Final Year Project

# Finding the Minimum Number of Sudoku Clues Through Information Theory

William Ikenna-Nwosu

Student ID: 16303711

A final report submitted in part fulfilment of the degree of

**BSc. (Hons.) in Computer Science**

**Supervisor:** Félix Balado Pumariño

UCD School of Computer Science
University College Dublin

May 11, 2020

# Table of Contents

# List of Figures

# List of Tables

# Glossary

$n$ Size of subgrid.

$N = n^2$ Size of grid.

$n^2 \times n^2 = N \times N$ Number of cells in grid.

$N_s$ Number of Sudoku grids.

**Band** Set of horizontally adjacent subgrids in a Sudoku.

**Cell** A square in a Sudoku grid that can contain a single symbol.

**Channel Capacity (C)** Channel capacity is the tight upper bound on the rate at which information can be reliably transmitted over a communication channel. Following the terms of the noisy-channel coding theorem, the channel capacity of a given channel is the highest information rate (in units of information per unit time) that can be achieved with arbitrarily small error probability. Claude E. Shannon in 1948, defines the notion of channel capacity and provides a mathematical model by which one can compute it. The key result states that the capacity of the channel, as defined above, is given by the maximum of the mutual information between the input and output of the channel, where the maximization is with respect to the input distribution..

**Clues** Initially filled cells in a Sudoku grid.

**Code** A code is a system of rules to convert information—such as a letter, word, sound, image, or gesture—into another form or representation, sometimes shortened or secret, for communication through a communication channel or storage in a storage medium. In information theory and computer science, a code is usually considered as an algorithm that uniquely represents symbols from some source alphabet, by encoded strings, which may be in some other target alphabet. An extension of the code for representing sequences of symbols over the source alphabet is obtained by concatenating the encoded strings..

**Codeword** Block of bits that the encoder is allowed to transmit over the channel e.g. in the context of this project, codewords which represent proper Sudoku after they are encoded by the encoder which will then be sent over the erasure channel.

**Complete Grid** Sudoku grid with all cells filled in that satisfies all rules of Sudoku.

**Error-Correcting Code (ECC)** Codes may also be used to represent data in a way more resistant to errors in transmission or storage. This so-called error-correcting code works by including carefully crafted redundancy with the stored (or transmitted) data. Examples include Hamming codes, Reed–Solomon, Reed–Muller, Walsh–Hadamard, Bose–Chaudhuri–Hochquenghem, Turbo, Golay, Goppa, low-density parity-check codes, and space–time codes. In computing, telecommunication, information theory, and coding theory, error-correcting codes are used for controlling errors in data over unreliable or noisy communication channels. The central idea is the sender encodes the message with redundant information in the form of an ECC. The redundancy allows the receiver to detect a limited number of errors that may occur anywhere in the message, and often to correct these errors without retransmission. The American mathematician Richard Hamming pioneered this field in the 1940s and invented the first error-correcting code in 1950: the Hamming (7,4) code. ECC contrasts with error detection in that errors that are encountered can be corrected, not simply detected. The advantage is that a system using ECC does not require a reverse channel to request retransmission of data when an error occurs. The downside is that there is a fixed overhead that is added to the message, thereby requiring a higher forward channel bandwidth. ECC is therefore applied in situations where retransmissions are costly or impossible, such as one-way communication links and when transmitting to multiple receivers in multicast..

**Erasure Channel** An erasure channel is a communication channel model wherein errors are described as erasures.

**Grid** Square matrix formed by $N$ rows, $N$ columns and $N$ subgrids, each having $N$ cells ($N \times N$ cells total).

**Information Theory** Information Theory studies the quantification, storage, transformation, communication, transmission, processing, extraction, utilization and coding of information. It is interdisciplinary; closely associated with with a collection of pure and applied disciplines including mathematics, statistics, physics, computer science, neurobiology, information engineering and electrical engineering. Abstractly, information can be thought of as the resolution of uncertainty. It was originally proposed by Claude Shannon in 1948 to find fundamental limits on signal processing and communication operations such as data compression, in a landmark paper titled "A Mathematical Theory of Communication".

**Latin Square** A Latin square of order $n$ is then an $n \times n$ matrix in which every row and column contains each of the $n$ symbols exactly once. We refer to the number n as the order of the Latin square.

**Law of Large Numbers (LLN)** Law of Large Numbers (LLN) is a theorem that describes the result of performing the same random experiment a large number of times. According to the law, the average of the results obtained from a large number of trials should be close to the expected value, and will tend to become closer to the expected value as more trials are performed. The LLN is important because it guarantees stable long-term results for the averages of some random variables.

**Magic Square** A magic square is an integer-valued matrix in which all the rows, columns and diagonals add up to the same sum.

**Minimum Sudoku Clue Problem (MSCP)** What is the smallest number of clues that can be given such that a sudoku puzzle has a unique completion?.

**Noisy-Channel Coding Theorem** The noisy-channel coding theorem (sometimes Shannon's theorem or Shannon's limit), establishes that for any given degree of noise contamination of a communication channel, it is possible to communicate discrete data (digital information) nearly error-free up to a computable maximum rate through the channel. The Shannon limit or Shannon capacity of a communication channel refers to the maximum rate of error-free data that can theoretically be transferred over the channel if the link is subject to random data transmission errors, for a particular noise level. The theorem states that given a noisy channel with channel capacity C and information transmitted at a rate R, then if R < C there exist codes that allow the probability of error at the receiver to be made arbitrarily small. This means that, theoretically, it is possible to transmit information nearly without error at any rate below a limiting rate, C. The converse is also important. If R > C, an arbitrarily small probability of error is not achievable. All codes will have a probability of error greater than a certain positive minimal level, and this level increases as the rate increases. So, information cannot be guaranteed to be transmitted reliably across a channel at rates beyond the channel capacity. No errorless communication is possible at R = C.

**Partial Grid** Sudoku grid with some cells filled in that satisfies all rules of Sudoku.

**Rate (R)** The rate is the average amount of information being transmitted over the channel every time the channel is used, i.e., average amount of information, in the original message that each of the symbols in the encoded message is responsible for representing, i.e., amount of information in bits each chosen symbol represents, i.e., average entropy per symbol. Measured in bits/channel use.

**Redundancy** Amount of redundant information in bits contained in the structure of the message. This extra information contained in the message is in fact redundant in the sense that, if it were missing the message would still be essentially complete, or at least could be completed. Measured in bits.

**Shidoku** 4 x 4 Sudoku.

**Stack** Set of vertically adjacent subgrids in a Sudoku.

**Subgrid** Inner $n \times n$ subgrids contained in $N \times N$ grid.

**Sudoku** A Sudoku puzzle is a 9 x 9 partial grid, some of whose cells already contain a digit between 1 and 9. The task is to complete the grid by filling in the remaining cells such that each row, column and subgrid contains each of the digits from 1 to 9 exactly once.

**Unique Grid** Sudoku puzzle that has only one completion or solution.

**Zero-Error Capacity of a Noisy Channel ($C_0$)** The zero error capacity $C_0$ of a noisy channel is defined as the least upper bound of rates at which it is possible to transmit information with zero probability of error.

# Abstract

Sudoku is a puzzle game where the player is presented with a 9 x 9 square grid called a Sudoku grid such that every row, column, and 3 x 3 square subgrid contain a permutation of the numbers 1 to 9. In a Sudoku puzzle, a few squares out of the 81 forming a chosen Sudoku grid are given as clues to the player, who has to guess the content of the remaining ones. In 2013, McGuire et al. found through a mathematically informed empirical search that 17 is the minimum number of clues for a Sudoku puzzle to be uniquely solvable. However there is no formula yet that tells us why this should be so. This research project aims to investigate the Minimum Number of Clues for a Sudoku puzzle to be uniquely solvable from a theoretical perspective; using the tools of Information Theory to model Sudoku as an erasure channel in an effort to produce theoretical results that can give insight into why exactly it is that 17 is the minimum number of clues for a Sudoku puzzle to be uniquely solvable.

**Keywords:** Sudoku, Latin Square, Information Theory, Erasure Communication Channel, Law of Large Numbers (LLN), Asymptotic Equipartition Property (AEP), Zero-Error Capacity of a Noisy Channel, Error-Free Coding, Noisy-Channel Coding Theorem, Puzzle, Minimum Sudoku Clue Problem, Coding

**New Chapters:**

- Data & Context
- Core Contribution
- Evaluation

**Modifications:**

- Added Appendices
- Added List of Tables
- Added List of Figures
- Added Glossary

# Chapter 1: **Introduction**

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | | | 8 | | 1 | | | |
| | | | | | | | 4 | 3 |
| 5 | | | | | | | | |
| | | | | 7 | | 8 | | |
| | | | | | | 1 | | |
| | 2 | | | 3 | | | | |
| 6 | | | | | | | 7 | 5 |
| | | 3 | 4 | | | | | |
| | | | 2 | | | 6 | | |

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

Figure 1.1: A partially filled 9 x 9 Sudoku with 17 clues and its solution (completely filled) side by side [1]

**Minimum Sudoku Clue Problem (MSCP):** What is the smallest number of clues that can be given such that a sudoku puzzle has a unique completion? i.e. What is the minimum number of clues a Sudoku of size $N \times N$ can have whilst still having a unique solution?

In the context of this research project, we are trying to answer the above question mathematically. We currently know the answer to this question is 17 thanks to the efforts of all who came before and McGuire et al.'s exhaustive computer search [2]. However, we don't yet have a formula that tells us why? In the remainder of this report, we discuss and go into detail on how the question has been answered or how the problem has developed depending on how you want to look at it and eventually describe how we approached the Minimum Sudoku Clue Problem using the tools of Information Theory.

# Chapter 2: Project Specification

**Core:**

Firstly, the student will become acquainted with Sudoku and its most relevant literature. A Sudoku grid can be seen as a 9-ary code where only certain codewords are allowed. Thus a Sudoku puzzle can be seen as putting a Sudoku grid (codeword) through a 9-ary erasure communications channel. The task of the player is to decode the original codeword (i.e. the original grid) from the noisy output of the channel. Since it is known how many Sudoku grids exist (i.e., the number of Sudoku grids has been enumerated), the rate of the code is also known. For this reason, information theory should be able to allow us to investigate the minimum number of clues in an analytical way.

With the help of the advisor, the student will study the problem from the point of view of standard information theory, where the uses of the channel go asymptotically to infinity.

**Advanced:**

To refine the results above it will be necessary to resort to zero-error capacity in the erasure channel, over a finite number of channel uses (i.e., 81 uses).

# Chapter 3: **Groundwork**

Some initial groundwork was performed to gain more context on the background of Sudoku and the Minimum Sudoku Clue Problem in an attempt to understand the problem at its core and fully appreciate the richness and variety of Sudoku's history and the progress made on the Minimum Sudoku Clue Problem to date.

## 3.1 Background Research

In this section, we investigate and trace the history of Sudoku, including its predecessors. Sudoku's predecessors are actually quite well known, documented, studied and are still relatively of interest to people in modern times. It is particularly interesting to explore them not only for their own historical merit but also because they may give insight into the underlying workings of Sudoku and provide some much needed and elusive generality in the combinatorics of Sudoku. We will also go into some detail on Sudoku's mathematical foundations to provide a good theoretical understanding of Sudoku's structure and its mathematical relation to its predecessors.

### 3.1.1 History of Sudoku

Below, is a timeline of Sudoku's history. Starting from the earliest known inception of Sudoku's predecessors to the best of the author's knowledge and continuing to present times.

- **Sudoku Predecessors (190 BC – 1782)**
  - **Magic Squares (190 BC):** A magic square is a matrix in which the integers in all the rows, columns and diagonals add up to the same sum. The third order magic square was known to Chinese mathematicians as early as 190 BC [3].
  - **Latin Squares (1700 – 1782):** A Latin square of order $n$ is a matrix whose rows and columns contain the numbers 1 to $n$ exactly once. The Korean mathematician Choi Seok-jeong was the first to publish an example of Latin squares of order nine, in order to construct a magic square in 1700 in his monograoh Koo-Soo-Ryak, predating Leonhard Euler by 67 years [4]. It is unlikely that this was the first appearance of Latin squares. Ahrens remarks that Latin square amulets go back to medieval Islam (c1200), and a magic square of al-Buni, c 1200, indicates knowledge of two $4 \times 4$ orthogonal latin squares. In 1723, a new edition of Ozanam's four-volume treatise [1712] presented a card puzzle that is equivalent to finding two orthogonal latin squares of order 4. Then in 1776, Euler presented a paper (De Quadratis Magicis) to the Academy of Sciences in St. Petersburg in which he again constructed magic squares of orders 3, 4, and 5 from orthogonal latin squares. He posed the question for order 6, now known as Euler's 36 Officers Problem. Euler was unable to find a solution and wrote a more extensive paper [798] in 1779/1782. He conjectured that no solution exists for order 6. Indeed he conjectured further that there exist orthogonal latin squares of all orders n except when n $\equiv$ 2 (mod 4): [5]. The definition of a Latin square is not too far from the definition of a Sudoku grid. In fact, every completed Sudoku grid is a Latin square but

not every Latin square is a Sudoku grid. Albeit, more on this in the next section 3.1.2, where the connection between the two will be explained.

- **First Appearance of Number Puzzles in Newspapers (1800s):** Number puzzles appeared in newspapers in the late 19th century, when French puzzle setters began experimenting with removing numbers from magic squares. Le Siècle, a Paris daily, published a partially completed 9x9 magic square with $3 \times 3$ subsquares on November 19, 1892. It was not a Sudoku because it contained double-digit numbers and required arithmetic rather than logic to solve, but it shared key characteristics: each row, column and subsquare added up to the same number [6].

- **Refined Puzzle from 'LaFrance' NewsPaper which Closely Resembled Modern Sudoku (1895):** On July 6, 1895, Le Siècle's rival, La France, refined the puzzle so that it was almost a modern Sudoku. It simplified the $9 \times 9$ magic square puzzle so that each row, column, and broken diagonals contained only the numbers 1 - 9, but did not mark the subsquares. Although they are unmarked, each $3 \times 3$ subsquare does indeed comprise the numbers $1 - 9$ and the additional constraint on the broken diagonals leads to only one solution. These weekly puzzles were a feature of French newspapers such as L'Echo de Paris for about a decade, but disappeared about the time of World War I [6].

- **Number Place (1979):** A standard Sudoku is like an order-9 Latin square, differing only in its added requirement that each subgrid contain the numbers 1 through 9. The first Sudoku puzzle appeared in the May 1979 edition of Dell Pencil Puzzles and Word Games and, according to research done by Will Shortz, the crossword editor of the New York Times, was apparently created by a retired architect named Howard Garns. Garns died in Indianapolis in 1989 (or 1981; accounts vary), too early to witness the global success of his invention. The game, published by Dell as "Number Place" not "Sudoku" actually [7].

- **Number Place in Japan, now given name Sudoku (1984):** Number Place jumped to a magazine in Japan in 1984, which ultimately named it "Sudoku," loosely translated as "single numbers." The magazine trademarked that moniker, and so copycats in Japan used the "Number Place" name. In yet another Sudoku-related irony, then, the Japanese call the puzzle by its English name, and English speakers call it by its Japanese name [7].

- **First Sudoku Generator and Modern Sudoku (1997 – Present):** Sudoku owes its subsequent success to Wayne Gould, a peripatetic retired judge living in Hong Kong, who came across it while visiting Japan in 1997 and wrote a computer program that automatically generates Sudoku grids. At the end of 2004 the London Times accepted his proposal to publish the puzzles, and in January 2005 the Daily Telegraph followed suit. Since then, several dozen daily papers in countries all over the world have taken to printing the game, some even putting it on the cover page as a promotional come-on. Specialty magazines and entire books devoted to this diversion have sprung up, as have tournaments, Web sites and blogs [7].

As you can see, the history of Sudoku is long and varied; dating back to BC times and extending forward to modern times with Sudoku as we know it to be. The one constant being that Sudoku has managed to maintain the interest of humans worldwide for generations. Both those studying it to advanced current knowledge of the game and those engaged with it for recreation and leisure to improve logical problem-solving or pass time. Try to hold onto this interest as we now gently go into the mathematics underpinning the game of Sudoku.

## 3.1.2 Mathematical Foundations of Sudoku

### Latin Squares

Indeed, aside from the numbers and the grid, Sudoku has almost nothing to do with the magic square but everything to do with the Latin square [7]. A Sudoku grid is a special kind of Latin square; a Latin square with an extra set of constraints. The mathematical connection between the two, will be further explained in the following paragraphs.

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 1 | | | | 5 | | | 8 | |
| | 3 | | | | | | | |
| | | | | 7 | | 9 | | 2 |
| | | 6 | | | | | | |
| 5 | | | | | | | | |
| | | 8 | | 1 | | 3 | | 5 |
| | | | 1 | | | | | |
| | 9 | | | | | | | 7 |
| | | | | | 5 | | | |

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

Figure 3.1: A partially filled $9 \times 9$ Latin square and its solution (completely filled) side by side

The name originates from the use of Latin letters, instead of digits when they were first being studied. Let us give a more precise definition. Imagine that you have a collection of $n$ distinct symbols. The first $n$ letters of the alphabet perhaps, or the first $n$ positive whole numbers. A Latin square of order $n$ is then an $n \times n$ array in which every row and column contains each of the $n$ symbols exactly once. We refer to the number $n$ as the order of the Latin square [8].

**More Mathematical Definition of Latin Square:** A Latin square of order $n$ is an $n \times n$ matrix, each row and column of which is a permutation of the set of letters $\mathcal{L} = \{1, 2, ..., n\}$ [9] i.e. the same symbol never appears more than once in the same row or column.

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

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

Figure 3.2: A partially filled 9 x 9 Sudoku and its solution (completely filled) side by side

**Additional Sudoku constraint:** A Sudoku square is then seen to be a Latin square of order 9 with an extra condition regarding the $3 \times 3$ blocks [8];The standard completed Sudoku grid is a $9 \times 9$ Latin square that meets the additional constraint of having each of its nine subgrids contain the digits 1 to 9. [7]

This additional constraint is in contrast to a $9 \times 9$ Latin square where it is not necessary for the $3 \times 3$ subgrids to contain one of each of the numbers from 1 – 9. See above figure 3.1

Therefore, every solution to a Sudoku is a solution to a Latin square but not every solution to a Latin square is a solution to a Sudoku. The implication only goes in one direction because Latin squares do not have the extra constraint which is imposed on Sudoku. This can be seen in the above figure 3.1 which displays a solution (completely filled) to a Latin square which is not a solution to a Sudoku because it does not satisfy the additional constraint whereby each 3 x 3 subgrid must be composed of numbers 1 – 9.

## Latin Squares: Theoretical Facts and Results

**A Formula to Compute the Total Number of Latin Squares of Order $n$, $L(n)$ [9]**

$$L_n = n! \sum_{A \in B_n} (-1)^{\sigma_0(A)} \binom{perA}{n}$$

- $B_n$ is the set of $n \times n$ (0,1) matrices
- $\sigma_0(A)$ is the number of zero elements of the matrix $A \in B_n$
- $perA$ is the permanent of the matrix $A$

- **Completing Latin Squares is an NP-Complete Problem:** It is known that completing partially filled Latin squares is an NP-Complete problem and there is a proof to go along with it, i.e., deciding whether a Latin square can be completed is NP-complete. The proof is done by establishing that every uniform tripartite graph is the defect of some partial Latin square [10].

| $n$ | $L_n$ |
|---|---|
| 1 | 1 |
| 2 | 2 |
| 3 | 12 |
| 4 | 576 |
| 5 | 161,280 |
| 6 | 812,851,200 |
| 7 | 61,479,419,904,000 |
| 8 | 108,776,032,459,082,956,800 |
| 9 | 5,524,751,496,156,892,842,531,225,600 |
| 10 | 9,982,437,658,213,039,871,725,064,756,920,320,000 |
| 11 | 776,966,836,171,770,144,107,444,346,734,230,682,311,065,600,000 |

Table 3.1: Table of all known exact values for number of Latin squares of order $n$ [4, 11]

**Remark 1** (Upper Bound on Number of $9 \times 9$ Sudoku Grids). *Value for $L_9$ is an upper bound on the number of $9 \times 9$ Sudoku grids.*

## Sudoku as a Code

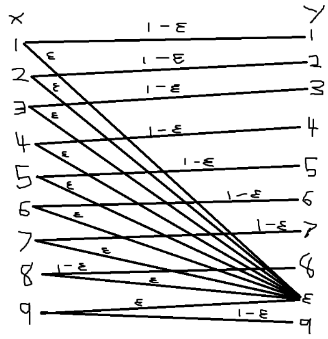

Figure 3.3: 9-ary erasure communication channel modelling Sudoku grid

A Code is a system of rules for converting information into another form for communication through a communication channel. A Sudoku grid is a codeword of the Sudoku code. An Error-Correcting Code (ECC) is used for controlling errors in data over unreliable or noisy communication channels. This is the case in Sudoku as some squares will be erased as the grid is encoded. This encoding will then lead to codewords representing unique Sudoku grids (refer to Unique Grid in glossary). A Codeword is a block of bits the encoder is allowed to transmit over the channel. So, a Sudoku grid can be seen as a 9-ary code where only certain codewords are allowed.

Thus, a Sudoku puzzle can be seen as putting a Sudoku grid (codeword) through a 9-ary erasure communications channel. A sudoku grid is a codeword from the $9 \times 9$ Sudoku code (which is a 9-ary code); a partially filled Sudoku grid is therefore a noisy codeword, as we have erased codeword symbols. The task of the player is to decode the original codeword (i.e. the original grid) from the noisy output of the channel. Since it is known how many Sudoku grids exist (i.e., the number of Sudoku grids has been enumerated), the Rate (R) of the code is also known. For this reason, information theory should be able to allow us to investigate the minimum number of clues in an analytical way.

Thus, the Minimum Sudoku Clue Problem can be studied from the point of view of standard information theory, where the uses of the channel go asymptotically to infinity. To refine the results above it will be necessary to resort to zero-error capacity [12] in the erasure channel, over a finite number of channel uses (i.e., 81 uses).

The tools of Information Theory — specifically codes, error-correcting codes and erasure communication channels — have now been introduced as a way to model Sudoku grids. We will now go into a bit more detail on some of the key ideas in Information Theory that allow us to create this model and provide a brief introduction to Information Theory as a field of study.

### 3.1.3 Information Theory Insights from Shannon

To gain additional historical context, foundational knowledge and background on the tools of Information Theory and its origin as a field of study; apart from the module I took here at UCD, Information Theory (COMP30690) taught by my supervisor Félix Balado Pumariño, I went through the paper on Claude E. Shannon written by some former colleagues of his at MIT [13]. This paper provided some interesting key findings that I would like to briefly discuss and communicate their relevance to the project in the following paragraphs.

## Shannon's Communication Diagram

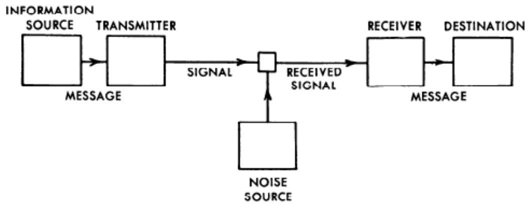


Figure 3.4: Shannon's communication diagram [13].

**Information Source:** The information source selects a desired message out of a set of possible messages e.g. selects which numbers from set $\{1, 2, ..., 9\}$ to place in which cells in the Sudoku grid.

**Transmitter (Encoder):** The transmitter changes this message into the signal which is actually sent over the communication channel from the transmitter to the receiver e.g. uses a code to convert Sudoku grid into a 9-ary code to be sent over the erasure channel.

**Channel:** The channel is merely the medium of communication used to transmit the signal from transmitter to receiver e.g. erasure communication channel.

**Signal:** Can travel through channel e.g. a bit.

**Noise:** In the process of being transmitted, it is unfortunately characteristic that certain things are added to the signal which were not intended by the information source e.g. random erasures in the channel.

**Receiver (Decoder):** The receiver is a sort of inverse transmitter, changing the transmitted signal back into a message, and handing this message on to the destination e.g. the player will be the receiver; decoding the codeword representing a completely filled Sudoku back into an actual Sudoku grid then deciphering the original grid from it.

**Destination:** The destination is the person (or thing) for whom the message is intended e.g. the player.

The above diagram abstracts away the details and semantics of an engineering problem at hand and provides a model for which to frame the engineering problem and analyse it using the rigorous Mathematical tools of Information Theory. In the context of this project, Sudoku will be modelled as a 9-ary erasure communication channel where only certain codewords will be allowed to be put through the channel. Any set of nine different symbols can be used but we will use the set $\{1, ..., 9\}$ for Sudoku (remember Sudoku constraint from previous section 1.2). Numbers will be selected from the set at the information source, encoded and then passed through the channel.

**Key Finding:** "The semantic aspects of communication are irrelevant to the engineering problem. The significant aspect is that the actual message is one selected from a set of possible messages" [13]

**Relevance to Project:** This is an interesting and seemingly counter-intuitive observation. In relation to this research project, that means the numbers 1,...,9 are irrelevant in determining the solution to the Minimum Sudoku Clue Problem. Any set of nine symbols could be used, so they can be abstracted away and focus can be centred on the important details; how are the cells in the Sudoku grid being filled? How are the cells being selected? Which cells are being filled first and what is the strategy behind their selection? These questions are independent of the symbol set being used.

## Shannon's Second Fundamental Theory of Communication

**Shannon's Second Fundamental Theory of Communication:** Let a discrete channel have a capacity C and a discrete source the entropy per second R (rate). If $R < C$ there exists a coding system such that the output of the source can be transmitted over the channel with an arbitrarily small frequency of errors. If $R > C$ it is possible to encode the source so that the equivocation is less than $R - C + \epsilon$, where $\epsilon$ is arbitrarily small. There is no method of encoding which gives an equivocation less than $R - C$ [13].

**Key Finding:** "The idea that Shannon is conveying is that no matter what the noise, there is an encoding scheme that allows you to transmit the information error-free over the channel (so long as $R < C$). Again, this idea was revolutionary as it was believed that after a certain level of noise, it would be impossible to transmit the desired signal." [13]

**Relevance to Project:** This theory in conjunction with Zero-Error Capacity over a Noisy Channel [12] is the foundation on which the advanced part of the project is based on. This theory states that no matter how noisy the channel is, there exists an encoding scheme that allows you to transmit the information error-free over the channel, so long as $R < C$. So, in theory, we need to determine whether $R < C$ for our 9-ary erasure communication channel which is modelling Sudoku and then that will tell us about the existence of the encoding scheme that will allow us to transmit the Sudoku codewords error-freely over the channel and then we will just have to find this encoding scheme with the assurance that it is out there, it exists.

Now the key ideas in Information Theory that will be used in this research project have been outlined, it should be clearer to see the connection between Sudoku and Information Theory. In the following section, the history of research done on the Minimum Sudoku Clue Problem will be traced and previous attempts at using Information Theory to approach the Minimum Sudoku Clue Problem will be briefly discussed.

## 3.2 Related Work and Ideas

Extensive research has been done on the Minimum Sudoku Clue Problem by a myriad of contributors and communities across the globe with a variety of methods, including a mix of exhaustive computer searches, formal and computer-assisted proofs. In this section along with the following subsections, we will provide a detailed account of the history of research on the Minimum Sudoku Clue Problem and mention some examples of previous attempts at using the tools of Information Theory to approach the Minimum Sudoku Clue Problem.

### 3.2.1 Previous Work on the Minimum Sudoku Clue Problem

Here, we have a timeline — largely constructed by McGuire et al [2] — of the previous research done on the Minimum Sudoku Clue Problem. Starting from around the time the first modern Sudoku grid appeared in newspapers and continuing to the most recent progress made on the Minimum Sudoku Clue Problem.

- **Sudoku introduced in Japan (1980s):** In Japan, Sudoku was introduced by the publisher Nikoli in the 1980s. Japanese puzzle creators have made puzzles with 17 clues, and with no doubt wondered whether 16 clues were possible. Nikoli have a rule that none of their puzzles will have more than 32 clues.
- **Collection of 17-clue Sudokus over the years:** Over the years, people have collected close to 50,000 different 17-clue Sudoku puzzles, which may be downloaded from Gordon Royle's homepage [14]. Most of these were found by Royle, who is a mathematician at the University of Western Australia and who compiled this list while searching for a 16-clue puzzle. On the Sudoku forums [15] McGuire et al. became aware of a rather special grid, which has 29 different 17-clue puzzles, also discovered by Gordon Royle. Many considered this grid a likely candidate to contain a 16-clue puzzle. Actually, McGuire et al. initially started to work on checker to search this particular grid.
- **Minimum Sudoku Clue Problem gaining widespread attention (2007):** By the end of 2007, the Sudoku minimum number of clues problem had been mentioned in several journal publications. The first one is an article by the French computer scientist Jean-Paul Delahaye entitled The Science behind Sudoku [7], which was first published in the June 2006 issue of the Scientific American. This article in fact quotes one of the authors of this study (Gary McGuire) in conjunction with the minimum number of clues problem.
- **Exhaustive Computer Search for 11-clue and 12-clue Sudoku (2007 - 2009):** Between autumn 2007 and spring 2008, a team at the University of Graz in Austria verified by distributed exhaustive computer search that there are no proper Sudoku puzzles having eleven clues. They continued for about another year, and by May 2009, they had apparently completed most of the computations necessary to show that twelve clues are never sufficient for a unique solution either. Their stated aim was to build up to proving that no 16-clue Sudoku puzzle exists, yet as of early 2010, that project seemed not to be active anymore.
- **Max Neunhöffer lectures on Mathematics of Sudoku (Since 2007):** Since 2007, Max Neunhöffer of the University of St. Andrews in Scotland has lectured several times at different venues on the mathematics of Sudoku, discussing in particular the minimum number of clues problem. According to the slides of his talks, Neunhöffer appears to even have written a computer program for searching a Sudoku solution grid for 16-clue puzzles.
- **17 year old girl submits potential proof of nonexistence of a 16-clue Sudoku (2008):** In 2008, a 17-year-old girl submitted a proof of the nonexistence of a 16-clue Sudoku puzzle

as an entry to Jugend forscht (the German national science competition for high-school students). She later published her work in the journal Junge Wissenschaft (No. 84, pp. 24–31). However, when Sascha Kurz, a mathematician at the University of Bayreuth, Germany, studied the proof closely, he found a gap that is probably very difficult, if not impossible, to fix.

- **Mladen Dobrichev's open-source GridChecker (mid-2010):** In mid-2010, the Bulgarian engineer Mladen Dobrichev released the initial version of his opensource tool GridChecker; he has since provided several updates. Dobrichev's programme, also written in C++, basically does the same thing as McGuire et al.'s original checker, although it is at least one order of magnitude faster.

- **Distributed search over the internet for 16-clue Sudokus using BOINC lead by Computer Scientists at the National Chiao Tong University, Taiwan (late-2010):** In October 2010, a group of computer scientists at the National Chiao Tung University, Taiwan, led by I-Chen Wu started a distributed search for 16-clue puzzles on the Internet using BOINC. Their strategy is exactly the same as McGuire et al.'s (i.e., consider each Sudoku solution grid individually and exhaustively search for a 16-clue subset whose only completion is the given grid). This had become feasible since Wu's team had succeeded in speeding up McGuire et al.'s original checker by a factor of 129, so that it would take them an estimated 2,400 processor-years to examine every grid for 16-clue puzzles. Wu also spoke about this in November 2010 at the International Conference on Technologies and Applications of Artificial Intelligence, and he and Hung-Hsuan Lin together published the article Solving the Minimum Sudoku Problem in the conference proceedings, describing some of the techniques they had used to improve checker. As far as the progress of the BOINC search is concerned, according to that project's website, as of 31st December 2011 - the day before McGuire et al. announced their result (arXiv :1201:0749v1) - they had checked 1,453,000,000 Sudokus (26.5 per cent of all cases that need to be considered).

- **Taking Sudoku Seriously book by Jason Rosenhouse and Laura Taalman (2011):** Toward the end of December 2011, the digital edition of the book Taking Sudoku Seriously: The Math Behind the World's Most Popular Pencil Puzzle by Jason Rosenhouse and Laura Taalman of James Madison University in Harrisonburg, Virginia, came out; the printed version followed several weeks later [8]. This book devotes the whole of Section 9.4 The Rock Star Problem to the minimum number of clues problem, saying:

  *"If you can figure it out, you will be a rock star in the universe of people who care about such things. Granted, that is a far smaller universe than the one full of people who care about actual rock stars, but still, it would be great."*

- **There is no 16-Clue Sudoku: Solving the Sudoku Minimum Number of Clues Problem via Hitting Set Enumeration (2013):** In 2013, Gary McGuire from University College Dublin, Bastian Tugemann a former student from Munich Germany and Gilles Civario from the Irish Centre for High-End Computing, Dublin, Ireland performed an exhaustive computer search for 16-clue Sudoku puzzles, and did not find any, thus proving empirically by exhaustive computer search that the minimum number of Sudoku clues for a Sudoku to have a unique solution is indeed 17. The paper they published "There is no 16-Clue Sudoku: Solving the Sudoku Minimum Number of Clues Problem via Hitting Set Enumeration" describes the hitting set enumeration method they used to reduce the search space considerably and efficiently search each of the hitting sets in time.

  *"As a side comment, there have been attempts to solve the Sudoku minimum number of clues problem using mathematics only, i.e., without the help of computers. However, nobody has made any real progress — though it is easy to see why a 7-clue puzzle cannot have a unique solution,1 a theoretical proof of the nonexistence of an 8-clue puzzle is still lacking. This is far from the answer of 17, so that a purely mathematical solution to the minimum number of clues problem is a long way off." [2]*

The Minimum Sudoku Clue Problem is challenging indeed. Even the mighty computer has faltered out as it has been pushed to its limits by finding the minimum number of clues for 9 x 9 Sudoku grids. Continuing along this computer-assisted proof path seems intractable at best as more computing power or more sophisticated search algorithms will need to be constructed. Even if results are found for Sudoku grids of higher dimensions, it still irks us to know why. Computer-assisted proofs and exhaustive searches provide empirical results at best through brute force. However, grounding the Minimum Sudoku Clue Problem in theory could produce better results when it comes to generalising the problem. Hence, the computer can be used to verify and evaluate these theoretical results in an ideal world. Though, having a solid theory combined with a rigorous proof would give us more confidence going forward for the Minimum Sudoku Clue Problem and remove that itch concerned with knowing why. A mathematical proof to show the non-existence of a 16-clue Sudoku grid has been attempted. Perhaps we can try a hand at a theoretical approach using the tools of information theory to allow us to analyse the Minimum Sudoku Clue Problem rigorously and solve it in general.

In the next section, we will look at examples of previous attempts to use the tools of Information Theory to approach the Minimum Sudoku Clue Problem.

### 3.2.2 Previous Attempts at Using Information Theory to Approach MSCP

There is minimal relevant literature regarding using the tools of Information Theory to approach the Minimum Sudoku Clue Problem. Most previous attempts at using Information Theory in Sudoku problems centre around constructing Sudoku grids by exploiting entropy [16–20] or by viewing Sudoku as matrices and performing operations on them and using Information Theory to inform heuristics. Overall, it seems like there is a common thread of a combination of a set of operations and a systematic computation strategy being guided by Information Theory to reduce the computational time complexity. Although these papers do not address the Minimum Sudoku Clue Problem directly, they provide some interesting, noteworthy methods and techniques which could shed some light on the problem and provide some insight into possible paths to follow and applications.

In the remainder of this report, we will go into detail on how we used the tools of Information Theory to approach the Minimum Sudoku Clue Problem.

# Chapter 4: Data & Context

In this chapter, the information deemed most relevant to the research project has been gathered including databases, theorems, empirical results and formulas. These will come into play when evaluating the accuracy and effectiveness of our theoretical results. Our theories need to be tested for us to be confident in their reliability. They should correspond with known empirical results and if they are tested adequately, should produce reliable and trustworthy results when generalized. In the following sections, we enumerate the different data sources and their potential contribution to the research project.

## 4.1 Online Database of Sudoku Puzzles

The website `menneske.no` hosts an online database of Sudoku puzzles of various shapes and sizes; some familiar and some unfamiliar which is being maintained by Vegard Hanssen [21]. This could be helpful when creating a database of Sudoku puzzles of size $N \times N$ with $q$ clues and modelling a pmf of Sudoku of size $N \times N$ with $q$ clues, e.g., plot distribution of number of clues amongst all $9 \times 9$ Sudoku grids.

- **Additional documentation:** There is also some interesting documentation here about "Reducing Methods" for Sudokus, which is essentially starting with all possible solutions in a sense and trying to centre on creating a Sudoku instance such that it has a unique solution. This is similar to what we are doing in the erasure channel.
- **Statistics on distribution of Sudoku in database and number of clues they have:** There is some useful empirical statistical data on the Sudoku/Show distribution section of the website; correlating the distribution of the number of clues and the frequency of puzzles in the database. It could present an interesting plot with a shape that could be familiar to us. This is immediately applicable to the aforementioned side goal of modelling a pmf of Sudoku of size $N \times N$ with $q$ clues.

## 4.2 Collection of All Currently Known 17-clue Sudoku Solutions

This is a collection of all currently known 17-clue Sudoku solutions being hosted on Gordon Royle's webpage [14]. It could prove useful for creating a Sudoku generator that can enumerate all Sudokus of size $N \times N$ with $q$ clues e.g. generating all possible Sudoku grids of size $9 \times 9$ with 17 clues. Should we be able to write a program to perform our erasure channel encoding and incorporate a parameter to tweak the number of clues, then this would not be too far off being feasible.

## 4.3 Existing Empirical Results on MSCP and Combinatorics of Sudoku

Most of the sources for the empirical results we will be using to test our theoretical hypotheses have already been mentioned and there is agreement on those results amongst these sources [2, 22, 23]. Relevant results are displayed in tables below.

| $N \times N$ | $N_s$ |
|---|---|
| $4 \times 4$ | 288 [8, 23] |
| $9 \times 9$ | 6,670,903,752,021,072,936,960 $\approx 6.7 \times 10^{21}$ [24] |
| $16 \times 16$ | unknown, estimated $5.9584 \times 10^{98}$ [15] |
| $25 \times 25$ | unknown, estimated $4.3648 \times 10^{308}$ [15] |

Table 4.1: Table of currently known $N_s$ for grids of size $N \times N$ [22]

| $N \times N$ | Minimum Number of Clues |
|---|---|
| $4 \times 4$ | 4 |
| $9 \times 9$ | 17 |
| $16 \times 16$ | At least one puzzle with 55 clues has been created. It is not known if this is the fewest possible |
| $25 \times 25$ | At least one puzzle with 151 clues has been created. It is not known if this is the fewest possible |

Table 4.2: Table of currently known minimum number of clues for Sudoku of size $N \times N$ [22]

From my research so far, Lauren Puskar has some well written definitions on the terminology regarding Sudoku such as the bands, cells, the technical details; the same terminology is used in this report. Her paper [25] is quite good for familiarising oneself on the technical details regarding Sudoku in order to look at or gain access to more sophisticated or advanced papers and materials on the topic or just out of interest. The EnjoySudoku forum [26] provides an excellent and comprehensive collection of the terminology used in the Sudoku community. Below, are listed some theorems which have been proven to be true by mathematical proof; regarding $4 \times 4$ and $9 \times 9$ Sudoku. Proofs can be found in the cited papers.

### 9 x 9 Sudoku Theorems on the Minimum Number of Clues for a Unique Solution

- **Lemma 3.7.** The pigeonhole principle states that if $n$ items are put into $m$ containers, with $n > m$, then at least one container must contain more than one item.

  The pigeonhole principle comes into play when manipulating columns of stacks in the Sudoku grid during the proofs of these theorems. More detail on this is provided in Puskar's paper [25].

- **Theorem 4.1.** If we are given exactly one clue, called $q$, which is a number from the set $\{1, ..., 9\}$, on a blank Sudoku board, then there is not a unique solution.

- **Theorem 4.2.** If we are given up to exactly 5 clues, called $q_1, q_2, q_3, q_4$ and $q_5$, which are numbers from the set $\{1, ..., 9\}$, on a blank Sudoku board, then there is not a unique solution.

- **Theorem 4.3.** If we are given up to exactly seven clues, called $q_1, q_2, q_3, q_4, q_5, q_6, q_7$ which are numbers from the set $\{1, ..., 9\}$, on a blank Sudoku board, then there is not a unique solution.

### Shidoku Theorems on the Minimum Number of Clues for a Unique Solution

- **Theorem 5.1.** If four clues are given along the diagonal of a $4 \times 4$ board and they need not be unique, then the solution board is not unique [25].

## 4.4 Kilfoil-Silver-Pettersen Formula Approximation of Total Number of Sudoku Grids of Size N x N

**Kilfoil-Silver-Pettersen's formula computes an approximation for the total number of Sudoku grids of size N x N [8, 22]**

$$\text{Number of Grids} \approx \frac{b_{R,C}^{C} \times b_{C,R}^{R}}{(RC)!^{RC}}$$

- Kilfoil-Silver-Pettersen Formula **asserts** that the **Sudoku row and column constraints** are, to **first approximation**, **conditionally independent** given the **box constraint**.
- $b_{R,C}$ is the number of ways of completing a Sudoku band of $R$ horizontally adjacent $R \times C$ boxes.
- $R$ Number of rows in the subgrid of the Sudoku.
- $C$ Number of columns in the subgrid of the Sudoku.

**Kilfoil-Silver-Pettersen Formula:** This formula generalises the computation of the total number of Sudoku grids of size $N \times N$ with a unique solution and achieves a very accurate approximation that is not too far from the exact values that are already known. It is currently the fastest known technique for computing approximations of these $b_{R,C}$.

**Remark 2** (Nonsquare Sudoku Grids). *The Kilfoil-Silver-Pettersen formula allows nonsquare Sudoku grids but we are always assuming $R = C$ throughout the document.*

| $N \times N$ | C (Number of Columns in Subgrid) | Approximation of $N_s$ |
|---|---|---|
| $4 \times 4$ | 2 | 256 |
| $9 \times 9$ | 3 | 6,657,084,616,885,512,582,463.49 |
| $16 \times 16$ | 4 | $5.9584 \times 10^{98}$ |
| $25 \times 25$ | 5 | $4.3648 \times 10^{308}$ |

Table 4.3: Table of approximations given by Kilfoil-Silver-Pettersen's formula to compute the total number of Sudoku grids of size $N \times N$ with a unique solution [6, 8]

A discussion thread on the Enjoy Sudoku forum [15] provides some insight on the Kilfoil-Silver-Pettersen formula. Apparently, the formula does not take size into account. The interested reader can find more details about this in the remainder of the discussion thread to get more context. In the following chapters, we discuss our research findings and evaluate our theoretical results against these data sources after applying the tools of Information Theory to the Minimum Sudoku Clue Problem.

# Chapter 5: Core Contribution

After applying the tools of Information Theory to the Minimum Sudoku Clue Problem; specifically coding, rates, capacity and erasure communication channels, we have been able to make a few significant conclusions and have also made some progress on the Minimum Sudoku Clue Problem. We document our findings in the remainder of the chapter, along with further details in the appendices A and B.

## 5.1 Rates of Sudoku Codes

The table A.1 contains our results for the computation of the rates for Sudoku of sizes 4 x 4, 9 x 9, 16 x 16 and 25 x 25. Thanks to the exhaustive computer search by McGuire et al., we are able to compute the rate exactly for 9 x 9 Sudoku grids but have to settle for approximations for grids in higher dimensions.

The formula for the computation of the rate of codes representing Sudoku grids of size $N \times N$ is presented in appendix A. However, it depends on $N_s$, the number of Sudoku grids of size $N \times N$. This is a challenge that can be overcome by either constructing a formula to compute $N_s$ (unsolved problem) or by creating a formula that provides a permissible approximation for $N_s$ within a margin of error. Luckily, we have such a formula for the latter with the Kilfoil-Silver-Pettersen formula (refer to page 23). Nevertheless, this formula takes factorial time complexity to compute and more analysis needs to be done to track its error margins against actual values.

In conclusion, both potential options presented to overcome the challenge of computing an exact value or an approximation for $N_s$ seem to be about even in terms of working out. Perhaps the best option is to compare the time complexity of both options and proceed with the option that would be more susceptible to simplification. As our options are either to throw more computing power at the problem, in that case we are limited by Moore's law and waiting for a breakthrough in computing infrastructure or to simplify one of these formulas by rephrasing their construction or gaining insight into on of the terms composing the formula. One could also start from scratch and count the number of Sudoku grids of size $N \times N$ in a completely different way. For now, we will have to settle with what we have but these results for the rate and the formulas to go along with them are a good start.

## 5.2 Capacity of Erasure Channels for Sudoku Codes

Modelling partial Sudoku grids as codewords allowed us to view the game of Sudoku as putting a Sudoku grid (codeword) through a 9-ary erasure communication channel with the task of the player being to decode the original codeword, i.e., the original grid. It has also allowed us to compute the capacity of the channel.

**9-ary Erasure Communication Channel for 9 x 9 Sudoku Codewords:** Figure 3.3

The capacity (C) of a channel provides a maximum upper bound on the rate as the number of channel uses goes asymptotically to infinity. For Sudoku, we have a finite number of channel uses, i.e. 81 uses. Therefore when $N = 9$, the capacity will be able to provide an approximation for the minimum number of clues. Specifically, an upper bound on the minimum number of clues which can be refined by resorting to Shannon's Zero-Error Capacity of a Noisy Channel ($C_0$) Theorem.

**Capacity C of N-ary Erasure Communication Channel for N x N Sudoku Codewords** = $(1 - \epsilon) \log_2 N$ bits (Appendix B)

Having these results for the capacity allows to immediately compute reliable approximations for the minimum number of clues in general and fine tune the accuracy of these approximations to our liking. We can be confident in the accuracy of these approximations as they are grounded in the rigorous workings of Information Theory.

An Analytical proof explaining in detail and providing justification for the above, boxed capacity result and the capacity of erasure communication channels in general, which Sudoku codewords will be put through, can be found in appendix B.

## 5.3 Bounds for the Minimum Sudoku Clue Problem

Tuning the probability of erasure, $\epsilon$, allows us to generate bound approximations for the Minimum Sudoku Clue Problem as $\epsilon$ is the sole variable in the computation of capacity (C). By informing our method for selecting an appropriate $\epsilon$, we can improve the accuracy of our bounds; tightening them as we centre on the true growth trend generalising the Minimum Sudoku Clue Problem. In the following subsections, we outline how we were able to generate upper and lower bounds with reasonable accuracy which provide good approximations for the Minimum Sudoku Clue Problem.

### 5.3.1 Upper Bound Approximation for the Minimum Sudoku Clue Problem

To obtain the upper bound promised by the capacity of the erasure channel we have to select a value for $\epsilon$, the probability of erasure in the channel. We select $\epsilon$ such that R = C. Where R is the rate of the Sudoku code, C is the capacity and $\epsilon$ is the probability of erasure in the channel. Selecting $\epsilon$ this way will immediately give us a good upper bound approximation on the Minimum Sudoku Clue Problem. Values for $\epsilon$ for codewords representing Sudoku grids of sizes $4 \times 4$, $6 \times 6$, $16 \times 16$ and $25 \times 25$ can be obtained using the following computations. The $4 \times 4$ example is detailed below;

## Computing Probability of Erasure $\epsilon$ such that $R = C$

As mentioned previously, setting $\epsilon$ such that $R = C$ will immediately provide us with a reasonable upper bound on the Minimum Sudoku Clue Problem. The $4 \times 4$ case is outlined in the following lines;

**Rate (R) of $4 \times 4$ Sudoku Code** = 0.5106203126 bits/channel use (Table A.1)
**Capacity (C) of Sudoku Code** = $(1 - \epsilon) \log_2 N$ bits (Appendix B)
R = C
0.5106203126 = $(1 - \epsilon) \log_2 4$
$\epsilon$ = 0.7446898437
$\therefore$ at $\epsilon$ = 0.7446898437, R = C for codewords modelling $4 \times 4$ Sudoku grids

The same computations can be used for Sudoku of other sizes. Below, we list $\epsilon$ values such that R = C for Sudoku grids of size $9 \times 9$, $16 \times 16$ and $25 \times 25$.

At $\epsilon$ = 0.7176458017, R = C for codewords modelling $9 \times 9$ Sudoku grids.
At $\epsilon$ = 0.6795661468, R = C for codewords modelling $16 \times 16$ Sudoku grids.
At $\epsilon$ = 0.6467488305, R = C for codewords modelling $25 \times 25$ Sudoku grids.

As you can see, the probability of erasure is decreasing as the size of the Sudoku grid increases i.e. as the size of the grid increases, each cell has a lower probability of being erased i.e. inversely proportional relationship between Sudoku grid size and probability of erasure in the erasure channel. The above computations are in line with Shannon's Noisy Channel Coding Theorem. This means that we can manipulate the probability of erasure $\epsilon$ to make it arbitrarily small. This means that, in theory, it is possible to encode the Sudoku grids nearly without error at any rate below a limiting rate, C. Graphs illustrating these trends will be displayed in the next sections.

# Chapter 6: **Evaluation**

Now we will evaluate the accuracy and effectiveness of our theoretical results by comparing them with known empirical results. In the following sections, we will present plots showing the trend of growth of our bounds detailed in the previous chapter. We will also plot other measures against each other to observe whether there is an interesting correlation between the measures in question and see how they inform our evaluation. First, we present our methodology for describing the encoding process occuring at the information source as this will produce results that appear on the plots and in the tables. We will then discuss these results and analyse both their accuracy and effectiveness for approaching the Minimum Sudoku Clue Problem.

## Processing Sudoku Grid in Erasure Channel

The number of erasures in the erasure channel modelling Sudoku of size $N \times N$ follows a binomial distribution Bi $(\epsilon,\, N^2)$.

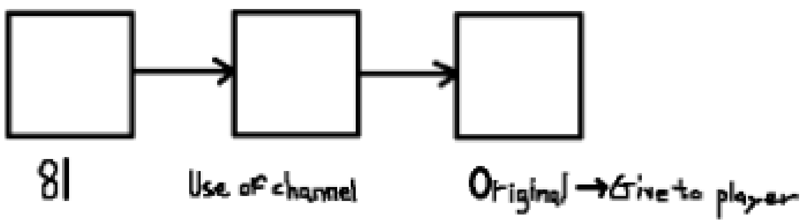


Figure 6.1: Processing Sudoku grid in erasure channel

- Channel Capacity (C) tends to infinity $\therefore$ the results for capacity are asymptotic and should be regarded as approximations. The point of intersection making R = C should be a good indicator of where the minimum number of clues resides as we have detailed in the previous chapter.
- **Probability of an erasure** $= \epsilon$
- **Probability of non-erasure** $= 1 - \epsilon$
- **Number of channel uses** $= N^2$
- **Number of non-erasures (clues)** $= N^2(1-\epsilon)$
- **Number of erasures** $= N^2\epsilon$
- $N^2(1-\epsilon) + N^2\epsilon = N^2(1-\epsilon+\epsilon) = N^2$

Now the encoding process has been explained, the plots for the bounds and measures can be presented with the confidence that they can be soundly interpreted.

## 6.1 Plotting Upper Bound Approximation for Minimum Sudoku Clue Problem

Presented are a plot and table of the previously mentioned upper bound approximation for the Minimum Sudoku Clue Problem, along with a brief analytical discussion of the plot that comments on its accuracy.

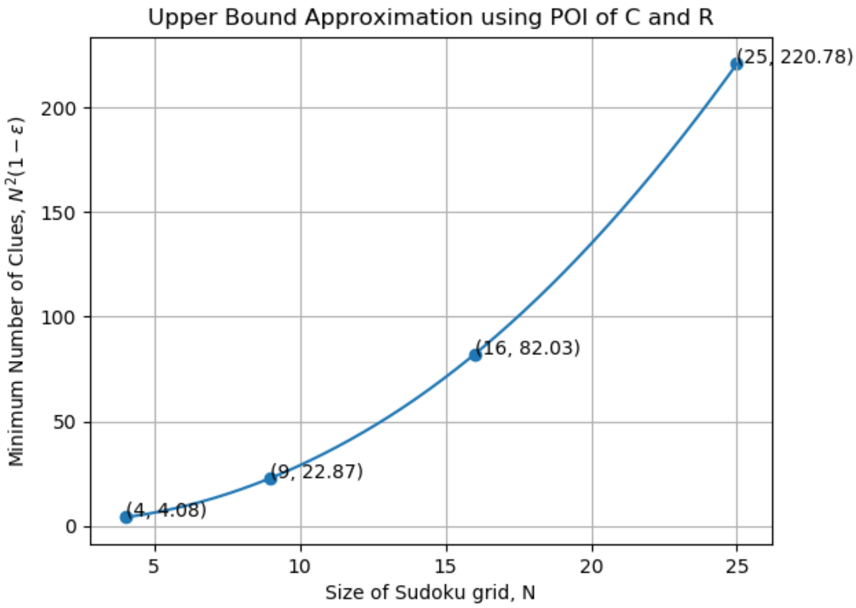


Figure 6.2: Plot of upper bound approximation for minimum number of clues in Sudoku of size $N \times N$ using probability of erasure, $\epsilon$ : R = C.

| $N \times N$ | Minimum Number of Clues Upper Bound | Percentage Error ($\frac{\lvert approx - actual \rvert}{actual} \times 100$) |
|---|---|---|
| $4 \times 4$ | 4.084962501 | 0% |
| $9 \times 9$ | 22.87069006 | 34.53% |
| $16 \times 16$ | 82.03106644 | 49.15% |
| $25 \times 25$ | 220.7819809 | 46.21% |

Table 6.1: Table showing upper bound approximations for minimum number of clues and percentage error

The above graph and table show a trend of overestimates becoming more accurate as $N$ increases. This is because capacity (C) tends to infinity and thus gives asymptotic results. To correct these overestimates and refine the results, Shannon's zero-error capacity theorem will need to be employed.

## 6.2 Plotting Capacity (C) against $\epsilon$ such that R = C for N x N Sudoku Grids

A plot and table of the capacity (C) against $\epsilon$ such that R = C for Sudoku grids of size $N \times N$ are presented, along with a paragraph commenting on the perceived trend in the graph and the overall verdict on the results in the table.

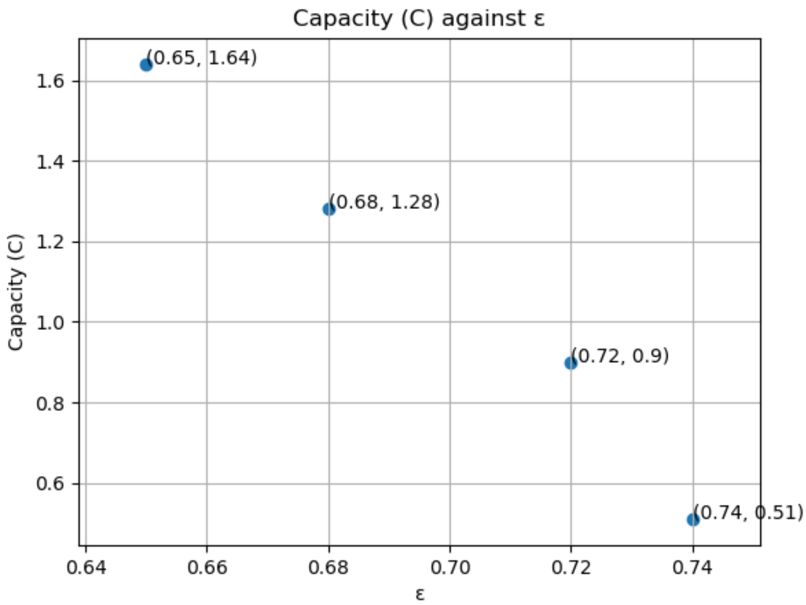


Figure 6.3: Plot of Capacity (C) against $\epsilon$ such that C = R for Sudoku grids of size $N \times N$

| $N \times N$ | Capacity (C) | $\epsilon$ |
|---|---|---|
| $4 \times 4$ | 0.5106203126 | 0.7446898437 |
| $9 \times 9$ | 0.8950416323 | 0.7176458017 |
| $16 \times 16$ | 1.281735413 (Approximation) | 0.6795661468 |
| $25 \times 25$ | 1.64044763 (Approximation) | 0.6467488305 |

Table 6.2: Table showing capacity (C) and probability of erasure, $\epsilon$ for Sudoku grids of size $N \times N$

From observing the graph, there is a seemingly linear trend going through the capacity for varying values of the probability of erasure, $\epsilon$. The main takeaway here is that there is an inversely proportional relationship between the capacity (C) and the probability of erasure, $\epsilon$.

# Chapter 7: **Summary and Conclusions**

Hopefully I have illustrated the rich history of Sudoku and the extensive research that has gone into the Minimum Sudoku Clue Problem to date which has been contributed to by a diverse range of people and communities. It is with great honour that this paper is able to add to that extensive research with the discussed results achieved through the use of the tools of information theory.

Using erasure channels to model the game of Sudoku proved to be a very effective method for tackling the Minimum Sudoku Clue Problem as the results have shown. The approximations given by using the actual rate of Sudoku grids for sizes $4 \times 4$ and $9 \times 9$ and the approximations for $16 \times 16$ and $25 \times 25$ provided a sufficient upper bound on the Minimum Sudoku Clue Problem which shows a trend of reduced percentage error in higher dimensions. However, for this upper bound to generalize, it is necessary to obtain a formula or approximations for the total number of Sudoku grids of size $N \times N$.

Some of the side goals were achieved but others are still outstanding. Formulas to compute the rate and capacity of Sudoku grids of size $N \times N$ have been created but still outstanding are;

1. Model pmf of Sudoku of size $4 \times 4$ with $q$ clues i.e. plot distribution of number of clues amongst all $4 \times 4$ Sudoku [21]
2. Model pmf of Sudoku of size $9 \times 9$ with $q$ clues i.e. plot distribution of number of clues amongst all $9 \times 9$ Sudoku e.g. 10,290 sudokus with 32 clues [21]
3. Model pmf of Sudoku of size $N \times N$ with $q$ clues i.e. plot distribution of number of clues amongst all $N \times N$ Sudoku [21]
4. Formula to Count Number of Sudoku of Size $N \times N$
5. Create database of Sudoku with $q$ clues e.g. Database storing number of known Sudoku grids of size $N \times N$ with $q$ clues i.e. Can store Sudoku grids in a way that differentiates them based on the number of clues they have.
6. Employ Zero-Error Capacity Theorem on current theoretical results to refine them
7. Zero-Error Capacity of an erasure channel
8. Sudoku generator to generate Sudoku with $q$ clues

Overall, the project has been a rewarding success and the main goals have been achieved. The progress made is satisfactory and there are still other opportunities to pursue in this line of research and the Minimum Sudoku Clue Problem at large.

# Chapter 8: Acknowledgements

I would like to express special thanks and appreciation for my supervisor, Félix Balado Pumariño for guiding my research project and assisting me; lending an ear to discuss any ideas or methods I had in mind, answering any questions I brought his way, making himself available and open to contact. Also, I would like to thank everyone who has worked on the Minimum Sudoku Clue Problem and The Enjoy Sudoku Forum; this forum filled with Sudoku enthusiasts has been quite helpful and a wonderful community in the Sudoku space. Finally, my family, for supporting me and helping me to do my best.

# Appendices

# Appendix A: Rates of Sudoku Codes

**Rate of N x N Sudoku Grid** $= \frac{\log_2 N_s}{N^2}$

**Redundancy of N x N Sudoku Grid** $= \log_2 N^2 - \log_2 N_s$

| $N \times N$ | $N_s$ | Rate of Code (bits/channel use) | Redundancy (bits) |
|---|---|---|---|
| $4 \times 4$ | 288 [8, 23] | 0.5106203126 | 23.830075 |
| $9 \times 9$ | 6,670,903,752,021,072,936,960 $\approx 6.7 \times 10^{21}$ [24] | 0.8950416323 | 84.2655529 |
| $16 \times 16$ | unknown, estimated $5.9584 \times 10^{98}$ [15] | 1.281735413 (Approximation) | 695.8757344 (Approximation) |
| $25 \times 25$ | unknown, estimated $4.3648 \times 10^{308}$ [15] | 1.64044763 (Approximation) | 1877.13035 (Approximation) |

Table A.1: Table showing rates and redundancies for Sudoku codes modelling Sudoku grids of size $N \times N$

Appendix B: **Analytical Proof of Capacity of Erasure Channels for Sudoku Codes**

## N-ary Erasure Channel

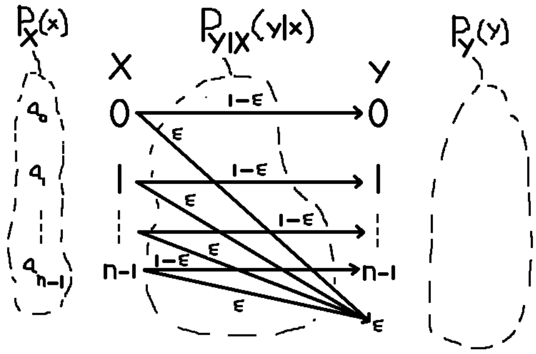


Figure B.1: N-ary Erasure Channel

- $\sum_i a_i = 1$
- $0 \leq a_i \leq 1$
- Transition probability matrix $= \Pi = [P_{X,Y}]$

$$\Pi = \begin{bmatrix} P_{Y|X}(0|0) & \dots & P_{Y|X}(n-1|0) & P_{Y|X}(\epsilon|0) \\ 0 & \ddots & 0 & P_{Y|X}(\epsilon|1) \\ \vdots & \dots & 0 & \vdots \\ 0 & \dots & P_{Y|X}(n-1|n-1) & P_{Y|X}(\epsilon|n-1) \end{bmatrix} = \begin{bmatrix} 1-\epsilon & 0 & \dots & 0 & \epsilon \\ 0 & \ddots & \ddots & \vdots & \epsilon \\ \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & \dots & 0 & 1-\epsilon & \epsilon \end{bmatrix}$$

Calculating $P_Y(y)$ in terms of $P_X(x)$

$$P_Y(y) = \sum_{x \in X} P_{X,Y}(x,y) = \sum_{x \in X} P_{Y|X}(y|x) P_X(x) = \sum_{x \in \{0,1,2\}} P_{Y|X}(y|x) P_X(x)$$

$$P_Y(i) = \sum_{x \in \{0,1,\dots,n-1\}} P_{Y|X}(i|x) P_X(x) = (1-\epsilon) a_i$$

$$P_Y(\epsilon) = \epsilon(a_0 + \cdots + a_{n-1}) = \sum_{i=1}^{n-1} = \epsilon(1) = \epsilon$$

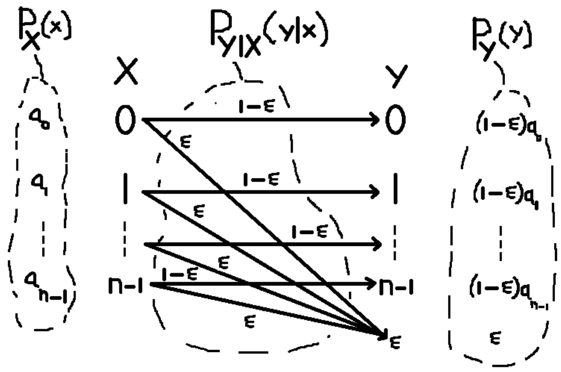


Figure B.2: N-ary Erasure Channel with $P_Y(y)$ inserted

Compute capacity (C) of N-ary Erasure Channel = $\max_{P_X(x)} I(X;Y) = H(Y) - H(Y|X)$

H(Y) = $-\sum_{y\in\{0,1,\ldots,n-1,\epsilon\}} P_Y(y)\log(P_Y(y)) = -(1-\epsilon)a_0\log((1-\epsilon)a_0)\cdots-(1-\epsilon)a_{n-1}\log((1-\epsilon)a_{n-1}) - \epsilon\log\epsilon$

H(Y|X) = $\sum_{x\in X} P_X(x)H(Y|X=x) = -\sum_{x\in\{0,1,\ldots,n-1\}} P_X(x)\sum_{y\in\{0,1,\ldots,n-1,\epsilon\}} P_{Y|X}(y|x)\log(P_{Y|X}(y|x))$

$= -\sum_{i\in\{0,1,\ldots,n-1\}} P_X(i)[P_{Y|X}(i|i)\log(P_{Y|X}(i|i)) + P_{Y|X}(\epsilon|i)\log(P_{Y|X}(\epsilon|i))]$

$= -\sum_{i\in\{0,1,\ldots,n-1\}} a_i[(1-\epsilon)\log(1-\epsilon)+\epsilon\log\epsilon]$

Entropy maximised when $a_i = \frac{1}{n}$

H(Y) = $-(\sum_{i=0}^{n-1}(1-\epsilon)\frac{1}{n}\log(\frac{1-\epsilon}{n})) - \epsilon\log\epsilon$

$= -(1-\epsilon)\log(\frac{1-\epsilon}{n}) - \epsilon\log\epsilon = -(1-\epsilon)\log(1-\epsilon) + (1-\epsilon)\log n - \epsilon\log\epsilon$

H(Y|X) = $-\sum_{i\in\{0,1,\ldots,n-1\}} a_i[(1-\epsilon)\log(1-\epsilon)+\epsilon\log\epsilon] = -(1-\epsilon)\log(1-\epsilon)+\epsilon\log\epsilon$

$= -\sum_{i=0}^{n-1}\frac{1}{n}[(1-\epsilon)\log(1-\epsilon)+\epsilon\log\epsilon]$

$= -(1-\epsilon)\log(1-\epsilon) - \epsilon\log\epsilon$

C = I(X;Y) = H(Y) - H(Y|X)

$= -(1-\epsilon)\log(1-\epsilon) + (1-\epsilon)\log n - \epsilon\log\epsilon + (1-\epsilon)\log(1-\epsilon) + \epsilon\log\epsilon$

$= (1-\epsilon)\log n$

**Capacity C of N-ary Erasure Channel** = I(X;Y) = H(Y) - H(Y|X) = $(1-\epsilon)\log_2 n$ bits

Taking base of logarithm as n instead of 2

**Capacity C of N-ary Erasure Channel** = I(X;Y) = H(Y) - H(Y|X) = $1 - \epsilon$ n-ary bits

**Remark 3** (Binary Erasure Channel). *The capacity becomes a known result in the binary case.*

□

# Bibliography


1. Chel, H. *A Sudoku with 17 Clues and its Unique Solution* https://www.researchgate.net/figure/A-Sudoku-with-17-clues-and-its-unique-solution_fig1_311250094/download. Accessed on 2019-11-17.
2. McGuire, G., Tugemann, B. & Civario, G. There is no 16-clue Sudoku: Solving the Sudoku Minimum Number of Clues Problem via Hitting Set Enumeration. *Experimental Mathematics* **23,** 190–217 (2014).
3. Wikipedia. *History of Magic Squares* https://en.wikipedia.org/wiki/Magic_square#Middle_East:_Persia,_Arabia,_North_Africa,_Muslim_Iberia. Accessed on 2020-03-10.
4. Wikipedia. *Latin Squares* https://en.wikipedia.org/wiki/Latin_square. Accessed on 2019-11-17.
5. Colburn, C. J. & Dinitz, J. H. *Handbook of Combinatorial Designs* 2nd (Chapman and Hall/CRC Press, 2007).
6. Wikipedia. *Sudoku* https://en.wikipedia.org/wiki/Sudoku. Accessed on 2019-11-17.
7. Delahaye, J.-P. The science behind Sudoku. *Scientific American* **294,** 80–87 (2006).
8. Rosenhouse, J. & Taalman, L. *Taking Sudoku Seriously* (Oxford University Press, 2011).
9. Shao, J.-y. & Wei, W.-d. TA Formula for the Number of Latin Squares. *Discrete Mathematics* **110,** 293–296 (1992).
10. Colburn, C. J. Complexity of Completing Partial Latin Squares. *Discrete Applied Mathematics* **8,** 25–30 (1984).
11. Of Integer Sequences, T. O.-I. E. *Number of Latin Squares of Order n* https://oeis.org/A002860. Accessed on 2020-03-24.
12. Shannon, C. The zero error capacity of a noisy channel. *IRE Transactions on Information Theory* **2,** 8–19 (1956).
13. Chiu, E., Lin, J., Mcferron, B., Petigara, N. & Seshasai, S. Mathematical Theory of Claude Shannon (2001).
14. Royle, G. *Minimum Sudoku* http://staffhome.ecm.uwa.edu.au/~00013890/sudokumin.php. Accessed on 2019-11-6.
15. Aiden, H. *estimate for 4 x 4 Sudoku* http://forum.enjoysudoku.com/su-doku-s-maths-t44-525.html#p15920, http://www.setbb.com/sudoku/. Accessed on 2019-11-20.
16. Gunther, J. & Moon, T. Entropy Minimization for Solving Sudoku (2011).
17. Et al., J. L. & 2513, T. When Sudoku Comes across Information Theory (2007).
18. Prates, M. O. R. & Lamb, L. C. Problem Solving at the Edge of Chaos: Entropy, Puzzles and the Sudoku Freezing Transition. *CoRR* **abs/1810.03742.** arXiv: 1810.03742. http://arxiv.org/abs/1810.03742 (2018).
19. Newton, P. K. & DeSalvo, S. A. The Shannon entropy of Sudoku matrices. *Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences* **466,** 1957–1975 (2010).
20. Ercsey-Ravasz, M. & Toroczkai, Z. The chaos within Sudoku. *Scientific reports* **2,** 725 (2012).
21. Hanssen, V. *Sudoku puzzles* http://www.menneske.no/sudoku/nobox/3/eng/, http://www.menneske.no/sudoku/eng/top10.html. Accessed on 2019-11-6.

22. Wikipedia. *Mathematics of Sudoku* https://en.wikipedia.org/wiki/Mathematics_of_Sudoku. Accessed on 2019-11-17.

23. Of Mathematics, C. D. *The Math Behind Sudoku* http://pi.math.cornell.edu/~mec/Summer2009/Mahmood/Home.html. Accessed on 2019-11-1.

24. Felgenhauer, B. & Jarvis, F. Enumerating possible Sudoku grids. *Preprint available at http://www.afjarvis.staf* (2005).

25. Puskar, L. An Exploration of the Minimum Clue Sudoku Problem (2017).

26. enjoysudoku. *Sudoku Terminology* http://sudopedia.enjoysudoku.com/Terminology.html. Accessed on 2020-03-20.