

# Approximating hv-Convex Binary Matrices and Images from Discrete Projections

Fethi Jarray<sup>1,2,\*</sup>, Marie-Christine Costa<sup>2</sup>, and Christophe Picouleau<sup>2</sup>

<sup>1</sup> Gabes University of Sciences 6072 Gabes, Tunisia

<sup>2</sup> Laboratoire CEDRIC, 292 rue Saint-Martin, 75003 Paris, France  
fethi\_jarray@yahoo.fr, {costa,chp}@cnam.fr

**Abstract.** We study the problem of reconstructing hv-convex binary matrices from few projections. We solve a polynomial time case and we determine some properties of the hv-convex matrices. Since the problem is NP-complete, we provide an iterative approximation based on a longest path and a min-cost/max-flow model. The experimental results show that the reconstruction algorithm performs quite well.

**Keywords:** Discrete Tomography; hv-convex; Image Reconstruction.

## 1 Introduction

Discrete tomography deals with the reconstruction of discrete homogeneous objects regarded as binary matrices from their projections. The problem of reconstructing a  $m \times n$  binary matrix from its orthogonal projections  $H$  and  $V$  is the following [12]: given  $H = (h_1, \dots, h_m)$  and  $V = (v_1, \dots, v_n)$  two nonnegative integer vectors find a binary matrix such that the number of ones in every row  $i$  (resp. column  $j$ ) equals  $h_i$  (resp.  $v_j$ ). Ryser [12] gives necessary and sufficient conditions for the existence of a solution. However, the problem is usually highly underdetermined and a large number of solutions may exist [13]. The reader is referred to the book of Herman and Kuba [9] for an overview on discrete tomography.

In many applications such as image processing and electron microscopy, the orthogonal projections alone are not sufficient to uniquely determine matrices or objects. Fortunately, objects that occur in practical applications usually exhibit certain properties. Hence we seek to reconstruct binary matrices under additional constraints like connectivity or convexity for instance. Woeginger [14] prove that the consistency problem for polyominoes (connected sets) is NP-complete. The consistency problem for h-convex objects (polyominoes or not) is also NP-complete [1]. The above result extends to the v-convex objects. The consistency problem for hv-convex objects is NP-complete [1,14]. Therefore, the reconstruction can be solved in polynomial time only for the hv-convex polyominoes objects [1,5].

In the present paper, we will deal with the problem of reconstructing hv-convex matrices. Since the problem is NP-complete, one way to solve it is to

---

\* The corresponding author.

concentrate on approximative solutions. There are few papers in the literature on algorithms providing approximating results, i.e. returning binary matrices respecting the orthogonal projections but not necessary hv-convex[6,3,2]. Dahl and Flatberg [6] provide an algorithm based on a lagrangian relaxation for reconstructing a nearly hv-convex matrix respecting the prescribed projections.

In this paper, we propose a new algorithm for approximately reconstructing hv-convex matrices from their projections. Our algorithm performs a sequence of related reconstruction, each using only one projection. The algorithm uses longest path and network flows algorithms to provide a solution.

The paper is organized as follows. Notations and properties of hv-convex matrices are introduced in section 2. In section 3, we study the problem of reconstructing a hv-convex matrix respecting only one orthogonal projection (H or V). In Section 4, we propose a heuristic based on longest path and network models to approximately solve the problem with both projections. Numerical results are presented and discussed in the last section.

## 2 Definitions and Properties

Let  $x$  be an  $m \times n$  binary matrix. The horizontal projection of  $x$  is the vector  $H = (h_1, \dots, h_m)$  such that  $h_i = \sum_{j=1}^n x_{ij}$  is the sum of the elements lying on row  $i$ . The vertical projection of  $x$  is the vector  $V = (v_1, \dots, v_n)$  where  $v_j = \sum_{i=1}^m x_{ij}$  is the sum of the elements on column  $j$ . Both projections  $H$  and  $V$  constitute the orthogonal projections of  $x$ . A matrix is horizontally convex (h-convex) if all the 1's of each row are adjacent. A matrix is vertically convex (v-convex) if all the 1's of each column are adjacent. A matrix is hv-convex if it is both h-convex and v-convex.

Because discrete tomography is very related to digital image processing we often refer to binary matrices as binary images and call the matrix entries pixels with values black (0) and white (1).

The related decision problem associated to the problem of reconstruction hv-convex matrix is defined as follows:

**Instance:** Given  $H = (h_1, \dots, h_m)$  and  $V = (v_1, \dots, v_n)$  two nonnegative integer vectors.

**Question:** Is there a  $m \times n$  hv-convex matrix respecting the horizontal projection  $H$  and the vertical projection  $V$ ?

**Definition 1.** Let  $x$  and  $y$  be two  $m \times n$  binary matrices. We define the resemblance  $R(x, y)$  between  $x$  and  $y$  as  $R(x, y) = \sum_{i=1}^m \sum_{j=1}^n x_{ij}y_{ij}$ .

$R(x, y)$  is the number of common 1's between  $x$  and  $y$ .

For any binary matrix  $x$ , denote the number of pairs of adjacent 1's (either horizontal or vertical) in  $x$  by  $n_x$ .

**Proposition 1.** Let  $x$  be a binary matrix with orthogonal projections  $(H, V)$  then

- i)  $n_x = \sum_{i=1}^m \sum_{j=1}^{n-1} x_{ij}x_{i,j+1} + \sum_{j=1}^n \sum_{i=1}^{m-1} x_{ij}x_{i+1,j}$
- ii)  $n_x \leq 2 \sum_{j=1}^n v_j - m - n$
- iii)  $x$  is hv-convex if and only if  $n_x = 2 \sum_{j=1}^n v_j - m - n$ .

*Proof.* Let  $x$  be a binary matrix respecting the projections  $(H, V)$ . The number of adjacent 1's on row  $i$  is  $\sum_{j=1}^{n-1} x_{ij}x_{i,j+1} \leq h_i - 1$ . Row  $i$  is h-convex if and only if the number of adjacent 1's is  $h_i - 1$ .

Analogously, the number of adjacent 1's on column  $j$  is  $\sum_{i=1}^{m-1} x_{ij}x_{i+1,j} \leq v_j - 1$ . Column  $j$  is v-convex if and only if the number of adjacent 1's is  $v_j - 1$ .

Thus we deduce that  $n_x = \sum_{i=1}^m \sum_{j=1}^{n-1} x_{ij}x_{i,j+1} + \sum_{j=1}^n \sum_{i=1}^{m-1} x_{ij}x_{i+1,j} \leq \sum_{i=1}^m (h_i - 1) + \sum_{j=1}^n (v_j - 1) = 2 \sum_{i=1}^m h_i - m - n = 2 \sum_{j=1}^n v_j - m - n$ . So  $n_x \leq 2 \sum_{j=1}^n v_j - m - n$  is an upper bound on  $n_x$ .

A matrix is hv-convex if and only if all the rows are h-convex and all the columns are v-convex. So, a matrix  $x$  is hv-convex if and only if  $n_x = 2 \sum_{j=1}^n v_j - m - n$ .

### 3 Solving Polynomial Cases

We propose two algorithms to solve two important particular subproblems.

#### 3.1 Reconstructing a hv-Convex Matrix Respecting One Projection hv-Convex(V)

In this section, we provide a polynomial time algorithm to reconstruct a hv-convex matrix respecting the vertical projection  $V$ .

We suppose that on each column  $j$  the 1's are placed on the rows from  $s_j$  to  $e_j = s_j + v_j - 1$  since the matrix is hv-convex. To reconstruct a solution, it is sufficient to determine  $s_j$  or  $e_j$  for each column.

**Proposition 2.** *A  $m \times n$  binary matrix with vertical projection  $V$  is hv-convex only if  $m \geq v_1 + \sum_{j=2}^n \max(v_j - v_{j-1}, 0)$*

*Proof.* We will demonstrate the proposition by induction on  $n$ . For  $n = 1$ , the proposition is true because  $m \geq v_1$ . Suppose that the proposition is true for a  $m \times (n - 1)$  matrix and show that is also verified for a  $m \times n$  matrix.

Let  $x$  be a  $m \times n$  hv-convex binary matrix with vertical projection  $V$ . Denote by  $x_{n-1}$  the submatrix of  $x$  of size  $m \times (n - 1)$  composed by the first  $n - 1$  columns. We have  $m \geq v_1 + \sum_{j=2}^{n-1} \max(v_j - v_{j-1}, 0)$  because  $x_{n-1}$  is also hv-convex.

Two cases can be distinguished according to the values of  $v_{n-1}$  and  $v_n$ .

- If  $v_n \leq v_{n-1}$ , the proposition is true because  $v_1 + \sum_{j=2}^{n-1} \max(v_j - v_{j-1}, 0) = v_1 + \sum_{j=2}^n \max(v_j - v_{j-1}, 0) \leq m$ .
- If  $v_n > v_{n-1}$ , we denote by  $A$  the set of rows of the submatrix  $x_{n-1}$  having at least a 1 from columns  $1, \dots, n - 1$ ,  $B$  the set of rows having a 1 on column  $n - 1$  and  $C$  the set of rows having a 1 on column  $n$ . We have  $|B| = v_{n-1}$ ,

$|C| = v_n$  and  $|A| \geq v_1 + \sum_{j=2}^{n-1} \max(v_j - v_{j-1}, 0)$  because the submatrix  $x_{n-1}$  is hv-convex. The set of rows of  $x$  is equal to  $A \cup C$ . Since the matrix  $x$  is h-convex then  $A \cap C \subset B$ . Thus  $m = |A \cap C| = |A| + |C| - A \cap C \geq |A| + |C| - |B| \geq v_1 + \sum_{j=2}^{n-1} \max(v_j - v_{j-1}, 0) + v_n - v_{n-1} = v_1 + \sum_{j=2}^n \max(v_j - v_{j-1}, 0)$ . We conclude that the proposition is true for the matrix of size  $m \times n$ .

In each step  $j$ , the following greedy algorithm sets to 1  $v_j$  cells from row  $s_j$  to row  $s_j + v_j - 1$ .

**Greedy algorithm**

$s_1 = 1$

**For**  $j = 2$  to  $n$  **do**

**if**  $v_{j-1} \leq v_j$  **then**  $s_j = s_{j-1}$

**if**  $v_{j-1} > v_j$  **then**  $e_j = e_{j-1}$

The following result establishes the validity of the greedy algorithm.

**Proposition 3.** *The greedy algorithm solves in polynomial time the problem hv-convex(V).*

*Proof.* It is obvious that the reconstructed matrix is v-convex and respects the vertical projection. This matrix is also h-convex because it is v-convex, that is,  $s_j \leq e_j$ ,  $j = 1, \dots, n$ , the sequence  $(s_1, \dots, s_n)$  is monotone non decreasing and the sequence  $(e_1, \dots, e_n)$  is also monotone non decreasing. The reconstructed matrix has a minimal number of rows ( $m = v_1 + \sum_{j=2}^n \max(v_j - v_{j-1}, 0)$ ).

We get similar result if the vertical projection is relaxed instead of the horizontal one. We establish the following general result:

**Proposition 4.** *A  $m \times n$  binary matrix with orthogonal projections  $(H, V)$  is hv-convex only if  $m \geq v_1 + \sum_{j=2}^n \max(v_j - v_{j-1}, 0)$  and  $n \geq h_1 + \sum_{i=2}^m \max(h_i - h_{i-1}, 0)$*

**3.2 Reconstructing a Nearly h-Convex Matrix Respecting the Horizontal Projection h-Convex(H,w)**

The problem *h-convex(H,w)* consists of reconstructing a binary matrix  $x$  respecting the horizontal projection  $H$ , having the maximum number of adjacent 1's on the rows and being as near as possible to a given binary matrix  $w$ . We define the problem *v-convex(V,w)* analogously. We formulate the reconstruction problem as a quadratic integer program. A very natural formulation is the following:

$$P_h \quad \begin{cases} \max \sum_{i=1}^m \sum_{j=1}^n x_{ij} w_{ij} + \sum_{i=1}^m \sum_{j=1}^{n-1} x_{ij} x_{i,j+1} \\ \sum_{j=1}^n x_{ij} = h_i \quad i = 1, \dots, m \\ x_{ij} \in \{0, 1\} \end{cases} \quad (1)$$

The objective function is the sum of the resemblance between the matrices  $x$  and  $w$  (first term) and the number of adjacent 1's on the rows of  $x$ . The

constraint (1) is necessary and sufficient to guarantee the satisfaction of the horizontal projection.

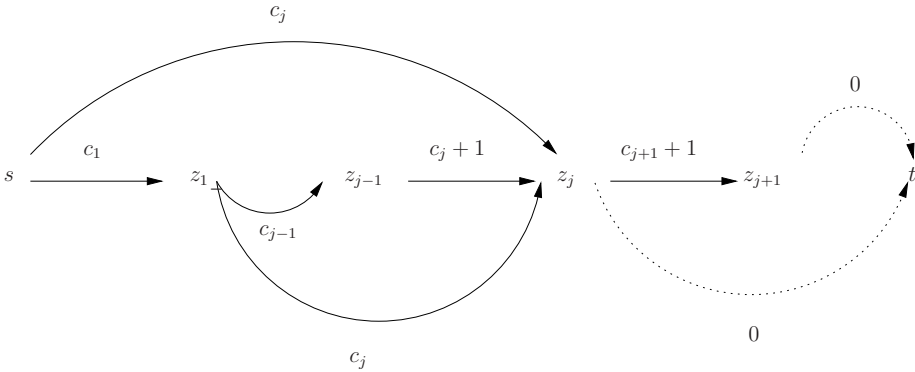
We note that the rows are decoupled and the problem  $P_h$  can be decomposed into  $m$  subproblems, one per row. For a given row  $i$ , the associate subproblem to solve has the following general form:

$$Q \quad \begin{cases} \max \sum_{j=1}^n z_j c_j + \sum_{j=1}^{n-1} z_j z_{j+1} \\ \sum_{j=1}^n z_j = b \\ z_j \in \{0, 1\} \end{cases} \quad (2)$$

with  $x_{ij} = z_j$ ,  $w_{ij} = c_j$  and  $h_i = b$ .

Dahl and Fatberg [6] show that the program  $Q$  is equivalent to a longest path problem with exactly  $b + 1$  vertices in the directed graph  $G$  presented in Figure 1. In this graph there is a node  $z_j$  for every column  $j$ . In addition, there is an artificial source node  $s$  and a sink node  $t$ . The arcs of  $G$  are defined as follows:

- There is an arc  $(s, z_j)$  with length  $l(s, z_j) = c_j$ ,  $j = 1, \dots, n$ .
- There is an arc  $(z_j, t)$ , with length  $l(z_j, t) = 0$ ,  $j = 1, \dots, n$ .
- There is an arc  $(z_j, z_{j+1})$ , with length  $l(z_j, z_{j+1}) = c_{j+1} + 1$ ,  $j = 1, \dots, n - 1$ .
- There is an arc  $(z_k, z_j)$  with length  $l(z_k, z_j) = c_j$ ,  $k = 1, \dots, j - 2$ .



**Fig. 1.** The program  $Q$  and the associated longest path problem

The longest path problem with given number of vertices can be solved in polynomial time by dynamic programming [6].

We define  $L(z_j, p)$  to be the length of the longest path from  $s$  to  $z_j$  using  $p$  vertices. The  $L(z_j, p)$  verifies the following relations:

$$\begin{cases} L(z_j, 1) = c_j, j = 1, \dots, n \\ L(z_j, p) = \max_{k < j} \{L(z_k, p - 1) + l(z_k, z_j)\} \end{cases}$$

The longest path problem with exactly  $b + 1$  vertices from  $s$  to  $t$  is with length  $L(t, b + 1) = \max_j \{L(z_j, b)\}$ . For  $Q$ , we have a complexity of  $O(bn^2)$ , yielding a time complexity of  $O(n^4)$  for  $P_h$  since  $h_i \leq n$ ,  $i = 1, \dots, m$ .

## 4 Heuristic

We provide the iterative algorithm A-hv(H,V) based on a longest path model to compute an approximate solution to the problem of reconstructing an hv-convex matrix. The aim is to maximize the resemblance between the solutions to  $h$ -convex( $H,w$ ) and  $v$ -convex( $V,w$ ). Every iteration consists of solving  $h$ -convex( $H,w$ ) or  $v$ -convex( $V,w$ ) with an objective function depending on the previous iteration. The longest path model is used to solve the problems  $h$ -convex( $H,w$ ) and  $v$ -convex( $V,w$ ) (see Section 3.2). The length matrix  $w$  is chosen in such a way that the new reconstruction has the maximal resemblance with the previous one.

Firstly, we compute a solution  $y^0$  to  $hv$ -convex( $V$ ) by using the greedy algorithm (see Section 3.1). Then, we determine  $x^1$  solution to  $h$ -convex( $H,y^0$ ) by solving the associated longest path problem where the length matrix  $w = y^0$ , i.e. a solution that differs from  $y^0$  in as few entries as possible. Subsequently  $x^1$  is used as a length matrix to determine a solution to  $v$ -convex( $V,x^1$ ). This procedure is repeated until an optimum is reached, i.e. the resemblance between the solutions to  $h$ -convex( $H,w$ ) and  $v$ -convex( $V,w$ ) becomes constant. As a summary we can describe the reconstruction algorithm as follows:

**Algorithm A-hv(H,V)**

Compute  $y^0$  solution to  $hv$ -convex( $V$ ).

$i = 0, R^{-1} = -1, R^0 = 0,$

**While**  $R^i > R^{i-1}$  **do**

    Compute  $x^{i+1}$  solution to  $h$ -convex( $H,y^i$ ) maximizing the resemblance with  $y^i$ .

    Compute  $y^{i+1}$  solution to  $v$ -convex( $V,x^{i+1}$ ) maximizing the resemblance with  $x^{i+1}$ .

$R^{i+1} = R(x^{i+1}, y^{i+1})$  and  $i = i + 1$ .

We will give some properties of this heuristic.

**Proposition 5.** *The resemblance is increasing from iteration to iteration, i.e.  $R^{i+1} \geq R^i$ .*

*Proof.*  $R(x^{i+1}, y^{i+1}) \geq R(x^i, y^i)$  because  $R(x^{i+1}, y^i) \geq R(x^i, y^i)$  and  $R(x^{i+1}, y^{i+1}) \geq R(x^{i+1}, y^i)$ .

We conclude that this heuristic is a polynomial time algorithm because the resemblance is not greater than the maximal number of adjacent 1's ( $n_x$ ).

**Proposition 6.** *If  $R(x^i, y^i) < R(x^{i+1}, y^{i+1})$  then  $x^{i+1} \neq x^j$  for  $j = 1, \dots, i$ .*

*Proof.* Suppose that  $x^{i+1} = x^j$  for some  $j$  in the range  $1 \leq j \leq i$ . By the proposition above,  $R(x^j, y^j) \leq R(x^i, y^i)$  and by the algorithm A-hv(H,V),  $R(x^j, y) \leq R(x^j, y^j)$  for every solution  $y$  to  $v$ -convex( $V$ ). Hence  $R(x^j, y) \leq R(x^j, y^j) \leq R(x^i, y^i) < R(x^{i+1}, y^{i+1})$  for all  $y$ . In particular for  $y = y^{i+1}$ , we get  $R(x^j, y^{i+1}) < R(x^{i+1}, y^{i+1})$ , a contradiction since  $x^{i+1} = x^j$ .

Let  $x$  and  $y$  be the solutions provided by the previous algorithm to the problems  $h$ -convex( $H,w$ ) and  $v$ -convex( $V,w$ ). If  $x = y$  then  $x$  is an exact solution

to  $hv(H, V)$ . Otherwise, we solve the following integer program to get a binary matrix  $z$  respecting  $(H, V)$  and nearly hv-convex.

$$\begin{cases} \max \sum_{i=1}^m \sum_{j=1}^n z_{ij}x_{ij} + \sum_{i=1}^m \sum_{j=1}^n z_{ij}y_{ij} \\ \sum_{j=1}^n z_{ij} = h_i \quad i = 1, \dots, m \\ \sum_{i=1}^m z_{ij} = v_j \quad j = 1, \dots, n \\ z_{ij} \in \{0, 1\} \end{cases}$$

The matrix  $z$  respects both projections and resembles  $x$  and  $y$  since the objective function is  $R(z, x) + R(z, y)$ . This program is equivalent to a min-cost/max-flow in a complete bipartite graph [6,10].

## 5 Computational Results

We have implemented our algorithm in language C. The min-cost/max-flow models used by the heuristic are solved by the *CS2* network flow library developed by Andrew Goldberg [8]. All results are obtained using a PC with 3.8 GHz processor and 512 MBs of RAM.

The main criterion to evaluate the performance of our heuristic is the ability to reconstruct hv-convex or nearly hv-convex matrix. For example, if the optimal solution has 100 adjacent 1’s and our heuristic provides a solution with 95 adjacent 1’s, we say that the algorithm is efficient. For each problem, the heuristic either converges to an hv-convex matrix or provides an approximate hv-convex matrix. As in [6,2], we have tested our heuristic with two sets of images.

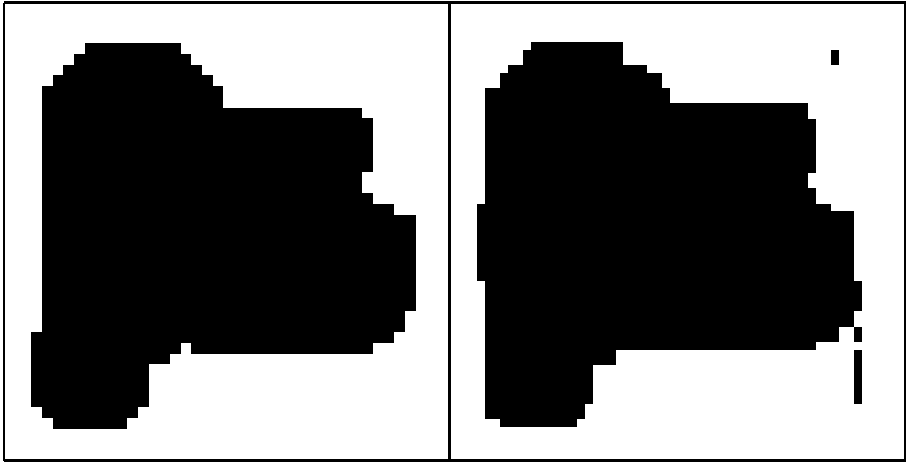
### 5.1 hv-Convex Images

The first set of test cases consists of hv-convex images of various sizes. Figure 2 illustrates an approximate solution provided by the algorithm.

The results of computational experiments are summarized in Table 1. The first column contains the size of the problem. The next column gives the upper bound on the number of adjacent 1’s given by Proposition 1 ( $n_x \leq 2 \sum_{j=1}^n v_j - m - n$ ). Adjacent is the number of adjacent 1’s on the provided solution. gap is the gap between the number of adjacent 1’s on the approximated solution and the upper bound. The last column displays the total CPU time in seconds.

For the small sizes, the upper bound is tight and the algorithm gives an hv-convex matrix. For the remaining instances, the algorithm finds an approximate solution respecting both projections and having the maximum number of adjacent 1’s.

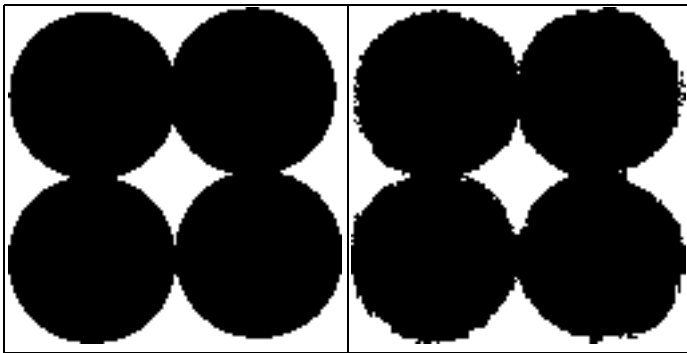
The gap between the number of adjacent 1’s on the approximative solution and the upper bound is about 2%. Since, the difference is negligible by regarding the size of the matrices, we conclude that the heuristic gives an approximate hv-convex matrice respecting as much as possible both projections.



**Fig. 2.** A  $40 \times 40$  image instance and its reconstruction

**Table 1.** Reconstruction results for hv-convex images

size	bound	adjacent	gap(%)	run time
(9,23)	66	66	0	0
(40,40)	96	96	0	0.36
(50,40)	2750	2673	2.8	0.62
(50,50)	3438	3270	5	0.77
(75,75)	7064	6973	1.3	3.21
(100,100)	10322	10203	1.2	8.45



**Fig. 3.** A  $40 \times 40$  image instance and its reconstruction



## 5.2 Random Circles

The second group of tests consists of random circles of various sizes. The matrices of this set are not necessary hv-convex (see Figure 3).

The computational results are presented in Table 2. This set of test cases exposes another advantage of the algorithm. For all the large tested problems, the algorithm provides a solution with an average number of adjacent 1's very close to the upper bound. The overall execution time of the algorithm is always under 30s.

In [6], the hv-convex problem was approximately solved by a lagrangian decomposition method. Regarding the processing time, our algorithm seems to be much faster than the previous algorithm. We note also that our heuristic is much simpler than the previous algorithm which uses a gradient procedure to solve the lagrangian problem.

**Table 2.** Reconstruction results for random circles

size	bound	adjacent	gap (%)	run time
(86,99)	7785	7540	3	5.13
(136,133)	18133	17591	3	22.86
(117,148)	19701	18766	4.75	29.70
(126,125)	24565	24328	1	22
(117,136)	17337	16788	3.2	22.85

## 6 Conclusion

In this paper, we have provided an iterative algorithm to approximate hv-convex matrices from orthogonal projections. For evaluation, we have considered two types of images: hv-convex and random circles. Given the size and the complexity of reconstructing hv-convex matrices, we believe that the computational results are very encouraging as they show that the problem can be solved to near optimality very fast.

## Acknowledgments

The authors would like to thank the referees for their valuable comments and suggestions.

## References

1. Barcucci, E., Del Lungo, A., Nivat, M., Pinzani, R.: Reconstructing convex polyominoes from their horizontal and vertical projections. *Theoretical computer science* 155(1), 321–347 (1996)

2. Batenburg, K.J.: An evolutionary algorithm for discrete tomography. *Discrete applied mathematics* 155(1), 36–54 (2005)
3. Boufkhad, Dubois, Nivat, M.: Reconstructing  $(h,v)$ -convex 2-dimensional patterns of objects from approximate horizontal and vertical projections. *Theoretical Computer Science* 290(3), 1647–1664 (2003)
4. Brualdi, R.A.: Matrices of zeros and ones with fixed row and column sum. *Linear algebra and its applications* 3, 159–231 (1980)
5. Chrobak, M., Dürr, C.: Reconstructing  $hv$ -convex polyominoes from orthogonal projection. *Inform. Process. Lett.* 69, 283–289 (1999)
6. Dahl, G., Fatberg, T.: Optimization and reconstruction of  $hv$ -convex  $(0,1)$ -matrices. *Discrete applied mathematics* 155(1), 93–105 (2005)
7. Gardner, R.J., Gritzmann, P., Prangenberg, D.: On the computational complexity of determining polyatomic structures by X-rays. *Theoretical computer science* 233, 91–106 (2000)
8. Goldberg, A.V.: An efficient implementation of a scaling minimum-cost flow algorithm. *Journal of algorithms* 22, 1–29 (1997)
9. Herman, G.T., Kuba, A.: *Discrete Tomography: Foundations, Algorithms and Applications*. Birkhäuser, Basel (1999)
10. Jarray, F.: Solving problems of discrete tomography. Applications in workforce scheduling, Ph.D. Thesis, University of CNAM, Paris (2004)
11. Picouleau, C.: Reconstruction of domino tiling from its two orthogonal projections. *Theoretical computer science* 255(1), 437–447 (2001)
12. Ryser, H.J.: Combinatorial Properties of Matrices of Zeros and Ones. *Canad. J. Math* 9, 371–377 (1957)
13. Wang, B., Zhang, F.: On the precise Number of  $(0,1)$ -Matrices in  $u(R,S)$ . *Discrete mathematics* 187, 211–220 (1998)
14. Woeginger, G.J.: The reconstruction of polyominoes from their orthogonal projections. *Information Processing Letters* 77(5-6), 225–229 (2001)