On Polyatomic Tomography over Abelian Groups: Some Remarks on Consistency, Tree Packings and Complexity

Peter Gritzmann[1] · Barbara Langfeld[2]

Abstract

The paper deals with an inverse problem of reconstructing matrices from their marginal sums. More precisely, we are interested in the existence of $r \times s$ matrices for which only the following information is available: The entries belong to known subsets of c distinguishable abelian groups, and the row and column sums of all entries from each group are given. This generalizes Ryser's classical problem of characterizing the set of all 0–1-matrices with given row and column sums and is a basic problem in (polyatomic) discrete tomography. We show that the problem is closely related to packings of trees in bipartite graphs, prove consistency results, give algorithms and determine its complexity. In particular, we find a somewhat unusual complexity behavior: the problem is hard for "small" but easy for "large" matrices.

Keywords Discrete inverse problem · Discrete tomography · Polyatomic tomography · Polynomial-time algorithm · NP-completeness

Mathematics Subject Classification Primary: 05 · 68R · 68Q17 · 82D · 90C · 49N45

1 Introduction

The present paper deals with the inverse problem of reconstructing matrices from their marginal sums. The task of understanding the combinatorial structure of all 0–1-

Peter Gritzmann
gritzmann@tum.de

Barbara Langfeld
langfeld@math.uni-kiel.de

[1] Zentrum Mathematik, Technische Universität München, 85747 Garching bei München, Germany

[2] Mathematisches Seminar, Christian-Albrechts-Universität zu Kiel, 24118 Kiel, Germany

Springer

matrices with given row and column sums was addressed by Ryser [17] already in 1960; see also [3]. It is closely related to degree sequences of graphs, and—by identifying the 1-entries of such a matrix with point sets in $\mathbb{R}^2$ or $\mathbb{Z}^2$—can, in retrospective, also be viewed as an anticipation of the field of *discrete tomography*. Since then, a rich theory has been developed which is centered around the task of determining (theoretically and algorithmically) discrete structures by information about their interaction with certain query sets. For surveys see [1,10,12], and other sources quoted there. More than half a century after Ryser's work discrete tomography is a well-established field with seminal results reaching into combinatorics, geometry, geometry of numbers, commutative algebra, optimization and other fields.

Its many practical applications in materials science, physics, and various other areas (see e.g. [1]) have fueled studies in dimensions higher than 2, for X-rays in more than two directions, for other query sets—most notably hyperplanes, leading to the *discrete Radon transform*—with other image domains (modelling e.g. grey values in images) and for objects which are composed of different types of elements, the *polyatomic* or *colored* case.

As Ryser [17] showed, the classical binary monoatomic case for 2 directions can be solved easily and in polynomial time. The problem becomes $\mathbb{NP}$-hard, however, when more than two directions are involved [8], see also [4]. The corresponding *integer case* where the entries of the unknown matrix are in $\mathbb{Z}$ can be modeled via systems of linear Diophantine equations and can therefore also be solved in polynomial time for any fixed number of directions.

The situation changes if more than one class of atoms (often called colors) is present. Then, the binary case becomes $\mathbb{NP}$-hard even when X-ray information about the unknown matrix is given only for the rows and columns, i.e., in the two coordinate directions [6].

As we will show, polyatomic tomography over $\mathbb{Z}$ depicts a different behavior, being $\mathbb{NP}$-hard for matrices with a small number of rows (or columns) but easy for matrices of larger size. It will turn out that the problem is intimately related to packings of trees in (bipartite) graphs, whose study was initiated by Tutte [20] and Nash-Williams [15]. This connection will be used to determine (even in the general case, i.e., for matrix entries restricted to distinguishable but otherwise arbitrary abelian groups) when the reconstruction problem from row and column sums can be solved for a given number of colors. Let us further remark that the problem we are considering can also be viewed as a special variant of the multi-commodity flow problem which therefore also depicts the same (possibly unexpected) complexity behavior.

The present paper is organized as follows. In Sect. 2 we will introduce the problems formally and state the main results. Section 3 will show how the polyatomic problem relates to tree packings, prove our main existence theorem, and the efficiency part of Theorem 2.2. In Sect. 4 we will deal with the computational complexity of the problem and prove the hardness part of Theorem 2.2. The final Sect. 5 will provide an interpretation as a multi-commodity flow problem, make some concluding remarks and state some open problems.

2 Notation and Main Results

We will now introduce the relevant notion in great enough generality to be able to formulate our main results and also place them into the perspective of, partly classical, known results.

Throughout the paper $\mathbb{N}$, $\mathbb{Z}$ and $\mathbb{R}$ denote the sets of positive integers, integers and reals, respectively, and let $\mathbb{N}_0 = \mathbb{N} \cup \{0\}$ and $[n] = \{j \in \mathbb{N} : j \leq n\}$ for each $n \in \mathbb{N}$.

Further, let $c, r, s \in \mathbb{N}$, and let $\mathbb{G}_1, \ldots, \mathbb{G}_c$ be abelian groups (written additively). (The groups need not be different but are to be distinguished, i.e., should be thought of as a pair consisting of an abelian group and an index from $[c]$, sometimes referred to as its *color*.) In the following, we are interested in (colored) $r \times s$ matrices $M = \left(\mu_{ij}^{(\ell_{ij})}\right)_{i \in [r], j \in [s]}$ with entries $\mu_{ij}^{(\ell_{ij})} \in \mathbb{G}_{\ell_{ij}}$ for some $\ell_{ij} \in [c]$ with prescribed row and column sums for each of the groups $\mathbb{G}_1, \ldots, \mathbb{G}_c$.

More precisely, suppose that, for $\ell \in [c]$, we are given $R^{(\ell)} = \left(\rho_1^{(\ell)}, \ldots, \rho_r^{(\ell)}\right) \in \mathbb{G}_\ell^r$, and $S^{(\ell)} = \left(\sigma_1^{(\ell)}, \ldots, \sigma_s^{(\ell)}\right) \in \mathbb{G}_\ell^s$. Then we want to find a matrix $M = \left(\mu_{ij}^{(\ell_{ij})}\right)$ such that

$$
\begin{aligned}
\sum_{\substack{j=1 \\ \ell_{ij}=\ell}}^{s} \mu_{ij}^{(\ell_{ij})} &= \rho_i^{(\ell)} \quad \text{for } i \in [r], \quad \ell \in [c] \quad \text{and} \\
\sum_{\substack{i=1 \\ \ell_{ij}=\ell}}^{r} \mu_{ij}^{(\ell_{ij})} &= \sigma_j^{(\ell)} \quad \text{for } j \in [s], \quad \ell \in [c].
\end{aligned}
\tag{2.1}
$$

Let us point out again that each element $\mu_{ij}^{(\ell_{ij})}$ of the matrix M carries the information which of the c groups it belongs to—specified by (its color) ℓ_{ij}. As usual, a sum over the empty set is, by convention, the neutral element of the underlying group.

Clearly, a necessary condition for such a matrix M to exist is that each $(R^{(\ell)}, S^{(\ell)})$ is *balanced*, i.e.,

$$
\sum_{i=1}^{r} \rho_i^{(\ell)} = \sum_{j=1}^{s} \sigma_j^{(\ell)} \quad \text{for } \ell \in [c]. \tag{2.2}
$$

Using the notation

$$
\mathcal{R} = (R^{(1)}, \ldots, R^{(c)}), \quad \mathcal{S} = (S^{(1)}, \ldots, S^{(c)}), \quad \mathscr{G} = (\mathbb{G}_1, \ldots, \mathbb{G}_c)
$$

we say that our row and column sum data $(\mathcal{R}, \mathcal{S})$ is *balanced* (for $\mathscr{G}$) if (2.2) holds for each $\ell \in [c]$. Given a balanced pair $(\mathcal{R}, \mathcal{S})$, let $\mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ denote the set of all colored matrices in $\left(\bigcup_{\ell \in [c]} \mathbb{G}_\ell\right)^{r \times s}$ satisfying (2.1).

In the following we will focus on the question when $\mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ is nonempty. In some cases, the elements of $M \in \mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ will be further restricted to subsets $W^{(\ell)} \subset \mathbb{G}_\ell$ for each $\ell \in [c]$. Setting $\mathscr{W} = (W^{(1)}, \ldots, W^{(c)})$, let for given $c, \mathscr{G}, \mathscr{W}$ the subset

of $\mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ of all matrices $\left(\mu_{ij}^{(\ell_{ij})}\right)_{i \in [r], j \in [s]}$ whose entries are restricted to $\mathscr{W}$, i.e., $\mu_{ij}^{(\ell_{ij})} \in W^{(\ell_{ij})}$ for each $(i, j) \in [r] \times [s]$, be denoted by $\mathfrak{A}_{\mathscr{G}, \mathscr{W}}^c(\mathcal{R}, \mathcal{S})$. Then we are dealing with the problem

COLRYS$_c(\mathscr{G}, \mathscr{W})$

Given r, s and corresponding balanced $(\mathcal{R}, \mathcal{S})$, does there exist $M \in \mathfrak{A}_{\mathscr{G}, \mathscr{W}}^c(\mathcal{R}, \mathcal{S})$?

Let us remark that—as the notation indicates—c, $\mathscr{G}$ and $\mathscr{W}$ are (arbitrary but) fixed whenever we refer to the problem COLRYS$_c(\mathscr{G}, \mathscr{W})$, while the dimensions r and s and the parameters in $(\mathcal{R}, \mathcal{S})$ specify the different instances. Note that we may assume without loss of generality that $2 \leq r \leq s$.

The name COLRYS$_c(\mathscr{G}, \mathscr{W})$ is chosen in reverence for Ryser, as it describes a colored version of his original "monoatomic" problem studied in [17]. In fact, for $c = 1$, $\mathscr{G} = (\mathbb{Z})$, $\mathscr{W} = (\{0, 1\})$, COLRYS$_c(\mathscr{G}, \mathscr{W})$ asks for a 0–1 matrix with given row and column sums (computed in $\mathbb{Z}$). As it is well known, this and the corresponding reconstruction problem to compute such a matrix can be solved in polynomial time, [17]; see also [3] or [12]. Let us mention that the monoatomic case has also been studied over rings other than $\mathbb{Z}$; see [2] and the papers quoted there.

Further, for $c = 1$, $\mathscr{G} = (\mathbb{Z})$, $\mathscr{W} = (\mathbb{Z})$, each instance of COLRYS$_c(\mathscr{G}, \mathscr{W})$ gives rise to a system of linear Diophantine equations and can therefore be solved efficiently. Even if X-ray information in more than two directions is given, the problem can be formulated as a system of linear Diophantine equations and still be solved in polynomial time; see e.g. [14, Sect. 7].

On the other hand, when $\mathbb{G}_\ell = \mathbb{Z}$ and $W_\ell = \{0, 1\}$ for all $\ell \in [c]$, COLRYS$_c(\mathscr{G}, \mathscr{W})$ constitutes the "classical" polyatomic case where a planar hybrid material is only accessible by X-ray information (at atomic scale) in the standard coordinate directions for each of the c types of different materials. Let us stress the fact that the positions of the colors, i.e., the assignments $(i, j) \mapsto \ell_{ij}$, are by no means fixed "externally" but are part of the solution. Therefore COLRYS$_c(\mathscr{G}, \mathscr{W})$ does not simply decompose into c monoatomic problems with restricted support which can be solved independently. In fact, as it is well known, COLRYS$_c(\mathscr{G}, \mathscr{W})$ is $\mathbb{NP}$-complete, for every $c \geq 2$, [6]. (Note that [6] regards the neutral elements of all groups together as an additional color.)

The present paper addresses existence issues and the question of efficient computability for the general problem COLRYS$_c(\mathscr{G}, \mathscr{W})$. In particular, we give the following existence theorem for general $\mathscr{G}$.

Theorem 2.1 *Let $c, r, s \in \mathbb{N}$ with $2 \leq r \leq s$ and*

$$c \leq \left\lfloor \frac{rs}{r + s - 1} \right\rfloor.$$

Then, for any corresponding pair $(\mathcal{R}, \mathcal{S})$ which is balanced over $\mathscr{G}$, there exists a matrix $M \in \mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$.

Note that the condition in Theorem 2.1 is, in particular, satisfied when

$$r \geq c + 1 \quad \text{and} \quad s \geq (r - 1)^2$$

or

$$r \geq 2c.$$

This means that the answer to any "large" instance of $\text{COLRYS}_c(\mathscr{G}, \mathscr{G})$ is always "yes". The proof of Theorem 2.1 will be given in Sect. 3. There, we will also discuss natural conditions under which the given condition is sharp; see Theorem 3.8.

The proof of Theorem 2.1 utilizes packings of spanning trees in complete bipartite graphs and will even show that a matrix in $\mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ can actually be computed efficiently, whenever the group operations are algorithmically accessible. Since we do not want to overload the paper with technical details, we will formulate our algorithmic results only for the *integer case*, i.e., for $\mathscr{G} = \mathscr{Z} = (\mathbb{Z}, \ldots, \mathbb{Z})$ where we can naturally employ the standard *binary Turing machine model*.

The following results deal with the complexity of $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ and $\text{COLRYS}_c (\mathscr{Z}, \mathscr{N}_0)$ where $\mathscr{N}_0 = (\mathbb{N}_0, \ldots, \mathbb{N}_0)$. While $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ is in $\mathbb{P}$ for $r \geq c + 1$ it turns out, in particular, that this problem is $\mathbb{NP}$-hard in general.

Theorem 2.2 *Let $c \in \mathbb{N}$. When all instances are restricted to those for which $c + 1 \leq r \leq s$, then $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ can be solved in polynomial time.*

For any $c \in \mathbb{N}\backslash\{1\}$, $\text{COLRYS}_c(\mathscr{Z}, \mathscr{W})$ is $\mathbb{NP}$-complete both for $\mathscr{W} = \mathscr{Z}$ and $\mathscr{W} = \mathscr{N}_0$. The $\mathbb{NP}$-hardness persists even if all instances are restricted to those where the number of rows is fixed to some $r^* \in \{2, \ldots, c\}$. Moreover, when $c \in \mathbb{N}\backslash\{1\}$, $\text{COLRYS}_c(\mathscr{Z}, \mathscr{N}_0)$ is $\mathbb{NP}$-complete for any fixed $r^* \geq c + 1$.

The first part of Theorem 2.2 will follow from the (more general) proof given in Sect. 3 while the hardness result will be proved in Sect. 4.

Let us remark that the dramatic drop in complexity of $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ from $r^* \leq c$ to $r^* \geq c + 1$ may be phrased intuitively as "Small instances are hard while large instances are easy." Of course, in the hardness statement only r^* is fixed but s is still part of the input. Anyway it is more common that problems do not get easier if previously fixed parameters become part of the input.

3 Tomography and Spanning Trees

In this section we will first consider the monoatomic case, i.e., $c = 1$, for which we can use a simplified notation. Let $\mathbb{G}$ be an abelian group, $0_{\mathbb{G}}$ its neutral element, $\mathscr{G} = (\mathbb{G})$, $r, s \in \mathbb{N}$ with $2 \leq r \leq s$, $R = (\rho_1, \ldots, \rho_r) \in \mathbb{G}^r$, $S = (\sigma_1, \ldots, \sigma_s) \in \mathbb{G}^s$, $\mathcal{R} = (R)$, and $\mathcal{S} = (S)$. As before, we assume that $(\mathcal{R}, \mathcal{S})$ is balanced, which simply means that (R, S) is balanced. Further, $\mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ will be denoted by $\mathfrak{A}_{\mathbb{G}}(R, S)$.

We show how the existence of a matrix $M \in \mathfrak{A}_{\mathbb{G}}(R, S)$ is related to the existence of spanning trees in complete bipartite graphs. As it is standard fare, (R, S) is naturally associated with a vertex weighted complete bipartite graph as follows.

Let $A = \{a_1, \ldots, a_r\}$, $B = \{b_1, \ldots, b_s\}$ denote the bipartition of the vertex set of $K_{r,s}$, and regard, for $i \in [r]$ and $j \in [s]$, the values ρ_i, σ_j as weights associated with the vertices a_i and b_j, respectively. We speak of $K_{r,s}$ as (R, S)-weighted.

Now, for $i \in [r]$ and $j \in [s]$, each edge e_{ij} of $K_{r,s}$ corresponds to the position (i, j) of the desired matrix M. Hence we are interested in assigning weights μ_{ij} to the edges whose row and column sums add up to R and S, respectively. Note that there is some connection to the 1–2–3-conjecture; see e.g. [19] and the papers quoted there.

Actually, for later applications to the polyatomic case, we are interested in sparsity, i.e., in assigning $0_{\mathbb{G}}$ to as many edges of $K_{r,s}$ as possible. Let $G = (V, E)$ be a spanning subgraph of $K_{r,s}$, i.e., $V = A \cup B$. The graph G naturally inherits the vertex weights (R, S), and is called *compensable* if there exist edge weights $\omega_{ij} \in \mathbb{G}$ such that

$$\sum_{\substack{j \in [s] \\ e_{ij} \in E}} \omega_{ij} = \rho_i, \quad i \in [r], \tag{3.1}$$

$$\sum_{\substack{i \in [r] \\ e_{ij} \in E}} \omega_{ij} = \sigma_j, \quad j \in [s]. \tag{3.2}$$

Edge weights satisfying (3.1) and (3.2) will be called *compensating*.

Recall that, by our general assumption, (R, S) is balanced, hence the sum of the weights of the vertices in A coincides with that of B, i.e., the trivial condition for compensability is satisfied. The following observation is obvious, but crucial.

Remark 3.1 Let $G = (V, E)$ be a spanning subgraph of the (R, S)-weighted complete bipartite graph $K_{r,s}$. Then edge weights $\omega_{ij} \in \mathbb{G}$ for $e_{ij} \in E$ are compensating if and only if the matrix $M = (\mu_{ij})$ with entries

$$\mu_{ij} = \begin{cases} \omega_{ij} & \text{if } e_{ij} \in E, \\ 0_{\mathbb{G}} & \text{else,} \end{cases}$$

for $i \in [r]$ and $j \in [s]$ is in $\mathfrak{A}_{\mathbb{G}}(R, S)$.

Next we show that actually each arbitrary spanning tree leads to a matrix in $\mathfrak{A}_{\mathbb{G}}(R, S)$.

Theorem 3.2 *Let $K_{r,s}$ be (R, S)-weighted. Then any spanning tree T in $K_{r,s}$ is compensable.*

Proof We select a vertex of T as root, say, without loss of generality, a_1, and arrange the other vertices of T in layers $L_0, \ldots, L_q$ according to their edge distance to a_1. Since $K_{r,s}$ is bipartite, the vertices of each layer belong either all to A or all to B and consecutive layers contain vertices of different sets of this bipartition. Of course, $L_0 = \{a_1\}$.

Note that for each $p \in \{1, \ldots, q\}$ and each vertex v in L_p there is a unique edge that joins v to a vertex from L_{p-1}; it will in the sequel be denoted by e_v. Further, let E_v denote the subset of the edge set E_T of T of those edges which join v to a vertex from L_{p+1}.
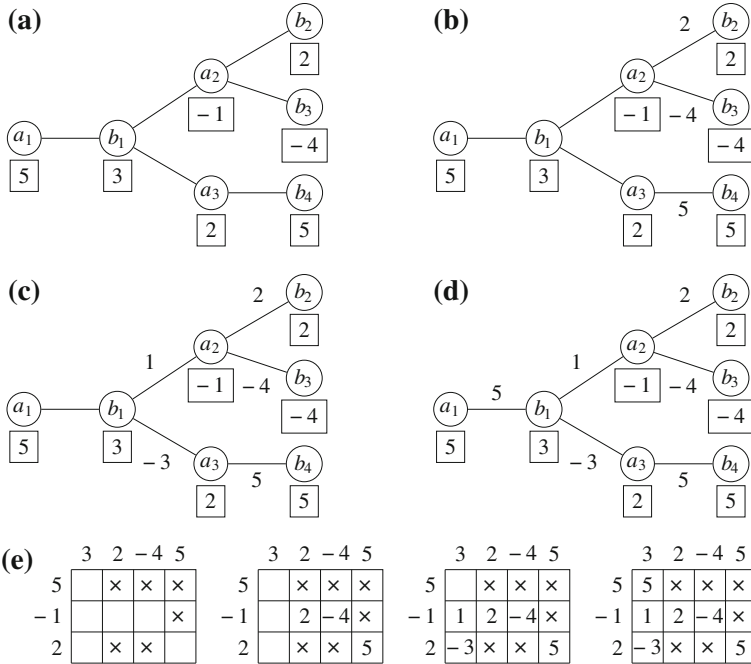
Fig. 1 (**a–d**) Illustration of the algorithm in the proof of Theorem 3.2 for $\mathbb{G} = \mathbb{Z}$. (**a**) The weights of the vertices are given below the nodes in the boxes. (**b–d**) Successively, the edges connected to the previously satisfied layer are set to their final value (unframed). (**e**) The same procedure is described as a successive construction of an integer 3×4 matrix with the prescr ibed row and column sums. Positions that are never touched are marked with "×". The other positions correspond to the edges of the tree and are updated step by step

We will now assign appropriate edge weights $\omega(e)$ for $e \in E_T$ layer by layer. For an illustration see Fig. 1.

To simplify the notation, $\tau(v)$ will denote the given weight of v, i.e., $\tau(v) = \rho_i$ for $v = a_i \in A$ and $\tau(v) = \sigma_j$ for $v = b_j \in B$. In the following, we say that the vertex v is satisfied if, for $v = a_i \in A$, the condition (3.1) and, for $v = b_j \in B$, the condition (3.2) is satisfied.

We start with the vertices of L_q. Since each such vertex v is a leaf of T we can set $\omega(e_v) = \tau(v)$, and thus satisfy v.

Now let $p \in [q-1]$, $P_p = L_{p+1} \cup \cdots \cup L_q$ and suppose, edge weights have been chosen for all edges in $\{e_v : v \in P_p\}$ in such a way that all the vertices in P_p are satisfied. For v in L_p we now choose

$$\omega(e_v) = \tau(v) - \sum_{e \in E_v} \omega(e),$$

which, consequently, satisfies v.

Equivalently, the construction can be described in terms of the weights $\tau(v)$ for v in L_p. Let T_v be the vertex set of the connected component of the subgraph of T induced by $\{v\} \cup P_p$ which contains v. Now, suppose that $v \in A$. Then

$$\omega(e_v) = \sum_{w \in T_v \cap A} \tau(w) - \sum_{w \in T_v \cap B} \tau(w). \tag{3.3}$$

For $v \in B$, the roles of A and B in (3.3) are reversed.

By the successive construction all vertices of $V \setminus \{a_1\}$ are satisfied. Moreover, since (R, S) is balanced, (3.3) yields

$$\sum_{v \in L_1} \omega(e_v) = \sum_{w \in B} \tau(w) - \sum_{w \in A \setminus \{a_1\}} \tau(w) = \sum_{j=1}^{s} \sigma_j - \sum_{i=2}^{r} \rho_i = \rho_1.$$

So a_1 is also satisfied, which completes the proof. $\qquad\square$

Note that, apart from simple computations in T, the construction only involves operations in $\mathbb{G}$. Hence, if these are algorithmically accessible, the corresponding matrix can be constructed efficiently. Since we do not want to go into additional technical details we formulate the result in the following corollary only for $\mathbb{Z}$, where we can apply the standard binary Turing machine model.

Corollary 3.3 *Let $\mathbb{G} = \mathbb{Z}$. Given an (R, S)-weighted $K_{r,s}$, and a spanning tree in $K_{r,s}$, compensating edge weights can be computed in polynomial time.*

Next we turn to the case of an arbitrary number c of colors. As Remark 3.1 and Theorem 3.2 suggest we will now aim at packing c spanning trees into $K_{r,s}$. Let us remark that the study of packings of subgraphs into graphs goes back a long way; e.g. [15,20] for spanning trees, [11] for cliques and stable sets, and [15,18] for other packing problems with a particular view towards combinatorial optimization.

A graph G admits a *packing of c spanning trees* if there exist (at least) c edge-disjoint spanning trees in G. Equivalently, we say that c spanning trees can be packed into G. Let stp(G) denote G's *spanning tree packing number*, i.e., the maximal number of spanning trees which can be packed into G. Initiated by work of Tutte [20] and Nash-Williams [15], stp(G) has been studied extensively. Of course, we are particularly interested in the spanning tree packing number for complete bipartite graphs. Note first that, as $K_{r,s}$ has rs edges while each spanning tree in $K_{r,s}$ has $r + s - 1$ edges, $\lfloor rs/(r + s - 1) \rfloor$ is a trivial upper bound for stp$(K_{r,s})$. As it turns out, this bound is tight, [16, Sect. 3.4]; see also the references cited there.

Proposition 3.4 *For $r, s \in \mathbb{N}$ with $2 \leq r \leq s$,*

$$\mathrm{stp}(K_{r,s}) = \left\lfloor \frac{rs}{r + s - 1} \right\rfloor. \tag{3.4}$$

It is now easy to complete the proof of Theorem 2.1.

Proof of Theorem 2.1 Let $r, s \in \mathbb{N}$ with $2 \leq r \leq s$ such that $c \leq \lfloor rs/(r + s - 1) \rfloor$ and let $(\mathcal{R}, \mathcal{S})$ be balanced over $\mathscr{G}$. Further, let $T^{(1)}, \ldots, T^{(c)}$ be edge-disjoint spanning trees according to Proposition 3.4. If we associate, for $\ell \in [c]$, with $T^{(\ell)}$ the vertex weights $(R^{(\ell)}, S^{(\ell)})$ then $T^{(\ell)}$ is compensable by Theorem 3.2. Finally, by Remark 3.1 any compensating edge weighting directly translates to an entry $\mu_{ij}^{(\ell)} \in \mathbb{G}_\ell$ of the desired matrix M satisfying (2.1). Note that the nonzero elements are restricted to the positions (i, j) whose corresponding edges are in $T^{(\ell)}$. Finally we fill all entries of M which do not correspond to an edge of any of the c trees with neutral elements $0_{\mathbb{G}_\ell}$ of any of the groups $\mathbb{G}_\ell$. Then $M \in \mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$. $\qquad\square$

As in the monoatomic case, the efficient computation of such a matrix $M \in \mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ relies on the appropriate algorithmic accessibility of the group operations. In addition, we need to be able to compute a maximum tree packing in $K_{r,s}$. This can be done efficiently even for general graphs; see [18, Sect. 51.5a] for a complexity survey with references.

Proposition 3.5 *For each graph G, $\mathrm{stp}(G)$ edge-disjoint spanning trees in G can be computed in polynomial time.*

With the aid of Corollary 3.3 and Proposition 3.5, the proof of Theorem 2.1 therefore yields, in particular, the following corollary.

Corollary 3.6 *When all instances are restricted to those for which $2 \leq r \leq s$ and $c \leq \lfloor rs/(r + s - 1) \rfloor$, then $\mathrm{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ can be solved in polynomial time.*

This corollary allows us to prove the efficiency part of Theorem 2.2. Recall that, by the remark after Theorem 2.1, Corollary 3.6 covers already the cases that $r \geq c + 1, s \geq (r - 1)^2$ and $r \geq 2c$. So we are left with the situation that r and s are bounded by constants, and we can use the fact that $\mathrm{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ becomes easy for *fixed* r and s. Indeed, we can simply enumerate all (constantly many) partitions $P_1, \ldots, P_c$ of $[r] \times [s]$ (where empty sets are allowed) such that the matrix positions in P_ℓ contain only elements of $\mathbb{G}_\ell$. Then, for $\ell \in [c]$, we solve in polynomial time the c systems of linear Diophantine equations in the variables $\xi_{ij}^{(\ell)}$ for $(i, j) \in P_\ell$ that encode the row and column sum conditions. This will produce integer solutions for each color or decide infeasibility in polynomial time.

Let us close this section with a result on the tightness of the bound $c \leq \lfloor rs/(r + s - 1) \rfloor$ in Theorem 2.1. While, by Proposition 3.4, the right hand side is tight for $\mathrm{stp}(K_{r,s})$, its tightness in Theorem 2.1 actually depends on the underlying groups. For instance, if all groups are trivial, i.e., $\mathbb{G}_\ell = (\{0_{\mathbb{G}_\ell}\}, +)$, it follows trivially from the convention that sums over the empty set yield the neutral element of the underlying group, that solutions exist for any number c.

As it turns out, however, the bound in Theorem 2.1 is tight if the involved groups are "arithmetically rich enough". To be more precise, let us call a group $\mathbb{G}$ *arithmetically rich*, if for each r, s with $2 \leq r \leq s$ there exist $R \in (\mathbb{G}\backslash\{0_{\mathbb{G}}\})^r, S \in (\mathbb{G}\backslash\{0_{\mathbb{G}}\})^s$ such that (R, S) is balanced and for each $I \subset [r], J \subset [s]$ with $1 \leq |I| \leq r - 1$ and $1 \leq |J| \leq s - 1$,

$$\sum_{i \in I} \rho_i \neq \sum_{j \in J} \sigma_j.$$

Such a pair (R, S) will be called *rich*.

Remark 3.7 $\mathbb{Z}$ is arithmetically rich.

Proof For $r = s = 2$ choose $(R, S) = ((2, 2), (1, 3))$. Then, of course, (R, S) is balanced and rich. So let $2 \leq r$ and $2 < s$ and take $R = (1, \ldots, 1) \in \mathbb{Z}^r$, $S = (r, \ldots, r, (2 - s)r) \in \mathbb{Z}^s$. Then (R, S) is balanced. Also, for $I \subset [r]$, $J \subset [s]$ with $1 \leq |I| \leq r - 1$ and $1 \leq |J| \leq s - 1$, we have $0 < \sum_{i \in I} \rho_i < r$ while $\sum_{j \in J} \sigma_j$ is a multiple of r. Thus $\sum_{i \in I} \rho_i \neq \sum_{j \in J} \sigma_j$, i.e., (R, S) is rich; hence $\mathbb{Z}$ is arithmetically rich. □

As it turns out, the bound in Theorem 2.1 is tight, whenever all involved groups are arithmetically rich.

Theorem 3.8 *Let $c \geq 2$ and $2 \leq r \leq s$, let all groups in $\mathscr{G}$ be arithmetically rich, and suppose $\mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S}) \neq \emptyset$ for all balanced $(\mathcal{R}, \mathcal{S})$. Then $c \leq \lfloor rs/(r + s - 1) \rfloor$.*

Proof For each $\ell \in [c]$, let $(R^{(\ell)}, S^{(\ell)})$ be rich. Let $M \in \mathfrak{A}_{\mathscr{G}}^c(\mathcal{R}, \mathcal{S})$ and, for each $\ell \in [c]$, let $P_\ell \subset [r] \times [s]$ denote the positions of M carrying elements from $\mathbb{G}_\ell$ different than $0_{\mathbb{G}_\ell}$. Note that each row and column of M must contain at least one such position. Let H_ℓ be the graph with node set $[r] \times [s]$ and edges between any two nodes in P_ℓ which differ only in one coordinate. Suppose that H_ℓ was not connected, and let $I_\ell \subset [r]$, $J_\ell \subset [s]$ be inclusion minimal such that one connected component of H_ℓ lies in $I \times J$. But then, of course, $1 \leq |I_\ell| \leq r - 1$, $1 \leq |J_\ell| \leq s - 1$, and

$$\sum_{i \in I_\ell} \rho_i^{(\ell)} = \sum_{j \in J_\ell} \sigma_j^{(\ell)},$$

contradicting the choice of $(\mathcal{R}, \mathcal{S})$. Hence H_ℓ is connected. Therefore the associated subgraph of $K_{r,s}$ induced by the edges that correspond to P_ℓ contains a spanning tree. Hence M leads to a packing of spanning trees $K_{r,s}$, and thus $c \leq \text{stp}(K_{r,s})$. □
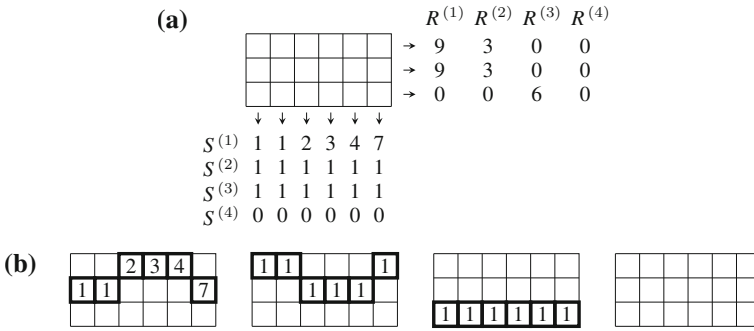
4 Computational Complexity

Let us now turn to the hardness results asserted in Theorem 2.2. First note that for $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$ and $\text{COLRYS}_c(\mathscr{Z}, \mathcal{N}_0)$ membership in $\mathbb{NP}$ is clear. To prove $\mathbb{NP}$-hardness we will use a transformation from

STRONG PARTITION

Given $n \in \mathbb{N}$ and $v_1, \ldots, v_n \in \mathbb{N}$ such that, both, n and $v = \sum_{i=1}^{n} v_i$ are even; does there exist $N \subset [n]$ such that

$$|N| = \frac{n}{2} \quad \text{and} \quad \sum_{i \in N} v_i = \sum_{i \in [n] \setminus N} v_i?$$

Fig. 2 Construction of the transformation for $\mathcal{I} = (6; 1, 1, 2, 3, 4, 7)$. $\mathcal{I}$ is a yes-instance of STRONG PARTITION since, for $N = \{3, 4, 5\}$, we have $\sum_{i \in N} v_i = \sum_{i \in [n] \setminus N} v_i$. (**a**) Corresponding instance $\mathcal{J}$ of COLRYS$_c(\mathcal{Z}, \mathcal{Z})$; here $c = 4$, $r^* = 3$, $n = 6$, $v = \sum_{i=1}^{n} v_i = 18$. (**b**) From left to right, four matrices having the given row and column sums $(R^{(\ell)}, S^{(\ell)})$ ($\ell \in [4]$). The nonzero entries constructed in the proof are highlighted

As it easily follows from [13] (see also [9, Prob. SP13 and comments]), STRONG PARTITION is $\mathbb{NP}$-complete. Let us now begin with COLRYS$_c(\mathcal{Z}, \mathcal{Z})$, and, accordingly, fix $r^* \in \{2, \ldots, c\}$.

Let $\mathcal{I} = (n; v_1, \ldots, v_n)$ be an instance of STRONG PARTITION and suppose without loss of generality that $n \geq r^*$. In the following we construct an equivalent instance $\mathcal{J} = (r^*, n; \mathcal{R}, \mathcal{S})$ of COLRYS$_c(\mathcal{Z}, \mathcal{Z})$; see Fig. 2 for an example illustrating the construction. For each $\ell \in [c]$ we define (potential) row and column sums

$$R^{(\ell)} = \left(\rho_1^{(\ell)}, \ldots, \rho_{r^*}^{(\ell)}\right) \in \mathbb{Z}^{r^*} \quad \text{and} \quad S^{(\ell)} = \left(\sigma_1^{(\ell)}, \ldots, \sigma_n^{(\ell)}\right) \in \mathbb{Z}^n$$

as follows. Let

$$\left(\rho_1^{(1)}, \ldots, \rho_{r^*}^{(1)}\right) = \left(\tfrac{v}{2}, \tfrac{v}{2}, 0, \ldots, 0\right), \quad \left(\sigma_1^{(1)}, \ldots, \sigma_n^{(1)}\right) = (v_1, \ldots, v_n),$$

$$\left(\rho_1^{(2)}, \ldots, \rho_{r^*}^{(2)}\right) = \left(\tfrac{n}{2}, \tfrac{n}{2}, 0, \ldots, 0\right), \quad \left(\sigma_1^{(2)}, \ldots, \sigma_n^{(2)}\right) = (1, \ldots, 1),$$

and for each $\ell \in [c]$ with $3 \leq \ell \leq r^*$ let

$$\left(\rho_1^{(\ell)}, \ldots, \rho_{r^*}^{(\ell)}\right) = (n \cdot \delta_{i\ell})_{1 \leq i \leq r^*}, \quad \left(\sigma_1^{(\ell)}, \ldots, \sigma_n^{(\ell)}\right) = (1, \ldots, 1),$$

where δ_{ij} is the usual Kronecker delta. Finally, for each $\ell \in [c]$ with $r^* < \ell \leq c$, let

$$\left(\rho_1^{(\ell)}, \ldots, \rho_{r^*}^{(\ell)}\right) = (0, \ldots, 0), \quad \left(\sigma_1^{(\ell)}, \ldots, \sigma_n^{(\ell)}\right) = (0, \ldots, 0).$$

Note, first, that each of the pairs $(R^{(\ell)}, S^{(\ell)})$ is balanced for each $\ell \in [c]$; hence we have constructed an instance $\mathcal{J}$ of COLRYS$_c(\mathcal{Z}, \mathcal{Z})$. Of course, the construction requires only polynomial time.

Now we show that $\mathcal{I}$ and $\mathcal{J}$ are equivalent. Suppose, first, that $\mathcal{I}$ is a yes instance of STRONG PARTITION and let $N \subset [n]$ be such that $\sum_{i \in N} v_i = \sum_{i \in [n] \setminus N} v_i$. We fill the entries of an $r^* \times n$ matrix $M = (\mu_{ij}^{(\ell_{ij})})$ as follows:

$$\mu_{1,j}^{(1)} = v_j \text{ for } j \in N, \text{ and } \mu_{2,j}^{(1)} = v_j \text{ for } j \in [n] \setminus N;$$
$$\mu_{1,j}^{(2)} = 1 \text{ for } j \in [n] \setminus N \text{ and } \mu_{2,j}^{(2)} = 1 \text{ for } j \in N;$$
$$\mu_{\ell,j}^{(\ell)} = 1 \text{ for } j \in [n] \text{ and } \ell \in \{3, \ldots, r^*\}.$$

Note that the matrix is now filled completely. In particular, colors $\ell > r^*$ are never used. Clearly, $M \in \mathfrak{A}_{\mathscr{Z},\mathscr{Z}}^c(\mathcal{R}, \mathcal{S})$.

Conversely, let $\mathcal{J}$ be a yes-instance of $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$, and let $M = (\mu_{ij}^{(\ell_{ij})}) \in \mathfrak{A}_{\mathscr{Z},\mathscr{Z}}^c(\mathcal{R}, \mathcal{S})$. By the choice of $\mathcal{S}$, each column contains at least one nonzero entry of each color $\ell \in [r^*]$, which adds up to a total of $r^* \cdot n$ positions filled with nonzero entries. Hence for each color $\ell \in [r^*]$ there is exactly one nonzero entry in each column, and its value is given by the prescribed column sum. This also implies that for none of the colors $\ell \geq r^* + 1$, any of the entries of M is of that color, i.e., $\ell_{ij} \neq \ell$ for $i \in [r^*]$, $j \in [n]$, $\ell \geq r^* + 1$. Hence $\mu_{\ell,j}^{(\ell)} = 1$ for $j \in [n]$ and $\ell \in \{3, \ldots, r^*\}$, i.e., the ℓ-th row of M is filled with 1's of the color ℓ for $\ell \in \{3, \ldots, r^*\}$. Also $\ell_{ij} \notin \{1, 2\}$ for $i \geq 3$. Now, with $N = \{j \in [n] : \ell_{1,j} = 1\}$, the row sum equalities imply that

$$\sum_{i \in N} v_i = \sum_{i \in [n] \setminus N} v_i = \frac{v}{2} \quad \text{and} \quad |N| = \frac{n}{2},$$

which concludes the proof of the reduction for $\text{COLRYS}_c(\mathscr{Z}, \mathscr{Z})$.

To prove the $\mathbb{NP}$-hardness of $\text{COLRYS}_c(\mathscr{Z}, \mathcal{N}_0)$ we can apply (a simplified version of) the same construction. Just observe that a row sum 0 for any color implies that no (nonzero) entry of the matrix M in that row can be of that color. Hence, for larger r, the construction can just be amended by 0 row sums. Now all hardness assertions of Theorem 2.2 have been established.

Note that over $\mathbb{Z}$, the argument on zero row sums forcing zero values in the matrix is obviously wrong. So this additional freedom of realizing row sums 0 in a nontrivial way is actually responsible for the drastic drop in complexity over $\mathbb{Z}$ which is shown in Theorem 2.2.

5 Final Remarks

We conclude with some final remarks and open problems.

Let us first point out that the problem we are considering can be viewed as a special variant of the *multi-commodity flow* problem. In fact, let the node sets $A = \{a_1, \ldots, a_r\}$ and $B = \{b_1, \ldots, b_s\}$ of $K_{r,s}$ correspond to r supply and s demand stations for c liquids which can be transported through pipes (modeled by the edges of $K_{r,s}$). No pipe can be used for different liquids, and the quantities of flow are integer (say, in order to guarantee a certain precision for subsequent mixtures).

Now, suppose we are given c different integer quantities of demand or supply, respectively, at each node such that the total demand equals the total supply for each of the c liquids. In this situation we will speak of a *service request*.

Given a service request, we are interested in facilitating the transport in integer quantities in such a way that all demands and supplies are met and no two liquids use the same pipe. Note that negative flows are not prohibited, i.e., flows may transport some liquid from a demand to a supply station in order to satisfy all demands and supplies at each node. Clearly, this asks for consistency (and subsequently, the reconstruction of a solution) of the corresponding instances of COLRYS$_c(\mathscr{L}, \mathscr{L})$ with all groups being $\mathbb{Z}$.

Actually, if we require a "general solution", i.e., that the pipes are reserved for individual liquids so as to guarantee that *every possible* service request can be executed with the same assignment of liquids to pipes, we are exactly in the situation of Theorem 2.1.

Of course, in the context of multi-commodity flow some questions arise naturally. For instance, bounds on the capacity of the pipes lead to corresponding restrictions of $\mathscr{W}$. Similarly, when only nonnegative flows are allowed the entries of our colored matrix are confined to $\mathbb{N}_0$. Then, of course, we are in the situation of COLRYS$_c(\mathscr{L}, \mathscr{N}_0)$ and Theorem 2.2 provides a corresponding $\mathbb{NP}$-hardness result. Note that, if the capacities are bounded above by 1 and negative flow is prohibited we are actually dealing with binary tomography for c classes of atoms which is $\mathbb{NP}$-hard for $c \geq 2$, [6].

Let us now mention two open problems in discrete tomography. In spite of the computational simplicity of the classical monochromatic binary tomography problem of reconstructing a 0–1 matrix from its row and column sums (the "classical Ryser problem"), the complexity status of the corresponding counting problem ("Determine the number of solutions!") is, most annoyingly, still open. Some restricted variants are, however, known to be related to the problem of computing the permanent of a matrix and are thus #$\mathbb{P}$-hard; [7]. See also [2] for counting results over finite fields. Perhaps even more surprisingly, while the consistency problem for the 2-atom case for two directions is $\mathbb{NP}$-hard, [6], the complexity status of the corresponding uniqueness and counting problems remains unknown. (The ambiguity of exchanging neutral elements $0_{\mathbb{G}_\ell}$ is avoided by regarding them together as an extra color.)

Our present paper has focused on the polyatomic case for X-rays in two directions. It is, of course, natural to ask similar questions for X-rays in more than two directions or query sets other than lines, particularly hyperplanes, as they lead to the Radon transform. As it turns out, the situation is more complicated then, exceeds by far the scope of the present paper and is therefore deferred to a subsequent article. Let us just mention that the results of Sect. 3 can, to some extent, be generalized, leading to packings of certain hypertrees in associated hypergraphs.

References

1. Alpers, A., Gritzmann, P.: On the reconstruction of static and dynamic discrete structures. In: Ramlau, R., Scherzer, O. (eds.) The Radon Transform: The First 100 Years and Beyond, pp. 297–342. De Gruyter, Berlin (2019)
2. Brawley, J.V., Carlitz, L.: Enumeration of matrices with prescribed row and column sums. Linear Algebra Appl. **6**, 165–174 (1973)
3. Brualdi, R.A.: Matrices of zeros and ones with fixed row and column sum vectors. Linear Algebra Appl. **33**, 159–231 (1980)
4. Brunetti, S., Del Lungo, A., Gritzmann, P., de Vries, S.: On the reconstruction of binary and permutation matrices under (binary) tomographic constraints. Theor. Comput. Sci. **406**(1–2), 63–71 (2008)
5. Cornuéjols, G.: Combinatorial Optimization: Packing and Covering. CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 74. SIAM, Philadelphia (2001)
6. Dürr, C., Guíñez, F., Matamala, M.: Reconstructing 3-colored grids from horizontal and vertical projections is NP-hard: a solution to the 2-atom problem in discrete tomography. SIAM J. Discrete Math. **26**(1), 330–352 (2012)
7. Gardner, R.J., Gritzmann, P., Prangenberg, D.: On the computational complexity of reconstructing lattice sets from their X-rays. Discrete Math. **202**(1–3), 45–71 (1999)
8. Gardner, R.J., Gritzmann, P., Prangenberg, D.: On the computational complexity of determining polyatomic structures by X-rays. Theor. Comput. Sci. **233**(1–2), 91–106 (2000)
9. Garey, M.R., Johnson, D.S.: Computers and Intractability: A Guide to the Theory of NP-Completeness. A Series of Books in the Mathematical Sciences. W.H. Freeman and Co., San Francisco (1979)
10. Grimm, U., Gritzmann, P., Huck, C.: Discrete tomography of model sets: reconstruction and uniqueness. In: Baake, M., Grimm, U. (eds.) Aperiodic Order, vol. 2. Encyclopedia of Mathematics and Its Applications, vol. 166, pp. 39–72. Cambridge University Press, Cambridge (2017)
11. Grünbaum, B.: A result on graph-coloring. Mich. Math. J. **15**, 381–383 (1968)
12. Herman, G.T., Kuba, A. (eds.): Discrete Tomography: Foundations, Algorithms, and Applications. Birkhäuser, Basel (1999)
13. Karp, R.M.: Reducibility among combinatorial problems. In: Miller, R.E., Thatcher, J.W. (eds.) Complexity of Computer Computations, pp. 85–103. Plenum, New York (1972)
14. Lagarias, J.C.: Point lattices. In: Graham, R.L., Grötschel, M., Lovász, L. (eds.) Handbook of Combinatorics, vols. 1–2. Elsevier, Amsterdam (1995)
15. Nash-Williams, C.St.J.A.: Edge-disjoint spanning trees of finite graphs. J. Lond. Math. Soc. **36**, 445–450 (1961)
16. Palmer, E.M.: On the spanning tree packing number of a graph: a survey. Discrete Math. **230**(1–3), 13–21 (2001)
17. Ryser, H.J.: Matrices of zeros and ones. Bull. Am. Math. Soc. **66**, 442–464 (1960)
18. Schrijver, A.: Combinatorial Optimization: Polyhedra and Efficiency. Algorithms and Combinatorics, vol. 24. Springer, Berlin (2003)
19. Seamone, B.: The 1-2-3 conjecture and related problems: a survey. arXiv:1211.5122 (2012)
20. Tutte, W.T.: On the problem of decomposing a graph into *n* connected factors. J. Lond. Math. Soc. **36**, 221–230 (1961)