The Reconstruction of Convex Polyominoes from Horizontal and Vertical Projections

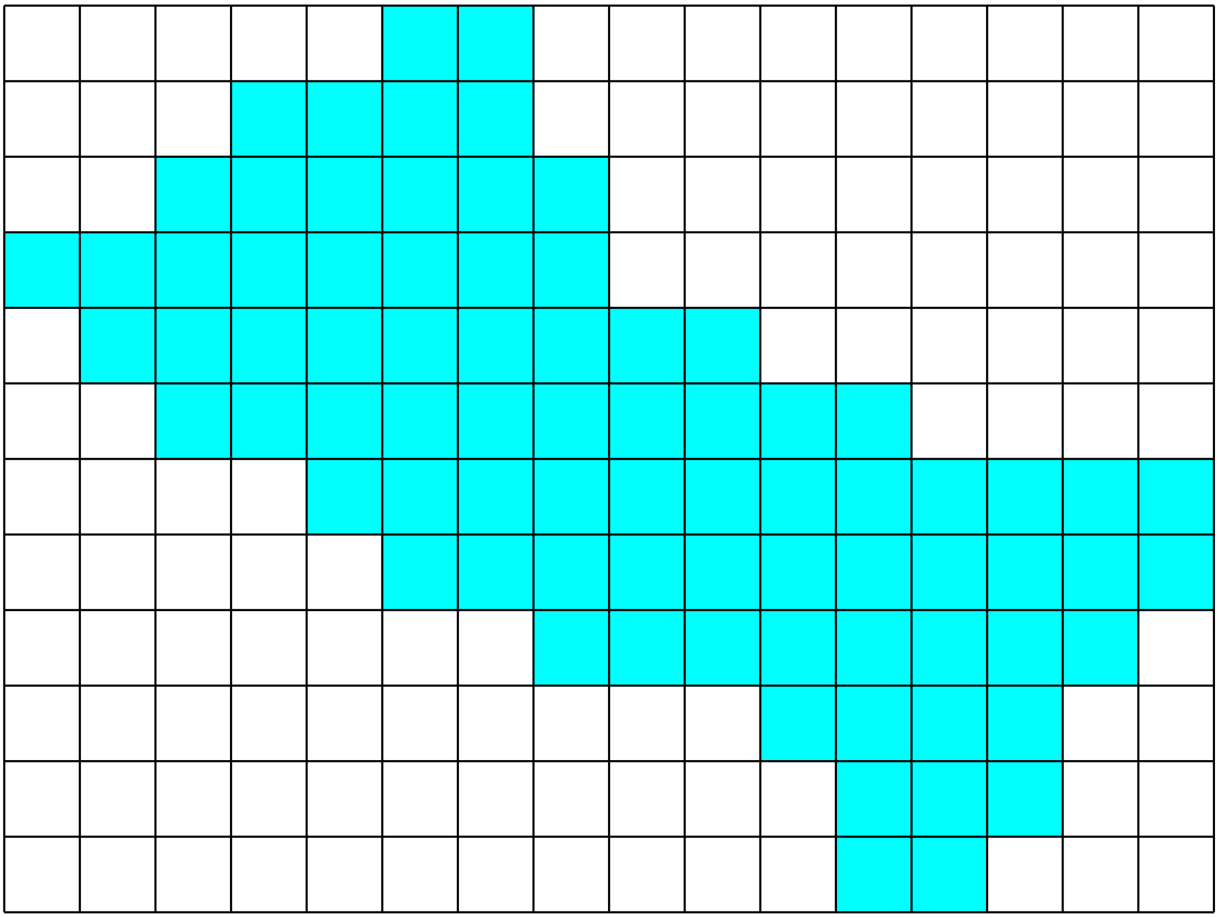
Maciej Gębala

SOFSEM'98

Institute of Computer Science
University of Wrocław

The Polyomino

A one-color picture on the grid.



A matrix $A_{m \times n} = [a_{ij}]$ represents the polyomino.

$$a_{ij} = \begin{cases} 1 & \text{if cell [i,j] belongs to the polyomino,} \\ 0 & \text{otherwise.} \end{cases}$$

The Convex Polyomino

The polyomino (represented by matrix A) is a convex polyomino if

(p) the set of 1's in matrix A is connected, and

(h) the set of 1's in every row of A is connected, and

(v) the set of 1's in every column of A is connected.

If a polyomino satisfies only one or two of above properties then the problem of reconstruction is in NP.

If a polyomino satisfies none or all of above properties then the problem of reconstruction is in P.

Projections

H is a horizontal projection of A

$$H(A)(i) = \sum_{j=1}^{n} a_{ij} \qquad i \in \{1, \ldots, m\}$$

V is a vertical projection of A

$$V(A)(j) = \sum_{i=1}^{m} a_{ij} \qquad j \in \{1, \ldots, n\}$$

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	H
1						1	1										2
2				1	1	1	1										4
3			1	1	1	1	1	1									6
4	1	1	1	1	1	1	1	1									8
5		1	1	1	1	1	1	1	1	1							9
6			1	1	1	1	1	1	1	1	1	1					10
7					1	1	1	1	1	1	1	1	1	1	1	1	12
8						1	1	1	1	1	1	1	1	1	1	1	11
9								1	1	1	1	1	1	1	1		8
10											1	1	1	1			4
11												1	1	1			3
12												1	1				2
V	1	2	4	5	6	8	8	7	5	5	5	7	6	5	3	2	

The Problem

Given two vectors

$$\begin{aligned} H = & \quad (h_1, h_2, \ldots, h_m) \quad \in \{1, \ldots, n\}^m \\ V = & \quad (v_1, v_2, \ldots, v_n) \quad \in \{1, \ldots, m\}^n \end{aligned}$$

decide whether there is at least one convex polyomino whose horizontal projection is described by H and whose vertical projection is described by V.

Necessary condition

$$\sum_{i=1}^{m} h_i = \sum_{j=1}^{n} v_j$$

The Algorithm

1) Choosing a initial position of some 1's.

2) Performing a filling procedure.

- Barcucci, Del Lungo, Nivat, Pinzani (1996)
  The filling procedure that costs $O(n^2m^2)$ starts for $O(n^2m^2)$ initial positions (all possible positions of 1's on four edges of matrix).

  Global cost: $O(n^4m^4)$

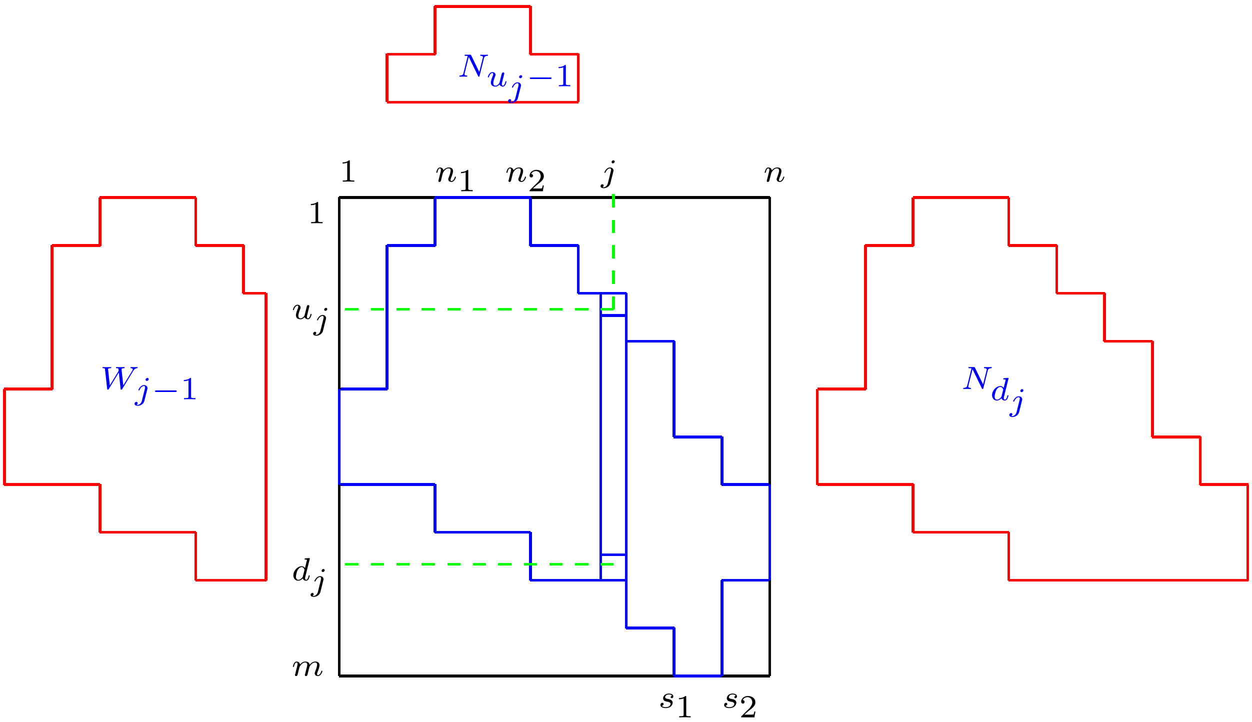
- Our result
  The filling procedure that costs $O(nm \log nm)$ starts for $O(\min(n,m)^2)$ initial positions (positions of 1's in the first and the last row and on the center of matrix).

  Global cost: $O(\min(n,m)^2 \cdot nm \log nm)$

Some Properties

- Let $v_i < m$ for $i \in \{1, \ldots, n\}$.
- Let 1's in the first row be on the left of 1's in the last row.

$$N_{u_j-1} \subset W_{j-1} \subset N_{d_j}$$



$$\sum_{k=1}^{u_j-1} h_k < \sum_{k=1}^{j-1} v_k < \sum_{k=1}^{d_j} v_k$$

Initial Positions

- Choose positions of 1's in the first and the last row.
- Estimate positions of 1's in the columns between the columns containing 1's from the first or the last row. Put 1's between U_j and D_j.

$$D_j = \min\{i \in [1..m-1] : \sum_{k=1}^{i} h_k > \sum_{k=1}^{j} v_k\},$$

$$U_j = \max\{i \in [2..m] : \sum_{k=1}^{i-1} h_k < \sum_{k=1}^{j-1} v_k\}.$$

- Properties

$$U_j \leq D_j$$

$$U_{j+1} \leq D_j + 1$$

The Filling Procedure

Alternating rows and columns perform operations $\oplus, \ominus, \otimes, \odot$ until a fixed point is reached.

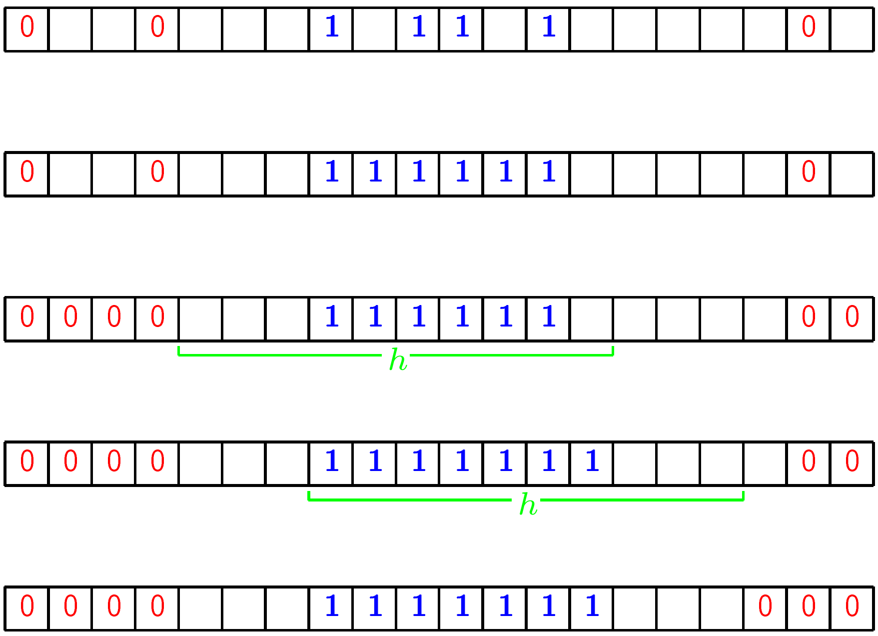
$h = 10$
The number of 1's

Operation $\oplus$
Connect the set of 1's

Operation $\ominus$
Connect sets of 0's

Operation $\otimes$
Extend the set of 1's

Operation $\odot$
Extend sets of 0's



After these operations the number of empty cells in a row or a column is equal twice number of missing 1's.

The Result of Procedure

Procedure return

- Fail if there is no convex polyomino with the fixed initial positions, or
- A convex polyomino which satisfies projection (H,V), or
- A convex polyomino with empty cycles (reduction to 2SAT problem)

0	0	0	0	0	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	$c1$	1	1	$c18$	0	0	0	0	0	0	0	0	0	0	0	0
0	0	0	$c13$	$c7$	1	1	1	1	$c12$	$c6$	0	0	0	0	0	0	0	0	0	0
0	$a1$	$b1$	1	1	1	1	1	1	1	$a6$	$b6$	0	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	0	0	0	0	0
0	$a2$	$b2$	$c14$	1	1	1	1	1	1	1	1	1	1	1	$a3$	$b3$	$c15$	0	0	0
0	0	0	0	$c8$	$c2$	1	1	1	1	1	1	1	1	1	1	1	1	$c9$	$c3$	0
0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	$c17$	$c11$	$c5$	1	1	1	1	1	1	$c16$	$c10$	$c4$	0
0	0	0	0	0	0	0	0	0	0	$a5$	$b5$	1	1	1	$a4$	$b4$	0	0	0	0
0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0