Efficient algorithms for the Zarankiewicz problem

Andrew Kay*
School of Computing and Digital Technology,
Birmingham City University

April 19, 2016

Abstract

The Zarankiewicz problem asks for the maximum number of 1s in an $m \times n$ matrix with no $s \times t$ minor containing only 1s. We present a general algorithm and a specific algorithm for the case $s = t = 2$, each substantially more efficient than previous work. The algorithms are based on a generalisation of Paige–Wexler canonical form for finite projective planes, and a new connection with symmetric inverse semigroups analogous to a connection between finite projective planes and symmetric groups. We obtain over 200 new exact values, and correct previously unreported errors in R. K. Guy's tables. Finally, we make some observations which may apply to the search for finite projective planes.

Keywords. Zarankiewicz problem, rectangle-free, forbidden minor, finite projective plane, extremal combinatorics, computational combinatorics, constraint programming.

1 Introduction

In 1951, Zarankiewicz posed some specific cases of a problem which, more generally, asks for the maximum number of 1s in an $m \times n$ matrix with no $s \times t$ minor containing only 1s. [Zar51] The problem is also stated in extremal graph theory as the maximal number of edges in a bipartite graph on vertex sets U, V with $\big(|U|, |V|\big) = (m, n)$ such that no s vertices from U and t vertices from V span a complete bipartite subgraph. [DHS13]

The problem is difficult in general; exact values known to date are due to a few theorems establishing exact values for some infinite classes of parameters, a patchwork of upper and lower bounds none comprehensively sharper than another, *ad hoc* methods for particular cases, and more recently by computer search. Most effort has been focused on the case $s = t = 2$. The problem remains an active area of research; see e.g. [DHS13, DDR13, FS13, SP12, Wer12].

*Electronic address: `Andrew.Kay@bcu.ac.uk`

1.1 Notation

Matrices in this paper have entries in $\{I, O\}$ rather than $\{1, 0\}$. This unusual choice of notation is to reduce cognitive dissonance, as in Section 1.3 we will define an ordering with $I < O$. The notation also extends more naturally to matrices with entries in semigroups in Section 3.2. We interpret I as an incidence flag with no arithmetic properties.

The set of natural numbers (including 0) is $\mathbb{N}$, and $\mathbb{N}_\infty = \mathbb{N} \cup \{\infty\}$. The set of $m \times n$ matrices with entries in a set Σ is $\mathrm{Mat}_\Sigma(m, n)$. The weight $w(A)$ of a matrix $A \in \mathrm{Mat}_{\{I,O\}}$ is its number of I entries. More generally, we allow a weight function $w : \Sigma \to \mathbb{N}$ to be extended to $w : \mathrm{Mat}_\Sigma(m, n) \to \mathbb{N}$ via the formula

$$w(A) = \sum_{i=1}^{m} \sum_{j=1}^{n} w(A_{i,j}) = \sum_{\sigma \in \Sigma} |A|_\sigma w(\sigma) \tag{1.1}$$

where $|A|_\sigma$ is the number of σ entries in A. The specific case has $w(A) = |A|_I$, implying $w(I) = 1$ and $w(O) = 0$.

Definition 1.2. *An (s, t)-rectangle is an $s \times t$ minor containing only I entries. A matrix with no (s, t)-rectangles is (s, t)-rectangle-free.*

Definition 1.3. *$z(m, n;\ s, t)$ is the maximum weight of an (s, t)-rectangle-free $m \times n$ matrix.*[1]

For brevity, in the case $s = t = 2$ we will simply write *rectangle*, *rectangle-free* and $z(m, n)$. We will leave O entries blank where this aids readability.

When referring to the rows and columns of a particular matrix, we will write R_i for row i and C_j for column j.

As usual, $[n] = \{1, 2, \ldots, n\}$ and $\binom{n}{k}$ is the binomial symbol. $<_{\text{lex}}$ is lexicographic order.

It will sometimes be useful to identify matrices in $\mathrm{Mat}_{\{I,O\}}(m, n)$ with subsets of $[m] \times [n]$; in this case, $(i, j) \in A$ if and only if $A_{i,j} = I$. We also write

- $\operatorname{dom} A \subseteq [m]$ for the domain, and $\operatorname{ran} A \subseteq [n]$ for the range,
- $U \nrightarrow V$ for the set of partial functions from U to V,
- $U \rightarrowtail\!\!\!\!\!\nrightarrow V$ for the set of partial injections[2] from U to V, and
- $B \circ A$ for backward relational composition,

as in the Z notation. [Spi92]

[1] Earlier literature (e.g. [Guy69, KST54, Rom75]) refers instead to the *minimum* weight such that *every* $m \times n$ matrix has an (s, t)-rectangle. This is of course $z(m, n;\ s, t) + 1$.

[2] Sometimes called partial bijections, or partial permutations.

1.2 Bounds and exact values

We summarise a number of theorems establishing bounds and exact values for $z(m, n;\ s, t)$. Some theorems are asymmetric in m and n, in that they might give a sharper bound on the transpose matrix. Clearly $z(m, n;\ s, t) = z(n, m;\ t, s)$.

The first result is used in various arguments e.g. in [Guy69, KST54, Rom75].

Definition 1.4. *For $A \in \mathrm{Mat}_\Sigma(m, n)$ with a weight function w,*

1. *The row weight distribution of A is a vector $\mathbf{r} \in \mathbb{N}^m$ with $r_i = w(R_i)$.*
2. *The column weight distribution of A is a vector $\mathbf{c} \in \mathbb{N}^n$ with $c_j = w(C_j)$.*

Theorem 1.5. *Call a vector $\mathbf{r} \in \mathbb{N}^m$ "admissible" if*

$$\sum_{i=1}^{m} \binom{r_i}{t} \leqslant (s-1)\binom{n}{t} \tag{1.6}$$

Then every (s, t)-rectangle-free matrix has an admissible row weight distribution, and

$$z(m, n;\ s, t) \leqslant \max_{\mathbf{r}} \sum_{i=1}^{m} r_i \tag{1.7}$$

where the maximum is taken over all admissible vectors.

Since the sum in (1.6) is a convex function of $\mathbf{r}$, the maximum in (1.7) can be attained by a vector with every $|r_i - r_k| \leqslant 1$. Therefore, this bound can be calculated directly without iterating over admissible vectors.

Note that an *admissible* vector is not necessarily *realisable* as the row weight distribution of an (s, t)-rectangle-free matrix.

Proof. The sum in (1.6) counts the total number of combinations of t columns spanned by the I entries in each row of A. If the row weight distribution $\mathbf{r}$ of A is not admissible, then by the pigeonhole principle there is a combination of t columns such that A has at least s rows with I entries in those columns. Hence, A is not (s, t)-rectangle-free.

Therefore an extremal (s, t)-rectangle-free matrix A has an admissible row weight distribution. $w(A) = \sum_{i=1}^{m} r_i$ gives the result. $\square$

The next theorem achieves equality for $m \gg n$, using rows of weight t and $t-1$.

Theorem 1.8 (Čulik [Čul56])**.** *If $m \geqslant (s-1)\binom{n}{t}$, then*

$$z(m, n;\ s, t) = (t-1)m + (s-1)\binom{n}{t}$$

The next theorem achieves equality in the case $s = t = 2$ for some square matrices.

Theorem 1.9 (Reiman [Rei58]). *Let $n = k^2 + k + 1$ for $k \geqslant 2$. Then*

$$z(n, n) \leqslant (k^2 + k + 1)(k + 1)$$

and a rectangle-free matrix of this weight must be the incidence matrix of a finite projective plane of order k.

The only known orders k are the prime powers. The next theorem also uses finite projective planes, and extends a lower bound of Dybizbański, Dzido & Radziszowski [DDR13].

Theorem 1.10 (Deletion principle). *Let $n = k^2 + k + 1$ where $k \geqslant 2$ is the order of a finite projective plane. Then for $c, d \in \mathbb{N}$, then*

$$z(n - c,\, n - d) \geqslant (k^2 + k + 1 - c - d)(k + 1) + X_k(c,\, d) \tag{1.11}$$

where $X_k(c, d)$ is the maximum weight of a $c \times d$ minor of an incidence matrix of a finite projective plane of order k. We have

$$\begin{aligned} X_k(0,\, d) &= 0 \\ X_k(1,\, d) &= \min\{d,\, k + 1\} \\ X_k(2,\, d) &= \min\{d + 1,\, 2k + 2\} && \text{for } d \geqslant 1 \\ X_k(3,\, d) &= \min\{d + 3,\, 3k + 3\} && \text{for } d \geqslant 3 \\ X_k(4,\, d) &= \min\{d + 6,\, 4k + 4\} && \text{for } d \geqslant 6 \end{aligned}$$

For k a power of 2, $X_k(7, 7) = 21$. Values valid for all k are given in the table below.

c, d	**1**	**2**	**3**	**4**	**5**	**6**	**7**
1	1∗	2	3				
2		3∗	4	5	6		
3			6∗	7	8	9	
4				9	10	12	
5					12	14	15
6						16	18

Furthermore, (1.11) is an equality in the cases marked ∗.

Proof. We obtain a rectangle-free matrix satisfying the lower bound by deleting c rows and d columns from the incidence matrix of a finite projective plane of order k. The weight of the resulting matrix is given by (1.11) when $X_k(c, d)$ is the weight of the $c \times d$ minor of their intersection, by the inclusion/exclusion principle.

For $X_k(1, d)$ choose any row and any $\min\{d,\, k + 1\}$ columns intersecting that row.

For $X_k(2, d)$ with $d \geqslant 1$, choose any two rows, the column intersecting both, and any other $\min\{d - 1,\, 2k\}$ columns intersecting either.

In the remaining cases, by the non-degeneracy axiom of a finite projective plane, there are four columns $C_1, \ldots, C_4$ such that no row intersects any three. There must be six distinct rows $R_1, \ldots, R_6$ intersecting each pair; then the three pairs of rows (R_1, R_6), (R_2, R_5) and (R_3, R_4) must each intersect in distinct columns C_5, C_6, C_7.

$$\begin{array}{cc} & \begin{array}{ccccccc} C_1 & C_2 & C_3 & C_4 & C_5 & C_6 & C_7 \end{array} \\ \begin{array}{c} R_1 \\ R_2 \\ R_3 \\ R_4 \\ R_5 \\ R_6 \end{array} & \left(\begin{array}{cccc|ccc} I & I & & & I & & \\ I & & I & & & I & \\ I & & & I & & & I \\ & I & I & & & & I \\ & I & & I & & I & \\ & & I & I & I & & \end{array}\right) \end{array} \tag{1.12}$$

It follows that any incidence matrix of a finite projective plane has (1.12) as a minor, and therefore has $c \times d$ (or $d \times c$) minors of weight $X_k(c, d)$ as given in the table above.

For $X_k(3, d)$ with $d \geqslant 3$, choose the 3×3 minor of weight 6, and any other $\min\{d - 3, 3k\}$ columns intersecting its rows.

For $X_k(4, d)$ with $d \geqslant 6$, choose the 6×4 minor of weight 12, and any other $\min\{d - 6, 4k - 8\}$ rows intersecting its columns.

For k a power of 2, a Fano subplane gives such a 7×7 minor. [Cam95, p. 671]

Equality in the cases marked $*$ is a theorem of Dybizbański et al. [DDR13]. $\square$

By Čulik's theorem (Theorem 1.8, [Čul56]) and exhaustion, the values given for $X_k(c, d)$ are equal to $z(c, d)$ for sufficiently large k, so they give the sharpest possible lower bound using this method.

In Section 2.4 we will use some other bounds and exact values which are established in [DHS13,FS13,KST54,Rei58,Rom75]. The remaining theorem is strictly weaker than a bound of Damásdi, Héger & Szőnyi [DHS13], but can be used dynamically on partial solutions in Section 2.4, when r is known.

Theorem 1.13.

$$z(m, n;\ s, t) \leqslant \max_{0 \leqslant r \leqslant n} \min \left\{ \begin{array}{ccc} & & r \\ rm, & + & z(m-1,\ r;\ s-1,\ t) \\ & + & z(m-1,\ n-r;\ s,\ t) \end{array} \right\}$$

Proof. Let $r = w(R_i)$ be the greatest row weight in an extremal (s, t)-rectangle-free matrix A. Trivially $w(A) \leqslant rm$.

Let M_I and M_O be the minors formed of all rows except R_i, and the columns C_j with $A_{i,j} = I$ and O respectively. M_I is $(m-1) \times r$ and $(s-1,\ t)$-rectangle-free, and M_O is $(m-1) \times (n-r)$ and (s, t)-rectangle-free. The result follows as $w(A) = r + w(M_I) + w(M_O)$. $\square$

Lastly we note that $z(m, n;\ s, t) \geqslant z(m-1,\ n;\ s, t) + t - 1$, and if $m \geqslant s+1$ and $n \geqslant t+1$ then $z(m, n;\ s, t) \geqslant z(m-1,\ n-1;\ s, t) + s + t - 1$. This bound is usually not sharp, but is exact in infinitely many cases.[3]

1.3 Lex-minimal form

Reiman's theorem (Theorem 1.9, [Rei58]) states that finite projective plane incidence matrices are extremal rectangle-free matrices. We begin by considering these matrices in Paige–Wexler canonical form. [PW53]

Example 1.14. *The finite projective plane of order 3 in Paige–Wexler canonical form.*

$$
\begin{array}{|cccc|ccc|ccc|ccc|}
\hline
I & I & I & I & & & & & & & & & \\
I & & & & I & I & I & & & & & & \\
I & & & & & & & I & I & I & & & \\
I & & & & & & & & & & I & I & I \\
\hline
 & I & & & I & & & I & & & I & & \\
 & I & & & & I & & & I & & & I & \\
 & I & & & & & I & & & I & & & I \\
\hline
 & & I & & I & & & & I & & & & I \\
 & & I & & & I & & & & I & I & & \\
 & & I & & & & I & I & & & & I & \\
\hline
 & & & I & I & & & & & I & & I & \\
 & & & I & & I & & I & & & & & I \\
 & & & I & & & I & & I & & I & & \\
\hline
\end{array}
$$

Paige & Wexler described their canonical form directly, then showed that any finite projective plane incidence matrix can be put into this form by row/column permutations. However, their canonical form can be described more succintly, and more generally.

Definition 1.15. *For a total order $<$ on $\mathrm{Mat}_\Sigma(m, n)$, the $<$-minimal form of a matrix is the $<$-minimal member of its orbit under row and column permutations, and for square matrices, transposition.*[4]

If $<$ is a total order on Σ, we define lex order on matrices by comparing them on the first row in which they differ.[5] We are interested in $<_{\text{lex}}$-minimal form.

For $\Sigma = \{I, O\}$ we define $I < O$. In this case, finite projective plane incidence matrices in lex-minimal form are necessarily in Paige–Wexler canonical form.[6] The lex-minimal form of a rectangle-free matrix has clear similarities with Paige–Wexler canonical form, as illustrated in the following example.

[3] E.g. in Čulik's theorem (Theorem 1.8, [Čul56]), or adding one row to a finite projective plane [DHS13].

[4] Or, in general, the symmetry group of the problem space.

[5] Since the order is defined only for matrices of equal dimensions, this is equivalent to $<_{\text{lex}}$ on the strings formed by concatenating their rows.

[6] A formal proof of this is possible either directly, or using Reiman's theorem [Rei58] and various results from this paper.

Example 1.16. *A matrix in lex-minimal form satisfying* $z(16, 22) = 83$.

I	I	I	I	I	I																
I						I	I	I	I	I											
I											I	I	I	I							
	I					I					I				I	I					
	I						I					I					I	I			
	I							I					I						I	I	
		I				I						I							I		I
		I						I						I	I		I				
		I							I				I			I		I			
			I			I								I				I		I	
			I				I						I		I						I
			I						I		I						I		I		
				I			I							I		I			I		
				I				I			I							I			I
				I					I			I			I					I	
					I					I						I	I			I	I

The use of lex order for symmetry reduction in matrix search problems has been noted e.g. in [FFH$^+$02] which, incidentally, uses $z(4, 3) = 7$ as an example — though the authors do not identify the Zarankiewicz problem by name. In Section 2.3 we will use an alternative ordering to achieve greater algorithmic efficiency for the Zarankiewicz problem.

1.4 Properties of lex-minimal form

Let A be a matrix in lex-minimal form, relative to a total order $<$ on the entry set Σ.

The main lemma resembles an existence proof in [FFH$^+$02].

Lemma 1.17. *The rows of A are in lex order, and the columns of A are in lex order.*

Proof. Suppose $R_i >_{\text{lex}} R_k$ for $i < k$. Then swapping rows i and k yields a matrix lower in lex order; a contradiction.

Suppose $C_j >_{\text{lex}} C_l$ for $j < l$. Then let i be the first row in which C_j and C_l differ. Swapping columns j and l leaves rows above i unchanged, but yields a row i lower in lex order; a contradiction. □

Lemma 1.18. *If $I \in \Sigma$ is $<$-minimal, then every $|R_i|_I \leqslant |R_1|_I$.*

Proof. If not, then swapping rows R_1 and R_i and sorting the columns in lex order yields a first row lower in lex order; a contradiction. □

The next lemma applies only to rectangle-free matrices, and will be used in Section 3.

Lemma 1.19. *Suppose also that $\Sigma = \{I, O\}$ with $I < O$, and A is rectangle-free. Then for all $i < k \leqslant w(C_1)$, we have $w(R_i) \geqslant w(R_k)$.*

Proof. Since $i < k \leqslant w(C_1)$ and the rows are in lex order, we must have $(i,1) \in A$ and $(k,1) \in A$. As A is rectangle-free, R_i and R_k do not have a common I in any other column. Since the columns of A are in lex order, the matrix must be as in the following diagram:

$$\begin{array}{c} R_1 \\ R_i \\ R_k \end{array}\begin{pmatrix} I\cdots\cdots I & & \\ I & I\cdots I & \\ I & & I\cdots\cdots I \end{pmatrix} \quad\mapsto\quad \begin{array}{c} R_1 \\ R_i \\ R_k \end{array}\begin{pmatrix} I\cdots\cdots I & & \\ I & I\cdots\cdots I & \\ I & & I\cdots I \end{pmatrix}$$

If $w(R_i) < w(R_k)$, then swapping rows i and k and sorting the columns in lex order leaves rows above i unchanged, but yields a row i lower in lex order; a contradiction. $\square$

Lemmas 1.18 and 1.19 are not true of the transpose, as seen in Example 1.16.

2 General algorithm

In this section we describe an algorithm for finding extremal (s,t)-rectangle-free matrices, as a specific instance of an abstract model for algorithms which solve combinatorial or constraint programming problems on matrices.

2.1 Abstract model

The algorithm is defined by matrix dimensions m, n, an entry set Σ, a weight function $w : \mathrm{Mat}_\Sigma(m,n) \to \mathbb{N}$, a set of procedures enforcing constraints, a set of bounding functions, and a guess function.

- A *partial solution* is a matrix $P \in \mathrm{Mat}_{\mathbb{P}\Sigma}(m,n)$. The entry $P_{i,j} \subseteq \Sigma$ contains the symbols which have not (yet) been eliminated as possible entries in completions of P.
- A matrix $A \in \mathrm{Mat}_\Sigma(m,n)$ is a *completion* of a partial solution P if every $A_{i,j} \in P_{i,j}$, and A is unchanged by the procedures enforcing constraints.[7]
- An entry $P_{i,j}$ is *confirmed* if $|P_{i,j}| = 1$. A partial solution is *complete* if all its entries are confirmed, and *consistent* if every $|P_{i,j}| > 0$.
- A procedure enforcing a *global* or *local constraint* operates on a partial solution P. If the procedure enforces a local constraint, it also takes the indices of a fresh entry of P as parameters. The procedure transforms P into a new partial solution P' with every $P'_{i,j} \subseteq P_{i,j}$. An entry (i,j) is *fresh* if it is confirmed in P' but not in P.
- A *bounding function* $\overline{w} : \mathrm{Mat}_{\mathbb{P}\Sigma}(m,n) \nrightarrow \mathbb{N}_\infty$ is defined on consistent partial solutions. $\overline{w}(P)$ gives an upper bound on the weight of a completion, i.e. $w(A) \leqslant \overline{w}(P)$ whenever A is a completion of P.

[7]We regard A as a complete, consistent partial solution by treating its entries as singleton sets.

- A *guess function* $G : \mathrm{Mat}_{\mathbb{P}\Sigma}(m, n) \nrightarrow [m] \times [n] \times \Sigma$ is defined on incomplete consistent partial solutions. $G(P)$ selects an entry and a symbol to guess once no further progress can be made otherwise. If $G(P) = (i, j, \sigma)$ then we require $|P_{i,j}| > 1$ and $\sigma \in P_{i,j}$.

Constraints must be used to enforce the rules of the problem being solved, but can also be used to break symmetry. In this paper, global constraints are not used.

The algorithm itself is a recursive tree search, using constraint propagation and bounds to eliminate branches. The algorithm is described below as a procedure operating on a partial solution P, a subset of fresh indices $F \subseteq [m] \times [n]$, a lower bound $L \in \mathbb{N}$ and an upper bound $U \in \mathbb{N}_\infty$. By default, every $P_{i,j} = \Sigma$, $F = \emptyset$, $L = 0$ and $U = \infty$. The values of L and U are preserved during recursive calls and backtracks. Backtracking from the initial call is equivalent to terminating.

1. Repeat until no changes are made to P:
   (a) While F is non-empty: take an $(i, j) \in F$, delete it from F, and then enforce each local constraint on P at (i, j), inserting any fresh indices into F.
   (b) Enforce each global constraint on P, inserting any fresh indices into F.

   If P becomes inconsistent during this loop, then reject P and backtrack.
2. For each bounding function $\overline{w}$: if $\overline{w}(P) < L$ then reject P and backtrack.
3. If P is complete:
   (a) If $w(P) \geqslant L$, output P; otherwise, reject P and backtrack.
   (b) If $w(P) = U$ then terminate; otherwise, let $L = w(P) + 1$ and backtrack.[8]
4. Let $(i, j, \sigma) = G(P)$. Copy P to form a new partial solution P', then let $P'_{i,j} = \{\sigma\}$.
5. Recursively solve P', using $F' = \{(i, j)\}$.
6. Delete σ from $P_{i,j}$. If now $|P_{i,j}| = 1$, insert (i, j) into F.
7. Repeat from 1.

The model is implemented in a Java package written by the author. A more thorough investigation of this abstract model will follow in a future paper; for now, we note that the algorithm's efficiency depends on

- The ability of the constraints to eliminate possible entries from partial solutions, so that they do not need to be eliminated by branching,
- The ability of the constraints to reject branches which, by symmetry, are isomorphic to other branches searched,

[8]If all solutions are desired, this step can be modified appropriately.

- The ability of the bounding functions to quickly detect fruitless branches,
- The ability of the guess function to discover good solutions quickly so that the lower bound L can be improved, and to discover latent inconsistencies quickly (i.e. when no completion exists),

among other factors. These should be considered together; for example, if G always guesses in the first incomplete row, then a bounding function which uses completed rows will be more useful than a bounding function using completed columns. Also, there is often a choice of which factor to improve by exploiting symmetry.

Since the number of branches is typically exponential, any polynomial-time improvement in these factors is likely to be (asymptotically) worth the investment.

2.2 Constraints and bounds for hereditary problems

This section applies generally to combinatorial optimisation problems on matrices for which the weight function is extended from $w : \Sigma \to \mathbb{N}$ to $w : \mathrm{Mat}_\Sigma \to \mathbb{N}$ as in (1.1), and which are *hereditary* in the sense that if a matrix is a valid solution, then its minors also are. This class of problems includes forbidden-minor problems such as the Zarankiewicz problem.

For generality, we write $W(m, n) = \max w(A)$, where the maximum is taken over valid matrices. For the Zarankiewicz problem, $W(m, n) = z(m, n;\ s, t)$. The following result generalises upper bounds given in [Guy68] and [DHS13].

Proposition 2.1 (Density principle)**.** *For $0 \leqslant c \leqslant m$ and $0 \leqslant d \leqslant n$,*

$$W(m, n) \leqslant \frac{mn}{cd}\, W(c, d) \tag{2.2}$$

and

$$W(m, n) \leqslant W(c, n) + (m - c) \left\lfloor \frac{W(c, n)}{c} \right\rfloor \tag{2.3}$$

Proof. Let A be an extremal $m \times n$ solution. Then

$$\binom{m-1}{c-1}\binom{n-1}{d-1} w(A) = \sum_M w(M) \leqslant \binom{m}{c}\binom{n}{d} W(c, d)$$

where the sum ranges over all $c \times d$ minors M of A. When $c = m - 1$, (2.3) is a special case of (2.2) with $d = n$, where the floor function may be used as W is integer-valued. The result follows by induction on $m - c$. □

Next we show results applying to partial solutions. Consider the lower and upper bounds

$$\underline{w}(P) = \sum_{i=1}^{m}\sum_{j=1}^{n} \min_{\sigma \in P_{i,j}} w(\sigma) \quad \leqslant \quad w(A) \quad \leqslant \quad \overline{w}(P) = \sum_{i=1}^{m}\sum_{j=1}^{n} \max_{\sigma \in P_{i,j}} w(\sigma) \tag{2.4}$$

which follow directly from (1.1) whenever A is a completion of P. The procedure enforcing the constraint should delete possible entries from P as appropriate.

Constraint 2.5. *If (i, j) is fresh and $W(m, n) \geqslant L$, then require $\overline{w}(R_i) \geqslant L - W(m-1,\, n)$, and $\overline{w}(C_j) \geqslant L - W(m,\, n-1)$.*

Proof. If $w(A) \geqslant L$ then deleting R_i yields an $(m-1)\times n$ solution such that

$$L - w(R_i) \leqslant w(A) - w(R_i) \leqslant W(m-1,\, n)$$

Similarly for C_j. □

As in Section 2.1, finding a solution allows L to be increased, improving the ability of the constraint to eliminate the rest of the search tree.

Bound 2.6. *If A is the completion of a partial solution P, then*

$$w(A) \leqslant \overline{w}(P) = W(c, d) + \sum_{A \setminus M} w(A_{i,j})$$

where M is a $c\times d$ minor of A containing all of the entries which are unconfirmed in P, and the sum ranges over entries not in M.

Proof. By (1.1) we have $w(A) = w(M) + \sum_{A\setminus M} w(A_{i,j})$, where $w(M) \leqslant W(c, d)$. □

2.3 Weightlex order

We look for solutions not in lex-minimal form, but in a similar form which takes better advantage of symmetry. This section applies generally to matrix search problems for which the weight function is extended from $w : \Sigma \to \mathbb{N}$ to $w : \mathrm{Mat}_\Sigma \to \mathbb{N}$ as in (1.1), and the set of valid matrices is closed under row and column permutations, and in the case of square matrices, transposition.

Definition 2.7. *Let $A, A' \in \mathrm{Mat}_\Sigma(m, n)$ where Σ has a total order and a weight function.*

Weightlex order: $A <_{\mathrm{wlex}} A'$ if either $w(A) > w(A')$, or $w(A) = w(A')$ and $A <_{\mathrm{lex}} A'$.

Row-weightlex order: $A <^{\mathrm{row}}_{\mathrm{wlex}} A'$ if $R_i <_{\mathrm{wlex}} R'_i$ on the first row in which they differ.

These orders have two purposes: firstly to try partial solutions of greater potential weight first, and secondly to allow assumptions about the row weights of partial solutions.

We are mainly interested in row-weightlex order. We have $A <^{\mathrm{row}}_{\mathrm{wlex}} A'$ if either $w(R_i) > w(R'_i)$, or $w(R_i) = w(R'_i)$ and $R_i <_{\mathrm{lex}} R'_i$, where i is the index of the first row in which A and A' differ. We proceed by showing properties of row-weightlex-minimal form.

Let A be a matrix in $<^{\mathrm{row}}_{\mathrm{wlex}}$-minimal form relative to a total order $<$ and weight function w on the entry set Σ.

Lemma 2.8. *The rows of A are in weightlex order, and the columns of A are in lex order.*

Proof. Suppose $R_i >_{w\text{lex}} R_k$ for $i < k$. Then swapping rows i and k yields a lower row i in row-weightlex order; a contradiction.

Suppose $C_j >_{\text{lex}} C_l$ for $j < l$. Then let i be the first row in which C_j and C_l differ. Swapping columns j and l leaves rows above i unchanged, but yields an equal weight row i lower in lex order; a contradiction. □

We immediately get a bounding function, which can be applied in the general case of searching for solutions in row-weightlex-minimal form.

Bound 2.9. *Let P be a partial solution of A, with row i of P complete for all $i \leqslant k$. Then*

$$w(A) \leqslant \overline{w}(P) = \sum_{i=1}^{k} w(R_i) + (m-k)\, w(R_k)$$

Proof. By Lemma 2.8, $w(R_i) \leqslant w(R_k)$ for $i > k$, so $\sum_{i=k+1}^{m} w(R_i) \leqslant (m-k)\, w(R_k)$. □

We also describe procedures enforcing the constraints given by Lemma 2.8, using the lower and upper bounds defined in (2.4). The procedures enforcing the constraints should delete possible entries from P as appropriate.

Constraint 2.10. *If R_i, R_{i+1} are consecutive rows with a fresh entry, require $\overline{w}(R_i) \geqslant \underline{w}(R_{i+1})$. If this is an equality, require $R_i \leqslant_{\text{lex}} R_{i+1}$.*

We may have other reasons that two rows must have equal weight, such as Constraint 2.5. In this case we can enforce both the required weights and the lex ordering separately.

Constraint 2.11. *If C_j, C_{j+1} are consecutive columns with a fresh entry, require $C_j \leqslant_{\text{lex}} C_{j+1}$.*

We also define a guess function which enables effective use of Bound 2.9 and Constraint 2.10.

Definition 2.12. *The guess function G_{lex} minimises i, then j, then σ, subject to the requirements of a guess function.*

Guesses made by G_{lex} fill in one row at a time from top to bottom. This is advantageous, as weightlex does less work on incomplete rows than lex. However, the difference is more than made up for by Bound 2.9; additionally, local weightlex breaks some symmetries which local lex does not (e.g. as in Lemma 1.19).

Weightlex orders and G_{lex} are more effective together when $\sigma < \tau$ implies $w(\sigma) \geqslant w(\tau)$, as partial solutions of greater weight will be tried first. This property holds for $\Sigma = \{I, O\}$, and also for the entry sets we will use in Section 3.1.

2.4 General algorithm for the Zarankiewicz problem

The algorithm for general $s, t \geqslant 2$ is a specific instance of the model in Section 2.1. We look for solutions in row-weightlex-minimal form, and use

- Lower and upper bounds L and U as in Section 1.2, and Proposition 2.1,[9]
- The constraints 2.5, 2.10, 2.11 and 2.13,
- The bounding functions 2.6, 2.9, 2.14 and 2.15, and
- The guess function G_{lex} as in Definition 2.12.

Whereas the definitions in Sections 2.2 and 2.3 are general, the definitions here are specific to the Zarankiewicz problem. Using the following constraint, the confirmed entries of a partial solution P remain (s,t)-rectangle-free.

Constraint 2.13. *If I is fresh at (i,j) then for each $s \times t$ minor including (i,j), if all but one of this minor's entries are confirmed as I, delete I from the remaining entry.*

Proof. Suppose P has an $s \times t$ minor of confirmed I entries. Let $P_{i,j}$ and $P_{k,l}$ be the most and second-most recently confirmed entries respectively. Then when $P_{k,l}$ was confirmed, I would have been deleted from $P_{i,j}$; a contradiction. □

Bound 2.9 eliminates partial solutions where some row weights are too small; the following bounding functions eliminate those where some row weights are too large.

Bound 2.14. *Let P be a partial solution of A, with row i of P complete for all $i \leqslant k$. Then*

$$w(A) \leqslant \overline{w}(P) = \max_{\mathbf{r}} \sum_{i=1}^{m} r_i$$

where the maximum is taken over all admissible vectors with $r_i = w(R_i)$ for $i \leqslant k$.

Proof. Similar to Theorem 1.5. As before, the bound can be calculated directly without iterating over admissible vectors.[10] □

Bound 2.15. *Let P be a partial solution of A, with R_1 complete and $w(R_1) = r$. Then*

$$\begin{aligned} w(A) \leqslant \overline{w}_{s,t}(P) = & \quad r \\ & + \min\{ \ \overline{w}_{s-1,\,t}(M_I), \quad z(m-1,\, r;\ s-1,\ t) \ \} \\ & + \min\{ \ \overline{w}_{s,t}(M_O), \quad z(m-1,\, n-r;\ s,t) \ \} \end{aligned}$$

where M_I and M_O are the minors of P as in Theorem 1.13.

[9] We increment L by 1 prior to the search; if no solution is found, the lower bound given in Section 1.2 is exact. As every lower bound in Section 1.2 is constructive, there is no disadvantage in doing this.

[10] If Constraint 2.5 gives a minimum row weight, there may be no admissible vectors. If so, we reject P.

Proof. Similar to Theorem 1.13, but note that the rows of M_I and M_O are not necessarily in descending weight order themselves, so rm is not an upper bound. We do not need to recurse on minors with fewer than s rows or t columns, as the recursive bound on the minor will equal the upper bound of (2.4). □

Finally we note that G_{lex} prefers to try rows of greatest weight first; this is advantageous except in the first row, as (heuristically by Theorem 1.5) we expect the optimal row weight distribution to be approximately uniform. Therefore we iteratively run the algorithm starting from partial solutions with confirmed first rows of weight $t \leqslant r \leqslant n$ for increasing r, so that the optimal solution is found sooner, and hence the improved lower bound L is available for more of the search.

We present results and analysis of this algorithm in Section 4.

3 Scaffolds

In this section we consider the case $s = t = 2$. The following is a motivating example.

Example 3.1. *The (improper) scaffold of Example 1.16.* $m_O = 0$.

$$
\begin{array}{r|cccccc|ccccc|cccc|c|c|c|c|c|c|c|}
\multicolumn{1}{r}{} & \multicolumn{1}{c}{} & \multicolumn{5}{c}{\overbrace{\hphantom{I\ I\ I\ I\ I}}^{p=5}} & \multicolumn{5}{c}{\overbrace{\hphantom{I\ I\ I\ I\ I}}^{n_1=5}} & \multicolumn{4}{c}{\overbrace{\hphantom{I\ I\ I\ I}}^{n_2=4}} & \multicolumn{7}{c}{\overbrace{\hphantom{I\ I\ I\ I\ I\ I\ I}}^{n_O=7}} \\
\cline{2-23}
 & I & I & I & I & I & I & & & & & & & & & & & & & & & & \\
q=2\left\{\right. & I & & & & & & I & I & I & I & I & & & & & & & & & & & \\
 & I & & & & & & & & & & & I & I & I & I & & & & & & & \\
\cline{2-23}
 & & I & & & & & \multicolumn{16}{c}{} \\
m_1=3\left\{\right. & & I & & & & & \multicolumn{16}{c}{} \\
 & & I & & & & & \multicolumn{16}{c}{} \\
\cline{2-7}
 & & & I & & & & \multicolumn{16}{c}{} \\
m_2=3\left\{\right. & & & I & & & & \multicolumn{16}{c}{} \\
 & & & I & & & & \multicolumn{16}{c}{} \\
\cline{2-7}
 & & & & I & & & \multicolumn{16}{c}{} \\
m_3=3\left\{\right. & & & & I & & & \multicolumn{16}{c}{} \\
 & & & & I & & & \multicolumn{16}{c}{} \\
\cline{2-7}
 & & & & & I & & \multicolumn{16}{c}{} \\
m_4=3\left\{\right. & & & & & I & & \multicolumn{16}{c}{} \\
 & & & & & I & & \multicolumn{16}{c}{} \\
\cline{2-7}
m_5=1\left\{\right. & & & & & & I & \multicolumn{16}{c}{} \\
\cline{2-7}
\end{array}
$$

This scaffold is "improper" as $q < m_1$.

Definition 3.2. *A scaffold is a tuple* $\mathcal{S} = (p, q, \mathbf{m}, \mathbf{n}, m_O, n_O)$ *where* $p, q, m_O, n_O \in \mathbb{N}$, $\mathbf{m} \in \mathbb{N}^p$ *and* $\mathbf{n} \in \mathbb{N}^q$, *such that* $q \geqslant m_1 \geqslant \cdots \geqslant m_p$ *and* $p \geqslant n_1 \geqslant \cdots \geqslant n_q$.

We say $\mathcal{S}$ is the scaffold of a matrix $A \in \mathrm{Mat}_{\{I,O\}}(m, n)$ if the first $q + 1$ rows and $p + 1$ columns of A are as in Example 3.1. In this case we must have

$$
m = 1 + q + \sum_{i=1}^{p} m_i + m_O \qquad \text{and} \qquad n = 1 + p + \sum_{j=1}^{q} n_j + n_O \tag{3.3}
$$

We say $\mathcal{S}$ is an $m\times n$ scaffold if m and n satisfy (3.3). We intend to search for extremal rectangle-free matrices by their scaffolds. Let $z(\mathcal{S})$ be the maximum weight of a rectangle-free matrix with scaffold $\mathcal{S}$.

Theorem 3.4. $z(m,n) = \max_{\mathcal{S}} z(\mathcal{S})$*, where the maximum is taken over all* $m\times n$ *scaffolds.*

Proof. Trivially every $z(\mathcal{S}) \leqslant z(m,n)$.

Let A be an extremal rectangle-free $m\times n$ matrix in lex-minimal form. By Lemmas 1.17 and 1.19, A has a (possibly improper) scaffold. We proceed by permuting rows and columns of A so that its scaffold is proper.

Let $p+1 = w(R_1)$; then by Lemma 1.18, every $w(R_i) \leqslant p+1$. Sort the first $p+1$ columns in descending weight order, then sort the rows in lex order. The first row is unchanged, and so the first $p+1$ columns are as required.

Let $q+1 = w(C_1)$; sort the first $q+1$ rows in descending weight order, then sort the columns in lex order. The first $p+1$ columns are unchanged, and so the first $q+1$ rows are as required.

Finally, let $m_i + 1 = w(C_{i+1})$ for $1 \leqslant i \leqslant p$, and $n_j + 1 = w(R_{j+1})$ for $1 \leqslant j \leqslant q$, and determine m_O and n_O by (3.3). The resulting scaffold $\mathcal{S}$ has $z(\mathcal{S}) \geqslant w(A) = z(m,n)$ by construction. □

3.1 Scaffold quotient matrices

Referring back to Example 1.14, the scaffold $p = q = 3$, $\mathbf{m} = \mathbf{n} = (3,3,3)$, $m_O = n_O = 0$ defines a natural partition on the rows and columns of the matrix, resulting in a block form for the remaining 9×9 minor.

Example 3.5. *Let* $I_3 = \begin{pmatrix} I & & \\ & I & \\ & & I \end{pmatrix}$, $J_3 = \begin{pmatrix} & I & \\ & & I \\ I & & \end{pmatrix}$ *and* $K_3 = \begin{pmatrix} & & I \\ I & & \\ & I & \end{pmatrix}$. *The scaffold quotient matrix of Example 1.14 is*

$$Q = \begin{pmatrix} I_3 & I_3 & I_3 \\ I_3 & J_3 & K_3 \\ I_3 & K_3 & J_3 \end{pmatrix}$$

where Q *is a* $p\times q$ *matrix, and its entries* $Q_{i,j}$ *are* $m_i\times n_j$ *blocks.*

To avoid confusion, we refer to the entries of Q as *blocks* rather matrices. If σ is a block, then its *height* is $a(\sigma) = \max \operatorname{dom} \sigma$ and its *width* is $b(\sigma) = \max \operatorname{ran} \sigma$. We will refer to the rows and columns of σ as *subrows* and *subcolumns*. Subrows and subcolumns of Q are the concatenations of the subrows and subcolumns of the blocks in a single row or column of Q respectively.

Note that we can have e.g. some $n_j = 0$; these need not be represented as columns in the scaffold quotient matrix. On the other hand, we could have e.g. $n_O > 0$; this corresponds with n_O additional columns of maximum block width 1. $m_i = 0$ and $m_O > 0$ are similar.

Therefore in general the scaffold quotient matrix is $\mu \times \nu$, where $\mu = \bar{p} + m_O$ and $\nu = \bar{q} + n_O$, and $\bar{p}$ and $\bar{q}$ are the numbers of non-zero entries in $\mathbf{m}$ and $\mathbf{n}$ respectively.

Let a_i and b_j be the maximum allowed block height in row i and block width in column j respectively:

$$a_i = \begin{cases} m_i & \text{for } i \leqslant \bar{p} \\ 1 & \text{for } i > \bar{p} \end{cases} \qquad \text{and} \qquad b_j = \begin{cases} n_j & \text{for } j \leqslant \bar{q} \\ 1 & \text{for } j > \bar{q} \end{cases} \tag{3.6}$$

Definition 3.7. *An abstract scaffold is a tuple $\mathcal{A} = (\mu, \nu, \mathbf{a}, \mathbf{b})$ where $\mu, \nu \in \mathbb{N}$, $\mathbf{a} \in \mathbb{N}^{\mu}$ and $\mathbf{b} \in \mathbb{N}^{\nu}$.*

The above defines a natural mapping $\mathcal{S} \mapsto \mathcal{A}$.

If we require the *expansion* $e(\mathcal{S}, Q) \in \mathrm{Mat}_{\{I,O\}}(m, n)$ of a scaffold $\mathcal{S}$ and scaffold quotient matrix Q to be rectangle-free, this imposes constraints on the blocks.

Theorem 3.8. *The expansion $e(\mathcal{S}, Q)$ is rectangle-free if and only if*

1. *Every block $Q_{i,j}$ has at most one I entry per subrow and per subcolumn, and*

2. *Every 2×2 minor of Q has a rectangle-free expansion.*

Proof. Let $A = e(\mathcal{S}, Q)$. By construction, either two or four of the I entries of a rectangle in A must be in $e(Q) \subset A$.

1. If the number is two, then the other two I entries in $e(\mathcal{S}) \subset A$ must be in the same row or column, and hence the two I entries in Q must be in the same subrow of the same column of Q, or the same subcolumn of the same row of Q respectively.

2. If the number is four, then by 1. they must be in distinct blocks forming a 2×2 minor of Q.

The converse is trivial. $\square$

A formal definition is now possible. For brevity we write *quotient matrix.*

Definition 3.9. *A quotient matrix for an abstract scaffold $\mathcal{A}$ (or a scaffold $\mathcal{S}$) is a matrix $Q \in \mathrm{Mat}_{\Sigma}(\mu, \nu)$ such that each block $Q_{i,j}$ has $a(Q_{i,j}) \leqslant a_i$ and $b(Q_{i,j}) \leqslant b_j$. The entry set $\Sigma \subseteq \mathrm{Mat}_{\{I,O\}}(\max a_i, \max b_j)$ consists of the blocks satisfying 3.8.1.*

A quotient matrix Q is rectangle-free if it satisfies 3.8.2.

Definition 3.10. *$z(\mathcal{A})$ is the maximum weight of a rectangle-free quotient matrix for the abstract scaffold $\mathcal{A}$.*

For convenience we use the same entry set $\Sigma \subseteq \mathrm{Mat}_{\{I,O\}}(\max a_i, \max b_j)$ for all entries, even those with smaller block dimensions. This is achieved by identifying e.g.

$$\begin{pmatrix} I & O & O \\ O & I & O \\ O & O & O \end{pmatrix} = \begin{pmatrix} I & O & O \\ O & I & O \end{pmatrix} = \begin{pmatrix} I & O \\ O & I \\ O & O \end{pmatrix}$$

since they are equal as subsets of $[3] \times [3]$, $[2] \times [3]$ and $[3] \times [2]$ respectively.

It follows from Lemma 3.8 that if a rectangle-free matrix A has a scaffold $\mathcal{S}$, then it also has a rectangle-free quotient matrix Q. We also have $w(A) = w(\mathcal{S}) + w(Q)$, where the *scaffold weight* $w(\mathcal{S})$ is the number of I entries in its expansion,

$$w(\mathcal{S}) = 1 + p + q + \sum_{i=1}^{p} m_i + \sum_{j=1}^{q} n_j \tag{3.11}$$

It follows that $z(\mathcal{S}) = w(\mathcal{S}) + \max w(Q)$ where the maximum is taken over rectangle-free quotient matrices Q for $\mathcal{A} \hookleftarrow \mathcal{S}$, and we can extend Theorem 3.4 to give

$$z(m,n) = \max_{\mathcal{S}} \Big\{ w(\mathcal{S}) + z(\mathcal{A}) \Big\} = \max_{\mathcal{S}} \Big\{ w(\mathcal{S}) + \max_{Q} w(Q) \Big\} \tag{3.12}$$

3.2 Connection with symmetric inverse semigroups

Identifying blocks with subsets of $[a] \times [b]$, a block has at most one I entry per subrow if and only if it is a partial function, and at most one I entry per subcolumn if and only if this partial function is injective.

Therefore, the entry set Σ of a quotient matrix can be identified with the set of partial injections $[a] \rightarrowtail\!\!\!\!\rightarrow [b]$. Furthermore, the rectangle-free property can be stated algebraically, using function composition. The following theorem resembles an observation of Montaron on finite projective planes. [Mon85]

Theorem 3.13. *A quotient matrix is rectangle-free if and only if it has no minor* $\left(\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right)$ *such that* $\gamma \circ \delta^\intercal \circ \beta \circ \alpha^\intercal$ *has a fixed point.*

Proof. Suppose a minor $\left(\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right)$ contains a rectangle given by subrow i of α, β, subrow k of γ, δ, subcolumn j of α, γ and subcolumn l of β, δ. Then $\alpha^\intercal(j) = i$, $\beta(i) = l$, $\delta^\intercal(l) = k$ and $\gamma(k) = j$, so j is a fixed point.

Conversely, if j is a fixed point of $\gamma \circ \delta^\intercal \circ \beta \circ \alpha^\intercal$ then let $i = \alpha^\intercal(j)$, $l = \beta(i)$ and $k = \delta^\intercal(l)$. We must have $\gamma(k) = i$, hence i, k and j, l define a rectangle as before. $\square$

Note that $\gamma \circ \delta^\intercal \circ \beta \circ \alpha^\intercal = (\gamma \circ \delta^\intercal) \circ (\alpha \circ \beta^\intercal)^\intercal$. Consequently, we can evaluate the compatibility relation in 3.8.2 by three applications of the mapping $(\sigma, \tau) \mapsto \sigma \circ \tau^\intercal$, which can be precomputed and cached in $\mathcal{O}\big(|\Sigma|^2\big)$ time and storage rather than the naïve $\mathcal{O}\big(|\Sigma|^4\big)$. This optimisation is necessary, as the fourth power of

$$|\Sigma| = \sum_{k=0}^{\min\{a,b\}} \binom{a}{k}\binom{b}{k} k!$$

would be infeasible even for small a, b.

If $a > b$ then in general $\sigma \circ \tau^{\intercal} \notin \Sigma$. In this case we can use a similar construction based on the mapping $(\sigma, \tau) \mapsto \sigma^{\intercal} \circ \tau$.

In the case $a = b$, the entry set is $\mathcal{I}_{[a]}$, the symmetric inverse semigroup acting on $[a]$. In general, Σ is a subset of the symmetric inverse semigroup acting on $[\max\{a, b\}]$. This connection may open up the Zarankiewicz problem to theoretical advances using algebraic approaches. For an overview of symmetric inverse semigroups, see [Lip96].

It is natural to generalise the Zarankiewicz problem to matrices with blocks as entries.

Definition 3.14. *$z(\mu, \nu \mid a, b)$ is the maximum weight of a rectangle-free $\mu{\times}\nu$ quotient matrix with entries in $[a] \rightarrowtail\!\!\!\!\rightarrow [b]$.*

Equivalently, $z(\mu, \nu \mid a, b) = z(\mathcal{A})$, where $\mathcal{A}$ is the $\mu{\times}\nu$ abstract scaffold with $\mathbf{a} = (a, \ldots, a)$ and $\mathbf{b} = (b, \ldots, b)$. We will investigate this generalisation further in Section 5.

3.3 Constraints and bounds on scaffolds

The number of $m{\times}n$ scaffolds[11] grows very quickly as a function of m, n. In this section we describe constraints and bounds which can be used to reject the vast proportion of scaffolds as candidates in (3.12), either by showing directly that $z(\mathcal{S}) < z(m, n)$, or that $z(\mathcal{S}) \leqslant z(\mathcal{S}')$ for another scaffold $\mathcal{S}'$. To avoid circular reasoning in the latter case, we require that $\mathcal{S}'$ precedes $\mathcal{S}$ in *scaffold order.*

Definition 3.15. *Scaffolds $\mathcal{S}, \mathcal{S}'$ are compared in scaffold order by $\mathcal{S}' <_{\mathrm{sc}} \mathcal{S}$ if*

- *$p' > p$, or*
- *$p' = p$ and $q' > q$, or*
- *…, $q' = q$, and $\sum_{i=1}^{p} m_i' > \sum_{i=1}^{p} m_i$, or*
- *…, $\sum_{i=1}^{p} m_i' = \sum_{i=1}^{p} m_i$, and $\mathbf{m}' >_{\mathrm{lex}} \mathbf{m}$, or*
- *…, $\mathbf{m}' = \mathbf{m}$, and $\sum_{j=1}^{q} n_j' > \sum_{j=1}^{q} n_j$, or*

[11] An exact formula for the number of $m{\times}n$ scaffolds is

$$\sum_{x=0}^{n-1}\sum_{y=0}^{m-1}\left(\sum_{N=0}^{n-x-1} p(x, y, N)\right)\left(\sum_{M=0}^{m-y-1} p(y, x, M)\right)$$

where $p(x, y, N)$ is the number of partitions of N into at most y parts each of size at most x.

- $\ldots$, $\sum_{j=1}^{q} n'_j = \sum_{j=1}^{q} n_j$, *and* $\mathbf{n}' >_{\text{lex}} \mathbf{n}$.

This is a total order, as m_O and n_O are determined by (3.3). We will not necessarily search for quotient matrices of scaffolds in this order.

Proposition 3.16. *Let r and c be minimum row and column weights respectively, as in Constraint 2.5. If $\mathcal{S} = (p, q, \mathbf{m}, \mathbf{n}, m_O, n_O)$ is a candidate in Theorem 3.4, we can require*

1. *$p + 1 \geqslant r$ and every $n_j + 1 \geqslant r$, similarly $q + 1 \geqslant c$ and every $m_i + 1 \geqslant c$.*
2. *(a) If $m = n$, then $p \geqslant q$.*
   (b) If $m = n$ and $p = q$, then $\sum_{i=1}^{p} m_i \geqslant \sum_{j=1}^{q} n_j$.
   (c) If $m = n$, $p = q$, and $\sum_{i=1}^{p} m_i = \sum_{j=1}^{q} n_j$, then $\mathbf{m} \geqslant_{\text{lex}} \mathbf{n}$.
3. *$m_O = 0$ or every $m_i > 0$, similarly $n_O = 0$ or every $n_j > 0$.*
4. *If $p \geqslant 2$ and $q \geqslant 2$, then either every $m_i > 0$ or every $n_j > 0$.*

Proof.

1. Trivial.
2. If not, then $\mathcal{S}^{\intercal} = (q, p, \mathbf{n}, \mathbf{m}, n_O, m_O)$ precedes $\mathcal{S}$ in scaffold order.
3. If not, form a new $m \times n$ scaffold $\mathcal{S}'$ by replacing the first $m_i = 0 \mapsto 1$ and decreasing m_O by 1. Then $w(\mathcal{S}') = w(\mathcal{S}) + 1$, but both $\mathcal{S}$ and $\mathcal{S}'$ map to the same abstract scaffold, so they have the same $\max_Q w(Q)$. Therefore $z(\mathcal{S}) < z(\mathcal{S}')$.
4. If not, then given a matrix achieving $z(\mathcal{S})$, delete the I entries at $(1,\ p{+}1)$ and $(q{+}1,\ 1)$ and insert I entries at $(2,\ p + 1)$, $(q + 1,\ 2)$ and $(q + 1,\ p + 1)$ as below.

$$\left[\begin{array}{ccc|c} I & I & I* & \\ I & & & \cdots \\ I* & & & \\ \hline & \vdots & & \ddots \end{array}\right] \mapsto \left[\begin{array}{ccc|c} I & I & & \\ I & & I* & \cdots \\ & I* & I* & \\ \hline & \vdots & & \ddots \end{array}\right]$$

   The affected entries are labelled $*$. This yields an $m \times n$ matrix of weight $z(\mathcal{S}) + 1$ which is rectangle-free, as R_{q+1} and C_{p+1} are otherwise empty. $\square$

Proposition 3.17. *The following are upper bounds for $z(\mathcal{S})$.*

1. *$(m - q)(p + 1) + q + \sum_{j=1}^{q} n_j$.*
2. *(a) $w(\mathcal{S}) + z(m - q - 1,\ n - p - 1)$.*
   (b) $1 + p + q + \sum_{i=1}^{p} m_i + z(m - 1,\ n - p - 1)$, and the transpose.
3. *Bound 2.14 with rows of weight $p + 1$ and each $n_i + 1$, and the transpose.*
4. *$w(\mathcal{S}) + \sum_{i=1}^{\mu} \sum_{j=1}^{\nu} \min\{a_i, b_j\}$.*

Proof.

1. If a rectangle-free matrix A with scaffold $\mathcal{S}$ has greater weight, it must have a row of weight $> p+1$. The lex-minimal form of A has a scaffold $\mathcal{S}'$ with $p' > p$, so $\mathcal{S}' <_{\mathrm{sc}} \mathcal{S}$. We also have $z(\mathcal{S}') \geqslant w(A) = z(\mathcal{S})$.

2. Follows directly from Bound 2.6.

3. Follows directly from Bound 2.14.

4. Follows directly from (2.4), using $w(A) = w(\mathcal{S}) + w(Q)$. □

The bound in Proposition 3.17.4 is effective for eliminating scaffolds with small p, q and large m_i, n_j, but overestimates the potential weight for scaffolds with large p, q and small m_i, n_j. If the abstract scaffold $\mathcal{A} \leftarrow\!\shortmid \mathcal{S}$ has a $c \times d$ minor of 1×1 blocks, then the upper bound given by Proposition 3.17.4 for the weight of this minor is cd. This bound could easily be improved to $z(c, d)$. We can apply this reasoning to other block sizes.

Definition 3.18. *If $\mathcal{A} = (\mu, \nu, \mathbf{a}, \mathbf{b})$ is an abstract scaffold with c rows of block height a and d columns of block width b, then*

1. *The band $\mathcal{A}|^a$ is the abstract scaffold $\big(c,\ \nu,\ (a, \ldots, a), \mathbf{b}\big)$.*

2. *The stack $\mathcal{A}|_b$ is the abstract scaffold $\big(\mu,\ d,\ \mathbf{a},\ (b, \ldots, b)\big)$.*

3. *The box $\mathcal{A}|^a_b$ is the abstract scaffold $\big(c,\ d,\ (a, \ldots, a),\ (b, \ldots, b)\big)$.*

If Q is a quotient matrix for $\mathcal{A}$, then $Q|^a$, $Q|_b$ and $Q|^a_b$ are the corresponding minors.

We now give upper bounds for $z(\mathcal{A})$ using bands, stacks and boxes, and hence upper bounds for $z(\mathcal{S}) = w(\mathcal{S}) + z(\mathcal{A})$.

Proposition 3.19. *Let $\mathcal{A}$ be an abstract scaffold. Then*

1. $\displaystyle\sum_a \sum_b z(\mathcal{A}|^a_b)$ *2.* $\displaystyle\sum_a z(\mathcal{A}|^a)$ *3.* $\displaystyle\sum_b z(\mathcal{A}|_b)$

are upper bounds for $z(\mathcal{A})$, where the sums range over block heights and widths.

Proof. Given an extremal rectangle-free quotient matrix Q for $\mathcal{A}$, we have

$$z(\mathcal{A}) = w(Q) = \sum_a \sum_b w(Q|^a_b) \leqslant \sum_a \sum_b z(\mathcal{A}|^a_b)$$

2. and 3. are similar. □

Note that $z(\mathcal{A}|_b^a) = z(c, d \mid a, b)$. These values can be precomputed and cached using the algorithm in Section 3.5. In contrast, the bounds on bands and stacks are unlikely to already be in the cache, and are the most computationally expensive, so they should only be computed if other bounds do not eliminate the scaffold first.

If $\mathbf{a}$ or $\mathbf{b}$ is (almost) uniform, then $Q|^a$ or $Q|_b$ respectively may be (almost) the whole quotient matrix. In this case, for efficiency reasons we do not compute exact values for 2. or 3.[12]

Since any quotient matrix with only one row is necessarily rectangle-free, (2.4) gives an exact upper bound for a band of one row. The same applies for stacks of one column.

Finally, we give a bound on rows and columns which is computationally cheap, and may be applied in Propositions 3.17.1 and 3.19.

Proposition 3.20. *If $(\mathcal{S}, Q)$ is a candidate in Theorem 3.4 and $m_i = q$, then we can assume that $w(R_i) \leqslant \sum_{j=1}^{q} n_j$. Similarly for columns with $n_j = p$.*

Proof. Suppose $A = e(\mathcal{S}, Q)$ is an extremal solution for $z(\mathcal{S})$, and the q subrows of R_i of Q have weights $r_1, \ldots, r_q$. Form a new matrix A' by swapping the corresponding q rows of A with rows $2, \ldots, q+1$ of A, then sorting them in descending weight order, as in the example below. Finally, sort the columns in lex order.

$$\begin{array}{r} \\ \mathbf{n}=(1,1)\left\{\begin{array}{c} \\ \\ \end{array}\right. \\ \mathbf{r}=(1,2)\left\{\begin{array}{c} \\ \\ \end{array}\right. \\ \\ \end{array}
\begin{array}{|ccc|c|c|c|c|} \hline I & I & I & & & & \\ I & & & I & & & \\ I & & & & I & & \\ \hline & I & & I & & & \\ & I & & & I & I & \\ \hline & & \vdots & & \vdots & & \ddots \\ \hline \end{array}
\quad \mapsto \quad
\begin{array}{r} \\ \mathbf{r}'=(2,1)\left\{\begin{array}{c} \\ \\ \end{array}\right. \\ \left\{\begin{array}{c} \\ \\ \end{array}\right. \\ \\ \end{array}
\begin{array}{|ccc|c|c|c|c|} \hline I & I & I & & & & \\ & I & & & I & I & \\ & I & & I & & & \\ \hline I & & & I & & & \\ I & & & & I & & \\ \hline & & \vdots & & \vdots & & \ddots \\ \hline \end{array}$$

The resulting matrix A' has scaffold $\mathcal{S}'$ formed by replacing $\mathbf{n}$ in $\mathcal{S}$ with the sorted $\mathbf{r}'$. We also have $z(\mathcal{S}') \geqslant w(A') = w(A) = z(\mathcal{S})$. If $w(R_i) = \sum_{k=1}^{q} r_k > \sum_{j=1}^{q} n_j$ then $\mathcal{S}' <_{\mathrm{sc}} \mathcal{S}$, so $\mathcal{S}$ can be rejected. The result for columns with $n_j = p$ is similar. $\square$

3.4 Quotient-weightlex-minimal form

As before, we define an ordering on solutions, then show results about the minimal form which can be used to define effective constraints and bounding functions.

Definition 3.21. *Quotient matrices Q, Q' for the same abstract scaffold $\mathcal{A}$ are compared in quotient-weightlex order by $Q <_{\mathrm{qu}} Q'$ if, on the first row in which they differ,*

- *$w(R_i) > w(R_i')$, or*

[12] In fact, since not all of the constraints in Section 3.5 can be imposed on individual bands or stacks, computing exact value of $z(\mathcal{A}|^a)$ or $z(\mathcal{A}|_b)$ may even be *slower* than computing $z(\mathcal{S})$. However, in many cases the upper bound given by 2. or 3. is enough to eliminate $\mathcal{S}$ without search; we use a heuristic to estimate which option is more efficient.

- *$w(R_i) = w(R'_i)$ and R_i has fewer pairs than R'_i of subrows not in descending weight order, or*
- *…, R_i and R'_i have equally many pairs of subrows not in descending weight order, and $R_i <_{\mathrm{lex}} R'_i$ where the entry set $\Sigma \subseteq \mathrm{Mat}_{\{I,O\}}(\max a_i, \max b_j)$ is ordered by $<_{w\mathrm{lex}}$.*

The *quotient-weightlex-minimal form* of a quotient matrix is the $<_{\mathrm{qu}}$-minimal member of its orbit under permutations of rows of equal block height, columns of equal block width, subrows of the same row and subcolumns of the same column, and if $\mathcal{A}^\intercal = \mathcal{A}$, transposition of the quotient matrix and its blocks.

Let Q be a quotient matrix in quotient-weightlex-minimal form for an abstract scaffold $\mathcal{A}$.

Lemma 3.22.

1. *Rows of Q of equal block height are in weightlex order.*
2. *Columns of Q of equal block width are in lex order.*
3. *Subrows of the same row of Q are in weightlex order.*
4. *Subcolumns of the same column of Q are in lex order.*

Proof.

1. Consider rows R_i, R_k of Q for $i < k$.
   (a) If $w(R_i) < w(R_k)$, then swapping the rows yields a row i of greater weight; a contradiction.
   (b) If $w(R_i) = w(R_k)$ but $R_i >_{\mathrm{lex}} R_k$, then swapping the rows does not change the weight of any row or subrow, but yields a row i lower in lex order; a contradiction.
2. If not, then let i be the first row in which the two columns of Q differ. Swapping the columns does not change the weight of any row or subrow, and does not change any rows above i, but yields a row i lower in lex order; a contradiction.
3. Consider subrows R_i, R_k in the expansion of a row of Q for $i < k$.
   (a) If $w(R_i) < w(R_k)$, then swapping the subrows does not change the row weight, but leaves fewer pairs of subrows not in descending weight order; a contradiction.
   (b) Let j be the column in Q of the first block in which R_i, R_k differ. If $w(R_i) = w(R_k)$ but $R_i >_{\mathrm{lex}} R_k$, then swapping the subrows does not change weight of the row or any subrow, and leaves blocks before j unchanged, but yields a block j of equal weight lower in lex order; a contradiction.
4. Consider subcolumns C_j, C_l in the expansion of a column of Q for $j < l$. If $C_j >_{\mathrm{lex}} C_l$, let i be the row in Q of the first block in which C_j, C_l differ. Swapping the subcolumns does not change the weight of any row, subrow, and leaves blocks above i unchanged, but yields a block i of equal weight lower in lex order; a contradiction. □

Lemma 3.23. *If subrows of the first row of Q have equal weight, then $\pi, \pi' \in \{I, O\}^\nu$ are in lex order, where π_j, π'_j indicate whether each subrow has an I entry in the block $Q_{1,j}$.*

Proof. If not, let j be the first column with $\pi_j = O$ and $\pi'_j = I$. Swap these subrows, then swap subcolumns of Q to revert any changes made to blocks before j in the first row, as in the example below.

$$\begin{array}{r} \pi = (I, O, \ldots) \\ \pi' = (I, I, \ldots) \\ \\ \\ \end{array}
\begin{array}{|cc|c|c|} \hline I & & & \cdots \\ & I & I & \\ \hline I & & \ddots & \ddots \\ \hline & \vdots & \ddots & \ddots \\ \hline \end{array}
\mapsto
\begin{array}{|cc|c|c|} \hline & I & I & \cdots \\ I & & & \\ \hline I & & \ddots & \ddots \\ \hline & \vdots & \ddots & \ddots \\ \hline \end{array}
\mapsto
\begin{array}{|cc|c|c|} \hline I & & I & \cdots \\ & I & & \\ \hline & I & \ddots & \ddots \\ \hline & \vdots & \ddots & \ddots \\ \hline \end{array}$$

This does not change the weight of the first row or any of its subrows, and leaves blocks before j in the first row unchanged, but yields a block $Q_{1,j}$ of equal weight lower in lex order; a contradiction. □

3.5 Quotient matrix algorithm

The algorithm for computing $z(\mathcal{A})$ is a specific instance of the model in Section 2.1, using

- Constraints 3.24, 3.25, 3.26, 3.27, and 3.28,
- The bounding function 3.29, and
- The guess function G_{lex} as in Definition 2.12.

The initial partial solution P has $\sigma \in P_{i,j}$ only if $a(\sigma) \leqslant a_i$ and $b(\sigma) \leqslant b_j$. As before, the procedures enforcing the constraints should delete possible entries from P as appropriate.

Constraint 3.24. *If (i, j) is fresh then for each $2{\times}2$ minor including (i, j) with three confirmed entries, delete all incompatible $\sigma \in \Sigma$ from the remaining entry, as in Theorem 3.13.*

For rows, $\overline{w}$ and $\underline{w}$ are as in (2.4), and for subrows we use a similar definition.

Constraint 3.25. *If R_i, R_{i+1} are consecutive rows with a fresh entry, and $a_i = a_{i+1}$, then require $\overline{w}(R_i) \geqslant \underline{w}(R_{i+1})$. If this is an equality, then require $R_i \leqslant_{\text{lex}} R_{i+1}$.*

Constraint 3.26. *If R_i, R_{i+1} are consecutive subrows of a row with a fresh entry, then require $\overline{w}(R_i) \geqslant \underline{w}(R_{i+1})$. If this is an equality, then require $R_i \leqslant_{\text{lex}} R_{i+1}$, and if also $i = 1$, also require $\pi \leqslant_{\text{lex}} \pi'$ as in Lemma 3.23.*

Constraint 3.27. *If C_j, C_{j+1} are consecutive columns with a fresh entry, and $b_j = b_{j+1}$, then require $C_j \leqslant_{\text{lex}} C_{j+1}$.*

Constraint 3.28. *If C_j, C_{j+1} are consecutive subcolumns of the same column with a fresh entry, then require $C_j \leqslant_{\text{lex}} C_{j+1}$.*

Proof. 3.24 is similar to 2.13, and 3.25–3.28 follow from Lemma 3.22. □

Bound 3.29. *If P is a partial solution for an extremal quotient matrix Q, then*

$$w(Q) \leqslant \overline{w}(P) = \min\left\{\sum_a \overline{w}_{\text{band}}(P|^a),\quad \sum_b \overline{w}_{\text{stack}}(P|_b)\right\}$$

where

1. *$\overline{w}_{\text{box}}(P|^a_b)$ is the minimum of $z(c, d \mid a, b)$ or an upper bound for it, (2.4), and Bound 2.6 with $W(m, n) = z(m, n \mid a, b)$,*
2. *$\overline{w}_{\text{band}}(P|^a)$ is the minimum of $z(\mathcal{A}|^a)$ or an upper bound for it, $\sum_b \overline{w}_{\text{box}}(P|^a_b)$, and Bound 2.9 on the rows, and*
3. *$\overline{w}_{\text{stack}}(P|_b)$ is the minimum of $z(\mathcal{A}|_b)$ or an upper bound for it, $\sum_a \overline{w}_{\text{box}}(P|^a_b)$, and Bound 2.14 on the subrows.*

Proof. Follows directly from Proposition 3.19. □

If $\mathcal{A}$ is a box (i.e. all blocks have the same dimensions) then the problem is hereditary, so we can also use the upper bounds in Proposition 2.1, and impose Constraint 2.5.

If $\mathcal{A} \hookleftarrow \mathcal{S}$ for a candidate $\mathcal{S}$ in Theorem 3.4, then we can also use bounds for $z(\mathcal{S})$ as in Section 3.3, and the following constraints. If $\mathcal{A}$ is a band or a stack in Proposition 3.19, we impose the applicable constraints to its rows or columns respectively.

Constraint 3.30. *If R_i has a fresh entry and $a_i = q$, require $\underline{w}(R_i) \leqslant \sum_{j=1}^{q} n_j$. If this is an equality and R_i is complete, then for the vector $\mathbf{r}$ of subrow weights of R_i, require $\mathbf{r} \leqslant_{\text{lex}} \mathbf{n}$.*

If C_j has a fresh entry and $b_j = p$, require $\underline{w}(C_j) \leqslant \sum_{i=1}^{p} m_i$.

Proof. As in Proposition 3.20. By Lemma 3.22.3 we do not need to sort $\mathbf{r}$. We do not require $\mathbf{c} \leqslant_{\text{lex}} \mathbf{m}$ for the (sorted) subcolumn weights, as columns of P are usually incomplete. □

Constraint 3.31. *Suppose $P_{i,j} = \sigma$ is fresh and $\sigma_{k,l} = I$. Let R_k and C_l be the corresponding subrow and subcolumn of Q, and let $\varepsilon, \varepsilon' \in \{0, 1\}$ indicate whether $e(\mathcal{S})$ has I entries in the corresponding row and column respectively.*[13] *We require*

1. *$\overline{w}(R_k) + \varepsilon \geqslant r$ and $\overline{w}(C_l) + \varepsilon' \geqslant c$, where r, c are minimum row and column weights respectively for $e(\mathcal{S}, Q)$ as in Constraint 2.5,*
2. *If $n_l = p$ then $\underline{w}(C_l) \leqslant q$, and*
3. *$\underline{w}(R_k) + \varepsilon \leqslant p + 1$, and if this is an equality, $\underline{w}(C_l) + \varepsilon' \leqslant q + 1$.*

[13] ε may vary on the band $a = 1$, and the bound is stricter when $\varepsilon = 1$. The subrow weights excluding ε are in descending order, so we should impose $\varepsilon = 1$ on the last m_O subrows, not the first. This occurs naturally when we reverse $\mathbf{a}$ and $\mathbf{b}$ anyway.

4. *If $n = m$ then require $\underline{w}(C_l) + \varepsilon' \leqslant p + 1$.*

Proof. 1. is trivial.

For 2. and 3., if there is an extremal rectangle-free $A \in \mathrm{Mat}_{\{I,O\}}(m, n)$ with a row of weight $> p+1$, or a column of weight $> q+1$ intersecting a row of weight $p+1$, then after permuting rows and columns it has a scaffold preceding $\mathcal{S}$ in scaffold order.

For 4., permuting rows and columns of $A^\intercal$ would yield a matrix with a scaffold preceding $\mathcal{S}$ in scaffold order. □

We also use Bound 2.14 on the expansion, and apply Proposition 3.20 in Bound 3.29.

The upper bounds in Proposition 3.19 still tend to overestimate the potential weight of quotient matrices with many small blocks, as individual bands and stacks can typically afford greater box weights than the full quotient matrix can. Most such scaffolds are fruitless. To alleviate this, we reverse **a** and **b** to put blocks of least height and width at the upper left of Q, where G_{lex} tries them first so that the bounding functions give sharper bounds. The results here do not depend on this ordering, so long as rows of equal a_i and columns of equal b_j respectively are consecutive.

3.6 Scaffold algorithm

In this section we describe the specific algorithm for the case $s = t = 2$, based on Theorem 3.4. The only parameters are m and n.

1. Get a lower bound L and an upper bound U using the results in Section 1.2. If an exact value is given directly, or $L = U$, then terminate. Otherwise, increment L by 1.[14]
2. Generate the set of candidate $m\times n$ scaffolds satisfying the constraints of Proposition 3.16, and for which the least upper bound U' given by Propositions 3.17 and 3.19.1 has $U' \geqslant L$. If $m \neq n$, similarly generate the set of $n\times m$ scaffolds.
3. If either set of candidate scaffolds is empty, terminate.
4. Choose a candidate scaffold $\mathcal{S}$, and delete it from the set it was taken from.
5. Compute upper bounds as in Propositions 3.19.2 and 3.19.3, and update U' accordingly. If now $U' < L$, reject $\mathcal{S}$ and repeat from 3.
6. Search for an extremal quotient matrix Q for $\mathcal{S}$ using the algorithm in Section 3.5, with lower bound L and upper bound $\min\{U, U'\}$. If no solution with $w(\mathcal{S}) + w(Q) \geqslant L$ is found, repeat from 3.
7. Output $e(\mathcal{S}, Q)$.

[14] As before, there is no disadvantage in doing this; if no solution is found, the result will be the original L, and the lower bounds used are constructive.

8. Let $L = w(\mathcal{S}) + w(Q) + 1$, and delete any scaffolds the set(s) of candidates according to the new lower bound L.

9. If $L \leqslant U$, repeat from 3.

In 4. we use two heuristics to choose $\mathcal{S}$: if $m \neq n$ we use an estimate of running time to choose from the set of candidate scaffolds which can be exhausted faster, and we choose a scaffold minimising $\mathrm{VAR}(q, m_1, \ldots, m_p) + \mathrm{VAR}(p, n_1, \ldots, n_q)$, as we expect an extremal solution to have approximately uniform row and column weight distributions.

If $n \neq m$ we can impose one further constraint in the algorithm of Section 3.5.

Constraint 3.32. *Let c be the greatest $p+1$ of a scaffold in the set of candidates from which $\mathcal{S}$ was* not *taken. Suppose $P_{i,j} = \sigma$ is fresh and $\sigma_{k,l} = I$. Let C_l be the corresponding subcolumn of Q, and $\varepsilon' \in \{0, 1\}$ indicate whether $e(\mathcal{S})$ has an I entry in the corresponding column. Then we require $\underline{w}(C_l) + \varepsilon' \leqslant c$.*

Proof. Let A be an $m \times n$ rectangle-free matrix with a column of weight $> c$. After permuting rows and columns of $A^\intercal$ we find a matrix of equal weight with a scaffold $\mathcal{S}'$ such that p' is greater than of any candidate in the other set. $\mathcal{S}'$ has already been eliminated as a candidate, and precedes all remaining $n \times m$ candidates in scaffold order, so A is not extremal. $\square$

4 Computational results and analysis

In this section we present our results, analyse the efficiency of the algorithms described in Sections 2.4 and 3.6, and compare with results of other researchers.

4.1 New Zarankiewicz numbers

As the case $s = t = 2$ is of most interest, we report the following new values, all of which were found by the scaffold algorithm on single-processor computers. In total we found 213 new values for $s, t \leqslant 4$ using the two algorithms. We give full tables in Appendix A.

$z(m, n)$

$n \backslash m$	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
21	–	–	85	–	–	–	–	–	=
22	–	83	87	91	96	–	–	108	=
23	–	85	89	93	98	103	108	110	115
24	82	87		96	100	105			118

The most comprehensive table of values we found for $z(m, n)$ was published by Damásdi et al. in [DHS13]. Tables for $z(m, n;\ s, t)$ with $s, t \leqslant 4$ were published by Guy in [Guy68] and [Guy69]. To our knowledge, these tables include all Zarankiewicz numbers for $s, t \leqslant 4$

reported prior to this paper.[15] We report a value as new if it is not found in these tables, unless the value would likely have been reported were it within the table's range for m, n. We also found 12 previously unreported errors in Guy's tables, which are corrected in Appendix A.

To illustrate the number and nature of new values found, for each s, t we plot the set of m, n for which $z(m, n;\ s, t)$ was previously reported, and overlay the boundary of the set of values we computed. Unshaded areas within this boundary represent new values.

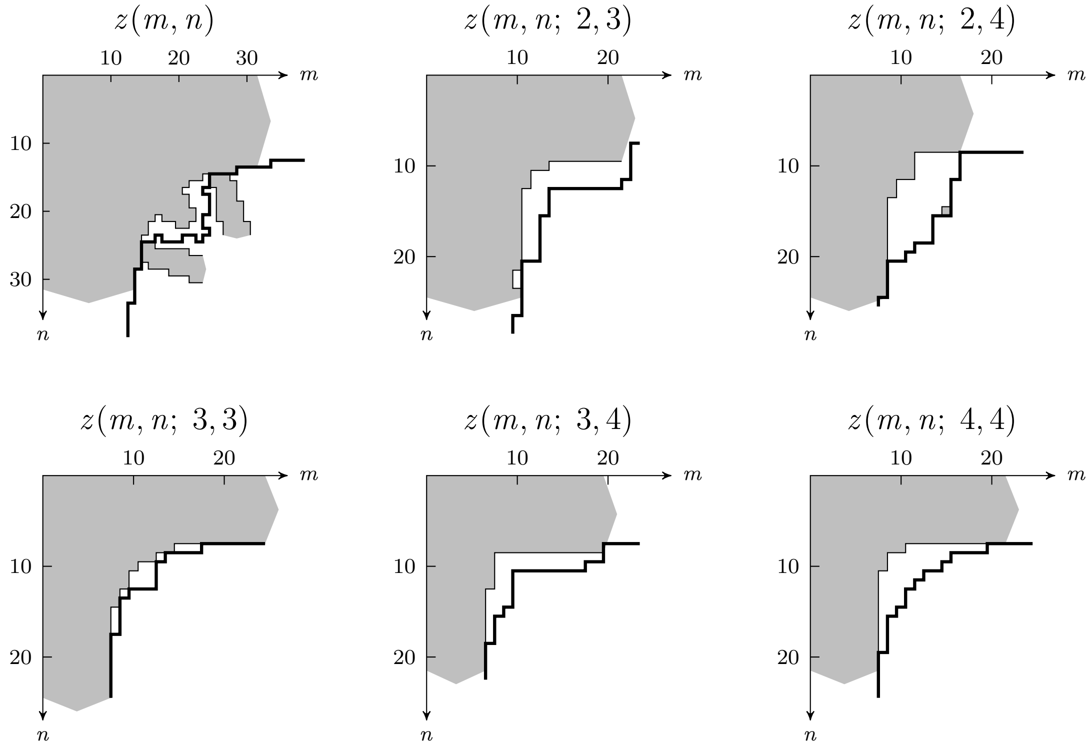

Figure 4.1: Computed values vs. those reported prior to this paper

Although Figure 4.1 gives an indication of our algorithms' feasible regions, we did not allocate CPU time across the possible parameters m, n, s, t in a systematic way. Also, minor variations had an enormous effect on feasibility; in one startling example, $z(12, 11;\ 2, 3)$ did not finish within one day, yet its transpose $z(11, 12;\ 3, 2)$ took under one second. It is therefore very likely that more values could be found within reasonable running times using essentially the same algorithms.

4.2 Efficiency

We measured the running time of each algorithm in the case $s = t = 2$ on a single 1.9 GHz processor. Due to the extensive use of recursive bounds, the running time to compute $z(m, n)$ depends on whether the cached values for smaller m, n are exact, or are themselves bounds.

[15] With the exception of $z(15, 15) = 61$, which was incorrect in Guy's tables, but not corrected in Damásdi et al.'s. The correct value appears in [DDR13].

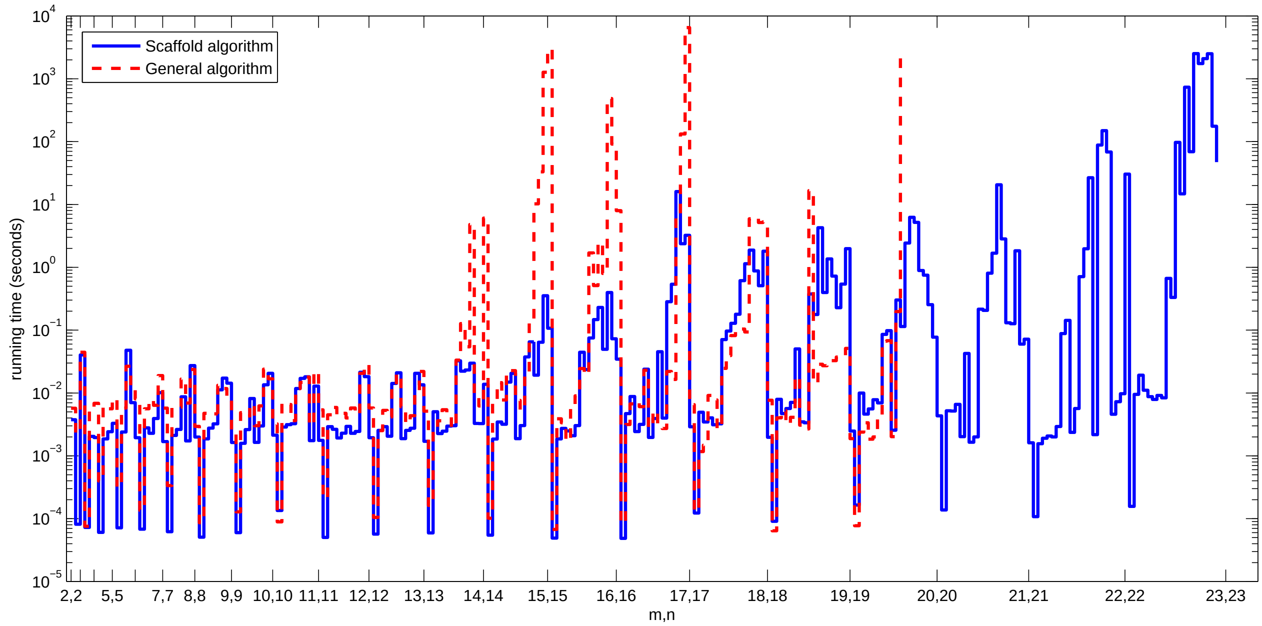

Figure 4.2: Running time to compute $z(m, n)$

We give running times using exact values for this purpose, so it is appropriate to also show the cumulative running time required to compute $z(m', n')$ for all $m' \leqslant m$ and $n' \leqslant n$.

In a few individual cases where the lower bound L equals the upper bound U, no search is required, and hence the running time is negligible. In general the sharpness of L has a large effect on the scaffold algorithm's efficiency, due to the number of candidate scaffolds, and the likelihood of searching scaffolds with no good solution. The effect on the general algorithm is mitigated by the likelihood of finding a good solution quickly. It would be possible to improve L by trying to insert additional rows and/or columns into solutions for smaller m, n.

4.3 Comparison with other computational approaches

Steinbach & Posthoff described an algorithm for $z(m, n)$ which recursively generates all rectangle-free matrices of an increasing weight. Their algorithm uses a *ternary vector* data structure to distinguish confirmed and unconfirmed entries, but does not use upper bounds or symmetry reduction. [SP12] Later they extended their algorithm with some symmetry reduction techniques, and were able to compute $z(10, 10)$ in ∼7 hours on a single 2.93 GHz processor. [SP14]

Werner described a backtracking algorithm for $z(m, n)$ which iteratively increments complete rows in a way which avoids rectangles. An upper bound resembling Bound 2.6 is used to reject partial solutions, and symmetry reduction is implemented by generating rows in lexicographic order. This algorithm computed $z(10, 10)$ in ∼18 minutes, and $z(11, 11)$ in ∼172 hours, on a single 2.67 GHz processor; a heuristic variant successfully found $z(12, 12)$

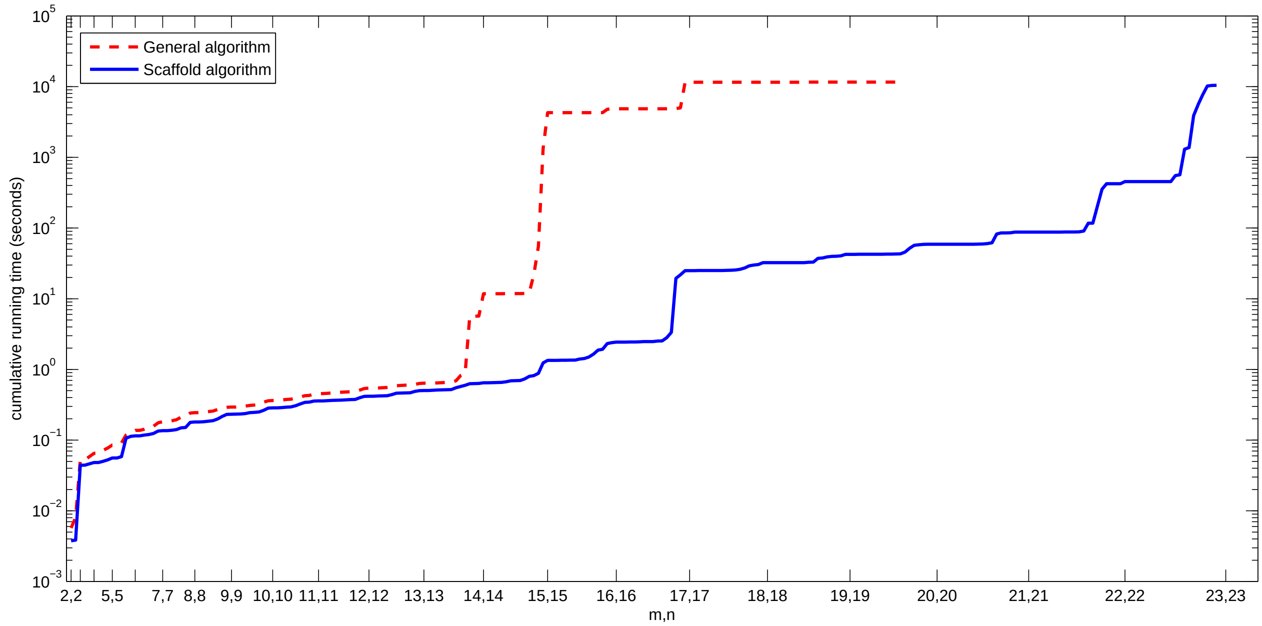

Figure 4.3: Cumulative running time to compute $z(m, n)$

in ∼12.5 hours. [Wer12]

Dybizbański et al. used the software package `nauty` [MP14] to compute $z(n, n)$ up to $n = 16$, but did not give details of this approach other than that it was infeasible for $n > 16$. [DDR13]

In comparison, on a single 1.9 GHz processor our general algorithm computes $z(m, n)$ for *every* $m, n \leqslant 16$ in ∼1.35 hours, or *every* $m, n \leqslant 19$ in ∼3.2 hours; the scaffold algorithm achieves these results in ∼2.5 seconds and ∼42.5 seconds respectively. Therefore, both of our algorithms improve on previous work in the case $s = t = 2$ by several orders of magnitude.

We did not find other computational approaches for general s, t.

5 Observations and questions

The generalisation of the Zarankiewicz problem to matrices with entries in $[a] \rightarrowtail\!\!\!\rightarrow [b]$ opens up further connections with other related combinatorial problems. We demonstrate the depth of the "block Zarankiewicz problem" with some examples of such connections.

Definition 5.1. *For an entry set* $\Sigma \subseteq [a] \rightarrowtail\!\!\!\rightarrow [b]$*,*

1. *We say* $(m, n;\ s, t)$ *is* Σ*-completable if there is a matrix in* $\mathrm{Mat}_\Sigma(m, n)$ *with an* (s, t)*-rectangle-free expansion.*
2. *In this case we define* $z(m, n;\ s, t \mid \Sigma)$ *as the maximum weight of an* (s, t)*-rectangle-free matrix in* $\mathrm{Mat}_\Sigma(m, n)$*.*

Trivially, if $O \in \Sigma$ then every $(m, n;\ s, t)$ is Σ-completable. As before, for brevity we omit s, t in the case $s = t = 2$. The following definition will also be useful.

Definition 5.2. *For $T \subseteq \mathbb{N}^2$, we define $z(m, n;\ T)$ as the maximum weight of a matrix in $\mathrm{Mat}_{\{I,O\}}(m, n)$ which is (s, t)-rectangle-free for all $(s, t) \in T$.*

Values of $z(m, n;\ T)$ can be computed with an algorithm similar to that in Section 2.4, using instances of the relevant constraints and bounds for each $(s, t) \in T$.

5.1 Rectangle-free coloring

The bipartite Ramsey number $b_k(s)$ is the minimum $n \in \mathbb{N}$ such that every k-edge-coloring of the complete bipartite graph $K_{n,n}$ has a monochromatic $K_{s,s}$ subgraph. [DDR13]

This problem of Ramsey theory is equivalent to a multicolor version of the Zarankiewicz problem, by identifying k-edge-colourings of $K_{n,n}$ as matrices in $\mathrm{Mat}_{[k]}(n, n)$. An edge-coloring has a monochromatic $K_{s,s}$ if and only if the corresponding matrix has a monochromatic $s \times s$ minor. The multicolor Zarankiewicz problem has been studied e.g. in [DDR13, FGGP10, GHO00]. We say $(m, n;\ s, t)$ is k-colorable if there is a matrix in $\mathrm{Mat}_{[k]}(m, n)$ with no monochromatic $s \times t$ minor.

Identifying $[k]$ with $\Delta_k = \{\delta_1, \ldots, \delta_k\}$ where each $\delta_i = \{(i, i)\}$, we see that a matrix in $\mathrm{Mat}_{[k]}$ has a monochromatic $s \times t$ minor if and only if the corresponding matrix in Mat_{Δ_k} has an (s, t)-rectangle. It follows that $(m, n;\ s, t)$ is k-colorable if and only if it is Δ_k-completable.

In the case $s = t = 2$, we use this notation to restate and generalise some definitions given by Fenner, Gasarch, Glover & Purewal in [FGGP10].

Definition 5.3.

1. *A half-mono rectangle is a 2×2 minor $\left(\begin{smallmatrix} \alpha & \beta \\ \gamma & \delta \end{smallmatrix}\right)$ with $\alpha = \gamma$ and $\beta = \delta$.*
2. *(m, n) is strongly-k-colorable if there is a matrix in $\mathrm{Mat}_{[k]}(m, n)$ with no half-mono rectangle.*

Identifying $[k]$ with $\Gamma_k = \{\gamma_1, \ldots, \gamma_k\}$ where $\gamma_i = \{(1, i)\}$, we see that a matrix in $\mathrm{Mat}_{[k]}$ has a half-mono rectangle if and only if the corresponding matrix in Mat_{Γ_k} has a rectangle. It follows that (m, n) is strongly-k-colorable if and only if it is Γ_k-completable.

The *obstruction set* OBS_k is the set of pairs[16] (m, n) which are not k-colorable, but for which (m', n') is k-colorable whenever $[m'] \times [n']$ is a proper subset of $[m] \times [n]$. It is natural to extend this definition to other entry sets.

Definition 5.4. $\mathrm{OBS}(\Sigma)$ *is the set of pairs (m, n) which are not Σ-completable, but for which (m', n') is Σ-completable whenever $[m'] \times [n']$ is a proper subset of $[m] \times [n]$.*

[16] Fenner et al.'s obstruction sets actually contain the sets $[m] \times [n]$ rather than the pairs (m, n).

Equivalently, (m', n') is Σ-completable if and only if $[m'] \times [n']$ is not a superset of any $[m] \times [n]$ with $(m, n) \in \mathrm{OBS}(\Sigma)$.

We have seen that $\mathrm{OBS}_k = \mathrm{OBS}(\Delta_k)$. We also note that when (m, n) is Δ_k-completable, we have $z\big(m, n \,\big|\, \Delta_k \cup \{O\}\big) = z\big(m, n;\ \mathrm{OBS}(\Delta_k)\big) = mn$. We therefore ask when this is true for other (m, n).

Question 5.5. *For which m, n, k is $z\big(m, n \,\big|\, \Delta_k \cup \{O\}\big) = z\big(m, n;\ \mathrm{OBS}(\Delta_k)\big)$?*

One direction is trivial: given an extremal $Q \in \mathrm{Mat}_{\Delta_k \cup \{O\}}(m, n)$, the indices of the non-O entries of Q must form an $\mathrm{OBS}(\Delta_k)$-rectangle-free subset of $[m] \times [n]$. The question is when this construction can be performed in reverse given an extremal $A \in \mathrm{Mat}_{\{I,O\}}(m, n)$.

Fenner et al. note that if (m, n) is k-colorable then $k\, z(m, n) \geqslant mn$. A similar argument shows that $k\, z(m, n) \geqslant z\big(m, n \,\big|\, \Delta_k \cup \{O\}\big)$, and hence we give an example to show that Question 5.5 is nontrivial.

Example 5.6. $\mathrm{OBS}_2 = \{\, (3, 7),\ (5, 5),\ (7, 3) \,\}$ *as shown in [FGGP10], so neither* $(7, 4)$ *nor* $(7, 5)$ *are* 2*-colorable.*

1. $z(7, 4;\ \mathrm{OBS}_2) = z\big(7, 4 \,\big|\, \Delta_2 \cup \{O\}\big) = 26$*, but*
2. $z(7, 5;\ \mathrm{OBS}_2) = 32$ *and* $z(7, 5) = 15$.

When (m, n) is k-colorable, a related question is whether a coloring is possible for which the entries of some color form an extremal $m \times n$ rectangle-free matrix. In general this cannot be done if the extremal rectangle-free matrix is given, as e.g. $(5, 3)$ is 2-colorable but there is an extremal 5×3 rectangle-free matrix with a non-rectangle-free complement.

Question 5.5 allows us to ask this question about other (m, n).

Question 5.7. *For which m, n, k is there an extremal rectangle-free matrix in $\mathrm{Mat}_{\Delta_k \cup \{O\}}(m, n)$ for which the entries of some color form an extremal rectangle-free matrix in $\mathrm{Mat}_{\{I,O\}}(m, n)$?*

5.2 Symmetric groups

Theorem 5.8. *For $k \geqslant 2$, we have $z(k, k \mid k, k) \leqslant k^3$, with equality if and only if there is a finite projective plane of order k.*

Proof. Trivially, k^3 is an upper bound, e.g. by (2.4), and is achieved only by a matrix Q with every $w(Q_{i,j}) = k$. Such a Q is precisely the quotient matrix for the scaffold with $\mathbf{m} = \mathbf{n} = (k, \ldots, k)$ and $m_O = n_O = 0$ which gives a finite projective plane incidence matrix in Paige–Wexler canonical form. [PW53] $\square$

In general, $z(m, n \mid k, k) = mnk$ if and only if there is a solution in which every block has weight k. Considering the blocks as partial functions, $w(\sigma) = k$ if and only if σ is a total function. In this case we can consider matrices with entries in the symmetric group S_k, and Theorem 5.8 can be restated as: (k, k) is S_k-completable if and only if there is a finite projective plane of order k.

Theorem 5.9. $\mathrm{OBS}(S_k) \supseteq \{\,(2,\,k+1),\,(k+1,\,2)\,\}$, *with equality if and only if there is a finite projective plane of order* k.

Proof. By the pigeonhole principle on the positions of the I entries in the first subrow of each block, $(2,\,k+1)$ and $(k+1,\,2)$ are not S_k-completable for any k. On the other hand, $(2,k)$ and $(k,2)$ are S_k-completable for all k, as the first row can be given by the identity element I_k and the second given by the monochrome sets of a Latin square of order k.

If (k,k) is S_k-completable then there is a finite projective plane of order k. Otherwise there is some $(m,n) \in \mathrm{OBS}(S_k)$ with $m \leqslant k$ and $n \leqslant k$. $\square$

Matrices in Mat_{S_k} in lex-minimal form have only the identity element I_k in the first row and column. By Theorem 3.13 every other block must be a derangement, and for blocks σ, τ in another row or column, $\sigma\tau^{-1}$ must be a derangement.

The $h-1$ non-constant rows of a matrix in $\mathrm{Mat}_{S_k}(h,k)$ in lex-minimal form correspond with Latin squares. Although it is well known that a finite projective plane is equivalent to a set of $k-1$ mutually orthogonal Latin squares (MOLS) of order k, the rows of an incidence matrix in Paige–Wexler canonical form do not themselves yield such a set. However, Paige & Wexler do describe a construction of a complete set of MOLS from the blocks of the incidence matrix.

Example 5.10. $\mathrm{OBS}(S_6) = \{\,(2,7),\,(3,6),\,(6,3),\,(7,2)\,\}$.

Proof. As in Theorem 5.9, $(2,6)$ is S_6-completable. Using the quotient matrix algorithm with $\Sigma = S_6$, we found that $(3,6)$ is not S_6-completable, and that $(5,5)$ is S_6-completable. $\square$

This may be related to the non-existence of a set of 2 MOLS of order 6. For an overview of MOLS, see e.g. [ABCD96]. The next case not covered by Theorem 5.9 is $k = 10$, for which the $\mathcal{O}\big(|\Sigma|^2\big)$ memory required to cache the compatibility relation was too large.

Question 5.11. *Is* (h,k) *is* S_k*-completable if and only if there is a set of* $h-1$ *mutually orthogonal Latin squares of order* k*?*

The following question is a direct analogue of Question 5.5.

Question 5.12. *For which* m,n,k *is* $z\big(m,n \mid S_k \cup \{\,O\}\big) = k\, z\big(m,n;\ \mathrm{OBS}(S_k)\big)$*?*

As before, the equality is trivial when (m,n) is S_k-completable, and given an extremal rectangle-free $Q \in \mathrm{Mat}_{S_k \cup \{O\}}(m,n)$ we can we can take $A \in \mathrm{Mat}_{\{I,O\}}(m,n)$ as the set of indices for which $w(Q_{i,j}) = k$. Again, the question of when the reverse construction is possible is non-trivial.

Example 5.13. *By Theorem 5.9 neither* $(10,5)$ *nor* $(10,6)$ *are* S_2*-completable.*

1. $z(10,5 \mid S_2) = 50$ *and* $z\big(10,5;\ \mathrm{OBS}(S_2)\big) = 25$*, but*
2. $z(10,6 \mid S_2) = 58$ *and* $z\big(10,6;\ \mathrm{OBS}(S_2)\big) = 30$.

Empirical results suggest that the lower bound $z(m,n \mid k,k) \geqslant z\big(m,n \bigm| S_k \cup \{O\}\big)$ is quite sharp and often exact; due to the relative size of the entry sets, the latter is far more efficient to compute.

5.3 Search for finite projective planes

We consider quotient matrices Q such that the expansion $e(\mathcal{S}, Q)$ is the incidence matrix of a finite projective plane of order k, where the scaffold $\mathcal{S}$ has $\mathbf{m} = \mathbf{n} = (k, \ldots, k)$ and $m_O = n_O = 0$.

For $k = 2,3,4,5$, the Q we found have only k distinct entries. This may be because the compatibility relation in Theorem 3.13 is easiest to satisfy over the whole quotient matrix if an entry set is chosen to maximise the proportion of possible $2{\times}2$ minors which are compatible, and this is done by minimising its size. Montaron reported in [Mon85] that $k = 7,8,9,11$ also have this property.

Question 5.14. *If* k *is the order of a finite projective plane, is there a rectangle-free matrix in* $\mathrm{Mat}_{S_k}(k,k)$ *with at most* k *distinct entries?*

Whether or not the property can be assumed to hold, we can look for entry sets $\Sigma \subset S_k$ with a high proportion of compatible $2{\times}2$ minors. Given a good lower bound on this proportion, most such sets should be eliminable without searching for quotient matrices.

We also ask the analogue of Question 5.7.

Question 5.15. *For which* m, n, Σ *is there an extremal rectangle-free matrix in* $\mathrm{Mat}_\Sigma(m,n)$ *for which the entries of some (non-*O*) symbol form an extremal rectangle-free matrix in* $\mathrm{Mat}_{\{I,O\}}(m,n)$*?*

For k, k, S_k this would be a finite projective plane, as above.

Since S_k acts on Q by permuting subrows, without loss of generality it is the I_k entries which form an extremal rectangle-free $k{\times}k$ matrix. By Reiman's theorem (Theorem 1.9, [Rei58]) the expansion $e(\mathcal{S}, Q)$ is itself an extremal rectangle-free $n{\times}n$ matrix, where $n = k^2+k+1$. We call such a k *sub-similar*.

Question 5.16. *Which orders* k *of a finite projective plane are sub-similar?*

If it can be shown that k is sub-similar, we can search for finite projective planes of order k from an initial configuration of the quotient matrix with $z(k,k)$ confirmed I_k entries, rather than $2k-1$ in the first row and column as in Paige–Wexler canonical form. We verified that

$k = 2, 3, 4, 5$ are sub-similar; $k = 6$ is not, as there is no finite projective plane of order 6. We were able to verify that $k = 7$ is not sub-similar; see Appendix B for details. It was infeasible to check $k \geqslant 8$ using the techniques described in this paper.

Due to the small number of non-isomorphic extremal rectangle-free matrices under row, column and transpose symmetries — e.g. the algorithm in Section 2.4, modified to search for all solutions, found 8 distinct extremal 12×12 rectangle-free matrices, all isomorphic — the cost of repeating the search for each initial configuration, if necessary, should be outweighed by the benefit of a shallower search tree. The entries in such a solution must still be I_k or derangements,[17] and the search space can still be reduced by symmetry, as the action of S_k by conjugation of entries preserves the initial configuration and the rectangle-free property.

We do not propose that algorithms similar to those presented in this paper would be feasible for $k = 12$, but the idea may be applicable in other approaches.

References

[ABCD96] R. Julian R. Abel, Andres E. Brouwer, Charles J. Colbourn, and Jeffrey H. Dinitz. Mutually orthogonal Latin squares (MOLS). In *The CRC handbook of combinatorial designs*, pages 160–193. CRC Press, Boca Raton, 1996.

[Cam95] Peter J. Cameron. Finite geometries. In *Handbook of Combinatorics*, pages 647–691. Elsevier, 1995.

[Čul56] K. Čulik. Teilweise Lösung eines verallgemeinerten problems von K. Zarankiewicz. In *Annales Polonici Mathematici*, volume 1, pages 165–168, 1956.

[DDR13] Janusz Dybizbański, Tomasz Dzido, and Stanisław Radziszowski. On some Zarankiewicz numbers and bipartite Ramsey numbers for quadrilateral. *arXiv preprint arXiv:1303.5475*, 2013.

[DHS13] Gábor Damásdi, Tamás Héger, and Tamás Szőnyi. The Zarankiewicz problem, cages, and geometries. *Annales Universitatis Scientiarum Budapestinensis de Rolando Eötvös Nominatae Sectio Mathematica*, 56(1):3–37, 2013.

[DM96] John D. Dixon and Brian Mortimer. *Permutation groups*, volume 163. Springer Science & Business Media, 1996.

[FFH+02] Pierre Flener, Alan M. Frisch, Brahim Hnich, Zeynep Kiziltan, Ian Miguel, Justin Pearson, and Toby Walsh. Breaking row and column symmetries in matrix models. In *Principles and Practice of Constraint Programming-CP 2002*, pages 462–477. Springer, 2002.

[FGGP10] Stephen Fenner, William Gasarch, Charles Glover, and Semmy Purewal. Rectangle free coloring of grids. *arXiv preprint arXiv:1005.3750*, 2010.

[17] Every entry $\sigma \neq I_k$ must be a corner of some $\left(\begin{smallmatrix} I_k & I_k \\ I_k & \sigma \end{smallmatrix}\right)$ minor, otherwise I_k could be inserted there in the initial configuration without creating a rectangle.

[FS13] Zoltán Füredi and Miklós Simonovits. The history of degenerate (bipartite) extremal graph problems. In *Erdős Centennial*, pages 169–264. Springer, 2013.

[GHO00] Wayne Goddard, Michael A. Henning, and Ortrud R. Oellermann. Bipartite Ramsey numbers and Zarankiewicz numbers. *Discrete Mathematics*, 219(1):85–95, 2000.

[Guy68] Richard K. Guy. A problem of Zarankiewicz. In *Theory of Graphs*, pages 119–150. Akadémai Kiadó, Budapest, 1968.

[Guy69] Richard K. Guy. A many-facetted problem of Zarankiewicz. In *The Many Facets of Graph Theory*, pages 129–148. Springer, 1969.

[KST54] T. Kóvari, V. Sós, and P. Turán. On a problem of K. Zarankiewicz. *Colloquium Mathematicae*, 3(1):50–57, 1954.

[Lip96] Stephen Lipscomb. *Symmetric inverse semigroups*, volume 46 of *Mathematical Surveys and Monographs*. American Mathematical Society, 1996.

[Mon85] Bernard Montaron. On incidence matrices of finite projective planes. *Discrete Mathematics*, 56(2):227–237, 1985.

[MP14] Brendan D. McKay and Adolfo Piperno. Practical graph isomorphism, II. *Journal of Symbolic Computation*, 60:94–112, 2014.

[PW53] Lowell J. Paige and Charles Wexler. A canonical form for incidence matrices of finite projective planes and their associated Latin squares. *Portugaliae mathematica*, 12(3):105–112, 1953.

[Rei58] Istvan Reiman. Über ein problem von K. Zarankiewicz. *Acta Mathematica Hungarica*, 9(3-4):269–273, 1958.

[Rom75] Steven Roman. A problem of Zarankiewicz. *Journal of Combinatorial Theory, Series A*, 18(2):187–198, 1975.

[SP12] Bernd Steinbach and Christian Posthoff. Search space restriction for maximal rectangle-free grids. In *10th International Workshop on Boolean Problems*, 2012.

[SP14] Bernd Steinbach and Christian Posthoff. The slot principle. In *Recent Progress in the Boolean Domain*. Cambridge Scholars Publisher, 2014.

[Spi92] J. Michael Spivey. *The Z notation: a reference manual*. International Series in Computer Science. Prentice-Hall, New York, NY, 1992.

[Wer12] Matthias Werner. An algorithmic approach for the Zarankiewicz problem. In *10th International Workshop on Boolean Problems*, 2012.

[Zar51] Kazimierz Zarankiewicz. Problem p 101. In *Colloq. Math*, volume 2, page 5, 1951.

A Appendix: Tables

Tables for $z(m,n;\ s,t)$ have been previously published by Guy in [Guy68] and [Guy69], and by Damásdi et al. in [DHS13] for the case $s=t=2$. Due to the number of new exact values found by computer search (in **bold**), and errors in Guy's tables (marked $*$), new tables may be of benefit. Some values reported by Damásdi et al. (in *italics*) were not feasible to confirm by computer search. We were able to check all of Guy's values. Some values are previously unreported, but likely only because they lie outside the range of earlier tables; these are not labelled as new.

A.1 Exact values of $z(m,n;\ 2,2)$

n \ m	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
4	9																			
5	10	12																		
6	12	14	16																	
7	13	15	18	21																
8	14	17	19	22	24															
9	15	18	21	24	26	29														
10	16	20	22	25	28	31	34													
11	17	21	24	27	30	33	36	39												
12	18	22	25	28	32	36	39	42	45											
13	19	23	27	30	33	37	40	44	48	52										
14	20	24	28	31	35	39	42	45	49	53	56									
15	21	25	30	33	36	40	44	47	51	55	58	61*								
16	22	26	31	34	38	42	46	50	53	57	60	64*	67							
17	23	27	32	36	39	43	47	51	55	59	63	67*	70	74						
18	24	28	33	37	41	45	49	53	57	61	65	69	73	77	81					
19	25	29	34	39	42	46	51	55	60	64	68	72	76	80	84	88				
20	26	30	35	40	44	48	52	57	61	66	70	75	80	84	88	92	96			
21	27	31	36	42	45	49	54	59	63	67	72	77	81	**85**	90	95	100	105		
22	28	32	37	43	47	51	55	60	65	69	73	78	**83**	**87**	**91**	**96**	101	106	**108**	
23	29	33	38	44	48	52	57	62	66	71	75	80	**85**	**89**	**93**	**98**	**103**	**108**	**110**	**115**
24	30	34	39	45	50	54	58	63	68	73	78*	**82**	**87**		**96**	**100**	**105**			**118**
25	31	35	40	46	51	55	60	65	70	75	80*	*85*	*90*							
26	32	36	41	47	53	57	61	66	72	78	**82***	*86*	*91*	*96*	*101*	*106*	*111*	*116*		
27	33	37	42	48	54	58	63	68	73	79	**84***	*88*	*93*	*98*	*103*	*108*	*113*	*118*	*123*	*128*
28	34	38	43	49	56	60	64	69	75	81	**86***		*96*	*101*	*106*	*111*	*116*	*121*	*126*	*131*
29	35	39	44	50	57	61	66	71	76	82						*114*	*120*	*125*	*130*	*135*
30	36	40	45	51	58	63	67	72	78	84									*132*	*138*
31	37	41	46	52	59	64	69	74	79	85										
32	38	42	47	53	60	66	70	75	81	87										
33	39	43	48	54	61	67	72	77	82	88										
34	40	44	49	55	62	69	73	78	84											
35	41	45	50	56	63	70	75	80	85											
36	42	46	51	57	64	72	76	81	87											
37	43	47	52	58	65	73	78	83	88											
38	44	48	53	59	66	74	79	84	90											

N.b. the unbolded values marked $*$ were corrected in [DHS13], except for $z(15,15)=61$ as noted by Héger after publication, and reported correctly by Dybizbański et al. in [DDR13].

A.2 Exact values of $z(m, n;\ 2, 3)$

n \ m	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
4	9	12	14	16	18	20	22	24	26	28	30	32	34	36	38	40	42	44	46	48	50
5	10	13	16	19	22	24	27	30	32	34	36	38	40	42	44	46	48	50	52	54	56
6	12	15	18	21	24	27	30	33	36	38	41	44	46	49	52	54	57	60	62	64	66
7	13	16	20	24	28	31	34	37	40	43	46	49	51	54	57	59	62	65	67	70	73
8	14	18	22	25	29	33	36	40	44	48	52	56	58	61	64	66	69	72	74	77	
9	15	19	23	27	31	35	39	42	46	50	54	58	61	64	68	72	75	78	81	84	
10	16	21	25	30	34	38	42	46	50	53	56	**60**	**64**	**68**	**72**	**76**	**79**	**83**	**86**	**90**	
11	17	22	26	31	36	40	45	50	55	**58**	**61**	**65**	**68**	**72**	**76**	**79**	**83**	**87**	**91**	**95**	
12	18	24	28	33	37*	42	46	51	56	**60**	**64**	**68**	**72**	**76**	**80**	**84**	**88**	**92**	**96**		
13	19	25	29	34	39	44	49	54	**58**	**63**	**67**										
14	20	26	31	36	42	46	51	56	**61**	**66**	**71**										
15	21	27	32	37	43	48	54	60	**65**	**70**	**75**										
16	22	28	34	39	45	50	56	61	**66**	**72**											
17	23	29	35	40	46	52	57	63	**69**	**74**											
18	24	30	37	42	48	54	**60***	66	**71**	**77**											
19	25	31	38	43	49	56	**62***	**68***	**74**	**79**											
20	26	32	40	45	51	58	64	70	**76**	**82**											
21	27	33	41	46	52	59	66	72													
22	28	34	42	48	54	61	68	**74**													
23	29	35	43	49	55	62	69	**76**													
24	30	36	44	51	57	64	72	78													
25	31	37	45	52	58	65	73	**80**													
26	32	38	46	54	60	67	75	**82**													
27	33	39	47	55	61	68	76														
28	34	40	48	57	63	70	78														

N.b. $z(7, 12;\ 2, 3) = 37$ was reported correctly in [Guy68] but incorrectly in [Guy69].

A.3 Exact values of $z(m, n;\ 2, 4)$

n \ m	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
5	12	16	20	23	26	29	32	35	38	41	44	47	50	53	56	59	62	65	68	71	74
6	13	17	21	25	29	33	37	41	45	48	52	56	60	63	66	69	72	75	78	81	84
7	15	19	23	27	31	35	39	43	47	51	55	59	63	67	71	75	79	83	87	91	95
8	16	21	26	30	35	40	44	48	52	56	60	64	68	72	76	80	84	88	92	96	100
9	18	22	27	32	37	42	47	52	57	**61**	**66**	**71**	**75**	**80**							
10	19	24	30	35	40	45	50	55	60	**65**	**70**	**75**	**80**	**85**							
11	20	25	31	37	42	48	54	60	66	**71**	**76**	**81**	**86**	**91**							
12	21	27	33	39	44	50	56	**61**	**67**	**73**	**78**	**84**	**90**								
13	22	28	34	40	46	52	58	**64**	**70**	**76**	**81**	**87**	**93**								
14	23	30	36	42	49	56	**62**	**68**	**74**	**80**	**86**	**92**	**98**								
15	24	31	37	44	51	57	**64**	**71**	**77**	**84**	**91**	**98**	105								
16	25	33	39	46	53	59	**66**	**73**	**79**	**86**	**93**										
17	26	34	40	48	55	62	**69**	**75**	**82**	**89**	**96**										
18	27	36	42	49	57	64	**72**	**78**	**85**	**92**	**99**										
19	28	37	43	51	58	66	**73**	**81**	**88**												
20	29	38	45	52	60	68	**75**	**83**													
21	30	39	46	54	63	70															
22	31	40	48	55	64	72															
23	32	41	49	57	66	74															
24	33	42	51	58	67	76															
25	34	43	52	60	69																

A.4 Exact values of $z(m, n;\ 3, 3)$

$_{n}\,^{m}$	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
4	13	16	18	21	24	26	28	30	32	34	36	38	40	42	44	46	48	50	52	54	56
5		20	22	25	28	30	33	36	38	41	44	46	49	52	54	57	60	62	64	66	68
6			26	29	32	36	39	42	45	48	50	53	56	58	61	64	66	69	72	74	77
7				33	37	40	44	47	50	53	56	60	63	66	69	72	75	78	81	84	87
8					42	45	50	53	57	60	64	**67**	**70**	**74**							
9						49	54	59	64	**67**											
10							60	**64**	**68**												
11								**69**	**74**												
12									**80**												

A.5 Exact values of $z(m, n;\ 3, 4)$

$_{n}\,^{m}$	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**
5	17	21	25	28	32	36	40	43	46	49	52	55	58	61	64	67	70	73	76	79
6	20	25	30	33	37	41	45	48	52	56	60	63	67	71	75	78	82	86	90	93
7	22	27	32	37	42	46	51	55	60	64	68	72	76	80	84	88	92	96	100	103
8	25	30	35	40	45	50	55	60	65	70	**75***	79	**83***	**88***	**92***	**97***				
9	28	33	39	44	**50**	**56**	**61**	**66**	**72**	**77**	**82**	**86**	**91**	**96**	**101**	**106**				
10	30	36	42	48	**54**	**60**	**66**	**72**	**78**	**83**	**89**	**94**	**100**	**105**						
11	33	39	46	52	**59**	**66**														
12	36	42	50	56	**64**	**72**														
13	38	44	52	**59**	**67**	**75**														
14	40	47	56	**63**	**70**	**78**														
15	42	50	60	**66**	**74**															
16	44	52	62	**70**																
17	46	55	65	**73**																
18	48	58	68	**77**																
19	50	60	70																	
20	52	63	73																	
21	54	66	76																	
22	56	68	78																	

A.6 Exact values of $z(m, n;\ 4, 4)$

$_{n}\,^{m}$	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**	**14**	**15**	**16**	**17**	**18**	**19**	**20**	**21**	**22**	**23**	**24**
5	22	26	30	33	37	41	45	48	52	56	60	63	66	69	72	75	78	81	84	87
6		31	36	39	43	47	51	55	59	63	67	71	75	78	82	86	90	93	97	101
7			42	45	49	54	58	63	68	72	77	82	87	90	95	100	105	108	112	116
8				51	55	60	**65**	**70**	**75**	**80**	**85**	**90**	**95**	**99**	**104**					
9					**61**	**67**	**72**	**78**	**84**	**88**	**94**									
10						**74**	**79**	**86**	**93**	**97**										
11							**86**	**93**												

B Appendix: $k = 7$ is not sub-similar

We searched for quotient matrices for a finite projective plane of order 7 from an initial configuration containing $z(7, 7) = 21$ blocks I_7. By Reiman's theorem (Theorem 1.9, [Rei58]) the I_7 blocks in this initial configuration must themselves form the incidence matrix of the unique finite projective plane of order 2, the Fano plane.

This would have been infeasible without the following observation specific to the case $k = 7$.

Lemma B.1. *By row, column and transpose symmetries, the indices of any unconfirmed entry in the initial configuration can be mapped to any other, while preserving the initial configuration.*

Proof. The automorphism group of the Fano plane acts transitively on antiflags. [DM96, p. 305] □

Therefore if the initial configuration plus one entry $\sigma \in S_7$ leads to no solutions, σ can be eliminated as a possibility in *all* entries. As the action of S_7 on the quotient matrix by conjugation of entries preserves the initial configuration and the rectangle-free property, if the initial configuration plus one entry σ leads to no solutions, conjugates of σ can similarly be eliminated everywhere.

As σ must be a derangement, it was therefore only necessary to perform four searches for an initial σ with cycle types $(2, 2, 3)$, $(2, 5)$, $(3, 4)$ and (7) respectively. If no solution is found, entries of this cycle type may be deleted from Σ in later searches.

These reductions in the search space made the search feasible. We used an instance of the abstract algorithm model in Section 2.1 with Constraint 3.24, and a guess function which chooses an unconfirmed entry with the fewest possible symbols.

We found no solution in any of the four cases. Therefore, there is no quotient matrix for a finite projective plane of order 7 with I_7 entries forming an extremal 7×7 rectangle-free matrix, and so $k = 7$ is not sub-similar. □