Available online at www.sciencedirect.com
SCIENCE DIRECT®
Theoretical Computer Science 347 (2005) 370–392
Theoretical Computer Science
www.elsevier.com/locate/tcs

An introduction to periodical discrete sets from a tomographical perspective

Andrea Frosini[a,*], Maurice Nivat[b], Laurent Vuillon[c]

[a]*Dipartimento di Scienze Matematiche ed Informatiche "Roberto Magari" Università degli Studi di Siena, Via del Capitano 15, 53100, Siena, Italy*
[b]*Laboratoire d'Informatique, Algorithmique, Fondements et Applications (LIAFA), Université Denis Diderot 2, place Jussieu 75251 Paris Cedex 05, France*
[c]*Laboratoire de Mathématiques, Université de Savoie, CNRS UMR 5127, 73376 Le Bourget du Lac, France*

Accepted 9 June 2005

Communicated by E. Pergola

Abstract

In this paper we introduce a new class of binary matrices whose entries show periodical configurations, and we furnish a first approach to their analysis from a tomographical point of view. In particular we propose a polynomial-time algorithm for reconstructing matrices with a special periodical behavior from their horizontal and vertical projections. We succeeded in our aim by using a reduction involving polyominoes which can be characterized by means of $2-SAT$ formulas.

Keywords: Discrete tomography; Computational complexity; Polyomino; $2-SAT$ reduction

1. Introduction

The present paper studies the possibility of determining some geometrical aspects of a discrete physical structure whose interior is accessible only through a small number of measurements of the atoms lying along a fixed set of directions. This is the central theme of *discrete tomography* and the principal motivation of this study is in the attempt to reconstruct

* Corresponding author.
E-mail addresses: frosini@unisi.it (A. Frosini), maurice.nivat@liafa.jussieu.fr (M. Nivat), laurent.vuillon@univ-savoie.fr (L. Vuillon).

0304-3975/$ - see front matter
doi:10.1016/j.tcs.2005.06.035

three-dimensional crystals from two-dimensional images taken by a transmission electron microscope. The quantitative analysis of these images can be used to determine the number of atoms lying in atomic lines along certain directions [14]. The question is to deduce the local atomic structure of the crystal from the atomic line count data. The goal is to use the reconstruction technique for quality control in VLSI (Very Large Scale Integration) technology. Before showing the results of this paper, we give a brief survey of the relevant contributions in discrete tomography.

Clearly, the best known and most important part of the general area of tomography is *computerized tomography*, an invaluable tool in medical diagnosis and many other areas including biology, chemistry and material science. Computerized tomography is the process of obtaining the density distribution within a physical structure from multiple X-rays. More formally, we attempt to reconstruct a density function $f(x)$ for x in $\mathbb{R}^2$ or $\mathbb{R}^3$, from the knowledge of its line integrals $X_f(L) = \int_L f(x)\,\mathrm{d}x$ for each line L through the space. A line integral is the *X-ray* of $f(x)$ along L. The mapping $f \to X_f$ is known as the *Radon transform*. The mathematics of computerized tomography is quite well understood. Appropriate quadratures [18] of the Radon inversion formula are used, with concepts from calculus and continuous mathematics playing the main role.

Discrete tomography is the area of computerized tomography which deals with discrete physical structures. These structures are usually homogeneous or present a small number of density values. Furthermore, there are strong technical reasons why very few X-rays can be sent through them. Discrete tomography has its own mathematical theory mostly based on discrete mathematics. It has some strong connection with combinatorics and geometry. We wish to point out that the mathematical techniques developed in discrete tomography have applications in other fields such as: image processing, statistical data security, biplane angiography, graph theory and so on. As a survey of the state of the art of discrete tomography we can suggest the book [13].

Interestingly, mathematicians have been concerned with abstract formulations of these problems before the emergence of the practical applications. Many problems of discrete tomography were first discussed as combinatorial problems during the late 1950s and early 1960s. In 1957 Ryser [17] and Gale [11] gave a necessary and sufficient condition for a pair of vectors being the discrete X-rays of an homogeneous planar physical structure, represented by a binary matrix, along the horizontal and vertical directions. The discrete X-rays in horizontal and vertical directions are equal to the *row* and *column sums* of the matrix. They gave an exact combinatorial characterization of the row and column sums that correspond to a binary matrix, and they derived an $\mathrm{O}(nm)$-time algorithm for reconstructing a matrix, with n and m denoting its sizes. We refer the reader to an excellent survey on binary matrices with given row and column sums by Brualdi [5].

In most practical applications we can use some a priori information about geometrical aspects of the image that we want to reconstruct, in order to guide the reconstruction process to a more accurate output. We can think to these a priori information in terms of subclasses of binary images to which the solution must belong. For instance, several papers study the reconstruction problem of binary images having convexity or connectivity properties, in particular there is a uniqueness result [12] for the subclass of *convex binary matrices*, (i.e. finite subsets of $\mathbb{Z}^n$ which are coincident with their convex hull). It is proved that a convex binary matrix is uniquely determined by its discrete X-rays in certain prescribed sets of four directions or in any seven non-parallel coplanar directions. Moreover, there are efficient algorithms for reconstructing binary matrices belonging to classes of subsets of $\mathbb{Z}^2$ characterized by means of convexity or connectivity properties, from their discrete X-rays. In particular we refer to the class of *hv-convex polyominoes* [4,9,3] (i.e., two-dimensional binary matrices which are 4-connected and convex in the horizontal and vertical directions) and to the class of convex binary matrices [6,7].

In this paper, we propose some new classes of binary matrices showing periodicity properties. The periodicity is a natural constraint, and it has not yet been studied in discrete tomography. We provide a polynomial-time algorithm for reconstructing $(1, q)$ periodical binary matrices whose horizontal and vertical projections, i.e. row and column sums, are “not too far" (in a sense explained later) from two given integer sequences. The reconstruction becomes exact when the periodicity is $(1, 1)$. The basic idea of the algorithm is to determine a polynomial transformation of our reconstruction problem to 2-Satisfiability problem which can be solved in linear time [2]. A similar idea has been described and successfully applied in [4,8]. We wish to point out that this paper is only an initial approach to the problem of reconstructing binary matrices having periodicity properties from a small number of discrete X-rays. There are many open problems on these classes of binary matrices of interest to researchers in discrete tomography and related fields: the problem of uniqueness, the problem of reconstruction from three or more X-rays, the problem of reconstructing binary matrices having convexity and periodicity properties, and so on.

2. Definitions and preliminaries

Let A be a $m \times n$ binary matrix, we choose to enumerate its rows and columns starting from row 1 and column 1 which intersect in its upper left position. For each $1 \leqslant i \leqslant m$ and $1 \leqslant j \leqslant n$, let $r_i = \sum_{j=1}^{n} a_{i,j}$ and $c_j = \sum_{i=1}^{m} a_{i,j}$. We define $R = (r_1, \ldots, r_m)$ and $C = (c_1, \ldots, c_n)$ as the vectors of *horizontal* and *vertical projections* of A, respectively. Matrix A is said to be *consistent* with R and C.

Let $0 < p < n$ and $0 < q < m$ be two integers. Matrix A is (p, q) *periodical* or, equivalently, has *period* (p, q), if it holds:

$$a_{i,j} = 1 \Rightarrow \begin{cases} a_{i+q,j+p} = 1 & \text{if } 1 \leqslant i+q \leqslant m \text{ and } 1 \leqslant j+p \leqslant n, \\ a_{i-q,j-p} = 1 & \text{if } 1 \leqslant i-q \leqslant m \text{ and } 1 \leqslant j-p \leqslant n, \end{cases}$$

for each $1 \leqslant i \leqslant m$ and $1 \leqslant j \leqslant n$ (see Fig. 1). In the sequel we indicate with $Per(p, q)$ the class of all binary matrices having period (p, q).

Remark. Since, by definition, a generic matrix A belongs to all classes $Per(p, q)$, with $p \geqslant n$ or $q \geqslant m$, then, in order to avoid non significative cases, we restrict to $p < n$ and $q < m$.

Let $mod_{[1..n]} : \mathbb{N} \to \mathbb{N}$ be the function

$$(x)mod_{[1..n]} = \begin{cases} (x)mod_n & \text{if } (x)mod_n \neq 0 \\ n & \text{otherwise,} \end{cases}$$

where $(x)mod_n$ is the usual modulo function.

$$\begin{array}{c|cccccccc}
 & 3 & 2 & 2 & 1 & 1 & 2 & 1 & 4 \\
\hline
3 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\
2 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
3 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 1 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
3 & 0 & 1 & 1 & 0 & 0 & 1 & 0 & 0
\end{array}$$

Fig. 1. A binary matrix having period (2, 3). The integers at the beginning of each row and column correspond to its horizontal and vertical projections, respectively. The circled entry 1 is linked, by periodicity, with the two pointed ones.

$$\begin{array}{cccccccc}
0 & 1 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} \\
0 & 0 & 0 & \mathbf{1} & 0 & 1 & 0 & 0 \\
\mathbf{0} & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
1 & 0 & 0 & 0 & \mathbf{1} & 0 & 1 & 0 \\
1 & \mathbf{0} & 0 & 1 & 0 & 0 & 0 & 0
\end{array}
\qquad
\begin{array}{cccccccc}
0 & 1 & 0 & 0 & 0 & 0 & \mathbf{1} & 1 \\
0 & 0 & 0 & 1 & 0 & \mathbf{1} & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & \mathbf{1} \\
1 & 0 & 0 & 0 & 1 & 0 & \mathbf{1} & 0 \\
\mathbf{1} & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{array}
\qquad
\begin{array}{cccccccc}
0 & 1 & 0 & 0 & 0 & 0 & 1 & \mathbf{1} \\
0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 \\
\mathbf{1} & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0
\end{array}$$

(a) (b) (c)

Fig. 2. Three copies of the same (1, 2) periodical matrix.

The concept of periodicity hides the following notion of *propagation* of a value inside a matrix: for any given position (i, j) of $A \in Per(p, q)$, we define *set of propagation* $P_{i,j}$ to be the set of all positions $(i + kq, t)$ such that

$$t = (j + kp)\ mod_{[1..n]} \quad \text{with } k \in \mathbb{Z} \text{ and } 1 \leqslant i + kq \leqslant m.$$

Finally, we define *line* to be each subset $\ell_{i,j}$ of elements of A such that
- $a_{i',j'} \in \ell_{i,j}$ if and only if $(i', j') \in P_{i,j}$;
- $a_{i',j'} \in \ell_{i,j}$ implies $a_{i',j'} = 1$;

and we define *length* of $\ell_{i,j}$ to be its cardinality. In words, line is each set of elements of A having value 1 and whose positions form a propagation set.

Each line $\ell_{i,j}$ has a *starting point* [*ending point*] which is the element $a_{i',j'} \in \ell_{i,j}$ such that, for each $a_{i'',j''} \in \ell_{i,j}$, it holds $i' \leqslant i''$ [$i' \geqslant i''$]. Furthermore we say that $\ell_{i,j}$ *starts* [*ends*] on column j'. In Fig. 2, three copies of the same (1, 2) periodical matrix are depicted: the highlighted entries
- of matrix (a) correspond to the elements whose positions belong to the two propagation sets $P_{3,1}$ and $P_{2,4}$;
- of matrix (b) correspond to the lines $\ell_{2,6}$ and $\ell_{5,1}$ of lengths two and three, respectively;
- of matrix (c) correspond to the two elements $a_{4,1}$ and $a_{1,8}$ having value 1 and not belonging to any line.

$$
\begin{array}{c|cccccccccc}
 & 2 & 2 & 2 & 2 & 2 & 1 & 1 & 3 & 3 & 3 \\
\hline
2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\
4 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\
2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
5 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 & 0 \\
2 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
5 & 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\
1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0
\end{array}
$$

Fig. 3. A (1, 2) periodical matrix whose highlighted positions form eight connected boxes.

The following notion of *box* has maximal relevance in our framework. Let A be a (p, q) periodical matrix: for each row i of A, the two sets of positions

$$(i, 1), \ldots, (i, p) \quad \text{and} \quad (i, n-p+1), \ldots, (i, n)$$

are called (the ith) *left* and *right box* of A, respectively. In the same way we can define, for each column j of A, the positions

$$(1, j), \ldots, (q, j) \quad \text{and} \quad (m-q+1, j), \ldots, (m, j)$$

to form (the jth) *upper* and *lower box* of A, respectively. As a direct consequence of the definition of boxes, it holds:

Proposition 1. *Let us indicate with b_i and b_{i+q} the sums of the elements of the ith right box and of the $(i+q)$th left box of A, respectively, and, analogously, let us indicate with b_j and b_{j+p} the sums of the elements of the jth lower box and of the $(j+p)$th upper box of A, respectively. We have:*
(a) *if $r_i + k = r_{i+q}$, with $k \geqslant 0$, then it holds $b_{i+q} - b_i = k$, else, if $k < 0$, then it holds $b_i - b_{i+q} = k$;*
(b) *if $c_j + k = c_{j+p}$, with $k \geqslant 0$, then it holds $b_{j+p} - b_j = k$, else, if $k < 0$, then it holds $b_j - b_{j+p} = k$.*

In Fig. 3, a (1, 2) periodical matrix is depicted: the highlighted positions form eight boxes which are grouped two by two. The difference between the sums of the elements inside each box of the same couple is different from 0, and can be computed from the horizontal and vertical projections of the matrix, as stated in Proposition 1.

Formalization of the main problems. The given definitions allow us to specify, inside our framework, some relevant problems of discrete tomography:

RECONSTRUCTION($Per(p, q)$, (R, C))

Instance: two vectors $R \in \mathbb{N}^m$ and $C \in \mathbb{N}^n$.

Output: an element of $Per(p, q)$, if it exists, having R and C as vectors of horizontal and vertical projections, respectively.

This problem requires to construct an element of $Per(p, q)$ which is consistent with two given horizontal and vertical projections. Such a task can be easily fulfilled by using a procedure which generates all the elements of $Per(p, q)$ of dimension $m \times n$ and, for each of them, checks its consistency with R and C. This elementary procedure, however, requires an amount of time which grows exponentially with the dimensions of R and C. In the sequel, we will focus our attention on its following variant:

REC-STRIP($Per(p, q), (R, C)$)

Instance: two vectors $R \in \mathbb{N}^m$ and $C \in \mathbb{N}^n$.

Output: an element A of $Per(p, q)$, if it exists, having C as vertical projections and such that

$$\sum_{i=kq+1}^{kq+t} r_i = \sum_{i=kq+1}^{kq+t} \sum_{j=1}^{n} a_{i,j} \quad \text{and} \quad \sum_{i=kq+t+1}^{(k+1)q} r_i = \sum_{i=kq+t+1}^{(k+1)q} \sum_{j=1}^{n} a_{i,j},$$

for each possible integer $k \geqslant 0$, and such that $t = (m)mod_q$.

In other words, we search for a (p, q) periodical matrix A consistent with C, and such that its horizontal projections are not considered one by one, but they grouped and summed up into alternate *strips* of height $(m)mod_q$ and $q - (m)mod_q$.

In this paper, we will define a procedure to solve REC-STRIP($Per(1, q), (R, C)$) in polynomial time.

A small remark is needed: the reconstruction of a $(0, q)$ periodical matrix from R and C is far from being a trivial problem. We choose to skip this case, at least for the moment, although it might seem a more natural starting point, since we are attracted by the connection between the reconstruction of $(1, q)$ periodical matrices and the reconstruction of horizontally and vertically convex discrete sets on a torus (starting in both cases from the horizontal and vertical projections). For, this connection, too, is non-trivial, as indicated in the next paragraph.

UNIQUENESS($Per(p, q)$)

Instance: an element $A \in Per(p, q)$.

Question: does there exist an element $A' \in Per(p, q)$, different from A, such that A and A' have the same horizontal and vertical projections?

The study of the conditions which assure the uniqueness of a matrix consistent with a given set of projections, usually starts from an analysis of its switching components with respect to the directions of projections (see [13] for details and examples). In the sequel we point out simple remarks about the uniqueness of the elements of the class $Per(1, 1)$. A deepest analysis of this problem together with the formalization of a switching theory for the whole class $Per(p, q)$ furnish material for future work. From a practical point of view, uniqueness is a crucial property when linked to an easy algorithm of reconstruction since the projections of the object can be used to efficiently characterize (and so encode) the object itself.

3. A general strategy for reconstructing periodical matrices

In this section we propose a general strategy for solving the above defined problem reconstruction problems, and then we successfully apply it to the class $Per(1, 1)$.

$$
A:\quad
\begin{array}{ccccccccccc}
1 & 0 & & & & & 0 & \mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{1} \\
0 & 0 & & & & & 0 & \mathbf{0} & \mathbf{1} & \mathbf{1} & \mathbf{1} \\
1 & 0 & 0 & & & & 0 & 0 & 1 & \mathbf{0} & \mathbf{0} \\
\mathbf{0} & \mathbf{1} & 1 & 0 & & & 0 & 1 & 0 & \mathbf{0} & \mathbf{1} \\
\mathbf{0} & \mathbf{1} & 0 & 0 & 0 & \dots & & 0 & 0 & \mathbf{0} & \mathbf{1} \\
\mathbf{1} & \mathbf{1} & 1 & 0 & 0 & & & & 0 & 0 & 1 \\
\mathbf{0} & \mathbf{1} & \mathbf{0} & \mathbf{1} & 1 & & & & & 1 & 0 \\
\mathbf{1} & \mathbf{0} & \mathbf{0} & \mathbf{1} & 0 & & & & & 0 & 0
\end{array}
$$

Fig. 4. The two zones of the $(2, 3)$ periodical matrix A where the entries not belonging to any line can lie.

We observe that the presence inside a periodical matrix of elements which do not belong to any line produces perturbations in its projections which partially reveal when examining its boxes. The knowledge of these elements become exact when the wideness of the boxes reduces to a single position, i.e. when one or both the components of the vector of periodicity have value 1. Furthermore, boxes provide useful information about the location of (the starting points of) the lines inside a periodical matrix, whose total number and lengths can be easily inferred from the projections. Different strategies which depend on the subclass of $Per(p, q)$ we are dealing with, merge all these information in order to successfully complete the reconstruction task. Our general approach to these reconstruction problems rely on two steps: a *Preprocessing* and a *Lines reconstruction*.

Preprocessing: it is created a partial solution to the reconstruction problem, i.e. a $m \times n$ matrix whose elements having value 1 are those which do not belong to any line of the final solution. These elements can be partially detected by computing the left and right boxes of the solution, and they lie in the union of two zones which comprehend two opposite corners of the matrix and whose extensions depends on the vector of periodicity (the highlighted entries in Fig. 4).

In these zones, and only here, the elements of the solution may not completely show the periodical behavior which characterizes the structure.

If we focus our attention on matrices which belong to $Per(1, q)$, we notice that their left and right boxes are composed by a single element. In such a case, the preprocessing can be used to reconstruct a set of elements which are fixed, i.e. which are common to all the solutions satisfying the given projections R and C.

Lines reconstruction: suitable positions for the lines which belong to the final solution are now detected. It is created a $m \times n$ matrix whose elements having value 1 form the lines of the final solution, and it is merged with that reconstructed in the preprocessing. In a word, it is now that the periodical behavior of the structure realizes. Different reconstruction strategies can be defined according to the different class of periodical structures we are dealing with: some specific properties of the solution, in fact, could greatly simplify this part of the reconstruction.

In the sequel, we present a reconstruction algorithm for the class $Per(1, 1)$ which is useful to better understand the more complex result involving the class $Per(1, q)$ described in next section. The simplicity of this example, allows both the preprocessing and the lines reconstruction to directly act and modify the final solution A.

The reconstruction of $Per(1,1)$ from two projections

Let I be an instance of RECONSTRUCTION($Per(1,1)$, (R, C)). We create a $m \times n$ matrix A and we initialize its entries to the *blank* value. Procedure 1 performs the preprocessing part of the reconstruction after observing that both the left and right boxes of A are composed by a single cell, and after checking that the differences between consecutive entries of R belong to $\{-1, 0, 1\}$ (if this assumption is not satisfied, then I has no solution). Two vectors R' and C', which are initialized to the values of R and C, respectively, support the computation. In particular, they are used to store, step by step, the horizontal and vertical projections of the entries 1 not yet placed in A.

Procedure 1. *Preprocessing*

```
for i = 1 to m − 1 do
                { Comment: search for an entry 1 in a left box}
  if R'[i] + 1 == R'[i + 1] then
    for j = 1 to min {m − i, n} do
      A[i + j][j] = 1; R'[i + j] = R'[i + j] − 1; C'[j] = C'[j] − 1;
    end for
  end if
                { Comment: search for an entry 1 in a right box}
  if R'[i] == R'[i + 1] + 1 then
    for j = 1 to min {n, i} do
      A[i − j + 1][n + 1 − j] = 1; R'[i − j + 1] = R'[i − j + 1] − 1;
      C'[n + 1 − j] = C'[n + 1 − j] − 1;
    end for
    i = max{i − n − 1, 0};
  end if
end for
```

From Procedure 1, one can easy check the theorem:

Theorem 2. *After performing Procedure* 1:
(a) *the elements of A having value* 1, *are common to all the solutions of instance I*;
(b) *the partial solution A does not contain any line*;
(c) *the vector R' is homogeneous*, *i.e. all its entries have the same value*.

Another simple and useful property is

Proposition 3. *After performing Procedure* 1, *if there exists $A' \in Per(1,1)$ which is consistent with R' and C', then A' is composed only by lines*.

Proof. By Theorem 2, for each $1 \leqslant i_0 < m$, it holds $r'_{i_0} = r'_{i_0+1}$, with $r'_{i_0} = \sum_{j=1}^{n} a'_{i_0,j}$ and $r'_{i_0+1} = \sum_{j=1}^{n} a'_{i_0+1,j}$.

Since by periodicity, $a'_{i_0,j} = a'_{i_0+1,j+1}$, for $1 \leqslant j < n$, then it holds that $a'_{i_0,n} = a'_{i_0+1,1}$, and the thesis is achieved. $\square$

Remark. This result allows us to map matrix A' on a cylinder (i.e. we can consider its first and last column as contiguous ones) without loosing its periodical behavior.

The part of the algorithm where lines are reconstructed is split into two procedures: a first one which places in A the lines whose positions are common to all solutions of I ($Line - rec$ procedure), and a second one which places the remaining ones, if any ($Loop - rec$ procedure). Since the procedure $Line - rec$ is very similar to the procedure *Preprocessing*, we give a brief description of it:

Procedure 2. $Line - rec$

Step 1: compute the upper boxes of A using vector C' and, for each of them containing an entry 1, place in A a line whose starting point is the element inside the box. If this line intersects a previously placed entry, then return FAILURE. Update C' and R'.

Step 2: compute the lower boxes of the solution using vector C' and, for each of them containing an entry 1, place in A a line whose ending point is the element inside the box. If this line intersects a previously placed entry, then return FAILURE. Update C' and R'.

Step 3: repeat Step 1 and Step 2 till no upper and lower boxes are detected.

From the definition of Procedure 2, it is straightforward that

Theorem 4. *After performing Procedures* 1 *and* 2:
(a) *the entries of A are common to all the solutions of the instance I*;
(b) *both R' and C' are homogeneous.*

We want to stress the following uniqueness result:

Corollary 5. *After performing Procedures* 1 *and* 2, *if all the elements of the vector R' have value* 0, *then the reconstructed solution A of I is unique*.

The last part of our reconstruction process needs one more definition: let us consider a generical $m \times n$ matrix $B \in Per(1, 1)$ as lying on a torus (of its same dimension), i.e. we consider its last and first row as to be consecutive, and the same for its last and first column. A sequence $\ell_1, \ldots, \ell_k$ of lines of B is called a *loop* if it constitutes a class modulo (1, 1) on the torus. In other words, for each $1 \leqslant i \leqslant k$, the ending column of ℓ_i and the starting column of $\ell_{(i+1) mod_{[1..k]}}$ are consecutive in B (an example of loop are the highlighted entries in Fig. 5(c)). Simple calculations show that the length of a loop is $l.c.m.\{m, n\}$ (the least common multiple of m and n).

Property 3.1. *Let us assume that both the vectors of the horizontal and vertical projections of B are homogeneous. The following statements need a simple check*:
(a) *B is composed only by loops*;
(b) *the number of loops of B is $\frac{m\,r}{l.c.m.\{m,n\}} = \frac{n\,c}{l.c.m.\{m,n\}}$, where r (resp. c) is the common value of its horizontal (resp. vertical) projection*;

$$
\begin{array}{ccccccccc}
0&0&0&0&0&0&0&\mathbf{1}&0\\
0&0&0&0&0&0&0&0&\mathbf{1}\\
0&0&0&0&0&0&0&0&0\\
\mathbf{1}&0&0&0&0&0&0&0&0\\
0&\mathbf{1}&0&0&0&0&0&0&0\\
\mathbf{1}&0&\mathbf{1}&0&0&0&0&0&0
\end{array}
\qquad
\begin{array}{ccccccccc}
0&\mathbf{1}&0&0&0&0&0&1&0\\
0&0&\mathbf{1}&0&0&0&0&0&1\\
0&0&0&\mathbf{1}&0&0&0&0&0\\
1&0&0&0&\mathbf{1}&0&0&0&0\\
0&1&0&0&0&\mathbf{1}&0&0&0\\
1&0&1&0&0&0&\mathbf{1}&0&0
\end{array}
\qquad
\begin{array}{ccccccccc}
0&1&\mathbf{1}&0&0&\mathbf{1}&0&1&\mathbf{1}\\
\mathbf{1}&0&1&\mathbf{1}&0&0&\mathbf{1}&0&1\\
0&\mathbf{1}&0&1&\mathbf{1}&0&0&\mathbf{1}&0\\
1&0&\mathbf{1}&0&1&\mathbf{1}&0&0&\mathbf{1}\\
\mathbf{1}&1&0&\mathbf{1}&0&1&\mathbf{1}&0&0\\
1&\mathbf{1}&1&0&\mathbf{1}&0&1&\mathbf{1}&0
\end{array}
$$

(a) (b) (c)

Fig. 5. The three stages of the reconstruction of a (1, 1) periodical matrix.

(c) *if we set* $\alpha = \frac{l.c.m.\{m,n\}}{m}$ *and* $\beta = \frac{l.c.m.\{m,n\}}{n}$, *i.e. the number of cells of each loop lying on any row and on any column of* B, *respectively, then a necessary and sufficient condition for* B *to exist is that* $(r)mod_\alpha = (c)mod_\beta = 0$;

(d) *for any loop* λ, *if* $B[1][j] \in \lambda$, *then* $B[1][(j + \frac{m\,n}{l.c.m.(m,n)})mod_{[1..n]}] \in \lambda$.

Now we are ready to move back to our reconstruction process and complete it by defining the procedure *Loop* − *rec* which scans A searching for free positions where its loops, if any, can be placed, and which bases its correctness on Property 3.1. Again, only a brief description of the procedure is given:

Procedure 3. *Loop* − *rec*

Step 1: mark with the symbol X the elements of the first row of A having value *blank*, and which cannot be starting points of a line, i.e.

for $i = 2$ to m **do**
  if $(A[i][1] == 1)$ & $(A[1][(n-i)mod_{[1..n]}] == blank)$ **then**
    $A[1][(n-i)mod_{[1..n]}] = X$
  end if
end for

Step 2: place a loop inside A such that the starting points of its lines do not intersect any element of value 1 or X, if possible, else return FAILURE;

Step 3: repeat step 2 till all the $\frac{m\,r'}{l.c.m.\{m,n\}}$ loops are placed;

Step 4: the elements having values X and *blank* change their value to 0.
  Return A.

Theorem 6. *The problem* RECONSTRUCTION$(Per(1,1),(R,C))$ *can be solved in polynomial time.*

Proof. Let I be an instance of RECONSTRUCTION$(Per(1,1),(R,C))$. Since *Preprocessing* and *Line* − *rec* reconstruct the entries which are common to all the solutions of I, then if they give FAILURE, the vectors R and C are not consistent. The same result holds if *Loop* − *rec* gives FAILURE, since there are not enough free positions in A for placing the required $\frac{m\,r'}{l.c.m.\{m,n\}}$ loops. Let us analyze the complexity of the three procedures:

Preprocessing: the vector R' of length m is scanned and, for each of its elements, at most $m-1$ entries of A are changed, with a computational complexity of $O(m\;min\{m,n\})$.

Line $-$ *rec*: the vector C' of length n is scanned at most n times and, for each of its elements, at most one line is added to the matrix A, i.e. m of its entries are changed. So the computational complexity is $O(m\,n^2)$.

Loop $-$ *rec*: taking into account Property 3.1, the check for the possible positions of a loop and its placement takes $O(m\,n)$. Finally, the substitution of the values *blank* with 0 takes $O(m\,n)$.

So, an element of $Per(1,1)$ can be reconstructed in $O(m\,n^2)$. $\square$

As a direct consequence of the defined reconstruction strategy, we have the following uniqueness result:

Corollary 7. *Let* $R \in \mathbb{N}^m$ *and* $C \in \mathbb{N}^n$. *If* $g.c.d.\{n, m\} = 1$, *then there is at most one* $(1,1)$ *periodical matrix consistent with R and C.*

The proof can be easily obtained by observing that either the solution has no loops, and so it is completely reconstructed during the preprocessing and the line reconstruction stage, or it contains one single loop, and consequently all its entries have value 1. As an example, if $R \in \mathbb{N}^m$ and $C \in \mathbb{N}^{m+1}$, then at most one solution to RECONSTRUCTION($Per(1,1)$, (R, C)) exists.

Example 8. Let us reconstruct an element of $Per(1,1)$ consistent with

$$R = (5, 5, 4, 5, 5, 6) \quad \text{and} \quad C = (4, 4, 4, 3, 3, 3, 3, 3, 3).$$

Preprocessing detects two left boxes in positions $(4,1)$ and $(6,1)$ and a right box in positions $(2,9)$. For each of them, the matrix A, whose elements are here immediately initialized to the value 0, is filled with entries which guarantee the $(1,1)$ periodicity, as shown in Fig. 5(a). The vectors R' and C' are now updated to $R' = (4,4,4,4,4,4)$ and $C' = (2,3,3,3,3,3,3,2,2)$, with R' homogeneous.

Line-rec scans vector C' and detects an upper box in position $(1,2)$, then it places the correspondent line (Fig. 5(b)). Both the updated vectors $R' = (3,3,3,3,3,3)$ and $C' = (2,2,2,2,2,2,2,2,2)$ are now homogeneous.

Loop $-$ *rec* performs the last stage of the reconstruction. Since $l.c.m.\{6, 9\} = 18$, each loop has $\frac{18}{6} = 3$ cells on every row, and $\frac{18}{9} = 2$ cells on every column. Since the entries of R' and C' are consistent with these two values, the placement of the loop goes on, after marking by X the positions $(1,5)$ and $(1,7)$. The following three sets of positions for the starting points of the lines are considered, and one of them is chosen:

- $S_1 = \{(1,1), (1,7), (1,4)\}$ cannot be chosen since position $(1,7)$ is marked;
- $S_2 = \{(1,2), (1,8), (1,5)\}$ cannot be chosen since positions $(1,2)$ and $(1,8)$ has value 1, and, furthermore, position $(1,5)$ is marked;
- $S_3 = \{(1,3), (1,9), (1,6)\}$ is chosen, and the placement of the loop is finally performed (Fig. 5(c)).

The final solution A is achieved after replacing the entries X with the value 0. Since only one choice is allowed for the placement of the loop, then the final solution is unique.

4. The reconstruction of $Per(1, q)$ with $1 < q < m$

In this section we concentrate on the problem REC-STRIP($Per(1, q)$, (R, C)), and we use the already introduced reconstruction strategy to solve it. We achieve one of its solutions, say A, as the union of two matrices A' and A''; this union is reached at a stage called *fusion*, while the two matrices are respectively obtained in a *preprocessing* stage which is very similar to the one for the class $Per(1, 1)$, and after a complex *line reconstruction* stage, which uses a reconstruction procedure for a special class of convex polyominoes. It is now clear how such a problem can be considered as a natural extension of the special case when m is a multiple of q. In fact, in this special case, the problem turns out to be equivalent to the reconstruction of a matrix which is (1, 1)-periodic, with the further assumption that each element of the matrix contributes to the horizontal projections for a $1/q$ fraction.

Obviously, it holds the following useful.

Remark. Let M be a solution of REC-STRIP. Inside the same strip of M, one can shift up or down an element in the same column, and then shift also the other elements which are linked to it by periodicity, in order to get another solution.

So, let I be an instance of REC-STRIP($Per(1, q)$, (R, C)), and let A', R' and C' be chosen and initialized as in the reconstruction of an element of $Per(1, 1)$ from two projections, defined in the previous section.

Preprocessing. The vector R' is again used to determine the elements which do not belong to any line of the solution A. These points are stored in a matrix A', whose elements are initialized to the value *blank*. The preprocessing for the class $Per(1, q)$ can be performed by using a modified version of Procedure 1, which takes into account the new period $(1, q)$, and which groups together the left [respectively, right] boxes belonging to the same strip, instead of considering each of them separately.

To support the computation, we define the vector $Sum = (s_1, \ldots, s_t)$, which contains the projections of the t strips of A, with $t = 2\lceil \frac{m}{q} \rceil - 1$. More precisely, for each $1 \leqslant i \leqslant t$, if i is odd, then

$$s_i = r'_{(\lceil \frac{i}{2} \rceil - 1)q+1} + \cdots + r'_{(\lfloor \frac{i}{2} \rfloor - 1)q+(m)mod_{[1..q]}},$$

else, if i is even, then

$$s_i = r'_{(\lceil \frac{i}{2} \rceil - 1)q+(m)mod_{[1..q]}+1} + \cdots + r'_{\lfloor \frac{i}{2} \rfloor q},$$

with the further assumption that if q divides m, then all s_i, with i even, are set to 0.

The procedures $Odd - Boxes$ (described below), and $Even - Boxes$ (which one can easily deduce from $Odd - Boxes$. We skip the obvious details) scan the elements of Sum having odd and even indices, respectively, in order to detect the sets of left and right boxes of the final solution A, where all the entries not belonging to any line lie.

We observe that

- in Step 2 of *Preprocessing*, it is required to mark with X the elements of the first q rows of A' which cannot be starting points of lines of A (as in Step 1 of Procedure 3);

Procedure 4. *Preprocessing*

Step 1: **for** $i = 1$ to $t - 2$ **do**
  Step 1.1: **if** i is odd **then** $Odd - Boxes(i)$ **endif**
  Step 1.2: **if** i is even **then** $Even - Boxes(i)$ **endif**
  Step 1.3: complete the matrix A' according with the $(1, q)$ periodicity;
  Step 1.4: update the vectors R', C', and *Sum*;
Step 2: for each element $A'[i][1] = 1$, if $1 \leqslant i' \leqslant q$ and the two positions (i', j') and $(i, 1)$ belong to the same propagation set, then $A'[i'][j'] = X$.

Procedure 5. *Odd − Boxes* (i)

if $s_i + h == s_{i+2}$ & $0 < h \leqslant (m)mod_{[1..q]}$ **then**
  $A'[\lceil \frac{i}{2} \rceil\, q + 1][1] = \cdots = A'[\lceil \frac{i}{2} \rceil\, q + h][1] = 1;$
else if $s_i == s_{i+2} + h$ & $0 < h \leqslant (m)mod_{[1..q]}$ **then**
  $A'[(\lceil \frac{i}{2} \rceil - 1)\, q + 1][n] = \cdots = A'[(\lceil \frac{i}{2} \rceil - 1)\, q + h][n] = 1;$
  $i = max\{i - n - 1, 0\};$
else if $h > (m)mod_{[1..q]}$ **then**
  FAILURE
end if

- in the procedures *Odd − boxes* and *Even − Boxes*, one performs a consistency check for the value of h, i.e. $h \geqslant 0$ and $h \leqslant (m)mod_{[1..q]}$;
- both the procedures *Odd − boxes* and *Even − Boxes* use a greedy strategy to place the detected entries in the left and right boxes of A'.

A result similar to Theorem 2 holds:

Theorem 9. *After performing the preprocessing stage*:
(a) *the matrix A' does not contain any line*;
(b) *for each $1 \leqslant i \leqslant t - 2$, it holds $s_i = s_{i+2}$.*
(c) *each solution of instance I has $s_1 + s_2$ lines at most.*

Line reconstruction. The reconstruction of the matrix A'' which contains exactly all the lines of A, and which is one of the solutions of REC-STRIP($Per(1, q)$, (R', C')) will be held in the three steps hereafter summarized:

Step 1: the instance I' of REC-STRIP($Per(1, q)$, (R', C')) is transformed into an instance I'' of the problem of reconstructing an horizontal and vertical convex discrete structure M lying on a torus from its vertical projections C', and from the partial knowledge of its horizontal projections. We call such a problem RECCPT(C', L, n_L, n_{L+1}, k), where the parameters L, n_L, n_{L+1}, k are computed from I';

Step 2: the instance I'' is characterized by means of a Boolean formula Ω belonging to 2-SAT, and it is solved in polynomial time by using standard techniques (see [2]);

Step 3: using the found solution of I'', we finally compute a solution of I'.

Step 1 (*Line reconstruction*): where the reconstruction of the matrix A'' reduces to an instance of the problem RECCPT(C', L, n_L, n_{L+1}, k).

Before introducing the problem RECCPT(C', L, n_L, n_{L+1}, k), we point out some properties of the matrix A'' and we define the parameters L, n_L, n_{L+1} and k:

Property 4.1. *In the matrix* A''
(a) *each entry 1 belongs to a line, and each line has length* L *or* $L+1$, *where* $L = \lfloor \frac{m}{q} \rfloor$;
(b) *the maximum number of lines of length* $L+1$ *which can start in the same column is* $n_{L+1} = (m) mod_q$, *while the maximum number of lines of length* L *which can start in the same column is* $n_L = q - n_{L+1}$;
(c) *the total number of lines is* $k = s_1 + s_2$, *where* s_1 *and* s_2 *are the numbers of lines of length* $L+1$ *and* L, *respectively.*

From statement (a) of Property 4.1, it follows that A'' maintains the $(1, q)$ periodicity when mapped on a cylinder, i.e. when its first and last columns are considered consecutive, and, consequently, that A'' is completely determined by the values of its first q rows. We can order the lines of A'' by ordering their starting points: let (i, j) and (i', j') be the starting points of two lines:
if $j < j'$ then $(i, j) < (i', j')$;
if $j > j'$ then $(i, j) > (i', j')$;
if $j = j'$ and $i < i'$ then $(i, j) > (i', j')$;
if $j = j'$ and $i > i'$ then $(i, j) < (i', j')$, else $(i, j) = (i', j')$.

Roughly speaking, we order the starting points of the lines of A'' from left to right, and from bottom to up.

Now we define the problem RECCPT(C', L, n_L, n_{L+1}, k), and we prove its equivalence with REC-STRIP($Per(1, q), (R', C')$): let us consider a torus having a squared surface of dimension $k \times n$ (k rows and n columns), and let us indicate with $\mathcal{T}_{h,v}$ the class of all its subsets which are horizontally and vertically convex, i.e. such that the cells of a generic element $M \in \mathcal{T}_{h,v}$ which lie on the same row or column form a single bar. We choose to represent M with a binary matrix (see matrix M in Example 10), and we define the problem of the Reconstruction of a Convex Polyomino on a Torus from partial projections:

RECCPT(C', L, n_L, n_{L+1}, k)

Instance: a vector $C' \in \mathbb{N}^n$ and four integers L, n_L, n_{L+1}, and k.

Output: a $k \times n$ matrix $M \in \mathcal{T}_{h,v}$, if it exists, such that:
- C' is the vector of its vertical projections;
- its horizontal projections have value L or $L+1$;
- on each column of M, at most n_L bars of length L, and n_{L+1} bars of length $L+1$ can start.

REC-STRIP($Per(1, q), (R', C')$) and RECCPT(C', L, n_L, n_{L+1}, k) are proved to be equivalent by defining a procedure which maps a solution B_0 of the instance I' of the first problem into a solution M of the instance I'' of the second one, and vice versa, mapping back M into a matrix B_1, in general different from B_0, which is again a solution of I'. The parameter k is the number of lines both in B_0 and in B_1, i.e. $k = s_1 + s_2$. In our reconstruction process, we will identify B_1 with the matrix A''.

So, let us start from B_0, and construct a matrix M of dimension $k \times n$, representing a convex set on a torus, as follows: for each $1 \leqslant i \leqslant k$, the ith row of M is composed by

$$
B_0:\ \begin{array}{ccccccc}
0&1&0&1&0&0&1\\
0&1&0&0&0&1&0\\
0&1&0&0&0&0&0\\
0&0&1&0&0&0&0\\
1&0&1&0&1&0&0\\
0&0&1&0&0&0&1\\
0&0&1&0&0&0&0\\
0&0&0&1&0&0&0\\
0&1&0&1&0&1&0
\end{array}
\qquad
M:\ \begin{array}{ccccccc}
0&1&1&0&0&0&0\\
0&1&1&0&0&0&0\\
0&1&1&1&0&0&0\\
0&0&1&1&0&0&0\\
0&0&0&1&1&1&0\\
0&0&0&0&0&1&1\\
1&1&0&0&0&0&1
\end{array}
\qquad
B_1:\ \begin{array}{ccccccc}
0&1&0&1&0&0&1\\
0&0&0&0&0&0&0\\
0&1&0&0&0&0&0\\
0&1&1&0&0&1&0\\
1&0&1&0&1&0&0\\
0&0&0&0&0&0&0\\
0&0&1&0&0&0&0\\
0&0&1&1&0&0&1\\
0&1&0&1&0&1&0
\end{array}
$$

Fig. 6. The equivalence between Rec-Strip and RecCPT.

a sequence of consecutive entries 1 which starts and ends in the same columns as ℓ_i, the ith line of B_0 with respect to the order above defined (see Example 10).

The obtained matrix M is a convex set on a torus, since, when moving on each column of M from up to bottom, the order from bottom to up defined on the lines of each column of B_0 allows the starting bars of length L to be encountered before those of length $L+1$. Furthermore, it is clear that M is a solution of I''.

On the contrary, given a solution M of I'', we construct a $m \times n$ matrix B_1, with $m = L\, n_L + (L+1)\, n_{L+1}$, as follows: for each row i of M, if there exists a bar of length L [respectively, $L+1$] lying on it, then we place in B_1 a line of length L [respectively, $L+1$], which is its ith one, and which starts in column j. The placement of the starting points of these lines can be performed with a greedy technique, since the problem Rec-Strip$(Per(1,q), (R', C'))$ requires to relax the constraints on the horizontal projections of B_1 imposed by R'. For this reason, the horizontal projections of B_1 *can differ* from those of B_0.

However, it can be easily checked that B_1 is a solution of instance I'. The following example tries to clarify the equivalence:

Example 10. Let us consider the $(1,4)$ periodical matrix B_0 in Fig. 6 of dimension 9×7 consistent with $R' = (3,2,1,1,3,2,1,1,3)$ and $C' = (1,4,4,3,1,2,2)$.

The values of its parameters are $L = \lfloor m/q \rfloor = \lfloor 9/4 \rfloor = 2$, $n_{L+1} = (m) mod_q = 1$, $n_L = q - n_{L+1} = 3$, $s_1 = r'_1 = 3$ and $s_2 = r'_2 + r'_3 + r'_4 = 2+1+1 = 4$.

The matrix B_0 has three lines of length $L+1$, exactly $\ell_{1,2}$, $\ell_{1,4}$ and $\ell_{1,7}$, and four lines of length L, exactly $\ell_{3,2}$, $\ell_{2,2}$, $\ell_{4,3}$ and $\ell_{2,6}$ (in Fig. 6, the zones where the lines of lengths 2 and 3 starts, are highlighted with different colors).

Starting from B_0, we construct the 7×7 matrix M which is horizontally and vertically convex on a torus and which has three bars of length three and four bars of length two, (one bar for each line of B_0) as depicted in Fig. 6. The starting column of each line of B_0 is the same as that of the corresponding bar in M.

On the other hand, starting from the matrix M, we compute the $(1,4)$ periodical matrix B_1 having the same number of lines and the same vertical projections as B_0, by placing in B_1 a line for each bar of M. Again the starting column of each bar of M is the same as that of the correspondent line in B_1, while the starting rows of the lines of B_1 are chosen with a greedy strategy.

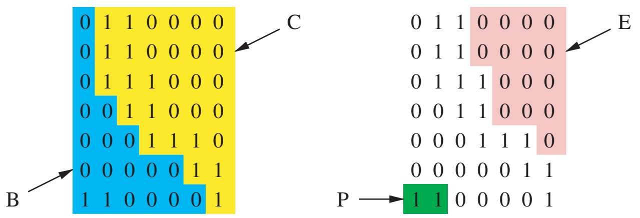

Fig. 7. The matrix M of Fig. 6 and its zones B, C, E and P.

Step 2 (*Lines reconstruction*): where a 2-SAT formula characterizes all the solutions of the instance I'' of RecCPT(C', L, n_L, n_{L+1}, k).

We observe that a solution M of I'' can be divided into four zones (i.e. subsets of positions), say B, C, E and P, such that position (i, j) belongs to $(C \cup P) - E$, if and only if $m_{i,j} = 1$ (symmetrically, position (i, j) belongs to $(B \cup E) - P$ if and only if $m_{i,j} = 0$). In Fig. 7, the four zones of the matrix M depicted in Example 10 are pointed out.

Starting from the instance I'' of RecCPT(C', L, n_L, n_{L+1}, k), we define a 2-SAT formula Ω (a formula in conjunctive normal form, where each clause has at most two literals) whose satisfiability is linked to the existence of a solution M for I'' in such a way: if Ω is satisfiable, then we are able to construct a solution for I'' in polynomial time and, vice versa, each solution of I'' gives, in polynomial time, an evaluation of the variables satisfying Ω.

The formula Ω determines M by characterizing its zones B, C, E, and P, and it is defined as the conjunction of three 2-SAT formulas:

- Ω_1 which encodes the geometrical constraints of M;
- Ω_2 which gives the consistency of M with the horizontal and vertical projections;
- Ω_3 which imposes the maximum number of bars of lengths L and $L+1$ starting on each column of M.

The variables of Ω belong to the union of the four sets of variables:

$$\mathcal{B} = \{b(i, j) : 1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n\}, \quad \mathcal{C} = \{c(i, j) : 1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n\},$$

$$\mathcal{P} = \{p(i, j) : 1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n\} \quad \text{and} \quad \mathcal{E} = \{e(i, j) : 1 \leqslant i \leqslant k, 1 \leqslant j \leqslant n\},$$

which represent B, C, P and E, respectively.

Coding in Ω_1 *the geometrical constraints of* M.

Let $1 \leqslant i \leqslant k$ and $1 \leqslant j \leqslant n$, and let us define Ω_1 as the conjunction of the following sets of clauses:

$$Corners = \bigwedge_{i,j} \left\{ \begin{array}{l} (x(i, j) \Rightarrow x(i-1, j)) \wedge (x(i, j) \Rightarrow x(i, j+1)) \\ (y(i, j) \Rightarrow y(i+1, j)) \wedge (y(i, j) \Rightarrow y(i, j-1)) \end{array} \right\}$$

for $x \in \mathcal{C} \cup \mathcal{E}$ and $y \in \mathcal{B} \cup \mathcal{P}$,

$$\begin{array}{lll} Disj & = & \bigwedge_{i,j} \{(b(i, j) \Rightarrow \overline{c}(i, j)) \wedge (p(i, j) \Rightarrow b(i, j)) \wedge (e(i, j) \Rightarrow c(i, j))\}, \\ Compl & = & \bigwedge_{i,j} \{\overline{b}(i, j) \Rightarrow c(i, j)\}, \\ Anch & = & \{\overline{e}(1, L) \wedge \overline{e}(k - c'_n + 1, n) \wedge \overline{p}(k, L+1) \wedge \overline{p}(k - c'_n, 1)\}. \end{array}$$

In the sequel, we indicate with $mathitCorner(X)$, $X \in \{B, C, E, P\}$, the subset of clauses of *Corners* whose variables belong to $\mathcal{X}$.

Now, let V_1 be an evaluation of the variables in $\mathcal{B}, \mathcal{C}, \mathcal{P}, \mathcal{E}$ which satisfies Ω_1. We define the binary matrix M of size $k \times n$ as follows:

$$(c(i,j)=1 \ \wedge \ e(i,j)=0) \Rightarrow m_{i,j}=1, \quad p(i,j)=1 \Rightarrow m_{i,j}=1,$$
$$(b(i,j)=1 \ \wedge \ p(i,j)=0) \Rightarrow m_{i,j}=0, \quad e(i,j)=0 \Rightarrow m_{i,j}=0.$$

It is immediate to check that M is well defined.

Lemma 11. *The following statements hold*:
(a) *$C-E$ and $B-P$ are h-convex and v-convex regions*;
(b) *$\{B, C\}$ is a partition of M, $P \subseteq B$ and $E \subseteq C$*;
(c) *there are no columns of M where both points of P and points of E lie*;
(d) *there are no rows of M where both points of P and points of E lie*.

Proof. (a) let us suppose that $C-E$ is not v-convex (h-convex as well), i.e. there exist three points m_{i_0,j_0}, m_{i_1,j_0}, and m_{i_2,j_0}, with $i_0 < i_1 < i_2$ such that $m_{i_0,j_0}, m_{i_2,j_0} \in C-E$ and $m_{i_1,j_0} \in E$. By $Corner(E)$, we get that if $m_{i_1,j_0} \in E$ then $m_{i_0,j_0} \in E$, a contradiction. A similar argument holds if the point $m_{i_1,j_0} \in B$. The convexity of the zone $B-P$ can be proved similarly.

(b) immediate from *Disj* and *Compl*.

(c) let us suppose that there exist two points $m_{i_0,j_0} \in P$ and $m_{i_1,j_0} \in E$. If $j_0 \leqslant L$ then, by $Corner(E)$, we get $m_{1,L} \in E$. Since *Anch* imposes $\overline{e}(1,L)$, we get a contradiction. On the other hand, if $j_0 > L$ then, by $Corner(P)$, we get $m_{k,L+1} \in P$. Since *Anch* imposes $\overline{p}(k, L+1)$, we obtain a contradiction (see Fig. 7).

(d) immediate from *Anch*. $\square$

Coding in Ω_2 the upper and lower bounds of the row and column sums of M.

Again we consider $1 \leqslant i \leqslant k$ and $1 \leqslant j \leqslant n$, and let row r be the first one where no points of E lie, as stated in Lemma 11, i.e. $r = k - c'_n$. The formula Ω_2 is the conjunction of the following sets of clauses:

$$LBC = \bigwedge\nolimits_{i,j} \left\{ \begin{array}{ll} \text{if } j > L, & e(i,j) \Rightarrow \overline{b}(i + c'_j, j) \\ \text{if } j \leqslant L, & b(i,j) \Rightarrow p(i + k - c'_j, j) \end{array} \right\},$$

$$UBC = \bigwedge\nolimits_{i,j} \left\{ \begin{array}{ll} \text{if } j > L, & \overline{e}(i,j) \Rightarrow b(i + c'_j, j) \\ \text{if } j \leqslant L, & \overline{b}(i,j) \Rightarrow \overline{p}(i + k - c'_j, j) \end{array} \right\},$$

$$UBR = \bigwedge\nolimits_{i,j} \left\{ \begin{array}{ll} \text{if } i \leqslant r, & \overline{b}(i,j) \Rightarrow e(i, j + L + 1) \\ \text{if } i > r, & p(i,j) \Rightarrow \overline{c}(i, j + n - L - 1) \end{array} \right\},$$

$$LBR = \bigwedge\nolimits_{i,j} \left\{ \begin{array}{ll} \text{if } i \leqslant r, & b(i,j) \Rightarrow \overline{e}(i, j + L) \\ \text{if } i > r, & \overline{p}(i,j) \Rightarrow c(i, j + n - L) \end{array} \right\}.$$

For each column j of M, the formulas *LBC* and *UBC* set the value c_j to be both the lower and the upper bound for the vertical projection of M, while the formulas *LBR* and *UBR* impose to each horizontal projection of M to be greater than L and smaller than $L+1$, respectively.

The constraints coded by *LBC* and *UBC* are expressed in two different ways, depending on the presence of the sets P or E in the columns of M:

- for each $1 \leqslant j \leqslant L$, we impose that the vertical projections of the zone $B - P$ have to be less than or equal to $k - c'_j$ (formula *LBC*), and greater than or equal to $k - c'_j$ (formula *UBC*);
- for each $L < j \leqslant n$, we impose that the vertical projections of the zone $C - E$ have to be greater than or equal to c'_j (formula *LBC*), and less than or equal to c'_j (formula *UBC*).

The constraints on the horizontal projections of M are set with a similar strategy: the matrix is split again into two parts, a first one from row 1 to row r, where the zone P is not present, and a second one from row r till the end of M, where the zone E is not present.

Lemma 12. *Let M be the binary matrix defined by means of the valuation V_2 which satisfies $\Omega_1 \wedge \Omega_2$. It holds*:
(a) *C' is the vector of the vertical projections of M*;
(b) *the value of each horizontal projection of M is L or $L + 1$.*

Proof. We only prove that the set *LBC* gives a lower bound to the vertical projections of M (in the proof we identify each variable with the correspondent truth value associated by V_2).

A complete proof of the lemma is furnished in [10]. Let us proceed by contradiction: if $j > L$, then let us suppose that there exists j_0 such that

$$c'_{j_0} > \sum_{i=1}^{k} c(i, j_0) - e(i, j_0).$$

It follows that there exist i_0 and i_1, with $i_0 < i_1$, $i_1 - i_0 \leqslant c'_{j_0}$ such that $e(i_0, j_0) = 1$ and $b(i_1, j_0) = 1$. By $Corner(B)$, $b(i_0 + c'_{j_0}, j_0) = 1$, and, by *LBC*, we get a contradiction, so

$$\sum_{i=1}^{k} c(i, j_0) - e(i, j_0) \geqslant c'_{j_0}.$$

If $j \leqslant L$, then let us suppose that there exists j_0 such that

$$c'_{j_0} > \sum_{i=1}^{k} c(i, j_0) + p(i, j_0) \quad \text{and so} \quad k - c'_{j_0} < \sum_{i=1}^{k} b(i, j_0) - p(i, j_0).$$

It follows that there exist i_0 and i_1, such that $i_0 < i_1$, $i_1 - i_0 > k - c'_{j_0}$, $b(i_0, j_0) = 1$ and $p(i_1, j_0) = 0$. By $Corner(P)$, it holds $p(i_0 + k - c'_{j_0}, j_0) = 0$, a contradiction, so

$$\sum_{i=1}^{k} b(i, j_0) - p(i, j_0) \leqslant k - c'_{j_0} \quad \text{and} \quad \sum_{i=1}^{k} c(i, j_0) + p(i, j_0) \geqslant c'_{j_0}.$$

In the same fashion, we can prove that *UBC* gives an upper bound to the vertical projections of M, and furthermore, that *LBR* and *UBR* set the bounds for the horizontal projections. □

Remark. The problem characterized by the formula $\Omega_1 \wedge \Omega_2 \wedge \Omega_3$, ($\Omega_3$ being defined hereafter) is slightly more general than $\text{RecCPT}(C', L, n_L, n_{L+1}, k)$, i.e. it is required that on each column j of M, at most max_j^L bars of length L and at most max_j^{L+1} bars of length $L+1$ start (remind that $0 \leqslant max_j^L \leqslant n_L$ and $0 \leqslant max_j^{L+1} \leqslant n_{L+1}$). This new problem, say RECCPT$(C', L, Max_L, Max_{L+1}, k)$, has to be introduced in order to avoid inconsistencies during the merging of the matrices A' and A'' (this last being computed directly from M). In a similar fashion, REC-STRIP can also be modified by strengthening the constraint on the number of lines of length L and $L+1$ starting on each column of its solutions, so that the equivalence described in Step 1 of the line reconstruction stage, is preserved.

Coding in Ω_3 the maximum number of bars of length L and $L+1$ starting on each column of M.

We consider again $r = k - c'_n$, and we define the two vectors

$$Max_L = (max_1^L, \ldots, max_n^L) \quad \text{and} \quad Max_{L+1} = (max_1^{L+1}, \ldots, max_n^{L+1})$$

by using the matrix A' computed in the preprocessing stage, as follows: for each $1 \leqslant j \leqslant n$, max_j^{L+1} is the number of entries 0 (i.e. the entries whose value is not 1 or X) from position $(1, j)$ to position (n_{L+1}, j) of A', and max_j^L is the number of entries 0 from position $(n_{L+1}+1, j)$ to position (q, j) of A'. Roughly speaking, the vectors Max_L and Max_{L+1} store, entry by entry, the maximum number of starting bars of length L and $L+1$ admitted on each column of M. The formula Ω_3 is the conjunction of the following sets of clauses:

$$BB_L = \bigwedge_{i,j} \left\{ \begin{array}{ll} \text{if } i \leqslant r, & b(i - max_j^L, j-1) \Rightarrow \overline{e}(i, j+L) \\ \text{if } (i > r \wedge j > n - L), & b(i - max_j^L, j-1) \Rightarrow p(i, j+L-n) \end{array} \right\},$$

$$BB_{L+1} = \bigwedge_{i,j} \left\{ \begin{array}{ll} \text{if } i \leqslant r, & c(i,j) \Rightarrow e(i - max_j^{L+1}, j+L) \\ \text{if } i > r, & c(i,j) \Rightarrow \overline{p}(i - max_j^{L+1}, j+L-n) \end{array} \right\}.$$

Lemma 13. *Let M be the binary matrix defined by means of the valuation V_3 which satisfies $\Omega_1 \wedge \Omega_2 \wedge \Omega_3$. It holds*:
(a) *on each column j of M, at most max_j^L bars of length L can start*;
(b) *on each column j of M, at most max_j^{L+1} bars of length $L+1$ can start.*

Proof. (a) We proceed by contradiction, and we suppose that there exists a column j_0 where $max_{j_0}^L + h$, with $h > 0$, (consecutive) bars of length L start, from row $i_0 - max_{j_0}^L - h + 1$ to row i_0:
if $i_0 \leqslant r$, then $b(i_0 + max_{j_0}^L, j_0 - 1) = 1$ and $e(i_0, j_0 + L) = 1$, so by BB_L, we obtain a contradiction;
if $i_0 > r$ and $j > n - L$, then $b(i_0 + max_{j_0}^L, j_0 - 1) = 1$ and $\overline{p}(i_0, j_0 + L - n) = 1$, so, by BB_L, we obtain a contradiction.
Hence, for all $1 \leqslant j \leqslant n$, column j contains at most max_j^L starting bars of length L (see Fig. 8).

Point (b) can be similarly proved (see [10]). □

The following theorem is a direct consequence of Lemmas 11–13:

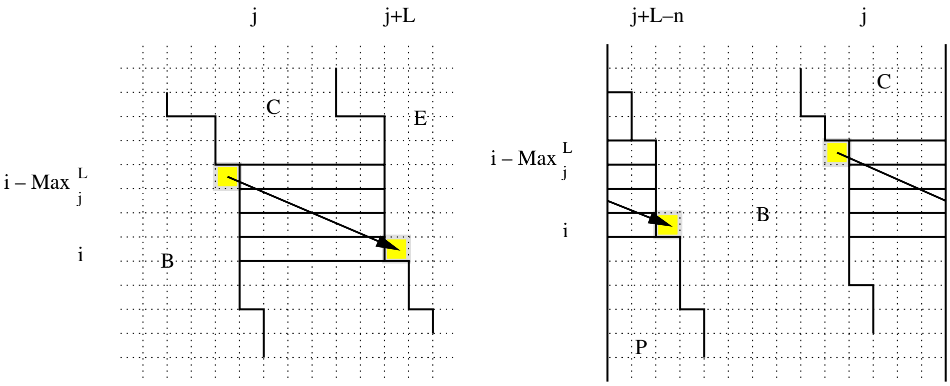

Fig. 8. BB_L prevents these two situations when $max_j^L = 3$, and $i \leqslant r$ or $(i > r \wedge j > n - L)$, respectively.

Theorem 14. *$\Omega_1 \wedge \Omega_2 \wedge \Omega_3$ is satisfiable if and only if there exists an element $M \in \mathcal{T}_{h,v}$ of dimension $k \times n$ which is consistent with C', and such its generic column j contains at most max_j^L starting bars of length L and at most max_j^{L+1} starting bars of length $L+1$.*

Since $\Omega_1 \wedge \Omega_2 \wedge \Omega_3$ is a 2-SAT formula which characterizes a generic instance of RECCPT $(C', L, Max_L, Max_{L+1}, k)$, then its solution requires an amount of time which is linear in the number of its clauses [2].
Step 3 (*Lines reconstruction*): Where matrix A'', solution of instance I', is computed from M.

Since the matrix M, obtained by a valuation of Ω, is a solution of I'', then the equivalence between the problems REC-STRIP $(Per(1, q), (R', C'))$ and RECCPT(C', L, n_L, n_{L+1}, k), proved at the beginning of this section, allows an easy computation of A''.
Fusion. In this final stage the matrices A' and A'' are merged, and the final solution A of REC-STRIP$(Per(1, q), (R, C))$ is achieved by using the Procedure 6, *Fusion* whose details are sketched below. The vectors

$$Start_L = (s_1^L, \ldots, s_n^L) \quad \text{and} \quad Start_{L+1} = (s_1^{L+1}, \ldots, s_n^{L+1})$$

support the computation by storing in s_j^L and s_j^{L+1}, with $1 \leqslant j \leqslant n$, the number of starting lines of length L and $L+1$ in column j of A'', respectively.

It is immediate to observe that Procedure 6 cannot generate inconsistences (by the definitions of A' and A''), and that its output A is one of the solutions of REC-STRIP$(Per(1, q)$, $(R, C))$, as desired. Since we already stressed that each step of the reconstruction is performed in polynomial time, then it holds that:

Theorem 15. *The problem* REC-STRIP$(Per(1, q), (R, C))$ *can be solved in polynomial time*.

Example 16. Let us reconstruct a solution of REC-STRIP$(Per(1, 3), (R, C))$ with

$$R = (4, 2, 5, 2, 3, 5, 5, 1) \quad \text{and} \quad C = (3, 2, 4, 3, 3, 3, 4, 1, 1, 3).$$

Procedure 6. *Fusion*

Initialize matrix A to the values of A';
Compute vector *Start* from A'' as already indicated;
for $j = 1$ to n **do**
  for $i = 1$ to q **do**
    if $(i \leqslant n_{L+1})$ & $(s_j^{L+1} > 0)$ & $(A[i][j] == 0)$ **then**
      $A[i, j] = 1; s_j^{L+1} = s_j^{L+1} - 1;$
    end if
    if $(i > n_{L+1})$ & $(s_j^{L} > 0)$ & $(A[i][j] == 0)$ **then**
      $A[i, j] = 1; s_j^{L} = s_j^{L} - 1;$
    end if
  end for
end for
Complete the lines of A from the placed starting points;
Change back to the value 0 all the elements of A of value X;
Return A as output.

$$
\mathrm{A}': \begin{array}{cccccccccc}
0&0&0&0&0&0&0&0&\mathrm{X}&1\\
0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0
\end{array}
\qquad
\mathrm{M}: \begin{array}{cccccccccc}
0&1&1&0&0&0&0&0&0&0\\
0&1&1&1&0&0&0&0&0&0\\
0&0&1&1&0&0&0&0&0&0\\
0&0&1&1&1&0&0&0&0&0\\
0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&1&1&1&0&0&0\\
0&0&0&0&0&1&1&0&0&0\\
0&0&0&0&0&0&1&1&0&0\\
1&0&0&0&0&0&0&0&1&1\\
1&0&0&0&0&0&0&0&0&1
\end{array}
\qquad
\mathrm{A}'': \begin{array}{cccccccccc}
0&0&0&0&1&0&0&0&0&0\\
0&1&1&0&1&0&0&0&1&0\\
0&1&1&0&0&1&1&0&0&1\\
0&0&0&0&0&1&0&0&0&0\\
0&0&1&1&0&1&0&0&0&1\\
1&0&1&1&0&0&1&1&0&0\\
0&0&0&0&0&0&1&0&0&0\\
1&0&0&1&1&0&1&0&0&0
\end{array}
$$

Fig. 9. The three matrices which support the computation of Rec-Strip($Per(1, 3), (R, C)$). The final solution is obtained by merging A' with A'', and then changing back all the X to the value 0.

Preprocessing: The 8×10 matrix A' is created and its elements are initialized to the value *blank*. It is computed the vector $Sum = (6, 5, 5, 5, 6)$ which allows to detect an entry 1 in the couple of left boxes (7, 1) and (8, 1), and an entry 1 in the couple of right boxes (1, 10) and (2, 10) not belonging to any line. These two entries are placed (with a greedy strategy) in positions (8, 1) and (2, 10), and position (2, 9) is marked with the symbol X in order to prevent lines to start there. Finally, all the *blank* are changed to the value 0, obtaining matrix A' of Fig. 9.
Lines reconstruction: starting from the updated vectors

$$R' = (3, 2, 5, 2, 3, 5, 4, 1), \quad C' = (2, 2, 4, 3, 3, 3, 4, 1, 1, 2) \quad \text{and}$$

$$Sum = (5, 5, 5, 5, 5),$$

the instance I'' of RECCPT(C', L, Max_L, Max_{L+1}, k) is created, where $L = 2$, $k = 10$, $Max_{L+1} = \{2, 2, 2, 2, 2, 2, 2, 2, 2, 1, 1\}$, and all the elements of Max_L are set to the same value $n_L = 1$.

Then, I'' is characterized by a 2-SAT formula Ω, one of whose valuations determines matrix M depicted in Fig. 9. One can immediately observe that M belongs to $\mathscr{T}_{h,v}$, it is consistent with C', its horizontal projections have values 2 or 3, and it satisfies the constraints imposed by the vectors Max_L and Max_{L+1}. The matrix A'' of Fig. 9 is computed from M. *Fusion:* The matrices A' and A'' merge into the final solution A. Notice again that no inconsistencies can occur at this stage, since the number of starting lines on each column of A is tuned by the entries of the vectors Max_L and Max_{L+1}.

5. Conclusions

Our main purpose here has been to introduce periodicity properties in terms relevant for discrete tomography. The periodicity is a natural constraint and it has not yet been studied in this environment. As pointed out in the Introduction, the motivation of this study is in the attempt of limiting the class of possible solutions when we reconstruct a discrete planar object using a priori information comprehending also its periodical behavior. This means that we modelled such a knowledge in terms of a subclass of binary images to which the object must belong.

It is not surprising that we obtain also some interesting uniqueness results, as pointed out for the class of binary matrices having period (1, 1). We have also shown a simple greedy algorithm for reconstructing an element of $Per(1, 1)$ consistent with a given couple of vectors of horizontal and vertical projections. Such a reconstruction becomes more difficult when dealing with binary matrices having period $(p, 1)$ or $(1, q)$. In these cases, we have described a polynomial time algorithm which solves the subproblem REC-STRIP, and which uses a reduction to 2-Satisfiability problem. We want to point out that an interesting property of this approach is that it uses a sub-procedure for reconstructing an element of $\mathcal{T}_{h,v}$ (a subclass of convex polyominoes lying on a torus) from the partial knowledge of its horizontal and vertical projections.

Future challenges will concern the general problem of the reconstruction of binary matrices with period $(1, q)$ and $(p, 1)$ from their projections, and the extension of this result to the class $Per(p, q)$. So, this paper is only an initial approach to the problem of reconstructing binary matrices having periodicity properties from a small number of discrete projections. Lot of work should be done to understand such environment: we only challenge the reconstruction problem from two projections in some special cases, but many consistency, reconstruction and uniqueness problems can be reformulated imposing periodical constraints.

References

[2] B. Aspvall, M.F. Plass, R.E. Tarjan, A linear-time algorithm for testing the truth of certain quantified Boolean formulas, Inform. Process. Lett. 8 (1979) 121–123.

[3] E. Barcucci, S. Brunetti, A. Del Lungo, M. Nivat, Reconstruction of lattice sets from their horizontal, vertical and diagonal X-rays, Discrete Math. 241 (2001) 65–78.
[4] E. Barcucci, A. Del Lungo, M. Nivat, R. Pinzani, Reconstructing convex polyominoes from their horizontal and vertical projections, Theoret. Comput. Sci. 155 (1996) 321–347.
[5] R.A. Brualdi, Matrices of zeros and ones with fixed row and column sum vectors, Linear Algebra Appl. 33 (1980) 159–231.
[6] S. Brunetti, A. Daurat, An algorithm reconstructing convex lattice sets, Theoret. Comput. Sci. (Issues 1–3) 304 (28) (2003) 35–57.
[7] S. Brunetti, A. Daurat, A. Del Lungo, Approximate X-rays reconstruction of special lattice sets, Pure Math. Appl. 11 (3) (2000) 409–425.
[8] M. Chrobak, C. Dürr, Reconstructing hv-convex polyominoes from orthogonal projections, Inform. Process. Lett. 69 (1999) 283–289.
[9] A. Del Lungo, M. Nivat, Reconstruction of connected sets from two projections, in: G.T. Herman, A. Kuba (Eds.), Discrete Tomography: Foundations, Algorithms and Applications, Birkhauser, Boston, MA, USA, 1999, pp. 85–113.
[10] A. Frosini, Complexity results and reconstruction algorithms for discrete tomography, Ph.D. Thesis, University of Siena, Siena, Italy, 2003.
[11] D. Gale, A theorem on flows in networks, Pacif. J. Math. 7 (1957) 1073–1082.
[12] R.J. Gardner, P. Gritzmann, Discrete tomography: determination of finite sets by X-rays, Trans. Amer. Math. Soc. 349 (1997) 2271–2295.
[13] G.T. Herman, A. Kuba (Eds.), Discrete Tomography: Foundations, Algorithms and Applications, Birkhauser Boston, Cambridge, MA, 1999.
[14] C. Kisielowski, P. Schwander, F.H. Baumann, M. Seibt, Y. Kim, A. Ourmazd, An approach to quantitative high-resolution transmission electron microscopy of crystalline materials, Ultramicroscopy 58 (1995) 131–155.
[17] H. Ryser, Combinatorial mathematics, The Carus Mathematical Monographs, Math. Assoc. America, Vol. 14, 1963.
[18] L.A. Shepp, J.B. Kruskal, Computerized tomography: the new medical X-ray technology, Amer. Math. Monthly 85 (1978) 420–439.