The Search for a Finite Projective Plane of Order 10

C. W. H. Lam *
Computer Science Department
Concordia University
Montréal Québec Canada H3G 1M8

November 30, 2005

Dedicated to the memory of Herbert J. Ryser

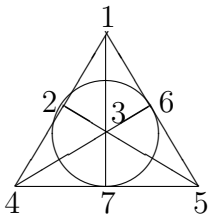
1 Prologue

When I was a graduate student looking for a thesis topic, Herbert Ryser advised me not to work on the projective plane of order 10. Even though he was extremely interested in this subject, he believed that it was too difficult and that I might get nowhere with it. I took his advice and chose another problem. Somehow, this problem has a beauty that fascinates me as well as many other mathematicians. Finally in 1980, I succumbed to the temptation and started working on it with some of my colleagues. We eventually managed to get somewhere, but unfortunately, Dr. Ryser is no longer with us to hear of the final result. This is an expository article describing the evolution of the problem and how computers were used to solve it.

2 The History of the Problem

A *finite projective plane of order* n, with $n > 0$, is a collection of n^2+n+1 lines and n^2+n+1 points such that

*This work was supported in part by the Natural Sciences and Engineering Research Council of Canada under Grants A9373, A9413, 0011 and by the Fonds pour la Formation de Chercheurs et l'Aide à la Recherche under Grants EQ2369 and EQ3886.



$$
\begin{aligned}
L1 &= \{1, 2, 4\} \\
L2 &= \{2, 3, 5\} \\
L3 &= \{3, 4, 6\} \\
L4 &= \{4, 5, 7\} \\
L5 &= \{5, 6, 1\} \\
L6 &= \{6, 7, 2\} \\
L7 &= \{7, 1, 3\}
\end{aligned}
$$

Figure 1: The finite projective plane of order two

1. every line contains $n + 1$ points,

2. every point is on $n + 1$ lines,

3. any two distinct lines intersect at exactly one point, and

4. any two distinct points lie on exactly one line.

There are several different definitions for a finite projective plane and this set of axioms is chosen to highlight the striking duality of lines and points. One can interchange the words "line" and "point" in the definition and obtain essentially the same axioms! The attractiveness of these objects is in their simplicity and their reliance on the language of geometry. One is tempted to start drawing lines on paper and may soon discover some simple examples.

The smallest example of a finite projective plane is a triangle, the plane of order one. The smallest non-trivial example is of order 2, as shown in Fig. 1. There are seven points labelled from 1 to 7. There are also seven lines labelled $L1$ to $L7$. Six of them are straight lines but $L6$ is represented by the circle through the points 2, 6, and 7. The reader is invited to verify that the axioms of a finite projective plane are satisfied.

An early reference to a finite projective plane is in the paper by Veblen [32], which studied the axioms for geometry and used the plane of order 2 as an example. Veblen also proved that this plane of order 2 cannot be drawn using only straight lines. In a series of

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

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

Figure 2: Two orthogonal latin squares of order four

11	22	33	44
23	14	41	32
34	43	12	21
42	31	24	13

Figure 3: A Graeco-Latin square of Order 4

papers [32, 33, 34], Veblen, Bussey and Wedderburn established the existence of most of the planes of small orders, as well as all four non-isomorphic planes of order 9. One of the orders missing is $n = 6$.

In 1938, Bose [4] explained why there is no projective plane of order 6. He related the existence of a finite projective plane of order n to the existence of a hyper-Graeco-Latin square. The term "Graeco-Latin square" originated with Euler in 1782. In our modern terminology, they are called orthogonal Latin squares. Let us define them.

A *Latin square* of order n is an $n \times n$ matrix satisfying the following properties:

1. all the entries are integers between 1 and n,

2. in every row, no entry is repeated, and

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

Figure 4: A Latin square orthogonal to those in Fig. 2

3. in every column, no entry is repeated.

Let $S_1 = [s_{ij}^{(1)}]$ and $S_2 = [s_{ij}^{(2)}]$ denote two Latin squares of order n. They are said to be *orthogonal* provided that the n^2 2-samples $(s_{ij}^{(1)}, s_{ij}^{(2)})$ for $i, j = 1, 2, \ldots, n$ are distinct. A simple way to visualize the definition is to put the second square, slightly shifted, on top of the first square. The resulting n by n array of 2-samples has no repeated entries, and is often referred to as a *Graeco-Latin square* or an *Euler square*.

Figure 2 contains two examples of Latin squares of order 4. They are orthogonal and the resulting Graeco-Latin square is shown in Fig. 3. Is there a third Latin square orthogonal to both the squares in Fig. 2? Yes, there is one, as shown in Fig. 4. Can one find a fourth? The answer is no. One can easily prove the following theorem [26, p. 80].

Theorem 1 *Let $S_1, S_2, \ldots, S_t$ be a set of t mutually orthogonal Latin squares of order $n \geq 3$. Then*

$$t \leq n - 1. \tag{1}$$

If equality holds in (1), then the orthogonal set is said to be *complete*. Finally, we are ready to state Bose's result [26, p. 92].

Theorem 2 *Let $n \geq 3$. We may construct a projective plane of order n if and only if we may construct a complete set of $n - 1$ mutually orthogonal Latin Squares of order n.*

Why does Bose's result explain the non-existence of a projective plane of order 6? It states that such a plane exists if and only if there exists a complete set of 5 mutually orthogonal Latin squares of order 6. The possible existence of even a pair of orthogonal Latin squares of order 6 was a much older problem.

In a 1782 paper [12], Euler started by stating the problem of the 36 officers. This problem asks for an arrangement of 36 officers of 6 ranks and from 6 regiments in a square formation of size 6 by 6. Each vertical and each horizontal line of this formation is to contain one

and only one officer of each rank and one and only one officer from each regiment. Euler denoted the 6 regiments by the Latin letters a, b, c, d, e, f and the 6 ranks by the Greek letters $\alpha, \beta, \gamma, \delta, \epsilon, \zeta$. He further remarked that the characteristic of an officer was determined by the two letters, one Latin and the other Greek, and that the problem consists of arranging the 36 combinations of two letters in a square in such manner that every row and every column contains the six Latin as well as the six Greek letters. This was the origin of the term "Graeco-Latin square". Euler observed that the first step was to arrange the Latin letters in a square so that no letter was missing either from any row or from any column. He called this square a Latin square. If he had chosen to arrange the Greek letters instead, then we would probably have Graeco squares rather than Latin squares. In any case, if we label both the ranks and the regiments from 1 through 6, then Euler's problem reduces to the construction of a pair of orthogonal Latin squares of order 6.

Euler found no solution to this particular problem. He then conjectured that no solution exists if the order of the square is of the form $n \equiv 2 \pmod 4$. This is the famous Euler's conjecture. The first case $n = 2$ is trivially impossible. Tarry around 1900 [27] verified by a systematic enumeration that Euler's conjecture holds for $n = 6$. Since there does not exist even a pair of orthogonal Latin squares, Bose's result implies the non-existence of a projective plane of order 6.

Yet, there is something unpleasant about a systematic hand enumeration: it is messy and it is error prone. Mathematicians did find a better explanation in the celebrated Bruck-Ryser theorem [7], which was published in 1949.

Theorem 3 (Bruck-Ryser) *If $n \equiv 1, 2 \pmod 4$, then a necessary condition for the existence of a finite projective plane of order n is that integers x, y exist satisfying $n = x^2 + y^2$.*

Since $6 \equiv 2 \pmod 4$ and it is not a sum of two integer squares, the Bruck-Ryser theorem implies the non-existence of a projective plane of order 6. How did they prove the theorem?

	1	2	3	4	5	6	7
L1	1	1	0	1	0	0	0
L2	0	1	1	0	1	0	0
L3	0	0	1	1	0	1	0
L4	0	0	0	1	1	0	1
L5	1	0	0	0	1	1	0
L6	0	1	0	0	0	1	1
L7	1	0	1	0	0	0	1

Figure 5: An incidence matrix for the plane of order two

We will not repeat it here, but one of the crucial steps involved the use of an incidence matrix.

The *incidence matrix* $A = [a_{ij}]$ of a projective plane of order n is an $n^2 + n + 1$ by $n^2 + n + 1$ matrix where the columns represent the points and the rows represent the lines. The entry a_{ij} is 1 if point j is on line i; otherwise, it is 0. For example, Fig. 5 gives the incidence matrix for the projective plane of order 2. In terms of an incidence matrix, the properties of being a projective plane are translated into:

1. A has constant row sum $n + 1$,

2. A has constant column sum $n + 1$,

3. the inner product of any two distinct rows of A is 1, and

4. the inner product of any two distinct columns of A is 1.

The conditions on row sums and row inner products can be encapsuled in the following matrix equation:

$$AA^T = nI + J, \tag{2}$$

where A^T denotes the transpose of the matrix A, I denotes the identity matrix, and J the matrix of all 1's. Every diagonal entry on the right hand side of Eq. (2) is $n + 1$, which implies that the inner product of any row of A with itself is $n + 1$ or, equivalently, its row

sum is $n+1$. Every off-diagonal entry is 1, which is the same as requiring the inner product of any two distinct rows of A to be 1. Ryser also proved that A is a normal matrix [25]; in other words,

$$AA^T = A^T A.$$

Hence, Eq. (2) also implies the conditions regarding column sums and column inner products. The Bruck-Ryser theorem starts from this equation and proves that it implies n is a sum of two integer squares when $n \equiv 1, 2 \pmod 4$.

It is surprising that the Bruck-Ryser theorem has a partial converse [13].

Theorem 4 *If $n \equiv 0, 3 \pmod 4$ or if $n \equiv 1, 2 \pmod 4$ and $n = x^2 + y^2$, then there exists a rational matrix A satisfying equation (2).*

Of course, if the matrix A is actually the incidence matrix of a projective plane, then it must have entries which are either 0 or 1. Although this theorem guarantees only a matrix with rational entries, it suggests that the necessary part of the Bruck-Ryser theorem is very close to being also sufficient.

Projective planes are special cases of a class of combinatorial objects called symmetric block designs. We are not going to discuss block designs, except to mention that Chowla and Ryser have generalized the Bruck-Ryser theorem to symmetric block designs [10], which it is now known as the Bruck-Ryser-Chowla theorem. Here again, a partial converse exists, providing more credence to the hope that the conditions in the Bruck-Ryser-Chowla theorem are both necessary and sufficient. This hope is now shattered by the non-existence of the finite projective plane of order 10.

Let us return to the history of projective planes. Now that we have a good explanation of the non-existence of a plane of order 6, what is the next unknown case? It is $n = 10$. Since $10 = 1^2 + 3^2$, a plane of order 10 would exist if the necessary condition of the Bruck-Ryser theorem is also sufficient. On the other hand, $10 \equiv 2 \pmod 4$, and so, if one believes

Euler's conjecture, then it does not exist.

First, Euler's conjecture was shown to be false. In 1959, Bose and Shrikhande [5] constructed a pair of orthogonal Latin squares of order 22. Then Parker [23, 24] constructed a pair of orthogonal Latin squares of order 10. Together they showed that Euler's conjecture is false for all orders greater than six [6]. This raised the hope for the existence of a plane of order 10.

History indicated that significant advances were made when one branch of mathematics was shown to be related to a different branch of mathematics. It is not surprising that the beginning of the end of the plane of order 10 occurred when people started studying the binary error-correcting code associated with it.

3 The Beginning of the End

Let A be the incidence matrix of a finite projective plane of order 10. Let V be the vector space generated by the rows of A over $\mathbf{F}_2$, the finite field with two elements $\{0,1\}$. A vector in V is called a *codeword*. The *weight* of a codeword is the number of 1's in the codeword. Let w_i be the number of codewords of weight i. We define the *weight enumerator* of V to be

$$\sum_{i=0}^{111} w_i x^i.$$

In 1970, Assmus gave a talk at Oberwolfach entitled "The projective plane of order ten?" which discussed the properties of V. After the talk, there was a lot of anticipation about this new approach. Maybe one could derive a contradiction such as showing that one of the weights is non-integral or negative. Unfortunately, no such simple contradictions were found.

However, there were several important advances. Assmus and Mattson showed [2] that the weight enumerator is uniquely determined by w_{12}, w_{15}, and w_{16}. Since their report is not readily available in most libraries, we shall refer to the paper of MacWilliams, Sloane,

and Thompson [22], which proved many of the same results. One of the many innocuous but extremely useful results was:

Theorem 5 *Let l be any line of the plane, and v any codeword of V. Then*

$$|v \cap l| \equiv |v| \pmod 2.$$

For example, every line must intersect a codeword of even weight in an even number of points. This gives us an extra condition beyond what is available from the definition of a projective plane. Roughly speaking, this condition reduces the number of possibilities for each line by half. The cumulative effect of this condition is tremendous and it is the main reason that makes an exhaustive search possible.

Furthermore, MacWilliams *et al.* showed that $w_{15} = 0$ after using about 3 hours of computer time on a General Electric 635. Bruen and Fisher later showed in [8], that $w_{15} = 0$ also followed from an earlier computer result by Denniston [11]. However, the method of MacWilliams *et al.* illustrated how to continue attacking the problem. This can be summarised as follows:

> Given any weight i, we assume that a codeword of weight i exists. By considering the intersection patterns of a few selected lines with the i points of this codeword, we arrive at a small number of starting configurations, each corresponding to a submatrix of the incidence matrix. Then, we try to complete the rest of the incidence matrix. If we succeed, then it is time to celebrate because we have constructed a plane. If none of the starting configurations can be so completed, then the plane of order 10 does not contain any codeword of weight i and $w_i = 0$.

This method requires first the generation of all the possible starting configurations. A good reference is the 1980 paper [14] by Marshall Hall, Jr., which analyzed in detail the starting configurations for codewords of small weight $i \leq 20$. Given a starting configuration,

the attempt to complete it is basically a backtracking process. The term "backtrack" was coined by D. H. Lehmer in the 1950's, but backtrack techniques have been used to solve puzzles for a long time. It is a tedious and lengthy task, one that is best suited for a computer.

3.1 Backtrack Search using a Computer

The first description of a generalized backtrack search algorithm was in [35]. Let us first define a restricted version of the search problem.

> **Search problem**:
>
> Given a collection of sets of candidates $C_1, C_2, C_3, \ldots, C_m$ and a boolean *compatibility* function $P(x, y)$ defined for all $x \in C_i$ and $y \in C_j$, find an *m-tuple* $(x_1, \ldots, x_m)$ with $x_i \in C_i$ such that $P(x_i, x_j)$ is true for all $i \neq j$.

The m-tuple satisfying the above condition is called a *solution*. The term *compatibility* was first introduced by Carter in 1974 [9]. This version is restricted because the compatibility function is only defined on ordered pairs rather than on all k-tuples, $1 \leq k \leq m$.

For example, if we take $m = n^2 + n + 1$ and let C_i be the set of all candidates for column i of the incidence matrix of a projective plane, then $P(x, y)$ can be defined as

$$P(x, y) = \begin{cases} \text{true} & \text{if } \langle x, y \rangle = 1 \\ \text{false} & \text{otherwise,} \end{cases}$$

where $\langle x, y \rangle$ denotes the inner product of columns x and y. A solution is then a complete incidence matrix.

In a backtrack approach, we generate *k-tuples* with $k \leq m$. A *k-tuples* $(x_1, \ldots, x_k)$ is a *partial solution* at level k if $P(x_i, x_j)$ is true for all $i \neq j \leq k$. The basic idea of the backtrack approach is to *extend* a partial solution at level k to one at level $k + 1$, and if this extension is impossible, then to go back to the partial solution at level $k - 1$ and attempt to generate a different partial solution at level k.

A nice way to organize the information inherent in the partial solutions is the backtrack search tree. Here, the empty partial solution ( ) is taken as the root, and the partial solution $(x_1, \ldots, x_{k-1}, x_k)$ is represented as the child of the partial solution $(x_1, \ldots, x_{k-1})$. Following the computer science terminology, we often called a partial solution a *node*. It is often true that the computing cost of processing a node is independent of its level k. Under this assumption, the total computing cost of a search is equal to the number of nodes in the search tree times the cost of processing a node.

The number of nodes in the search tree can often be reduced by using the *symmetry* or *property-preserving* operations. If A is the incidence matrix of a projective plane, then the operations of permuting the rows and permuting the columns of A correspond only to reordering the lines and points of the plane. These operations preserve the property of being a projective plane. To see how they reduce the size of a search tree, consider the plane of order 10. There are $\binom{111}{11} \approx 4.7 \times 10^{14}$ choices for the first column, corresponding to the number of ways of placing 11 ones in the 111 rows. By using the row permutations, we can assume that all these ones are placed on the first 11 rows, reducing the number of partial solutions (x_1) from 4.7×10^{14} to 1. Mathematicians love to use the phrase "without loss of generality" to indicate a simplification by symmetry operations. So without loss of generality, the second column has only one choice — with a one in the first row and the remaining 10 ones in rows 12 to 21. Now, row 1 has nine remaining ones. By permuting columns, we can assume that these remaining ones of row 1 are in columns 3 to 11. Next, by row permutation, the remaining 10 ones of column 3 can be placed in rows 22 to 31. Continuing in this manner, it is not difficult to show that there is only one choice up to column 21. Beyond this point, symmetry operations are difficult to visualize, because they often involve combinations of row and column permutations.

This process of reducing the search tree by using symmetry operations is called *isomorph rejection*. Two partial solutions are said to be *isomorphic* if there is a symmetry operation

mapping one to the other. To determine whether a solution exists, it is only necessary to extend the non-isomorphic partial solutions, because if a partial solution can be extended to a complete solution, then every other isomorphic partial solution can also be so extended. The testing of whether two partial solutions are isomorphic is called *isomorphism testing*. The set of symmetry operations mapping a partial solution to itself forms an *automorphism group*. In the context of projective planes, an automorphism of a complete solution is also called a *collineation*.

The efficient use of the symmetry operations is one of the most difficult problem in a backtrack search. It is both a blessing and a curse — a blessing because there is always hope of further optimization and a curse because it is the source of many programming errors. However, following the method of MacWilliams *et al.*, it is clear that the existence question of the plane of order 10 is headed towards a computer-based solution and that symmetry consideration is a necessary tool for efficiency reasons.

In his 1974 Ph. D. thesis, Carter picked up where MacWilliams *et al.* had left off. He showed that there are six possible starting configurations associated with a codeword of weight 16. By using the computer, he managed to eliminate four of the cases, as well as one subcase of the fifth. He used a total of about 100 hours of computer time, mostly on a CDC 6600 at the Institute for Defense Analyses (IDA) in Princeton, New Jersey, and also on a CDC 7600 at the Lawrence Radiation Laboratories in Berkeley, California. His search investigated about 3×10^8 nodes, or about 10^3 nodes per second.

This was the situation when we entered the picture. While we knew $w_{15} = 0$, the search for weight 16 codewords was about three-quarter done and the search for weight 12 codewords was presumed to be too difficult. The person who suggested the problem to us was John Thompson.

4 The Home Stretch

Around 1980, we purchased a share in a VAX-11/780 with the intention of running long mathematical programs during the off-hours when it was otherwise unused. In the same year, Thompson sent John McKay two papers [28, 29] outlining a connection between a codeword of weight 12 and a set of fixed-point free involutions. When McKay showed me the papers, I was finally hooked. We had a computer looking for a problem and the search for codewords of weight 12 was a problem looking for a computer. I started by writing a simple computer program to play with the ideas. Little did I know that it would become a 9 year undertaking!

We realised very early that the weight 12 case was going to take a lot of computer time. So I showed my simple program to Larry Thiel, who has a reputation of making any computer program run faster. Right away, he saw ways of speeding it up. He and Stanley Swiercz wrote most of our computer programs.

We realised also that it is important to estimate how long a computer search would take. It would be useless to speed up a program by a factor of ten, if the resulting program still requires 100 years to run. However, speeding it up by a factor of 1000 would make the search feasible in our environment. By adapting Knuth's Monte Carlo method [16], we estimated that the search tree had about 4×10^{11} nodes. Hoping that we could process 10^5 nodes per second, we arrived at an estimate of 50 days of computer time.

I was going to present the final results at the Ninth Australian Conference on Combinatorial Mathematics in 1981, but the program development was slower than expected. At the conference, I had to quickly change my plan and talked about a feasibility study of such a search [17]. The program was finally developed and the search finished late in 1982. We did not find a completed incidence matrix and so, $w_{12} = 0$ [18]. Ryser was happy with the result and encouraged us to continue. Hall was both excited and pessimistic. He wrote, "For the

first time, I doubt that a plane of order 10 exists."

The search for weight 12 codewords took 183 days instead of the predicted 50. The main reason for the discrepancy was that we could process only about 3×10^4 nodes per second instead of 10^5. Yet, we learned a lot about the planning of the search, the estimation process, and the optimization techniques — knowledge which became useful in our later programs. In retrospect, we probably could have reduced the size of the search tree by another factor of 10 and come within the predicted time.

Even before the search for the weight 12 codeword was finished, we were looking for a way to solve the whole problem. One possibility is to determine w_{16} and then the weight enumerator. Unless the weight enumerator provides a contradiction, we still have to perform another search. Hall in [14] proved that if $w_{12} \leq 1211$, then there exist primitive weight 20 codewords, which intersect the lines of the plane in at most four points. We might be able to skip the step of finding w_{16} if we could go directly to weight 20. Unfortunately, we obtained an estimate of at least 10^{13} weight 20 starting configurations. Even though, as Hall remarked, it required only a few minutes of computer time to try each starting configuration, a few minutes times 10^{13} equaled 10^7 years — clearly beyond our capability. There was no short cut which could avoid computing the weight enumerator.

During my visit to the University of Cambridge in 1982, Thompson showed me Carter's thesis and suggested that we might be able to finish the remaining cases. It was the first time I read his thesis and I was surprised by how familiar his computing techniques were. We had rediscovered a lot of what was already in his thesis! Thompson also made me promise to redo Carter's work on weight 16, just to have an independent verification of its correctness. Although I have yet to do it, I still intend to fulfil this promise.

In the mean time, Larry had developed an extremely useful prototyping program, called NPL. It had a very humble beginning. Working with incidence matrices meant that we often had to draw them and fill in large numbers of zeros and ones. As a labour saving move, we

started drawing templates and then making photocopies. Larry took this one step further, printing the matrices with a computer rather than drawing the templates by hand. Soon we included the capability of initializing the starting configurations and later, the ability to backtrack was also included. Before long, it became a full-blown prototyping program, capable of performing estimations and adapting to new configurations. It turned out to be an extremely useful tool. We could quickly explore an idea for an "improvement" to the search. Ideas that turned out to increase the required CPU time were discarded. We could use it to plan our final search in detail, without having to write the program first. The most exciting stage was this planning stage.

Using NPL, we found a feasible way to solve the remaining cases of Carter. Compared to Carter's work, we found a better search order and a better use of the symmetry, both reducing the size of the search tree. Carefully optimised programs to implement the planned attack were written for each of the remaining cases. Running on two VAX computers, it took the equivalent of 80 days of computing on a VAX-11/780 [19]. No completed incidence matrix was found and hence $w_{16} = 0$.

Now that w_{12}, w_{15} and w_{16} were all known to be zero, we could compute the weight enumerator of the binary error-correcting code for the plane of order 10 [20]. What catches the eye is that $w_{19} = 24,675$. If one follows the same method of finding starting configurations and trying to extend them, then one will either construct the plane or show that it does not exist.

Even before we finished our search for the weight 16 case, Thompson was already busy at work. In a letter dated October 24, 1983, he enclosed a copy of his unpublished note [30] with a list of 82 starting configurations for the weight 19 case. He constructed them by hand and had not checked for pairwise inequivalence. He ended the letter with "I hope you'll press on with this question, although perhaps I am too greedy".

It was a challenge that we could not refuse. I quickly wrote up a program and found

65 starting configurations. Larry also upgraded his NPL program to include isomorphism testing and found 66 configurations! After cross checking our results, I found that his numbers were correct. We also reconciled our numbers with those of Thompson. At last, we could see the finish line at a distance.

By simple counting arguments, 17 of the 66 cases can easily be eliminated. Four other cases are eliminated in a more *ad hoc* manner. Scott Crossfield, working with me as an NSERC Summer Undergraduate Assistant, obtained estimates for each of the remaining 45 cases. Our preliminary results indicated that the problem could be solved with another two years of computing time. These arguments and estimation results were presented in [19].

We decided to solve these cases by starting from the easy ones and hoping to discover better methods for the more difficult ones. We hedged our preliminary estimates by saying that it might be too optimistic. Our estimate of how fast the computer could test the acceptability of a column was based on our experience of the weight 12 and 16 cases. For weight 19, this test was much slower for technical reasons which were related to the fact that 19 is an odd number whereas the other two numbers are even. We could proceed only about 60 nodes per second, giving an estimate of 100 years to finish the search. To make the search feasible, we needed to speed it up by another factor of 100. The only possibility left was to use a faster computer, bringing to mind a supercomputer. Fortunately, our method could be easily adapted to use the vectoring capability of a supercomputer [31]. It was time to ask for help. Both Hall and Thompson suggested that IDA might be willing because it had helped Carter previously. After some discussion with Nick Patterson, the deputy director of the Communication Research Division at IDA, he agreed to run our yet undeveloped program on their CRAY-1A as the lowest priority job using up the otherwise idling computer. He and Douglas Wiedemann, who was on leave from IDA pursuing graduate studies at the University of Waterloo, gave us many suggestions on the efficient usage of the CRAY. In addition, Patterson looked after the day-to-day running of the program for over two years.

He was the unsung hero in the successful completion of our work.

In the mean time, Concordia had acquired more and more VAX computers. Soon, we were running our program on five different computers. We figured that there was enough computing resources at Concordia to solve all but a few of the most difficult cases. So we targeted the CRAY program for only one type of starting configuration which contained all the most difficult cases. The other configurations were left for our VAX computers to solve. All the cases at Concordia finished around January 1987 and took the equivalent of 800 days of VAX-11/780 computer time.

There are many considerations that go into developing a computer program which runs for months and years. Interruptions, ranging from power failures to hardware maintenance, are to be expected. A program should not have to restart from the very beginning for every interruption; otherwise, it may never finish. To solve the restarting problem, we partitioned our starting configurations into smaller cases and a message was output onto a log file every time one of them was finished. If there was an interruption, we just looked up the last completed case and continued from that point. For our programs, a convenient subdivision was at the "$A2$" boundary; more details can be found in [19, 21]. The messages written on the log file also contained statistics about the run to allow comparison and verification of the results.

In order to avoid any unpleasant surprises involved in running programs at a long distance, we also put in a number of safety checks. Most of them were related to testing whether the sizes of the internal data structures were large enough. For lack of memory space, one could not merely allocate the largest expected size for every structure. Instructions were given on how to selectively increase the sizes. Some data structures were so specialized that their size could not be increased. We temporarily ignored the problem by deciding to handle them later, if necessary. Of course, we hoped that it would not be necessary.

We finished developing our CRAY program in 1986 and we were relieved to obtain a

final estimate of about 3 months of CRAY computer time. It started running at IDA in the fall of 1986 and was to await completion in two to three years.

5 The Finish Line

On November 11, 1988, Patterson called and said that the run was finished. Always careful, he proposed to verify that there was a record of each subcase before sending the results on a magnetic tape back to us. Somehow, the news of the end the plane of order 10 was spread around the combinatorics community and we were deluged with inquiries. On November 18, Patterson called again with some bad news. There was an error number 4 for one of the $A2$'s. What is an error number 4? It had been two years since we last looked at our CRAY program. Taking out an old listing, it took us a while to determine that error number 4 was a size problem for a data structure that could not be enlarged. This $A2$ could not be solved with the existing CRAY program! If it had been able to handle this $A2$, it would have taken about 30 seconds. The only other program around that could handle this case was our slow but adaptable NPL program. After further dividing this $A2$ into about 200 smaller subcases, it took NPL one day to solve one of them, giving an estimate of 200 days just to solve this $A2$! Meanwhile, Barry Cipra from Science called and said he would like to write an article about the non-existence of a plane of order 10. What should we tell everybody?

We decided that it was prudent to solve the problem first. We found that, with a small modification of the CRAY program, it could handle all but one of the subcases. So, we ran this offending subcase using NPL and bypassed the CRAY. The plane of order 10 was again dead on November 29, 1988.

There was a surprising amount of public interest in the non-existence of a finite projective plane of order 10. Besides Science, it was also reported in the New York Times and in Scientific American. One often asked question is, "How much computer time it took on the CRAY?" Unfortunately, we did not keep track of the computer time used on the CRAY,

because we thought it was a useless figure other than demonstrating the difficulty of the problem. We were more interested in keeping statistics that might be useful for a future verification by someone else. So, we had to make an educated guess and said 3,000 hours. Later, Patterson suggested that it was probably closer to 2,000 hours. So, our CRAY program was proceeding nodes at a rate of about 2×10^4 nodes per second.

After the burst of publicity, we finally managed to read the magnetic tape containing the statistics. To our horror, we found another $A2$ with an error number 4. However, we knew exactly what to do this time and there was no panic. It was handled exactly the same way as the previous one. By the end of January 1989, the plane of order 10 was dead a third and hopefully the final time.

6 Is This Really the End?

This is not the first time that a computer has played an important role in "proving" a theorem. A notable earlier example is the four-color theorem [3]. Yet, these are not proofs in the traditional mathematical sense. It is impossible for any human being to check through all the calculations. From personal experience, it is extremely easy to make programming mistakes. We have taken many precautions, including the use of two different programs to cross check selective sample cases and the checking of internal consistency when isomorphism testing is performed. Yet, I want to emphasize that this is only an experimental result and it desperately needs an independent verification, or better still, a theoretical explanation.

There is, moreover, the possibility of an undetected hardware failure. A common error of this type is the random changing of bits in a computer memory, which could mean the loss of a branch of the search tree. This is the worst kind of hardware error, because we might loose solutions without realizing it. The CRAY-1A is reported to have such errors at the rate of about one per one thousand hours of computing. At this rate, we expect to encounter two to three errors! We did discover one such error by chance. After a hardware problem,

Patterson reran the 1,000 $A2$'s just before the failure and the statistics have changed for the $A2$ processed just prior to the malfunction. How should one receive a "proof" that is almost guaranteed to contain several random errors?

Unfortunately, this is an unavoidable nature of a computer-based proof — it is never absolute. However, despite this reservation, we argued in [21] that the possibility of hardware errors leading us to a wrong conclusion is extremely small. Since each $A2$ is in a separate run and there are about half a million non-isomorphic $A2$'s, the probability of one random hardware error affecting one specific $A2$ is about 2×10^{-6}. Suppose we accept that the weight enumerator is correct. Then if an undiscovered plane of order 10 exists, it would contain 24,675 weight 19 codewords. The 19 points in each such codeword give rise to an $A2$. Since we have searched through all non-isomorphic $A2$'s, we must have encountered these special $A2$'s. If all these 24,675 special $A2$'s are isomorphic, then only one out of about half a million non-isomorphic $A2$'s can be extended to the undiscovered plane. Even under this assumption, the probability of this special $A2$ being affected by two or three undetected hardware errors is less than 10^{-5}. Is it likely that all 24,675 $A2$'s arising from an undiscovered plane are isomorphic? Since the plane is known to have a trivial collineation group [1, 15, 36], it is more likely that there are two or more non-isomorphic $A2$'s amongst the 24,675 cases. In this situation, the probability of hardware errors affecting all of them is infinitesimal. The same argument can be used even if we do not assume the correctness of previous computer-based results. Basically, the argument depends on the observation that if a plane of order 10 exists, then it can be constructed from many different starting points. Random hardware failures are unlikely to eliminate all of them. In other words, the fact that no one has yet constructed one is a very strong indication that it does not exist.

7 Epilogue

While we were tracing the origin of the existence problem of the plane of order 10, we asked Dan Hughes, who has worked in this area for a long time and is famous for the Hughes planes which are named after him. He recounted the following story. In about 1957, at a Chinese restaurant in Chicago, Reinhold Baer, another mathematician well known for his work in group theory and projective planes, was trying to impress the younger Hughes by remarking that if the plane of order 10 was settled by a computer, he hoped not be alive to see it. Baer got his wish but I do not think Herb Ryser shared this opinion. Ryser was happy that the weight 12 case was settled by a computer. I can only extrapolate and hope that he would also be happy that the whole problem has been "settled", even if by a computer.

References

[1] R. P. Anstee, M. Hall, Jr., and J. G. Thompson, "Planes of order 10 do not have a collineation of order 5," J. Comb. Theory, Series A, Vol. 29(1980), p. 39–58.

[2] E. F. Assmus, Jr. and H. F. Mattson, Jr., "On the Possibility of a Projective Plane of Order 10," Algebraic Theory of Codes II, Air Force Cambridge Research Laboratories Report AFCRL-71-0013, Sylvania Electronic Systems, Needham Heights, Mass., 1970.

[3] K. Appel and W. Haken, "Every planar map is four-colorable," Bull. Amer. Math. Soc., Vol. 82(1976), p. 711–712.

[4] R. C. Bose, "On the application of the properties of Galois fields to the problem of construction of hyper-Graeco-Latin squares," Sankhyã, Vol. 3(1938), p. 323–338.

[5] R. C. Bose and S. S. Shrikhande, "On the falsity of Euler's Conjecture about the non-existence of two orthogonal latin squares of order $4t + 2$," Proc. Nat. Acad. Sci. U. S. A. Vol. 45(1959), p. 734–737.

[6] R. C. Bose, S. S. Shrikhande, and E. T. Parker, "Further results on the construction of mutually orthogonal latin squares and the falsity of Euler's conjecture," Canad. J. Math. Vol. 12(1960), p. 189–203.

[7] R. H. Bruck and H. J. Ryser, "The non-existence of certain finite projective planes," Can. J. Math., Vol. 1(1949), p. 88–93.

[8] A. Bruen and J. C. Fisher, "Blocking Sets, k-arcs and Nets of Order Ten," Advances in Math., Vol. 10(1973), p. 317–320.

[9] J. L. Carter, "On the Existence of a Projective Plane of Order Ten," Ph. D. thesis, Univ. of Calif., Berkeley, 1974.

[10] S. Chowla and H. J. Ryser, "Combinatorial problems," Can. J. Math., Vol. 2(1950), p. 93–99.

[11] R. H. F. Denniston, "Non-existence of a Certain Projective Plane," J. Austral. Math. Soc., Vol. 10(1969), p. 214–218.

[12] L. Euler, "Recherches sur une nouvelle espèce de quarrés magiques," Verh. Zeeuwsch. Genootsch. Wetensch. Vlissengen Vol. 9(1782), p. 85–239.

[13] M. Hall, Jr. and H. J. Ryser, "Normal completions of incidence matrices," Amer. J. Math, Vol. 76(1954), p. 581–589.

[14] M. Hall, Jr., "Configurations in a plane of order 10," Ann. Discrete Math., Vol. 6(1980), p. 157–174.

[15] Z. Janko and T. van Trung, "Projective planes of order 10 do not have a collineation of order 3," J. Reine Angew. Math., Vol. 325(1981), p. 189–209.

[16] D. E. Knuth, "Estimating the Efficiency of Backtrack Programs," Mathematics of Computations, Vol. 29(1975), p. 121–136.

[17] C. W. H. Lam, L. Thiel, and S. Swiercz, "A feasibility study of a search for ovals in a projective plane of order 10," Proceeding of the Ninth Australian Conference on Combinatorial Mathematics, Springer-Verlag Lecture Notes in Mathematics, Vol. 952(1082), p. 349–352.

[18] C. W. H. Lam, L. Thiel, S. Swiercz, and J. McKay, "The Nonexistence of ovals in a projective plane of order 10," Discrete Mathematics, Vol. 45(1983), p. 319–321.

[19] C. Lam, S. Crossfield, and L. Thiel, "Estimates of a computer search for a projective plane of order 10," Congressus Numerantium, Vol. 48(1985), p. 253–263.

[20] C. W. H. Lam, L. Thiel, and S. Swiercz, "The Nonexistence of Code Words of Weight 16 in a Projective Plane of Order 10," J. of Combinatorial Theory, Series A, Vol. 42(1986), p. 207–214.

[21] C. W. H. Lam, L. H. Thiel, and S. Swiercz, "The Non-existence of Finite Projective Planes of Order 10," to appear.

[22] J. MacWilliams, N. J. A. Sloane, and J. G. Thompson, "On the existence of a projective plane of order 10," J. Combinatorial Theory, Sec. A., Vol. 14(1973), p. 66–78.

[23] E. T. Parker, "Construction of some sets of mutually orthogonal Latin squares," Proc. Amer. Math. Soc., Vol. 10(1959), p. 946–949.

[24] E. T. Parker, "Orthogonal Latin squares," Proc. Nat. Acad. Sci. U. S. A., Vol. 45(1959), p. 859–862.

[25] H. J. Ryser, "A note on a combinatorial problem," Proc. Amer. Math. Soc., Vol. 1(1950), p. 422–424.

[26] H. J. Ryser, "Combinatorial Mathematics," The Carus Mathematical Monographs, Math. Assoc. of America, 1963.

[27] G. Tarry, "Le problème des 36 officiers," C. R. Assoc. Fran. Av. Sci. Vol. 1(1900), p. 122–123, Vol. 2(1901), p. 170–203.

[28] J. G. Thompson, "Fixed Point Free Involutions and Finite Projective Planes," Durham Conference on Finite Groups, ed. M. Collins, 1978.

[29] J. G. Thompson, "Ovals in a projective plane of order 10," unpublished.

[30] J. G. Thompson, "Extremal 19-sets in the F_2-code of a projective plane of order 10," unpublished.

[31] L. H. Thiel, C. Lam, and S. Swiercz, "Using a CRAY-1 to perform backtrack search," Proc. of the Second International Conference on Supercomputing, San Francisco, USA, Vol. III(1987), p. 92–99.

[32] O. Veblen, "A system of axioms for geometry," Trans. Amer. Math. Soc., Vol. 5(1904), p. 343–384.

[33] O. Veblen and W. H. Bussey, "Finite Projective Geometries," Trans. Amer. Math. Soc., Vol. 7(1906), p. 241–259.

[34] O. Veblen and J. H. M. Wedderburn, "Non-Desarguesian and non-Pascalian Geometries," Trans. Amer. Math. Soc., Vol. 8(1907), p. 379–388.

[35] R. J. Walker, "An Enumerative Technique for a Class of Combinatorial Problems," Proc. AMS Symp. Appl. Math., Vol. X(1960), p. 91–94.

[36] S. H. Whitesides, "Collineations of projective planes of order 10," Parts I and II, J. Comb. Theory, Series A, Vol. 26(1979), p. 249–277.