UNIVERSITY OF WARSAW

Parameterized Complexity of MinCSP

Marcin Pilipczuk
m.pilipczuk@mimuw.edu.pl

based on joint work with
Eun Jung Kim, Stefan Kratsch, and Magnus Wahlström
and many others

European Research Council
Established by the European Commission

PACS, 2024-07-07

CSP dichotomy

Theorem (Bulatov, Zhuk, 2017)

For every finite D and Γ, the corresponding CSP is either NP-complete and polynomial time solvable.

CSP dichotomy

Theorem (Bulatov, Zhuk, 2017)

For every finite D and Γ, the corresponding CSP is either NP-complete and polynomial time solvable.

$\textsc{MinCSP}(D, \Gamma)$: Can you delete k constraints to make the instance satisfiable?

- Interesting if $\text{CSP}(D, \Gamma)$ is P-time.
- Trivial $n^{\mathcal{O}(k)}$ algorithm.
- Is it FPT parameterized by k?

Min SAT

MinCSP(D, Γ): Can you delete k constraints to make the instance satisfiable?

Min SAT

MinCSP(D, Γ): Can you delete k constraints to make the instance satisfiable?

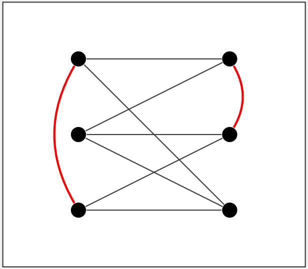
- $D = \{0, 1\}, \Gamma = \{x \neq y\}$.

Min SAT

UNIVERSITY OF WARSAW

MINCSP(D, Γ): Can you delete k constraints to make the instance satisfiable?

- $D = \{0, 1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!



Min SAT

MinCSP(D, Γ): Can you delete k constraints to make the instance satisfiable?

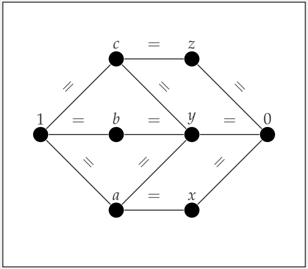
- $D = \{0, 1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0, 1\}, \Gamma = \{x = 0, x = 1, x = y\}$.

Min SAT

UNIVERSITY OF WARSAW

MinCSP(D,Γ): Can you delete k constraints to make the instance satisfiable?

- $D = \{0,1\}, \Gamma = \{x \neq y\}$.
    - Edge Bipartization!
- $D = \{0,1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
    - Undirected Minimum Cut!



Min SAT

UNIVERSITY OF WARSAW

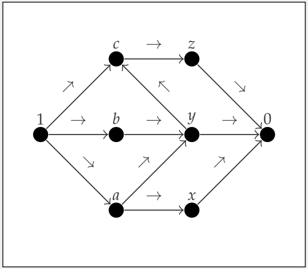
$\textsc{MinCSP}(D,\Gamma)$: Can you delete k constraints to make the instance satisfiable?

- $D = \{0,1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0,1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
  - Undirected Minimum Cut!
- $D = \{0,1\}, \Gamma = \{x \to 0, 1 \to x, x \to y\}$.

Min SAT

MINCSP(D,Γ): Can you delete k constraints to make the instance satisfiable?

- $D = \{0,1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0,1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
  - Undirected Minimum Cut!
- $D = \{0,1\}, \Gamma = \{x \to 0, 1 \to x, x \to y\}$.
  - Directed Minimum Cut!



Min SAT

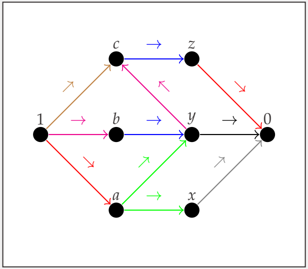
MinCSP(D, Γ): Can you delete k constraints to make the instance satisfiable?

- $D = \{0, 1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0, 1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
  - Undirected Minimum Cut!
- $D = \{0, 1\}, \Gamma = \{x \to 0, 1 \to x, x \to y\}$.
  - Directed Minimum Cut!
- $D = \{0, 1\}, \Gamma = \{x \to 0, 1 \to x, (x \to y) \wedge (u \to v)\}$.

Min SAT

UNIVERSITY OF WARSAW

MINCSP(D,Γ): Can you delete k constraints to make the instance satisfiable?



- $D = \{0,1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0,1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
  - Undirected Minimum Cut!
- $D = \{0,1\}, \Gamma = \{x \to 0, 1 \to x, x \to y\}$.
  - Directed Minimum Cut!
- $D = \{0,1\}, \Gamma = \{x \to 0, 1 \to x, (x \to y) \wedge (u \to v)\}$.
  - Bundled Cut with bundles of size 2!
    W[1]-hard (Marx, Razgon 2009)

Min SAT

UNIVERSITY OF WARSAW

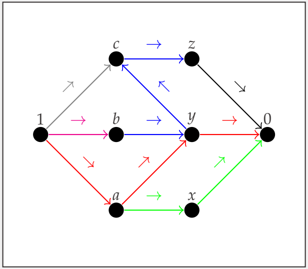
$\textsc{MinCSP}(D, \Gamma)$: Can you delete k constraints to make the instance satisfiable?

- $D = \{0, 1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0, 1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
  - Undirected Minimum Cut!
- $D = \{0, 1\}, \Gamma = \{x \to 0, 1 \to x, x \to y\}$.
  - Directed Minimum Cut!
- $D = \{0, 1\}, \Gamma = \{x \to 0, 1 \to x, (x \to y) \wedge (u \to v)\}$.
  - Bundled Cut with bundles of size 2!
    W[1]-hard (Marx, Razgon 2009)
- $D = \{0, 1\}$,
  $\Gamma = \{x \to 0, 1 \to x, (x \to y) \wedge (y \to z) \wedge (z \to v)\}$.

Min SAT

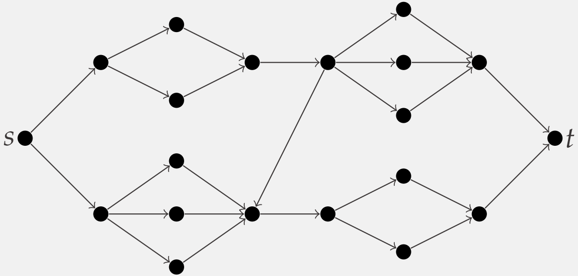
UNIVERSITY OF WARSAW

$\textsc{MinCSP}(D, \Gamma)$: Can you delete k constraints to make the instance satisfiable?



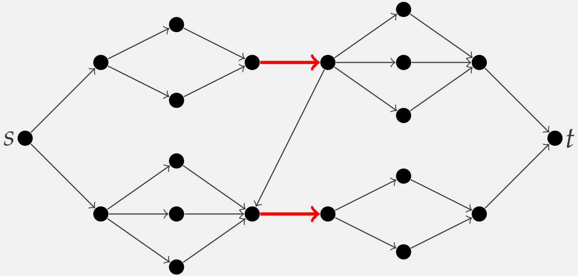
- $D = \{0, 1\}, \Gamma = \{x \neq y\}$.
  - Edge Bipartization!
- $D = \{0, 1\}, \Gamma = \{x = 0, x = 1, x = y\}$.
  - Undirected Minimum Cut!
- $D = \{0, 1\}, \Gamma = \{x \to 0, 1 \to x, x \to y\}$.
  - Directed Minimum Cut!
- $D = \{0, 1\}, \Gamma = \{x \to 0, 1 \to x, (x \to y) \wedge (u \to v)\}$.
  - Bundled Cut with bundles of size 2!
    W[1]-hard (Marx, Razgon 2009)
- $D = \{0, 1\}$,
  $\Gamma = \{x \to 0, 1 \to x, (x \to y) \wedge (y \to z) \wedge (z \to v)\}$.
  - 3-Chain SAT!
    FPT status was open for some time.

Minimum (s,t)-cut

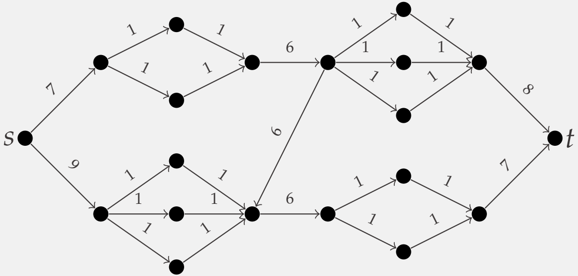


- Minimum (s,t)-cut problem: P-time.

Minimum (s, t)-cut

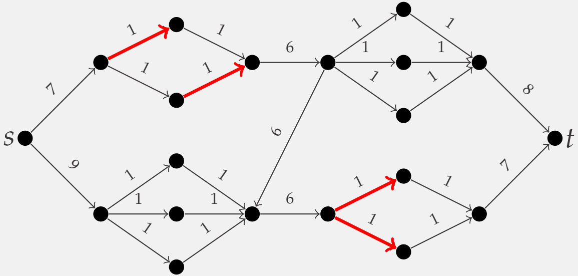


- Minimum (s, t)-cut problem: P-time.



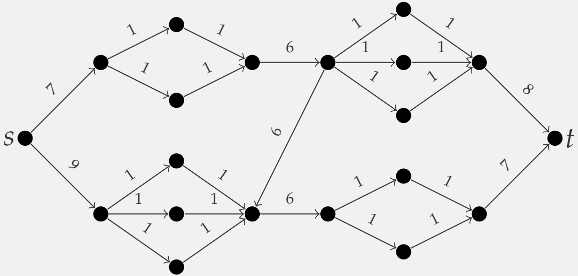
- Minimum (s, t)-cut problem: P-time.
- Still P-time if you add weights (capacities).

Minimum (s,t)-cut



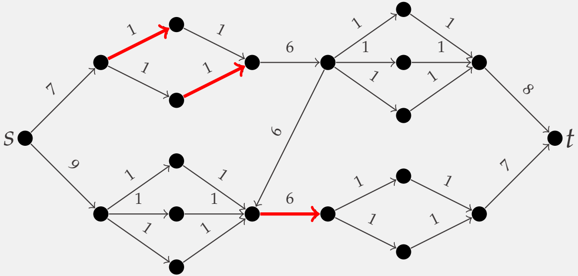
- Minimum (s,t)-cut problem: P-time.
- Still P-time if you add weights (capacities).

Minimum (s, t)-cut



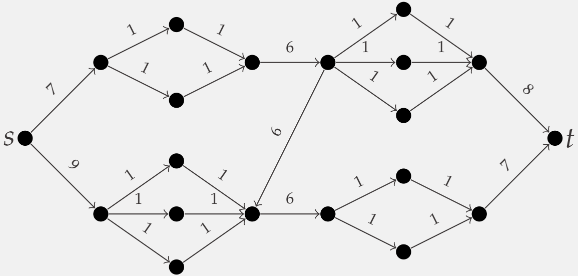
- Minimum (s, t)-cut problem: P-time.
- Still P-time if you add weights (capacities).
- What about bi-objective?
  - Cardinality $\leq k$. (Figure: $k = 3$.)
  - Minimum weight.

Minimum (s,t)-cut



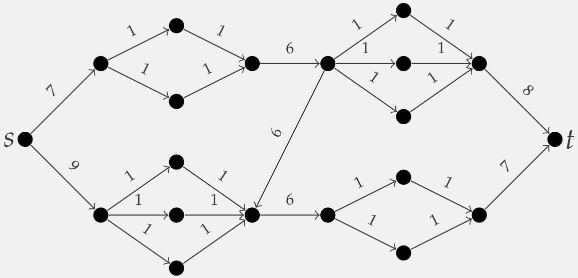
- Minimum (s,t)-cut problem: P-time.
- Still P-time if you add weights (capacities).
- What about bi-objective?
  - Cardinality $\leq k$. (Figure: $k = 3$.)
  - Minimum weight.

Minimum (s,t)-cut



- Minimum (s,t)-cut problem: P-time.
- Still P-time if you add weights (capacities).
- What about bi-objective?
    - Cardinality $\leq k$. (Figure: $k = 3$.)
    - Minimum weight.
- This version is NP-hard.

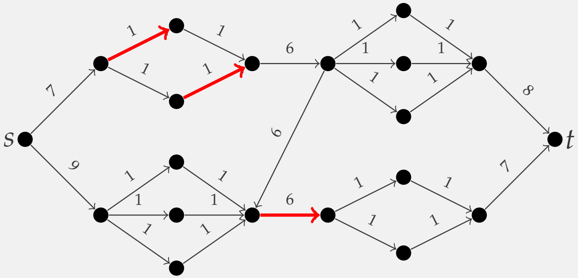
Bi-objective (s, t)-cut



Bi-objective (s, t)-Cut

Input: digraph $G, s, t \in V(G), \omega : E(G) \to \mathbb{Z}_+, k, W \in \mathbb{Z}$.
Question: is there an (s, t)-cut Z with $|Z| \leq k$ and $\omega(Z) \leq W$.

Bi-objective (s, t)-cut

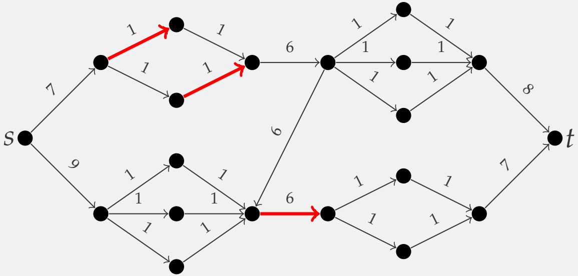


Bi-objective (s, t)-Cut

Input: digraph $G, s, t \in V(G), \omega : E(G) \to \mathbb{Z}_+, k, W \in \mathbb{Z}$.
Question: is there an (s, t)-cut Z with $|Z| \leq k$ and $\omega(Z) \leq W$.

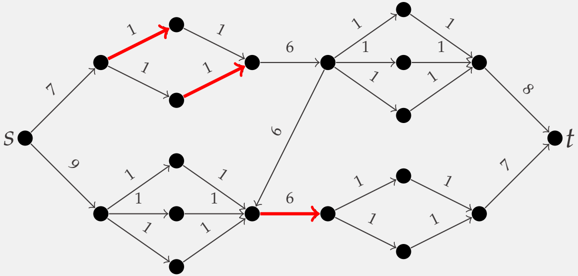
- Figure: $k = 3, W = 8 \longrightarrow$ YES.

Bi-objective (s,t)-cut

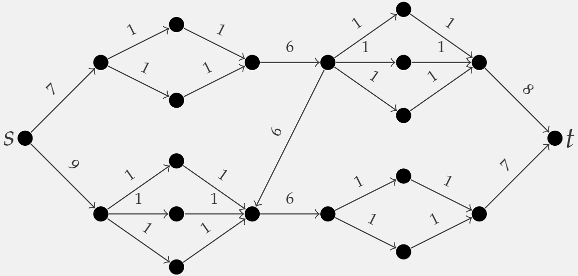


- NP-hard.
- FPT when parameterized by $k + W$.
    - Multi-budgeted important separators.
    - Kratsch, Li, Marx, P., Wahlström, IPEC 2018.

Bi-objective (s,t)-cut

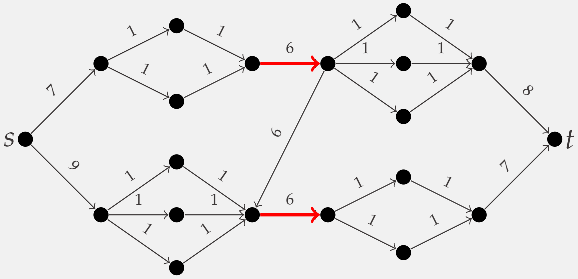


- NP-hard.
- FPT when parameterized by $k + W$.
  - Multi-budgeted important separators.
  - Kratsch, Li, Marx, P., Wahlström, IPEC 2018.
- Parameterization by k only?
  - Undirected: can hammer down with randomized contractions / treewidth reductions.

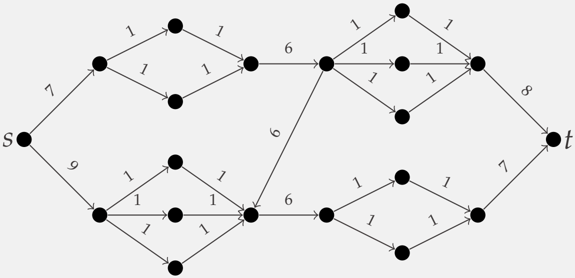


- Important observation: P-time for $k = \lambda_G(s,t)$.
  - Example: $k = 2$.

Bi-objective (s, t)-cut



- Important observation: P-time for $k = \lambda_G(s, t)$.
    - Example: $k = 2$.



- Important observation: P-time for $k = \lambda_G(s,t)$.
  - Example: $k = 2$.
- Set

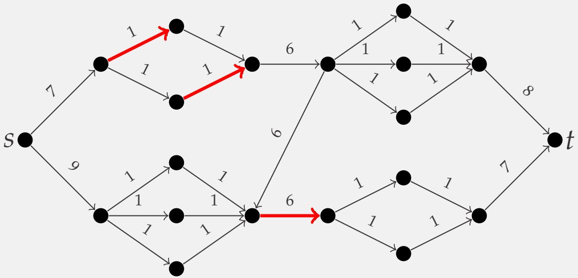
$$M := 1 + \sum_{e \in E(G)} \omega(e),$$

$$\mathrm{cap}(e) := M + \omega(e),$$

and ask for cut of capacity $\leq kM + W$.

Directed flow-augmentation

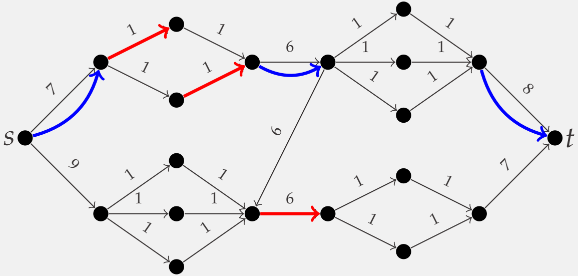
UNIVERSITY OF WARSAW



Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s, t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s,t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s,t)-cut in $G + A$.

Directed flow-augmentation



Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s, t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s,t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s,t)-cut in $G + A$.

Solving bi-objective (s,t)-cut

Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s,t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s,t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s,t)-cut in $G + A$.

- Repeat $2^{\mathcal{O}(k^4 \log k)}$ times:
  - Invoke flow-augmentation, obtaining A.
  - Find min-weight solution of cardinality $\lambda_{G+A}(s,t)$ in $G + A$.
- Bi-objective (s,t)-cut is randomized FPT when parameterized by k.

Solving bi-objective (s, t)-cut

Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s, t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s, t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s, t)-cut in $G + A$.

- Exponent not optimized, but seems hard to get $o(k^2)$.

Solving bi-objective (s, t)-cut

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s, t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s, t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s, t)-cut in $G + A$.

- Exponent not optimized, but seems hard to get $o(k^2)$.
- No real obstacle to derandomize.
    - Stack of color-coding tricks.
    - Derandomization on arXiv.

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s, t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s,t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s,t)-cut in $G + A$.

- Exponent not optimized, but seems hard to get $o(k^2)$.
- No real obstacle to derandomize.
  - Stack of color-coding tricks.
  - Derandomization on arXiv.
- Works in slightly more general setting of *star (s,t)-cut*.
  - (s,t)-cut whose every arc goes from reachable-from-s to non-reachable-from-s.

Solving bi-objective (s, t)-cut

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström)

There exists a randomized polynomial-time algorithm that, given a digraph G, $s, t \in V(G)$, and $k \in \mathbb{Z}$, outputs $A \subseteq V(G) \times V(G)$ so that for every minimal (s, t)-cut Z of $|Z| \leq k$, with probability $2^{-\mathcal{O}(k^4 \log k)}$ the set Z is a minimum (s, t)-cut in $G + A$.

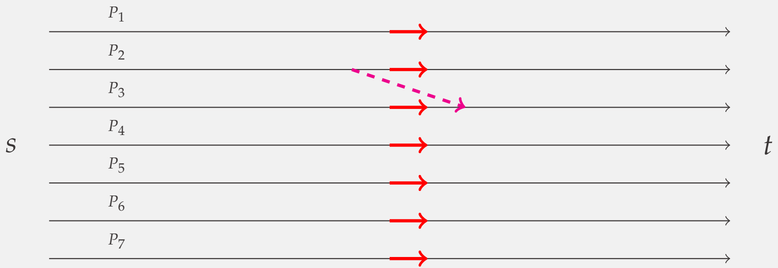
- Exponent not optimized, but seems hard to get $o(k^2)$.
- No real obstacle to derandomize.
  - Stack of color-coding tricks.
  - Derandomization on arXiv.
- Works in slightly more general setting of *star* (s, t)*-cut*.
  - (s, t)-cut whose every arc goes from reachable-from-s to non-reachable-from-s.
- Undirected graphs: $2^{-\mathcal{O}(k \log k)}$ success probability.

PoV: space of all mincuts

PoV: space of all mincuts

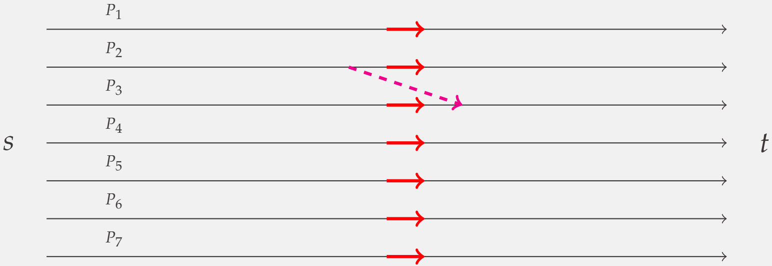
- Minimum cut: one edge per flow path.
  Path $P_i \sim$ variable x_i where it is cut.

PoV: space of all mincuts



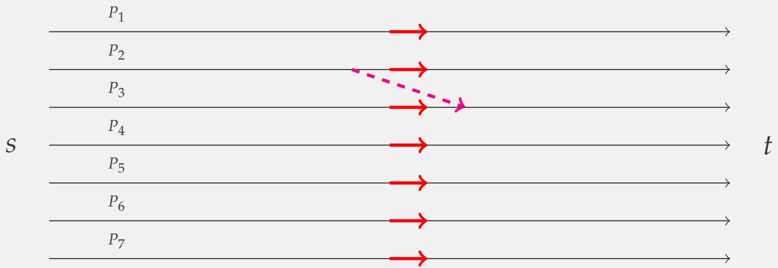
- Minimum cut: one edge per flow path.
  Path $P_i \sim$ variable x_i where it is cut.
- Path Q from $u \in P_i$ to $v \in P_j$, edge-disjoint with the flow.

PoV: space of all mincuts



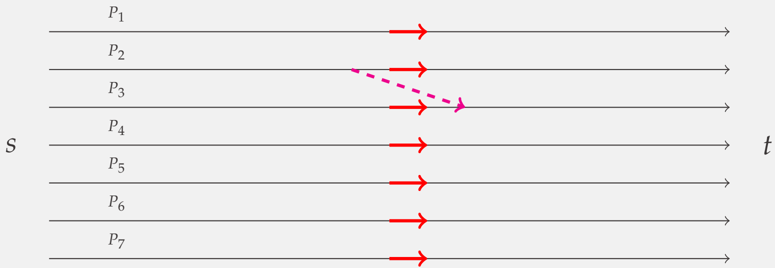
- Minimum cut: one edge per flow path.
  Path $P_i \sim$ variable x_i where it is cut.
- Path Q from $u \in P_i$ to $v \in P_j$, edge-disjoint with the flow.
- $x_i \leq u \vee x_j \geq v$.

PoV: space of all mincuts



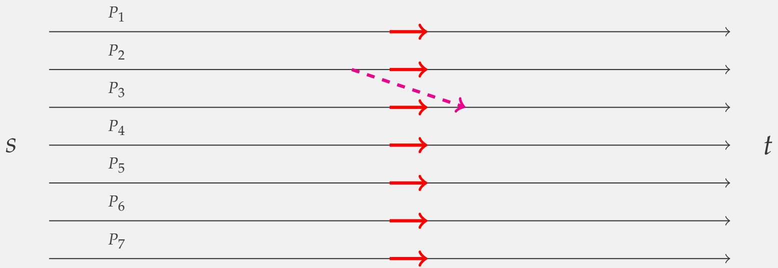
- Minimum cut: one edge per flow path.
  Path $P_i \sim$ variable x_i where it is cut.
- Path Q from $u \in P_i$ to $v \in P_j$, edge-disjoint with the flow.
- $x_i \leq u \vee x_j \geq v$.
- Write such clause for every choice of (i, j, u, v) for $\exists Q$.

PoV: space of all mincuts

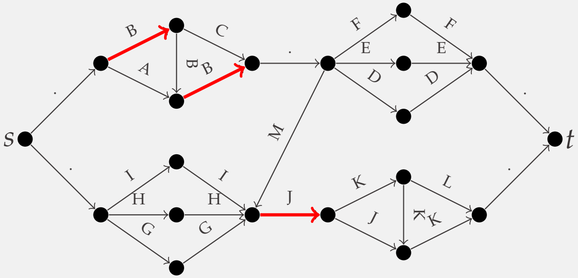


- Minimum cut: one edge per flow path.
  Path $P_i \sim$ variable x_i where it is cut.
- Path Q from $u \in P_i$ to $v \in P_j$, edge-disjoint with the flow.
- $x_i \leq u \vee x_j \geq v$.
- Write such clause for every choice of (i, j, u, v) for $\exists Q$.
- Space of all CSP solutions $\equiv$ space of all *st*-mincuts.

PoV: space of all mincuts



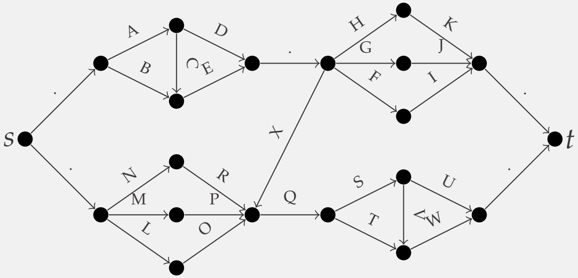
- Minimum cut: one edge per flow path.
  Path $P_i \sim$ variable x_i where it is cut.
- Path Q from $u \in P_i$ to $v \in P_j$, edge-disjoint with the flow.
- $x_i \leq u \vee x_j \geq v$.
- Write such clause for every choice of (i, j, u, v) for $\exists Q$.
- Space of all CSP solutions $\equiv$ space of all st-mincuts.
- Flow-augmentation: cover the space of *minimal st*-cuts of size $\leq k$ with small number of such CSP instances.



Bundled cut

Input: digraph G, $s, t \in V(G)$, $k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

Figure: solution of cost 2.

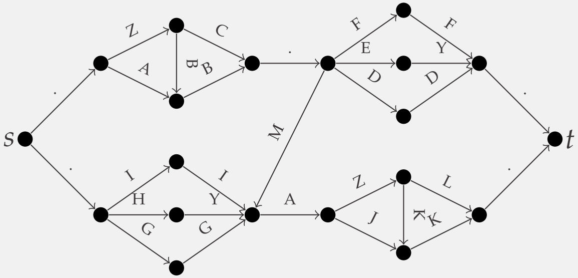


Bundled cut

Input: digraph G, $s, t \in V(G)$, $k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

All $B \in \mathcal{B}$ singletons $\longrightarrow$ minimum (s,t)-cut.

Bundled Cut

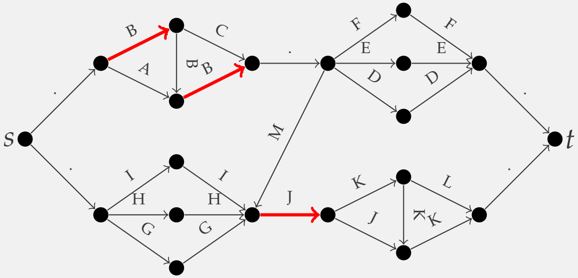


Bundled cut

Input: digraph $G, s, t \in V(G), k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

Every $B \in \mathcal{B}$ of size at most 2 $\longrightarrow$ $W[1]$-hard with param. k.

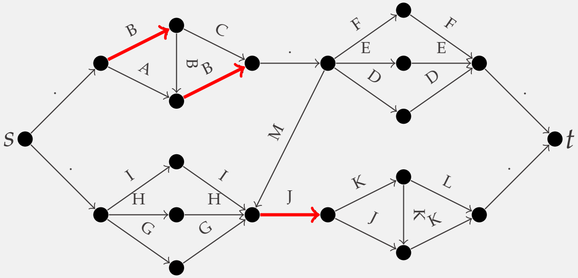
Bundled Cut



Bundled cut

Input: digraph $G, s, t \in V(G), k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

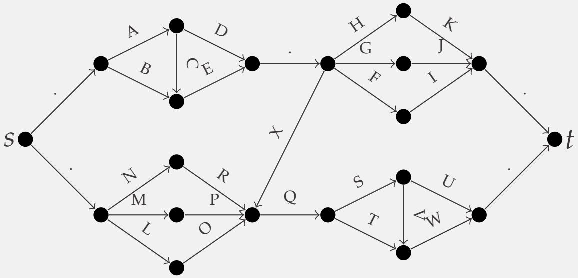
Every $B \in \mathcal{B}$ is a path of length $\leq \ell \longrightarrow \ell$-CHAIN SAT.



ℓ-Chain SAT

Input: digraph $G, s, t \in V(G), k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$, each being a path of length $\leq \ell$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

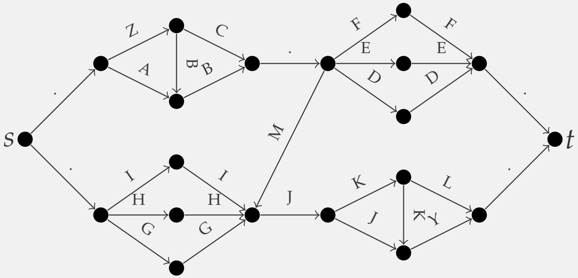
Figure: $\ell = 3$.



ℓ-Chain SAT

Input: digraph $G, s, t \in V(G), k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$, each being a path of length $\leq \ell$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

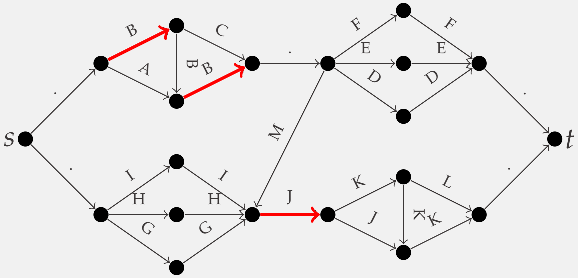
Case $\ell = 1$: Minimum (s,t)-Cut



ℓ-Chain SAT

Input: digraph G, $s, t \in V(G)$, $k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$, each being a path of length $\leq \ell$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

Case $\ell = 2$: still Minimum (s,t)-Cut.

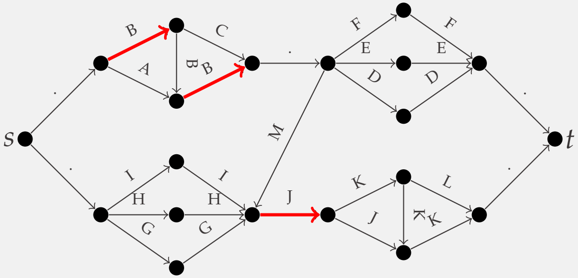


ℓ-Chain SAT

Input: digraph $G, s, t \in V(G), k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$, each being a path of length $\leq \ell$.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

Case $\ell = 3$: starts to be interesting!

Chain SAT

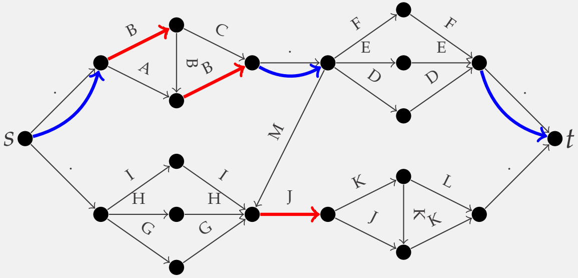


3-Chain SAT

Input: digraph G, $s, t \in V(G)$, $k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$, each being a path of length ≤ 3.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

Focus on case $\ell = 3$.

Chain SAT



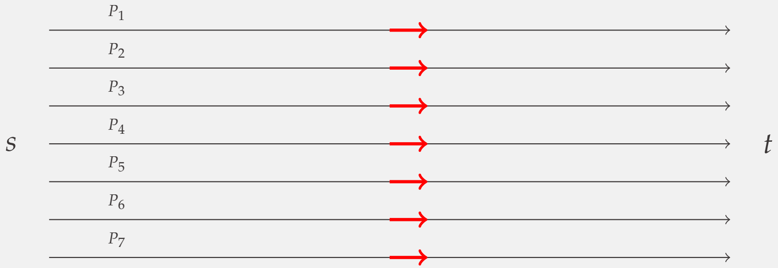
3-Chain SAT

Input: digraph $G, s, t \in V(G), k \in \mathbb{Z}$, family $\mathcal{B}$ of pairwise disjoint subsets of $E(G)$, each being a path of length ≤ 3.
Question: does there exist a minimal (s,t)-cut $Z \subseteq \bigcup \mathcal{B}$ with $|\{B \in \mathcal{B} \mid B \cap Z \neq \emptyset\}| \leq k$.

Flow-augmentation: focus on the case Z is a mincut ($|Z| \leq 2k$).

3-Chain SAT

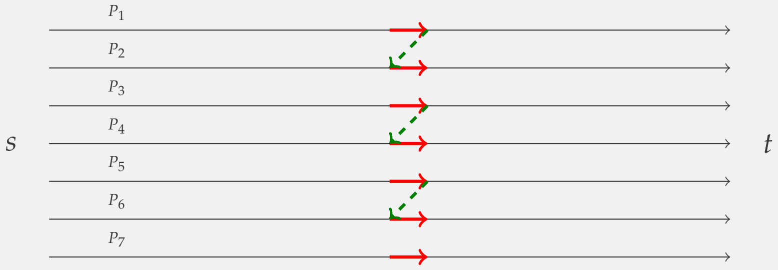
3-Chain SAT



3-Chain SAT

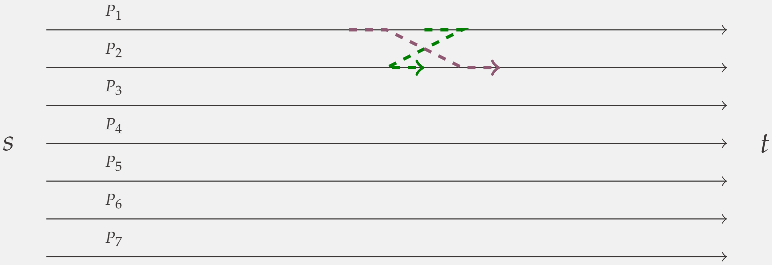
UNIVERSITY OF WARSAW

- Guess how cuts on paths group in $\mathcal{B}$.



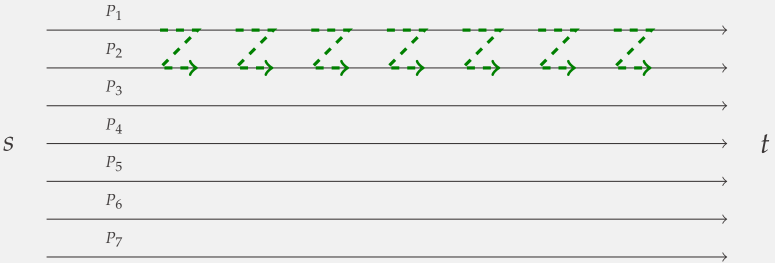
- Guess how cuts on paths group in $\mathcal{B}$.

3-Chain SAT



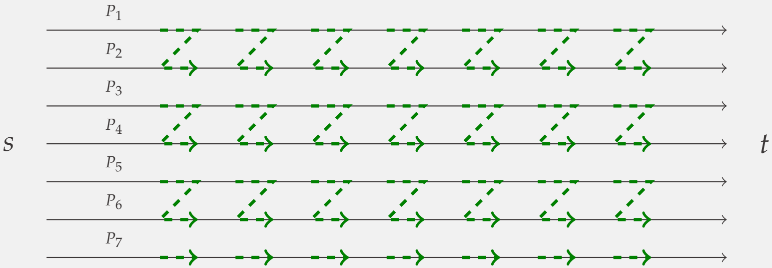
- Guess how cuts on paths group in $\mathcal{B}$.
- On one pair, candidates cannot cross.

3-Chain SAT

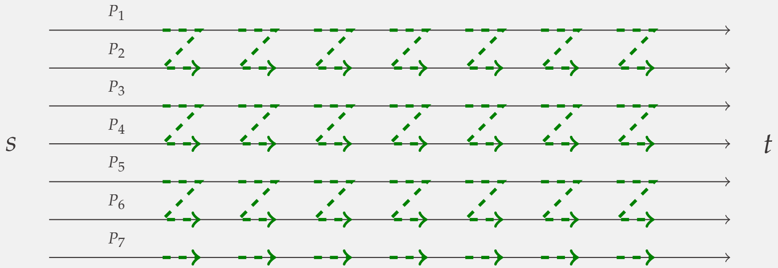


- Guess how cuts on paths group in $\mathcal{B}$.
- On one pair, candidates cannot cross.

3-Chain SAT



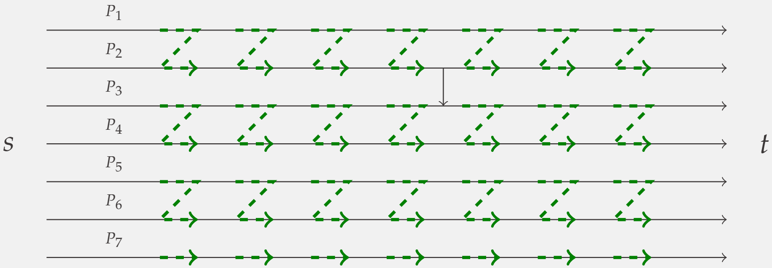
- Guess how cuts on paths group in $\mathcal{B}$.
- On one pair, candidates cannot cross.



- Guess how cuts on paths group in $\mathcal{B}$.
- On one pair, candidates cannot cross.
- Collapse pair into one flow path.

3-Chain SAT

UNIVERSITY OF WARSAW



- Guess how cuts on paths group in $\mathcal{B}$.
- On one pair, candidates cannot cross.
- Collapse pair into one flow path.
- The same (P-time) description as the space of all mincuts.

ℓ-Chain SAT

Theorem (Kim, Kratsch, P., Wahlström)

CHAIN ℓ-SAT *is FPT when param. by k and* ℓ.

Works also in the weighted setting.
(Every $B \in \mathcal{B}$ has its weight and we do not want to exceed total weight budget.)

ℓ-Chain SAT

Theorem (Kim, Kratsch, P., Wahlström)

CHAIN ℓ-SAT *is FPT when param. by k and ℓ.*

Works also in the weighted setting.
(Every $B \in \mathcal{B}$ has its weight and we do not want to exceed total weight budget.)

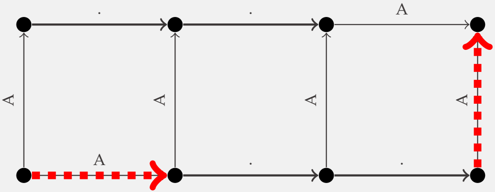
Works for slightly more general problem.

Theorem (Kim, Kratsch, P., Wahlström)

(WEIGHTED) BUNDLED CUT WITH PAIRWISE LINKED DELETABLE EDGES *is FPT when param. by k and maximum bundle size.*

For every $B \in \mathcal{B}$ and $e, f \in B$, there exists a directed path between an endpoint of e and an endpoint of f (in one of the directions) that does not use edge of a different bundle.

ℓ-Chain SAT



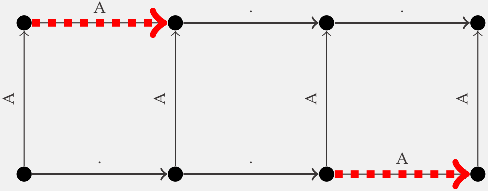
Works for slightly more general problem.

Theorem (Kim, Kratsch, P., Wahlström)

(WEIGHTED) BUNDLED CUT WITH PAIRWISE LINKED DELETABLE EDGES *is FPT when param. by k and maximum bundle size.*

For every $B \in \mathcal{B}$ and $e, f \in B$, there exists a directed path between an endpoint of e and an endpoint of f (in one of the directions) that does not use edge of a different bundle.

ℓ-Chain SAT

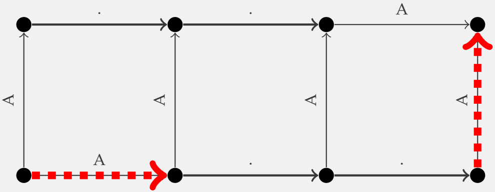


Works for slightly more general problem.

Theorem (Kim, Kratsch, P., Wahlström)

(Weighted) Bundled Cut with Pairwise Linked Deletable Edges *is FPT when param. by k and maximum bundle size.*

For every $B \in \mathcal{B}$ and $e, f \in B$, there exists a directed path between an endpoint of e and an endpoint of f (in one of the directions) that does not use edge of a different bundle.



Works for slightly more general problem.

Theorem (Kim, Kratsch, P., Wahlström)

(WEIGHTED) BUNDLED CUT WITH PAIRWISE LINKED DELETABLE EDGES *is FPT when param. by k and maximum bundle size.*

For every $B \in \mathcal{B}$ and $e, f \in B$, there exists a directed path between an endpoint of e and an endpoint of f (in one of the directions) that does not use edge of a different bundle.

Min SAT dichotomy

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- MIN SAT(Γ): For fixed boolean language Γ, given an instance and an integer k, can one delete k constraints to make it satisfiable?

Min SAT dichotomy

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- MIN SAT(Γ): For fixed boolean language Γ, given an instance and an integer k, can one delete k constraints to make it satisfiable?
- ℓ-CHAIN SAT was a roadblock.
  - constraints of the form $(x_0 \to x_1 \to \ldots \to x_\ell)$;
  - shorthand for $(x_0 \to x_1) \vee (x_1 \to x_2) \vee \ldots \vee (x_{\ell-1} \to x_\ell)$;

Min SAT dichotomy

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- MIN SAT(Γ): For fixed boolean language Γ, given an instance and an integer k, can one delete k constraints to make it satisfiable?
- ℓ-CHAIN SAT was a roadblock.
  - constraints of the form $(x_0 \to x_1 \to \ldots \to x_\ell)$;
  - shorthand for $(x_0 \to x_1) \vee (x_1 \to x_2) \vee \ldots \vee (x_{\ell-1} \to x_\ell)$;
- Recall: $\Gamma = \{1 \to x, x \to 0, (x \to y) \wedge (u \to v)\}$ is W[1]-hard.

Min SAT dichotomy

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- MIN SAT(Γ): For fixed boolean language Γ, given an instance and an integer k, can one delete k constraints to make it satisfiable?
- ℓ-CHAIN SAT was a roadblock.
    - constraints of the form $(x_0 \to x_1 \to \ldots \to x_\ell)$;
    - shorthand for $(x_0 \to x_1) \vee (x_1 \to x_2) \vee \ldots \vee (x_{\ell-1} \to x_\ell)$;
- Recall: $\Gamma = \{1 \to x, x \to 0, (x \to y) \wedge (u \to v)\}$ is W[1]-hard.
- $\Gamma = \{1 \to x, x \to 0, (x \to y) \wedge (u \to v) \wedge (\neg x \vee \neg u)\}$???

Min SAT dichotomy

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- MIN SAT(Γ): For fixed boolean language Γ, given an instance and an integer k, can one delete k constraints to make it satisfiable?
- ℓ-CHAIN SAT was a roadblock.
  - constraints of the form $(x_0 \to x_1 \to \ldots \to x_\ell)$;
  - shorthand for $(x_0 \to x_1) \vee (x_1 \to x_2) \vee \ldots \vee (x_{\ell-1} \to x_\ell)$;
- Recall: $\Gamma = \{1 \to x, x \to 0, (x \to y) \wedge (u \to v)\}$ is W[1]-hard.
- $\Gamma = \{1 \to x, x \to 0, (x \to y) \wedge (u \to v) \wedge (\neg x \vee \neg u)\}$???
  - FPT!

Min SAT dichotomy

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- Post's lattice: meet with all old friends.

Min SAT dichotomy

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- Post's lattice: meet with all old friends.
- Luckily, this has been mostly done by (Bonnet, Egri, Marx, ESA 2016).
  - They provided *FPT constant-approximation* dichotomy.

Min SAT dichotomy

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- Post's lattice: meet with all old friends.
- Luckily, this has been mostly done by (Bonnet, Egri, Marx, ESA 2016).
    - They provided *FPT constant-approximation* dichotomy.
- Two "clones" to investigate.
    - ID_2: constraints definable using 2SAT formulae.
    - IS_{10}^{d}: constraints definable using constants, implications, and d-ary clauses will all variables negated.

Min SAT dichotomy

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- Post's lattice: meet with all old friends.
- Luckily, this has been mostly done by (Bonnet, Egri, Marx, ESA 2016).
  - They provided *FPT constant-approximation* dichotomy.
- Two "clones" to investigate.
  - ID_2: constraints definable using 2SAT formulae.
  - IS_{10}^{d}: constraints definable using constants, implications, and d-ary clauses will all variables negated.
- Main algorithmic part: Each of these contain a new tractability island.

Min SAT dichotomy

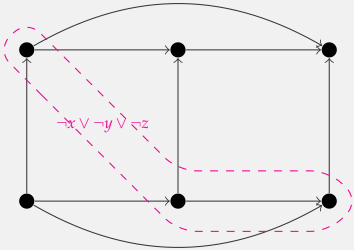
UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström, SODA 2023)

FPT/W[1]-hard dichotomy for MIN SAT *for boolean domain.*

- Post's lattice: meet with all old friends.
- Luckily, this has been mostly done by (Bonnet, Egri, Marx, ESA 2016).
  - They provided *FPT constant-approximation* dichotomy.
- Two "clones" to investigate.
  - ID_2: constraints definable using 2SAT formulae.
  - IS^d_{10}: constraints definable using constants, implications, and d-ary clauses will all variables negated.
- Main algorithmic part: Each of these contain a new tractability island.
- Hardness: Marx-Razgon reduction is **the** hardness reduction.

Tractability Island 1



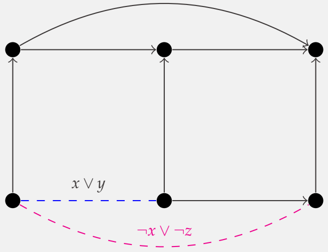
- Constraints defined as an AND of:
  - implications (can use constants 1 and 0);
  - ORs of negated variables.

  such that the implication graph is $2K_2$-free.

Then, MIN SAT is FPT when parameterized by k and max constraint arity (but only unweighted here; weighted is W[1]-hard).

Tractability Island 2

UNIVERSITY OF WARSAW



- Constraints defined as an AND of 2-clauses (can use constants 1 and 0).
- Assumption: for every constraint, the graph of 2-clauses is $2K_2$-free (after deleting 1 and 0).

Then, Min UnSAT is FPT when parameterized by k and max constraint arity (also in the weighted setting).

MinCSP
parameterized complexity
classification programme

Goal: Provide FPT vs W[1]-hard dichotomy theorems for MinCSP parameterized by the number of unsatisfied constraints for various classes of languages Γ.

MinCSP
parameterized complexity
classification programme

Goal: Provide FPT vs W[1]-hard dichotomy theorems for MinCSP parameterized by the number of unsatisfied constraints for various classes of languages Γ.

Note: requires $\text{CSP}(\Gamma)$ to be P-time.

Weighted DFAS

- Weighted Directed Feedback Arc Set
  (delete at most k arcs of minimum total weight to get a DAG)

Weighted DFAS

- Weighted Directed Feedback Arc Set
  (delete at most k arcs of minimum total weight to get a DAG)
- Weighted Directed Feedback Arc Set reduces to Weighted Skew Multicut with $\ell \sim k$ by the standard Iterative Compression trick.

Weighted DFAS

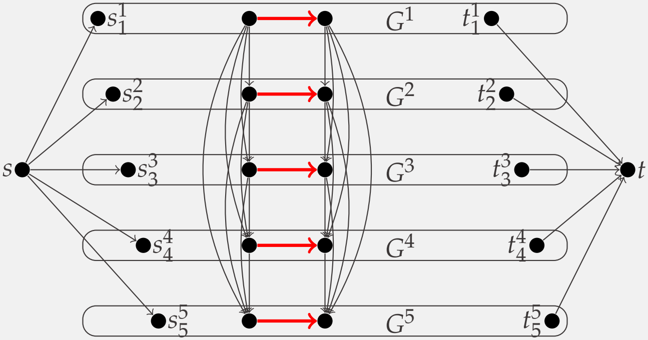
- Weighted Directed Feedback Arc Set
  (delete at most k arcs of minimum total weight to get a DAG)
- Weighted Directed Feedback Arc Set reduces to Weighted Skew Multicut with $\ell \sim k$ by the standard Iterative Compression trick.
- Weighted Skew Multicut
  ($s_1, \ldots, s_\ell, t_1, \ldots, t_\ell \in V(G)$, delete at most k arcs of minimum total weight to break all $s_i \to t_j$ paths for $1 \leq i \leq j \leq \ell$)

Weighted DFAS

- Weighted Directed Feedback Arc Set
  (delete at most k arcs of minimum total weight to get a DAG)
- Weighted Directed Feedback Arc Set reduces to Weighted Skew Multicut with $\ell \sim k$ by the standard Iterative Compression trick.
- Weighted Skew Multicut
  ($s_1, \ldots, s_\ell, t_1, \ldots, t_\ell \in V(G)$, delete at most k arcs of minimum total weight to break all $s_i \to t_j$ paths for $1 \leq i \leq j \leq \ell$)
- Weighted Skew Multicut unravels to a Bundled Cut with Pairwise Linked Deletable Edges instance.

Weighted Skew Multicut

$(s_1, \ldots, s_\ell, t_1, \ldots, t_\ell \in V(G)$, delete at most k arcs of minimum total weight to break all $s_i \to t_j$ paths for $1 \leq i \leq j \leq \ell)$



Weighted DFAS

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström)

(Weighted) Directed Feedback Arc Set *reduces to* (Weighted) BCwPLDE.

Directed Feedback Arc Set: delete at most k arcs to get an acyclic graph.

Weighted DFAS

UNIVERSITY OF WARSAW

Theorem (Kim, Kratsch, P., Wahlström)

(Weighted) Directed Feedback Arc Set *reduces to* (Weighted) BCwPLDE.

Directed Feedback Arc Set: delete at most k arcs to get an acyclic graph.

Theorem (Kim, Masařík, P., Sharma, Wahlström)

(Weighted) Directed Subset Feedback Arc Set *reduces to* (Weighted) BCwPLDE.

Directed Subset Feedback Arc Set: the input digraph G is equipped with $R \subseteq E(G)$ and the goal is to delete at most k arcs so that no cycle contains an arc of R.

Undirected Multicut

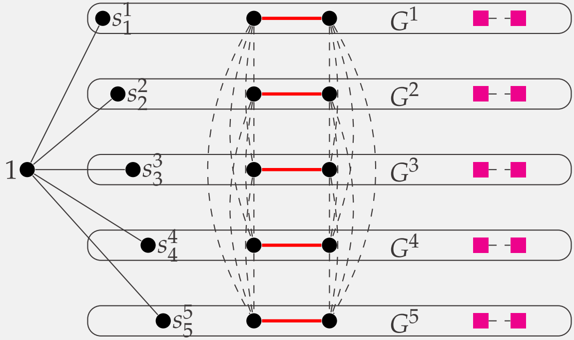
- UNDIRECTED MULTICUT: Graph G, terminal pairs $\mathcal{T} \subseteq \binom{V(G)}{2}$, $k \in \mathbb{Z}$; delete at most k arcs so that every $st \in \mathcal{T}$ is separated.

Undirected Multicut

- UNDIRECTED MULTICUT: Graph G, terminal pairs $\mathcal{T} \subseteq \binom{V(G)}{2}$, $k \in \mathbb{Z}$; delete at most k arcs so that every $st \in \mathcal{T}$ is separated.
- Iterative compression:
  - $S = \{s_1, \ldots, s_{k+1}\} \subseteq V(G)$ that separates every $st \in \mathcal{T}$;
  - solution needs also to separate S.

Undirected Multicut

- UNDIRECTED MULTICUT: Graph G, terminal pairs $\mathcal{T} \subseteq \binom{V(G)}{2}$, $k \in \mathbb{Z}$; delete at most k arcs so that every $st \in \mathcal{T}$ is separated.
- Iterative compression:
  - $S = \{s_1, \ldots, s_{k+1}\} \subseteq V(G)$ that separates every $st \in \mathcal{T}$;
  - solution needs also to separate S.



Undirected Multicut

UNDIRECTED MULTICUT: Graph G, terminal pairs $\mathcal{T} \subseteq \binom{V(G)}{2}$, $k \in \mathbb{Z}$; delete at most k arcs so that every $st \in \mathcal{T}$ is separated.

Theorem (Kim, Masařík, P., Sharma, Wahlström)

WEIGHTED UNDIRECTED MULTICUT *can be solved using the algorithm of Tractability Island 2 as a black-box.*

Beyond boolean domain

- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).

Beyond boolean domain

- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).
- $\Gamma = \{<, \leq\}$ is equivalent to DIRECTED SUBSET FEEDBACK ARC SET!

Beyond boolean domain

- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).
- $\Gamma = \{<, \leq\}$ is equivalent to Directed Subset Feedback Arc Set!
- (Osipov, P., Wahlström, ESA 2024)
  $\Gamma = \{<, \neq, =\}$ is FPT.
  $\Gamma = \{<, \leq, \neq, =\}$ is W[1]-hard.
  - Graph formulation: Symmetric Multicut. Directed graph G, unordered pairs of terminals $\mathcal{T}$, integer k. Delete k edges so that for every $st \in \mathcal{T}$, s and t are not in the same strong component.

Beyond boolean domain

- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).
- $\Gamma = \{<, \leq\}$ is equivalent to DIRECTED SUBSET FEEDBACK ARC SET!
- (Osipov, P., Wahlström, ESA 2024)
  $\Gamma = \{<, \neq, =\}$ is FPT.
  $\Gamma = \{<, \leq, \neq, =\}$ is W[1]-hard.

Beyond boolean domain

- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).
- $\Gamma = \{<, \leq\}$ is equivalent to DIRECTED SUBSET FEEDBACK ARC SET!
- (Osipov, P., Wahlström, ESA 2024)
  $\Gamma = \{<, \neq, =\}$ is FPT.
  $\Gamma = \{<, \leq, \neq, =\}$ is W[1]-hard.
- (Osipov, Wahlström, ESA 2023) Full dichotomy for FO over $\{=, \neq\}$.

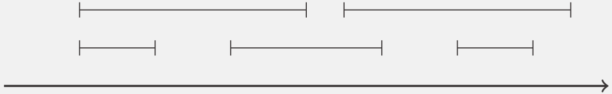
Beyond boolean domain

- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).
- $\Gamma = \{<, \leq\}$ is equivalent to DIRECTED SUBSET FEEDBACK ARC SET!
- (Osipov, P., Wahlström, ESA 2024)
  $\Gamma = \{<, \neq, =\}$ is FPT.
  $\Gamma = \{<, \leq, \neq, =\}$ is W[1]-hard.
- (Osipov, Wahlström, ESA 2023) Full dichotomy for FO over $\{=, \neq\}$.
- Most cases beyond look W[1]-hard.

Beyond boolean domain

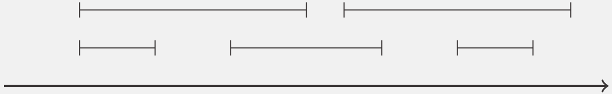
- Point Line Algebra:
  - Domain: $\mathbb{Q}$.
  - Constraints have access to $<, \leq, =, \neq$ (and are FO formulae).
- $\Gamma = \{<, \leq\}$ is equivalent to DIRECTED SUBSET FEEDBACK ARC SET!
- (Osipov, P., Wahlström, ESA 2024)
  $\Gamma = \{<, \neq, =\}$ is FPT.
  $\Gamma = \{<, \leq, \neq, =\}$ is W[1]-hard.
- (Osipov, Wahlström, ESA 2023) Full dichotomy for FO over $\{=, \neq\}$.
- Most cases beyond look W[1]-hard.
- **OPEN**: Full dichotomy.

Beyond boolean domain



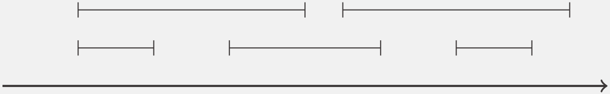
- Allen Algebra:
  - Domain: intervals $\{[a, b] \mid a, b \in \mathbb{Q}, a < b\}$.
  - Basic constraints: “precedes”, ”starts”, ”ends”, ”meets”, ”during”, ”overlap”, ”equal”.

Beyond boolean domain



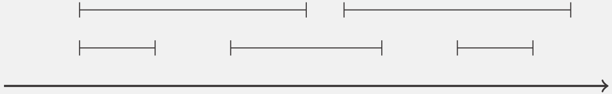
- Allen Algebra:
  - Domain: intervals $\{[a, b] \mid a, b \in \mathbb{Q}, a < b\}$.
  - Basic constraints: “precedes”, ”starts”, ”ends”, ”meets”, ”during”, ”overlap”, ”equal”.
- (Dabrowski, Jonsson, Ordyniak, Osipov, P., Sharma, IPEC 2023) classification of languages that are just subsets of basic constraints.

Beyond boolean domain



- Allen Algebra:
  - Domain: intervals $\{[a, b] \mid a, b \in \mathbb{Q}, a < b\}$.
  - Basic constraints: “precedes”, ”starts”, ”ends”, ”meets”, ”during”, ”overlap”, ”equal”.
- (Dabrowski, Jonsson, Ordyniak, Osipov, P., Sharma, IPEC 2023) classification of languages that are just subsets of basic constraints.
  - Highlight 1: there is always an FPT 2-approximation by using Directed Subset Feedback Arc Set separately on endpoints.

Beyond boolean domain



- Allen Algebra:
    - Domain: intervals $\{[a, b] \mid a, b \in \mathbb{Q}, a < b\}$.
    - Basic constraints: “precedes”, ”starts”, ”ends”, ”meets”, ”during”, ”overlap”, ”equal”.
- (Dabrowski, Jonsson, Ordyniak, Osipov, P., Sharma, IPEC 2023) classification of languages that are just subsets of basic constraints.
    - Highlight 1: there is always an FPT 2-approximation by using Directed Subset Feedback Arc Set separately on endpoints.
    - Highlight 2: {“precedes”, “starts”, “equals”} has an FPT algorithm via reduction to BCwPLDE.

Linear equations

- Domain: field / ring $\mathbb{K}$.

Linear equations

- Domain: field / ring $\mathbb{K}$.
- Constraints: linear equations.

Linear equations

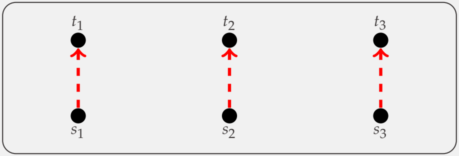
- Domain: field / ring $\mathbb{K}$.
- Constraints: linear equations.
- Example: EDGE BIPARTIZATION
  - vertices $\sim$ variables over $\mathbb{F}_2$;
  - edge $uv \sim u = 1 + v$.

Linear equations

UNIVERSITY OF WARSAW

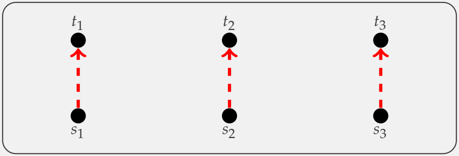
- Domain: field / ring $\mathbb{K}$.
- Constraints: linear equations.
- Example: EDGE BIPARTIZATION
  - vertices $\sim$ variables over $\mathbb{F}_2$;
  - edge $uv \sim u = 1 + v$.
- (Dabrowski, Jonsson, Ordyniak, Osipov, Wahlström, SODA 2023)
  - FPT if two variables per equation and $\mathbb{K}$ is an Euclidean domain;
  - W[1]-hard for three variables per equation;
  - W[1]-hard for some commutative rings (e.g., $\mathbb{Z}/6\mathbb{Z}$).

3-terminal Directed Multicut



MULTICUT: graph G, terminal pairs $(s_i, t_i)_{i=1}^{\ell}$, integer k.
Goal: delete k edges so that no $s_i \to t_i$ path remains.

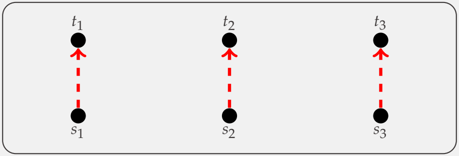
3-terminal Directed Multicut



MULTICUT: graph G, terminal pairs $(s_i, t_i)_{i=1}^{\ell}$, integer k.
Goal: delete k edges so that no $s_i \to t_i$ path remains.

- [Marx, Razgon, STOC'11]: ℓ arbitrary is W[1]-hard.
- [Chitnis, Hajiaghayi, Marx, SODA'12]: $\ell = 2$ is FPT.
- [P. Wahlström, SODA'16]: $\ell = 4$ is W[1]-hard.

3-terminal Directed Multicut



MULTICUT: graph G, terminal pairs $(s_i, t_i)_{i=1}^{\ell}$, integer k.
Goal: delete k edges so that no $s_i \to t_i$ path remains.

- [Marx, Razgon, STOC'11]: ℓ arbitrary is W[1]-hard.
- [Chitnis, Hajiaghayi, Marx, SODA'12]: $\ell = 2$ is FPT.
- [P. Wahlström, SODA'16]: $\ell = 4$ is W[1]-hard.

Theorem (Hatzel, Jaffke, Lima, Masařík, P., Sharma, Sorge, SODA'23)

$\ell = 3$ case is FPT! (Uses twin-width and flow augmentation.)

Summary

Flow augmentation looks like the missing tool in directed graph separation problems.

Summary

Flow augmentation looks like the missing tool in directed graph separation problems.

Closed some dichotomies and long-standing open problems.

Summary

UNIVERSITY OF WARSAW

Flow augmentation looks like the missing tool in directed graph separation problems.

Closed some dichotomies and long-standing open problems.

MinCSP parameterized complexity classification programme.

Summary

Flow augmentation looks like the missing tool in directed graph separation problems.

Closed some dichotomies and long-standing open problems.

MINCSP parameterized complexity classification programme.

Open problems:

- Full dichotomy for Point Line Algebra?
- Good methodology for closing dichotomies?

Summary

Flow augmentation looks like the missing tool in directed graph separation problems.

Closed some dichotomies and long-standing open problems.

MinCSP parameterized complexity classification programme.

Open problems:

- Full dichotomy for Point Line Algebra?
- Good methodology for closing dichotomies?

Thanks!