How Many Mates Can a Latin Square Have?

Megan Bryant
mrlebla@g.clemson.edu

James Figler
figler@live.marshall.edu

Roger Garcia
garcroge@kean.edu

Yudhishthir Singh
ysingh@crimson.ua.edu

Marshall University
August 8, 2012

Acknowledgements

We would like to thank our mentor, Dr. Carl Mummert from Marshall Universty, who was an invaluable resource.

This research was conducted during the 2011 and 2012 Marshall University REU, which was supported by NSF award OCI-1005117 and by Marshall University.

The Big Green computational cluster was supported by NSF award EPS-0918949.

Introduction

Our goal was to research the upper bound on the number of mates for latin squares as a function of the size.

We used an exhaustive computational search to calculate the mate frequencies for latin squares of sizes 7 and 8.

Introduction

Our goal was to research the upper bound on the number of mates for latin squares as a function of the size.

We used an exhaustive computational search to calculate the mate frequencies for latin squares of sizes 7 and 8.

We used the data gathered from our search to formulate an algorithm that constructs mates for particular latin power squares.

This algorithm led to a new proof of a theorem regarding the existence of mates of latin power squares of size 2^n.

Latin Square

Latin squares were being studied as early as 650 BC for their supposed mystic properties. The modern definition of a latin square is as follows:

Definition

A *latin square* of order n is an array of size $n \times n$ with n symbols each of which appears exactly once in each row and each column. These symbols are most frequently denoted by the integers $1, \ldots n$.

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

Cyclic Squares

Definition

A *cyclic latin square* of order n is formed by filling the first row with symbols in any order. Fill the next row by shifting all of the symbols left one place and move the first symbol to the end. Continue like this, shifting each row one place to the left of the previous row.

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{pmatrix}$$

We denote a cyclic latin square of size n as C_n.

Since there is a cyclic square of every size n, there is a latin square of every size n.

Reduced Latin Squares

For computational purposes, we used reduced latin squares in both in our calculations and theorems.

Definition

A latin square with symbols $1, \ldots, n$ is said to be *reduced* if the first row and the first column are in the natural order $1, \ldots, n$.

Square A below is an example of a latin square in reduced form, while B and C are latin squares that are not reduced in their rows and columns, respectively.

$$\underset{A}{\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}} \quad \underset{B}{\begin{pmatrix} 1 & 3 & 2 \\ 2 & 1 & 3 \\ 3 & 2 & 1 \end{pmatrix}} \quad \underset{C}{\begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}}$$

Semireduced Latin Squares

We also considered partially reduced squares in our research.

Definition

A latin square is in *semireduced* form if the first row is in the natural order $1, \ldots, n$.

The following square is an example of a square in semireduced form, but not reduced form.

$$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

Orthogonal Latin Squares

Our research focused on counting mates of reduced latin squares.

Definition

Latin squares A and B of the same size are *orthogonal* to each other, written $A \perp B$, if every possible ordered pair is present when the squares are superimposed. A square that is orthogonal to another square is called a *mate* of that square.

The following squares are orthogonal to each other.

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

Why?

Orthogonal Latin Squares

These squares are orthogonal:

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix} \quad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

When they are superimposed, they form the following latin square, which contains all possible ordered pairs of the numbers 1, 2, and 3.

$$\begin{pmatrix} (1,1) & (2,2) & (3,3) \\ (2,3) & (3,1) & (1,2) \\ (3,2) & (1,3) & (2,1) \end{pmatrix}$$

Transversals

Whether a latin square has a mate can be determined with the use of transversals.

Definition

A *transversal* of a latin square is a list of cells, one in each row and each column, which contain all the possible symbols from the square.

A square has a mate if and only if it can be covered in non-overlapping transversals.

Transversals

A transversal is a path through a latin square where each symbol, column, and row appears exactly once.

$$\begin{pmatrix} 1 & \textcircled{2} & 3 \\ 2 & 3 & \textcircled{1} \\ \textcircled{3} & 1 & 2 \end{pmatrix}$$

A transversal (circled cells)

$$\begin{pmatrix} \boxed{1} & \textcircled{2} & \diamond 3 \\ \diamond 2 & \boxed{3} & \textcircled{1} \\ \textcircled{3} & \diamond 1 & \boxed{2} \end{pmatrix}$$

Covered with transversals

$$\begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

Mate with

$\square = 1$,
$\bigcirc = 2$,
$\diamond = 3$.

Product Operation

The *product operation* allows for the formation of a larger latin square from two smaller latin squares. The product of a size n square with a size m square is a square of size nm.

For example, the product of the following 2×2 and 3×3 squares

$$\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix} \times \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix}$$

is the 6×6 latin square:

$$\left(\begin{array}{ccc|ccc} (1,1) & (1,2) & (1,3) & (2,1) & (2,2) & (2,3) \\ (1,2) & (1,3) & (1,1) & (2,2) & (2,3) & (2,1) \\ (1,3) & (1,1) & (1,2) & (2,3) & (2,1) & (2,2) \\ \hline (2,1) & (2,2) & (2,3) & (1,1) & (1,2) & (1,3) \\ (2,2) & (2,3) & (2,1) & (1,2) & (1,3) & (1,1) \\ (2,3) & (2,1) & (2,2) & (1,3) & (1,1) & (1,2) \end{array}\right).$$

Product Operation

When the ordered pairs are replaced with the symbols $1, \ldots, n$, the following latin square is revealed.

$$\left(\begin{array}{ccc|ccc} (1,1) & (1,2) & (1,3) & (2,1) & (2,2) & (2,3) \\ (1,2) & (1,3) & (1,1) & (2,2) & (2,3) & (2,1) \\ (1,3) & (1,1) & (1,2) & (2,3) & (2,1) & (2,2) \\ \hline (2,1) & (2,2) & (2,3) & (1,1) & (1,2) & (1,3) \\ (2,2) & (2,3) & (2,1) & (1,2) & (1,3) & (1,1) \\ (2,3) & (2,1) & (2,2) & (1,3) & (1,1) & (1,2) \end{array}\right) = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 5 & 6 & 4 \\ 3 & 1 & 2 & 6 & 4 & 5 \\ 4 & 5 & 6 & 1 & 2 & 3 \\ 5 & 6 & 4 & 2 & 3 & 1 \\ 6 & 4 & 5 & 3 & 1 & 2 \end{pmatrix}$$

A product of a square with itself is a special case. It creates a *power square*.

The product of a cyclic square of size n with itself m times is denoted C_n^m.

Computation

The first phase of the project, conducted by James Figler and Yudhishthir Sigh in Summer 2011, involved the development of custom software to perform an exhaustive search for mates of reduced latin squares of size 7.

Computation

The first phase of the project, conducted by James Figler and Yudhishthir Sigh in Summer 2011, involved the development of custom software to perform an exhaustive search for mates of reduced latin squares of size 7.

In Summer 2012, Megan Bryant and Roger Garcia used the developed software to calculate the number of semireduced mates for reduced latin squares of size 8.

Our results led to the development of an algorithm and a theorem for finding semireduced mates.

Big Green

The computation was performed using Marshall University computational cluster “Big Green” and a lab of commodity PCs.

The search for size 8 would have required approximately 5.5 years of processor time if run on a single processor.

Program

The parallel program searches for semireduced mates of reduced latin squares.

Lemma

The total number of squares of size n is $n!(n-1)!$ *times the number of reduced squares.*

These additional latin squares have the same number of mates as the reduced latin square.

Program

Lemma

A semireduced mate can be used to generate $n!$ distinct mates of the same square through symbol permutation.

Thus each mate that our program found represents $n!$ distinct mates.

Finding Mates

To count the number of mates of a square, the program searched for ways to cover it with transversals.

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix} \qquad \begin{pmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{pmatrix} \qquad \begin{pmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{pmatrix}$$

A transversal (circled cells) | Covered with transversals | Mate

It counted each distinct transversal covering as a mate to the corresponding reduced latin square.

We found the use of transversals to be far more efficient in finding mates than simply testing arbitrary squares.

Results

The program generated a frequency list for the number of mates for the reduced latin squares of sizes 7 and 8.

For size 7, the program found 6 distinct semireduced mate frequencies.

Mates	*Frequency*
0	16,765,080
1	105,840
2	52,920
8	210
635	120

Table: Reduced latin squares of size 7 by number of semireduced mates

Results

In size 8 the program found 115 distinct semireduced mate frequencies compared to the 6 distinct semireduced mate for size 7.

The most frequent mate count was 0 with approximately 532 billion squares.

The largest and least frequent was $70,272$ with only 30 squares.

More than half of the frequencies were under $1,000$ semireduced mates.

Theoretical Results

Our experimental data gave us several interesting leads from which we developed an algorithm and, subsequently, a theorem.

The most important revelation was that the 30 squares which had the largest number of mates were all variations of the power square of size 8.

An algorithm was created to generate a mate for each power square of size 2^n. The algorithm begins with a 4×4 power square C_2^2 and a particular mate M_4.

Algorithm

The algorithm begins by using C_2^2 and a particular mate M_4.

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{pmatrix} \qquad \begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \\ 2 & 1 & 4 & 3 \end{pmatrix}$$

C_2^2 M_4

M_4 will serve as a blueprint to generate an 8×8 mate.

Algorithm

This algorithm will use four 2×2 matrices A_1, A_2, A_3 and A_4 for the top half of the new mate.

$$\underset{A_1}{\begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix}} \quad \underset{A_2}{\begin{pmatrix} 3 & 4 \\ 1 & 2 \end{pmatrix}} \quad \underset{A_3}{\begin{pmatrix} 5 & 6 \\ 7 & 8 \end{pmatrix}} \quad \underset{A_4}{\begin{pmatrix} 7 & 8 \\ 5 & 6 \end{pmatrix}}$$

For the bottom half another four 2×2 matrices tA_1, tA_2, tA_3 and tA_4 will be used.

$$\underset{tA_1}{\begin{pmatrix} 4 & 3 \\ 2 & 1 \end{pmatrix}} \quad \underset{tA_2}{\begin{pmatrix} 2 & 1 \\ 4 & 3 \end{pmatrix}} \quad \underset{tA_3}{\begin{pmatrix} 8 & 7 \\ 6 & 5 \end{pmatrix}} \quad \underset{tA_4}{\begin{pmatrix} 6 & 5 \\ 8 & 7 \end{pmatrix}}$$

tA_1, tA_2, tA_3 and tA_4 are transformed versions of A_1, A_2, A_3 and A_4 obtained by flipping the original four 2×2 matrices vertically and horizontally.

Algorithm

The elements of M_4 are replaced with the 2×2 matrices.

$$\begin{pmatrix} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \\ 2 & 1 & 4 & 3 \end{pmatrix} \qquad \left(\begin{array}{cccc} A_1 & A_2 & A_3 & A_4 \\ A_3 & A_4 & A_1 & A_2 \\ \hline tA_4 & tA_3 & tA_2 & tA_1 \\ tA_2 & tA_1 & tA_4 & tA_3 \end{array}\right)$$

M_4 After replacment

In the top half of the matrix above, 1s have been replace with A_1, 2s with A_2, 3s with A_3 and 4s with A_4. In the bottom half, 1s have been replace with tA_1, 2s with tA_2, 3s with tA_3 and 4s with tA_4.

Algorithm

Together the four 2×2 matrices yield an 8×8 mate.

$$
\underset{C_2^3}{\begin{pmatrix}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
2 & 1 & 4 & 3 & 6 & 5 & 8 & 7 \\
3 & 4 & 1 & 2 & 7 & 8 & 5 & 6 \\
4 & 3 & 2 & 1 & 8 & 7 & 6 & 5 \\
5 & 6 & 7 & 8 & 1 & 2 & 3 & 4 \\
6 & 5 & 8 & 7 & 2 & 1 & 4 & 3 \\
7 & 8 & 5 & 6 & 3 & 4 & 1 & 2 \\
8 & 7 & 6 & 5 & 4 & 3 & 2 & 1
\end{pmatrix}}
\qquad
\underset{M_8}{\left(\begin{array}{cc|cc|cc|cc}
1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 \\
3 & 4 & 1 & 2 & 7 & 8 & 5 & 6 \\
\hline
5 & 6 & 7 & 8 & 1 & 2 & 3 & 4 \\
7 & 8 & 5 & 6 & 3 & 4 & 1 & 2 \\
\hline
6 & 5 & 8 & 7 & 2 & 1 & 4 & 3 \\
8 & 7 & 6 & 5 & 4 & 3 & 2 & 1 \\
\hline
2 & 1 & 4 & 3 & 6 & 5 & 8 & 7 \\
4 & 3 & 2 & 1 & 8 & 7 & 6 & 5
\end{array}\right)}
$$

The lines indicate 2×2 subsquares which have been substituted for entries of M_4.

Algorithm

To make a mate of C_2^4, which is 16×16, the algorithm would begin with M_8.

Eight 2×2 matrices would be used for the top half and a transformed version for the bottom half to yield a particular mate M_{16}.

In order to continue the algorithm for a larger square of size 2^n, the new mate will be obtained from a mate of size 2^{n-1}.

An existence theorem

The algorithm developed led to a new proof of a theorem on mates of power squares of size 2^n.

Theorem

For every n, the power square C_2^n of size 2^n has at least one mate.

Remember: C_2 is a cyclic square of size 2

$$\begin{pmatrix} 1 & 2 \\ 2 & 1 \end{pmatrix}$$

Future Work

- Perform a partial search of mate frequencies for reduced latin squares of size 9×9.
- Analyze the mates of the 8x8 power square C_2^3 in more detail.
- Prove a better estimate for the number of mates of a power square as a function of the size.
- Find asymptotic bounds on the number of mates that a square can have as a function of the size.

References

J. Denes and A. D. Keedwell, *Latin squares and their applications*, Academic Press, New York, 1974.

Charles F. Laywine and Gary L. Mullen, *Discrete mathematics using latin squares*, Wiley-Interscience Series in Discrete Mathematics and Optimization, Wiley, New York, 1998.

Gary L. Mullen and Carl Mummert, *Finite fields and applications*, Student Mathematical Library, vol. 41, American Mathematical Society, Providence, RI. 2007.

Ian M. Wanless, *Transversals in latin squares*, Quasigroups and Related Systems vol. 15 n. 1, 2007.

A paper containing detailed results of this project and the code that was developed may be found at:

http://science.marshall.edu/mummertc/latin2012