Introduction

The main objective of this note is to describe a connection between multimagic series and certain multivariate polynomials, and to show how these polynomials can be used to compute the number of multimagic series of various types. Multimagic series appear in the context of multimagic squares, cubes, and so on. Christian Boyer's excellent website http://www.multimagie.com contains a lot of interesting information about the subject in general (including many references), and has two pages about multimagic series for squares[1] and cubes[2].

This note starts by setting up a general framework which can be used to attack many similar combinatorial problems. After describing the connection with multivariate polynomials, we show how these polynomials can be used to compute the number of multimagic series of various types by computing the product of a series of simple binomials, reducing the result obtained after each multiplication by dropping irrelevant terms. The final polynomial will always contain only one monomial, and its coefficient provides the final result. Some new exact results for bimagic series are included, as well as comparisons with the best available estimates.

This note follows the commonly used convention that the value of a sum over an empty set is 0, and that the value of a product over an empty set is 1.

General framework

Before discussing the general case covering multimagic series, let's start with another example: the well-known *binomial identity*

$$(x+y)^n = \sum_{k=0}^{n} \binom{n}{k} x^k y^{n-k}$$

The coefficient of the general term in the summation is a binomial coefficient, and represents the number of distinct subsets of k elements that can be chosen from a set of n elements (of course if $k < 0$ or $k > n$ no such subsets exist, and the binomial coefficient is zero). In case you don't immediately see why this is true, check out the sections about the combinatorial interpretation and proof in http://en.wikipedia.org/wiki/Binomial_theorem (example and general case). The intuitive explanation is as follows (I just quote from the Wikipedia page): "if we write $(x+y)^n$ as a product

$$(x+y)(x+y)(x+y)\ldots(x+y),$$

then, according to the distributive law, there will be one term in the expansion for each choice of either x or y from each of the binomials of the product. For example, there will only be one term x^n, corresponding to choosing x from each binomial. However, there will be several terms of the form $x^{n-2}y^2$, one for each way of choosing exactly two binomials to contribute a y. Therefore, after combining like terms, the coefficient of $x^{n-2}y^2$ will be equal to the number of ways to choose exactly 2 elements from an n-element set." Of course we can replace y by 1 to obtain

$$(1+x)^n = \sum_{k=0}^{n} \binom{n}{k} x^k$$

This idea can be taken one step further by replacing the term x in the binomials by powers of x as follows:

$$\prod_{i=1}^{n} \left(1 + x^{f(i)}\right) = \sum_{k=0}^{n} c_k^{(n)} x^k$$

Using the same line of reasoning as above, it can be shown that the coefficient $c_k^{(n)}$ is equal to the number of subsets T that can be chosen from the set $S = \{1,2,\ldots,n\}$ such that $\sum_{i \in T} f(i) = k$.

Note that if we take $f(i) = 1$ for all $i \in S$, then $\sum_{i \in T} f(i)$ is the number of elements of T, so we have

$$c_k^{(n)} = \binom{n}{k}$$

and recovered the form of the binomial identity with $y = 1$.

If we need to satisfy several equations simultaneously we have to introduce more variables. For example, if we write

$$\prod_{i=1}^{n} \left(xy^i + xz^i + y^i z^i\right) = \sum_{u,v,w} c_{u,v,w} x^u y^v z^w$$

then $c_{u,v,w}$ is equal to the number of ways to color n balls numbered from 1 to n using three colors (say red, green and blue), such that the number of balls which are not blue is equal to u, the sum of the numbers on the balls which are not green is equal to v, and the sum of the numbers on the balls which are not red is equal to w. To see why this is true, assume that the factor i corresponds to the ball with number i, and choose the first term xy^i if this ball is red, the second term xz^i if this ball is green, and the third term $y^i z^i$ if this ball is blue.

We will use *multi-index notation* to avoid lengthy expressions: if the variables are indexed from 1 to m, $x = (x_1, x_2, \ldots, x_m)$, and $u = (u_1, u_2, \ldots, u_m) \in \mathbb{Z}^m$, we use the following abbreviated notation:

$$c_u x^u = c_{u_1,u_2,\ldots,u_m} {x_1}^{u_1} {x_2}^{u_2} \ldots {x_m}^{u_m}$$

Note that we can still write $x^u x^v = x^{u+v}$, now for all $u, v \in \mathbb{Z}^m$. A nice thing about this notation is that it helps us to reason about multivariate polynomials as if they are just polynomials in one variable.

To solve enumeration problems for multimagic series, the most general form we need is a product of simple binomials of the form $1 + x^{f(i)}$. Let S be a finite set, and let f be a function with domain S and codomain $\mathbb{Z}^m$. For any subset $T \subseteq S$, let $F(T) = \sum_{i \in T} f(i)$. Then we have

$$\prod_{i \in S} \left(1 + x^{f(i)}\right) = \sum_{T \subseteq S} x^{F(T)}$$

To see why this is true, notice as before that in order to form a single term in the summation on the right, one of both terms must be chosen from each factor (binomial) in the product on the left. For each element $i \in S$ choose the second term ($x^{f(i)}$) whenever $i \in T$, and choose the first term (1) whenever $i \notin T$. After multiplying the chosen terms we obtain the term $x^{F(T)}$ in the summation on the right.

In this summation, *like terms* (terms with matching exponent vectors u) can be combined as before to a single term with an integer coefficient c_u:

$$\sum_{T \subseteq S} x^{F(T)} = \sum_{u} c_u x^u$$

We have shown that the number of ways to choose a subset T of S such that $\sum_{i \in T} f(i) = u$ is equal to (using brackets for the "coefficient of" operator)

$$c_u = [x^u] \prod_{i \in S} \left(1 + x^{f(i)}\right)$$

For example, for bimagic series for squares of order N it is natural to take $S = \{1, 2, \dots, N^2\}$, $f(i) = (1, i, i^2)$ and $u = (N, S_1, S_2)$, where S_1 and S_2 are the magic sums for this type of magic series, hence $S_1 = N(N^2 + 1)/2$ and $S_2 = N(N^2 + 1)(2N^2 + 1)/6$. Then c_u is the number of bimagic series for squares of order N. In this example the first variable is introduced to fix the length of the series, the second variable to fix its sum, and the third variable to fix its sum of squares. The same result can also be found by taking $S = \{0, 1, 2, \dots, N^2 - 1\}$, $f(i) = (1, i, i(i - 1)/2)$ and $u = (N, \sigma_1, \sigma_2)$, where $\sigma_1 = N(N^2 - 1)/2$ and $\sigma_2 = N(N^2 - 1)(N^2 - 2)/6$. We leave the proof to the reader.

As in this example, it is often the case that all the exponents are nonnegative. This does not have to be the case in general, and much of what follows also applies (or can easily be adapted) to the general case. But in the rest of the note we are assuming that f has codomain $\mathbb{Z}_{\geq 0}^m$.

So far we have reformulated the problem of enumerating multimagic series, and many similar problems, to the computation of a certain coefficient of a certain multivariate polynomial. There are several ways to compute such a coefficient. For example, since it is relatively easy to *evaluate* the polynomial

$$P(x) = \prod_{i \in S} \left(1 + x^{f(i)}\right)$$

one can use the multivariate inverse discrete Fourier transform to find $c_u = [x^u]P(x)$. But, even considering the fact that usually the number of terms can be reduced substantially, this approach will quickly result in a very high number of evaluations of $P(x)$. In this note we will use a different approach.

Since S is finite, we can assume without loss of generality that $S = \{0, 1, 2, \dots, M - 1\}$ with $M = |S|$. First consider the following sequence of polynomials:

$$P_n(x) = \prod_{i=n}^{M-1} \left(1 + x^{f(i)}\right), n = 0, 1, \dots, M$$

We can start from the last polynomial $P_M(x) = 1$, and successively compute

$$P_n(x) = \left(1 + x^{f(n)}\right) P_{n+1}(x, y, z)$$

for $n = M - 1$ down to $n = 0$, writing each $P_n(x)$ in the form

$$P_n(x) = \sum_{\alpha} c_{\alpha}^{(n)} x^{\alpha}$$

where $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_m)$. From the final polynomial $P_0(x) = P(x)$ we can extract

$$c_u^{(0)} = [x^u] P_0(x) = [x^u] P(x) = c_u$$

Because we are only interested in one particular coefficient $c_u^{(0)}$ we don't need to compute the others, obviously. As we will see, this will allow us to drop most terms from each of the intermediate polynomials $P_n(x)$ as well. But how do we determine which terms can safely be dropped without affecting the end result? In order to answer this question, consider the "complementary" polynomials, defined by

$$Q_n(x) = \prod_{i=0}^{n-1} \left(1 + x^{f(i)}\right), n = 0, 1, \dots, M$$

Then we have, for $n = 0, 1, \dots, M$,

$$P_n(x) Q_n(x) = P(x)$$

Instead of computing the sequence of polynomials $P_n(x)$, we will compute a sequence of "reduced" polynomials

$$\tilde{P}_n(x) = \sum_{\alpha} \tilde{c}_{\alpha}^{(n)} x^{\alpha}$$

such that the following two conditions are satisfied for $n = 0, \dots, M$:

$$\tilde{c}_{\alpha}^{(n)} = 0 \vee \tilde{c}_{\alpha}^{(n)} = c_{\alpha}^{(n)}, \text{for all } \alpha$$

$$[x^u](\tilde{P}_n(x) Q_n(x)) = [x^u](P_n(x) Q_n(x)) = c_u$$

If we also write $Q_n(x)$ as a sum,

$$Q_n(x, y, z) = \sum_{\alpha} d_{\alpha}^{(n)} x^{\alpha}$$

then the second condition can be expressed as

$$\sum_{\alpha} \tilde{c}_{\alpha}^{(n)} d_{u-\alpha}^{(n)} = \sum_{\alpha} c_{\alpha}^{(n)} d_{u-\alpha}^{(n)} = c_u$$

Therefore, if we know that $d_{u-\alpha}^{(n)} = 0$, we can take $\tilde{c}_{\alpha}^{(n)} = 0$ (i.e., drop the term) without violating the two conditions. For example, because $Q_n(x)$ does not have any terms with negative exponents, we can take $\tilde{c}_{\alpha}^{(n)} = 0$ whenever $u - \alpha \notin \mathbb{Z}_{\geq 0}{}^m$. But we'll see that more terms can be dropped.

Of course we don't want to compute $Q_n(x)$ from its definition to find out which coefficients are equal to zero, and which ones are not. But it is relatively easy to compute bounds for the exponents in the nonzero terms appearing in $Q_n(x)$.

Before explaining how this can be done, we introduce some definitions. We start by rewriting $Q_n(x)$ as a polynomial in the first variable x_1:

$$Q_n(x) = \sum_i d_i^{(n)}(x)\, x_1{}^i$$

where each coefficient $d_i^{(n)}(x)$ is a polynomial that does not contain the variable x_1. Continuing in the same way we can write:

$$d_i^{(n)}(x) = \sum_j d_{i,j}^{(n)}(x)\, x_2{}^j\,;\; d_{i,j}^{(n)}(x) = \sum_k d_{i,j,k}^{(n)}(x) x_3{}^k\,; \ldots$$

where each coefficient $d_{i,j}^{(n)}(x)$ is a polynomial that does not contain the variables x_1 and x_2, each coefficient $d_{i,j,k}^{(n)}(x)$ is a polynomial that does not contain the variables x_1, x_2 and x_3, and so on. Let $L^{(n)}$ and $M^{(n)}$ be the minimal and the maximal value of i such that $d_i^{(n)}(x) \neq 0$. Similary, for all i such that $L^{(n)} \leq i \leq M^{(n)}$, let $L_i^{(n)}$ and $M_i^{(n)}$ be the minimal and the maximal value of j such that $d_{i,j}^{(n)}(x) \neq 0$. Continuing in the same way, for all i and j such that $L^{(n)} \leq i \leq M^{(n)}$ and $L_i^{(n)} \leq j \leq M_i^{(n)}$, let $L_{i,j}^{(n)}$ and $M_{i,j}^{(n)}$ be the minimal and the maximal value of k such that $d_{i,j,k}^{(n)} \neq 0$, and so on. Let $S^{(n)}$ be the set of all $(i, j, k, \ldots) \in \mathbb{Z}_{\geq 0}^m$ satisfying the set of inequalities

$$\begin{aligned}
&L^{(n)} \leq i \leq M^{(n)} \\
&L_i^{(n)} \leq j \leq M_i^{(n)} \text{ for all } i \text{ satisfying the first inequality} \\
&L_{i,j}^{(n)} \leq k \leq M_{i,j}^{(n)} \text{ for all } i, j \text{ satisfying the first two inequalities}
\end{aligned}$$

and so on.

Because $P_n(x)Q_n(x) = P(x)$, we can drop all the terms $c_{\alpha}^{(n)} x^{\alpha}$ from $P_n(x)$ which do not have a complementary term $d_{u-\alpha}^{(n)} x^{u-\alpha}$ with $d_{u-\alpha}^{(n)} > 0$ in $Q_n(x)$, as we already mentioned. We know that $d_{u-\alpha}^{(n)} = 0$ unless $u - \alpha \in S^{(n)}$. So if one or more of the above inequalities are not satisfied, the term $c_{\alpha}^{(n)} x^{\alpha}$ can be dropped from $P_n(x)$, resulting in our reduced polynomial $\tilde{P}_n(x)$. Note that with this particular sequence of reduced polynomials, the final polynomial $\tilde{P}_0(x)$ will consist of only one term: $\tilde{P}_0(x) = c_u^{(0)} x^u$.

Now let's see how the bounds can be computed in general. To simplify the equations we'll extend the above definitions and assume that $L_i^{(n)} = +\infty$ and $M_i^{(n)} = -\infty$ unless $L^{(n)} \leq i \leq M^{(n)}$, that $L_{i,j}^{(n)} = +\infty$ and $M_{i,j}^{(n)} = -\infty$ unless $L^{(n)} \leq i \leq M^{(n)}$ and $L_i^{(n)} \leq j \leq M_i^{(n)}$, and so on.

The case $Q_0(x) = 1$ is trivial: $L^{(0)} = M^{(0)} = L_0^{(0)} = M_0^{(0)} = L_{0,0}^{(0)} = M_{0,0}^{(0)} = \cdots = 0$. Writing $f(n) = (f_1(n), \ldots, f_m(n))$, the other bounds can be computed from the recurrence

$$Q_{n+1}(x) = \left(1 + x^{f(n)}\right) Q_n(x) = \left(1 + x_1{}^{f_1(n)} \ldots x_m{}^{f_m(n)}\right) Q_n(x)$$

So we have

$$\begin{aligned}
L^{(n+1)} &= \min\{L^{(n)}, L^{(n)} + f_1(n)\} = L^{(n)} = 0 \text{ (using } f_1(n) \geq 0) \\
M^{(n+1)} &= \max\{M^{(n)}, M^{(n)} + f_1(n)\} = M^{(n)} + f_1(n) \\
L_i^{(n+1)} &= \min\{L_i^{(n)}, L_{i-f_1(n)}^{(n)} + f_2(n)\} \\
M_i^{(n+1)} &= \max\{M_i^{(n)}, M_{i-f_1(n)}^{(n)} + f_2(n)\} \\
L_{i,j}^{(n+1)} &= \min\{L_i^{(n)}, L_{i-f_1(n),j-f_2(n)}^{(n)} + f_3(n)\} \\
M_{i,j}^{(n+1)} &= \max\{M_i^{(n)}, M_{i-f_1(n),j-f_2(n)}^{(n)} + f_3(n)\}
\end{aligned}$$

and so on. In the next section we will show that the bounds can be computed *directly* for any given n (not by recursion over n as in the general case described here), but keep in mind that this is not always possible in general.

Application to bimagic series

Let's reconsider the problem of determining the number of bimagic series for squares of order N, which as we already mentioned can be found by taking $S = \{0, 1, 2, \ldots, N^2 - 1\}$, $m = 3$, $f_1(i) = 1$, $f_2(i) = i$, $f_3(i) = i(i-1)/2$, $u_1 = N$, $u_2 = \sigma_1$ and $u_3 = \sigma_2$, where $\sigma_1 = N(N^2 - 1)/2 =$ and $\sigma_2 = N(N^2 - 1)(N^2 - 2)/6$. With these settings, c_u will be the number of bimagic series for squares of order N. In this case we have, writing all variables explicitly:

$$P_n(x_1, x_2, x_3) = \prod_{i=n}^{N^2-1} \left(1 + x_1 x_2{}^i x_3{}^{i(i-1)/2}\right); \; Q_n(x_1, x_2, x_3) = \prod_{i=0}^{n-1} \left(1 + x_1 x_2{}^i x_3{}^{i(i-1)/2}\right)$$

for $n = 0, 1, \ldots, N^2$. This paragraph applies to multimagic series for cubes and hypercubes of any dimension as well, provided that every appearance of N^2 is substituted by N^3 (for cubes) or N^4 (for hypercubes of dimension 4), and so on.

Now it is possible to compute the bounds defined at the end of the previous section *directly* for any given n. From the definition of $Q_n(x_1, x_2, x_3)$ it is clear that $L^{(n)} = 0$ and $M^{(n)} = n$. For $L_i^{(n)}$ and $M_i^{(n)}$ we are dealing only with monomials containing $x_1{}^i$, so the exponent of x_2 must be a sum of i distinct numbers taken from $\{0, 1, \ldots, n-1\}$. If the i smallest numbers are taken we obtain

$$L_i^{(n)} = \sum_{j=0}^{i-1} j = \frac{i(i-1)}{2}$$

If the i largest numbers are taken, we obtain

$$M_i^{(n)} = \sum_{j=n-i}^{n-1} j = \frac{n(n-1)}{2} - \frac{(n-i)(n-i-1)}{2} = i(n-1) - L_i^{(n)}$$

For $L_{i,j}^{(n)}$ and $M_{i,j}^{(n)}$ we are dealing only with monomials containing ${x_1}^i {x_2}^j$, so the exponent of x_3 must be the sum of the $k(k-1)/2$ of i distinct numbers k taken from $\{0, 1, \dots, n-1\}$ having sum j. This time we cannot just choose the i smallest or the i largest numbers from the set, because of the additional requirement that the chosen numbers must have sum j. But there is a very simple algorithm to compute all the $L_{i,j}^{(n)}$ and $M_{i,j}^{(n)}$ for any given combination of n and i. Consider an array of i integer elements $a[0], a[1], \dots, a[i-1]$ satisfying $0 \leq a[0] < a[1] < \cdots < a[i-1] \leq n-1$, and start by assigning the smallest possible number from the set $\{0, 1, \dots, n-1\}$ to each element of the array, so $a[0] = 0, a[1] = 1, \dots, a[i-1] = i-1$. This initial assignment will have the smallest possible sum of elements ($L_i^{(n)}$). Now we will perform a number of steps. In each step one of the numbers (and thus also their sum) will be increased by 1, but in such a way that after each step we still have $0 \leq a[0] < a[1] < \cdots < a[i-1] \leq n-1$.

In the first step we can only increase $a[i-1]$ (at least if $i < n$, the case $i = n$ is trivial), but after that there may be more than one element in the array that can be increased without violating the above constraints. Two particular sequences of steps are of interest here. The first sequence, which we will call the *minimal sequence*, will allow us to compute all values $L_{i,j}^{(n)}$ for the given values of n and i. The second sequence, which we will call the *maximal sequence*, will allow us to compute all values $M_{i,j}^{(n)}$ for the given values of n and i.

For the minimal sequence, increase the array element with the *smallest* possible index in each step. For the maximal sequence, increase the array element with the *largest* possible index in each step. Both sequences terminate when no more elements can be increased, which (for both sequences) will be the case when $a[0] = n-i, a[1] = n-i+1, \dots, a[i-1] = n-1$. This final assignment will have the largest possible sum of elements ($M_i^{(n)}$). Note that these rules completely define both sequences.

Now for each array in the minimal sequence we can compute the two sums:

$$v = \sum_{k=0}^{i-1} a[k]\,;\; w = \sum_{k=0}^{i-1} \binom{a[k]}{2}$$

We claim (and prove in the appendix) that $L_{i,v}^{(n)} = w$. Since this is true after each step, and since the sequence starts with the smallest sum and ends with the largest sum, passing over all intermediate sums, we will find all $L_{i,j}^{(n)}$. Repeating the same for the maximal sequence allows us to find all $M_{i,j}^{(n)}$ as well. Of course the sums v and w can easily be computed incrementally from the result of the previous step. This claim can also be stated somewhat more formally and perhaps more precisely as follows. Let $S_i^{(n)}$ be the set of all i-tuples $(a_0, a_1, \dots, a_{i-1})$ of integers $a_0, a_1, \dots, a_{i-1}$ such that $0 \leq a_0 < a_1 < \cdots < a_{i-1} \leq n-1$, and define a partial order $\preceq$ between the elements of $S_i^{(n)}$ such that

$$(a_0, a_1, \dots, a_{i-1}) \preceq (b_0, b_1, \dots, b_{i-1}) \Leftrightarrow \sum_{k=0}^{i-1} a[k] = \sum_{k=0}^{i-1} b[k] \wedge \sum_{k=0}^{i-1} \binom{a[k]}{2} \leq \sum_{k=0}^{i-1} \binom{b[k]}{2}$$

Then we claim that the arrays (i-tuples) in the minimal sequence are the minimal elements of $(S_i^{(n)}, \preceq)$, and similarly that the arrays (i-tuples) in the maximal sequence are the maximal elements of $(S_i^{(n)}, \preceq)$.

For example, with $N = 4$ (note really important here), $n = 10$ and $i = 5$, we have $L^{(10)} = 0$, $M^{(10)} = 10$, $L_5^{(10)} = 10$, $M_5^{(10)} = 35$, and the minimal sequence goes like this (from top to bottom):

$a[0]$	$a[1]$	$a[2]$	$a[3]$	$a[4]$	v	w
0	1	2	3	4	10	10
0	1	2	3	5	11	14
0	1	2	4	5	12	17
0	1	3	4	5	13	19
…	…	…	…	…	…	…
1	3	4	5	6	19	34
2	3	4	5	6	20	35
…	…	…	…	…	…	…
5	6	7	8	9	35	110

The corresponding maximal sequence goes like this (from top to bottom):

$a[0]$	$a[1]$	$a[2]$	$a[3]$	$a[4]$	v	w
0	1	2	3	4	10	10
0	1	2	3	5	11	14
0	1	2	3	6	12	19
0	1	2	3	7	13	25
…	…	…	…	…	…	…
0	1	4	8	9	22	70
0	1	5	8	9	23	74
…	…	…	…	…	…	…
5	6	7	8	9	35	110

For example, we can read from the last columns that $L_{5,12}^{(10)} = 17$ and $M_{5,12}^{(10)} = 19$, implying that for any combination of 5 distinct numbers chosen from $\{0, \dots, 9\}$ with sum $v = 12$ we have $17 \leq w \leq 19$. Note that the sum v ranges over all values such that $L_5^{(10)} \leq v \leq M_5^{(10)}$ as required.

Algorithm (outline)

In this section we slightly refine the basic algorithm. To simplify the discussion we'll make abstraction of the minor complication that we'll actually work with reduced polynomials $\tilde{P}_n(x)$ instead of $P_n(x)$. So the basic idea is to start from the polynomial $P_M(x) = 1$ (where $M = N^2$ for magic squares, $M = N^3$ for magic cubes, and so on), and in each step to compute $P_n(x)$ from $P_{n+1}(x)$ using:

$$P_n(x) = (1 + x^{(1,n,n(n-1)/2)})P_{n+1}(x)$$

Because the larger polynomials $P_n(x)$ are too large to be stored in memory, we will represent each polynomial $P_n(x)$ by a file. In each step we can read the file representing $P_{n+1}(x)$ and write the file representing $P_n(x)$. After successfully completing each step, the file representing $P_{n+1}(x)$ can be deleted.

The general operation of a single step is illustrated in the following diagram, which shows the positions of the two input terms that have to be read in order to produce the next output term (this example applies to bimagic series for squares of order $N = 6$, with $n = 10$). The two arrows in the left column show the position of two "input heads" moving sequentially through the same input file, in synch with the position of the "output head" represented by the arrow in the right column. Note that this diagram is a simplification, because it makes abstraction of the input and output buffers, which may contain thousands or even millions of terms.

$$\begin{array}{cc} & P_{11}(x) \\ & \vdots \end{array} \qquad\qquad\qquad\qquad \begin{array}{c} P_{10}(x) \\ \vdots \end{array}$$

$$\left.\begin{array}{cccccc} I2 \rightarrow & 5x^{(4,93,1144)} & * & x^{(1,10,45)} & = & 5x^{(5,103,1189)} \\ & \vdots & & & & \\ I1 \rightarrow & 2x^{(5,103,1189)} & * & 1 & = & 2x^{(5,103,1189)} \\ & \vdots & & & & \end{array}\right\} \rightarrow \begin{array}{c} \vdots \\ 7x^{(5,103,1189)} \\ \vdots \\ \vdots \end{array} \rightarrow OUT$$

Rather than performing one multiplication per step, it is more efficient to combine two subsequent multiplications. We can do this as follows: if N is even (so also M is even), start from $P_M(x) = 1$ as before, and if N is odd (so also M is odd), start from $P_{M-1}(x)$. In each step we now compute $P_n(x)$ from $P_{n+2}(x)$, where n will always be even, using

$$\begin{aligned} P_n(x) &= (1 + x^{(1,n,n(n-1)/2)})(1 + x^{(1,n+1,n(n+1)/2)})P_{n+2}(x) \\ &= (1 + x^{(1,n,n(n-1)/2)} + x^{(1,n+1,n(n+1)/2)} + x^{(2,2n+1,n^2)})P_{n+2}(x) \end{aligned}$$

This is illustrated in the following diagram, which shows the positions of the four input terms that have to be read in order to produce the next output term (also this example applies to bimagic series for squares of order $N = 6$, with $n = 10$). Now each of the arrows in the left column shows the position of one of the four "input heads" moving sequentially through the same input file, in synch with the position of the "output head" represented by the arrow in the right column.

$$\begin{array}{cc} & P_{12}(x) \\ & \vdots \end{array} \qquad\qquad\qquad\qquad \begin{array}{c} P_{10}(x) \\ \vdots \end{array}$$

$$\left.\begin{array}{cccccc} I4 \rightarrow & x^{(3,82,1089)} & * & x^{(2,21,100)} & = & x^{(5,103,1189)} \\ & \vdots & & & & \\ I3 \rightarrow & x^{(4,92,1134)} & * & x^{(1,11,55)} & = & x^{(5,103,1189)} \\ & \vdots & & & & \\ I2 \rightarrow & 4x^{(4,93,1144)} & * & x^{(1,10,45)} & = & 4x^{(5,103,1189)} \\ & \vdots & & & & \\ I1 \rightarrow & x^{(5,103,1189)} & * & 1 & = & x^{(5,103,1189)} \\ & \vdots & & & & \end{array}\right\} \rightarrow \begin{array}{c} \vdots \\ \vdots \\ \vdots \\ 7x^{(5,103,1189)} \\ \vdots \\ \vdots \\ \vdots \\ \vdots \end{array} \rightarrow OUT$$

Combining more than two multiplications does not help because the number of input heads/buffers and hence the number of reads performed doubles with each additional factor (the number of reads is equal to the number of terms in the pre-factor), whereas the number of steps does not halve any more.

Representation of the polynomials

An important consideration is how to represent the terms of the polynomials $\tilde{P}_n(x)$. Perhaps the main criterion for our purpose is the “compactness” of the representation (the average number of bytes per term). We’ll assume without loss of generality that there are three variables. As a first attempt we could start from

$$\tilde{P}_n(x) = \sum_u \tilde{c}_u^{(n)} x^u = \sum_{u_1,u_2,u_3} \tilde{c}_{u_1,u_2,u_3}^{(n)} x_1^{u_1} x_2^{u_2} x_3^{u_3}$$

and represent this sum as a lexicographically ordered list of all 4-tuples $(u_1, u_2, u_3, \tilde{c}_{u_1,u_2,u_3}^{(n)})$ such that $\tilde{c}_{u_1,u_2,u_3}^{(n)} \neq 0$. To compress this list, for every element $(u_1, u_2, u_3, \tilde{c}_{u_1,u_2,u_3}^{(n)})$ after the first, consider the previous element $(v_1, v_2, v_3, \tilde{c}_{v_1,v_2,v_3}^{(n)})$. Then, if $u_1 = v_1$, drop the first component u_1, reducing the element to a triple. Next, if also $u_2 = v_2$, drop the second component u_2 as well, further reducing the element to a pair. And finally, if also $u_3 = v_3 + 1$, drop the third component u_3 as well, leaving only the coefficient $\check{c}_{u_1,u_2,u_3}^{(n)}$. Especially if $\tilde{P}_n(x)$ has many terms, it will almost always be the case that whenever $\tilde{c}_{u_1,u_2,u_3}^{(n)} \neq 0$, then also $\tilde{c}_{u_1,u_2,u_3-1}^{(n)} \neq 0$. This essentially reduces the representation to a long list of coefficients, with only a very small overhead for the exponents.

Lee Morgenstern’s algorithm

The basic algorithm described in this note is in fact quite similar to the algorithm developed by Lee Morgenstern[3], although his algorithm was derived without using polynomials. His program was used to compute the largest known exact results for bimagic series. It can be shown that his lists $L[r][s]$ can be interpreted as representations of certain polynomials related to polynomials defined in this note, and that his idea of truncating these lists is equivalent to our idea of working with reduced polynomials $\tilde{P}_n(x)$. Because his program keeps all data in memory it is faster, but fails to work for larger values of N.

New results for bimagic series for squares and cubes

With the ingredients described in this note I wrote my own (C++) program to compute the exact number of bimagic series for squares and cubes of order N. The program confirms the known results for squares ($N \leq 28$) and cubes ($N \leq 12$). Of course I also computed some new results:

- exact number of bimagic series for squares of order $N = 29$:

$$397017067970073855910942668942599683652058914$$

- exact number of bimagic series for squares of order $N = 30$:

$$67063309991205148544594890672812817873237628826$$

- exact number of bimagic series for cubes of order $N = 13$:

$$450285458654002877929960$$

The same program can be used for larger values of N (without modifications), and it can be adapted to handle bimagic series for hypercubes as well (by changing one simple expression in the beginning of the program).

The new results are consistent with the known estimates found by Michael Quist[4] (see the tables below). His formula for squares is

$$Est_1(N) = \frac{6}{\pi^{3/2}} \sqrt{\frac{30}{e}} \cdot \frac{(Ne)^N}{N^{15/2}} \left(1 + \frac{1787}{2940} N^{-1}\right)$$

The estimates get better if the next term is added to his asymptotic series. Adding the next term (see the note[(*)] at the end of this section) results in the following approximation:

$$Est_2(N) = \frac{6}{\pi^{3/2}} \sqrt{\frac{30}{e}} \cdot \frac{(Ne)^N}{N^{15/2}} \left(1 + \frac{1787}{2940} N^{-1} - \frac{11522051}{70630560} N^{-2}\right)$$

The results for squares are listed in the following table, where $Ex(N)$ is the exact result (rounded to 7 digits), and the relative errors of the two estimates ($k = 1$ and $k = 2$) are defined as

$$Err_k(N) = (Est_k(N) - Ex(N))/Ex(N)$$

N	$Ex(N)$	$Est_1(N)$	$Err_1(N)$	$Est_2(N)$	$Err_2(N)$
29	$3.970171 \cdot 10^{44}$	$3.971414 \cdot 10^{44}$	$+0.0313\%$	$3.970660 \cdot 10^{44}$	$+0.0123\%$
30	$6.706331 \cdot 10^{46}$	$6.708319 \cdot 10^{46}$	$+0.0296\%$	$6.707127 \cdot 10^{46}$	$+0.0119\%$

Adding the next term should improve the estimates for larger values of N as well.

Quist's formula for *cubes* is

$$Est_1(N) = \frac{6\sqrt{30}}{\pi^{3/2}} \cdot \frac{(N^2 e)^N}{N^{21/2}} \left(1 - \frac{1201}{980} N^{-1}\right)$$

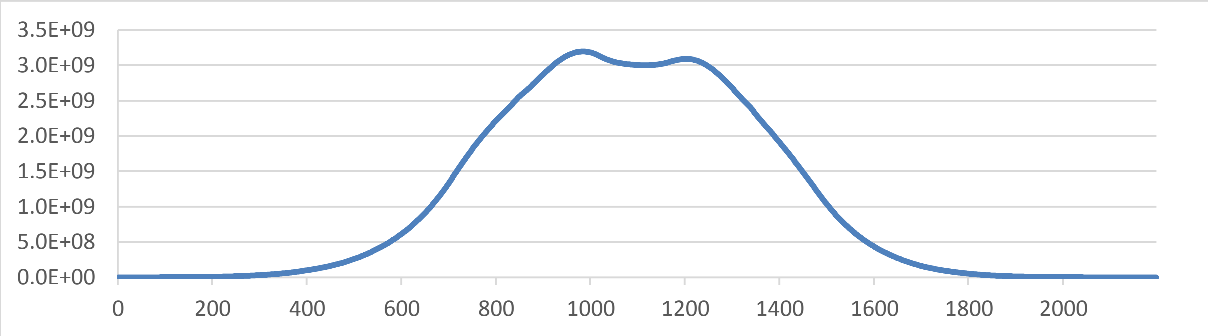
Again, the estimate gets better if the next term is added to his asymptotic series, resulting in the following approximation:

$$Est_2(N) = \frac{6\sqrt{30}}{\pi^{3/2}} \cdot \frac{(N^2 e)^N}{N^{21/2}} \left(1 - \frac{1201}{980} N^{-1} + \frac{16333813}{7847840} N^{-2}\right)$$

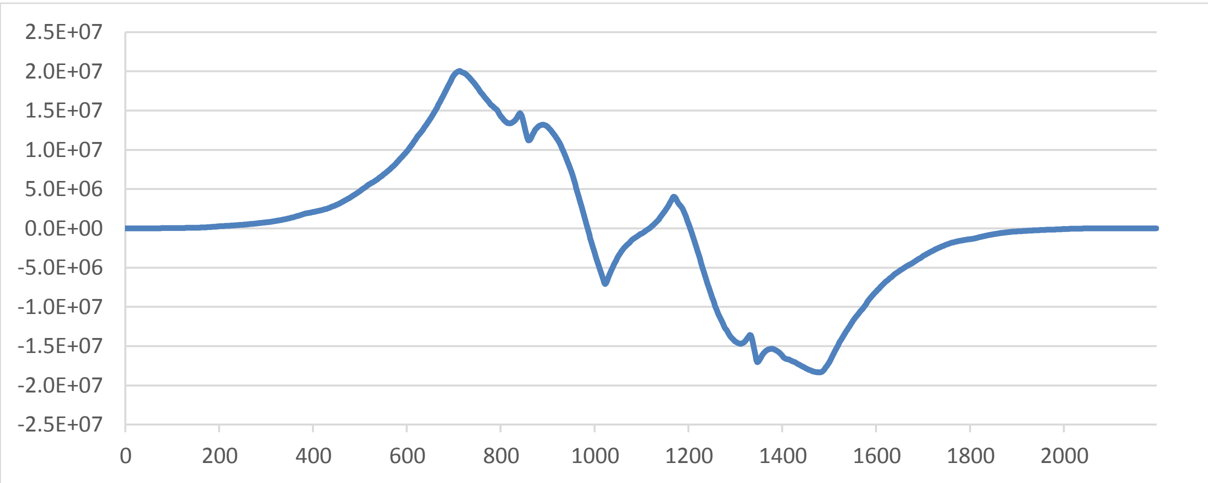
The following table compares the new result for cubes with the two estimates, where $Ex(N)$ is again the exact result (rounded to 7 digits), and the relative errors of the two estimates are defined as above.

N	$Ex(N)$	$Est_1(N)$	$Err_1(N)$	$Est_2(N)$	$Err_2(N)$
13	$4.502855 \cdot 10^{23}$	$4.364515 \cdot 10^{23}$	-3.072%	$4.423861 \cdot 10^{23}$	-1.754%

The following graph shows the number of terms of $\tilde{P}_n(x)$ in function of n, for the case of bimagic series for cubes of order $N = 13$:



Perhaps more interesting is the graph of the first differences:



I don't have an explanation for this rather complex graph. It could be interesting to examine how it changes for different values of N and for different dimensions. The graph of the second differences (not shown here) reveals even more irregularities at a smaller scale.

(*) The computation of the extra terms in the formulas for $Est_2(N)$ is too complicated to describe in this note (I found a way to compute any number of terms of the non-Gaussian factor defined in Michael Quist's paper, without computing all the associated diagrams). Currently I only have detailed notes for ordinary magic series. For bimagic series I just have plain text draft notes in Dutch containing the derivations of all the required formulas, but if anyone is interested I could spend some time to make them readable...

Appendix: proofs related to the minimal and maximal sequences

Here we will prove that if for any array $a[0], a[1], \dots, a[i-1]$ in the minimal sequence we compute

$$v = \sum_{k=0}^{i-1} a[k]\,;\ w = \sum_{k=0}^{i-1} \binom{a[k]}{2}$$

then it is always the case that $L_{i,v}^{(n)} = w$, in other words, that this array has the smallest w of all possible arrays of i strictly increasing numbers taken from $\{0, 1, \dots, n-1\}$ having sum v. We will also prove that if we do the same for the maximal sequence, then it is always the case that $M_{i,v}^{(n)} = w$, in other words, that this array has the largest w of all possible arrays of i strictly increasing numbers taken from $\{0, 1, \dots, n-1\}$ having sum v.

Let's start with the (maybe somewhat sketchy) proof for the minimal sequence. Because of the way in which the minimal sequence is constructed, it is immediately clear that each array in the sequence has a very specific form: either the array consists of i consecutive integers (the initial and the final array in the sequence are examples of this case), or the array consists of two blocks of consecutive integers, with a gap of length 1 between them. As an example, take the table of the minimal sequence from our previous example with $N = 4, n = 10$ and $i = 5$. One of the arrays in this sequence is $a[0] = 0, a[1] = 1, a[2] = 3, a[3] = 4, a[4] = 5$. As shown in the diagram below, the first two elements of this array form a first block (containing the consecutive integers 0 and 1), and the other three elements form the second block (containing the consecutive integers 3, 4 and 5). The gap between the two blocks has length 1 as required (only the number 2 is missing).

<table>
<tr><td>0</td><td>1</td><td>2</td><td>3</td><td>4</td><td>5</td><td>6</td><td>7</td><td>8</td><td>9</td></tr>
<tr><td>$a[0]$</td><td>$a[1]$</td><td></td><td>$a[2]$</td><td>$a[3]$</td><td>$a[4]$</td><td colspan="4"></td></tr>
<tr><td colspan="2">block</td><td>gap</td><td colspan="3">block</td><td colspan="4">gap</td></tr>
</table>

It is also clear that every strictly increasing array of integers taken from $\{0, 1, \dots, n-1\}$ with this form will appear in the minimal sequence.

Now we must prove that every strictly increasing array of integers taken from $\{0, 1, \dots, n-1\}$ with a given sum v has a larger w than the (unique) array with the same sum v from the minimal sequence. We do this by transforming the initial array in a finite number of reduction steps, where each step keeps the sum v constant, but strictly decreases w, and such that after the final step we always obtain that unique array with the same sum v from the minimal sequence. If we can prove that such a reduction is always possible, the proof is complete.

The reduction step can be defined as follows (assuming that the array a is not already in the required form, after which no more steps are needed). First choose two array elements $a[r]$ and $a[s]$, with $r + 1 < s - 1$ and $a[r] + 1 < a[r+1] < a[s-1] < a[s] - 1$. This is always possible: choose two consecutive blocks with a gap of length greater than 1 between them (which is always possible if the array a is not already in the required form), and take the largest element of the first block as $a[r]$, and the smallest element of the second block as $a[s]$). Then, add 1 to the array element $a[r]$ and subtract 1 from the array element $a[s]$. The resulting array a' will still be a strictly increasing array of integers taken from $\{0, 1, \dots, n-1\}$ with the same sum v. We still have to show that each step strictly decreases w:

$$w' - w = \sum_{k=0}^{i-1} \left[\binom{a'[k]}{2} - \binom{a[k]}{2} \right] = \binom{a'[r]}{2} - \binom{a[r]}{2} + \binom{a'[s]}{2} - \binom{a[s]}{2}$$
$$= \binom{a[r]+1}{2} - \binom{a[r]}{2} + \binom{a[s]-1}{2} - \binom{a[s]}{2} = a[r] - (a[s]-1) < 0$$

This completes the proof for the minimal sequence.

The proof for the maximal sequence is very similar. Because of the way in which the maximal sequence is constructed, it is immediately clear that each array in the sequence has the following form: either the array consists of i consecutive integers (the initial and the final array in the sequence are examples of this case), or the array contains a single gap, or exactly two gaps with a single block of length 1 between them. As an example, consider the maximal sequence from the previous example with $N = 4, n = 10$ and $i = 5$. One of the arrays in this sequence is $a[0] = 0, a[1] = 1, a[2] = 4, a[3] = 8, a[4] = 9$. As shown in the diagram below, there are exactly two gaps in this array, the first one between the elements $a[1]$ and $a[2]$ (the numbers 2 and 3 are missing), and the other one between the elements $a[2]$ and $a[3]$ (the numbers 5, 6 and 7 are missing), and there is a single block between the gaps (containing the number 4), with length 1 as required.

0	1	2	3	4	5	6	7	8	9
$a[0]$	$a[1]$			$a[2]$				$a[3]$	$a[4]$
block		gap		block	gap			block	

It is also clear that every strictly increasing array of integers taken from $\{0, 1, \dots, n-1\}$ with this form will appear in the maximal sequence.

Now we must prove that every strictly increasing array of integers taken from $\{0, 1, \dots, n-1\}$ with a given sum v has a smaller w than the (unique) array with the same sum v from the maximal sequence. We do this by transforming the initial array in a finite number of reduction steps, where each step keeps the sum v constant, but strictly increases w, and such that after the final step we always obtain that unique array with the same sum v from the maximal sequence. If we can prove that such a reduction is always possible, the proof is complete.

The reduction step can now be defined as follows (assuming that the array a is not already in the required form, after which no more steps are needed). First choose two array elements $a[r]$ and $a[s]$, with $r < s$ and $a[r] - 1 > a[r-1]$ (or $a[r] > 0$ if $r = 0$) and $a[s] + 1 < a[s+1]$ (or $a[s] < n - 1$ if $s = i - 1$). This is always possible: choose one or more blocks not containing 0 or $n - 1$ with total length greater than 1 (which is always possible if the array a is not already in the required form), and take the smallest element of these blocks as $a[r]$, and the largest element as $a[s]$. Then, subtract 1 from the array element $a[r]$ and add 1 to the array element $a[s]$. The resulting array a' will still be a strictly increasing array of integers taken from $\{0, 1, \dots, n-1\}$ with the same sum v. We still have to show that each step strictly increases w:

$$
\begin{aligned}
w' - w &= \sum_{k=0}^{i-1} \left[\binom{a'[k]}{2} - \binom{a[k]}{2} \right] = \binom{a'[r]}{2} - \binom{a[r]}{2} + \binom{a'[s]}{2} - \binom{a[s]}{2} \\
&= \binom{a[r]-1}{2} - \binom{a[r]}{2} + \binom{a[s]+1}{2} - \binom{a[s]}{2} = -(a[r]-1) + a[s] > 0
\end{aligned}
$$

This completes the proof for the maximal sequence.

Dirk Kinnaes

August 2015

References

[1] Christian Boyer, Multimagic series for squares
http://www.multimagie.com/English/Series.htm

[2] Christian Boyer, Multimagic series for cubes
http://www.multimagie.com/English/CubeSeries.htm

[3] Lee Morgenstern, Counting Magic and Multimagic Series
http://home.earthlink.net/~morgenstern/magic/series/counting.htm

[4] Michael Quist, Asymptotic enumeration of magic series
http://arxiv.org/abs/1306.0616