Machine Proof of a Theorem on Cubic Residues

By D. H. Lehmer, E. Lehmer, W. H. Mills, and J. L. Selfridge

If p is a prime of the form $6m + 1$, the numbers

$$1^3, 2^3, \cdots, (p - 1)^3$$

when reduced modulo p consist of only $(p - 1)/3 = 2m$ distinct numbers between 1 and $p - 1$. These $2m$ numbers are known as the cubic residues of p. Thus, the cubic residues of 13 are 1, 5, 8, and 12, while those of 97, when arranged monotonely, begin

$$1, 8, 12, 18, 19, 20, 22, 27, 28, 30, \cdots .$$

Here we observe the triplet (18, 19, 20) of three consecutive numbers among the cubic residues of 97, while no such phenomenon exists for $p = 13$. We will call any set of three consecutive positive integers a *triplet.* A prime $p = 6m + 1$ is called *exceptional* if it does not have a triplet of cubic residues. Thus 13 is an exceptional prime, and 97 is not an exceptional prime.[1] It has been known since 1928 [1] that all "sufficiently large" primes have a triplet of cubic residues. Thus there are only a finite number of exceptional primes. By using machine methods we have proved much more, namely:

Theorem 1.

(a) *The only exceptional primes are*

$$2, 3, 7, 13, 19, 31, 37, 43, 61, 67, 79, 127, 283.$$

(b) *Every non-exceptional prime has a triplet of cubic residues that does not exceed*

$$(23532, 23533, 23534). \tag{1}$$

(c) *There are infinitely many primes whose smallest triplet of cubic residues is* (1). *Hence, result* (b) *is the best possible.*

Remark. The referee comments that the proof of Theorem 1, described below, is "not a machine proof in the sense of the theorem-proving programs now being developed." This is true. The aim of most writers on this subject is to consider a very general program enabling a digital computer to prove a wide class of theorems at a very low level, beginning with the axioms, setting its own goals, and trying to achieve them without human intervention. This is, in a way, a simulation problem. Speculations about such programs involve (significantly) such notions as decidability. Meanwhile, no really new theorems seem to emerge. Perhaps too much is expected of a single program.

In our work, instead of starting with axioms, we did not hesitate to use any device or previously known result that might be useful. In particular, the authors aided and abetted the machine in its search for a theorem and its proof. Neverthe-

Received April 3, 1962.

[1] Every prime $p = 6m - 1$ is non-exceptional since every number less than p is a cubic residue of such a prime.

less, all three results (a), (b), and (c) are due to the machine. Even the verification of these results using the data supplied by the machine would be far too long and hazardous a calculation to do by hand.

It is perhaps worth noting that (a), (b), and (c), though proved in a finite number of steps, are statements about infinite classes. For example (a) does not assert that the only exceptional primes less than one million are 7, 13, $\cdots$, 283. This would have been merely a new finite result, easily obtainable by the machine, but not a "genuine" theorem.

We give in what follows an explanation of how the computer was programmed to carry out the immense number of steps needed to prove this theorem. For discussion of the general problem for runs of kth power residues, the reader may consult previous papers [3] and [4]. We note here that a corresponding theorem for pairs of consecutive cubic residues has been proved by M. Dunton [2], and that there is no such theorem for four consecutive cubic residues [3].

For a prime $p = 6m + 1$, the $2(p - 1)/3 = 4m$ non-residues fall into two classes such that the product of a cubic residue by a non-residue of one class is congruent modulo p to a non-residue of the same class. The product of two non-residues of the same class is congruent to a non-residue of the other class, while the product of two non-residues of different classes is congruent to a cubic residue of p. We call these two classes of non-residues Class 1 and Class 2 respectively. This definition becomes unambiguous as soon as one member is assigned to Class 1. Let Class 0 denote the class of cubic residues. Thus the numbers from 1 to $p - 1$ are divided into three classes, each having $(p - 1)/3$ elements. We set $R(s) = i$, if s is congruent modulo p to a member of Class i. Thus for every integer s not divisible by p, $R(s)$ is defined and $R(s) = 0$, 1, or 2. Moreover it follows from the above discussion that

$$R(s_1s_2) \equiv R(s_1) + R(s_2) \pmod 3 \tag{2}$$

for any s_1 and s_2 not divisible by p.

Next let S be a given finite set of distinct primes, say

$$S = \{q_1, q_2, \cdots, q_t\}, \qquad q_1 < q_2 < \cdots < q_t.$$

The vector $A = [a_1, a_2, \cdots, a_t]$, where each $a_i = 0$, 1, or 2, will be called an S-vector. If p is a prime not in the set S, and $R(q_1) = a_1$, $R(q_2) = a_2$, $\cdots$, $R(q_t) = a_t$, then A will be called an S-vector belonging to p. If all the primes q_i in S are cubic residues of p, then the zero vector belongs to p. Except for this case, there are two S-vectors belonging to a given prime p, due to the choice in the definition of Class 1 and Class 2.

There are 3^t possible S-vectors. According to a theorem of Kummer (See [5], pp. 426–428), each of these 3^t possible S-vectors belongs to an infinite number of primes. Thus the primes of the form $6m + 1$, not in the set S, are divided into $\frac{1}{2}(3^t + 1)$ subsets, and each of these subsets contains an infinite number of primes.

Now let n be an integer whose prime factors all belong to the set S, so that

$$n = q_1^{\alpha_1}q_2^{\alpha_2} \cdots q_t^{\alpha_t}, \qquad (\alpha_i \geqq 0),$$

and let

$$\alpha_i = 3m_i + b_i \qquad (0 \leqq b_i \leqq 2).$$

Thus

$$[b_1, b_2, \cdots, b_t] \tag{3}$$

is a sequence of ternary digits uniquely determined by n and S. We call (3) the decomposition vector of n. Suppose A belongs to the prime p. Then (2) yields

$$R(n) \equiv a_1b_1 + a_2b_2 + \cdots + a_tb_t \pmod 3.$$

In particular n is a cubic residue of p if and only if

$$a_1b_1 + a_2b_2 + \cdots + a_tb_t \equiv 0 \pmod 3. \tag{4}$$

More picturesquely we say that n is a cubic residue of p if and only if the decomposition vector (3) of n and an S-vector A belonging to p are mutually perpendicular or orthogonal modulo 3.

If each member of the triplet $(n - 1, n, n + 1)$ has its prime factors restricted to the set S, and if each of the three decomposition vectors of $n - 1, n, n + 1$ respectively is orthogonal to the S-vector A modulo 3, then any prime p to which A belongs has this triplet of cubic residues. In this case we say that the triplet $(n - 1, n, n + 1)$ disposes of the S-vector A. If we can dispose of each of the 3^t possible S-vectors this way using triplets not exceeding $(N - 1, N, N + 1)$, then it follows that every prime not in the set S has a triplet of cubic residues not exceeding $(N - 1, N, N + 1)$. Finally, to show that there exist primes such that $(N - 1, N, N + 1)$ is the smallest triplet of cubic residues, we must take S^* to be the set of all primes less than $N + 2$ and find an S^*-vector A^* such that $(N - 1, N, N + 1)$ is the smallest triplet that disposes of A^*. If we can do all this with $N = 23533$, then we have proved Theorem 1.

Before we made the machine runs we had no way of knowing that 23533 was the correct value of N, in fact we did not even know if there was a value of N with the required properties. Hence it was necessary to experiment with different values of N—in fact we made machine runs with seven distinct values of N.

The success of our program depended very largely on the selection of a suitable set S. If p_0 is an exceptional prime not in the set S, then no triplet will dispose of the S-vectors belonging to p_0. Hence if we are to dispose of all S-vectors, the set S must include all the exceptional primes. Unfortunately, while we know that the set of exceptional primes is finite, we have no way of finding them all in advance, or even of finding an upper bound for them. However a preliminary machine run was designed to test individual primes. With this program we tested all primes less than 11243. This gave us the exceptional primes 2, 3, 7, 13, 19, 31, 37, 43, 61, 67, 79, 127, and 283. It seemed unlikely that there were any more, but we could not be sure of this until we ran the main program on the machine.

The most important consideration in the choice of the set of primes S is the necessity of having a large number of suitable triplets available. We can estimate the number of triplets required by a crude probabilistic argument. If none of the three numbers, $n - 1, n, n + 1$ is a cube and if all of their prime factors are in S, then the *a priori* probability that $(n - 1, n, n + 1)$ disposes of a given S-vector A is $1/27$. From this we estimate that the number of triplets required to

TABLE 1

First Members of Triplets

t	q_t	N_t	
2	3	2	1, 2
3	5	3	3, 4, 8
4	7	5	5, 6, 7, 14, 48
5	11	5	9, 10, 20, 54, 98
6	13	9	11, 12, 13, 24, 25, 26, 63, 64, 350
7	17	10	15, 16, 32, 33, 34, 49, 50, 119, 168, 440
8	19	12	17, 18, 19, 38, 55, 75, 76, 152, 169, 208, 323, 2430
9	23	11	21, 22, 23, 44, 68, 90, 160, 207, 322, 390, 2023
10	29	17	27, 28, 56, 114, 115, 143, 174, 230, 288, 493, 550, 782, 1274, 2000, 3248, 9800, 13310
11	31	16	29, 30, 31, 62, 91, 124, 153, 154, 340, 341, 494, 527, 713, 1518, 1519, 13454
12	37	24	35, 36, 37, 74, 110, 184, 185, 220, 259, 405, 406, 665, 702, 960, 999, 1330, 1443, 1664, 1700, 2736, 3625, 5290, 7104, 17575
13	41	27	39, 40, 80, 123, 203, 245, 246, 285, 286, 287, 368, 492, 574, 1023, 1024, 1188, 1517, 1680, 1681, 1885, 2294, 3772, 4959, 5082, 29600, 32798, 212380

For each t, q_t denotes the tth prime and N_t denotes the number of triplets in which the largest prime factor involved is q_t . The smallest member of each of these N_t triplets is listed. The fact that the table is complete is established in [6]

dispose of all 3^t vectors is approximately

$$\frac{t \log 3}{\log (27/26)} \doteq 29t.$$

However, the actual number of triplets available for small values of t is much less than $29t$. For example, if S is the set consisting of the first t primes, then the number of triplets $\leqq 442224$ surpasses $29t$ only for $t \geqq 25$. For $t = 13$ the total number of such triplets, irrespective of size, is only 141. The first members of these 141 triplets are given in Table 1. For $t = 50$, however, there are more than 1800 triplets less than 25000, which compares favorably with $29t = 1450$.

It is clear that before the proposed program can be attempted we must have the machine supply itself with a large list of triplets $(n - 1, n, n + 1)$ whose prime factors are in S and where $n \leqq N$. This preliminary program involves choosing a set S of primes and the limit N. For the first run one should take N rather large in order to be on the safe side.

To decide quickly whether the prime factors of a number $n \leqq N$ belong to the set S there are three methods available. One may simply attempt to factor n using as trial divisors only the primes q_i in S. If this fails to give a complete factorization, then n is not acceptable, since it contains a prime factor not included in set S. At least t divisions are required for each nonacceptable n. A second method involves the construction of the product

$$M = q_1^{\beta_1} q_2^{\beta_2} \cdots q_t^{\beta_t},$$

where β_i is determined by the inequality

$$q_i^{\beta_i} \leqq N < q_i^{\beta_i+1}.$$

Then n is acceptable if and only if n divides M. The number M will ordinarily consist of several machine words. However,

$$\log M \leqq t \log N$$

so that approximately $t \log N/\log W$ division instructions need to be executed for each n, where W stands for the largest integer representable by a machine word. Since $N \ll W$, the number of divisions required is much less than in the first method. The third method consists of sifting out the multiples of primes not in the set S from the numbers from 1 to N. If a binary machine is used, a compact bit representation technique is available in which one obtains a binary number of N bits whose nth bit is 1 or 0 according as n is acceptable or not. In this case it is particularly easy to look for three consecutive acceptable n's that make a triplet.

The problem of registering the list of triplets so as to make it most readily available to the main routine will be discussed later. We next consider the ordering of the list.

If in the vector (3), $b_d \neq 0$ and $b_h = 0$ for all $h > d$, then we say that the vector is of dimension d. The dimension of the zero vector is defined to be zero. The dimension of a triplet $(n - 1, n, n + 1)$ is defined to be the maximum of the dimensions of the three decomposition vectors of $n - 1$, n, $n + 1$ respectively. For example, if S contains the first 10 primes, then the triplet

$$\begin{aligned} n - 1 &= 9800 = 2^3 \cdot 5^2 \cdot 7^2 \sim [0, 0, 2, 2, 0, \cdots, 0] \\ n &= 9801 = 3^4 \cdot 11^2 \sim [0, 1, 0, 0, 2, 0, \cdots, 0] \\ n + 1 &= 9802 = 2 \cdot 13^2 \cdot 29 \sim [1, 0, 0, 0, 0, 2, 0, 0, 0, 1, 0, \cdots, 0] \end{aligned}$$

has dimension 10. As a final step in our preparation of the list of triplets for the main routine, we sort the list by dimension and prepare a small table in which the machine can look up the address of the first triplet of each dimension d.

In actual practice the number t is large enough so that it is prohibitive to consider all 3^t possible S-vectors separately. Therefore, it is necessary to dispose of many of them at oncc. We do this by means of the concept of case vector. Let $d \leqq t$. A case vector of dimension d is a vector

$$C = [a_1, a_2, \cdots, a_d], \tag{5}$$

where $a_i = 0$, 1, or 2. If we can find a triplet $(n - 1, n, n + 1)$ of dimension d, whose three decomposition vectors are orthogonal to C modulo 3, then in one blow, we have disposed of all those 3^{t-d} original S-vectors whose first d components agree with C.

We can now describe the main routine and its methodical disposal of these case vectors. This is easily done with the flow chart in Figure 1 and a brief explanation.[2]

[2] Figure 1 describes a program in which all 3^t possible S-vectors are examined. However, by symmetry and other considerations, it is sufficient to examine the vectors from [0, 2] to [2].

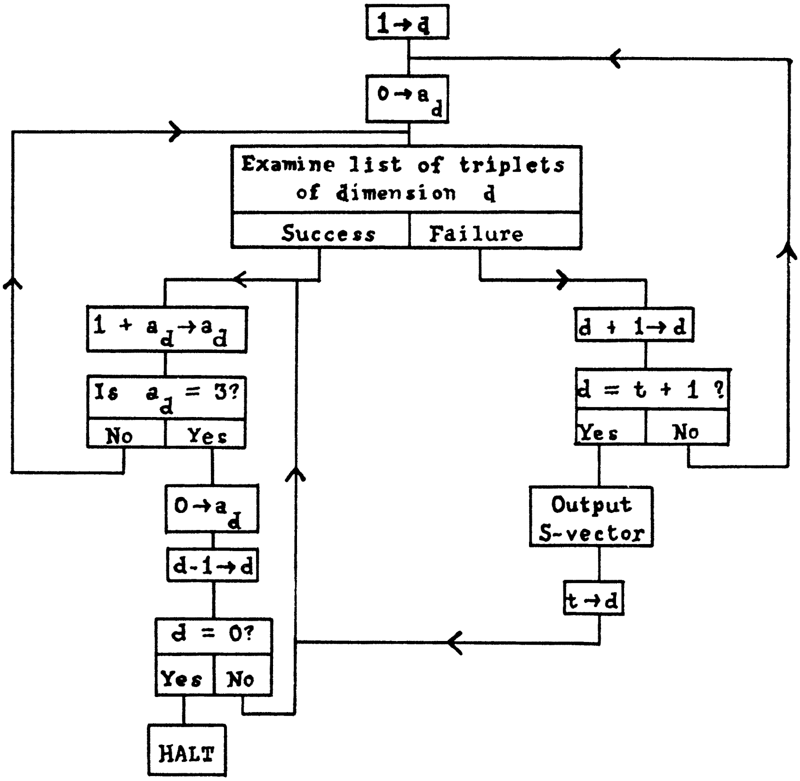

FIG. 1

By "success" we mean, of course, that the machine has discovered a triplet of dimension d which disposes of the current case vector. As the proof proceeds the dimension d rises and falls, never rising more than a unit at a time, but often falling more than a unit in an irregular way. In fact, if one imagines a ternary point to the left of a_1, the case vector (5) becomes a rational variable, written to the base 3, that rises monotonely from zero to one with irregular speed, certain intervals being much more difficult than others. Since the dimension d rises by one unit each time there is no triplet available of dimension d which disposes of the current case vector, it follows that when the dimension becomes $t + 1$, then the machine has examined the entire list of triplets without success. It then reports the S-vector as one that the data cannot handle. The processing of this output is discussed later.

It is clear from the diagram that the machine is spending almost all its time examining the list of triplets. It is also clear that the basic operation here is that of finding the inner product of the case vector and a decomposition vector, so that every effort should be made to shorten this program loop. To this effect we exploit the fact that decomposition vectors are usually quite sparse, i.e., they consist mostly of zeros. Hence, it is advantageous to use a more condensed format in which only the nonzero components of the decomposition vectors are involved. In one such representation the coded word

$$u_1, v_1; u_2, v_2; \cdots; u_k, v_k; 0 \tag{6}$$

corresponds to the decomposition vector in which the $(v_i + 1)$-st component is u_i, $i = 1(1)k$, and all other components are zero. For example, the number $n = 15678 = 2 \cdot 3^2 \cdot 13 \cdot 67$ has the decomposition vector

$$[1, 2, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1]$$

which corresponds to the coded word

$$1, 0; 2, 1; 1, 5; 1, 18; 0.$$

Each u_i is 1 or 2, and two binary bits are used to represent it. Suppose $2^{w-1} < t \leqq 2^w$. Then w bits are required for each v_i, and the word (6) is composed of $(w + 2)$-bit subwords.

With the decomposition vector written in the form (6), its inner product (4) with the case vector (5) is

$$\sum_{i=1}^{k} u_i a_{v_i+1} .$$

Since we need only evaluate this modulo 3 we interpret $u_i = 1$ as "add" and $u_i = 2$ as "subtract". Thus, we compute the inner product without multiplications. The case vector (5) is stored in t successive words in the memory, and the word (6) is unpacked by a series of left shifts in an obvious manner, so that each $(w + 2)$-bit subword constructs a command in which the operation is either addition or subtraction, as determined by its first two bits, and whose address is determined by its remaining w bits. The last symbol, zero, in the word (6) is interpreted as "end of message," and when that is reached the accumulator total is reduced to 0, 1, or 2 by a few additions or subtractions of 3. A simple test for zero now suffices to tell whether n is a cubic residue of those primes p that correspond to the current case vector.

We began with $t = 32$, $w = 5$, so that $w + 2 = 7$, and with our 35-bit binary words we could use decomposition vectors with 5 or fewer nonzero entries. This was sufficient for every triplet that we had occasion to use. Later we increased t to 50 and then to 55, so that we had to take $w = 6$, $w + 2 = 8$, and therefore we had to omit a few triplets which involve numbers with 5 distinct prime factors.

In conclusion, we give a brief account of the successive runs made by the machine which culminated in Theorem 1.

The preliminary search for exceptional primes had revealed only one of them greater than 127, namely 283. It was decided that for the first run the set S would consist of the first 31 primes, i.e., those primes $\leqq 127$, and 283. A high limit of $N = 442224$ was used, in the hope of providing the machine with an adequate supply of triplets. This actually gave us 1381 triplets. This run was successful in that all S-vectors were disposed of. At this point the machine had proved the (a) part of Theorem 1, so that we knew that our list of exceptional primes was complete. Furthermore, we knew that results of the type (b) and (c) were indeed possible.

This first run required 59 minutes on the IBM 704 during which time the machine considered about 101,000 case vectors.

The limit was now lowered to $N = 2^{17} = 131{,}072$ and a second successful run was made. The third run with $N = 2^{16} = 65536$, however, resulted in the output of eight vectors, which were easily disposed of by including the primes 137 and 139 in the set S—an operation that did not require the use of the computer. In the fourth run with $N = 2^{15} = 32768$ the machine reported 120 vectors, in two distinct regions, which it could not handle. We were able to dispose of these 120 vectors by increasing the set S. An attempt to make a run at 2^{14} was abandoned because of the large amount of output due to the fact that our supply of triplets had fallen below 1000. It now became apparent that many more primes would have to be included in the set S. Accordingly t was increased to 50 and a run was made with $N = 3 \cdot 2^{13} = 24576$. This resulted in an output of more than 512 vectors, all but 4 of which started with

$$[0, 2, 0, 2, 0, 2, 2, 0, 2, 2, 2, 2, 2, 2, 2, 2, 0, \cdots]. \tag{7}$$

By including the prime 271 all these vectors can be disposed of by the three triplets:

$$\begin{array}{lll} 8671 = 13 \cdot 23 \cdot 29, & 8672 = 2^5 \cdot 271, & 8673 = 3 \cdot 7^2 \cdot 59 \\ 9212 = 2^2 \cdot 7^2 \cdot 47, & 9213 = 3 \cdot 37 \cdot 83, & 9214 = 2 \cdot 17 \cdot 271 \\ 24388 = 2^2 \cdot 7 \cdot 13 \cdot 67, & 24389 = 29^3, & 24390 = 2 \cdot 3^2 \cdot 5 \cdot 271. \end{array}$$

The remaining 4 vectors which started with

$$[1, 2, 2, 2, 1, 0, 1, 2, 1, 0, 0, 1, \cdots]$$

were all disposed of by the triplet

$$20008 = 2^3 \cdot 41 \cdot 61, \qquad 20009 = 11 \cdot 17 \cdot 107, \qquad 20010 = 2 \cdot 3 \cdot 5 \cdot 23 \cdot 29,$$

which the machine did not possess because 20010 has five distinct prime factors.

At this point we thought that Theorem 1 might hold with $N = 24389$ instead of 23533. A sixth machine run was made with $N = 24389$. There were a number of additional output vectors, but the extra vectors all started as in (7) and were disposed of with the prime 271 as before. We now had part (b) of the theorem with $N = 24389$. In an attempt to prove part (c) with this value of N, a "case test" program was written. In this program we let S^* be the set of all primes less than 24391, and we take a particular S^* vector A^*. The program then finds the smallest triplet that disposes of A^*. We took for A^* various extensions of the vector (7) in which nearly all the components corresponding to large primes q were 2. The largest triplet put out by these runs was

$$23532 = 2^2 \cdot 3 \cdot 37 \cdot 53, \qquad 23533 = 101 \cdot 233, \qquad 23534 = 2 \cdot 7 \cdot 41^2.$$

The vector A^* which produced this result consists of

$$\left\{\begin{array}{l} \text{0's corresponding to } q = 2, 5, 11, 19, 59, 79, 89, 113, 191, 211, 223, 229, \\ \qquad\qquad 269, 373, 577, 829, 839, 1613, 2393; \\ \text{1's corresponding to } q = 233, 313, 353, 919, 967, 2671 \\ \text{2's corresponding to all other primes } q < 23535. \end{array}\right. \tag{8}$$

This and other outputs of the case test runs suggested that the primes $q = 233$, 313, 331, 449, and 967 should be incorporated in the main run. The final successful run was therefore made with $N = 23533$ and $t = 55$. This run alone constitutes a proof of parts (a) and (b) of Theorem 1, while the case test of the vector (8) proves part (c) of the theorem.

The last main run was done on the IBM 7090 in about 40 minutes and required the examination by the machine of about 250,000 case vectors.

The various machine runs were made at the Computer Centers of the University of California at Berkeley, Los Angeles, and Livermore, and at the University of Washington in Seattle, and we are grateful to the directors of these laboratories for the free use of their equipment. The machines used were the IBM 701, 704, 709, and 7090. We are also grateful to John Brillhart, David Mapes, and Vance Vaughn for donating their time to this unsupported research.

The University of California
Los Angeles 25, California

Yale University
New Haven, Connecticut, and

The University of Washington
Seattle, Washington

1. A. Brauer, "Über Sequenzen von Potenzresten," *S.-B Deutsch Berlin*, S., 1928, p. 9–16.
2. M. Dunton, "A bound for consecutive pairs of cubic residues," (to be published).
3. D. H. & Emma Lehmer, "On runs of residues," *Proc. Amer. Math. Soc.*, v. 13, 1962, p. 102–106.
4. D. H. & Emma Lehmer & W. H. Mills, "Pairs of consecutive power residues, *Canad. J. Math.*
5. D. Hilbert, *Jber. Deutsch. Math. Verein.*, v. 4, 1897, p. 175–546.
6. D. H. Lehmer, "On a problem of Störmer," (to be published).