Binary Tomography Using Geometrical Priors: Uniqueness and Reconstruction Results

Péter Balázs

Department of Image Processing and Computer Graphics
University of Szeged

June 2007

A DISSERTATION SUBMITTED FOR THE DEGREE OF DOCTOR OF PHILOSOPHY OF THE UNIVERSITY OF SZEGED

University of Szeged
Doctoral School in Mathematics and Computer Science
Ph.D. Program in Informatics

Preface

Computerized tomography (CT) is a diagnostic procedure for obtaining the density distribution within the human body from its x-ray projections, which makes it possible to recognize disorders within certain human organs. In the last few decades applications of CT have spread to a wide area beyond medical imaging. These include fields of industry, biology, physics and chemistry. In most of these new applications the number of the possible density values of the object in investigation is small, but there is a practical limitation that only a few projections of the object can be made. In contrast, in medical imaging the density values can vary across a wide range and the number of available projections is usually a few hundred. Due to the very limited number of projections in the new applications the usual classical reconstruction methods of CT do not work that well.

Discrete tomography (DT) investigates how, with prior knowledge, we can reconstruct an object. The reconstruction should only contain a few values that can be exploited to eliminate the problems arising from using a small number of available projections. A more restrictive but still very common case is when the density values can only be elements of the set $\{0, 1\}$. This dissertation is concerned with this latter field called binary tomography (BT) and starts out by introducing the main problems and applications of this area and connections of BT with other mathematical fields. In the subsequent chapters we will study the problems of uniqueness and reconstruction when two, three or four projections are available. The several theoretical results presented in this dissertation are also interesting in their own right, but the insight that we can gain from these results on the mathematical behaviour of BT might be more important and far-reaching. Of course, an understanding of the theory of BT is necessary in designing efficient reconstruction algorithms that can be applied in practice.

The first of my **acknowledgements** goes to my supervisors, Prof. János Csirik for supporting my research and letting me work at the Department of Computer Algorithms and Artificial Intelligence, and Prof. Attila Kuba for introducing me to the fascinating world of discrete tomography, and inspiring me to do my research in this field. I cannot express the deep sorrow that I feel of Attila's early death, which suddenly interrupted our common work on several interesting problems of DT. My thanks also goes to all my colleagues at the Institute of Informatics who helped me by giving good advice (even when my problems were of a non-scientific nature). I am very grateful to Eörs Máté and my brother, László Balázs for carefully checking the chapters and making some valuable comments. I would also like to express my gratitude to David P. Curley for scrutinizing and correcting this thesis from a linguistic point of view. I wish to thank my mother, my sister, my brother and my girlfriend Kátya for all the love they have given me. Finally, I also would like to thank the many friends, especially Lajos Fröhlich, László Tóth and László Márk for the all-night-long conversations and for all the beers we had together.

My work was partially supported by grants OTKA T032241, OTKA T048476 and NSF DMS 0306215.

Péter Balázs, June 2007.

Notations

$\hat{F} = (\hat{f}_{ij})_{m\times n}$	the binary matrix that represents the discrete set F of size $m \times n$
$\mathcal{H}(F) = (h_1, \ldots, h_m)$	the horizontal projection of the discrete set F of size $m \times n$
$\mathcal{V}(F) = (v_1, \ldots, v_n)$	the vertical projection of the discrete set F of size $m \times n$
$\mathcal{D}(F) = (d_1, \ldots, d_{m+n-1})$	the diagonal projection of the discrete set F of size $m \times n$
$\mathcal{A}(F) = (a_1, \ldots, a_{m+n-1})$	the antidiagonal projection of the discrete set F of size $m \times n$
$\widetilde{\mathcal{H}}(F)$, $\widetilde{\mathcal{V}}(F)$, $\widetilde{\mathcal{D}}(F)$, $\widetilde{\mathcal{A}}(F)$	the cumulated vectors of the discrete set F
$\mathcal{F}$	the class of all discrete sets
$\mathcal{HV}$	the class of hv-convex discrete sets
$\mathcal{S}_4$	the class of hv-convex 4-connected discrete sets
$\mathcal{S}_8$	the class of hv-convex 8-connected discrete sets
$\mathcal{S}'_8$	the class of hv-convex 8-connected discrete sets which have two or more components, i.e. $\mathcal{S}'_8 = \mathcal{S}_8 \setminus \mathcal{S}_4$
$\mathcal{Q}'$	the class of discrete sets that are Q-convex along the horizontal and vertical directions and have two or more components
$\mathcal{DCP}^{NE}_{(a,b)}$	the class of NE-directed polyominoes that are convex along the direction (a, b)
$\mathcal{DEC}$	the class of decomposable discrete sets
$\mathcal{DEC}_{\mathcal{C}}$	the class of decomposable discrete sets which have components from the class $\mathcal{C}$
$\mathcal{S}^{NW}$	the class of decomposable discrete sets of NW type
$\mathcal{S}^{NE}$	the class of decomposable discrete sets of NE type
$\mathcal{L}_2$	the set of directions $\{(1,0),(0,1)\}$
$\mathcal{L}_4$	the set of directions $\{(1,0),(0,1),(1,1),(-1,1)\}$

List of Figures

2.1 A discrete set represented by its elements (left), a binary picture (center) and a binary matrix (right) . . . 6
2.2 The projection of a discrete set in direction $v = (1, 2)$ . . . 7
2.3 Formulating a reconstruction problem as a linear equation system . . . 13
2.4 A polyomino (a), an hv-convex polyomino (b), an hv-convex 8- but not 4-connected discrete set (c), and a general hv-convex discrete set (d) . 14
2.5 A Q-convex (left), and a non-Q-convex discrete set (right) with the four quadrants around the point P (denoted by a black dot) . . . 16

3.1 Two directed hv-convex polyominoes which have the same horizontal and vertical projections . . . 18
3.2 Proof of Lemma 3.1.2 . . . 19
3.3 A discrete set that is neither 4-connected nor diagonally convex, but it can be reconstructed by steps 1-3 of Algorithm DCP . . . 20
3.4 Exponentially many polyominoes of $\mathcal{DCP}^{NE}_{(-1,1)}$ with the same horizontal and vertical projections. (a)-(d) Proof of Theorem 3.3 for the case $k = 2$. The sets from left to right are $F_2^{\{\}}$, $F_2^{\{1\}}$, $F_2^{\{2\}}$, and $F_2^{\{1,2\}}$, respectively. Elements of the switching components with a value of 1 are denoted by black dots. (e) Another set with the same projections . 22

4.1 Proof of Lemma 4.1.1. The set S is denoted by a grey square . . . 26
4.2 A discrete set of $\mathcal{Q}'$ of NW type (left) and of NE type (right) . . . 27
4.3 A discrete set $F \in \mathcal{Q}'$ of NW type (left) and a discrete set $F' \in \mathcal{Q}'$ of NE type (right). The elements of C_F and $C_{F'}$ are depicted by white dots 28
4.4 Six sets of $\mathcal{Q}'$ with the same horizontal and vertical projections . . . 31
4.5 An example which shows how Algorithm 2-RECQ' works . . . 32
4.6 Proof of Lemma 4.4.1 . . . 34
4.7 Two sets of $\mathcal{Q}'$ with an empty row and column and the same horizontal and vertical projections . . . 36

5.1 The gluing operators: NW, NE, SE, and SW gluings (from left to right) 40
5.2 Configurations of the SCDRs of four components F_1, F_2, F_3, and F_4 which satisfy property (γ) (first row), and do not satisfy it (second row) 41
5.3 An example showing that the converse of Lemma 5.1.3 does not hold . 44

5.4 (a) A discrete set which has no NW component although the $(5, 4)$ position satisfies the conditions of Theorem 5.1 with the polyomino in (b) 45

5.5 An example which shows how Algorithm 4-RECDEC works. The located discrete rectangles – which must contain the next component – are denoted by bold squares. The column assumed to be a column of $C(F)$ is represented with slanted lines 49

5.6 Three different decomposable discrete sets with the same projections . 49

5.7 A Q-convex (a), and an hv-convex but non-Q-convex discrete set (b) with the four quadrants around the point P (denoted by a black dot) . 51

5.8 An hv-convex decomposable discrete set 53

5.9 (a) An hv-convex discrete set which is possibly reconstructible by Algorithm 4-RECDEC and (b) an hv-convex discrete set with the same projections but with different components. This shows that the set in (a) does not satisfies property (α) 54

5.10 Two discrete sets that are not decomposable but have decomposable configurations 56

6.1 The relative position of the minimal bounding rectangles of the components $B_1, \ldots, B_k$ in the $\mathcal{S}'$ class 65

6.2 An example which shows the connection between elements of the $\mathcal{HV}$ and $\mathcal{S}$ classes 68

6.3 All the hv-convex discrete sets of size 2×3. The numbers tell us that there are other solutions which can be obtained by mirroring or/and rotating the given set 68

6.4 Some hv-convex binary pictures with a perimeter value of 10. The numbers tell us that there are other solutions that can be obtained by mirroring or/and rotating the given polyomino 72

6.5 The distribution of the number of components in the test dataset (solid lines) and the corresponding normal distribution (dashed lines) for sets of $\mathcal{S}'$ of sizes 200×200 (left) and 500×500 (right) 75

6.6 The distributions of the number of components – which depend on the size of the test data – in the $\mathcal{HV}'$ ((a)-(f)) and $\mathcal{HV}$ ((g)-(j)) classes . . 76

7.1 The connections between the classes studied in this dissertation 80

List of Tables

1.1 The connection between the thesis points and the corresponding publications 4

4.1 Average execution times in seconds of Algorithm 2-REC8' and Algorithm C in [11], which both depend on the size of the discrete set 36

5.1 The number of hv-convex polyominoes in the test datasets that are not uniquely determined by two, three, or four projections 58
5.2 Average execution times for reconstructing all hv-convex polyominoes with the same horizontal and vertical projections, which depends on the size of hv-convex polyomino. The values were taken from [11] 59
5.3 Accuracy and average running time of Algorithm 4-RECHV on the test datasets 60

6.1 The values of $P(n)$, $S'(n)$, $S(n)$, $HV'(n)$, and $HV(n)$ 71
6.2 The number of components of 1000 sets generated from the $\mathcal{S}'$ class (top) and $\mathcal{S}$ class (bottom) using uniform random distributions 74
6.3 The expectation value $E_{\mathcal{HV}'}(n)$ $(E_{\mathcal{HV}}(n))$ and the variance $D^2_{\mathcal{HV}'}(n)$ $(D^2_{\mathcal{HV}}(n))$ of the components of a set with a minimal bounding rectangle of size $n \times n$ in the $\mathcal{HV}'$ $(\mathcal{HV})$ class. The values have been rounded to 5 digits 75

A.1 The connection between the thesis points and the corresponding publications 84

B.1. A tézispontok és a Szerző publikációinak viszonya 88

Contents

Notations v

1 Introduction 1
1.1 Summary by Chapters 2
1.2 Summary by Results 3

2 Binary Tomography: Definitions, Basic Problems and Applications 5
2.1 Computerized, Discrete, and Binary Tomography 5
2.2 Definitions and Basic Problems 6
2.3 Applications of Binary Tomography 9
2.3.1 Data Compression, Data Security 9
2.3.2 Image Processing 10
2.3.3 Electron Microscopy 10
2.3.4 Angiography 10
2.3.5 Computerized Tomography 11
2.3.6 Non-Destructive Testing 11
2.4 Related Fields 11
2.4.1 Analysis of Measurable Plane Sets 11
2.4.2 Convex Geometry 12
2.4.3 Linear Algebra, Matrix Theory 12
2.4.4 Graph Theory, Network Flows 12
2.4.5 Linear Programming, Combinatorial Optimization 12
2.5 Discrete Sets with Geometrical Properties 13

3 Uniqueness and Non-Uniqueness Results for Directed Polyominoes 17
3.1 Diagonally Convex Directed Polyominoes 18
3.2 Antidiagonally Convex Directed Polyominoes 20
3.3 Generalisation to Arbitrary Line-Convexity 22
3.4 Summary 23

4 Reconstruction of Q-Convex non-4-Connected Discrete Sets 25
4.1 Components of Q-Convex Sets 25
4.2 Locating the Components 27
4.3 The Reconstruction Algorithm 30
4.4 Connection with the $\mathcal{S}'_8$ Class 32

4.4.1 Components of a Set of $\mathcal{S}_8'$ 33
4.4.2 Experimental Results 36
4.5 Summary 36

5 Reconstruction from Four Projections 39
5.1 Decomposability: A New Property for Reconstruction 39
5.1.1 The Center of a Decomposable Discrete Set 41
5.1.2 Finding a Component 42
5.1.3 The Reconstruction Algorithm 46
5.2 The Connection with Q-Convexes 51
5.3 Reconstructing hv-Convex Discrete Sets from Four Projections 52
5.3.1 Reconstruction of hv-Convex Decomposable Discrete Sets . . . 52
5.3.2 Three Projections: A Negative Result 53
5.3.3 A Heuristic for Reconstructing hv-Convex Discrete Sets with Decomposable Configurations 55
5.4 Conclusions and Discussion 60

6 Random Generation of hv-Convex Discrete Sets 63
6.1 Generation of Special hv-Convex Discrete Sets 64
6.2 Generation of hv-Convex Discrete Sets 67
6.3 Possible Generalisations 69
6.4 Statistics on hv-Convex Discrete Sets 71
6.4.1 The Number of hv-Convex Discrete Sets 71
6.4.2 The Number of Components 73
6.5 Conclusions and Discussion 77

7 Conclusions 79

Appendices 81

Appendix A Summary in English 81
A.1 Summary by Chapters 82
A.2 Key Points of the Thesis 83

Appendix B Summary in Hungarian 85
B.1. A fejezetek áttekintése 86
B.2. Az eredmények tézisszerű összefoglalása 87

Bibliography 89

Chapter 1

Introduction

"Not knowing ourselves - can be weighed up
by rereading what we have written"
Paul Valéry

Binary tomography (BT) is a technique which seeks to reconstruct a binary function defined over the two-dimensional integer lattice $\mathbb{Z}^2$ from its line-sums along several directions. In the terminology of BT the function is often called the discrete set, while the line-sums are referred to as projections of the discrete set. The main challenge in BT is that practical limitations every time reduce the number of available projections to at most about ten (usually far fewer) – which results in ambiguous reconstruction, i.e. the number of possible solutions of the same reconstruction task can be extremely large. This can cause the reconstructed discrete set to be quite unlike the original one. In addition, the reconstruction problem can be NP-hard, depending on the number and directions of the projections. One way of eliminating these problems is to use meta-heuristics (like simulated annealing [76], and genetic algorithms [77]) to find possibly good but not necessarily exact solutions. Another strategy is to suppose that we have some prior knowledge of the set to be reconstructed so we can reduce the search space of the possible solutions. It can be assumed that the discrete set arises from a class with a certain known distribution [73; 75] or it has some geometrical properties. Incorporating geometrical knowledge into the reconstruction is an intensively studied field of BT with numerous surprising theoretical results and interesting applications.

This dissertation summarizes the author's research results in reconstructing discrete sets that have certain geometrical properties. The main motivation behind this work was to extend already known reconstruction algorithms to broader classes of discrete sets and to find new properties that can guarantee fast and less ambiguous reconstruction. Always great care is taken to investigate how the new results relate to the former ones in order to give an overview of the field of discrete reconstruction using geometrical priors. I think that the investigation of certain geometrical properties – whether or not they can facilitate the reconstruction – can lead to a better understanding of the problems present in discrete tomography. It seems to me now that it might be the key to designing fast and accurate reconstruction systems which can be applied to real-world problems.

1.1 Summary by Chapters

Chapter 2 does not contain scientific contributions from the author but it does try to provide an overview of the field of binary tomography. This chapter will present the basic definitions and notations of the field that will be used in the subsequent chapters. Later it will provide an overview of the main problems and applications of binary tomography and its mathematically related fields.

The remaining chapters roughly follow an order of solving first simpler, then evermore complicated problems. Chapter 3 investigates the problem of reconstructing directed convex polyominoes from two projections. Although in this case the restrictions for the set to be reconstructed are quite strong, the chief difficulties of the reconstruction task can still be examined in this case. This chapter also introduces some mathematical tools and results that will be useful in the later chapters where more difficult reconstruction problems will be studied.

Chapter 4 presents an algorithm for reconstructing Q-convex discrete sets which have at least two components from two projections. The relation between the classes of Q-convexes and hv-convex 8-connected discrete sets will also be investigated. The reconstruction algorithm that we developed outperforms the best-known ones for these types of problem. The mathematical results and the algorithm described in this chapter form the basis of the reconstruction algorithms presented in the subsequent chapters.

Chapter 5 describes a decomposition technique that can reconstruct all the discrete sets which satisfy some specific properties and have the same four projections. In fact, this algorithm is one of the first that can perform an exact reconstruction from four projections in polynomial time. The technique presented here is based on the results of the previous chapters, but some other technical theorems and lemmas are also needed. The effectiveness of this algorithm is examined by describing its worst-case performance on several previously studied classes. We show that for each studied class the use of two additional projections can considerably speed up the reconstruction and reduce the number of solutions. A negative result is also given to show that three projections are insufficient to achieve this result. The main strength of the decomposition technique is that it can be applied to various classes of discrete sets by introducing some minor modifications on the decomposition framework. In the final part of this chapter we show how to modify our framework to achieve a fast and accurate reconstruction heuristic using four projections for the class of hv-convexes.

Comparing reconstruction algorithms working on the same class from the viewpoint of average accuracy or running time makes it possible for us to examine the pros and cons of using particular algorithms. Keeping our findings in mind, more efficient reconstruction algorithms can be designed. In Chapter 6 we introduce a method by which hv-convex discrete sets can be generated using a uniform random distribution. Generating benchmark sets with this method, the efficiency of several reconstruction algorithms can be compared on the class of hv-convexes. Some statistics in this chapter are also given that are of great help in analysing the performance of reconstruction algorithms developed for this class. Besides this, we also show that the generation method can be adapted to various classes of discrete sets that have disjoint components.

1.2 Summary by Results

In the following a listing of the most important results of the dissertation is given. Table 1.1 shows which thesis point is described in which publication by the author.

I.) Horizontally or vertically convex NE-directed polyominoes can be reconstructed from their horizontal and vertical projections uniquely in polynomial time. The author investigated how the change of the direction of convexity influences the above result. He found that it can be extended for diagonally convex NE-directed polyominoes as well, but assuming convexity along any other (that is, not horizontal, vertical or diagonal) direction on the NE-directed polyomino to be reconstructed can cause the number of solutions of the same reconstruction task to become exponentially large.

II.) The author examined the problem of reconstruction in the class of discrete sets that are Q-convex along the horizontal and vertical directions and have at least two components. He developed an algorithm that reconstructs sets of this class from the horizontal and vertical projections in polynomial time. The algorithm finds all solutions of a given reconstruction task.

III.) The author showed that the class of hv-convex 8-connected discrete sets is a subclass of the class of Q-convexes. Then he compared his algorithm on the class of hv-convex 8-connected sets to previously published ones and found that the new algorithm outperforms the former ones from the viewpoint of the worst case or the viewpoint of the average execution time complexity. He also showed that for Q-convex but not 8-connected sets his algorithm can be speeded up and in this case the number of possible solutions of the same reconstruction problem is at most two.

IV.) The author introduced a new property of discrete sets called decomposability and gave a polynomial-time algorithm for reconstructing every decomposable discrete sets that has the same four projections. He also showed that three projections are usually not sufficient to achieve the above result. Then he investigated the relationship between the classes of decomposables and Q-convexes and demonstrated its consequences on the reconstruction complexity for some well-known classes when four projections are accessible.

V.) The author investigated the possibility of extending the decomposition technique to the class of hv-convexes. He gave a reconstruction heuristic for hv-convex sets with decomposable configurations. He then developed a test environment to demonstrate that the heuristic locates the solutions very quickly and with good accuracy.

VI.) The efficiency of newly developed exact or heuristical reconstruction algorithms is often tested on the class of hv-convexes. The author described a method for generating elements of this class using a uniform random distribution where

	[2]	[3]	[5]	[6]	[7]	[8]
I.	•					
II.		•				
III.		•	•			
IV.		•	•	•		
V.					•	
VI.						•

Table 1.1: The connection between the thesis points and the corresponding publications

an exact comparison of several reconstruction algorithms is possible from the viewpoint of accuracy or average execution time. The generation method can readily be extended to other classes of discrete sets that have disjoint components. Using this method the author presented several statistics on hv-convex sets that can affect the accuracy or the complexity of the reconstruction.

Chapter 2

Binary Tomography: Definitions, Basic Problems and Applications

2.1 Computerized, Discrete, and Binary Tomography

Computerized tomography was originally a method of diagnostic radiology that was used to obtain the density distribution within the human body based on x-ray projection samples. From a mathematical point of view it seeks to determine an unknown function $f(x, y, z)$ defined over the three-dimensional (3D) Euclidean space $\mathbb{E}^3$ from weighted integrals over subspaces of $\mathbb{E}^3$ (called projections). In this form we face two problems. First, the values of f can vary over a wide range so a huge number of projections are needed to ensure an accurate reconstruction of it. Second, the projections may be known only approximately thus only an approximate determination of f may be possible.

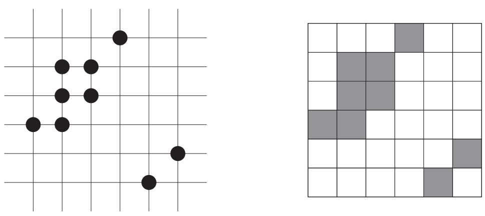
There are other situations where we want to reconstruct an object but it is not possible to examine the object itself as only its projections are available. For example, in industrial applications like non-destructive testing or reverse engineering we normally cannot investigate the interior of an industrial object itself. However, its x-ray projections can be measured so one can make assumptions about the shape of the object and the material(s) it is made of. Other examples arise from the field of electron microscopy where the task is to identify biological macromolecules composed of ice, protein, and RNA or to determine crystalline structures. In most of these applications there is a restriction that only a small number of projections can be taken, thus methods of computerized tomography usually cannot approximate the function f that well. However, there is a hope that f can be determined from just a small number of projections too, since we have some prior knowledge here. In these applications the range of f is discrete and consists of only a small number of possible values. This leads us to the field of *discrete tomography* where it is assumed that only a handful of projections are available, but the prior information about the possible values of f can be used. Clearly, this is very different case from computerized tomography and so discrete tomography needs its own special mathematical tools and theory.

A more restricted situation is when f can only take the values 0 or 1. This kind of tomography is called *binary tomography*, which also has a variety of applications. For example, in electron microscopy 0 and 1 can represent the absence and presence of a certain atom in the crystalline structure, respectively. Similarly, in angiography the values 0 and 1 can describe the absence or presence of a contrast agent in heart chambers or in segments of blood vessels.

Although tomographical problems can also be investigated in higher dimensions, in fact almost all the 3D applications use two-dimensional (2D) slices of the object being studied, so reducing a 3D tomography problem to a 2D one. After the reconstruction the 2D slices are integrated together to produce a 3D object. In this chapter we first give an overview of the basic definitions and problems of 2D binary tomography. Then we present several interesting situations where binary tomography can be used. After, we discuss the connections between binary tomography and different fields of mathematics.

2.2 Definitions and Basic Problems

The finite subsets of the 2D integer lattice $\mathbb{Z}^2$ are called *discrete sets*. The *size* of a discrete set is the size of its minimal bounding discrete rectangle. In this dissertation, unless we state otherwise we will assume that the size of the discrete set F under investigation is $m \times n$. A discrete set is defined up to a translation and it can be represented by a binary picture formed from unitary cells or by a binary matrix $\hat{F} = (\hat{f}_{ij})_{m\times n}$. To be consistent with the corresponding matrix representation we shall assume that the vertical axis of the 2D integer lattice is top-down directed and that the upper left corner of the minimal bounding rectangle of F is the position $(1, 1)$. For the same reason we will refer to any element of a discrete set by its matrix position (i.e. not by the position of the element in the 2D integer lattice). Figure 2.1 shows the three most commonly used representations of a discrete set.



$$\begin{bmatrix} 0&0&0&1&0&0\\ 0&1&1&0&0&0\\ 0&1&1&0&0&0\\ 1&1&0&0&0&0\\ 0&0&0&0&0&1\\ 0&0&0&0&1&0 \end{bmatrix}$$

Figure 2.1: A discrete set represented by its elements (left), a binary picture (center) and a binary matrix (right)

A *lattice direction* v is represented by a non-zero vector $(a, b) \in \mathbb{Z}^2$ such that a and b are coprimes. Without loss of generality we will assume that $b \geq 0$, and $b = 0$ only if $a = 1$. A *lattice line* in direction v is a line in the 2D Euclidean space $\mathbb{E}^2$ that is parallel to v and passes through at least one point of $\mathbb{Z}^2$. Now let us denote the set of lattice lines in direction v by $\mathcal{L}^{(v)}$. Then the *projection* of the discrete set F in direction v is defined as the function $\mathcal{P}_F^{(v)} : \mathcal{L}^{(v)} \to \mathbb{N}_0$ where $\mathcal{P}_F^{(v)}(\ell) = |F \cap \ell|$ for each $\ell \in \mathcal{L}^{(v)}$. Projections of a discrete set outside its minimal bounding discrete rectangle always have a value of 0 so they are not interesting in this study. Figure 2.2 shows some lattice lines and the projection of a discrete set in direction $v = (1, 2)$. Recall that the vertical axis of $\mathbb{Z}^2$ is top-down directed.

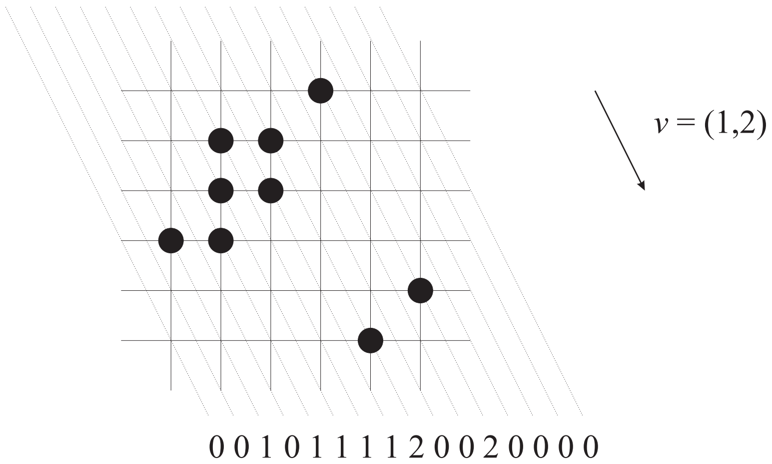


Figure 2.2: The projection of a discrete set in direction $v = (1, 2)$

The class that includes all the discrete sets will be denoted by $\mathcal{F}$. Given an arbitrary class $\mathcal{G} \subseteq \mathcal{F}$ of discrete sets we say that the discrete set $G \in \mathcal{G}$ is *uniquely determined* in the class $\mathcal{G}$ (with respect to some projections) if there is no other discrete set $G' \in \mathcal{G}$ with the same projections. Now three main problems arise in discrete tomography. Given an arbitrary class of discrete sets $\mathcal{G} \subseteq \mathcal{F}$ and an arbitrary set $\mathcal{L} = (v_1, \ldots, v_q)$ of q lattice directions these problems can be stated as follows.

Consistency($\mathcal{G}$, $\mathcal{L}$)

Instance: *For $k = 1, \ldots, q$ a function $p^{(k)} : \mathcal{L}^{(v_k)} \to \mathbb{N}_0$ with finite support.*

Task: *Decide whether there exists a discrete set $F \in \mathcal{G}$ such that $\mathcal{P}_F^{(v_k)} = p^{(k)}$ for $k = 1, \ldots, q$.*

Reconstruction($\mathcal{G}$, $\mathcal{L}$)

Instance: *For $k = 1, \ldots, q$ a function $p^{(k)} : \mathcal{L}^{(v_k)} \to \mathbb{N}_0$ with finite support.*

Task: *Construct a discrete set $F \in \mathcal{G}$ such that $\mathcal{P}_F^{(v_k)} = p^{(k)}$ for $k = 1, \ldots, q$.*

Uniqueness($\mathcal{G}$, $\mathcal{L}$)

Instance: *An $F \in \mathcal{G}$.*

Task: *Decide whether F is uniquely determined in the class $\mathcal{G}$ with respect to its projections in the directions of $\mathcal{L}$.*

Clearly, once the Reconstruction($\mathcal{G}$, $\mathcal{L}$) task has been solved the Consistency($\mathcal{G}$, $\mathcal{L}$) task is also solved for an arbitrary class of discrete sets $\mathcal{G} \subseteq \mathcal{F}$ and an arbitrary set of lattice directions $\mathcal{L}$.

In this thesis we will use two special sets of directions, namely $\mathcal{L}_2 = \{(1,0),(0,1)\}$ and $\mathcal{L}_4 = \{(1,0),(0,1),(1,1),(-1,1)\}$. For the sake of technical simplicity we introduce the notations $\mathcal{H}(F) = (h_1,\ldots,h_m)$, $\mathcal{V}(F) = (v_1,\ldots,v_n)$, $\mathcal{D}(F) = (d_1,\ldots,d_{m+n+1})$, and $\mathcal{A}(F) = (a_1,\ldots,a_{m+n-1})$ for the projections of F in the $(1,0)$, $(0,1)$, $(1,1)$, and $(-1,1)$ directions, respectively. We will refer to them as the *horizontal, vertical, diagonal*, and *antidiagonal projections* of F, respectively. That is

$$h_i = \sum_{j=1}^{n} \hat{f}_{ij}, \qquad i = 1,\ldots,m \; , \tag{2.1}$$

$$v_j = \sum_{i=1}^{m} \hat{f}_{ij}, \qquad j = 1,\ldots,n \; , \tag{2.2}$$

$$d_k = \sum_{(m-i)+j=k} \hat{f}_{ij}, \qquad k = 1,\ldots,m+n-1 \; , \tag{2.3}$$

$$a_k = \sum_{i+j=k+1} \hat{f}_{ij}, \qquad k = 1,\ldots,m+n-1 \; . \tag{2.4}$$

The *cumulated vectors* of a discrete set F are often useful in investigating the properties of F. They will be denoted by $\widetilde{\mathcal{H}}(F) = (\widetilde{h}_1,\ldots,\widetilde{h}_m)$, $\widetilde{\mathcal{V}}(F) = (\widetilde{v}_1,\ldots,\widetilde{v}_n)$, $\widetilde{\mathcal{D}}(F) = (\widetilde{d}_1,\ldots,\widetilde{d}_{m+n-1})$, and $\widetilde{\mathcal{A}}(F) = (\widetilde{a}_1,\ldots,\widetilde{a}_{m+n-1})$, and defined by the following formulas

$$\widetilde{h}_i = \sum_{l=1}^{i} h_l, \qquad i = 1,\ldots,m \; , \tag{2.5}$$

$$\widetilde{v}_j = \sum_{l=1}^{j} v_l, \qquad j = 1,\ldots,n \; , \tag{2.6}$$

$$\widetilde{d}_k = \sum_{l=1}^{k} d_l, \qquad k = 1,\ldots,m+n-1 \; , \tag{2.7}$$

$$\widetilde{a}_k = \sum_{l=1}^{k} a_l, \qquad k = 1,\ldots,m+n-1 \; . \tag{2.8}$$

Example 2.2.1

The horizontal, vertical, diagonal, and antidiagonal projections and the cumulated vectors of the discrete set F in Fig. 2.1 are given by the following

$$\mathcal{H}(F) = (1,2,2,2,1,1), \qquad \widetilde{\mathcal{H}}(F) = (1,3,5,7,8,9) \; ; \tag{2.9}$$

$$\mathcal{V}(F) = (1,3,2,1,1,1), \qquad \widetilde{\mathcal{V}}(F) = (1,4,6,7,8,9) \; ; \tag{2.10}$$

$$\mathcal{D}(F) = (0,0,1,0,2,2,2,1,1,0,0), \qquad \widetilde{\mathcal{D}}(F) = (0,0,1,1,3,5,7,8,9,9,9) \; ; \tag{2.11}$$

$$\mathcal{A}(F) = (0,0,1,4,2,0,0,0,0,2,0), \qquad \widetilde{\mathcal{A}}(F) = (0,0,1,5,7,7,7,7,7,9,9) \; . \tag{2.12}$$

In the coming chapters we will examine the tasks $\textsc{Reconstruction}(\mathcal{G},\ \mathcal{L})$ and $\textsc{Uniqueness}(\mathcal{G},\ \mathcal{L})$ for several classes of discrete sets using the direction sets $\mathcal{L}_2$ and $\mathcal{L}_4$. But before going further let us mention here some standard results.

To decide $\textsc{Consistency}(\mathcal{G},\ \mathcal{L}_2)$ in an arbitrary class $\mathcal{G} \subseteq \mathcal{F}$ the following simple necessary condition should be met

(i) $p_i^{(1)} \leq n$, for $1 \leq i \leq m$, and $p_j^{(2)} \leq m$, for $1 \leq j \leq n$;

(ii) $\sum_{i=1}^{m} p_i^{(1)} = \sum_{j=1}^{n} p_j^{(2)}$, i.e., the vectors $p^{(1)}$ and $p^{(2)}$ have the same total sums.

In this case we say that the vectors $p^{(1)}$ and $p^{(2)}$ are *compatible*. A similar condition can be given for the $\textsc{Consistency}(\mathcal{G},\ \mathcal{L}_4)$ task.

In [80] a necessary and sufficient condition was given for the $\textsc{Consistency}(\mathcal{G},\ \mathcal{L}_2)$ task. In the same paper the first algorithm was presented which solves the $\textsc{Reconstruction}(\mathcal{F},\ \mathcal{L}_2)$ task in $O(mn + n \log n)$ time. In addition a necessary and sufficient condition was presented for the $\textsc{Uniqueness}(\mathcal{F},\ \mathcal{L}_2)$ task. It was shown that the discrete set $F \in \mathcal{F}$ is uniquely determined w.r.t. its horizontal and vertical projections if and only if $\hat{F}$ does not contain a *switching component* which is a submatrix of $\hat{F}$ of the form

$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{or} \quad \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} . \tag{2.13}$$

This result is based on the observation that, when interchanging 0s and 1s in a switching component, the horizontal and vertical projections of F remain invariant.

In contrast to the previous positive results for the horizontal and vertical projections, it was shown that the $\textsc{Consistency}(\mathcal{F},\ \mathcal{L})$, $\textsc{Reconstruction}(\mathcal{F},\ \mathcal{L})$, and $\textsc{Uniqueness}(\mathcal{F},\ \mathcal{L})$ tasks are all NP-hard if $\mathcal{L} = \mathcal{L}_4$. In fact, these problems are intractable even if $\mathcal{L}$ is a set of three arbitrary directions [48].

2.3 Applications of Binary Tomography

Although binary tomography is a quite restricted subfield of discrete tomography, it actually has a wide range of applications. In many reconstruction tasks that arise in practice it can be assumed that the object being studied is made of or consists of a homogeneous material and hence 0 and 1 values can represent the absence or the presence of this material, respectively. In this section we give an overview of the most successful applications of BT. This summary is by no means exhaustive.

2.3.1 Data Compression, Data Security

One can think of projections of a discrete set F as an encoding of some digital data stored in a binary matrix $\hat{F}$. Since the $\textsc{Reconstruction}(\mathcal{F},\ \mathcal{L})$ task is in general NP-complete for $|\mathcal{L}| > 2$ [48] we can make the data secure by encoding it via the projections along the directions in $\mathcal{L}$ [60]. On the other hand, if for a certain class

$\mathcal{G} \subseteq \mathcal{F}$ and a set of directions $\mathcal{L}$ the $\textsc{Reconstruction}(\mathcal{G}, \mathcal{L})$ task can be solved quickly (say, in polynomial time) then the projections can be considered as a compressed form of the binary matrix $\hat{F}$. If $\hat{F}$ is uniquely determined in $\mathcal{G}$ w.r.t. its projections along the directions of $\mathcal{L}$ the compression will then be lossless; otherwise, it will be lossy.

2.3.2 Image Processing

Discrete sets can also be regarded as binary pictures while their projections can be viewed as their (lossy or lossless) encoded representations. Several authors show that certain image processing operators and methods (e.g. in image registration and thinning) have analogous counterparts that can be applied directly on projections [44; 45; 74; 82]. In [85] a discrete deconvolution technique is also described that is based on binary tomography theory.

2.3.3 Electron Microscopy

There are two main areas of electron microscopy where BT can be applied quite effectively. In structural biology the aim is to describe how a macromolecule is built at the highest possible resolution (as close to the atomic resolution as possible). This can be done through transmission electron microscopy by obtaining images of the specimen under study and then applying a reconstruction from projections technique to get an estimation of the original shape and volume of the macromolecule. Most biological macromolecules in fact contain essentially two or three (protein, ice, and nucleic acid) components. This prior information can be exploited to reduce ambiguity arising from the limited number of available projections, thus making it possible to reconstruct images with higher resolutions. For more details, see [31].

With a technique called QUANTITEM [61; 84], an approach based on high-resolution electron microscopy, it is possible to count the approximate number of atoms lying on the same atomic column. This opens the way to analysing crystalline structures and supplying useful information required in certain areas of physics and chemistry. Here again, techniques of BT can be applied effectively as the presence or absence of atoms can be represented by a 1 value or a 0 value, respectively. For an introduction to DT techniques applied in determining crystalline structures the reader should peruse [18] and [81].

2.3.4 Angiography

When cardiologists wish to identify abnormalities of a cardiac ventricle or a blood vessel it is often done by injecting some Roentgen contrast agent into the part of the body being examined and taking some cross-sectional x-ray images. Then, after the contrast agent is no longer present, a second set of x-ray images are taken. By subtracting the second set of images from the first one, we should get projections of the part of the human body being examined. Assuming that the dye is homogeneous and has unit absorption it leads to the problem of reconstructing a binary image from its projections (where the 1 and 0 values stand for the presence or absence of the dye, respectively). Several algorithms have been proposed for the problem in this special context which usually exploit the fact that the successive slices are very similar. For more details, the reader should see [32] and [78].

2.3.5 Computerized Tomography

Although CT methods are unsuitable for reconstructing discrete sets from just a few projections, images reconstructed by CT methods can be regarded as approximations of the true discrete image. Thus CT images could be the input data for further DT (or BT) procedures which refine them to reconstruct a discrete image. In this way using the fact that the image can contain only a small number of grey-intensity values, this prior knowledge can be incorporated into a second reconstruction phase to produce images with a better resolution and accuracy. For further details of these applications, see [20; 35].

2.3.6 Non-Destructive Testing

Getting information about the interior of an object without damaging it in any way is important if we wish to test the quality of certain industrial parts. For example, this information can help us to detect pipe corrosion, fractures or dangerous engine deformations without damaging them. For this purpose x-ray, gamma, or neutron imaging techniques are used. However, the acquisition of the projections is often time-consuming and expensive, hence the goal is to carry out the reconstruction based on perhaps only a handful of projections. Assuming that the object of interest is made of a homogeneous material one can apply BT techniques to achieve accurate reconstructions even from few projections [19; 62].

2.4 Related Fields

DT and BT have a number of connections with different fields of mathematics. This section provides a brief overview of the main connections between DT and areas of mathematics.

2.4.1 Analysis of Measurable Plane Sets

In 1949 Lorentz investigated the problem of whether a function pair can be the projections of a planar measurable set [72]. He formulated consistency and uniqueness conditions that can be considered as the first results of DT theory. Although later problems of DT were mostly investigated by using other mathematical tools, it was always an important question of whether the results achieved relating to consistency and uniqueness tasks could be extended to measurable plane sets as well [34; 70; 89].

2.4.2 Convex Geometry

In [49] the authors showed that four properly chosen projections uniquely determine all planar convex sets. An analogue of this latter theorem for convex discrete sets that are intersections of convex polygons and $\mathbb{Z}^2$ was then given in [47]. Later, this result was generalized to the broader class of Q-convexes too [39]. We should also mention here that an algorithm for the approximate reconstruction of plane convex bodies based on a certain set of projections was once described in [63]. Some advances relating to the generalization of this result can be found in [26–28].

2.4.3 Linear Algebra, Matrix Theory

Since discrete sets can be represented as binary matrices there is a natural connection between DT and matrix theory. Most of the problems of DT can also be formulated in the language of matrices. In fact, Ryser's reconstruction algorithm [80] was also a matrix theoretical result. Later, several variants of this algorithm were developed. Another question arising in matrix theory is the determination of the number of binary matrices that have certain fixed row and column sums. For a good introduction of this, see Section 1.2.2 in [56] and the references given therein. Another useful survey can be found in [21]. Some of the results have also been generalized to not necessarily binary but two-valued matrices [67]. Yet another algorithm for reconstructing binary matrices from their projections was presented in [53] based on an algebraic analysis of the general solution set of a reconstruction task [54]. Recently, researchers have begun to study the reconstruction task when, besides the row and column sums, some fixed elements of the matrix are also known in advance [22; 65].

2.4.4 Graph Theory, Network Flows

Some tasks of DT can be reformulated as network flows as well [46]. A few newly developed reconstruction algorithms use this approach [16; 17]. Another novel approach in DT is to apply reconstruction algorithms when the discrete-valued function is defined over the vertex and/or edge set of a graph. This field is called and has its applications in Internet research and electric circuit analysis [59].

2.4.5 Linear Programming, Combinatorial Optimization

By introducing one variable for each lattice point of the 2D integer grid, the binary reconstruction problem can be formulated as a linear equation system $Ax = b$ where the binary matrix A describes the correspondence between the lattice points of $\mathbb{Z}^2$ and the projections, the integer vector b stands for the measured projections and the binary vector x represents the unknown discrete set to be reconstructed.

Example 2.4.1
Consider the task of reconstructing the discrete set illustrated in Fig. 2.3 from the horizontal and vertical projections.

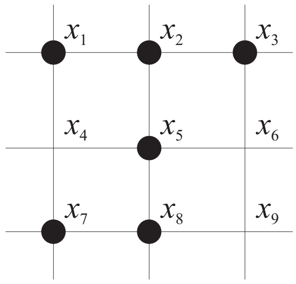


Figure 2.3: Formulating a reconstruction problem as a linear equation system

In this case the unknown vector is $x^T = (x_1, \ldots, x_9)$ and one has to solve the linear equation system $Ax = b$ defined by

$$A = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}, \quad \text{and} \quad b = \begin{pmatrix} 3 \\ 1 \\ 2 \\ 2 \\ 3 \\ 1 \end{pmatrix} . \tag{2.14}$$

The binary equation system $Ax = b$ can be solved for instance by iterative CT reconstruction algorithms [55]. However, it can also be formulated as an optimization problem where the goal is to find the global minimum of the objective function

$$C(x) = ||Ax - b||^2 + \gamma \cdot \Phi(x) . \tag{2.15}$$

The first term on the right hand side of (2.15) ensures that the projections of the reconstructed solution will be approximately the same as the true ones. In the second term, $\Phi(x)$ allows one to incorporate some known facts into the reconstruction. It could determine the smoothness of the reconstructed set or the difference between a model set x_{mod} and the reconstructed one. Finally, $\gamma \geq 0$ is a suitably chosen weighting factor where we can emphasize whether the measured projections are or the prior information is more reliable. This combinatorial optimization problem can then be solved using simulated annealing [19], dc-programming (the abbreviation dc stands for *difference of convex functions*) [83] or genetic algorithms [15].

2.5 Discrete Sets with Geometrical Properties

Due to the small number of available projections in binary tomography the reconstruction of a discrete set is usually an underdetermined and/or an NP-hard problem. Despite this, there is a hope of avoiding intractability and of reducing the number of possible solutions if some prior knowledge can be incorporated into the reconstruction process. The most commonly used properties that the discrete set to be reconstructed has to satisfy to facilitate the reconstruction are of a geometrical nature like connectedness and convexity. Below we will briefly describe the most frequently used geometrical features that can make the reconstruction task easier.

Connectedness Two points $P = (p_1, p_2)$ and $Q = (q_1, q_2)$ in a discrete set F are said to be *4-adjacent* if $|p_1 - q_1| + |p_2 - q_2| = 1$. The points P and Q are said to be *8-adjacent* if they are 4-adjacent or ($|p_1 - q_1| = 1$ and $|p_2 - q_2| = 1$). The sequence of distinct points $P_0, \ldots, P_k$ is a *4/8-path* from point P_0 to point P_k in a discrete set F if each point of the sequence is in F and P_l and P_{l-1} are 4/8-adjacent, respectively, for each $l = 1, \ldots, k$. A discrete set F is *4/8-connected* if, for any two points of F, there is a 4/8-path in F, respectively, between them. The 4-connected set is also known as a *polyomino* [52]. If the discrete set is not 4-connected then it consists of several polyominoes. The reader can readily check that the maximal 4-connected subsets of a discrete set F give a uniquely determined partition of F. Such a maximal 4-connected subset of a discrete set F is called a *component* of F. The discrete set in Fig. 2.1 has 4 components, namely $\{(1, 4)\}$, $\{(2, 2), (2, 3), (3, 2), (3, 3), (4, 1), (4, 2)\}$, $\{(5, 6)\}$, and $\{(6, 5)\}$.

Convexity The discrete set F is *convex along the direction* $v = (a, b)$ if for any two points $(i_1, j_1) \in F$ and $(i_2, j_2) \in F$ if $(i_2, j_2) = (i_1, j_1) + k \cdot v$ for some $k \in \mathbb{Z}$ then $(i_1, j_1) + t \cdot v \in F$ for every $t \in \{0, \ldots, k\}$ as well. In particular, the discrete set is

- *horizontally convex* if it is convex along the $(1, 0)$ direction,
- *vertically convex* if it is convex along the $(0, 1)$ direction,
- *diagonally convex* if it is convex along the $(1, 1)$ direction,
- *antidiagonally convex* if it is convex along the $(-1, 1)$ direction.

For example, Fig. 2.4a depicts a diagonally and horizontally convex polyomino. If a discrete set is both horizontally and vertically convex it is then called hv-convex (see Figs. 2.4b-2.4d). The class of hv-convex discrete sets and its subclasses are studied very frequently in discrete tomography and they will play a central role in this thesis too.

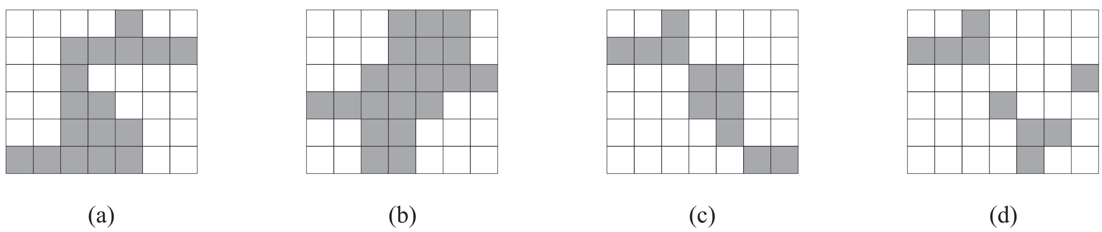


Figure 2.4: A polyomino (a), an hv-convex polyomino (b), an hv-convex 8- but not 4-connected discrete set (c), and a general hv-convex discrete set (d)

Now let us introduce the following notations for certain subclasses of hv-convex discrete sets

- $\mathcal{S}_4$ for the class of 4-connected hv-convex sets;
- $\mathcal{S}_8$ for the class of 8-connected hv-convex sets;
- $\mathcal{S}_8'$ for the class of 8- but not 4-connected hv-convex sets;
- $\mathcal{HV}$ for the class of hv-convex sets.

Evidently, every 4-connected set is also 8-connected, thus $\mathcal{S}_4 \subset \mathcal{S}_8$ (for the real inclusion see Fig. 2.4c for example) and so $\mathcal{S}_8' = \mathcal{S}_8 \setminus \mathcal{S}_4 \neq \emptyset$. Furthermore, it is also clear that $\mathcal{S}_8 \subset \mathcal{HV}$ (see Fig. 2.4d for the real inclusion).

The first reconstruction algorithm for the $\mathcal{HV}$ class using the horizontal and vertical projections was presented in [64], and later it was shown that this reconstruction task is NP-complete [88]. In the same paper the author also demonstrated that the reconstruction of polyominoes from the two orthogonal projections is NP-hard. Further NP-completeness results were presented in [13] for horizontally or vertically convex discrete sets, and horizontally or vertically convex polyominoes. That is, just assuming hv-convexity or 4-connectedness for the set to be reconstructed cannot facilitate the reconstruction when using the horizontal and vertical projections. In spite of this, we have some positive results related to hv-convex polyominoes (i.e. when the set is both 4-connected and hv-convex). There are upper and lower bounds on the number of elements of $\mathcal{S}_4$ that have the same horizontal and vertical projections [42]. In addition, several polynomial-time reconstruction methods have been presented for this class (see [13; 14; 36; 50]). All of these algorithms were later extended to cover the $\mathcal{S}_8$ class as well [30; 66].

In discrete tomography there is another frequently used generalization of the continuous concept of convexity called Q-convexity. For a point $P = (p_1, p_2)$ we can define the four *quadrants around* P by

$$\begin{aligned}
R_0(P) &= \{Q = (q_1, q_2) \mid q_1 \leq p_1 \text{ and } q_2 \leq p_2\} \,, \\
R_1(P) &= \{Q = (q_1, q_2) \mid q_1 \geq p_1 \text{ and } q_2 \leq p_2\} \,, \\
R_2(P) &= \{Q = (q_1, q_2) \mid q_1 \geq p_1 \text{ and } q_2 \geq p_2\} \,, \\
R_3(P) &= \{Q = (q_1, q_2) \mid q_1 \leq p_1 \text{ and } q_2 \geq p_2\} \,.
\end{aligned} \tag{2.16}$$

A discrete set F is *Q-convex* if $R_k(P) \cap F \neq \emptyset$ for all $k \in \{0, 1, 2, 3\}$ implies $P \in F$. Figure 2.5a shows a Q-convex discrete set with the four quadrants around a point P, while the discrete set in Fig. 2.5b is not Q-convex.

Q-convexity was introduced in [24] as an alternative generalization of the continuous concept of convexity. In this article Q-convexity was defined in a more general way along an arbitrary set of directions. More precisely, the above definition corresponds to Q-convexity along the horizontal and vertical directions. Since we only consider Q-convex sets defined in the above manner, for the sake of simplicity we shall use this abbreviated form. The class of Q-convex sets is one of the broadest classes in discrete tomography for which a polynomial-time reconstruction algorithm is known [24]. We will denote the class of Q-convex sets which have several components by $\mathcal{Q}'$. From Fig. 2.5a it is apparent that this class is nonempty.

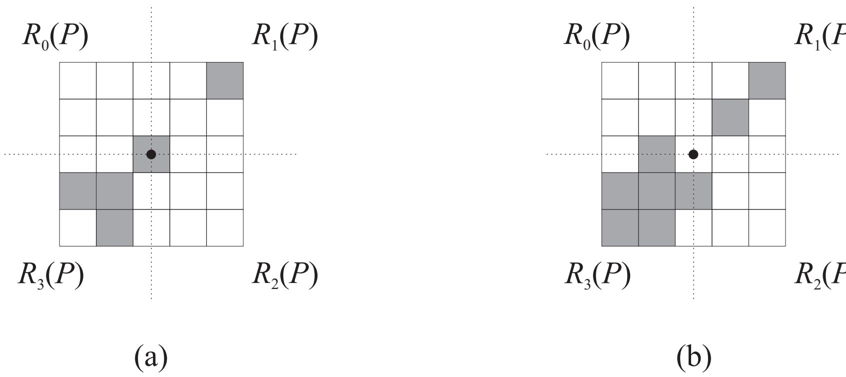


Figure 2.5: A Q-convex (left), and a non-Q-convex discrete set (right) with the four quadrants around the point P (denoted by a black dot)

Directedness We say that the point $(i, j-1)/(i+1, j)/(i, j+1)/(i-1, j)$ lies in the north/east/south/west direction of the point (i, j), respectively. Recall that the vertical axis of $\mathbb{Z}^2$ is top-down directed. A 4-path in a discrete set F is a *northeast path* (NE-path for short) from point P_0 to point P_t if each point P_l of the path is north or east of P_{l-1} for each $l = 1, \ldots, t$. SW-, SE-, NW-paths can be similarly defined. The discrete set F is *NE-directed* if there is a particular point of F, called the *source*, such that there is an NE-path in F from the source to any other point of F. It follows from the definition that the source of a NE-directed set is necessarily the point $(m, 1)$. For example the polyomino in Fig. 2.4a is NE-directed. Similar definitions can be given for SW-, SE-, and NW-directedness. We also say that the discrete set is *directed* if it is NE-, SW-, SE-, or NW-directed. The directedness of discrete sets (in discrete tomography) was first introduced in [40], while the reconstruction in certain classes of directed discrete sets was investigated in [68]. In this thesis we will examine classes of directed polyominoes that are convex along a certain direction. Among such classes, the classes of NE-directed polyominoes will be of especial importance. For a given direction (a, b) such a class will be denoted by $\mathcal{DCP}^{NE}_{(a,b)}$.

Other Properties Connectedness, convexity, and directedness properties are very often used in reconstruction algorithms to reduce ambiguity and to avoid intractability and they will play a central role here as well. There are some other less frequently studied properties of discrete sets that can facilitate the reconstruction process but we will not pursue them further in this thesis. For the interested reader we recommend [33] and [34] for the discussion of a special convexity property. Two other articles which focus on the reconstruction of binary matrices subject to special adjacency constraints are [37; 79]. Yet another property called periodicity was introduced in [41]. In the same article reconstruction algorithms are described for classes of discrete sets with certain periodicity properties.

Chapter 3

Uniqueness and Non-Uniqueness Results for Directed Polyominoes

Consider the task of reconstructing a polyomino such that its horizontal and vertical projections are equal to a given vector $H \in \mathbb{N}^m$ and $V \in \mathbb{N}^n$, respectively. This task is in general NP-hard even if we assume that the polyomino to be reconstructed is horizontally or vertically convex [88]. Curiously though, if we assume convexity in both the horizontal and vertical directions then the above reconstruction task is no longer intractable [13; 14; 36]. However, even in this case there can be exponentially many solutions for certain pairs of vectors H and V [42]. Thus further restrictions have to be imposed on the polyomino to be reconstructed in order to eliminate ambiguity.

As was shown in [40], the assumption that the polyomino in question is convex along at least one of the horizontal and vertical directions, and that it is also NE-directed can ensure that the solution of the above problem is uniquely determined. Later an algorithm was also presented for reconstructing polyominoes like this of size $m \times n$ using their horizontal and vertical projections in $O(mn)$ time [68]. From the constructions given in [40] and [68] it also follows that SE-, NW- or SW-directed polyominoes which are horizontally or vertically convex can also be reconstructed uniquely and with the same time complexity. These results on the uniqueness and reconstruction of directed polyominoes can be summarized in the following

Theorem 3.1 *[40; 68] Every horizontally or vertically convex directed polyomino of size $m \times n$ can be reconstructed from its source and its horizontal and vertical projections uniquely in $O(mn)$ time.*

It is important to note that if the source of the polyomino in question (or equivalently, the direction it points in) is not known in advance then the reconstruction is not necessarily unique (see, for example, Fig. 3.1).

In this chapter we will examine whether Theorem 3.1 can also be generalized to other directions of convexity. We will see how the change of direction of convexity affects the number of possible solutions of the same reconstruction task. Although we will only present results for NE-directed polyominoes here, the reader can easily devise a way of getting similar results for SE-, NW or SW-directed polyominoes as well.

Figure 3.1: Two directed hv-convex polyominoes which have the same horizontal and vertical projections

3.1 Diagonally Convex Directed Polyominoes

In this section we will focus on diagonally convex NE-directed polyominoes, i.e. on the class $\mathcal{DCP}^{NE}_{(1,1)}$. We will use the following simple lemma which holds for an arbitrary NE-directed polyomino.

Lemma 3.1.1 *Let D be a NE-directed polyomino with $\mathcal{H}(D) = (h_1, \ldots, h_m)$ and $\mathcal{V}(D) = (v_1, \ldots, v_n)$. Then $(m, j) \in D$ if and only if $1 \leq j \leq h_m$, and $(i, 1) \in D$ if and only if $m - v_1 < i \leq m$.*

Proof
It as an immediate consequence of NE-directedness and 4-connectedness. □

Now consider an arbitrary polyomino $D \in \mathcal{DCP}^{NE}_{(1,1)}$. From Lemma 3.1.1, the subset F of the polyomino D (which consists of all the elements of the first column and the last row of D) is determined by the vector components h_m and v_1. On the basis of the following lemma the remaining elements of D can be determined by the set F.

Lemma 3.1.2 *Let $D \in \mathcal{DCP}^{NE}_{(1,1)}$, $F \subset D$ and $(i, j) \in \{1, \ldots, m-1\} \times \{2, \ldots, n\}$ be a position such that for every $(i', j') \neq (i, j)$ if $i' \geq i$ and $j' \leq j$ then $(i', j') \in D \leftrightarrow (i', j') \in F$. Then $\sum_{t=i+1}^{n} \hat{f}_{tj} < v_j$ and $\sum_{t=1}^{j-1} \hat{f}_{it} < h_i$ are necessary and sufficient conditions for $(i, j) \in D$.*

Proof
Let (i, j) be a position which satisfies the conditions of the lemma.

The necessary part is trivial since $i' \geq i$, $j' \leq j$ and $(i', j') \neq (i, j)$ implies $(i', j') \in D \leftrightarrow (i', j') \in F$ and so the inequalities $\sum_{t=i+1}^{n} \hat{f}_{tj} < v_j$ and $\sum_{t=1}^{j-1} \hat{f}_{it} < h_i$ must hold.

To prove the sufficient part assume that $\sum_{t=i+1}^{n} \hat{f}_{tj} < v_j$ and $\sum_{t=1}^{j-1} \hat{f}_{it} < h_i$ and assume that $(i, j) \notin D$, that is $\hat{d}_{ij} = 0$.

If $i = 1$ then a contradiction arises from $\sum_{t=i+1}^{n} \hat{f}_{tj} < v_j$ and the fact that $(i', j) \in F \leftrightarrow (i', j) \in D$ holds for every position (i', j) if $i' > 1$. Similarly, if $j = n$ then the contradiction arises from $\sum_{t=1}^{j-1} \hat{f}_{it} < h_i$ and the fact that $(i, j') \in F \leftrightarrow (i, j') \in D$ holds for every position (i, j') if $j' < n$.

In the other cases, since the conditions of the lemma hold, there exist $i'' < i$ and $j'' > j$ for which $\hat{d}_{i'',j} = \hat{d}_{i,j''} = 1$. Since D is NE-directed there is an NE-path from $(m, 1)$ to (i'', j) such that for every point (c_1, c_2) of this path $c_2 \leq j$ holds. Therefore, the lattice line ℓ in the $(1, 1)$ direction which contains the point (i, j) also contains at least one point of D, say (i_1, j_1), for which $j_1 < j$. Similarly, we find that there is a NE-path from $(m, 1)$ to (i, j'') and thus ℓ contains at least one point of D, say (i_2, j_2), for which $j_2 > j$. We get $\hat{d}_{i_1,j_1} = 1$, $\hat{d}_{ij} = 0$ and $\hat{d}_{i_2,j_2} = 1$ with $j_1 < j < j_2$, which contradicts the diagonal convexity property of D (see Fig. 3.2). □

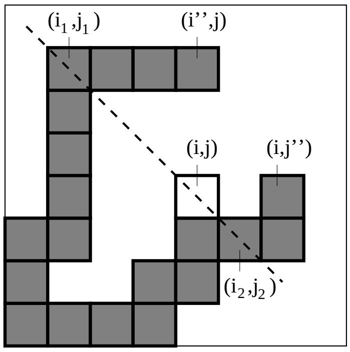


Figure 3.2: Proof of Lemma 3.1.2

The following theorem states that every diagonally convex NE-directed polyomino is uniquely determined by its horizontal and vertical projections.

Theorem 3.2 *Let $H \in \mathbb{N}^m$ and $V \in \mathbb{N}^n$. In the class $\mathcal{DCP}^{NE}_{(1,1)}$ there is at most one polyomino P such that $\mathcal{H}(P) = H$ and $\mathcal{V}(P) = V$.*

Proof

From Lemma 3.1.1, the first column and the last row of P are uniquely determined by v_1 and h_m, respectively, i.e. a subset F of the polyomino P can be found (consisting of all the positions of the last row and first column of P).

Then for the position $(m-1, 2)$ the conditions of Lemma 3.1.2 hold. Therefore, using Lemma 3.1.2 we can establish whether the position $(m-1, 2)$ belongs to P and, if so, we set $F = F \cup \{(m-1, 2)\}$. Taking each position bottom up and left to right, F always satisfies the conditions of Lemma 3.1.2 and so the above procedure can be repeated. If H and V are the projections of a diagonally convex NE-directed polyomino then we will eventually get $F = P$.

The uniqueness property follows from the construction. □

The proof of Theorem 3.2 is constructive, that is an algorithm (called Algorithm DCP) similar to the one in [68] can also be described to reconstruct the possibly existing polyomino of the class $\mathcal{DCP}^{NE}_{(1,1)}$ with given horizontal and vertical projections.

This algorithm works as follows. Step 1 is for the initialization of the matrix $\hat{F}$ and the auxiliary vectors H' and V'.

In Step 2 a subset F of the polyomino to be reconstructed is defined based on Lemma 3.1.1.

Then in each iteration of Step 3 we check whether the conditions of Lemma 3.1.2 hold and, if so, we update the matrix $\hat{F}$ and the vectors H' and V'. Using the vectors H' and V' Step 3 runs in $O(mn)$ time.

Lastly, in Step 4 we check whether the reconstructed set is a diagonally convex NE-directed polyomino and has the given projections H and V which can also be performed in $O(mn)$ time. The importance of this step will become clear when the reader takes a look at Fig. 3.3. It shows a discrete set that does not belong to the class $\mathcal{DCP}^{NE}_{(1,1)}$ but it can be reconstructed by applying just the first three steps of Algorithm DCP. In the final check in Step 4 of Algorithm DCP we can make sure that the algorithm does not

Algorithm DCP for reconstructing polyominoes of $\mathcal{DCP}^{NE}_{(1,1)}$ from two projections

Input: Two compatible vectors, $H \in \mathbb{N}^m$ and $V \in \mathbb{N}^n$.
Output: The matrix $\hat{F}$ which represents the uniquely determined polyomino $F \in \mathcal{DCP}_{(1,1)}$ with projections $\mathcal{H}(F) = H$ and $\mathcal{V}(F) = V$ (when such a solution exists).

Step 1 $\hat{F} := (0)_{m\times n}$; $H' := H$; $V' := V$;
Step 2 **for** $i := m - v_1 + 1, \ldots, m$ $\{\hat{f}_{i1} := 1;$ **dec**(h'_i);$\}$
  for $j := 1, \ldots, h_m$ $\{\hat{f}_{mj} := 1;$ **dec**(v'_j);$\}$
Step 3 **for** $i := m - 1, \ldots, 1$
  for $j := 2, \ldots, n$
    if $(h'_i > 0$ **and** $v'_j > 0)$ **then** $\{\hat{f}_{ij} := 1;$ **dec**(h'_i); **dec**(v'_j);$\}$
Step 4 **if** $(\mathcal{H}(F) \neq H$ **or** $\mathcal{V}(F) \neq V$ **or** $F \notin \mathcal{DCP}^{NE}_{(1,1)})$
  then exit(no solution) **else return** $\hat{F}$;

give a solution if there is no element of $\mathcal{DCP}^{NE}_{(1,1)}$ with the given horizontal and vertical projections, H and V respectively. In this way Algorithm DCP can handle not just the $\textsc{Reconstruction}(\mathcal{DCP}^{NE}_{(1,1)}, \mathcal{L}_2)$ task but also the corresponding consistency issue.

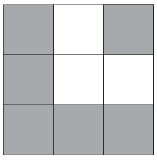


Figure 3.3: A discrete set that is neither 4-connected nor diagonally convex, but it can be reconstructed by steps 1-3 of Algorithm DCP

Summarizing the last few paragraphs, we state the following

Corollary 3.1 *Let $H \in \mathbb{N}^m$ and $V \in \mathbb{N}^n$. If there is a polyomino $P \in \mathcal{DCP}^{NE}_{(1,1)}$ such that $\mathcal{H}(P) = H$ and $\mathcal{V}(P) = V$ then Algorithm DCP reconstructs it in $O(mn)$ time. Otherwise, the algorithm does not give a solution and terminates.*

3.2 Antidiagonally Convex Directed Polyominoes

In this section we will show that there will be a drastic change in the number of NE-directed polyominoes which have the same horizontal and vertical projections if, instead of diagonal convexity, it is assumed that the polyomino is antidiagonally convex. The following theorem demonstrates that assuming antidiagonal convexity on the NE-directed polyomino which has fixed horizontal and vertical projections will not eliminate ambiguity.

Theorem 3.3 *In the class $\mathcal{DCP}^{NE}_{(-1,1)}$ there can be exponentially many polyominoes with the same horizontal and vertical projections.*

Proof

We show that for any $k \in \mathbb{N}$ there are at least 2^k polyominoes of size $(6k-1) \times 3$ in the class $\mathcal{DCP}^{NE}_{(-1,1)}$ with the same horizontal and vertical projections. Let

$$\hat{B} = \begin{pmatrix} 1 & 1 & 0 \end{pmatrix}, \quad \hat{F}_1 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ & \hat{B} & \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad \text{and} \quad \hat{F}_k = \begin{pmatrix} \hat{F}_1 \\ \hat{B} \\ \hat{F}_{k-1} \end{pmatrix} \quad \text{for} \quad k > 1 \ . \tag{3.1}$$

For a given $k \in \mathbb{N}$ and for any $l \in \mathbb{N}$ $(1 \leq l \leq k)$ we will refer to the submatrix of $\hat{F}_k$ consisting of the rows $6(l-1)+i \quad (i = 1, \ldots, 5)$ as the l-th *level* of $\hat{F}_k$ and to the submatrix $\hat{B}$ in the row $6l$ as the l-th *bridge* of $\hat{F}_k$ (omitting the case $k = l$). For any l the set of positions $\{(6(l-1)+4, 2), (6(l-1)+4, 3), (6(l-1)+1, 2), (6(l-1)+1, 3)\}$ form a switching component in $\hat{F}_k$ and we will refer to it as the l-th switching component of $\hat{F}_k$.

Next let $\hat{F}'_1$ be the binary matrix we get by interchanging the 0s and 1s in the first switching component of $\hat{F}_1$, that is

$$\hat{F}'_1 = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 1 & 1 \end{pmatrix} .$$

Evidently, $F_1, F'_1 \in \mathcal{DCP}^{NE}_{(-1,1)}$ and so the theorem holds for the case $k = 1$.

For the case $k > 1$ let $\hat{F}^S_k$ where $S = \{s_1, \ldots, s_n\} \subseteq \{1, \ldots, k\}$ $(n \leq k)$ denote the binary matrix we get from $\hat{F}_k$ by switching the s_1-th, ..., s_n-th switching components. Note that from the viewpoint of directedness, 4-connectedness or antidiagonal convexity the l-th bridge only affects the $(l+1)$-th and l-th levels and vice versa (when they exist). In order to prove that $F^S_k \in \mathcal{DCP}^{NE}_{(-1,1)}$ for any $k \in \mathbb{N}$ and $S \subseteq \{1, \ldots, k\}$ it is sufficient to study the submatrices of $\hat{F}^S_k$ which consists of the l-th level and the l-th bridge and the l-th bridge and the $(l+1)$-th level. These matrices can only be of the form $\begin{pmatrix} \hat{B} \\ \hat{F}_1 \end{pmatrix}$, $\begin{pmatrix} \hat{F}_1 \\ \hat{B} \end{pmatrix}$, $\begin{pmatrix} \hat{B} \\ \hat{F}'_1 \end{pmatrix}$, or $\begin{pmatrix} \hat{F}'_1 \\ \hat{B} \end{pmatrix}$. Actually, it can be readily shown that the four sets represented by these matrices are antidiagonally convex and NE-directed.

For a given k, S can be any subset of $\{1, \ldots, k\}$, thus the number of solutions with the same projections is at least 2^k, and so the theorem is proved (see Fig. 3.4 for the case $k = 2$). $\square$

Remark 3.2.1

The bound 2^k in the proof of Theorem 3.3 is not tight. See, for instance, the discrete set in Fig. 3.4e.

Notice that Theorem 3.3 above does not tell us whether the reconstruction from the horizontal and vertical projections in the class $\mathcal{DCP}^{NE}_{(-1,1)}$ is solvable in polynomial time

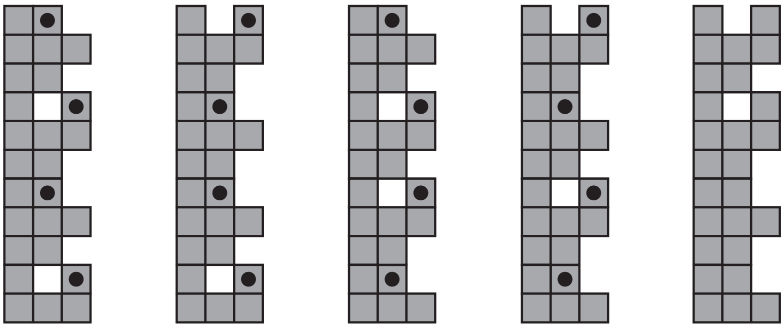


Figure 3.4: Exponentially many polyominoes of $\mathcal{DCP}^{NE}_{(-1,1)}$ with the same horizontal and vertical projections. (a)-(d) Proof of Theorem 3.3 for the case $k = 2$. The sets from left to right are $F_2^{\{\}}$, $F_2^{\{1\}}$, $F_2^{\{2\}}$, and $F_2^{\{1,2\}}$, respectively. Elements of the switching components with a value of 1 are denoted by black dots. (e) Another set with the same projections

as the $\textsc{Reconstruction}(\mathcal{DCP}^{NE}_{(-1,1)},\ \mathcal{L}_2)$ procedure seeks to find just one solution. Despite this, one consequence of Theorem 3.3 is the following.

Corollary 3.2 *If there is an algorithm which reconstructs all the discrete sets of class $\mathcal{DCP}^{NE}_{(-1,1)}$ with the horizontal and vertical projections $H \in \mathbb{N}^m$ and $V \in \mathbb{N}^n$, respectively, then there are some pairs of vectors H and V for which the time complexity of the algorithm is exponential.*

3.3 Generalisation to Arbitrary Line-Convexity

From Theorem 3.2 and Theorem 3.3 it is quite apparent that the direction of convexity plays an important role in determining whether ambiguity can be eliminated. In this section we see in more detail how the direction of convexity affects the number of NE-directed polyominoes. Our key finding here is that the construction given in the proof of Theorem 3.3 can be adapted to polyominoes which are convex along an arbitrary lattice direction $d \notin \{(1,0),(0,1),(1,1)\}$.

Theorem 3.4 *Let $d = (a,b)$ be a lattice direction such that $d \notin \{(1,0),(0,1),(1,1)\}$. In the class $\mathcal{DCP}^{NE}_{(a,b)}$ there can be exponentially many polyominoes with the same horizontal and vertical projections.*

Proof

Assume that a direction $d = (a,b)$ is given such that $d \notin \{(1,0),(0,1),(1,1)\}$. We will give a construction similar to the one used in the proof of Theorem 3.3. The idea is that with some additional rows of appropriate size we can construct polyominoes such that two consecutive levels will not affect each other convexity-wise along the d direction.

First, observe that for each $k \geq 1$ the matrices of (3.1) have just 3 columns so they are also line-convex along any direction (a,b) if $|a| \geq 2$.

Moreover, $a = 0$ implies $b = 1$. Since $d \neq (0,1)$ the theorem holds for any direction (a,b) where $|a| \neq 1$.

Consider $|a| = 1$ and construct the bridge $\hat{B}^{(b)}$ of size $b \times 3$ and the matrix $\hat{F}_1^{(b)}$ of size $(4+b) \times 3$ from the matrices of (3.1) in the following way

$$\hat{B}^{(b)} = \begin{pmatrix} 1 & 1 & 0 \\ & \vdots & \\ 1 & 1 & 0 \end{pmatrix}, \quad \hat{F}_1^{(b)} = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 1 & 1 \\ & \hat{B}^{(b)} & \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{pmatrix}, \quad \text{and} \quad \hat{F}_k^{(b)} = \begin{pmatrix} \hat{F}_1^{(b)} \\ \hat{B}^{(b)} \\ \hat{F}_{k-1}^{(b)} \end{pmatrix} \quad \text{for} \quad k > 1\,. \tag{3.3}$$

Notice that $\hat{B}^{(1)} = \hat{B}$ and so $\hat{F}_1^{(1)} = \hat{F}_1$. Then for any $b \in \mathbb{N}$ the proof can be completed in a similar way to the proof of Theorem 3.3. $\square$

Using Theorem 3.2 and Theorem 3.4 we can also examine the problem of uniqueness when the NE-directed polyomino is line-convex in several directions.

Theorem 3.5 *Let $D = \{(a_i, b_i) \mid i = 1, \ldots, l\}$ be a finite set of lattice directions and let $\mathcal{DCP}_D^{NE}$ denote the class of NE-directed polyominoes that are convex along every direction of D, and $P \in \mathcal{DCP}_D^{NE}$. Then P is uniquely determined by its horizontal and vertical projections in the class $\mathcal{DCP}_D^{NE}$ and it can be reconstructed in $O(mn)$ time if $D \cap \{(1,0),(0,1),(1,1)\} \neq \emptyset$. Otherwise if $D \cap \{(1,0),(0,1),(1,1)\} = \emptyset$ there can be exponentially many polyominoes in the class $\mathcal{DCP}_D^{NE}$ with the same horizontal and vertical projections as P.*

Proof

The uniqueness part follows from Theorem 3.1 and Theorem 3.2.

To prove the second part consider $D \cap \{(1,0),(0,1),(1,1)\} = \emptyset$. In this case there are two possibilities. If there is at least one direction $(a_j, b_j) \in D$ for a $j \in \{1, \ldots, l\}$ such that $|a_j| = 1$ then we have to apply the construction of (3.3) with the argument $b = \max\{b_i : |a_i| = 1, (a_i, b_i) \in D\}$.

Otherwise (if $|a_i| > 1$ for all $i = 1, \ldots, l$) we can use the construction of (3.1). $\square$

Remark 3.3.1

The construction applied in the proof of Theorem 3.5 can also be extended to any infinite set D of directions except the case when $|a_i| = 1$ for infinitely many $(a_i, b_i) \in D$. In this latter case the integer $\max\{b_i : |a_i| = 1, (a_i, b_i) \in D\}$ does not exist.

3.4 Summary

In this chapter we studied the problem of uniqueness in classes of directed polyominoes when the horizontal and vertical projections are available.

We showed that in the class of diagonally convex NE-directed polyominoes these two projections uniquely determine the polyomino and it can be reconstructed in $O(mn)$ time. In an analogous way we can prove that the same holds true for diagonally convex

SW-directed, and antidiagonally convex NW- or SE-directed polyominoes (geometrically, we simply rotate the 2D integer lattice). These results complement an earlier theorem about horizontally or vertically convex directed polyominoes (see Theorem 3.1).

In addition, we also proved that assuming convexity for the NE-directed polyomino in any other remaining direction does not eliminate ambiguity, i.e. there can be exponentially many polyominoes with the properties mentioned which have the same horizontal and vertical projections. Even in these cases the results can be adapted to SE-, SW- or NW-directed polyominoes in a straightforward way.

After, the results were generalised to directed polyominoes which satisfy convexity properties along an arbitrary set of finite directions, and also for some infinite sets of directions.

We should also mention here that the results of this chapter were published in [2].

Chapter 4

Reconstruction of Q-Convex non-4-Connected Discrete Sets

Q-convexity is a property that was introduced in [24]. The class of the Q-convexes is one of the broadest known classes in which polynomial-time reconstruction is possible using just two projections. Moreover, for this class the uniqueness results of Gardner-McMullen [49] and Gardner-Gritzmann [47] can be extended [39]. It [24] it was also shown that every discrete set which is Q-convex along the horizontal and vertical directions is also hv-convex. This result is especially important since it implies that in a subclass of hv-convex discrete sets the reconstruction can be performed in polynomial time using horizontal and vertical projections, while in the general class of hv-convexes this problem is NP-hard [88]. The above results related to Q-convex discrete sets help explain the importance of this class in discrete tomography.

As was shown in [29], the reconstruction of discrete sets that are Q-convex along the horizontal and vertical directions can be performed in $O(N^4 \log N)$ time (where $N = \max\{m, n\}$) using horizontal and vertical projections. In this chapter we present a faster algorithm for this class when it is also known in advance that the Q-convex set to be reconstructed is not 4-connected, that is it consists of two or more components.

4.1 Components of Q-Convex Sets

Let F be a non-4-connected Q-convex set which has components $F_1, \ldots, F_k$ $(k \geq 2)$ such that $\{i_l, \ldots, i'_l\} \times \{j_l, \ldots, j'_l\}$ is the smallest containing discrete rectangle (SCDR) of the l-th $(l = 1, \ldots, k)$ component of F. Since every Q-convex set is also hv-convex the sets of the row/column indices of the components of F consist of consecutive indices and they are disjoint. Without loss of generality we can assume that

$$1 = i_1 \leq i'_1 < i_2 \leq i'_2 < \cdots \leq i'_k = m \; . \tag{4.1}$$

The following lemma shows that the SCDRs of the components of F can be arranged in just two possible ways.

Lemma 4.1.1 *Let $F \in \mathcal{Q}'$ which has components $F_1, \ldots, F_k$ $(k \geq 2)$ with the SCDRs $\{i_l, \ldots, i'_l\} \times \{j_1, \ldots, j'_l\}$ $(l = 1, \ldots, k)$ such that (4.1) holds. Then exactly one of the following cases is possible*

(1) $1 = j_1 \leq j'_1 < j_2 \leq j'_2 < \cdots \leq j'_k = n$*, or*

(2) $n = j_1 \geq j'_1 > j_2 \geq j'_2 > \cdots \geq j'_k = 1$*.*

Proof
Obviously, both case (1) and case (2) cannot hold simultaneously.

For $k = 2$ the statement is trivial. Assume that $k > 2$ and neither case (1) nor case (2) is satisfied. Then there exist three components, say F_{l_1}, F_{l_2}, and F_{l_3}, such that $i_{l_1} < i_{l_2} < i_{l_3}$ and exactly one of the following relations hold

(a) $j_{l_1} < j_{l_3} < j_{l_2}$, or

(b) $j_{l_3} < j_{l_1} < j_{l_2}$, or

(c) $j_{l_2} < j_{l_1} < j_{l_3}$, or

(d) $j_{l_2} < j_{l_3} < j_{l_1}$.

Assume that case (a) is true and let $S = \{i_{l_2}, \ldots, i'_{l_2}\} \times \{j_{l_3}, \ldots, j'_{l_3}\}$. Since F is Q-convex it is hv-convex as well, thus $F \cap S = \emptyset$. Moreover, for an arbitrary position $M \in S$ all the four quadrants along M are non-empty, i.e. $R_k(M) \cap F \neq \emptyset$ for all $k \in \{0, 1, 2, 3\}$, which contradicts the Q-convexity property (see Fig. 4.1).

The proof is similar in cases (b), (c), and (d). □

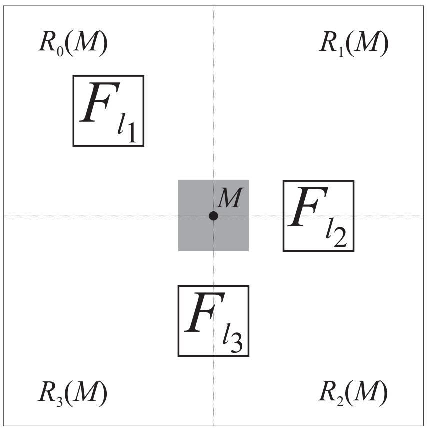


Figure 4.1: Proof of Lemma 4.1.1. The set S is denoted by a grey square

If case 1 of Lemma 4.1.1 holds for a set $F \in \mathcal{Q}'$ then we say that F is of *NW type*. Similarly, if case 2 of Lemma 4.1.1 is satisfied then we say that F is of *NE type* (see Fig. 4.2).

Before going further we will give a simple description of directed hv-convex polyominoes which directly follows from the definition of directedness.

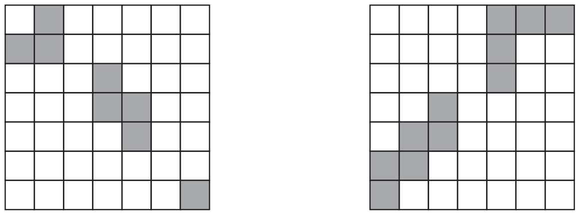


Figure 4.2: A discrete set of $\mathcal{Q}'$ of NW type (left) and of NE type (right)

Proposition 4.1 *Let $G \in \mathcal{S}_4$ and $\{i', \dots i''\} \times \{j', \dots j''\}$ be its SCDR. Then*

(1) G is SE-directed if and only if $\hat{g}_{i',j'} = 1$;

(2) G is NW-directed if and only if $\hat{g}_{i'',j''} = 1$;

(3) G is SW-directed if and only if $\hat{g}_{i',j''} = 1$;

(4) G is NE-directed if and only if $\hat{g}_{i'',j'} = 1$.

The following lemma is about the directedness of the components, which depends on the type of the set.

Lemma 4.1.2 *Let $F \in \mathcal{Q}'$ that has components $F_1, \dots, F_k$ $(k \geq 2)$ with the SCDRs $\{i_l, \dots, i'_l\} \times \{j_l, \dots, j'_l\}$ $(l = 1, \dots, k)$ such that (4.1) holds. If F is of NW/NE type then $F_1, \dots, F_{k-1}$ are NW/NE-directed and $F_2, \dots, F_k$ are SE/SW-directed, respectively.*

Proof

Assume that F is of NW type and let M denote the bottom right position of the SCDR of F_1. If F_1 is not NW-directed then $M \notin F$ using Proposition 4.1. But $R_k(M) \cap F \neq \emptyset$ for all $k \in \{0, 1, 2, 3\}$, which contradicts the Q-convexity property.

In the same way we find that $F_2, \dots, F_{k-1}$ are also NW-directed. The SW-directedness of the components $F_2, \dots, F_k$ can be proved in an analogous way by examining the top left positions of the SCDRs of the components.

The proof is similar when F is of NE type. □

4.2 Locating the Components

Now we will demonstrate how we can represent a set of $\mathcal{Q}'$. From points discussed in the previous section the following corollary can be stated.

Corollary 4.1 *Let $F \in \mathcal{Q}'$ which has components $F_1, \dots, F_k$. Then there are uniquely determined row indices $0 < i_1 < \dots < i_k = m$ and column indices $0 < j_1 < \dots < j_k \leq n$ such that for each $l = 1, \dots, k$ $(k \geq 2)$ (i_l, j_l) is the bottom-right position of the SCDR of F_l when F is of NW type and (i_l, j_{k-l+1}) is the bottom-left position of the SCDR of F_l when F is of NE type.*

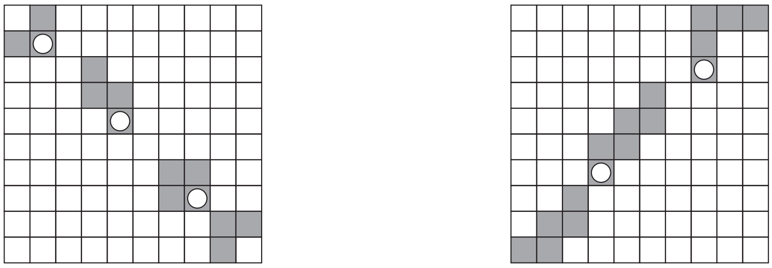


Figure 4.3: A discrete set $F \in \mathcal{Q}'$ of NW type (left) and a discrete set $F' \in \mathcal{Q}'$ of NE type (right). The elements of C_F and $C_{F'}$ are depicted by white dots

Depending on the type of F, let us define

$$C_F = \begin{cases} \{(i_l, j_l) \mid l = 1, \dots, k-1\}, & \text{when } F \text{ is of NW type ,} \\ \{(i_l, j_{k-l+1}) \mid l = 1, \dots, k-1\}, & \text{when } F \text{ is of NE type ,} \end{cases} \tag{4.2}$$

where $i_1, \dots, i_k$ and $j_1, \dots, j_k$ denote the uniquely determined indices mentioned in Corollary 4.1. That is, using Proposition 4.1 C_F consists of the source points of the NW-/NE-directed components $F_1, \dots, F_{k-1}$ when F is of NW/NE type, respectively (see Fig. 4.3). The knowledge of any element of C_F is useful in the reconstruction of a set $F \in \mathcal{Q}'$, which can be stated in following theorem.

Theorem 4.1 *Any $F \in \mathcal{Q}'$ is uniquely determined in $\mathcal{Q}'$ by its horizontal and vertical projections, its type, and an arbitrary element of C_F.*

Proof

Let us suppose that F is of NW type and $(i_l, j_l) \in C_F$ is given for some $l \in \{1, \dots, k-1\}$. Then (i_l, j_l) is the source of the NW-directed component F_l which can be reconstructed uniquely using Theorem 3.1. Next, suppose that the SCDR of F_l is $\{i', \dots, i_l\} \times \{j', \dots, j_l\}$. Then the source (i^*_{l+1}, j^*_{l+1}) of the SE-directed component F_{l+1} and the source (i_{l-1}, j_{l-1}) of the NW-directed component F_{l-1} are uniquely determined. Namely,

$$\begin{aligned} i^*_{l+1} &= \min\{i \mid i > i_l \text{ and } h_i \neq 0\} , \\ j^*_{l+1} &= \min\{j \mid j > j_l \text{ and } v_j \neq 0\} , \\ i_{l-1} &= \max\{i \mid i < i' \text{ and } h_i \neq 0\} , \\ j_{l-1} &= \max\{j \mid j < j' \text{ and } v_j \neq 0\} . \end{aligned} \tag{4.3}$$

Again, by Theorem 3.1 F_{l+1} and F_{l-1} can be reconstructed uniquely. Then the sources of F_{l+2} and F_{l-2} are determined using the formulas of (4.3). This method can be continued until F_1 and F_k are reconstructed.

The proof is similar when F is of NE type. □

As a direct consequence, we get from Theorem 4.1 that different solutions of the $\textsc{Reconstruction}(\mathcal{Q}', \mathcal{L}_2)$ task with the same type have different source points.

Corollary 4.2 *If $F, F' \in \mathcal{Q}'$ are different solutions of the same reconstruction task $\textsc{Reconstruction}(\mathcal{Q}', \mathcal{L}_2)$ and they have the same type then $C_F \cap C_{F'} = \emptyset$.*

Using Theorem 4.1 we can reconstruct the Q-convex set F from its horizontal and vertical projections if we know the type of F and at least one element of C_F. In the following we will show that the cumulated vectors of F are appropriate for determining elements of C_F.

Let $\widetilde{H}$ and $\widetilde{V}$ be the cumulated vectors of $F \in \mathcal{Q}'$. We say that the position $(i,j) \in \{1,\ldots,m-1\} \times \{1,\ldots,n-1\}$ is an *equality position of NW type* of F if $\widetilde{h}_i = \widetilde{v}_j$. In an analogous way, we say that $(i,j) \in \{1,\ldots,m\} \times \{2,\ldots,n+1\}$ is an *equality position of NE type* of F if $\widetilde{h}_i = \widetilde{v}_n - \widetilde{v}_{j-1}$. Equality positions (i,j) for which $h_i = 0$ and/or $v_j = 0$ are not of interest to us here and in the following we will omit them. Now let us denote the set of equality positions of NW/NE type of F by L_F^{NW}/L_F^{NE}, respectively.

The following lemma reveals the connection between the set of equality positions of F and the set of source points C_F.

Lemma 4.2.1 *Let $F \in \mathcal{Q}'$. Then $C_F \subseteq L_F^{NW}$ when F is of NW type and $C_F \subseteq L_F^{NE}$ when F is of NE type.*

Proof
Let us suppose that F is of NW type and define a set E as follows

$$E = (\{1,\ldots,i\} \times \{j+1,\ldots,n\}) \cup (\{i+1,\ldots,m\} \times \{1,\ldots,j\}) \; . \tag{4.4}$$

If $(i,j) \in C_F$ then $F \cap E = \emptyset$. It is not hard to see that $C_F \subseteq L_F^{NW}$ since

$$\begin{aligned} \widetilde{h}_i = \sum_{t=1}^{i} h_t &= |F \cap \{1,\ldots,i\} \times \{1,\ldots,n\}| = |F \cap \{1,\ldots,i\} \times \{1,\ldots,j\}| \\ &= |F \cap \{1,\ldots,m\} \times \{1,\ldots,j\}| = \sum_{t=1}^{j} v_t = \widetilde{v}_j \; . \end{aligned} \tag{4.5}$$

If F is of NE type then define E' as follows

$$E' = (\{1,\ldots,i\} \times \{1,\ldots,j-1\}) \cup (\{i+1,\ldots,m\} \times \{j,\ldots,n\}) \; . \tag{4.6}$$

If $(i,j) \in C_F$ then $F \cap E' = \emptyset$. As before, it is obvious that $C_F \subseteq L_F^{NE}$ since

$$\begin{aligned} \widetilde{h}_i &= \sum_{t=1}^{i} h_t = |F \cap \{1,\ldots,i\} \times \{1,\ldots,n\}| = |F \cap \{1,\ldots,i\} \times \{j,\ldots,n\}| \\ &= |F \cap \{1,\ldots,m\} \times \{j,\ldots,n\}| = \sum_{t=j}^{n} v_t = \widetilde{v}_n - \widetilde{v}_{j-1} \; . \end{aligned} \tag{4.7}$$

□

4.3 The Reconstruction Algorithm

Now we are ready to describe the algorithm for reconstructing sets of $\mathcal{Q}'$ from their horizontal and vertical projections. A simple description of the algorithm is given in the following.

Algorithm 2-RECQ' for reconstructing sets of $\mathcal{Q}'$ from two projections

Input: Two compatible vectors, $H \in \mathbb{N}^m$ and $V \in \mathbb{N}^n$.
Output: All the binary matrices $\hat{F}$ representing sets of $\mathcal{Q}'$ having projections $\mathcal{H}(F) = H$ and $\mathcal{V}(F) = V$.

Step 1 identify the sets L_F^{NW} and L_F^{NE};
Step 2 while $L_F^{NW} \neq \emptyset$ **do**
    $F = \emptyset$; $l := 1$;
    take an arbitrary element $(i,j) \in L_F^{NW}$ and let $L_F^{NW} = L_F^{NW} \setminus \{(i,j)\}$;
    try to reconstruct the NW-directed component F_1 from source (i,j)
    with $\mathcal{H}(F_1) = (h_1, \ldots, h_i)$ and $\mathcal{V}(F_1) = (v_1, \ldots, v_j)$;
    repeat
        if F_l is not reconstructed **then break**;
        $F := F \cup F_l$; $l := l + 1$;
        identify the source (i_l^*, j_l^*) of F_l and try to reconstruct the
        SE-directed component F_l from source (i_l^*, j_l^*) with the
        corresponding projections;
        if F_l is reconstructed **then**
            if F_l is the last component **then** output $F \cup F_l$;
            else if F_l is NW-directed **then** $L_F^{NW} = L_F^{NW} \setminus \{(i_l, j_l)\}$;
    until F_l is the last component **or** F_l is not NW-directed;
Step 3 try to reconstruct solutions of NE type in an analogous way as in Step 2,
    using the set L_F^{NE};

This algorithm works as follows. First we assume that the set $F \in \mathcal{Q}'$ to be reconstructed is of NW type. Then using Theorem 4.1 it is sufficient to find an arbitrary element of C_F to uniquely reconstruct F from its horizontal and vertical projections. The elements of C_F are equality positions of NW type using Lemma 4.2.1. Thus in order to find all the solutions of the reconstruction problem we should test every element of L_F^{NW} to see whether it is in C_F as well.

Without any loss of a solution we can assume that if a given equality position $(i,j) \in L_F^{NW}$ is in C_F then it is the source of the first component F_1. From Lemma 4.1.2 we know that this component is NW-directed. Now, in order to decide whether (i,j) is the source of F_1, we will try to reconstruct an hv-convex NW-directed polyomino G with source (i,j) such that $\mathcal{H}(G) = (h_1, \ldots, h_i)$ and $\mathcal{V}(G) = (v_1, \ldots, v_j)$.

If there is no such polyomino then obviously (i,j) cannot be the source of F_1 and we will go on to the next equality position from L_F^{NW}.

Otherwise, we can assume that $F_1 = G$ and we will try to reconstruct the remaining components iteratively. The l-th component F_l ($l = 2, \ldots$) is SE-directed (see Lemma 4.1.2), so we will try to reconstruct an hv-convex SE-directed polyomino G

such that the number of its points in each row and column is equal to the corresponding elements of H and V, respectively. The source of F_l must be the position (i^*_{l+1}, j^*_{l+1}) where i^*_{l+1} and j^*_{l+1} are defined by (4.3).

If it is not possible to reconstruct the SE-directed polyomino, then clearly (i^*_{l+1}, j^*_{l+1}) cannot be the source of F_l, which contradicts the assumption that (i,j) is the source of F_1 and we will go on to the next equality position from L_F^{NW}.

If it is possible, we should check to see whether the reconstructed component was the last one. This can be done by checking the bottom right position (i'', j'') of G. If $(i'', j'') \neq (m,n)$, G cannot be the last component. Then, using Lemma 4.1.2, G must be NW-directed. This can be examined by using Proposition 4.1.

If $\hat{g}_{i'',j''} \neq 1$ then G is not NW-directed and clearly $F_l \neq G$, which contradicts the assumption that (i,j) is the source of F_1. In this case we can go on to the next equality position from L_F^{NW}.

Otherwise, that is when $\hat{g}_{i'',j''} = 1$, we can assume that $F_l = G$. From Corollary 4.2, G cannot be the first component of any other solution of the same type therefore (i'', j'') can be eliminated from L_F^{NW} and we continue with the next iteration.

If $(i'', j'') = (m,n)$ then $F_l = G$ and $F = F_1 \cup \cdots \cup F_l$. We have found one solution and we can go on to the next equality position from L_F^{NW} to find another solution.

The second part of the algorithm is similar. Here we will assume that F is of NE type and we will try to reconstruct NE- and SW-directed components from the corresponding sources.

If no solutions are found after examining all the equality positions of both types then the assumption that $F \in \mathcal{Q}'$ is not met, i.e. there is no discrete set with the given projections which is Q-convex and has at least two components. However in some cases there can be several solutions (see Fig. 4.4).

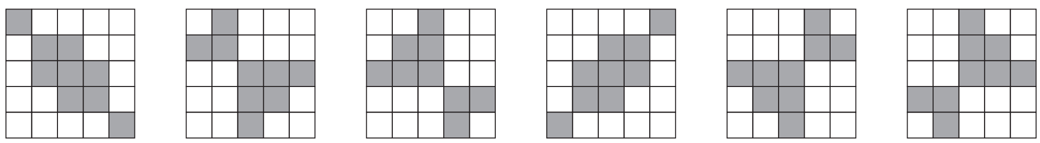


Figure 4.4: Six sets of $\mathcal{Q}'$ with the same horizontal and vertical projections

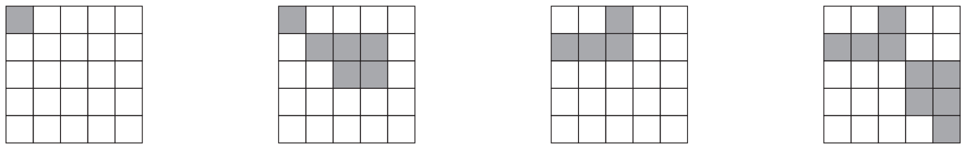
Example 4.3.1

Suppose that we wish to reconstruct the Q-convex discrete set F in Fig. 4.5d from the vectors $H = (1,3,2,2,1)$ and $V = (1,1,2,2,3)$. In this case $\widetilde{H} = (1,4,6,8,9)$, $\widetilde{V} = (1,2,4,6,9)$, $L_F^{NW} = \{(1,1),(2,3),(3,4)\}$, $L_F^{NE} = \{(4,2)\}$, and $C_F = \{(2,3)\}$.

First, we assume that the set to be reconstructed is of NW type. Assuming that $(1,1)$ is the source of the NW-directed component F_1 (Fig. 4.5a) we delete the $(1,1)$ position from L_F^{NW}. Then the algorithm reconstructs the SE-directed F_2 (Fig. 4.5b) and it deletes the $(3,4)$ position from L_F^{NW}. In the next iteration of the repeat loop, F_3 cannot be reconstructed so the algorithm breaks the loop.

Assuming that $(2,3)$ is the source of F_1 (Fig. 4.5c), the algorithm deletes $(2,3)$ from L_F^{NW} and gives a solution of NW type (Fig. 4.5d). Since L_F^{NW} is empty now, we go to Step 3.

Finally, assuming that the set to be reconstructed is of NE type, we can suppose that $(4, 2)$ is the source of F_1. Since there is no room for F_1 the algorithm terminates (there is no solution of NE type).



(a) (b) (c) (d)

Figure 4.5: An example which shows how Algorithm 2-RECQ' works

As regards the analysis of the complexity of Algorithm 2REC-Q', we can state the following

Theorem 4.2 *Algorithm 2-RECQ' solves the* $\textsc{Reconstruction}(\mathcal{Q}', \mathcal{L}_2)$ *task in* $O(mn \cdot \min\{m, n\})$ *time. The algorithm finds all sets of* $\mathcal{Q}'$ *with the given projections.*

Proof
Every row and column index can be in an equality position of both types at most once. This means that we have at most $\min\{m, n\}$ equality positions of NW type and at most $\min\{m, n\}$ equality positions of type 2. Moreover, equality positions can be found in time $O(m + n)$ by comparing the cumulated horizontal and vertical vectors.

Reconstructing the (directed) components of F – after assuming that an equality position (i, j) is in C_F – takes $O(mn)$ time (see Theorem 3.1). We have to test every equality position to see whether it is in C_F as well, so we get the execution time of $O(mn \cdot \min\{m, n\})$ in the worst case.

From Lemma 4.1.2 and Theorem 4.1 the reconstructed sets are Q-convex, and have the given projections H and V.

Moreover, Theorem 4.1 states that any element of C_F together with the projections and the type of F is sufficient to reconstruct F uniquely. Elements of C_F are equality positions as well – which follows from Lemma 4.2.1. Algorithm 2-RECQ' tests every equality position to see whether it is in C_F, and so the second part of the theorem follows. □

4.4 Connection with the $\mathcal{S}'_8$ Class

The reconstruction of hv-convex discrete sets has good theoretical foundations. The first reconstruction algorithm for this class using two projections was published in [64]. As it later transpired, the reconstruction task in this class is NP-complete [88], hence efforts have been made to find subclasses of the class of hv-convex sets where the reconstruction can be solved in polynomial time. An algorithm for reconstructing hv-convex polyominoes was presented in [13; 14]. Afterwards the method was improved to reconstruct discrete sets of $\mathcal{S}_8$ as well [30]. The worst case computational complexity of this algorithm is of $O(mn \cdot \log(mn) \cdot \min\{m^2, n^2\})$. In [36] another reconstruction algorithm was published for the class of hv-convex polyominoes which has a worst case time complexity of $O(mn \cdot \min\{m^2, n^2\})$. With a slight modification this algorithm is also suitable for reconstructing hv-convex 8-connected discrete sets [66]. After implementing the two methods for reconstructing sets of $\mathcal{S}_8$ [11] it turned out that the first algorithm ([30]) in general reconstructs the solutions faster than the other one ([66]) in almost every case that has been studied. During the testing of the programs a third algorithm – a combination of the previous ones – was developed, which has the same worst case computational complexity as the second algorithm but it remains as fast as the first one in the average case. In this sense it is the best known algorithm so far for the reconstruction task of hv-convex 8-connected sets. In this section we will discuss the connection between the classes $\mathcal{Q}'$ and $\mathcal{S}_8'$ and show that Algorithm 2-RECQ' can also be applied to reconstruct discrete sets of this latter class as well.

4.4.1 Components of a Set of $\mathcal{S}_8'$

Let $F \in \mathcal{S}_8'$ which has components $F_1, \ldots, F_k$ such that $\{i_l, \ldots, i_l'\} \times \{j_l, \ldots, j_l'\}$ is the SCDR of the l-th ($l = 1, \ldots, k$) component of F. In a similar way as in the $\mathcal{Q}'$ class, the sets of the row/column indices of the components of F consist of consecutive indices and they are disjoint since F is hv-convex. Without loss of generality we may again assume that (4.1) holds. Then a result similar to Lemma 4.1.1 can be stated.

Lemma 4.4.1 *Let $F \in \mathcal{S}_8'$ which has components $F_1, \ldots, F_k$ $(k \geq 2)$ with the SCDRs $\{i_l, \ldots, i_l'\} \times \{j_l, \ldots, j_l'\}$ $(l = 1, \ldots, k)$ such that (4.1) holds. Then one and only one of the following cases is possible*

(1) $1 = j_1 \leq j_1' < j_2 \leq j_2' < \cdots \leq j_k' = n$, *or*

(2) $n = j_1 \geq j_1' > j_2 \geq j_2' > \cdots \geq j_k' = 1$.

Proof
Evidently, both case (1) and case (2) cannot hold simultaneously.

For $k = 2$ the statement is trivial. Assume that $k > 2$ and neither case (1) nor case (2) is satisfied. Then there exist three components, say F_{l_1}, F_{l_2}, and F_{l_3}, such that $i_{l_1} < i_{l_2} < i_{l_3}$ and exactly one of the following relations hold

(a) $j_{l_1} < j_{l_3} < j_{l_2}$, or

(b) $j_{l_3} < j_{l_1} < j_{l_2}$, or

(c) $j_{l_2} < j_{l_1} < j_{l_3}$, or

(d) $j_{l_2} < j_{l_3} < j_{l_1}$.

Next, assume that case (a) is true and define a set S as

$$S = (\{i_{l_2}, \ldots, i_{l_2}'\} \times \{1, \ldots, j_{l_2} - 1\}) \cup (\{1, \ldots, i_{l_3} - 1\} \times \{j_{l_3}, \ldots, j_{l_3}'\}) \qquad (4.8)$$

Then it follows from the hv-convexity of F that $F \cap S = \emptyset$ and so the elements of F_{l_1} cannot be connected by an 8-path to elements of F_{l_2} – which contradicts the 8-connectedness of F (see Fig.4.6).

The proof is similar in cases (b), (c), and (d). □

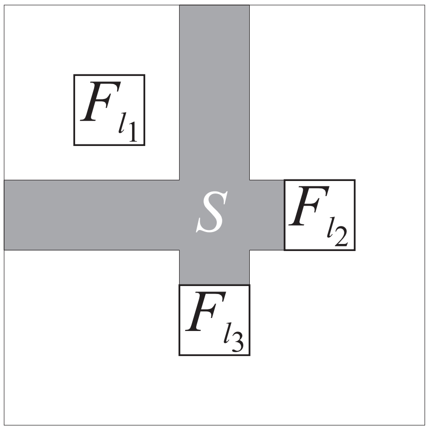


Figure 4.6: Proof of Lemma 4.4.1

In an analogous way as for sets in $\mathcal{Q}'$, we can say that the set $F \in \mathcal{S}_8'$ is of NW/NE type, if for F, case 1/2 of Lemma 4.4.1 holds, respectively.

Now we can determine the directedness of the components of F, which depends on the type of F.

Lemma 4.4.2 *Let $F \in \mathcal{S}_8'$ which has components $F_1, \ldots, F_k$ $(k \geq 2)$ with the SCDRs $\{i_l, \ldots, i_l'\} \times \{j_l, \ldots, j_l'\}$ $(l = 1, \ldots, k)$ such that (4.1) holds. If F is of NW/NE type then $F_1, \ldots, F_{k-1}$ are NW/NE-directed and $F_2, \ldots, F_k$ are SE/SW-directed, respectively.*

Proof

First, let us suppose that F is of NW type. Since F is 8-connected $F_1 \cup F_2$, $F_2 \cup F_3, \ldots, F_{k-1} \cup F_k$ are also 8-connected. Knowing the relative positions of the SCDRs of F_1 and F_2 (see Corollary 4.1), it is obvious that $F_1 \cup F_2$ is 8-connected if and only if

$$\hat{f}_{i_1,j_1} = \hat{f}_{i_1+1,j_1+1} = 1 \; . \tag{4.9}$$

Similarly, we get from the 8-connectedness of F that

$$\begin{aligned} \hat{f}_{i_2,j_2} = \hat{f}_{i_2+1,j_2+1} = 1 \; , \\ \vdots \\ \hat{f}_{i_{k-1},j_{k-1}} = \hat{f}_{i_{k-1}+1,j_{k-1}+1} = 1 \; . \end{aligned} \tag{4.10}$$

From Proposition 4.1 we know that (4.9) and (4.10) are actually equivalent to the NW-directedness of $F_1, \ldots, F_{k-1}$ and to the SE-directedness of $F_2, \ldots, F_k$.

In an analogous way, we can prove the second part of the theorem when F is of NE type. In this case

$$\begin{aligned} \hat{f}_{i_1,j_{k-1}+1} &= \hat{f}_{i_1+1,j_{k-1}} = 1 \ , \\ \hat{f}_{i_2,j_{k-2}+1} &= \hat{f}_{i_2+1,j_{k-2}} = 1 \ , \\ &\vdots \\ \hat{f}_{i_{k-1},j_1+1} &= \hat{f}_{i_{k-1}+1,j_1} = 1 \ . \end{aligned} \tag{4.11}$$

It is not hard to see that the positions given in (4.9), (4.10), and (4.11) are the source points of the corresponding components of F. □

The above observations can be summarized in the following theorem.

Theorem 4.3 $\mathcal{S}_8' \subset \mathcal{Q}'$.

Proof

Using Lemma 4.4.1 and Lemma 4.4.2, sets of $\mathcal{S}_8'$ have the same properties as sets of $\mathcal{Q}'$. The only difference is that the SCDRs of the components of a set in the class $\mathcal{Q}'$ may be separated (i.e. there may be empty rows and/or columns between two consecutive components), while in the $\mathcal{S}_8'$ class they are always 8-connected. □

Theorem 4.3 immediately implies that the $\textsc{Reconstruction}(\mathcal{S}_8',\ \mathcal{L}_2)$ task can also be solved in $O(mn \cdot \min\{m,n\})$ time. However, there is a nice improvement in the complexity of the reconstruction when the set of $\mathcal{Q}'$ to be reconstructed is not 8-connected.

Theorem 4.4 *The* $\textsc{Reconstruction}(\mathcal{Q}'\backslash\mathcal{S}_8',\ \mathcal{L}_2)$ *task can be solved in* $O(mn)$ *time. The number of solutions is at most two.*

Proof

Let $F \in \mathcal{Q}'\backslash\mathcal{S}_8'$. Since F is not 8-connected in the SCDR of F there must exist an empty row (say the i-th) such that $\widetilde{h}_i \neq 0$ or/and an empty column (say the j-th) such that $\widetilde{v}_j \neq 0$. Assume that the i-th row of the SCDR of F is empty and $\widetilde{h}_i \neq 0$. Let

$$i^* = \max\{i' \mid \widetilde{h}_{i'} = \widetilde{h}_i \text{ and } h_{i'} \neq 0\} \ , \text{ and} \tag{4.12}$$

$$j^* = \begin{cases} \min\{j' \mid \widetilde{v}_{j'} = \widetilde{h}_i\}, & \text{if } F \text{ is of type } 1 \ , \\ \max\{j' \mid \widetilde{v}_n - \widetilde{v}_{j'-1} = \widetilde{h}_i\}, & \text{if } F \text{ is of type } 2 \ . \end{cases} \tag{4.13}$$

Notice that (i^*, j^*) can be found in $O(m+n)$ time. Since F is Q-convex the position (i^*, j^*) must be the source of a NW/NE-directed component when F is of NW/NE type, respectively. Therefore $(i^*, j^*) \in C_F$ and the solution is uniquely determined using Theorem 4.1. Consequently, ambiguity can only arise in the type of F, so the number of solutions is at most two. In some cases there exist solutions of both types (see Fig. 4.7).

An analogous proof can be given when the SCDR of F has an empty column. □

Figure 4.7: Two sets of $\mathcal{Q}'$ with an empty row and column and the same horizontal and vertical projections

4.4.2 Experimental Results

In [11] an algorithm (called Algorithm C) is presented which has a worst case complexity of $O(mn \cdot \min\{m^2, n^2\})$ and so far has the best average time complexity for reconstructing hv-convex 8-connected discrete sets using two projections. In order to compare the average execution times of this algorithm and Algorithm 2-RECQ' on the $\mathcal{S}_8'$ class we generated sets of $\mathcal{S}_8'$ at random from a uniform distribution. This was carried out by the method given in [11], which generates sets of $\mathcal{S}_8$ that have fixed row and column numbers from a uniform distribution. This method is also suitable for generating sets of $\mathcal{S}_8'$ from a uniform distribution (if the generated set is 4-connected then we simply omit it). In our experiments we generated discrete sets of $\mathcal{S}_8'$ with different sizes. Each set of test data consisted of 1000 hv-convex 8-connected but not 4-connected discrete sets. Then we reconstructed them by using both algorithms. We used a PC with AMD Athlon processor of 1.4 GHz and 1.5 GB RAM under Red Hat Linux release 7.3. The programs were written in C++.

The average execution times in seconds for obtaining all the solutions of different test sets are presented in Table 4.1. The results indicate that not just the worst case complexity of our algorithm is better (see Theorem 4.2), but also that its average execution time is much better on all of the five test sets.

Table 4.1: Average execution times in seconds of Algorithm 2-REC8' and Algorithm C in [11], which both depend on the size of the discrete set

Size $n \times n$	2-REC8'	C in [11]
20×20	0.000272	0.011511
40×40	0.001064	0.032524
60×60	0.002597	0.065897
80×80	0.004746	0.116505
100×100	0.007831	0.178633

4.5 Summary

In this chapter we studied the problem of reconstructing Q-convex discrete sets from their horizontal and vertical projections, making use of the assumption that the set to be reconstructed consists of two or more components. First, we showed that sets belonging to this class (denoted by $\mathcal{Q}'$) can be decomposed into components which can be uniquely reconstructed by using horizontal and vertical projections. Second, we recognized the very important fact that the SCDRs of the components of a set $F \in \mathcal{Q}'$ can be arranged in just two possible ways. Omitting possible empty rows and columns they are connected to each other by their bottom right and upper left (sets of NW type) or by their bottom left and upper right (sets of NE type) positions. Next, we introduced the concept of equality positions. Then, with the aid of equality positions, the directed components of F could be reconstructed independently. Based on these results we described an algorithm 2REC-Q' for handling the $\textsc{Reconstruction}(\mathcal{Q}\text{'}, \mathcal{L}_2)$ task in $O(mn \cdot \min\{m, n\})$ time – in the worst case. Our algorithm has the advantage that it finds all the solutions of the same reconstruction problem in polynomial time. We should also mention that the algorithm developed can be used to reconstruct discrete sets that are Q-convex *along the horizontal and vertical directions*. But deciding whether there is a way of adapting our algorithm to the reconstruction of discrete sets which are Q-convex along other directions (i.e., not along the horizontal and vertical ones) we require more study.

We then showed that the $\mathcal{S}_8'$ class of hv-convex 8- but not 4-connected discrete sets form a subclass of $\mathcal{Q}'$. Afterwards our newly developed algorithm was applied to solve the reconstruction task in this class as well. We compared algorithm 2REC-Q' with a previous (more general) one given in [11] from the viewpoint of average running time. Our experimental results demonstrate that algorithm 2-RECQ' also outperforms the earlier algorithm in average running time. It is also interesting to note that the assumption on a set being 8-connected but not 4-connected leads to an improvement in the reconstruction complexity, while other algorithms which exploit topological properties like thinning algorithms usually do not run as fast on the $\mathcal{S}_8'$ class as on the $\mathcal{S}_4$ class (cf. [71]).

We showed too that there is an improvement in the reconstruction complexity if we wish to reconstruct sets of $\mathcal{Q}' \setminus \mathcal{S}_8'$ using horizontal and vertical projections. For this class we found that the $\textsc{Reconstruction}(\mathcal{Q}' \setminus \mathcal{S}_8', \mathcal{L}_2)$ task is solvable in $O(mn)$ time and the number of sets in this class which has the same horizontal and vertical projections is at most two.

Our algorithm in its original form was described to work in the class of hv-convex 8- but not 4-connected discrete sets in [4]. The theoretical results which form the basis of the algorithm was later proved in [5]. The extended version of the algorithm which is also able to reconstruct sets of $\mathcal{Q}'$ is described in [3]. In the same work the reader can find the complexity and ambiguity results for the $\mathcal{Q}' \setminus \mathcal{S}_8'$ class.

Chapter 5

Reconstruction from Four Projections

A lot of work has been done in designing efficient reconstruction algorithms for different classes of discrete sets using just two projections (e.g. [11; 13; 23; 36; 42; 64; 66; 80]). However, the theory of reconstructing discrete sets from three or more projections is, currently, far from complete. In contrast to the positive results for the $\textsc{Consistency}$ $(\mathcal{F},\ \mathcal{L}_2)$ task with the $\textsc{Reconstruction}(\mathcal{F},\ \mathcal{L}_2)$, and $\textsc{Uniqueness}(\mathcal{F},\ \mathcal{L}_2)$ tasks [80] it was shown that all three problems become NP-hard if the set of directions $\mathcal{L}_2$ is replaced by an arbitrary set $\mathcal{L}$ of directions such that $|\mathcal{L}| \geq 3$ [48]. Nevertheless, there are a few algorithms that can reconstruct some special discrete sets using three or more projections [12; 24]. In this chapter we will describe a method that uses four projections to reconstruct a discrete set which has at least two components. Our algorithm is initially based on some of the observations of Chapter 4 and works for a special class of discrete sets that have disjoint components called decomposable discrete sets. We will also examine whether this method can be adapted to facilitate the reconstruction process in the classes $\mathcal{Q}'$, $\mathcal{S}_8'$ and $\mathcal{HV}$.

5.1 Decomposability: A New Property for Reconstruction

Let us first introduce some new concepts that will be useful later on. Given two discrete sets C and D represented by the binary matrices $\hat{C} = (\hat{c}_{ij})_{m_1 \times n_1}$ and $\hat{D} = (\hat{d}_{ij})_{m_2 \times n_2}$, respectively, we say that we get the discrete set F represented by the binary matrix $\hat{F} = (\hat{f}_{ij})_{m_3 \times n_3}$ by *northwest gluing* (or NW gluing in short) C to D if

$$\hat{F} = \begin{pmatrix} \hat{C} & \mathbf{0} \\ \mathbf{0} & \hat{D} \end{pmatrix} , \tag{5.1}$$

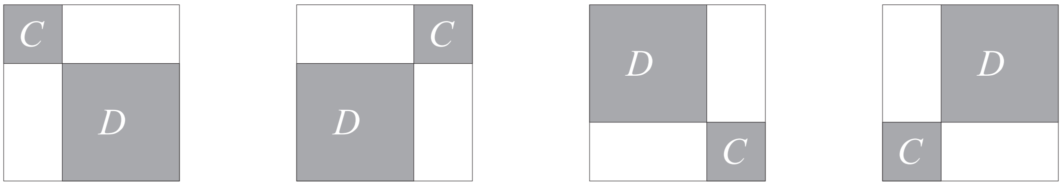


Figure 5.1: The gluing operators: NW, NE, SE, and SW gluings (from left to right)

such that $m_3 \geq m_1 + m_2$ and $n_3 \geq n_1 + n_2$. We should mention here that in the resulting set F there can be empty rows or/and columns between the subsets C and D (namely, when $m_3 > m_1 + m_2$ or/and $n_3 > n_1 + n_2$). If C is a single component then we say that C is the *NW component* of F. NE, SE, SW gluings and components are defined in a similar way (see Fig. 5.1).

Two components F_1 and F_2 of F are *disjoint* if both the sets of the row indices and the sets of the column indices of F_1 and F_2 are disjoint. We say that a discrete set F made up of k $(k \geq 2)$ components is *decomposable* if all of the following properties are fulfilled:

(α) the components of F are uniquely reconstructible from their horizontal and vertical projections in the class of all polyominoes in polynomial time,

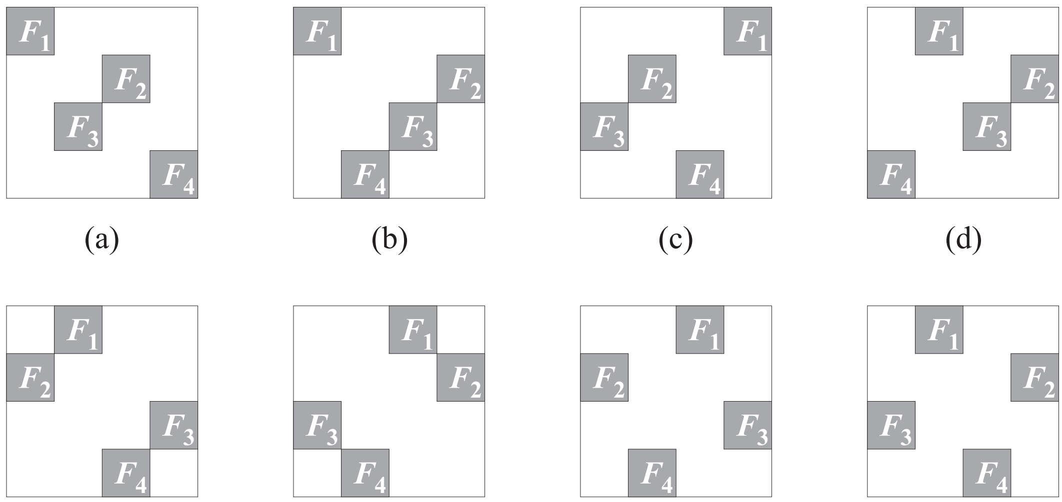
(β) the sets of the row and column indices of the components' smallest containing discrete rectangles (SCDR) are pairwisely disjoint,

(γ) in the case when $k > 2$ we get F by gluing a single component to a decomposable discrete set made up of $k-1$ components using one of the four gluing operators.

One direct consequence of this definition is that we find that every discrete set which consists of one component is non-decomposable and every discrete set which consists of two or three components and satisfies properties (α) and (β) is decomposable as they also satisfy property (γ). Figure 5.2 shows some configurations of the SCDRs of the components which satisfy property (γ) when the set satisfies property (β) and consists of four components. The same figure also shows some situations where property (γ) is not satisfied.

Actually, property (α) is usually not satisfied. But one can force the components to belong a certain class $\mathcal{C}$ of polyominoes such that every polyomino of $\mathcal{C}$ is uniquely determined in this class by its horizontal and vertical projections. Some of these classes were analysed in Chapter 3. In this case in the following when necessary we will say that the discrete set is decomposable w.r.t. the class $\mathcal{C}$.

The type of a decomposable discrete set can be defined in much the same way as that for sets of $\mathcal{Q}'$. If by omitting empty rows and columns, the SCDRs of the components of a decomposable discrete set F are related to each other by their bottom right hand and upper left hand (bottom left hand and upper right hand) corners then we say that F is of *NW/NE type*, respectively. That is, F is of NW type if, during the construction of F, only NW or/and SE gluings are used. Similarly, F is of NE type if we obtain it by using just SW or/and NE gluings.



(a) (b) (c) (d)

(e) (f) (g) (h)

Figure 5.2: Configurations of the SCDRs of four components F_1, F_2, F_3, and F_4 which satisfy property (γ) (first row), and do not satisfy it (second row)

Now let us introduce the following set of notation.

- $\mathcal{DEC}$ for the class of decomposable discrete sets;
- $\mathcal{DEC}_{\mathcal{C}}$ for the class of decomposable discrete sets w.r.t. the class $\mathcal{C}$;
- $\mathcal{S}^{NW}$, $\mathcal{S}^{NE}$ for the class of decomposable discrete sets of NW/NE type, respectively.

Manifestly, $\mathcal{S}^{NW} \cap \mathcal{S}^{NE} = \emptyset$ and $\mathcal{S}^{NW} \cup \mathcal{S}^{NE} \subset \mathcal{DEC}$ (for the real inclusion see Figs. 5.2a-d, say).

5.1.1 The Center of a Decomposable Discrete Set

We can provide a description of decomposable discrete sets by stating and proving the following lemma:

Lemma 5.1.1 *A discrete set F is decomposable if and only if it satisfies property (α) and there exists a sequence of discrete sets $F^{(1)}, \ldots, F^{(k)}$ $(k \geq 2)$ such that $F^{(1)}$ consists of one component, $F^{(k)} = F$, and for each $l = 1, \ldots, k-1$ we get $F^{(l+1)}$ by gluing a component to $F^{(l)}$ using a gluing operator.*

Proof
It is a direct consequence of the definition of decomposability. □

For a given discrete set $F \in \mathcal{DEC}$ the sequence described in this lemma is not uniquely determined. But we will call any sequence which satisfies the properties of it as a *gluing sequence* of F.

Example 5.1.1
The discrete set $F = \{(1,1),(2,2)\}$ has two gluing sequences: $F^{(1)} = \{(1,1)\}$, $F^{(2)} = \{(1,1),(2,2)\}$, and $G^{(1)} = \{(2,2)\}$, $G^{(2)} = \{(1,1),(2,2)\}$.

Naturally, every gluing sequence of a given decomposable discrete set must have the same length (namely, a length that is always equal to the number of components of the discrete set). If the discrete set $F \in \mathcal{DEC}$ consists of two components then, obviously, either $F \in \mathcal{S}^{NW}$ or $F \in \mathcal{S}^{NE}$. Moreover, from property (γ) it follows that if $F^{(1)}, \ldots, F^{(k)}$ is a gluing sequence of $F \in \mathcal{DEC} \setminus (\mathcal{S}^{NW} \cup \mathcal{S}^{NE})$ there exists a unique integer $1 < j < k$ such that $F^{(j)} \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ and $F^{(j+1)} \notin \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$. The following lemma shows the connection between two arbitrary gluing sequences of the same decomposable discrete set.

Lemma 5.1.2 *Let $F^{(1)}, \ldots, F^{(k)}$ and $G^{(1)}, \ldots, G^{(k)}$ be two different gluing sequences of the same discrete set $F \in \mathcal{DEC} \setminus (\mathcal{S}^{NW} \cup \mathcal{S}^{NE})$. Furthermore, let $1 < j_f, j_g < k$ such that $F^{(j_f)}, G^{(j_g)} \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ and $F^{(j_f+1)}, G^{(j_g+1)} \notin \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$. Then $j_f = j_g$ and $F^{(j_f)} = G^{(j_g)}$.*

Proof
Consider $F \in \mathcal{DEC} \setminus (\mathcal{S}^{NW} \cup \mathcal{S}^{NE})$. If $F^{(j_f)} \neq G^{(j_g)}$ then there exists a position (p,q) such that $(p,q) \in F^{(j_f)} \setminus G^{(j_g)}$ or $(p,q) \in G^{(j_g)} \setminus F^{(j_f)}$.

Assume that $(p,q) \in F^{(j_f)} \setminus G^{(j_g)}$ and the SCDR of $G^{(j_g)}$ is $I \times J = \{i', \ldots, i''\} \times \{j', \ldots, j''\}$. Then, on the basis of property (β), $(p,q) \in \bar{I} \times \bar{J}$ where $\bar{I} = \{1, \ldots, m\} \setminus I$ and $\bar{J} = \{1, \ldots, n\} \setminus J$. Since $(p,q) \in F^{(j_f)}$ and $F^{(j_f)} \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ it follows that $F^{(j_f)} \subset \bar{I} \times \bar{J}$ hence $F^{(j_f)} \cap G^{(j_g)} = \emptyset$. Since $F^{(j_f+1)}, G^{(j_g+1)} \notin \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ the components of $G^{(j_g)}$ cannot be glued to a subset of F which contains $F^{(j_f)}$ or vice versa thus $F^{(j_f)} \cup G^{(j_g)} \not\subseteq F$ is a contradiction.

The proof is similar when it is assumed that $(p,q) \in G^{(j_g)} \setminus F^{(j_f)}$. □

From Lemma 5.1.2 we can say that for every set $F \in \mathcal{DEC}$ there exists a unique integer j such that for every gluing sequence $F^{(1)}, \ldots, F^{(k)} = F$ $F^{(j)}$ is the same, $F^{(j)} \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$, and $j = k$ or $F^{(j+1)} \notin \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$. The uniquely determined set $F^{(j)}$ is called the *center* of F and shall be denoted by $C(F)$.

Example 5.1.2
If the configuration of the components of the set F is the same as that given in Fig. 5.2a then $C(F) = F_2 \cup F_3$, while for the case given in Fig. 5.2b $C(F) = F_2 \cup F_3 \cup F_4$.

5.1.2 Finding a Component

Making use of properties (α) and (β) in the reconstruction of a decomposable discrete set, it is sufficient to identify the SCDRs of the components. In this subsection we shall only deal with NW components. However, the results given below can be easily modified to determine NE, SE, and SW components as well. In order to find the SCDR of a NW component we will first supply a necessary condition that is quite similar to that stated in Lemma 4.2.1.

Lemma 5.1.3 *Let $F \in \mathcal{DEC}$. If (i,j) is the bottom right hand position of the SCDR of the NW component of F then i is the smallest integer for which there exists an integer j with $\widetilde{h}_i = \widetilde{v}_j = \widetilde{a}_{i+j-1} > 0$ and $a_{i+j} = 0$.*

Proof

Define a set E as follows

$$E = (\{1,\dots,i\} \times \{j+1,\dots,n\}) \cup (\{i+1,\dots,m\} \times \{1,\dots,j\}) \ . \tag{5.2}$$

If (i,j) is the bottom right hand position of the SCDR of the NW component of F, then clearly $\widetilde{h}_i > 0$, $\widetilde{v}_j > 0$, and $\widetilde{a}_{i+j-1} > 0$.

Moreover, $F \cap E = \emptyset$, and so

$$\begin{aligned} \widetilde{h}_i = \sum_{t=1}^{i} h_t &= |F \cap \{1,\dots,i\} \times \{1,\dots,n\}| = |F \cap \{1,\dots,i\} \times \{1,\dots,j\}| \\ &= |F \cap \{1,\dots,m\} \times \{1,\dots,j\}| = \sum_{t=1}^{j} v_t = \widetilde{v}_j \ . \end{aligned} \tag{5.3}$$

Furthermore, for each $k = 1,\dots,m+n-1$ define the set A_k by

$$A_k = \{(i,j) \in \mathbb{Z}^2 \ : \ i+j = k+1\} \ . \tag{5.4}$$

Note that $a_k = |F \cap A_k|$ and $(F \cap A_k) \cap E \subseteq F \cap E = \emptyset$ for every $k = 1,\dots,m+n-1$. Then we find that

$$\widetilde{a}_{i+j-1} = \sum_{k=1}^{i+j-1} |F \cap A_k| = |F \cap \{1,\dots,i\} \times \{1,\dots,j\}| = \widetilde{h}_i = \widetilde{v}_j \ . \tag{5.5}$$

In addition, $A_{i+j} \subseteq E$ implies $F \cap A_{i+j} \subseteq F \cap E = \emptyset$ and we get that

$$a_{i+j} = |F \cap A_{i+j}| \leq |F \cap E| = 0 \ . \tag{5.6}$$

Next, assume that an integer $i' < i$ exists for which an integer j' exists such that $\widetilde{h}_{i'} = \widetilde{v}_{j'} = \widetilde{a}_{i'+j'-1} > 0$ and $a_{i'+j'} = 0$. Evidently, in this case $j' < j$. Since (i,j) is the bottom right hand position of the SCDR of the NW component of F and every component is a polyomino we see that the 1st, $\dots$, $(i+j)$-th coordinates of the antidiagonal projection of F have to be of the form $(0,\dots,0,a_{k_1},\dots,a_{k_2},0,\dots,0)$, where $1 \leq k_1 \leq k_2 < i+j$ and $a_l \neq 0$ for every $k_1 \leq l \leq k_2$. But then $a_{i'+j'} = 0$ only if $i'+j' < k_1$ or $i'+j' > k_2$.

If $i'+j' < k_1$ then $\widetilde{a}_{i'+j'-1} = 0$ which is of course a contradiction.

Otherwise, if $i'+j' > k_2$ then $\widetilde{h}_i \geq \widetilde{h}_{i'}$ since $i > i'$. If $\widetilde{h}_i > \widetilde{h}_{i'}$ then $\widetilde{h}_{i'} = \widetilde{a}_{i'+j'-1} = \widetilde{a}_{i+j-1} = \widetilde{h}_i > \widetilde{h}_{i'}$ is a contradiction. If $\widetilde{h}_i = \widetilde{h}_{i'}$ then $h_i = 0$, which contradicts the assumption that (i,j) is the bottom right hand position of the SCDR of the NW component of F. $\square$

As luck would have it, Lemma 5.1.3 does not give a sufficient condition for identifying the SCDR of a component of an $F \in \mathcal{DEC}$. For example, if the decomposable discrete set F is like that given in Fig. 5.3 then for the $(2,2)$ position the conditions of Lemma 5.1.3 hold (since $\widetilde{h}_2 = \widetilde{v}_2 = \widetilde{a}_3 = 3$ and $a_4 = 0$) but F has no NW component.

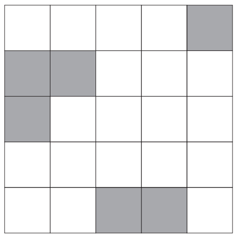


Figure 5.3: An example showing that the converse of Lemma 5.1.3 does not hold

The example of Fig. 5.3 tells us that some more conditions need to be introduced to find NW components. With the aid of the following theorem it is possible to test whether the decomposable discrete set has an NW or SE component.

Theorem 5.1 *Let $\mathcal{C}$ be an arbitrary class of polyominoes which can be uniquely reconstructed in this class from their horizontal and vertical projections in polynomial time. Let $F \in \mathcal{DEC}_{\mathcal{C}}$, $\mathcal{H}(F) = (h_1, \ldots, h_m)$, $\mathcal{V}(F) = (v_1, \ldots, v_n)$, and $\mathcal{A}(F) = (a_1, \ldots, a_{m+n-1})$. If (i,j) is a position that satisfies the necessary conditions of Lemma 5.1.3 such that a polyomino $P \in \mathcal{C}$ exists with $\mathcal{H}(P) = (h_1, \ldots, h_i)$, $\mathcal{V}(P) = (v_1, \ldots, v_j)$, and $\mathcal{A}(P) = (a_1, \ldots, a_{i+j-1})$ then P is the NW component of F or/and F has a SE component. If no such position exists then F has no NW component.*

Proof

For each $k = 1, \ldots, m+n+1$ define the set A_k by (5.4). Moreover, let $T = \bigcup_{k=1}^{i+j-1} A_k$ and $Q = F \cap T$.

We have $\mathcal{A}(Q) = \mathcal{A}(P)$ so the sum of the terms of $\mathcal{H}(Q)$ is equal to the sum of the terms in $\mathcal{H}(P)$. Furthermore, $\mathcal{H}(Q) \leq \mathcal{H}(P)$, so $\mathcal{H}(Q) = \mathcal{H}(P)$. In a similar way we get $\mathcal{V}(Q) = \mathcal{V}(P)$. Then $Q \subseteq \{1, \ldots, i\} \times \{1, \ldots, j\}$ thus $F \cap E = \emptyset$, where E is defined by (5.2). Hence F has neither an NE nor SW component, i.e. it has a NW or a SE component.

If F has a NW component then it must be the polyomino P based on Lemma 5.1.3 and property (α). Thus the first part of the theorem is proven.

The second part of the theorem follows from the fact that the position which satisfies the necessary conditions of Lemma 5.1.3 is uniquely determined. $\square$

Note that if the conditions of the above theorem hold then in some cases the discrete set can have both NW and SE components – for example when the discrete set is in $\mathcal{S}^{NW}$. However, Theorem 5.1 does not tell us whether the discrete set necessarily has a NW component (see Fig. 5.4 for instance).

Remark 5.1.1

The polyomino in Fig. 5.4b is not uniquely determined by its horizontal and vertical projections. Its symmetrical pair has the same projections. But assuming the restriction

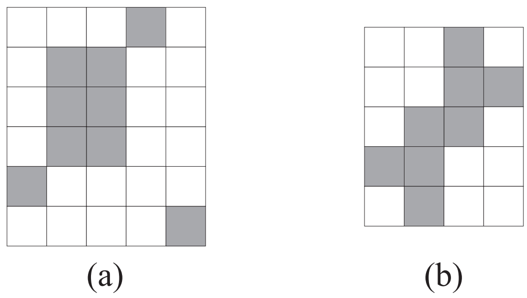


Figure 5.4: (a) A discrete set which has no NW component although the $(5, 4)$ position satisfies the conditions of Theorem 5.1 with the polyomino in (b)

that the discrete set to be reconstructed can have only rectangle polyominoes or the polyomino shown in Fig. 5.4b as components one can guarantee that each component is uniquely determined by its horizontal and vertical projections in the class $\mathcal{C}$ which consists of all the rectangle polyominoes and the polyomino of Fig. 5.4b. That is, Fig. 5.4 provides an example which shows that Theorem 5.1 does not always give a sufficient condition for the existence of a NW-component in the class $\mathcal{DEC}_{\mathcal{C}}$.

If a decomposable discrete set F is in $\mathcal{S}^{NW}/\mathcal{S}^{NE}$ then with the aid of the NW/NE-version of Theorem 5.1 it is possible to find the SCDR of the NW/NE component of F, respectively. This means that once we have decomposed all the components around the center of F, Theorem 5.1 will provide an effective tool for reconstructing the center itself. On the basis of the following theorem one can find the NW component of F (if exists) if $F \in \mathcal{DEC} \setminus (\mathcal{S}^{NW} \cup \mathcal{S}^{NE})$ as well.

Theorem 5.2 *Assume that $F \in \mathcal{DEC} \setminus (\mathcal{S}^{NW} \cup \mathcal{S}^{NE})$ and a position (i, j) satisfies the conditions of Theorem 5.1 with a polyomino P. Moreover, let $\{i_1, \dots, i_2\} \times \{j_1, \dots, j_2\}$ be the SCDR of $C(F)$. Then P is the NW component of F if and only if there exists $i' \in \{i_1, \dots, i_2\}$ such that $i < i'$ or there exists $j' \in \{j_1, \dots, j_2\}$ such that $j < j'$.*

Proof
From the definition of the center it is obvious that $C(F) \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$. Now assume that $C(F) \in \mathcal{S}^{NW}$. Since Theorem 5.1 holds F cannot have a NE or SW component.

If F has a NW component then the gluing sequence of F is of the form

$$F^{(1)}, \dots, F^{(l)} = C(F), F^{(l+1)}, \dots, F^{(k)} = F \ , \tag{5.7}$$

where $F^{(l+1)}$ must exist (since $F \notin \mathcal{S}^{NW}$) and we get it from $C(F)$ by a NE or SW gluing. Clearly, in both cases for the bottom right hand (i, j) position of the SCDR of the NW component of F $i < i_1$ and $j < j_1$ must hold (plainly, $i < i'$ and $j < j'$ for any $i' \in \{i_1, \dots, i_2\}$ and $j' \in \{j_1, \dots, j_2\}$ is also true).

On the other hand, if $i < i'$ for an arbitrary $i' \in \{i_1, \dots, i_2\}$ or $j < j'$ for an arbitrary $j' \in \{j_1, \dots, j_2\}$ then the center lies to the southeast of (i, j) and $F \cap E = \emptyset$ where E is defined by (5.2) (since (i, j) satisfies the conditions of Theorem 5.1). So F must have a NW component which, by Theorem 5.1, must be P .

The proof is similar when $C(F) \in \mathcal{S}^{NE}$. □

Finally, an observation similar to Corollary 4.2 will help speed up the reconstruction process.

Corollary 5.1 *If $F, F' \in \mathcal{DEC}$ are different solutions of the same reconstruction task then the SCDRs of $C(F)$ and $C(F')$ are disjoint.*

5.1.3 The Reconstruction Algorithm

An algorithm can be outlined to reconstruct decomposable discrete sets with given horizontal, vertical, diagonal, and antidiagonal projections. But first we will describe a procedure for decomposing a NW component of a discrete set that is decomposable w.r.t the class $\mathcal{C}$.

Procedure DecomposeNW for decomposing a NW component of a set of $\mathcal{DEC}_{\mathcal{C}}$

Step 1 find the position (i, j) for which $\widetilde{h}_i = \widetilde{v}_j = \widetilde{a}_{i+j-1} > 0$ and $a_{i+j} = 0$;
if no such position exists **then return**(no component);
Step 2 **if** $j \geq l$ **then return**(no component);
Step 3 construct a polyomino $P \in \mathcal{C}$ with $\mathcal{H}(P) = (h_1, \ldots, h_i)$, $\mathcal{V}(P) = (v_1, \ldots, v_j)$, and test if $\mathcal{A}(P) = (a_1, \ldots, a_{i+j-1})$;
if no such polyomino exists **then return**(no component);
Step 4 $F := F \cup P$;
Step 5 update H, V, D, and A according to the projections of P;
Step 6 **return**;

The procedures DecomposeNE, DecomposeSE, and DecomposeSW for decomposing components from the other three directions can be outlined similarly. The main algorithm for reconstructing discrete sets that are decomposable w.r.t. the class $\mathcal{C}$ uses these four procedures to reconstruct the solutions component by component. Evidently, in most general form $\mathcal{C}$ may be equal to the class of all polyominoes hence the same algorithm can be used to reconstruct sets of $\mathcal{DEC}$ as well. The algorithm we used can be described in the following concise way.

This algorithm works as follows. First, the set of solutions is S is empty and the set C contains the possible columns of a solution's center (Step 1).

Next, in Step 2 we define $F := \emptyset$, restore the vectors H, V, D, and A to their original values and we try to find a solution such that the leftmost column of its center is $l = \min C$.

In Step 3 we first try to decompose a NW component. This is done by Procedure DecomposeNW which looks for the SCDR of the NW component of F by trying to find the uniquely determined position which satisfies the conditions of Theorems 5.1 and 5.2 with a polyomino P (Steps 1–3 of Procedure DecomposeNW).

If no such position exists then either the assumption that the set to be reconstructed has a NW component was false or the l-th column cannot be a column of the center. In both cases the procedure simply returns to the main algorithm.

Algorithm 4-RECDEC for reconstructing sets of $\mathcal{DEC}_\mathcal{C}$ from four projections

Input: Four compatible vectors $H \in \mathbb{N}_0^m$, $V \in \mathbb{N}_0^n$, $D \in \mathbb{N}_0^{m+n-1}$, and $A \in \mathbb{N}_0^{m+n-1}$.
Output: The set S which contains all the discrete sets of $\mathcal{DEC}_\mathcal{C}$ with the horizontal, vertical, diagonal, and antidiagonal projections H, V, D, and A, respectively.

Step 1 $C := \{1, \ldots, n\} \setminus \{j \mid v_j = 0\}$; $S = \emptyset$;
Step 2 $F := \emptyset$; $l := \min C$; **restore** H, V, D, and A;
Step 3 **repeat**
call DecomposeNW;
if (no component) **then** call DecomposeNE;
if (no component) **then** call DecomposeSE;
if (no component) **then** call DecomposeSW;
until (no component);
Step 4 try to reconstruct the last component;
Step 5 **if** $D = 0$ **and** $A = 0$ **then** $\{$ $S := S \cup \{F\}$;
$C := C \setminus \{$columns of $C(F)\}$; $\}$
Step 6 **if** $C = \emptyset$ **then return** S **else goto** Step 2;

Otherwise, the polyomino P is assumed to be the NW component of F, we simulate how this component affects the projections of F (Steps 4 and 5 of Procedure DecomposeNW) and then we return to the main algorithm.

If we were not able to find a NW component then we try to decompose a NE, SE, then SW component in this order. Note that Theorem 5.2 cannot be applied on sets of $F \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ but this does not cause our algorithm to fail since in this case we will decompose a SE/SW-component instead of a NW/NE component if the set is of NW/NE type, respectively.

If we found a component (which will be one of the four kinds) then we go on and try to decompose other components from the remaining set. We repeat this until we cannot decompose a component. Then the remaining set is undecomposable hence it must consist of a single component. We try to reconstruct this last component in Step 4.

Next, in Step 5 we check whether the reconstructed set F has the given projections, and if so we set $S := S \cup \{F\}$. Using Corollary 5.1 the columns of $C(F)$ cannot be the columns of any other solution hence we can eliminate the columns of $C(F)$ from C and go on to Step 2 to try to find other solutions (if any) with other centers.

Even if the part of the discrete set which is not still reconstructed is an element of $\mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ we can continue to impose a condition using the variable l (normally imposed by Theorem 5.2) because there is always a component which does not contain the l-th column.

In each iteration we start the reconstruction from scratch using the original vectors H, V, D, and A (see Step 2) and we put the reconstructed set F into S if it is a different solution from the given reconstruction task.

Algorithm 4-RECDEC examines, in each iteration, whether a solution with a certain center exists by assuming that a given column is also a center column. Using Theorem 5.2 this strategy can also be applied to the center rows. If $m < n$ then we choose the latter version of our strategy (this decision is taken in Step 1).

Example 5.1.3
Suppose we want to reconstruct discrete sets of $\mathcal{DEC}_\mathcal{C}$ (where $\mathcal{C}$ is the same class as mentioned in Remark 5.1.1) from the vectors

$$\begin{aligned} H &= (1,2,2,2,1,1,2,2,2,1,1)\ , \\ V &= (1,3,3,1,0,1,3,3,1,1)\ , \\ D &= (0,0,0,0,0,0,1,2,2,3,4,2,2,1,0,0,0,0,0,0)\ , \\ A &= (0,0,1,3,3,1,0,0,0,0,0,0,1,3,3,1,0,0,0,1)\ . \end{aligned}$$

First, it is assumed that the first column is a column of the center so the position (i,j) which satisfies $\tilde{h}_i = \tilde{v}_j = \tilde{a}_{i+j-1} > 0$ and $a_{i+j} = 0$ contradicts Theorem 5.2 (Fig. 5.5a).

Next, the algorithm reconstructs SW components in two iterations (Figs. 5.5b and 5.5c).

After, in the next two iterations the algorithm decomposes two NE components in the remaining part of the discrete set (Figs. 5.5d and 5.5e).

Next, the algorithm reconstructs the last component and finds a solution F (Fig. 5.5f). After eliminating the columns of $C(F)$ from C we go on by assuming that the sixth column is a column of a possible another solution's center. With this assumption we decompose a NW component (Fig. 5.5g).

Then, a NW component cannot be decomposed in the remaining part as its SCDR contradicts Theorem 5.2 (Fig. 5.5h).

In the following we will decompose a SE, a NE and a NE component once again (Fig. 5.5i, 5.5j, and 5.5k, respectively).

Reconstructing the last component (Fig. 5.5l) we find another solution F'.

Next, assuming that the last column is the center we decompose two NW components (Figs. 5.5m and 5.5n) and reconstruct the last component (Fig. 5.5o).

However, the reconstructed set does not have the given diagonal projection therefore the algorithm terminates and the number of solutions is two.

Using the same method as in Example 5.1.3 it is not hard to construct examples where the number of solutions is k for an arbitrary $k \geq 2$ (see Fig. 5.6 for the case $k = 3$).

Turning to the analysis of Algorithm 4-RECDEC, we can state the following

Theorem 5.3 *Let $\mathcal{C}$ be an arbitrary class of polyominoes that can be reconstructed in this class uniquely from their horizontal and vertical projections in polynomial time (in $O(f(m,n))$ time, say). Then Algorithm 4-RECDEC solves the* $\textsc{Reconstruction}(\mathcal{DEC}_\mathcal{C}, \mathcal{L}_4)$ *task in $O(\min\{m,n\} \cdot f(m,n))$ time. The algorithm finds all sets of $\mathcal{DEC}_\mathcal{C}$ with the given projections.*

Proof
Let F be an arbitrary set of the solution set S of Algorithm 4-RECDEC.

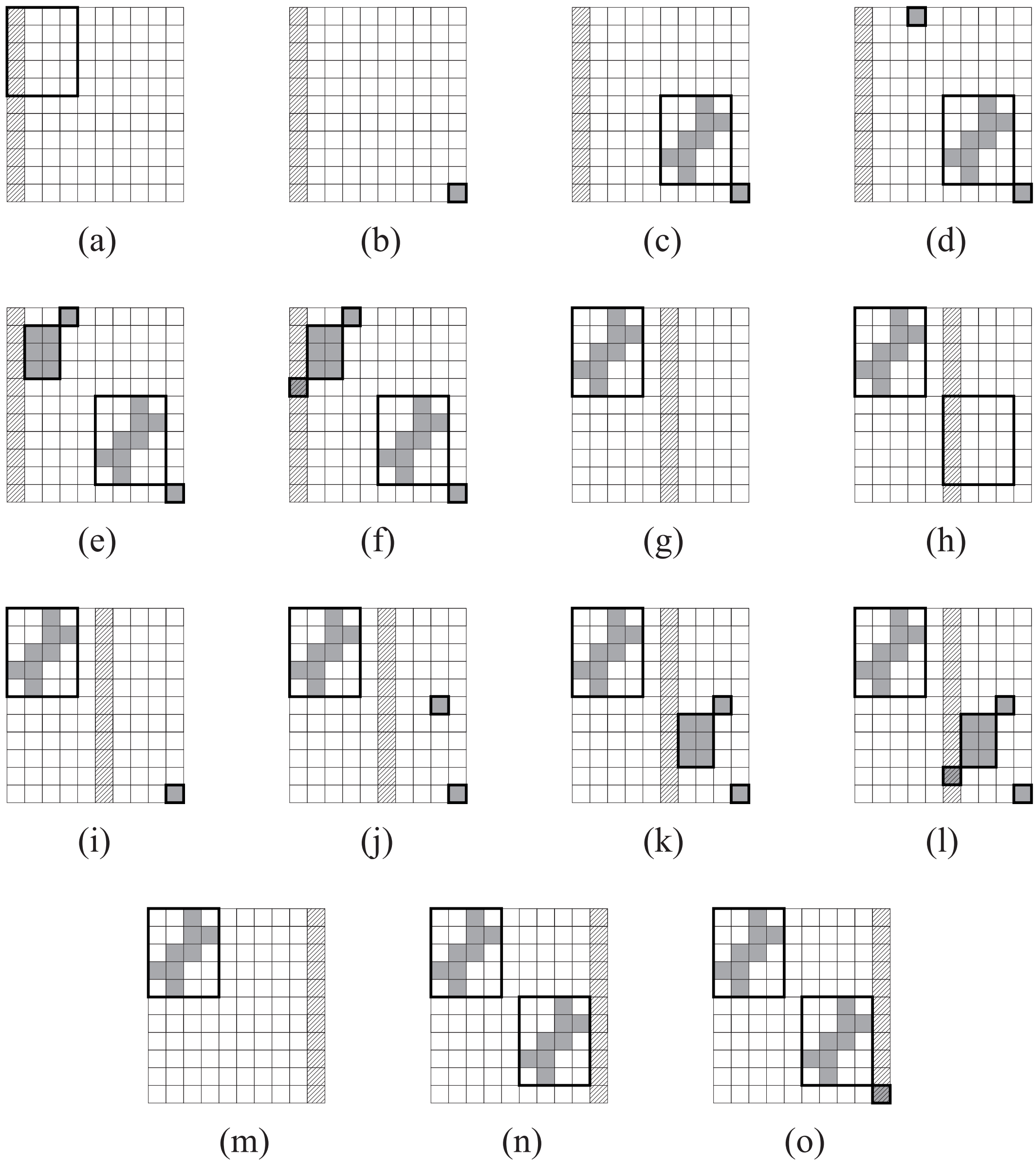


Figure 5.5: An example which shows how Algorithm 4-RECDEC works. The located discrete rectangles – which must contain the next component – are denoted by bold squares. The column assumed to be a column of $C(F)$ is represented with slanted lines

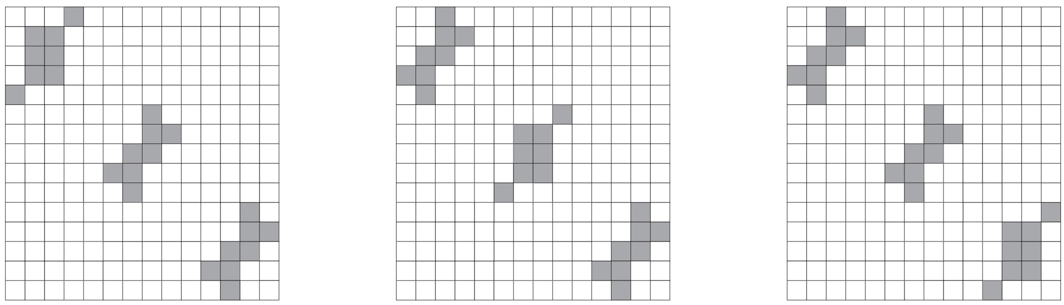


Figure 5.6: Three different decomposable discrete sets with the same projections

From Theorems 5.1 and 5.2 it follows that F is decomposable. Due to Step 3 of Procedure DecomposeNW (and its modified versions for the remaining three directions), the horizontal and vertical projections of F are equal to the given vectors H and V, respectively. Moreover, Step 5 of Algorithm 4-RECDEC guarantees that the diagonal and antidiagonal projections of F are also equal to the vectors D and A, respectively.

Assuming that the l-th ($l = 1, \ldots, k$) component to be reconstructed is a NW component, it takes $O(m + n)$ time to find the (uniquely determined) position which satisfies the necessary conditions of Lemma 5.1.3. We do it simply by scanning the vectors $\widetilde{H}$, $\widetilde{V}$, and $\widetilde{A}$.

In order to test whether this position is the bottom right hand position of the SCDR of the NW component we try to reconstruct this component using Theorem 5.1, which takes $O(f(m, n))$ time.

The same is true when the l-th component is a NE, SE or SW component. In the worst case the component is a SW component, i.e. we try to reconstruct the l-th component at most four times and so the time complexity of reconstructing all the components (Steps 3 and 4 of Algorithm 4-RECDEC) will be of $O(f(m, n))$, which using property (α), is polynomial.

In the worst case we iterate Steps 3 and 4 of Algorithm 4-RECDEC by assuming in turn that the first, then the second, ..., then the n-th column of G is also a column of $C(G)$ (if $n < m$) or by assuming in turn that the first, then the second, ..., then the m-th row of G is also a row of $C(G)$ (if $m \leq n$). This means that Steps 3 and 4 of Algorithm 4-RECDEC must be repeated at most $\min\{m, n\}$ times. So we find that the total execution time of Algorithm 4-RECDEC is of $O(\min\{m, n\} \cdot f(m, n))$ in the worst case.

As we check every possible column (row) of F to see whether it is a column (row) of $C(F)$ it follows that the algorithm reconstructs all sets of $\mathcal{DEC}_\mathcal{C}$ with the given projections. Consequently, if the algorithm returns an $S = \emptyset$ then there is no solution of the $\textsc{Reconstruction}(\mathcal{DEC}_\mathcal{C},\ \mathcal{L}_4)$ task. $\square$

Remark 5.1.2

When a decomposable discrete set is of NW type then the horizontal, vertical and antidiagonal projections will be sufficient for the reconstruction task. In this case the NW or SE versions of Theorem 5.1 provide both the necessary and sufficient conditions for the existence of a NW or SE component, respectively. In a similar way, to reconstruct a decomposable set of NE type it is sufficient to know its horizontal, vertical and diagonal projections. In both cases the solution is uniquely determined owing to Corollary 5.1. That is, knowing the type of the decomposable discrete set the reconstruction can be achieved even with just three projections, uniquely.

We should also add that the class of decomposable discrete sets for which the NW/NE/SE/SW version of Theorem 5.1 provides both the necessary and sufficient conditions for the existence of a NW/NE/SE/SW component, respectively, is a broader class than $\mathcal{S}^{NW} \cup \mathcal{S}^{NE}$. This class of sets was studied in [10].

5.2 The Connection with Q-Convexes

In this section we will demonstrate that decomposability is a more general property than Q-convexity along the horizontal and vertical directions when the discrete set is composed of at least two components. We will prove that every Q-convex set made up of at least two components is decomposable as well. In fact, a somewhat stronger theorem can also be stated.

Theorem 5.4 $\mathcal{Q}' \subset \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$.

Proof
Let $F \in \mathcal{Q}'$ which has components $F_1, \ldots, F_k$ $(k \geq 2)$ such that (4.1) holds.

Since F is non-4-connected it has at least two components. Obviously, the components of F are hv-convex and using Lemma 4.1.2 they are also directed, hence they can be reconstructed uniquely from the horizontal and vertical projections in $O(mn)$ time (see Theorem 3.1), i.e. property (α) is satisfied here.

Furthermore, from Lemma 4.1.1 the configuration of the components of F can be just one of two possibilities. In both cases the sequence $F^{(i)} = \bigcup_{l=1}^{i} F_l$ $(i = 1, \ldots, k)$ is a gluing sequence of F so $F \in \mathcal{DEC}$ using Lemma 5.1.1.

If case (1) of Lemma 4.1.1 holds then we use only SE gluings to build the set F. Otherwise, if case (2) of Lemma 4.1.1 is true then we use only SW gluings to build F. Then, $F \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ (see Fig. 5.7a for instance).

For $\mathcal{Q}' \neq \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$ consider Fig. 5.7b for example. □

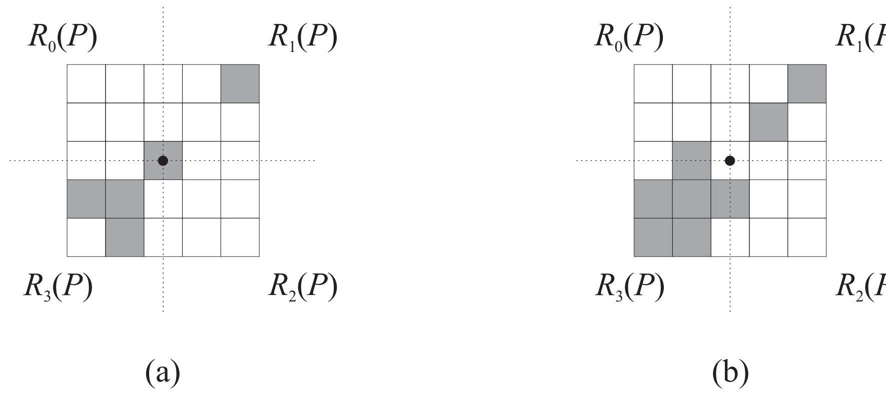


Figure 5.7: A Q-convex (a), and an hv-convex but non-Q-convex discrete set (b) with the four quadrants around the point P (denoted by a black dot)

Theorem 5.4 has an important consequence on the issue of reconstruction complexity in the class $\mathcal{Q}'$ if we use four projections.

Corollary 5.2 *The* $\textsc{Reconstruction}(\mathcal{Q}', \mathcal{L}_4)$ *task can be solved in* $O(mn)$ *time. The solution is uniquely determined in the class* $\mathcal{Q}'$*.*

Proof
Let $F \in \mathcal{Q}'$. From Lemma 4.1.2 we know that the components of F belong to the class $\mathcal{C}$ of hv-convex, directed polyominoes. Moreover, reconstructing an element of $\mathcal{C}$ takes $O(mn)$ time (see Theorem 3.1).

From Theorem 5.4 $F \in \mathcal{S}^{NW} \cup \mathcal{S}^{NE}$. If $F \in \mathcal{S}^{NW} / F \in \mathcal{S}^{NE}$ then each time we call the procedures for the decomposition in Step 3 of Algorithm 4-RECDEC we can decompose a NW/NE component, respectively. Then Algorithm 4-RECDEC reconstructs the solution in the first iteration (i.e., if $l = 0$) and this solution is uniquely determined. Since no more iterations are needed the factor $\min\{m, n\}$ can be eliminated from the formula of the reconstruction complexity given in Theorem 5.3. □

As a direct consequence of Corollary 5.2 and Theorem 4.3 we have

Corollary 5.3 *The* $\textsc{Reconstruction}(\mathcal{S}_8', \mathcal{L}_4)$ *task can be solved in* $O(mn)$ *time and its solution is uniquely determined in the class* $\mathcal{S}_8'$.

5.3 Reconstructing hv-Convex Discrete Sets from Four Projections

The class of hv-convex discrete sets (denoted by $\mathcal{HV}$) is a frequently studied class in discrete tomography. In [64] an algorithm was published which makes it possible to reconstruct sets of this class using two projections. Unfortunately as it turned out later the reconstruction problem in this class is NP-hard [88]. At the same time, for certain subclasses of the $\mathcal{HV}$ class it was found that the reconstruction task using two projections can be accomplished in polynomial time [5; 11; 13; 30; 36]. Somewhat surprisingly, it was also shown that the reconstruction is no longer intractable if absorption in the projections is present (at least for certain absorption coefficients) [69]. Later, the fact that reconstructing an hv-convex discrete set from its horizontal and vertical projections is NP-hard meant that the class of hv-convexes became one of the indicators of newly developed exact or heuristical reconstruction algorithms from the viewpoint of accuracy or running time [15; 38]. In this subsection we investigate the reconstruction of hv-convex discrete sets based on their horizontal, vertical, diagonal, and antidiagonal projections. Although the complexity of this problem is not known we think that there is less hope that a polynomial-time algorithm exists for solving this task (unless P=NP). Still, the results of this section provide an insight into the difficulties involved in the reconstruction task in the very important $\mathcal{HV}$ class.

5.3.1 Reconstruction of hv-Convex Decomposable Discrete Sets

Let us first examine the class $\mathcal{HV} \cap \mathcal{DEC}$, i.e. the class of hv-convex decomposable discrete sets. The following theorem tells us that, with the aid of the decomposition technique described in Section 5.1.3, it is possible to solve the reconstruction problem in this class using four projections in polynomial time.

Theorem 5.5 *The algorithm 4-RECDEC solves the* $\textsc{Reconstruction}(\mathcal{HV} \cap \mathcal{DEC}, \mathcal{L}_4)$ *task in* $O(mn \cdot \min\{m^3, n^3\})$ *time. The algorithm should find all sets of* $\mathcal{HV} \cap \mathcal{DEC}$ *with the given projections.*

Proof

We know that the components of the set F to be reconstructed are uniquely determined by their horizontal and vertical projections in the class of all polyominoes (otherwise F would not satisfy property (α) and would not belong to the class $\mathcal{DEC}$). Moreover, the components are hv-convex as well. In [11] (and also in some previous articles cited there) it was proved that reconstructing a member of the class of hv-convex polyominoes with given horizontal and vertical projections takes theoretically $f(m,n) = O(mn \cdot \min\{m^2, n^2\})$ time. Of course the same is true if we know beforehand that the polyomino is uniquely determined by these projections. Then the truth of this theorem can be seen after applying Theorem 5.3 to the class $\mathcal{C}$ of uniquely determined hv-convex polyominoes. □

Theorem 5.5 states that if the hv-convex discrete set is decomposable then it can be reconstructed in polynomial time. In this case property (α) is guaranteed simply by assuming that the set is decomposable, i.e. finding a component implies that there is no hv-convex polyomino with the same horizontal and vertical projections. An element of this class can be seen in Fig. 5.8.

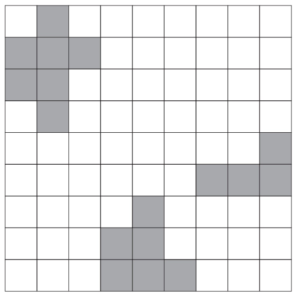


Figure 5.8: An hv-convex decomposable discrete set

Unfortunately, Algorithm 4-RECDEC is not suitable for solving the corresponding consistency task, i.e. for determining the $\textsc{Consistency}(\mathcal{HV} \cap \mathcal{DEC},\ \mathcal{L}_4)$ task. It may happen that the algorithm reconstructs an hv-convex set with the given projections despite the assumption that the components are uniquely determined is wrong and so property (α) is not satisfied. Clearly in this case the reconstructed set is not in $\mathcal{HV} \cap \mathcal{DEC}$. It is caused by the fact that, in Step 3 of the procedures used for decomposition, we can construct a polyomino only by using the general algorithm to reconstruct an hv-convex polyomino (see [11] for the algorithm). This algorithm reconstructs a polyomino from its horizontal and vertical projections but there is no guarantee that the reconstructed polyomino is the only one with these projections. In some cases it does not cause Algorithm 4-RECDEC to fail (see Fig. 5.9). But this drawback also means that Algorithm 4-RECDEC in certain cases can reconstruct hv-convex discrete sets in a somewhat broader class than that of the decomposable ones.

5.3.2 Three Projections: A Negative Result

In the last section it was shown that every hv-convex decomposable discrete set which has the same horizontal, vertical, diagonal, and antidiagonal projections can be recon-

(a) (b)

Figure 5.9: (a) An hv-convex discrete set which is possibly reconstructible by Algorithm 4-RECDEC and (b) an hv-convex discrete set with the same projections but with different components. This shows that the set in (a) does not satisfies property (α)

structed in polynomial time. Plainly this also means that the number of solutions is polynomial as well. However, it remained an open question of whether the use of four projections is necessary to achieve this result. The following theorem addresses this issue.

Theorem 5.6 *For some vectors H, V, and D there can be exponentially many hv-convex decomposable binary matrices with the same horizontal, vertical, and diagonal projections H, V, D, respectively.*

Proof

Consider the following matrices

$$M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 1 \\ 0 & 1 & 0 \end{pmatrix} \quad \text{and} \quad M' = \begin{pmatrix} 0 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} . \tag{5.8}$$

Clearly, M and M' are decomposable and have the same horizontal, vertical, and diagonal projections.

Now for a given integer $k \geq 1$ and for any $S \subseteq \{1, \ldots, k\}$ let the matrix X_k^S be defined as follows

$$X_k^S = \begin{pmatrix} M_1 & & & \\ & M_2 & & \\ & & \ddots & \\ & & & M_k \end{pmatrix} \quad \text{where} \quad M_i = \begin{cases} M & \text{if } i \in S \\ M' & \text{if } i \notin S \end{cases} \qquad i = 1, \ldots, k \ . \tag{5.9}$$

The matrices defined by (5.9) are, of course, hv-convex and decomposable and have the same horizontal, vertical, and diagonal projections. Furthermore, S can be any subset of $\{1, \ldots, k\}$ which yields 2^k matrices with the above-mentioned properties. □

As a consequence of this theorem we get

Corollary 5.4 *If there is an algorithm which reconstructs all the hv-convex decomposable discrete sets using the horizontal, vertical, and diagonal projections H, V, and D, respectively, then there are some vectors H, V, and D for which the complexity of the algorithm is exponential.*

Naturally we get the same results when we replace the diagonal projections with the antidiagonal projections.

5.3.3 A Heuristic for Reconstructing hv-Convex Discrete Sets with Decomposable Configurations

In Section 5.3.1 we mentioned that there is an unexpected feature of Algorithm 4-RECDEC when we apply it to the reconstruction task of hv-convex decomposable discrete sets, because in some cases it reconstructs discrete sets from a somewhat broader class than $\mathcal{HV} \cap \mathcal{DEC}$. The aim of this section is to show how this 'problem' can be turned to our advantage to get a fast and accurate reconstruction heuristic that works for a much broader subclass of hv-convexes than the decomposable subclass. Our strategy will be somewhat different than that in the previous sections. We will describe a reconstruction algorithm that has exponential time complexity in the worst case – unlike the previously presented ones that all run in polynomial time. Despite this, we will also show that this algorithm remains quite fast on average and in almost every case it will find a solution to the given reconstruction task.

In order to introduce the broader class which we want to develop a reconstruction algorithm for, we need to recall the definition of decomposability. A discrete set F consisting of $k \geq 2$ components is *decomposable* if the following properties simultaneously hold:

(α) the components of F are uniquely reconstructible from their horizontal and vertical projections in the class of polyominoes in polynomial time,

(β) the sets of the row and column indices of the components' SCDRs are pairwisely disjoint,

(γ) in the case when $k > 2$ we get F by gluing a single component to a decomposable discrete set made up of $k-1$ components using one of the four gluing operators.

If F satisfies properties (β) and (γ) but not necessarily property (α) then we say that the components of F are in a *decomposable configuration*. Needless to say, the components of an arbitrary decomposable discrete set are in a decomposable configuration but the converse is not true. For example, Fig. 5.10 shows a situation where the components are in a decomposable configuration but property (α) does not hold since the bottom left hand components of both sets have the same horizontal and vertical projections.

Our aim now is to modify the reconstruction Algorithm 4-RECDEC that was presented in Subsection 5.1.3 in such a way that it will be suitable for reconstructing hv-convex discrete sets that have decomposable configurations and which are not necessarily decomposable. Before going further we note there is a restricted form of Algorithm 4-RECDEC which reconstructs an hv-convex decomposable discrete set with the given four projections in polynomial time (the restriction here is that we are interested in finding only one solution).

(a) (b)

Figure 5.10: Two discrete sets that are not decomposable but have decomposable configurations

Algorithm 4-RECHV for reconstructing some sets of $\mathcal{HV}$ from four projections

Input: Four compatible vectors $H \in \mathbb{N}_0^m$, $V \in \mathbb{N}_0^n$, $D \in \mathbb{N}_0^{m+n-1}$, and $A \in \mathbb{N}_0^{m+n-1}$.
Output: An hv-convex discrete set F that has components in a decomposable configuration such that $\mathcal{H}(F) = H$, $\mathcal{V}(F) = V$, $\mathcal{D}(F) = D$, and $\mathcal{A}(F) = A$.

Step 1 $C := \{1, \ldots, n\} \setminus \{j \mid v_j = 0\}$;
Step 2 $F := \emptyset$; $l := \min C$; **restore** H, V, D, and A;
Step 3 repeat
call DecomposeNW';
if (no component) **then** call DecomposeNE';
if (no component) **then** call DecomposeSE';
if (no component) **then** call DecomposeSW';
until (no component);
Step 4 try to reconstruct the last component;
Step 5 if $D = 0$ **and** $A = 0$ **then return** F **else** $C := C \setminus \{l\}$;
Step 6 if $C = \emptyset$ **then return**(no solution found) **else goto** Step 2;

Notice that this algorithm uses slightly different procedures to decompose a component than Algorithm 4-RECDEC. These procedures are called DecomposeNW', DecomposeNE', DecomposeSE', and DecomposeSW'. Below the reader will only find the pseudo-code of DecomposeNW', but the remaining three procedures used by Algorithm 4-RECHV can be described analogously.

Now let us examine the differences between the procedures DecomposeNW and DecomposeNW' in more detail.

First of all, note that the components of an arbitrary hv-convex set are also hv-convex, hence we can restrict ourselves to the class of hv-convex polyominoes when reconstructing a component.

Of course, all four procedures in their original forms just apply the third (diagonal or antidiagonal) projection to test whether the component that was reconstructed from the horizontal and vertical projections also has the proper third projection. There are several algorithms for reconstructing an hv-convex polyomino with given horizontal and vertical projections (see [11] for a comparison). However, all of these algorithms can find only one of the solutions in polynomial time. If the discrete set is decomposable then using property (α) this solution is the true component of the discrete set we wish

Procedure DecomposeNW'

Step 1 find the position (i,j) for which $\widetilde{h}_i = \widetilde{v}_j = \widetilde{a}_{i+j-1} > 0$ and $a_{i+j} = 0$;
if no such position exists **then return**(no component);
Step 2 if $j \geq l$ **then return**(no component);
Step 3' construct all the hv-convex polyominoes with horizontal and vertical projections $(h_1, \ldots, h_i)$, and $(v_1, \ldots, v_j)$, respectively;
if no such polyomino exists **then return**(no component);
Step 3'' select randomly a polyomino P reconstructed in Step 3' such that $\mathcal{A}(P) = (a_1, \ldots, a_{i+j-1})$;
if no such polyomino exists **then return**(no component);
Step 4 $F := F \cup P$;
Step 5 update H, V, D, and A according to the projections of P;
Step 6 return;

to reconstruct (since the components are uniquely determined). Still, if property (α) is not satisfied then it can happen that the reconstructed polyomino is not the true component of the discrete set. For example, during the reconstruction of the discrete set in Fig. 5.10a it may happen that the reconstructed polyomino in the SW corner is the one of Fig. 5.10b which causes the original Algorithm 4-RECDEC to fail. The idea of our heuristic is that we try to eliminate this ambiguity by using directly the third projection in the reconstruction. In order to do this we replace Step 3 of the procedures used for the decomposition in the original decomposition procedures by two steps. In Step 3' we reconstruct all the candidates with the corresponding horizontal and vertical projections and in Step 3'' we randomly choose one of them that has the proper third projection. We should point out that if there is only one candidate in each decomposition step then we get the original form of Algorithm 4-RECDEC.

We conducted experiments to see how an ambiguity of the components (which are hv-convex polyominoes) can affect the performance of Algorithm 4-RECHV. Using the methods given in [58], we generated hv-convex polyominoes with different sizes sampled from uniform random distributions. Each test dataset consisted of 5000 polyominoes all of the same size. The second column of Table 5.1 lists the number of polyominoes in the test datasets that have ambiguous solutions when only two projections are used to reconstruct them. Note that unless the size of the polyomino is small, ambiguity occurs in only 3-6% of the cases (the values in this column are essentially the same as those that were established independently by a similar study in [30]). This means that if the components of an hv-convex discrete set form a decomposable configuration and they are relatively large then it is highly likely that the set will be decomposable, i.e. it will satisfy property (α) too. Evidently, the more components the discrete set has the less likely that all the components will be uniquely determined by the horizontal and vertical projections as ambiguity can occur in any component. Moreover, if the set has small components then ambiguity is more likely to occur and cause the algorithm to fail.

The accuracy of Algorithm 4-RECHV mostly depends on whether the third projec-

Table 5.1: The number of hv-convex polyominoes in the test datasets that are not uniquely determined by two, three, or four projections

Size $n \times n$	H, V	H, V, D	H, V, A	H, V, D, A
4×4	1393	40	52	18
5×5	1442	33	36	16
7×7	967	13	8	2
10×10	586	4	6	1
20×20	312	2	1	1
40×40	210	1	0	0
60×60	162	1	0	0
80×80	148	0	0	0
100×100	171	0	0	0

tion can effectively eliminate ambiguity. The third and fourth columns of Table 5.1 give the number of polyominoes in the test dataset that were not uniquely determined by three projections (actually, due to symmetry the two columns have nearly the same entries). These results tell us that if the size of the polyomino is greater than 3×3 then ambiguity occurs in less than 1% of the cases, and it markedly decreases as the size of the set increases. Moreover, one can readily verify that when the size of the set is smaller then 4×4 then there is no ambiguity in the case of three projections. As before, the more components the discrete set has, the more likely ambiguity will occur (since it can occur in any component).

If we have several candidates with the same three projections for a component then the only thing that affects the remaining part of the reconstruction is the fourth projection of the component. The fifth column of Table 5.1 makes it clear that even in the case when there are several candidates with the same three projections it is still possible that the fourth projection of the chosen set will be the same as the true component's projection and then the algorithm will not fail.

The computational cost of Algorithm 4-RECHV mainly depends on whether Step 3' can be performed quickly. In this step we reconstruct hv-convex polyominoes from their horizontal and vertical projections. Several algorithms have been developed for solving this problem (see [11] for a comparison of them). However, all of them can find just one solution in polynomial time. Since the number of hv-convex polyominoes with the same horizontal and vertical projections can be exponentially large [42], executing Step 3' in some cases may take an exponential time. Fortunately, on average this task can be performed in a few hundredths of a second even on a PC with a processor of speed 533 MHz and 192 MB of RAM (see Table 5.2).

Based on our observations that all the hv-convex polyominoes that have the same two projections can be found quite quickly, and the number of ambiguous cases is very small when three projections are used to reconstruct them, we expected our novel heuristic to reconstruct hv-convex discrete sets which have decomposable configurations (i.e. when property (α) does not always hold) quickly and in most cases accurately. To test this we carried out a series of experiments. We generated 5 datasets, each

Table 5.2: Average execution times for reconstructing all hv-convex polyominoes with the same horizontal and vertical projections, which depends on the size of hv-convex polyomino. The values were taken from [11]

Size $n \times n$	Avg. time (s)
20×20	0.006
40×40	0.008
60×60	0.018
80×80	0.031
100×100	0.048

of them containing 1000 hv-convex sets with decomposable configurations that had k components of size $n \times n$ for some fixed k and n. The generation method was the following. As before, using the methods given in [58], we generated a sequence of k hv-convex polyominoes of size $n \times n$ sampled from a uniform random distribution. Afterwards, we generated a random sequence of the NW, NE, SE, and SW elements. If the discrete set to be generated had k components then the length of the sequence was $k-1$, and it showed how the k components have to be glued to each other in the right order. For the 5 test datasets the k and n parameters had the following values:

- Test 1: $k = 10$, $n = 5$;
- Test 2: $k = 20$, $n = 5$;
- Test 3: $k = 30$, $n = 5$;
- Test 4: $k = 10$, $n = 10$;
- Test 5: $k = 20$, $n = 10$.

Thus, for instance, Test 1 contained 1000 discrete sets of size 50×50, and each of them had 10 components of size 5×5. The reconstruction heuristic was implemented in C++ and the run was performed on an Intel Pentium IV 3.2 GHz with 1GB RAM under Debian GNU/Linux 3.1.

Table 5.3 shows the average running times, and the number of correct and incorrect solutions for the 5 test datasets. In the last column of this table we also list the number of cases when the algorithm did not find a solution. From the entries of Table 5.3 we can see that the number of incorrect solutions (and also the number of cases when no solution is found) goes up as the number of components goes up. But we should mention here that in the first three tests the resulting inaccurate reconstruction differed from the original set by only one component, and in only 8 positions. More precisely, the original and the reconstructed components always formed a pair like

Table 5.3: Accuracy and average running time of Algorithm 4-RECHV on the test datasets

Test	#correct sol.	#incorrect sol.	#no sol.	time (s)
Test 1	939	14	47	0.600
Test 2	891	27	82	0.847
Test 3	851	41	108	2.322
Test 4	998	0	2	0.660
Test 5	994	0	6	5.676

$$M = \begin{pmatrix} 0 & 0 & X & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ X & 1 & 1 & 1 & X \\ 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & X & 0 & 0 \end{pmatrix} \quad \text{and} \quad M' = \begin{pmatrix} 0 & 1 & X & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \\ X & 1 & 1 & 1 & X \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 0 & X & 1 & 0 \end{pmatrix}, \tag{5.10}$$

where the positions marked by X took arbitrary values and were the same in M and M'.

Moreover, according to Table 5.1 if the set has components of size 10×10 (or greater) then the algorithm can accurately reconstruct the set in almost every case. In fact, in Tests 4 and 5 there were no inaccurate reconstructions (although there is a small chance that it will occur).

Quite clearly, the larger the set is, the more time will be needed to reconstruct it. But even for the biggest sets of Test 5 the average running time of the algorithm was quite fast. In summary, we can say that Algorithm 4-RECHV has a good performance in terms of both quality and running time.

5.4 Conclusions and Discussion

In this chapter we studied the reconstruction problem using four projections. We introduced the class of decomposable discrete sets and presented a polynomial-time algorithm for reconstructing elements of this class. This algorithm is able to find all the solutions for a given reconstruction problem. The complexity of reconstructing decomposable discrete sets from two or three projections remains an open question, however.

Analysing the connection of the class of decomposable discrete sets with other frequently studied classes in discrete tomography we proved that every Q-convex set which consists of at least two components is decomposable. As a consequence, we found that the reconstruction from four projections in the class $\mathcal{Q}'$ can be solved uniquely in $O(mn)$ time. Since $\mathcal{S}'_8 \subset \mathcal{Q}'$ the reconstruction of hv-convex 8- but not 4-connected sets from four projections can also be solved uniquely in $O(mn)$ time.

We used the diagonal and antidiagonal projections to decompose the components of a discrete set. Notice that if the set belongs to the class $\mathcal{S}^{NW}/\mathcal{S}^{NE}$ then the use of the antidiagonal/diagonal projection, respectively, is sufficient to decompose the set into components, i.e. if the type of the set is known in advance then three projections are sufficient to reconstruct the object. And we also showed that in both classes the reconstruction is faster than in the general class of decomposables. Actually, without going into technical details we should mention here that instead of using the diagonal/antidiagonal projection one can use any projection in a direction that has a positive/negative gradient, respectively.

We also investigated the feasibility of applying the decomposition technique to reconstruct hv-convex discrete sets based on four projections and found that the reconstruction of hv-convex decomposable discrete sets can be performed in $O(mn \cdot \min\{m^3, n^3\})$ time. Since the decomposition technique can find all solutions of a given reconstruction problem this also means that the number of solutions of this reconstruction task is polynomial. With a counterexample we also showed that three projections are not sufficient to achieve this result, i.e. there can be exponentially many hv-convex decomposable discrete sets with the same three projections.

We found that the decomposition algorithm has a problem when we apply it to reconstruct hv-convex decomposable discrete sets. In some cases it can reconstruct discrete sets in such a way that the components are not necessarily uniquely determined – which is one of the conditions a decomposable discrete set has to satisfy. By exploiting this feature we developed a reconstruction heuristic for the class of hv-convex discrete sets that have so-called decomposable configurations. We conducted experiments to learn about the quality of our heuristic. Although our heuristic can solve the reconstruction in the above class in the worst case in exponential time, we showed that on average the algorithm is pretty fast. We also found that the algorithm is quite accurate as it reconstructed, in most of the test cases, the original discrete set or a closely similar one.

The concept of decomposability was introduced in [9], and the decomposition technique was described in [6]. The applications of this to several decomposable classes can be found in [3; 6; 10]. The negative result concerning the ambiguity problem for three projections and the reconstruction heuristic for the class of discrete sets which have decomposable configurations came from results presented in [7].

Chapter 6

Random Generation of hv-Convex Discrete Sets

When analysing reconstruction algorithms from the viewpoint of running time, the worst case time complexity is just one feature where the effectiveness of the algorithm can be described. In fact, when a reconstruction algorithm is applied in practice the average running time of the algorithm is also an important characteristic of its efficiency. Besides the worst case scenario, the average performance of an algorithm is also important even when other parameters of its efficiency are tested such as accuracy and the memory requirements. Moreover, it is also important to describe which constraining assumptions (in our case, which kind of discrete sets) the algorithm performs under for the reconstruction to be fast and accurate, and what are the (perhaps less likely) situations where the algorithm attains its theoretically worst performance.

For example, for the reconstruction of hv-convex polyominoes or hv-convex 8-connected sets from the horizontal and vertical projections, two very different algorithms have been developed. One of them (let us call it Algorithm A) approximates the solution iteratively by a nondecreasing sequence of so-called kernel sets and by a nonincreasing sequence of so-called shell sets (see [13; 14], and [30] for the improved version of the algorithm which can reconstruct sets of $\mathcal{S}'_8$ too). Algorithm A and its modified version have a worst case time complexity of $O(mn \cdot \log mn \cdot \min\{m^2, n^2\})$. The other approach (let us call it Algorithm B) is based on an observation that the reconstruction task can be transformed into a 2SAT task that is solvable in polynomial time (as was proved in [1]). This algorithm was presented in [36], while the improved form which can also reconstruct 8-connected hv-convex sets was published in [66]. Both forms of Algorithm B have a worst case time complexity of $O(mn \cdot \min\{m^2, n^2\})$. However, in [11] Algorithms A and B were compared from the viewpoint of the average execution time and it was found that Algorithm A reconstructs the solution faster than Algorithm B in almost every case studied. The observations concerning the average execution times of the two reconstruction approaches led to the design of a third reconstruction algorithm that has the same worst case time complexity as Algorithm B but remains as fast as Algorithm A in the average case [11]. This hybrid algorithm is frequently used today.

To be able to analyse the average performance of a reconstruction algorithm that was developed for a given class $\mathcal{G} \subseteq \mathcal{F}$ of discrete sets, one has to generate elements of the class $\mathcal{G}$ of the same size using uniform random distributions. For the class $\mathcal{F}$ this task is not all that complicated. To generate a discrete set $F \in \mathcal{F}$ of size $m \times n$ from a uniform random distribution one simply has to put elements of the set $\{0, 1\}$ from a uniform random distribution into the mn positions of the representative matrix $\hat{F}$ of F.

The reader may have realized by now that for some classes of discrete sets it is far from trivial to generate elements of that class using a uniform random distribution. For instance, the comparison that was described in the previous paragraph was based on a sophisticated method of [58] where it is possible to generate hv-convex polyominoes of given size using a uniform random distribution. Another important class of discrete sets is the class of Q-convexes that was examined in some detail in Chapter 4. Inspired by [58], in [25] a method was described that can generate elements of this broader class from uniform random distributions as well, by which important statistics from the Q-convex sets could be collected [27; 29].

The more general class of hv-convex discrete sets is one of the most important classes in discrete tomography. As we have repeatedly mentioned in this dissertation, the reconstruction from two projections in this class is NP-hard [88]. Still, several methods can solve this problem by applying some heuristic [64], metaheuristic [15] or optimization [38] technique. And it was also shown that the reconstruction task in this class is no longer intractable if absorption in the projections is present (for certain absorption coefficients) [69]. Based on these findings, in the last few years the class of hv-convex discrete sets has become one of the statistical indicators of newly developed exact or heuristical reconstruction algorithms which characterises the effectiveness of a given technique. This means that the performance of the newly developed techniques are often tested on this class. Unhappily researchers had to acknowledge the fact that no method was known for generating hv-convex sets of a given size using uniform random distributions, and hence no exact comparison of the techniques was possible. In actual fact, the efficiency of these techniques was usually tested on only 5-10 elements (or just one!) of the class of hv-convexes that were chosen ad hoc. Obviously, this does not say too much about the average performance of the reconstructed method on that class.

In this chapter we will present an algorithm for generating hv-convex discrete sets of a moderate size from uniform random distributions and study the properties of uniformly generated sets which can affect the performance of certain reconstruction algorithms. We also will show how this generation technique can be adapted to some other classes of discrete sets as well.

6.1 Generation of Special hv-Convex Discrete Sets

The first result concerning the uniform random generation of hv-convex discrete sets was given by Delest and Viennot [43], who proved that the number P_n of hv-convex polyominoes with a perimeter value of $2n + 8$ is

$$P_n = (2n + 11)4^n - 4(2n + 1)\binom{2n}{n}. \tag{6.1}$$

Later, based on the above result in [51] it was shown that the number $P_{m+1,n+1}$ of hv-convex polyominoes with a minimal bounding rectangle of size $(m+1) \times (n+1)$ is

$$P_{m+1,n+1} = \frac{m+n+mn}{m+n}\binom{2m+2n}{2m} - \frac{2mn}{m+n}\binom{m+n}{m}^2. \tag{6.2}$$

We will now consider a special class of hv-convex discrete sets (denoted by $\mathcal{S}'$), namely where the minimal bounding rectangles of the components are always connected to each other with their bottom right hand and upper left hand corners and there are no empty rows and columns in the discrete sets (see Fig. 6.1).

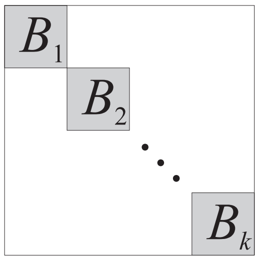


Figure 6.1: The relative position of the minimal bounding rectangles of the components $B_1, \ldots, B_k$ in the $\mathcal{S}'$ class

It is not hard to see that a discrete set $F \in \mathcal{S}'$ of size $m \times n$ is either a polyomino (i.e. it has just one component) or it contains a polyomino of size $k \times l$ (where $k < m$ and $l < n$) as a subset in the upper left hand corner and the remaining part of F is a discrete set of size $(m-k) \times (n-l)$ which also belongs to the $\mathcal{S}'$ class (see Fig. 6.1 again). Denoting the number of discrete sets of $\mathcal{S}'$ with size $m \times n$ by $S'_{m,n}$, this observation can be concisely expressed by a simple recursive formula. It is

$$S'_{m,n} = P_{m,n} + \sum_{k<m,\ l<n} P_{k,l} \cdot S'_{m-k,n-l}\ . \tag{6.3}$$

Using Equation (6.2) and the initial values $S'_{1,j} = P_{1,j} = 1$ $(j = 1, \ldots, n)$ and $S'_{i,1} = P_{i,1} = 1$ $(i = 1, \ldots, m)$ $S'_{m,n}$ can be calculated via a dynamic programming approach in $O(m^2n^2)$ time with $O(mn)$ memory requirements. From this, we now can describe the algorithm for generating hv-convex discrete sets of $\mathcal{S}'$ of a given size using a uniform random distribution.

This algorithm is called GENHV-S' and it works as follows.

First, in Step 1, it calculates $S'_{m,n} = P_{m,n} + P_{1,1} \cdot S'_{m-1,n-1} + P_{1,2} \cdot S'_{m-1,n-2} + P_{2,1} \cdot S'_{m-2,n-1} + \cdots + P_{m-1,n-1} \cdot S'_{1,1}$.

Choosing a number r using a uniform random distribution in the interval $[1, S'_{m,n}]$

Algorithm GENHV-S' for generating hv-convex discrete sets of $\mathcal{S}'$ from a uniform random distribution

Input: The integers m and n.
Output: The hv-convex discrete set $F \in \mathcal{S}'$ of size $m \times n$.

Step 1 calculate $S'_{m,n}$;
Step 2 generate a number $r \in [1, S'_{m,n}]$ from a uniform random distribution;
Step 3 **if** ($r > P_{m,n}$)
{ $r = r - P_{m,n}$;
for $k = 1$ **to** $m - 1$
for $l = 1$ **to** $n - 1$
if ($r > P_{k,l} \cdot S'_{m-k,n-l}$) $r = r - P_{k,l} \cdot S'_{m-k,n-l}$;
else call Algorithm GENHV-S' with parameters $m - k$ and $n - l$; }
Step 4 generate the components using a uniform random distribution;

(Step 2), one must determine which one of the following intervals r falls into;

$$\begin{aligned}
I_0 &= [1, P_{m,n}] \ , \\
I_{1,1} &= [P_{m,n} + 1, P_{m,n} + P_{1,1} \cdot S'_{m-1,n-1}] \ , \\
I_{1,2} &= [P_{m,n} + P_{1,1} \cdot S'_{m-1,n-1} + 1, P_{m,n} + P_{1,1} \cdot S'_{m-1,n-1} + P_{1,2} \cdot S'_{m-1,n-2}] \ , \\
&\vdots \\
I_{m-1,n-1} &= [S'_{m,n} - P_{m-1,n-1} \cdot S'_{1,1} + 1, S'_{m,n}] \ .
\end{aligned} \tag{6.4}$$

If $r \in I_0$ then the whole discrete set to be generated will consist of just one component, i.e. it will be an hv-convex polyomino. If $r \in I_{1,1}$ then the upper left hand component of the set to be generated will have a size 1×1. Similarly, if $r \in I_{1,2}$ then the upper left hand component will have a size 1×2. And so on. Finally, if $r \in I_{m-1,n-1}$ then the upper left hand component will have a size $(m-1) \times (n-1)$. The intervals of (6.4) give a partition of the interval $[1, S'_{m,n}]$, hence the size of the upper left hand component of the set to be generated can be found and this method can be repeated for the remaining set as well (Step 3).

Next, we only have to generate the components themselves from uniform random distributions by knowing the sizes of their bounding rectangles – which is possible with the algorithm given in [58] (Step 4). From the above points mentioned here it follows that Algorithm GENHV-S' generates sets of $\mathcal{S}'$ of size $m \times n$ from a uniform random distribution.

The above method can also be extended to special hv-convex discrete sets which may have empty rows or/and columns (but still have the same component configuration as shown in Fig. 6.1). This class will be denoted by $\mathcal{S}$. Naturally, $\mathcal{S}' \subset \mathcal{S}$. Then a discrete set $F \in \mathcal{S}$ of size $m \times n$ is either a polyomino or it contains a polyomino of size $k \times l$ (where $k < m$ and $l < n$) as a subset in the upper left hand corner, and the remaining part of F is a discrete set of size $(m-k) \times (n-l)$ which may have some empty rows or/and columns in the upper left hand corner and the remaining part belongs to the $\mathcal{S}$ class. Denoting the number of discrete sets of $\mathcal{S}$ with size $m \times n$ by

$S_{m,n}$, we get a formula similar to Equation (6.3). That is,

$$S_{m,n} = P_{m,n} + \sum_{k<m,\ l<n} P_{k,l} \cdot \Big(\sum_{i \le m-k,\ j \le n-l} S_{i,j} \Big) \ . \tag{6.5}$$

As before, using (6.2) and the initial values $S_{1,j} = P_{1,j} = 1$ ($j = 1, \ldots, n$) and $S_{i,1} = P_{i,1} = 1$ ($i = 1, \ldots, m$), (6.5) can be evaluated via a dynamic programming approach in $O(m^3n^3)$ time[1] with $O(mn)$ memory requirements. Next, an algorithm similar to Algorithm GENHV-S' can be supplied to generate hv-convex discrete sets of given size from the $\mathcal{S}$ class based on a uniform random distribution (let us call it Algorithm GENHV-S). But we should add that if we allow the generated set to have empty rows or/and columns then the evaluation of the corresponding recursive formula will take more time than in the case when the set does not have empty or/and columns. That is, the generation task with this method in the broader $\mathcal{S}$ class will take longer.

6.2 Generation of hv-Convex Discrete Sets

The method of Section 6.1 provides a useful tool for reconstructing some special hv-convex discrete sets from uniform random distributions. It needs, though, some re-thinking to adapt Algorithm GENHV-S' (and Algorithm GENHV-S) to the whole class of hv-convexes. We will use the following lemma whose content is quite trivial.

Lemma 6.2.1 *Let F be an arbitrary hv-convex set that has $k \ge 2$ components. Then there is a uniquely determined set $F' \in \mathcal{S}$ and a uniquely determined permutation π of order k such that F and F' have the same components and if the SCDR of the l-th component of F' is $\{i_l, \ldots, i'_l\} \times \{j_l, \ldots, j'_l\}$ then the SCDR of the l-th component of F is $\{i_l, \ldots, i'_l\} \times \{j_{\pi(l)}, \ldots, j'_{\pi(l)}\}$.*

Figure 6.2 shows an example of this connection between sets of $\mathcal{S}$ and arbitrary hv-convex discrete sets. Of course if the set F in Lemma 6.2.1 does not have empty rows and columns then the corresponding set F' also belongs to the $\mathcal{S}'$ class.

The reader may think that it is now straightforward to generate sets of $\mathcal{HV}$ from a uniform random distribution by doing the following. One simply generates a set of $\mathcal{S}$ from a uniform random distribution and then one gets the final hv-convex set by permuting the column sets of the components with a randomly chosen permutation. Unfortunately, this method does not generate the sets from a uniform random distribution. Let us see why this is so in an illustrative example.

Example 6.2.1
Suppose we want to generate a discrete set of $\mathcal{S}'$ with size 2×3 from a uniform random distribution. Then,

$$S'_{2,3} = P_{2,3} + P_{1,2} \cdot S'_{1,1} + P_{1,1} \cdot S'_{1,2} = 13 + 1 \cdot 1 + 1 \cdot 1 = 15 \ . \tag{6.6}$$

Fig. 6.3 shows the 15 possible discrete sets of $\mathcal{S}'$ with size 2×3.

[1]With a more sophisticated approach this computational complexity can be reduced to $O(m^2n^2)$.

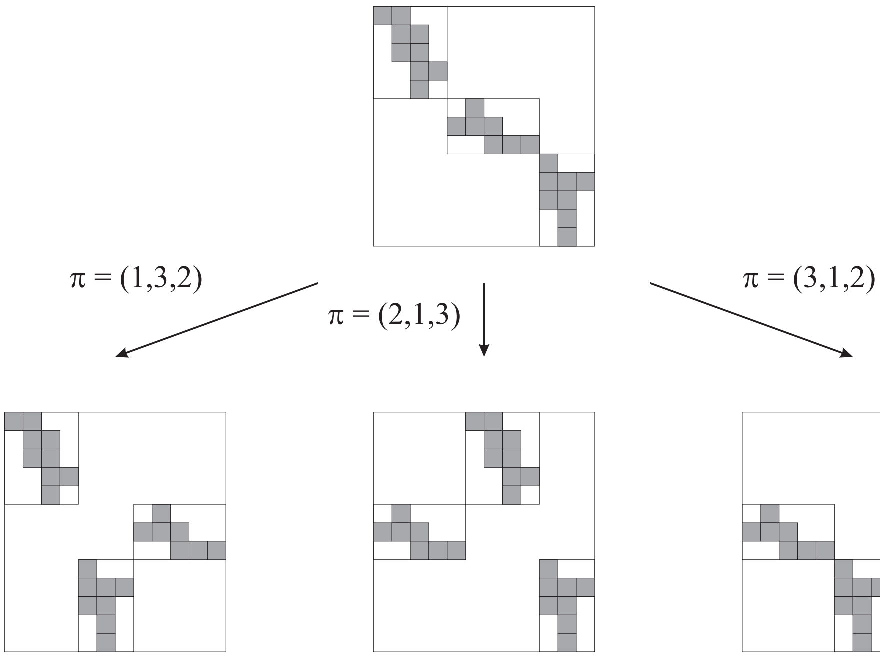


Figure 6.2: An example which shows the connection between elements of the $\mathcal{HV}$ and $\mathcal{S}$ classes

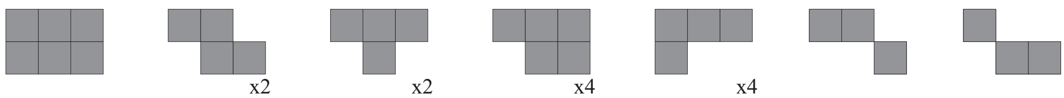


Figure 6.3: All the hv-convex discrete sets of size 2×3. The numbers tell us that there are other solutions which can be obtained by mirroring or/and rotating the given set

That is, the probability that the generated set will be a polyomino is 13/15, while the probability that the result will be the last or second last set of Fig. 6.3 is 1/15 in both cases. There are two other hv-convex sets of size 2×3 that we can get by mirroring the last two sets of Fig. 6.3 horizontally (or equivalently, by permuting the column sets of the components of these sets according to the permutation $\pi = (2, 1)$). Thus there exist 17 hv-convex discrete sets of size 2×3 that do not contain empty rows and columns. Following the naive generation method described above, the probability that the generated set will be the last set of Fig. 6.3, say, is $\frac{1}{15} \cdot \frac{1}{2}$ (since either we keep this set in its original form or we apply the permutation $\pi = (2, 1)$ on the column sets of it). But the true probability for a uniform random generation must be 1/17 – which shows that the naive method does not work well.

We need to make some small changes in Algorithm GENHV-S' to achieve our goal. In fact, what we have to do is to have proper weights of the discrete sets of $\mathcal{S}'$ that can represent several hv-convex sets. These sets of $\mathcal{S}'$ are exactly the ones which consist of several components. Moreover, with the help of Lemma 6.2.1 it is also clear that each such set represents $k!$ sets if it has k components.

Now, the recursive formulas used in the generation will change in the following way. Let $HV'^{(t)}_{m,n}$ denote the number of arbitrary hv-convex discrete sets with nonempty rows and columns that have minimal bounding rectangles of size $m \times n$ and exactly

t components. Then $HV'^{(t)}_{i,j} = 0$ if $i < t$ or $j < t$, and $HV'^{(1)}_{i,j} = P_{i,j}$ for each $i = 1, \ldots, m$ and $j = 1, \ldots, n$. Finally, based on similar observations as the formula of (6.3) for $t > 1$ the following recursion holds

$$HV'^{(t)}_{m,n} = \sum_{k<m,\ l<n} P_{k,l} \cdot HV'^{(t-1)}_{m-k,n-l} \cdot t \ . \tag{6.7}$$

Here, the factor t decreases by one at each step as we go deeper and deeper in the recursion. Eventually this will yield a factor of $t!$ for a set that consists of t components – which is exactly the weight we need.

With a simple calculation we find that the total number $HV'_{m,n}$ of arbitrary hv-convex discrete sets of size $m \times n$ with nonempty rows and columns is

$$HV'_{m,n} = \sum_{t=1}^{\min\{m,n\}} HV'^{(t)}_{m,n} \ . \tag{6.8}$$

With this modification we will get the right probabilities like that for the problem presented in Example 6.2.1, that is

$$\begin{aligned} HV'_{2,3} &= HV'^{(1)}_{2,3} + HV'^{(2)}_{2,3} = P_{2,3} + P_{1,2} \cdot HV'^{(1)}_{1,1} \cdot 2 + P_{1,1} \cdot HV'^{(1)}_{1,2} \cdot 2 = \\ &= 13 + 1 \cdot 1 \cdot 2 + 1 \cdot 1 \cdot 2 = 17 \ . \end{aligned} \tag{6.9}$$

Now, in an analogous way to Algorithm GENHV-S', we shall describe the generation method called Algorithm GENHV' for the whole class of hv-convexes with nonempty rows and columns, also taking into account what Lemma 6.2.1 says – that the components need to be permuted by a randomly chosen permutation.

We should also add that with the help of equations (6.5) and (6.7) we can develop Algorithm GENHV in a straightforward way for generating arbitrary hv-convex sets – perhaps with empty rows or/and columns – from a uniform random distribution. After, the total number of hv-convex discrete sets can be calculated by a formula similar to (6.8). Unfortunately, the recursive formula calculated by Algorithm GENHV'/GENHV is more complex than the formula used in Algorithm GENHV-S'/GENHV-S, respectively so the more general generation methods have a bigger time complexity. Generality, it seems, comes at a price.

6.3 Possible Generalisations

The main advantage of the methods of Section 6.1 and Section 6.2 is that they can be readily extended to several classes of discrete sets which have disjoint components (even when the components are not hv-convex) if the components themselves can be generated from a uniform random distribution – knowing their bounding rectangles – and if it is possible to enumerate them.

As a simple example let us assume that the discrete set to be generated does not have empty rows and columns and all the components are discrete rectangles. Here

Algorithm GENHV' for generating hv-convex discrete sets with nonempty rows and columns using a uniform random distribution

Input: The integers m and n.
Output: The hv-convex discrete set F of size $m \times n$.

Step 1 calculate $HV'_{m,n}$;
Step 2 generate a number $r \in [1, HV'_{m,n}]$ using a uniform random distribution;
Step 3 **if** ($r \in [1, HV'^{(1)}_{m,n}]$) **goto** Step 7;
  else let t be the integer for which $r \in [HV'^{(t-1)}_{m,n} + 1, HV'^{(t)}_{m,n}]$;
Step 4 generate a number $r \in [1, HV'^{(t)}_{m,n}]$;
Step 5 **if** ($r > HV'^{(1)}_{m,n}$)
  { $r = r - HV'^{(1)}_{m,n}$;
   for $k = 1$ **to** $m - 1$
    for $l = 1$ **to** $n - 1$
     if ($r > P_{k,l} \cdot HV'^{(t-1)}_{m-k,n-l} \cdot t$) $r = r - P_{k,l} \cdot HV'^{(t-1)}_{m-k,n-l} \cdot t$;
     else goto Step 3 with parameters $m - k$, $n - l$, and $t - 1$; }
Step 6 generate a permutation π of order t from a uniform random distribution and permute the column sets of the components according to π;
Step 7 generate the components from a uniform random distribution;

we will not assume that the components have a special configuration but just suppose that the sets of their row and column indices are disjoint. Let $R^{(t)}_{m,n}$ denote the number of such discrete sets of size $m \times n$ having exactly t components. For each i and j there exists exactly one discrete rectangle of size $i \times j$, i.e. $R^{(1)}_{i,j} = 1$ for each $i = 1, \dots, m$ and $j = 1, \dots, n$. Moreover, $R^{(t)}_{i,j} = 0$ if $i < t$ or $j < t$, and for $t > 1$ the following recursive formula holds

$$R^{(t)}_{m,n} = \sum_{i<m,\ j<n} R^{(t-1)}_{i,j} \cdot t \ , \tag{6.10}$$

where the factor t represents the proper weights for describing the possible permutations of the column set of the t components. After, the total number of such discrete sets of size $m \times n$ is $\sum_{t=1}^{\min\{m,n\}} R^{(t)}_{m,n}$ and the generation algorithm can be adapted in a straightforward way.

Another way of generalising the method is by not generating the elements of a certain class using uniform random distributions, but by generating some sets with bigger probabilities than others. By manipulating the weight factor t in the recursive formula used in the generation algorithm we can assign bigger weights to some sets we prefer and smaller ones to the sets that we want to occur less often in the generation process. For example, we can select generated sets that have more components and not ones that have fewer components. Actually, $w(t) = t$ is just a special weight function that yields a uniform random distribution.

6.4 Statistics on hv-Convex Discrete Sets

The methods of Section 6.1 and Section 6.2 allow us to examine some important properties of hv-convex discrete sets. In order to get some statistical information about them we generated test datasets with Algorithms GENHV-S' and GENHV' and their modified versions GENHV' and GENHV for hv-convex discrete sets with possible empty rows or/and columns. Each set of test data consisted of 1000 hv-convex discrete sets of the same size generated from the given class using a uniform random distribution. The algorithms were implemented in C++ and the long integer functions of library NTL-5.4 [86] were employed. The test was run on a PC with Intel Pentium 4 processor of 3.2 GHz and 1 GB RAM under Debian GNU/Linux 3.1 with Kernel 2.6.17.13.

6.4.1 The Number of hv-Convex Discrete Sets

Our first simple investigation did not use the test datasets but focused on the number of hv-convex discrete sets in the classes studied. Introducing the notations $\mathcal{P}$ for the class of hv-convex polyominoes and $\mathcal{HV}'$ for the class of hv-convex discrete sets with non-empty rows and columns we can proceed further. Next, recall that the class of hv-convex discrete sets is denoted by $\mathcal{HV}$.

Table 6.1 shows the number of elements in the classes $\mathcal{P}$, $\mathcal{S}'$, $\mathcal{S}$, $\mathcal{HV}'$, and $\mathcal{HV}$ with bounding rectangles of semi-perimeter n for the first 15 values of n – represented by $P(n)$, $S'(n)$, $S(n)$, $HV'(n)$, and $HV(n)$, respectively (the first column can also be calculated via formula (6.1) and it enumerates the first 15 elements of Sequence A005436 in [87]).

Table 6.1: The values of $P(n)$, $S'(n)$, $S(n)$, $HV'(n)$, and $HV(n)$

n	$P(n)$	$S'(n)$	$S(n)$	$HV'(n)$	$HV(n)$
2	1	1	1	1	1
3	2	2	2	2	2
4	7	8	8	9	9
5	28	32	34	36	40
6	120	139	150	162	184
7	528	618	674	732	860
8	2344	2779	3056	3368	4058
9	10416	12528	13898	15520	19240
10	46160	56404	63178	71618	91440
11	203680	253152	286570	329988	435136
12	894312	1131849	1296008	1518090	2072672
13	3907056	5040412	5842442	6971112	9883264
14	16986352	22359981	26255254	3196392	47193776
15	73512288	98837102	117642282	146390016	225779728

For $n = 5$ the corresponding binary pictures of the classes of hv-convex polyominoes ($\mathcal{S}'$ and $\mathcal{S}$) are shown in Fig. 6.4. The remaining elements of the $\mathcal{HV}$ class with semi-perimeter value $n = 5$ can be obtained by permuting the column sets of the discrete sets of the bottom row of Fig. 6.4.

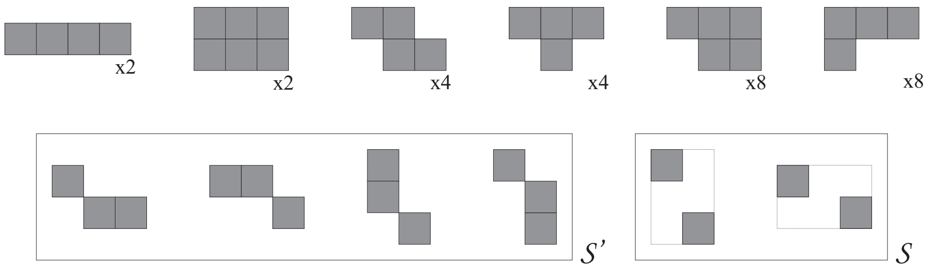


Figure 6.4: Some hv-convex binary pictures with a perimeter value of 10. The numbers tell us that there are other solutions that can be obtained by mirroring or/and rotating the given polyomino

From the definitions of the classes studied, the following inclusions are trivial

$$\mathcal{P} \subset \mathcal{S}' \subset \mathcal{S} \subset \mathcal{HV} \quad \text{and} \quad \mathcal{P} \subset \mathcal{S}' \subset \mathcal{HV}' \subset \mathcal{HV} \,. \tag{6.11}$$

Knowing the relations of (6.11) and with the aid of the statistics presented in Table 6.1, we can describe the relative cardinality of the classes examined. With this information we can, for example, address questions concerning the relative occurrence of certain hv-convex discrete sets and calculate the probability that an hv-convex discrete set chosen from a uniform random distribution has some special properties which can facilitate the reconstruction task.

Example 6.4.1

Using the entries of Table 6.1 we can calculate the probability that an hv-convex discrete set with semi-perimeter value of 6 chosen from a uniform random distribution is an hv-convex polyomino (i.e. it consists of one component), which turns out to be $120/184 \approx 0.65$. If we increase the semi-perimeter value to 10, say, then this probability decreases to $46160/91440 \approx 0.50$. Such information is especially useful in the reconstruction task as hv-convex polyominoes can be reconstructed from their horizontal and vertical projections in polynomial time. In contrast, if the hv-convex set has at least two components then the reconstruction is NP-hard (see the introduction here). Hence with this method we can calculate the probability that the reconstruction of the randomly chosen hv-convex set can be performed using a polynomial-time algorithm to reconstruct an hv-convex polyomino.

Example 6.4.2

From Table 6.1 we can also calculate whether an hv-convex discrete set F that was chosen using a uniform random distribution with a given semi-perimeter value belongs to the $\mathcal{S}$ class and it has at least two components. For instance, the probability that an hv-convex discrete set with semi-perimeter value of 6 chosen from a uniform random distribution belongs to the $\mathcal{S}$ class but is not a polyomino is $(150 - 120)/184 \approx 0.16$. As we mentioned in Remark 5.1.2 and touched on in Section 5.3.3, such a set can be reconstructed (with a heuristic algorithm) quite fast and accurately from three projections. Using Table 6.1, say, we can address the important issue of whether such situations occur frequently.

6.4.2 The Number of Components

The second experiment examines the number of components of an hv-convex set. This piece of information is also very useful when reconstructing images like these.

For example, as was discussed in the introduction here, if the hv-convex set consists of a single component then the reconstruction from two projections can be solved in polynomial time, otherwise it is NP-hard. Furthermore, the number of components of an hv-convex set also affects the accuracy of the reconstruction heuristic that was presented in Section 5.3.3. Namely, the more components the hv-convex discrete set has, the more likely that ambiguity will occur in the reconstruction.

As regards the classes $\mathcal{S}'$ and $\mathcal{S}$, Table 6.2 gives the number of components – which depends on the size – in the 1000 generated discrete sets of each dataset when the discrete set does not have empty rows and columns (top half of the table) and when empty rows and columns are also permitted (bottom half of the table). Note that the sum of the elements in the last two rows is smaller than 1000. Owing to lack of space we did not include 2 sets of size 80×80 and 22 sets of size 100×100 that have over 15 components.

This numerical investigation makes it clear that when generating a special hv-convex set using a uniform random distribution there is a high probability that the set will consist of a single component if the size of the set is small (i.e. less than or equal to 20×20), but there is almost no chance of applying the well-known polynomial-time algorithms which reconstruct hv-convex polyominoes for sets of a greater size. Although this fact could also have been inferred from Table 6.1, Table 6.2 is more informative in the sense that the expected number of components of the special hv-convex discrete set that was generated from a uniform random distribution can also be estimated in advance by knowing just the size of the discrete set. This could be quite useful in the reconstruction task.

Let $E_{\mathcal{C}}(n)$ and $D^2_{\mathcal{C}}(n)$ respectively denote the expected number and the variance of the components of a discrete set of size $n \times n$ generated from a given $\mathcal{C}$ class using a uniform random distribution. If $n \leq 100$ then the estimated values of $E_{\mathcal{S}'}(n)$, $E_{\mathcal{S}}(n)$, $D^2_{\mathcal{S}'}(n)$, and $D^2_{\mathcal{S}}(n)$ can be calculated directly from Table 6.2. For larger sets a good estimation can be given using the following simple formulas:

$$E_{\mathcal{S}'}(n) \approx 0.075n \quad \text{and} \quad D^2_{\mathcal{S}'}(n) \approx 0.04n \ , \tag{6.12}$$

in the case of sets with non-empty rows and columns and

$$E_{\mathcal{S}}(n) \approx 0.100n \quad \text{and} \quad D^2_{\mathcal{S}}(n) \approx 0.06n \tag{6.13}$$

in the case of possibly empty rows and columns.

In addition, in both classes $\mathcal{S}'$ and $\mathcal{S}$ and for each size of sets the number of

Table 6.2: The number of components of 1000 sets generated from the $\mathcal{S}'$ class (top) and $\mathcal{S}$ class (bottom) using uniform random distributions

Size	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
2×2	837	163													
3×3	834	152	14												
4×4	840	145	15												
5×5	785	191	23	1											
7×7	746	225	27	2											
10×10	659	272	60	9											
20×20	314	403	196	73	12	2									
40×40	29	189	318	257	135	54	8	6	3	1					
60×60	2	36	118	240	260	171	106	40	19	6	1	1			
80×80		3	34	100	183	216	186	138	85	36	12	6	1		
100×100		1	9	30	69	160	189	190	149	92	55	32	17	6	1
2×2	824	176													
3×3	792	201	7												
4×4	770	206	23	1											
5×5	725	241	32	2											
7×7	656	280	54	9	1										
10×10	518	335	120	22	5										
20×20	175	326	315	123	51	10									
40×40	9	72	206	271	205	132	57	33	11	1	3				
60×60		10	33	102	169	224	192	126	87	28	12	5	1	1	
80×80			5	24	55	106	148	196	178	122	75	49	21	14	5
100×100				2	11	30	98	123	147	169	146	94	77	52	29

components follows a normal-like distribution with expectation value $E_{\mathcal{S}'}(n)$ and with variance $D^2_{\mathcal{S}'}(n)$ (and with expectation value $E_{\mathcal{S}}(n)$ and with variance $D^2_{\mathcal{S}}(n)$ in the $\mathcal{S}$ class). In order to verify this we decided to generate two more test sets made up of 1000 uniformly chosen discrete sets of sizes 200×200 and 500×500 from the $\mathcal{S}'$ class (the generation of this latter set took about half a day). Figure 6.5 shows the differences between the test results and the normal distributions with the estimated parameters.

With the aid of the formulas (6.7) and (6.8) (and with similar formulas in the classes $\mathcal{S}$, $\mathcal{S}'$, and $\mathcal{HV}$) it is also possible to describe the true distribution of the number of components of the generated hv-convex discrete set of the $\mathcal{HV}'$ class since, in this case, we can enumerate the discrete sets of a given class that has k components. Recall, however, that if we use recursive formulas in the reconstruction process which determine the exact number of the sets with a fixed number of components in a certain class then the generation complexity increases relative to the complexity of the simpler generation methods. Table 6.3 lists the expectation values and the variances of the variables which represent the number of components of a discrete set generated using a uniform random distribution from the $\mathcal{HV}'$ and $\mathcal{HV}$ classes when the size of the

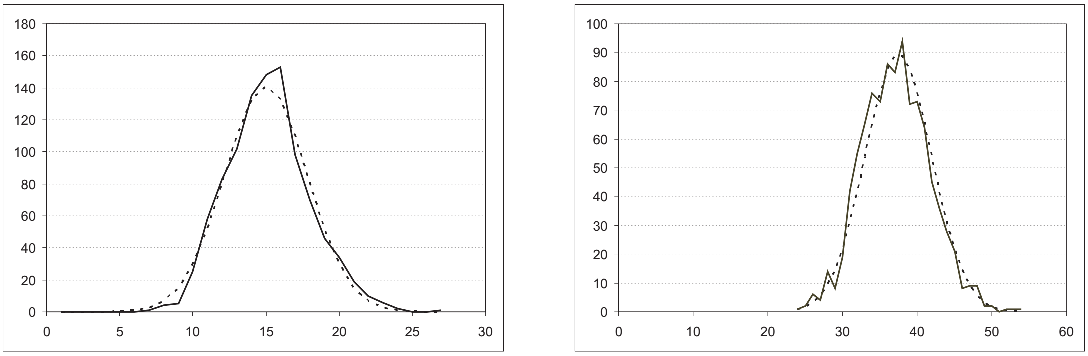


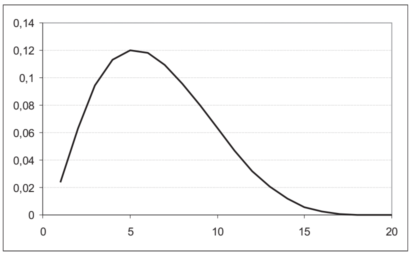
Figure 6.5: The distribution of the number of components in the test dataset (solid lines) and the corresponding normal distribution (dashed lines) for sets of $\mathcal{S}'$ of sizes 200×200 (left) and 500×500 (right)

Table 6.3: The expectation value $E_{\mathcal{HV}'}(n)$ ($E_{\mathcal{HV}}(n)$) and the variance $D^2_{\mathcal{HV}'}(n)$ ($D^2_{\mathcal{HV}}(n)$) of the components of a set with a minimal bounding rectangle of size $n \times n$ in the $\mathcal{HV}'$ ($\mathcal{HV}$) class. The values have been rounded to 5 digits

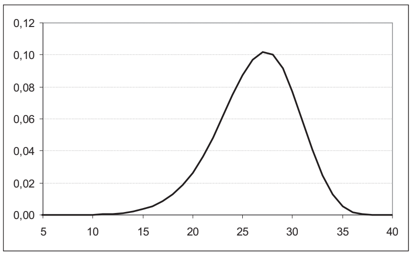
n	$E_{HV'}(n)$	$D^2_{HV'}(n)$	$E_{HV}(n)$	$D^2_{HV}(n)$
20	6.53981	9.84446	9.03570	8.12406
40	26.33821	16.00766	26.11090	9.54114
60	46.30283	12.92260	43.68220	10.00145
80	65.70631	12.05665	61.49588	10.72577
100	84.99456	11.80716	–	–
200	181.53870	12.45513	–	–

minimal bounding rectangle is $n \times n$ for a fixed $n \in \mathbb{N}$. In addition, the corresponding distributions are depicted in Fig. 6.6. For the $\mathcal{HV}$ class the statistics for sets of size larger than 80×80 could not be determined due to the huge time complexity of the generation algorithm. Even for sets of the $\mathcal{HV}$ class of size 80×80 the generation algorithm takes about a week to run. This shows the (perhaps single) drawback of our generation method, namely that it is applicable for generating general hv-convex discrete sets of moderate size only.

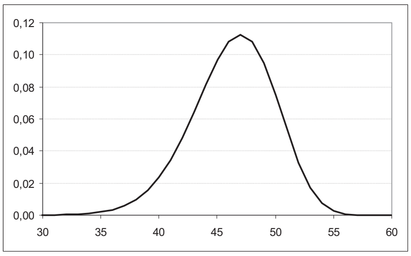
Despite this, statistics about the expected number of components can be especially useful in the reconstruction task. It tells us something about the discrete set to be reconstructed before we attempt to reconstruct it. Thus, such statistics opens the to for designing reconstruction algorithms in the future that exploit information known beforehand about the expected number of components. The author believes that such algorithms could be more effective in practice than the previously developed ones which do not make use of such prior knowledge.



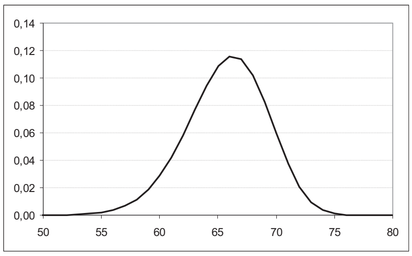
(a)



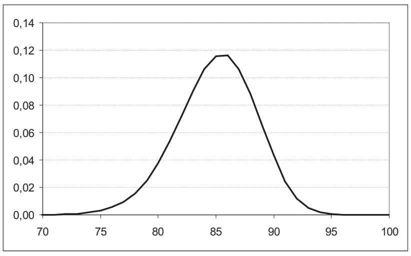
(b)



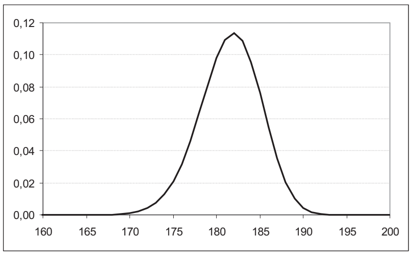
(c)



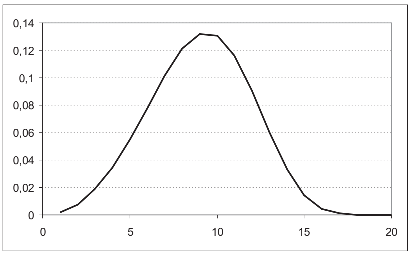
(d)



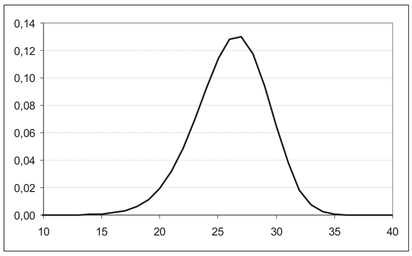
(e)



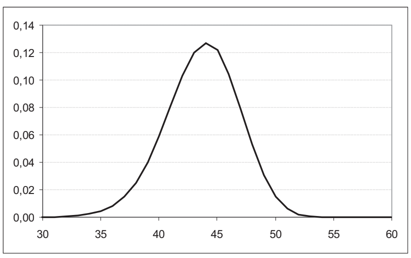
(f)



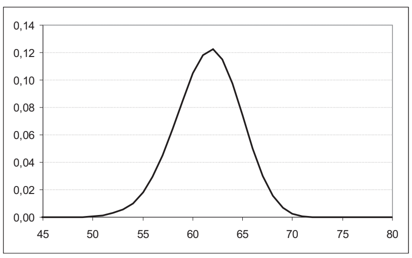
(g)



(h)



(i)



(j)

Figure 6.6: The distributions of the number of components – which depend on the size of the test data – in the $\mathcal{HV}'$ ((a)-(f)) and $\mathcal{HV}$ ((g)-(j)) classes

6.5 Conclusions and Discussion

In this chapter we developed several variants of a technique for reconstructing hv-convex discrete set using uniform random distributions.

The main advantage of our method is that it can be applied to any class of discrete set that have disjoint components if the components themselves can be generated from uniform random distributions by knowing their bounding rectangles, and if it is possible to enumerate them. To demonstrate this, we presented a simple example of generating discrete sets made up of discrete rectangles using uniform random distributions.

A further generalisation of our generation method is also possible, which can be used to generate discrete sets of certain classes using non-uniform distributions.

Finally, several statistical results are given that can be useful in analysing the complexity of reconstruction algorithms. Because of its huge computational complexity our generation method is usually applicable for discrete sets of moderate size only. This seems to be the only disadvantage of the method presented. Although certain statistics could be found using it, the on-line generation of sets of large size with this method is not feasible. The author thinks that this is not really such a big problem as benchmark sets used in testing reconstruction algorithms have to be generated once and can then be used over and over again. But, unfortunately, with this method the generation of large-size hv-convex sets (which also have empty rows or/and columns) using a uniform random distribution is not feasible. Hence it is an important and interesting open question of whether more sophisticated and more efficient generation techniques for the class of hv-convexes can be developed to overcome current difficulties.

The generation method described in this section and some of the statistics presented here can be found in [8].

Chapter 7

Conclusions

In binary tomography the use of geometrical priors in the reconstruction of binary images (or, equivalently, discrete sets) from their projections is an effective tool for overcoming ambiguity and intractability problems which arise from using a limited number of available projections. Variants of connectedness, convexity, and directedness are the most frequently used geometrical properties to facilitate the reconstruction process. In several points of this thesis it was mentioned that if the discrete set to be reconstructed has only one of these properties then the reconstruction usually does not become easier. In this dissertation we gave a far from exhaustive description of the hybrid combinations of the above key properties that can guarantee fast and rare ambiguous reconstruction.

We showed that the assumption that the discrete set to be reconstructed is *directed and convex* along certain directions can yield a unique reconstruction using the horizontal and vertical projections in polynomial time, but this result is very sensitive in the sense that even small changes in the direction of convexity can yield an enormous number of solutions with the same two projections.

Connectedness and convexity used together can also guarantee polynomial-time reconstruction using the horizontal and vertical projections. In binary tomography there are several discrete analogues of the continuous concept of convexity like hv-convexity and Q-convexity. The property of connectedness can also be defined on 8-adjacent points or only on 4-adjacent ones. Using any combination of the connectedness and convexity properties in the reconstruction process as prior information we get some good theoretical results about the number of possible solutions and some good information about the reconstruction complexity.

In theoretical binary tomography one of the most important tasks is to find ever broader classes of discrete sets in which the reconstruction using two or four projections can be achieved in polynomial time – in such a way that the reconstructed set is the same as the original one, i.e. it is uniquely determined by its projections. A commonly used approach here is to choose a class that has good properties in this sense and then attempt to weaken the geometrical properties that were described for the class. With this approach we also introduced the concept of decomposability and showed that every decomposable discrete set can be reconstructed from the horizontal, vertical, diagonal and antidiagonal projections in polynomial time. Then, after studying the reconstruction task in the class of decomposable discrete sets we found that it has a feature that can be exploited to develop a fast and accurate reconstruction heuristic for the class of hv-convex decomposable discrete sets. Figure 7.1 shows the relations between the classes studied in this dissertation, where it can be seen the class of decomposables introduced is in some sense one of the broadest known classes in which the reconstruction from a few projections can be performed in polynomial time.

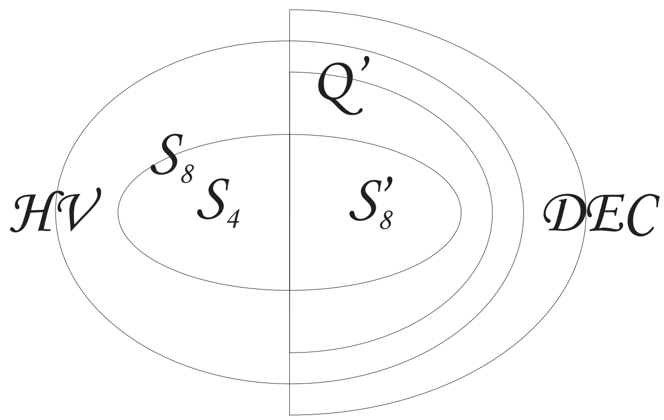


Figure 7.1: The connections between the classes studied in this dissertation

In the analysis of the performance of reconstruction algorithms we developed, the worst case complexity is not the only parameter which can be used to analyse the algorithm's performance. From an applications point of view it is even more important to know how the algorithm works in the average case. To be able to test the average time complexity (or the average accuracy) of a reconstruction algorithm which was developed for a certain class of discrete sets one has to generate elements of that class from a uniform random distribution. In this dissertation we also describe a technique which can generate elements of some classes (like the important class of hv-convex discrete sets) in a uniform way. The technique presented can easily be adapted to a wide variety of generation tasks including situations where by assigning weighting factors, some sets are preferred to others during the generation process.

With the aid of the generation method presented we were able to get some statistics about hv-convex discrete sets which are of help in designing efficient reconstruction algorithms.

The theoretical results given in this dissertation are interesting on their own right as well, but the insights we gain into the mathematical behaviour of binary tomography could be more important. I think that understanding the theory of binary tomography is the key to designing efficient reconstruction algorithms that can be applied in practice as well.

Appendix A

Summary in English

Computerized tomography is an imaging procedure that reconstructs two dimensional images from several hundred projections. In certain applications of tomography the acquisition of projections is time-consuming or/and expensive, but the number of possible grey scale values of the image to be reconstructed is quite small – in some cases just two (i.e. the image to be reconstructed is black-and-white). Binary tomography (BT) exploits the fact that the reconstruction should contain only black and white pixels in order to reduce the problems arising from using a small number of available projections.

Due to the limited number of available projections the reconstruction task in BT is often underdetermined, which means that there may be several different binary image solutions with the same projections. And a further complication is that the reconstruction from projections taken from certain directions can be NP-hard. One way of overcoming these problems is to suppose that the image to be reconstructed has certain geometrical properties like connectedness or convexity. By making use of such geometrical priors the reconstruction of binary images from their horizontal, vertical, diagonal and antidiagonal projections can be accomplished in polynomial time and the number of ambiguous solutions can be drastically reduced or in some cases even totally eliminated.

The chief aim of this dissertation was to present several techniques developed by the author which can be used to reconstruct binary images from their projections or to study the performance of reconstruction algorithms. Some of the techniques described are algorithms by which discrete sets with certain geometrical properties and given projections can be reconstructed. Most of these algorithms use just the horizontal and vertical projections to reconstruct the solution. But we also found a procedure that reconstructs so-called decomposable discrete sets from the horizontal, vertical, diagonal, and antidiagonal projections. This algorithm is an especially important result of this thesis since up to now very few reconstruction algorithms are known that use four projections. In addition, this algorithm can serve a basis for some reconstruction heuristics that use four projections as well.

Besides this, tools are presented by which allow one to investigate the number of solutions of a given reconstruction task. Some of them can be applied to demonstrate that the number of discrete sets with the same projections in a given class of discrete sets is quite limited; in some cases the solution of a reconstruction task is even uniquely determined. With the help of other methods presented it can be shown that in some classes there can be exponentially many discrete sets with the same projections.

After, a method is presented which allows one to generate discrete sets from several classes from a uniform random distribution. The generation method presented can be adapted to many classes. Such generation methods are very important in the study of the average performance of the reconstruction algorithms.

During the writing of this dissertation we always took great care to see how the newly achieved results relate to the known ones. Although - naturally - some questions was left open, the author hopes that the reader found this dissertation enjoyable and useful.

A.1 Summary by Chapters

Though Chapter 2 does not contain scientific contributions from the author, it does seek to provide an overview of the field of binary tomography. This chapter contains the basic definitions and notations that will be used in the later chapters. Moreover, it surveys of the main problems and applications of binary tomography and its mathematically related fields.

The remaining chapters follow an order of solving first simpler then evermore complicated problems. Chapter 3 examines the problem of reconstructing directed convex polyominoes from two projections. Although the restrictions for the set to be reconstructed are quite strong, the main difficulties of the reconstruction task can be examined for this class as well. This chapter also introduces some mathematical tools and findings that will be useful in the subsequent chapters where more difficult reconstruction problems will be investigated.

Chapter 4 presents an algorithm for reconstructing Q-convex discrete sets which have at least two components from two projections. The relationship between the classes of Q-convexes and hv-convex 8-connected discrete sets is also investigated. The reconstruction algorithm we developed outperforms the known ones for these classes. The theorems and mathematical results of this chapter form the basis of the reconstruction algorithms presented in the subsequent chapters.

Chapter 5 describes a decomposition technique which can reconstruct all the discrete sets that satisfy some additional properties and have the same four projections. Actually, this algorithm is numbered among the first ones that can carry out an exact reconstruction from four projections in polynomial time. This technique was the practical outcome of findings presented in the previous chapters, but some more technical theorems and lemmas are also needed. The effectiveness of this algorithm is tested by describing its worst-case performance on several previously studied classes. The main strength of the decomposition technique is that it can be applied to a wide variety of discrete sets after introducing some slight modifications in the decomposition framework. After analysing the feasibility of applying this technique to the reconstruction of hv-convex sets with a polynomial time complexity, we develop a heuristic reconstruction algorithm for the class of hv-convexes.

An analysis of the efficiency of reconstruction algorithms applied on the same class from the viewpoint of accuracy or running time allows us to examine under which circumstances certain algorithms outperform other ones. Using these findings we can design better reconstruction algorithms. In Chapter 6 we present a method for generating hv-convex discrete sets using uniform random distributions. With the help of this algorithm we can compare the effectiveness of reconstruction algorithms in the class of hv-convex discrete sets. In this chapter we also present some statistical results that are useful when we make this comparison. Afterwards, we demonstrate how this method can be adapted to generate elements from other classes of discrete sets using uniform random distributions as well. We also touch upon the issue of how the generation technique can be modified to generate discrete sets from non-uniform distributions.

A.2 Key Points of the Thesis

In the following a listing of the most important results of the dissertation is given. Table A.1 shows which thesis point is described in which publication by the author.

I.) Horizontally or vertically convex NE-directed polyominoes can be reconstructed from their horizontal and vertical projections uniquely in polynomial time. The author investigated how varying the direction of convexity influences the above result. He found that the above result can be extended to diagonally convex NE-directed polyominoes as well, but assuming convexity along any other directions can yield an exponentially large number of solutions.

II.) The author developed a fast algorithm for reconstructing Q-convex discrete sets which have at least two components using two projections. The author's algorithm has a worst case time complexity of $O(mn \cdot \min\{m^2, n^2\})$ and it can locate all the solutions of a given reconstruction task.

III.) The author showed that the class of hv-convex 8-connected discrete sets form a subclass of the class of Q-convexes. He compared his algorithm on the class of hv-convex 8-connected sets to previously published ones and found that the new algorithm outperforms the former ones from the viewpoint of the worst case and the viewpoint of the average execution time complexity. He also showed that for Q-convex but not 8-connected sets his algorithm can be speeded up to having a worst case time complexity of $O(mn)$, and in this case the number of possible solutions of the same reconstruction problem is at most two.

IV.) The author introduced the class of decomposable discrete sets and presented a polynomial-time reconstruction algorithm for this class using four projections. He investigated the relationship between the classes of decomposables and Q-convexes and demonstrated its consequences for the reconstruction task complexity for some well-known classes when four projections are available.

	[2]	[3]	[5]	[6]	[7]	[8]
I.	•					
II.		•				
III.		•	•			
IV.		•	•	•		
V.					•	
VI.						•

Table A.1: The connection between the thesis points and the corresponding publications

V.) The author investigated the possibility of extending the decomposition technique to the class of hv-convexes. He presented a fast and accurate heuristic reconstruction algorithm for hv-convex sets with decomposable configurations.

VI.) The efficiency of recently developed exact or heuristical reconstruction algorithms are often tested on the class of hv-convexes. The author described a method for generating elements of this class using a uniform random distribution where an exact comparison of several reconstruction algorithms can be made from the viewpoint of average execution time. The method can readily be extended to other classes of discrete sets which have disjoint components. Using this method the author presented statistics on properties of hv-convex sets that can affect the performance of reconstruction algorithms.

Appendix B

Summary in Hungarian

A számítógépes tomogáfia egy olyan képalkotó eljárás, mely kétdimenziós képek előállítására nyújt lehetőséget pár száz úgy nevezett képi vetület felhasználásával. Bizonyos alkalmazások esetén a vetületek készítése nagyon időigényes vagy költséges lehet, ugyanakkor ezzel egyidőben ismert, hogy az előállítandó képen csak néhány szürkeségi érték megjelenése várható (például, ha a rekonstruálandó kép fekete-fehér). A bináris tomográfia (BT) azt vizsgálja, hogyan használható ki az az információ, hogy a rekonstrukció fekete-fehér, annak érdekében, hogy a kevés számú rendelkezésre álló vetületből adódó nehézségeket kiküszöböljük.

A BT-ban mindig erősen korlátozott a rendelkezésre álló vetületek száma, így a rekonstrukciós feladat általában jelentősen aluldefiniált. Ez azt eredményezi, hogy ugyanazokkal a vetületekkel több (esetenként lényegesen eltérő) kép is előállhat. Mindezek mellett, bizonyos irányból vett vetületek esetén a rekonstrukció NP-teljes is lehet. Az egyik megközelítés, melynek segítségével ezek a problémák áthidalhatóak, azzal a feltételezéssel él, hogy a rekonstruálandó kép rendelkezik valamilyen geometriai tulajdonsággal (például konvex, összefüggő, stb.). Ilyen geometriai tulajdonoságok feltételezésével elérhető, hogy a rekonstrukció a horizontális, vertikális, diagonális és antidiagonális vetületekből polinomiális időben végrehajtható legyen és a lehetséges megoldások száma is jelentősen csökkenjen.

Jelen disszertáció célja a Szerző által kidolgozott olyan technikák bemutatása volt, melyek segítségével lehetőség nyílik bináris képek vetületekből történő előállítására, illetve a rekonstrukció teljesítményének vizsgálatára. A közölt algoritmusok nagy része horizontális és vertikális vetületeket használ a kép megalkotásához. Azonban egy olyan technika is ismertetésre került, mely emelett még a diagonális és antidiagonális vetületeket is alkalmazza úgy nevezett felbontható halmazok rekonstruálására. Ez az eljárás különösen fontos eredménye a disszertációnak, mivel a mai napig igen kevés rekonstrukciós algoritmus ismert, mely négy vetület igénybevételével pontos megoldást tud szolgáltatni. Továbbá ez a módszer alapját képezheti újabb, négy vetületet használó rekonstrukciós heurisztikák kidolgozásának is.

Ezek mellett olyan eszközöket is ismertettünk, melyekkel a rekonstrukció lehetséges megoldásainak számát vizsgálhatjuk. Megmutattuk, hogy bizonyos geometriai tulajdonságok teljesülése esetén a rekonstrukció egyértelmű, míg más esetekben exponenciálisan sok megoldása lehet ugyanannak a rekonstrukciós feladatnak.

Végezetül egy olyan eljárást mutattunk be, melynek segítségével bizonyos osztályokból véletlenszerűen, egyenletes eloszlás mellett generálhatunk diszkrét halmazokat. Ilyen generáló algoritmusok megtervezése rendkívül fontos a különböző rekonstrukciós algoritmusok átlagos teljesítményének vizsgálatakor.

A disszertáció írása során végig igyekeztünk nagy figyelmet fordítani annak vizsgálatára, miként kapcsolódnak új megállapításaink, eszközeink a terület korábbi eredményeihez. Elkerülhetetlenül néhány kérdés nyitott maradt, ennek ellenére a Szerző reméli, hogy az Olvasó élveztesnek és hasznosnak találja a dolgozatot.

B.1. A fejezetek áttekintése

A 2. fejezet nem a Szerző eredményeivel foglalkozik. Ezen fejezet célja, hogy betekintést adjon a bináris tomográfiába. Itt ismertetjük a dolgozatban későbbiekben használt definíciókat és jelöléseket, és itt tárgyaljuk azt is, hogy hogyan kapcsolódik a bináris tomográfia a matematika néhány fontos területéhez.

A további fejezetekben először egyszerűbb, majd egyre bonyolultabb problémák megoldását mutatjuk be. A 3. fejezet irányított konvex poliominók két vetületből történő rekonstruálásával foglalkozik. Habár ebben az esetben a rekonstruálandó halmazra tett megszorítások meglehetősen szigorúak, a rekonstrukciós feladatok megoldása során felmerülő fő problémák már ebben a halmazosztályban is vizsgálhatók. Ez a fejezet vezet be néhány olyan matematikai fogást és eredményt, melyek a további fejezetek bonyolultabb problémáinak vizsgálatában is hasznos eszköznek bizonyulnak.

A 4. fejezet egy olyan algoritmust ismertet, amely két vetületből rekonstruál legalább két komponensből álló, Q-konvex diszkrét halmazokat. Megvizsgáljuk a Q-konvex és a hv-konvex 8-összefüggő diszkrét halmazok kapcsolatát is. Ezen fejezet tételei és matematikai eredményei képezik az alapját az 5. fejezetben ismertetésre kerülő rekonstrukciós algoritmusoknak.

Az 5. fejezet egy dekompozíciós technikát ír le, amely az összes olyan diszkrét halmazt képes négy vetületből rekonstruálni, melyek kielégítenek néhány speciális geometriai feltételt. Tulajdonképpen ez az algoritmus egyike az első olyanoknak, melyek pontos rekonstrukciót adnak négy vetületből polinomiális időben. Az ismertetett eljárás jórészt az előző fejezet eredményein alapszik, de néhány újabb technikai jellegű lemmára és tételre is szükség van. Az algoritmus hatékonyságát néhány előzőleg bemutatott osztályon elemezzük a legrosszabb eset vizsgálattal. Megmutatjuk, hogy ezekben az osztályokban a diagonális és antidiagonális vetületek használata minden esetben jelentősen gyorsabb és megbízhatóbb rekonstrukciót eredményez, mintha csak a horizontális és vertikális vetületeket használnánk. A dekompozíciós technika egyik fő ereje abban rejlik, hogy néhány apróbb módosítással egyéb halmazosztályokra is alkalmazható. A fejezet utolsó részében azt mutatjuk meg, hogyan illeszthető a technika a hv-konvex halmazok osztályára. Megvizsgáljuk, hogy milyen lehetőségeink vannak arra, hogy a dekompozíciós technikát úgy használjuk erre az osztályra, hogy a rekonstrukció még polinomiális időben elvégezhető legyen. Ezek után kifejlesztünk egy gyors és pontos rekonstrukciós heurisztikát a hv-konvex halmazosztály egy részosztályára.

Ugyanarra a halmazosztályra kidolgozott különböző rekonstrukciós algoritmusok összehasonlítása átlagos pontosság és futási idő szempontjából lehetőséget ad arra, hogy megvizsgáljuk, mely algoritmusok hatékonyabbak másikaknál bizonyos körülmények között. Ezen ismeretek alapján újabb, még hatékonyabb algoritmusok tervezése válik lehetővé. A 6. fejezetben egy olyan eljárást ismertetünk, mellyel hv-konvex halmazok generálhatóak véletlenszerűen, egyenletes eloszlás mellett. Ezen eljárás segítségével összehasonlíthatjuk a hv-konvex halmazosztályra kidolgoztt rekonstrukciós algoritmusok hatékonyságát. A fejezetben az összehasonlítás szempontjából fontos statisztikákat is közlünk. Emellett megmutatjuk, hogy az ismertetett algoritmus hogyan módosítható, amennyiben más osztályból, vagy más eloszlás szerint származó halmazok generálására van igény.

B.2. Az eredmények tézisszerű összefoglalása

Az alábbiakban hat tézispontba rendezve összegezzük a Szerző kutatási eredményeit. A kutatásokból származó publikációkat, valamint azok tartalmának az egyes tézispontokhoz való viszonyát a B.1. táblázat tekinti át.

I.) Egy korábbi eredmény szerint a horizontálisan vagy vertikálisan konvex ÉK-irányított poliominók a horizontális és vertikális vetületeikből egyértelműen rekonstruálhatók polinomiális időben. A Szerző megvizsgálta, hogy a konvexitás irányának változtatása milyen módon befolyásolja a fenti eredményt. Azt tapasztalta, hogy a fenti tétel továbbra is igaz marad, ha diagonális konvexitást feltételezünk a rekonstruálandó ÉK-irányított poliominóról. A Szerző azt is bizonyította, hogy bármilyen más irányú konvexitás feltételezése estén előfordulhat, hogy exponenciálisan sok megoldás lesz ugyanazokkal a horizontális és vertikális vetületekkel.

II.) A Szerző kidolgozott egy $O(mn \cdot \min\{m^2, n^2\})$ legrosszabb futási idejű algoritmust, mely a horizontális és vertikális vetületekből rekonstruálja az összes azokkal a vetületekkel rendelkező olyan Q-konvex halmazt, melynek legalább két komponense van. Az algoritmus az adott feladat összes megoldását megtalálja.

III.) A Szerző megmutatta, hogy a hv-konvex 8-összefüggő halmazok részosztályát képezik a Q-konvex halmazok osztályának. Összehasonlítva az általa kidolgozott algoritmust a korábban publikáltakkal azt találta, hogy a hv-konvex 8-összefüggő halmazok osztályán az nemcsak a legrosszabb eset, de az átlagos futási idő tekintetében is gyorsabb a korábbiaknál. Ezen kívül a Szerző azt is megmutatta, hogy a Q-konvex, de nem 8-összefüggő halmazok esetén a rekonstrukció a horizontális és vertikális vetületekből $O(mn)$ időben megoldható és a lehetséges megoldások száma legfeljebb kettő.

IV.) A Szerző bevezette a felbontható diszkrét halmazok osztályát és egy olyan polinomiális futási idejű rekonstrukciós algoritmust adott erre az osztályra, mely négy

	[2]	[3]	[5]	[6]	[7]	[8]
I.	•					
II.		•				
III.		•	•			
IV.		•	•	•		
V.					•	
VI.						•

B.1. táblázat. A tézispontok és a Szerző publikációinak viszonya

vetületet használ. Megvizsgálta a kapcsolatot a felbontható és a Q-konvex halmazok osztályai között és összegezte az ebből a kapcsolatból adódó következményeket néhány jól ismert halmazosztály rekonstrukciós bonyolultságára vonatkozólag, amennyiben a rekonstrukció során négy vetület használható.

V.) A szerző megvizsgálta, hogyan terjeszthető ki a négy vetületet használó rekonstrukciós technika a hv-konvex halmazok osztályra. Ezek alapján kidolgozott egy gyors és pontos heurisztikát olyan diszkrét halmazok négy vetületből történő rekonstruálására, melyek hv-konvexek és a komponenseik úgy nevezett felbontható konfigurációt alkotnak.

VI.) Az újonnan ismertetésre kerülő egzakt vagy heurisztikus rekonstrukciós algoritmusok hatékonyságát gyakran tesztelik a hv-konvex osztályon. A Szerző bemutatott egy eljárást, mellyel ennek az osztálynak az elemei egyenletes eloszlás szerint generálhatók. Ezen eljárás segítségével a különböző rekonstrukciós algoritmusok pontosan összemérhetővé válnak az átlagos futási idő tekintetében. A Szerző néhány fontos statisztikát is ismertetett a hv-konvex halmazosztályra vonatkozólag, melyek összefüggésben állnak a rekonstrukció nehézségével. Az ismertetett generáló metódus könnyen kiterjeszthető számos olyan diszkrét halmazokat tartalmazó osztályra, melyek elemeinek komponensei diszjunktak.

Bibliography

[1] 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 (3)** (1979) 121–123.

[2] P. Balázs, The number of line-convex directed polyominoes having the same orthogonal projections, *Lecture Notes in Comput. Sci.* **4245** (2006) 77–85.

[3] P. Balázs, Decomposition algorithms for reconstructing discrete sets with disjoint components, In [57], pp. 153–174 (2007).

[4] P. Balázs, E. Balogh, A. Kuba, A fast algorithm for reconstructing hv-convex 8-connected but not 4-connected discrete sets, *Lecture Notes in Comput. Sci.* **2886** (2003) 388–397.

[5] P. Balázs, E. Balogh, A. Kuba, Reconstruction of 8-connected but not 4-connected hv-convex discrete sets, *Disc. Appl. Math.* **147** (2005) 149–168.

[6] P. Balázs, A decomposition technique for reconstructing discrete sets from four projections, *Image and Vision Computing*, in press.

[7] P. Balázs, On the ambiguity of reconstructing hv-convex binary matrices with decomposable configurations, *Acta Cybernetica*, accepted.

[8] P. Balázs, Generation and empirical investigation of hv-convex discrete sets, *Lecture Notes in Comput. Sci.* **4522** (2007) 344–353.

[9] P. Balázs, Reconstruction of decomposable discrete sets from four projections, *Lecture Notes in Comput. Sci.* **3429** (2005) 104–114.

[10] P. Balázs, Reconstruction of discrete sets from four projections: strong decomposability, *Elec. Notes in Discrete Math.* **20** (2005) 329–345.

[11] E. Balogh, A. Kuba, Cs. Dévényi, A. Del Lungo, Comparison of algorithms for reconstructing hv-convex discrete sets, *Lin. Alg. Appl.* **339** (2001) 23–35.

[12] E. Barcucci, S. Brunetti, A. Del Lungo, M. Nivat, Reconstruction of discrete sets from three or more projections, *Lecture Notes in Comput. Sci.* **1767** 199–210.

[13] E. Barcucci, A. Del Lungo, M. Nivat, R. Pinzani, Reconstructing convex polyominoes from horizontal and vertical projections, *Theor. Comput. Sci.* **155** (1996) 321–347.

[14] E. Barcucci, A. Del Lungo, M. Nivat, R. Pinzani, Medians of polyominoes: A property for the reconstruction, *Int. J. Imaging Systems and Techn.* **9** (1998) 69–77.

[15] K.J. Batenburg, An evolutionary approach for discrete tomography, *Disc. Appl. Math.* **151** (2005) 36–54.

[16] K.J. Batenburg, A new algorithm for 3D binary tomography, *Elec. Notes in Discrete Math.* **20** (2005) 247–261.

[17] K.J. Batenburg, A network flow algorithm for binary image reconstruction, *Lecture Notes in Comput. Sci.* **4245** (2006) 86–97.

[18] K.J. Batenburg, W.J. Palenstijn, On the reconstruction of crystals through discrete tomography, *Lecture Notes in Comput. Sci.* **3322** (2004) 23–37.

[19] J. Baumann, Z. Kiss, S. Krimmel, A. Kuba, A. Nagy, L. Rodek, B. Schillinger, J. Stephan, Discrete tomography methods for non-destructive testing, In [57], pp. 303–332 (2007).

[20] J.A. Browne, M. Koshy, CT-assisted engineering and manufacturing, In [56], pp. 345–383 (1999).

[21] R.A. Brualdi, Matrices of zeros and ones with fixed row and column sum vectors, *Lin. Algebra and Its Appl.* **33** (1980) 159–231.

[22] R.A. Brualdi, G. Dahl, Constructing $(0,1)$-matrices with given line sums and certain fixed zeros In [57], pp. 113–124 (2007).

[23] S. Brunetti, A. Daurat, Reconstruction of discrete sets from two or more X-rays in any direction, *Proceedings of the seventh International Workshop on Combinatorial Image Analysis* (2000) 241–258.

[24] S. Brunetti, A. Daurat, An algorithm reconstructing convex lattice sets, *Theor. Comput. Sci.* **304**, 35–57 (2003).

[25] S. Brunetti, A. Daurat, Random generation of Q-convex sets, *Theor. Comput. Sci.* **347** (2005) 393–414.

[26] S. Brunetti, A. Daurat, Determination of Q-convex bodies by X-rays, *Elec. Notes in Discrete Math.* **20** (2005) 67–81.

[27] S. Brunetti, A. Daurat, Stability in Discrete Tomography: Some Positive Results, *Disc. Appl. Math.* **147(2-3)** (2005) 207–226.

[28] S. Brunetti, A. Daurat, A. Del Lungo, Approximate X-rays reconstruction of special lattice sets, *Pure Math. Appl.* **11(3)** (2000) 409–425.

[29] S: Brunetti, A. Daurat, A. Kuba, Fast filling operations used in the reconstruction of convex lattice sets, *Lecture Notes in Comput. Sci.* **4245** (2006) 98–109.

[30] S. Brunetti, A. Del Lungo, F. Del Ristoro, A. Kuba, M. Nivat, Reconstruction of 4- and 8-connected convex discrete sets from row and column projections, *Lin. Alg. Appl.* **339** (2001) 37–57.

[31] J.M. Carazo, C.O. Sorzano, E. Rietzel, R. Schröder, R. Marabini, Discrete tomography in electron microscopy In [56], pp. 405–416 (1999).

[32] B.M. Carvalho, G.T. Herman, S. Matej, C. Salzberg., E. Vardi, Binary tomography for triplane cardiography, *Lecture Notes in Comput. Sci.* **1613** (1999) 29–41.

[33] G. Castiglione, A. Restivo, Reconstruction of L-convex polyominoes, *Elec. Notes in Discrete Math.* **12** (2003).

[34] G. Castiglione, A. Frosini, A. Restivo, S. Rinaldi, A tomographical characterization of L-convex polyominoes, *Lecture Notes in Comput. Sci* **3425** (2005) 115–125.

[35] M.T. Chan, G. T. Herman, E. Levitan, Probabilistic modeling of discrete images, In [56], pp. 213–235 (1999).

[36] M. Chrobak, Ch. Dürr, Reconstructing hv-convex polyominoes from orthogonal projections, *Inform. Process. Lett.* **69(6)** (1999) 283–289.

[37] M.C. Costa, F. Jarray, C. Picouleau, Reconstruction of binary matrices under adjacency constraints, *Elec. Notes in Discrete Math.* **20** (2005) 281–197.

[38] G. Dahl, T. Flatberg, Optimization and reconstruction of hv-convex $(0,1)$-matrices, *Disc. Appl. Math.* **151** (2005) 93–105.

[39] A. Daurat, Determination of Q-convex sets by X-rays, *Theor. Comput. Sci.* **332(1-3)** (2005) 19–45.

[40] A. Del Lungo, Polyominoes defined by two vectors, *Theor. Comput. Sci.* **127** (1994) 187–198.

[41] A. Del Lungo, A. Frosini, M. Nivat, L. Vuillon, Discrete tomography: reconstruction under periodicity constraints, *Lecture Notes in Comput. Sci.* **2380** (2002) 38–56.

[42] A. Del Lungo, M. Nivat, R. Pinzani, The number of convex polyominoes recostructible from their orthogonal projections, *Discrete Math.* **157** (1996) 65–78.

[43] M.P. Delest, G. Viennot, Algebraic laguages and polyominoes enumertaion, *Theor. Comput. Sci.* **34** (1984) 169–206.

[44] A. Fazekas, Skeletonization on projection pictures, *Lecture Notes in Comput. Sci.* **1689** (1999) 419–426.

[45] A. Fazekas, G.T. Herman, A. Matej, On processing binary pictures via their projections, *Int. J. Imaging Systems and Techn.* **9** (1998) 99–100.

[46] D. Gale, A theorem on flows in networks, *Pacific J. Math.* **7** (1957) 1073–1082.

[47] R.J. Gardner, P. Gritzmann, Discrete tomography: determination of finite sets by X-rays, *Trans. Amer. Math. Soc.* (1997) 2271–2295.

[48] R.J. Gardner, P. Gritzmann, Uniqueness and complexity in discrete tomography, In [56], 85–113 (1999).

[49] R.J. Gardner, P. McMullen, On Hammers's X-ray problem, *J. London Math. Soc.* **21** (1980) 171–175.

[50] M. Gebala, The reconstruction of convex poyominoes from horizontal and vertical projections, *Lecture Notes in Comput. Sci.* **1521** (1998) 350–359.

[51] I. Gessel, On the number of convex polyominoes, *Ann. Sci. Math. Québec* **24** (2000) 63–66.

[52] S.W. Golomb, *Polyominoes*, Charles Scriber's Sons, New York, 1965.

[53] L. Hajdu, R. Tijdeman, An algorithm for discrete tomography, *Lin. Alg. Appl.* **339** (2001) 147–169.

[54] L. Hajdu, R. Tijdeman, Algebraic aspects of discrete tomography, *Journal für die reine und angewandte Math.* **534** (2001) 119–128.

[55] G.T. Herman, Reconstruction of binary patterns from a few projections, In *Intern. Computing Symposium 1973* (1974) 371–378.

[56] G.T. Herman, A. Kuba (Eds.), *Discrete Tomography: Foundations, Algorithms and Applications*, Birkhäuser, Boston, 1999.

[57] G.T. Herman, A. Kuba (Eds.), *Advances in Discrete Tomography and its Applications*, Birkhäuser, Boston, 2007.

[58] W. Hochstättler, M. Loebl, C. Moll, Generating convex polyominoes at random, *Discrete Math.* **153** (1996) 165–176.

[59] A. Imiya, A. Torii, K. Sato, Tomography on finite graphs, *Elec. Notes in Discrete Math.* **20** (2005) 217–232.

[60] R.W. Irving, M.R. Jerrum, Three-dimensional data security problems, *SIAM J. Comput.* **23** (1994) 170–184.

[61] C. Kisielowski, P. Schwander, F.H. Baumann, M. Seibt, Y. Kim, A. Ourmazd, An approach to quantitative high-resolution electron micorscopy of crystalline materials, *Ultramicroscopy* **58** (1995) 131-155.

[62] Z. Kiss, L. Rodek, A. Kuba, Image reconstruction and correction methods in tomography, *Acta Cybernetica* **17(3)** 557–587.

[63] D. Kölzow, A. Kuba, A. Volcic, An algorithm for reconstructing convex bodies from their projections, *Discrete and Comput. Geom.* **4** (1989) 205–237.

[64] A. Kuba, The reconstruction of two-directionally connected binary patterns from their two orthogonal projections, *Comp. Vision, Graphics, and Image Proc.* **27** (1984) 249–265.

[65] A. Kuba, Reconstruction of unique binary matrices with prescibed elements, *Acta Cybernetica* **12** (1995) 57–70.

[66] A. Kuba, Reconstruction in different classes of 2D discrete sets, *Lecture Notes in Comput. Sci.* **1568** (1999) 153–163.

[67] A. Kuba, Reconstruction of two-valued functions and matrices, In [56], pp. 137–162 (1999).

[68] A. Kuba, E. Balogh, Reconstruction of convex 2D discrete sets in polynomial time, *Theor. Comput. Sci.* **283** (2002) 223-242.

[69] A. Kuba, A. Nagy, E. Balogh, Reconstruction of hv-convex binary matrices from their absorbed projections, *Disc. Appl. Math.* **139** (2004) 137–148.

[70] A. Kuba, A. Volcic, Characterisation of measurable plane sets which are reconstructible from their two projections, *Inverse Problems* **4** (1988) 513–527.

[71] L. Latecki, U. Eckhardt, A. Rosenfeld, Well-composed sets, *Computer Vision and Image Understanding* **61(1)** (1995) 70–83.

[72] G.G. Lorentz, A problem of plane measure, *Am. J. Math.* **71** (1949) 417–426.

[73] E. Levitan, M. Chan, G.T. Herman, Image-modeling Gibbs priors, *Graph. Models Image Proc.* **57** (1995) 117–130.

[74] Z. Mao, R. N. Strickland, Image sequence processing for target estimation in forward-looking infrared imagery, *Optical Engrg.* **27** (1988) 541–549.

[75] S. Matej, A. Vardi, G.T. Herman, E. Vardi, Binary tomography using Gibbs prios, In [56], pp. 191–212 (1999).

[76] N. Metropolis, A. Rosenbluth, A. T. Rosenbluth., E. Teller, Equation of state calculation by fast computing machines, *Journal of Chem. Phys.* **21** (1953) 1087–1092.

[77] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolutionary Programs*, Springer Verlag, 1996.

[78] D.G.W. Onnasch, G.M.P. Prause Heart chamber reconstructionfrom biplane angiography, In [56], pp. 385–403 (1999).

[79] C. Picouleau, S. Brunetti, A. Frosini, Reconstructing a binary matrix under timetabling constraints, *Elec. Notes in Discrete Math.* **20** (2005) 99–112.

[80] H.J. Ryser, Combinatorial properties of matrices of zeros and ones, *Canad. J. Math.* **9** (1957) 371–377.

[81] P.M. Salzberg, R. Figueroa, Tomography on the 3D-torus and crystals, In [56], pp. 417–434 (1999).

[82] A.R. Schilferstein, Y.T. Chien, Some properties of image processing operations on projection sets obtained from digital pictures *IEEE Trans. Comput.* **C-26** (1977) 958–970.

[83] T. Schüle, C. Schnörr, S. Weber, J. Hornegger, Discrete tomography by convex-concave regularization and D.C. programming, *Disc. Appl. Math.* **151** (2005) 229–243.

[84] P. Schwander, C. Kisielowski, M. Seibt, F.H. Baumann, Y. Kim, A. Ourmazd, Mapping projected potential, interfacial roghness, and composition in general crystalline solids by quantitative transmission electron microscopy, *Phys. Rev. Letters* **71** (1993) 4150–4053.

[85] B. Sharif, B. Sharif, Discrete tomography in discrete deconvolution: deconvolution of binary images using Ryser's algorithm, *Elec. Notes in Discrete Math.* **20** (2005) 555–571.

[86] V. Shoup, NTL: A library for doing number theory
`http://www.shoup.net/ntl`

[87] N.J.A. Sloane, The on-line encyclopedia of integer sequences
`http://www.research.att.com/~njas/sequences/`

[88] G.W. Woeginger, The reconstruction of polyominoes from their orthogonal projections, *Inform. Process. Lett.* **77** (2001) 225–229.

[89] S. Zopf, A. Kuba, Reconstruction of measurable plane sets from two generalized projections, *Elec. Notes in Discrete Math.* **20** (2005) 47–66.

Index

$C(F)$, 42
C_F, 28
L_F^{NE}, 29
L_F^{NW}, 29
$\mathcal{HV}'$ class, 71
$\mathcal{S}$ class, 66
$\mathcal{S}'$ class, 65

adjacent points, 14
Algorithm
- 2-RECQ', 30
- 4-RECDEC, 47
- 4-RECHV, 56
- DCP, 20
- GENHV', 70
- GENHV-S', 66

bridge, 21

center, 42
compatible vectors, 9
component, 14
- NW, 40

connectedness, 14
consistency, 7
convexity, 14
- Q-, 15

cumulated vectors, 8

decomposable
- configuration, 55
- discrete set, 40

directedness, 16
discrete set, 6
- size of, 6
- uniquely determined, 7

equality position, 29

gluing, 39
gluing sequence, 41

lattice direction, 6
lattice line, 6
level, 21

path
- 4, 8-, 14
- northeast, 16

polyomino, 14
Procedure
- DecomposeNW, 46
- DecomposeNW', 57

projection, 7
- antidiagonal, 8
- diagonal, 8
- horizontal, 8
- vertical, 8

property
- (α), (β), (γ), 40

quadrant, 15

reconstruction, 7

SCDR, 25
source, 16
switching component, 9

tomography
- binary, 6
- computerized, 5
- discrete, 5

type
- of a set of $\mathcal{Q}'$, 26
- of a set of $\mathcal{DEC}$, 40

uniqueness, 7