On Separators in Temporal Graphs

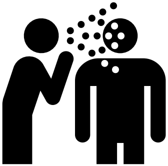
Hendrik Molter

Algorithmics and Computational Complexity,
TU Berlin, Germany

Algorithmic Aspects of Temporal Graphs,
Satellite Workshop of ICALP 2018, Prague

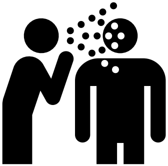
Based on joint work with Till Fluschnik, Rolf Niedermeier and Philipp Zschoche.

Introduction
Motivation: Separators

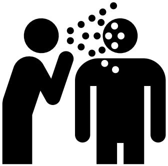


- Disease Spreading

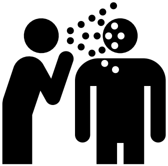
Introduction
Motivation: Separators



- Disease Spreading
- Rumor Spreading



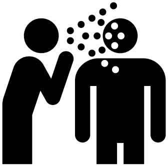
- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks



- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks

- Robustness of Connections

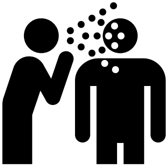
Introduction
Motivation: Separators



- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks

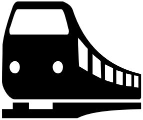
- Robustness of Connections
- Traffic Networks

Introduction
Motivation: Separators

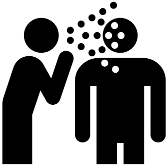


- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks

- Robustness of Connections
- Traffic Networks

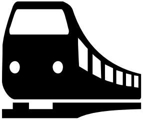


Introduction
Motivation: Separators



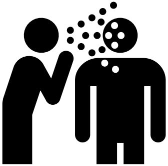
- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks

- Robustness of Connections
- Traffic Networks



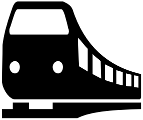
- Malware Spreading

Introduction
Motivation: Separators

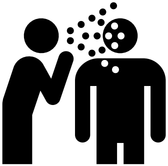


- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks

- Robustness of Connections
- Traffic Networks

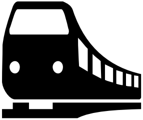


- Malware Spreading
- Rumor Spreading



- Disease Spreading
- Rumor Spreading
- Physical Proximity Networks

- Robustness of Connections
- Traffic Networks



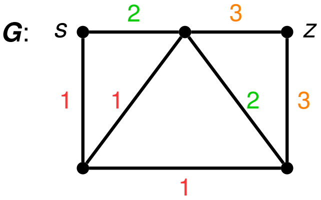
- Malware Spreading
- Rumor Spreading
- Social Networks / Computer Networks

Temporal Graph

A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.

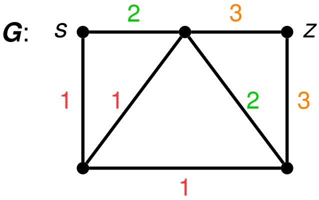
Temporal Graph

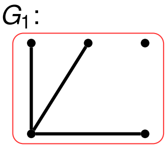
A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.



Temporal Graph

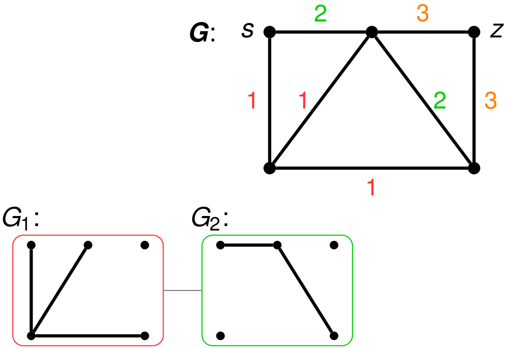
A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.





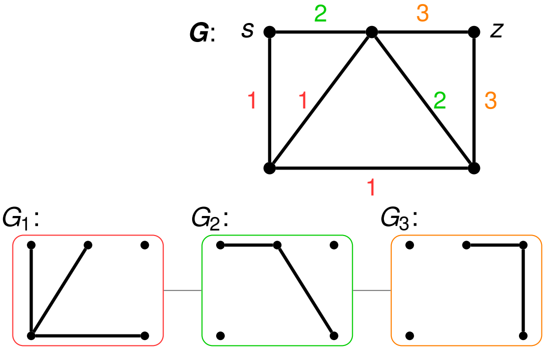
Temporal Graph

A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.



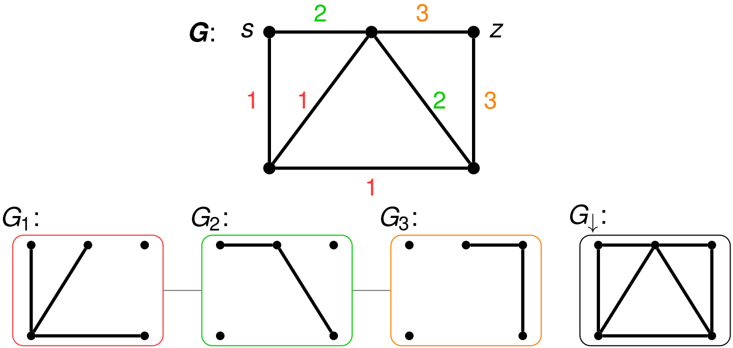
Temporal Graph

A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.



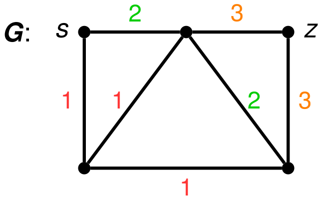
Temporal Graph

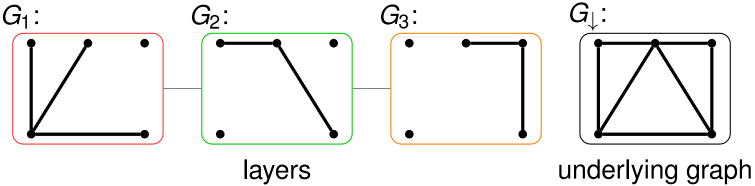
A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.



Temporal Graph

A **temporal graph** $\boldsymbol{G} = (V, E_1, E_2, \ldots, E_\tau)$ is defined as vertex set V with a list of edge sets $E_1, \ldots, E_\tau$ over V, where τ is the lifetime of $\boldsymbol{G}$.





Temporal Paths

A **strict** (s, z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$.

Temporal Paths

A **(non-)strict** (s, z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

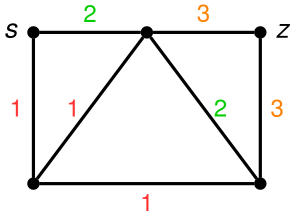
Temporal Paths

A **(non-)strict** (s,z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

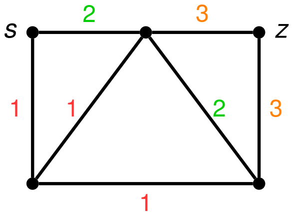
$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

strict temporal (s,z)**-paths:**



temporal (s,z)**-paths:**



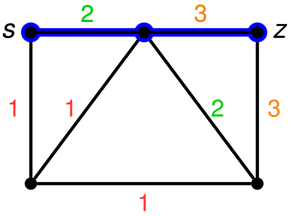
Temporal Paths

A **(non-)strict** (s, z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

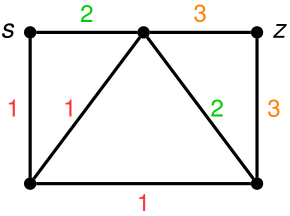
$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

strict temporal (s, z)**-paths:**



temporal (s, z)**-paths:**



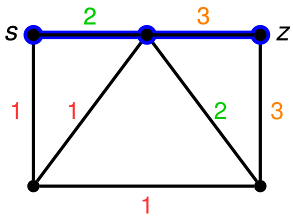
Temporal Paths

A **(non-)strict** (s,z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

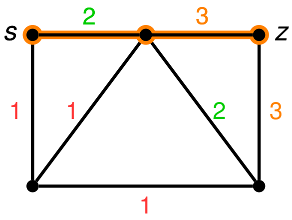
$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

strict temporal (s,z)**-paths:**



temporal (s,z)**-paths:**



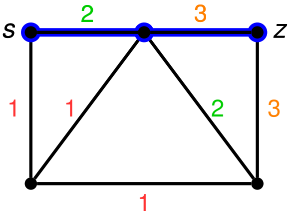
Temporal Paths

A **(non-)strict** (s, z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

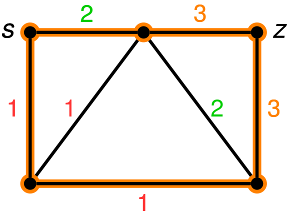
$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

strict temporal (s, z)**-paths:**



temporal (s, z)**-paths:**



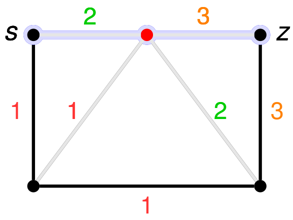
Temporal Paths

A **(non-)strict** (s, z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

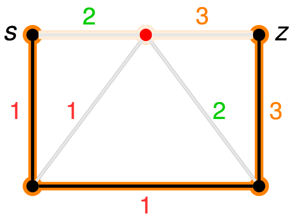
$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1] : t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

strict temporal (s, z)**-paths:**



temporal (s, z)**-paths:**



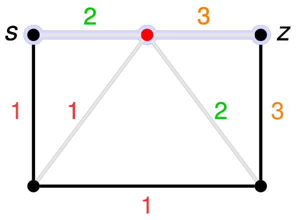
Temporal Paths

A **(non-)strict** (s, z)**-path** of length ℓ in $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is a list

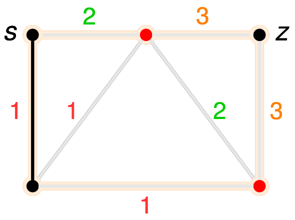
$$P = ((\{s = v_0, v_1\}, t_1), \ldots, (\{v_{\ell-1}, v_\ell = z\}, t_\ell)),$$

where $\{v_{i-1}, v_i\} \in E_{t_i}$ for all $i \in [\ell]$ and $v_i \neq v_j$ for all $i, j \in \{0, \ldots, \ell\}$ with $i \neq j$ and for all $i \in [\ell - 1]$: $t_i < t_{i+1}$ $(t_i \leq t_{i+1})$.

strict temporal (s, z)**-paths:**



temporal (s, z)**-paths:**



Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no strict (s, z)-path in $\boldsymbol{G} - S$?

(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

(Non-)Strict (s, z)-Separation

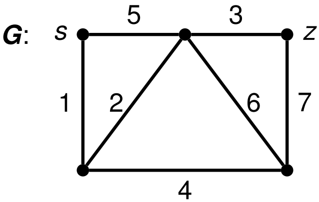
Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).

(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

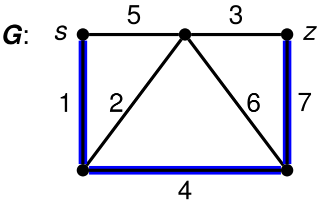
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

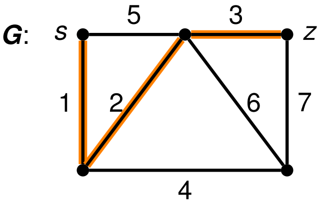
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

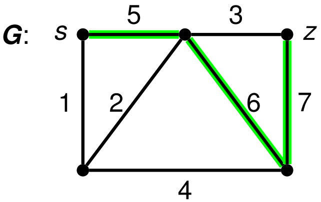
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

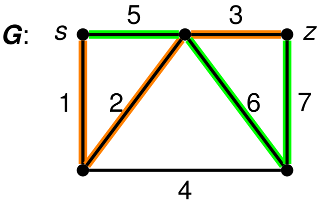
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

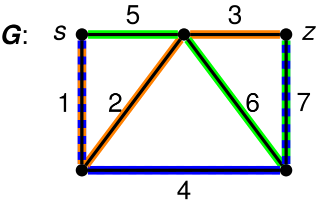
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

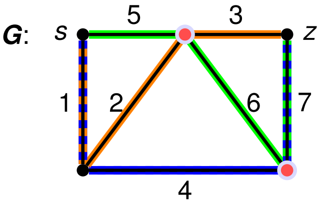
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

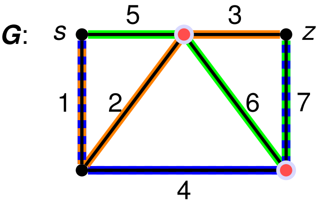
Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



(Non-)Strict (s, z)-Separation

Input: A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with two distinct vertices $s, z \in V$, and an integer k.
Question: Is there a subset $S \subseteq V \setminus \{s, z\}$ of size at most k such that there is no (non-)strict (s, z)-path in $\boldsymbol{G} - S$?

Berman [1996, Networks] showed that for temporal graphs Menger's Theorem **fails** (vertex-variant).



The edge-deletion variant can be computed in polynomial-time.

Kempe, Kleinberg, and Kumar [2002, JCSS] showed that

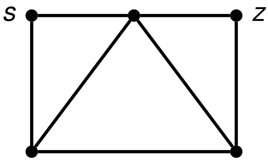
- (Non-)Strict (s, z)-Separation is NP-hard.

Kempe, Kleinberg, and Kumar [2002, JCSS] showed that

- (Non-)Strict (s, z)-Separation is NP-hard.
- Menger's Theorem holds if the underlying graph **excludes** a fixed minor.

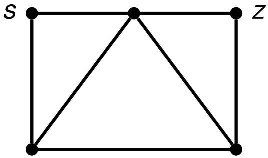
Kempe, Kleinberg, and Kumar [2002, JCSS] showed that

- (Non-)Strict (s, z)-Separation is NP-hard.
- Menger's Theorem holds if the underlying graph **excludes** a fixed minor.



Kempe, Kleinberg, and Kumar [2002, JCSS] showed that

- (Non-)Strict (s, z)-Separation is NP-hard.
- Menger's Theorem holds if the underlying graph **excludes** a fixed minor.



This presentation is based on Fluschnik et al. [2018, WG] and Zschoche et al. [2018, MFCS]. (Both to appear, available on arXiv.)

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.

n: instance size

k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.

n: instance size

k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.

Parameterized Hardness

- **W[1]-hard**: Presumably no FPT algorithm (XP algorithm possible).

n: instance size k: parameter

Parameterized Tractability

- **FPT** (fixed-parameter tractable): Solvable in $f(k) \cdot n^{O(1)}$ time.
- **XP**: Solvable in $n^{g(k)}$ time.

Parameterized Hardness

- **W[1]-hard**: Presumably no FPT algorithm (XP algorithm possible).
- **para-NP-hard**: NP-hard for constant k (no XP algorithm).

n: instance size k: parameter

Basic Results.

Parameter	(s, z)-Separation Strict	Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	

Basic Results.

Parameter	(s, z)-Separation Strict	(s, z)-Separation Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	
k	W[1]-hard	W[1]-hard

Basic Results.

Parameter	(s, z)-Separation Strict	(s, z)-Separation Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	
k	W[1]-hard	W[1]-hard
$\tau + k$	FPT	*open*

Basic Results.

Parameter	(s, z)-Separation Strict	(s, z)-Separation Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	
k	W[1]-hard	W[1]-hard
$\tau + k$	FPT	*open*

Complexity of Finding Temporal Separators

Basic Results

Basic Results.

Parameter	(s, z)-Separation: Strict	(s, z)-Separation: Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	para-NP-hard
k	W[1]-hard	W[1]-hard
$\tau + k$	FPT	*open*

Canonical next step: Restrict input graphs.

Basic Results.

Parameter	(s,z)-Separation Strict	(s,z)-Separation Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	para-NP-hard
k	W[1]-hard	W[1]-hard
$\tau + k$	FPT	*open*

Canonical next step: Restrict input graphs.

- Restrict each layer.

Basic Results.

Parameter	(s, z)-Separation: Strict	(s, z)-Separation: Non-Strict
$2 \leq \tau \leq 4$	poly-time	para-NP-hard
$\tau \geq 5$	para-NP-hard	
k	W[1]-hard	W[1]-hard
$\tau + k$	FPT	*open*

Canonical next step: Restrict input graphs.

- Restrict each layer.
- Restrict the underlying graph.

Complexity of Finding Temporal Separators
Restricting each Layer

(Non-)Strict (s,z)-Separation with **restricted layers**.

Layer Restriction	Complexity

Complexity of Finding Temporal Separators
Restricting each Layer

(Non-)Strict (s, z)-Separation with **restricted layers**.

Layer Restriction	Complexity
at most one edge	NP-hard and W[1]-hard wrt. k

Complexity of Finding Temporal Separators
Restricting each Layer

(Non-)Strict (s, z)-Separation with **restricted layers**.

Layer Restriction	Complexity
at most one edge	NP-hard and W[1]-hard wrt. k
forest unit interval	para-NP-hard wrt. τ

(Non-)Strict (s, z)-Separation with **restricted layers**.

Layer Restriction	Complexity
at most one edge	NP-hard and W[1]-hard wrt. k
forest unit interval	para-NP-hard wrt. τ

Complexity of Finding Temporal Separators
Restricting each Layer

(Non-)Strict (s, z)-Separation with **restricted layers**.

Layer Restriction	Complexity
at most one edge	NP-hard and W[1]-hard wrt. k
forest unit interval	para-NP-hard wrt. τ

Take away message:
Layer restrictions do not help much.

Complexity of Finding Temporal Separators
Restricting the Underlying Graph

(Non-)Strict (s, z)-Separation with **restricted underlying graph.**

Underlying Graph Restriction	Complexity

Complexity of Finding Temporal Separators
Restricting the Underlying Graph

(Non-)Strict (s, z)-Separation with **restricted underlying graph**.

Underlying Graph Restriction	Complexity
bounded treewidth	poly-time (FPT wrt. $\text{tw} + \tau$)

(Non-)Strict (s, z)-Separation with **restricted underlying graph**.

Underlying Graph Restriction	Complexity
bounded treewidth	poly-time (FPT wrt. $\text{tw} + \tau$)
bounded vertex cover	poly-time (FPT)

Complexity of Finding Temporal Separators

Restricting the Underlying Graph

(Non-)Strict (s, z)-Separation with **restricted underlying graph.**

Underlying Graph Restriction	Complexity
bounded treewidth	poly-time (FPT wrt. $\text{tw} + \tau$)
bounded vertex cover	poly-time (FPT)
complete $- \{s, z\}$ bipartite line graph	para-NP-h wrt. τ / W[1]-h wrt. k

Complexity of Finding Temporal Separators

Restricting the Underlying Graph

(Non-)Strict (s, z)-Separation with **restricted underlying graph.**

Underlying Graph Restriction	Complexity
bounded treewidth	poly-time (FPT wrt. $\text{tw} + \tau$)
bounded vertex cover	poly-time (FPT)
complete $- \{s, z\}$ bipartite line graph	para-NP-h wrt. τ / W[1]-h wrt. k
planar	NP-hard (Strict: FPT wrt. τ)

Complexity of Finding Temporal Separators
Restricting the Underlying Graph

(Non-)Strict (s, z)-Separation with **restricted underlying graph.**

Underlying Graph Restriction	Complexity
bounded treewidth	poly-time (FPT wrt. $\text{tw} + \tau$)
bounded vertex cover	poly-time (FPT)
complete $- \{s, z\}$ bipartite line graph	para-NP-h wrt. τ / W[1]-h wrt. k
planar	NP-hard (Strict: FPT wrt. τ)

Complexity of Finding Temporal Separators

Restricting the Underlying Graph

(Non-)Strict (s, z)-Separation with **restricted underlying graph**.

Underlying Graph Restriction	Complexity
bounded treewidth	poly-time (FPT wrt. $\mathrm{tw} + \tau$)
bounded vertex cover	poly-time (FPT)
complete $- \{s, z\}$ bipartite line graph	para-NP-h wrt. τ / W[1]-h wrt. k
planar	NP-hard (Strict: FPT wrt. τ)

Take away message:

Underlying graph restrictions help sometimes.

We have seen so far:

Complexity of Finding Temporal Separators
First Summary

We have seen so far:

- Layer restrictions:

Complexity of Finding Temporal Separators

First Summary

We have seen so far:

- Layer restrictions: do not seem to help.

Complexity of Finding Temporal Separators
First Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions:

Complexity of Finding Temporal Separators
First Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.

Complexity of Finding Temporal Separators
First Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.

Observation

All these restrictions are invariant under reordering of layers!

Complexity of Finding Temporal Separators

First Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.

Observation

All these restrictions are invariant under reordering of layers!

Idea: Restrict "temporality" of the input graph.

Complexity of Finding Temporal Separators
Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	Non-Strict
p-monotone		

Definition (cf. Khodaverdian et al. [2016]; Casteigts et al. [2012])

$\boldsymbol{G} = (V, E_1, \dots, E_\tau)$ is p-**monotone** if there are $1 = i_1 < \dots < i_{p+1} = \tau$ such that for all $\ell \in [p]$ it holds that $E_j \subseteq E_{j+1}$ or $E_j \supseteq E_{j+1}$ for all $i_\ell \leq j < i_{\ell+1}$.

Complexity of Finding Temporal Separators
Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s, z)-Separation Strict	(s, z)-Separation Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$

Definition (cf. Khodaverdian et al. [2016]; Casteigts et al. [2012])

$\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is p-**monotone** if there are $1 = i_1 < \cdots < i_{p+1} = \tau$ such that for all $\ell \in [p]$ it holds that $E_j \subseteq E_{j+1}$ or $E_j \supseteq E_{j+1}$ for all $i_\ell \leq j < i_{\ell+1}$.

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	(s,z)-Separation Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
q-periodic		

Definition (cf. Liu and Wu [2009]; Casteigts et al. [2012]; Flocchini et al. [2013])

$\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is ***q*-periodic** if $E_i = E_{i+q}$ for all $i \in [\tau - q]$. We call $r := \tau/q$ the number of periods.

Complexity of Finding Temporal Separators

Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s, z)-Separation: Strict	(s, z)-Separation: Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
q-periodic	NP-h for $q \geq 1$	poly-time for $q = 1$, NP-h for $q \geq 2$

Definition (cf. Liu and Wu [2009]; Casteigts et al. [2012]; Flocchini et al. [2013])

$\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is ***q*-periodic** if $E_i = E_{i+q}$ for all $i \in [\tau - q]$. We call $r := \tau/q$ the number of periods.

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	(s,z)-Separation Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
q-periodic	NP-h for $q \geq 1$	poly-time for $q = 1$, NP-h for $q \geq 2$
	poly-time if $r \geq n$	poly-time if $r \geq n$

Definition (cf. Liu and Wu [2009]; Casteigts et al. [2012]; Flocchini et al. [2013])

$\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is ***q*-periodic** if $E_i = E_{i+q}$ for all $i \in [\tau - q]$. We call $r := \tau/q$ the number of periods.

Complexity of Finding Temporal Separators
Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
T-interval connected		

Definition (Kuhn et al. [2010])

$\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is T**-interval connected** if for every $t \in [\tau - T + 1]$ the graph $G = (V, \cap_{i=t}^{t+T-1} E_i)$ is connected.

Complexity of Finding Temporal Separators

Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	(s,z)-Separation Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
T-interval connected	NP-h for $T \geq 1$	NP-h for $T \geq 1$

Definition (Kuhn et al. [2010])

$\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is ***T*-interval connected** if for every $t \in [\tau - T + 1]$ the graph $G = (V, \cap_{i=t}^{t+T-1} E_i)$ is connected.

Complexity of Finding Temporal Separators
Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	(s,z)-Separation Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
T-interval connected	NP-h for $T \geq 1$	NP-h for $T \geq 1$

Definition

$G = (V, E_1, \ldots, E_\tau)$ is **λ-steady** if for all $t \in [\tau - 1]$ we have that $|E_t \triangle E_{t+1}| \leq \lambda$.

λ-steady

Complexity of Finding Temporal Separators

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
T-interval connected	NP-h for $T \geq 1$	NP-h for $T \geq 1$
λ-steady	NP-h for $\lambda \geq 0$	poly-time for $\lambda = 0$, NP-h for $\lambda \geq 1$

Definition

$G = (V, E_1, \ldots, E_\tau)$ is **λ-steady** if for all $t \in [\tau - 1]$ we have that $|E_t \triangle E_{t+1}| \leq \lambda$.

Temporal graph classes with temporal aspects:

Restriction	(s, z)-Separation: Strict	(s, z)-Separation: Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
q-periodic	NP-h for $q \geq 1$	poly-time for $q = 1$, NP-h for $q \geq 2$
	poly-time if $r \geq n$	poly-time if $r \geq n$
T-interval connected	NP-h for $T \geq 1$	NP-h for $T \geq 1$
λ-steady	NP-h for $\lambda \geq 0$	poly-time for $\lambda = 0$, NP-h for $\lambda \geq 1$

Complexity of Finding Temporal Separators
Temporal Restrictions

Temporal graph classes with temporal aspects:

Restriction	(s,z)-Separation Strict	(s,z)-Separation Non-Strict
p-monotone	NP-h for $p \geq 1$	poly-time for $p = 1$, NP-h for $p \geq 2$
q-periodic	NP-h for $q \geq 1$	poly-time for $q = 1$, NP-h for $q \geq 2$
	poly-time if $r \geq n$	poly-time if $r \geq n$
T-interval connected	NP-h for $T \geq 1$	NP-h for $T \geq 1$
λ-steady	NP-h for $\lambda \geq 0$	poly-time for $\lambda = 0$, NP-h for $\lambda \geq 1$

Complexity of Finding Temporal Separators
Second Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.

Complexity of Finding Temporal Separators
Second Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.
- Temporal restrictions:

Complexity of Finding Temporal Separators
Second Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.
- Temporal restrictions: do not seem to help.

Complexity of Finding Temporal Separators
Second Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.
- Temporal restrictions: do not seem to help.

Idea: Tailored restrictions that do not fit into the above categories.

Complexity of Finding Temporal Separators
Second Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.
- Temporal restrictions: do not seem to help.

Idea: Tailored restrictions that do not fit into the above categories.

- Order-Preserving Temporal
  Unit Interval Graphs.

Complexity of Finding Temporal Separators
Second Summary

We have seen so far:

- Layer restrictions: do not seem to help.
- Underlying graph restrictions: help only in few cases.
- Temporal restrictions: do not seem to help.

Idea: Tailored restrictions that do not fit into the above categories.

- Order-Preserving Temporal Unit Interval Graphs.
- Temporal Graph with bounded-sized Temporal Core.

Order-Preserving Temporal Unit Interval Graph

A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is an **order-preserving temporal unit interval graph** if

Order-Preserving Temporal Unit Interval Graph

A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is an **order-preserving temporal unit interval graph** if

- each layer is a unit interval graph, and

Order-Preserving Temporal Unit Interval Graph

A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is an **order-preserving temporal unit interval graph** if

- each layer is a unit interval graph, and
- there is a total ordering $<_V$ which is **compatible** with each layer.

Order-Preserving Temporal Unit Interval Graph

A temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is an **order-preserving temporal unit interval graph** if

- each layer is a unit interval graph, and
- there is a total ordering $<_V$ which is **compatible** with each layer.

Recall: $<_V$ is compatible with a unit interval graph $G = (V, E)$ if $\{x, y\} \in E$ with $x <_V y$ implies $\{v \in V \mid x \leq_V v \leq_V y\}$ is a clique.

Order-Preserving Temporal Unit Interval Graph

A temporal graph $\mathcal{G} = (V, E_1, \ldots, E_\tau)$ is an **order-preserving temporal unit interval graph** if

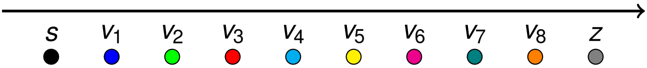
- each layer is a unit interval graph, and
- there is a total ordering $<_V$ which is **compatible** with each layer.

Recall: $<_V$ is compatible with a unit interval graph $G = (V, E)$ if $\{x, y\} \in E$ with $x <_V y$ implies $\{v \in V \mid x \leq_V v \leq_V y\}$ is a clique.

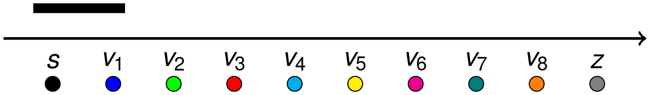
Motivation: Physical proximity networks in one-dimensional spaces.

(s, z)-Separation on Temporal Unit Interval Graphs

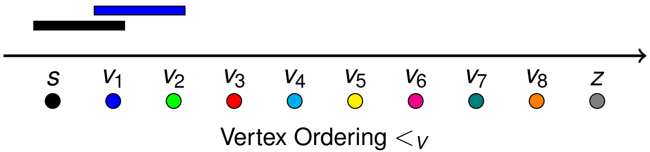
Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



Vertex Ordering $<_V$



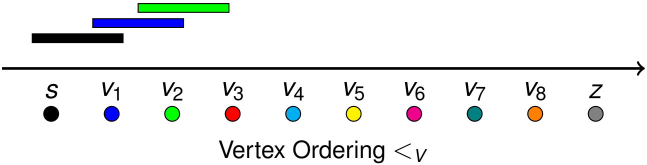
Vertex Ordering $<_V$



s
v1
v2
v3
v4
v5
v6
v7
v8
z
Vertex Ordering <V

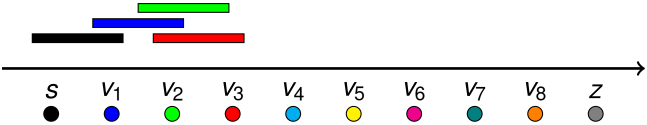
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



(s, z)-Separation on Temporal Unit Interval Graphs

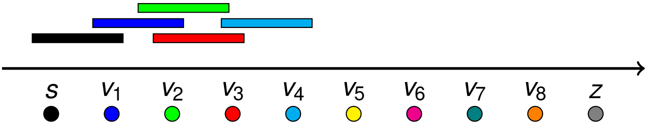
Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



Vertex Ordering $<_V$

(s, z)-Separation on Temporal Unit Interval Graphs

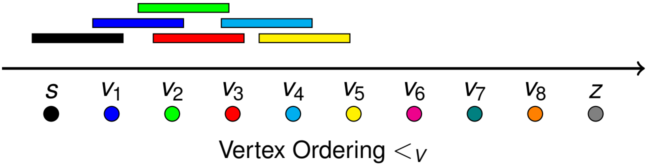
Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



Vertex Ordering $<_V$

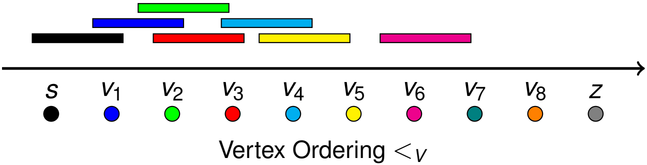
(s,z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s,z)-Separation Order-Preserving Temporal Unit Interval Graphs



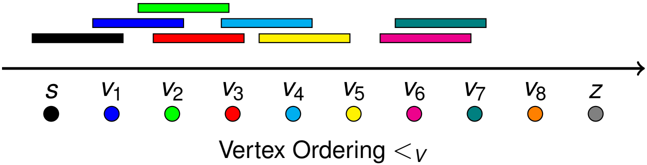
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs

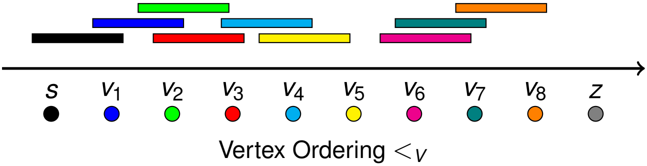


(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs

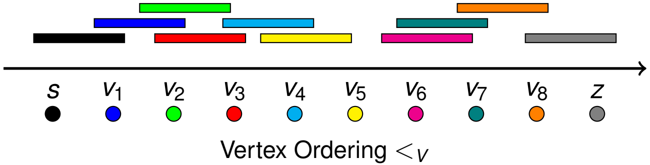


(s,z)-Separation on Temporal Unit Interval Graphs
Poly-time Algo for Non-Strict (s,z)-Separation Order-Preserving Temporal Unit Interval Graphs



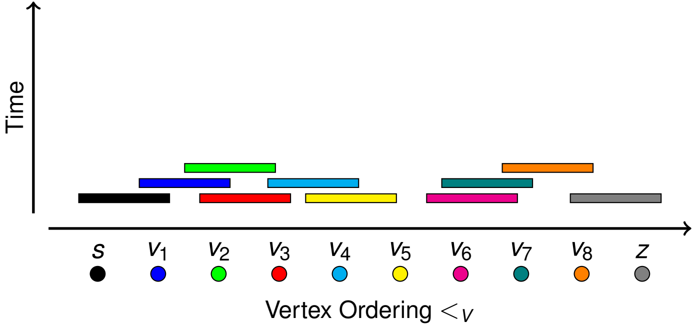
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



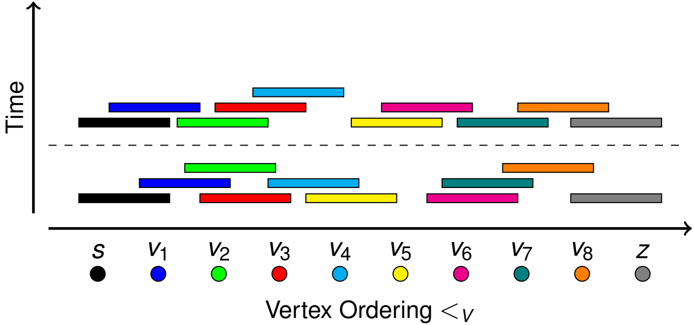
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



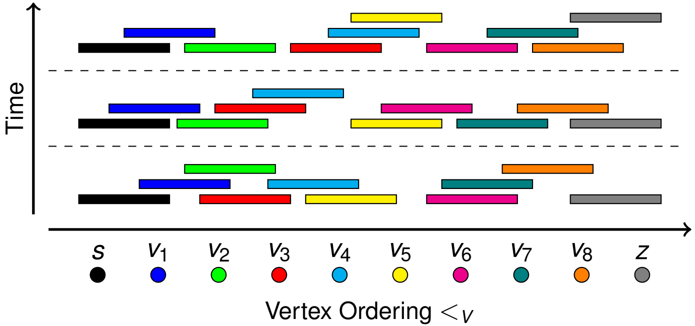
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



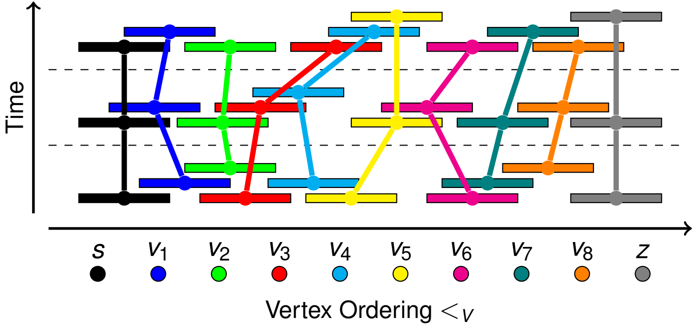
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



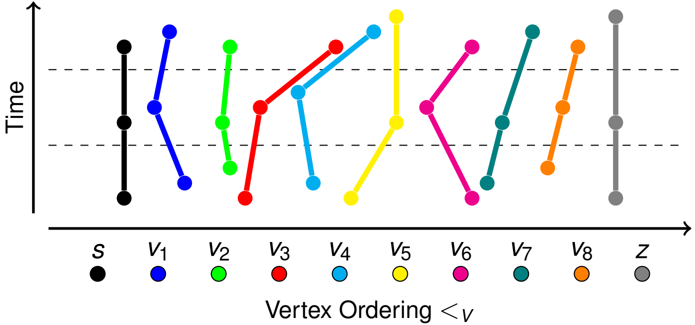
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



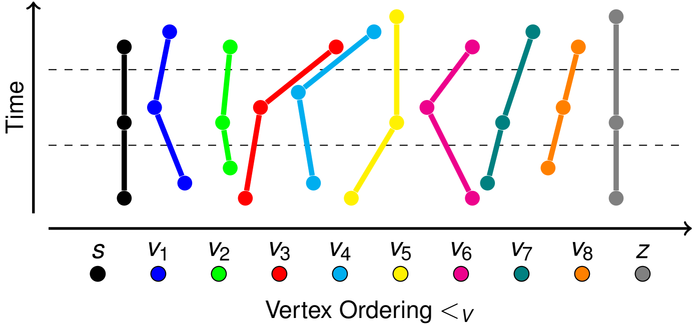
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs

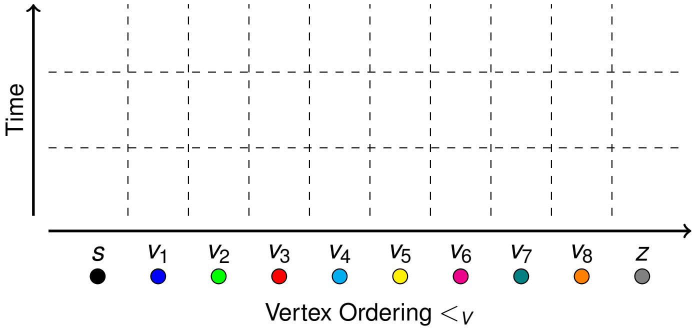


Observation

"Compatible" means these lines do not cross.

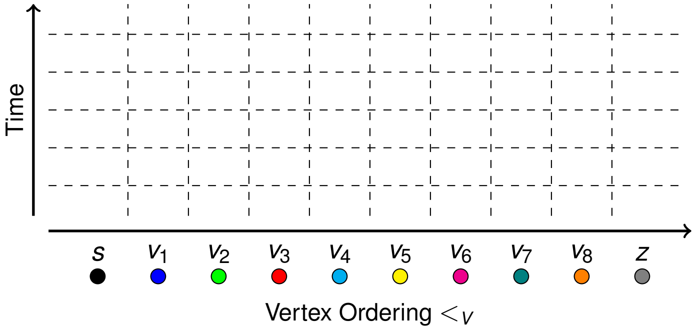
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



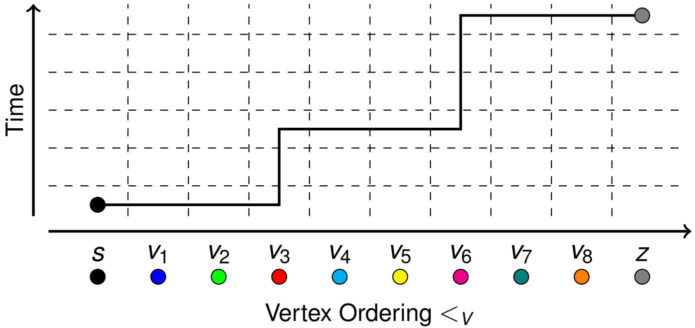
(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs

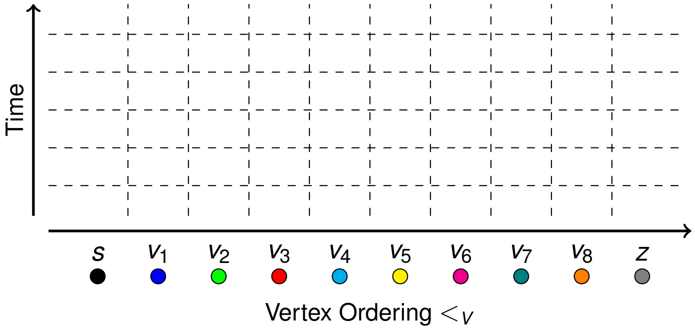


Observation

There are always temporal paths that follow the vertex ordering.

(s, z)-Separation on Temporal Unit Interval Graphs
Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs

DP-Table T:

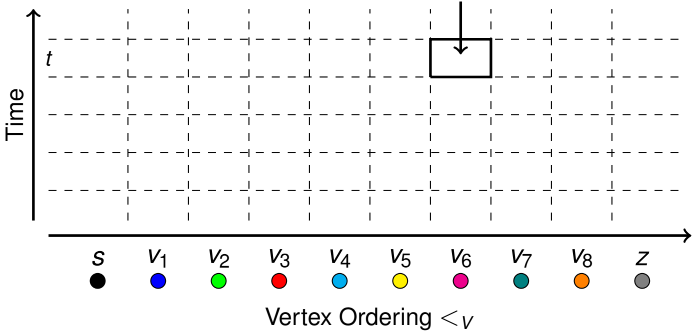


(s, z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs

DP-Table T:

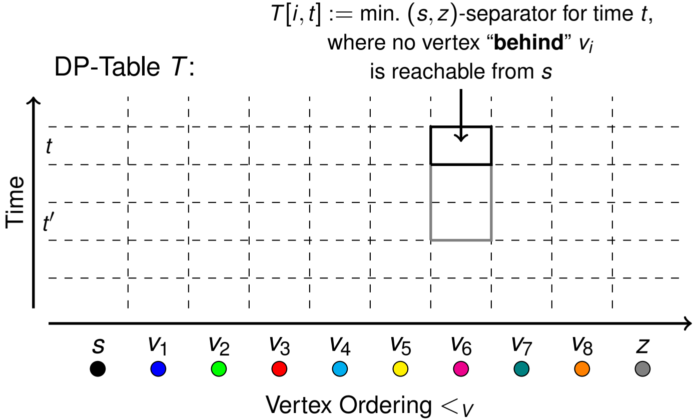
$T[i, t] :=$ min. (s, z)-separator for time t, where no vertex "**behind**" v_i is reachable from s



(s,z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s,z)-Separation Order-Preserving Temporal Unit Interval Graphs

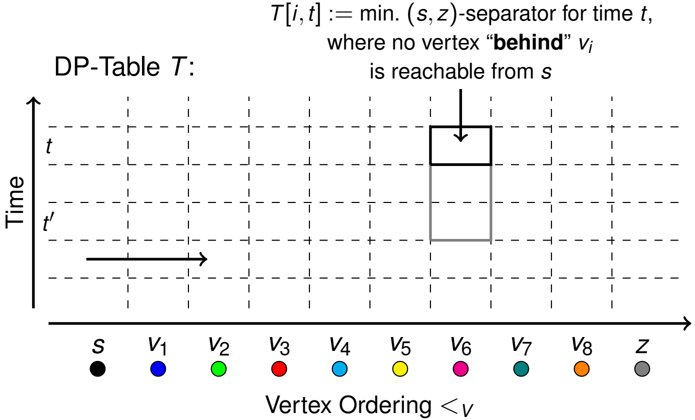
DP-Table T:



- Guess earliest time t' when v_i is reachable from s.

(s, z)-Separation on Temporal Unit Interval Graphs

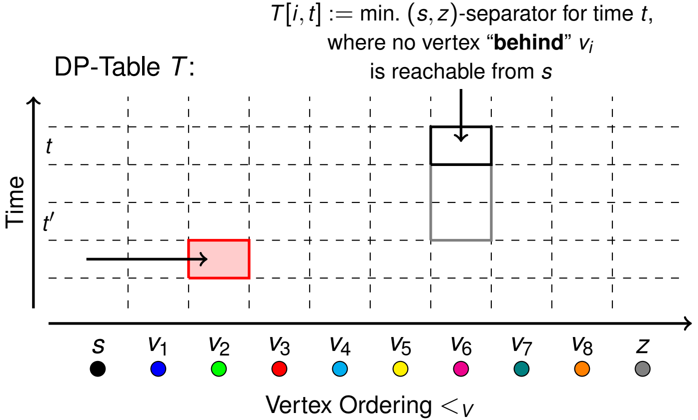
Poly-time Algo for Non-Strict (s, z)-Separation Order-Preserving Temporal Unit Interval Graphs



- Guess earliest time t' when v_i is reachable from s.
- Guess furthest vertex v_j reachable from s in $t' - 1$.

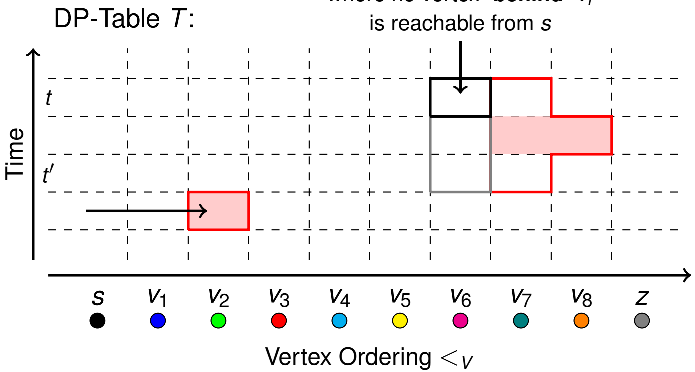
(s,z)-Separation on Temporal Unit Interval Graphs

Poly-time Algo for Non-Strict (s,z)-Separation Order-Preserving Temporal Unit Interval Graphs



- Guess earliest time t' when v_i is reachable from s.
- Guess furthest vertex v_j reachable from s in $t'-1$.
- $T[i,t] = T[j,t'-1]$

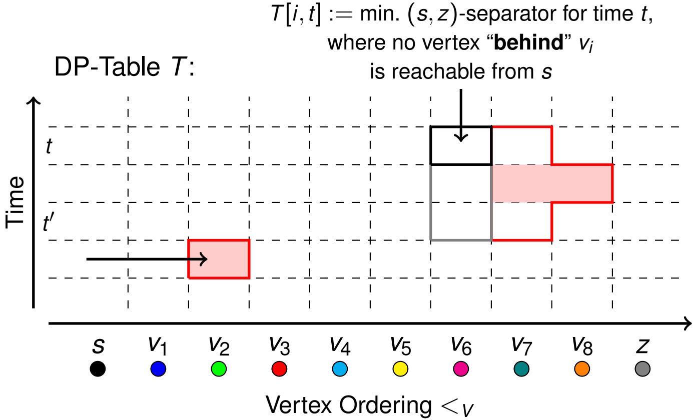
$T[i, t] :=$ min. (s, z)-separator for time t,
where no vertex "**behind**" v_i
is reachable from s



- Guess earliest time t' when v_i is reachable from s.
- Guess furthest vertex v_j reachable from s in $t' - 1$.
- $T[i, t] = T[j, t' - 1] \cup$ max. "right" neighborhood of v_i in $[t', t]$.

(s,z)-Separation on Temporal Unit Interval Graphs

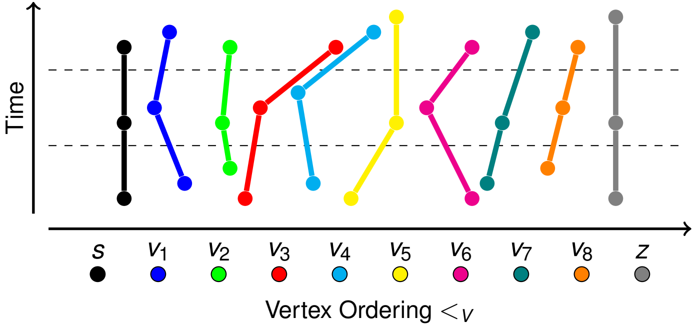
Poly-time Algo for Non-Strict (s,z)-Separation Order-Preserving Temporal Unit Interval Graphs



Theorem

Non-Strict (s,z)-Separation on order-preserving temporal unit interval graphs is poly-time solvable.

(s, z)-Separation on Temporal Unit Interval Graphs
Almost Order-Preserving Temporal Unit Interval Graphs

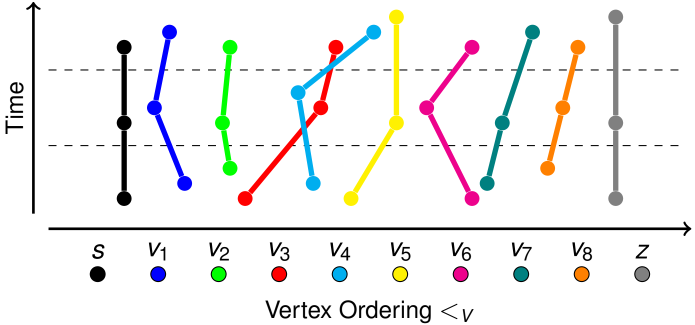


Observation

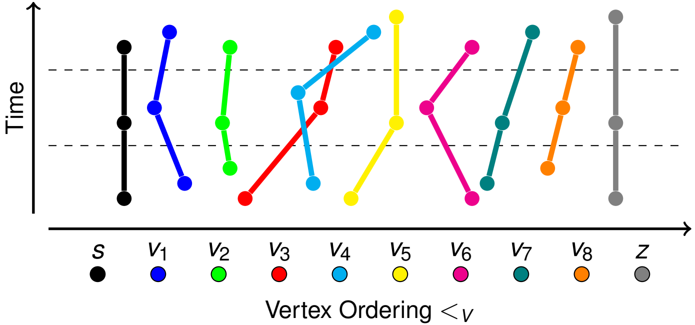
"Compatible" means these lines do not cross.

(s, z)-Separation on Temporal Unit Interval Graphs

Almost Order-Preserving Temporal Unit Interval Graphs



(s, z)-Separation on Temporal Unit Interval Graphs
Almost Order-Preserving Temporal Unit Interval Graphs

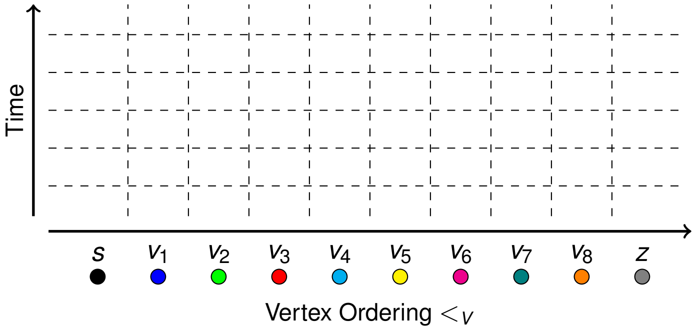


- Idea: Bound number of crossings between consecutive time steps.

  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.

(s, z)-Separation on Temporal Unit Interval Graphs

Almost Order-Preserving Temporal Unit Interval Graphs

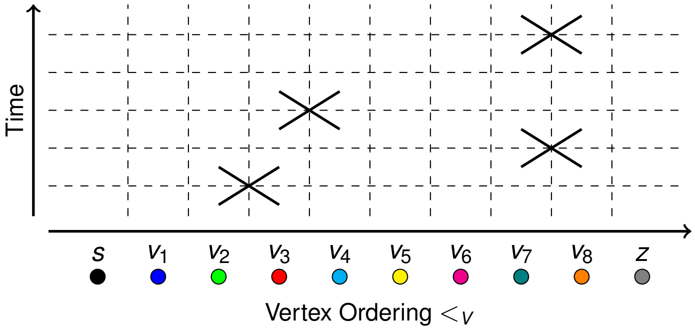


- Idea: Bound number of crossings between consecutive time steps.

  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.

(s, z)-Separation on Temporal Unit Interval Graphs

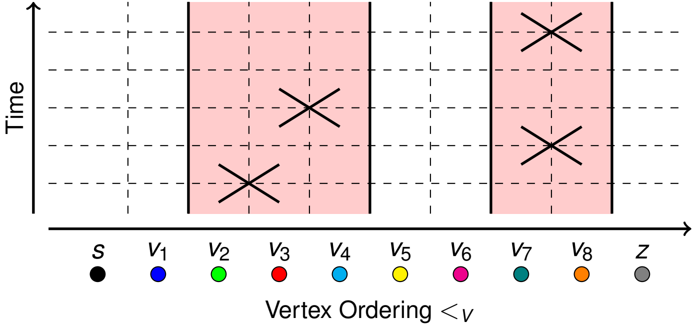
Almost Order-Preserving Temporal Unit Interval Graphs



- Idea: Bound number of crossings between consecutive time steps.

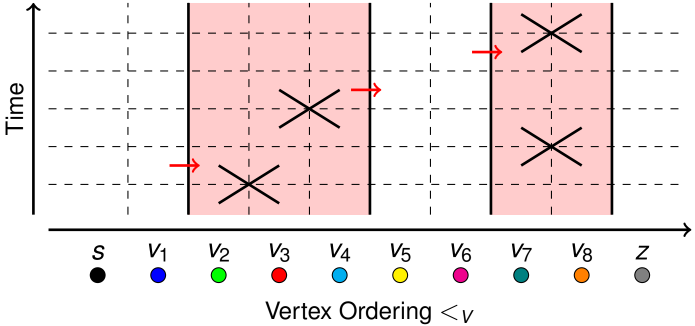
  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.

(s, z)-Separation on Temporal Unit Interval Graphs
Almost Order-Preserving Temporal Unit Interval Graphs



- Idea: Bound number of crossings between consecutive time steps.
  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.
- Brute-force the "regions" where crossings happen.

(s, z)-Separation on Temporal Unit Interval Graphs
Almost Order-Preserving Temporal Unit Interval Graphs

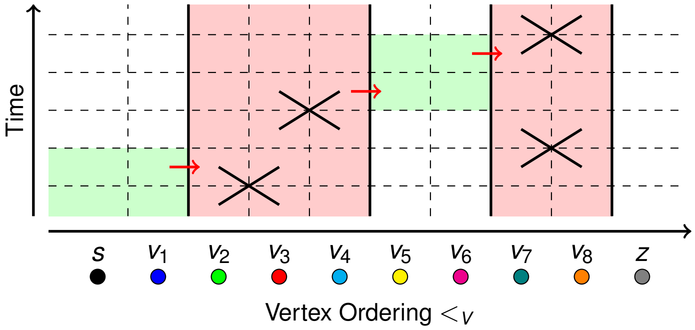


- Idea: Bound number of crossings between consecutive time steps.

  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.
- Brute-force the “regions” where crossings happen.

(s, z)-Separation on Temporal Unit Interval Graphs

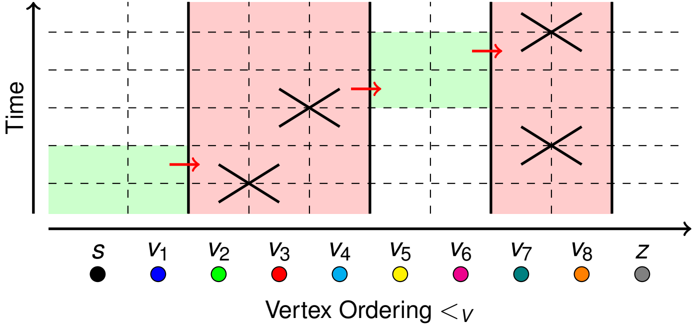
Almost Order-Preserving Temporal Unit Interval Graphs



- Idea: Bound number of crossings between consecutive time steps.
  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.
- Brute-force the "regions" where crossings happen.
  Solve the rest with the poly-time algorithm.

(s, z)-Separation on Temporal Unit Interval Graphs

Almost Order-Preserving Temporal Unit Interval Graphs



- Idea: Bound number of crossings between consecutive time steps.
  $\Leftrightarrow$ Vertex orderings have bounded **Kendall tau distance** κ.
- Brute-force the "regions" where crossings happen.
  Solve the rest with the poly-time algorithm.
- Size of regions bounded by κ and the lifetime τ.

(s, z)-Separation on Temporal Unit Interval Graphs
Summary

Theorem

(Non-)Strict (s, z)-Separation on order-preserving temporal unit interval graphs is poly-time solvable.

Theorem

(Non-)Strict (s, z)-Separation on order-preserving temporal unit interval graphs is poly-time solvable.

Theorem

(Non-)Strict (s, z)-Separation on temporal unit interval graphs is FPT wrt. $(\kappa + \tau)$.

(s, z)-Separation on Temporal Unit Interval Graphs
Summary

Theorem

(Non-)Strict (s, z)-Separation on order-preserving temporal unit interval graphs is poly-time solvable.

Theorem

(Non-)Strict (s, z)-Separation on temporal unit interval graphs is FPT wrt. $(\kappa + \tau)$.

Theorem

(Non-)Strict (s, z)-Separation on temporal unit interval graphs is para-NP-hard wrt. κ and para-NP-hard wrt. τ.

Temporal Core

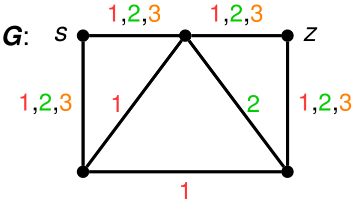
The **temporal core** of $G = (V, E_1, \ldots, E_\tau)$ is the vertex set

$$W = \{v \in V \mid \exists \{v, w\} \in (\bigcup_{i=1}^{\tau} E_i) \setminus (\bigcap_{i=1}^{\tau} E_i)\}.$$

Temporal Core

The **temporal core** of $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is the vertex set

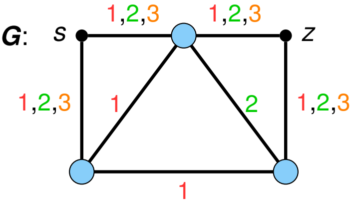
$$W = \{v \in V \mid \exists \{v, w\} \in (\bigcup_{i=1}^{\tau} E_i) \setminus (\bigcap_{i=1}^{\tau} E_i)\}.$$



Temporal Core

The **temporal core** of $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is the vertex set

$$W = \{v \in V \mid \exists \{v, w\} \in (\bigcup_{i=1}^{\tau} E_i) \setminus (\bigcap_{i=1}^{\tau} E_i)\}.$$



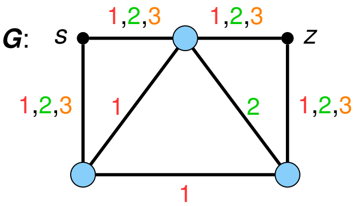
Temporal Core

Motivation and Definition

Temporal Core

The **temporal core** of $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ is the vertex set

$$W = \{v \in V \mid \exists \{v, w\} \in (\bigcup_{i=1}^{\tau} E_i) \setminus (\bigcap_{i=1}^{\tau} E_i)\}.$$

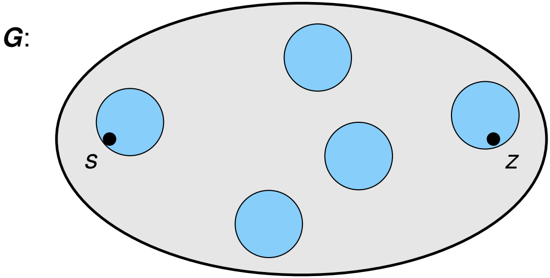


Recall: Strict (s, z)-Separation is NP-hard even if $W = \emptyset$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

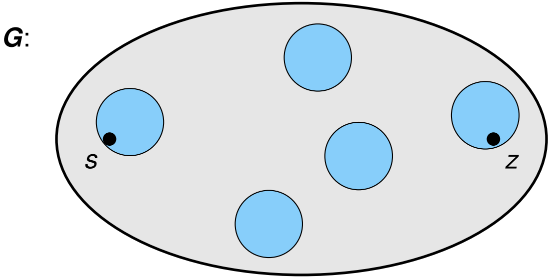


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

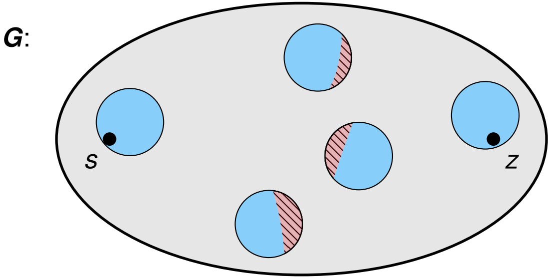
- Guess which core vertices are part of the separator.



Non-Strict (s, z)-Separation with small Temporal Cores
FPT Algorithm for “Size of the Temporal Core”

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator.

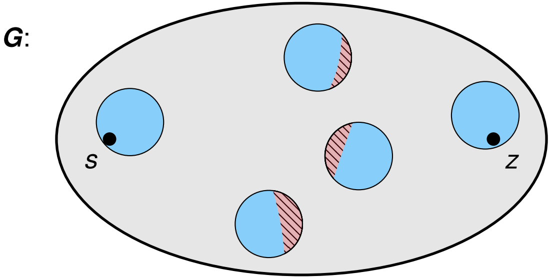


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator.
- Guess which core vertices need to be separated from each other.

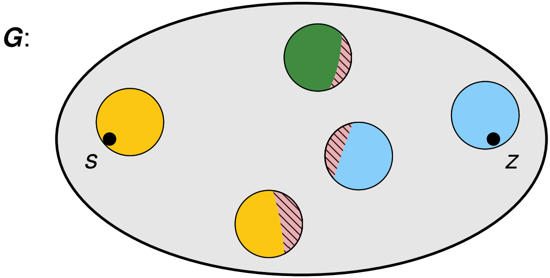


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator.
- Guess which core vertices need to be separated from each other.

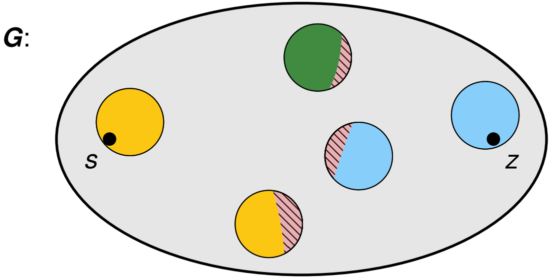


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for “Size of the Temporal Core”

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator.
- Guess which core vertices need to be separated from each other.
- Use an algorithm for **Node Multiway Cut**.

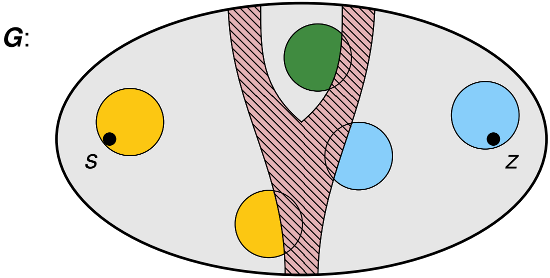


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator.
- Guess which core vertices need to be separated from each other.
- Use an algorithm for **Node Multiway Cut**.



Node Multiway Cut

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

Node Multiway Cut

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

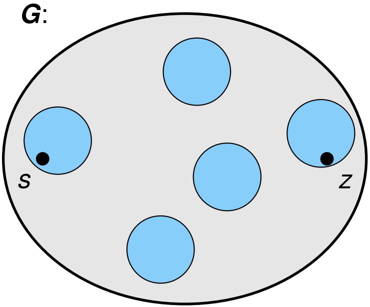
Theorem (Cygan et al. [2013], TOCT)

Node Multiway Cut can be solved in $2^{k-b} \cdot |V|^{O(1)}$ time,
where $b = \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

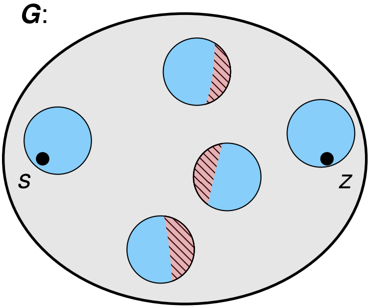
- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.



Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

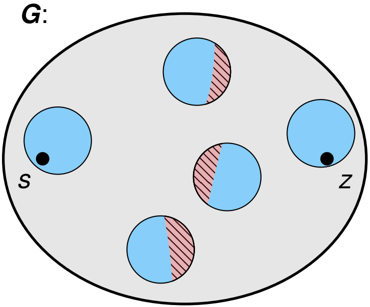
- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.



Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

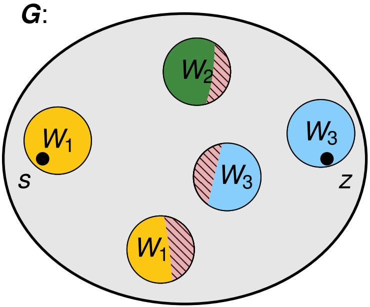
- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \ldots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.



Non-Strict (s,z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

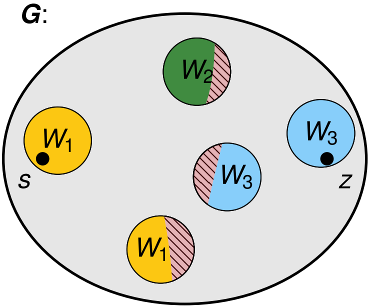
- Guess a set $S_W \subseteq (W \setminus \{s,z\})$ of size at most k.
- Guess a partition $\{W_1,\ldots,W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.



Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for “Size of the Temporal Core”

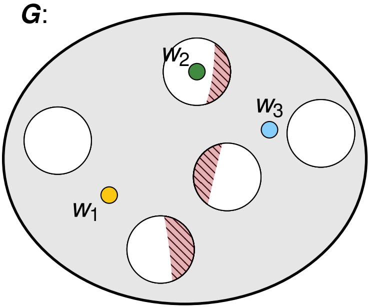
- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \ldots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_\downarrow - W$ and adding a vertex w_i for each part W_i.



Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \ldots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_{\downarrow} - W$ and adding a vertex w_i for each part W_i.

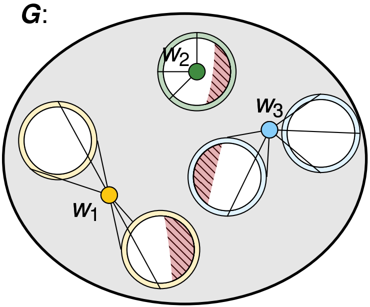


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \ldots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_\downarrow - W$ and adding a vertex w_i for each part W_i.

  For all $i \in [r]$, add edge sets $\{\{v, w_i\} \mid v \in N_{G_\downarrow}(W_i) \setminus W\}$.

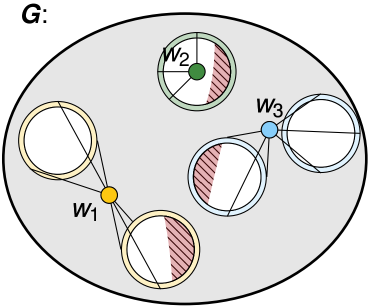


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for “Size of the Temporal Core”

- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \dots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_\downarrow - W$ and adding a vertex w_i for each part W_i.

  For all $i \in [r]$, add edge sets $\{\{v, w_i\} \mid v \in N_{G_\downarrow}(W_i) \setminus W\}$.
- Solve Node Multiway Cut instance $(G', \{w_1, \dots, w_r\}, k - |S_W|)$.

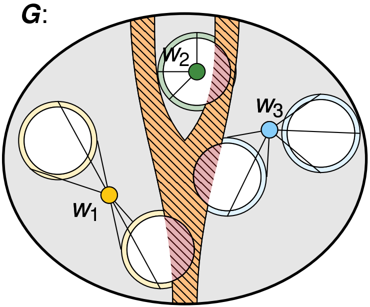


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \dots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_\downarrow - W$ and adding a vertex w_i for each part W_i.

  For all $i \in [r]$, add edge sets $\{\{v, w_i\} \mid v \in N_{G_\downarrow}(W_i) \setminus W\}$.
- Solve Node Multiway Cut instance $(G', \{w_1, \dots, w_r\}, k - |S_W|)$.

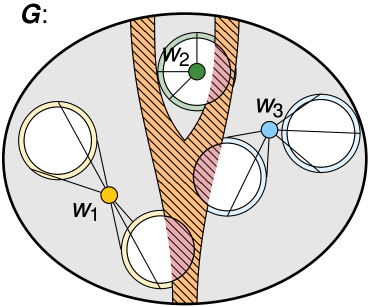


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \dots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_\downarrow - W$ and adding a vertex w_i for each part W_i.

  For all $i \in [r]$, add edge sets $\{\{v, w_i\} \mid v \in N_{G_\downarrow}(W_i) \setminus W\}$.
- Solve Node Multiway Cut instance $(G', \{w_1, \dots, w_r\}, k - |S_W|)$.
- Check whether the solution is correct.

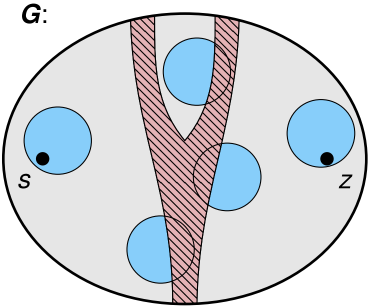


Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

- Guess a set $S_W \subseteq (W \setminus \{s, z\})$ of size at most k.
- Guess a partition $\{W_1, \ldots, W_r\}$ of $W \setminus S_W$ such that s and z are not in the same part.
- Construct the graph G' by copying $G_\downarrow - W$ and adding a vertex w_i for each part W_i.

  For all $i \in [r]$, add edge sets $\{\{v, w_i\} \mid v \in N_{G_\downarrow}(W_i) \setminus W\}$.
- Solve Node Multiway Cut instance $(G', \{w_1, \ldots, w_r\}, k - |S_W|)$.
- Check whether the solution is correct.



Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for “Size of the Temporal Core”

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\mathbf{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**
- Use an algorithm for Node Multiway Cut.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**
- Use an algorithm for Node Multiway Cut.

Theorem (Cygan et al. [2013], TOCT)

Node Multiway Cut can be solved in $2^{k-b} \cdot |V|^{O(1)}$ time,
where $b = \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**
- Use an algorithm for Node Multiway Cut.
  - Let L be a minimum (s, z)-separator in $G_\downarrow - (W \setminus \{s, z\})$.

Theorem (Cygan et al. [2013], TOCT)

Node Multiway Cut can be solved in $2^{k-b} \cdot |V|^{O(1)}$ time,
where $b = \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**
- Use an algorithm for Node Multiway Cut.
  - Let L be a minimum (s, z)-separator in $G_\downarrow - (W \setminus \{s, z\})$.
  - If $k \geq |W \setminus \{s, z\}| + |L|$, **Ok!**

Theorem (Cygan et al. [2013], TOCT)

Node Multiway Cut can be solved in $2^{k-b} \cdot |V|^{O(1)}$ time,
where $b = \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**
- Use an algorithm for Node Multiway Cut.
  - Let L be a minimum (s, z)-separator in $G_\downarrow - (W \setminus \{s, z\})$.
  - If $k \geq |W \setminus \{s, z\}| + |L|$, **Ok!** Otherwise, $k - b \leq k - |L| < |W|$.

Theorem (Cygan et al. [2013], TOCT)

Node Multiway Cut can be solved in $2^{k-b} \cdot |V|^{O(1)}$ time,
where $b = \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$.

Non-Strict (s, z)-Separation with small Temporal Cores

FPT Algorithm for "Size of the Temporal Core"

Theorem

Non-Strict (s, z)-Separation is FPT wrt. $|W|$.

Given a temporal graph $\boldsymbol{G} = (V, E_1, \ldots, E_\tau)$ with temporal core W:

- Guess which core vertices are part of the separator. **Ok!**
- Guess which core vertices need to be separated from each other. **Ok!**
- Use an algorithm for Node Multiway Cut. **Ok!**
  - Let L be a minimum (s, z)-separator in $G_\downarrow - (W \setminus \{s, z\})$.
  - If $k \geq |W \setminus \{s, z\}| + |L|$, **Ok!** Otherwise, $k - b \leq k - |L| < |W|$.

Theorem (Cygan et al. [2013], TOCT)

Node Multiway Cut can be solved in $2^{k-b} \cdot |V|^{O(1)}$ time,
where $b = \max_{x \in T} \min\{|S| \mid S \subseteq V \text{ is an } (x, T \setminus \{x\})\text{-separator}\}$.

Outlook
and Future Work

Summary:

Outlook
and Future Work

Summary:

- (Non-)Strict (s,z)-Separation is hard, even in very restricted cases.

Outlook
and Future Work

Summary:

- (Non-)Strict (s, z)-Separation is hard, even in very restricted cases.
- Tractable cases: Almost order-preserving temporal unit interval graphs and temporal graphs with bounded temporal core.

Outlook
and Future Work

Summary:

- (Non-)Strict (s,z)-Separation is hard, even in very restricted cases.
- Tractable cases: Almost order-preserving temporal unit interval graphs and temporal graphs with bounded temporal core.

Discussion:

Outlook
and Future Work

Summary:

- (Non-)Strict (s,z)-Separation is hard, even in very restricted cases.
- Tractable cases: Almost order-preserving temporal unit interval graphs and temporal graphs with bounded temporal core.

Discussion:

- One-dimensional physical proximity not very interesting in practice.

Outlook
and Future Work

Summary:

- (Non-)Strict (s, z)-Separation is hard, even in very restricted cases.
- Tractable cases: Almost order-preserving temporal unit interval graphs and temporal graphs with bounded temporal core.

Discussion:

- One-dimensional physical proximity not very interesting in practice.
- Strict (s, z)-Separation seems to be more realistic, however many positive results only hold in the non-strict case.

Outlook
and Future Work

Summary:

- (Non-)Strict (s, z)-Separation is hard, even in very restricted cases.
- Tractable cases: Almost order-preserving temporal unit interval graphs and temporal graphs with bounded temporal core.

Discussion:

- One-dimensional physical proximity not very interesting in practice.
- Strict (s, z)-Separation seems to be more realistic, however many positive results only hold in the non-strict case.

Thank you!

References I

Computer Icon on Slide 2 taken from https://commons.wikimedia.org/wiki/File:Blue_computer_icon.svg (CC BY-SA 3.0).

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

Casteigts, A., Flocchini, P., Quattrociocchi, W., and Santoro, N. (2012). Time-varying graphs and dynamic networks. *International Journal of Parallel, Emergent and Distributed Systems*, 27(5):387–408.

Cygan, M., Pilipczuk, M., Pilipczuk, M., and Wojtaszczyk, J. O. (2013). On multiway cut parameterized above lower bounds. *ACM Transactions on Computation Theory*, 5(1):3:1–3:11.

Flocchini, P., Mans, B., and Santoro, N. (2013). On the exploration of time-varying networks. *Theoretical Computer Science*, 469:53–68.

Fluschnik, T., Molter, H., Niedermeier, R., and Zschoche, P. (2018). Temporal graph classes: A view through temporal separators. In *Proceedings of the 44th International Workshop on Graph-Theoretic Concepts in Computer Science (WG'18)*, LNCS. Springer. Accepted for publication. To appear.

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

Khodaverdian, A., Weitz, B., Wu, J., and Yosef, N. (2016). Steiner network problems on temporal graphs. *CoRR*, abs/1609.04918v2.

Kuhn, F., Lynch, N. A., and Oshman, R. (2010). Distributed computation in dynamic networks. In *Proceedings of the 42nd Annual ACM Symposium on the Theory of Computing (STOC '10)*, pages 513–522. ACM.

Liu, C. and Wu, J. (2009). Scalable routing in cyclic mobile networks. *IEEE Transactions on Parallel and Distributed Systems*, 20(9):1325–1338.

Zschoche, P., Fluschnik, T., Molter, H., and Niedermeier, R. (2018). The complexity of finding small separators in temporal graphs. In *Proceedings of the 43rd International Symposium on Mathematical Foundations of Computer Science (MFCS'18)*, LIPIcs. Schloss Dagstuhl—Leibniz Center for Informatics. Accepted for publication. To appear.