Smallest Entry in a 3x3 Magic Square of Squares

Lee Morgenstern

June, 2007

Abstract

This paper develops a computer procedure that has proved that all 9 entries of a 3x3 magic square of distinct squares must be at least the squares of 8-digit numbers.

Introduction

A necessary condition for a 3x3 magic square of distinct squares is a solution to any of its 7-square subsets.
This paper studies the following 7-square subset where a,b,c > 0 and thus, c-(a+b) is the smallest entry.

```
-------  c-(a+b)  -------
c+(b-a)  c        c-(b-a)
c-b      c+(a+b)  c-a
```

This configuration contains three arithmetic progressions having the same starting value:

```
c-(a+b), c-b, c-(b-a), with step value a;
c-(a+b), c-a, c+(b-a), with step value b; and
c-(a+b), c,   c+(a+b), with step value a+b.
```

Since these values must be squares, this motivates the study of 3-square arithmetic progressions having a fixed starting value.
If we can find a set of three progressions that also have the step value relationship (a, b, a+b), we will have found a magic square containing at least 7 squared entries.

Suppose we have a 3-square arithmetic progression given by L^2, M_n^2, H_n^2 where $0 < L < M_n < H_n$, and suppose that the smallest value, L = 1.
Here is a list of the progressions for n = 1 ... 5.

n	L^2	M_n^2	H_n^2	STEP
1	1^2	5^2	7^2	24
2	1^2	29^2	41^2	840
3	1^2	169^2	239^2	28560
4	1^2	985^2	1393^2	970224
5	1^2	5741^2	8119^2	32959080

Every 3-square arithmetic progression starting from 1 can be produced using a simple recursion. Note how fast the step values increase in size.
If there were a magic square having 1 as the smallest value, but with the largest value having 100 digits, that solution could be found in about 66 iterations of the recursion.

But because any choice of the largest step value is more than twice the size of any step value below it, there can't be a selection of three step values with the relationship (a, b, a+b). Therefore L = 1 cannot be the smallest entry in this 7-square configuration. This also means that 1 cannot be the smallest entry in any 3x3 magic square of distinct squares.

Other starting values can be proved impossible using a termination condition, derived in this paper. It has been found that the termination condition occurs surprisingly early in a generated sequence of candidate step values. So the impossibility of each starting value takes very little time to prove.

==================

This paper contains a proof that all 3-square arithmetic progressions with a fixed starting value can be produced by simple recursions of an easily obtainable finite set of generators.

Each recursively produced sequence consists of values in a near-geometric progression increasing by about 34 times for each iteration, so the numbers get very large very fast. This makes it possible to do a quick but complete search for magic squares having extremely large numbers.

But better than that, this paper contains a proof that if an enumeration is done for a particular starting value, then once a certain condition occurs, it is no longer possible to achieve a solution, and so the enumeration can be terminated as impossible to satisfy.

Part 1
Generating All Arithmetic Progressions

This part of the paper develops and proves a technique for generating all possible 3-square arithmetic progressions having a fixed starting value. The technique uses a recursive formula which takes a "generator" progression and produces an infinite series of progressions with increasing values.

The Arithmetic Progression Formula

The 3-square arithmetic progression under study is

L^2, M_n^2, H_n^2,

where L is fixed.

The step value of the progression = $M_n^2 - L^2 = H_n^2 - M_n^2$.

This is expressed in this paper as

$$\textbf{(1)}\quad L^2 = 2M_n^2 - H_n^2; \quad 0 < L < M_n < H_n.$$

We want to represent a given value of L^2 as this binary quadratic form in all possible ways. We also want to generate those representations, as needed, in H_n order: $H_1 < H_2 < H_3$, and so on.

The recursion described below will accomplish this, however, it must be noted that putting a representation using (M_n,H_n) into the recursion does not in general produce (M_{n+1},H_{n+1}). For example, if L is prime, then putting (M_n,H_n) into the recursion will produce (M_{n+3},H_{n+3}). Thus, in this case we need three different recursive sequences in order to produce all the representations.

In this paper, the number of different recursive sequences needed to produce all H_n values for a given L is given by g. That is, putting (M_n,H_n) into the recursion produces (M_{n+g},H_{n+g}).

Forward Recursion Theorem

Given a representation of (1) using (M_n,H_n) for a given L, then (M_{n+g},H_{n+g}) given by

(2a) $M_{n+g} = 3M_n + 2H_n,$

(2b) $H_{n+g} = 3H_n + 4M_n$

is also a representation.

Proof

Starting with

$$2M_{n+g}^2 - H_{n+g}^2,$$

substitute (2a), (2b),

$$2(3M_n + 2H_n)^2 - (3H_n + 4M_n)^2,$$

expand the terms,

$$18M_n^2 + 8H_n^2 + 24H_nM_n - (9H_n^2 + 16M_n^2 + 24H_nM_n),$$

and cancel, producing

$$2M_n^2 - H_n^2,$$

which from (1) is equal to L^2.

Thus

$$2M_{n+g}^2 - H_{n+g}^2 = L^2.$$

Remark

Note that the recursion has only positive coefficients, therefore starting with a representation using positive numbers, the recursion produces a sequence of representations using increasingly larger positive numbers.

Example

The number of different recursion sequences that can be produced for a given L is given by g. Suppose that L = 7. It will be found that g = 3.
The smallest representation uses

$(M_1,H_1) = (13, 17);\ 7^2 = 2(13^2) - 17^2.$

Putting these values into the recursion produces

$(M_4,H_4) = (73,103);\ 7^2 = 2(73^2) - 103^2;$

$(M_7,H_7) = (425,601);\ 7^2 = 2(425^2) - 601^2.$

Here is the second smallest representation for L = 7.
It produces a different sequence.

$(M_2,H_2) = (17, 23);\ 7^2 = 2(17^2) - 23^2.$

Putting these values into the recursion produces

$(M_5,H_5) = (97,137);\ 7^2 = 2(97^2) - 137^2;$

$(M_8,H_8) = (565,799);\ 7^2 = 2(565^2) - 799^2.$

Here is the third smallest representation where $H_g = 7L$.

$(M_3,H_3) = (35, 49);\ 7^2 = 2(35^2) - 49^2.$

Putting these values into the recursion produces

$(M_6,H_6) = (203, 287);\ 7^2 = 2(203^2) - 287^2;$

$(M_9,H_9) = (1183,1673);\ 7^2 = 2(1183^2) - 1673^2.$

Remark

Note that H_1 through H_9 are in increasing order, even though they came from multiple sequences. It will be proved below that this will always be true for any L.

Reverse Recursion Formulas

Solving for the inverse recursion formulas from (2a), (2b),

(3a) $M_n = 3M_{n+g} - 2H_{n+g},$

(3b) $H_n = 3H_{n+g} - 4M_{n+g}.$

Remark

Since the forward recursion takes positive values and produces a sequence of ever-increasing values, the reverse recursion must produce a sequence of ever-decreasing values.

5/7 Lemma

Given a representation of (1) using (M_n,H_n) for a given L,
if

$$H_n > 7L,$$

then

$$M_n < (5/7)H_n.$$

Proof

If $H_n > 7L$ then

$$L < H_n/7$$

and combining with (1)

$$2M_n^2 = H_n^2 + L^2 < H_n^2 + H_n^2/49$$

or

$$M_n^2 < (25/49)H_n^2$$

or

$$M_n < (5/7)H_n.$$

Reverse Recursion Reduction Lemma

Given a representation of (1) using (M_{n+g},H_{n+g}) for a given L,
then applying the reverse recursion (3a),(3b) to produce (M_n,H_n),
if

$$H_{n+g} > 7L,$$

then

$$H_n > L.$$

Proof

If $H_{n+g} > 7L$, then from the 5/7 Lemma

$$M_{n+g} < (5/7)H_{n+g}.$$

Combining with (3a),

$$H_n = 3H_{n+g} - 4M_{n+g} > 3H_{n+g} - 4(5/7)H_{n+g}$$

or

$$H_n > H_{n+g}/7.$$

And since $H_{n+g} > 7L$,

$$H_n > L.$$

Finite Generator Theorem

A generator is a representation of (1) that is not produced by any other representation using the recursion (2a),(2b). For a given value of L, there is a finite number, g, of generators using (M_1,H_1) ... (M_g,H_g) and $L < H_1 < \ldots < H_g \leq 7L$.

Proof

Starting from any representation where $H_n > 7L$ and applying the reverse recursion (3a),(3b) repeatedly, we will eventually encounter a terminal representation with $H_t \leq 7L$, for some t.
When this happens, the previous representation that produced it must have had $H_{t+g} > 7L$, thus this terminal representation must have $H_t > L$ by the Reverse Recursion Reduction Lemma.

Therefore, this terminal representation has $L < H_t \leq 7L$, and is the generator of the sequence of representations that were encountered using the above reverse recursion procedure. When the forward recursion (2a),(2b) is applied, all representations in that sequence are reachable.

Since starting from any representation and applying the reverse recursion always eventually produces a representation with $L < H_t \leq 7L$, then all representations with numbers greater than that range are reachable by starting with a representation in that range and applying the forward recursion. Since that range is finite for a given value of L, the number of generators is finite for a given L.

Remark

Finding all generators is just a matter of testing values for H_n, n = 1 ... g-1, in the range L ... 7L that satisfy (1);
then adding $H_g = 7L$, which is always a generator.
Part 3 of this paper shows that there exist faster ways.

Generator Nonredundancy Theorem

All representations using (M_n,H_n) with $L < H_n \leq 7L$ are generators.

Remark

When searching for generators in the finite range, there is no chance that one of them could be produced by another in the same range. Therefore, in a computer procedure, there is no need to check for duplicate values.

Proof

Given a representation using (H_n,M_n) with $L < H_n \leq 7L$, then from (1),

$$2M_n^2 = H_n^2 + L^2 > 2L^2$$

thus

$$M_n > L.$$

From the recursion (2a),

$$H_{n+g} = 3H_n + 4M_n$$

and since

$$3H_n + 4M_n > 3L + 4L = 7L$$

we have

$$H_{n+g} > 7L.$$

Therefore, one application of the recursion on a generator always produces a representation past the range of the generators. So there is no chance of finding a non-generator in the generator range.

Relative Placement Theorem

Any two different recursive sequences produce mutually exclusive sets of representations and their relative placements within the sets of sequences never change. This is because $H_j < H_k$ implies $H_{j+g} < H_{k+g}$ for any j and k.

Remark

This is important for the orderly enumeration of representations. The example above with $L = 7$ and $g = 3$ shows that the H_n values are produced in increasing order by applying the recursion once for each generator sequence, then repeating the recursion once each for the produced values, and so on.

Proof

Suppose (M_j,H_j) and (M_k,H_k) are used in two different representations produced from possibly different generated sequences with

$$H_j < H_k.$$

We have

$$2M_j^2 - H_j^2 = L^2,$$
$$2M_k^2 - H_k^2 = L^2,$$

thus

$$2M_j^2 = H_j^2 + L^2$$

and since $H_j < H_k$

$$2M_j^2 < H_k^2 + L^2$$

or

$$2M_j^2 < 2M_k^2$$

or

$$M_j < M_k.$$

Therefore,

$$H_j < H_k \text{ implies } M_j < M_k.$$

Suppose that g is the number of generators and the forward recursion (2a),(2b) produces H_{j+g} from H_j and H_{k+g} from H_k in the sets of sequences.

Then

$$H_{j+g} = 3H_j + 4M_j$$

and since $H_j < H_k$ and $M_j < M_k$

$$H_{j+g} < 3H_k + 4M_k$$

or

$$H_{j+g} < H_{k+g}.$$

Therefore,

$$H_j < H_k \text{ implies } H_{j+g} < H_{k+g}$$

for any j and k, and their relative placement never changes.
This also means that their values can never be the same, thus multiple recursive sequences produce mutually exclusive values.

Part 2
Enumerating Potential Magic Squares

This part of the paper describes a procedure for enumerating a sequence of 3-square arithmetic progressions in order to find a magic square. When certain conditions occur in the sequence, the procedure terminates because it is no longer possible to satisfy the magic square requirements. These termination conditions are proved below.

Magic Square Requirements

See the Introduction to this paper for a description of the 7-square subset of the 3x3 magic square of distinct squares that we are studying. For a solution to this configuration, we need to find three 3-square arithmetic progressions having the same starting value and their step values must have the relationship (a, b, a+b). In other words, the sum of the step values of the first two arithmetic progressions equals the step value of the third.

We can also express this with twice the step values.
Twice the step value of arithmetic progression L^2, M_n^2, H_n^2 is $(H_n^2 - L^2)$.
So we need to find three representations of L^2 that use (M_i,H_i), (M_j,H_j), (M_k,H_k) and that satisfy the equation

$$(H_i^2 - L^2) + (H_j^2 - L^2) = (H_k^2 - L^2).$$

Note that H_k must be the largest term.
Adding $2L^2$ to both sides gives

(4) $$H_i^2 + H_j^2 = H_k^2 + L^2; \quad H_i < H_j < H_k.$$

Magic Square Enumeration Procedure

After finding the generators (M_1,H_1) ... (M_g,H_g), arranged in increasing H_n value, then the rest of the representations can be produced in order, one at a time, and successive values of H_n^2 put into a list. The increasing order is guaranteed by the Relative Placement Theorem.

Each time a new H_k^2 is produced, it can be checked with combinations of H_i^2 values that come before it in that list to try and satisfy (4).

Checking is quicker than you might think. This is because, most of the time, $H_{k-2}^2 + H_{k-1}^2 < H_k^2 + L^2$.
This means that the biggest sum that can be made using previous H_i values is too small.
So, after just a single check, all possible combinations using the latest H_k can be rejected.

It will also be seen below that if the sum is too small by $2L^2$, that is, $H_{k-2}^2 + H_{k-1}^2 < H_k^2 - L^2$, then H_{k+g}, H_{k+2g}, etc. can also be rejected.

When this simple case does not occur, checking combinations using H_k can be done in at worst linear time in the length of the list.
Here is a general procedure.

```
Set k := n; i := 1; j := k-1
Do
  Compare Hi^2 + Hj^2 to Hk^2 + L^2;
  If too small, increment i;
  Else If too large, decrement j;
  Else just right; (you have found a magic square of 7 squares)
While i < j (which will be at most k-2 times)
```

Example

Let's use the L = 7 numbers above.

n	1	2	3	4	5	6	7	8	9
H_n	17	23	49	103	137	287	601	799	1673

i=1,j=2,k=3
$H_3^2 + L^2 = 49^2 + 7^2 = 2450$.
$H_1^2 + H_2^2 = 17^2 + 23^2 = 818$, too small by more than $2\text{x}7^2$.

i=2,j=3,k=4
$H_4^2 + L^2 = 103^2 + 7^2 = 10658$.
$H_2^2 + H_3^2 = 23^2 + 49^2 = 2930$, too small by more than $2\text{x}7^2$,
so it's not necessary to check i=1
because the sum will be even smaller.

i=3,j=4,k=5
$H_5^2 + L^2 = 137^2 + 7^2 = 18818$.
$H_2^2 + H_3^2 = 49^2 + 103^2 = 13010$, too small by more than $2\text{x}7^2$,
so it's not necessary to check i=1 or i=2
because the sum will be even smaller.

As the following proofs will show, we need not enumerate any further.

To prove the Enumeration Termination Theorem, we need two lemmas.

Changing Geometric Progression Lemma

If (M_j,H_j) and (M_k,H_k) with $j < k$ are used in (1), then

$$H_{j+g}/H_j > H_{k+g}/H_k$$

and

$$M_{j+g}/M_j < M_{k+g}/M_k.$$

Proof

If (M_j,H_j) and (M_k,H_k) with $j < k$ are used in (1), then

$$H_j < H_k$$

thus

$$L^2/H_j^2 > L^2/H_k^2.$$

Dividing (1) by H_n^2 gives

$$L^2/H_n^2 = 2M_n^2/H_n^2 - 1.$$

Substituting j and k for n and combining the above result,

$$2M_j^2/H_j^2 - 1 > 2M_k^2/H_k^2 - 1$$

or

$$M_j/H_j > M_k/H_k.$$

Using the forward recursion formula (2b) and dividing by H_n,

$$H_{n+g}/H_n = 3 + 4M_n/H_n.$$

Substituting j and k for n and combining the above result,

$$H_{j+g}/H_j = 3 + 4M_j/H_j > 3 + 4M_k/H_k = H_{k+g}/H_k,$$

Therefore

$$H_{j+g}/H_j > H_{k+g}/H_k.$$

Repeating the derived condition

$$M_j/H_j > M_k/H_k$$

we also have

$$H_j/M_j < H_k/M_k.$$

Using the forward recursion formula (2a) and dividing by M_n,

$$M_{n+g}/M_n = 3 + 2H_n/M_n.$$

Substituting j and k for n and combining the above result,

$$M_{j+g}/M_j = 3 + 2H_j/M_j < 3 + 2H_k/M_k = M_{k+g}/M_k.$$

Therefore

$$M_{j+g}/M_j < M_{k+g}/M_k.$$

Corresponding Combination Rejection Lemma

Given the values H_i, H_j, H_k from a sequence of representations of (1) for a given L, with $i < j < k$,

(Case 1)
if
$$H_i^2 + H_j^2 < H_k^2 - L^2$$
then
$$H_{i+g}^2 + H_{j+g}^2 < H_{k+g}^2 - L^2;$$

(Case 2)
if
$$H_i^2 + H_j^2 > H_k^2 + L^2$$
then
$$H_{i+g}^2 + H_{j+g}^2 > H_{k+g}^2 + L^2.$$

Remark

This means that once one of the above two cases becomes true, it will also be true for the corresponding combination of values with index offsets of multiples of g. That is, if it's true for the combination (H_i,H_j,H_k), then it will be true for $(H_{i+g},H_{j+g},H_{k+g})$, $(H_{i+2g},H_{j+2g},H_{k+2g})$, and so on.

All of them will produce a sum outside of the interval $H_k^2 \pm L^2$, and therefore can never satisfy (4). Therefore, it will no longer be necessary to check those combinations in the rest of the enumeration for a given value of L.

Proof

(Case 1)
If
$$H_i^2 + H_j^2 < H_k^2 - L^2,$$
then adding $2L^2$ to both sides and using (1) we get
$$2M_i^2 + 2M_j^2 < 2M_k^2.$$
Multiplying by $(M_{k+g}/M_k)^2$,
$$2M_i^2(M_{k+g}/M_k)^2 + 2M_j^2(M_{k+g}/M_k)^2 < 2M_{k+g}^2.$$

From the Changing Geometric Progression Lemma with $j < k$,
$$M_{j+g}/M_j < M_{k+g}/M_k.$$
Squaring and rearranging, we get
$$M_{j+g}^2 < M_j^2(M_{k+g}/M_k)^2.$$
Similarly, with $i < k$,
$$M_{i+g}^2 < M_i^2(M_{k+g}/M_k)^2.$$

Combining the above results,

$$2M_{i+g}^2 + 2M_{j+g}^2 < 2M_{k+g}^2$$

and then subtracting $2L^2$ from both sides and using (1),

$$H_{i+g}^2 + H_{j+g}^2 < H_{k+g}^2 - L^2.$$

(Case 2)

If

$$H_i^2 + H_j^2 > H_k^2 + L^2,$$

then multiplying by $(H_{k+g}/H_k)^2$, we get

$$H_i^2(H_{k+g}/H_k)^2 + H_j^2(H_{k+g}/H_k)^2 > H_{k+g}^2 + L^2(H_{k+g}/H_k)^2.$$

From the Changing Geometric Progression Lemma with $j < k$,

$$H_{j+g}/H_j > H_{k+g}/H_k.$$

Squaring and rearranging, we get

$$H_{j+g}^2 > H_j^2(H_{k+g}/H_k)^2.$$

Similarly, with $i < k$,

$$H_{i+g}^2 > H_i^2(H_{k+g}/H_k)^2.$$

Combining the above results,

$$H_{i+g}^2 + H_{j+g}^2 > H_{k+g}^2 + L^2(H_{k+g}/H_k)^2,$$

and since $H_{k+g}/H_k > 1$,

$$H_{i+g}^2 + H_{j+g}^2 > H_{k+g}^2 + L^2.$$

Enumeration Termination Theorem

If for g consecutive values of H_k, $(r+1 \leq k \leq r+g)$, for some r,

(5) $$H_1^2 + H_{k-1}^2 < H_k^2 - L^2$$

and for each combination of H_i and H_j, $i < j < k$,

either

(6) $$H_i^2 + H_j^2 < H_k^2 - L^2$$

or

(7) $$H_i^2 + H_j^2 > H_k^2 + L^2,$$

then it is impossible for any H_i, H_j, H_k combination where $i < j < k$ and $k \geq r+1$, to satisfy (4) and the magic square enumeration for the given value of L can be terminated.

Proof

If (6) or (7) applies, then we know from the Corresponding Combination Rejection Lemma that all future corresponding combinations will not satisfy (4).

But this doesn't cover all possible future combinations.

New combinations will be created because there are more representations in the list.

For example, suppose (7) is met using i=2,j=k-1; the sum is too big.
Then we also know that (7) is met using i=2+g,j=k-1+g. But then,
new combinations where i=1...g+1 and j=k-1+g could have a smaller sum.
These would need to be tested -- unless you knew that (5) was met.
Then all new combinations would have a sum which was too small.
Therefore, after g consecutive rejections using the above criteria,
all combinations, old and new, can be rejected.

Example

Just for completeness, let's try the enumeration for L = 1, to prove
that the comments in the Introduction are true.
Only the first three representations are needed because g = 1.

n	1	2	3
H_n	7	41	239

i=1,j=2,k=3

$H_3^2 - L^2 = 239^2 - 1^2 = 57{,}120.$

$H_1^2 + H_2^2 = 7^2 + 41^2 = 1730.$

Conditions (5) and (6) are met and since g = 1, we are done. Thus,
1 cannot be the lowest entry in any 3x3 magic square of distinct squares.

Example

L = 41 is the smallest value of L where condition (7) appears.
Since L is prime, g = 3. We only need the first five representations.

n	1	2	3	4	5
H_n	113	119	287	679	713

i=1,j=2,k=3

$H_3^2 - L^2 = 287^2 - 41^2 = 80{,}688.$

$H_1^2 + H_2^2 = 113^2 + 119^2 = 26{,}930.$

Conditions (5) and (6) are met.

i=2,j=3,k=4

$H_4^2 - L^2 = 679^2 - 41^2 = 459{,}360.$

$H_2^2 + H_3^2 = 119^2 + 287^2 = 96{,}530.$

Condition (6) is met for the largest values of i and j,
therefore condition (6) is met for all combinations,
condition (5) being one of them.

i=3,j=4,k=5

$H_5^2 + L^2 = 713^2 + 41^2 = 510{,}050.$

$H_3^2 + H_4^2 = 287^2 + 679^2 = 543{,}410.$

Condition (7) is met, so we need to test k=5 further.

i=2,j=4,k=5
$H_5^2 - L^2 = 713^2 - 41^2 = 506,688$
$H_2^2 + H_4^2 = 119^2 + 679^2 = 475,202$.
Condition (6) is met. All the rest of the combinations will have lower sums than this one, so they all satisfy condition (6). Condition (5) is also satisfied.

We have 3 consecutive rejections of all combinations, so 41^2 cannot be the lowest entry in any 3x3 magic square of distinct squares.

Example

L = 71 is the smallest value of L where a value of H_k cannot be rejected and more representations have to be generated.
Since L is prime, g = 3.

n	1	2	3	4	5	6	7
H_n	97	391	497	631	2297	2911	3689

i=1,j=2,k=3
$H_3^2 - L^2 = 497^2 - 71^2 = 241,968$.
$H_1^2 + H_2^2 = 97^2 + 391^2 = 162,290$.
Conditions (5) and (6) are met.

i=2,j=3,k=4
$H_4^2 - L^2 = 631^2 - 71^2 = 393,120$.
$H_4^2 + L^2 = 631^2 + 71^2 = 403,202$.
$H_2^2 + H_3^2 = 391^2 + 497^2 = 399,890$.
Neither condition (6) or (7) is met, so we can't reject this combination. Since the sum is too small for (4) to be satisfied, there's no need to try i=1,j=3 or i=1,j=2, which would make an even smaller sum.

i=3,j=4,k=5
$H_5^2 - L^2 = 2297^2 - 71^2 = 5,271,168$.
$H_3^2 + H_4^2 = 497^2 + 631^2 = 645,170$.
Condition (6) is met and thus is met by all other combinations including condition (5).

k=6
Since all combinations of k=3 were rejected with (5) and (6), all future combinations of k=6, k=9, etc. are also rejected and need not be tested.

i=5,j=6,k=7
$H_7^2 + L^2 = 3689^2 + 71^2 = 13,613,762$.
$H_5^2 + H_6^2 = 2297^2 + 2911^2 = 13,750,130$.
Condition (7) is met, so we need to test k=7 further.

i=4,j=6,k=7
$H_7^2 - L^2 = 3689^2 - 71^2 = 13{,}603{,}680$.
$H_4^2 + H_6^2 = 631^2 + 2911^2 = 8{,}872{,}082$.
Condition (6) is met. All the rest of the combinations will have lower sums than this one, so they all satisfy condition (6).
Condition (5) is also satisfied.

We now have 3 consecutive rejections of all combinations, so 71^2 cannot be the lowest entry in any 3x3 magic square of distinct squares.

Example
L = 49 is not a prime and g = 5.

n	1	2	3	4	5	6	7
H_n	71	119	161	257	343	457	721

The first 5 tests all meet conditions (5) and (6), so we have 5 consecutive rejections of all combinations.

Example
L = 119 is not a prime and g = 9.
It will be found that k = 3,4,6,7,8 can't be rejected,
but k = 9 through 17 are rejected, making 9 consecutive rejections.

Part 3
Finding Generators

This part of the paper describes efficient methods for finding generators. For large values of L, finding all the generators can be more time-consuming than doing the calculations to search for the magic square requirements and to reject combinations.

Prime Factor Reduction

In a 3-square arithmetic progression, if either the lowest or highest values have a prime factor of the form 8k+3 or 8k+5, then all three terms will have that factor. This also means that all 7 of the entries in a magic square will have that factor. So we can divide out the factor producing a smaller magic square. Therefore it is not necessary to test values of L and H having those primes as factors.

This leaves values of L and H having prime factors of only 8k+1 and 8k+7.
The first of these numbers are
1, 7, 17, 23, 31, 41, 47, 49, 71, 73, 79, 89, 97, 103, 113, 119.

This reduces the number of L values that need to be tested and also reduces the number of H values that need to be searched to find the generators. They both come from the same list.

Proof

Given

$$L^2 = 2M^2 - H^2,$$

we factor this in the unique factorization domain Z[√(2)] as

$$L^2 = (M\sqrt{2} + H)(M\sqrt{2} - H).$$

If a prime of the form 8k+3 or 8k+5, which is also a prime in the domain Z[√(2)], is a factor of L, then it must also be a factor of (M√(2) + H) or (M√(2) - H). If it is a factor of one of them, it is also a factor of the conjugate, so it is a factor of both. If it is a factor of both, then it is also a factor of their difference, which is 2H. Since the factor is odd, it is a factor of L. If it is a factor of H and L and odd, then it is also a factor of M. Thus, it is a factor of all three terms and can be factored out to produce a smaller solution. Therefore, values of L having prime factors of 8k+3 or 8k+5 need not be tested.

Formula for the Number of Generators

If the prime factorization of

$$L = p^a q^b \ldots r^c,$$

then the number of generators

$$g = (2a + 1)(2b + 1) \ldots (2c + 1).$$

Examples

For L = 1, g = 1.

Given that p, q, r are primes:

If L = p, then g = 3.

If $L = p^2$, then g = 5.

If $L = p^3$, then g = 7.

If L = pq, then g = 9.

If $L = p^2q$, then g = 15.

If L = pqr, then g = 27.

Generators Come In Pairs

If (M_n,H_n) is a generator for a given L, then so is (M_p,H_p) given by

$$M_p = 17M_n - 12H_n,$$
$$H_p = 24M_n - 17H_n.$$

Proof

We have to prove that (M_p,H_p) gives a representation of L^2 using (1).
Starting with

$$2M_p^2 - H_p^2$$

and substituting the relation above

$$2(17M_n - 12H_n)^2 - (24M_n - 17H_n)^2$$

expand the terms and cancel, producing

$$2M_n^2 - H_n^2$$

which is equal to L^2.

We also have to prove that the formula produces generators
from other generators.
If (M_n,H_n) is a generator,
then their values range from

$M = L$, $H = L$ to $M = 5L$, $H = 7L$.

Thus H_p ranges between

$24L - 17L$ and $24x5L - 17x7L$

or

$7L$ and L,

the same range as H_n when it is a generator.

Also, the case where $H_p = H_n$ doesn't exist since if

$$H_n = H_p = 24M_n - 17H_n.$$

then

$$(3/4)H_n = M_n$$

and putting this into (1) gives

$$2(9/16)H_n^2 - H_n^2 = L_n^2$$

or

$$H_n^2 = 8L^2$$

or

$$H_n = L\sqrt{(8)}$$

which can never be an integer.

The Pre-Generator Method

In the formula for the number of generators, g is always an odd number.
But $H_g = 7L$ is always a generator.
So that leaves an even number of other generators between L and 7L.
As the above proof shows, these other generators come in related pairs.

Suppose that we have a complete set of generators for a given L,
Applying the reverse recursion to these values
produces all values less than L so that $H_{n-g} < M_{n-g} < L$.
All M_{n-g} values will be positive, but exactly half of the H_{n-g} values will be negative.
If $H_n > L\sqrt{(8)}$, then $H_{n-g} > 0$.
If $H_n < L\sqrt{(8)}$, then $H_{n-g} < 0$.

If $(M_{n-g},-H_{n-g})$ satisfies (1), then so does (M_{n-g},H_{n-g}).
This means that the negative values of H_{n-g} and positive values of H_{n-g} come in pairs of equal absolute value.

Therefore, we can do a faster search for generators by
looking for values of H_{n-g} in the pre-generator range $0 < H_{n-g} < L$,
which is 1/6 the range of $L < H_n < 7L$.
For each of these pre-generators,
we list both positive and negative values of H_{n-g}.
We then put the pre-generators into the forward recursion
to get the generators. Then we add $H_g = 7L$ to the list.

The Prime Factor Reduction also applies to these H_{n-g} values.
It is sufficient to list values of H_{n-g} that have prime factors of only 8k+1 and 8k+7.

Example

For L = 7, g = 3.
In the range 1 ... 7, we find 1 pre-generator.
Listing both the negative and positive pre-generators.

n	1	2
H_{n-g}	-1	1
M_{n-g}	5	5

Putting these through the forward recursion and adding 7L produces

n	1	2	3
H_n	17	23	49

Example

For L = 119 = 7x17, g = 9.
In the range 1 ... 119, we find 4 pre-generators.
Listing both the negative and positive pre-generators.

n	1	2	3	4	5	6	7	8
H_{n-g}	-79	-49	-41	-17	17	41	49	79
M_{n-g}	101	91	89	85	85	89	91	101

Putting these through the forward recursion and adding 7L produces

n	1	2	3	4	5	6	7	8	9
H_n	167	217	233	289	391	479	511	641	833

Composition of Forms

Suppose we have already tested L_u and L_v and have saved the list of pre-generators for each. We then come to $L_w = L_uL_v$. We can directly compute all the pre-generators for L_w using the lists of pre-generators for L_u and L_v.

To do this we use the following composition and scaling formulas.

(8a) $M_w = (2M_uM_v + H_uH_v) - (M_uH_v + H_uM_v)$
(8b) $H_w = |(2M_uM_v + H_uH_v) - 2(M_uH_v + H_uM_v)|$

(9a) $M_w = (2M_uM_v - H_uH_v) - |M_uH_v - H_uM_v|$
(9b) $H_w = (2M_uM_v - H_uH_v) - 2|M_uH_v - H_uM_v|$

(10a) $M_w = M_uL_v$
(10b) $H_w = H_uL_v$

(11a) $M_w = M_vL_u$
(11b) $H_w = H_vL_u$

To use the above formulas for the case where L_u and L_v have no prime factors in common, follow this procedure.

```
For each (Mu,Hu),
  For each (Mv,Hv),
    Use (8a),(8b);
    Use (9a),(9b).

For each Mu,Hu),
  Use (10a),(10b).

For each (Mv,Hv),
  Use (11a),(11b).
```

If L_w is a power of a prime, p^a, then its pre-generators can be computed from the pre-generators for $L_u = p$ and $L_v = p^{a-1}$.
The procedure is different, depending on whether the power is even or odd.
It is also required to keep track of the one primitive pre-generator in each list of pre-generators for prime powers.

For a prime, there is exactly one pre-generator and it is primitive.
For a prime power, follow this procedure.

```
To compute the scaled pre-generators of Lw:
For each (Mv,Hv),
  Use (11a),(11b).

To compute the primitive pre-generator of Lw:
If the power, a, is even
   Use (8a),(8b) on both primitive pre-generators.
Else
   Use (9a),(9b) on both primitive pre-generators.
```

Example

$L_w = 7^2$.
For $L_u = 7$, $(M_u,H_u) = \{(5,1)\}$ and is primitive.
For $L_v = 7$, $(M_v,H_v) = \{(5,1)\}$ and is primitive.
The power of the prime is even, so we use (8a),(8b) and (11a),(11b).

	L_u	M_u	H_u	M_v	H_v	M_w	H_w	
(8a), (8b) -->	--	5	1	5	1	41	31	(primitive)
(11a),(11b) -->	7	-	-	5	1	35	7	

Example

$L_w = 7^3$.

For $L_u = 7$, $(M_u,H_u) = \{(5,1)\}$ and is primitive.

For $L_v = 7^2$, $(M_v,H_v) = \{(35,7), (41,31)\}$,
the last one being primitive.
The power of the prime is odd, so we use (9a),(9b) and (11a),(11b).

	L_u	M_u	H_u	M_v	H_v	M_w	H_w	
(9a), (9b) -->	--	5	1	41	31	265	151	(primitive)
(11a),(11b) -->	7	-	-	41	31	287	217	
(11a),(11b) -->	7	-	-	35	7	245	49	

Example

$L_w = 391 = 17 \times 23$

For $L_u = 23$, $(M_u,H_u) = \{(17,7)\}$.

For $L_v = 17$, $(M_v,H_v) = \{(13,7)\}$.

	L_u	L_v	M_u	H_u	M_v	H_v	M_w	H_w
(8a), (8b) -->	--	--	17	7	13	7	281	71
(9a), (9b) -->	--	--	17	7	13	7	365	337
(10a),(10b) -->	--	17	17	7	--	-	289	119
(11a),(11b) -->	23	--	-	-	13	7	299	161

Example

$L_w = 19159 = 49 \times 391 = 7^2 \times (17 \times 23)$

For $L_u = 49$, $(M_u,H_u) = \{(41,31),(35,7)\}$.

For $L_v = 391$, $(M_v,H_v) = \{(281,71),(289,119),(299,161),(365,337)\}$.

	L_u	L_v	M_u	H_u	M_v	H_v	M_w	H_w
(8a), (8b) -->	--	---	41	31	281	71	13621	1999
(8a), (8b) -->	--	---	41	31	289	119	13549	289
(8a), (8b) -->	--	---	41	31	299	161	13639	2231
(8a), (8b) -->	--	---	41	31	365	337	15245	9887
(8a), (8b) -->	--	---	35	7	281	71	15715	11263
(8a), (8b) -->	--	---	35	7	289	119	14875	8687
(8a), (8b) -->	--	---	35	7	299	161	14329	6601
(8a), (8b) -->	--	---	35	7	365	337	13559	791

	L_u	L_v	M_u	H_u	M_v	H_v	M_w	H_w
(9a), (9b) -->	--	---	41	31	281	71	15041	9241
(9a), (9b) -->	--	---	41	31	289	119	15929	11849
(9a), (9b) -->	--	---	41	31	299	161	16859	14191
(9a), (9b) -->	--	---	41	31	365	337	16981	14479
(9a), (9b) -->	--	---	35	7	281	71	18655	18137
(9a), (9b) -->	--	---	35	7	289	119	17255	15113
(9a), (9b) -->	--	---	35	7	299	161	16261	12719
(9a), (9b) -->	--	---	35	7	365	337	13951	4711

	L_u	L_v	M_u	H_u	M_v	H_v	M_w	H_w
(10a),(10b) -->	--	391	41	31	--	--	16031	12121
(10a),(10b) -->	--	391	35	7	--	--	13685	2737
(11a),(11b) -->	49	---	-	-	281	71	13769	3479
(11a),(11b) -->	49	---	-	-	289	119	14161	5831
(11a),(11b) -->	49	---	-	-	299	161	14651	7889
(11a),(11b) -->	49	---	-	-	365	337	17885	16513

Finding Pre-Generators For Primes

The Pre-Generator Method reduces the problem from scanning for generators in the range L ... 7L to instead scanning in the 1/6 size range 0 ... L.
The Composition of Forms reduces the problem to scanning for pre-generators only for primes, while pre-generators for composites are directly computed.
This section shows how to further reduce the work in finding the pre-generator for a prime by scanning in the much smaller range 0 ... $\sqrt{L}$.
Also, since there is only one pre-generator for a prime, once you find it, you can stop scanning.

Instead of looking for a representation for L^2, look for the representation,

$$L = 2m^2 - h^2,$$

then use one of the composition formulas to compute the pre-generator for L^2.

(12a) $$M = 2m^2 + h^2 - 2mh$$

(12b) $$H = |2m^2 + h^2 - 4mh|$$

Examples

L	m	h	M	H		L	m	h	M	H
7	2	1	5	1		89	7	3	65	23
17	3	1	13	7		97	7	1	85	71
23	4	3	17	7		103	8	5	73	7
31	4	1	25	17		113	9	7	85	41
41	5	3	29	1		127	8	1	113	97
47	6	5	37	23		137	9	5	97	7
71	6	1	61	49		151	10	7	109	31
73	7	5	53	17		191	10	3	149	89
79	8	7	65	47		199	10	1	181	161

Remark

Here are some observations that lead to some extra time savings.
All the h values are odd. An m value is odd when L = 8k+1.
An m value is even when L = 8k+7. Will this always be true?

Proof
We have

$$h^2 = 2m^2 - L.$$

$2m^2$ is even and L is odd, therefore h must be odd.

The square of an odd number has the form 8k+1.
The square of an even number has the form 8k or 8k+4.
We have

$$2m^2 = L + h^2.$$

Since h is odd, h^2 has the form 8k+1.
If L has the form 8k+1, then $2m^2$ must have the form 8k+2, and m must be odd.
If L has the form 8k+7, then $2m^2$ must have the form 8k, and m must be even.

Part 4
Future Research

This part of the paper contains ideas for extending the results to larger numbers and determining places where a magic square might be hiding.

Extending the Results

The enumeration described in the Introduction was performed using the Pre-Generator Method, but without the Composition of Forms or √(L) Prime scanning. Thus, looking for pre-generators consumed most of the time. So the enumeration was aborted after examining values of L up to 7 digits.

Using the new methods should speed up the operation enough so that much larger values for L can be tested. A new implementation would require a technique to build values of L from a list of 8k+1/8k+7 primes, similar to the technique that Christian Boyer used in his search.

I look forward to an independent verification of my results and a possible extension of it, possibly even finding a magic square.

Magic Square Search Ideas

Here are some ideas for trying various values of L to try and find a magic square without enumerating everything.

The best value of L to use for finding a magic square would be one that had a lot of different small prime factors. This is because the density of H values would be greatest, increasing the chance for (4) to be met. Using higher powers of the same prime does not increase the density as much as different primes. This can be seen from the formula for the number of generators.

A procedure to search for a magic square is to try values of L in the following sequence,

7, 7x17, 7x17x23, 7x17x23x31, etc.

including the next prime for each search.
The idea is that if there is a solution using a subset of these primes, then you would find a scaled version of the solution. So there is no need to try every subset when you can do them all at once.
Make sure you have your giant integer arithmetic package ready.