A TABU SEARCH APPROACH FOR THE RECONSTRUCTION OF BINARY IMAGES WITHOUT EMPTY INTERIOR REGION

ALAIN BILLIONNET, FETHI JARRAY, GHASSEN TLIG AND EZZEDINE ZAGROUBA

Abstract. In this paper, we are concerned with a discrete tomography problem. We seek to reconstruct a binary image from its orthogonal projections, i.e., its horizontal and vertical line sums without interior black holes. We provide a tabu search approach to minimize the number of holes while satisfying the projections. We test our approach on some random binary images. Computational results show that the algorithm proposed produces near-optimal solutions for all test problems.

1. Introduction

Discrete tomography is a modern inverse problem in which we consider the horizontal and the vertical line sums of an image as a basis to obtain the pixels of the image. Binary images are most commonly represented by binary matrices with values 1 (white pixel) and 0 (black pixel). We will refer to any pixel of the image by its matrix position (see Fig. 1).

Discrete tomography is applicable in many interesting contexts to reconstruct discrete structures such as Workforce Scheduling [9, 16], Data Compression and Data Security [15], Industrial Non-Destructive Testing [4], Medical Imaging [13, 17] and Timetabling [5].

In a binary image, we call the horizontal projection of row \(i\), the number of white pixels in this row. Similarly, we call the vertical projection of column \(j\), the number of white pixels in column \(j\). The basic problem of reconstructing a binary matrix from its orthogonal projections is defined as follows: given the horizontal projection of each row and the vertical projection of each column, find a binary image that fits with the prescribed projections. Ryser described [18] a polynomial time algorithm to solve such a problem and gave necessary and sufficient conditions on the projections for the existence of a reconstruction.

Generally, the basic problem is underdetermined and many solutions may exist. To reduce the space of feasible solutions, various forms of prior knowledge on size, shape, smoothness, etc. are integrated to uniquely reconstruct the original structure from the projections. Incorporating such information into a method is a difficult task and the reconstruction problem is usually reduced to an optimization problem to select the best solution in a certain sense. In the literature, several additional constraints and prior knowledge have been considered to reduce the set


Keywords: discrete tomography, tabu search, adjacency binary images, interior holes binary images.
of the feasible solutions of the basic problem. These constraints include periodicity [10, 11], convexity [1–3, 6, 8, 14], adjacency [5], connectedness [3, 7, 12, 20] and timetabling [5].

In this paper, we deal with the reconstruction of binary images without interior black holes. A hole can be considered as an individual 0 (black pixels) surrounded by 1s (white pixels). In particular, our research objective is to develop a tabu search approach to reconstruct a binary image with a minimal number of black holes. As far as the authors know, no one has so far published research on the reconstruction of such images.

The remainder of this paper is organized as follows. In Section 2, we introduce some definitions and notation. In Section 3, we propose a tabu search approach. In Section 4, we present and discuss the numerical results.

2. Preliminaries

Throughout this paper, we assume that the binary images are of size $m \times n$. Given an $m \times n$ binary image $A = [a_{i,j}]$, the horizontal projection of $A$ is the vector $H = (h_1, \ldots, h_m)$ such that $h_i = \sum_{j=1}^{n} a_{i,j}$ is the number of white cells on row $i$. The vertical projection of $A$ is defined analogously as the vector $V = (v_1, \ldots, v_n)$ where $v_j = \sum_{i=1}^{m} a_{i,j}$ is the number of white cells on column $j$. Both projections $H$ and $V$ constitute the orthogonal projections of $A$ (see Fig. 1). We denote by $BM(H,V)$ the class of all binary images satisfying the orthogonal projections $H$ and $V$. The condition $\sum_{i=1}^{m} h_i = \sum_{j=1}^{n} v_j \ (C_0)$ is obviously necessary for the existence of a binary image satisfying both projections.

![Figure 1. A binary image with horizontal projection $H = (5, 3, 5, 6, 4)$ and vertical projection $V = (2, 1, 3, 3, 2, 2, 3, 4)$.](image)

The reconstruction of a black and white image from horizontal and vertical line sums is stated as follows:

**Reconstruction Binary Image: **RBI($H,V$)

**Given:** $H = (h_1, \ldots, h_m) \in \mathbb{N}^m$ and $V = (v_1, \ldots, v_n) \in \mathbb{N}^n$.

**Goal:** Construct an $m \times n$ binary image that satisfies $H$ and $V$, i.e., row $i$ has exactly $h_i$ black cells and column $j$ has exactly $v_j$ black cells.
The number of binary images in the class $BM(H, V)$ is very large [19]. The definition of switching components is an essential concept to describe the characteristic of the class $BM(H, V)$.

**Definition 2.1.** A switching component of a binary image is a $2 \times 2$ sub image of the form \[
\begin{pmatrix}
1 & 0 \\
0 & 1
\end{pmatrix}
or \[
\begin{pmatrix}
0 & 1 \\
1 & 0
\end{pmatrix}.
\]
A switching operation consists in interchanging the 0’s and 1’s of a switching component.

We note that a switching operation preserves the orthogonal projections of images. The following theorem is the starting point for the meta-heuristics of the reconstruction of binary images under various constraints [18] since it shows that the graph of the neighborhood is connected.

**Theorem 2.2.** [18] Let $A$ and $B$ in $BM(H, V)$, $B \neq A$. Then, $A$ is transformable into $B$ (or vice versa) by finite sequences of switching operations.

**Definition 2.3.** For a binary image a cell $(i, j)$ is a hole if $(i, j)$ is a black pixel surrounded by 4 white pixels.

![Figure 2. Illustration of a black hole pixel.](image)

We say that a binary image fulfills the non-hole constraint if none of its black pixels is a hole. We denote $RNH(H, V)$ the problem of reconstructing an $m \times n$ binary image without hole pixels from its horizontal $H = (h_1, \ldots, h_m)$ and vertical $V = (v_1, \ldots, v_n)$ projections. The problem $RNH(H, V)$ can be defined as follows:

**Reconstruction of Non-Hole Binary Image: $RNH(H, V)$**

**Given:** $H = (h_1, \ldots, h_m) \in \mathbb{N}^m$ and $V = (v_1, \ldots, v_n) \in \mathbb{N}^n$.

**Goal:** Construct an $m \times n$ binary image without hole that satisfies the row and column sums $H$ and $V$.

In this paper, we are mainly interested in the problem $RNH(H, V)$, i.e., we consider the four neighbors of each pixel.

3. Tabu search approach of non interior hole binary images

We develop a tabu search algorithm for approximating binary images without interior holes.

3.1. Initial solution

We use Ryser’s classical algorithm $O(mn + \max(m \log n, n \log m))$ to reconstruct an initial solution satisfying the orthogonal projections. This polynomial time algorithm can be described as follows:
Ryser’s algorithm

Input: \{ \( H = (h_1, ..., h_m) \), \( V = (v_1, ..., v_n) \) two nonnegative integer vectors\}.
Output: \{ \( A \in BM(H, V) \) \}.

For \( i = 1 \) to \( n \) do
   If (the number of available rows \( \geq v_j \)) then
      Set \( v_j \) ones in priorities row;
      For each one placed in row \( i \) do: \( h_i \leftarrow h_i - 1 \);
   Else no solution;
End For

In each step, row ‘i’ is called available if \( h_i > 0 \)

3.2. Evaluation criterion

The objective function consists in minimizing the number of holes. The following function \( F_{NH} \) counts the number of holes in image \( A \):

\[
F_{NH}(A) = \sum_i \sum_j (1 - a_{i,j}) * a_{i-1,j} * a_{i,j-1} * a_{i+1,j} * a_{i,j+1}
\]

3.3. Neighborhood structure

An admissible solution is a binary image satisfying the orthogonal projections \( (H, V) \) i.e., we relax the hole constraint in the definition of feasible solutions. We define the neighborhood of a solution as the set of images obtained by a single switching operation. Recall that a switching operation preserves the orthogonal projections. Fig. 4 illustrates an example of the switching operation.

This definition of neighborhood induces a graph \( G(N, E) \) where the nodes of \( G \) correspond to the binary images of \( BM(H, V) \) and the edge set \( E \) represents the neighborhood relation: \( (A, B) \in E \) if and only if the binary images \( A \) and \( B \) are neighbors. We develop the following algorithm to select the best neighbor of the current solution according to the objective function:

Procedure for best neighbor solution
Input: \{ S \text{ a current solution } (S \in BM(H, V)) \}.

Output: \{ S' \text{ a best neighbor binary image of } S (S' \in BM(H, V)) \}.

Repeat

Select a switching structure: \{ (i, j), (i, k), (h, j), (h, k) \}

Apply switching operation;

Evaluate \( S' \);

If \( (F_{NH}(S') \leq F_{NH}(S)) \)

Update the best neighbor obtained till now;

End If

Until no switches can be found;

3.4. TS algorithm

The parameters of the tabu search algorithm are defined as follows:

- **Neighborhood move**: The best neighbor binary image with the minimum holes which is tabu is selected as the new current solution. If all neighbors binary images are tabu, then the oldest binary image solution is selected as current solution.

- **Tabu list and updating**: The length of the tabu list is 5 binary images and tabu list is updated after each neighborhood switching operation.

- **Termination criterion**: When the best solution has not improved for the last 100 iterations or when the total number of iterations reaches 10000.

The tabu search algorithm of reconstructing non hole binary images \( RNH(H, V) \) is the following:

**Tabu search algorithm**

Input: \{binary image (Ryser solution)\}.

Output: \{nearly non hole binary image respecting \( H \) and \( V \)\}.

While \(((total \leq 10000) \text{ and } (Nbr \leq 100)) \) do

Apply the procedure of the best neighbor solution;

Update the tabu list;

Update total and Nbr;

End While
The variable \( Nbr \) contains the number of iterations since the last improvement of the objective function. The variable \( total \) counts the number of iterations since the beginning of the algorithm.

4. Numerical experiments

We have implemented our algorithm in language C, using the gcc compiler. All our experiments were run on an AMD Athlon XP-M 1.7 GHz PC with 512 Mb of memory.

In order to compare the relative efficiency of Ryser’s algorithm and the tabu search approach for reconstructing matrices, we have used a large set of random binary images of various sizes. We generate a large set of square binary images with size varying from \( 40 \times 40 \) to \( 200 \times 200 \). For each size, we generate three classes of binary images (A, B, and C). For each class A, B, and C, we generate binary images with 10\%, 20\%, and 40\% of black pixels, respectively. For each class, we take the average of 10 instances for each size. The results of computational experiments are summarized in Table 1 for class A, Table 2 for class B, and Table 3 for class C. In these tables, the sub-column (hole cell) contains the number of holes provided by each method. The sub-column labeled (Time) indicates the running time (in seconds) required by each method.

We note that the tabu search algorithm outperforms the classical approach of Ryser’s even if the running time is slightly longer. In fact Ryser’s algorithm can be regarded as a single iteration tabu search.

<table>
<thead>
<tr>
<th>Image</th>
<th>Ryser Solution</th>
<th>T.S. Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% hole</td>
<td>Time</td>
</tr>
<tr>
<td>40 × 40</td>
<td>2.01%</td>
<td>0.00</td>
</tr>
<tr>
<td>60 × 60</td>
<td>2.47%</td>
<td>0.01</td>
</tr>
<tr>
<td>80 × 80</td>
<td>2.23%</td>
<td>0.01</td>
</tr>
<tr>
<td>100 × 100</td>
<td>1.95%</td>
<td>0.01</td>
</tr>
<tr>
<td>120 × 120</td>
<td>1.95%</td>
<td>0.15</td>
</tr>
<tr>
<td>140 × 140</td>
<td>2.12%</td>
<td>0.15</td>
</tr>
<tr>
<td>160 × 160</td>
<td>2.22%</td>
<td>0.15</td>
</tr>
<tr>
<td>180 × 180</td>
<td>2.07%</td>
<td>0.24</td>
</tr>
<tr>
<td>200 × 200</td>
<td>2.13%</td>
<td>0.31</td>
</tr>
</tbody>
</table>
Table 2. Reconstruction results for TS algorithm (Class B).

<table>
<thead>
<tr>
<th>Image</th>
<th>Ryser Solution</th>
<th>T.S. Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% hole</td>
<td>Time</td>
</tr>
<tr>
<td>40 × 40</td>
<td>8.00%</td>
<td>0.00</td>
</tr>
<tr>
<td>60 × 60</td>
<td>9.88%</td>
<td>0.01</td>
</tr>
<tr>
<td>80 × 80</td>
<td>8.93%</td>
<td>0.01</td>
</tr>
<tr>
<td>100 × 100</td>
<td>7.80%</td>
<td>0.02</td>
</tr>
<tr>
<td>120 × 120</td>
<td>7.80%</td>
<td>0.16</td>
</tr>
<tr>
<td>140 × 140</td>
<td>8.51%</td>
<td>0.15</td>
</tr>
<tr>
<td>160 × 160</td>
<td>6.22%</td>
<td>0.15</td>
</tr>
<tr>
<td>180 × 180</td>
<td>6.07%</td>
<td>0.22</td>
</tr>
<tr>
<td>200 × 200</td>
<td>6.13%</td>
<td>0.33</td>
</tr>
</tbody>
</table>

Table 3. Reconstruction results for TS algorithm (Class C).

<table>
<thead>
<tr>
<th>Image</th>
<th>Ryser Solution</th>
<th>T.S. Solution</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>% hole</td>
<td>Time</td>
</tr>
<tr>
<td>40 × 40</td>
<td>16.00%</td>
<td>0.00</td>
</tr>
<tr>
<td>60 × 60</td>
<td>19.77%</td>
<td>0.00</td>
</tr>
<tr>
<td>80 × 80</td>
<td>17.87%</td>
<td>0.02</td>
</tr>
<tr>
<td>100 × 100</td>
<td>15.60%</td>
<td>0.01</td>
</tr>
<tr>
<td>120 × 120</td>
<td>15.61%</td>
<td>0.16</td>
</tr>
<tr>
<td>140 × 140</td>
<td>17.02%</td>
<td>0.15</td>
</tr>
<tr>
<td>160 × 160</td>
<td>17.62%</td>
<td>0.15</td>
</tr>
<tr>
<td>180 × 180</td>
<td>16.61%</td>
<td>0.21</td>
</tr>
<tr>
<td>200 × 200</td>
<td>17.04%</td>
<td>0.32</td>
</tr>
</tbody>
</table>

5. Conclusion

We have proposed a tabu-search-based approach to reconstructing binary images without interior hole pixels from its horizontal and vertical projections. The minimization of hole black pixels has been used as the objective function. The results obtained by this metaheuristic are encouraging. Our methodology can easily be applied to other types of hole pixels.

References


Alain Billionnet, Laboratoire Cedric-CNAM, 292 rue St-Martin, Paris, France
e-mail: alain.billionnet@ensiie.fr

Fethi Jarray, Higher Institute of Computer Science of Medenine, Tunisia
e-mail: fethi.jarray@yahoo.fr

Ghassen Tlig, Laboratoire LIMTIC, Institut Supérieur d’Informatique, Université de Tunis El Manar and Higher Institute of Computer Science of Mednine, Tunisia
e-mail: ghassen.tlik@gmail.com

Ezzedine Zagrouba, Laboratoire LIMTIC, Institut Supérieur d’Informatique, Université de Tunis El Manar, Tunisia
e-mail: ezzeddine.zagrouba@fsm.rnu.tn