Theoretical Computer Science 411 (2010) 3123–3128

Contents lists available at ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Reconstructing $h\upsilon$-convex multi-coloured polyominoes

Adam Bains, Therese Biedl *

David R. Cheriton School of Computer Science, University of Waterloo, Waterloo, ON N2L 3G1, Canada

ARTICLE INFO

Article history:
Received 4 December 2009
Received in revised form 20 April 2010
Accepted 25 April 2010
Communicated by B. Durand

Keywords:
Discrete tomography
Combinatorial problems
Algorithms
Computational geometry
Theory of computation

ABSTRACT

In this paper, we consider the problem of reconstructing polyominoes from information about the thickness in vertical and horizontal directions. We focus on the case where there are multiple disjoint polyominoes (of different colours) that are $h\upsilon$-convex, i.e., any intersection with a horizontal or vertical line is contiguous. We show that reconstruction of such polyominoes is polynomial if the number of colours is constant, but NP-hard for an unbounded number of colours.

1. Introduction

The field of discrete tomography concerns reconstruction of objects given information about the thickness of the object in various projections. See the books by Herman and Kuba [5,6] for an extensive overview of this exciting field with many applications in medical imaging.

One special case is when the object to be reconstructed is a binary matrix with m rows and n columns, and the given information is the row-sums and column-sums of the matrix. Testing whether such a matrix exists and finding it can be done easily with flow methods. However, of more interest is the case where the object is supposed to be a *polyomino*, i.e., from every *black cell* (an entry of the matrix that is 1) to every other black cell, there exists a path along black cells that are adjacent horizontally or vertically.

Reconstructing polyominoes is NP-hard, even if all black cells within each row are contiguous (the polyomino is *h-convex*) or all black cells within each column are contiguous (the polyomino is *υ-convex*). Surprisingly, if the polyomino must be *$h\upsilon$-convex* (i.e., both h-convex and υ-convex), reconstructing it from row-sums and column-sums becomes polynomial. See Chapter 7 of [5] for references and an overview of these results.

We study here reconstruction of objects that are the union of multiple disjoint objects, each of which has a different colour. This has applications in the reconstruction of polyatomic crystals: the number of atoms of each kind in a projection can be determined using a high-resolution transmission electron microscope. See [7,9] for details. This problem also appears in a recreational puzzle called "Color Pic-a-pix"; see www.conceptispuzzles.com. The general problem (with no restriction on the shape of the objects) was proved to be NP-hard even for three colours [3], and very recently even for two colours [4]. The NP-hardness proofs for various cases of a single-colour immediately imply NP-hardness of multi-colour versions of the same cases. Hence, the only case that could possibly be polynomial is the case of multiple $h\upsilon$-convex polyominoes.

We resolve this case here, and hence study the following problem: Given C colours $\{1, \ldots, C\}$, and C sets of density-vectors (h_i^c) and (υ_j^c) for $i = 1, \ldots, m$, $j = 1, \ldots, n$ and $c = 1, \ldots, C$, do there exist C binary matrices $(x_{i,j}^c)$ such that

* Corresponding author. Tel.: +1 519 888 4567.
E-mail address: biedl@uwaterloo.ca (T. Biedl).

0304-3975/$ – see front matter © 2010 Elsevier B.V. All rights reserved.
doi:10.1016/j.tcs.2010.04.041

for each c the matrix $(x^c_{i,j})$ is an hv-convex polyomino with row-sums and column-sums (h^c_i) and (v^c_j), and such that the polyominoes are disjoint, i.e., for any i, j we have $\sum_c x^c_{i,j} \leq 1$? We call this the *C-colour hv-convex polyomino reconstruction problem.*

We show that this problem is polynomial if the number of colours is a constant, but becomes NP-hard if the number of colours is unbounded.

2. Few colours

The natural approach for reconstructing C hv-convex polyominoes is to take one of the existing algorithms to reconstruct a single hv-convex polyomino and modify it so that it handles multiple polyominoes and ensures that they are disjoint. This can in fact be done easily with the algorithm given by Barcucci et al. [1], and yields an algorithm for the C-colour reconstruction that takes time $O(C^2m^{2C+2}n^{2C+2})$. We will not give the details of this, since with a different approach the time complexity can be improved significantly.

There are faster algorithms for single-colour hv-convex polyomino reconstruction, and we tried to generalize the currently fastest known, which is the one given by Chrobak and Dürr [2] and takes $O(\min\{m, n\}^2mn)$ time. We did not succeed in generalizing this algorithm to multiple colours. The main difficulty is that this algorithm stores the computed polyomino implicitly (by storing the "blank area" around it), and hence there is no easy way to add a constraint to ensure that multiple polyominoes are disjoint.

In this paper, we first develop a different algorithm for single-coloured hv-convex polyomino reconstruction, whose run-time matches that of Chrobak and Dürr's algorithm. We then show that it can be generalized to multiple colours easily, yielding a run-time of $O(C^2 \min\{m, n\}^{2C}mn)$.

2.1. Single-colour reconstruction

We first explore the single-colour hv-convex polyomino reconstruction. So assume that we are given vectors (h_i) and (v_j) and we want to find a binary matrix $(x_{i,j})$ that is an hv-convex polyomino and has row-sums (h_i) and column-sums (v_j). Note that necessarily $\sum_i h_i = \sum_j v_j$, since otherwise no solution can exist. We also assume that $h_i > 0$ and $v_j > 0$ for all i and j. As before, we say that cell (i, j) is *black* if and only if $x_{i,j} = 1$ and *white* otherwise.

A *foot* of a polyomino is a place where the polyomino intersects the leftmost/rightmost column or top/bottom row, and the four feet are named after the meridian directions. For our algorithm, assume that $m \leq n$ after possible rotation. Try every possible *west foot* and *east foot position*, i.e., all possible indices $w, e \in \{1, \dots, m\}$. We say that a polyomino *respects these feet* if cells $(w, 1)$ and (e, n) are black.[1] We now show how to find a polyomino that respects these feet, if one exists, in $O(nm)$ time using a 2-SAT formulation. We give the algorithm only for the case where $w \leq e$; the other case is similar.

Barcucci et al. [1] used as one of their main ingredients that in any row between w and e, they can find cells that are guaranteed to be in any polyomino respecting the feet. More precisely, let S_{WE} be the set of all cells (i, j) with $w \leq i \leq e$, $\sum_{k=1}^{j} v_k \geq \sum_{k=1}^{i-1} h_k$, and $\sum_{k=1}^{i} h_k \geq \sum_{k=1}^{j-1} v_k$. Set S_W to be the first h_w cells in row w and let S_E to be the last h_e cells in row e. Exactly as in [1], one can show that any cell in $S_W \cup S_E \cup S_{WE}$ must be black in any polyomino that respects the west foot and east foot. Furthermore, the cells in $S_W \cup S_E \cup S_{WE}$ form a polyomino that contains at least one cell in every column.

Now define a 2-SAT instance. We have two variables $x_{i,j}$ and $R_{i,j}$ for every cell, where $x_{i,j} =$ TRUE means that cell (i, j) is black, and $R_{i,j} =$ TRUE means that cell (i, j) is to the right of the polyomino, i.e., it and all cells to its right are white.[2] For every column j, let α_j be such that (α_j, j) is in $S_{WE} \cup S_W \cup S_E$; recall that at least one such cell must exist for any j, and it must be black in any polyomino respecting the feet.

Add the following clauses for all $j = 1, \dots, n$[3]:

$$\begin{array}{ll} x_{i,j} = \text{TRUE} & \text{for } i = \alpha_j \\ x_{i,j} = \text{FALSE} & \text{for } i \notin [\alpha_j - v_j + 1, \alpha_j + v_j - 1] \\ x_{i,j} \Leftrightarrow \overline{x_{i+v_j,j}} & \text{for } i \in [\alpha_j - v_j + 1, \alpha_j - 1] \\ x_{i,j} \Rightarrow x_{i+1,j} & \text{for } i \in [\alpha_j - v_j, \alpha_j - 2] \\ x_{i,j} \Rightarrow x_{i-1,j} & \text{for } i \in [\alpha_j + 2, \alpha_j + v_j]. \end{array}$$

One can easily verify that these clauses ensure that $x_{i,j}$ is true for exactly v_j cells in column j, and these cells are contiguous. For each i and j, also add the clauses

$$R_{i,j} \Rightarrow R_{i,j+1} \quad \text{and} \quad R_{i,j} \Rightarrow \overline{x_{i,j}} \quad \text{and} \quad x_{i,j} \Rightarrow R_{i,j+h_i},$$

[1] The term "feet" comes from [1]; a similar concept was the "anchor" used in [2]. Cell $(w, 1)$ is an arbitrary black cell in column 1 (i.e., it need not be the topmost one), so one polyomino may respect multiple foot positions.

[2] This is loosely inspired by the variables for white corner regions used by Chrobak and Dürr [2].

[3] To ease notation we allow indices of variables to be outside $[1 \dots m] \times [1 \dots n]$; any clause containing such variables should be omitted.

which ensures that $R_{i,j}$ does indeed describe the white region to the right of the polyomino in row i, and row i contains at most h_i cells for which $x_{i,j}$ is true. Furthermore, if row i contains exactly h_i cells for which $x_{i,j}$ is true, then these cells must be contiguous.

If this 2-SAT instance has a solution, then define a cell to be black if and only if $x_{i,j}$ is true. Then the total number of cells that are black is exactly $\sum_j v_j$ and at most $\sum_i h_i$, which implies equality since $\sum_i h_i = \sum_j v_j$. So each column j contains v_j black cells (and they are contiguous), and each row i contains h_i black cells (and they are contiguous). Since $S_W \cup S_E \cup S_{WE}$ was connected and every column is contiguous, the resulting polyomino is also connected, and hence the desired reconstruction.

Computing the set S_{WE}, building the 2-SAT instance, and solving it can be done in $O(mn)$ time. Trying this for the $O(\min\{m, n\}^2)$ possible foot configurations yields the answer to the reconstruction problem in $O(\min\{m, n\}^2 mn)$ time, which matches the run-time achieved by Chrobak and Dürr [2].

2.2. Fast multi-coloured reconstruction

Our single-colour algorithm easily generalizes to multiple colours. For each colour c, let j^c_ℓ, j^c_r be the leftmost/rightmost column j with $v^c_j > 0$. Choose an east and west foot for the polyomino of colour c in columns j^c_ℓ and j^c_r. Then build the 2-SAT instance for colour c, using variables $x^c_{i,j}$ and $R^c_{i,j}$, but using only the range where i, j satisfy $h^c_i > 0$ and $v^c_j > 0$. Outside the range, we still have variables $x^c_{i,j}$ and $R^c_{i,j}$ (because these will be needed in other constraints), but do not allow the polyomino to expand into this region in the obvious way: add the constraint $x^c_{i,j} = \text{FALSE}$ whenever $h^c_i = 0$ or $v^c_j = 0$, and add $R^c_{i,j} = \text{TRUE}$ whenever $j > j^c_r$.

Combine all these 2-SAT instances for all C colours into one 2-SAT instance with $O(Cnm)$ variables and $O(Cnm)$ clauses. To ensure that no two colours use the same cell, add the exclusion clauses

$$x^{c_1}_{i,j} \Rightarrow \overline{x^{c_2}_{i,j}} \quad \text{for all colours } c_1 \neq c_2 \text{ and each cell } (i, j).$$

This adds $O(C^2mn)$ clauses. Hence the time needed to build and solve the 2-SAT instance for one fixed set of foot configurations is $O(C^2mn)$.

Each colour has at most m^2 possible foot configurations (assuming $m \leq n$ after possible rotation), so the total number of combinations of foot configurations is at most $\min\{m, n\}^{2C}$. Solving the 2-SAT instance for each of them hence computes disjoint hv-convex polyominoes, if they exist, in $O(\min\{m, n\}^{2C} mnC^2)$ time.

Theorem 1. *The C-colour reconstruction problem for hv-convex polyominoes can be solved in time $O(\min\{m, n\}^{2C} mnC^2)$.*

2.3. Variants

The variables $R_{i,j}$ for the "white region to the right" turn out to be useful for two variants of the C-colour reconstruction problem. The first variant concerns ordered reconstruction. As defined, a C-colour reconstruction problem gives the row/column-sums as an unordered set (one for each colour). A variant, typically used in the Pic-a-pix game, would be to give an ordered set, i.e., all cells of colour c_1 must be to the left of all cells of colour c_2, etc.

We can solve this variant easily. Whenever colour c_1 must be left of colour c_2 in row i, add the clause $x^{c_2}_{i,j} \Rightarrow R^{c_1}_{i,j}$ for all j. So if the cell has colour c_2, then it must be in the right region with respect to colour c_1, so all cells of colour c_1 must be to its left as desired. Similarly we can add clauses for columns, after defining another variable (say $B^c_{i,j}$) that expresses that a cell is below all cells of colour c.

If the order of colours is total, and we add this clause only for consecutive colours, then we can even drop the exclusion clauses (which are then always satisfied), which reduces the run-time of our algorithm to $O(\min\{m, n\}^{2C} mnC)$.

A second variant concerns the shape of the union of all coloured polyominoes. The current setup does not put any restriction on this shape. But if we wanted the union to be h-convex, we could simply add the clause $x^{c_1}_{i,j} \Rightarrow R^{c_2}_{i,j+h_i}$, for any i, j, c_1, c_2, where $h_i = \sum_c h^c_i$. Row i then can contain at most h_i cells of any colour, which ensures that coloured cells are contiguous. Similarly (again after adding variables $B^c_{i,j}$) we can ensure v-convexity of the union.

3. NP-hardness results

The previous section gave a polynomial algorithm for multi-colour reconstruction for a constant number C of colours. The dependency on C is exponential, and thus it is no surprise that if C is not constant, the problem becomes NP-complete. We prove this now.

Surely C-colour reconstruction is in NP. To see that it is NP-hard, we use a reduction from 3-SAT. We use three gadgets: a transmitter gadget, a splitter gadget, and a crosser gadget (see Fig. 1.) Each occurrence of a gadget has its own unique colour (and three colours for a splitter gadget).

Assume we are given an instance of 3-SAT with variables $x_1, \ldots, x_N$ and clauses $c_1, \ldots, c_M$, where each clause contains exactly three literals. Create an instance with $6N + 8$ rows and $21M + 2$ columns. There are six rows reserved for each variable (specifically, counting rows from top to bottom, rows $6i - 5, \ldots, 6i$ are dedicated to variable x_i for $i = 1, \ldots, N$).

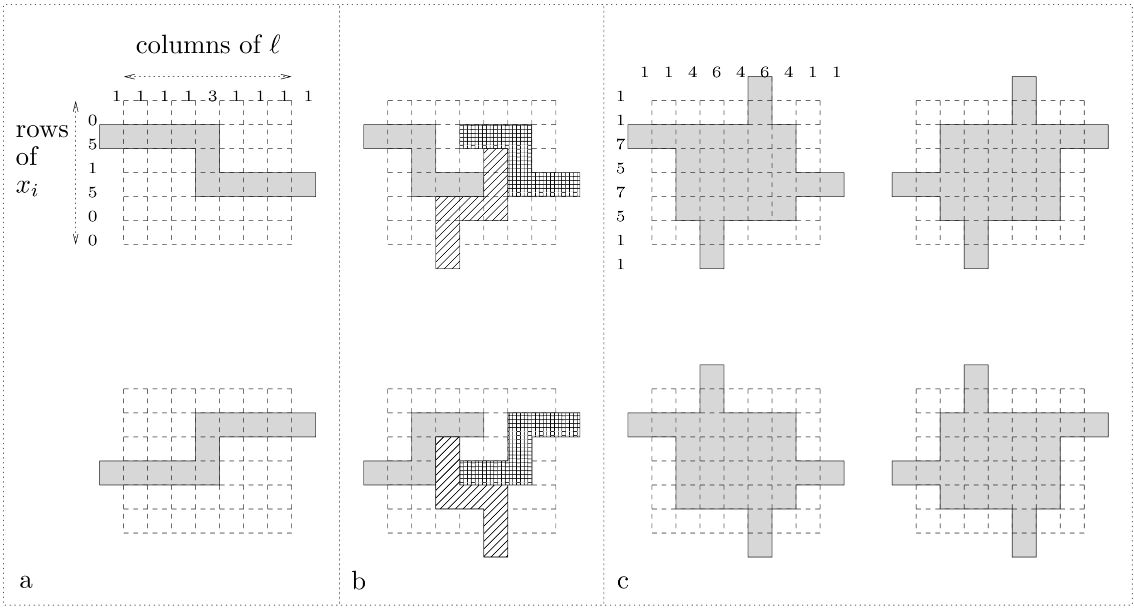

Fig. 1. (a) The transmitter gadget in two realizations, (b) the splitter gadget in two realizations, and (c) the crosser gadget in four realizations. We omit the row/column-sums for the splitter gadget to avoid confusion among the three colours.

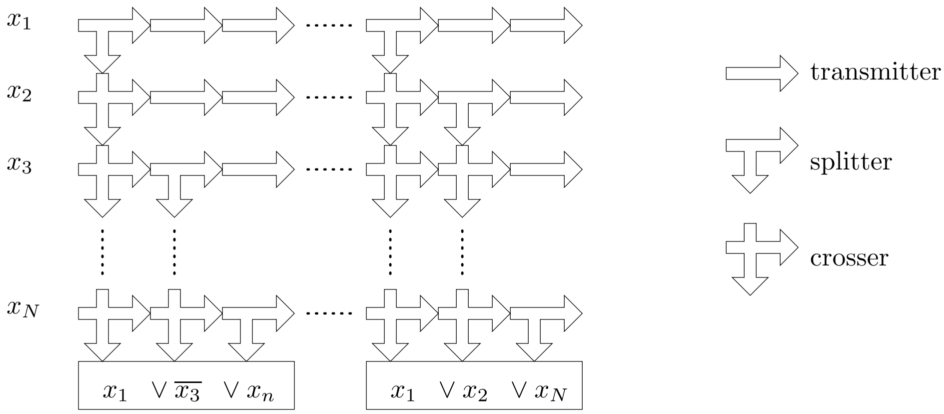

Fig. 2. The overall layout.

The last eight rows (i.e., rows $6N + 1, \ldots, 6N + 8$) are reserved for the clause gadgets that we will describe later. Columns 1 and $21M + 2$ are boundary columns needed for overhang from adjacent gadgets, and seven columns are dedicated to each literal of a clause (specifically, the kth literal of clause c_j uses columns $21j + 7k - 26$ through $21j - 7k + 20$, for $k = 1, 2, 3$ and $j = 1, \ldots, M$).

At the place where the rows for variable x_i meet the columns of literal ℓ, we place a splitter gadget if ℓ uses x_i (i.e., if $\ell = x_i$ or $\ell = \overline{x_i}$), a crosser gadget if ℓ uses some x_h with $h < i$, and a transmitter gadget otherwise. See Fig. 2.

Each gadget has some "legs" sticking out, i.e., places where it enters the region of adjacent gadgets (or the boundary columns; note that there are no crosser gadgets in the top rows). Using hv-convexity, one can show that a transmitter gadget and a splitter gadget have exactly two possible realizations each: either the upper leg (in row 2) sticks out to the left and the lower leg (in row 4) sticks out to the right, or vice versa. For the splitter gadget, this also determines whether the leg to the bottom sticks out in column 3 or column 5. A crosser gadget has four legs sticking out (in rows 2 and 4 and columns 3 and 5), and exactly four possible realizations. See Fig. 1.

Note that all gadgets within one row must either all have their upper legs to the left or all have them to the right; otherwise they would overlap. Given a realization of this coloured polyomino instance, set variable x_i to be true if all upper legs in gadgets in the rows assigned to x_i stick out to the left. If $\ell_j = x_i$ or $\ell_j = \overline{x_i}$, then this also means that in the columns of ℓ_j, the leg of the splitter gadget (in the rows of x_i) sticks out to the bottom in column 3, and this transmits along all crosser gadgets below the rows of x_i.

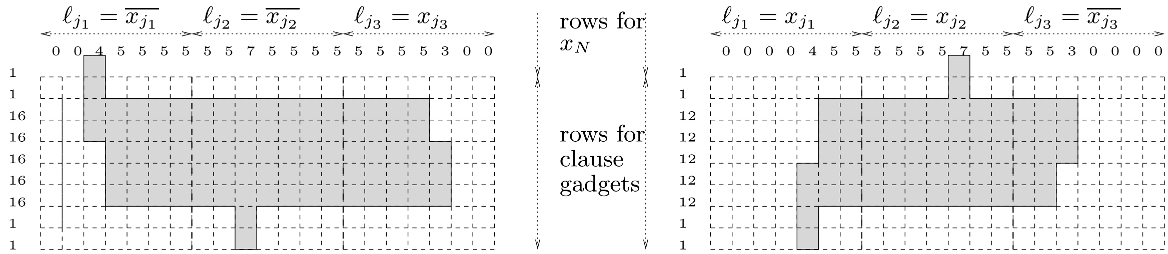

Fig. 3. Two of the eight possible clause gadgets.

Finally we define eight different clause gadgets (depending on the negation status of the literals in the clause), and place for each clause the corresponding gadget in the columns of the clause, below all other gadgets. Fig. 3 shows the clause gadget for $\overline{x_{j_1}} \vee \overline{x_{j_2}} \vee x_{j_3}$ and $x_{j_1} \vee x_{j_2} \vee \overline{x_{j_3}}$. Generally, the column-sum vector for this gadget is

$$(\underbrace{0\dots0}_{a_1}\,4\,\underbrace{5\dots5}_{6-a_1}\,\underbrace{5\dots5}_{a_2}\,7\,\underbrace{5\dots5}_{6-a_2}\,\underbrace{5\dots5}_{a_3}\,3\,\underbrace{0\dots0}_{6-a_3}),$$

where $a_k = 4$ if the kth literal of the clause is positive and $a_k = 2$ otherwise. The row-sum vector is $(1, 1, W, W, W, W, W, 1, 1)$, where $W = 14 + a_3 - a_1$.

The clause gadget has $W + 1$ non-zero columns and five rows with density W; by hv-convexity this implies that all non-zero columns except the first and last have black cells in these rows. Thus any column with density 5 has all its black cells in the rows with density W. But there must be some column that has a black cell in the top two rows (creating another leg that sticks out to the top by hv-convexity). This leg hence must occur in one of the *special columns*, which are the columns with density 4, 7, or 3. On the other hand, for each of these three special columns, there does indeed exist a reconstruction of the clause gadget where the leg is in this column; Fig. 3 shows two of them.

The clause gadget has been configured such that the special columns coincide exactly with the column where the gadget in the rows above has a leg sticking out if this literal is false. So if the coloured polyomino instance can be realized, then to avoid overlap for every clause there must be at least one true literal, so 3-SAT has a solution. Similarly one shows that any 3-SAT solution gives a realization, which finishes the reduction.

Clearly the reduction is polynomial: it uses $O(N)$ rows, $O(M)$ columns, and $O(NM)$ colours. More precisely, note that we have one splitter gadget for each of the $3M$ literals, and hence $3NM - 3M$ gadgets that are transmitter or crosser gadgets. We also have M clause gadgets, so the total number of colours is $3 \cdot 3M + (3NM - 3M) + M = 3NM + 7M$.

Theorem 2. *C-colour hv-convex reconstruction is NP-complete if C is part of the input.*

4. Conclusion and open problems

In this note, we showed that reconstruction of multiple disjoint hv-convex polyominoes from their projections is polynomial if the number of colours is constant, but NP-hard if the number of colours is unbounded.

The main remaining open problem is to determine the dependency on the number of colours C. Is it possible to separate the exponentiality in C from the size of the grid; in other words, is the problem fixed-parameter tractable in the number of colours? (See for example Niedermeier [8] for an introduction to fixed-parameter tractability.) We suspect that this is the case, but have not been able to prove it; the main obstacle is that foot positions of different colours need not be in the same column, and we have not been able to bound the number of relevant foot positions in terms of C only.

For discrete tomography applications, it is vital not only to construct some polyomino that satisfies the constraints, but also to establish whether the answer is unique. Note that our algorithm does not detect uniqueness. Is it NP-hard, given one hv-convex solution, to test whether there exists another? To our knowledge this is open even in the single-colour case.

Finally, are there faster algorithms for single-coloured hv-convex polyomino reconstruction? And can they be generalized to multiple colours?

Acknowledgement

This research was supported by NSERC.

References

[1] E. Barcucci, A. Del Lungo, M. Nivat, R. Pinzani, Medians of polyominoes: a property for the reconstruction, International Journal of Imaging Systems and Technology 9 (2–3) (1998) 69–77.

[2] M. Chrobak, C. Dürr, Reconstructing hv-convex polyominoes from orthogonal projections, Information Processing Letters 69 (1999) 283–289.

[3] M. Chrobak, C. Dürr, Reconstructing polyatomic structures from discrete x-rays: NP-completeness proof for three atoms, Theoretical Computer Science 259 (1–2) (2001) 81–96.

[4] C. Dürr, F. Guiñez, M. Matamala, Reconstructing 3-colored grids from horizontal and vertical projections is NP-hard, in: Amos Fiat, Peter Sanders (Eds.), European Symposium on Algorithms, ESA 09, in: Lecture Notes in Computer Science, vol. 5757, Springer, 2009, pp. 776–787.
[5] G.T. Herman, A. Kuba (Eds.), Discrete Tomography: Foundations, Algorithms, and Applications, Birkhäuser, 1999.
[6] G.T. Herman, A. Kuba (Eds.), Advances in Discrete Tomography and Its Applications, Birkhäuser, 2007.
[7] C. Kisielowski, P. Schwander, F.H. Baumann, M. Seibt, Y. Kim, A. Ourmazd, An approach to quantitative high-resolution transmission electron microscopy of crystalline materials, Ultramicroscopy 58 (2) (1995) 131–155.
[8] Rolf Niedermeier, Invitation to Fixed-Parameter Algorithms, Oxford University Press, 2006.
[9] P. Schwander, C. Kisielowski, M. Seibt, F.H. Baumann, Y. Kim, A. Ourmazd, Mapping projected potential, interfacial roughness, and composition in general crystalline solids by quantitative transmission electron microscopy, Physical Review Letters 71 (25) (1993) 4150–4153.