Multistage Graph Problems on a Global Budget

K. Heeger, A. S. Himmel, F. Kammer,
R. Niedermeier, M. Renken, and A. Sajenko

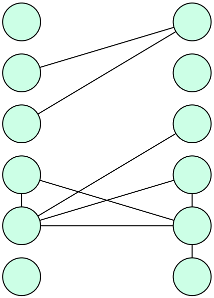
THM, University of Applied Sciences Mittelhessen

July 2021

Temporal Graphs

temporal graph

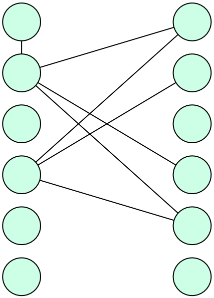
A graph in which the edge set can change in every (time) step.
$n := \#$vertices and $\tau :=$ maximum number of steps (lifetime)



Temporal Graphs

temporal graph

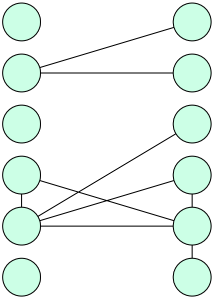
A graph in which the edge set can change in every (time) step.
$n := \#$vertices and $\tau :=$ maximum number of steps (lifetime)



Temporal Graphs

temporal graph

A graph in which the edge set can change in every (time) step.
$n :=$ #*vertices and* $\tau :=$ *maximum number of steps (lifetime)*



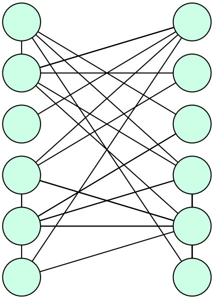
Temporal Graphs

temporal graph

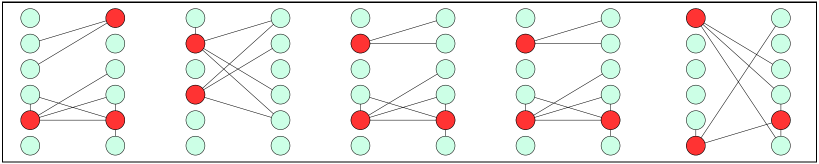
A graph in which the edge set can change in every (time) step.
$n :=$ #vertices and $\tau :=$ maximum number of steps (lifetime)

underlying graph

The graph with all edges that are present in at least one step.

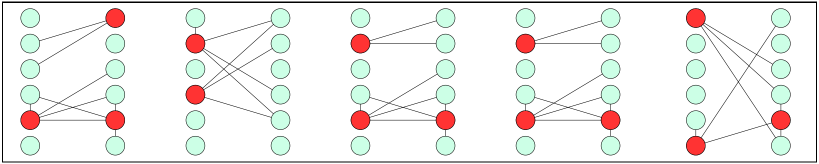


VERTEX COVER (steps are independent) k :=solution size



Multistage Graph Problems

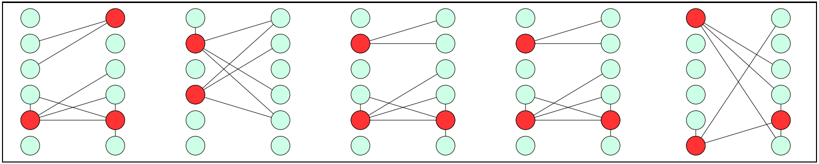
VERTEX COVER (steps are independent) k :=solution size



multistage problems on temporal graphs

Find a small solution for each layer of the temporal graph such that the solutions of two subsequent layers differ not too much.

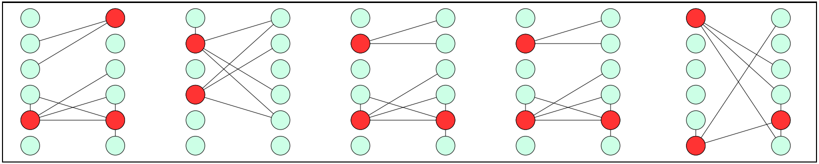
VERTEX COVER (steps are independent) k :=solution size



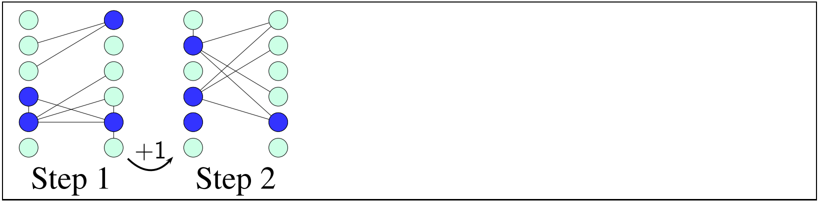
MULTISTAGE VERTEX COVER
with local budget $r := 1$ (changes from one step to the next)

Multistage Graph Problems

Vertex Cover (steps are independent) k :=solution size



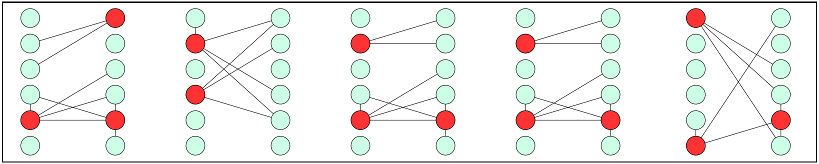
Multistage Vertex Cover
with local budget $r := 1$ (changes from one step to the next)



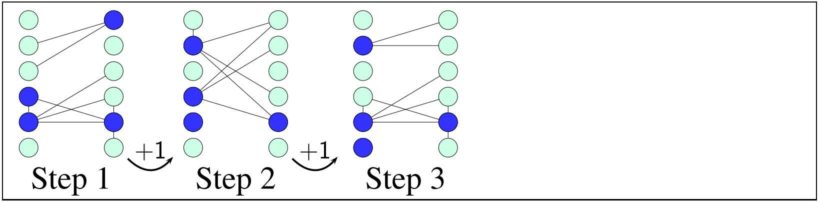
$k := 4$ $n := 12, \tau := 5$

Multistage Graph Problems

Vertex Cover (steps are independent) k :=solution size



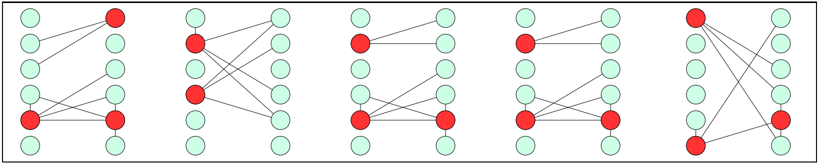
Multistage Vertex Cover
with local budget $r := 1$ (changes from one step to the next)



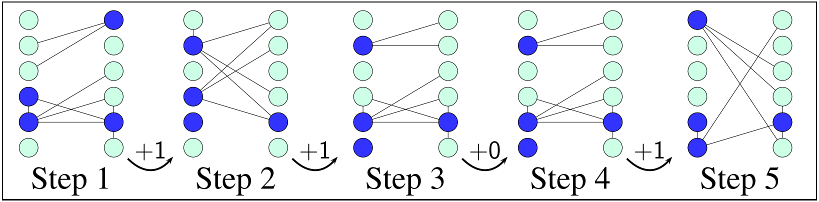
$k := 4$ $n := 12, \tau := 5$

Multistage Graph Problems

VERTEX COVER (steps are independent) k :=solution size



MULTISTAGE VERTEX COVER
with local budget $r := 1$ (changes from one step to the next)



$k := 4$ $n := 12, \tau := 5$

Recent Results

multistage problems on temporal graphs

Find a small solution for each layer of the temporal graph such that the solutions of two subsequent layers differ not too much.

- Fluschnik, Niedermeier, Rohm, and Zschoche — IPEC 2019
  Multistage vertex cover
- Fluschnik, Niedermeier, Schubert, and Zschoche — ISAAC 2020
  Multistage s-t path

Recent Results

classical multistage problems on temporal graphs

Find a small solution for each layer of the temporal graph such that the solutions of two subsequent layers differ not too much.

- Fluschnik, Niedermeier, Rohm, and Zschoche — IPEC 2019
  Multistage vertex cover
- Fluschnik, Niedermeier, Schubert, and Zschoche — ISAAC 2020
  Multistage s-t path

Recent Results

classical multistage problems on temporal graphs

Find a small solution for each layer of the temporal graph such that the solutions of two subsequent layers differ not too much.

- Fluschnik, Niedermeier, Rohm, and Zschoche — IPEC 2019
  Multistage vertex cover
- Fluschnik, Niedermeier, Schubert, and Zschoche — ISAAC 2020
  Multistage s-t path

Further kinds of multistage problems

- Bampis, Escoffier, Lampis, and Paschos — SWAT 2018
  Multistage matchings
- Bampis, Escoffier, and Teiller — MFCS 2019
  Multistage knapsack
- Bredereck, Fluschnik, and Kaczmarczyk — arXiv 2020
  Multistage committee election

New Perspective

classical multistage graph problems

bounding changes between solution sets for subsequent layers

***global* multistage graph problems**

bounding total number ℓ of changes between subsequent solutions

New Perspective

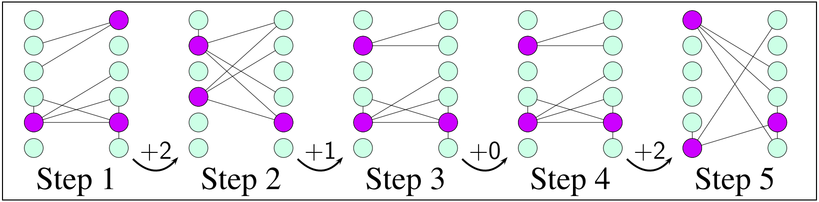
classical multistage graph problems

bounding changes between solution sets for subsequent layers

***global* multistage graph problems**

bounding total number ℓ of changes between subsequent solutions

GLOBAL MULTISTAGE VERTEX COVER with $\ell := 5$ and $k := 3$



New Perspective

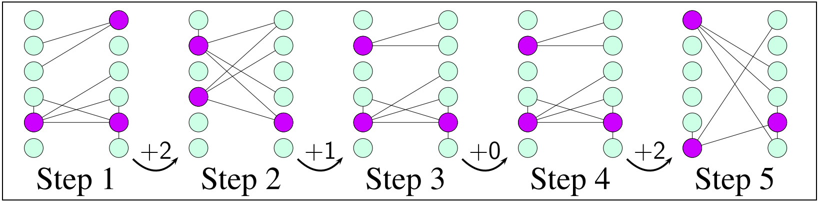
classical multistage graph problems

bounding changes between solution sets for subsequent layers

***global* multistage graph problems**

bounding total number ℓ of changes between subsequent solutions

Global Multistage Vertex Cover with $\ell := 5$ and $k := 3$



- Rohm — Bachelor Thesis 2018
  Poly-sized Kernel parameterized by $k + \tau$

Some Motivation for Global Multistage Problems

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

Some Motivation for Global Multistage Problems

- Placement of supply units, e.g., cranes, containers for employees.
- k := number of supply units $\quad \ell$:= relocation costs

Construction of BER airport. Source: youtube.com

Some Motivation for Global Multistage Problems

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

Some Motivation for Global Multistage Problems

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

Some Motivation for Global Multistage Problems

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

Some Motivation for Global Multistage Problems

- Placement of supply units, e.g., cranes, containers for employees.
- $k :=$ number of supply units $\qquad \ell :=$ relocation costs

Construction of BER airport. Source: youtube.com

New Results—TCS 2021

problem	k	$k+\ell$	$k+\tau$
Vertex Cover	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Path Contraction	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Cluster Editing	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Cluster Edge Del.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Planar Dom. Set	W[2]	W[2]†	?
Edge Dom. Set	W[2]	W[2]†	?
s-t-Path	W[1]	W[1]†	W[1]†
s-t-Cut	W[2]	W[2]†	?
Matching	W[1]	W[1]†	?

all problems: NP-hard w.r.t. parameter ℓ even if $\ell = O(1)$

$n :=$ #vertices and $\tau :=$ lifetime of the temporal graph
$k :=$ solution size, $\ell :=$ global budget

working space of all algorithms: $\text{poly}(k+\ell) + k \log n + \log \tau$ bits

$^{\dagger)}$ hardness even for $\ell = 0$, i.e., also for the classical multistage

New Results—TCS 2021

problem	k	$k+\ell$	$k+\tau$
Vertex Cover	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Path Contraction	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Cluster Editing	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Cluster Edge Del.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Planar Dom. Set	W[2]	W[2]†	?
Edge Dom. Set	W[2]	W[2]†	?
s-t-Path	W[1]	W[1]†	W[1]†
s-t-Cut	W[2]	W[2]†	?
Matching	W[1]	W[1]†	?

all problems: NP-hard w.r.t. parameter ℓ even if $\ell = O(1)$

$n :=$ #vertices and $\tau :=$ lifetime of the temporal graph
$k :=$ solution size, $\ell :=$ global budget

Optimal for $k, \ell = O(1)$.

working space of all algorithms: $\text{poly}(k+\ell) + k \log n + \log \tau$ bits

$^{\dagger)}$ hardness even for $\ell = 0$, i.e., also for the classical multistage

Some Definitions

Let f be a computable function and k the solution size.

graph property $\mathcal{P}_\mathcal{X}$ is *enumerable with size f*

$$\left|\left\{S \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\right\}\right| \leq f(k)$$

Some Definitions

Let f be a computable function and k the solution size.

graph property $\mathcal{P}_\mathcal{X}$ is *enumerable with size f*

$$\left|\left\{S \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\right\}\right| \leq f(k)$$

graph property $\mathcal{P}_\mathcal{X}$ is *superset-enumerable with size f*

For any given set F:

$$\left|\left\{S \supseteq F \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\right\}\right| \leq f(k)$$

Some Definitions

Let f be a computable function and k the solution size.

graph property $\mathcal{P}_{\mathcal{X}}$ is *enumerable with size f*

$\left|\left\{S \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_{\mathcal{X}}\right\}\right| \leq f(k)$

graph property $\mathcal{P}_{\mathcal{X}}$ is *superset-enumerable with size f*

For any given set F:

$\left|\left\{S \supseteq F \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_{\mathcal{X}}\right\}\right| \leq f(k)$

graph property $\mathcal{P}_{\mathcal{X}}$ is *monotone*

S *satisfies* $\mathcal{P}_{\mathcal{X}} \Rightarrow$ *any superset of* S *satisfies* $\mathcal{P}_{\mathcal{X}}$

Some Definitions

Let f be a computable function and k the solution size.

graph property $\mathcal{P}_\mathcal{X}$ is *enumerable with size f*

$\left|\left\{S \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\right\}\right| \leq f(k)$

graph property $\mathcal{P}_\mathcal{X}$ is *superset-enumerable with size f*

For any given set F:
$\left|\left\{S \supseteq F \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\right\}\right| \leq f(k)$

graph property $\mathcal{P}_\mathcal{X}$ is *monotone*

S satisfies $\mathcal{P}_\mathcal{X} \Rightarrow$ any superset of S satisfies $\mathcal{P}_\mathcal{X}$

full kernel

kernel contains all minimal solutions of the original instance

Some Definitions

Let f be a computable function and k the solution size.

graph property $\mathcal{P}_\mathcal{X}$ is *enumerable with size f*

$\left|\{S \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\}\right| \leq f(k)$

graph property $\mathcal{P}_\mathcal{X}$ is *superset-enumerable with size f*

For any given set F:
$\left|\{S \supseteq F \mid S \textit{ minimal solution of size} \leq k \textit{ satisfying } \mathcal{P}_\mathcal{X}\}\right| \leq f(k)$

graph property $\mathcal{P}_\mathcal{X}$ is *monotone*

S *satisfies* $\mathcal{P}_\mathcal{X} \Rightarrow$ *any superset of* S *satisfies* $\mathcal{P}_\mathcal{X}$

full kernel

kernel contains all minimal solutions of the original instance

monotone & full kernel of size $f(k) \Rightarrow$ (superset-)enum. with size $2^{f(k)}$

New Framework

$\mathcal{G}$: n-vertex temporal graph with τ time steps
parameters k : solution size, ℓ : global budget

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *graph property that is superset-enumerable with size f* $\Rightarrow$
time: $\mathrm{poly}(n)\tau\ell(k + f(k))^{2\ell+k+1}$
space: $O((k+\ell)\log f(k) + k\log n + \log\tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

New Framework

$\mathcal{G}$: n-vertex temporal graph with τ time steps
parameters k : solution size, ℓ : global budget

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *graph property that is superset-enumerable with size f* $\Rightarrow$
time: $\mathrm{poly}(n)\tau\ell(k+f(k))^{2\ell+k+1}$
space: $O((k+\ell)\log f(k) + k\log n + \log\tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

(monotone & full kernel $\Rightarrow$ (superset-)enumerable) $\Rightarrow$

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *monotone graph property with full kernel* $\Rightarrow \exists$ *function* f' :
time: $\mathrm{poly}(n)\tau\ell f'(k+\ell)$
space: $O(f'(k+\ell) + k\log n + \log\tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

New Framework

$\mathcal{G}$: n-vertex temporal graph with τ time steps
parameters k : solution size, ℓ : global budget

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *graph property that is superset-enumerable with size f* $\Rightarrow$
time: $\text{poly}(n)\tau\ell(k + f(k))^{2\ell+k+1}$
space: $O((k + \ell)\log f(k) + k \log n + \log \tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

(monotone & full kernel $\Rightarrow$ (superset-)enumerable) $\Rightarrow$

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *monotone graph property with full kernel* $\Rightarrow \exists$ *function* f' :
time: $\text{poly}(n)\tau\ell f'(k + \ell)$
space: $O(f'(k + \ell) + k \log n + \log \tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

Second framework: We can solve $\mathcal{P}_X$ with better bounds.

New Framework

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: graph property that is superset-enumerable with size f $\Rightarrow$
time: $\mathrm{poly}(n)\tau\ell(k + f(k))^{2\ell+k+1}$
space: $O((k + \ell) \log f(k) + k \log n + \log \tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

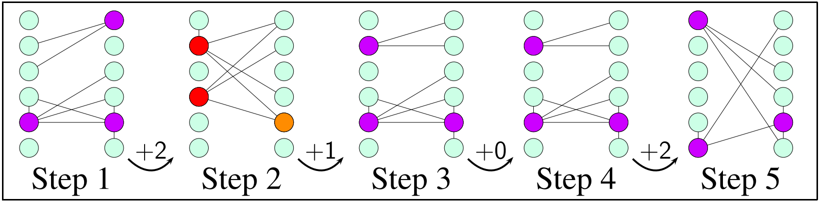
New Framework

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *graph property that is superset-enumerable with size f* $\Rightarrow$
time: $\mathrm{poly}(n)\tau\ell(k + f(k))^{2\ell+k+1}$
space: $O((k+\ell)\log f(k) + k\log n + \log\tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

key observation

$\exists$ *optimal solution where the solution S_i in every Step i:*
S_i is the union of a minimal solution for Step i and vertices of S_{i-1}



New Framework

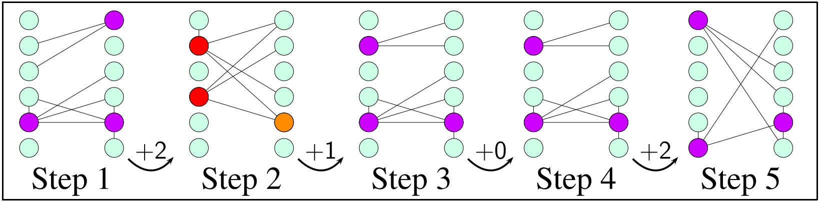
algorithm sketch:

for *step* $i := 1$ **to** τ **do**

1. *Restart loop with* $i+1$ *if solution* S_{i-1} *is solution for Step* i

key observation

$\exists$ *optimal solution where the solution* S_i *in every Step* i*:*
S_i *is the union of a minimal solution for Step* i *and vertices of* S_{i-1}



New Framework

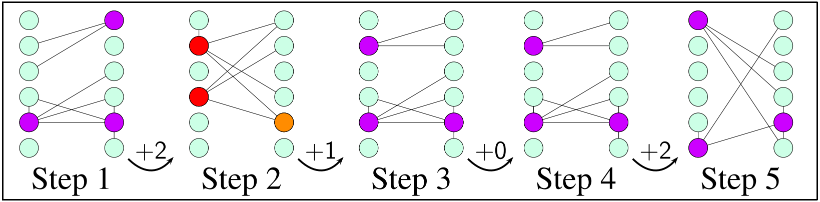
algorithm sketch:

for *step* $i := 1$ **to** τ **do**

1. *Restart loop with* $i+1$ *if solution* S_{i-1} *is solution for Step* i
2. *Number the minimal solutions* $S_i^1, \ldots, S_i^{f(k)}$ *in Step* i

key observation

$\exists$ *optimal solution where the solution* S_i *in every Step* i*:*
S_i *is the union of a minimal solution for Step* i *and vertices of* S_{i-1}



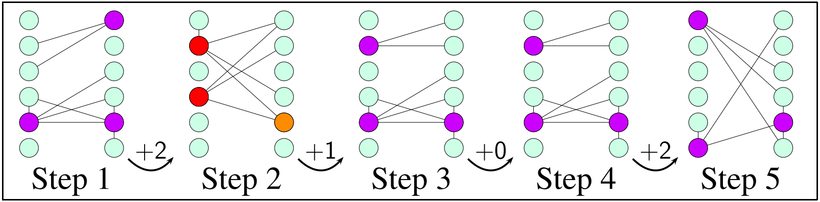
algorithm sketch:

for *step* $i := 1$ **to** τ **do**

1. *Restart loop with $i+1$ if solution S_{i-1} is solution for Step i*
2. *Number the minimal solutions $S_i^1, \ldots, S_i^{f(k)}$ in Step i*
3. *Guess: Solution S_i^j for Step i and take-over vertices of S_{i-1}*

key observation

$\exists$ optimal solution where the solution S_i in every Step i:
S_i is the union of a minimal solution for Step i and vertices of S_{i-1}



New Framework

algorithm sketch:

for *step* $i := 1$ **to** τ **do**

1. *Restart loop with* $i + 1$ *if solution* S_{i-1} *is solution for Step* i
2. *Number the minimal solutions* $S_i^1, \ldots, S_i^{f(k)}$ *in Step* i
3. *Guess:* *Solution* S_i^j *for Step* i *and take-over vertices of* S_{i-1}

run time and space bounds – sketch

ℓ *times we have to guess a number between* $1, \ldots, f(k)$
$(k+\ell)$ *such numbers & some numbers of size* $O(n)$ *and* $O(\tau)$

New Framework

algorithm sketch:

for *step* $i := 1$ **to** τ **do**

1. *Restart loop with* $i+1$ *if solution* S_{i-1} *is solution for Step* i
2. *Number the minimal solutions* $S_i^1, \ldots, S_i^{f(k)}$ *in Step* i
3. *Guess:* *Solution* S_i^j *for Step* i *and take-over vertices of* S_{i-1}

run time and space bounds – sketch

ℓ *times we have to guess a number between* $1, \ldots, f(k)$
$(k+\ell)$ *such numbers & some numbers of size* $O(n)$ *and* $O(\tau)$

global multistage problem for property $\mathcal{P}_X$ on $\mathcal{G}$

$\mathcal{P}_X$: *graph property that is superset-enumerable with size* $f \Rightarrow$
time: $\mathrm{poly}(n)\tau\ell(k+f(k))^{2\ell+k+1}$
space: $O((k+\ell)\log f(k) + k\log n + \log\tau)$ *bits*
$+$ *time/space needed to enumerate the solutions of the time steps*

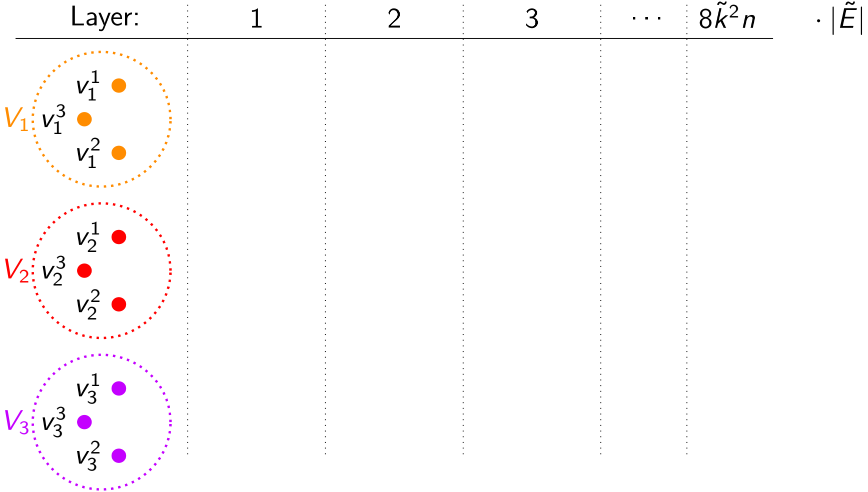
GM VERTEX COVER is W[1]-hard w.r.t. k

Red. to CLIQUE-instance $((\tilde{V}, \tilde{E}), \tilde{k})$.

Layer:	1	2	3	$\cdots$	$8\tilde{k}^2 n$	$\cdot \lvert\tilde{E}\rvert$

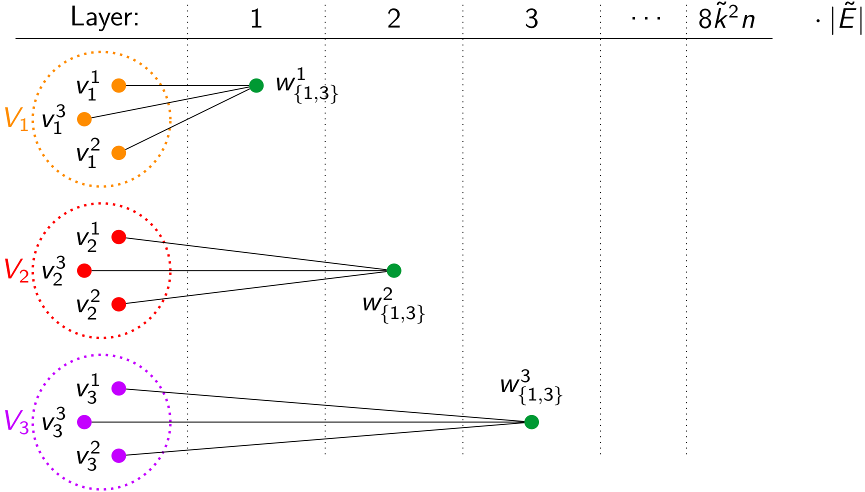
GM Vertex Cover is W[1]-hard w.r.t. k

Red. to Clique-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall$ $\tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$



Red. to Clique-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall$ $\tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$

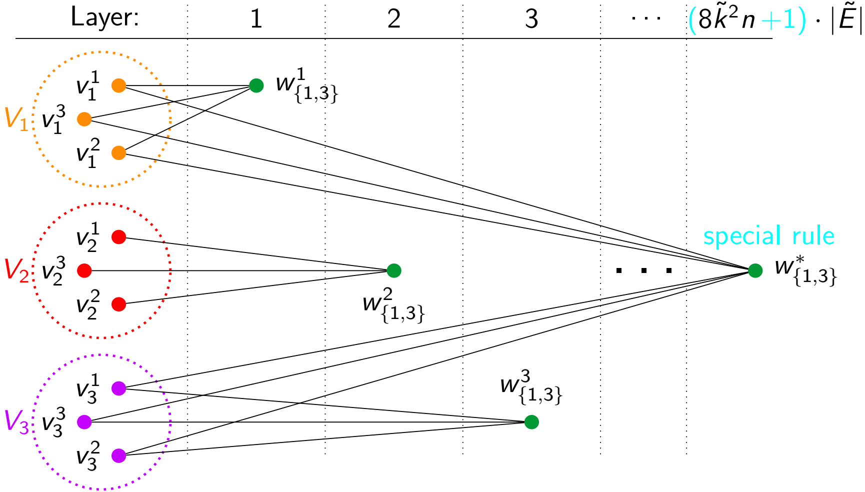
$\forall$ $\tilde{e}_{\{1,3\}}, \ldots \in \tilde{E}$: $8\tilde{k}^2 n$ vertices v_e^i connected to $V_{i \bmod n}$



GM Vertex Cover is W[1]-hard w.r.t. k

Red. to Clique-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall\ \tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$

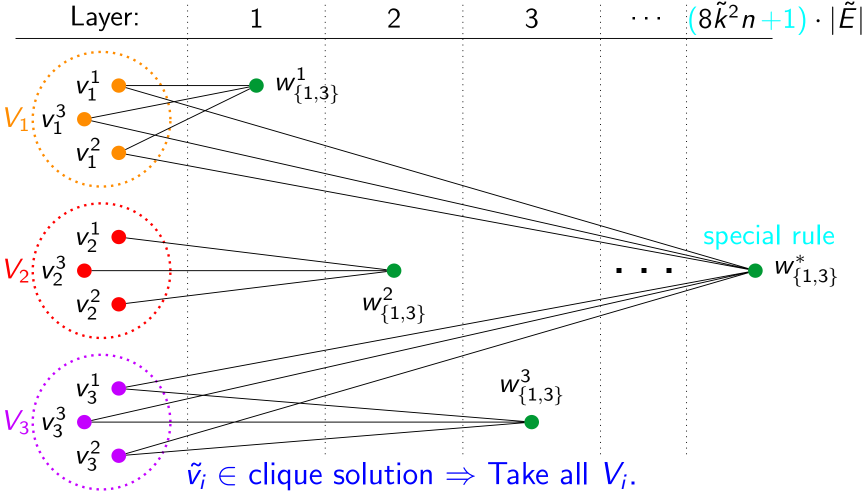
$\forall\ \tilde{e}_{\{1,3\}}, \ldots \in \tilde{E}$: $8\tilde{k}^2 n + 1$ vertices v_e^i connected to $V_{i \bmod n}$



GM Vertex Cover is W[1]-hard w.r.t. k

Red. to Clique-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall$ $\tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$

$\forall$ $\tilde{e}_{\{1,3\}}, \ldots \in \tilde{E}$: $8\tilde{k}^2 n + 1$ vertices v_e^i connected to $V_{i \bmod n}$

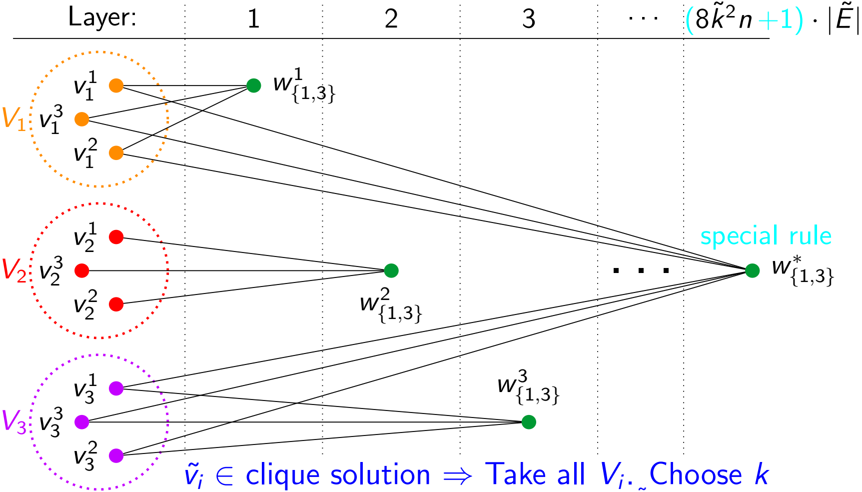


$\tilde{v}_i \in$ clique solution $\Rightarrow$ Take all V_i.

GM Vertex Cover is W[1]-hard w.r.t. k

Red. to Clique-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall\ \tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$

$\forall\ \tilde{e}_{\{1,3\}}, \ldots \in \tilde{E}$: $8\tilde{k}^2 n + 1$ vertices v_e^i connected to $V_{i \bmod n}$

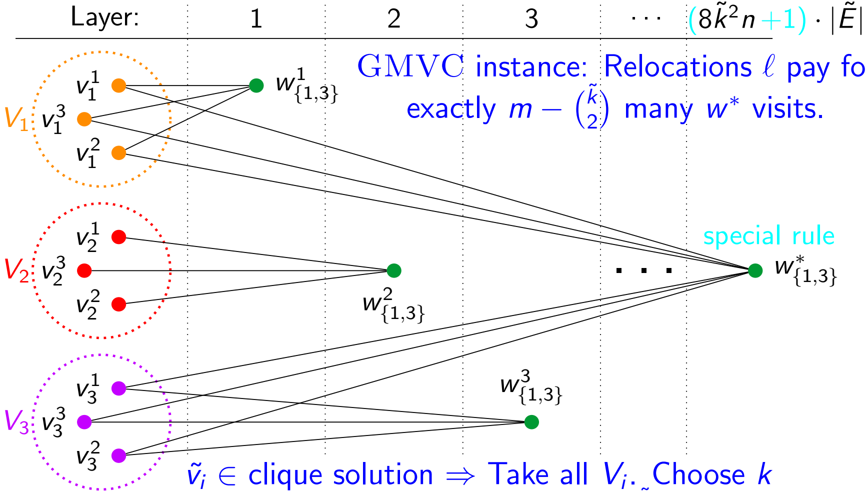


$\tilde{v}_i \in$ clique solution $\Rightarrow$ Take all V_i. Choose k s.t. one jumping vertex: must visit all except $\binom{\tilde{k}}{2}$ vertices w^*.

GM Vertex Cover is W[1]-hard w.r.t. k

Red. to Clique-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall\ \tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$

$\forall\ \tilde{e}_{\{1,3\}}, \ldots \in \tilde{E}$: $8\tilde{k}^2 n + 1$ vertices v_e^i connected to $V_{i \bmod n}$



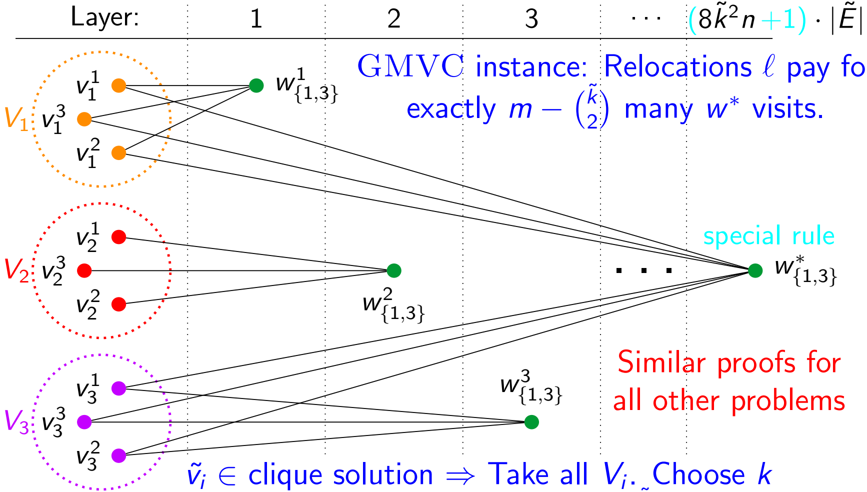
GMVC instance: Relocations ℓ pay for exactly $m - \binom{\tilde{k}}{2}$ many w^* visits.

$\tilde{v}_i \in$ clique solution $\Rightarrow$ Take all V_i. Choose k s.t. one jumping vertex: must visit all except $\binom{\tilde{k}}{2}$ vertices w^*.

GM VERTEX COVER is W[1]-hard w.r.t. k

Red. to CLIQUE-instance $((\tilde{V}, \tilde{E}), \tilde{k})$. $\forall\ \tilde{v}_1, \tilde{v}_2, \ldots \in \tilde{V}$: $V_1, V_2, \ldots$

$\forall\ \tilde{e}_{\{1,3\}}, \ldots \in \tilde{E}$: $8\tilde{k}^2 n + 1$ vertices v_e^i connected to $V_{i \bmod n}$



GMVC instance: Relocations ℓ pay for exactly $m - \binom{\tilde{k}}{2}$ many w^* visits.

Similar proofs for all other problems

$\tilde{v}_i \in$ clique solution $\Rightarrow$ Take all V_i. Choose k s.t. one jumping vertex: must visit all except $\binom{\tilde{k}}{2}$ vertices w^*.

New Results and Open Problems

problem	k	$k+\ell$	$k+\tau$
Vertex Cover	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Path Contraction	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Cluster Editing	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Cluster Edge Del.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Planar Dom. Set	W[2]	W[2]†	?
Edge Dom. Set	W[2]	W[2]†	?
s-t-Path	W[1]	W[1]†	W[1]†
s-t-Cut	W[2]	W[2]†	?
Matching	W[1]	W[1]†	?

$^{\dagger)}$ hardness even for $\ell = 0$, i.e., also for the classical multistage

Open Problems

- *Parameter $k+\tau$?*

New Results and Open Problems

problem	k	$k+\ell$	$k+\tau$
VERTEX COVER	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
PATH CONTRACTION	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
CLUSTER EDITING	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
CLUSTER EDGE DEL.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
PLANAR DOM. SET	W[2]	W[2]†	?
EDGE DOM. SET	W[2]	W[2]†	?
s-t-PATH	W[1]	W[1]†	W[1]†
s-t-CUT	W[2]	W[2]†	?
MATCHING	W[1]	W[1]†	?

†) hardness even for $\ell = 0$, i.e., also for the classical multistage

Open Problems

- *Parameter $k+\tau$? tw(underl.graph)+??*

New Results and Open Problems

problem	k	$k+\ell$	$k+\tau$
Vertex Cover	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Path Contraction	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Cluster Editing	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Cluster Edge Del.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Planar Dom. Set	W[2]	W[2]†	?
Edge Dom. Set	W[2]	W[2]†	?
s-t-Path	W[1]	W[1]†	W[1]†
s-t-Cut	W[2]	W[2]†	?
Matching	W[1]	W[1]†	?

$^{\dagger)}$ hardness even for $\ell = 0$, i.e., also for the classical multistage

Open Problems

- *Parameter $k+\tau$? tw(underl.graph)+??*
- *Weighted versions?*

New Results and Open Problems

problem	k	$k+\ell$	$k+\tau$
Vertex Cover	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Path Contraction	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
Cluster Editing	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Cluster Edge Del.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
Planar Dom. Set	W[2]	W[2]†	?
Edge Dom. Set	W[2]	W[2]†	?
s-t-Path	W[1]	W[1]†	W[1]†
s-t-Cut	W[2]	W[2]†	?
Matching	W[1]	W[1]†	?

$^{\dagger)}$ hardness even for $\ell = 0$, i.e., also for the classical multistage

Open Problems

- *Parameter $k+\tau$? tw(underl.graph)+??*
- *Weighted versions? Solve problems as* Independent Set, Feedback Vertex Set, Colourability, *etc.?*

New Results and Open Problems

problem	k	$k+\ell$	$k+\tau$
VERTEX COVER	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
PATH CONTRACTION	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)\log k)}$	poly. kernel
CLUSTER EDITING	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
CLUSTER EDGE DEL.	W[1]	$\text{poly}(n)\tau\ell 2^{O((k+\ell)k)}$	?
PLANAR DOM. SET	W[2]	W[2]†	?
EDGE DOM. SET	W[2]	W[2]†	?
s-t-PATH	W[1]	W[1]†	W[1]†
s-t-CUT	W[2]	W[2]†	?
MATCHING	W[1]	W[1]†	?

†) hardness even for $\ell = 0$, i.e., also for the classical multistage

Open Problems

- *Parameter $k+\tau$? tw(underl.graph)+??*
- *Weighted versions? Solve problems as* INDEPENDENT SET, FEEDBACK VERTEX SET, COLOURABILITY, *etc.?*
- *Modify algorithms above to run in para-L?*