LATIN SQUARES OF ORDER 10

Submitted: August 20, 1995; Accepted: August 25, 1995 (revised)

Brendan D. McKay
Department of Computer Science
Australian National University
Canberra, ACT 0200, Australia
bdm@cs.anu.edu.au

Eric Rogoyski
Cadence Design Systems, Inc.
555 River Oaks Parkway
San Jose, CA 95134, USA
rogoyski@cadence.com

Abstract.

We describe two independent computations of the number of Latin squares of order 10. We also give counts of Latin rectangles with up to 10 columns, and estimates of the number of Latin squares of orders up to 15.

Mathematics Reviews Subject Classifications: 05B15, 05-04

1. Introduction.

A $k \times n$ *Latin rectangle* is a $k \times n$ matrix with entries from $\{1, 2, \ldots, n\}$ such that the entries in each row and the entries in each column are distinct. A *Latin square* of order n is an $n \times n$ Latin rectangle.

A Latin rectangle is said to be *normalized* if the first row and first column read $[1, 2, \ldots n]$ and $[1, 2, \ldots, k]$, respectively. For example,

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ 2 & 3 & 1 & 5 & 6 & 4 \\ 3 & 5 & 4 & 6 & 2 & 1 \end{bmatrix}$$

is a normalized 3×6 Latin rectangle. It is not hard to see that the total number of $k \times n$ Latin rectangles is $n!\,(n-1)!\,L(k,n)/(n-k)!$, where $L(k,n)$ is the number of normalized $k \times n$ Latin rectangles.

The values of $L(n,n)$ for $n = 7, 8, 9$ were found by Sade [9], Wells [11], and Bammel and Rothstein [2], respectively. A recurrence for $L(3,n)$ was found by Kerewala [5], and a complicated summation for $L(4,n)$ by Athreya, Pranesachar and Singhi [1]. General formulae for $L(n,n)$ appear in [3], [8] and [10], but do not appear useful for either exact or asymptotic computation. The asymptotic value of $L(k,n)$ for $k = o(n^{6/7})$ was found by Godsil and McKay [4].

The numbers of distinct Latin squares under various symmetry operations are given to $n = 8$ in [6].

The value of $L(10,10)$ was computed independently by the two authors in 1991 and 1990, respectively, using similar but not identical methods. In chronological order, we will refer to these as the “first” and “second” computations throughout this paper. They have much in common with each other, and also with the method of [2], so we will describe them together. We wish to thank Neil Sloane for introducing us.

2. Description of the computations.

Let R be a $k \times n$ Latin rectangle. We can define an associated k-regular bipartite graph $G = G(R)$ thus: $V(G) = C \cup S$, where $C = \{c_1, c_2, \ldots, c_n\}$ and $S = \{s_1, s_2, \ldots, s_n\}$ and $E(G) = \{c_i s_j \mid \text{ column } i \text{ contains symbol } j\}$. We will call this graph the *template* of R. Clearly, many Latin rectangles may have the same template; for example, every Latin square of order n has the complete bipartite graph $K_{n,n}$ as its template.

A *one-factor* of a graph G is a spanning regular subgraph of degree one. A *one-factorization* of G is a partition of $E(G)$ into one-factors. Clearly, the rows of a Latin rectangle R correspond to the one-factors in a one-factorization of $G(R)$. For any template G, define $N(G)$ to be the number of one-factorizations of G, or equivalently the number of normalised Latin squares with template G. In forming this count, one-factorizations which differ only in the order of the one-factors are not counted separately. The value of $N(G)$ can be found from the following recursion:

$$N(G) = \sum_F N(G - F), \tag{1}$$

where the sum is over one-factors F of G which contain some fixed edge of G. The feasibility of this computation for $n = 10$ is due to the fact that $N(G)$ is an invariant of the isomorphism class of G. Thus, we need only apply (1) to one member of each isomorphism class. The challenge with efficiency is that the templates on the right need to be identified according to which isomorphism class they belong.

The two computations differed in the types of isomorphism recognised between two templates. In the first computation, isomorphisms fixing the sets C and S were used, while in the second the exchange of C and S was also permitted. In order to apply recursion (1), it is necessary to be able to identify $G - F$ from amongst the templates for which the value of $N(\,)$ is already known. In the first computation, this was achieved by defining a canonical labelling for templates. Templates were stored in canonical form, and templates $G - F$ were identified by converting them to canonical form. In the second computation, a combinatorial invariant was devised such that no two templates had the same invariant. The invariant had two components. The first component was a quickly-computed number depending on the distribution of the cycles of length 4 in $G - F$. This proved sufficient to identify the great majority of templates uniquely. For those not uniquely identified, there was a second component formed from a canonical labelling of the template, using the first author's graph isomorphism program `nauty` [7]. The set of nonisomorphic templates was determined in advance using `nauty`.

The number of distinct templates under the two definitions of equivalence for $n = 10$ and $k = 1, \ldots, 5$ were 1, 12, 1165, 121790, 601055 for the first computation, and 1, 12, 725, 62616, 304496 for the second computation.

When $N(G)$ is known for each template G, the number of normalized Latin rectangles can be determined. In terms of the second computation, we have

$$L(k, n) = 2nk!\,(n-k)! \sum_G \frac{N(G)}{|\mathrm{Aut}(G)|}, \tag{2}$$

where the sum is over all templates of degree k, and $\mathrm{Aut}(G)$ is the automorphism group of G. The reason for (2) is that $2n!\,k!\,(n-k)!/|\mathrm{Aut}(G)|$ is the number of labellings of G in which the

n	k	$L(k,n)$	n	k	$L(k,n)$	n	k	$L(k,n)$
1	1	1	7	1	1	9	1	1
2	1	1		2	309		2	16687
	2	1		3	35792		3	1034 43808
3	1	1		4	1293216		4	20 76245 60256
	2	1		5	11270400		5	11268 16430 83776
	3	1		6	16942080		6	12 95260 54043 81184
4	1	1		7	16942080		7	224 38296 79166 91456
	2	3	8	1	1		8	377 59757 09642 58816
	3	4		2	2119		9	377 59757 09642 58816
	4	4		3	1673792	10	1	1
5	1	1		4	4209 09504		2	1 48329
	2	11		5	27206 658048		3	81549 99232
	3	46		6	33 53901 89568		4	14717 45210 59584
	4	56		7	53 52814 01856		5	746 98838 30762 86464
	5	56		8	53 52814 01856		6	8 70735 40559 10037 09440
6	1	1					7	1771 44296 98305 41859 22560
	2	53					8	42920 39421 59185 42730 03520
	3	1064					9	75807 21483 16013 28114 89280
	4	6552					10	75807 21483 16013 28114 89280
	5	9408						
	6	9408						

Table 1. Numbers of normalized Latin rectangles.

neighbours of c_1 are $\{s_1, \ldots, s_k\}$, and $(n-1)!$ is removed to allow for normalization of the first row. In the case of $k = n$, equation (2) simplifies to $L(n,n) = N(K_{n,n})/(n-1)!$.

Each method requires a few days on a fast workstation. The results are listed in Table 1. To obtain the total number of Latin rectangles, not necessarily normalized, multiply $L(k,n)$ by $n!\,(n-1)!/(n-k)!$.

3. Estimates.

For $n = 11$ and $k = 1, \ldots, 5$, the numbers of distinct templates under the weaker of our two definitions of isomorphism are 1, 14, 7454, 5582612, 156473848. Allowing interchange of C and S gains almost a factor of two, but still the numbers are too large for $L(11,11)$ to be computable by our method in a reasonable time. However, we can find approximate values for higher order using a probabilistic method.

Suppose we form a "random" normalized Latin square with rows $R_1, \ldots, R_n$ by this process: R_1 is the usual first row for a normalized square. For $i = 2, \ldots, n$, R_i is chosen uniformly at random from amongst those extensions of $[R_1, \ldots, R_{i-1}]$ which have i in the first position. For $i = 1, \ldots, n-1$, let e_i be the total number of such extensions of $[R_1, \ldots, R_{i-1}]$. Then it is easy to show that $e_1 e_2 \cdots e_{n-1}$ is an unbiased estimator of $L(n, n)$. (This is an example of a method originally due to Knuth.) For best experimental efficiency, we computed e_i exactly for $i > n - 8$ and by testing random permutations for smaller i.

In Table 2, we present our estimates and the number of trials used for each n. It is not possible to be precise about the accuracy, but we feel that probably these numbers are correct to within one value of the least significant digit.

n	*trials*	$L(n, n)$
11	1000000	5.36×10^{33}
12	1100000	1.62×10^{44}
13	400000	2.51×10^{56}
14	200000	2.33×10^{70}
15	20000	1.5×10^{86}

Table 2. Estimates of $L(n, n)$ for larger n.

References.

[1] K. B. Athreya, C. R. Pranesachar and N. M. Singhi, On the number of Latin rectangles and chromatic polynomial of $L(K_{r,s})$, *European J. Combinatorics*, **1** (1980) 9–17.

[2] S. E. Bammel and J. Rothstein, The number of 9×9 Latin squares, *Discrete Math.*, **11** (1975) 93–95.

[3] I. Gessel, Counting Latin rectangles, *Bull. Amer. Math. Soc.*, **16** (1987) 79–83.

[4] C. D. Godsil and B. D. McKay, Asymptotic enumeration of Latin rectangles, *J. Combinatorial Theory, Ser. B*, **48** (1990) 19–44.

[5] S. M. Kerewala, The enumeration of Latin rectangle of depth three by means of difference equation, *Bull. Calcutta Math. Soc.*, **33** (1941) 119–127.

[6] G. Kolesova, C. W. H. Lam and L. Thiel, On the number of 8×8 Latin squares, *J. Combinatorial Theory, Ser. A*, **54** (1990) 143–148.

[7] B. D. McKay, nauty users' guide (version 1.5), Technical Report TR-CS-90-02, Computer Science Dept., Australian National University, 1990.

[8] J. R. Nechvatal, Asymptotic enumeration of generalised Latin rectangles, *Utilitas Math.*, **20** (1981) 273–292.

[9] A. Sade, Enumeration des carrés latins. Application au 7^e ordre. Conjecture pour les ordres supérieurs, Marseille, 1948, 8pp.

[10] Jia-yu Shao and Wan-di Wei, A formula for the number of Latin squares, *Discrete Math.*, **110** (1992) 293–296.

[11] M. B. Wells, The number of Latin squares of order eight, *J. Combinatorial Theory*, **3** (1967) 98–99.