

MINIMUM IDENTIFYING CODES IN CIRCULAR STRIPS OF GIVEN HEIGHT

MARWANE BOUZNIF

G-SCOP laboratory
Grenoble FRANCE

23/11/2011

joint work with

Julien DARLAY, Julien MONCEL, Myriam PREISSMANN

Problem studied

Question :

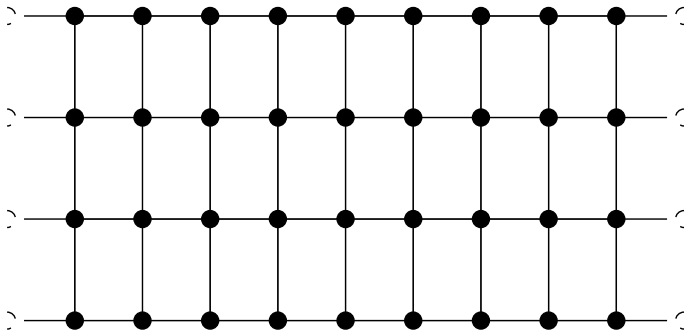
What is the minimum density of an identifying code on the infinite strip of height 3 ?

- 1 Definitions and motivations
- 2 The algorithm
- 3 Conclusion and perspective

Summary

- 1 Definitions and motivations
- 2 The algorithm
- 3 Conclusion and perspective

The infinite strip of height 4 : \mathcal{S}_4^∞



Optimization problem

finite graph VS infinite graph

finite graph : minimum identifying code

infinite graph : identifying code of minimum density

Periodicity of an identifying code on the infinite strip

Proposition

At least one identifying code of \mathcal{S}_h^∞ of minimum density is periodic (has a pattern which repeats itself along the strip).

What is known ?

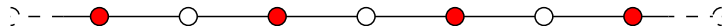
Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$

What is known ?

Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$ (period 2)



What is known ?

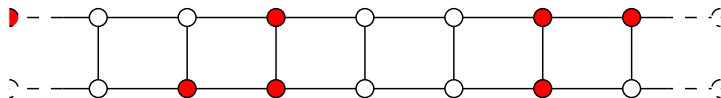
Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$ (period 2)
- $d^*(\mathcal{S}_2^\infty) = \frac{3}{7}$

What is known ?

Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$ (period 2)
- $d^*(\mathcal{S}_2^\infty) = \frac{3}{7}$ (period 7)



What is known ?

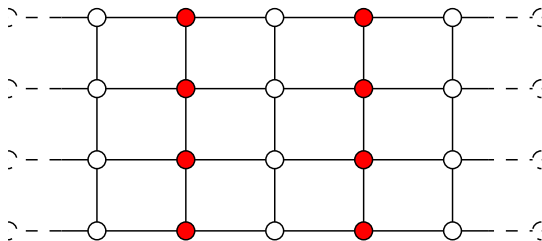
Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$ (period 2)
- $d^*(\mathcal{S}_2^\infty) = \frac{3}{7}$ (period 7)
- $\frac{7}{20} - \frac{1}{2k} \leq d^*(\mathcal{S}_{k \geq 3}^\infty) \leq \min\left(\frac{2}{5}, \frac{7}{20} + \frac{2}{k}\right)$

What is known ?

Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$ (period 2)
- $d^*(\mathcal{S}_2^\infty) = \frac{3}{7}$ (period 7)
- $\frac{7}{20} - \frac{1}{2k} \leq d^*(\mathcal{S}_{k \geq 3}^\infty) \leq \min(\frac{2}{5}, \frac{7}{20} + \frac{2}{k})$



What is known ?

Proposition (Daniel, Gravier, Moncel 2004)

- $d^*(\mathcal{S}_1^\infty) = \frac{1}{2}$ (period 2)
- $d^*(\mathcal{S}_2^\infty) = \frac{3}{7}$ (period 7)
- $\frac{7}{20} - \frac{1}{2k} \leq d^*(\mathcal{S}_{k \geq 3}^\infty) \leq \min(\frac{2}{5}, \frac{7}{20} + \frac{2}{k})$

Theorem (Cohen, Gravier, Honkala, Lobstein, Mollard, Payan, Zémor 1999 and Ben-Haim, Litsyn 2005)

$$d^*(\mathbb{Z}^2) = \frac{7}{20}$$

Methodology

Circular and infinite strips : graphs with a repetitive structure along a path or a cycle : rotagraphs and fasciagraphs.

Methodology

Circular and infinite strips : graphs with a repetitive structure
along a path or a cycle : rotagraphs and fasciagraphs.

A good characterization of "local" problems in rotagraphs and
fasciagraphs (generalizing results of Klavžar, Vesel 2003) linked
to

Methodology

Circular and infinite strips : graphs with a repetitive structure
along a path or a cycle : rotagraphs and fasciagraphs.

A good characterization of "local" problems in rotagraphs and fasciagraphs (generalizing results of Klavžar, Vesel 2003) linked to a generic algorithm using dynamic programming in dioid $(\min, +, \infty, 0)$ (inspired from Klavžar, Žerovnik 1996 and Žerovnik 1999).

Methodology

Circular and infinite strips : graphs with a repetitive structure
along a path or a cycle : rotagraphs and fasciagraphs.

A good characterization of "local" problems in rotagraphs and fasciagraphs (generalizing results of Klavžar, Vesel 2003) linked to a generic algorithm using dynamic programming in dioid $(\min, +, \infty, 0)$ (inspired from Klavžar, Žerovnik 1996 and Žerovnik 1999).

Identifying code being a "local" problem,

Methodology

Circular and infinite strips : graphs with a repetitive structure
along a path or a cycle : rotagraphs and fasciagraphs.

A good characterization of "local" problems in rotagraphs and fasciagraphs (generalizing results of Klavžar, Vesel 2003) linked to a generic algorithm using dynamic programming in dioid $(\min, +, \infty, 0)$ (inspired from Klavžar, Žerovnik 1996 and Žerovnik 1999).

Identifying code being a "local" problem, generic algorithm can be used in circular strips.

Summary

- 1 Definitions and motivations
- 2 The algorithm**
- 3 Conclusion and perspective

Proposition

C is an identifying code of $G = (V, E)$

\Leftrightarrow

for each subset W of $V(G)$, vertices in $\{u \mid N_G[u] \subseteq W\}$ are dominated, and pairwise separated, by vertices of $C \cap W$.

Proposition

C is an identifying code of $G = (V, E)$

\Leftrightarrow

for each subset W of $V(G)$, vertices in $\{u | N_G[u] \subseteq W\}$ are dominated, and pairwise separated, by vertices of $C \cap W$.

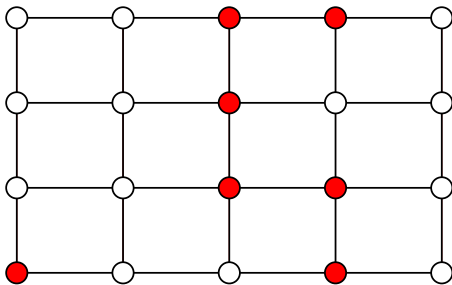
Proposition

C is an identifying code of \mathcal{S}°

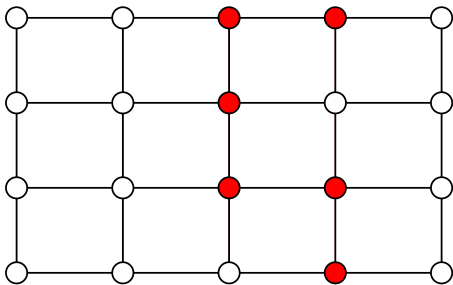
\Leftrightarrow

for each subset of 5 consecutive columns, vertices in the three middle columns are dominated, and pairwise separated, by vertices of C in these five columns.

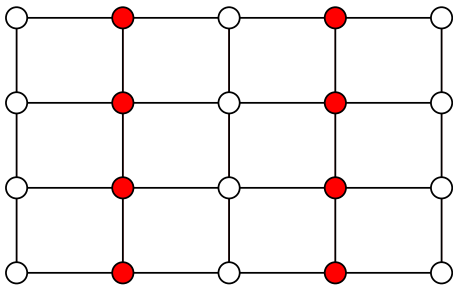
Example of a weak-code on 5 columns



Example of a weak-code on 5 columns



Example of a weak-code on 5 columns

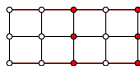
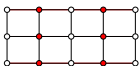
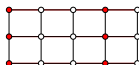
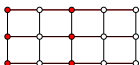
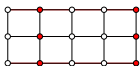


Construction of the auxiliary graph $\vec{\mathcal{G}} = (V, A)$

The vertices

$$V = \{\text{weak codes on 5 columns}\}$$

Construction of the auxiliary graph $\vec{\mathcal{G}} = (V, A)$

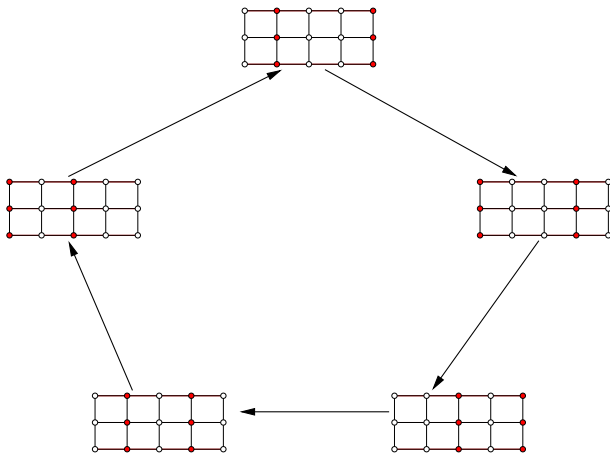


Construction of the auxiliary graph $\vec{\mathcal{G}} = (V, A)$

The arcs

$$A = \{(u, v) \mid 4 \text{ last columns of } u \text{ and } 4 \text{ first columns of } v \text{ coincide}\}$$

Construction of the auxiliary graph $\vec{\mathcal{G}} = (V, A)$

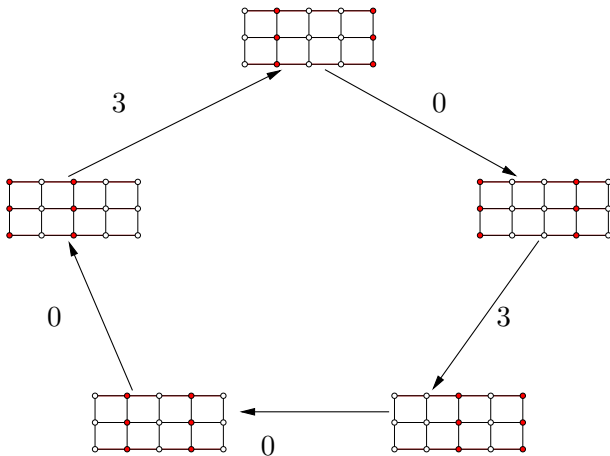


Construction of the auxiliary graph $\vec{\mathcal{G}} = (V, A)$

The weights w

$(u, v) \in A$, $w((u, v)) =$ number of vertices of the last column of v in the code.

Construction of the auxiliary graph $\vec{\mathcal{G}} = (V, A)$



Solutions and circuits of min weight in the auxiliary graph

Minimum identifying codes and circuits

Circuit of length s and weight w

\Leftrightarrow

Identifying code of $\mathcal{S}_{h \times s}^\circ$ of cardinality w .

Solutions and circuits of min weight in the auxiliary graph

Minimum identifying codes and circuits

Circuit of length s and weight w

\Leftrightarrow

Identifying code of $\mathcal{S}_{h \times s}^\circ$ of cardinality w .

Minimum weight of a circuit of length s can be obtained by computing the s -th power of the adjacency matrix \mathcal{M} of the auxiliary graph on the dioid $(\text{Min}, +, \infty, 0)$.

Solutions and circuits of min weight in the auxiliary graph

Minimum identifying codes and circuits

Minimum weight of a circuit of length s can be obtained by computing the s -th power of the adjacency matrix \mathcal{M} of the auxiliary graph on the dioid $(\text{Min}, +, \infty, 0)$.

Sequence of powers of \mathcal{M} and pseudo-period

Existence of a pseudo-period in the sequence of powers of \mathcal{M} :
 exist integers $r, P > 0, c$ such that

$$\mathcal{M}^{r+P} = \mathcal{M}^r + cJ$$

\Rightarrow only a FINITE number of matrices to compute

The algorithm

Preprocess, executed only once

- **Input** : an integer h

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Step 1** : compute **EXHAUSTIVELY** the weak-codes of $\mathcal{S}_{h,5}$.

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Step 1** : compute **EXHAUSTIVELY** the weak-codes of $\mathcal{S}_{h,5}$.
- **Step 2** : build the auxiliary graph

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Step 1** : compute **EXHAUSTIVELY** the weak-codes of $\mathcal{S}_{h,5}$.
- **Step 2** : build the auxiliary graph
- **Step 3** : compute \mathcal{M}^n until reaching the pseudo-period P .

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Step 1** : compute **EXHAUSTIVELY** the weak-codes of $\mathcal{S}_{h,5}$.
- **Step 2** : build the auxiliary graph
- **Step 3** : compute \mathcal{M}^n until reaching the pseudo-period P .
- **Output** : integers r_0, P, c and a vector τ such that $\tau(i)$ is the minimum weight of a circuit of length i in the auxiliary graph for every integer $3 \leq i \leq r_0 + P$.

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Output** : integers r_0, P, c and a vector τ such that $\tau(i)$ is the minimum weight of a circuit of length i in the auxiliary graph for every integer $3 \leq i \leq r_0 + P$.

The algorithm for $\mathcal{S}_{h,s}^\circ$, constant number of elementary operations

- **Input** : Data of preprocess and the integer s ,

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Output** : integers r_0, P, c and a vector τ such that $\tau(i)$ is the minimum weight of a circuit of length i in the auxiliary graph for every integer $3 \leq i \leq r_0 + P$.

The algorithm for $\mathcal{S}_{h,s}^\circ$, constant number of elementary operations

- **Input** : Data of preprocess and the integer s ,
- read s

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Output** : integers r_0, P, c and a vector τ such that $\tau(i)$ is the minimum weight of a circuit of length i in the auxiliary graph for every integer $3 \leq i \leq r_0 + P$.

The algorithm for $\mathcal{S}_{h,s}^\circ$, constant number of elementary operations

- **Input** : Data of preprocess and the integer s ,
- read s
- deduce integers s' and q from s, r_0 , and P
 (for $s \geq r_0$, $s' = r_0 + (s - r_0) \bmod (P)$ and $q = \frac{s - s'}{P}$)

The algorithm

Preprocess, executed only once

- **Input** : an integer h
- **Output** : integers r_0, P, c and a vector τ such that $\tau(i)$ is the minimum weight of a circuit of length i in the auxiliary graph for every integer $3 \leq i \leq r_0 + P$.

The algorithm for $\mathcal{S}_{h,s}^\circ$, constant number of elementary operations

- **Input** : Data of preprocess and the integer s ,
- read s
- deduce integers s' and q from s, r_0 , and P
 (for $s \geq r_0$, $s' = r_0 + (s - r_0) \bmod (P)$ and $q = \frac{s - s'}{P}$)
- **Output** : $\tau(s') + qc$: the minimum weight of an identifying code of $\mathcal{S}_{h,s}^\circ$.

Optimal solution for the infinite strip of height 3

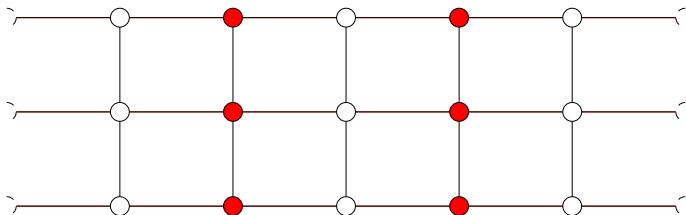


Figure: upper bound = optimal solution ? (density : $2/5$)

Optimal solution for the infinite strip of height 3

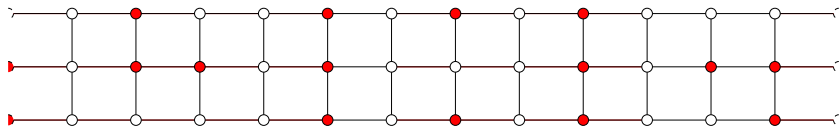


Figure: optimal solution (density : $7/18 < 2/5$)

Summary

- 1 Definitions and motivations
- 2 The algorithm
- 3 Conclusion and perspective**

Conclusion

Program used to compute also in king and triangular circular strips

Similar algorithms for finite grids (deal with side effects)

Combinatorial proof (F. Havet) with discharging method for minimum density of the infinite strip of height 3.

Perspectives

try to improve the program to work for infinite strip of height 4 and more (for now too much memory used)

in progress (F. Havet) : with discharging method, improve the known bound for height > 3 .

in progress : investigate other repetitive structure (fractal graphs ...)

Questions ?