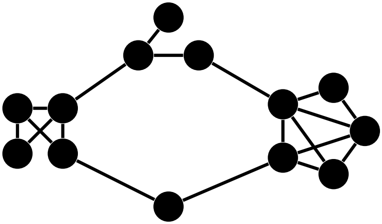
Subexponential parameterized complexity of completion problems
Survey of the upper bounds

Marcin Pilipczuk

University of Warsaw

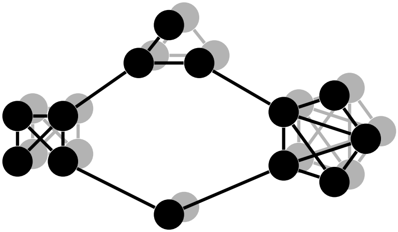
4th Nov 2015

Graph modification problems



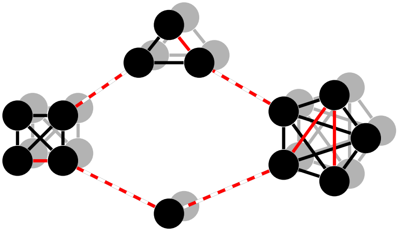
- An input graph: a result of an experiment.

Graph modification problems



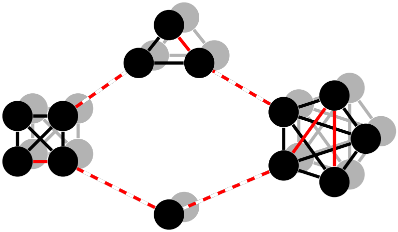
- An input graph: a result of an experiment.
- Theory: result should have some property.

Graph modification problems



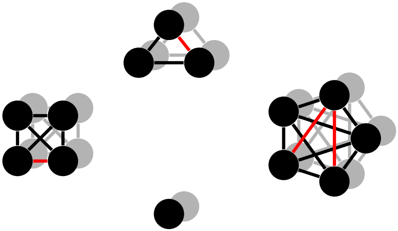
- An input graph: a result of an experiment.
- Theory: result should have some property.
- Modification: error in our measurments.

Graph modification problems



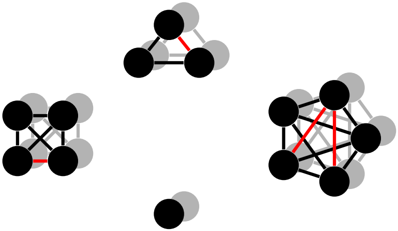
- An input graph: a result of an experiment.
- Theory: result should have some property.
- Modification: error in our measurments.
- A few errors ⇒ #modifications as a parameter.

Completions



completion = only edge insertions are allowed

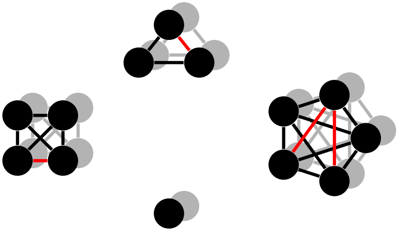
Completions



completion = only edge insertions are allowed

$\mathcal{G}$ Completion
obtain a graph $\in \mathcal{G}$

Completions

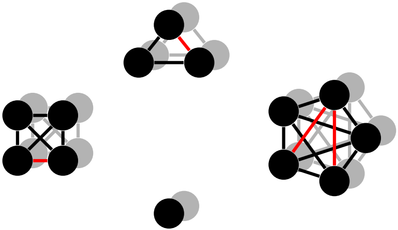


completion = only edge insertions are allowed

$\mathcal{G}$ COMPLETION
obtain a graph $\in \mathcal{G}$

$\mathcal{F}$-COMPLETION
kill all induced subgraphs $\in \mathcal{F}$

Completions



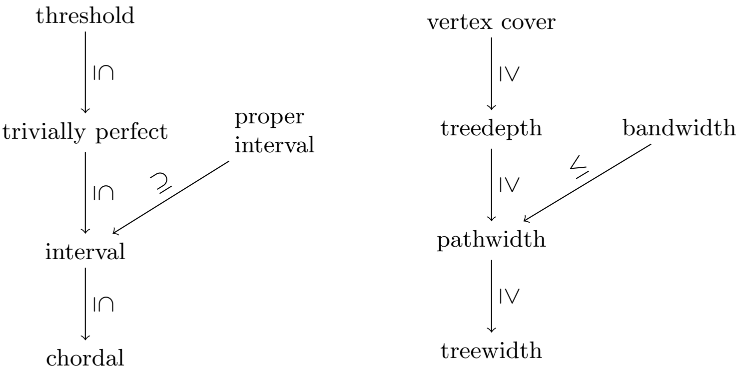
completion = only edge insertions are allowed

$\mathcal{G}$ Completion
obtain a graph $\in \mathcal{G}$

$\mathcal{F}$-Completion
kill all induced subgraphs $\in \mathcal{F}$

Side note: no known P vs NP dichotomy for completion problems.

Interesting classes for completion



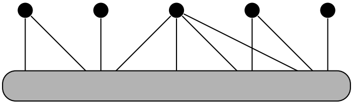
$$\mathrm{measure}(G) \simeq \min\{\omega(H) : H \in \mathcal{G} \text{ and } H \text{ is a completion of } G\}.$$

Disclaimer

In this talk we mostly focus on $f(k)$
in the FPT running time $f(k)n^{\mathcal{O}(1)}$.

We denote $\mathcal{O}^*(f(k)) = f(k)n^{\mathcal{O}(1)}$.

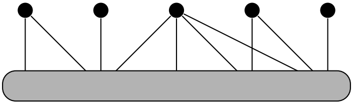
Finite set $\mathcal{F}$



Split Completion

$\{2K_2, C_4, C_5\}$-Completion

Finite set $\mathcal{F}$



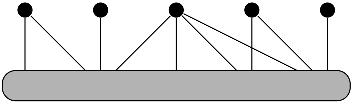
SPLIT COMPLETION

$\{2K_2, C_4, C_5\}$-COMPLETION

Theorem (Cai, IPL'96)

A simple branching strategy $\Rightarrow \mathcal{O}^*(c^k)$ *FPT algorithm*

Finite set $\mathcal{F}$



SPLIT COMPLETION $\{2K_2, C_4, C_5\}$-COMPLETION

Theorem (Cai, IPL'96)

A simple branching strategy $\Rightarrow \mathcal{O}^*(c^k)$ FPT algorithm

Works also for:
CO-CLUSTER, COGRAPH, THRESHOLD, PSEUDOSPLIT, ...

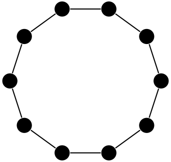
Chordal completion

CHORDAL COMPLETION

$\{C_4, C_5, C_6, \ldots\}$-COMPLETION

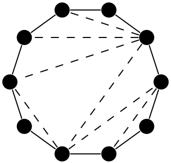
Chordal completion

Chordal Completion $\{C_4, C_5, C_6, \ldots\}$-Completion



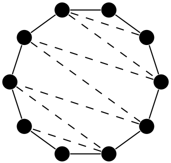
Chordal completion

CHORDAL COMPLETION $\{C_4, C_5, C_6, \ldots\}$-COMPLETION



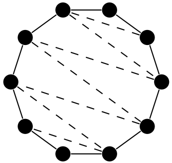
Chordal completion

Chordal Completion $\{C_4, C_5, C_6, \ldots\}$-Completion



Chordal completion

CHORDAL COMPLETION $\{C_4, C_5, C_6, \ldots\}$-COMPLETION

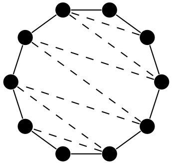


Theorem (Kaplan, Shamir, Tarjan, SICOMP'99)

Large hole
$\Rightarrow$ *many options but big cost (Catalan number $C_{\ell-2}$ for $\ell - 3$ edges)*
$\Rightarrow$ $\mathcal{O}^*(4^k)$ *branching.*

Chordal completion

Chordal Completion $\{C_4, C_5, C_6, \ldots\}$-Completion



Theorem (Kaplan, Shamir, Tarjan, SICOMP'99)

Large hole
$\Rightarrow$ many options but big cost (Catalan number $C_{\ell-2}$ for $\ell - 3$ edges)
$\Rightarrow$ $\mathcal{O}^*(4^k)$ branching.

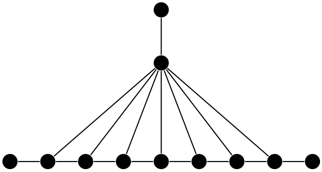
Gives also $\mathcal{O}^*(c^k)$ FPT algorithm for Proper Interval Completion, as Chordal $\longrightarrow$ Proper Interval means killing $\{S_3, \mathrm{claw}, \mathrm{net}\}$.

Interval completion

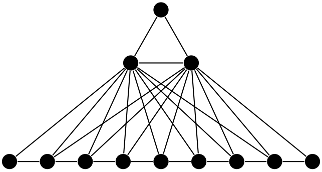
INTERVAL COMPLETION $\{\text{holes}, \text{ATs}\}$-COMPLETION

Interval completion

Interval Completion

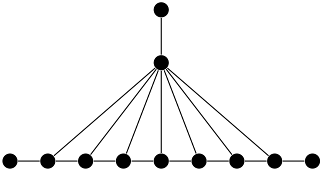


{holes, ATs}-Completion

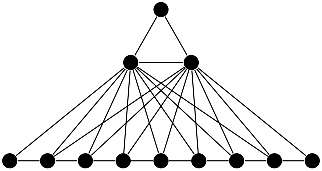


Interval completion

INTERVAL COMPLETION



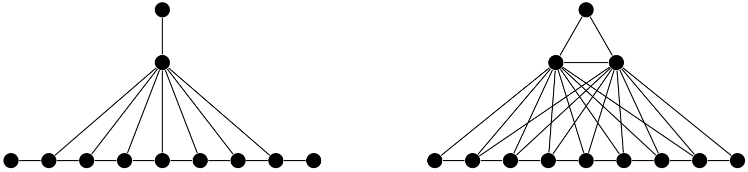
{holes, ATs}-COMPLETION



Long-standing open problem for more than a decade.

Interval completion

Interval Completion	{holes, ATs}-Completion



Long-standing open problem for more than a decade.

Theorem (Villanger, Heggernes, Paul, Telle, SICOMP'09)

Branching still doable! An $\mathcal{O}^*(k^{2k})$ FPT algorithm.

Theorem (Cao, SODA'16)

Can be solved in $\mathcal{O}(c^k(n+m))$ time.

Polynomial kernels

Theorem (Kaplan, Shamir, Tarjan, SICOMP'99)

CHORDAL COMPLETION *admits a polynomial kernel.*

Polynomial kernels

Theorem (Kaplan, Shamir, Tarjan, SICOMP'99)

CHORDAL COMPLETION *admits a polynomial kernel.*

Theorem (Guo, ISAAC'07)

CHAIN, SPLIT, THRESHOLD *and* TRIVIALLY PERFECT COMPLETION *admit polynomial kernels.*

Polynomial kernels

Theorem (Kaplan, Shamir, Tarjan, SICOMP'99)

CHORDAL COMPLETION *admits a polynomial kernel.*

Theorem (Guo, ISAAC'07)

CHAIN, SPLIT, THRESHOLD *and* TRIVIALLY PERFECT COMPLETION *admit polynomial kernels.*

Theorem (Kratsch, Wahlström, IWPEC'09)

There is a finite $\mathcal{F}$ *such that* $\mathcal{F}$-COMPLETION *does not admit a polynomial kernel unless* $\mathrm{NP} \subseteq \mathrm{coNP/poly}$.

Polynomial kernels

Theorem (Kaplan, Shamir, Tarjan, SICOMP'99)

CHORDAL COMPLETION *admits a polynomial kernel.*

Theorem (Guo, ISAAC'07)

CHAIN, SPLIT, THRESHOLD *and* TRIVIALLY PERFECT COMPLETION *admit polynomial kernels.*

Theorem (Kratsch, Wahlström, IWPEC'09)

*There is a finite $\mathcal{F}$ such that $\mathcal{F}$-*COMPLETION *does not admit a polynomial kernel unless* $\mathrm{NP} \subseteq \mathrm{coNP/poly}$*.*

Theorem (Guillemot, Havet, Paul, Perez, Algorithmica'13)

COGRAPH COMPLETION *admits a polynomial kernel.*

Theorem (Bessy, Perez, Inf. Comp.'13)

PROPER INTERVAL COMPLETION *admits a polynomial kernel.*

Look at the bright side!

$\mathcal{G}$ COMPLETION

$\mathcal{F}$-COMPLETION

Look at the bright side!

$\mathcal{G}$ Completion $\mathcal{F}$-Completion

Fomin, Villanger, SODA'12:
think positively!

Look at the bright side!

$\mathcal{G}$ Completion $\mathcal{F}$-Completion

Fomin, Villanger, SODA'12:
think positively!

Theorem (Fomin, Villanger, SODA'12)

Chordal Completion *can be solved in time* $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$.

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Strategy:

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Strategy:

1. Apply known polynomial kernel.

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Strategy:

1. Apply known polynomial kernel.
2. Identify and enumerate $2^{o(k)}$ candidates for crucial structures in G.

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Strategy:

1. Apply known polynomial kernel.
2. Identify and enumerate $2^{o(k)}$ candidates for crucial structures in G.
   - crucial structure $\simeq$ a maximal clique;

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Strategy:

1. Apply known polynomial kernel.
2. Identify and enumerate $2^{o(k)}$ candidates for crucial structures in G.
   - crucial structure $\simeq$ a maximal clique;
   - often $n^{\mathcal{O}(\sqrt{k})}$ candidate structures; with poly kernel $\Rightarrow 2^{o(k)}$ bound

Look at the bright side!

Fomin, Villanger, SODA'12:
think positively!

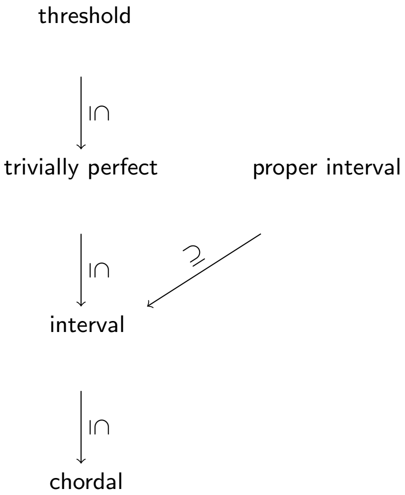
Build the structure of $(G + \mathrm{completion})$ by dynamic programming.

Key observation: there are $2^{o(k)}$ reasonable 'partial structures', and hence $2^{o(k)}$ reasonable DP states.

Strategy:

1. Apply known polynomial kernel.
2. Identify and enumerate $2^{o(k)}$ candidates for crucial structures in G.
   - crucial structure $\simeq$ a maximal clique;
   - often $n^{\mathcal{O}(\sqrt{k})}$ candidate structures; with poly kernel $\Rightarrow 2^{o(k)}$ bound
3. Candidate structure $\longrightarrow$ DP state.

Look at the bright side!



Look at the bright side!

threshold
$\mathcal{O}^*\left(k^{\mathcal{O}(\sqrt{k})}\right)$

$\downarrow \subseteq$

trivially perfect
$\mathcal{O}^*\left(k^{\mathcal{O}(\sqrt{k})}\right)$

proper interval
$\mathcal{O}^*\left(k^{\mathcal{O}(k^{2/3})}\right)$

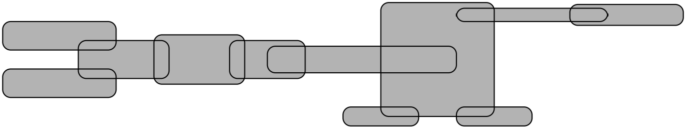
$\downarrow \subseteq$

interval
$\mathcal{O}^*\left(k^{\mathcal{O}(\sqrt{k})}\right)$

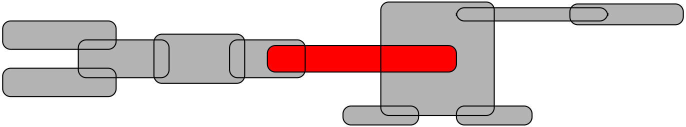
$\downarrow \subseteq$

chordal
$\mathcal{O}^*\left(k^{\mathcal{O}(\sqrt{k})}\right)$

Chordal completion

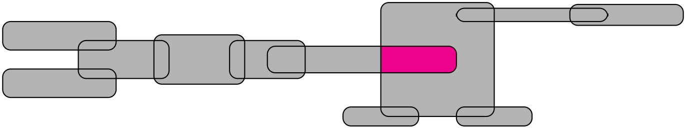


Chordal completion



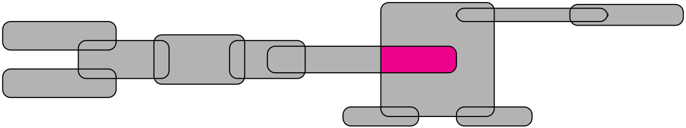
crucial structure = maximal clique

Chordal completion



crucial structure = maximal clique / minimal clique separator

Chordal completion

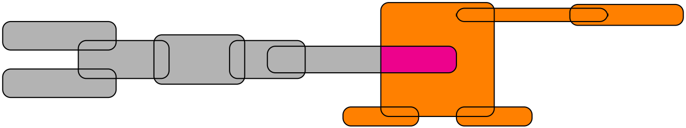


crucial structure = maximal clique / minimal clique separator

Theorem (Fomin, Villanger, SODA'12)

One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

Chordal completion



crucial structure = maximal clique / minimal clique separator

Theorem (Fomin, Villanger, SODA'12)

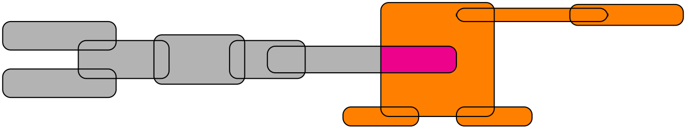
One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

DP state = clique separator Ω +

one connected component C of $G \setminus \Omega$,

value = minimum completion of $G[C \cup \Omega]$ that cliquifies Ω.

Chordal completion



crucial structure = maximal clique / minimal clique separator

Theorem (Fomin, Villanger, SODA'12)

One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

DP state = clique separator Ω +

one connected component C of $G \setminus \Omega$,

value = minimum completion of $G[C \cup \Omega]$ that cliquifies Ω.

Corollary: Chain Completion in $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ time.

Enumerating potential maximal cliques

Theorem (Fomin, Villanger, SODA'12)

One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

Enumerating potential maximal cliques

Theorem (Fomin, Villanger, SODA'12)

One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

- Multiple-step, involved combinatorial analysis.

Enumerating potential maximal cliques

Theorem (Fomin, Villanger, SODA'12)

One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

- Multiple-step, involved combinatorial analysis.
- Will give a flavour on the INTERVAL COMPLETION case.

Enumerating potential maximal cliques

Theorem (Fomin, Villanger, SODA'12)

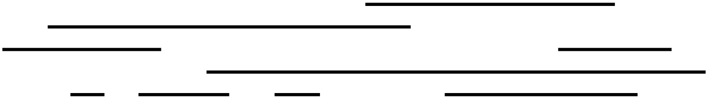
One can either enumerate $k^{\mathcal{O}(\sqrt{k})}$ candidate maximal cliques and candidate clique separators, or perform a $k^{\mathcal{O}(\sqrt{k})}$ branching.

- Multiple-step, involved combinatorial analysis.
- Will give a flavour on the Interval Completion case.

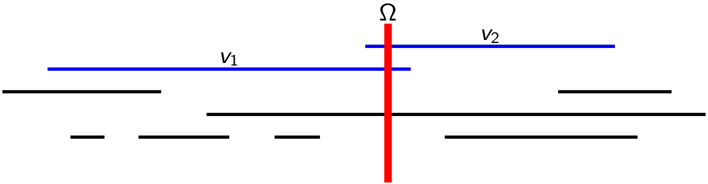
Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

There are $n^{\mathcal{O}(\sqrt{k})}$ reasonable candidates for maximal cliques in the Interval Completion *case.*

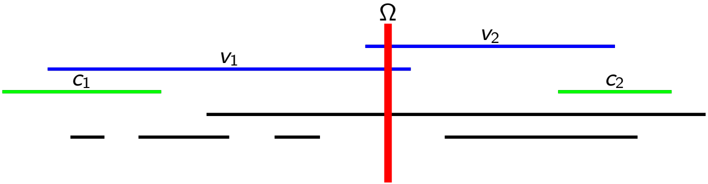
Maximal cliques in Interval Completion



Maximal cliques in Interval Completion

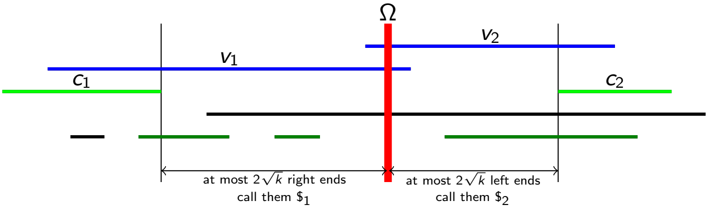


Maximal cliques in Interval Completion



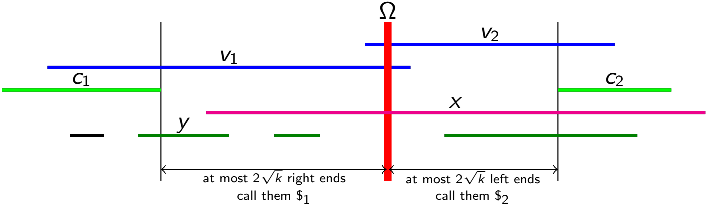
cheap v = at most $\sqrt{k}$ incident solution edges.
c_1 := last ending cheap before Ω
c_2 := first starting cheap after Ω

Maximal cliques in Interval Completion



cheap v = at most $\sqrt{k}$ incident solution edges.
c_1 := last ending cheap before Ω
c_2 := first starting cheap after Ω

Maximal cliques in Interval Completion



cheap v = at most $\sqrt{k}$ incident solution edges.

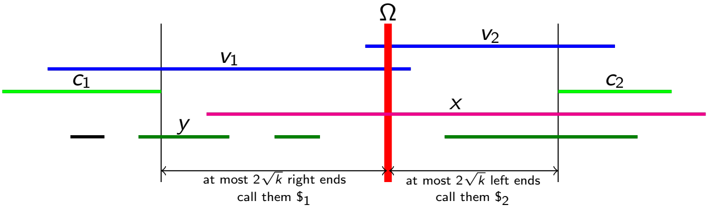
$c_1 :=$ last ending cheap before Ω

$c_2 :=$ first starting cheap after Ω

Why $x \in \Omega$?

x has left end before Ω because some $y \in N_G(x)$ has right end before Ω.

Maximal cliques in Interval Completion



cheap v = at most $\sqrt{k}$ incident solution edges.

$c_1 :=$ last ending cheap before Ω

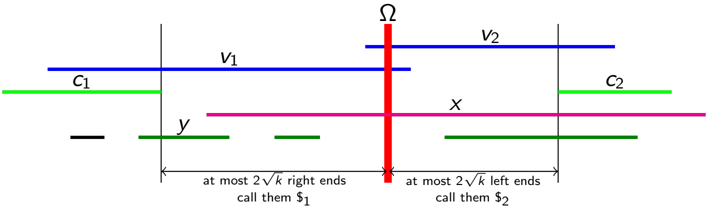
$c_2 :=$ first starting cheap after Ω

Why $x \in \Omega$?

x has left end before Ω because some $y \in N_G(x)$ has right end before Ω.

$y \in N_G(x)$ for some y with right end before $\Omega \Rightarrow$
$x \in N_G(v_1) \cup N_G(\$_1) \cup N_{G+F}(c_1)$.

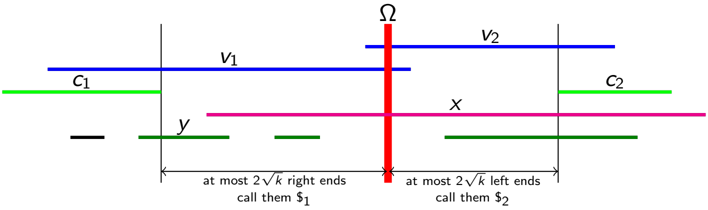
Maximal cliques in Interval Completion



Lemma

$$\Omega = (N_G[v_1] \cup N_G(\$_1) \cup N_{G+F}(c_1)) \cap (N_G[v_2] \cup N_G(\$_2) \cup N_{G+F}(c_2)).$$

Maximal cliques in Interval Completion



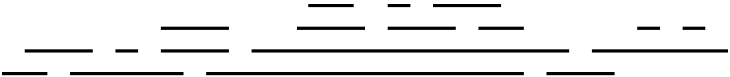
Lemma

$$\Omega = (N_G[v_1] \cup N_G(\$_1) \cup N_{G+F}(c_1)) \cap (N_G[v_2] \cup N_G(\$_2) \cup N_{G+F}(c_2)).$$

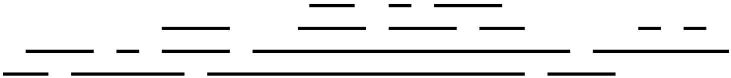
There are $n^{\mathcal{O}(\sqrt{k})}$ choices for:

- v_1, v_2, c_1, c_2;
- $\$_1$ and $\$_1$, as there are of size $\mathcal{O}(\sqrt{k})$;
- solution edges incident to c_1 and c_2, as both c_1 and c_2 are cheap.

Interval Completion

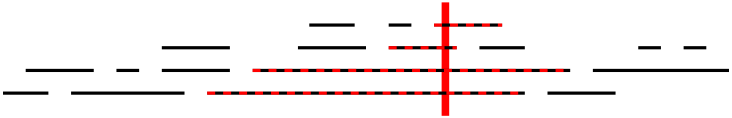


Interval Completion



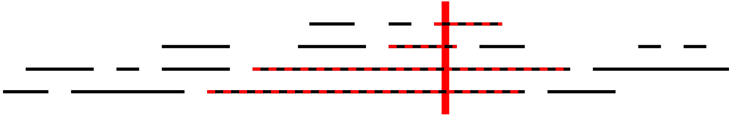
Problem 1: no known polynomial kernel!

Interval Completion



Problem 1: no known polynomial kernel!

Interval Completion

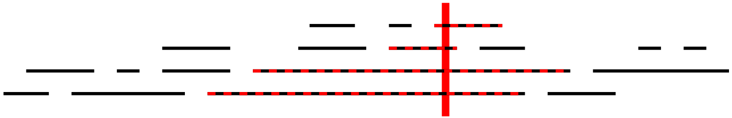


Problem 1: no known polynomial kernel!

Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

There are $n^{\mathcal{O}(\sqrt{k})}$ reasonable candidates for maximal cliques.

Interval Completion



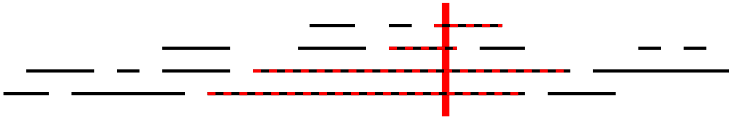
Problem 1: no known polynomial kernel!

Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

There are $n^{\mathcal{O}(\sqrt{k})}$ reasonable candidates for maximal cliques.

Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

For any vertex v, there are $k^{\mathcal{O}(t+\sqrt{k})} n^{\mathcal{O}(1)}$ reasonable ways to choose completion edges incident to v, as long as there are at most t of them*.



Problem 1: no known polynomial kernel!

Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

There are $n^{\mathcal{O}(\sqrt{k})}$ reasonable candidates for maximal cliques.

Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

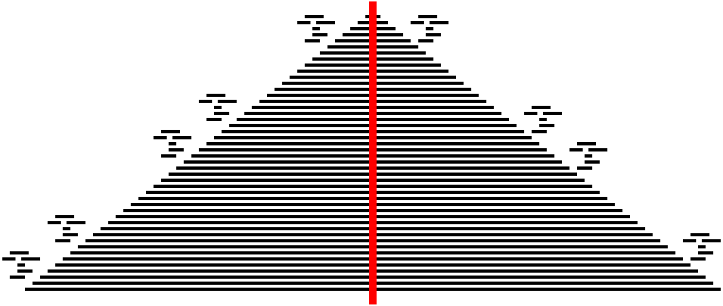
For any vertex v, there are $k^{\mathcal{O}(t+\sqrt{k})}n^{\mathcal{O}(1)}$ reasonable ways to choose completion edges incident to v, as long as there are at most t of them*.

Theorem (Bliznets, Fomin, P., Pilipczuk, 2014)

There are $k^{\mathcal{O}(\sqrt{k})}n^8$ reasonable candidates for maximal cliques*.

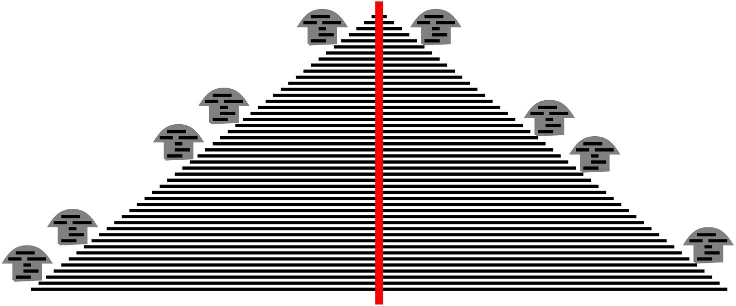
* or we can reduce something.

Interval Completion



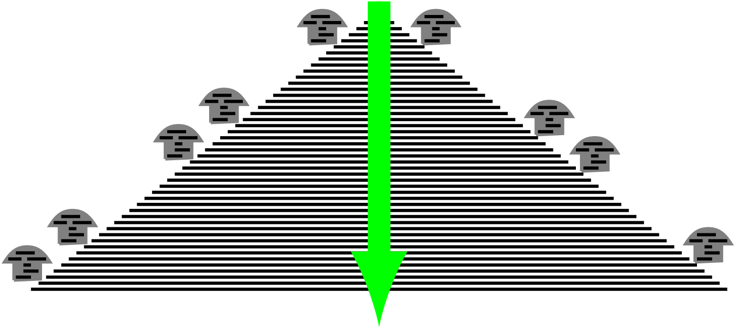
Problem 2: history is hard to deduce!

Interval Completion



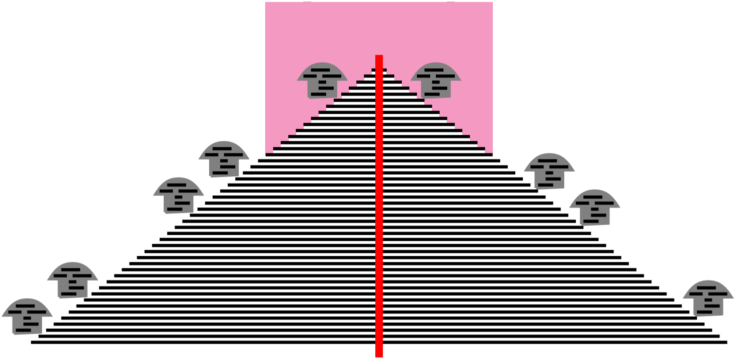
Problem 2: history is hard to deduce!

Interval Completion



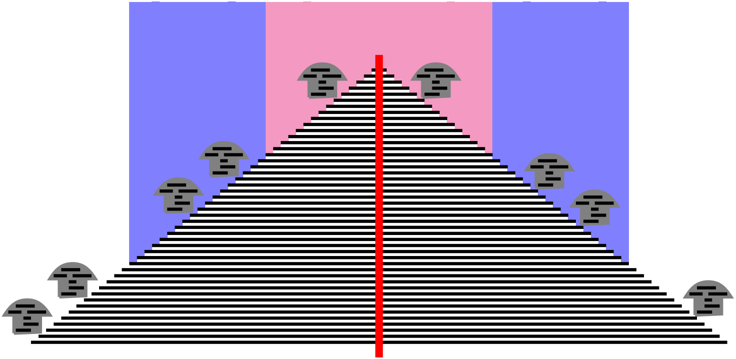
Problem 2: history is hard to deduce!

Interval Completion



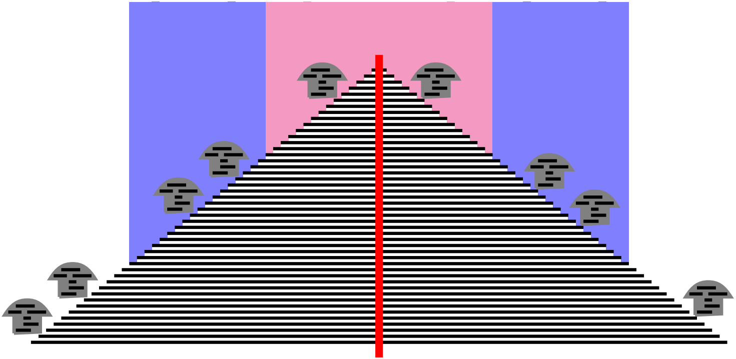
Problem 2: history is hard to deduce!

Interval Completion



Problem 2: history is hard to deduce!

Interval Completion



Problem 2: history is hard to deduce!

Solution: make much more complicated DP states.

More completions

Theorem (Ghosh, Kolay, Kumar, Misra, Panolan, Rai, Ramanujan, SWAT'12)

An $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ *algorithm for* Split Completion *via chromatic coding.*

More completions

Theorem (Ghosh, Kolay, Kumar, Misra, Panolan, Rai, Ramanujan, SWAT'12)

An $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ *algorithm for* SPLIT COMPLETION *via chromatic coding.*

Theorem (Drange, Fomin, Pilipczuk, Villanger, STACS'14)

$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ *algorithms for:*

1. TRIVIALLY PERFECT COMPLETION,
2. THRESHOLD COMPLETION
   via chromatic coding + reduction to CHAIN COMPLETION,
3. PSEUDOSPLIT COMPLETION
   similarly as SPLIT COMPLETION.

Summary

CHORDAL	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[FV, SODA'12]
CHAIN	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[FV, SODA'12]
SPLIT	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[GKKMPRR, SWAT'12]
TRIVIALLY PERFECT	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[DMPV, STACS'14]
THRESHOLD	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[DMPV, STACS'14]
PSEUDOSPLIT	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[DMPV, STACS'14]
INTERVAL	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[BFPP, SODA'16]
PROPER INTERVAL	$\mathcal{O}^*(k^{\mathcal{O}(k^{2/3})})$	[BFPP, ESA'14]

Summary

CHORDAL	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[FV, SODA'12]
CHAIN	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[FV, SODA'12]
SPLIT	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[GKKMPRR, SWAT'12]
TRIVIALLY PERFECT	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[DMPV, STACS'14]
THRESHOLD	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[DMPV, STACS'14]
PSEUDOSPLIT	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[DMPV, STACS'14]
INTERVAL	$\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$	[BFPP, SODA'16]
PROPER INTERVAL	$\mathcal{O}^*(k^{\mathcal{O}(k^{2/3})})$	[BFPP, ESA'14]
CO-CLUSTER	ETH-hard	[KU, DAM'12]
COGRAPH	ETH-hard	[DMPV, STACS'14]
CO-TRIVIALLY PERFECT	ETH-hard	[DMPV, STACS'14]

Open problems

- Can you get time $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ for Proper Interval Completion?

Open problems

- Can you get time $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ for Proper Interval Completion?
- Does Interval Completion admit a polynomial kernel?

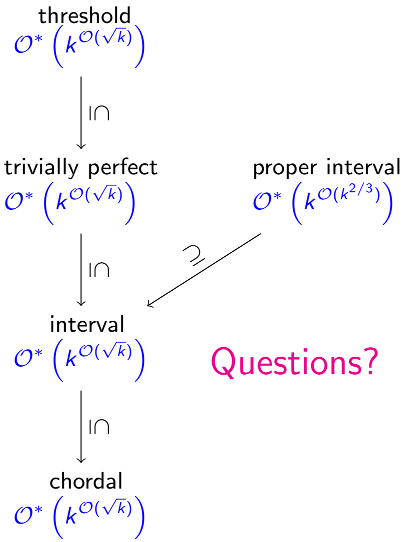
Open problems

- Can you get time $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ for Proper Interval Completion?
- Does Interval Completion admit a polynomial kernel?
- Can you provide $2^{\Omega(\sqrt{k})}$ lower bounds under ETH?
  - An $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k})})$-time algorithm for one of the problems?
  - An $\mathcal{O}^*(2^{o(\sqrt{k})})$-time algorithm seems hard, as it gives $2^{o(n)}$ bound.

Open problems

- Can you get time $\mathcal{O}^*(k^{\mathcal{O}(\sqrt{k})})$ for Proper Interval Completion?
- Does Interval Completion admit a polynomial kernel?
- Can you provide $2^{\Omega(\sqrt{k})}$ lower bounds under ETH?
  - An $\mathcal{O}^*(2^{\mathcal{O}(\sqrt{k})})$-time algorithm for one of the problems?
  - An $\mathcal{O}^*(2^{o(\sqrt{k})})$-time algorithm seems hard, as it gives $2^{o(n)}$ bound.
- Is there any meta-explanation why there are subexponential algorithms for this family of problems?

Summary diagram



Questions?