



HAL
open science

Temporally connected components

Stefan Balev, Yoann Pigné, Eric Sanlaville, Jason Schoeters

► **To cite this version:**

Stefan Balev, Yoann Pigné, Eric Sanlaville, Jason Schoeters. Temporally connected components. 2023.
hal-03966327

HAL Id: hal-03966327

<https://hal.science/hal-03966327v1>

Preprint submitted on 31 Jan 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

1 Temporally connected components

2 **Stefan Balev** ✉

3 Normandie Univ, UNIHAVRE, LITIS, 76600 Le Havre, France

4 **Yoann Pigné** ✉ 🏠

5 Normandie Univ, UNIHAVRE, LITIS, 76600 Le Havre, France

6 **Eric Sanlaville** ✉ 🏠

7 Normandie Univ, UNIHAVRE, LITIS, 76600 Le Havre, France

8 **Jason Schoeters** ✉ 🏠

9 University of Cambridge, United Kingdom

10 — Abstract —

11 We discuss a variety of extensions of connected components in temporal graphs, focusing on extensions
12 using connectivity over time through temporal paths (or journeys). Starting with components induced
13 by temporal sources or sinks, we build up to components induced by multiple sources or sinks,
14 and eventually components where all vertices are sources and sinks, *i.e.* temporally connected
15 components. Our contributions mainly include structural results on the number of components, and
16 algorithmic and complexity results of corresponding decision problems.

17 **2012 ACM Subject Classification** Theory of computation → Problems, reductions and completeness;
18 Networks → Network components; Mathematics of computing → Graph algorithms; Mathematics
19 of computing → Paths and connectivity problems

20 **Keywords and phrases** temporal graph theory, NP-complete problem, polynomial-time reduction,
21 algorithm, connected component, temporal connectivity

22 **Funding** *Eric Sanlaville*: DyNet RIN Tremplin Région Normandie 2020-2022

23 *Jason Schoeters*: DyNet RIN Tremplin Région Normandie 2020-2022

24 **1** Introduction

25 Temporal graphs have become increasingly more popular in the literature over the years,
26 and with good reason. Dynamic settings, whether failure-prone systems or highly mobile
27 entities, which static graphs fail to model, can be modeled naturally with temporal graphs
28 [9, 24, 48]. Many problems in temporal graphs can be solved by extending static graph
29 structures and related problems and algorithms into the dimension of time, resulting in
30 various more complicated extensions. The structures considered in this paper use journeys
31 (also called temporal paths) which allow for connectivity over time through increasing time
32 labels on successive edges of the path [12, 25, 28, 33, 35]. Concerning the problems, often these
33 become harder in terms of time and/or space complexity [1, 4, 40]. Naturally, this leads to
34 results considering specific classes of temporal graphs, often restraining the underlying graph
35 (or footprint), approximation results, and/or fixed-parameter tractability results (or, in the
36 negative case, $W[1]$ or even $W[2]$ hardness results) [11, 15, 16, 19, 29, 50]. Temporal graph
37 theory has also natural links with gossip theory [3, 21, 22], and with rainbow structures in
38 edge-colored static graphs [7, 13, 38].

39 The aim of this paper is to present a comprehensive study of the fundamental question
 40 “What is a connected component in temporal graphs?” Some works have appeared over the
 41 years offering answers to this question, in one way or another. Here, we wish to regroup
 42 these results, not unlike a survey, and add or complete missing pieces of information relevant
 43 to this central question. We choose to consider a well-established hierarchy of temporal
 44 connectivity properties proposed by Casteigts *et al.* in [9], and recently revisited in [8], as
 45 a basis for all sorts of connectivity which may arise or exist in temporal graphs. For some
 46 of these properties defined through journeys, we define the corresponding component and
 47 present results on computational complexity, algorithms, and the number of components.

48 In Section 2, we present the main concepts regarding temporal graphs, different types of
 49 connectivity such as journeys, and components in such graphs. In Section 3, we state results
 50 regarding temporal components in which one vertex connects to/from all others. Then, in
 51 Section 4, we present results regarding temporal components in which all vertices connect
 52 to/from all others. Finally, in Section 5, we conclude by giving a summary of this paper and
 53 discuss some open questions and future works.

54 Note that as the paper presents a number of new results, for ease of reading, most of the
 55 technical proofs are moved to the appendices. The concerned results are marked with a \star .

56 **2 Main concepts of temporal graph connectivity**

57 **2.1 Temporal graphs**

58 We use standard notation and terminology from graph theory [17]. Temporal graphs can be
 59 defined in a number of (more or less) equivalent ways, and as a result exist under a multitude
 60 of distinct names, including dynamic graphs, stream graphs, link streams, time-varying
 61 graphs and evolving graphs [2, 9, 18, 39]. We choose the following, often referred to as the
 62 compact representation.

63 A temporal graph $\mathcal{G} = (G, \lambda)$ is defined by a static graph $G = (V, E)$ with vertex set $V(G)$
 64 and edge set $E(G) \subseteq V(G) \times V(G)$, and a labelling of the edges $\lambda : E \rightarrow 2^{\mathbb{N}} \setminus \emptyset$ determining
 65 at which discrete times edges are present. G is commonly called the underlying graph (or
 66 footprint) of \mathcal{G} , sometimes denoted as $G_{\downarrow}(\mathcal{G})$ or simply G_{\downarrow} . In this paper, we consider
 67 only finite-time temporal graphs, with the largest label, or lifetime, of \mathcal{G} being denoted
 68 as $T(\mathcal{G})$. When there is no ambiguity, we denote $V(G)$, $E(G)$ and $T(\mathcal{G})$ as simply V , E
 69 and T respectively, and $n = |V|$ and $m = |E|$. We use the notation $G_t(\mathcal{G})$ (or simply G_t)
 70 for the static graph corresponding to temporal graph \mathcal{G} at time t , *i.e.* $G_t = (V, E_t \subseteq E)$
 71 with $e \in E_t \iff t \in \lambda(e)$. G_t is called a snapshot of \mathcal{G} . It is sometimes useful to
 72 represent temporal graphs as sequences of snapshots, *e.g.* $\mathcal{G} = (G_1, G_2, \dots, G_T)$. Let the
 73 restriction of \mathcal{G} on time window $[t_1, t_2]$ for some $t_1 \leq t_2 \leq T$ be the temporal graph denoted
 74 $\mathcal{G}_{[t_1, t_2]} = (G_{t_1}, G_{t_1+1}, \dots, G_{t_2})$, and $G_{[t_1, t_2]}$ denote its footprint. Finally, let the subgraph of
 75 \mathcal{G} induced by vertex set $V' \subseteq V$ be the temporal graph denoted $\mathcal{G}[V'] = (G' = (V', E'), \lambda)$,
 76 with $\{u, v\} \in E' \iff u, v \in V'$ and $\{u, v\} \in E$.

77 **Size of a temporal graph**

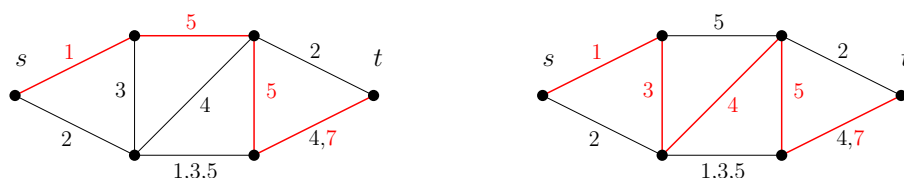
78 Whether one considers the compact representation, or the sequence of snapshots represent-
 79 ation, in both cases the size of a temporal graph is $O(n + mT)$. In specific cases, more
 80 resourceful representations exist, such as to store only the differences from a snapshot to
 81 another (added and removed edges), which is specially effective for temporal graphs with few

82 changes between snapshots, or low dynamics. In this paper however, we do not specifically
 83 consider temporal graphs with low dynamics. Without loss of generality, we assume $m \geq n$.¹
 84 In this setting, an algorithm on temporal graphs is polynomial if and only if it runs in
 85 polynomial time regarding n, m , and T .

86 2.2 Journeys and temporal connectivity

87 In static graphs, including snapshots of temporal graphs, vertices s, t are adjacent if they are
 88 connected by edge $\{s, t\}$, and edges $\{u, v\}, \{v, w\}$ are said to be incident. An (elementary)
 89 path is defined as a (non-repeating) sequence of adjacent vertices. We denote a path from
 90 u to v as $u \rightarrow v$ or, since we consider undirected graphs, $u \leftrightarrow v$. If for all pairs of vertices
 91 in G there exists a path including both vertices, then G is said to be connected. If G isn't
 92 connected, then G can be partitioned into subgraphs such that each subgraph is connected.
 93 Such subgraphs, and equivalently the associated vertex sets, are referred to as the connected
 94 components of G .

95 In temporal graphs, a natural extension of paths exists, in which even though two vertices
 96 may admit no path in any of the snapshots, over time they might still be connected thanks
 97 to some structures called temporal paths or journeys. More formally, a journey from s to t
 98 in \mathcal{G} is a path from s to t in the footprint G with non-decreasing labels on successive edges
 99 (see Figure 1). We then say s can reach t and t can be reached by s . We denote a journey,
 100 or the existence of a journey, from u to v , as $u \rightsquigarrow v$. Journeys, as opposed to paths, are
 101 neither symmetrical nor transitive, *i.e.* $u \rightsquigarrow v$ and $v \rightsquigarrow w$ does not necessarily imply $u \rightsquigarrow w$.
 102 The notation $u \rightsquigarrow v \rightsquigarrow w$ implies that journey $v \rightsquigarrow w$ takes place later in time than journey
 103 $u \rightsquigarrow v$ (which also implies $u \rightsquigarrow w$). When a temporal graph \mathcal{G} admits journeys from all
 104 vertices to all other vertices, the graph is said to be temporally connected, or \mathcal{TC} . We use
 105 the notation $u \circlearrowleft v$ for a round-trip journey $u \rightsquigarrow v \rightsquigarrow u$



■ **Figure 1** Example of temporal graph \mathcal{G}_1 (presented in the compact representation), with two journeys (in red) but no static path from vertex s to vertex t in any of the snapshots.

106 Besides the footprint, another useful structure concerning temporal graphs is the transitive
 107 closure, which represents the reachability (through journeys) of all vertices of the graph.
 108 Formally, the transitive closure of a temporal graph \mathcal{G} corresponds to the directed graph
 109 $H(\mathcal{G}) = (V, A \subseteq V \times V)$, with $A = \{(v, v') : v \rightsquigarrow v'\}$. Note that this definition introduced by
 110 [6] uses the notion of closure in a temporal sense: as stated previously, $u \rightsquigarrow v$ and $v \rightsquigarrow w$
 111 implies $u \rightsquigarrow w$ only if the second journey takes place later than the first one. So $H(\mathcal{G})$ is not,
 112 as usual, the transitive closure of some relation, for instance the relation associated to the
 113 existence of journeys between vertex couples, but precisely the graph of this relation. When
 114 clear from the context, we simply denote $H(\mathcal{G})$ as H .

¹ If for some temporal graph \mathcal{G} we have $m < n$, *i.e.* the footprint admits multiple connected components V' , precomputation allows us to reduce to $O(n)$ cases of temporal graphs $\mathcal{G}' = \mathcal{G}[V']$ where $m' \geq n'$.

4 Temporally connected components

115 For example, the transitive closure of the temporal graph \mathcal{G}_1 from Figure 1 corresponds to a
116 complete bidirectional graph (all arcs in both directions exist) except for arc (t, s) which is
117 missing, meaning the only journey missing for \mathcal{G}_1 to be temporally connected is from t to s .

118 Constructing the transitive closure can be done in time $O(n(m \log T + n \log n))$ through n
119 calls of an adaptation of Dijkstra’s algorithm [46], or in time $O(\max(|E_i|)nT)$ through an
120 online algorithm gradually building the transitive closure snapshot by snapshot [5]. The latter
121 is more efficient in specific cases when the lifetime $T = O(n)$ and when the snapshot density
122 $\max(|E_i|) = o(m)$. For simplicity, we will consider the former time complexity throughout this
123 paper, and we will refer to the adaptation of Dijkstra’s algorithm as the temporal Dijkstra
124 algorithm.

125 2.3 Hierarchy of connectivity properties for temporal graphs

126 In [9], and recently revisited in [8], Casteigts *et al.* introduce a hierarchy of temporal
127 properties, including among others temporal connectivity, \mathcal{TC} . The properties are given
128 mnemonic names using key concepts *e.g.* \mathcal{J} for “journey”, and \mathcal{TC} for “temporal connectivity”.
129 Superscript adds restraints, for example \mathcal{B} meaning “in each bounded time window”, and
130 $\forall 1$ meaning “from all vertices to one”. This gives rise to properties such as $\mathcal{J}^{1\forall}$, having a
131 vertex which is connected through journeys to all other vertices, and $\mathcal{TC}^{\mathcal{B}}$, each window of
132 some given size Δ^2 being temporally connected. The properties of interest for this paper
133 are formally defined in Sections 3 and 4 when we define their corresponding components.
134 For clarity, we’ve opted to change some mnemonic names of the hierarchy for shorter ones
135 without superscript (so we may add superscript later on). \mathcal{S} for “source” will replace $\mathcal{J}^{1\forall}$,
136 and since “sink” unfortunately also starts with an S, \mathcal{T} for “target” will replace $\mathcal{J}^{\forall 1}$. By
137 slight abuse of notation, we will often use the mnemonic names to denote the property as
138 well as the set of graphs admitting the property.

139 2.4 From connectivity properties to temporal components

140 The rest of the paper is organized to define and study, for every journey-based temporal
141 property \mathcal{X} , the analogue of a connected component corresponding to this property, which
142 we will simply refer to as a \mathcal{X} component.

143 **► Definition 1 (\mathcal{X} component).** *Given a temporal graph \mathcal{G} , a \mathcal{X} component is a maximal*
144 *subset $V' \subseteq V$ such that \mathcal{X} is respected by V' in \mathcal{G} .*

145 Note the maximality requirement for V' , which mimics the maximality requirement for static
146 connected components. It is natural to look for largest components also in the temporal case.
147 Theorem 1 can be ambiguous concerning the latter part, since technically \mathcal{X} is a property
148 of temporal graphs, not of a set of vertices in a temporal graph. For now, it is sufficient
149 to say that we simply aim to present the difference between \mathcal{X} components and closed \mathcal{X}
150 components, Theorem 1 and Theorem 2 *resp.* Also, these definitions are formal and clear
151 when presented using concrete properties in Sections 3 and 4.

152 Using terminology from [6, 23, 44], we say a \mathcal{X} component V' is closed if journeys between
153 vertices $u, v \in V'$ necessary for property \mathcal{X} , do not use vertices from $V \setminus V'$. In other words,

² In this paper, Δ will by default be used to denote the size of a time window, and $\Delta(G)$ for the maximum degree of a static graph G .

154 to verify if V' is a closed \mathcal{X} component, it suffices to verify the \mathcal{X} property on $\mathcal{G}[V']$, and
 155 this verification does not depend on the rest of \mathcal{G} .

156 ► **Definition 2** (Closed \mathcal{X} component). *Given a temporal graph \mathcal{G} , a closed \mathcal{X} component is*
 157 *a maximal subset $V' \subseteq V$ such that \mathcal{X} is respected by $\mathcal{G}[V']$.*

158 Open \mathcal{X} components are defined as \mathcal{X} components in which there exists at least one journey
 159 necessary for \mathcal{X} which goes outside of the component. Most of the components studied in
 160 static graphs are closed (an exception being k -connected components [45] which may be
 161 open). We give the following results for \mathcal{X} components, and if applicable, also for closed \mathcal{X}
 162 components.

163 For each property \mathcal{X} , we start by studying the worst-case number of \mathcal{X} components, *i.e.* the
 164 maximum number of \mathcal{X} components which may exist in a given temporal graph. This can be
 165 useful for enumeration and partition problems.

166 For each property \mathcal{X} , the corresponding decision problem \mathcal{X} COMPONENT is defined as
 167 follows.

168 ► **Definition 3** (\mathcal{X} COMPONENT decision problem).

169 **Input:** temporal graph \mathcal{G} , integer k (and integer Δ if \mathcal{X} is a windowed property).

170 **Question:** does \mathcal{G} admit a \mathcal{X} component of size at least k ?

171 Algorithms and complexity results for \mathcal{X} COMPONENT are presented. As stated in Section 1,
 172 depending on property \mathcal{X} , some results may already exist in the literature. However we
 173 additionally determine the boundary (if one exists) between polynomial-time solvability and
 174 NP -hardness depending on the lifetime of the graph. To obtain such results, we modify
 175 reductions from the literature or use different reductions altogether.

176 Regarding hardness implications, Bhadra *et al.* [6] give the following argument for \mathcal{TC}
 177 components and closed \mathcal{TC} components, which we generalize for any temporal properties
 178 $\mathcal{X}_1, \mathcal{X}_2$. Although \mathcal{X}_1 components are a special case of \mathcal{X}_2 components, the NP -hardness
 179 of \mathcal{X}_1 COMPONENT does not directly imply that \mathcal{X}_2 COMPONENT is NP -hard as well. This
 180 is because a possible polynomial time algorithm for \mathcal{X}_2 COMPONENT need only answer the
 181 decision problem and not identify the components of size at least k , thus potentially making it
 182 difficult to verify if at least one such a component is a \mathcal{X}_1 component. Also, the same temporal
 183 graph may contain both a \mathcal{X}_1 component (of indeterminate size) and a \mathcal{X}_2 component of size
 184 k , so the decision problem for the latter would always return “yes”, ignoring the presence
 185 or absence of a \mathcal{X}_1 component of size k , thereby leaving its decision problem unsolved. Of
 186 course, the other way around, since \mathcal{X}_1 components are a special case of \mathcal{X}_2 components, if
 187 \mathcal{X}_2 COMPONENT is NP -hard, then \mathcal{X}_1 COMPONENT is not necessarily NP -hard either. Since
 188 hardness results do not transfer one way or the other, note that this implies that neither do
 189 results on polynomial-time solvability.

190 Regarding hardness proofs, *i.e.* reductions, we do however use a trick from [6] making one
 191 reduction work for both \mathcal{X} COMPONENT and CLOSED \mathcal{X} COMPONENT, which is to make
 192 sure all \mathcal{X} components in the transformed instance are closed.

193 **3 One to/from all** ($\mathcal{S}, \mathcal{S}^B, \mathcal{S}^D, \mathcal{T}, \mathcal{T}^B, \mathcal{T}^D$)

194 A temporal source is a vertex u such that all other vertices v in the temporal graph admit
 195 a journey $u \rightsquigarrow v$. The concept is mainly useful to model a network over which one agent,

6 Temporally connected components

196 say the leader, can control the entire network, diffusing information, messages or influence
 197 throughout the network using peer-to-peer broadcasting [14, 31, 49]. The property of having
 198 a temporal source is denoted by \mathcal{S} .

199 ► **Definition 4** (\mathcal{S} component). *An \mathcal{S} component of a temporal graph \mathcal{G} is a maximal subset
 200 $V' \subseteq V$ such that $\exists u \in V', \forall v \in V', u \rightsquigarrow v$ in \mathcal{G} .*

201 Adding the natural constraint of time windows has mainly two advantages, first a time bound
 202 after which one is ensured all other vertices can be reached from the source, and second
 203 the possibility for the source to reach other vertices multiple times over the lifetime of the
 204 network. The property of having a vertex which is a temporal source for each time window
 205 is denoted by \mathcal{S}^B .

206 ► **Definition 5** (\mathcal{S}^B component). *An \mathcal{S}^B component of duration Δ of a temporal graph \mathcal{G} is
 207 a maximal subset $V' \subseteq V$ such that $\exists u \in V', \forall t \leq T - \Delta + 1, \forall v \in V', u \rightsquigarrow v$ in $\mathcal{G}_{[t, t+\Delta-1]}$.*

208 A natural relaxation of the latter allows for any vertex to be the source in a time window,
 209 not necessarily the same vertex for all time windows. In other words, the source may be
 210 dynamic and change over time. This property is denoted by \mathcal{S}^D .

211 ► **Definition 6** (\mathcal{S}^D component). *An \mathcal{S}^D component of duration Δ of a temporal graph \mathcal{G} is
 212 a maximal subset $V' \subseteq V$ such that $\forall t \leq T - \Delta + 1, \exists u \in V', \forall v \in V', u \rightsquigarrow v$ in $\mathcal{G}_{[t, t+\Delta-1]}$.*

213 Closely related to source vertices are sink vertices, where other vertices are able to reach
 214 such a vertex, leading to properties \mathcal{T} , \mathcal{T}^B , and \mathcal{T}^D . While often studied independently and
 215 having its own specific applications such as collecting and analysing data (e.g. [30, 34, 47]),
 216 we simplify this paper through the following observation.

217 ► **Lemma 7.** *\mathcal{T} (resp. \mathcal{S} , \mathcal{T}^B , \mathcal{S}^B , \mathcal{T}^D , \mathcal{S}^D) components in temporal graph $\mathcal{G} = (G, \lambda)$
 218 of lifetime T correspond to \mathcal{S} (resp. \mathcal{T} , \mathcal{S}^B , \mathcal{T}^B , \mathcal{S}^D , \mathcal{T}^D) components in temporal graph
 219 $\mathcal{G}' = (G, \lambda')$ of lifetime T , where $\forall e \in E(G), \lambda'(e) = \bigcup_{\ell \in \lambda(e)} T - \ell$.*

220 **Proof.** Any journey in \mathcal{G} from vertex u to v is reversed in \mathcal{G}' . Thus, a vertex able to reach
 221 all vertices in \mathcal{G} (or in some $\mathcal{G}_{[t, t+\Delta-1]}$) can be reached by all vertices in \mathcal{G}' (or in some
 222 $\mathcal{G}'_{[t', t'+\Delta-1]}$) and vice versa. ◀

223 Theorem 7 allows us to focus only on \mathcal{S} components, since any structural result transfers to
 224 \mathcal{T} components, and algorithmic results transfer too, with a polynomial overhead of $O(mT)$
 225 to build the reversed graph (but they can be adapted easily without any overhead). The
 226 same holds for \mathcal{S}^B components and \mathcal{T}^B components, as well as for \mathcal{S}^D components and \mathcal{T}^D
 227 components.

228 3.1 \mathcal{S} components

229 ► **Lemma 8.** *\mathcal{S} components are necessarily closed.*

230 **Proof.** In an \mathcal{S} component, all vertices on a journey from the source to some other vertex
 231 also admit a journey from the source, and are thus by maximality included in the component
 232 as well. ◀

233 ► **Lemma 9.** *A vertex can only be a source for one \mathcal{S} component.*

234 **Proof.** If a vertex was a source for two distinct \mathcal{S} components, then at least one wouldn't be
 235 maximal since their union would result in a larger \mathcal{S} component. ◀

236 ▶ **Observation 10.** *An isolated vertex in the footprint is an \mathcal{S} component.*

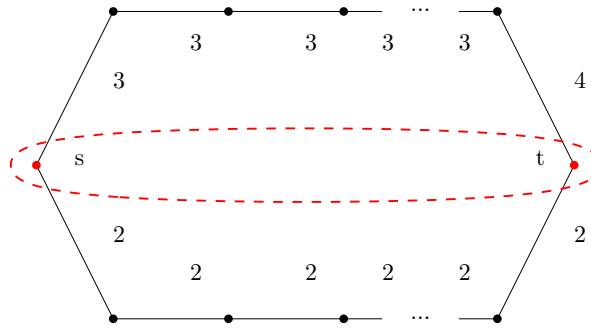
237 ▶ **Theorem 11** (★). *The worst-case number of \mathcal{S} components is n .*

238 ▶ **Theorem 12** (★). *The worst-case number of \mathcal{S} components is $\frac{n}{2}$ in temporal graphs without
 239 isolated vertices.*

240 ▶ **Theorem 13** (★). *\mathcal{S} COMPONENT is solvable in polynomial time $O(n(m \log T + n \log n))$.*

241 3.2 \mathcal{S}^B components

242 ▶ **Observation 14.** *Contrary to \mathcal{S} components, \mathcal{S}^B components can be open, as shown in
 243 Figure 2.*



244 ■ **Figure 2** A graph family admitting an open \mathcal{S}^B component for $\Delta = 2$ (in dashed and red with
 vertex s as source) and an open \mathcal{T}^B component for $\Delta = 2$ (in dashed and red with vertex t as sink).

244 ▶ **Lemma 15.** *A vertex can only be a source for one \mathcal{S}^B component.*

245 **Proof.** If a vertex was a source for two distinct \mathcal{S}^B components, then at least one wouldn't
 246 be maximal since their union would result in a larger \mathcal{S}^B component. ◀

247 ▶ **Theorem 16** (★). *The worst-case number of \mathcal{S}^B components is n .*

248 ▶ **Theorem 17** (★). *\mathcal{S}^B COMPONENT is solvable in polynomial time $O((T - \Delta)n(m \log \Delta +$
 249 $n \log n))$.*

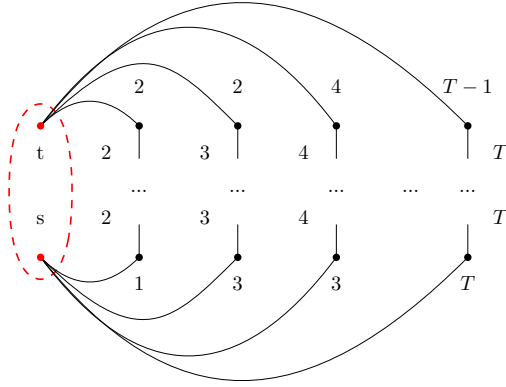
250 ▶ **Theorem 18** (★). *The worst-case number of closed \mathcal{S}^B components is n .*

251 ▶ **Theorem 19** (★). *CLOSED \mathcal{S}^B COMPONENT is solvable in polynomial time $O(n(n - k)(T -$
 252 $\Delta)(m \log \Delta + n \log n))$.*

253 3.3 \mathcal{S}^D components

254 Some results from \mathcal{S}^B components transfer for \mathcal{S}^D components, or give bounds, since \mathcal{S}^B
 255 components are \mathcal{S}^D components in which the source for every window is the same vertex.

256 ▶ **Observation 20.** *Even \mathcal{S}^D components where sources change between successive windows
 257 can be open, such as shown in Figure 3.*



■ **Figure 3** A graph family admitting an open $\mathcal{S}^{\mathcal{D}}$ component for $\Delta = 2$ (in dashed and red with vertex s as source for windows [odd, even] and t as source for windows [even, odd]).

258 ▶ **Lemma 21.** *An ordered set of vertices (one source per window possibly with repetitions)*
 259 *corresponds to sources of at most one $\mathcal{S}^{\mathcal{D}}$ component.*

260 **Proof.** If such a set of vertices were sources for two distinct $\mathcal{S}^{\mathcal{D}}$ components, then at least
 261 one wouldn't be maximal since their union would result in a larger $\mathcal{S}^{\mathcal{D}}$ component. ◀

262 ▶ **Theorem 22** (★). *The worst-case number of $\mathcal{S}^{\mathcal{D}}$ components is at least n , and at most*
 263 *$\min(2^n, n^{T-\Delta+1})$.*

264 ▶ **Theorem 23** (★). *$\mathcal{S}^{\mathcal{D}}$ COMPONENT is solvable in time $O(n^{\min(k, T-\Delta+2)}(T-\Delta)k(m \log \Delta +$
 265 *$n \log n))$.**

266 ▶ **Theorem 24** (★). *$\mathcal{S}^{\mathcal{D}}$ COMPONENT is NP-complete.*

267 ▶ **Theorem 25** (★). *The worst-case number of closed $\mathcal{S}^{\mathcal{D}}$ components is at least n , and at*
 268 *most $\min(2^n, n^{T-\Delta+1})$.*

269 ▶ **Theorem 26** (★). *CLOSED $\mathcal{S}^{\mathcal{D}}$ COMPONENT is solvable in time $O(\min(2^n, n^{T-\Delta+1})(T -$
 270 *$\Delta)n(m \log \Delta + n \log n))$.**

271 ▶ **Theorem 27** (★). *CLOSED $\mathcal{S}^{\mathcal{D}}$ COMPONENT is NP-complete.*

272 **4 All to/from all (\mathcal{TC} , $\mathcal{TC}^{\mathcal{B}}$, $\mathcal{TC}^{\mathcal{C}}$)**

273 A graph is said to be temporally connected if there exists a journey from each vertex to every
 274 other vertex. In other words, over time, each vertex is able to connect to the whole temporal
 275 graph, and is used for example in contexts with multi-hop message passing, distributed
 276 mobile agents, or social communication networks [26, 36, 37].

277 ▶ **Definition 28** (\mathcal{TC} component). *A \mathcal{TC} component of a temporal graph \mathcal{G} is a maximal*
 278 *subset $V' \subseteq V$ such that $\forall u, v \in V'$, $u \rightsquigarrow v$.*

279 Similar to $\mathcal{S}^{\mathcal{B}}$, time windows add multiple advantages to temporal connectivity. Gomez *et*
 280 *al.* suppose a $\mathcal{TC}^{\mathcal{B}}$ component exists in their work on agreement in dynamic systems in [23].
 281 In [10], Casteigts *et al.* present a general framework for computing parameters in temporal
 282 graphs, one of which being Δ for which the given graph is $\mathcal{TC}^{\mathcal{B}}$. Closed $\mathcal{TC}^{\mathcal{B}}$ components

283 are studied by Huyghues-Despointes *et al.* in [27]. They propose polynomial-time algorithms
 284 for computing lower and upper bounds on the maximum component size.

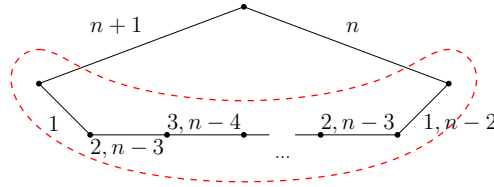
285 ► **Definition 29** ($\mathcal{TC}^{\mathcal{B}}$ component). A $\mathcal{TC}^{\mathcal{B}}$ component of duration Δ of a temporal graph
 286 \mathcal{G} is a maximal subset $V' \subseteq V$ such that $\forall v, v' \in V', \forall t \leq T - \Delta + 1, v \rightsquigarrow v'$ in temporal
 287 graphs $\mathcal{G}_{[t, t+\Delta-1]}$.

288 Another parameter Casteigts *et al.* are interested in is the round trip temporal diameter,
 289 being the shortest duration for which there exist round trip journeys between every pair of
 290 vertices. More generally, round-trip connectivity can represent systems in which feedback
 291 or acknowledgements are needed in a connection, such as Transmission Control Protocol
 292 (TCP).

293 ► **Definition 30** (\mathcal{TC}° component). A \mathcal{TC}° component of a temporal graph \mathcal{G} is a maximal
 294 subset $V' \subseteq V$ such that $\forall u, v \in V', u \circ v$ in \mathcal{G} .

295 4.1 \mathcal{TC} components

296 ► **Observation 31.** \mathcal{TC} components can be open such as shown in Figure 4.



■ **Figure 4** A graph family admitting an open \mathcal{TC} component (in dashed and red).

297 ► **Theorem 32** (\star). The worst-case number of \mathcal{TC} components is at least $2^{0.52\sqrt{n}}$, and at
 298 most $2^{0.53n}$.

299 Note (see appendix) that the exact bounds are $3^{(\sqrt{n+1}+1)/3}$ and $3^{n/3}$ respectively. We used
 300 $2^{0.52k} < 3^{k/3} < 2^{0.53k}$ so that results are comparable with results using powers of 2.

301 An algorithm is mentioned by Bhadra *et al.* [6] and implemented and experimented on by
 302 Nicosia *et al.* [42]. To the best of our knowledge, no complexity analysis of this algorithm
 303 has been performed.

304 ► **Theorem 33** (\star). \mathcal{TC} COMPONENT is solvable in time $O(nm \log T + \min(n^k k^2, 2^{0.25n}))$.

305 Bhadra *et al.* prove that \mathcal{TC} COMPONENT is NP-complete. They use a reduction from
 306 CLIQUE in which they produce a temporal graph with a lifetime $T = 4$ (so using labels
 307 between 1 and 4 included). This leaves the NP-hardness question open for lifetimes $1 < T < 4$
 308 ($T = 1$ being polynomial-time solvable). We treat the case of $T = 2$, filling the gap and
 309 proving general NP-hardness for $T > 1$.

310 ► **Theorem 34** (\star). \mathcal{TC} COMPONENT is NP-complete, for all constant lifetimes $T > 1$.

311 ► **Theorem 35** (\star). The worst-case number of closed \mathcal{TC} components is at least $2^{0.52\sqrt{n}}$, and
 312 at most $2^{0.53n}$.

313 ► **Theorem 36** (\star). CLOSED \mathcal{TC} COMPONENT is solvable in time $O(2^n n(m \log T + n \log n))$.

10 Temporally connected components

314 Bhadra *et al.* also prove CLOSED \mathcal{TC} COMPONENT is NP-complete through the same
 315 reduction as for \mathcal{TC} COMPONENT. We use the same trick to show CLOSED \mathcal{TC} COMPONENT
 316 is NP-complete for all constant lifetimes $T > 1$.

317 ► **Theorem 37** (\star). CLOSED \mathcal{TC} COMPONENT is NP-complete, for all constant lifetimes
 318 $T > 1$.

319 4.2 \mathcal{TC}^B components

320 ► **Lemma 38**. \mathcal{TC}^B components can be open.

321 **Proof**. Since \mathcal{TC} components are \mathcal{TC}^B components for the specific setting of $\Delta = T$, The-
 322 orem 32 transfers directly. ◀

323 ► **Theorem 39** (\star). The worst-case number of \mathcal{TC}^B components is at least $2^{0.52\sqrt{n}}$ and at
 324 most $2^{0.53n}$.

325 ► **Theorem 40** (\star). \mathcal{TC}^B COMPONENT is solvable in time $O((T - \Delta)n(m \log \Delta + n \log n) +$
 326 $\min(n^k k^2, 2^{0.25n}))$.

327 Again, since \mathcal{TC} components are \mathcal{TC}^B components by setting window size $\Delta = T$, NP-hardness
 328 transfers. We however prove a more general hardness for \mathcal{TC}^B COMPONENT regarding any
 329 constant lifetime and window size.

330 ► **Theorem 41** (\star). \mathcal{TC}^B COMPONENT is NP-complete, for all constant lifetimes $T > 1$ and
 331 window sizes $\Delta > 1$.

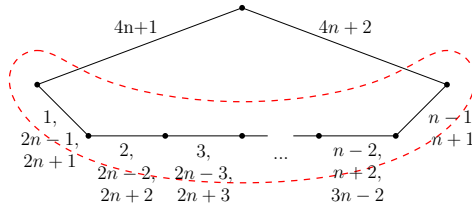
332 ► **Theorem 42** (\star). The worst-case number of closed \mathcal{TC}^B components is at least $2^{0.52\sqrt{n}}$
 333 and at most $2^{0.53n}$.

334 ► **Theorem 43** (\star). CLOSED \mathcal{TC}^B COMPONENT is solvable in time $O(2^n(T - \Delta)n(m \log \Delta +$
 335 $n \log n))$.

336 ► **Theorem 44** (\star). CLOSED \mathcal{TC}^B COMPONENT is NP-complete, for all constant lifetimes
 337 $T > 1$ and window sizes $\Delta > 1$.

338 4.3 \mathcal{TC}° components

339 ► **Observation 45**. \mathcal{TC}° components can be open, such as shown in Figure 5.



■ **Figure 5** Graph family admitting an open \mathcal{TC}° component (in red and dashed).

340 ► **Theorem 46** (\star). The worst-case number of \mathcal{TC}° components is at least $2^{0.52\sqrt{n}}$, and at
 341 most $2^{0.53n}$.

342 ► **Theorem 47** (\star). \mathcal{TC}° COMPONENT is solvable in time $O(n^2(m \log T + n \log n) + \min(n^k k^2, 2^{0.25n}))$.

343 Outside of the trivial case of lifetime $T = 1$, \mathcal{TC}° COMPONENT can be solved in polynomial
344 time for lifetime $T = 2$ as well.

345 ▶ **Theorem 48** (★). \mathcal{TC}° COMPONENT is solvable in polynomial time $O(n + m)$ on temporal
346 graphs with constant lifetimes $T \leq 2$.

347 ▶ **Theorem 49** (★). \mathcal{TC}° COMPONENT is NP-complete, for all constant lifetimes $T > 2$.

348 ▶ **Theorem 50** (★). The worst-case number of closed \mathcal{TC}° components is at least $2^{0.52\sqrt{n}}$,
349 and at most $2^{0.53n}$.

350 ▶ **Theorem 51** (★). CLOSED \mathcal{TC}° COMPONENT is solvable in time $O(2^n(n^2(m \log T +$
351 $n \log n)))$.

352 ▶ **Theorem 52** (★). CLOSED \mathcal{TC}° COMPONENT is solvable in polynomial time $O(n + m)$ on
353 temporal graphs with constant lifetimes $T \leq 2$.

354 ▶ **Theorem 53** (★). CLOSED \mathcal{TC}° COMPONENT is NP-complete, for all constant lifetimes
355 $T > 2$.

356 5 Conclusion

357 5.1 Summary of this paper

358 A first summary of this paper is given in the following table, where we precise for each \mathcal{X}
359 component the worst-case number of components, or the presented bounds on this number
360 (using notation [lower bound, upper bound]), as well as the complexity of the corresponding
361 decision problem and its corresponding complexity class (P referring to polynomial-time
362 solvable, NPC to NP-complete). For \mathcal{X} components defined through the \mathcal{TC} property (lower
363 part of the table) we give the constant values of T (and Δ when applicable) for which the
364 corresponding decision problem is NP-complete.

Component	Worst-case Number	Complexity	Class
\mathcal{S}	n ($\frac{n}{2}$ no isolated vertices)	$O(n(m \log T + n \log n))$	P
\mathcal{S}^B	n	$O((T - \Delta)n(m \log \Delta + n \log n))$	P
Closed \mathcal{S}^B	n	$O(n(T - \Delta)(n - k)(m \log \Delta + n \log n))$	P
\mathcal{S}^D	$[n, \min(2^n, n^{T-\Delta+1})]$	$O(n^{\min(k, T-\Delta+2)}(T - \Delta)k(m \log \Delta + n \log n))$	NPC
Closed \mathcal{S}^D	$[n, \min(2^n, n^{T-\Delta+1})]$	$O(\min(2^n, n^{T-\Delta+1})(T - \Delta)n(m \log \Delta + n \log n))$	NPC
\mathcal{TC}	$[2^{0.52\sqrt{n}}, 2^{0.53n}]$	$O(nm \log T + \min(n^k k^2, 2^{0.25n}))$	NPC ($T > 1$)
Closed \mathcal{TC}	$[2^{0.52\sqrt{n}}, 2^{0.53n}]$	$O(2^n n(m \log T + n \log n))$	NPC ($T > 1$)
\mathcal{TC}^B	$[2^{0.52\sqrt{n}}, 2^{0.53n}]$	$O((T - \Delta)n(m \log \Delta + n \log n) + \min(n^k k^2, 2^{0.25n}))$	NPC ($T \& \Delta > 1$)
Closed \mathcal{TC}^B	$[2^{0.52\sqrt{n}}, 2^{0.53n}]$	$O(2^n (T - \Delta)n(m \log \Delta + n \log n))$	NPC ($T \& \Delta > 1$)
\mathcal{TC}°	$[2^{0.52\sqrt{n}}, 2^{0.53n}]$	$O(n^2(m \log T + n \log n) + \min(n^k k^2, 2^{0.25n}))$	NPC ($T > 2$)
Closed \mathcal{TC}°	$[2^{0.52\sqrt{n}}, 2^{0.53n}]$	$O(2^n(n^2(m \log T + n \log n)))$	NPC ($T > 2$)

365 Temporal graphs with lifetime $T = 1$ trivially admit polynomial-time algorithms for all
366 these problems (since they reduce to finding connected components in the static graph), so
367 they are not presented. Note that all these results are to be found in this paper, although
368 some parts were first presented in related works. We also prove \mathcal{TC}° COMPONENT and
369 CLOSED \mathcal{TC}° COMPONENT can be solved in polynomial time for temporal graphs with
370 lifetime $T = 2$.

371 5.2 A short discussion about optimisation of windowed components

372 This paper provided algorithms (polynomial or not) to solve the windowed versions of our
 373 problems. Some of these consider treat the windows in the specific order $([1, \Delta], [2, \Delta +$
 374 $1], \dots, [T - \Delta + 1, T])$. We note this order isn't necessary, in the sense that the result would
 375 remain unchanged if it considered the time windows in other orders. A consequence potentially
 376 useful for optimisation is that one could reorder the time windows in the most "favorable" way.
 377 For example, for \mathcal{S}^B , start with the window in which the corresponding temporal graph may
 378 have few and small candidate components, and start the intersection process with windows
 379 in which the corresponding temporal graphs may have very distinct \mathcal{S} components from the
 380 candidate components. Of course, theoretically this would only be useful if such an estimation
 381 and reordering of time windows could be done in time $O((T - \Delta)(n(m \log \Delta + n \log n)))$ as
 382 well, and even then practically this may still induce a significant change in running time.

383 Another possible optimisation concerns some bounded versions of problems related to \mathcal{TC} ,
 384 where transitive closures are computed for each time window. Can one do better by computing
 385 a modified transitive closure over the whole graph, which iteratively keeps track of how old
 386 journeys' starting dates are, and removes them over time if too old? Our first investigations
 387 did not allow to show this idea brings a breakthrough.

388 Both of these directions we leave as open research avenues in this paper.

389 5.3 A quick note on parameterised complexity

390 Among the NP-hard problems in this paper, all but two are para-NP-hard concerning
 391 the lifetime parameter, *i.e.* for some constant value of T the problem is NP-hard. The
 392 two problems which are not para-NP-hard regarding lifetime T are \mathcal{S}^D COMPONENT and
 393 CLOSED \mathcal{S}^D COMPONENT. Both are in XP, *i.e.* the problems are polynomial-time solvable for
 394 all constant T (and \mathcal{S}^D COMPONENT is in XP as well concerning the solution size parameter
 395 k). It may be of interest to determine where exactly in XP the problems are situated, *e.g.*
 396 FPT or W[1]-hard. Of course, outside of T , other natural parameters may be considered.
 397 Again, this research is left as open avenues.

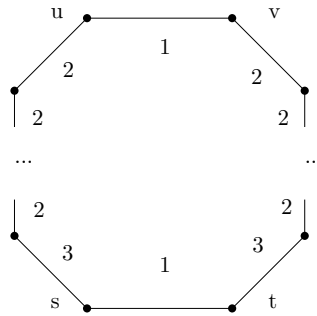
398 — **References** —

- 399 1 Eleni C Akrida, Leszek Gasieniec, George B Mertzios, and Paul G Spirakis. The complexity of
400 optimal design of temporally connected graphs. *Theory of Computing Systems*, 61(3):907–944,
401 2017.
- 402 2 Aris Anagnostopoulos, Ravi Kumar, Mohammad Mahdian, Eli Upfal, and Fabio Vandin.
403 Algorithms on evolving graphs. In *Proceedings of the 3rd Innovations in Theoretical Computer
404 Science Conference, ITCS '12*, pages 149–160, 2012.
- 405 3 Brenda Baker and Robert Shostak. Gossips and telephones. *Discrete Mathematics*, 2(3):191–
406 193, June 1972.
- 407 4 Stefan Balev, Yoann Pigné, Eric Sanlaville, and Mathilde Vernet. Complexité du problème de
408 Steiner dynamique. In *23ème congrès annuel de la Société Française de Recherche Opération-
409 nelle et d'Aide à la Décision*, Villeurbanne - Lyon, France, February 2022. INSA Lyon.
- 410 5 Matthieu Barjon, Arnaud Casteigts, Serge Chaumette, Colette Johnen, and Yessin M Neggaz.
411 Testing temporal connectivity in sparse dynamic graphs. *arXiv preprint arXiv:1404.7634*,
412 2014.
- 413 6 Sandeep Bhadra and Afonso Ferreira. Computing multicast trees in dynamic networks and
414 the complexity of connected components in evolving graphs. *Journal of Internet Services and
415 Applications*, 3(3):269–275, 2012.
- 416 7 Peter Bradshaw and Bojan Mohar. A Rainbow Connectivity Threshold for Random Graph
417 Families. In Jaroslav Nešetřil, Guillem Perarnau, Juanjo Rué, and Oriol Serra, editors, *Extended
418 Abstracts EuroComb 2021*, Trends in Mathematics, pages 842–847. Springer International
419 Publishing, 2021.
- 420 8 Arnaud Casteigts. *A Journey Through Dynamic Networks (with Excursions)*. Habilitation à
421 diriger des recherches, Université de Bordeaux, June 2018.
- 422 9 Arnaud Casteigts, Paola Flocchini, Walter Quattrociocchi, and Nicola Santoro. Time-varying
423 graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed
424 Systems*, 27(5):387–408, 2012.
- 425 10 Arnaud Casteigts, Ralf Klasing, Yessin M Neggaz, and Joseph G Peters. Computing parameters
426 of sequence-based dynamic graphs. *Theory of Computing Systems*, 63(3):394–417, 2019.
- 427 11 Arnaud Casteigts, Joseph G Peters, and Jason Schoeters. Temporal cliques admit sparse
428 spanners. *Journal of Computer and System Sciences*, 121:1–17, 2021.
- 429 12 Arnaud Casteigts, Michael Raskin, Malte Renken, and Viktor Zamaraev. Sharp thresholds in
430 random simple temporal graphs. In *2021 IEEE 62nd Annual Symposium on Foundations of
431 Computer Science (FOCS)*, pages 319–326. IEEE, 2022.
- 432 13 Lily Chen, Xueliang Li, and Yongtang Shi. The complexity of determining the rainbow
433 vertex-connection of a graph. *Theoretical Computer Science*, 412(35):4531–4535, 2011.
- 434 14 Julia Chuzhoy and Sanjeev Khanna. A new algorithm for decremental single-source shortest
435 paths with applications to vertex-capacitated flow and cut problems. In *Proceedings of the
436 51st Annual ACM SIGACT Symposium on Theory of Computing, STOC 2019*, pages 389–400.
437 Association for Computing Machinery, 2019.
- 438 15 Pierluigi Crescenzi, Clémence Magnien, and Andrea Marino. Approximating the temporal
439 neighbourhood function of large temporal graphs. *Algorithms*, 12(10):211, 2019.
- 440 16 Argyrios Deligkas, Eduard Eiben, and George Skretas. Minimizing reachability times on
441 temporal graphs via shifting labels. *arXiv preprint arXiv:2112.08797*, 2021.

- 442 **17** Reinhard Diestel. *Graph Theory*, volume 173 of *Graduate Texts in Mathematics*. Springer,
443 Berlin, Heidelberg, 2017.
- 444 **18** Antoine Dutot, Frédéric Guinand, Damien Olivier, and Yoann Pigné. Graphstream: A tool
445 for bridging the gap between complex systems and dynamic graphs. In *Emergent Properties*
446 *in Natural and Artificial Complex Systems. (ECCS'2007)*, 2007.
- 447 **19** Jessica Enright, Kitty Meeks, and Hendrik Molter. Counting temporal paths. *arXiv preprint*
448 *arXiv:2202.12055*, 2022.
- 449 **20** Jeff Erickson. *Algorithms*. self-publishing, 2019.
- 450 **21** Satoshi Fujita, Stephane Perennes, and Joseph G Peters. Neighbourhood gossiping in hyper-
451 cubes. *Parallel Processing Letters*, 8(02):189–195, 1998.
- 452 **22** Frits Göbel, J Orestes Cerdeira, and Hendrik Jan Veldman. Label-connected graphs and the
453 gossip problem. *Discrete Mathematics*, 87(1):29–40, 1991.
- 454 **23** Carlos Gómez-Calzado, Arnaud Casteigts, Alberto Lafuente, and Mikel Larrea. A connectivity
455 model for agreement in dynamic systems. In Jesper Larsson Träff, Sascha Hunold, and
456 Francesco Versaci, editors, *Euro-Par 2015: Parallel Processing*, Lecture Notes in Computer
457 Science, pages 333–345, Berlin, Heidelberg, 2015. Springer, Springer.
- 458 **24** John P. Hayes. A graph model for fault-tolerant computing systems. *IEEE Transactions on*
459 *Computers*, 25(09):875–884, 1976.
- 460 **25** Anne-Sophie Himmel, Hendrik Molter, Rolf Niedermeier, and Manuel Sorge. Adapting the
461 bron–kerbosch algorithm for enumerating maximal cliques in temporal graphs. *Social Network*
462 *Analysis and Mining*, 7(1):35, 2017.
- 463 **26** Petter Holme. Network reachability of real-world contact sequences. *Physical Review E*,
464 71(4):046119, 2005.
- 465 **27** Charles Huyghues-Despointes, Binh-Minh Bui-Xuan, and Clémence Magnien. Forte delta-
466 connexité dans les flots de liens. In *ALGOTEL 2016-18èmes Rencontres Francophones sur les*
467 *Aspects Algorithmiques des Télécommunications*, 2016.
- 468 **28** Allen Ibiapina and Ana Silva. Mengerian temporal graphs revisited. In *International Symposium*
469 *on Fundamentals of Computation Theory*, pages 301–313. Springer, Springer International
470 Publishing, 2021.
- 471 **29** David Ilcinkas, Ralf Klasing, and Ahmed Mouhamadou Wade. Exploration of constantly
472 connected dynamic graphs based on cactuses. In *International Colloquium on Structural*
473 *Information and Communication Complexity*, pages 250–262. Springer, 2014.
- 474 **30** Joe-Air Jiang, Chien-Hao Wang, Min-Sheng Liao, Xiang-Yao Zheng, Jen-Hao Liu, Cheng-Long
475 Chuang, Che-Lun Hung, and Chia-Pang Chen. A wireless sensor network-based monitoring
476 system with dynamic convergecast tree algorithm for precision cultivation management in
477 orchid greenhouses. *Precision agriculture*, 17(6):766–785, 2016.
- 478 **31** Amol Kapoor, Xue Ben, Luyang Liu, Bryan Perozzi, Matt Barnes, Martin Blais, and Shawn
479 O’Banion. Examining covid-19 forecasting using spatio-temporal graph neural networks. *arXiv*
480 *preprint arXiv:2007.03113*, 2020.
- 481 **32** Richard M. Karp. Reducibility among Combinatorial Problems. In Raymond E. Miller,
482 James W. Thatcher, and Jean D. Bohlinger, editors, *Complexity of Computer Computations*,
483 The IBM Research Symposia Series, pages 85–103. Springer US, Boston, MA, 1972.
- 484 **33** David Kempe, Jon Kleinberg, and Amit Kumar. Connectivity and inference problems for
485 temporal networks. *Journal of Computer and System Sciences*, 64(4):820–842, 2002.

- 486 **34** Abdul Waheed Khan, Abdul Hanan Abdullah, Mohammad Hossein Anisi, and Javed Iqbal
487 Bangash. A comprehensive study of data collection schemes using mobile sinks in wireless
488 sensor networks. *Sensors*, 14(2):2510–2548, 2014.
- 489 **35** Nina Klobas, George B Mertzios, Hendrik Molter, and Paul G Spirakis. The complexity of
490 computing optimum labelings for temporal connectivity. *arXiv preprint arXiv:2202.05880*,
491 2022.
- 492 **36** Gueorgi Kossinets, Jon Kleinberg, and Duncan Watts. The structure of information pathways
493 in a social communication network. In *Proceedings of the 14th ACM SIGKDD international*
494 *conference on Knowledge discovery and data mining*, KDD '08, pages 435–443. Association for
495 Computing Machinery, August 2008.
- 496 **37** Vassilis Kostakos. Temporal graphs. *Physica A: Statistical Mechanics and its Applications*,
497 388(6):1007–1023, 2009.
- 498 **38** Michael Krivelevich and Raphael Yuster. The rainbow connection of a graph is (at most)
499 reciprocal to its minimum degree. *Journal of Graph Theory*, 63(3):185–191, 2010.
- 500 **39** Matthieu Latapy, Tiphaine Viard, and Clémence Magnien. Stream graphs and link streams
501 for the modeling of interactions over time. *Social Network Analysis and Mining*, 8(1):61, 2018.
- 502 **40** George B Mertzios, Hendrik Molter, Rolf Niedermeier, Viktor Zamaraev, and Philipp Zschoche.
503 Computing maximum matchings in temporal graphs. *arXiv preprint arXiv:1905.05304*, 2019.
- 504 **41** John W Moon and Leo Moser. On cliques in graphs. *Israel journal of Mathematics*, 3(1):23–28,
505 1965.
- 506 **42** Vincenzo Nicosia, John Tang, Mirco Musolesi, Giovanni Russo, Cecilia Mascolo, and Vito
507 Latora. Components in time-varying graphs. *Chaos: An interdisciplinary journal of nonlinear*
508 *science*, 22(2):023101, 2012.
- 509 **43** John M Robson. Finding a maximum independent set in time $o(2n/4)$. Technical report,
510 Technical Report 1251-01, LaBRI, Université Bordeaux I, 2001.
- 511 **44** Mathilde Vernet, Yoann Pigné, and Éric Sanlaville. A study of connectivity on dynamic
512 graphs: Computing persistent connected components. *4OR*, April 2022.
- 513 **45** Dong Wen, Lu Qin, Ying Zhang, Lijun Chang, and Ling Chen. Enumerating k-Vertex
514 Connected Components in Large Graphs. In *2019 IEEE 35th International Conference on*
515 *Data Engineering (ICDE)*, pages 52–63, April 2019.
- 516 **46** B Bui Xuan, Afonso Ferreira, and Aubin Jarry. Computing shortest, fastest, and foremost
517 journeys in dynamic networks. *International Journal of Foundations of Computer Science*,
518 14(02):267–285, 2003.
- 519 **47** Hiroyuki Yomo, Akitoshi Asada, and Masato Miyatake. On-demand data gathering with
520 a drone-based mobile sink in wireless sensor networks exploiting wake-up receivers. *IEICE*
521 *Transactions on Communications*, 101(10):2094–2103, 2018.
- 522 **48** Ping Yu, Zhiping Wang, Peiwen Wang, Haofei Yin, and Jia Wang. Dynamic evolution of
523 shipping network based on hypergraph. *Physica A: Statistical Mechanics and its Applications*,
524 598:127247, July 2022.
- 525 **49** Wenyu Zang, Peng Zhang, Chuan Zhou, and Li Guo. Discovering Multiple Diffusion Source
526 Nodes in Social Networks. *Procedia Computer Science*, 29:443–452, 2014.
- 527 **50** Philipp Zschoche, Till Fluschnik, Hendrik Molter, and Rolf Niedermeier. The complexity
528 of finding small separators in temporal graphs. *Journal of Computer and System Sciences*,
529 107:72–92, 2020.

530 **A** Auxiliary proofs531 **A.1** \mathcal{S} components532 **► Theorem 11.** *The worst-case number of \mathcal{S} components is n .*533 **Proof.** Theorem 9 implies that at most n \mathcal{S} components can exist, which by Theorem 10 is
534 tight when considering the trivial case of the empty temporal graph, *i.e.* a temporal graph
535 composed of empty graph snapshots $G_i = (V, \emptyset)$. ◀536 For Theorem 12, we establish a relation between \mathcal{S} components and locally minimum label
537 edge sets, *i.e.* connected edge sets sharing a label which have no incident edges with lower
538 labels other than themselves.539 **► Lemma 54.** *The number of \mathcal{S} components is at most the number of locally minimum edge
540 sets in a temporal graph without isolated vertices.*541 **Proof.** For any locally minimum label edge set $E' = \{\{u_1, u_2\}, \dots, \{u_j, u_k\}\}$, observe that \mathcal{S}
542 components S_{u_i} with source $u_i \in V(E')$ are identical (in terms of their vertex set), since all
543 vertices u_i can reach all vertices reachable by all other vertices of E' through the locally
544 minimum label edge set. Hence, such vertices u_i of a locally minimum label edge set produce
545 altogether at most one \mathcal{S} component. Concerning vertices which are not part of a locally
546 minimum label edge set, say vertex v , observe that since v is not part of a locally minimum
547 label edge set, this means that its incident edge with the smallest label, say $\{v, x\}$, must have
548 some incident edge $\{x, y\}$ with an even smaller label (or $\{v, x\}$ is part of a connected edge
549 set of the same label which has such an incident edge). Now, either $\{x, y\}$ is part of a locally
550 minimum label edge set or it has again some incident edge with a smaller label (or again its
551 connected edge set of the same label does). Continue this process until having found a locally
552 minimum label edge set, say $E' = \{\{u_1, u_2\}, \dots, \{u_j, u_k\}\}$. Note that any u_i can reach v at
553 the latter's earliest incident edge's label, implying any vertex v can reach, can be reached by
554 u_i as well, through v . \mathcal{S} component S_v with source v thus has to be contained (in terms of
555 their vertex set) within the larger \mathcal{S} component S_{u_i} with source u_i , since u_i can reach all
556 vertices v can reach. Hence, vertices v which are not part of a locally minimum label edge
557 set cannot be the source of a \mathcal{S} component, since it would not be maximal. This combined
558 with locally minimum label edge sets producing at most one \mathcal{S} component each, effectively
559 bounds the number of \mathcal{S} components by the number of locally minimum label edge sets. ◀560 Note that there is not necessarily equality for Theorem 54, for example when one component
561 induced by a locally minimum label edge set is included in another, such as shown in Figure 6.562 **► Lemma 55.** *At most $\frac{n}{2}$ locally minimum label edge sets may exist in any temporal graph,
563 which is tight.*564 **Proof.** Since locally minimum label edge sets cannot be incident by definition, and cover at
565 least two vertices each, at most $\frac{n}{2}$ locally minimum label edge sets exist in any graph. This
566 is shown tight as follows. Take a complete graph on n vertices $K_n = (V, E)$ without any
567 labels. Now, take a maximum matching in K_n and assign small labels to it, and to the rest
568 of the edges, either assign large labels or remove these edges. The result is a temporal graph
569 with exactly $\frac{n}{2}$ locally minimum label edge sets, corresponding to the maximum matching's
570 $\frac{n}{2}$ edges.

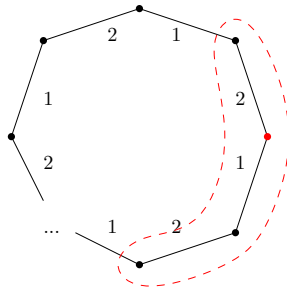


■ **Figure 6** Example graph family for which Theorem 54 does not induce an equality between the number of \mathcal{S} components (only one exists composed of all vertices with source either u or v) and the number of locally minimum edge sets (two such sets exist: $\{u, v\}$ and $\{s, t\}$).

571

572 ▶ **Theorem 12.** *The worst-case number of \mathcal{S} components is $\frac{n}{2}$ in temporal graphs without*
 573 *isolated vertices.*

574 **Proof.** In other words, we prove that at most $\frac{n}{2}$ \mathcal{S} components may exist in any temporal
 575 graph without isolated vertices, and we prove this is tight. Figure 7 shows a graph family
 576 which admits $\frac{n}{2}$ \mathcal{S} components, which by Theorem 54 and Theorem 55 finishes the proof.



■ **Figure 7** A graph family with $\frac{n}{2}$ \mathcal{S} components. A component with its corresponding source is marked in red.

577

578 ▶ **Theorem 13.** *\mathcal{S} COMPONENT is solvable in polynomial time $O(n(m \log T + n \log n))$.*

579 **Proof.** We propose the following algorithm for \mathcal{S} COMPONENT. First, construct the transitive
 580 closure H of the given temporal graph \mathcal{G} , then check if $k \leq \Delta^+(H) + 1$, where $\Delta^+(H)$
 581 denotes the maximum outdegree of H . The described algorithm has a time complexity of
 582 $O(n(m \log T + n \log n))$ to construct the transitive closure H , and an additional $O(n^2)$ to
 583 compute the maximum outdegree $\Delta^+(H)$.

584 ▶ **Theorem 56.** *For a given temporal graph, all \mathcal{S} components can be enumerated in time*
 585 *$O(nm \log T + n^3)$.*

586 **Proof.** Start by computing the transitive closure H , taking time $O(n(m \log T + n \log n))$.
 587 Then, let the set of \mathcal{S} components be composed of vertex v with all its out-neighbours in H ,
 588 for all vertices $v \in V$. Now, some of these \mathcal{S} components may be included in others, meaning
 589 they may be non-maximal. To remove non-maximal \mathcal{S} components, start by sorting the
 590 vertices in each \mathcal{S} component, taking time $O(n^2 \log n)$. Then, for each pair of \mathcal{S} components,
 591 of which there are $O(n^2)$, check whether one is a subset of the other, and remove it if so,
 592 which can be done in time $O(n)$. All in all, enumeration of \mathcal{S} components can be done in
 593 time $O(n(m \log T + n \log n) + n^2 \log n + n^3) = O(nm \log T + n^3)$. ◀

594 A.2 $\mathcal{S}^{\mathcal{B}}$ components

595 ▶ **Theorem 16.** *The worst-case number of $\mathcal{S}^{\mathcal{B}}$ components is n .*

596 **Proof.** Theorem 15 implies at most n $\mathcal{S}^{\mathcal{B}}$ components may exist, which is tight for example
 597 for the empty temporal graph. In fact, it is tight even for much denser graphs; a temporal
 598 graph with one window of size Δ containing only empty snapshots is sufficient to obtain n
 599 $\mathcal{S}^{\mathcal{B}}$ components of size 1, meaning even a temporal graph with a complete graph as footprint
 600 may attain n $\mathcal{S}^{\mathcal{B}}$ components. ◀

601 ▶ **Theorem 17.** *$\mathcal{S}^{\mathcal{B}}$ COMPONENT is solvable in polynomial time $O((T - \Delta)n(m \log \Delta +$
 602 $n \log n))$.*

603 **Proof.** We propose the following algorithm for $\mathcal{S}^{\mathcal{B}}$ COMPONENT. Start by enumerating the
 604 \mathcal{S} components of the windowed temporal graph $\mathcal{G}_{[1, \Delta]}$, which by Theorem 56 can be done in
 605 time $O(nm \log \Delta + n^3)$. However, we are not interested in maximal \mathcal{S} components, so the
 606 added $O(n^3)$ can be removed. Thus, we enumerate \mathcal{S} components, some of which may not
 607 be maximal, in time $O(n(m \log \Delta + n \log n))$. Keep only those of size at least k . These are
 608 candidate components for $\mathcal{S}^{\mathcal{B}}$ components. Let us denote these candidate components as S_s by
 609 their unique corresponding source vertex s (Theorem 9). Then, for each windowed temporal
 610 graph $\mathcal{G}_{[t, t + \Delta - 1]}$, of which there are $O(T - \Delta)$, obtain their \mathcal{S} components in the same manner,
 611 again in time $O(n(m \log \Delta + n \log n))$. Denote these \mathcal{S} components as $S_s[t, t + \Delta - 1]$. For
 612 each candidate component S_s , of which there are $O(n)$, update S_s by setting it to the
 613 intersection of S_s and $S_s[t, t + \Delta - 1]$ which is doable in time $O(n \log n)$. Remove S_s from
 614 the candidate components if its size is less than k . If no candidate components remain at
 615 some point, the instance is negative, otherwise it is positive. All in all, this gives a total
 616 time complexity of $O(n(m \log \Delta + n \log n) + (T - \Delta)n(m \log \Delta + n \log n + n(n \log n))) =$
 617 $O((T - \Delta)n(m \log \Delta + n \log n))$. ◀

618 ▶ **Theorem 57.** *For a given temporal graph, all $\mathcal{S}^{\mathcal{B}}$ components can be enumerated in time*
 619 $O((T - \Delta)n(m \log T + n \log n) + n^3)$.

620 **Proof.** Run the algorithm from Theorem 17 and at the end add the verification and removal
 621 of non-maximal $\mathcal{S}^{\mathcal{B}}$ components, doable as in Theorem 56 in time $O(n^3)$, for a total of
 622 $O((T - \Delta)n(m \log T + n \log n) + n^3)$. ◀

623 ▶ **Theorem 18.** *The worst-case number of closed $\mathcal{S}^{\mathcal{B}}$ components is n .*

624 **Proof.** Theorem 16 directly adapts for closed $\mathcal{S}^{\mathcal{B}}$ components, since n $\mathcal{S}^{\mathcal{B}}$ components of
 625 size 1 are necessarily closed. ◀

626 ► **Lemma 58.** *Verifying a solution vertex subset S_s of source s and of size k for CLOSED \mathcal{S}^B COM-*
 627 *PONENT takes time $O((T - \Delta)(\min(m, k^2) \log \Delta + k \log k))$.*

628 **Proof.** For each windowed temporal subgraph $\mathcal{G}[S_s]_{[t, t+\Delta-1]}$, of which there are $O(T - \Delta)$,
 629 apply the temporal Dijkstra algorithm from vertex s to obtain its reachability which is done
 630 in time $O(\min(m, k^2) \log \Delta + k \log k)$. If for all windows, its reachability covers the whole
 631 vertex set S_s , then S_s is a solution, else it is not. ◀

632 ► **Theorem 19.** *CLOSED \mathcal{S}^B COMPONENT is solvable in polynomial time $O(n(n - k)(T -$
 633 $\Delta)(m \log \Delta + n \log n))$.*

634 **Proof.** We propose the following algorithm for CLOSED \mathcal{S}^B COMPONENT. Run the algorithm
 635 for \mathcal{S}^B COMPONENT. At the end of the algorithm, the set of candidates contains all \mathcal{S}^B
 636 components S_s of size at least k . For all S_s , test if it is closed, *i.e.* if for all $t \leq T - \Delta + 1$, S_s
 637 is an \mathcal{S} component of temporal graph $\mathcal{G}[S_s]_{[t, t+\Delta-1]}$. If some component is, then the instance
 638 is positive. If however S_s is open, meaning at some time $t \leq T - \Delta + 1$, vertex s cannot
 639 reach all vertices in $\mathcal{G}[S_s]_{[t, t+\Delta-1]}$, then S_s can be shrunk down to contain only the reachable
 640 vertices in $\mathcal{G}[S_s]_{[t, t+\Delta-1]}$. This shrunk down S_s can now be tested again: if it is of size at
 641 least k and closed, then the instance is positive, and if not we can again shrink S_s *etc.* If no
 642 closed \mathcal{S}^B component remains of suitable size, then the instance is negative. The algorithm
 643 for \mathcal{S}^B COMPONENT runs in time $O((T - \Delta)n(m \log \Delta + n \log n))$, after which $O(n)$ candidate
 644 components are to test whether they contain a large enough closed \mathcal{S}^B component. For each
 645 of these, testing if they are closed, takes time $O((T - \Delta)(m \log \Delta + n \log n))$ by Theorem 58.
 646 This process is repeated in case the component is not closed and a shrunk component
 647 has to be tested which again isn't closed but contains another shrunk component to be
 648 tested *etc.* This can happen at most $n - k$ times, since the size of a component is at most n
 649 and we're not interested in components of size less than k . Hence, all together, we obtain
 650 a complexity of $O(n(T - \Delta)(m \log \Delta + n \log n) + n(n - k)(T - \Delta)(m \log \Delta + n \log n)) =$
 651 $O(n(n - k)(T - \Delta)(m \log \Delta + n \log n))$. ◀

652 A.3 \mathcal{S}^D components

653 ► **Theorem 22.** *The worst-case number of \mathcal{S}^D components is at least n , and at most*
 654 *$\min(2^n, n^{T-\Delta+1})$.*

655 **Proof.** The lower bound comes from Theorem 18, with an example being the empty temporal
 656 graph. For the upper bounds, since \mathcal{S}^D components are subsets of vertices which may intersect,
 657 at most 2^n \mathcal{S}^D components may exist. By Theorem 21, representing components through
 658 their sources, *i.e.* ordered sets of vertices (one source per window), another bound of $n^{T-\Delta+1}$
 659 is found. ◀

660 ► **Lemma 59.** *Verifying a solution vertex subset V' of size k for \mathcal{S}^D COMPONENT takes*
 661 *time $O((T - \Delta)k(m \log \Delta + n \log n))$.*

662 **Proof.** For each window $[t, t + \Delta - 1]$, of which there are $T - \Delta$, one constructs a partial
 663 transitive closure H' of $\mathcal{G}_{[t, t+\Delta-1]}$ in which only reachability for the vertices V' is computed.
 664 This takes time $O(k(m \log \Delta + n \log n))$. For each induced subgraph $H'[V']$, checking if
 665 a vertex with outdegree $k - 1$ exists takes time $O(k^2)$. In total, this takes time $O((T -$
 666 $\Delta)(k(m \log \Delta + n \log n) + k^2)) = O((T - \Delta)k(m \log \Delta + n \log n))$. ◀

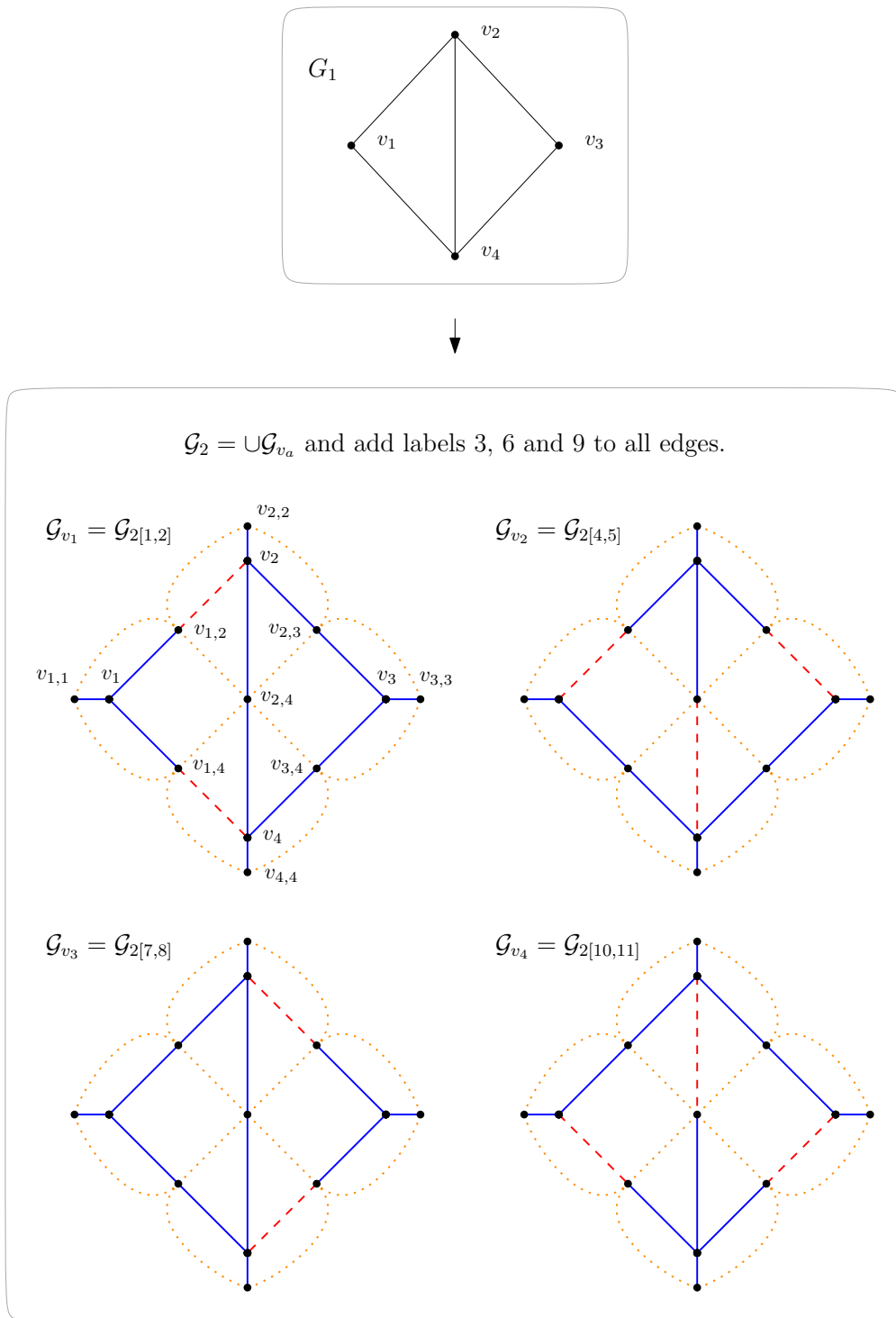
667 ► **Theorem 23.** $\mathcal{S}^{\mathcal{D}}$ COMPONENT is solvable in time $O(n^{\min(k, T-\Delta+2)}(T-\Delta)k(m \log \Delta +$
 668 $n \log n))$.

669 **Proof.** We present two algorithms, the first being the brute force algorithm of testing
 670 all possible subsets of vertices of size k , the other a generalization of the algorithm for
 671 $\mathcal{S}^{\mathcal{B}}$ COMPONENT. Let us start by analysing the brute force algorithm first. For all possible
 672 subsets of vertices of interest, of which there exist $\binom{n}{k} = O(n^k)$, we verify if it is a suitable
 673 solution. By Theorem 59, one verification takes time $O((T-\Delta)k(m \log \Delta + n \log n))$, meaning
 674 the brute force algorithm runs in time $O(n^k(T-\Delta)k(m \log \Delta + n \log n))$. Concerning
 675 the generalization of the algorithm for $\mathcal{S}^{\mathcal{B}}$ COMPONENT, only one change needs to be
 676 made. Instead of shrinking candidate component S_s at every window by taking the union
 677 with $S_s[t, t + \Delta - 1]$, we will need to instead shrink the candidate component with all
 678 $S_v[t, t + \Delta - 1]$ for all $v \in S_s$ (which could all correspond to a source for that time window),
 679 splitting a candidate component up into potentially $O(n)$ distinct components every time
 680 window. A result of this is that at every time window $[t, t + \Delta - 1]$, not $O(n)$ candidate
 681 components S_s exist from previous windows, but at most $n^t = O(n^{T-\Delta+1})$ candidate
 682 components $S_{(s,t,u,\dots,v)}$. Analysing the complexity gives $O(n^{T-\Delta+1}n(T-\Delta)(m \log \Delta +$
 683 $n \log n))$. ◀

684 ► **Theorem 24.** $\mathcal{S}^{\mathcal{D}}$ COMPONENT is NP-complete.

685 **Proof.** Theorem 59 proves $\mathcal{S}^{\mathcal{D}}$ COMPONENT is in NP. To prove NP-hardness, we give a
 686 polynomial-time reduction from CLIQUE to $\mathcal{S}^{\mathcal{D}}$ COMPONENT, *i.e.* we show how to transform
 687 any instance of CLIQUE to an instance of $\mathcal{S}^{\mathcal{D}}$ COMPONENT such that the instance of CLIQUE
 688 is positive if and only if the instance of $\mathcal{S}^{\mathcal{D}}$ COMPONENT is positive.

689 Given an instance of CLIQUE, being a graph G_1 of n_1 vertices and m_1 edges and an integer k_1 ,
 690 we describe the following polynomial-time transformation to an instance of $\mathcal{S}^{\mathcal{D}}$ COMPONENT,
 691 being a temporal graph $\mathcal{G}_2 = (G_2, \lambda_2)$ of lifetime T_2 and integers Δ_2 and k_2 . Set $T_2 = 3n_1 - 1$,
 692 $\Delta_2 = 2$ and $k_2 = m_1 + n_1 + k_1$. Let's construct the footprint G_2 , initially the empty
 693 graph. Suppose $V(G_1) = \{v_1, v_2, \dots, v_{n_1}\}$. For each vertex $v_i \in V(G_1)$, add vertices v_i and
 694 $v_{i,i}$ with edge $\{v_i, v_{i,i}\}$ to G_2 . We refer to vertices v_i in G_2 as original vertices, and to
 695 vertices $v_{i,i}$ as satellite vertices. Then, for each edge $\{v_i, v_j\} \in E(G_1)$, add vertex $v_{i,j} = v_{j,i}$
 696 and edges $\{v_i, v_{i,j}\}$, $\{v_{i,j}, v_j\}$ to G_2 . These edges will allow for v_i to reach v_j and/or vice
 697 versa, depending on the labelling. We refer to vertices $v_{i,j}$ as intermediary vertices. Finally,
 698 $\forall v_{i,j}, v_{h,k} \in G_2$, add edges $\{v_{i,j}, v_{h,k}\}$ (essentially creating a clique of all intermediary and
 699 satellite vertices, of size $m_1 + n_1$). Let us refer to these last edges as the background edges.
 700 This concludes the construction of G_2 . Note that $n_2 = m_1 + 2n_1$. Concerning λ_2 (see also
 701 Figure 8), for each vertex $v_a \in V(G_1)$ with $1 \leq a \leq n_1$, we create a temporal graph \mathcal{G}_{v_a}
 702 using labels $3a - 2$ and $3a - 1$. Afterwards, we take the union of these temporal graphs and
 703 add some extra labels to all edges to obtain \mathcal{G}_2 . For now, construct each \mathcal{G}_{v_a} by placing label
 704 $3a - 1$ on all edges $\{v_i, v_{i,a}\}$ and on the background edges, and label $3a - 2$ on all other
 705 edges. Note that the intermediary vertices allow for v_i and v_j to reach each other, except for
 706 temporal graphs \mathcal{G}_{v_i} in which only v_i can reach v_j , and vice versa for \mathcal{G}_{v_j} . Now, let \mathcal{G}_2 be
 707 the union of all \mathcal{G}_{v_a} , for all $1 \leq a \leq n_1$, and add labels $3a$ on all edges, for all $1 \leq a \leq n_1 - 1$.
 708 This concludes the polynomial-time transformation.



■ **Figure 8** Example of transformation from a CLIQUE instance graph G_1 to a \mathcal{S}^D COMPONENT instance temporal graph \mathcal{G}_2 . For visibility, for all \mathcal{G}_{v_a} , blue full edges correspond to edges with label $3a - 2$, red dashed edges to edges with label $3a - 1$, and orange dotted edges to background edges with label $3a - 1$. Also for visibility, not all background edges are represented.

709 Now let's show that the instance of CLIQUE, being (G_1, k_1) , is positive if and only if the
710 instance of $\mathcal{S}^{\mathcal{D}}$ COMPONENT, being $(\mathcal{G}_2, \Delta_2, k_2)$, is positive.

711 **(G_1, k_1) is positive $\implies (\mathcal{G}_2, \Delta_2, k_2)$ is positive:**

712 Suppose that a clique exists in G_1 of size k_1 , composed of vertices $V' = v_h, v_{h+1}, \dots, v_{h+k_1}$.
713 Note that in all windows of size $\Delta_2 = 2$ in \mathcal{G}_2 , either the window contains $3a$ for some a and
714 thus all vertices can reach each other in this window using only edges with label $3a$, or the
715 window does not contain $3a$ in which case it must contain $3a - 1$, meaning $m_1 + n_1$ vertices
716 can reach each other by using the background edges with label $3a - 1$. In either case, this
717 implies a $\mathcal{S}^{\mathcal{D}}$ component of size at least $m_1 + n_1$ must exist in \mathcal{G}_2 . To prove a larger $\mathcal{S}^{\mathcal{D}}$
718 component exists, of size $k_2 = m_1 + n_1 + k_1$, we can ignore windows containing $3a$, since in
719 these windows a \mathcal{S} component of size $n_2 \geq k_2$ exists and can thus be shrunk to adapt to any
720 $\mathcal{S}^{\mathcal{D}}$ component suitable for the other windows. The remaining windows correspond exactly
721 to the temporal graphs \mathcal{G}_{v_i} for all $1 \leq i \leq n_1$. Now, all edges $\{v_i, v_j\}$ composing our clique
722 in G_1 exist by definition as edges $\{v_i, v_{i,j}\}$ and $\{v_{i,j}, v_j\}$, with either labels 1 and 1 *resp.* or
723 1 and 2 *resp.*, or 2 and 1 *resp.* in temporal graphs \mathcal{G}_{v_a} . Let us iterate over these temporal
724 graphs (with variable b), in which two cases are distinguished:

- 725 ■ If the edges of the clique in G_1 are all transformed in edges with label 1 in the temporal
726 graph \mathcal{G}_{v_b} , then vertex v_h is a source able to reach V' in \mathcal{G}_{v_b} , for a total of at least
727 $m_1 + n_1 + k_1 = k_2$ vertices.
- 728 ■ If some edges of the clique in G_1 are transformed into edges with label 2 in the temporal
729 graph \mathcal{G}_{v_b} , then v_b must be a part of the clique in G_1 , since by definition only adjacent
730 edges to v_b are transformed into edges with label 2 in \mathcal{G}_{v_b} (outside of background edges).
731 Also, v_b (and only v_b amongst the original vertices) corresponds to a source able to reach
732 V' in \mathcal{G}_{v_b} , for a total of at least $m_1 + n_1 + k_1 = k_2$ vertices.

733 In both cases, and thus for all \mathcal{G}_{v_a} , a \mathcal{S} component exists containing the aforementioned
734 $m_1 + n_1$ vertices, as well as the k_1 vertices from the clique in G_1 . The result of a $\mathcal{S}^{\mathcal{D}}$
735 component of size $m_1 + n_1 + k_1 = k_2$ existing in \mathcal{G}_2 follows directly.

736 **$(\mathcal{G}_2, \Delta_2, k_2)$ is positive $\implies (G_1, k_1)$ is positive:**

737 Again, observe that any $\mathcal{S}^{\mathcal{D}}$ component in \mathcal{G}_2 must contain the $m_1 + n_1$ vertices which the
738 background edges cover. Let us look at the other $k_2 - (m_1 + n_1) = k_1$ vertices. These vertices
739 must be v_i for some i (since vertices $v_{i,j}$ are already contained through the background edges),
740 meaning they correspond directly to vertices v_i of G_1 . Let's denote these vertices as V' .
741 Consider temporal graph \mathcal{G}_{v_b} for some v_b , and observe that any vertex v_i can, by construction,
742 only reach other vertices v_j by traversing intermediary vertices $v_{i,j}$. Also, at most one original
743 vertex v_j can be reached by a journey starting at a vertex v_i . This implies all vertices in V'
744 must be linked pairwise through intermediary vertices in \mathcal{G}_{v_b} . By construction, intermediary
745 vertices between vertices V' can only exist in \mathcal{G}_{v_b} if the corresponding edges exist between
746 V' in G_1 . Thus a clique of size $|V'| = k_1$ must exist in G_1 . ◀

747 ▶ **Theorem 25.** *The worst-case number of closed $\mathcal{S}^{\mathcal{D}}$ components is at least n , and at most*
748 *$\min(2^n, n^{T-\Delta+1})$.*

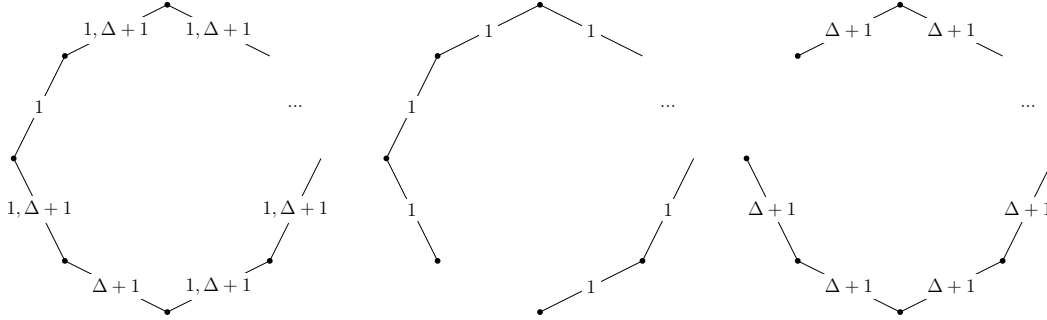
749 **Proof.** The lower bound from Theorem 22 works as $\mathcal{S}^{\mathcal{D}}$ components of size 1 are closed. The
750 upper bounds transfer directly from Theorem 22. ◀

751 ► **Lemma 60.** *Verifying a solution vertex subset V' of size k for CLOSED $\mathcal{S}^{\mathcal{D}}$ COMPONENT*
 752 *takes time $O((T - \Delta)k(\min(m, k^2) \log \Delta + k \log k))$.*

753 **Proof.** For each window $[t, t + \Delta - 1]$, of which there are $T - \Delta$, constructing the transitive
 754 closure of $\mathcal{G}[V']_{[t, t + \Delta - 1]}$ takes time $O(k(\min(m, k^2) \log \Delta + k \log k))$. In each transitive closure,
 755 checking if a vertex with outdegree $k - 1$ exists takes time $O(k^2)$. In total, this takes time
 756 $O((T - \Delta)(k(\min(m, k^2) \log \Delta + k \log k) + k^2)) = O(k(T - \Delta)(\min(m, k^2) \log \Delta + k \log k))$. ◀

757 ► **Lemma 61.** *The existence of a closed $\mathcal{S}^{\mathcal{D}}$ component of size $> k$ does not necessarily*
 758 *imply the existence of a (non-maximal) closed $\mathcal{S}^{\mathcal{D}}$ component of size k .*

759 **Proof.** Consider Figure 9, in which an infinite family of temporal graphs is presented. For
 760 each of these graphs, say \mathcal{G} , a closed $\mathcal{S}^{\mathcal{D}}$ component exists of size n , with any vertex as
 761 source for both windows. However, no (non-maximal) closed $\mathcal{S}^{\mathcal{D}}$ component exists of size
 762 $n - 1$, since considering any such set of vertices V' would result in either $G(\mathcal{G}[V']_{[1, \Delta]})$ or
 763 $G(\mathcal{G}[V']_{[2, \Delta + 1]})$ to be disconnected, implying no temporal source can exist in that window.



764 **Figure 9** Temporal graph family in which a closed $\mathcal{S}^{\mathcal{D}}$ component of size n exists but no non-
 765 maximal component of size $n - 1$ exists. The temporal graphs corresponding to time window $[1, \Delta]$
 766 and time window $[2, \Delta + 1]$ are shown in the middle and on the right respectively.

767 ► **Theorem 26.** *CLOSED $\mathcal{S}^{\mathcal{D}}$ COMPONENT is solvable in time $O(\min(2^n, n^{T - \Delta + 1})(T - \Delta)n(m \log \Delta + n \log n))$.*

768 **Proof.** Due to Theorem 61, a brute force algorithm cannot simply verify all possible subsets
 769 of size k , as a closed $\mathcal{S}^{\mathcal{D}}$ component of size $> k$ may still exist even if one of size k
 770 does not. Hence, all possible subsets of size at least k will have to be checked, of which
 771 there are $O(\min(2^n, n^{T - \Delta + 1}))$ by Theorem 25. Verifying a solution takes time $O((T - \Delta)k(\min(m, k^2) \log \Delta + k \log k))$ by Theorem 60, meaning the brute force algorithm runs in
 772 time $O(\min(2^n, n^{T - \Delta + 1})(T - \Delta)n(m \log \Delta + n \log n))$. ◀

773 ► **Theorem 27.** *CLOSED $\mathcal{S}^{\mathcal{D}}$ COMPONENT is NP-complete.*

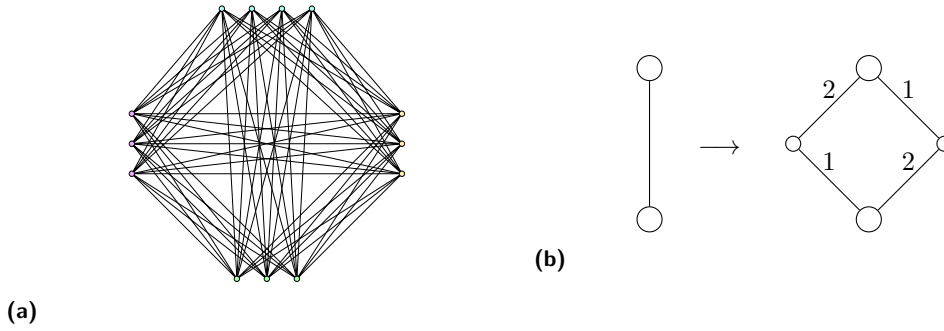
774 **Proof.** To prove CLOSED $\mathcal{S}^{\mathcal{D}}$ COMPONENT is in NP, we show how to verify a solution $V' \subseteq V$
 775 in polynomial time. For every window $[t, t + \Delta - 1]$, compute the transitive closure of
 776 temporal graph $\mathcal{G}[V']_{[t, t + \Delta - 1]}$ and check if at least one source, *i.e.* a vertex able to reach all
 777 others, exists. If this is the case in all windows, then V' is a solution, and else it is not.

778 To prove NP-hardness, the same reduction as in Theorem 24 is used, since all the $\mathcal{S}^{\mathcal{D}}$
 779 components in the transformed instance of the reduction are closed $\mathcal{S}^{\mathcal{D}}$ components. ◀

780 **A.4 \mathcal{TC} components**

781 ▶ **Theorem 32.** *The worst-case number of \mathcal{TC} components is at least $2^{0.52\sqrt{n}}$, and at most*
 782 *$2^{0.53n}$.*

783 **Proof.** The lower bound has been obtained by Casteigts *et al.* [8] by adapting a Moon and
 784 Moser graph [41] on n vertices and $\binom{n}{2} - n$ edges admitting $3^{n/3}$ cliques. Every edge of
 785 this graph is replaced by a semaphore gadget (see Figure 10) creating a temporal graph on
 786 $N = n^2 - 2n$ vertices in which every initial clique is now a \mathcal{TC} component, obtaining a total
 787 of $3^{n/3} = 3^{(\sqrt{N+1+1})/3} > 3^{\sqrt{N}/3} > 2^{0.52\sqrt{N}}$ \mathcal{TC} components.



■ **Figure 10** A graph with at least $2^{0.52\sqrt{n}}$ \mathcal{TC} components is constructed by taking a Moon and Moser graph (left) admitting $3^{n/3}$ cliques, and replacing each edge with a semaphore gadget (right). All initial cliques now correspond to \mathcal{TC} components, which are maximal when including semaphore vertices.

788 Outside of the trivial upper bound of 2^n , a better upper bound can be obtained as follows.
 789 Since \mathcal{TC} components of \mathcal{G} correspond to bidirectional cliques in the transitive closure of \mathcal{G}
 790 (see [6]), modify the transitive closure as follows. For all arcs (u, v) such that (v, u) exists
 791 as well, replace both arcs by one (undirected) edge $\{u, v\}$. All other arcs are removed. The
 792 corresponding graph is undirected and cliques in this graph correspond to \mathcal{TC} components.
 793 By Moon and Moser [41], at most $3^{n/3}$ cliques can exist in undirected graphs, which implies
 794 the same upper bound $3^{n/3} < 2^{0.53n}$ holds for \mathcal{TC} components in temporal graphs. ◀

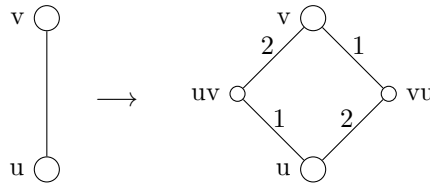
795 ▶ **Theorem 33.** *\mathcal{TC} COMPONENT is solvable in time $O(nm \log T + \min(n^k k^2, 2^{0.25n}))$.*

796 **Proof.** The algorithm starts by constructing the transitive closure of the input temporal
 797 graph. This is done in time $O(n(m \log T + n \log n))$. Then, it searches for bidirectional cliques
 798 (a subset of vertices with arcs in both directions between vertices) in the transitive closure.
 799 This has the same asymptotic complexity as searching for cliques. Since we're interested in
 800 cliques of size at least k , the brute force algorithm of testing each subset of size k runs in time
 801 $O(n^k k^2)$. Also, one can obtain a maximum-size clique in time $O(2^{n/4})$ [43]. All together,
 802 \mathcal{TC} COMPONENT can thus be solved in time $O(n(m \log T + n \log n) + \min(n^k k^2, 2^{n/4})) =$
 803 $O(nm \log T + \min(n^k k^2, 2^{0.25n}))$. ◀

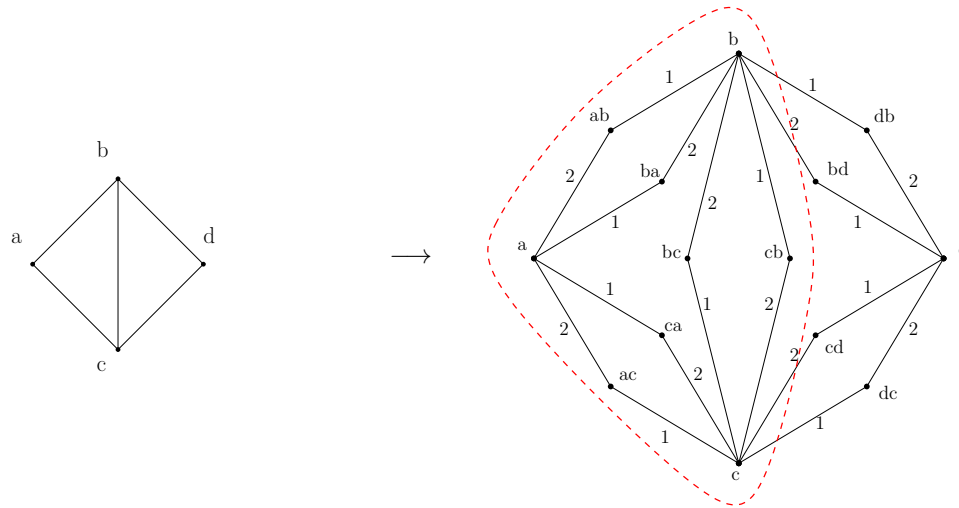
804 ▶ **Theorem 34.** *\mathcal{TC} COMPONENT is NP-complete, for all constant lifetimes $T > 1$.*

805 **Proof.** We use a specific version of CLIQUE, in which one has to decide if a clique of size
 806 $k = \frac{n}{3}$ exists in a given graph on n vertices. This specific problem was indirectly proven
 807 NP -hard by Erickson [20] through his reduction from 3-SAT to INDEPENDENTSET (in [32],
 808 Karp reduces SAT to CLIQUE in a similar fashion). Let us refer to this problem as $\frac{n}{3}$ CLIQUE.

809 We show how to reduce $\frac{n}{3}$ CLIQUE to \mathcal{TC} COMPONENT in polynomial time. Given a graph
 810 $G = (V, E)$ for the $\frac{n}{3}$ CLIQUE problem, create temporal graph \mathcal{G} with (initially) the same
 811 vertex set V . For each edge $(u, v) \in E$, add a semaphore gadget between u and v in \mathcal{G} , with
 812 semaphore “intermediary” vertices uv and vu (see Figure 11). See Figure 12 for an example
 813 of how to construct \mathcal{G} .



■ **Figure 11** An edge is replaced by a semaphore gadget, adding two intermediary vertices.



■ **Figure 12** On the left an example instance of CLIQUE with a clique of size 3 which, when transformed as shown on the right, allows for a \mathcal{TC} component of size 9 (shown in red and dashed).

814 We prove a clique of size $k = \frac{n}{3}$ exists in G if and only if a \mathcal{TC} component of size $k^2 = \frac{n^2}{9}$
 815 exists in \mathcal{G} .

816 $\frac{n}{3}$ **Clique** \implies **\mathcal{TC} Component**

817 If a clique of size k exists in G of vertices $K = \{v_1, v_2, \dots, v_k\}$, then consider the corresponding
 818 vertices in \mathcal{G} , as well as intermediary vertices $\{v_i v_j : v_i, v_j \in K, v_i \neq v_j\}$. Let’s refer to this
 819 set of vertices in \mathcal{G} as V' , which is of size $k + 2\frac{k(k-1)}{2} = k^2$. V' is a \mathcal{TC} component, since all
 820 vertices can reach each other:

- 821 ■ a vertex v_i can reach all other vertices v_j through journey $(v_i, v_i v_j, v_j)$ using time steps
 822 $(1, 2)$ respectively;
- 823 ■ a vertex v_i can reach all intermediary vertices $v_j v_k$, through journey $(v_i, v_i v_k, v_k, v_j v_k)$
 824 with time steps $(1, 2, 2)$ respectively;

825 ■ an intermediary vertex $v_i v_j$ can reach all vertices v_k through journey $(v_i v_j, v_i, v_i v_k, v_k)$
 826 using time steps $(1, 1, 2)$ respectively;

827 ■ an intermediary vertex $v_i v_j$ can reach all other intermediary vertices $v_k v_\ell$ through journey
 828 $(v_i v_j, v_i, v_i v_\ell, v_\ell, v_k v_\ell)$ using time steps $(1, 1, 2, 2)$ respectively.

829 **\mathcal{TC} Component $\implies \frac{n}{3}$ Clique**

830 If a \mathcal{TC} component V' of size $k^2 = \frac{n^2}{9}$ exists in \mathcal{G} , then consider all vertices $K = V' \cap V$.
 831 Trivially $|K| > 0$, furthermore $|K| > 1$ since otherwise the lone vertex $v \in K$ is part of a
 832 \mathcal{TC} component of size at most $2\Delta + 1$ (including all adjacent intermediary vertices), which
 833 leads to the asymptotic contradiction $k^2 = \frac{n^2}{9} \leq 2\Delta + 1 < 2n + 1$. We prove that K forms a
 834 clique of size k in G :

835 ■ for all pairs of vertices $v_i, v_j \in K$, intermediary vertex $v_i v_j$ (*resp.* $v_j v_i$) must be included
 836 in V' , since otherwise no journey $v_i \rightsquigarrow v_j$ (*resp.* $v_j \rightsquigarrow v_i$) exists in V' ;

837 ■ for all $v_k \notin K$, vertices $v_k v_\ell$ (*resp.* $v_\ell v_k$) cannot reach (*resp.* be reached by) vertices $v_i v_j$
 838 where $v_i, v_j \in V'$, thus $v_k v_\ell \notin V'$ (*resp.* $v_\ell v_k \notin V'$);

839 ■ since V' is composed solely of vertices K and all their corresponding intermediary vertices,
 840 we have that $|V'| = k^2 = |K| + 2 \frac{|K|(|K|-1)}{2} = |K|^2$, leading to $|K| = k$;

841 ■ finally, since all intermediary vertices of K are present in V' which can only be created
 842 through a semaphore gadget, which in turn can only be created if a corresponding edge
 843 is present in G , K forms a clique in G .

844 ◀

845 ▶ **Theorem 35.** *The worst-case number of closed \mathcal{TC} components is at least $2^{0.52\sqrt{n}}$, and at*
 846 *most $2^{0.53n}$.*

847 **Proof.** Theorem 32 directly adapts for closed components, as the components obtained by
 848 the lower bound's construction are closed. ◀

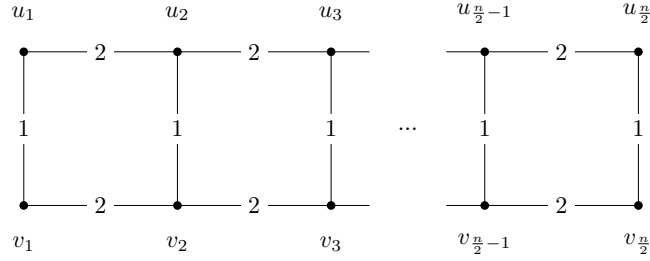
849 ▶ **Lemma 62.** *Verifying a solution vertex subset V' of size k for CLOSED \mathcal{TC} COMPONENT*
 850 *takes time $O(k(\min(m, k^2) \log T + k \log k))$.*

851 **Proof.** To verify a solution V' , we construct the transitive closure of temporal graph $\mathcal{G}[V']$
 852 and check if it is a (bidirectional) complete graph, doable in time $O(k(\min(m, k^2) \log T +$
 853 $k \log k) + k^2) = O(k(\min(m, k^2) \log T + k \log k))$. ◀

854 ▶ **Lemma 63.** *The existence of a closed \mathcal{TC} component of size $> k$ does not necessarily*
 855 *imply the existence of a (non-maximal) closed \mathcal{TC} component of size k .*

856 **Proof.** Consider Figure 13, in which an infinite family of temporal graphs is presented.
 857 For each of these graphs, say \mathcal{G} , a closed \mathcal{TC} component exists of size n . However, no
 858 (non-maximal) closed \mathcal{TC} component exists of size $n - 1$, since considering any such set of
 859 vertices $V \setminus u_i$ (*resp.* v_i) would result in v_i (*resp.* u_i) not being able to reach any vertex v_j
 860 (*resp.* u_j).

861 ◀



■ **Figure 13** Temporal graph family in which a closed \mathcal{TC} component of size n exists but no non-maximal component of size $n - 1$ exists.

862 ► **Theorem 36.** *CLOSED \mathcal{TC} COMPONENT is solvable in time $O(2^n n(m \log T + n \log n))$.*

863 **Proof.** We describe the brute force algorithm of testing all possible subsets of vertices. Due to
 864 Theorem 63, we cannot only test subsets of size k , needing to possibly test all subsets of size at
 865 least k , of which there are $O(2^n)$. Testing a vertex subset takes time $O(n(m \log T + n \log n))$
 866 by Theorem 62, giving a total time complexity of $O(2^n n(m \log T + n \log n))$.

867

868 ► **Theorem 37.** *CLOSED \mathcal{TC} COMPONENT is NP-complete, for all constant lifetimes $T > 1$.*

869 **Proof.** By Theorem 62, a solution vertex subset can be verified in time $O(n(m \log T + n \log n))$,
 870 and thus CLOSED \mathcal{TC} COMPONENT is in NP.

871 In the construction from Theorem 34, all \mathcal{TC} components in \mathcal{G} are closed. This suffices to
 872 prove that OPEN \mathcal{TC} COMPONENT is NP-hard as well, for all lifetimes $T > 1$.

873 A.5 \mathcal{TC}^B components

874 ► **Theorem 39.** *The worst-case number of \mathcal{TC}^B components is at least $2^{0.52\sqrt{n}}$ and at most
 875 $2^{0.53n}$.*

876 **Proof.** The lower bound transfers directly from Theorem 32.

877 The upper bound is obtained in a similar manner as for Theorem 32. Start by creating an
 878 adaptation of the transitive closure, in which an arc (u, v) is present if and only if a journey
 879 exists from u to v in every window of size Δ . (This can be computed easily using temporal
 880 Dijkstra algorithms from [46] in each windowed graph.) Then, for all arcs (u, v) such that
 881 (v, u) exists as well, replace both arcs by one (undirected) edge $\{u, v\}$. All other arcs are
 882 removed. The corresponding graph is undirected and cliques in this graph correspond to
 883 \mathcal{TC}^B components. By Moon and Moser [41], at most $3^{n/3}$ cliques can exist in undirected
 884 graphs, which implies the same upper bound $3^{n/3} < 2^{0.53n}$ holds for \mathcal{TC}^B components in
 885 temporal graphs.

886 ► **Theorem 40.** *\mathcal{TC}^B COMPONENT is solvable in time $O((T - \Delta)n(m \log \Delta + n \log n) +$
 887 $\min(n^k k^2, 2^{0.25n}))$.*

888 **Proof.** For all windowed temporal graphs $\mathcal{G}_{[t,t+\Delta-1]}$, construct transitive closures $H_{[t,t+\Delta-1]}$,
 889 taking time $O((T - \Delta)(n(m \log \Delta + n \log n)))$. Afterwards, take the intersection of these
 890 transitive closures, denoted by H . An arc (u, v) in H means that u can reach v in all
 891 windowed temporal graphs. A bidirectional clique of size k in H thus represents a $\mathcal{TC}^{\mathcal{B}}$
 892 component of size k . As described in Theorem 33, such a clique of size at least k can be
 893 found in time $O(\min(n^k k^2, 2^{n/4}))$. All in all, $\mathcal{TC}^{\mathcal{B}}$ COMPONENT can thus be solved in time
 894 $O((T - \Delta)(n(m \log \Delta + n \log n)) + \min(n^k k^2, 2^{0.25n}))$. ◀

895 ▶ **Theorem 41.** $\mathcal{TC}^{\mathcal{B}}$ COMPONENT is NP-complete, for all constant lifetimes $T > 1$ and
 896 window sizes $\Delta > 1$.

897 **Proof.** We prove that instances of \mathcal{TC} COMPONENT are equivalent to firstly, instances
 898 of $\mathcal{TC}^{\mathcal{B}}$ COMPONENT such that Δ is arbitrarily larger than T , and secondly, instances of
 899 $\mathcal{TC}^{\mathcal{B}}$ COMPONENT such that Δ is arbitrarily smaller than T . We also prove these cover the
 900 precised bounds of $T > 1$ and $\Delta > 1$.

901 Concerning $\Delta \geq T$, we trivially have that any instance (\mathcal{G}, k) for \mathcal{TC} COMPONENT, where
 902 the lifetime of \mathcal{G} is T , is equivalent to instance $(\mathcal{G}, k, \Delta \geq T)$ for $\mathcal{TC}^{\mathcal{B}}$ COMPONENT. This
 903 combined with Theorem 34 proves that $\mathcal{TC}^{\mathcal{B}}$ COMPONENT is NP-hard for all $T > 1$ and all
 904 $\Delta > 1$ such that $\Delta \geq T$.

905 Concerning $\Delta < T$, any instance (\mathcal{G}, k) for \mathcal{TC} COMPONENT, where the lifetime of \mathcal{G} is T , is
 906 equivalent to instance $(\mathcal{G}', k, \Delta = T)$ for $\mathcal{TC}^{\mathcal{B}}$ COMPONENT, where \mathcal{G}' corresponds to \mathcal{G} with
 907 some additional complete snapshots, i.e. $\mathcal{G}' = \mathcal{G} \frown (G_i = K_n)_{i=T+1}^j$ for some arbitrary $j > T$.
 908 Again with Theorem 34, this proves that $\mathcal{TC}^{\mathcal{B}}$ COMPONENT is NP-hard for all $T > 1$ and
 909 all $\Delta > 1$ such that $\Delta < T$. ◀

910 ▶ **Theorem 42.** The worst-case number of closed $\mathcal{TC}^{\mathcal{B}}$ components is at least $2^{0.52\sqrt{n}}$ and at
 911 most $2^{0.53n}$.

912 **Proof.** The lower bound from Theorem 32 transfers since the construction produces closed
 913 \mathcal{TC} components, which are also closed $\mathcal{TC}^{\mathcal{B}}$ components for $\Delta = T$. The upper bounds from
 914 Theorem 39 transfer directly. ◀

915 ▶ **Lemma 64.** Verifying if a vertex subset V' of size k is a solution for CLOSED $\mathcal{TC}^{\mathcal{B}}$ COM-
 916 PONENT takes time $O((T - \Delta)k(\min(m, k^2) \log \Delta + k \log k))$.

917 **Proof.** For each window $[t, t + \Delta - 1]$, compute the transitive closure $H_{[t,t+\Delta-1]}$ of temporal
 918 graph $\mathcal{G}[V']_{[t,t+\Delta-1]}$. V' is a solution for CLOSED $\mathcal{TC}^{\mathcal{B}}$ COMPONENT if and only if all
 919 $H_{[t,t+\Delta-1]}$ are bidirectional complete graphs. ◀

920 ▶ **Theorem 43.** CLOSED $\mathcal{TC}^{\mathcal{B}}$ COMPONENT is solvable in time $O(2^n(T - \Delta)n(m \log \Delta +$
 921 $n \log n))$.

922 **Proof.** We present the brute force algorithm. Due to Theorem 63, we know it may not be
 923 sufficient to verify only the subsets of size exactly k ; we may need to verify all subsets of
 924 size at least k , of which there exist $O(2^n)$. By Theorem 64, verifying such a subset requires
 925 $O((T - \Delta)n(m \log \Delta + n \log n))$ time. In total, the brute force algorithm thus takes time
 926 $O(2^n(T - \Delta)n(m \log \Delta + n \log n))$. ◀

927 ► **Theorem 44.** *CLOSED \mathcal{TC}^B COMPONENT is NP-complete, for all constant lifetimes $T > 1$*
 928 *and window sizes $\Delta > 1$.*

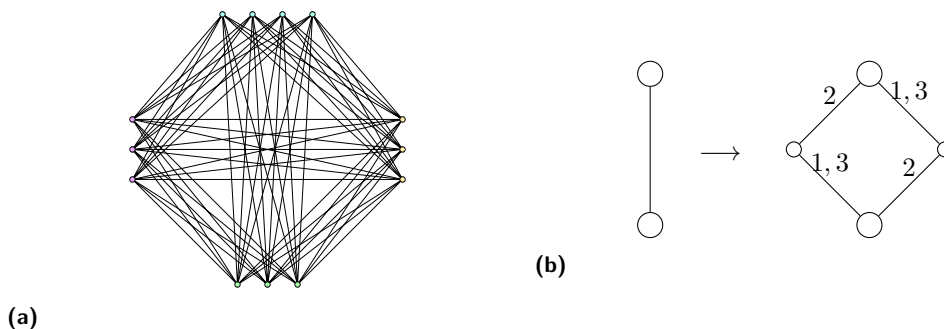
929 **Proof.** By Theorem 64, a vertex subset can be verified to be a solution for CLOSED \mathcal{TC}^B COM-
 930 PONENT in time $O((T - \Delta)n(m \log \Delta + n \log n))$. CLOSED \mathcal{TC}^B COMPONENT is thus in
 931 NP.

932 In the reduction used in Theorem 41, all \mathcal{TC}^B components in the transformed instance are
 933 closed. This suffices to prove that CLOSED \mathcal{TC}^B COMPONENT is NP-hard, for any constant
 934 lifetime $T > 1$ and window size $\Delta > 1$. ◀

935 A.6 \mathcal{TC}° components

936 ► **Theorem 46.** *The worst-case number of \mathcal{TC}° components is at least $2^{0.52\sqrt{n}}$, and at most*
 937 *$2^{0.53n}$.*

938 **Proof.** The lower bound can be obtained as follows. Take a Moon and Moser graph [41] on
 939 n vertices and $\binom{n}{2} - n$ edges admitting $3^{n/3}$ cliques. Every edge of this graph is replaced
 940 by a semaphore gadget (see Figure 10) creating a temporal graph on $N = n^2 - 2n$ vertices
 941 in which every initial clique is now a \mathcal{TC}° component, obtaining a total of $3^{n/3} > 2^{0.52\sqrt{N}}$
 942 \mathcal{TC}° components.



■ **Figure 14** A graph with at least $2^{0.52\sqrt{n}}$ \mathcal{TC}° components is constructed by taking a Moon and Moser graph (left) admitting $3^{n/3}$ cliques, and replacing each edge with a semaphore gadget (right). All initial cliques now correspond to \mathcal{TC}° components, which are maximal when including semaphore vertices.

943 Outside of the trivial upper bound of 2^n , a better upper bound can be obtained as follows.
 944 Consider the following adaptation of the transitive closure: instead of adding an arc between
 945 vertices u and v when a journey exists, add an arc when a round trip exists. (Detecting round
 946 trips between all pairs of vertices is easily done adapting the temporal Dijkstra algorithm
 947 from [46].) By definition, \mathcal{TC}° components of \mathcal{G} correspond to bidirectional cliques in this
 948 adapted transitive closure of \mathcal{G} . Further modify the transitive closure as follows. For all arcs
 949 (u, v) such that (v, u) exists as well, replace both arcs by one (undirected) edge $\{u, v\}$. All
 950 other arcs are removed. The corresponding graph is undirected and cliques in this graph
 951 correspond to \mathcal{TC}° components. By Moon and Moser [41], at most $3^{n/3}$ cliques can exist
 952 in undirected graphs, which implies the same upper bound $3^{n/3} < 2^{0.53n}$ holds for \mathcal{TC}°
 953 components in temporal graphs. ◀

954 ► **Theorem 47.** *\mathcal{TC}° COMPONENT is solvable in time $O(n^2(m \log T + n \log n) + \min(n^k k^2, 2^{0.25n}))$.*

955 **Proof.** We start by constructing a modified version of the transitive closure H , where an
 956 arc (u, v) represents $u \circ v$ instead of $u \rightsquigarrow v$. Initialize H with vertex set V . For all pairs of
 957 vertices, of which there are $O(n^2)$, do the following. Apply the temporal Dijkstra algorithm
 958 from [46] starting from u to obtain the smallest label at which v is reached from u through
 959 a journey, suppose label t . If v is unreachable from u , suppose $t = \infty$. This takes time
 960 $O(m \log T + n \log n)$. Then, if $t \neq \infty$, on temporal graph $\mathcal{G}_{[t, T]}$, apply again the temporal
 961 Dijkstra algorithm to check if u is reachable from v . This again takes time $O(m \log T + n \log n)$.
 962 If $t \neq \infty$ and if u is reachable from v in $\mathcal{G}_{[t, T]}$, then add arc (u, v) to H . Apply the same
 963 process in the other direction to check if $v \circ u$. After having done this for all pairs of vertices,
 964 modify H as follows. For all arcs (u, v) such that (v, u) exists as well, replace both arcs by
 965 an (undirected) edge $\{u, v\}$. Remove all other arcs. H is now an undirected graph in which
 966 cliques correspond to \mathcal{TC}° components. Since we're interested in cliques of size at least k ,
 967 the brute force algorithm of testing each subset of size k runs in time $O(n^k k^2)$. Also, one
 968 can obtain a maximum-size clique in time $O(2^{n/4})$ [43]. All together, \mathcal{TC}° COMPONENT can
 969 thus be solved in time $O(n^2(m \log T + n \log n) + \min(n^k k^2, 2^{0.25n}))$. ◀

970 ▶ **Theorem 48.** \mathcal{TC}° COMPONENT is solvable in polynomial time $O(n + m)$ on temporal
 971 graphs with constant lifetimes $T \leq 2$.

972 **Proof.** The case of $T = 1$ trivially reduces to finding connected components in the snapshot,
 973 doable in time $O(n + m)$.

974 For the case of $T = 2$, note that three possible \mathcal{TC}° components can exist. The first (*resp.*
 975 second) type of \mathcal{TC}° components are those which use only labels 1 (*resp.* 2). These \mathcal{TC}°
 976 components correspond to connected components in the corresponding snapshot, and can
 977 thus be computed in time $O(n + m)$ as well. The third, and final, type of \mathcal{TC}° components
 978 use both labels 1 and 2. We continue by proving such \mathcal{TC}° components do not exist, finishing
 979 the proof.

980 By contradiction, let us suppose one such a \mathcal{TC}° component exists, say V' . For any two
 981 vertices $u, v \in V'$, if a journey using only labels 1 (*resp.* 2) does not exist between u and v ,
 982 then a journey using only labels 2 (*resp.* 1) must exist between u and v (or V' would not be
 983 a \mathcal{TC}° component). Let us simplify this as a graph G with a copy of V' in which an edge
 984 with label i is drawn between vertices u and v if the corresponding journey between u and v
 985 uses only labels i . G is thus a complete graph with some arbitrary labelling of one label per
 986 edge, being either 1 or 2. We finish by showing that in G there is necessarily a spanning
 987 structure using only label 1, or using only label 2, implying that in fact V' is not a \mathcal{TC}°
 988 component needing both labels 1 and 2, as a spanning structure using only labels 1 or only
 989 labels 2 exists which is sufficient for \mathcal{TC}° .

990 **G contains a spanning structure using only labels 1 or only labels 2:**

991 We prove this property by induction. A K_2 graph trivially admits such a spanning structure.
 992 Now, if a K_n graph has this property, suppose *w.l.o.g.* that the spanning structure has only
 993 labels 1. Adding a vertex so as to obtain a K_{n+1} implies adding n edges connecting this
 994 vertex to the other vertices. To avoid having this property in the newly constructed K_{n+1} ,
 995 none of these edges can have label 1 (since otherwise a spanning structure with only labels 1
 996 exists), but they cannot all have label 2 either (since otherwise a spanning structure with
 997 only labels 2 exists), which is impossible. ◀

998 ▶ **Theorem 49.** \mathcal{TC}° COMPONENT is NP-complete, for all constant lifetimes $T > 2$.

999 **Proof.** We show how to reduce $\frac{n}{3}$ CLIQUE to \mathcal{TC}^\odot COMPONENT in polynomial time. Given
 1000 a graph $G = (V, E)$ for the $\frac{n}{3}$ CLIQUE problem, create temporal graph \mathcal{G} with (initially) the
 1001 same vertex set V . For each edge $\{u, v\} \in E$, add a semaphore gadget between u and v in
 1002 \mathcal{G} , with semaphore “intermediary” vertices uv and vu . On edges $\{u, uv\}$ and $\{v, vu\}$ place
 1003 labels 1 and 3, and on edges $\{u, vu\}$ and $\{v, uv\}$ place labels 2.

1004 We prove a clique of size $k = \frac{n}{3}$ exists in G if and only if a \mathcal{TC}^\odot component of size k^2 exists
 1005 in \mathcal{G} .

1006 $\frac{n}{3}$ Clique $\implies \mathcal{TC}^\odot$ Component

1007 If a clique of size k exists in G of vertices $K = \{v_1, v_2, \dots, v_k\}$, then consider the corresponding
 1008 vertices in \mathcal{G} , as well as intermediary vertices $\{v_i v_j : v_i, v_j \in K, v_i \neq v_j\}$. Let’s refer to this
 1009 set of vertices in \mathcal{G} as V' , which is of size $k + 2\frac{k(k-1)}{2} = k^2$. V' is a \mathcal{TC}^\odot component, since
 1010 all vertices can reach each other and then reach back:

- 1011 ■ a vertex v_i can reach all other vertices v_j and back through journey $(v_i, v_i v_j, v_j, v_i v_j, v_i)$
 1012 using time steps $(1, 2, 2, 3)$ respectively;
- 1013 ■ a vertex v_i can reach all intermediary vertices $v_j v_k$ and back through journey $(v_i, v_i v_k, v_k, v_j v_k, v_k, v_i v_k, v_i)$
 1014 with time steps $(1, 2, 2, 2, 2, 3)$ respectively;
- 1015 ■ an intermediary vertex $v_i v_j$ can reach all vertices v_k and back through journey $(v_i v_j, v_i, v_i v_k, v_k, v_i v_k, v_i, v_i v_j)$
 1016 using time steps $(1, 1, 2, 2, 3, 3)$ respectively;
- 1017 ■ an intermediary vertex $v_i v_j$ can reach all other intermediary vertices $v_k v_\ell$ and back
 1018 through journey $(v_i v_j, v_i, v_i v_\ell, v_\ell, v_k v_\ell, v_\ell, v_i v_\ell, v_i, v_i v_j)$ using time steps $(1, 1, 2, 2, 2, 2, 3, 3)$
 1019 respectively.

1020 \mathcal{TC}^\odot Component $\implies \frac{n}{3}$ Clique

1021 If a \mathcal{TC}^\odot component V' of size $k^2 = \frac{n^2}{9}$ exists in \mathcal{G} , then consider all vertices $K = V' \cap V$.
 1022 Trivially $|K| > 0$, furthermore $|K| > 1$ since otherwise the lone vertex $v \in K$ is part of a
 1023 \mathcal{TC}^\odot component of size at most $2\Delta + 1$ (including all adjacent intermediary vertices), which
 1024 leads to the asymptotic contradiction $k^2 = \frac{n^2}{9} \leq 2\Delta + 1 < 2n + 1$. We prove that K forms a
 1025 clique of size k in G :

- 1026 ■ for all pairs of vertices $v_i, v_j \in K$, intermediary vertex $v_i v_j$ (*resp.* $v_j v_i$) must be included
 1027 in V' , since otherwise no round trip $v_i \odot v_j$ (*resp.* $v_j \odot v_i$) exists in V' ;
- 1028 ■ for all intermediary vertices $v_k v_j : v_j \in K, v_k \notin K$ (thus having label 2), round trip
 1029 $v_k v_j \odot v_i$, where $v_i \in K$ isn’t possible without v_k , *i.e.*, vertices $v_k v_j$ cannot be included
 1030 in V' ;
- 1031 ■ for all intermediary vertices $v_j v_k : v_j \in K, v_k \notin K$ (thus having labels 1 and 3), round trip
 1032 $v_i \odot v_j v_k$, where $v_i \in K$ isn’t possible without v_k , *i.e.*, vertices $v_j v_k$ cannot be included
 1033 in V' ;
- 1034 ■ since V' is composed solely of vertices K and all their corresponding intermediary vertices,
 1035 we have that $|V'| = k^2 = |K| + 2\frac{|K|(|K|-1)}{2} = |K|^2$, leading to $|K| = k$;
- 1036 ■ finally, since all intermediary vertices of K are present in V' which can only be created
 1037 through a semaphore gadget, which in turn can only be created if a corresponding edge
 1038 is present in G , K forms a clique in G .

1039



1040 ► **Theorem 50.** *The worst-case number of closed \mathcal{TC}° components is at least $2^{0.52\sqrt{n}}$, and*
 1041 *at most $2^{0.53n}$.*

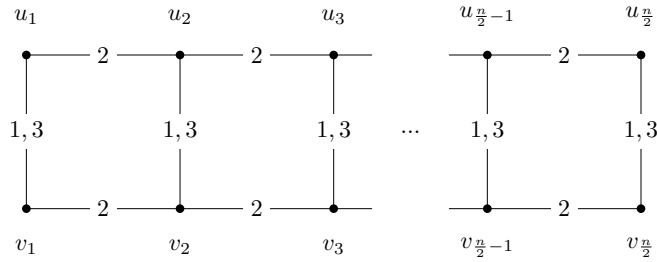
1042 **Proof.** The lower bound comes from the fact that all \mathcal{TC}° components in the construction
 1043 from Theorem 46 are closed, and the upper bound transfers directly from Theorem 46. ◀

1044 ► **Lemma 65.** *Verifying if a vertex subset V' of size k is a solution for CLOSED \mathcal{TC}° COM-*
 1045 *ONENT takes time $O(k^2(\min(m, k^2) \log T + k \log k))$.*

1046 **Proof.** Construct a modified transitive closure H in the following manner. In $\mathcal{G}[V']$, apply
 1047 for all pairs of vertices u and v , of which there are $O(k^2)$, the temporal Dijkstra algorithm
 1048 from [46], starting from vertex u to obtain the smallest label t at which u reaches v .
 1049 If u cannot reach v , then V' is not a solution vertex set. This can be done in time
 1050 $O(\min(m, k^2) \log T + k \log k)$. Then, check if v can reach u in temporal graph $\mathcal{G}[V']_{[t, T]}$,
 1051 doable again in time $O(\min(m, k^2) \log T + k \log k)$. If v cannot reach u , then V' is not
 1052 a solution vertex set. Else, it means $u \circ v$ in $\mathcal{G}[V']$. Apply the same process in the
 1053 other direction to check if $v \circ u$. When this is checked for all pairs of vertices, then
 1054 V' is a solution vertex set for CLOSED \mathcal{TC}° COMPONENT. In total, this thus takes time
 1055 $O(k^2(\min(m, k^2) \log T + k \log k))$. ◀

1056 ► **Lemma 66.** *The existence of a closed \mathcal{TC}° component of size $> k$ does not necessarily*
 1057 *imply the existence of a (non-maximal) closed \mathcal{TC}° component of size k .*

1058 **Proof.** Consider Figure 15, in which an infinite family of temporal graphs is presented.
 1059 For each of these graphs, say \mathcal{G} , a closed \mathcal{TC}° component exists of size n . However, no
 1060 (non-maximal) closed \mathcal{TC}° component exists of size $n - 1$, since considering any such set of
 1061 vertices $V \setminus u_i$ (*resp.* v_i) would result in v_i (*resp.* u_i) not being able to reach any vertex v_j
 1062 (*resp.* u_j) and back.



■ **Figure 15** Temporal graph family in which a closed \mathcal{TC}° component of size n exists but no non-maximal component of size $n - 1$ exists.

1063 ◀

1064 ► **Theorem 51.** *CLOSED \mathcal{TC}° COMPONENT is solvable in time $O(2^n(n^2(m \log T + n \log n)))$.*

1065 **Proof.** We present the brute force algorithm. Due to Theorem 66, we know it may not
 1066 be sufficient to verify only the subsets of size exactly k ; we may need to verify all subsets
 1067 of size at least k , of which there exist $O(2^n)$. By Theorem 65, verifying such a subset
 1068 requires $O(n^2(m \log T + n \log n))$ time. In total, the brute force algorithm thus takes time
 1069 $O(2^n(n^2(m \log T + n \log n)))$. ◀

1070 ► **Theorem 52.** *CLOSED \mathcal{TC}° COMPONENT is solvable in polynomial time $O(n + m)$ on*
 1071 *temporal graphs with constant lifetimes $T \leq 2$.*

1072 **Proof.** In Theorem 48, we prove that all \mathcal{TC}° components in temporal graphs of lifetime
 1073 $T \leq 2$ correspond to connected components in \mathcal{G}_1 or \mathcal{G}_2 . Connected components are
 1074 necessarily closed, thus all \mathcal{TC}° components in temporal graphs of lifetime $T \leq 2$ are
 1075 necessarily closed. Since \mathcal{TC}° COMPONENT can be solved in time $O(m + n)$ on such temporal
 1076 graphs, CLOSED \mathcal{TC}° COMPONENT can be solved in time $O(m + n)$ as well. ◀

1077 ► **Theorem 53.** *CLOSED \mathcal{TC}° COMPONENT is NP-complete, for all constant lifetimes*
 1078 *$T > 2$.*

1079 **Proof.** By Theorem 65, a vertex subset can be verified to be a solution for CLOSED \mathcal{TC}° COM-
 1080 PONENT in time $O(k^2(\min(m, k^2) \log T + k \log k))$. CLOSED \mathcal{TC}° COMPONENT is thus in
 1081 NP.

1082 In the reduction used in Theorem 49, all \mathcal{TC}° components in the transformed instance are
 1083 closed. This suffices to prove that CLOSED \mathcal{TC}° COMPONENT is NP-hard, for any constant
 1084 lifetime $T > 2$. ◀