Pseudoku: A Sudoku Adjacency Algebra and Fractional Completion Threshold

by

Kate Nimegeers

A Thesis Submitted in Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

in the Department of Mathematics and Statistics

We acknowledge with respect the Lekwungen peoples on whose traditional territory the university stands, and the Songhees, Esquimalt, and W̱SÁNEĆ peoples whose historical relationships with the land continue to this day.

Pseudoku: A Sudoku Adjacency Algebra and Fractional Completion Threshold

by

Kate Nimegeers

Supervisory Committee

Dr. Peter J. Dukes, Supervisor
(Department of Mathematics and Statistics)

Dr. Natasha Morrison, Committee Member
(Department of Mathematics and Statistics)

ABSTRACT

The standard Sudoku puzzle is a 9×9 grid partitioned into 3×3 square boxes and partially filled with symbols from the set $\{1, 2, ..., 9\}$, with the goal of the puzzle being to complete the grid so that each symbol appears once and only once in each row, column, and box. We study generalized Sudoku puzzles, set on an $n \times n$ grid with cells partitioned into n boxes (sometimes called cages) of height h and width w such that $hw = n$. Throughout this work, these generalized Sudoku are referred to as (h, w)-Sudoku when h and w are significant, but as simply Sudoku otherwise. The goal of solving a partially filled (h, w)-Sudoku puzzle remains the same; complete the Sudoku by assigning placements in the grid to each symbol from $\{1, 2, ..., hw\}$ so that each symbol appears once and only once in each row, column, and box.

This thesis is specifically concerned with establishing conditions which guarantee a *fractional* Sudoku completion. A fractional Sudoku completion is an assignment of a set of weights to each symbol-cell incidence, representing the proportion of the symbol for that specific cell. The total weight of symbols for each cell must sum to one, and the sum of the weights for each symbol must be exactly one across the cells from each row, column, and box. These conditions still require a balanced distribution of symbols throughout the grid, but with considerably more flexibility than the typical Sudoku conditions.

In order to apply graph theoretic techniques to the problem, we develop a 4-partite graph representation, G_P, for a partial Sudoku, P. The 4 parts correspond to the rows, columns, symbols, and boxes of P, and the edges of G_P indicate the conditions for a completed Sudoku that remain unsatisfied in P. We then introduce the concept of a *tile*: a 4-vertex subgraph of G_P, which represents a valid symbol placement in P. Completing P is equivalent to decomposing the edges of G_P into these tiles. We then use an edge-tile inclusion matrix to relate the existence of such a decomposition to the existence of an solution vector with $\{0, 1\}$ entries for a specific linear system. It is here that we move to the fractional setting through a relaxation of what constitutes an acceptable solution to the linear system - specifically, we are satisfied with solution vectors for which all entries are non-negative.

To find conditions that guarantee such a solution exists we study the Gram matrix of the edge-tile inclusion matrix for the *empty* (h, w)-Sudoku, denoted M. We show that M is symmetric and that each element of M corresponds to a pair of edges in the graph representation G_{hw} of the empty (h, w)-Sudoku grid. We then leverage the inherent symmetry of equivalence relations between these edges to establish a Sudoku adjacency algebra which contains M. This allows us to explicitly construct a generalized inverse

for M. This generalized inverse, along with some applied perturbation theory, is used to show that given large enough h and w, the linear system for any sufficiently sparse partial (h, w)-Sudoku is a minor perturbation of the linear system for the empty (h, w)-Sudoku, and therefore allows a fractional completion.

After presenting this main result, we take a brief detour to consider the unique case of Sudoku puzzles with *thin boxes*, examining how fixing the box width variable w while allowing height h to grow asymptotically influences the density conditions necessary for fractional completion. We also give an overview of our exploratory use of the Schur complement for matrix decomposition. Although this method didn't directly feed into our primary results, it was instrumental in the discovery of the equivalence relations we used to construct our Sudoku adjacency algebra. Finally, we explore the potential applicability of our methodologies to certain Sudoku variants and acknowledge the limitations inherent in our approach.

In the appendices, we provide additional resources that complement the main body of our work. In Appendix A, we give a factorization of the Sudoku matrix M and its eigenvectors as Kronecker products for readers who wish to more directly compare our methodology to algebraic graph theory work done on Sudoku by other researchers. Appendix B presents a series of interactive and educational activities designed to introduce students to the basic principles of Latin squares in a fun spy-themed setting.

Table of Contents

List of Tables

List of Figures

ACKNOWLEDGEMENTS

I would like to take a moment to thank:

Barron & Pam, for always believing I am capable of amazing things.

Brandon, for making my good days great and my bad days better.

Peter, for all your guidance and for countless hours talking about my two favourite hyper-fixations: math and Stardew Valley.

Breanna, for always encouraging me to chase after the best version of myself.

Akina, for always laughing the hardest when I make math puns.

The Combi-Crew, for being a proper subset of my greatest friends.

The AWM and Sigmas at UVic, for being pivotal in connecting me with fellow mathematics enthusiasts. Special thanks to Elena, Ashna, Aaron, and Ravessa for making me feel at home within these clubs.

URImprov, for being the highlight of my undergraduate experience. A special mention to Nando, who inspired me to get involved.

John & Myrtle Tilley, Ian Evans, the McCall MacBain Foundation, & FGS, for making sure I could afford enough coffee to power a math thesis.

"They're cheering for you," she said with a smile.
"But I could never have done it," he objected,
"without everyone else's help."
"That may be true," said Reason gravely,
"but you had the courage to try;
and what you can do is often simply
*a matter of what you *will* do."*
- Norton Juster, The Phantom Tollbooth

DEDICATION

To Fernando,

With profound appreciation, I dedicate this Masters thesis to you. As you know, when we met I had no ambitions of becoming a mathematician - I just wanted to pass calculus. Your support and unwavering belief in my potential, at a time when understanding mathematics felt utterly beyond my capabilities, made all of this possible. I am forever grateful for both your mentorship and friendship.

With gratitude,
Kate Nimegeers

Chapter 1
Introduction

1.1 The Latin square completion problem

A *Latin square* of order n is an $n \times n$ array with each cell containing an entry from a set of n symbols, with the condition that each symbol must appear exactly once in every row and every column. Latin squares and their variations, such as Sudoku, are regularly encountered as puzzles, where the entries are only partially filled, challenging a player to complete the puzzle while maintaining these key conditions. Presenting these structures as puzzle games necessitates the concept of a partially filled Latin square. A *partial Latin square* of order n is an $n \times n$ array with cells that may either remain empty or be filled in such a way that each symbol appears at most once in every row and column.

One can construct a set of ordered triples that exactly represents a partial Latin square L by representing the presence of symbol s_k in row r_i and column c_j as the ordered triple (i, j, k). The conditions placed on a Latin square mean that each pair of distinct triples in L agree in at most one coordinate. Using this notation, we can say that a partial Latin square L is *completable* if there exists a Latin square L' for which $L \subseteq L'$. If this is the case, we say that L' is a *completion* of L. Conversely, we say that L is *incompletable* if there does not exist a Latin square L' such that $L \subseteq L'$. A simple example of a partial Latin square L and a completion of L can be seen in Figure 1.1.

	2		
2	3	4	
3	4	1	
	1		3

1	2	3	4
2	3	4	1
3	4	1	2
4	1	2	3

Figure 1.1: A partial Latin square L, and a completion of L.

One intriguing challenge in the study of Latin squares is the creation of incompletable partial Latin squares which have very few filled entries. It was conjectured in 1960 by Evans [10] and later proven in 1981 by Smetaniuk [17] that any partial

Latin square of order n with at most $(n-1)$ filled entries is completable. This bound is indeed the best possible, as there exist a variety of constructions of partial Latin squares of order n that do not allow a completion but which have only n filled entries.

For instance, one can place a specific symbol k along the diagonal in every entry except one. By inserting any symbol $k' \in [n]$ with $k' \neq k$ in that final entry, the partial Latin square becomes incompletable. Another method is to start with an empty grid, populate a specific row with all symbols $\{1, 2, ..., n-1\}$, and then insert the symbol n in the last column but in a different row. This approach also results in an incompletable partial Latin square. A similar strategy can be applied by almost completely filling a specific column and introducing a similar 'blockage'. Small examples of these constructions which 'block' completion are depicted in Figure 1.2.

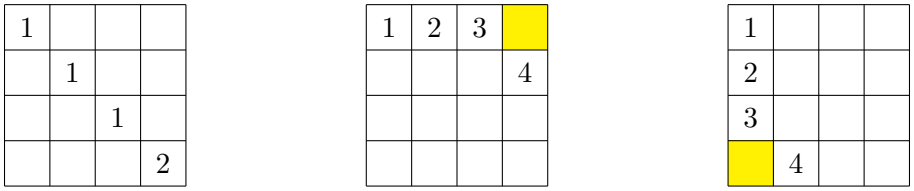


Figure 1.2: Incompletable partial Latin squares of order 4 with 4 filled entries.

In all of these incompletable partial Latin squares, only n entries are filled. However, it is readily apparent that the feature being leveraged in their construction is that some row, column, or symbol is employed exactly $n-1$ times and then a blockage is formed. This observation naturally leads to a more robust challenge by adding restrictions on the frequency of use for each row, column, or symbol.

Definition 1.1.1. A partial Latin square of order n is *ϵ-dense* if no row, column, or symbol is used more than ϵn times.

For example, to construct an incompletable partial Latin square of order n with the property that no row, column, or symbol is used more than $\lceil \frac{n}{2} \rceil$ times, we populate a specific row i with all symbols $\{1, 2, ..., \lceil \frac{n}{2} \rceil\}$, and then insert symbols $\{\lceil \frac{n}{2} \rceil + 1, \lceil \frac{n}{2} \rceil + 2, ..., n\}$ anywhere in column j, avoiding the entry (i, j). It is clear that this partial Latin square is incompletable because the entry (i, j) cannot be filled. If n is even, then L is $\frac{1}{2}$-dense. A example of this construction for a partial Latin square of order 4 is depicted in Figure 1.3 with the 'blocked' entry highlighted yellow.

Inspired by a famous unproven conjecture by Nash and Williams [14] on graph decompositions, Daykin and Häggkvist conjectured [8] that a partial Latin square which is $\frac{1}{4}$-dense should allow a completion. They proved a weaker version of this conjecture, replacing $\epsilon \leq 1/4$ with $\epsilon \leq 2^{-9} n^{-1/2}$ and requiring that n be divisible by 16. Subsequent work by Chetwynd and Häggkvist [6], followed by Gustavsson [11], further advanced this field by proving that for *any* $n \in \mathbb{N}$, there exists an $\epsilon > 0$ such that it

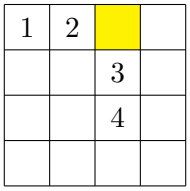


Figure 1.3: A $\frac{1}{2}$-dense partial Latin square of order 4.

is always possible to complete an ϵ-dense partial Latin square of order n. Bartlett [4] then demonstrated that all ϵ-dense partial Latin squares have a completion, with $\epsilon = 9.8(10^{-5})$. Subsequently, over two papers, this threshold was improved to *approximately* $\epsilon = \frac{1}{25}$, given that n is sufficiently large. One of the papers [5], by Bowditch and Dukes, established this threshold for a fractional relaxation of the problems, while the other paper [3] by Barber, Kühn, Lo, Osthus, and Taylor showed that the fractional threshold suffices for sufficiently large instances of the exact completion problem.

We now provide a more detailed explanation of fractional completion in Latin squares. Let L be a partial Latin square which we represent as a function $f_L : [n]^3 \to 0, 1$, where $f_L(i, j, k)$ denotes the presence (1) or absence (0) of symbol k in cell (i, j) of L. We define $L(i, j)$ as a mapping from the cell (i, j) to a set that is either empty, indicating no symbol has been assigned, or which contains exactly one symbol, k, reflecting the specific symbol assigned to cell (i, j) in L.

Definition 1.1.2. Let L be a partial Latin square of size $n \times n$. A *fractional completion* of L is a function, $f : [n]^3 \to [0, 1]$, such that for all $i, j, k \in [n]$, the following conditions hold:

1. *Consistency with Partial Latin Square L:* If $L(i, j) = \{k\}$ then $f(i, j, k) = 1$.
2. *Row, Column, and Symbol Constraints:* The following sums hold:
   - For every $j, k \in [n]$, $\sum_{i=1}^{n} f(i, j, k) = 1$.
   - For every $i, k \in [n]$, $\sum_{j=1}^{n} f(i, j, k) = 1$.
   - For every $i, j \in [n]$, $\sum_{k=1}^{n} f(i, j, k) = 1$.

Figure 1.4 provides a visual representation of a fractional completion of a Latin square of order 4 where the four colors represent the symbols, the solid disks represent pre-filled cells of a partial Latin square, and the multi-colored disks represent a fractional completion.

1.2 Extending the completion problem to Sudoku

Let h and w be integers with $h, w \geq 2$ and $hw = n$. A *Sudoku Latin square* of type (h, w) is an $n \times n$ Latin square partitioned into $h \times w$ sub-grids, called *boxes*, for which

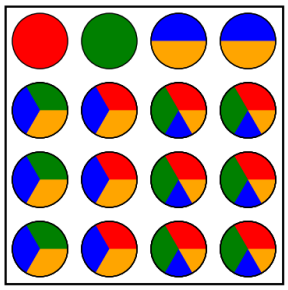


Figure 1.4: Illustration of a fractional completion of a Latin square of order 4.

the regular conditions of a Latin square must hold and every symbol must appear precisely once in each box. For brevity, we will henceforth refer to a Sudoku Latin square of type (h, w) as an (h, w)-*Sudoku* or, when h and w are not relevant, a *Sudoku*. We define the concepts of a partial Sudoku and its completion analogously to those definitions provided for Latin squares, with examples shown in Figure 1.5.

1			
	4		
			3
		2	1

1	2	3	4
3	4	1	2
2	1	4	3
4	3	2	1

Figure 1.5: A partial $(2, 2)$-Sudoku P and its completion.

In an (h, w)-Sudoku, we let $\text{box}(i, j)$ denote the box containing cell (i, j). If boxes are numbered left to right then top to bottom, then we have the convenient function which takes the cell position (i, j) as input and returns the box number that the cell belongs to:

$$\text{box}(i, j) = h \left\lfloor \frac{i-1}{h} \right\rfloor + \left\lfloor \frac{j-1}{w} \right\rfloor + 1. \tag{1.1}$$

For example, if boxes are numbered left to right then top to bottom in the $(2, 3)$-Sudoku, then cell $(5, 3)$ is in the fifth box, denoted b_5, which we can find without examining the $(2, 3)$-Sudoku because

$$5 = 2 \left\lfloor \frac{5-1}{2} \right\rfloor + \left\lfloor \frac{3-1}{3} \right\rfloor + 1.$$

Definition 1.2.1. Given an (h, w)-Sudoku, the term *row-bundle* is used to describe any set of h rows in that (h, w)-Sudoku that all intersect the same h boxes, and the term *column-bundle* to describe any set of w columns in that (h, w)-Sudoku that all intersect the same w boxes.

Note that by this definition, an (h, w)-Sudoku contains exactly w row-bundles and

h column-bundles. Figure 1.6 shows the empty $(2,3)$-Sudoku grid with the first row-bundle highlighted in yellow and, beside it, the same grid with the first column-bundle highlighted in green.

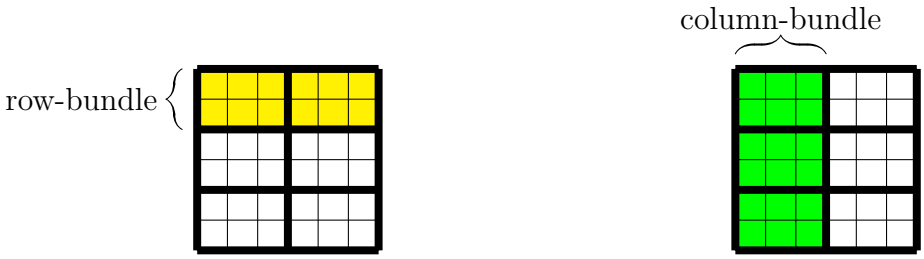


Figure 1.6: An illustration of a row-bundle beside an illustration of a column-bundle, each shown in a partial $(2,3)$-Sudoku.

We now define a fractional completion of a partial Sudoku. We use the function $P(i,j)$ as a mapping from the cell (i,j) to a set that is either empty, indicating no symbol has been assigned, or which contains exactly one symbol, k, reflecting the specific symbol assigned to cell (i,j) in P.

Definition 1.2.2. Let P be a partial Sudoku of size $n \times n$. A *fractional completion* of P is a function, $f : [n]^3 \to [0,1]$, such that for all $i,j,k,\ell \in [n]$, the following hold:

1. *Consistency with P:* If $P(i,j) = k$ then $f(i,j,k) = 1$.
2. *Row, Column, Symbol, and Box Constraints:* The following sums hold:
   - For every $j,k \in [n]$, $\sum_{i=1}^{n} f(i,j,k) = 1$.
   - For every $i,k \in [n]$, $\sum_{j=1}^{n} f(i,j,k) = 1$.
   - For every $i,j \in [n]$, $\sum_{k=1}^{n} f(i,j,k) = 1$.
   - For every $k,\ell \in [n]$, $\sum_{(i,j):\mathrm{box}(i,j)=\ell} f(i,j,k) = 1$.

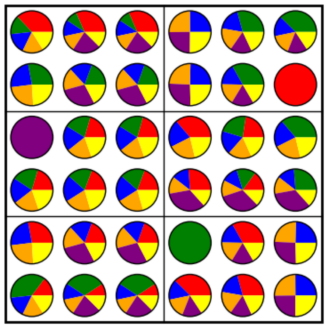


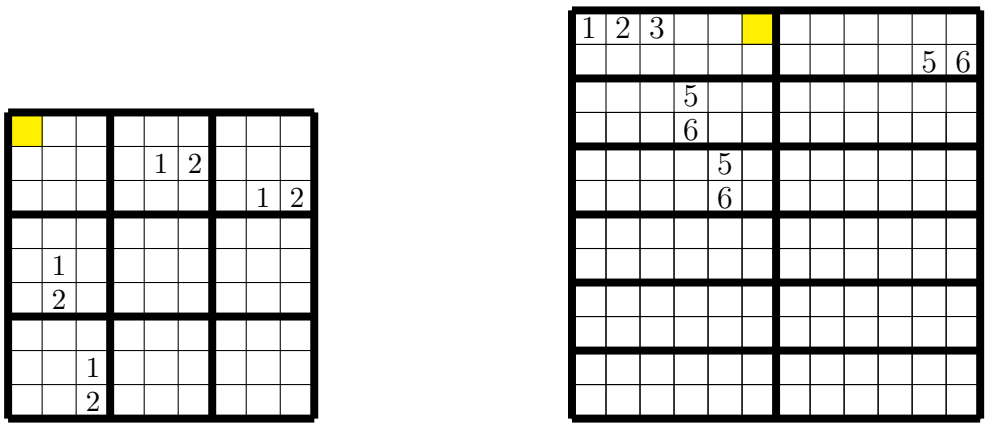
Figure 1.7: Illustration of a fractional completion of a $(2,3)$-Sudoku.

Figure 1.7 provides a visual representation of a fractional $(2, 3)$-Sudoku, where solid disks represent pre-filled cells of a partial Sudoku, and multi-colored disks represent a fractional completion.

Just as any Sudoku is necessarily a Latin square, any fractional Sudoku must also be a fractional Latin square. Therefore, most of the conditions are inherited directly from the Latin square. The final condition is unique to Sudoku because it describes the necessary constraints for boxes. Also, notice that if P is a partial Sudoku with a fractional completion f such that f only takes values in $\{0, 1\}$ then f fits the original requirements of a completion of P. In this case, we refer to f as an *exact* completion.

In the Sudoku context, the definition of ϵ-density requires additional strengthening. First of all, it is natural to impose constraints on the number of filled cells within each box. If we do not impose such constraints, completion can be easily obstructed by placing symbols $\{1, 2, ..., n-1\}$ in any single box and symbol n in alignment with the empty cell of that box. This effectively utilizes each symbol exactly once while limiting each row and column to at most $\max\{h, w\}$ occurrences. Therefore, in cases where $h \approx w$, the number of filled entries in each row or column is sub-linear in n.

We must also prevent symbol occurrences from becoming too imbalanced relative to the box partition. In an (h, w)-Sudoku, it is possible to force a particular symbol into the $(1, 1)$-entry, for example, by placing it outside the top-left box in rows $2, \ldots, h$ and in columns $2, \ldots, w$. By arranging this scenario for different symbols, such as 1 and 2, we can block completion. An illustration of this method is given in Figure 1.8a with a simple variation for Sudoku with $h \neq w$ shown in Figure 1.8b. In both figures, the problematic cell that requires multiple entries is coloured yellow.



(a) (b)

Figure 1.8: Two sparse partial Sudoku allowing no fractional completion.

Notice that this construction only uses each row or column at most twice and employs each symbol merely $h + w - 2$ times, which is sub-linear in cases where $h \approx w$.

Definition 1.2.3. A partial (h, w)-Sudoku is *ϵ-dense* if no row, column, or box is utilized more than ϵn times *and* each symbol occurs at most ϵh times in any row-bundle, likewise, at most ϵw times in any column-bundle.

Our main result gives a guarantee on fractional completion of ϵ-dense Sudoku.

Theorem 1.2.4. *For $\epsilon < \frac{1}{101}$ and sufficiently large h, w, every ϵ-dense partial (h, w)-Sudoku admits a fractional completion.*

1.3 Outline of thesis

This thesis is organized into six main chapters, each delving into specific aspects of the Sudoku completion problem and its fractional relaxation. An outline of each chapter is given below.

Serving as the foundation for the thesis, we begin Chapter 2 by introducing essential terminology and context. We then present a detailed exploration of Latin squares and their representation as balanced tripartite graphs, as seen in [4, 5]. The chapter then gives a natural extension into the representation of a partial Sudoku, P, as a balanced 4-partite graph with edges representing unsatisfied conditions in P that would need to be satisfied in order for P to be completed. We then introduce the concept of a *tile*; a 4-vertex subgraph representing a valid symbol placement in P. We show that completing P by fulfilling currently unmet conditions through valid symbol placements is equivalent to decomposing the edges of G_P into these tiles. Using a $\{0, 1\}$ edge-tile inclusion matrix, we relate the existence of such a decomposition to the existence of an entrywise $\{0, 1\}$ solution vector for a specific linear system built from the edge-tile inclusion matrix. The rank of this matrix is discussed in detail and an explicit construction of a set of basis vectors in the kernel is provided.

Chapter 3 explores the symmetries present in the Sudoku conditions via a thorough examination of the set of equivalence relations that exist between pairs of edges in the graph representation. We show that these equivalence relations, when written as $\{0, 1\}$ adjacency matrices, form a basis of a Sudoku adjacency algebra, $\mathfrak{A}$. The Gram matrix of the edge-tile inclusion matrix for the empty Sudoku, denoted M and described in Chapter 2, resides within $\mathfrak{A}$.

In Chapter 4 we use the properties of $\mathfrak{A}$ to give a spectral decomposition of M. As M is a real-valued, symmetric matrix, it possesses real eigenvalues and is orthogonally diagonalizable with a complete set of $4n^2$ orthonormal eigenvectors spanning $\mathbb{R}^{4n^2}$. The chapter's primary objective is to construct bases of the eigenspaces of M. Finally, as a key ingredient of the main theorem, the chapter explores the eigenprojectors of M which also reside in $\mathfrak{A}$, emphasizing their usefulness in computing a generalized inverse for M and establishing a norm bound for the generalized inverse.

Chapter 5 shifts our attention away from the empty Sudoku, instead focusing on proving the existence of a fractional completion for a partially filled Sudoku, given certain conditions. We begin by estimating the behaviour of a generalized inverse for perturbed versions of M corresponding to non-empty partial Sudokus. Then Lemma 5.3.1 is presented, giving the conditions on these perturbations such that they will still allow an entrywise non-negative solution to our relevant linear system. The chapter concludes by piecing all of these elements together into a concise proof of our main result, Theorem 1.2.4.

The final chapter, Chapter 6, details an added result for Sudoku with 'small' fixed w, which we say have *thin boxes*, Proposition 6.1.5. This was interesting to us, as our main result relies on letting both h and w grow sufficiently large but it is entirely possible to have a very large Sudoku with thin boxes. As thin boxes imply greater similarity to the Latin square case, we were interested to determine if a density condition could be found that relied only on the Latin square definition of ϵ-density. Although this was not the main focus of our research, it is an interesting potential continuation of the project. Finally, we conclude with remarks on our research methodology, findings of the thesis, and potential directions for future work.

Chapter 2
Preliminaries

2.1 Key terminology and historical context

In this chapter, we begin by representing Latin squares and the Sudoku variation as graphs. Consequently, we offer some background information on the essential graph-theoretical terms required to understand this methodology.

A graph of order n, denoted as $G = (V, E)$, consists of three essential components: a *vertex set* $V(G)$ with a cardinality of n, an *edge set* $E(G)$, and a *relation* that connects each edge to exactly two vertices. We use the term *endpoints* to describe the vertices connected by a specific edge, denoted as e. If the vertices u and v are the endpoints of the edge e, we may denote it as uv or vu depending on the context. When an edge uv is part of the edge set $E(G)$, we say that vertex u is *adjacent* to vertex v, and we consider the edge uv to be *incident* to both u and v.

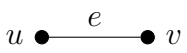


Figure 2.1: An edge e incident to endpoints u and v.

The *degree* of a vertex, $d(v)$, is defined as the number of edges incident to it, while the *minimum degree* of a graph, $\delta(G)$, is the smallest degree among all its vertices. If there exists a subset of vertices, $\mathcal{I}$, where no edges connect any pair of vertices within $\mathcal{I}$, we call $\mathcal{I}$ an *independent set*. Figure 2.2 shows a graph, denoted P_2, with three vertices, u, v, and w with $d(v) = 2$ and $d(u) = d(w) = 1$. Note that $\delta(P_2) = 1$ and that $\{u, w\}$ is an independent set.

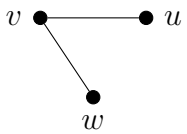


Figure 2.2: An illustration of P_2.

In *simple* graphs, each pair of vertices are connected with a maximum of one edge, and no edge is created between a single vertex and itself, which is called a *loop*. A *multi-graph* is a graph where any pair of distinct vertices can be connected by any number of edges. For a multi-graph G^+, we say that the *multiplicity* of an edge uv, denoted $m(uv)$, is the number of times an edge between vertices u and v occurs in $E(G^+)$.

It is worth noting that most of the graphs in this thesis are simple, so we use the term *graph* to mean a simple graph. In Section 2.2 we do briefly introduce a potential multi-graph representation of Sudoku, but we denote all multi-graphs with an superscripted + sign to indicate that they are not simple.

A *complete graph*, denoted by K_n, is a graph where every pair of distinct vertices is connected by an edge. In simpler terms, a complete graph consists of n vertices, and there exists an edge between every pair of vertices. As an illustrative example, Figure 2.3 displays the complete graph K_4.

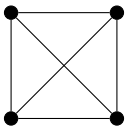
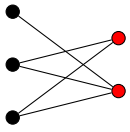
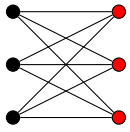


Figure 2.3: A complete graph on 4 vertices; K_4.

An *m-partite graph* is a type of graph whose vertices are partitioned into precisely m independent sets. When each set of the partition has an equal number of vertices, we refer to the m-partite graph as *balanced*. In this thesis, we emphasize balanced m-partite graphs because they are well-suited for modeling Latin squares and Sudokus. A *complete m-partite graph* is a simple m-partite graph where all vertices within each independent set are connected to every vertex in every other independent set. We can denote a complete m-partite graph as $K_{n_1,n_2,\ldots,n_m}$, with n_i denoting the number of vertices in independent set i. Figures 2.4a and 2.4b both show 2-partite graphs, often called *bipartite* graphs, with partite sets in red and black. Note that Figure 2.4b shows $K_{3,3}$ which is both complete and balanced.



(a) A bipartite graph with partite sets in red and black.



(b) Complete balanced bipartite graph, $K_{3,3}$.

A *subgraph* of a graph G, denoted as $H \subseteq G$, is a graph in which both the vertex set, $V(H)$, and the edge set, $E(H)$, are subsets of the corresponding sets in G. Figure 2.5 shows a red K_3 subgraph of K_4 as an example of this concept.

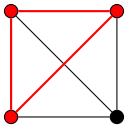


Figure 2.5: A red K_3 subgraph of K_4.

An *isomorphism* between a graph G and another graph G' is a function $f : V(G) \to V(G')$ such that $uv \in E(G)$ implies $f(u)f(v) \in E(G')$ and there exists a function $f^{-1} : V(G') \to V(G)$ such that for all $v \in V(G)$, $f^{-1}(f(v)) = v$. When an isomorphism exists between G and G', we say that G and G' are *isomorphic*, denoted as $G \simeq G'$. Figure 2.6 shows an example of isomorphic graphs G and G' with isomorphism $f : G \to G'$ with $f(1) = a, f(2) = b, f(3) = c, f(4) = d$, and $f(5) = e$.

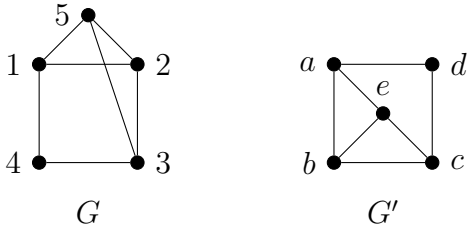
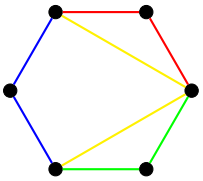
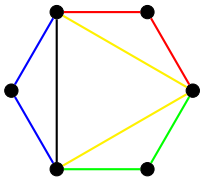


Figure 2.6: Two isomorphic graphs, G and G'.

An *F-decomposition of a graph G* is a partition of $E(G)$ into subgraphs $\{F_1, F_2, ..., F_n\}$, so that each subgraph in the set is isomorphic to F. Figure 2.7a gives a P_2-decomposition of a graph G, with each subgraph isomorphic to P_2 depicted in green, red, blue, and yellow.



(a) Graph G allowing a P_2-decomposition.



(b) Graph G' which is not P_2-divisible.

Figure 2.7: Comparison of two graphs with respect to P_2-decomposition.

For the existence of an F-decomposition of graph G, two necessary conditions must be met:

- *Divisibility of Edge Counts:* The number of edges in G must be divisible by the number of edges in F.
- *Vertex Degree Compatibility:* The degree of every vertex in G must be divisible by the greatest common divisor of the degrees of vertices in F.

A graph G that satisfies these two conditions is described as *F-divisible*. Let us take a brief detour to explore why *F-divisibility* is necessary for the existence of an F-decomposition. First, let G and F be graphs such that the number of edges in G is not divisible by the number of edges in F. Then, even if we do not worry about maintaining the structure of F, if we partition the edges of G into sets of size $|E(F)|$, we will be left with a remainder of edges. As an illustration, Figure 2.7b shows a graph G' with 9 edges, which is not P_2-divisible and therefore does not have a P_2-decomposition.

Next, let G and F be graphs such that there exists an F-decomposition of G. Since $E(G)$ is partitioned into copies of F, we can fix some vertex $v \in V(G)$ which is incident to at least one edge. We can suppose without loss of generality that there are m copies of F in this F-decomposition of G which contain vertex v and that all of the edges adjacent to v belong to one of these copies of F. Since every vertex in each instance of F has a degree that is a multiple of the greatest common divisor, d, the degree of v in G must also be a multiple of d. As v was an arbitrarily chosen vertex in G, this condition must hold for all vertices of G. As a simple example of how an F-decomposition would fail if this condition is not met, one might imagine attempting to construct a K_3-decomposition of a graph G which has even one vertex of odd degree.

Although F-divisibility is necessary for the existence of an F-decomposition, it is not a sufficient condition. The search for sufficient conditions allowing F-decompositions for various specific F is an open area of research. A notable conjecture proposed by Nash-Williams [14] asks whether all K_3-divisible graphs of order n with a minimum degree of at least $\frac{3}{4}n$ admit a K_3-decomposition. As of today, this conjecture remains unanswered. However, in recent years, progress has been made on the minimum degree conditions for the existence of K_3-decompositions using fractional K_3-decompositions.

The notion of a fractional F-decomposition provides a way to represent a decomposition of a graph G into subgraphs isomorphic to F in a more flexible and fractional manner.

Definition 2.1.1. We say that a graph G admits a *fractional F-decomposition* if there exists a set of pairs, $\{(F_1, w_1), (F_2, w_2), \ldots, (F_n, w_n)\}$, where each F_i is a subgraph of G which is isomorphic to F and each weight w_i is a non-negative real number such that for each edge $e \in E(G)$, the sum of the weights of the copies of F that contain e as an edge is exactly 1. That is,

$$\sum_{i:e\in E(F_i)} w_i = 1.$$

Put more plainly, a fractional F-decomposition of a graph G is a way of decomposing G into a set of copies of F that are each assigned *weights* so that when you put all these copies of F together you get the entire graph. The weight of each copy of F represents the amount contributed to any edge that is a part of that copy. As a simple example, if an edge is present in both F_i and $F_{i'}$ with weights $w_i = 0.3$

and $w_{i'} = 0.7$, then that edge is fully accounted for without any excess in the set $\{(F_1, w_1), (F_2, w_2), \ldots, (F_n, w_n)\}$. Figure 4.9 gives a visual representation of a fractional K_3-decomposition of K_4 with each K_3 receiving a weight of 0.5. Quick examination reveals that no exact K_3-decomposition of K_4 exists.

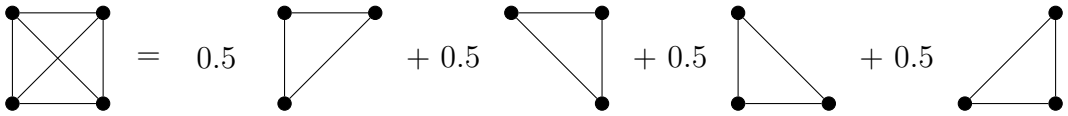


Figure 2.8: A visualization of a fractional K_3-decomposition of K_4.

A result of Barber, Kühn, Lo and Osthus in [2] connects the minimum degree threshold for F-decompositions of graphs with sufficiently large order to the minimum degree threshold for fractional F-decompositions. Combined with a result of Postle and Delcourt [7], which says that any graph G of order n with minimum degree at least $\frac{7+\sqrt{21}}{14}n$ admits a fractional K_3-decomposition, this gives the following theorem:

Theorem 2.1.2 (from [2] and [7]). *Let $\epsilon > 0$. Every sufficiently large K_3-divisible graph G with $\delta(G) \geq (\frac{7+\sqrt{21}}{14} + \epsilon)n \simeq (0.827 + \epsilon)n$ admits a K_3-decomposition.*

In this thesis our main focus is on finding a fractional completion threshold for Sudoku and, to do so, we are modelling our approach based on the work done by Bowditch and Dukes [5] on the Latin square completion problem. We therefore place a strong emphasis on K_3-decompositions of balanced tripartite graphs because of their direct relevance to the graph representations of both objects. This particular research question inspires one to wonder whether there is some partite analogue to the research presented in [2]. To address this specific scenario, Barber, Kühn, Lo, Osthus, and Taylor presented an m-partite variant of their result for K_m-decompositions [3].

A key observation in this setting is that if G is m-partite and one wants a K_m-decomposition of G, then every vertex in G must have the same number of neighbors in each of the other $(m-1)$ parts. To see why this is necessary, simply assume that G is m-partite and allows a K_m-decomposition. If $v \in V(G)$ is an endpoint of any edge, that edge must be included in some copy of K_m in that K_m-decomposition. This means that v must have an edge to each of the $(m-1)$ other parts. We call a graph which satisfies this property *locally balanced*. This condition is also necessary for the fractional relaxation, as we can apply the same sort of proof but with the appropriate weights on each copy of K_m. To reflect this condition, the authors of [3] use the term *K_m-divisible* to describe any balanced m-partite graph which is locally balanced.

In [3] Barber, Kühn, Lo, Osthus, and Taylor define $\tau_F(n, m)$ as the smallest τ such that every locally balanced m-partite graph G with $\delta(G) \geq \tau(m-1)n$ admits a fractional K_m-decomposition and set $\tau_F(m)$ as the limit when n approaches infinity of $\tau_F(n, m)$. With this definition, they give the following result:

Theorem 2.1.3 (from [3]). *If G is a balanced and locally balanced m-partite graph with $\delta(G) \geq (\tau_F(m) + \epsilon)(m-1)n$ for some $\epsilon > 0$ and large enough n, then G admits a K_m-decomposition.*

In their paper [5], Bowditch and Dukes show that $\tau_F(3) < 0.96$. This means that for large enough n, every locally balanced tripartite graph G on $3n$ vertices with $\delta(G)/2n \geq 0.96$ has a fractional K_3-decomposition. Their proof is based on using a linear system for assigning non-negative real weights to triangles in G so that the sum of weights on each edge is constant. For $K_{n,n,n}$ this system has a natural solution of assigning every position an equal portion of each symbol. They then show that as long as G is a sufficiently small perturbation of $K_{n,n,n}$, then there exists a non-negative solution. We now explain how these results directly correlate to Latin square completion conditions (fractional and otherwise), which we later extend to the Sudoku variation.

2.1.1 Latin squares as balanced tripartite graphs

To make the connection between K_3-decompositions of tripartite graphs and Latin square completions more concrete, we present a representation of Latin squares as balanced tripartite graphs, as used in [4] and [5]. This representation neatly captures the relationships between rows, columns, and symbols within a Latin square, ultimately allowing the application of graph theoretic concepts such as F-decompositions and fractional F-decompositions to the Latin square completion problem.

Consider a balanced tripartite graph, G_L, with $3n$ vertices, each belonging to one of the three sets: R for the n rows, C for the n columns, and S for the n symbols. That is,

$$V(G_L) = \{r_1, \ldots, r_n\} \cup \{c_1, \ldots, c_n\} \cup \{s_1, \ldots, s_n\} = R \cup C \cup S.$$

To define the edges in this representation, the focus is on the relationships between key components and identifying what is missing in order for partial Latin square L to be completed:

- If $L(i, j)$ is empty, then $r_i c_j \in E(G_L)$.
- If symbol s_k is missing in row r_i in L, then $r_i s_k \in E(G_L)$.
- If symbol s_k is missing in column c_j in L, then $c_j s_k \in E(G_L)$.

Simply put, the edges of G_L signify the absence of specific elements within L. By forming edges in this manner, the resulting balanced tripartite graph captures the requirements that must be fulfilled in order to form a completion of L in terms of edges.

1			4
3	4	1	2
2	1	4	3
4	3	2	1

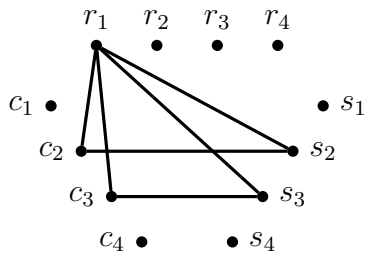


Figure 2.9: A partial Latin square of order 4, L, depicted alongside graph G_L.

This means that if an edge remains in G_L then there is a condition still unfulfilled for the completion of L. Figure 2.9 shows a simple example of a completable partial Latin square, L, alongside G_L. Note that, as defined, our graph representation of an empty partial Latin square of order n is $K_{n,n,n}$ and a completed Latin square of order n is represented by a graph on $3n$ vertices with no edges.

By construction, there are no edges within the parts R, C, and S, and as such we note that the only K_3 subgraphs possible must use exactly one vertex from each partite set. We see that any K_3 subgraph in G_L must represent a placement of an available symbol in a cell dictated by a row and column, neither of which already contain that symbol. Thus, G_L functions as a structured graph representation of the partial Latin square L and any K_3-decomposition of this graph G_L will represent a means of fulfilling these requirements.

Recall that we can represent a partial Latin square as a set L of ordered triples. In this representation, if symbol k appears in row i and column j of the partial Latin square, we include the ordered triple (r_i, c_j, s_k) in the set L. If L has been represented as the graph G_L, then any K_3 in G_L is similarly equivalent to an ordered triple. As such, any K_3-decomposition of G_L can also be represented as a set T of ordered triples, (r_i, c_j, s_k). Because the subgraphs in a K_3-decomposition are edge-disjoint, we can ascertain that each triple in T shares at most one coordinate with any other triple in T and with those ordered triples already present in L. Therefore, the union of sets L and T yields a completion of L.

Similarly, a fractional K_3-decomposition of G_L can be represented as a set T_f of ordered triples, (r_i, c_j, s_k) paired with non-negative weights w_{ijk}. In this case the ordered triples, (r_i, c_j, s_k), must still share at most one coordinate with any other triple L, but they need not maintain this condition with the other triples in T_f. Instead, we relax this requirement and ask only that for each row/column, row/symbol, or column/symbol pair the sum total of the weights of triples containing that pair is exactly 1. Combining L with T thus yields a fractional completion of L.

2.2 Sudoku as balanced 4-partite graphs

Similarly to Latin squares, a partial (h, w)-Sudoku can also be effectively represented as a set P of ordered triples. As before, if symbol k appears in row i and column j of the partial Sudoku, we include the ordered triple (r_i, c_j, s_k) in the set P. However, unlike the Latin square case, the requirement that these triples share at most one coordinate with any other triple in the set P is not sufficient to satisfy the Sudoku conditions. The condition on how often a symbol may appear in a given box must also be taken into account. There are two ways that we considered for doing so in the graph representation, and they are distinguished by how we establish the edges.

2.2.1 Explicit box relations and K_4-decompositions

Consider a balanced 4-partite multi-graph, G_P^+, with $4n$ vertices, each belonging to one of the four sets: R for the n rows, C for the n columns, S for the n symbols, and B for the n boxes. That is,

$$V(G_P^+) = \{r_1, \ldots, r_n\} \cup \{c_1, \ldots, c_n\} \cup \{s_1, \ldots, s_n\} \cup \{b_1, \ldots, b_n\} = R \cup C \cup S \cup B,$$

To define the edges in our representation, we identify what is missing in order for partial Sudoku P to be completed:

- If entry (i, j) is empty in P and $\text{box}(i, j) = \ell$, then $r_i c_j, r_i b_\ell, c_j b_\ell \in E(G_P^+)$. Furthermore, $m(r_i b_\ell)$ is equal to the number of empty cells in $r_i \cap b_\ell$ in P, and $m(c_j b_\ell)$ is equal to the number of empty cells in $c_j \cap b_\ell$ in P.
- If symbol s_k is missing in row r_i in P, then $r_i s_k \in E(G_P^+)$.
- If symbol s_k is missing in column c_j in P, then $c_j s_k \in E(G_P^+)$.
- If symbol s_k is missing in box b_ℓ in P, then $b_\ell s_k \in E(G_P^+)$.

1	2		
3	4	1	2
2	1	4	3
4	3	2	1

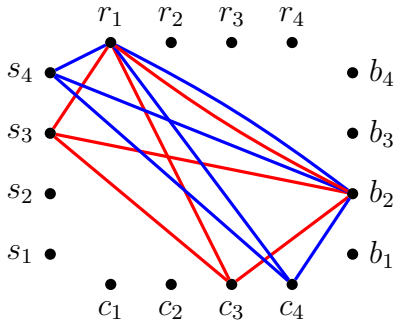


Figure 2.10: A partial $(2, 2)$-Sudoku, P, depicted alongside graph G_P^+.

As in the Latin square case, the edges of G_P^+ signify the unfulfilled conditions within P, effectively translating the puzzle requirements into a graph format. Thus, G_P^+ functions as a structured graph representation of the partial Sudoku and any K_4-decomposition of G_P^+ will represent a means of fulfilling these requirements. An example is illustrated in Figure 2.10 with a K_4-decomposition $\{K_4^{(1)}, K_4^{(2)}\}$ depicted with $K_4^{(1)}$ in blue and $K_4^{(2)}$ in red. This is written more formally as Proposition 2.2.1.

Proposition 2.2.1. *Any partial Sudoku P has a completion if and only if G_P^+ permits a K_4-decomposition.*

Proof. Suppose P is any partial Sudoku which has a completion, P'. For any empty cell (i, j) with $\text{box}(i, j) = \ell$ in P we can insert the corresponding symbol s_k from entry (i, j) in P' into that cell in P. In doing so, we remove an edge between each pair from $\{r_i, c_j, s_k, b_\ell\}$ from G_P^+ - thereby removing exactly one K_4 from G_P^+. If there are t empty cells in P and we do this for each of them then we have removed t copies of K_4, each of which would be necessarily be edge disjoint. There can be no remaining edges in G_P^+. Thus, the t copies of K_4 that were removed in this process are a K_4 decomposition of G_P^+.

Conversely, let P be any partial Sudoku and suppose G_P^+ allows a K_4-decomposition $\{F_1, F_2, ..., F_t\}$. Without loss of generality, look at any particular F_m. It must be that $V(F_m)$ has exactly one vertex from each of the partite sets in G and each edge in $E(F_m)$ represents an unfulfilled condition for completion. That is, there must be an empty cell where row r_i, column c_j, and box b_ℓ intersect and each of these must be missing symbol s_k. Thus, placing symbol s_k in cell (i, j) does not violate any of the necessary conditions. As this process can be followed for each $F_m \in \{F_1, F_2, ..., F_t\}$ and $\{F_1, F_2, ..., F_t\}$ is a K_4-decomposition of G_P^+ then this can be done repeatedly until there are no empty positions in the Sudoku and none of the Sudoku conditions have been violated. □

The model using the multi-graph G_P^+ has the advantage that the extra edges force all 4-cliques to correspond to valid tiles. However, since it is somewhat less convenient to work with a multi-graph and no new information is carried by the extra edges, we consider a second graph representation which omits the edges of type row-box and column-box.

2.2.2 Implicit box relations and $(K_3 + e)$ tile-decompositions

We again start by setting up a balanced 4-partite graph, G_P, as a representation of our partial Sudoku, P, with $4n$ vertices representing the different features of the Sudoku,

$$V(G_P) = \{r_1, \ldots, r_n\} \cup \{c_1, \ldots, c_n\} \cup \{s_1, \ldots, s_n\} \cup \{b_1, \ldots, b_n\} = R \cup C \cup S \cup B,$$

The edges of G_P are defined similarly to those in G_P^+. However, we ignore the relationships between rows and boxes and between columns and boxes because any r_ic_j edge will determine $\text{box}(i,j)$ through equation (1.1), so these edges do not add new information. We define the edges of G_P as follows:

- If entry (i,j) is empty in P then $r_ic_j \in E(G_P)$.
- If symbol s_k is missing in row r_i in P then $r_is_k \in E(G_P)$.
- If symbol s_k is missing in column c_j in P then $c_js_k \in E(G_P)$.
- If symbol s_k is missing in box b_ℓ in P then $b_\ell s_k \in E(G_P)$.

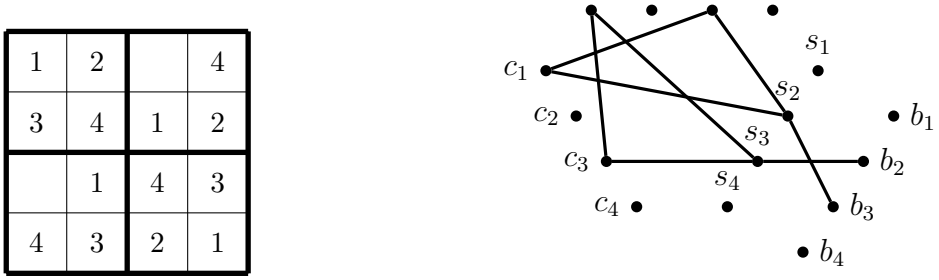


Figure 2.11: A partial $(2,2)$-Sudoku, P, depicted alongside graph G_P.

As with G_P^+, the edges of G_P signify the absence of specific elements within P. However, now when we add a symbol s_k to a cell (i,j) in the Sudoku P, we are not removing a K_4 from G_P. Instead, we are removing a triangle, $r_ic_js_k$ (analogous to the Latin square case), with an additional edge from the symbol s_k to box b_ℓ with $\ell = \text{box}(i,j)$. To capture this behavior, we define a graph $K_3 + e$ by its vertex set

$$V(K_3 + e) = \{r_i, c_j, s_k, b_\ell\}$$

and edge set

$$E(K_3 + e) = \{(r_i, c_j), (r_i, s_k), (c_j, s_k), (s_k, b_\ell)\}.$$

However, unlike the case of K_4 decompositions of G_P^+ where the row-box and column-box relations were explicitly defined, the existence of a $(K_3 + e)$-decomposition of G_P doesn't necessarily imply the existence of a completion of P. To address this, we introduce the concept of a tile.

Definition 2.2.2. A *tile*, t, is defined as a specific instance of $K_3 + e$ in G_P with vertex set $\{r_i, c_j, s_k, b_\ell\}$ where $\text{box}(i,j) = \ell$.

For convenience of notation, specific tiles are sometimes denoted by juxtaposing vertices without additional punctuation. That is, we may use short-hand to write a tile t as $r_i c_j s_k b_\ell$ when, in fact, t is the set of edges $\{r_i c_j, r_i s_k, c_j s_k, s_k b_\ell\} \subseteq E(G_P)$. We let $\mathcal{T}(G_P)$ denote the set of all tiles in G_P. This allows us to rework our notion of graph decompositions into something more suited for our purposes with G_P.

Definition 2.2.3. Given a partial Sudoku P and graph representation G_P, a *tile-decomposition* of G_P is a partition of $E(G_P)$ into tiles $\{t_1, t_2, ..., t_m\} \subseteq \mathcal{T}(G_P)$.

This notion of tile-decomposition paves the way for a slight variant of Proposition 2.2.1:

Proposition 2.2.4. *Any partial Sudoku, P, has a completion if and only if G_P permits a tile-decomposition.*

Proof. This proof is analogous to the one given for Proposition 2.2.1. The only difference is that the definition of a tile-decomposition imposes the condition that row r_i and column c_j are located within box b_ℓ, rather than enforcing this property with extra edges. □

As before, we can relax the notion of tile-decompositions through a fractional variation.

Definition 2.2.5. Given a partial Sudoku P and graph representation G_P, a *fractional tile-decomposition* of G_P is set of tile-weight pairs $\{(t_1, w_1), (t_2, w_2), ..., (t_m, w_m)\}$ such that each w_i is non-negative and, for each edge e in G_P, the sum of the weights of all tiles containing e is exactly 1. This is expressed as the sum:

$$\sum_{i:e \in t_i} w_i = 1.$$

Through a simple application of Definitions 2.2.5 and 1.2.2, we see that Proposition 2.2.4 implies the following corollary for fractional completions of partial Sudoku:

Corollary 2.2.6. *Any partial Sudoku, P, has a fractional completion if and only if G_P permits a fractional tile-decomposition.*

Equipped with this corollary, our next goal is to determine an ϵ-density threshold for a partial Sudoku P which guarantees the existence of a fractional tile-decomposition of G_P, and therefore a fractional completion of P. To do this, we begin by translating the graph representation for each partial Sudoku into an inclusion matrix. From this point onward it is assumed that the reader is familiar with basic definitions and principles of linear algebra, as would be seen in an introductory course on the subject.

2.3 Introduction to the inclusion matrix, W

An inclusion matrix for a graph G is a means of representing relationships between substructures in G as a matrix. We use the inclusion matrix as a tool for proving or disproving the existence of a tile-decomposition of G_P, so we construct an inclusion matrix that models the relationship between the edges, $E(G_P)$, and tiles, $\mathcal{T}(G_P)$.

Consider a case where G_P has m edges and k tiles within it. To construct our inclusion matrix, W_P, we create a binary matrix with dimensions $m \times k$. In this matrix, the rows are indexed by $E(G_P)$, and the columns are indexed by $\mathcal{T}(G_P)$. The elements of W_P are defined as follows:

$$W_P(e,t) = \begin{cases} 1 & \text{if } e \in t, \\ 0 & \text{otherwise.} \end{cases}$$

Now, let $\mathbb{1}$ denote the $m \times 1$ all-ones vector and consider the linear system,

$$W_P\mathbf{z} = \mathbb{1}. \tag{2.1}$$

A tile-decomposition of G_P is equivalent to a solution $\mathbf{z} \in \{0,1\}^k$ to this system. This is because each component $\mathbf{z}(t)$ of the vector $\mathbf{z}$ corresponds to including or excluding tile t. In this manner, we can think of $\mathbf{z}$ as a function that maps each element of the set $\mathcal{T}(G_P)$ to a binary value, either 0 or 1. If no such $\mathbf{z}$ exists, it indicates the absence of a valid H-decomposition for graph G. For example, Figure 2.12 shows a single row of $W_{P'}$, indexed by edge e, for some partial Sudoku P' with $G_{P'}$ containing 12 tiles and edge $e \in t_3$ and $e \in t_7$. In this case, we can deduce that if $\mathbf{z} \in \{0,1\}^{12}$ then it must be such that either component $\mathbf{z}(t_3) = 1$ or component $\mathbf{z}(t_7) = 1$, but not both. That is, if we are trying to construct a tile-decomposition of $G_{P'}$ we must include t_3 or t_7, but not both.

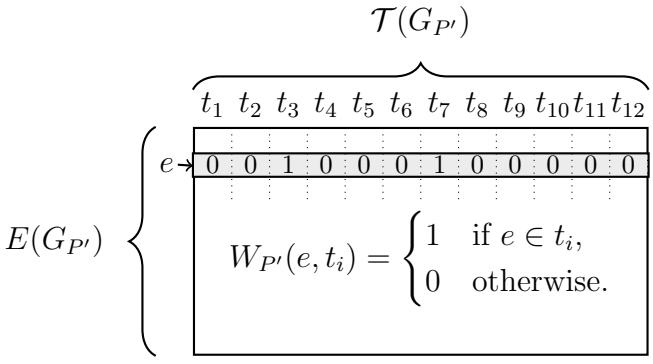


Figure 2.12: An illustration of a row of $W_{P'}$ for some partial Sudoku P' where $e \in t_3$ and $e \in t_7$.

A natural relaxation of this linear system is to consider $\mathbf{z}$ with non-negative entries, denoted as $\mathbf{z} \in \mathbb{R}^k_{\geq 0}$. This corresponds to a fractional tile-decomposition since, rather than merely including or excluding a tile t, each component $\mathbf{z}(t)$ of vector $\mathbf{z}$ represents the weight assigned to that tile. In this context, $\mathbf{z}$ functions as a mapping from the set $\mathcal{T}(G_P)$ to the non-negative reals. If we define $T_e \subseteq \mathcal{T}(G_P)$ to be the set of all tiles that contain edge e then the linear system (2.1) requires $\mathbf{z}$ to satisfy,

$$\sum_{t \in T_e} \mathbf{z}(t) = 1$$

for all edges $e \in E(G_P)$. For example, looking back at Figure 2.12 we can deduce that if $\mathbf{z} \in \mathbb{R}^k_{\geq 0}$ then it must be such that component $\mathbf{z}(t_3) = m$ for some $m \in [0, 1]$ and component $\mathbf{z}(t_7) = (1 - m)$. That is, if we are trying to construct a fractional tile-decomposition of $G_{P'}$ we must have $\mathbf{z}(t_3) + \mathbf{z}(t_7) = 1$.

When the linear system in Equation (2.1) has a valid solution, $\mathbf{z} \in \mathbb{R}^k_{\geq 0}$, it validates that the weight allocation adheres to the condition that the total weight assigned to each edge equals 1. Conversely, if no such $\mathbf{z}$ exists, it implies that graph G does not have a fractional tile-decomposition.

Now, we define the symmetric $m \times m$ matrix $M_P = W_P(W_P)^\top$ (sometimes called the Gram matrix of W_P). In this matrix, both the rows and columns are indexed by $E(G_P)$. The elements of M_P are defined as follows:

$$M_P(e, f) = |\{t \in \mathcal{T}(G_P) : e, f \in t\}|.$$

Any solution $\mathbf{x} \in \mathbb{R}^k_{\geq 0}$ to the linear system:

$$M_P \mathbf{x} = \mathbb{1}, \tag{2.2}$$

implies a solution $\mathbf{z} = W_P^\top \mathbf{x} \in \mathbb{R}^k_{\geq 0}$ to the linear system described in Equation (2.1). Again, we can think of the vector $\mathbf{x}$ as a function that maps the set $E(G)$ to non-negative weights from $\mathbb{R}^k_{\geq 0}$. Then $\mathbf{z} = (W_P)^\top \mathbf{x}$ will map tile t to $\sum_{e \in t} x_e$. That is, rather than assigning weights to the tiles directly, we assign weights to each edge and then assign each tile the sum of the weights assigned to each of its edges.

If M_P were invertible, we could guarantee the existence of a unique solution, $\mathbf{x} = M_P^{-1}\mathbb{1}$. However, M_P is not guaranteed to be invertible. In fact, in all cases where the linear system (2.2) has multiple solutions or no solution at all, M_P cannot be invertible. In the next section, we discuss the empty Sudoku which, of course, allows many fractional completions and therefore has multiple solutions to its corresponding linear system. A detailed discussion of its rank is presented in Section 2.4.1.

2.4 The linear system for empty Sudoku

In their paper on Latin square completion conditions, Bowditch and Dukes [5] showed that the partial Latin squares they were interested in were essentially small perturbations of the *empty* partial Latin square. We follow a similar methodology, beginning by narrowing our focus to the specific linear system for the empty (h, w)-Sudoku, G_{hw}. We define the graph G_{hw} as follows. Its vertex set is

$$V(G_{hw}) = \{r_1, \ldots, r_n\} \cup \{c_1, \ldots, c_n\} \cup \{b_1, \ldots, b_n\} \cup \{s_1, \ldots, s_n\},$$

with the four sets corresponding to rows, columns, boxes, and symbols, respectively. Its edge set is

$$E(G_{hw}) = \bigcup_{i,j=1}^{n} \{\{r_i, c_j\}, \{r_i, s_j\}, \{c_i, s_j\}, \{b_i, s_j\}\}. \tag{2.3}$$

In other words, exactly one edge is present for every combination of row-column, row-symbol, column-symbol, and box-symbol. As a point of notation, G_{hw} depends only on $n = hw$; indeed, if we omit indices in (2.3) it is seen to be simply a blow-up of the graph $K_3 + e$ by independent sets of size n. With this in mind, the subscript in G_{hw} can reasonably be interpreted as the product of h and w, though we find it useful to keep the parameters separate so that it is clear that G_{hw} is being discussed in connection with the (h, w)-Sudoku. Note that the subgraph of G_{hw} induced by rows, columns, and symbols is just the complete 3-partite graph $K_{n,n,n}$. Also notice that for any non-empty partial (h, w)-Sudoku P, the graph G_P is the graph obtained from G_{hw} by deleting the edges of tiles corresponding to the pre-filled entries in P.

In order to avoid unnecessary subscripts, we henceforth denote the edge-tile inclusion matrix for the empty Sudoku as W and we let $M = WW^\top$. Note that W is indexed by $E(G_{hw})$ and $\mathcal{T}(G_{hw})$ and that for any partial Sudoku P, the inclusion matrix W_P is a submatrix of W because $E(G_P) \subseteq E(G_{hw})$ and $\mathcal{T}(G_P) \subseteq \mathcal{T}(G_{hw})$.

Consequently, the linear system that we are focused on can be expressed as:

$$M\mathbf{x} = \mathbb{1}. \tag{2.4}$$

One can determine the (e, f) entry of M combinatorially as the number of tiles shared by both edges e and f. As M is the inclusion matrix for the empty Sudoku, we

can determine the entries through simple counting:

$$M(e,f) = \begin{cases} n & \text{if } e = f, \\ 1 & \text{if } e \neq f \text{ and } e, f \in \{r_i c_j, b_\ell s_k\} \text{ for any } i,j,k \in [n] \text{ st box}(i,j) = \ell, \\ 1 & \text{if } e \neq f \text{ and } e, f \in \{r_i c_j, c_j s_k\} \text{ for any } i,j,k \in [n], \\ 1 & \text{if } e \neq f \text{ and } e, f \in \{r_i c_j, r_i s_k\} \text{ for any } i,j,k \in [n], \\ 1 & \text{if } e \neq f \text{ and } e, f \in \{r_i s_k, c_j s_k\} \text{ for any } i,j,k \in [n], \\ w & \text{if } e \neq f \text{ and } e, f \in \{r_i s_k, b_\ell s_k\} \text{ for any } i,k,\ell \in [n], \\ h & \text{if } e \neq f \text{ and } e, f \in \{c_j s_k, b_\ell s_k\} \text{ for any } j,k,\ell \in [n], \\ 0 & \text{otherwise.} \end{cases} \tag{2.5}$$

In order to better understand the structure of M, it is beneficial to order the edges in a logical manner. For this purpose, we require a definition of lexicographic order.

Definition 2.4.1. Let A and B be two sets, and consider the Cartesian product $A \times B$. The lexicographic order on $A \times B$ is defined as follows: for any two pairs (a_1, b_1) and (a_2, b_2) from $A \times B$, we say $(a_1, b_1) < (a_2, b_2)$ if and only if one (or both) of the following conditions is met:

- $a_1 < a_2$ in the order of set A, or
- $a_1 = a_2$ and $b_1 < b_2$ in the order of set B.

Using this definition of lexicographic order, we fix an ordering of edges in W, and consequently in M, as follows:

- The first n^2 edges will be row-column edges, $r_i c_j$, ordered lexicographically.
- The second set of n^2 edges will be row-symbol edges, $r_i s_k$, ordered lexicographically.
- The third set of n^2 edges will be column-symbol edges, $c_j s_k$, ordered lexicographically.
- The final n^2 edges being the box-symbol edges, $b_\ell s_k$, ordered lexicographically.

That is, given two row-column edges $r_i c_j$ and $r_{i'} c_{j'}$, we first compare the rows r_i and $r_{i'}$. If, without loss of generality, $i < i'$ then $r_i c_j < r_{i'} c_{j'}$ lexicographically and therefore $r_i c_j$ is placed earlier than $r_{i'} c_{j'}$ in the edge ordering. If $r_i = r_{i'}$, we then compare the columns c_j and $c_{j'}$ to determine the ordering. If, without loss of generality, $j < j'$ then $r_i c_j < r_{i'} c_{j'}$ lexicographically and therefore $r_i c_j$ is placed earlier than $r_{i'} c_{j'}$ in the edge ordering.

With this edge ordering we see that the top-left $3n^2 \times 3n^2$ matrix $M[K_{n,n,n}]$, denoting M restricted to $K_{n,n,n}$, is a principal sub-matrix of M, as illustrated in Figure 2.13. Thus, $M[K_{n,n,n}]$ and the top $3n^2$ entries of $\mathbf{x}$ together give a simplified linear system corresponding to fractional Latin square completion, as seen in [5]. Restricting our linear system in this way means we retain the $3n^2$ Latin square conditions but eliminate the n^2 box conditions which are unique to Sudoku.

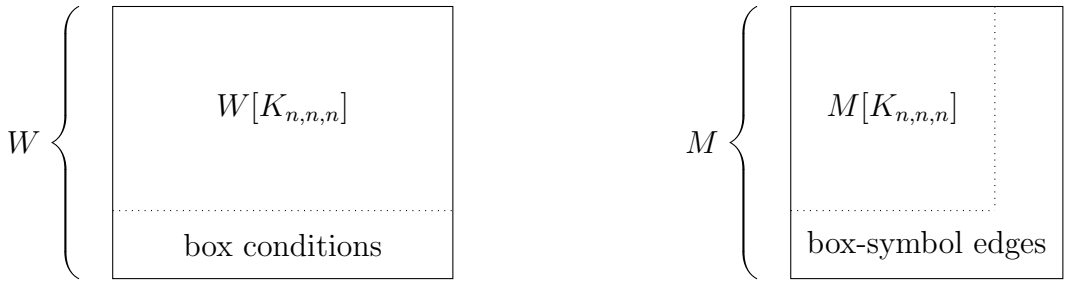


Figure 2.13: $W[K_{n,n,n}]$ and $M[K_{n,n,n}]$ as principal sub-matrices of W and M, respectively.

This arrangement of the edges of M allows us to interpret M as a block matrix consisting of sixteen $n^2 \times n^2$ blocks, dictated by two choices; the type of edges indexing the rows and the type of edges indexing the columns.

2.4.1 The rank of M

In this section we construct a set of $n^3 - (n-1)^3 + (n-1)(h-1)(w-1)$ linearly independent column vectors of W, directly corresponding to tiles in $\mathcal{T}(G_{hw})$, and show that all column vectors in W can be written as linear combinations of vectors from that set. This gives us the rank of W. It is a fundamental property of matrices that for a matrix A with entries from $\mathbb{R}$ or $\mathbb{C}$, we have $\text{rank}(A) = \text{rank}(AA^\top)$ [15, pg.63] and so, $\text{rank}(M)$ is also $n^3 - (n-1)^3 + (n-1)(h-1)(w-1)$. By showing that $\text{rank}(M) < 4n^2$ we confirm that the kernel of M, $\ker(M)$, is non-trivial, justifying the need for the additive shift we employ in Section 5.2. Also, knowing the rank of M allows us to confirm that the linearly independent sets of eigenvectors that we construct for each of M's nonzero eigenspaces in Chapter 4 imply the dimension of their corresponding eigenspace because the size of these sets together sum to $\text{rank}(M)$.

In the proof that follows we rely on a slight abuse of notation to remain concise. When referring to tiles in the graph $V(G_{hw})$, we omit conventional set notation and instead juxtapose vertices from different partite sets of $V(G_{hw})$. For example, a tile t with edges r_ic_j, r_is_k, c_js_k, and s_kb_ℓ will be denoted as $r_ic_js_kb_\ell$. Additionally, we describe the column vectors of matrix W using the tiles that they correspond to. Specifically, the column vector indexed with $r_ic_js_kb_\ell$ has entries indexed by r_ic_j, r_is_k, c_js_k, and s_kb_ℓ equal to 1 and all other entries as 0.

In the context of dealing with sparse vectors, we define the *support* of a vector to be the set of its nonzero elements. Hence, when we state that a vector $\mathbf{v}$ is supported by certain entries, we mean that these entries are nonzero. On the contrary, if we describe a vector as not supported by specific entries, it indicates that these entries are zero. So, we can refine our earlier statement by saying the column vector with corresponding tile $r_i c_j s_k b_\ell$ is supported by the entries $r_i c_j$, $r_i s_k$, $c_j s_k$, and $s_k b_\ell$.

Proposition 2.4.2. *We have* $\operatorname{rank}(M) = \operatorname{rank}(W) = n^3 - (n-1)^3 + (n-1)(h-1)(w-1)$.

Proof. Let $\mathcal{T}_1$ be the set of $n^3 - (n-1)^3$ tiles which intersect at least one of r_1, c_1, s_1. Each such tile corresponds with a column vector in W, each of which is indexed by the edges in G_{hw}. It was shown by Bowditch and Dukes in [5, Proposition 2.3] that $\mathcal{T}_1$ is linearly independent when W is restricted to the edges of $K_{n,n,n}$, rather than all of the edges of G_{hw} (in this case, we restrict each of the tiles to their K_3 counterparts, each still corresponding to a position in the (h, w)-Sudoku grid). As $E(K_{n,n,n}) \subset E(G_{hw})$, this tells us that $\mathcal{T}_1$ is also a linearly independent set of vectors in W.

For each $k = 2, \ldots, n-1$ and each of the $(h-1)(w-1)$ boxes that do not intersect either row 1 or column 1, pick a distinct tile represented by $r_i c_j s_k b_\ell$. Define $\mathcal{T}_2$ to be the set containing the $(n-1)(h-1)(w-1)$ column vectors corresponding to these tiles. Since each vector in $\mathcal{T}_2$ exhibits a unique box-symbol pairing, the set $\mathcal{T}_2$ is linearly independent. As no box-symbol pairing is represented in $\mathcal{T}_1$, it follows that the combined set $\mathcal{T}_1 \cup \mathcal{T}_2$ is linearly independent.

We next show that linear combinations of vectors from $\mathcal{T}_1 \cup \mathcal{T}_2$ can be used to generate any given column of W, say the one corresponding to tile $r_i c_j s_k b_\ell$. Suppose $i \leq h$ and $j \leq w$. Then $\ell = \operatorname{box}(i, j) = 1$. We can form the target tile as a linear combination in $\mathcal{T}_1$, namely as:

$$r_i c_j s_k b_1 = r_1 c_j s_k b_1 + r_i c_1 s_k b_1 + r_i c_j s_1 b_1 - r_1 c_1 s_k b_1 - r_1 c_j s_1 b_1 - r_i c_1 s_1 b_1 + r_1 c_1 s_1 b_1.$$

Suppose next that $i > h$ and $j \leq w$. This is the case where the column c_j belongs to the same column-bundle as c_1. As above, we have the linear combination

$$r_i c_j s_k b_\ell = r_1 c_j s_k b_1 + r_i c_1 s_k b_\ell + r_i c_j s_1 b_\ell - r_1 c_1 s_k b_1 - r_1 c_j s_1 b_\ell - r_i c_1 s_1 b_1 + r_1 c_1 s_1 b_1.$$

A similar calculation in the case when r_i belongs to the same row-bundle as r_1 shows that $\mathcal{T}_1$ generates any column vector of W corresponding with a tile with $i \leq h$ and $j > w$.

Suppose, then, that $i > h$ and $j > w$. This is the case where entry (i, j) is in one of the $(h-1)(w-1)$ boxes which do not intersect either row 1 or column 1. If $k = 1$, the corresponding column vector belongs to $\mathcal{T}_1$, so instead assume $k > 1$. Put $p = \operatorname{box}(1, j)$ and $q = \operatorname{box}(i, 1)$, and note that all tiles meeting these boxes correspond with column vectors in the span of $\mathcal{T}_1$, as shown above. Since $i > h$, $j > w$, and $k > 1$, we know that

s_k and b_ℓ occur together in some vector $r_{i'}c_{j'}s_kb_\ell \in \mathcal{T}_2$. Using this and other column vectors generated so far, we compute:

$$\begin{aligned} r_ic_js_kb_\ell = {} & r_ic_js_1b_\ell + r_1c_js_kb_p + r_ic_1s_kb_q - r_1c_1s_kb_1 - r_1c_js_1b_p - r_ic_1s_kb_q + r_{i'}c_{j'}s_kb_\ell \\ & - r_{i'}c_{j'}s_1b_\ell - r_1c_{j'}s_kb_p - r_{i'}c_1s_kb_q + r_1c_1s_kb_1 + r_1c_{j'}s_1b_p + r_{i'}c_1s_1b_q - r_1c_1s_1b_1. \end{aligned}$$

We have shown that $\mathcal{T}_1 \cup \mathcal{T}_2$ spans each column of W, and hence is a basis for range(W). □

Applying this alongside the rank-nullity theorem, we get a nice corollary which gives the dimension of $\ker(M)$.

Corollary 2.4.3. *We have* $\dim(\ker(W^\top)) = \dim(\ker(M)) = 3n + (n-1)(h+w)$.

Proof. Our matrix M is $4n^2 \times 4n^2$, so we apply Proposition 2.4.2 and the rank-nullity theorem to get:

$$\begin{aligned} \dim(\ker(W^\top)) &= 4n^2 - n^3 - (n-1)^3 + (n-1)(n-h-w+1) \\ &= 3n^2 - n^3 - (n-1)^3 + n^2 - (n-1)(n-h-w+1) \\ &= (3n-1) + n + (n-1)(h+w-1) \\ &= 3n + (n-1)(h+w). \end{aligned}$$

□

A robust discussion of $\ker(M)$ is included in Chapter 4, where we construct a set of basis vectors for each of the eigenspaces of M.

Chapter 3
The Sudoku Adjacency Algebra

The purpose of this chapter is to construct the Sudoku adjacency algebra that we use in our main result. However, in order to provide intuition and context for the methods of this construction, we begin with a short introduction to an algebraic structure called an association scheme followed by a generalization of this structure called a coherent configuration. We will then explore how these are applied in the Sudoku setting.

3.1 An introduction to association schemes

Let n be a natural number and consider the set $N \times N = \{(i,j) : i,j \in [n]\}$. We can partition this set using two fundamental relations: R_1 (equality) and R_2 (inequality) so that $(i,j) \in R_1$ if $i = j$ and $(i,j) \in R_2$ otherwise. These relations can be represented quite simply using the two $n \times n$ adjacency matrices, A_1 and A_2, defined as:

$$A_1(i,j) = \begin{cases} 1 & \text{if } i = j, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$A_2(i,j) = \begin{cases} 0 & \text{if } i = j, \\ 1 & \text{otherwise.} \end{cases}$$

Notice that A_1 and A_2 are symmetric matrices and that the sum $A_1 + A_2$ equals J, the all-ones $n \times n$ matrix.

Now, let us pivot for a moment and consider the set $S = \{aI + bJ : a, b \in \mathbb{R}\}$, where I is the $n \times n$ identity matrix. It is relatively straightforward to check that S satisfies the conditions of a vector space. Through a quick calculation we see that S is also closed under multiplication and that this operation is commutative. That is, for all $aI + bJ, cI + dJ \in S$ we have,

$$(aI+bJ)(cI+dJ) = (cI+dJ)(aI+bJ) = acI^2+(ad+bc)JI+bdJ^2 = acI+(ad+bc+bdn)J.$$

Together, these conditions tell us that S is an *algebra*: a vector space equipped with a bi-linear operation (in the case of S that operation is multiplication). Moreover, S is a *commutative algebra* because the multiplication operation is commutative. Now notice that any matrix in S can be expressed in terms of the adjacency matrices A_1 and A_2. Any matrix $aI + bJ \in S$ corresponds to:

$$(aI + bJ) = (a + b)A_1 + bA_2.$$

Therefore, the span of these adjacency matrices over $\mathbb{R}$ forms the commutative algebra, S. This simple example illustrates the foundational idea behind association schemes. A more formal definition of an association scheme is as follows:

Definition 3.1.1. Let X be a finite set. An *association scheme* on X is a partition of $X \times X$ into a set of nonempty symmetric binary relations $\mathcal{R} = \{R_1, \ldots, R_d\}$ satisfying the following properties:

1. R_1 is the identity relation, and
2. For each i, j, k, there exists a structure constant, p_{ij}^k, such that for any $x, z \in X$ with $(x, z) \in R_k$, there are precisely p_{ij}^k elements y such that $(x, y) \in R_i$ and $(y, z) \in R_j$.

As we did in our early example, for any given association scheme on set X we can consider the corresponding adjacency matrix A_i, associated with each relation index $i = 1, 2, \ldots, d$, and indexed by the elements of X:

$$A_i(x, x') = \begin{cases} 1 & \text{if } (x, x') \in R_i, \\ 0 & \text{otherwise.} \end{cases}$$

Since the relations partition $X \times X$, we have

$$\sum_{i=1}^{d} A_i = J,$$

and by applying the definition of the structure constants we see that,

$$A_i A_j = \sum_{k=1}^{d} p_{ij}^k A_k.$$

The algebra generated by these adjacency matrices is called the *Bose-Mesner* algebra of X.

In [5], Bowditch and Dukes show that the edge relations in $M[K_{n,n,n}]$ constitute an association scheme and, consequently, $M[K_{n,n,n}]$ is in the commutative *Bose-Mesner*

algebra on $E(K_{n,n,n})$. They then use the properties of the algebra to find a generalized inverse for $M[K_{n,n,n}]$. Applying this generalized inverse enabled them to find conditions which guarantee the existence of a solution to the linear system (2.2) when restricted to $E(K_{n,n,n})$, resulting in a fractional completion threshold for Latin squares.

When we attempted to apply this same method in the Sudoku setting, we found that the edge relations in M did not define an association scheme. Instead, the relations in M define a more general object called a coherent configuration.

Definition 3.1.2. Let X be a finite set. A *coherent configuration* on X is a partition of $X \times X$ into a set of nonempty binary relations $\mathcal{R} = \{R_1, \ldots, R_d\}$ satisfying the following properties:

1. The diagonal $\{(x, x) : x \in X\}$ is the union of certain relations in $\mathcal{R}$;
2. For each relation R in $\mathcal{R}$, its transpose $R^\top = \{(y, x) : (x, y) \in R\}$ is also contained in $\mathcal{R}$;
3. For each i, j, k, there exists a structure constant, p_{ij}^k, such that for any x, z with $(x, z) \in R_k$, there are precisely p_{ij}^k elements y such that $(x, y) \in R_i$ and $(y, z) \in R_j$.

As with an association scheme, in a coherent configuration the relations partition $X \times X$, so we again have the property:

$$\sum_{i=1}^{d} A_i = J.$$

Also, as the condition for structure constants is the same as in an association scheme, we also have:

$$A_i A_j = \sum_{k=1}^{d} p_{ij}^k A_k.$$

However, there are two key differences between an association scheme and a coherent configuration. They are as follows:

1. A coherent configuration does not require relations to be symmetric, only that their transpose is also in $\mathcal{R}$.
2. A coherent configuration allows the identity to be made up of the union of various relations, rather than requiring a sole relation to be the identity.

Unlike $K_{n,n,n}$, which has symmetric (and therefore interchangeable) relations on the edges between rows, columns and symbols; in the setting of G_{hw} the four types of edges have more complex relations and are not interchangeable. This meant we needed four separate relations to make up the diagonal $\{(x, x) : x \in E(G_{hw})\}$. These relations will be examined in further detail in the sections to follow.

3.2 A coherent configuration for Sudoku

We now set up a coherent configuration on the ground set $X = E(G_{hw})$ and use it to define a *Sudoku adjacency algebra*, $\mathfrak{A}$. To do this, we begin with an examination of the relations between individual vertices.

3.2.1 Equivalence relations on $V(G_{hw}) \times V(G_{hw})$

We define the relation $\sim$ to denote whether two rows or columns are in the same row or column bundle. So, given two rows r_i and $r_{i'}$, we write $r_i \sim r_{i'}$ if and only if $\lfloor (i-1)/h \rfloor = \lfloor (i'-1)/h \rfloor$. Similarly, given two columns c_j and $c_{j'}$, we write $c_j \sim c_{j'}$ if and only if $\lfloor (j-1)/w \rfloor = \lfloor (j'-1)/w \rfloor$. We write $r_i \ncong r_{i'}$ if $r_i \sim r_{i'}$ but $r_i \neq r_{i'}$ and we similarly define $\ncong$ for columns. The relation $\sim$ is an equivalence relation on both rows and columns.

The relation $\smile$ is used to denote whether two boxes intersect a common set of rows. Given two boxes b_ℓ and $b_{\ell'}$, we write $b_\ell \smile b_{\ell'}$ if $\lfloor (\ell-1)/h \rfloor = \lfloor (\ell'-1)/h \rfloor$. Similarly, we let the relation $\frown$ track whether two boxes occur in common columns with $b_\ell \frown b_{\ell'}$ being valid if and only if $\ell \equiv \ell' \pmod{h}$. Both $\smile$ and $\frown$ are equivalence relations on pairs of boxes. We extend this notation, somewhat informally, to pairs of rows and boxes, $r_i \smile b_\ell$, to mean that the corresponding row and box intersect, and similarly for columns and boxes, $c_j \frown b_\ell$.

	rows	cols	symbols	boxes
rows	$=, \ncong, \nsim$			$\smile, \not\smile$
cols		$=, \ncong, \nsim$		$\frown, \not\frown$
symbols			$=, \neq$	
boxes	$\smile, \not\smile$	$\frown, \not\frown$		$=, \smile, \frown, \not=$

Table 3.1: Relations on vertices of G_{hw}

The ordered pairs of edges are now categorized into the 69 edge relations formed by all the relevant combinations of individual vertex relations, as shown in Table 3.1. A blank within the table indicates the trivial relation.

3.2.2 Relations on $E(G_{hw}) \times E(G_{hw})$ forming a coherent configuration

Now, transitioning from vertices to edges, these classifications induce a partition of $E(G_{hw})^2$ into relations according to Figure 3.1, with the number of possible relation types between any two edge types given in Table 3.2.

	$r_i c_j$	$r_i s_k$	$c_j s_k$	$b_\ell s_k$
$r_{i'} c_{j'}$	9	3	3	4
$r_{i'} s_{k'}$	3	6	2	4
$c_{j'} s_{k'}$	3	2	6	4
$b_{\ell'} s_{k'}$	4	4	4	8

Table 3.2: Relations on vertices of G_{hw}

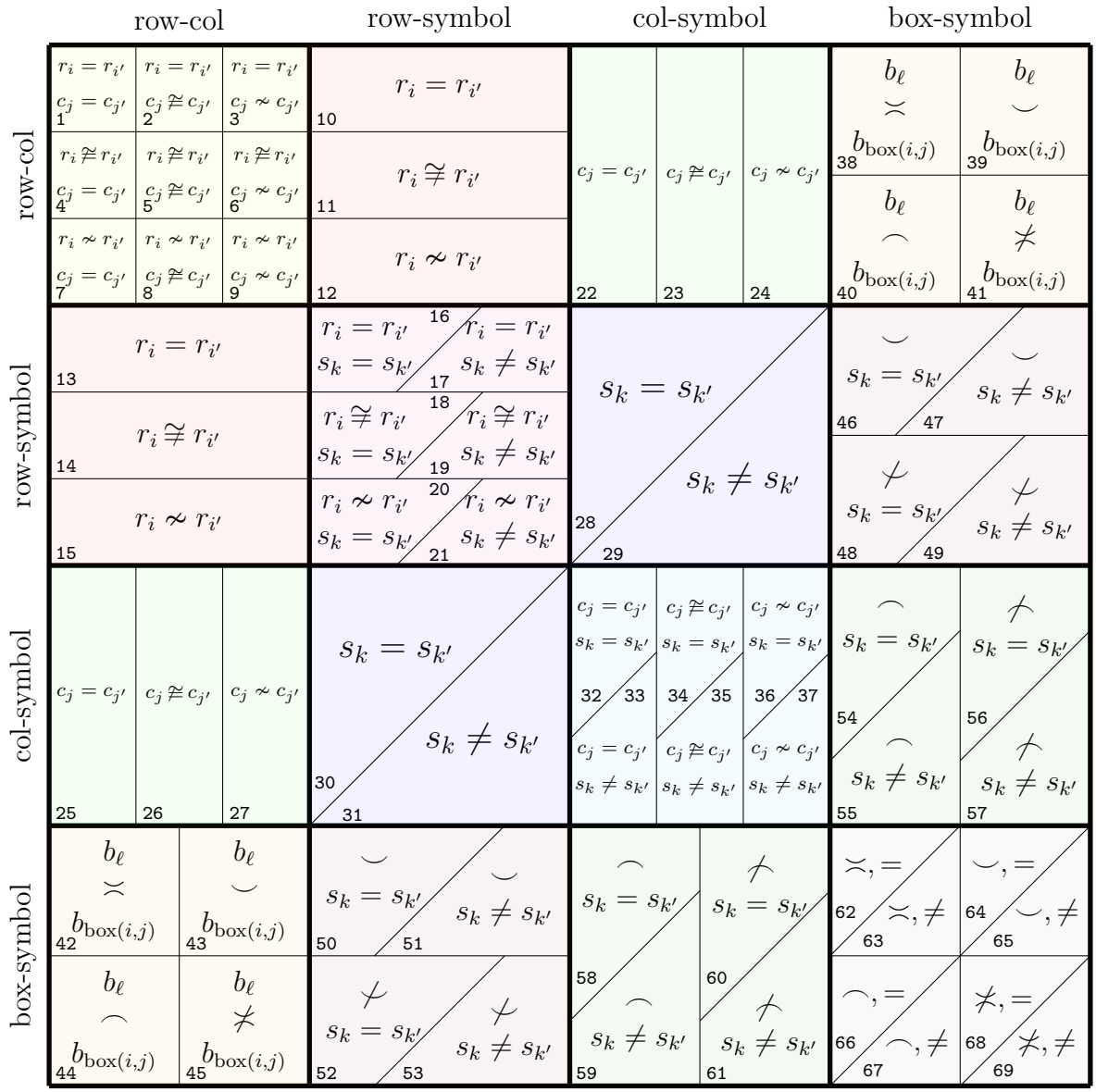


Figure 3.1: Relations on edges of G_{hw}.

Proposition 3.2.1. *The relations $R_1, \ldots, R_{69}$ given in Figure 3.1 define a coherent configuration on $E(G_{hw})$.*

Proof. Checking the first condition, we see that the diagonal relation $\{(x, x) : x \in E(G_{hw})\}$ is a union of the relations $R_1, R_{16}, R_{32}, R_{62}$. These four relations are the instances of equality for each of the four edge types: row-column, row-symbol, column-symbol, and box-symbol, respectively.

As for the second condition, for any relation R_i the transpose $R_i^\top$ is another relation in our family. We do not need to further explain that this condition is met, as the index of each R_i and corresponding $R_i^\top$ can be read off directly from Figure 3.1. For example, $R_{22}^\top = R_{25}$.

Checking the third condition amounts to showing that the number of choices $y \in E(G_{hw})$ with $(x, y) \in R_i$ and $(y, z) \in R_j$ depends only on i, j, k and not on the specific choice of x and z with $(x, z) \in R_k$. Consider the four (or fewer) vertices in $x \cup z$. We argue that there is a canonical choice depending on the relation label k. If $x \cup z$ contains exactly one symbol, we may assume without loss of generality that it is s_1. On the other hand, if x and z each contain a symbol, we may assume either a common symbol s_1 or distinct symbols $s_1 \neq s_2$, depending on k.

A similar canonical choice can be made for rows, except that unequal rows could be related under $\sim$, in which case our canonical choice is r_1, r_2, or not, in which case our canonical choice is r_1, r_{h+1}. The same holds for columns. Whenever a box b_ℓ appears in $x \cup z$, we can take a canonical choice that aligns with the decisions made for rows, columns, or potentially another box. Importantly, none of the preceding choices impact the count of elements y satisfying the given relations with x and z. □

With extensive case analysis, it would be possible to demonstrate formulas for the structure constants p_{ij}^k. However, to avoid presenting such details and to reduce the possibility of error, we implemented a computer-assisted procedure to calculate explicit structure constants, and then used interpolation to determine the symbolic expressions. For this purpose, we first bound the degrees of the polynomial for each structure constant, p_{ij}^k.

Proposition 3.2.2. *In our coherent configuration based on $E(G_{hw})$, each structure constant p_{ij}^k is a polynomial of degree at most 2 in each of h and w.*

Proof. Fix two edges $x, z \in E(G_{hw})$ with $(x, z) \in R_k$. The value p_{ij}^k gives a count of all edges $y \in E(G_{hw})$ with $(x, y) \in R_i$ and $(y, z) \in R_j$. This quantity is zero unless the relations R_i and R_j simultaneously allow one of the four types of edges for y.

Given relations R_i and R_j which admit a choice of y, we must choose either a row-column pair, a row-symbol pair, a column-symbol pair, or a box-symbol pair. The two components of each pair can be selected separately, leading to a product of choices for the two components.

The number of choices for a row is an element of $\{0, 1, h-2, h-1, h, n-2h, n-h, n\}$. These values come directly from the relations which induce the coherent configuration and from the canonical choices described in Proposition 3.2.2. The 0 and 1 come from tests of equality whereas the $h-2, h-1$, and h come from tests relating to sharing a bundle. The $n-2h$ and $n-h$ come from relations which require different bundles and, finally, the n comes from the case where no relation is required and any choice of a row will do. Analogously, the number of choices for a column is an element of $\{0, 1, w-2, w-1, w, n-2w, n-w, n\}$.

Similarly, the number of choices for a symbol is an element of $\{0, 1, n-2, n-1, n\}$ where, again, the 0 and 1 come from tests of equality. However, symbols do not have a relation relating to bundles. Instead, they have the relation of non-equality resulting in $n-2$ and $n-1$. As in the case of rows, the n comes from the case where no relation is required and any choice of a symbol will do.

Finally, the number of choices for a box is a product of an element of $\{0, 1, h-2, h-1, h\}$ with an element of $\{0, 1, w-2, w-1, w\}$. This is because the choice of both a row-bundle and a column-bundle equates to the choice of a box. Thus, let us go through the choices of a row-bundle. As before, 0 and 1 come from tests of equality and $h-2, h-1$, and h come from tests relating to sharing a bundle. The choices of a column-bundle are analogous.

Recalling that the structure constants are the product of available choices for the two components, we note that the number of choices for any single component is of degree at most 1 in each of h and w. Thus, their product is of degree at most 2 in each of h and w. □

For the sake of clarity, we now provide explicit examples to illustrate the potential choices for a component of edge y, based on the relations R_i, R_j, and R_k, as discussed in the above proof. We begin with concrete examples for each of the number of choices available for a row component of y.

- 0: No row satisfies the conditions imposed by both R_i and R_j. For example, if $x = r_1c_1$ and $z = r_2s_1$ with $(x, z) \in R_{11}$ then there is no row choice r for edge y that gives $(x, y) \in R_1$ and $(y, z) \in R_{10}$ because R_1 requires $r = r_1$ and R_{10} requires $r = r_2$.

- 1: Exactly one row meets the conditions of both R_i and R_j. For example, if $x = r_1c_1$ and $z = r_1c_1$ with $(x, z) \in R_1$ then there is one row choice r for edge y that gives $(x, y) \in R_{10}$ and $(y, z) \in R_{13}$ because R_{10} requires $r = r_1$ and R_{13} requires $r = r_1$.

- $h-2$: All but two of the rows in a bundle satisfy the conditions. For example, if $x = r_1c_1$ and $z = r_2c_2$ with $(x, z) \in R_4$ then there are $h-2$ choices of row r for edge y that gives $(x, y) \in R_5$ and $(y, z) \in R_6$ because R_5 requires $r \ncong r_1$ and

R_6 requires $r \ncong r_2$ so r can be any of the other rows from the same bundle as r_1 and r_2.

- $h-1$: All but one of the rows in a bundle satisfy the conditions. For example, if $x = r_1c_1$ and $z = r_1c_1$ with $(x, z) \in R_1$ then there are $h-1$ choices of row r for edge y that gives $(x, y) \in R_4$ and $(y, z) \in R_5$ because both R_4 and R_5 require $r \ncong r_1$ so r can be any of the other rows from the same bundle as r_1.

- h: Every row in a particular bundle satisfies the conditions of both relations. For example, if $x = b_1s_1$ and $z = b_2s_1$ with $(x, z) \in R_{62}$ then $b_1 \smile b_2$ and there are h choices of row r for edge y that gives $(x, y) \in R_{42}$ and $(y, z) \in R_{38}$ because R_{42} requires $r \smile b_1$ and R_{38} requires $r \smile b_2$. Thus, r can be any of the rows from the bundle that intersects both boxes.

- $n-2h$: Every row satisfies the conditions, except those in one of two possible specific bundles. For example, if $x = b_1s_1$ and $z = b_ns_1$ with $(x, z) \in R_{66}$ then $b_1 \not\smile b_n$ so there are $n-2h$ choices of row r for edge y that gives $(x, y) \in R_{45}$ and $(y, z) \in R_{41}$ because R_{45} requires $r \not\smile b_1$ and R_{38} requires $r \not\smile b_n$. Thus, r can be any of the rows, excepting those from one of the two bundles that intersect one of the two boxes.

- $n-h$: Every row except those in a specific bundle satisfies the conditions. For example, if $x = b_1s_1$ and $z = b_2s_1$ with $(x, z) \in R_{62}$ then $b_1 \smile b_2$ so there are $n-h$ choices of row r for edge y that gives $(x, y) \in R_{45}$ and $(y, z) \in R_{41}$ because R_{45} requires $r \not\smile b_1$ and R_{38} requires $r \not\smile b_2$. Thus, r can be any of the rows, excepting those from the bundle that intersects both boxes.

- n: Every row satisfies the conditions. For example, if $x = c_1s_1$ and $z = c_1s_1$ with $(x, z) \in R_{32}$ then there are n choices of row r for edge y that gives $(x, y) \in R_{28}$ and $(y, z) \in R_{29}$ because R_{28} and R_{29} only place restrictions on the symbol in edge y, meaning that r can be any of the n rows.

Now we provide a concrete example of relations R_i, R_j, and R_k for each of the number of choices available for a symbol component of y.

- 0: No symbol satisfies the conditions imposed by both R_i and R_j. For example, if $x = r_1s_1$ and $z = r_1s_2$ with $(x, z) \in R_{17}$ then there is no symbol choice s for edge y that gives $(x, y) \in R_{28}$ and $(y, z) \in R_{30}$ because R_{28} requires $s = s_1$ and R_{30} requires $s = s_2$.

- 1: Exactly one symbol meets the conditions of both R_i and R_j. For example, if $x = r_1s_1$ and $z = r_1s_1$ with $(x, z) \in R_{16}$ then there is one symbol choice s for edge y that gives $(x, y) \in R_{28}$ and $(y, z) \in R_{30}$ because both R_{28} and R_{30} require $s = s_1$.

- $n-2$: All but two symbols meets the conditions of both R_i and R_j. For example, if $x = r_1 s_1$ and $z = r_n s_2$ with $(x, z) \in R_{21}$ then there are $n-2$ symbol choices s for edge y that gives $(x, y) \in R_{21}$ and $(y, z) \in R_{19}$ because R_{21} requires $s \ncong s_1$ and R_{19} requires $s \ncong s_2$ so s can be any symbol except for s_1 or s_2.

- $n-1$: All but one of the symbols meets the conditions of both R_i and R_j. For example, if $x = r_1 s_1$ and $z = r_1 s_1$ with $(x, z) \in R_{16}$ then there are $n-1$ symbol choices s for edge y that gives $(x, y) \in R_{21}$ and $(y, z) \in R_{19}$ because R_{19} and R_{21} both requires $s \ncong s_1$ so s can be any symbol except for s_1.

- n: Every symbol satisfies the conditions of both relations. For example, if $x = r_1 c_1$ and $z = r_2 c_2$ with $(x, z) \in R_5$ then there are n choices of symbol s for edge y that gives $(x, y) \in R_{13}$ and $(y, z) \in R_{11}$ because R_{11} and R_{13} place no restrictions on the symbol choice.

Finally, we provide explicit examples of relations R_i, R_j, and R_k for each of the number of choices available for a box component of y. We do this with just the number of choices for a row-bundle, as the column-bundle cases are analogous and the choices for the box component result from a product of the row-bundle and column-bundle selections.

- 0: No row-bundle satisfies the conditions imposed by both R_i and R_j. For example, if $x = r_1 s_1$ and $z = r_1 s_1$ with $(x, z) \in R_{16}$ then there is no row-bundle choice for edge y that gives $(x, y) \in R_{46}$ and $(y, z) \in R_{53}$ because R_{46} requires $b_\ell \smile r_1$ and R_{53} requires $b_\ell \not\smile r_1$.

- 1: Exactly one row-bundle meets the conditions of both R_i and R_j. For example, if $x = r_1 s_1$ and $z = r_1 s_1$ with $(x, z) \in R_{16}$ then there is one row-bundle choice for edge y that gives $(x, y) \in R_{46}$ and $(y, z) \in R_{51}$ because R_{46} and R_{51} both require $b_\ell \smile r_1$.

- $h-2$: All but two row-bundles meet the conditions of both R_i and R_j. For example, if $x = r_1 s_1$ and $z = r_n s_2$ with $(x, z) \in R_{21}$ then there are $h-2$ row-bundle choices for edge y that gives $(x, y) \in R_{48}$ and $(y, z) \in R_{52}$ because R_{48} requires $b_\ell \not\smile r_1$ and R_{52} requires $b_\ell \not\smile r_n$ so the row-bundle chosen can be any row-bundle except the one containing r_1 or the one containing r_n.

- $h-1$: All but one of the row-bundles meets the conditions of both R_i and R_j. For example, if $x = r_1 s_1$ and $z = r_1 s_2$ with $(x, z) \in R_{16}$ then there are $h-1$ row-bundle choices for edge y that gives $(x, y) \in R_{48}$ and $(y, z) \in R_{52}$ because R_{48} and R_{52} require $b_\ell \not\smile r_1$ so the row-bundle chosen can be any row-bundle except the one containing r_1.

- h: Every row-bundle satisfies the conditions of both relations. For example, if $x = c_1 s_1$ and $z = c_1 s_3$ with $(x, z) \in R_{33}$ then any row-bundle choice for edge y gives $(x, y) \in R_{55}$ and $(y, z) \in R_{60}$ because R_{55} and R_{60} place no restrictions on the row-bundle choice.

The degree bound in Proposition 3.2.2 enabled us to construct a Vandermonde matrix with columns indexed by $\{1, h, w, h^2, hw, w^2, h^2w^2\}$, thereby facilitating the interpolation of symbolic structure constants from the computationally straightforward cases: $2 \leq h, w \leq 4$. The computation of all structure constants for these small cases was carried out on computer by explicitly listing all $4n^2$ edges and counting incidences. From this, we were able to build a vector of computed structure constants, $\mathbf{d}$, indexed by these nine cases. We then interpolated to find symbolic expressions using the 9×9 Vandermonde matrix, V, illustrated in Table 3.3. Solving the linear system $V \cdot \mathbf{c} = \mathbf{d}$ gave us the vector of coefficients, $\mathbf{c}$, for the above terms. Readers interested in examining the complete list of all symbolic structure constants computed in this way can find it on our GitHub page, https://github.com/pbd345/sudoku, listed under 'structure_constants.sage.' Note that in this file the structure constants p_{ij}^k are indexed lexicographically in the form of ordered triples, (i, j, k).

	h^0w^0	h^1w^0	h^0w^1	h^2w^0	h^1w^1	h^0w^2	h^2w^1	h^2w^2	h^1w^2
$h = 2, w = 2$	1	2	2	4	4	4	8	16	8
$h = 2, w = 3$	1	2	3	4	6	9	12	36	18
$h = 2, w = 4$	1	2	4	4	8	16	16	64	32
$h = 3, w = 2$	1	3	2	9	6	4	18	36	12
$h = 3, w = 3$	1	3	3	9	9	9	27	81	27
$h = 3, w = 4$	1	3	4	9	12	16	36	144	48
$h = 4, w = 2$	1	4	2	16	8	4	32	64	16
$h = 4, w = 3$	1	4	3	16	12	9	48	144	36
$h = 4, w = 4$	1	4	4	16	16	16	64	256	64

Table 3.3: The 9×9 Vandermonde Matrix, V, used for interpolation.

3.2.3 Constructing the Sudoku adjacency algebra $\mathfrak{A}$

For each relation index $i = 1, \ldots, 69$, we can consider its corresponding adjacency matrix A_i, which is a square matrix with the same indexing as M with entries:

$$A_i(e, f) = \begin{cases} 1 & \text{if } (e, f) \in R_i, \\ 0 & \text{otherwise.} \end{cases}$$

Let $\mathfrak{A}$ denote the ($\mathbb{Q}$, $\mathbb{R}$, or $\mathbb{C}$)-vector space that is spanned by these A_i. The conditions of a coherent configuration ensure that the product of any two of these adjacency

matrices is a linear combination of other matrices in the set, and therefore is in span($\mathfrak{A}$). This closure under multiplication, along with regular addition and scalar multiplication, satisfies the definition of an algebra. We therefore henceforth refer to $\mathfrak{A}$ as the *Sudoku adjacency algebra*.

As an interesting aside, if we view each relation as a graph, then $\{R_i : i = 1, \ldots, 69\}$ is a decomposition of the line graph of G_{hw} into degree-regular graphs.

Definition 3.2.3. For any graph G the line graph of G, denoted as $L(G)$, is defined as follows:

- *Vertices of $L(G)$:* Each vertex v_e of $L(G)$ represents an edge $e \in E(G)$.
- *Edges of $L(G)$:* Two distinct vertices in $L(G)$ are connected by an edge if and only if their corresponding edges in G are incident to a common vertex in G. That is, for $e \neq f$, $\{v_e, v_f\} \in E(L(G)) \Leftrightarrow e \cap f \neq \emptyset$.

The degree, $\deg(R_i)$, for each regular graph R_i is the nonzero row sum of the corresponding adjacency matrix, A_i. We give the degrees for each of the relations in Table 3.4. For clarity, these have been arranged into columns according to the four edge types: row-column, row-symbol, column-symbol, and box-symbol. Consider, for example, the degree $\deg(R_{47})$. Given a row-symbol edge, say $r_1 s_1$, this degree counts the symbol-box edges $s_k b_\ell$ with $s_k \neq s_1$ and b_ℓ in the same row bundle as r_1. There are $n-1$ choices for s_k and h choices for b_ℓ, since every row is incident with exactly h boxes. So $\deg(R_{47}) = (n-1)h$.

3.3 The structure of coefficient matrix M

Recall that W denotes the $\{0,1\}$ inclusion matrix of edges versus tiles in G_{hw}, and that $M = WW^\top$. We computed the rank in Section 2.4.1 and will construct a basis for the kernel of M in Section 4.1.4. A key observation is that M belongs to our adjacency algebra.

Proposition 3.3.1. *The matrix $M = WW^\top$ lies in $\mathfrak{A}$, with*

$$\begin{aligned} M =& hw(A_1 + A_{16} + A_{32} + A_{62}) + w(A_{46} + A_{50}) + h(A_{54} + A_{58}) \\ &+ A_{10} + A_{13} + A_{22} + A_{25} + A_{28} + A_{30} + A_{38} + A_{42}. \end{aligned} \tag{3.1}$$

Proof. Given two edges e, f in G_{hw}, the entry $M(e, f)$ counts the number of tiles containing both e and f. Since a tile contains exactly one row, column, box, and symbol, this number is zero whenever $e \cup f$ contains two unequal vertices in any of these categories. Moreover, since the box in a tile must correspond to the choice of row and column, $M(e, f)$ is zero if $e \cup f \supset \{r_i, b_\ell\}$ or $\{c_j, b_\ell\}$ with, respectively, $r_i \not\sim b_\ell$ or

row-column		row-symbol		column-symbol		box-symbol	
i	$\deg(R_i)$	i	$\deg(R_i)$	i	$\deg(R_i)$	i	$\deg(R_i)$
1	1	13	n	25	n	42	n
2	$w-1$	14	$n(h-1)$	26	$n(w-1)$	43	$n(h-1)$
3	$(h-1)w$	15	$nh(w-1)$	27	$n(h-1)w$	44	$n(w-1)$
4	$h-1$	16	1	30	n	45	$n(h-1)(w-1)$
5	$(h-1)(w-1)$	17	$n-1$	31	$n(n-1)$	50	h
6	$(h-1)^2w$	18	$h-1$	32	1	51	$(n-1)h$
7	$h(w-1)$	19	$(n-1)(h-1)$	33	$n-1$	52	$h(w-1)$
8	$h(w-1)^2$	20	$h(w-1)$	34	$w-1$	53	$(n-1)h(w-1)$
9	$n(h-1)(w-1)$	21	$(n-1)h(w-1)$	35	$(n-1)(w-1)$	58	w
10	n	28	n	36	$(h-1)w$	59	$(n-1)w$
11	$n(h-1)$	29	$(n-1)n$	37	$(n-1)(h-1)w$	60	$(h-1)w$
12	$nh(w-1)$	46	h	54	w	61	$(n-1)(h-1)w$
22	n	47	$(n-1)h$	55	$(n-1)w$	62	1
23	$n(w-1)$	48	$h(w-1)$	56	$(h-1)w$	63	$n-1$
24	$n(h-1)w$	49	$(n-1)h(w-1)$	57	$(n-1)(h-1)w$	64	$h-1$
38	n					65	$(n-1)(h-1)$
39	$n(h-1)$					66	$w-1$
40	$n(w-1)$					67	$(n-1)(w-1)$
41	$n(h-1)(w-1)$					68	$(h-1)(w-1)$
						69	$(n-1)(h-1)(w-1)$

Table 3.4: Relation degrees $\deg(R_i)$; alternatively the nonzero row sums of A_i.

$c_j \not\frown b_\ell$. It suffices to consider those remaining cases when there do exist tiles extending $e \cup f$.

If $e = f$, we claim that there are n such tiles, regardless of the type of edge. For $e = \{r_i, c_j\}$, any of the n symbols extend e to a tile (and there is a unique box involved). For $e = \{r_i, s_k\}$, there are any of n columns (and one corresponding box for each) extending e. This is similar when we exchange the roles of rows and columns. Finally, for $e = \{b_\ell, s_k\}$, any of the n cells (i, j) with $\text{box}(i, j) = \ell$ extend e to a tile. The identity relation in our setup decomposes into the identity on the four types of edges; in terms of matrices,

$$I = A_1 + A_{16} + A_{32} + A_{62}.$$

We have shown that the diagonal entries of M, and hence the coefficient for each of these four adjacency matrices, equals n.

Next, consider $e = \{r_i, s_k\}$ and $f = \{b_\ell, s_k\}$. In the event that $r_i \smile b_\ell$, we obtain w possible column indices j such that $\text{box}(i, j) = \ell$, and $e \cup f \cup \{c_j\}$ defines a valid tile. The two relations corresponding to this choice of e and f (transposes of each other) have indices 46 and 50 in our labelling. If instead we take $e = \{c_j, s_k\}$ for $c_j \frown b_\ell$, there are likewise exactly h extensions to a tile via a row index. This choice corresponds to relations numbered 54 and 58.

Finally, in each of the following possibilities for $\{e, f\}$, there is a unique tile extending $e \cup f$:

$$\{r_i c_j, r_i s_k\}, \{r_i c_j, c_j s_k\}, \{r_i s_k, c_j s_k\}, \{r_i c_j, b_{\mathrm{box}(i,j)} s_k\}.$$

The corresponding relation labels are (in pairs) $10, 13, 22, 25, 28, 30, 38, 42$, as numbered in Figure 3.1. □

As a concrete visual of the structure of M, we give an illustration in the case of the $(2, 3)$-Sudoku with graph $G_{2,3}$ in Figure 3.2. In this image the intensity of blue corresponds to the extremity of the positive values.

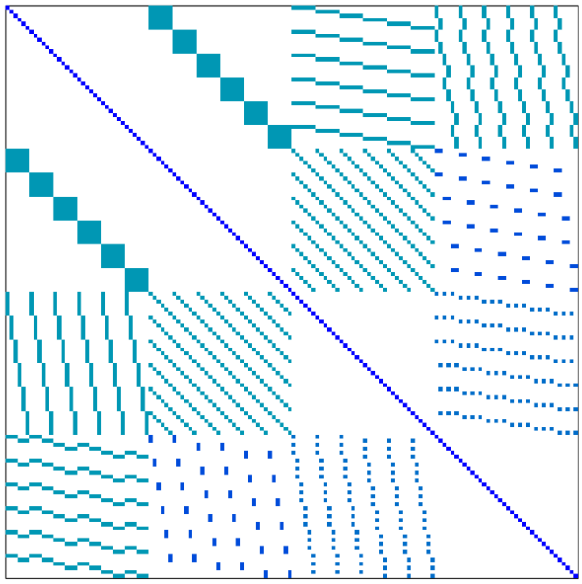


Figure 3.2: Structure of M for $G_{2,3}$.

Chapter 4
A Spectral Decomposition of M

In this chapter, we delve into a spectral decomposition of the matrix $M = WW^\top$, explicitly constructing a set of five integral eigenvalues and constructing a basis of eigenvectors for each such eigenvalue. By Equation (3.1), we know that $M \in \mathfrak{A}$. In Section 4.6.3 a generalized inverse for M is expressed with a list of coefficients in $\mathfrak{A}$. For the discovery of these coefficients, it is helpful to have a good understanding of the spectral decomposition of M.

Since $M = WW^\top$ and W has only $\{0, 1\}$ entries, we know M is a real symmetric matrix and therefore has real eigenvalues. The spectral theorem of symmetric matrices guarantees that any real symmetric matrix A can be orthogonally diagonalized. A consequence of this is the existence of a complete set of $4n^2$ orthonormal eigenvectors for M which are linearly independent and span $\mathbb{R}^{4n^2}$.

From Section 2.4, we recall that zero is an eigenvalue of M with multiplicity $3n + (h + w)(n - 1)$. This chapter begins with the construction of an explicit basis for the nullspace/kernel of M, equivalently the left nullspace of W^T. We do so by constructing a set of linearly independent kernel vectors and then showing that this set has cardinality equal to $\dim(\ker(M))$, as determined by Corollary 2.4.3. As this set of linearly independent kernel vectors matches the dimension of $\ker(M)$, it must constitute a basis of $\ker(M)$.

We then move on to the nonzero eigenvalues of M, first noting that M has constant row sums equal to $4n$, since every edge belongs to n tiles, and every tile has four edges. This gives an eigenvalue $4n$ corresponding to the one-dimensional eigenspace of constant vectors for which the all-ones vector on $4n^2$ entries is a basis. We compute all other nonzero eigenvalues for M and find a corresponding set of linearly independent eigenvectors for each eigenvalue. By calculating the sum of the cardinalities of these linearly independent sets from each eigenspace, we confirm the dimension of each of these eigenspaces collectively. We do so by verifying that their total dimension is equal to the rank of M, as given by $\text{rank}(M)$. This approach demonstrates how the nonzero eigenspaces contribute to the matrix's overall structure. This is summarized in the following proposition, with verification appearing in the remainder of the chapter.

Proposition 4.0.1. *The eigenvalues of M are $\theta_j = jn$, $j = 0, 1, \ldots, 4$ and we denote the eigenspace for eigenvalue θ_j as $\mathcal{E}_j$. Each eigenspace has a basis of eigenvectors consisting of vectors with entries in $\{0, \pm 1\}$.*

Note that as a direct consequence of this proposition and the fact that M is symmetric, we can apply the spectral theorem for symmetric matrices to conclude that

$$\sum_{j=0}^{4} \dim(\mathcal{E}_j) = 4n^2.$$

Within each eigenspace, eigenvectors are categorized based on distinct types, often defined by a primary characteristic of a Sudoku puzzle, like a row, column, symbol, box, bundle, or a combination of these elements. An informal explanation and brief validation is provided for every type of eigenvector, demonstrating that each proposed eigenvector $\mathbf{v}$ satisfies $M\mathbf{v} = \theta_j \mathbf{v}$.

In each such case, the verification follows the same approach. Take each edge $f \in E(G_{hw})$ and extend it to a tile $t \supseteq f$ in all possible ways. Then, sum the values of entries in $\mathbf{v}$ on the four edges of t and check that this total equals $\theta_j \mathbf{v}(f)$. This often equals zero, either from cancelling signs or when the support of $\mathbf{v}$ is disjoint from the relevant tiles t. For these arguments, we will regularly abuse notation by using t to mean the set $\{r_i, c_j, s_k, b_\ell, r_i c_j, r_i s_k, c_j s_k, s_k b_\ell\}$ corresponding to tile t induced by r_i, c_j, s_k, and b_ℓ. This is so we can say things like $r_i \in t$ or $c_j s_k \in t$, meaning that a vertex or edge is in a tile, in a clear, concise way.

For each eigenspace respectively, Figures 4.1, 4.5, 4.6, and 4.7 provide diagrams illustrating the eigenvector varieties in the case $(h, w) = (2, 3)$. In these diagrams, the four sections index the four edge types: row-column (top left), row-symbol (top right), symbol-column (bottom left), and box-symbol (bottom right). Symbols $+$ and $-$ denote vector entries 1 and -1, respectively, and blanks represent 0 in the corresponding positions. These figures are included largely because they give an nice visual illustration of the orthogonality of eigenvectors of different types within a given eigenspace.

4.1 Eigenspace $\mathcal{E}_0 = \ker(M)$ for eigenvalue $\theta_0 = 0$

From a combinatorial perspective, we view $\ker(M)$ as the set of all edge-weightings in G_{hw} for which each tile has a vanishing total weight over its four edges.

4.1.1 Type A_0: The row and column $\mathcal{E}_0$ vectors.

We construct the type A_0 vectors by first choosing a distinguished row, column or symbol vertex. Suppose, for now, that we chose a row, r_i. The entries of a row-

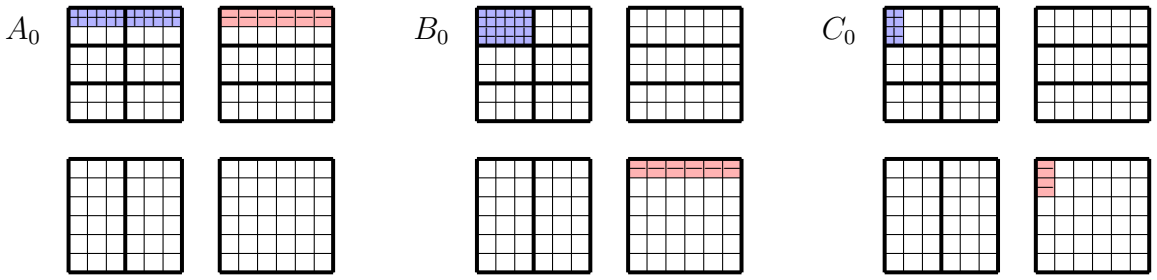


Figure 4.1: Eigenvectors for $\mathcal{E}_0$

determined type A_0 vector are as follows:

$$\mathbf{v}_{r_i}(e) = \begin{cases} 1 & \text{if } e = r_i c_j \text{ for some } c_j, \\ -1 & \text{if } e = r_i s_k \text{ for some } s_k, \\ 0 & \text{otherwise.} \end{cases}$$

In plain language, $\mathbf{v}_{r_i}$ encodes the condition that the number of times any column is used with row r_i is equal to the number of times a symbol is used in r_i.

Let $\mathbf{v}_{r_i}$ be any row-determined type A_0 vector. To confirm that $\mathbf{v}_{r_i}$ belongs to $\ker(M)$, we use the characteristic vector $\mathbf{t}$ of an arbitrarily selected tile t. If $r_i \in t$, then since t contains exactly one column and one symbol, we have $\mathbf{t} \cdot \mathbf{v}_{r_i} = 1 - 1 = 0$. On the other hand, if $r_i \notin t$, the support of t is disjoint from the support of $\mathbf{v}_{r_i}$, so we again have $\mathbf{t} \cdot \mathbf{v}_{r_i} = 0$. This confirms that all row-determined type A_0 vectors are necessarily in $\ker(M)$.

An example of a row-determined kernel vector in graph form is shown in Figure 4.2 with $(r_i c_j)$ edges shown in blue and $(r_i s_k)$ edges shown in red to indicate the weightings.

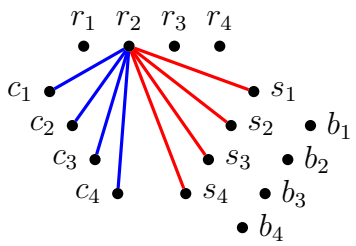


Figure 4.2: A graph representation of the type A_0 vector $\mathbf{v}_{r_2}$ for $G_{2,2}$.

Similarly, to build a column-determined type A_0 vector we fix a column c_j for $j \in [n]$ and define $\mathbf{v}_{c_j}$ as:

$$\mathbf{v}_{c_j}(e) = \begin{cases} 1 & \text{if } e = c_j s_k \text{ for some } s_k, \\ -1 & \text{if } e = r_i c_j \text{ for some } r_i, \\ 0 & \text{otherwise.} \end{cases}$$

Since the column-determined type A_0 vectors are constructed symmetrically to the row-determined type A_0 vectors, the proof of their inclusion in $\ker(M)$ is analogous and is therefore omitted.

There are $2n$ vectors of type A_0, as any row or column determines a unique type A_0 vector. It is quite straightforward to see that all $2n$ of the type A_0 vectors are linearly independent because they each involve a unique row (or column), and therefore have disjoint supports on row-symbol (or column-symbol) edges.

4.1.2 Type B_0: The box $\mathcal{E}_0$ vectors.

We construct the type B_0 vectors by first choosing a box, b_ℓ. We can then define the vector $\mathbf{v}_{b_\ell}$ as follows:

$$\mathbf{v}_{b_\ell}(e) = \begin{cases} 1 & \text{if } e = r_i c_j \text{ for some } r_i \text{ and } c_j \text{ such that } r_i \smile b_\ell \text{ and } c_j \frown b_\ell \\ -1 & \text{if } e = b_\ell s_k \text{ for some } s_k, \\ 0 & \text{otherwise.} \end{cases}$$

A type B_0 vector encodes the condition that the number of times a position is used in a given box is equal to the number of times a symbol is used in that same box. Figure 4.3 depicts a type B_0 vector centered on b_1 in $G_{2,2}$ as a graph with blue edges having weight $+1$ and red edges having weight -1.

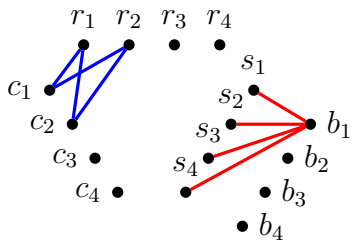


Figure 4.3: A graph representation of the type B_0 vector $\mathbf{v}_{b_1}$ for $G_{2,2}$.

To confirm that $\mathbf{v}_{b_\ell}$ belongs to the kernel of M, we again utilize the characteristic vector $\mathbf{t}$ of a tile t. If $b_\ell \in t$, then we have $\mathbf{t} \cdot \mathbf{v}_{b_\ell} = 1 - 1 = 0$ because t must also contain exactly one row-column edge and exactly one symbol. On the other hand, if $b_\ell \notin t$, the support of t is disjoint from the support of $\mathbf{v}_{b_\ell}$, hence we again have $\mathbf{t} \cdot \mathbf{v}_{b_\ell} = 0$. This confirms that all type B_0 vectors are in $\ker(M)$.

It is quite straightforward to see that all n of the type B_0 vectors are linearly independent because they each involve a unique box, and therefore have disjoint supports on box-symbol edges. Similarly, because the type A_0 vectors are linearly independent and have no supports on box-symbol edges, it follows that the combined set of all type A_0 and type B_0 vectors is linearly independent.

4.1.3 Type C_0: The bundle $\mathcal{E}_0$ vectors.

To build a row-bundle type C_0 vector, we first fix a particular symbol, s_k, and a row bundle $a \in [w]$. We define R_a to be the set containing all rows from that chosen row-bundle *and* all boxes which intersect the rows from that row-bundle. We can then define the vector $\mathbf{v}_{s_k,R_a}$ as follows:

$$\mathbf{v}_{s_k,R_a}(e) = \begin{cases} 1 & \text{if } e = r_i s_k \text{ for some } r_i \text{ such that } r_i \in R_a, \\ -1 & \text{if } e = b_\ell s_k \text{ for some } b_\ell \text{ such that } b_\ell \in R_a, \\ 0 & \text{otherwise.} \end{cases}$$

Similarly, we build a column-bundle type C_0 vector by again fixing a symbol s_k and a column-bundle $b \in [h]$. As before, we define C_b to be the set containing all columns from that chosen column-bundle *and* all boxes which intersect the columns from that row-bundle. We define the vector $\mathbf{v}_{s_k,C_b}$ as follows:

$$\mathbf{v}_{s_k,C_b}(e) = \begin{cases} 1 & \text{if } e = c_j s_k \text{ for some } c_j \text{ such that } c_j \in C_b, \\ -1 & \text{if } e = b_\ell s_k \text{ for some } b_\ell \text{ such that } b_\ell \in C_b, \\ 0 & \text{otherwise.} \end{cases}$$

Informally, $\mathbf{v}_{s_k,R_a}$ encodes the condition that the number of times symbol s_k is used in a row from a row-bundle, a, must equal the number of times s_k is used in a box which intersects that row-bundle. Figure 4.4 depicts a row-bundle type C_0 vector with blue edges having weight $+1$ and red edges having weight -1.

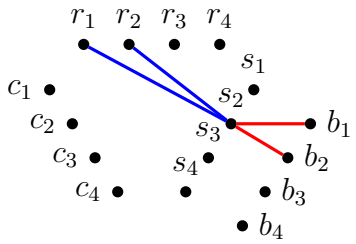


Figure 4.4: A graph representation of the type A_0 vector $\mathbf{v}_{s_2,R_1}$ for $G_{2,2}$.

To confirm that $\mathbf{v}_{s_k,R_a} \in \ker(M)$, we again utilize the characteristic vector $\mathbf{t}$ of an arbitrarily selected tile t. If $s_k \in t$ and for $r_i \in t$ we have $r_i \in R_a$ then $\mathbf{t} \cdot \mathbf{v}_{s_k,R_a} =$

$1 - 1 = 0$. On the other hand, if $s_k \neq s_{k'}$ or for $r_i \in t$, $r_i \notin R_a$, then the support of $\mathbf{t}$ is disjoint from the support of $\mathbf{v}_{s_k,R_a}$ and we again have $\mathbf{t} \cdot \mathbf{v}_{s_k,R_a} = 0$. The proof that $\mathbf{v}_{s_k,C_b}$ belongs to the kernel of M is analogous.

Of course, each $\mathbf{v}_{s_k,R_a}$ contains unique row-symbol supports and each $\mathbf{v}_{s_k,C_b}$ contains unique column-symbol supports, so the set of all $(h+w)n$ type C_0 vectors is linearly independent.

4.1.4 Constructing a basis for $\ker(M) = \mathcal{E}_0$

In this section, we construct a basis for $\ker(M) = \mathcal{E}_0$. To do this, we first determine the dimensions of the subspaces generated by vector types A_0, B_0, and C_0. We then examine how these dimensions relate to the overall dimension of $\ker(M)$, as determined in Corollary 2.4.3:

$$\dim(\ker(M)) = 3n + (h+w)(n-1).$$

Finally, we identify a linearly independent subset of these vectors with cardinality equal to $3n + (h+w)(n-1)$, thereby forming a basis for $\ker(M)$.

Lemma 4.1.1. *Let $\mathcal{E}_0^{\mathcal{A}}, \mathcal{E}_0^{\mathcal{B}}, \mathcal{E}_0^{\mathcal{C}}$ be the subspaces of $\mathbb{R}^{4n^2}$ spanned by the vectors of type A_0, B_0, and C_0 respectively. Then, the following dimensions are established:*

a) $\dim \mathcal{E}_0^{\mathcal{A}} = 2n$;

b) $\dim \mathcal{E}_0^{\mathcal{B}} = n$;

c) $\dim(\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}}) = 3n$;

d) $\dim \mathcal{E}_0^{\mathcal{C}} = (h+w)n$.

Proof. The dimensions of the subspaces are a direct consequence of the linear independence of their respective generating sets. □

The subspace intersection lemma, which can be found in [13, pg.4], states that for any finite-dimensional vector space V with subspaces U_1 and U_2:

$$\dim(U_1 + U_2) = \dim U_1 + \dim U_2 - \dim(U_1 \cap U_2).$$

Therefore, since $\ker(M)$ is a finite dimensional vector space, in order to show that the vectors of type A_0, B_0, and C_0 span $\ker(M)$, we let $(\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}})$ and $\mathcal{E}_0^{\mathcal{C}}$ take the roles of U_1 and U_2 respectively and bound the size of their intersection.

Lemma 4.1.2. $\dim((\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}}) \cap \mathcal{E}_0^{\mathcal{C}}) \leq h + w$.

Proof. Consider a vector $\mathbf{u}$ in the intersection $(\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}}) \cap \mathcal{E}_0^{\mathcal{C}}$.

First, observe that every vector of type A_0 or B_0 is invariant under symbol permutation. That is, for the vectors of type A_0 the row-symbol entries depend only on the row, and the column-symbol entries depend only on the column. Similarly, for the vectors of type B_0 the box-symbol entries depend only on the box. Therefore, the vector $\mathbf{u} \in \mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}}$ must be invariant under symbol permutation.

By assumption, $\mathbf{u}$ also lies in $\mathcal{E}_0^{\mathcal{C}}$ and can therefore be represented as a linear combination of type C_0 vectors. However, this does not change the fact that $\mathbf{u}$ is invariant under symbol permutation. This means that $\mathbf{u}$ must be expressible as a linear combination of a limited set of vectors. Specifically, for any row-bundle $a \in [h]$ the vector $\mathbf{v}_{s_{k'},R_a}$ can only contribute to $\mathbf{u}$ if, for all $k \in [n]\backslash\{k'\}$ the vector $\mathbf{v}_{s_k,R_a}$ contributes in equal measure, otherwise there would be box-symbol entries of $\mathbf{u}$ with different values depending on the symbol. This holds analogously for any column-bundle type C_0 vector $\mathbf{v}_{s_{k'},C_b}$ that contributes to $\mathbf{u}$. More succinctly, $\mathbf{u}$ must be a linear combination of $\sum_{k=1}^{n} \mathbf{v}_{s_k,R_a}$ for row-bundles $a \in [h]$ and $\sum_{k=1}^{n} \mathbf{v}_{s_k,C_b}$ for column-bundles $b \in [w]$.

As there are only $h + w$ such vectors, the number of independent vectors that can be formed as linear combinations of type C_0 vectors, while also lying in $\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}}$, is limited to at most $h + w$. This completes the proof. □

Applying the subspace intersection lemma and the dimensions established in Lemmas 4.1.1 and 4.1.2, the following proposition follows immediately.

Proposition 4.1.3. *We have* $\dim(\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}} + \mathcal{E}_0^{\mathcal{C}}) \geq 3n + (h+w)(n-1)$ *and hence the vectors of types* A_0, B_0, C_0 *span* $\ker(M)$.

Finally, we present the construction of the basis of $\ker(M)$.

Proposition 4.1.4. *A basis for* $\ker(M)$ *can be built from all of the vectors of type* A_0, B_0*, together with those vectors of type* C_0 *which avoid one particular symbol, say* s_n.

Proof. As we already have established that $\dim(\ker(M)) = 3n + (h + w)(n - 1)$ and that the vectors of types A_0, B_0, C_0 span $\ker(M)$, it is enough to show that the set described is linearly independent.

Suppose not. Then, there must exist some non-trivial linear combination of the vectors from this set which equals zero. Let $\mathbf{u}$ be the portion of this linear combination coming from vectors of types A_0 and B_0. As in the proof of Lemma 4.1.2, $\mathbf{u} \in (\mathcal{E}_0^{\mathcal{A}} + \mathcal{E}_0^{\mathcal{B}})$, implies that $\mathbf{u}$ is invariant under symbol permutation. In particular, any box-symbol entry of $\mathbf{u}$ depends only on the box.

However, $-\mathbf{u}$ is a linear combination of type C_0 vectors avoiding s_n and we know that the entry $b_\ell s_n$ of $\mathbf{u}$ must be 0 because no type C_0 vector using s_n was used in the linear combination for $-\mathbf{u}$. This implies that all box-symbol edges of $\mathbf{u}$ must also

be 0, therefore implying that $\mathbf{u}$ is a vanishing linear combination of type C_0 vectors. Since the type C_0 vectors are linearly independent, this implies that $\mathbf{u}$ is simply the zero vector. As the set of all type A_0 and B_0 vectors is also linearly independent, it follows that all coefficients in our original linear combination vanish. This gives the necessary contradiction, as it implies that the only linear combination of vectors from this set equally zero is trivial. □

4.2 Eigenspace $\mathcal{E}_1$ for eigenvalue $\theta_1 = n$

We now present a description of the eigenvectors in eigenspace $\mathcal{E}_1$ for eigenvalue $\theta_1 = n$.

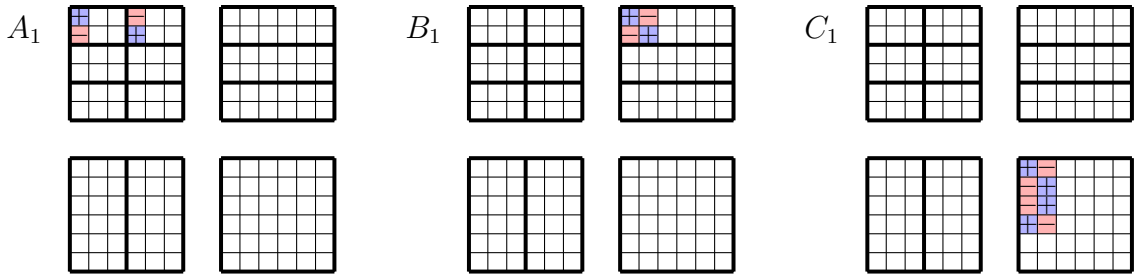


Figure 4.5: Eigenvectors for $\mathcal{E}_1$.

4.2.1 Type A_1: The row-column $\mathcal{E}_1$ vectors.

We construct the type A_1 vectors by selecting two distinct rows, r_{i_0} and r_{i_1}, and two distinct columns, c_{j_0} and c_{j_1}, ensuring that either the rows are within the same row-bundle or the columns are within the same column-bundle. We adopt the simplified notation $\mathbf{v}$ for a type A_1 vector with $r_{i_0}, r_{i_1}, c_{j_0}$ and c_{j_1} fixed to enhance clarity while minimizing the complexity of indices. The components of a type A_1 vector, based on the given configuration, are determined as follows:

$$\mathbf{v}(r_i c_j) = \begin{cases} (-1)^{\alpha+\beta} & \text{if } (i,j) = (i_\alpha, j_\beta) \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and, in all cases, $\mathbf{v}(c_j s_k) = \mathbf{v}(r_i s_k) = \mathbf{v}(b_\ell s_k) = 0$.

For an edge $f = r_{i_\alpha} c_{j_\beta}$, we have n tiles extending f corresponding to a choice of symbol s_k. Each such tile contains at most one non-vanishing edge, namely that corresponding to f. So $Mv(f) = nv(f)$. For f of type row-symbol, column-symbol, or box-symbol, we have $Mv(f) = 0$, either from cancellation or disjoint supports. Importantly, having either $r_{i_0} \sim r_{i_1}$ or $c_{j_0} \sim c_{j_1}$ ensures cancellation within each box.

Proposition 4.2.1. *There exists a linearly independent set $\mathcal{A}_1$ of $(n-1)^2-(h-1)(w-1)$ eigenvectors generated by type A_1 vectors.*

Proof. We first build a set $\mathcal{A}_1$ of linearly independent vectors and show that $|\mathcal{A}_1| = (n-1)^2 - (h-1)(w-1)$. It is important to note that since we've established the eigenvector property of all type A_1 vectors, any linear combination of these vectors must also belong to $\mathcal{E}_1$.

Fix the top left-most cell from each box as the *representative cell* from that box. Then fix the top left-most box b_1, noticing that $(1,1)$ is the representative cell in b_1. Next, choose any cell (i,j) where $i \neq 1$ and $j \neq 1$, which is *not* a representative cell of any box. To count the number of available cells we could choose, we first consider the total number of potential cells (i,j), where both i and j range from 2 to n, excluding the first row and column to avoid choosing representative cells. This initial count gives us a total of $(n-1)^2$ cells. However, within this total, we must account for the cells that serve as representative cells for other boxes. Since each of the remaining boxes—aside from b_1—has a unique representative cell, and there are $(h-1)(w-1)$ such boxes, we subtract this number from our initial count. Therefore, after excluding the representative cells of all boxes except for b_1, we are left with $(n-1)^2-(h-1)(w-1)$ viable cells (i,j) that could be chosen. Since entry (i,j) must be in some box b_ℓ, we then let (a,b) be the representative cell in box b_ℓ.

We are now able to define the following vectors w_{ij}, x_{ij}, $z_{ij} \in \mathcal{E}_1$ which depend on cell (i,j) as follows:

- Type A_1 vector w_{ij} is such that $r_{i_0} = r_i$, $c_{j_0} = c_j$, $r_{i_1} = r_a$, and $c_{j_1} = c_b$.
- Type A_1 vector x_{ij} is such that $r_{i_0} = r_i$, $c_{j_0} = c_b$, $r_{i_1} = r_a$, and $c_{j_1} = c_1$.
- Type A_1 vector z_{ij} is such that $r_{i_0} = r_a$, $c_{j_0} = c_j$, $r_{i_1} = r_1$, and $c_{j_1} = c_b$.

Let $v_{ij} = w_{ij} + x_{ij} + z_{ij}$. Of course, if $i = a$, then $v_{ij} = z_{ij}$. Similarly, if $j = b$, then $v_{ij} = x_{ij}$. If $(a,b) = (1,1)$, then $v_{ij} = w_{ij}$ because $x_{ij} = z_{ij} = 0$. In the case that $i \neq a, j \neq b$ and $(a,b) \neq (1,1)$, we note that v_{ij} is not of type A_1, but v_{ij} is constructed from a linear combination of vectors from type A_1.

Let $\mathcal{A}_1$ be the set of all v_{ij} vectors determined by cells (i,j) where $i \neq 1$ and $j \neq 1$ and such that (i,j) is *not* a representative cell of any box. Note that by this construction $|\mathcal{A}_1| = (n-1)^2 - (h-1)(w-1)$ and $\mathcal{A}_1$ is linearly independent because each such vector has the unique support $r_i c_j$. This completes the proof. □

4.2.2 Type B_1: The bundle-symbol $\mathcal{E}_1$ vectors

We construct the row-bundle type B_1 vectors by first choosing two distinct rows from a row-bundle and two distinct symbols. Let the rows be denoted as r_{i_0} and r_{i_1}, while

the symbols are represented by s_{k_0} and s_{k_1}. The components of a type B_1 vector, based on the given configuration, are determined as follows:

$$\mathbf{v}(r_i s_k) = \begin{cases} (-1)^{\alpha+\beta} & \text{if } (i,k) = (i_\alpha, k_\beta) \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and, in all cases, $\mathbf{v}(c_j s_k) = \mathbf{v}(r_i c_j) = \mathbf{v}(b_\ell s_k) = 0$.

Constructing the column-bundle type B_1 vectors is similar, except we then choose two unique columns from a column-bundle, rather than rows from a row-bundle. The verification that $M\mathbf{v} = n\mathbf{v}$ mirrors the justification presented for type A_1 vectors.

Importantly, one can observe that the type B_1 vectors are linearly independent of the type A_1 vectors because row-symbol and column-symbol entries are not supported in the type A_1 vectors.

Proposition 4.2.2. *There exists a set $\mathcal{B}_1$ of $(n-1)(h(w-1)+w(h-1))$ linearly independent type B_1 vectors.*

Proof. To form $\mathcal{B}_1$, we start by fixing the first row (or first column) for each row-bundle (or column-bundle) as a representative row (column) and a specific symbol $s_n = s_{k_0}$. We then select any of the $(n-1)$ other symbols as s_{k_1}. Subsequently, we opt for a row-bundle (from the available w choices) or a column-bundle (from h choices). Without loss of generality, if we choose a row-bundle, we set the representative row as r_{i_0} and pick another row as r_{i_2}. All vectors of this kind are added to $\mathcal{B}_1$. The process is similarly repeated for columns. The cardinality of $\mathcal{B}_1$ is $(n-1)(h(w-1)+w(h-1))$ and the vectors in $\mathcal{B}_1$ are linearly independent, as each has a unique row-symbol or column-symbol support. □

4.2.3 Type C_1: The box-symbol $\mathcal{E}_1$ Vectors

We construct the type C_1 vectors by first choosing four distinct boxes, $b_{\ell_{00}}, b_{\ell_{10}}, b_{\ell_{01}}$, and $b_{\ell_{11}}$, that form a rectangle and two distinct symbols s_{k_0} and s_{k_1}. Note that choosing two distinct row-bundles, denoted X_0, X_1, and two distinct column-bundles, denoted Y_0, Y_1, is a way to choose four such boxes since their intersection points necessarily define four boxes in a rectangle. Given this configuration, the entries of a type C_1 vector are determined as:

$$\mathbf{v}(b_\ell s_k) = \begin{cases} (-1)^{\alpha+\beta+\gamma} & \text{if } (k,\ell) = (k_\gamma, \ell_{\alpha\beta}) \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise} \end{cases}$$

and, in all cases, $\mathbf{v}(c_j s_k) = \mathbf{v}(r_i s_k) = \mathbf{v}(r_i c_j) = 0$.

Cancellation occurs if we sum over rows, columns, or symbols. For a symbol-box edge $f = s_k b_\ell$, the n tiles extending f correspond to a choice of entry in box ℓ. This

picks up the value of $\mathbf{v}(f)$ with a multiplicity of n. So $M\mathbf{v} = n\mathbf{v}$.

Proposition 4.2.3. *There exists a set $\mathcal{C}_1$ of $(n-1)(h-1)(w-1)$ linearly independent type C_1 vectors.*

Proof. To form $\mathcal{C}_1$, we start by fixing $s_n = s_{k_0}$, while designating any of the other $(n-1)$ symbols as s_{k_1}. We then fix the first row-bundle as X_0 and the first column-bundle as Y_0, while designating any of the $w-1$ other row-bundles as X_1 and any of the $h-1$ other column-bundles as Y_1. There are $(n-1)(h-1)(w-1)$ ways to choose some combination of s_{k_1}, X_1, and Y_1. These vectors are easily verified as a linearly independent set as each vector of the set contains a unique symbol-box support. □

4.2.4 A lower bound on $\dim(\mathcal{E}_1)$

Proposition 4.2.4. *The set $\mathcal{I}_1 = \mathcal{A}_1 \cup \mathcal{B}_1 \cup \mathcal{C}_1$ is linearly independent, and therefore $\dim(\mathcal{E}_1) \geq 4n^2 - (2n-3)(h+w) - 5n - 1$.*

Proof. The supports of vectors of type A_1, type B_1, and type C_1, as described above, all correspond to different edge types and so the linearly independent sets $\mathcal{A}_1, \mathcal{B}_1$, and $\mathcal{C}_1$ define pairwise orthogonal subspaces of eigenspace $\mathcal{E}_1$. As such, their union is linearly independent. □

Recall that the spectral theorem for symmetric matrices tells us that we have $\sum_{j=0}^{4} \dim(\mathcal{E}_j) = 4n^2$. Given that the set $\mathcal{I}_1$ is linearly independent and has a cardinality of $4n^2 - (2n-3)(h+w) - 5n - 1$, if we can identify linearly independent sets within the other eigenspaces such that their combined cardinalities equal $4n^2$, then we can conclude that $\mathcal{I}_1$ forms a basis for the eigenspace $\mathcal{E}_1$.

4.3 Eigenspace $\mathcal{E}_2$ for eigenvalue $\theta_2 = 2n$

We now present a description of the eigenvectors in eigenspace $\mathcal{E}_2$ for eigenvalue $\theta_2 = 2n$.

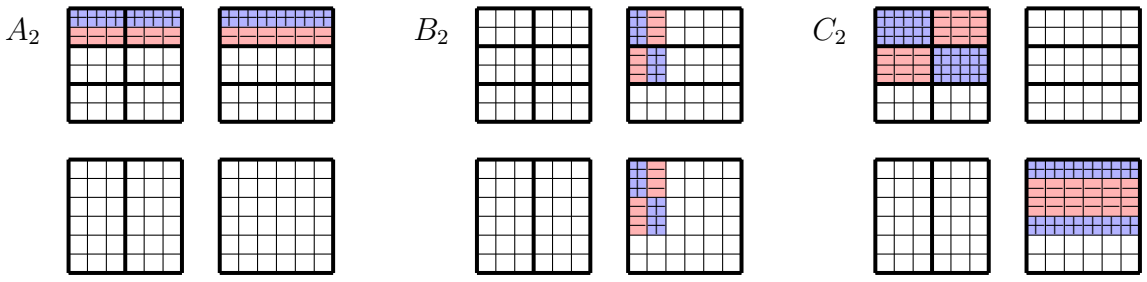


Figure 4.6: Eigenvectors in $\mathcal{E}_2$.

4.3.1 Type A_2: The bundle $\mathcal{E}_2$ vectors.

We construct the row-bundle type A_2 vectors by first choosing two unique rows from a row-bundle. Let the rows be denoted as r_{i_0} and r_{i_1}. The components of a type A_2 vector, based on the given configuration, are determined as follows:

$$\mathbf{v}(r_i s_k) = \mathbf{v}(r_i c_j) = \begin{cases} (-1)^\alpha & \text{if } i = i_\alpha \text{ for } \alpha \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and, in all cases, $\mathbf{v}(c_j s_k) = \mathbf{v}(b_\ell s_k) = 0$.

An edge $f = r_i c_j$ or $r_i s_k$ has exactly n extensions to a tile, each of which has two edges of a common sign. So $M\mathbf{v}(f) = 2n\mathbf{v}(f)$ in those cases. It is easy to see that $M\mathbf{v}(f) = 0$ on all other edges due to cancellation on rows. A similar variety of type A_2 vectors exists with rows and columns swapped, and the verification that they belong in $\mathcal{E}_2$ is analogous.

Proposition 4.3.1. *There exists a set $\mathcal{A}_2$ of $h(w-1)+w(h-1) = 2n-h-w$ linearly independent type A_2 vectors.*

Proof. Fix the first row of each row-bundle as the representative row and the first column from each column-bundle as the representative column. For the row-bundles, there are w ways to choose a row-bundle. Let r_{i_0} be the representative row of a chosen bundle, and r_{i_1} be any of the remaining $h-1$ rows from that row-bundle. By adding each such vector to $\mathcal{A}_2$, we obtain $w(h-1)$ linearly independent vectors because each such vector has a unique row-symbol support. Notably, this set can generate all other row-bundle type A_2 vectors by taking the difference between two such vectors.

For the column-bundles, we mirror the process with the only difference being that there are h ways to choose a column-bundle and $w-1$ ways to choose the other column from that column-bundle. This set of $h(w-1)$ vectors is also linearly independent, with each such vector having a unique column-symbol support. It can generate all other column-bundle type A_2 vectors by taking the difference between two such vectors.

These vectors can be included in $\mathcal{A}_2$ to form a linearly independent set with cardinality $h(w-1)+w(h-1) = 2n-h-w$. □

4.3.2 Type B_2: The bundle-symbol $\mathcal{E}_2$ vectors

We construct the row-bundle type B_2 vectors by first choosing two distinct row-bundles. Let the row-bundles be denoted as X_0 and X_1 and choose a row from each, r_{i_0} from X_0 and r_{i_1} from X_1. Finally, choose two distinct symbols s_{k_0} and s_{k_1}. The components of a type B_2 vector, based on the given configuration, are determined as follows:

$$\mathbf{v}(r_i s_k) = \begin{cases} (-1)^{\alpha+\beta} & \text{if } r_i \sim r_{i_\alpha} \text{ and } s_k = s_{k_\beta} \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\mathbf{v}(b_\ell s_k) = \begin{cases} (-1)^{\alpha+\beta} & \text{if } b_\ell \smile X_\alpha \text{ and } s_k = s_{k_\beta} \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise.} \end{cases}$$

In all cases, $\mathbf{v}(c_j s_k) = \mathbf{v}(r_i c_j) = 0$.

A similar variety exists with rows and columns swapped. To verify that type B_2 vectors belong in $\mathcal{E}_2$, we again check their interactions with an edge, f. If f is a row-column edge, the cancellation on symbols gives $M\mathbf{v}(f) = 0$. Likewise, if f is a column-symbol edge, the cancellation on rows gives $M\mathbf{v}(f) = 0$. For an edge f which is either of the other two edge types, there are n extensions to a tile, and the nonzero edges agree in sign so $M\mathbf{v}(f) = 2n\mathbf{v}(f)$.

Proposition 4.3.2. *There exists a set $\mathcal{B}_2$ of $(n-1)(h+w-2)$ linearly independent type B_2 vectors.*

Proof. Begin by fixing a symbol $s_n = s_{k_0}$ and let $s_k = s_{k_1}$ for any $k \in [n-1]$. Designate $r_n = r_{i_0}$ and set $r_i = r_{i_1}$ for any $i \in [n-1]$ that satisfies the congruence condition $i \equiv 0 \bmod h$ (we do this because, for any pair of symbols, if the two rows chosen are from the same bundle then the resulting vectors will be identical). Adding each such vector to $\mathcal{B}_2$ yields $(w-1)$ type B_2 vectors which can be verified as linearly independent by the unique row-symbol support in each vector. Furthermore, this set generates all other row-bundle type B_2 vectors by taking the difference between two such vectors.

Mirroring this approach with the roles of rows and columns swapped reveals the existence of a set containing $(n-1)(h-1)$ column-bundle type B_2 vectors which can be verified as linearly independent because they have unique column-symbol supports. This set generates all other column-bundle type B_2 vectors by taking the difference between two such vectors.

These vectors can be included in $\mathcal{B}_2$ to form a linearly independent set with cardinality $(n-1)((h-1)+(w-1)) = (n-1)(h+w-2)$. □

4.3.3 Type C_2: The box $\mathcal{E}_2$ vectors

We construct the type C_2 vectors by first choosing four distinct boxes, $b_{\ell_{00}}, b_{\ell_{10}}, b_{\ell_{01}}$, and $b_{\ell_{11}}$. Given this configuration, the entries of a type C_1 vector are determined as:

$$\mathbf{v}(b_\ell s_k) = \begin{cases} (-1)^{\alpha+\beta} & \text{if } \ell = \ell_{\alpha\beta} \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and

$$\mathbf{v}(r_i c_j) = \begin{cases} (-1)^{\alpha+\beta} & \text{if box}(i,j) = \ell_{\alpha\beta} \text{ for } \alpha, \beta \in \{0,1\}, \\ 0 & \text{otherwise.} \end{cases}$$

In all cases, $\mathbf{v}(c_j s_k) = \mathbf{v}(r_i s_k) = 0$.

To verify that type C_2 vectors belong in $\mathcal{E}_2$, we again check their interactions with an edge, f. If f is a row-symbol or column-symbol edge, the extension to a tile leads to cancellation. However, if f is a row-column edge then there are n extensions to a tile by selecting a symbol, and each has two matching edges from the entry and box. So $M\mathbf{v}(f) = 2n\mathbf{v}(f)$. Similarly, for a box-symbol edge f, we have $M\mathbf{v}(f) = 2n\mathbf{v}(f)$.

Proposition 4.3.3. *There exists a set $\mathcal{C}_2$ of $(h-1)(w-1) = n-h-w+1$ linearly independent type C_2 vectors.*

Proof. Recall that the choice of four boxes that form a rectangle is equivalent to choosing two distinct row-bundles, denoted X_0, X_1, and two distinct column-bundles, Y_0, Y_1. Thus, we start by fixing the first row-bundle as X_0 while allowing the choice of any of the $w-1$ other row-bundles as X_1 and fixing the first column-bundle as Y_0 while allowing the choice of any of the $h-1$ other column-bundles as Y_1. Together, there are $(h-1)(w-1)$ ways to choose X_1 and Y_1. These vectors are easily verified as linearly independent since each contains a unique box, as dictated by the intersection of X_1 and Y_1, and therefore contains unique symbol-box edges. Thus, we can add all such vectors to $\mathcal{C}_2$, forming a linearly independent set of size $(h-1)(w-1)$. □

4.3.4 A lower bound on $\dim(\mathcal{E}_2)$

Proposition 4.3.4. *The set $\mathcal{I}_2 = \mathcal{A}_2 \cup \mathcal{B}_2 \cup \mathcal{C}_2$ is linearly independent, and therefore $\dim(\mathcal{E}_2) \geq (n-3)(h+w-1) + 2n$.*

Proof. Unlike in $\mathcal{E}_1$, we see that the three types of vectors from $\mathcal{E}_2$ do not have entirely disjoint supports. However, for any type A_2 vector $\mathbf{a}$, type B_2 vector $\mathbf{b}$, and type C_2 vector $\mathbf{c}$, we have

$$\mathbf{a} \cdot \mathbf{b} = \mathbf{a} \cdot \mathbf{c} = \mathbf{b} \cdot \mathbf{c} = 0.$$

This can be easily verified by examining the definitions of each vector type. Notice that any pair of vectors from any two of the three types will have the property that the number of mutual supports which share the same sign will equal the number of mutual supports which have opposite signs, therefore leading to cancellation.

Since $\mathcal{A}_2$, $\mathcal{B}_2$, and $\mathcal{C}_2$ have all been shown to be, themselves, linearly independent sets, this mutual orthogonality indicates that the union of $\mathcal{A}_2$, $\mathcal{B}_2$, and $\mathcal{C}_2$ is linearly independent. Thus, $\dim(\mathcal{E}_2) \geq |\mathcal{A}_2| + |\mathcal{B}_2| + |\mathcal{C}_2| = (n-3)(h+w-1) + 2n$. This completes the proof. □

Again, recall that according to the spectral theorem for symmetric matrices, we have:

$$\sum_{j=0}^{4} \dim(\mathcal{E}_j) = 4n^2. \tag{4.1}$$

Given that the set $\mathcal{I}_2$ is linearly independent and has a cardinality of $(n-3)(h+w-1)+2n$, if we can identify other linearly independent sets within $\mathcal{E}_0, \mathcal{E}_1, \mathcal{E}_3$, and $\mathcal{E}_4$ such that their combined cardinalities equal $4n^2$, then we can conclude that the set $\mathcal{I}_2$ spans all of $\mathcal{E}_2$ and therefore forms a basis for $\mathcal{E}_2$.

4.4 Eigenspace $\mathcal{E}_3$ for eigenvalue $\theta_3 = 3n$

We now present a description of the eigenvectors in eigenspace $\mathcal{E}_3$ for eigenvalue $\theta_3 = 3n$.

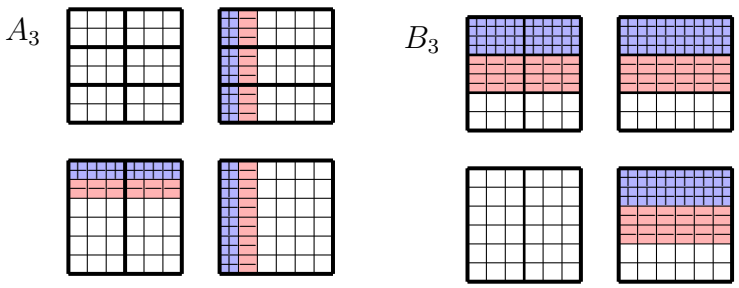


Figure 4.7: Eigenvectors in $\mathcal{E}_3$.

4.4.1 Type A_3: The symbol $\mathcal{E}_3$ vectors

We construct the type A_3 vectors by first choosing two distinct symbols s_{k_0} and s_{k_1}. The components of a type A_3 vector, based on the given configuration, are determined as follows:

$$\mathbf{v}(r_i s_k) = \mathbf{v}(c_j s_k) = \mathbf{v}(b_\ell s_k) = \begin{cases} (-1)^\alpha & \text{if } s_k = s_{k_\alpha} \text{ for } \alpha \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and, in all cases, $\mathbf{v}(r_i c_j) = 0$.

To verify that type A_3 vectors belong in $\mathcal{E}_3$, we again check their interactions with an edge, f. If f is any edge involving a symbol s_{k_γ}, the n tiles extending f each have (if any) three nonzero edges of matching sign. So $M\mathbf{v}(f) = 3n\mathbf{v}(f)$. In other cases, it is easy to see that $M\mathbf{v}(f) = 0$ by cancellation.

Proposition 4.4.1. *There exists a set $\mathcal{A}_3$ of $(n-1)$ linearly independent type A_3 vectors.*

Proof. We begin by fixing $s_n = s_{k_0}$, and letting $s_k = s_{k_1}$ for any $k \in [n-1]$ to form $(n-1)$ type A_3 vectors. Each such vector can be included in $\mathcal{A}_3$, which will remain linearly independent because each such vector is supported on unique row-symbol and column-symbol entries. Note that by iteratively forming differences between the vectors in $\mathcal{A}_3$, we can generate the rest of the type A_3 vectors. □

4.4.2 Type B_3: The bundle $\mathcal{E}_3$ vectors

We construct the row-bundle type B_3 vectors by first choosing two distinct row-bundles. We use X_0 to denote the set of rows in the first row-bundle and the boxes that intersect those rows. We use X_1 to denote the set of rows in the second row-bundle and the boxes that intersect those rows. The components of a type B_3 vector, based on the given configuration, are determined as follows:

$$\mathbf{v}(r_i s_k) = \mathbf{v}(r_i c_j) = \mathbf{v}(b_\ell s_k) = \begin{cases} (-1)^\alpha & \text{if } r_i \in X_\alpha \text{ for } \alpha \in \{0,1\}, \\ (-1)^\alpha & \text{if } b_\ell \in X_\alpha \text{ for } \alpha \in \{0,1\}, \\ 0 & \text{otherwise,} \end{cases}$$

and, in all cases, $\mathbf{v}(c_j s_k) = 0$.

A similar variety exists using column-bundles, rather than row-bundles. The verification here is analogous to that of type A_3, except that the bundles take the role of symbols.

Proposition 4.4.2. *There exists a set $\mathcal{B}_3$ of $(h-1)+(w-1)$ linearly independent type B_3 vectors.*

Proof. To construct the row-bundle type B_3 vectors which can be included in $\mathcal{B}_3$ fix the first row-bundle as the row-bundle containing row r_1 and let the other row-bundle be any of the remaining $(w-1)$ row-bundles. Each vector is supported on unique row-symbol entries, so they are linearly independent. This set can generate all other row-bundle type B_3 vectors by taking the difference between two such vectors.

The $(h-1)$ column-bundle type B_3 vectors which can be included in $\mathcal{B}_3$ are formed analogously and each resulting vector is supported on unique column-symbol entries, so they are linearly independent from one another and from the row-bundle type B_3 vectors. This set can generate all other column-bundle type B_3 vectors by taking the difference between two such vectors.

Thus, we let $\mathcal{B}_3$ be the linearly independent set of all such vectors, and observe that $\mathcal{B}_3$ has cardinality $(h-1)+(w-1)$. This completes the proof. □

4.4.3 A lower bound on $\dim(\mathcal{E}_3)$

Proposition 4.4.3. *The set $\mathcal{I}_3 = \mathcal{A}_3 \cup \mathcal{B}_3$ is linearly independent, and therefore $\dim(\mathcal{E}_3) \geq n + h + w - 3$.*

Proof. As with the previous eigenspaces, for any type A_3 vector $\mathbf{a}$ and type B_3 vector $\mathbf{b}$,

$$\mathbf{a} \cdot \mathbf{b} = 0.$$

To see why this is, consider $\mathbf{a}$ generated by fixed symbols s_{k_0} and s_{k_1} and $\mathbf{b}$ as a row-bundle type B_3 vector generated by fixed row-bundles with corresponding sets X_0 and X_1. The entries which intersect and are in the support of both vectors are $r_i s_k$ with $r_i \in X_0 \cup X_1$ and $s_k \in \{s_{k_0}, s_{k_1}\}$ and $b_\ell s_k$ with $b_\ell \smile r_i$ for some $r_i \in X_0 \cup X_1$ and $s_k \in \{s_{k_0}, s_{k_1}\}$. For $\mathbf{a}$ the $r_i s_k$ entry will be constant for any fixed s_k whereas for $\mathbf{b}$ they will be positive for $r_i \in X_0$ and negative for $r_i \in X_1$. As there are an equal number of rows in X_0 as there are in X_1, these values cancel to give 0. Similarly, for $\mathbf{a}$ the $b_\ell s_k$ entries will be constant for any fixed s_k whereas for $\mathbf{b}$ they will be positive for $b_\ell \smile X_0$ and negative for $b_\ell \smile X_1$. As there are an equal number of boxes which intersect X_0 as there are for X_1, these values again cancel to give 0. A similar proof can be used to show that this holds when $\mathbf{b}$ is a column-bundle type B_3 vector.

Since $\mathcal{A}_3$ and $\mathcal{B}_3$ are themselves linearly independent sets, this mutual orthogonality indicates that the union of $\mathcal{A}_3$ and $\mathcal{B}_3$ is linearly independent. Thus, $\dim(\mathcal{E}_3) \geq |\mathcal{A}_3| + |\mathcal{B}_3| = |\mathcal{I}_3| = n + h + w - 3$. This completes the proof. □

4.5 Confirming bases for eigenspaces $\mathcal{E}_1$, $\mathcal{E}_2$, and $\mathcal{E}_3$

Finally, we confirm that the independent sets of eigenvectors constructed in Sections 4.2, 4.3, and 4.4 form bases of their respective eigenspaces.

Recall that the dimension of $\mathcal{E}_4$ is 1, as it simply consists of the all-ones vector indexed by $E(G_{hw})$. Therefore, together Propositions 4.2.4, 4.3.4, and 4.4.3 give:

$$\sum_{j=0}^{4} \dim(\mathcal{E}_j) = 4n^2 \geq \dim(\ker(M)) + |\mathcal{I}_1| + |\mathcal{I}_2| + |\mathcal{I}_3| + 1,$$

and the sum of these lower bounds on the dimensions of $\mathcal{E}_j$ gives:

$$\begin{aligned}
&(4n - 1 + (n-1)(h+w-1)) + (4n^2 - (2n-3)(h+w) - 5n - 1) \\
&+ ((n-3)(h+w-1) + 2n) + (n+h+w-3) + 1 \\
&= 4n^2,
\end{aligned}$$

so we deduce that the lower bounds in Propositions 4.2.4, 4.3.4, and 4.4.3 must, in fact,

be met with equality. We can therefore conclude that $\mathcal{I}_1$, $\mathcal{I}_2$, and $\mathcal{I}_3$ serve as bases for the eigenspaces $\mathcal{E}_1$, $\mathcal{E}_2$, and $\mathcal{E}_3$, respectively.

4.6 Eigenprojectors of M

We have constructed a basis of $4n^2$ eigenvectors for M. Since M is a symmetric matrix, we know that these eigenvectors necessarily span $\mathbb{R}^{4n^2}$. That is, any vector in $\mathbb{R}^{4n^2}$ can be uniquely represented as a linear combination of eigenvectors of M. Additionally, for each of M's eigenspaces, $\mathcal{E}_j$, there is an eigenprojector, E_j, that maps vectors onto that eigenspace. These eigenprojectors are orthogonal idempotents, and when you sum the projections of a vector onto each eigenspace, you get the original vector. This relationship is captured by the equation:

$$\sum_{j=0}^{4} E_j = I.$$

The eigenprojectors can be computed as $E_i = V_i(V_i^\top V_i)^{-1}V_i^\top$, where V_i is a matrix whose columns are a basis of eigenvectors for $\mathcal{E}_i$ such as the one we constructed earlier. As a special case, since V_4 is the $(4n^2 \times 1)$ all-ones vector, we have $E_4 = \frac{1}{4n^2}J$. Furthermore, for each $j \in \{0, 1, 2, 3, 4\}$, E_j belongs to $\mathfrak{A}$. This is a standard property of coherent configurations, as discussed in [12].

The structure of entries for each of the other eigenprojectors is shown in Figure 4.8, mostly for the sake of interest. Intensity of blue/red correspond respectively to extreme positive/negative entries, while shades of green/yellow correspond to positive/negative entries which are smaller in magnitude.

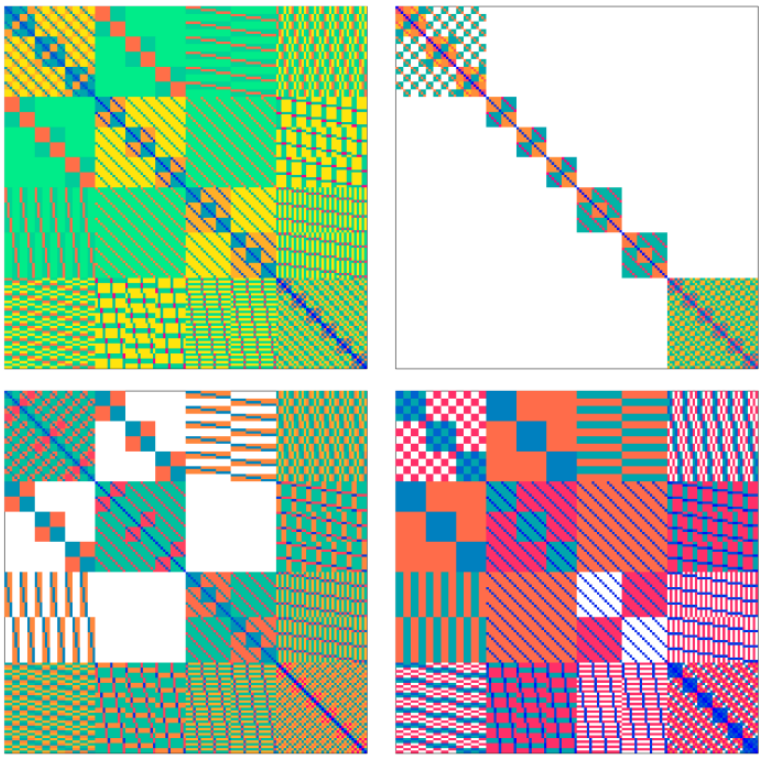


Figure 4.8: Structure of entries of E_0, E_1 (top), E_2, E_3 (bottom) for $(h, w) = (2, 3)$.

4.6.1 Using eigenprojectors to construct a generalized inverse

We now detail how the eigenprojectors of M can be used to compute a generalized inverse M^+ satisfying $MM^+M = M$. We explain this computation in the rest of this section. The spectral decomposition of M is given by:

$$M = \sum_{j=0}^{4} (jn)E_j = 0E_0 + nE_1 + 2nE_2 + 3nE_3 + 4nE_4.$$

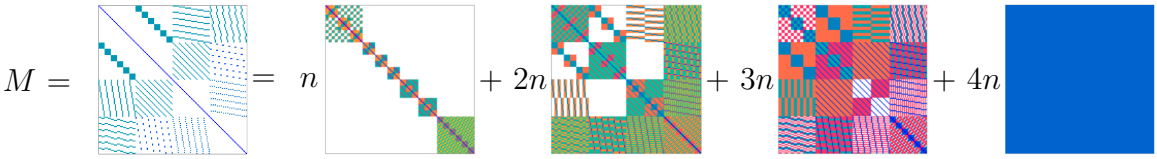


Figure 4.9: A visualization of the equation $M = \sum_{j=0}^{4}(jn)E_j$ for $(h, w) = (2, 3)$.

In what follows, E_0 will also be denoted K, since it projects onto the kernel of M. Although M itself is not invertible, if we take $\eta \neq 0$, say $\eta = n/x$, the additive shift $M + \eta K$ is invertible. This is because the eigenprojectors K, E_1, E_2, E_3, and E_4 are pairwise orthogonal, meaning that for any E_j with $j \in \{1, 2, 3, 4\}$ we have

$$(M + \eta K)E_j = ME_j + \eta KE_j = ME_j = (jn)E_j.$$

However, as K is idempotent, we get that

$$(M + \eta K)K = MK + \eta K^2 = \eta K.$$

Thus, we have replaced the eigenvalue 0 with the eigenvalue η. This means that $(M + \eta K) = \eta K + \sum_{j=1}^{4}(jn)E_j$ has rank $4n^2$ and is therefore invertible.

We can invert the additive shift $M + \eta K$ as

$$(M + \eta K)^{-1} = \frac{1}{n}\left(xK + \sum_{j=1}^{4} \frac{1}{j} E_j\right). \tag{4.2}$$

To confirm this formula works, we apply the fact that these eigenprojectors are orthogonal idempotents and that the formula $\sum_{j=0}^{4} E_j = I$ holds as we multiply $(M + \eta K)(M + \eta K)^{-1}$ to verify that it does, indeed, give the identity.

$$
\begin{aligned}
(M+\eta K)(M+\eta K)^{-1} &= \left(\eta K + \sum_{j=1}^{4}(jn)E_j\right)\frac{1}{n}\left(xK + \sum_{j=1}^{4}\frac{1}{j}E_j\right) \\
&= \left(\frac{n}{x}K + \sum_{j=1}^{4}(jn)E_j\right)\frac{1}{n}\left(xK + \sum_{j=1}^{4}\frac{1}{j}E_j\right) \\
&= \left(\frac{1}{x}K + \sum_{j=1}^{4}(j)E_j\right)\left(xK + \sum_{j=1}^{4}\frac{1}{j}E_j\right) \\
&= \frac{1}{x}K\left(xK + \sum_{j=1}^{4}\frac{1}{j}E_j\right) + \left(\sum_{j=1}^{4}(j)E_j\right)\left(xK + \sum_{j=1}^{4}\frac{1}{j}E_j\right) \\
&= (K^2 + 0) + (0 + \sum_{j=1}^{4}E_j^2) \\
&= K + \sum_{j=1}^{4}E_j \\
&= I.
\end{aligned}
$$

Then, we closely follow the methodology of Bowditch and Dukes, recalling [5, Lemma 2.6]. The idea lets us solve an under-determined system $A\mathbf{x} = \mathbf{b}$ by inverting an additive shift of A.

Lemma 4.6.1 (see [5])**.** *Let A and B be symmetric $N \times N$ real matrices with $AB = O$, $A + B$ invertible, and $B\mathbf{b} = \mathbf{0}$. Then $A(A+B)^{-1}\mathbf{b} = \mathbf{b}$.*

Thus applying this lemma with M as A, ηK as B, and $\mathbf{b}$ as the all-ones vector provides an opportunity to solve our linear system for fractional Sudoku by making use of a generalized inverse of the form shown in (4.2). We found that $x = 3/2$, or $\eta = 2n/3$, is an optimal choice so we let our generalized inverse M^+ be $(M + \frac{2n}{3}K)^{-1}$. A discussion of how this value was found and the functions it optimizes is given in Subsection 4.6.3.

4.6.2 Expressing eigenprojectors and M^+ in $\mathfrak{A}$

Using further interpolation and computer-assisted algebra, we found coefficients to express the eigenprojectors E_j for $j \in \{0, 1, 2, 3, 4\}$ and M^+ in the basis $\{A_i : i = 1, \ldots, 69\}$ for the adjacency algebra $\mathfrak{A}$. The coefficients of M^+ are expressed in Table 4.1. For increased readability, we cleared a denominator of $9n^3$ and then applied an additive shift of $5/16$. To aid in understanding how these computations were performed, we provide some background on the methodology for finding and verifying the

coefficients for each eigenprojector. These coefficients, along with the formula (4.2), enabled us to express M^+, as seen in Table 4.1.

relations	coefficients	relations	coefficients
1	$9n^2+h+w$	32	$9n^2+n+h$
2, 4, 5	$h+w$	34	$n+h$
3, 6, 17, 19, 39, 43, 47, 51	w	38, 42	$-9n/2+h+w$
7, 8, 33, 35, 40, 44, 55, 59	h	46, 50	$-9nw/2+n+w$
10, 13	$-9n/2+w+1$	54, 58	$-9nh/2+n+h$
11, 14	$w+1$	62	$9n^2+n+h+w-1$
12, 15, 24, 27, 29, 31	1	63	$h+w-1$
16	$9n^2+n+w$	64	$n+w-1$
18	$n+w$	65	$w-1$
20, 36, 48, 52, 56, 60	n	66	$n+h-1$
22, 25	$-9n/2+h+1$	67	$h-1$
23, 26	$h+1$	68	$n-1$
28, 30	$-7n/2+1$	69	-1

Table 4.1: Coefficients of A_i for $i \in [69]$ in the expression $9n^3(M+\eta K)^{-1}+\frac{5}{16}J$.

To find the symbolic coefficients for each eigenprojector, we conducted a computational search for specific coefficients for $2 \leq h, w \leq 4$, sufficient due to the degree bound in Proposition 3.2.2. We then performed interpolation using the Vandermonde matrix, V, from Table 3.3 to find the symbolic coefficients. Finally, we verified that these coefficients indeed produce the eigenprojectors. To do this, we wrote each eigenprojector as a linear combination of the basis matrices of $\mathfrak{A}$ and then checked that they satisfy:

- Idempotence: $E_jE_j = E_j$ for all j,
- Orthogonality: $E_iE_j = 0$ for all $i \neq j$, and
- Completeness: $\sum_{j=0}^{4} E_j = I$.

These checks were performed computationally, leveraging the fact that for any two matrices A, B in $\mathfrak{A}$, we can express A and B in terms of the basis matrices $\{A_i : i = 1, \ldots, 69\}$ of $\mathfrak{A}$ using coefficient vectors c_i and d_j. To see why this is, first we must recall that for any $i, j \in [69]$, the structural constants of our coherent configuration allow us to write the product A_iA_j as:

$$A_iA_j = \sum_{k=1}^{69} p_{ij}^k A_k.$$

Now, suppose:

$$A = \sum_{i=1}^{69} c_i A_i, \quad \text{and} \quad B = \sum_{j=1}^{69} d_j A_j.$$

Then their product can be written as:

$$\begin{aligned} AB &= \left(\sum_{i=1}^{69} c_i A_i \right) \left(\sum_{j=1}^{69} d_j A_j \right) \\ &= \sum_{i,j=1}^{69} c_i d_j A_i A_j \\ &= \sum_{i,j=1}^{69} c_i d_j \left(\sum_{k=1}^{69} p_{ij}^k A_k \right) \\ &= \sum_{k=1}^{69} A_k \left(\sum_{i,j=1}^{69} p_{ij}^k c_i d_j \right). \end{aligned}$$

This property means that after interpolating the p_{ij}^k structure constants (as detailed in Chapter 3), we are able to verify the idempotence, orthogonality, and completeness conditions of the eigenprojectors using just the coefficient vectors.

Once the conditions of the eigenprojectors are verified, Equation (4.2) allows us to determine the coefficients of the generalized inverse of M. Readers interested in verifying these calculations can use our Sage worksheet, available at `https://github.com/pbd345/sudoku`. This worksheet enables the computation of various symbolic products in $\mathfrak{A}$ using the eigenprojector coefficients and the coefficients of $(M + \eta K)^{-1}$ from Table 4.1.

4.6.3 A norm bound on the generalized inverse

In Chapter 5 we will establish the need for a norm bound for our perturbed system because this bound will help ensure the presence of a non-negative, entrywise solution to our linear system (2.2). Instead of finding an exact bound, we apply the sub-multiplicative property of the induced matrix ∞-norm to determine a general upper limit. Therefore, our goal of this section is to set a norm boundary for the generalized inverse, $(M + \eta K)^{-1}$.

We begin the chapter with a quick reminder of the definition of the ∞-norm for vectors:

$$\|\mathbf{x}\|_\infty = \max_i |x_i|.$$

This means that the ∞-norm of a vector equals the entry in that vector with the greatest absolute value. Similarly, the induced ∞-norm for a matrix, A, is the largest

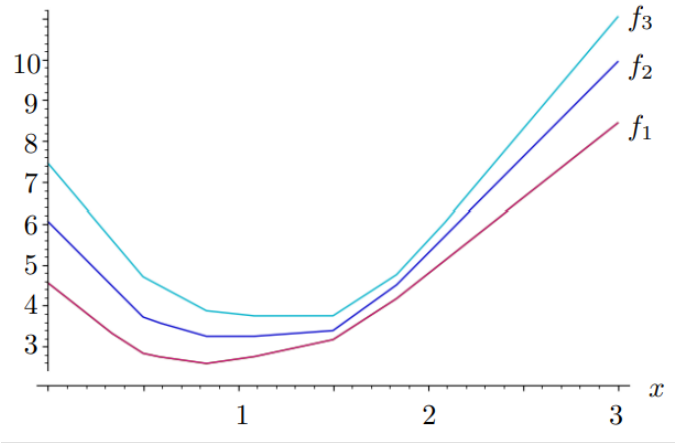


Figure 4.10: $(M + \eta K)^{-1}$ norm plots (f_1, f_2, f_3) as functions of $x := n/\eta$.

sum of absolute values within a single row of that matrix written as:

$$\|A\|_\infty = \max_i \sum_j |A_{ij}|.$$

To obtain a bound on the ∞-norm of Equation (4.2) using the values in Tables 3.4 and 4.1, we store the coefficients of the eigenprojectors and make note of their signs. For each of the four sections corresponding to the four edge types (row-column, row-symbol, column-symbol, and box-symbol) we used computational assistance to sum the absolute values of the eigenprojector coefficients multiplied by the relevant adjacency matrix row sums from that section (from Table 3.4). When we combine these, as shown in Equation (4.2), the result is three piece-wise linear functions (a duplicate occurs for the sections corresponding to row-symbol and column-symbol edges), each multiplied by n^{-1}. These computer-generated functions give the value of the dominant term in our norm bound of the generalized inverse and can be simplified to the following:

$$\begin{aligned}
f_1(x) &= 3|\tfrac{x}{2} - \tfrac{3}{4}| + 4|\tfrac{x}{6} - \tfrac{5}{36}| + 2|\tfrac{x}{12} - \tfrac{7}{144}| + 2|\tfrac{x}{12} - \tfrac{13}{144}| + 2|\tfrac{x}{3} - \tfrac{11}{18}| + 3|\tfrac{x}{2} - \tfrac{1}{4}| + 1, \\
f_2(x) &= 2|\tfrac{x}{2} - \tfrac{3}{4}| + 4|\tfrac{x}{6} - \tfrac{5}{36}| + 2|\tfrac{x}{12} - \tfrac{7}{144}| + 2|\tfrac{x}{12} - \tfrac{13}{144}| + |\tfrac{x}{3} - \tfrac{11}{18}| + |\tfrac{x}{3} - \tfrac{1}{9}| + 2|\tfrac{x}{2} - \tfrac{1}{4}| + 1, \\
f_3(x) &= 3|\tfrac{x}{2} - \tfrac{3}{4}| + |\tfrac{x}{4} - \tfrac{25}{48}| + 6|\tfrac{x}{6} - \tfrac{5}{36}| + 3|\tfrac{x}{12} - \tfrac{13}{144}| + 3|\tfrac{x}{3} - \tfrac{11}{18}| + 3|\tfrac{x}{2} - \tfrac{1}{4}| + 1.
\end{aligned}$$

Graphs for these functions are shown in Figure 4.10. Through computer assistance we computed a minimizing value for $\max\{f_i(x) : i = 1, 2, 3\}$, $x = 3/2$ or equivalently, $\eta = 2n/3$. Plugging this value into $\max\{f_i(x) : i = 1, 2, 3\}$ gives a dominant term $15/4n$, which is an upper bound of $\|(M + \eta K)^{-1}\|_\infty$ for all choices of $h, w \geq 2$. The result of computing a norm bound of our generalized inverse in the form (4.2) when $x = 3/2$ is summarized in the following lemma.

Lemma 4.6.2. *Let* $A = M + \frac{2n}{3}K$. *Then*

$$\|A^{-1}\|_\infty = \frac{15}{4n} - \frac{7(h+w)}{8n^2} - \frac{4}{9n^2} + \frac{31(h+w)-21}{72n^3} < \frac{15}{4n}.$$

We also use the coefficients of E_0 in $\mathfrak{A}$ and the row-sums (degrees) from Table 3.4 to find the following bound on the maximum row-sum of $K = E_0$.

Lemma 4.6.3. *We have* $\|K\|_\infty \leq \frac{11}{2} - \frac{17(h+w)}{6n} + O(n^{-1})$.

Chapter 5
Perturbation and the Main Result

In Section 4.6 we gave a construction for a generalized inverse of M, which we optimized and denoted as M^+. Here we show that for a non-empty partial Sudoku, P, it is possible to construct a similar generalized inverse for M_P by restricting the kernel to the edges of G_P. That is, we let $K[P]$ denote the principal sub-matrix of K whose rows and columns correspond to the edges of G_P for some partial Sudoku P. The following proposition gives key orthogonality relations, showing that the range of $K[P]$ is orthogonal to the all-ones vector and to the range of M_P. They are similar to those detailed in [5, Proposition 2.5].

Proposition 5.0.1. (a) $K[P]\mathbb{1} = \mathbf{0}$*; and* (b) $K[P]M_P = O$.

Proof. For any matrices and vectors indexed by $E(G_{hw})$, sort the indices so that those corresponding to $E(G_P)$ come first. Similarly, for those indexed by $\mathcal{T}(G_{hw})$, sort the indices so that those corresponding to $\mathcal{T}(G_P)$ come first. Now, let L be the inclusion map from edges of G_P to edges of G_{hw}. Specifically, L is a matrix where the rows correspond to edges of G_{hw} and the columns correspond to edges of G_P. The top of L is an identity matrix, I, representing the direct mapping of edges in G_P to themselves in G_{hw}, and the bottom is a zero matrix, O, indicating that there is no mapping between the additional edges in G_{hw} to any edges in G_P. We can write $K[P] = L^\top KL$. Similarly, let Q be the inclusion map from tiles of G_P to tiles of G_{hw}, with a similar structure. The rows correspond to tiles of G_{hw} and the columns correspond to tiles of G_P, so we again get an identity matrix on top and a zero matrix on the bottom.

Let $\mathbb{1}_P = (\mathbb{1} \mid \mathbf{0})$ be the $4n^2 \times 1$ $\{0,1\}$ indicator vector of $E(G_P)$ in $E(G_{hw})$. Alternatively, $\mathbb{1}_P$ is obtained from the $4n^2 \times 1$ all-ones vector by subtracting indicator vectors of tiles corresponding to symbols placed in P. It follows that $\mathbb{1}_P$ is contained in the range of $W^\top$, and hence is orthogonal to $\ker(W^\top) = \ker(M)$. As such, we have

$(K\mathbb{1}_P) = \mathbf{0}$. We prove (a) with a simple computation:

$$\begin{aligned} K[P]\mathbb{1} &= L^\top KL\mathbb{1} \\ &= (L^\top K)(L\mathbb{1}) \\ &= (L^\top K)\mathbb{1}_P \\ &= L^\top(K\mathbb{1}_P) \\ &= \mathbf{0}. \end{aligned}$$

To prove (b) we sort the rows and columns of W as before to get

$$W = \left[\begin{array}{c|c} W_P & * \\ \hline O & * \end{array}\right].$$

This works because the edges that are in G_{hw} but not in G_P cannot be in the tiles that are still full in G_P. We then can notice that LW_P and WQ are both indexed with rows corresponding to edges of G_{hw} and the columns correspond to tiles of G_P. It is straightforward enough to calculate that for any edge $e \in E(G_{hw})$ and tile $t \in \mathcal{T}(G_P)$, we have

$$LW_P(e,t) = WQ(e,t) = \begin{cases} 1 \text{ if } e \in t, \\ 0 \text{ otherwise.} \end{cases}$$

So, we see that $LW_P = WQ = [W_P \mid O]^\top$. Working with these,

$$\begin{aligned} K[P]M_P &= (L^\top KL)(W_P W_P^\top) \\ &= (L^\top K)(LW_P)(W_P^\top) \\ &= (L^\top K)(WQ)(W_P^\top) \\ &= (L^\top)(KW)(QW_P^\top) \\ &= O, \end{aligned}$$

since $\ker(M) = \ker(W^T)$ and therefore $KW = O$. □

Proposition 5.0.1 means that we can apply Lemma 4.6.1 with M_P taking on the role of A, a nonzero multiple of $K[P]$ taking the role of B, and $\mathbb{1} = \mathbf{b}$ as long as $M_P + \eta K[P]$ is invertible. Therefore, for any partial Sudoku P for which $M_P + \eta K[P]$ is invertible, we have a guaranteed solution to the linear system (2.2), of the form $(M_P + \eta K[P])^{-1}\mathbb{1}$. Recall we want such a solution to be entrywise non-negative. Thus, for the remainder of this chapter we seek a minimum degree condition on G_P for which it is guaranteed that $M_P + \eta K[P]$ is invertible and that $(M_P + \eta K[P])^{-1}\mathbb{1}$ is non-negative.

5.1 From ϵ-density to $(1-\delta)$-availability

This thesis is concerned with partial Sudoku which are nearly empty. Recall the definition of ϵ-dense discussed in Section 1 which we strengthened for the case of Sudoku. This definition leads easily to various degree bounds in G_P, summarized here.

Lemma 5.1.1. *Suppose P is ϵ-dense. Then in the graph G_P, the number of edges from vertex*

- *c_j to the row partite set is at least $(1-\epsilon)n$;*
- *r_i to the column partite set is at least $(1-\epsilon)n$;*
- *s_k to any row bundle is at least $(1-\epsilon)h$;*
- *s_k to any column bundle is at least $(1-\epsilon)w$;*
- *r_i, c_j or b_ℓ to the symbol partite set is at least $(1-\epsilon)n$;*
- *s_k to the box partite set is at least $(1-\epsilon)n$.*

This inspires an alternate sparseness definition which is well suited to our approach.

Definition 5.1.2. Let us say that a partial Sudoku P has the $(1-\delta)$-*availability property* if every edge $e \in E(G_P)$ is contained in at least $(1-\delta)n$ tiles in $\mathcal{T}(G_P)$.

With this definition, we are now equipped to explain the relationship between the notions of $(1-\delta)$-availability, essentially $(1-\delta)$-availability, and ϵ-density of a partial Sudoku.

Lemma 5.1.3. *An ϵ-dense partial Sudoku necessarily possesses the $(1-3\epsilon)$-availability property. Conversely, if P exhibits the $(1-\delta)$-availability property, then P is δ-dense, with the exception of completely filled rows, columns, boxes, or cases where a symbol appears h times in a row-bundle or w times in a column-bundle, which indicates that the symbol cannot be used again within that bundle.*

Proof. Given an ϵ-dense partial Sudoku, we first want to show that it has the $(1-3\epsilon)$-availability property. To do this, we consider the various edge types:

1. For an edge $\{r_i, c_j\}$:
   - At most ϵn symbols are already used in row i.
   - At most ϵn symbols are already used in column j.
   - At most ϵn symbols are already used in box(i,j).

   Even if these symbols are distinct, we still have at least $(1-3\epsilon)n$ available tiles for an edge of this type.

2. For an edge $\{r_i, s_k\}$:
   - At most ϵn columns are filled in row i.

- At most ϵn columns already contain symbol k.
- At most ϵn columns are unavailable due to boxes along row i already having symbol k.

Again, we see that there are at least $(1-3\epsilon)n$ available tiles for an edge of this type. Edges of the form $\{c_j, s_k\}$ behave similarly to the $\{r_i, s_k\}$ edge type.

3. Finally, for an edge $\{s_k, b_\ell\}$:
   - At most ϵn options are unavailable due to filled cells in box ℓ.
   - There are up to $(\epsilon h)w$ cells of box ℓ unavailable because symbol k is present elsewhere in the same row bundle, and likewise there are $(\epsilon w)h$ cells unavailable because k is present elsewhere in the same column bundle. If we ignore overlap where the row and column bundles intersect, these sum to at most $2\epsilon n$ unavailable tiles.

Thus, for each of these edge types, we observe that there are at least $(1-3\epsilon)n$ available tiles in $\mathcal{T}(G_P)$.

Next, we show that if P exhibits the $(1-\delta)$-availability property then every row, column, and box which is not completely filled has at most δn filled entries, and every symbol which does not saturate a row (column) bundle has at most δh (respectively δw) occurrences in that bundle. Assuming P has the $(1-\delta)$-availability property, we know that any row, column, symbol, or box that isn't fully occupied will have at most δn occurrences. Further, if symbol s_k appears fewer than h times in a row bundle, it can only appear at most δh times. Otherwise, we would be able to take a box b_ℓ in this bundle with empty cells and observe that $\{s_k, b_\ell\}$ has fewer than $n - w(\delta h) = (1-\delta)n$ tiles available in $\mathcal{T}(G_P)$. An analogous argument can be made for column bundles.

In other words, the $(1-\delta)$-availability property implies P is δ-dense, except possibly for completely filled rows, columns, boxes, or any symbols fully used in a bundle. This completes the proof. □

5.2 Changes to M resulting from pre-filled entries

Let P be a partial (h, w)-Sudoku, with $hw = n$. Recall that G_P is the graph obtained from G_{hw} by deleting the edges of tiles corresponding to pre-filled entries in P. Suppose P has the $(1-\delta)n$ availability property. Let $M = WW^\top$ and $M_P = W_P W_P^\top$, as usual. To set up our perturbation argument, we are interested in quantifying the change in M resulting from pre-filling the entries of P. It makes no sense to subtract M_P from M directly, since these matrices have different sizes. However, we can use a convenient border.

Let $\widetilde{M}$ denote the $4n^2 \times 4n^2$ matrix, indexed by edges of G_{hw}, whose entries are given by:

$$\widetilde{M}(e,f) = \begin{cases} M_P(e,f) & \text{if } e,f \in E(G_P); \\ 0 & \text{if } e \in E(G_P) \text{ and } f \notin E(G_P); \\ M(e,f) & \text{if } e \notin E(G_P). \end{cases} \tag{5.1}$$

If we sort the rows and columns so that those indexed by $E(G_P)$ come first, then the structure of $\widetilde{M}$ is as follows:

$$\widetilde{M} = \left[\begin{array}{c|c} M_P & O \\ \hline \text{as in } M & \end{array}\right]. \tag{5.2}$$

Sorting the edges of M in the same order as in $\widetilde{M}$, we can define $\Delta M = M - \widetilde{M}$. We next estimate $\|\Delta M\|_\infty$ under our sparseness assumption. For an edge $e \in E(G_P)$, let $U(e)$ denote the set of unavailable options:

$$U(e) = \{t \in \mathcal{T}(G_{hw}) : e \in t \text{ and } f \in t \text{ for some } f \in E(G_{hw}) \setminus E(G_P)\}.$$

Let $u(e)$ be the number of unavailable tiles that use edge e. That is, $u(e) = |U(e)|$. We can then define the vector, $\mathbf{u} = (u(e) : e \in E(G_P))$. We can now examine each case in more detail.

- If e is an edge of type row-column, say $e = r_i c_j$, then $U(e)$ tracks those symbols k which are not able to be placed in cell (i,j) because k already appears in row i, column j, or the box that cell (i,j) belongs to.

- If e is an edge of type row-symbol, say $e = r_i s_k$, then $U(e)$ tracks those columns j which are unavailable for symbol k in row i, either because cell (i,j) was pre-filled or k appears somewhere else in either column j or box $\text{box}(i,j)$. Several columns might be eliminated as options if k appears in a box intersecting row i. Edges of type column-symbol behave in an analogous way.

- If e is an edge of type box-symbol, say $e = b_\ell s_k$, then $U(e)$ tracks those cells (i,j) in box b_ℓ for which k is not allowed, either because (i,j) was already filled in P, or because k already appears in the row or column bundle for box b_ℓ.

Lemma 5.2.1. *We have* $\mathbf{0} \leq (\Delta M)\mathbb{1} \leq 4\mathbf{u}$ *entrywise. In particular,*

$$\|\Delta M\|_\infty \leq 4\|\mathbf{u}\|_\infty.$$

Proof. Entry e of the vector $(\Delta M)\mathbb{1}$ equals $\sum_f \Delta M(e,f)$. The summand is the number of unavailable tiles t with $e \in t$ and $f \in t$. Since each tile t contains four edges, this count is at most $4u(e)$. □

If P has the $(1-\delta)$-availability property, then by definition we have $\|\mathbf{u}\|_\infty \leq \delta n$. Recall that if P has the ϵ-dense property, then it has the $(1-3\epsilon)$-availability property, as explained in Section 5.1. These, together with Lemma 5.2.1 give bounds on ΔM.

Lemma 5.2.2. *With ΔM constructed from P as above, we have*

1. *$\|\Delta M\|_\infty \leq 4\delta n$ if P has the $(1-\delta)$-availability property; and*

2. *$\|\Delta M\|_\infty \leq 12\epsilon n$ if P is ϵ-dense;*

Proof. If P has the $(1-\delta)$-availability property, then by definition this means that each edge $e \in E(G_P)$ has, at most, δn unavailable tiles. So,

$$\|\Delta M\|_\infty \leq 4\|\mathbf{u}\|_\infty = 4(\max_e |u(e)|) \leq 4\delta n.$$

If P is ϵ-dense then P has the $(1-3\epsilon)$-availability property so,

$$\|\Delta M\|_\infty \leq 3(4\epsilon n).$$

This completes the proof. □

As an aside, if, say, row r_i is completely filled in P then no edge using r_i belongs to G_P. In this case, edges using r_i make no contribution to ΔM.

5.3 A guarantee on non-negative solutions

The following lemma provides a guarantee for non-negative solutions to a linear system.

Lemma 5.3.1. *Let A be an invertible matrix over the reals. Suppose $A - \Delta A$ is a perturbation. Then $A - \Delta A$ is invertible provided that*

$$\|A^{-1}\Delta A\|_\infty < 1$$

and there exists an entrywise nonnegative solution $\mathbf{x}$ to $(A-\Delta A)\mathbf{x} = A\mathbb{1}$, provided that

$$\|A^{-1}\Delta A\|_\infty \leq \frac{1}{2}.$$

Lemma 5.3.1 was used in a similar manner by Bowditch and Dukes for the Latin square case in [5, Section 3]. As noted there, the first part of Lemma 5.3.1 can be proved using the Neumann series expansion for matrix inverses, $(A-\Delta A)^{-1} = \sum_{k=0}^{\infty}(A^{-1}\Delta A)^k A^{-1}$, which converges when $A^{-1}\Delta A$ has norm less than 1. More details on the proof of this lemma can be found in [5, Section 3] and in Horn and Johnson's book [13, Section 5.8].

Applying submultiplicativity to Lemma 5.3.1 implies the existence of an entrywise non-negative solution $\mathbf{x}$ if:

$$\|A^{-1}\|_\infty \|\Delta A\|_\infty \leq \frac{1}{2}. \tag{5.3}$$

For more details, including a proof that the induced ∞-norm on matrices is submultiplicative, we again refer the reader to Horn and Johnson [13, Section 5.6].

We would like to apply this to the perturbation

$$M - \Delta M = M - (M - \widetilde{M}) = \widetilde{M},$$

but we must take care to handle the nontrivial kernel. Of various possible approaches, one convenient thing to do is to place those columns of $K = E_0$ corresponding to non-edges of G_P into the perturbation.

Thus, we let $A = M + \eta K$ and observe that

$$A\mathbb{1} = (M + \eta K)\mathbb{1} = M\mathbb{1} + \eta K\mathbb{1} = 4n\mathbb{1}.$$

That is, for this choice of A, the right side of the system in Lemma 5.3.1 is just a scalar multiple of the all-ones vector.

Define $\Delta A = \Delta M + \eta K'$, where

$$K'(e, f) = \begin{cases} 0 & \text{if } f \in E(G_P); \\ K(e, f) & \text{otherwise.} \end{cases}$$

An illustration of K' is included in Figure 5.1 for clarity.

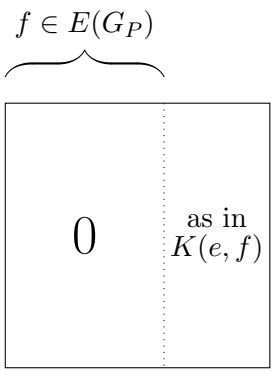


Figure 5.1: An illustration of K'.

Note that

$$A^{-1}(\eta K') = \left(\frac{1}{\eta} K + \sum_{j=1}^{4} \frac{1}{jn} E_j\right)(\eta K') = K', \tag{5.4}$$

since $KK' = K'$ and the columns of K' are orthogonal to each of the other eigenspaces.

The subsequent lemma demonstrates that if our partial Sudoku P is sparse, then the presence of zero columns in K' significantly reduces the ∞-norm in comparison to that of K.

Lemma 5.3.2. *Suppose P is ϵ-dense. Then, for large h and w,*

$$\|K'\|_\infty \leq (\epsilon + o(1))\|K\|_\infty.$$

Proof. Write $K = \sum_{i=1}^m c_i A_i$. Fix $e \in E(G_{hw})$. Then we have

$$\sum_{f \in E(G_{hw})} |K(e,f)| = \sum_i |c_i| d_i(e),$$

where $d_i(e) = |\{f : (e,f) \in R_i\}|$.

Recall that $d_i(e)$ is zero and contributes nothing to this sum unless e is of an edge type corresponding to the first coordinate of R_i, and we may assume a canonical choice r_1c_1, r_1s_1, c_1s_1, or s_1b_1 for e. Using the Sage worksheet at `https://github.com/pbd345/sudoku`, one can compute the leading terms in the product $c_i d_i(e)$ for each relation index i. The results are shown in Table 5.1, divided into columns according to the edge type of e. With the exception of $i \in I := \{1, 2, 4, 16, 32, 62\}$, each relation R_i has an associated feature which, owing to our ϵ-density assumption, limits the number of missing edges f in G_P with $(e,f) \in R_i$. These sparse features are indicated in Table 5.1, with a legend and upper bound on missing edges given in Table 5.2.

Let $\overline{G_P}$ denote the complement of G_P in G_{hw}. Then we have

$$\sum_{f \in E(G_{hw})} |K'(e,f)| = \sum_{f \in E(\overline{G_P})} |K(e,f)| = \sum_{i=1}^m |c_i| d_i'(e), \tag{5.5}$$

where $d_i'(e)$ is the number of edges $f \in E(\overline{G_P})$ with $(e,f) \in R_i$. When h and w are large, it suffices to sum over indices $i \notin I$. For such i, we have $d_i'(e) \leq (\epsilon + o(1))d_i(e)$. The lower order term is one of a few possibilities depending on i. For instance, relation R_3 corresponds to edges with equal rows and columns from different bundles. We have $d_3'(e) \leq \epsilon n$, while $d_3(e) = n - w$, so their ratio is $\epsilon + O(w/n) = \epsilon + O(h^{-1})$. Similar estimates hold for the other relations. The result follows from (5.5) by upper-bounding each term with $i \notin I$. □

i	leading term	sparse feature	i	leading term	sparse feature	i	leading term	sparse feature	i	leading term	sparse feature
1	$3/2n$	-	13	$-1/2$	r	25	$-1/2$	c	42	$-1/2$	b
2	$1/h$	-	14	$1/6$	rb	26	$1/6$	cb	43	$1/6$	rb
3	$1/2$	r	15	$1/12$	all	27	$1/12$	all	44	$1/6$	cb
4	$1/w$	-	16	$1/2h$	-	30	$-1/3$	s	45	$-1/12$	all
5	$1/2$	b	17	$1/2$	r	31	$1/12$	all	50	$-1/2$	srb
6	$-1/3$	rb	18	$1/2$	srb	32	$1/2w$	-	51	$1/6$	rb
7	$1/2$	c	19	$-1/3$	rb	33	$1/2$	c	52	$1/6$	s
8	$-1/3$	cb	20	$1/6$	s	34	$1/2$	scb	53	$-1/12$	all
9	$-1/12$	all	21	$-1/12$	all	35	$-1/3$	cb	58	$-1/2$	scb
10	$-1/2$	r	28	$-1/3$	s	36	$1/6$	s	59	$1/6$	cb
11	$1/6$	rb	29	$1/12$	all	37	$-1/12$	all	60	$1/6$	s
12	$1/12$	all	46	$-1/2$	srb	54	$-1/2$	scb	61	$-1/12$	all
22	$-1/2$	all	47	$1/6$	rb	55	$1/6$	cb	62	$(h+w)/2n$	-
23	$1/6$	cb	48	$1/6$	s	56	$1/6$	s	63	$1/2$	b
24	$1/12$	all	49	$-1/12$	all	57	$-1/12$	all	64	$1/2$	srb
38	$-1/2$	b							65	$-1/3$	rb
39	$1/6$	rb							66	$1/2$	scb
40	$1/6$	cb							67	$-1/3$	cb
41	$-1/12$	all							68	$-1/3$	s
									69	$1/4$	all

Table 5.1: Terms contributing to $\|K\|_\infty$

	sparse feature	bound		sparse feature	bound
r	cells filled in a row	ϵn	b	cells filled in a box	ϵn
c	cells filled in a column	ϵn	s	occurrences of a symbol	ϵn
rb	cells filled in a row bundle	ϵnh	srb	times a symbol is in a row bundle	ϵh
cb	cells filled in a column bundle	ϵnw	scb	times a symbol is in a column bundle	ϵw
all	cells filled in all of P	ϵn^2			

Table 5.2: Legend for Table 5.1 and ϵ-density bounds

Putting together Lemmas 4.6.2, 4.6.3, 5.2.2 and 5.3.2, we obtain a bound on $A^{-1}\Delta A$.

Proposition 5.3.3. *Suppose P is an ϵ-dense partial (h, w)-Sudoku where h, w are large. Then $\|A^{-1}\Delta A\|_\infty < 101\epsilon/2 + o(1)$.*

Proof. From Equation (5.4), we derive the following relationship:

$$\|A^{-1}\Delta A\|_\infty = \|A^{-1}(\Delta M) + K'\|_\infty.$$

Applying the triangle inequality to the right-hand side, we obtain:

$$\|A^{-1}(\Delta M) + K'\|_\infty \leq \|A^{-1}(\Delta M)\infty + \|K'\|_\infty.$$

Next, utilizing the property of submultiplicativity, we further refine our bounds:

$$\|A^{-1}(\Delta M)\infty + \|K'\|_\infty \leq \|A^{-1}\infty\|(\Delta M)\infty + \|K'\|_\infty.$$

Incorporating Lemma 5.3.2 under the assumption that both h and w are sufficiently large, the expression can be rewritten:

$$\|A^{-1}\infty\|(\Delta M)\infty + \|K'\|_\infty \leq \|A^{-1}\infty\|(\Delta M)\infty + (\epsilon + o(1))\|K\|_\infty.$$

Finally, by applying the constraints from Lemmas 4.6.2, 4.6.3, and 5.2.2, we arrive at a more concise inequality:

$$\|A^{-1}\infty\|(\Delta M)\infty + (\epsilon + o(1))\|K\|_\infty \leq \left(\frac{15}{4n}\right)(12\epsilon n) + \frac{11}{2}\epsilon + o(1).$$

Putting this all together and simplifying, we conclude that:

$$\|A^{-1}\Delta A\|_\infty < \frac{101\epsilon}{2} + o(1).$$

□

It is worth a remark that the lower order terms in Lemmas 4.6.2 and 4.6.3 are actually negative. This means our hypothesis of large h and w is only really used to control the mild $o(1)$ term in K'. In general, our method is robust for small partial Sudoku, often succeeding in practice with densities much larger than $1/101$.

5.4 Proving the main result

We are now ready to prove our result on fractional completion of partial Sudokus under the ϵ-dense assumption.

Proof of Theorem 1.2.4. Apply Lemma 5.3.1 to A and ΔA constructed as above.

Under the assumption $\epsilon < 1/101$, Proposition 5.3.3 gives

$$\|A^{-1}\Delta A\|_\infty < 1/2$$

for sufficiently large h, w. This implies the existence of an entrywise nonnegative solution to

$$(A - \Delta A)\mathbf{x} = \mathbb{1}.$$

Let $\mathbf{x}'$ denote the restriction of $\mathbf{x}$ to $E(G_P)$. Since $A - \Delta A$ is block lower-triangular with respect to the partition into edges and non-edges of G_P, it follows that

$$(M_P + \eta K[P])\mathbf{x}' = \mathbb{1}.$$

We note that both M_P and $K[P]$ are symmetric and satisfy the conditions in Proposition 5.0.1. Therefore, Lemma 4.6.1 implies $M_P\mathbf{x}' = \mathbb{1}$. This, in turn, implies an entrywise non-negative solution to the linear system (2.1), and thus implies the existence of a fractional completion for partial Sudoku P. □

Chapter 6
Final remarks

6.1 Thin boxes

When we began our work we considered whether it made sense to restrict our generalization of Sudoku so that the boxes remained square. Ultimately, the choice to allow rectangular boxes was made for two reasons. Rectangular boxes give a more general result and allow more opportunities for application while also providing the very practical advantage of allowing us to explore more cases of low-valued h and w computationally. The allowance for rectangular boxes enabled an exploration of small cases such as $(h, w) \in \{(2, 2), (3, 2), (4, 2), (3, 3), (2, 5), (3, 4)\}$ whereas restricting to square boxes would have meant computing and examining exceptionally large versions of our matrix M very quickly. However, the statement of our main result includes the condition that both dimensions h and w need to be sufficiently large. In contrast, this section delves into a detailed investigation of (h, w)-Sudoku with a fixed width w and sufficiently large height $h = n/w$. We refer to these Sudoku as having *thin boxes*.

In Section 1, we observed that there exist incompletable partial $n \times n$ Sudoku with no row, column, symbol, or box used very often. This motivated our bundle condition in our definition of ϵ-density for Sudoku. The idea is that symbol k can be forced in an entry (i, j) using only $O(h)$ pre-filled occurrences of k in the row bundle containing r_i and $O(w)$ pre-filled occurrences of k in the column bundle containing c_j. For $n \times n$ Sudoku with square or approximately square boxes, this is sub-linear in n. However, for Sudoku with thin boxes this construction requires a symbol be used $O(n)$ times. In this way, the structure of Sudoku with thin boxes, particularly when considering much larger values of h, is very reminiscent of Latin squares. In fact, for the sake of analogy, one can interpret the columns of a Latin square of order n as tall, thin boxes of width 1 and height n. With this observation, we postulated that all sufficiently large partial Sudoku of type $(\frac{n}{w}, w)$ which are ϵ-dense *in the sense of Latin squares* could still admit a completion. We were able to prove this in the fractional sense with Proposition 6.1.5, with the added condition that the value of ϵ is dependent on w.

This result does not follow directly from our earlier work. For example, consider

the case where $w = 2$. Here, a cell in the top-left box that has been pre-filled with symbol k prevents any placement of k in the same column's bottom-left box. This eliminates $n/2$ options for placing k in that box, resulting in a perturbation which is far too severe for the methods we employ in Chapter 5. Figure 6.1 illustrates this, with red shading in the cells in the bottom-left box that are unavailable for symbol k.

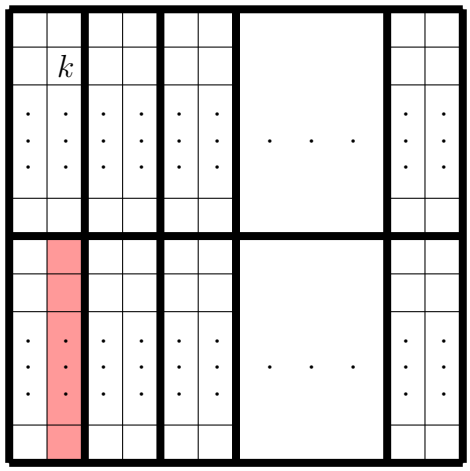


Figure 6.1: Severe perturbation visualization in a $(\frac{n}{2}, 2)$-Sudoku.

We now present a general construction that establishes a barrier to fractional completion in the context of a fixed box width, w. This construction is a generalization of the one illustrated in Figure 1.8b.

Proposition 6.1.1. *Let w be a fixed integer, $w \geq 2$. For any $\epsilon > \frac{1}{3w}$, there exists a partial (h, w)-Sudoku with no completion that uses every row, column, symbol, and box at most ϵhw times.*

Proof. Suppose $h \geq w$. Put $a = \lceil (h + w)/3 \rceil$ and $n = hw$. As $n = hw \geq 2a$ when $h \geq w \geq 2$, we can choose $2a$ unique symbols and distribute them into two disjoint sets A and A' of size a. Let $\mathcal{L}$ be the column bundle containing the first w columns and box numbers $hj + 1$ for $j \in \{0, 1, 2, ..., w - 1\}$. To construct such a partially filled (h, w)-Sudoku S, start with an empty Sudoku grid and then:

(a) Put the elements of A' in entries $(i, 1)$, $i = 1, \ldots, a$;

(b) Put the elements of A in column j of the jth box of $\mathcal{L}$ for each $j = 2, \ldots, w$; and

(c) Put the elements of A strictly to the right of $\mathcal{L}$ in row i for $i = a+1, \ldots, 2a-w+1$.

Note that placements (a) and (b) fit within their respective boxes, as we assume $h \geq w \geq 2$ which ensures $a \leq h$. We also claim that placement (c) can be executed without repeating elements in any $h \times w$ box. To achieve this, we order the elements of A somehow in the first of these rows, starting at column $w + 1$, and then shift this

same ordering to the left by multiples of w in each successive row. To check that there is room for this, we note that $(a-w)w + a \leq w(h-1) = n - w$. This holds with equality for $h = w = a = 2$ and otherwise can be easily verified with the estimate $a \leq (h+w+2)/3$. By construction, every row, column, and box contains at most a elements. Each symbol from A occurs exactly $(w-1) + (a-w+1) = a$ times and each symbol from A' occurs once and only once.

Now, suppose that there exists a fractional completion of S. We start by examining the placement of elements from A in the first box. By condition (b), columns $\{2, \ldots, w\}$ are unavailable and, similarly, condition (a) and (c) block the placement in rows $\{1, \ldots, 2a-w+1\}$ of the first column. So, the elements of A must fit into the entries $(i, 1)$ for $2a - w + 1 < i \leq h$, but there are only $h + w - 2a - 1 < a$ such entries. This is a contradiction, and hence S has no fractional completion. Taking h sufficiently large, we can ensure $a/n < \epsilon$. □

Proposition 6.1.1 shows that, in the case of thin boxes with fixed w, any result guaranteeing fractional completion with every row, column, symbol, and box used at most ϵhw times must have ϵ depending on w. Our methods can produce a result of this form with only minor adaptation. First, we establish a technical lemma for adding entries to Latin rectangles.

Definition 6.1.2. A *Latin rectangle* is an $m \times n$ array with each cell containing an entry from a set of n symbols, with the condition that each symbol must appear exactly once in every row and at most once in every column.

Of course, the notion of a partial Latin rectangle is analogous to that of a partial Latin square. However, the notion of ϵ-density must be adjusted in the case of Latin rectangles.

Definition 6.1.3. An $m \times n$ partial Latin rectangle is *ϵ-dense* if every row has at most ϵn filled cells, every column has at most ϵm filled cells, and every symbol is used at most ϵm times.

With these definitions in hand, we can state our lemma.

Lemma 6.1.4. *Suppose $0 < \epsilon, \delta < \frac{1}{10}$. Let $n \geq m$ and suppose we are given an ϵ-dense $m \times n$ partial Latin rectangle P. Let $A_1, \ldots, A_n \subset [n]$ with $|A_j| < \delta m$ for each j and $|\{j : k \in A_j\}| < \delta m$ for each k. Then P is contained in a $3(\delta + \epsilon)$-dense $m \times n$ partial Latin rectangle P' such that, for each $j = 1, \ldots, n$, column j of P' contains the elements of A_j.*

Proof. We construct P' from P by adding symbols from A_j one column at a time. It can be assumed that A_j is disjoint from the set of symbols already in column j of P by replacing A_j with a smaller set. Suppose we have extended P in the first $j-1$ columns

using $A_1, A_2, ..., A_{j-1}$ for some $1 \leq j \leq n$, and let P_j be the resulting array. We may sort the rows of P_j in weakly increasing order of how many symbols are in each. Let B_j consist of the first $\lceil(\epsilon + 2\delta)m\rceil$ row indices; those correspond to rows with the fewest number of symbols. Let G_j denote the bipartite graph with vertex partition A_j, B_j and an edge drawn between $k \in A_j$ and $r \in B_j$ if and only if symbol k is available to be placed in row r of column j.

We claim that G_j has a matching that uses every element of A_j. Consider symbol k in A_j. It can be placed in any row that doesn't already have symbol k. But k appears in at most ϵm rows of P and was previously added at most δm times in forming P_j. Therefore,

$$\deg_{G_j}(k) \geq (\epsilon + 2\delta)m - \epsilon m - \delta m = \delta m > |A_j|.$$

So our claim of the existence of the matching follows by Hall's Theorem. Let P' be the resulting array after extending P_n.

Checking that P' is $3(\epsilon + \delta)$-dense, the column condition and symbol condition follow immediately from P being ϵ-dense and the hypotheses on the sets A_j. Suppose for contradiction that row r contains more than $3(\epsilon + \delta)n$ symbols. By choice of B_j, when the last symbol was added to row r, there were at least $\lfloor(1 - \epsilon - 2\delta)m\rfloor$ other rows with at least $3(\delta + \epsilon)n - 1 \geq 2(\delta + \epsilon)n$ symbols. (Here, we used $\epsilon n \geq 1$ since otherwise P is empty.) These rows, along with row r, would account for at least $\frac{m}{2} \times 2n(\delta + \epsilon) = mn(\delta + \epsilon)$ filled entries. But this exceeds the total number of filled entries in P', leading to a contradiction. □

We now present a result guaranteeing fractional completion for the case of thin boxes. To achieve this, we utilize Lemma 6.1.4, which involves adding symbols with low availability to boxes in the same column bundle as necessary to produce a partial Sudoku that possesses the $(1 - \delta)$-availability property, first discussed in Section 5.1. We will then complete the process using the methods and estimates outlined in Lemmas 5.2.2 and 5.3.2.

Proposition 6.1.5. *For each $w \geq 2$, there exists a constant ϵ, dependent on w, with the following property: If a partial Sudoku of type $(\frac{n}{w}, w)$ is sufficiently large and each of its rows, columns, symbols, and boxes is utilized no more than ϵn times, then that partial Sudoku admits a fractional completion.*

Proof. Let S be an ϵ-dense partial Sudoku of type $(n/w, w)$ and let P be the restriction of S to the top row bundle. Then P is an $m \times n$ partial latin rectangle, where $m = n/w$, and P is $w\epsilon$-dense. Consider the column bundles of S. For the i^{th} column bundle, let Z_i be the set of symbols which occur in this bundle outside of P. Our strategy is to add the symbols of Z_i to the i^{th} column bundle of P.

Let A_j, $j = 1, \ldots, n$, be any sets of symbols whose disjoint union over the ith bundle equals Z_i for each i. Note that $|A_j| \leq |Z_i| \leq \epsilon nw = \epsilon m w^2$ and every symbol

appears in at most $\epsilon n = \epsilon mw$ of the A_j. Lemma 6.1.4 allows us to add the entries of A_j to column j of P, creating a Latin rectangle P' that is $O(w^2)\epsilon$-dense. Let us carry out the same procedure for each of the w row bundles of S. This produces a partial Sudoku S' with the following properties:

1. Every row, column, symbol and box is used at most $O(w^3)\epsilon n$ times in S';
2. For every box b of S', the symbols that occur in the column bundle containing b also occur within b itself.

These two properties mean that a column-symbol edge $\{c_j, s_k\} \in E(G_{S'})$ loses no options for tiles because of symbol k occurring in a different box of the column bundle containing c_j and, similarly, a box-symbol edge $\{b_\ell, s_k\} \in E(G_{S'})$ loses no options for tiles. This means S' has the $(1-\delta)$-availability property, where $\delta = C_1\epsilon$ for some constant C_1 depending only on w.

Let us construct ΔM based on S' as in Section 5.2. By Lemma 5.2.2, we have $\|\Delta M\|_\infty < 6C_1\epsilon n$. We make one minor change to the perturbation set-up of Section 5.3. Define $\Delta A = \Delta M + \eta K''$, where

$$K''(e,f) = \begin{cases} 0 & \text{if } e \notin E(G_{S'}) \text{ or } f \in E(G_{S'}); \\ K(e,f) & \text{otherwise.} \end{cases}$$

For clarity, we provide a simple illustration of K'' in Figure 6.2.

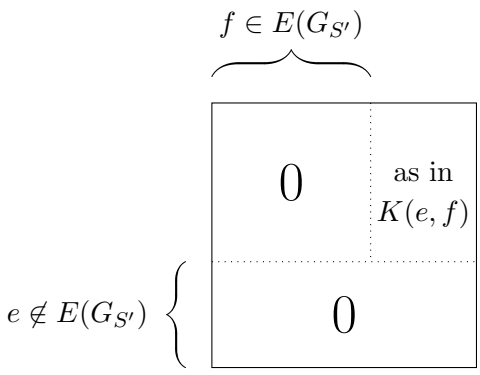


Figure 6.2: An illustration of K''.

For $\|K''\|_\infty$, it is straightforward to adapt Lemma 5.3.2 to get a weaker upper bound of the form $\|K''\|_\infty < C_2\epsilon$ where C_2 depends on w. A few additional remarks are useful. First, K'' vanishes on the diagonal and so receives no contribution from relations R_i, for $i \in \{1, 16, 32, 62\}$. Next, K'' also vanishes on entries supported by those relations R_i, for $i \in \{34, 54, 58, 66\}$, all of which involve constraints on the number of times a symbol can be present in a column bundle (those marked 'scb' in Table 5.2). This is because, by the second property of S', if $(e,f) \in R_i$ for $i \in \{34, 54, 58, 66\}$ with

$e \in E(G_{S'})$ then that implies $f \in E(G_{S'})$. Finally, since w is fixed and $h = n/w$ is sufficiently large, the number of nonzero entries of K'' supported by relation R_4 is bounded by ϵh times a function of w from the sparsity of each column of S'. Bounds on contributions from all other relations are as in Table 5.1.

Letting $A = M + \eta K$ for (say) $\eta = n$, the bound in Lemma 4.6.2 gives $\|A^{-1}\|_\infty < C_3 n^{-1}$. Choose $\epsilon(w) = \frac{1}{2C_3(4C_1+C_2)}$. Then with this choice of ϵ, we have

$$\|\Delta A\|_\infty \leq \|A^{-1}\|_\infty(\|\Delta M\|_\infty + \|nK''\|_\infty) < C_3(4C_1 + C_2)\epsilon = \frac{1}{2}.$$

It follows from Lemma 5.3.1 that the system $(A-\Delta A)\mathbf{x} = \mathbb{1}$ has a nonnegative solution $\mathbf{x}$, and we finish the proof as before by restricting $\mathbf{x}$ to the edges of G_S. □

We have omitted explicit estimates on the constants C_i. We believe that the main place where an improvement could be made is in the constant C_1 related to our construction of S'.

6.2 Exploration using the Schur complement

The work presented in this thesis was, of course, nowhere near as straightforward as this structured thesis would suggest. Once found, the Sudoku adjacency algebra $\mathfrak{A}$ was an obvious and elegant means to proving our main result. However, it was not immediately apparent. Early in our work we saw that a nice low-dimensional algebra arising from an association scheme, like the one found in the analogous case of Latin squares in [5], would not work for Sudoku. As such, we spent many hours trying to cobble together a bound on the infinity norm of a generalized inverse of our matrix M, a key ingredient in our main result. One tool that we thought might be appropriate for approaching this problem was the Schur complement.

Definition 6.2.1. Given a block matrix

$$A = \begin{bmatrix} A_1 & A_2 \\ A_3 & A_4 \end{bmatrix},$$

with A_4 invertible, we can decompose A into the following:

$$A = \begin{bmatrix} I & A_4^{-1}A_2 \\ 0 & I \end{bmatrix} \quad \begin{bmatrix} A/A_4 & 0 \\ 0 & A_4 \end{bmatrix} \quad \begin{bmatrix} I & 0 \\ A_4^{-1}A_3 & I \end{bmatrix}$$

where $A/A_4 = A_1 - A_2A_4^{-1}A_3$ is called the *Schur complement* of A with respect to A_4. A key property of the Schur complement is that A is invertible if and only if A/A_4 is

invertible and, in fact, a formula for A^{-1} can be written in terms of A_1, A_2, A_3, A_4^{-1} and $(A/A_4)^{-1}$ [15, pg. 47].

We applied this to our Sudoku setting by considering M as a block matrix,

$$M = \begin{bmatrix} M[K_{n,n,n}] & B \\ B^\top & nI \end{bmatrix}$$

where the blocks B, nI and $B^\top$ are the border of box-symbol edges in M depicted in Figure 2.13 from Section 2.4. We were then able to apply the Schur complement formula to decompose M as follows:

$$M = \begin{bmatrix} I & (\frac{1}{n})B \\ 0 & I \end{bmatrix} \quad \begin{bmatrix} M[K_{n,n,n}] - \frac{1}{n}BB^\top & 0 \\ 0 & nI \end{bmatrix} \quad \begin{bmatrix} I & 0 \\ (\frac{1}{n})B^\top & I \end{bmatrix}.$$

As we already had a deep understanding of $M[K_{n,n,n}]$ from the work presented in [5], this transition from M to $M/(nI)$ meant that the bulk of the new work for Sudoku could be done in terms of understanding the structure of $\frac{1}{n}BB^\top$ in terms of h and w. We also noted that reducing the size of the matrix in question from $4n^2 \times 4n^2$ to $3n^2 \times 3n^2$ made it significantly easier to examine cases of low-valued h and w visually.

Our main focus was on proving a general formula for constructing $BB^\top$ and identifying all of the eigenvectors of $M/(nI)$ in order to bound the infinity norm of each of the projectors onto the eigenspaces of $M/(nI)$. Although, in the end, this approach was not alluded to in our main result, it was due to this effort that we discovered clearly emerging patterns in our formulae and noted that they seemed to be arising from a subset of the relations discussed in Section 3.2.2.

Primarily, we noticed patterns in the structure of $BB^\top$ because the entry $BB^\top(e, f)$ appeared to depend on the relations between edges e and f. Once we had the eigenvectors of $M/(nI)$ constructed and were using them to build low-value examples of eigenprojectors, these same relations were proving useful in predicting the entries of the projector matrices. Once we had built a comprehensive set of such relations and had confirmed that they did indeed form a coherent configuration, this discovery rendered this more tedious approach unnecessary and allowed us to adopt the more theoretically elegant solution presented in the main text.

That said, we do acknowledge that the number of relations from Section 3.2.2 is quite large, and this pushed many of our estimates and calculations into computer-assisted territory. It is possible that the complexity of the algebra can be reduced. We conjecture that $M/(nI)$ lives in an algebra (over $\mathbb{Q}, \mathbb{R}$, or $\mathbb{C}$) of lower dimension than $\mathfrak{A}$. It may be possible to gain further insights into the Sudoku completion problem by continuing this line of inquiry.

6.3 Sudoku Variations

As mentioned in Section 1, (h, w)-Sudoku are a variation of an $n \times n$ Latin square. Even the notion of fractional completion acts as a variation on the original versions of these puzzles while simultaneously offering a potential approach to making advances on the original puzzle, like in the case of the Latin square completion conditions in [5]. There are many other variations on this type of puzzle game that may present intriguing research opportunities, either because they unlock avenues of exploration in the original problem or because they are interesting in their own right.

Many of these Sudoku variants are detailed comprehensively by Rosenhouse and Taalman in their book, [16, Chapter 10], so we refrain from going into too much detail. In this section we give a description of some of the Sudoku variants for which we believe our methods can be extended.

6.3.1 Free-form and α-approximate (h, w)-Sudoku

Suppose we generalize our setting so that each Sudoku box is an arbitrary subset of n cells. That is, the Sudoku conditions remain but the n-to-1 function box(i, j) mapping cells to boxes changes. This change can be relatively unrestricted as long as each box still contains exactly n cells, leading to what is known as a free-form Sudoku, as discussed by Alomari, Abudayah, and Sander in [1]. In free-form Sudoku, the boxes aren't required to be contiguous, but if the boxes do form a single continuous shape, then Sudoku is referred to as a jigsaw or polyomino Sudoku. An example of a polyomino Sudoku for $n = 9$ is depicted in Figure 6.3, as presented in [16, pg. 177].

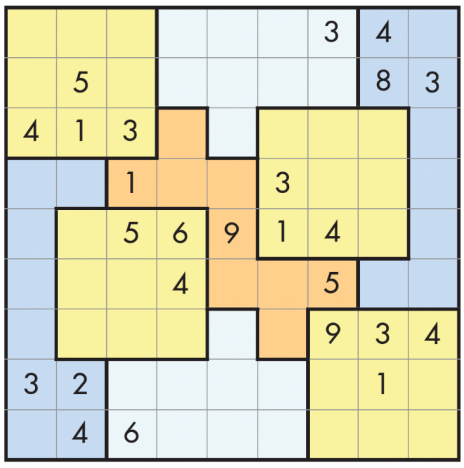


Figure 6.3: An illustration of a free-form Sudoku from [16, pg. 177]

This change to box shapes directly alters the relationships discussed in Section 3.2.1, particularly those dependent on bundles because non-rectangular boxes means

that row and column bundles are no longer well-defined in this setting. However, by restricting the change to box(i, j) so that the difference is sufficiently small, we were able to show that the perturbation methods we applied in Section 5.2 could also give a fractional completion guarantee for partial Sudoku of small density. To add some precision, we need a definition of what is meant by sufficiently small change.

Definition 6.3.1. We define an α-approximate (h, w)-Sudoku to be an Sudoku for which every box is a collection of $hw = n$ cells bounded between a $(1-\alpha)h \times (1-\alpha)w$ rectangle and a $(1+\alpha)h \times (1+\alpha)w$ rectangle.

The 0-approximate case coincides with our standard setting of rectangular boxes. Equipped with this definition, our paper [9] gives the following result:

Proposition 6.3.2. *There exist positive constants α and ϵ such that every ϵ-dense partial α-approximate (h, w)-Sudoku with large h and w has a fractional completion.*

We refer the reader to [9, Section 7] for the details of this proof.

Our approach for encoding Sudoku puzzles as graphs and applying algebraic graph theory is a novel methodology. However, there is an already existing method of encoding Sudoku puzzles as a graph G with n^2 vertices representing each of the cells and edges between vertices representing cells with shared rows, columns, or boxes. In this setting Sudoku completion is viewed as a proper edge-colouring problem, with a colour for each symbol. In [1], the authors delve into the eigenvalues and eigenvectors of the adjacency matrices for Sudoku graphs of this type. Their focus lies in comparing the spectral properties of the adjacency matrices for the (h, w)-Sudoku as compared to those of Sudoku with free-form boxes. Notably, they use the Kronecker product to construct $\{\pm 1, 0\}$-valued eigenvectors for both (h, w)-Sudoku and free-form Sudoku. For the sake of allowing comparison between our approaches and resulting eigenvectors, we have included an encoding of our matrix M and the eigenvectors of M from Chapter 4 as Kronecker-structured $\{\pm 1, 0\}$-valued eigenvectors in Appendix A. For readers interested in extending our methodologies to the case of free-form Sudoku, this may be especially valuable.

6.3.2 Simultaneous Sudoku

Another generalization of Sudoku which our methods could apply to is the allowance of simultaneous box patterns. This is natural because the row and column conditions in a Latin square can already be viewed as degenerate box conditions. An example for $n = 6$ with both 2×3 (in red) and 3×2 (in blue) boxes is given in Figure 6.4.

Using a generalized notion of tile and a suitably enlarged linear system, similar methods as in this thesis could, in principle, apply to the simultaneous Sudoku. Unlike the free-form Sudoku case, we do note that the equivalence relation $\sim$ from Section

3.2.2 would still induce coherent configuration relations, but *more* of them would be induced because they would need to be related to two separate sets of boxes. We foresee this adding an additional level of complexity which might make the resulting algebraic structure more difficult to analyze, requiring far more reliance on computational assistance.

1	2	3	4	5	6
4	5	6	1	2	3
3	6	2	5	1	4
5	1	4	3	6	2
2	3	1	6	4	5
6	4	5	2	3	1

Figure 6.4: A variation of a Sudoku puzzle with simultaneous box conditions.

6.4 Conclusion

In concluding this thesis, it is important to note that our main result specifically establishes conditions for *fractional* completion guarantees in the context of (h, w)-Sudoku and that we make no claims about *exact* completion guarantees. This contrasts with the case of partial Latin squares, where exact completion guarantees have been established. As explained in Section 1, Bowditch and Dukes' [5] results on fractional completion guarantees for partial Latin squares were combined with the absorber methods of Barber, Kühn, Lo, and Osthus [3], which successfully linked approximate K_r-decompositions of balanced and locally balanced r-partite graphs to their exact counterparts.

Such an approach to exact completion does not directly translate to the Sudoku via our encoding of the Sudoku graph because of two reasons. First, a tile, unlike its Latin square counterpart K_3, is not a true subgraph because not all instances of $K_3 + e$ are tiles. Secondly, although the graph G_{hw} is presented as a balanced 4-partite graph, it is not locally balanced because the vertices corresponding to boxes have no edges to row or column vertices. Arguably, G_{hw} is an imbalanced 3-partite graph because the box and row vertices (or column vertices) can be combined into an independent set.

This marks a clear direction for future research; the question of determining an exact completion threshold for Sudoku puzzles. We do not know whether a modified version of the methods used for Latin squares could be applicable in the Sudoku context.

Appendix A
Kronecker Products

In this appendix we give a description of the matrix M and the eigenvectors of M as Kronecker Products.

Definition A.0.1. The *Kronecker product*, denoted as $\otimes$, is a binary operation used to combine two matrices into a larger matrix [15, Section 10.2]. Given two matrices, A and B, $A \otimes B$ is defined as follows:

$$A \otimes B = \begin{bmatrix} a_{11}B & a_{12}B & \cdots & a_{1n}B \\ a_{21}B & a_{22}B & \cdots & a_{2n}B \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1}B & a_{m2}B & \cdots & a_{mn}B \end{bmatrix}.$$

If A is an $m \times n$ matrix and B is a $p \times q$ matrix then the result $A \otimes B$ is an $mp \times nq$ matrix.

We use the following standard property of Kronecker products [15, pg. 60], called the *mixed-product property*, in our analysis of the eigenvectors of M in Section A.2:

$$(A \otimes B)(C \otimes D) = AC \otimes BD.$$

A.1 Factoring M as a Kronecker Product

It is quite convenient to present M as a block matrix with a block partition corresponding to the four edge types, ordered as in Section 2.4. With this structure, each block is an $n^2 \times n^2$ matrix which can be factored as a Kronecker product.

We slightly abuse the Kronecker product in the following way. In forming $A \otimes B$, each factor is indexed by one of our four Sudoku objects: rows, columns, symbols, and boxes. The product is then indexed by corresponding pairs of elements. To this purpose, we define the following matrices and vectors, each with specified indexing:

- **All-Ones Vectors:** Let $\mathbf{j}_x$ for $x \in \{r, c, s, b\}$ be an $n \times 1$ all-ones vector, indexed as $\{x_1, x_2, \ldots, x_n\}$.

- **Indicator Vectors:** Let $\mathbf{e}_{x_i}$ for $x \in \{r, c, s, b\}$ be an $n \times 1$ vector, with elements indexed by $x_1, x_2, \ldots, x_n$. These vectors have the value 1 in position x_i and 0 elsewhere. We omit the second subscript and write e_x if i is unimportant or clear.
- **Identity Matrices:** Let I_x for $x \in \{r, c, s, b\}$ be an identity matrix, with both rows and columns indexed by $\{x_1, x_2, \ldots, x_n\}$.
- **All-Ones Matrices:** Let J_{xy} for $x, y \in \{r, c, s\}$ be an all-ones matrix, with rows indexed by $\{x_1, x_2, \ldots, x_n\}$ and columns indexed by $\{y_1, y_2, \ldots, y_n\}$.
- **Intersection Matrix:** Let the matrix H_{xb} for $x \in \{r, c\}$ be an $n \times n$ zero-one matrix, with rows indexed by $x_1, x_2, \ldots, x_n$ and columns indexed by $b_1, b_2, ..., b_n$ with $H_{xb}(x_i, b_\ell) = 1$ if and only if x_i intersects box b_ℓ and 0 otherwise.
- **Box Matrix:** Let the matrix H_{rcb} be an $n^2 \times n$ zero-one matrix, with rows indexed by row/column edges of the form $r_i c_j$ lexicographically and columns indexed by $b_1, b_2, ..., b_n$ with $H_{rcb}(r_i c_j, b_\ell) = 1$ if and only if $\text{box}(i, j) = \ell$ and 0 otherwise.

Using these, we are able to present the block structure of M in Figure A.1.

$nI_r \otimes I_c$	$I_r \otimes \mathbf{j}_c \otimes \mathbf{j}_s^\top$	$\mathbf{j}_r \otimes I_c \otimes \mathbf{j}_s^\top$	$H_{rcb} \otimes \mathbf{j}_s^\top$
$I_r \otimes \mathbf{j}_c^\top \otimes \mathbf{j}_s$	$nI_r \otimes I_s$	$\mathbf{j}_r \otimes \mathbf{j}_c^\top \otimes I_s$	$wH_{rb} \otimes I_s$
$\mathbf{j}_r^\top \otimes I_c \otimes \mathbf{j}_s$	$\mathbf{j}_r^\top \otimes \mathbf{j}_c \otimes I_s$	$nI_c \otimes I_s$	$hH_{cb} \otimes I_s$
$H_{rcb}^\top \otimes \mathbf{j}_s$	$wH_{rb}^\top \otimes I_s$	$hH_{cb}^\top \otimes I_s$	$nI_b \otimes I_s$

Figure A.1: Illustration of the block matrix structure of M as Kronecker products.

The Kronecker product $I_r \otimes I_c$ results in an $n^2 \times n^2$ identity matrix indexed by edges from G_P of the form $r_i c_j$ for $i, j \in [n]$, arranged in lexicographic order. This

ordering aligns with the edge order for the $(1,1)$-block of M. Recall that $M(e,f) = 0$ for edges $e = r_i c_j$ and $f = r_{i'} c_{j'}$ unless $i = i'$ and $j = j'$; otherwise, $M(e,f) = n$ because two distinct row-column edges cannot share a tile, and each edge in G_{hw} exists in precisely n tiles. Consequently, the $(1,1)$-block is represented as $n(I_r \otimes I_c)$. Similar constructions apply to blocks $(2,2)$, $(3,3)$ and $(4,4)$, as depicted in Figure A.1.

We can also illustrate that the $(1,2)$-block of M can be represented as $I_r \otimes J_{cs}$, where I_r is the identity matrix indexed by $r_1, \ldots, r_n$ and J_{cs} is the all-ones matrix whose rows are indexed by $c_1, \ldots, c_n$ and columns are indexed by $s_1, \ldots, s_n$. Notice that J_{cs} can be factored as $\mathbf{j}_c \otimes \mathbf{j}_s^\top$. We note that the resulting matrix from this Kronecker product has rows indexed by edges $e = r_i c_j$, columns indexed by edges $f = r_{i'} s_{k'}$, and the (e,f)-entry is 1 if and only if $i = i'$. This precisely corresponds to the condition for e and f sharing a common tile in the $(1,2)$-block. Additionally, due to the symmetry of M, the block $(2,1)$ is the transpose of the block $(1,2)$. Therefore, we can decompose the $(1,2)$-block of M into $(I_r \otimes J_{cs})^\top = I_r \otimes \mathbf{j}_c^\top \otimes \mathbf{j}_s$.

In a similar analysis, the $(1,3)$-block of M can be represented as $\mathbf{j}_r \otimes I_c \otimes \mathbf{j}_s^\top$, because the resulting matrix from this Kronecker product has rows indexed by edges $e = r_i c_j$, columns indexed by edges $f = c_{j'} s_{k'}$, and the (e,f)-entry is 1 if and only if $j = j'$. Again, this exactly corresponds to the condition for e and f sharing a common tile in the $(1,3)$-block and, as before, the $(3,1)$-block is, by the symmetry of M the transpose of block $(1,3)$, and is therefore $\mathbf{j}_r^\top \otimes I_c \otimes \mathbf{j}_s$.

The $(1,4)$-block of M_P has rows indexed by edges $e = r_i c_j$ and columns indexed by edges $f = b_\ell s_k$. The (e,f)-entry is 1 if and only if $\mathrm{box}(i,j) = \ell$. It can therefore be written as the Kronecker product $H_{rcb} \otimes \mathbf{j}_s^\top$. The $(4,1)$-block is the transpose, $(H_{rcb} \otimes \mathbf{j}_s^\top)^\top = H_{rcb}^\top \otimes \mathbf{j}_s$.

The $(2,3)$-block of M can be represented as $\mathbf{j}_r \otimes \mathbf{j}_c^\top \otimes I_s$, because the resulting matrix from this Kronecker product has rows indexed by edges $e = r_i s_k$, columns indexed by edges $f = c_{j'} s_{k'}$, and the (e,f)-entry is 1 if and only if $k = k'$. This exactly corresponds to the condition for e and f sharing a common tile in the $(2,3)$-block and, as before, the $(3,2)$-block is, by the symmetry of M the transpose of block $(1,3)$, and is therefore $\mathbf{j}_r^\top \otimes \mathbf{j}_c \otimes I_s$.

Finally, the $(2,4)$-block of M_P has rows indexed by edges $e = r_i s_k$ and columns indexed by edges $f = b_\ell s_{k'}$. The (e,f)-entry is w if and only if $k = k'$, because there are exactly w columns in b_ℓ with which it is possible to form a tile using r_i, s_k, and b_ℓ. It can therefore be written as the Kronecker product $wH_{rb} \otimes \mathbf{I}_s$. The $(4,2)$-block is the transpose, $(wH_{rb} \otimes I_s)^\top = wH_{rb}^\top \otimes I_s$.

A.2 Eigenvectors as Kronecker products

As with our matrix M, it is possible to write the eigenvectors of M using the $\otimes$ operation.

Recall the vectors $\mathbf{j}_x$ and $\mathbf{e}_x$ for $x \in \{r, c, s, b\}$ as defined in Section A.1. Any difference $\mathbf{e}_{x_i} - \mathbf{e}_{x_{i'}}$ where $i \neq i'$ is denoted $\mathbf{d}_x$. We also need a few minor variants of these. A difference $\mathbf{e}_{r_i} - \mathbf{e}_{r_{i'}}$ where $r_i \ncong r_{i'}$ is denoted $\mathbf{d}_r^{\sim}$; an analogous vector $\mathbf{d}_c^{\sim}$ is defined for columns in the same bundle. If $\ell_{00}, \ell_{01}, \ell_{10}, \ell_{11}$ are box indices forming a rectangle, the alternating sum of their indicator vectors is denoted $\mathbf{d}_b^{\square}$. Given two distinct row bundles, say B_0 and B_1, a combination,

$$\sum_{r \in B_0} \mathbf{e}_r - \sum_{r \in B_1} \mathbf{e}_r,$$

is denoted $\mathbf{d}_r^{\dagger}$. Similar vectors can be defined for columns and boxes. We remark that all vectors involving the letter '$\mathbf{d}$' have a sum of coordinates equal to zero.

As an important convention, when we take a Kronecker product of vectors from the above, it is regarded as a $4n^2 \times 1$ vector with zeros inserted as needed. For instance, $\mathbf{j}_r \otimes \mathbf{j}_c$ has entries equal to 1 supported on all row-column edges, and is assumed to vanish on edges of all other types (even though these do not appear in the product). Additionally, choices are assumed to be canonical and consistent within a vector. For instance, if $\mathbf{d}_s$ appears twice, it is assumed to represent the same symbol indices; if $\mathbf{d}_r^{\dagger}$ and $\mathbf{d}_b^{\dagger}$ appear together, it is assumed the row bundle and box bundle coincide.

Eigenvalue	Type	Eigenvector	
0	A_0	$\mathbf{e}_r \otimes \mathbf{j}_c - \mathbf{e}_r \otimes \mathbf{j}_s$	$*$
	B_0	$H_{rcb}\mathbf{e}_b - \mathbf{j} \otimes \mathbf{e}_b$	
	C_0	$\mathbf{d}_r^{\dagger} \otimes \mathbf{e}_s - \mathbf{d}_b^{\dagger} \otimes \mathbf{e}_s$	$*$
n	A_1	$\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c$	$*$
	B_1	$\mathbf{d}_r^{\sim} \otimes \mathbf{d}_s$	$*$
	C_1	$\mathbf{d}_s \otimes \mathbf{d}_b^{\square}$	
$2n$	A_2	$\mathbf{d}_r^{\sim} \otimes \mathbf{j}_c + \mathbf{d}_r^{\sim} \otimes \mathbf{j}_s$	$*$
	B_2	$\mathbf{d}_r^{\dagger} \otimes \mathbf{d}_s + \mathbf{d}_b^{\dagger} \otimes \mathbf{d}_s$	$*$
	C_2	$\mathbf{j}_s \otimes \mathbf{d}_b^{\square} + H_{rcb}\mathbf{d}_b^{\square}$	
$3n$	A_3	$\mathbf{d}_s \otimes \mathbf{j}_r + \mathbf{d}_s \otimes \mathbf{j}_c + \mathbf{d}_s \otimes \mathbf{j}_b$	
	B_3	$\mathbf{d}_r^{\dagger} \otimes \mathbf{j}_c + \mathbf{d}_r^{\dagger} \otimes \mathbf{j}_s + \mathbf{d}_b^{\dagger} \otimes \mathbf{j}_s$	$*$
$4n$		$\mathbf{j}_r \otimes \mathbf{j}_c + \mathbf{j}_r \otimes \mathbf{j}_s + \mathbf{j}_c \otimes \mathbf{j}_s + \mathbf{j}_s \otimes \mathbf{j}_b$	

Table A.1: Eigenvectors of M; $*$ denotes row/column variants

Although it does not convey much combinatorial understanding, eigenvectors in this form can be checked using block matrix multiplication. As an example, consider $\mathbf{v} = \mathbf{d}_r^{\sim} \otimes \mathbf{d}_c$, a Type A_1 vector. The four blocks of the product $M\mathbf{v}$ can be computed

one at a time by applying the mixed-product property:

$$\begin{aligned}
(nI_r \otimes I_c)(\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c) &= n\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c \\
(I_r \otimes J_{sc})(\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c) &= \mathbf{d}_r^{\sim} \otimes \mathbf{0} = \mathbf{0} \\
(J_{sr} \otimes I_c)(\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c) &= \mathbf{0} \otimes \mathbf{d}_c^{\sim} = \mathbf{0} \\
(H_{rcb}^{\top} \otimes \mathbf{j}_s)(\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c) &= H_{rcb}^{\top}(\mathbf{d}_r^{\sim} \otimes \mathbf{d}_c) \otimes \mathbf{j}_s = \mathbf{0} \otimes \mathbf{j}_s = \mathbf{0},
\end{aligned}$$

where the latter is because every box intersects both or neither of the row indices defining $\mathbf{d}_r^{\sim}$. It follows that $M\mathbf{v} = n\mathbf{v}$.

Appendix B
Mission: Math Outreach

This thesis would not have been possible without an excess of support from the math education community. Teachers, professors, study groups, and outreach opportunities; they all made a difference in this arduous (but also fun and worthwhile) journey. Therefore, this section is designed to be a first step in giving back to the math education community with the goal of creating opportunities for the next generation of young people to discover that math is *fun*.

Here we share a simple lesson plan designed to introduce students to the principles behind Latin squares. Of course, these principles are simple enough that they could be explained directly to many students. However, the main goal with this lesson plan is to encourage young people to engage with the creative and playful sides of math.

As a fun way to promote deeper engagement, the exercises presented are framed as a **set of missions for secret agents**. The idea is that these activities will be more fun when we treat the the problems like they are a game: getting instructions over an earpiece, cracking codes, and saving the world. The activities range from understanding grid coordinates to solving small Latin square puzzles and, finally, a simple Sudoku! Each activity is accompanied by a clear set of objectives, materials required, and step-by-step instructions.

Many of the exercises use the numbers 1 to 9. However, if the children who are going through the exercises are working on learning goals that involve recognizing new symbols, there is opportunity to use those symbols instead. For example, letters from a particular alphabet or the symbols for numbers in another language could be used. Note that we do not include a recommended age group, as the exercises written are just a starting point and we believe that they could be modified to be engaging to a wide variety of children.

Operation 1: Covert Landing Training Exercise

Agent Gear:

- A large grid (we suggest starting with a 3×3 grid which can be drawn on a classroom board or on the sidewalk with chalk - a 5×5 grid can be used if this becomes too easy for the group.).
- Smaller grid(s) (these can be drawn on paper).
- Stickers or sticky notes.
- Markers or chalk for drawing grids.

Set-Up at Headquarters (teacher instructions)

- Create the large grid (using the board or sidewalk), which should be big enough to be seen by the whole group of students. This grid represents the *landing zone*.
- Select one agent (student) to be the Field Operative. The Field Operative must face away from the grid so that they cannot see it. This agent will be given a grid (with the same dimensions as the large grid) on paper - this is their mission map.
- Place a marker (a drawn 'X', sticker, or sticky note) in one of the cells on the large visible grid. This marks the critical landing zone for the Field Operative.

Reminder: The objective of the game is to learn to describe the grid in terms of row-column coordinates. However, it is okay if the students try other techniques first! This is a training mission, so you can give them advice or guidance towards this goal as needed. This mission can be repeated several times with new Field Operatives, as long as the game remains fun!

Mission Description:

1. Agents, our special Field Operative has lost visuals of the map and is flying blind! We will need to talk them through their landing if we want their mission to be a success!
2. Agents must take turns giving instructions to the Field Operative, guiding them to place their marker on the corresponding cell in their mission map, aiming for a precise landing.
3. The mission is successful when the Field Operative lands (places their marker or sticker) on the targeted cell in their mission map.

Operation 2: Eye Scan Entry

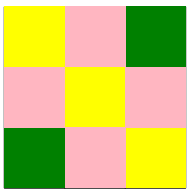
Agent Gear:

- Eye scan grid paper (a print-out of a 3×3 grid)
- the eye color instructions (for added secret agent fun, the instructions can come out of an envelope).
- Grid Tools (markers or crayons for colouring in the grids).

Mission Description:

1. You have successfully landed in the enemy base and infiltrated their hideout. However, your command center has received intel that the secret code we need to crack is locked behind an eye-scanner. Luckily, you have been given the necessary technology that can help us imitate the enemy leader's eyes. You have each been given a 3×3 grid blueprint. If you fill in the grid with the correct colours, you can imitate the enemy leader's eyes and scan the grid to gain access to the secrets!
2. Here are your mission instructions:
   - Color the following cells green:
     - Row 1, Column 3
     - Row 3, Column 1
   - Color the following cells yellow:
     - Row 1, Column 1
     - Row 2, Column 2
     - Row 3, Column 3
   - Color the following cells pink:
     - Row 1, Column 2
     - Row 2, Column 1
     - Row 2, Column 3
     - Row 3, Column 2

Answer Key:



Operation 3: Number Cipher

Agent Gear:

- Classified code document (a print-out of the rows of numbers from below).

Mission Description:

1. Junior agents, on your last mission you were able to retrieve a code from the enemy base. Your next mission is to decipher the code and reveal the lock combination that must be used to receive your final piece of key intel. The code consists of exactly eight rows of numbers. Each row is *supposed* to contain every number from 1 to 9, without repetition.

2. To complete your mission you must identify which number is missing from each row and add it to the end of that row. Order these numbers from the top of the page to the bottom. This sequence of numbers will form the first eight entries of the secret code.

3. The final piece of the code is the only number from 1 to 9 that hasn't been used in your sequence. Finding this number will complete the code and let you unlock the secrets necessary for the next stage of your mission.

1	3	5	9	7	2	6	4	
6	2	8	3	9	4	1	7	
9	8	4	1	2	7	5	6	
1	5	7	9	4	6	8	3	
2	7	5	1	8	4	9	3	
9	6	5	1	4	3	2	8	
6	3	4	2	5	8	7	9	
6	3	4	2	5	8	7	9	
8	1	2	7	5	4	3	6	

Operation 4: Secret Message

Agent Gear:

- Classified Code Document (a print-out of all following Puzzles and instructions).

Mission Description:

Junior agents, you have successfully received and delivered the information we needed to our base. Great job! Your final mission is to decipher a hidden message from our leader! Command Center has relayed the message and made it clear it is vital that this message be for *your eyes only*. To complete your mission and uncover the message, you must solve all five of the letter puzzles.

Puzzle 1:

To solve this puzzle, use the letters {O,U,Y} as a set of symbols. You must place the symbols so that they appear *once and only once* in each row and column of the puzzle. The yellow highlighted word is the first word of the code.

	O	U
	U	Y
	Y	O

Puzzle 2:

To solve this puzzle, use the letters {R,A,F,E} as a set of symbols. You must place the symbols so that they appear *once and only once* in each row and column of the puzzle. The yellow highlighted word is the second word of the code.

F		R	E
	E	F	R
	F	E	
	R		

Puzzle 3:

To solve this puzzle, use the letters {Y,K,M} as a set of symbols. You must place the symbols so that they appear *once and only once* in each row and column of the puzzle. The yellow highlighted word is the third word of the code.

		K
K		M

Puzzle 4:

To solve this puzzle, use the letters {S,B,T,A,E} as a set of symbols. You must place the symbols so that they appear *once and only once* in each row and column of the puzzle. The yellow highlighted word is the fourth word of the code.

	B		E	S
	T	S	A	
	S	B	T	A
	E	A		
	A		S	

Puzzle 5:

To solve this final puzzle, use the letters {G,T,N,H,A,E} as a set of symbols. You must place the symbols so that they appear *once and only once* in each row and column of the puzzle. You will notice that the puzzle grid is divided into 6 rectangular boxes. As an *additional challenge* you must also make sure that each symbol appears *once and only once* in each box. The yellow highlighted word is the final word of the code. **Hint: There is a box that is only missing the letter 'T'! If you are stuck, start there!**

	H		N		G
		T	E	A	H
			A		
	A	G	H		T
	E	N		H	A
		A			E

Bibliography

[1] O. Alomari, M. Abudayah, and T. Sander. Integral free-form Sudoku graphs. *Electronic notes in discrete mathematics*, 68:47–52, 2018.

[2] B. Barber, D. Kühn, A. Lo, and D. Osthus. Edge-decompositions of graphs with high minimum degree. *Adv. Math.*, 288:337–385, 2016.

[3] B. Barber, D. Kühn, A. Lo, D. Osthus, and A. Taylor. Clique decompositions of multipartite graphs and completion of latin squares. *J. Combin. Theory Ser. A*, 151:146–201, 2017.

[4] P. Bartlett. Completions of ϵ-dense partial Latin squares. *J. Combin. Des.*, 21:447–463, 2013.

[5] F.C. Bowditch and P.J. Dukes. Fractional triangle decompositions of dense 3-partite graphs. *J. Comb.*, 10(2):255–282, 2019.

[6] A.G. Chetwynd and R. Häggkvist. Completing partial $n \times n$ Latin squares where each row, column and symbol is used at most cn times. *Reports, Dept. of Mathematics, University of Stockholm*, 1985.

[7] M. Delcourt and L. Postle. Progress towards Nash-Williams' conjecture on triangle decompositions. *J. Combin. Theory Ser. B*, 146:382–416, 2021.

[8] P.J. Dukes and D. Horsley. On the minimum degree required for a triangle decomposition. *SIAM J. Discrete Math.*, 34:597–610, 2020.

[9] P.J. Dukes and K.I. Nimegeers. The linear system for Sudoku and a fractional completion threshold, 2023. Available at https://arxiv.org/abs/2310.15279.

[10] T. Evans. Embedding incomplete Latin rectangles. *Amer. Math. Monthly*, 67:958–961, 1960.

[11] T. Gustavsson. *Decompositions of large graphs and digraphs with high minimum degree*. PhD thesis, Doctoral Dissertation, Department of Mathematics, Stockholm University, 1991.

[12] D.G. Higman. Coherent configurations. I. *Rend. Sem. Mat. Univ. Padova*, 44:1–25, 1970.

[13] R.A. Horn and C.R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 2nd edition, 2013.

[14] C. Nash-Williams. An unsolved problem concerning decomposition of graphs into triangles. In *Combinatorial Theory and its Applications III*, pages 1179–1183. North Holland, 1970.

[15] K.B. Petersen and M.S Pedersen. The Matrix Cookbook, 2012. Accessed: 2022-07-14.

[16] J. Rosenhouse and L. Taalman. *Taking Sudoku Seriously: The Math Behind the World's Most Popular Pencil Puzzle*. Oxford University Press, Incorporated, New York, United States, 2012.

[17] B. Smetaniuk. A new construction on Latin squares - I: A proof of the Evans conjecture. *Ars Combinatoria*, (9):155–72, 1981.