = w_0 \prec w_1 \prec \cdots \prec w_r = v \) in such an ordering; the construction of our function \( t \) therefore implies that \( t(\bar{w}_{i-1}w_i) < t(\bar{w}_iw_{i+1}) \) for \( 1 \leq i \leq k-1 \), so we have a strict temporal path from \( u \) to \( v \) in \((G,E,t)\), as required. ▶

**Theorem 28.** Max-Min Reachability Temporal Ordering is NP-complete even if \( G \) is a tree obtained from a path by adding additional leaf vertices adjacent to its endpoints.

**Proof.** We give a reduction from the following problem, shown to be NP-complete in [12].
Let \((G, k)\) be the input to our instance of Vertex Cover, and suppose that \(V(G) = \{v_1, \ldots, v_m\}\) and \(E(G) = \{e_1, \ldots, e_m\}\). We will construct an instance \((G', \mathcal{E}', k')\) of Max-Min Reachability Temporal Ordering which is a yes-instance if and only if \((G, k)\) is a yes-instance for Vertex Cover.

Let \(P\) be a path on \(n - k\) vertices; we will refer to the endpoints of \(P\) as \(s\) and \(r\). We obtain \(G'\) from \(P\) by adding a set \(W = \{w_1, \ldots, w_m\}\) of \(m\) new leaves adjacent to \(r\), and a set \(U = \{u_1, \ldots, u_{m+1}\}\) of \(m + 1\) new leaves adjacent to \(s\). We set \(\mathcal{E} = \{E_1, \ldots, E_n\}\) where, for each \(i\),

\[
E_i = E(P) \cup \{su_j : 1 \leq j \leq m + 1\} \cup \{rw_j : v_i \text{ is incident with } e_j\}.
\]

Finally, we set \(k' = n + m - k + 1\).

Observe that, for every \(v \in V(G') \setminus W\) and any bijection \(t : \mathcal{E} \to [n]\), we have \(|\text{reach}_{G', \mathcal{E}, t}(v)| \geq n + m - k + 1\): since all edges not incident with some element of \(W\) are active at all timesteps, and the distance between any two vertices not in \(W\) is at most \(n - k\), we have \(|\text{reach}_{G', \mathcal{E}, t}(v)| \geq |V(P) \cup W|\), so \(|\text{reach}_{G', \mathcal{E}, t}(v)| \geq n - k + m + 1\). Thus we have a yes-instance to Max-Min Reachability Temporal Ordering if and only if there is a bijection \(t : \mathcal{E} \to n\) such that \(|\text{reach}_{G', \mathcal{E}, t}(v)| \geq n + m - k + 1\) for all \(v \in W\).

Suppose first that \(G\) contains a vertex-cover \(X\) of size at \(k\). We fix any function \(t : \mathcal{E} \to [n]\) with the property that \(t(E_i) \leq k\) whenever \(v_i \in X\). Fix an arbitrary \(w \in W\), and suppose that \(w = v_jv_j\). Since \(U\) is a vertex cover for \(G\), we must have that at least one of \(t(E_i)\) and \(t(E_j)\) is at most \(k\); thus the edge \(wr\) is active during at least one of the first \(k\) timesteps. Since the distance from \(r\) to all vertices in \(V(P) \cup U\) is at most \(n - k\), and all edges along the paths from \(r\) to such vertices are active at all timesteps, it follows that every vertex in \(V(P) \cup U\) is in the reachability set of \(w\). Hence \(|\text{reach}_{G', \mathcal{E}, t}(w)| \geq 1 + n - k + m + 1 > n + m - k + 1\), as required.

Conversely, suppose that we have a bijection \(t : \mathcal{E} \to [n]\) such that \(|\text{reach}_{G', \mathcal{E}, t}(w)| \geq n + m - k + 1\) for every \(w \in W\). For each \(w \in W\), write \(t_0(w)\) for the first timestep at which the edge \(wr\) is active, and fix \(w_j\) such that \(t_0(w)\) is as large as possible. Note that \(|W \cup V(P)| = m + n - k\), so as \(|\text{reach}_{G', \mathcal{E}, t}(w_j)| \geq n + m - k + 1\) there must be some vertex \(u \in \text{reach}_{G', \mathcal{E}, t}(w_j) \cap U\). As there are \(n - k\) edges on the path from \(r\) to \(u\), and there must be some sequence \(t_0(w_j) \leq t_1 < \cdots < t_{n-k}\) such that the \(i\)th edge on the path is active at time \(t_i\), we must have \(t_0(w_j) \leq n - (n - k) = k\). Now set \(X = \{v_i : t(E_i) \leq k\}\). It is clear that \(X \subseteq V(G)\) and, by bijectivity of \(t\), \(|X| = k\). To show that \(X\) is a vertex cover for \(G\), we consider an arbitrary edge \(e_t = vqvq\). By choice of \(w_j\), we know that \(t_0(w_j) \leq t_0(w_j) \leq k\), so one \(wjr\) is active at some timestep less than or equal to \(k\); it follows that at least one of \(t(E_q)\) and \(t(E_q)\) is at most \(k\), so \(\{v_p, v_q\} \cap X \neq \emptyset\). Thus \(X\) is indeed a vertex cover for \(G\). □