

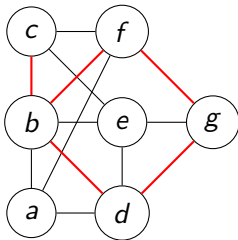
Twin-width IV: ordered graphs and matrices

Édouard Bonnet, Ugo Giocanti, Patrice Ossona de Mendez,
Pierre Simon, Stéphan Thomassé, and Szymon Toruńczyk

ENS Lyon, LIP

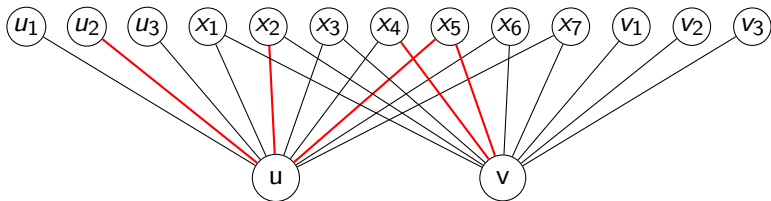
STOC 2022, Rome

Trigraphs



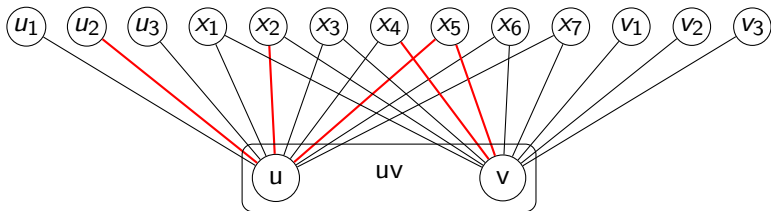
Three outcomes between a pair of vertices:
edge, or non-edge, or red edge (error edge)

Contractions in trigraphs



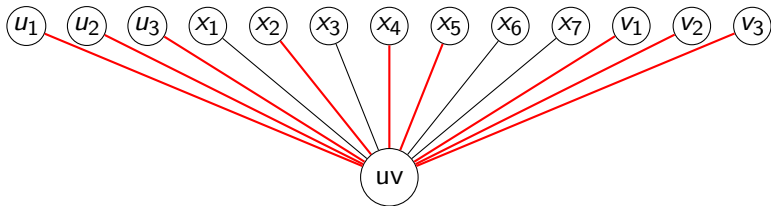
Identification of two non-necessarily adjacent vertices

Contractions in trigraphs



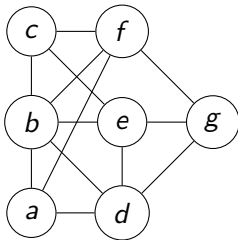
Identification of two non-necessarily adjacent vertices

Contractions in trigraphs



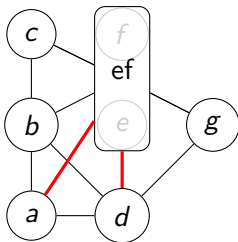
edges to $N(u) \Delta N(v)$ turn red, for $N(u) \cap N(v)$ red is absorbing

Contraction sequence



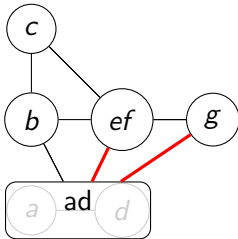
A contraction sequence of G :
Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that G_i is obtained by performing one contraction in G_{i+1} .

Contraction sequence



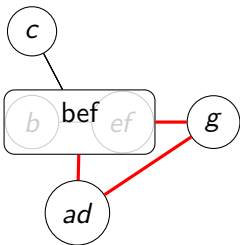
A contraction sequence of G :
Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that
 G_i is obtained by performing one contraction in G_{i+1} .

Contraction sequence



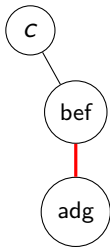
A contraction sequence of G :
Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that
 G_i is obtained by performing one contraction in G_{i+1} .

Contraction sequence



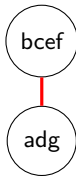
A contraction sequence of G :
Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that
 G_i is obtained by performing one contraction in G_{i+1} .

Contraction sequence



A contraction sequence of G :
Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that
 G_i is obtained by performing one contraction in G_{i+1} .

Contraction sequence



A contraction sequence of G :
Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that
 G_i is obtained by performing one contraction in G_{i+1} .

Contraction sequence

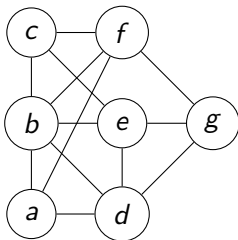


A contraction sequence of G :

Sequence of trigraphs $G = G_n, G_{n-1}, \dots, G_2, G_1$ such that G_i is obtained by performing one contraction in G_{i+1} .

Twin-width

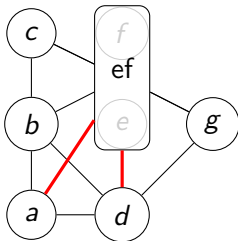
$\text{tww}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 0
overall maximum red degree = 0

Twin-width

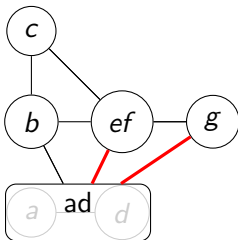
$\text{tw}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 2
overall maximum red degree = 2

Twin-width

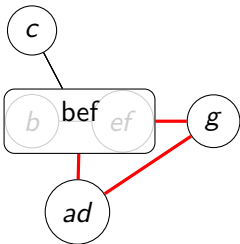
$\text{tw}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 2
overall maximum red degree = 2

Twin-width

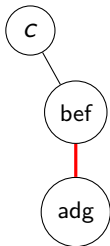
$\text{tw}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 2
overall maximum red degree = 2

Twin-width

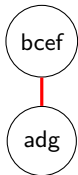
$\text{tww}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 1
overall maximum red degree = 2

Twin-width

$\text{tww}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 1
overall maximum red degree = 2

Twin-width

$\text{tww}(G)$: Least integer d such that G admits a contraction sequence where all trigraphs have *maximum red degree* at most d .



Maximum red degree = 0
overall maximum red degree = 2

Theorem (B., Geniet, Kim, Thomassé, Watrigant '20 & '21)

The following classes have bounded twin-width, and $O(1)$ -sequences can be computed in polynomial time.

- ▶ *Bounded rank-width, and even, boolean-width graphs,*
- ▶ *every hereditary proper subclass of permutation graphs,*
- ▶ *posets of bounded antichain size,*
- ▶ *unit interval graphs,*
- ▶ *K_t -minor free graphs,*
- ▶ *map graphs with embedding,*
- ▶ *d -dimensional grids,*
- ▶ *K_t -free unit d -dimensional ball graphs,*
- ▶ *$\Omega(\log n)$ -subdivisions of all the n -vertex graphs,*
- ▶ *cubic expanders defined by iterative random 2-lifts from K_4 ,*
- ▶ *flat classes,*
- ▶ *subgraphs of every $K_{t,t}$ -free class above,*
- ▶ *first-order transductions of all the above.*

First-order model checking on graphs

GRAPH FO MODEL CHECKING

Parameter: $|\varphi|$

Input: A graph G and a first-order sentence $\varphi \in FO(\{E\})$

Question: $G \models \varphi?$

First-order model checking on graphs

GRAPH FO MODEL CHECKING

Parameter: $|\varphi|$

Input: A graph G and a first-order sentence $\varphi \in FO(\{E\})$

Question: $G \models \varphi?$

Example:

$$\varphi = \exists x_1 \exists x_2 \cdots \exists x_k \forall x \forall y (E(x, y) \Rightarrow \bigvee_{1 \leq i \leq k} x = x_i \vee y = x_i)$$

$G \models \varphi? \Leftrightarrow k$ -VERTEX COVER

First-order model checking on graphs

GRAPH FO MODEL CHECKING

Parameter: $|\varphi|$

Input: A graph G and a first-order sentence $\varphi \in FO(\{E\})$

Question: $G \models \varphi?$

Example:

$$\varphi = \exists x_1 \exists y_1 \cdots \exists x_k \exists y_k \bigwedge_{\{x,y\} \in (\{x_1,y_1, \dots, x_k,y_k\})} x \neq y$$

$$\wedge E(x, y) \Leftrightarrow \bigvee_{1 \leq i \leq k} (x = x_i \wedge y = y_i) \vee (x = y_i \wedge y = x_i)$$

$$G \models \varphi? \Leftrightarrow$$

First-order model checking on graphs

GRAPH FO MODEL CHECKING

Parameter: $|\varphi|$

Input: A graph G and a first-order sentence $\varphi \in FO(\{E\})$

Question: $G \models \varphi?$

Example:

$$\varphi = \exists x_1 \exists y_1 \cdots \exists x_k \exists y_k \bigwedge_{\{x,y\} \in \binom{\{x_1, y_1, \dots, x_k, y_k\}}{2}} x \neq y$$

$$\wedge E(x, y) \Leftrightarrow \bigvee_{1 \leq i \leq k} (x = x_i \wedge y = y_i) \vee (x = y_i \wedge y = x_i)$$

$G \models \varphi? \Leftrightarrow k$ -INDUCED MATCHING

First-order model checking on graphs

GRAPH FO MODEL CHECKING

Parameter: $|\varphi|$

Input: A graph G and a first-order sentence $\varphi \in FO(\{E\})$

Question: $G \models \varphi?$

Example:

$$\varphi = \bigvee_{1 \leq q \leq k, q \text{ is odd}} \exists x_1 \notin \{s\} E(s, x_1) \wedge (\forall x_2 \notin \{s, x_1\} \neg E(x_1, x_2) \vee$$

$$(\exists x_3 \notin \{s, x_1, x_2\} E(x_2, x_3) \wedge (\forall x_4 \cdots (\exists x_q \notin \{s, x_1, \dots, x_{q-1}\} E(x_{q-1}, x_q) \wedge (\forall x_{q+1} \neg E(x_q, x_{q+1}) \vee x_{q+1} \in \{s, x_1, \dots, x_q\}))) \cdots)))$$

$$G \models \varphi? \Leftrightarrow$$

First-order model checking on graphs

GRAPH FO MODEL CHECKING

Parameter: $|\varphi|$

Input: A graph G and a first-order sentence $\varphi \in FO(\{E\})$

Question: $G \models \varphi?$

Example:

$$\varphi = \bigvee_{1 \leq q \leq k, q \text{ is odd}} \exists x_1 \notin \{s\} E(s, x_1) \wedge (\forall x_2 \notin \{s, x_1\} \neg E(x_1, x_2) \vee$$

$$(\exists x_3 \notin \{s, x_1, x_2\} E(x_2, x_3) \wedge (\forall x_4 \cdots (\exists x_q \notin \{s, x_1, \dots, x_{q-1}\} E(x_{q-1}, x_q) \wedge (\forall x_{q+1} \neg E(x_q, x_{q+1}) \vee x_{q+1} \in \{s, x_1, \dots, x_q\}))) \cdots)))$$

$G \models \varphi? \Leftrightarrow$ SHORT GENERALIZED GEOGRAPHY

FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

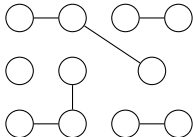
FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

FO transduction: color by $O(1)$ unary relations, interpret, delete



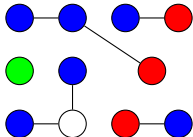
FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

FO transduction: color by $O(1)$ unary relations, interpret, delete



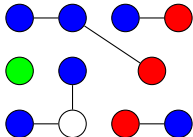
FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

FO transduction: color by $O(1)$ unary relations, interpret, delete



$$\varphi(x, y) = E(x, y) \vee (G(x) \wedge B(y) \wedge \neg \exists z R(z) \wedge E(y, z)) \\ \vee (R(x) \wedge B(y) \wedge \exists z R(z) \wedge E(y, z) \wedge \neg \exists z B(z) \wedge E(y, z))$$

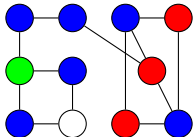
FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

FO transduction: color by $O(1)$ unary relations, interpret, delete



$$\varphi(x, y) = E(x, y) \vee (G(x) \wedge B(y) \wedge \neg \exists z R(z) \wedge E(y, z)) \\ \vee (R(x) \wedge B(y) \wedge \exists z R(z) \wedge E(y, z) \wedge \neg \exists z B(z) \wedge E(y, z))$$

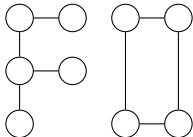
FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

FO transduction: color by $O(1)$ unary relations, interpret, delete



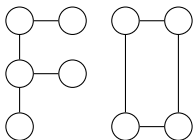
FO interpretations and transductions

FO simple interpretation: redefine the edges by a first-order formula

$$\varphi(x, y) = \neg E(x, y) \quad (\text{complement})$$

$$\varphi(x, y) = E(x, y) \vee \exists z E(x, z) \wedge E(z, y) \quad (\text{square})$$

FO transduction: color by $O(1)$ unary relations, interpret, delete



Theorem (B., Kim, Thomassé, Watrigant '20)

Transductions of bounded twin-width classes have bounded twin-width.

Dependence and monadic dependence

A class \mathcal{C} is

dependent, if the hereditary closure of every interpretation of \mathcal{C} misses some graph

monadically dependent, if every transduction of \mathcal{C} misses some graph [Baldwin, Shelah '85]

Dependence and monadic dependence

A class \mathcal{C} is

dependent, if the hereditary closure of every interpretation of \mathcal{C} misses some graph

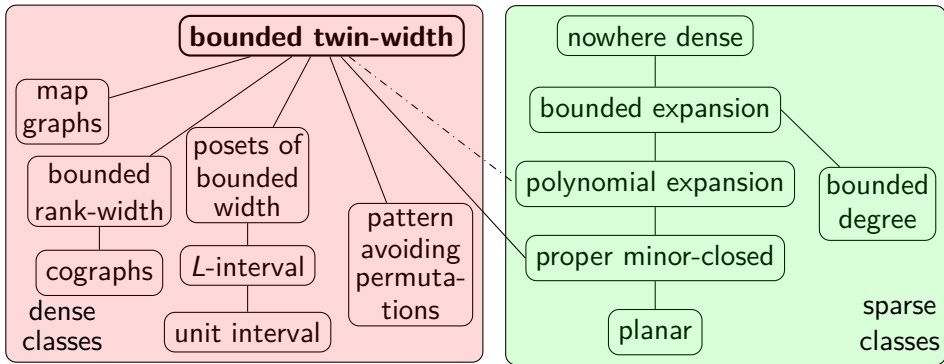
monadically dependent, if every transduction of \mathcal{C} misses some graph [Baldwin, Shelah '85]

Theorem (Downey, Fellows, Taylor '96)

FO model checking is AW[]-complete on general graphs,
thus unlikely FPT on independent classes*

Could it be that on every dependent class, it is FPT?

Classes with known tractable FO model checking



Theorem (B., Kim, Thomassé, Watrigant '20)

FO MODEL CHECKING solvable in $f(|\varphi|, d)n$ on graphs with a d -sequence.

Small classes

Small: class with at most $n!c^n$ labeled graphs on $[n]$.

Theorem (B., Geniet, Kim, Thomassé, Watrigant '21)

Bounded twin-width classes are small.

Unifies and extends the same result for:

σ -free permutations [Marcus, Tardos '04]

K_t -minor free graphs [Norine, Seymour, Thomas, Wollan '06]

Small classes

Small: class with at most $n!c^n$ labeled graphs on $[n]$.

Theorem (B., Geniet, Kim, Thomassé, Watrigant '21)

Bounded twin-width classes are small.

Is the converse true for hereditary classes?

Conjecture (small conjecture, now refuted in general, Tww VII)

A hereditary class has bounded twin-width if and only if it is small.

Recap of the main questions

- ▶ Can we efficiently approximate twin-width?
- ▶ Can we solve FO model checking on every dependent class?
- ▶ Is every hereditary small class of bounded twin-width?

Recap of the main questions

- ▶ Can we efficiently approximate twin-width?
- ▶ Can we solve FO model checking on every dependent class?
- ▶ Is every hereditary small class of bounded twin-width?

We answer all these questions positively in the case of ordered binary structures \equiv matrices on a finite alphabet

Twin-width for unordered matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Encode a bipartite graph (or, if symmetric, a graph)

Twin-width for unordered matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Contraction of two columns (similar with two rows)

Twin-width for unordered matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & r & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & r & 0 & 1 & 1 & 0 \\ 1 & 0 & r & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

The red degree is now the max number of r per row/column

Twin-width for unordered matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & r & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & r & 0 & 1 & 1 & 0 \\ 1 & 0 & r & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

In the non-bipartite case, we force symmetric pairs of contractions

Twin-width for matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

That was *not* the twin-width of **ordered** matrices

Twin-width for matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & r & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & r & 0 & 1 & 1 & 0 \\ 1 & 0 & r & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

Let's also record the columns disagreeing with the contraction

Twin-width for matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & r & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & r & 0 & 1 & 1 & 0 \\ 1 & 0 & r & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

$$\max_{\text{row, column}} (\text{number of red entries} + \text{red degree})$$

Twin-width for matrices

$$\begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

If you find it too clumsy, encode the linear order

Twin-width for matrices

$$\begin{bmatrix} 3 & 3 & 3 & 3 & 3 & 3 & 1 \\ 2 & 3 & 3 & 2 & 2 & 0 & 1 \\ 2 & 2 & 2 & 2 & 0 & 0 & 0 \\ 2 & 3 & 2 & 0 & 1 & 1 & 0 \\ 3 & 2 & 0 & 1 & 1 & 1 & 0 \\ 2 & 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{bmatrix}$$

and we're back to the unordered definition

Partition viewpoint

Matrix partition: partitions of the row set and of the column set

Matrix division: same but all the parts are *consecutive*

1	1	1	1	1	1	1	0
0	1	1	0	0	1	0	1
0	0	0	0	0	0	0	1
0	1	0	0	1	0	1	0
1	0	0	1	1	0	1	0
0	1	1	1	1	1	0	0
1	0	1	1	1	0	0	1

Partition viewpoint

Matrix partition: partitions of the row set and of the column set

Matrix division: same but all the parts are *consecutive*

1	1	1	1	1	1	1	0
0	1	1	0	0	1	0	1
0	0	0	0	0	0	0	1
0	1	0	0	1	0	1	0
1	0	0	1	1	0	1	0
0	1	1	1	1	1	0	0
1	0	1	1	1	0	0	1

Maximum number of non-constant zones per column or row part
= error value

Partition viewpoint

Matrix partition: partitions of the row set and of the column set

Matrix division: same but all the parts are *consecutive*

1	1	1	1	1	1	1	0
0	1	1	0	0	1	0	1
0	0	0	0	0	0	0	1
0	1	0	0	1	0	1	0
1	0	0	1	1	0	1	0
0	1	1	1	1	1	0	0
1	0	1	1	1	0	0	1

Maximum number of non-constant zones per column or row part
... until there are a single row part and column part

Partition viewpoint

Matrix partition: partitions of the row set and of the column set

Matrix division: same but all the parts are *consecutive*

1	1	1	1	1	1	1	0
0	1	1	0	0	1	0	1
0	0	0	0	0	0	0	1
0	1	0	0	1	0	1	0
1	0	0	1	1	0	1	0
0	1	1	1	1	1	0	0
1	0	1	1	1	0	0	1

**Twin-width as maximum error value
of a contraction sequence**

Matrix FO model checking

Signature for 0,1-matrices $\sigma = \{R^{(1)}, <^{(2)}, E^{(2)}\}$
($E^{(2)}$ becomes $E_1^{(2)}, \dots, E_t^{(2)}$ for $[0, t]$ -matrices)

Matrix FO model checking

Signature for 0,1-matrices $\sigma = \{R^{(1)}, <^{(2)}, E^{(2)}\}$
($E^{(2)}$ becomes $E_1^{(2)}, \dots, E_t^{(2)}$ for $[0, t]$ -matrices)

- ▶ $M \models R(x)$ iff x is a row index
- ▶ $M \models x < y$ iff x is a smaller index than y
- ▶ $M \models E(x, y)$ iff $M_{x,y} = 1$

Matrix FO model checking

Signature for 0,1-matrices $\sigma = \{R^{(1)}, <^{(2)}, E^{(2)}\}$
($E^{(2)}$ becomes $E_1^{(2)}, \dots, E_t^{(2)}$ for $[0, t]$ -matrices)

- ▶ $M \models R(x)$ iff x is a row index
- ▶ $M \models x < y$ iff x is a smaller index than y
- ▶ $M \models E(x, y)$ iff $M_{x,y} = 1$

tractable class: FO model checking solvable in time $f(\varphi)|M|^{O(1)}$

Growth of classes

Our matrix *classes* are closed under taking submatrices

- ▶ Small class: $\#n \times n$ matrices is $2^{O(n)}$
- ▶ Subfactorial: ultimately, $\#n \times n$ matrices $< n!$

No non-trivial automorphism in totally ordered structures,
so no need for labels

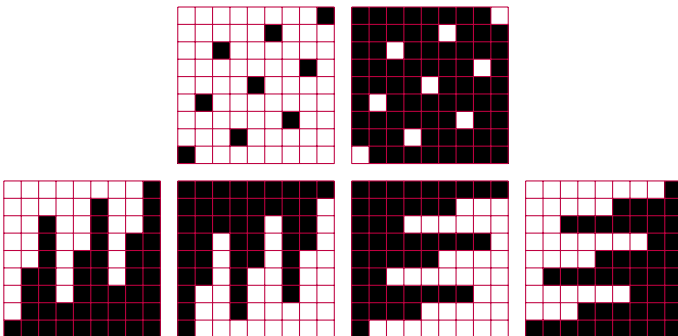
Equivalences in the matrix language

Theorem

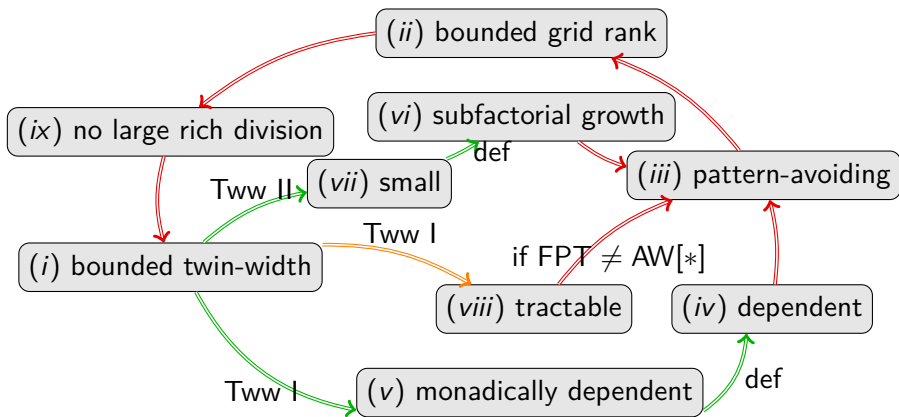
For every matrix class \mathcal{M} , the following are equivalent.

- (i) \mathcal{M} has bounded twin-width.
- (ii) \mathcal{M} has **bounded grid rank**. (division property)
- (iii) \mathcal{M} is **pattern-avoiding**.
(not including any of 6 “permutation-universal” classes)
- (iv) \mathcal{M} is dependent.
- (v) \mathcal{M} is monadically dependent.
- (vi) \mathcal{M} has subfactorial growth.
- (vii) \mathcal{M} is small.
- (viii) \mathcal{M} is tractable. (only if $\text{FPT} \neq \text{AW}[*]$.)
- (ix) \mathcal{M} has no large **rich division**. (division property)

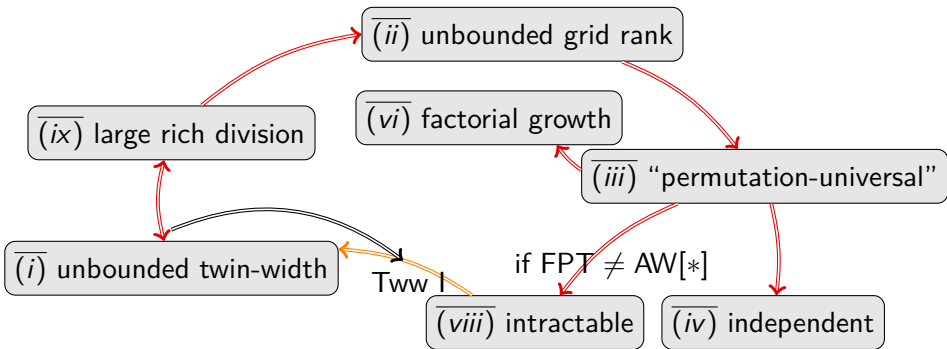
The 6 permutation-universal classes



Roadmap



Roadmap



Equivalences in the ordered graph language

Theorem

Let \mathcal{C} be a hereditary class of ordered graphs. The following are equivalent.

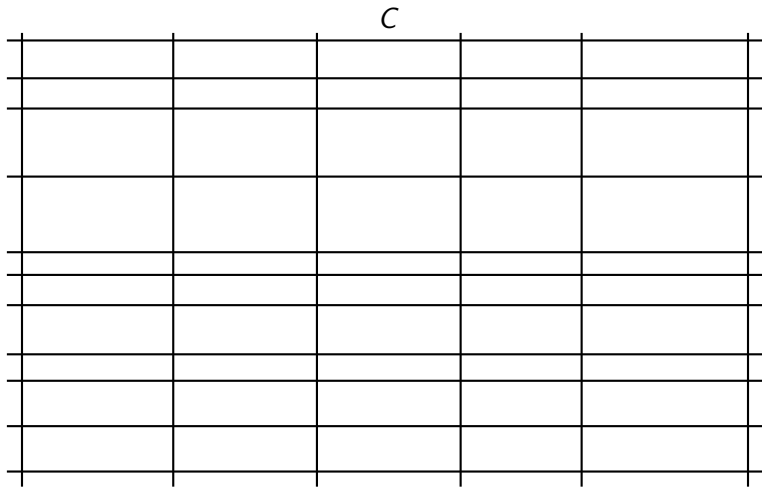
- (1) \mathcal{C} has bounded twin-width.
- (2) \mathcal{C} is monadically dependent.
- (3) \mathcal{C} is dependent.
- (4) \mathcal{C} is small.
- (5) \mathcal{C} contains $2^{O(n)}$ ordered n -vertex graphs.
- (6) \mathcal{C} contains less than $\sum_{k=0}^{\lfloor n/2 \rfloor} \binom{n}{2k} k!$ ordered n -vertex graphs, for some n .
- (7) \mathcal{C} does not include one of 25 hereditary ordered graph classes $\mathcal{M}_{s,\lambda,\rho}$ and \mathcal{P} with unbounded twin-width.
- (8) FO-model checking is fixed-parameter tractable on \mathcal{C} .

k -Rich division

A 10x5 grid representing a k -Rich division. The grid consists of 10 rows and 5 columns, with a vertical line on the far left and far right. The grid is empty, intended for a diagram illustrating the concept of a k -Rich division.

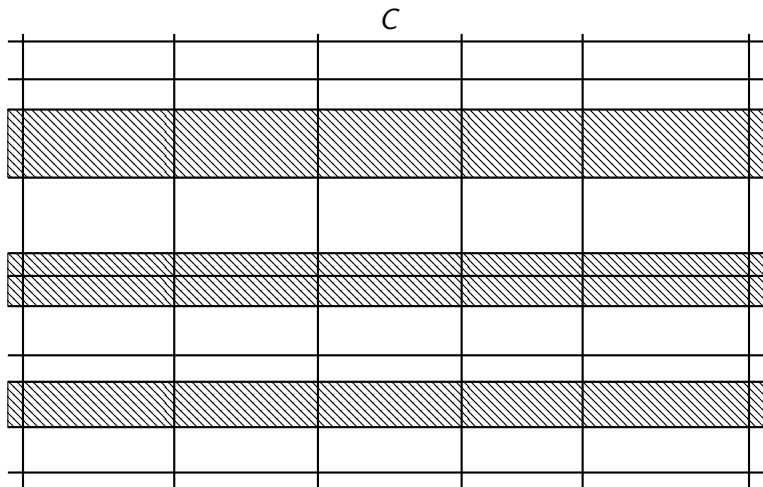
Division

k -Rich division



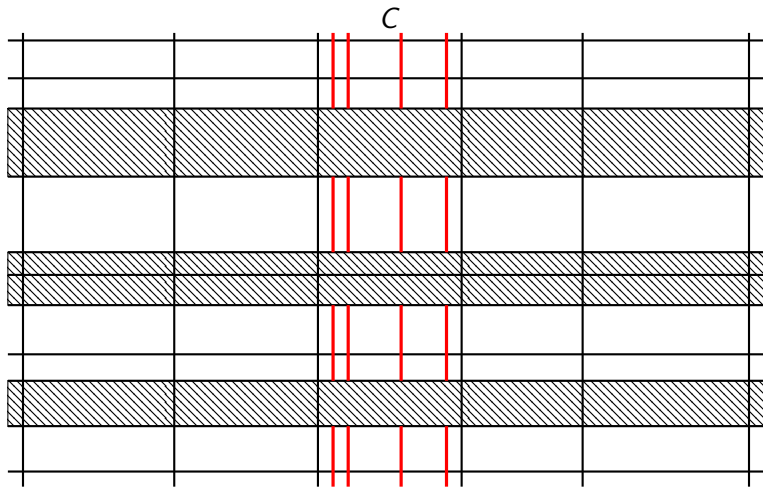
Division such that for each, say, column part C

k -Rich division



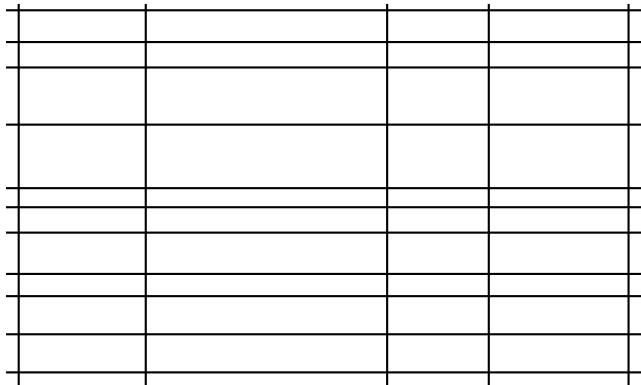
Division such that for each, say, column part C no removal of k row parts

k -Rich division



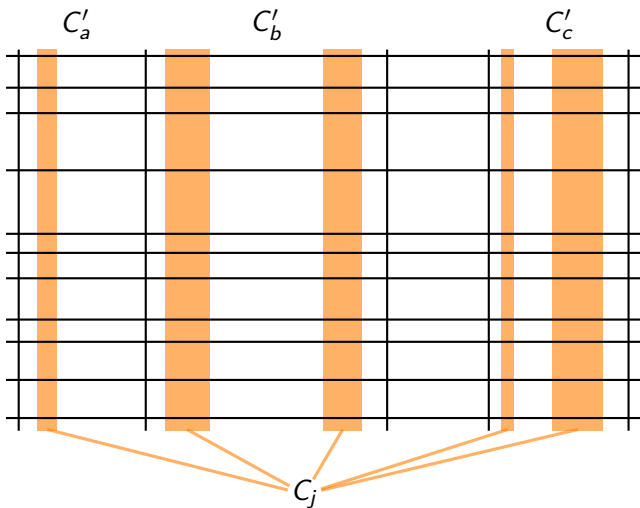
Division such that for each, say, column part C no removal of k row parts leaves C with less than k distinct column vectors

Large rich division \Rightarrow unbounded twin-width



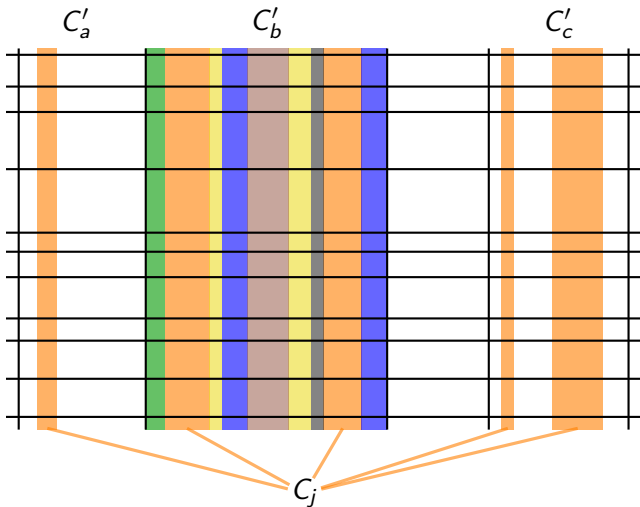
Fix an $2k(k+1)$ -rich division \mathcal{D} , and assume there is a k -sequence \mathcal{S}

Large rich division \Rightarrow unbounded twin-width



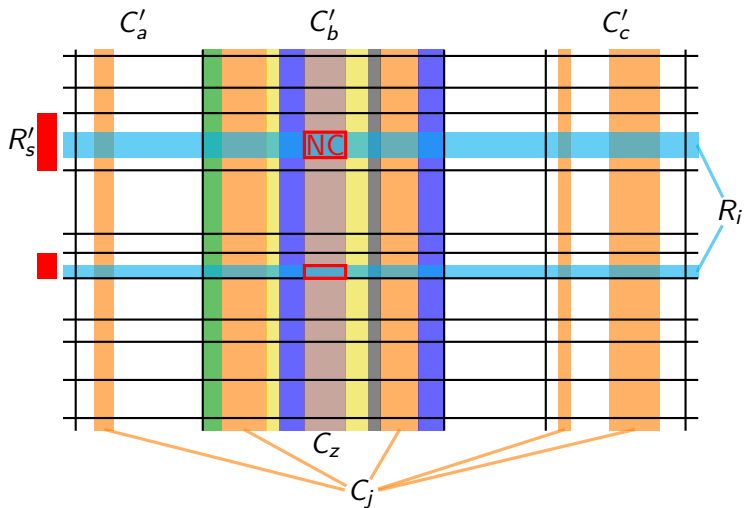
Consider the first time a part of \mathcal{S} intersects 3 parts of \mathcal{D}

Large rich division \Rightarrow unbounded twin-width



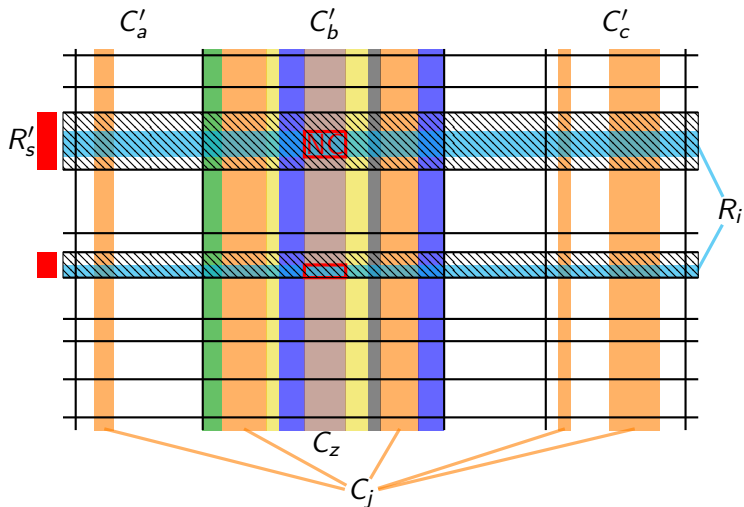
There are at most k other column parts intersecting C'_b (red degree of C_j)

Large rich division \Rightarrow unbounded twin-width



Each such part C_z is non-constant in at most $2k$ zones of \mathcal{D}

Large rich division \Rightarrow unbounded twin-width



Thus removing $2k(k + 1)$ row parts of $\mathcal{D} \rightarrow \leq k + 1$ distinct columns

No large rich division \Rightarrow bounded twin-width

Build greedily a division where every part contradicts the richness

- ▶ can only be stopped by a large rich division
- ▶ turned into a contraction sequence as in Tww I

No large rich division \Rightarrow bounded twin-width

Build greedily a division where every part contradicts the richness

- ▶ can only be stopped by a large rich division
- ▶ turned into a contraction sequence as in Tww I

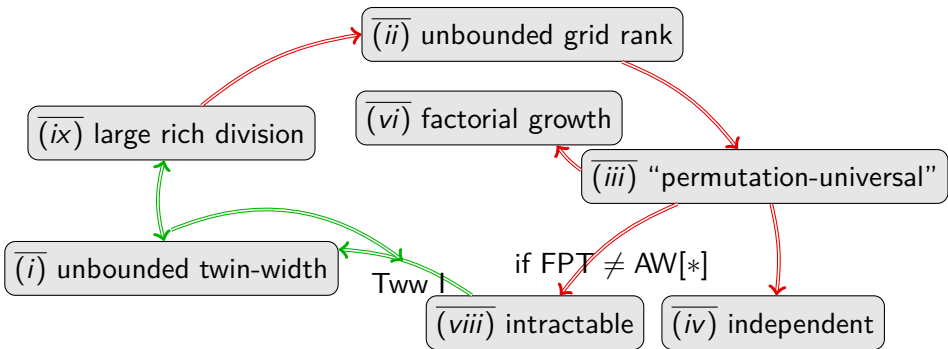
\rightarrow approximation of twin-width for ordered binary structures

Theorem

There is a fixed-parameter algorithm, which, given an ordered binary structure G and a parameter k , either outputs

- ▶ *a $2^{O(k^4)}$ -sequence of G , implying that $\text{tw}(G) = 2^{O(k^4)}$, or*
- ▶ *a $2k(k+1)$ -rich division of $M(G)$, implying that $\text{tw}(G) > k$.*

Roadmap



k -rank division

$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$
$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$
$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$
$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$

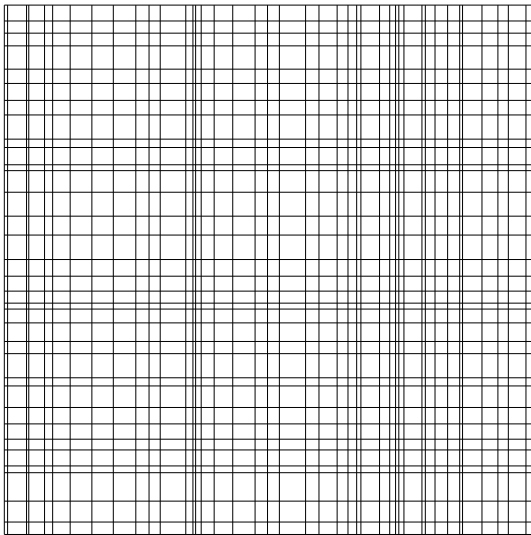
k -by- k division where every cell has rank at least k

k -rank division

$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$
$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$
$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$
$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$	$\text{rank} \geq k$

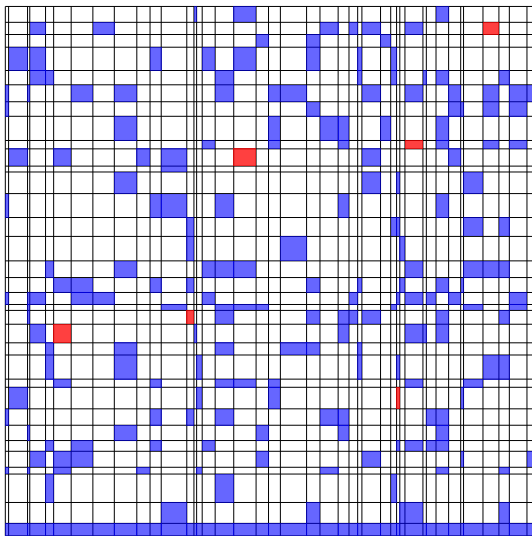
Grid rank of M = largest k such that M admits a k -rank division

Large rich division \Rightarrow unbounded grid rank



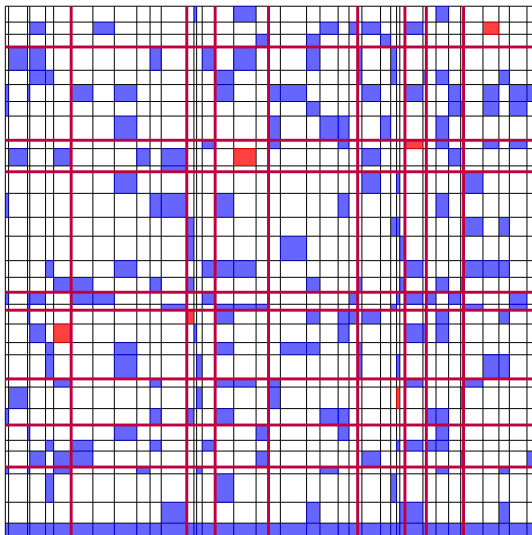
Fix a large rich division \mathcal{D}

Large rich division \Rightarrow unbounded grid rank



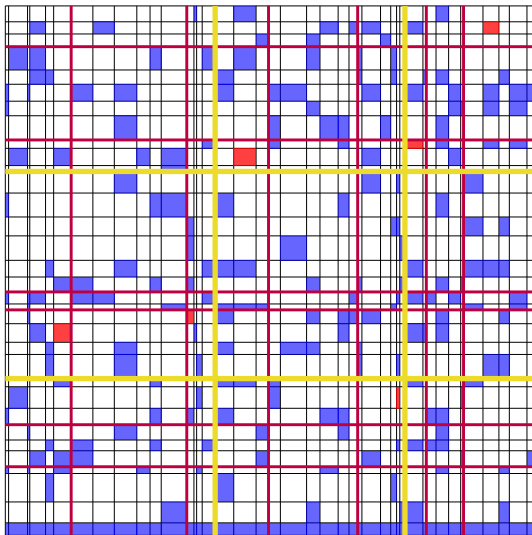
Red zones = large rank; Blue zones = first of its column to contain a particular row vector

Large rich division \Rightarrow unbounded grid rank



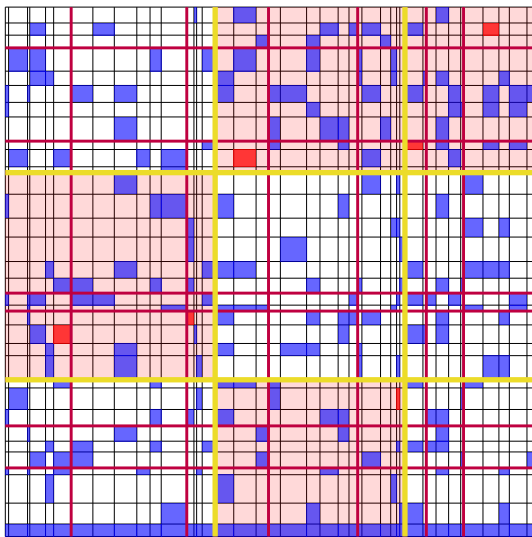
Marcus-Tardos theorem applied to the colored zones \rightarrow division \mathcal{D}'

Large rich division \Rightarrow unbounded grid rank



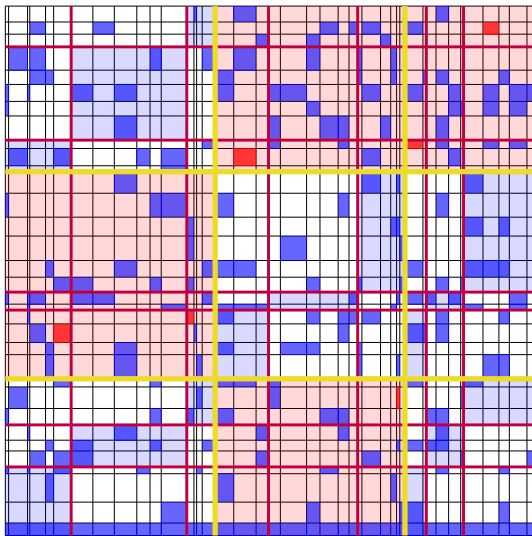
Coarser division \mathcal{D}'' , 1 zone of $\mathcal{D}'' \equiv 2^k \times 2^k$ zones of \mathcal{D}'

Large rich division \Rightarrow unbounded grid rank



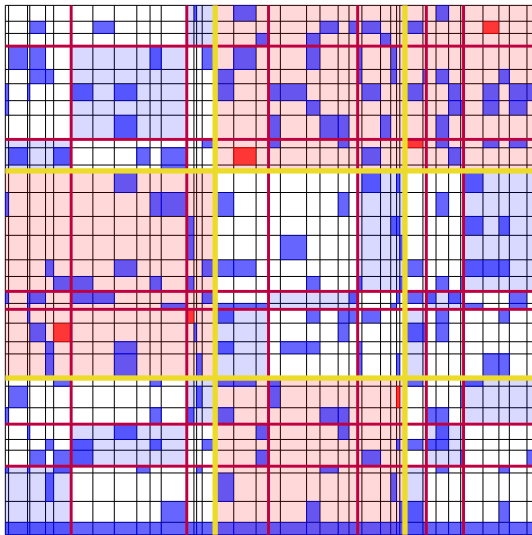
A zone of \mathcal{D}'' containing a red zone has large rank

Large rich division \Rightarrow unbounded grid rank



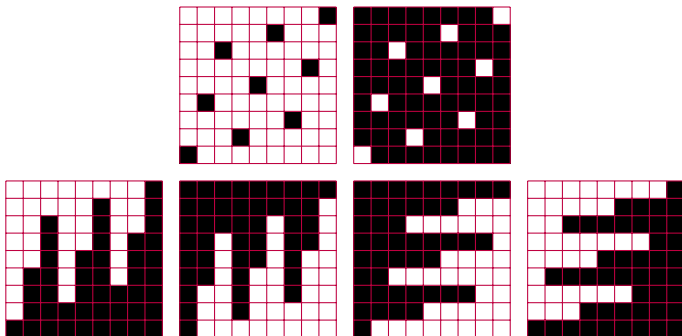
Other zones have diagonals of blue zones

Large rich division \Rightarrow unbounded grid rank



2^k distinct row vectors in each zone of \mathcal{D}''

Regularizing with Ramsey's theorem



Roadmap

