Extending Partial Representations of Proper and Unit Interval Graphs

Pavel Klavík

Joint work with: Jan Kratochvíl, Yota Otachi, Ignaz Rutter, Toshiki Saitoh, Maria Saumell, and Tomáš Vyskočil.

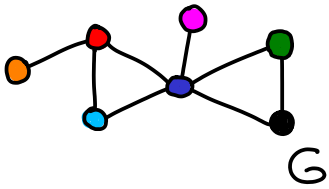
SWAT 2014 — Copenhagen

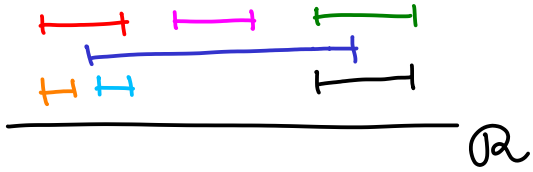
Intersection representations $\mathcal{R}$:

→ assign sets R_u $\forall u \in V(G)$,

→ encode edges by intersections

$$uv \in E(G) \iff R_u \cap R_v \neq \emptyset.$$





Interval graphs (INT): Each R_u is an interval.

→ Applied motivations (biology, etc.),

→ nice theoretical properties.

Big Picture

→ Many different graph classes.

→ Studied two types of questions:

- Recognition, equivalent characterizations
- Classical graph properties:

  max. clique, coloring, hamiltonicity.

→ Surprisingly, not much geometry!

We propose: → Study of structural properties of geometric representations.

→ Restricted representation problems:

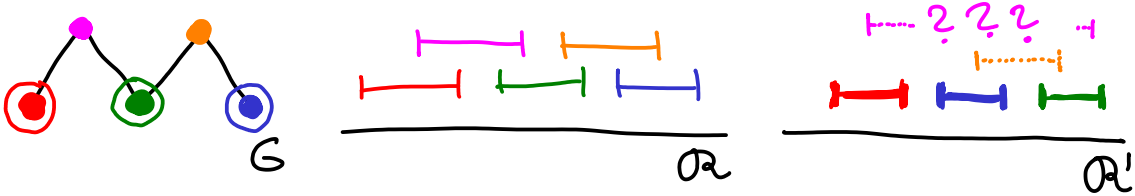
Addional constraints posed on $\mathcal{R}$.

Partial Representation Extension (RepExt)

→ Generalization of recognition.

Input: G and $\mathcal{R}'$. ← a partial representation

Question: Can $\mathcal{R}'$ be extended to $\mathcal{R}$.

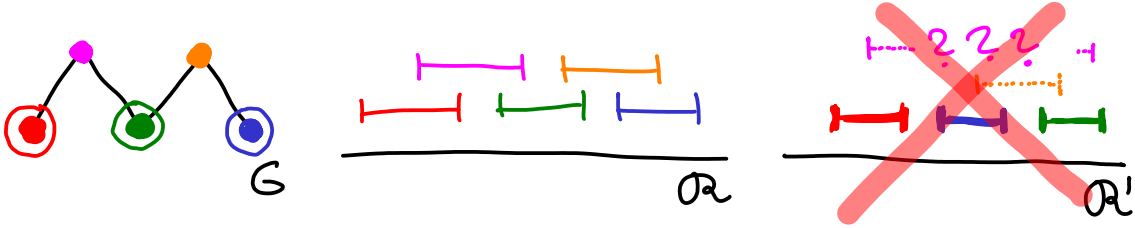


Partial Representation Extension (RepExt)

→ Generalization of recognition.

Input: G and $\mathcal{R}'$. ← a partial representation

Question: Can $\mathcal{R}'$ be extended to $\mathcal{R}$.



Introduced: K., Kratochvíl, Vyskočil (TAMC 2011)

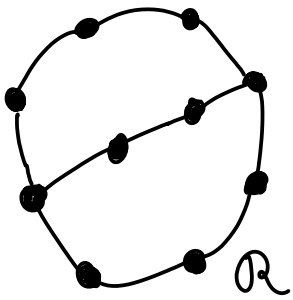
→ RepExt(INT) ..... $O(n^2)$

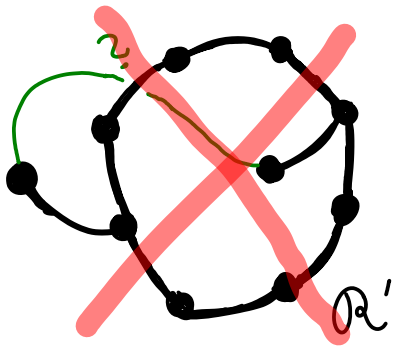
→ RepExt(PROPER INT) ..... $O(nm)$

Previous Results

PLANAR:

- Angelini, Batista, Frati, Jelínek, Kratochvíl, Patrignani, Rutter (SODA 2010). linear
- Straight-line: Partrignani (GD 2006). NP-c
- Kuratowski-type minors: Jelínek, Kratochvíl, Rutter (SoCG 2011).





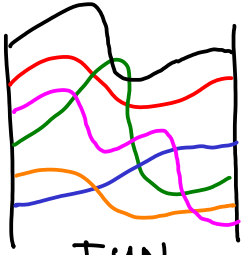
Previous Results

INT:

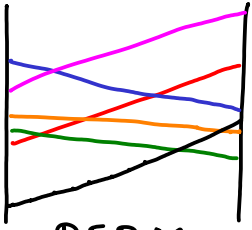
- Bläsius, Rutter (SODA 2013) — linear
- K., Kratochvíl, Otachi, Saitoh, Vyskočil (2014) — linear
- Minimal obstructions: K., Saumell (2014)

FUN, PERM: K., Kratochvíl, Krawczyk, Walczak (ESA 2012) — poly

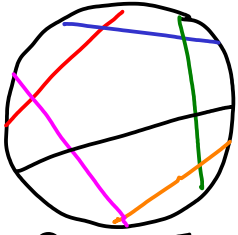
CIRCLE: Chaplick, Fulek, K. (GD 2013) — poly



FUN



PERM



CIRCLE

Previous Results

CHOR:

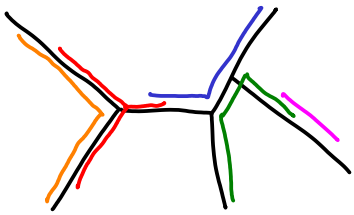
K., Kratochvíl, Otachi, Saitoh (ISAAC 2012) NP-c

CONTACT PLANAR:

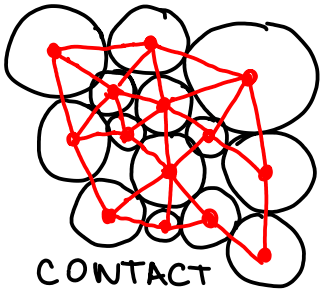
Chaplick, Dorbec, Kratochvíl, Montassier, Stacho (WG 2014) NP-c

CIRCULAR-ARC:

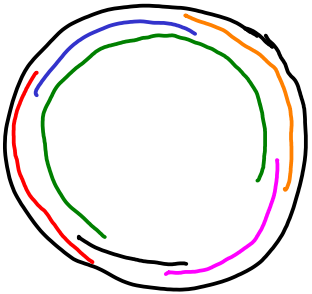
Open and very interesting.



CHOR



CONTACT PLANAR

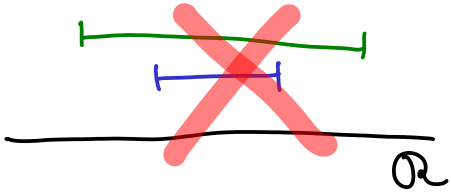


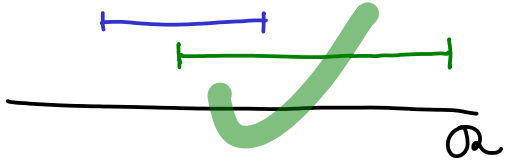
CIRCULAR-ARC

We study two subclasses of INT:

Proper interval graphs (PROPER INT)

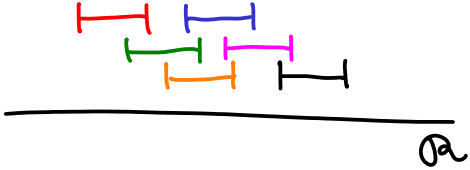
No proper inclusions $R_u \subsetneq R_v$.





Unit interval graphs (UNIT INT)

All interval of length one.



Our Results: RepExt(PROPER INT) .... $O(n+m)$.

RepExt(UNIT INT) .... $O(n^2 + n \cdot D(r))$.

Robert's Theorem (1969)

PROPER INT = UNIT INT.

Mathematically, this has the following meaning:

PROPER INT = UNIT INT

$\uparrow f : \mathcal{R} \mapsto G$

PROPER $\mathcal{R}$ $\supsetneq$ UNIT $\mathcal{R}$

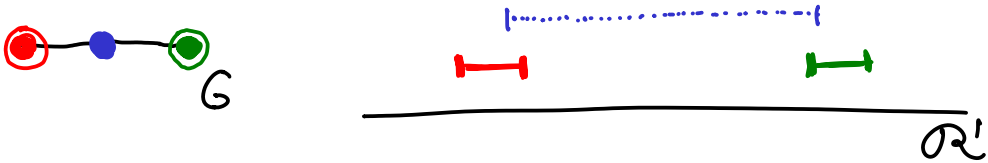
Proper interval representations TOPOLOGY

„Only the left-to-right ordering matters."

Unit interval representations GEOMETRY

„Precise geometric positions matter."

RepExt distinguishes these graph classes

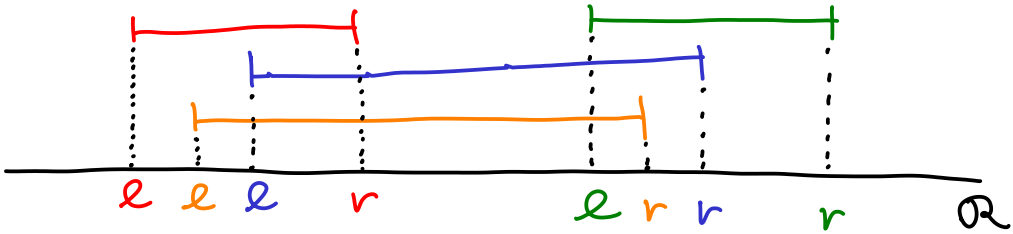


Extendible as PROPER INT, but not as UNIT INT.

Outline:

1. RepExt(PROPER INT).
2. RepExt(UNIT INT) using LP.
3. Alternative algorithm for RepExt(UNIT INT).

① RepExt(PROPER INT).



Left and right endpoints appear in the same ordering from left to right:

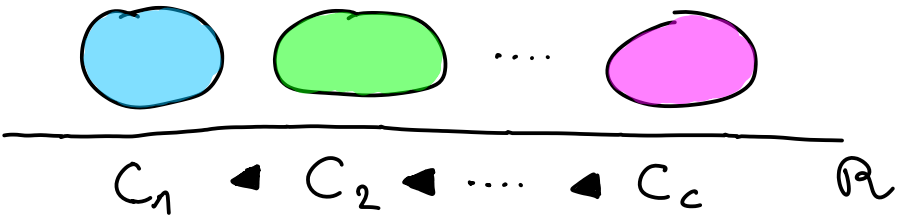
$$\bullet < \bullet < \bullet < \bullet .$$

Deng et al.: For a connected PROPER INT, the ordering $<$ is essentially unique, up to a reversal.

① RepExt(PROPER INT).

So it is sufficient to check whether $<^{\mathcal{R}'}$ is compatible with $<$ or its reversal.

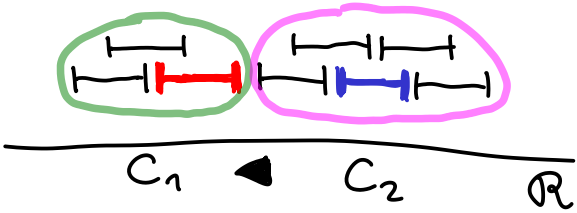
Concerning connected components, they are ordered from left to right ($\blacktriangleleft$).



The partial representation has to place intervals of each component consecutively.

② RepExt(UNIT INT).

Components are no longer independent!



Reversing the ordering $<$ of C_1 would block C_2.

We again know $\blacktriangleleft$.

(i) We process the components $C_1 \blacktriangleleft \cdots \blacktriangleleft C_c$ from left to right.

(ii) We push each C_t as far to the left as possible.

We solve a sequence of linear programs.

For C_t with $v_1 < v_2 < \cdots < v_k$:

ℓ_i's are variables

<u>Minimize:</u> $E_t := \ell_k + 1,$

<u>subject to:</u> $\ell_i \le \ell_{i+1},$

$\ell_i = \ell_i'$ if v_i is predrawn,

$$\forall\, v_i < v_j : \begin{cases} \ell_i \ge \ell_j - 1 & v_i v_j \in E(G), \\ \ell_i \le \ell_j - 1 - \varepsilon & v_i v_j \notin E(G), \end{cases}$$

$\ell_1 \ge E_{t-1} + 1 + \varepsilon.$

We solve a sequence of linear programs.

For C_t with $v_1 < v_2 < \cdots < v_k$: push the rep. as far to the left as possible

Minimize: $E_t := \ell_k + 1,$

subject to: $\ell_i \le \ell_{i+1},$ ordering preserved

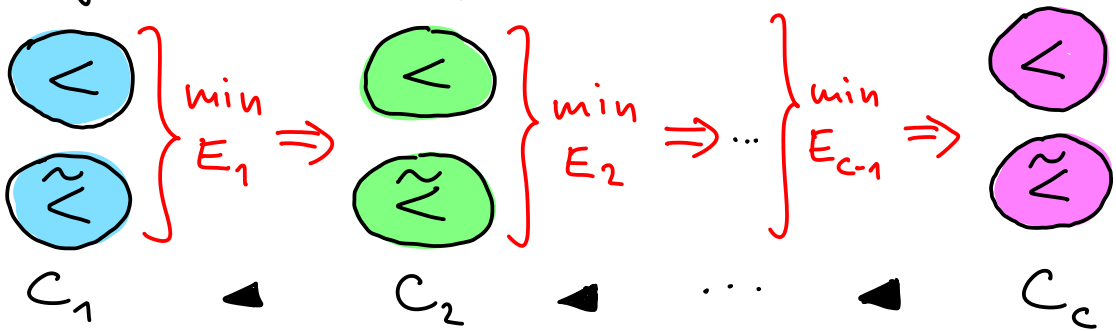
$\mathcal{R}$ extends $\mathcal{R}'$ $\quad \ell_i = \ell'_i$ if v_i is predrawn,

$$\forall\, v_i < v_j : \begin{cases} \ell_i \ge \ell_j - 1 & v_i v_j \in E(G), \\ \ell_i \le \ell_j - 1 - \varepsilon & v_i v_j \notin E(G), \end{cases}$$

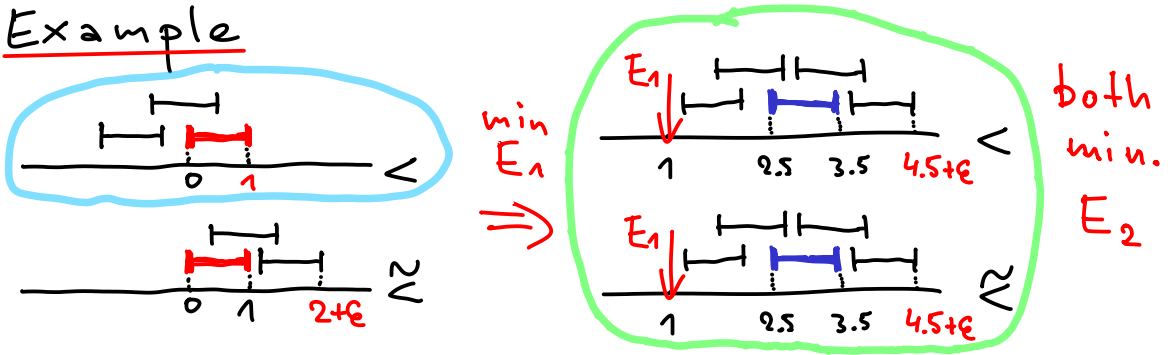
correct UNIT INT represen.

$\ell_1 \ge E_{t-1} + 1 + \varepsilon.$ on the right of the prev. component

For each C_t, we solve two linear programs: for $<$, and for the reversal $\tilde{<}$.



Example



The value ε can be always chosen

$$\varepsilon := \frac{1}{n \cdot \mathrm{lcm}(q_1, \dots, q_n)}, \quad \text{where } \ell_i' = \frac{p_i}{q_i}.$$

Therefore, REPEXT(UNIT INT) $\in$ P.

The LP's are systems of difference constraints, so they can be solved using Bellman-Ford shortest path algorithm, giving running time:

$$O(n^2 r + n D(r))$$

the size of input in binary

The value ε can be always chosen

$$\varepsilon := \frac{1}{n \cdot \mathrm{lcm}(q_1, \dots, q_n)}, \quad \text{where } \ell_i' = \frac{p_i}{q_i}.$$

Therefore, REPEXT(UNIT INT) $\in$ P.

The LP's are systems of difference constraints, so they can be solved using Bellman-Ford shortest path algorithm, giving running time:

$$O(n^2 r + n D(r)) \xrightarrow{\text{we improve}} O(n^2 + n D(r)).$$

almost quadratic

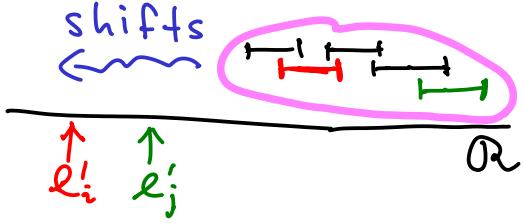
③ Alternative algorithm for RepExt(unit int).

(i) We start with some initial representation, modify it by a sequence of shifts, preserving correctness of representation.

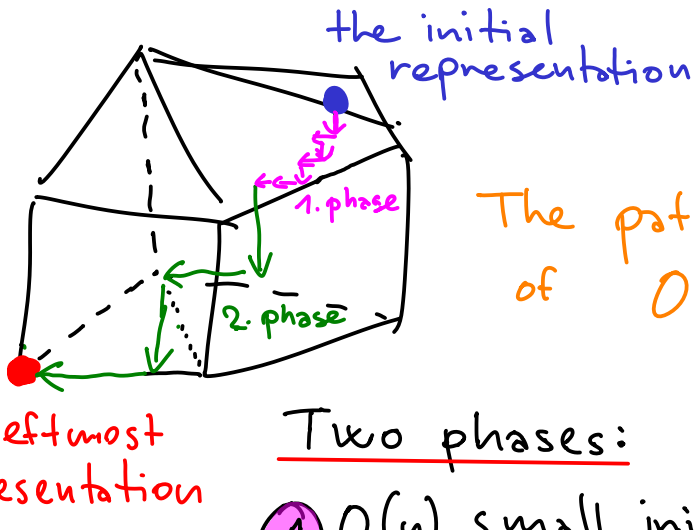
(ii) We end up with the leftmost rep. minimizing all $\ell_1, \dots, \ell_k$, with the change that $\ell_i \geq \ell'_i$ $\forall v_i$ predrawn.

(iii) We test whether $\ell_i = \ell'_i$.

Can always be done using $O(n^2)$ left shifts.



Geometrically, we traverse the polytope of feasible solutions by Manhatten path.



The path consists of $O(n^2)$ shifts.

Two phases:

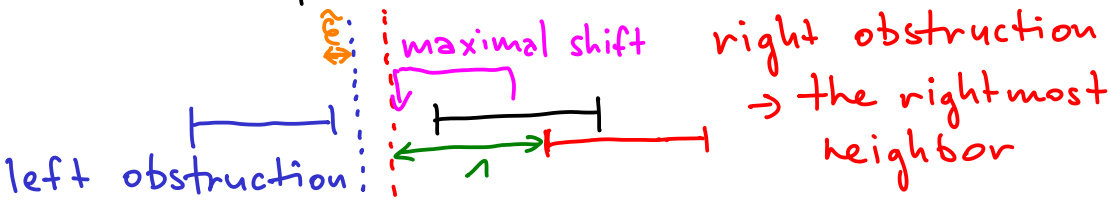
1. $O(n)$ small initial shifts.
2. $O(n^2)$ long shifts, each by at least $\frac{n-1}{n}$.

Several tricks involved to get $O(n^2 + nD(r))$

→ Most of the time, we shift in a grid $\tilde{\varepsilon} = 1/n^2$ to avoid long numbers.

→ The precise position is calculated only when an interval reaches its final destination.

→ For shifting of every interval, there are two possible obstructions.



→ the rightmost non-neighbor on the left

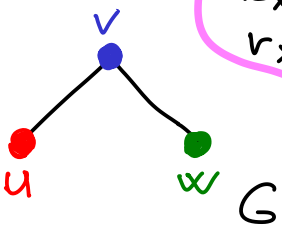
Why PROPER INT and UNIT INT are very different?

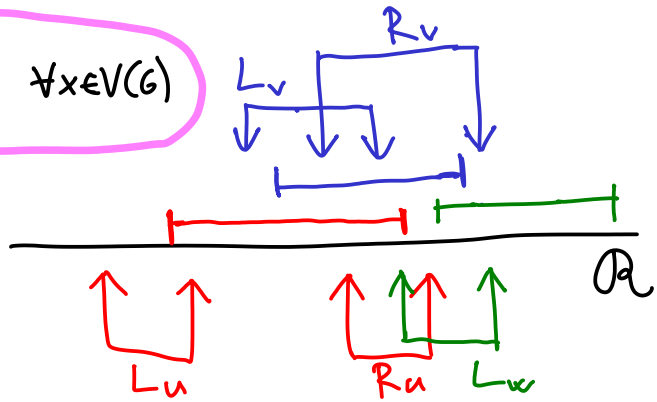
It is known that geometry is much more complicated than topology.

Bounded Representations (BOUNDREP)

→ generalizes REPEXT

$$\ell_x \in L_x \quad r_x \in R_x \quad \forall x \in V(G)$$





→ BoundRep(UNIT INT) ∈ NP-complete,

the ordering ◀ cannot be derived!

→ But by Balko, K., Otachi (ISAAC 2013):

BoundRep(PROPER INT) ∈ P,

the ordering ◀ can be derived.

So unless P = NP, proper and unit interval representations are vastly different.

Thank You!