NEWMAN POLYNOMIALS WITH PRESCRIBED VANISHING AND INTEGER SETS WITH DISTINCT SUBSET SUMS

PETER BORWEIN AND MICHAEL J. MOSSINGHOFF

ABSTRACT. We study the problem of determining the minimal degree $d(m)$ of a polynomial that has all coefficients in $\{0,1\}$ and a zero of multiplicity m at -1. We show that a greedy solution is optimal precisely when $m \le 5$, mirroring a result of Boyd on polynomials with ± 1 coefficients. We then examine polynomials of the form $\prod_{e \in E}(x^e+1)$, where E is a set of m positive odd integers with distinct subset sums, and we develop algorithms to determine the minimal degree of such a polynomial. We determine that $d(m)$ satisfies inequalities of the form $2^m + c_1 m \le d(m) \le \frac{103}{96} \cdot 2^m + c_2$. Last, we consider the related problem of finding a set of m positive integers with distinct subset sums and minimal largest element and show that the Conway-Guy sequence yields the optimal solution for $m \le 9$, extending some computations of Lunnon.

1. INTRODUCTION

A Newman polynomial is a univariate polynomial with all coefficients in $\{0,1\}$. We denote the set of such polynomials by $\mathcal{N}$. In this paper, we study Newman polynomials having a zero of prescribed multiplicity at $x = -1$ and degree as small as possible. To this end, we define

$$d(m) = \min\{\deg(f) : (x+1)^m \mid f(x) \text{ and } f \in \mathcal{N}\}. \tag{1}$$

Similar quantities have been studied for other families of polynomials. Boyd [4, 5] investigates an analogous problem for the class $\mathcal{L}$ of Littlewood polynomials, which have all coefficients in $\{-1,1\}$. Byrnes asked if the minimal degree of a Littlewood polynomial having a zero of multiplicity m at $x = 1$ is $2^m - 1$, since the polynomial

$$\prod_{k=1}^{m} \left(x^{2^{k-1}} - 1\right) \tag{2}$$

satisfies the criteria. Boyd proves that this greedy construction is optimal precisely for $m \le 5$, determines the minimal degrees for $m = 6$ and 7, and shows that the minimal degree in general exceeds $\exp(\sqrt{m}(1+o(1)))$.

The authors [3] study the similar problem of minimizing the degree of a polynomial with $\{-1,0,1\}$ coefficients and prescribed vanishing at $x = 1$. The minimal degree $d_1(m)$ in this case is known to satisfy $m^2 \ll d_1(m) \ll m^2 \log m$, and we determine the exact value of $d_1(m)$ for $m \le 10$ and find explicit upper bounds for $m \le 21$. Central to this work and the research of Boyd is the observation that

1991 *Mathematics Subject Classification.* Primary: 11C08, 11B75, 11P99; Secondary: 05D99, 11Y55.

Key words and phrases. Newman polynomial; pure product; sum-distinct sets; Conway-Guy sequence.

polynomials with low height, bounded degree, and a high order of vanishing at $x = 1$ must also vanish at pth roots of unity, for several small p.

Of course, since the set of allowed coefficients is symmetric in these other two problems, the results apply equally well when we require m-fold vanishing at -1 instead of 1. Clearly $f(1) > 0$ for a nonzero $f \in \mathcal{N}$, so it is natural to inquire about vanishing at -1 in this case.

The distribution of complex zeros of Newman polynomials is studied by Odlyzko and Poonen in [13].

In Section 2, we develop an algorithm for computing $d(m)$, trimming the search space by exploiting some arithmetic properties of the terms appearing in an extremal polynomial. We show that a simple greedy construction analogous to (2) for Littlewood polynomials is optimal for $m \le 5$ but not for $m \ge 6$, just as in Boyd's result.

In Section 3, we study polynomials of the form

$$\prod_{k=1}^{m} (x^{e_k} + 1). \tag{3}$$

We call such a polynomial a *pure product* and denote it by $[e_1, \ldots, e_k]$. Certainly a pure product is a Newman polynomial if and only if the subsets of $\{e_1, \ldots, e_m\}$ have distinct sums, and we say a set with this property is *sum-distinct*. We are interested in the case where the e_k are all odd integers, so that the resulting polynomial has an m-fold zero at -1. Let

$$e(m) = \min \left\{ \sum_{u \in U} u : U \subset 2\mathbf{Z}^+ - 1,\ |U| = m, \text{ and } U \text{ is sum-distinct} \right\}. \tag{4}$$

We develop an algorithm to compute $e(m)$ and use it to find the minimal pure product polynomials for $m \le 11$. Our results allow us to refine our upper bound on $d(m)$ for larger m.

In Section 4, we apply our methods to a similar problem posed by Erdős: Determine the smallest possible value of the largest element of a sum-distinct set. Let

$$w(m) = \min \left\{ \max(U) : U \subset \mathbf{Z}^+,\ |U| = m, \text{ and } U \text{ is sum-distinct} \right\}. \tag{5}$$

Erdős offered \$500 to determine if $w(m) = \Omega(2^m)$, and Guy [8] makes the stronger conjecture that $w(m) > 2^{m-3}$. This problem also appears in [9, problem C8]. The best known asymptotic result on this problem is due to Bohman [2], who shows that $w(m) < .22002 \cdot 2^m$ for sufficiently large m. This slightly improves the bound of $.23513 \cdot 2^m$ given by the Conway-Guy sequence, defined by

$$\begin{aligned} &u_0 = 0, \quad u_1 = 1, \\ &u_{m+1} = 2u_m - u_{m-\langle\sqrt{2m}\rangle}, \quad m \ge 1, \end{aligned} \tag{6}$$

where $\langle \alpha \rangle$ denotes the integer nearest α. The first few terms of this sequence are

$$0, 1, 2, 4, 7, 13, 24, 44, 84, 161, 309, 594, 1164, 2284, 4484, 8807, \ldots. \tag{7}$$

Let U_m denote the set $\{u_m - u_k : 0 \le k < m\}$. Conway and Guy [6] show that U_m is sum-distinct for $m \le 40$, Lunnon [10] for $m \le 79$, and Bohman [1] for all positive m. Lunnon also shows that $w(m) = u_m$ for $m \le 8$. We show that $w(9) = u_9$ and that U_9 is the unique optimal set.

2. Optimal Newman polynomials

In this section, we first determine an upper bound on $d(m)$ by explicitly constructing Newman polynomials with high orders of vanishing at -1. We then derive some properties of the optimal Newman polynomials we seek, and use them to develop an algorithm for finding them.

2.1. **Properties of optimal polynomials.** A sequence is *superincreasing* if each element in the sequence is larger than the sum of all the preceding elements. If $\{e_1, \ldots, e_m\}$ is a superincreasing sequence of positive odd integers, then clearly the pure product (3) is a Newman polynomial with a zero of order m at $x = -1$. We may therefore derive an upper bound on $d(m)$ by using the following greedy construction. Define the sequence $\{J_k\}$ by

$$\begin{aligned} J_1 &= 1, \\ J_k &= r_k + \sum_{i=1}^{k-1} J_i, \quad k \geq 2, \end{aligned}$$

where $r_k = 1$ if k is odd and $r_k = 2$ if k is even. Setting $J_0 = 1$ for convenience, it is easy to derive the equivalent recurrence

$$\begin{aligned} J_0 &= J_1 = 1, \\ J_k &= J_{k-1} + 2J_{k-2}, \quad k \geq 2, \end{aligned}$$

and a routine calculation shows that

$$J_k = \frac{1}{3}\left(2^{k+1} + (-1)^k\right).$$

The sequence $\{J_k\}$ is called the *Jacobsthal sequence*. Many of its properties appear in [**?**]. Let

$$g_m(x) = \prod_{k=1}^{m} \left(x^{J_k} + 1\right). \tag{8}$$

We conclude that

$$d(m) \leq \deg(g_m) = \frac{4}{3} \cdot 2^m - \frac{3}{2} + \frac{(-1)^m}{6}. \tag{9}$$

We would like to determine if this greedy construction is optimal.

We require the following notation. For a polynomial $f \in \mathcal{N}$, let $E(f)$ denote the set of exponents appearing in f, so $f(x) = \sum_{e \in E(f)} x^e$. Let $E_0(f)$ denote the subset of even integers in $E(f)$, and $E_1(f)$ the subset of odd integers. Let $N(f)$ denote the number of monomials of f, so $N(f) = |E(f)|$, and let $N_{a,b}(f)$ denote the number of elements of $E(f)$ that are congruent to a modulo b. Last, let $\Phi_n(x)$ denote the nth cyclotomic polynomial.

The following lemma lists some properties of the polynomials we seek.

Lemma 1. *Suppose $f \in \mathcal{N}$ has a zero of order m at -1 and $\deg(f) = d(m)$. Then*

(*i*) $N(f) = 2^m$.

(*ii*) *For $0 \leq k < m$, we have* $\displaystyle\sum_{r \in E_0(f)} r^k = \sum_{s \in E_1(f)} s^k$.

(*iii*) *If $m \geq 4$ and $k \geq 0$ then 2^{k+1} divides* $\displaystyle\sum_{r \in E_0(f)} r^k$.

(*iv*) *If* $m = 5$, *then* $N_{0,4}(f) \equiv N_{2,4}(f) \equiv 2 \pmod 4$; *if* $m \geq 6$, *then* $N_{0,4}(f) \equiv N_{2,4}(f) \equiv 0 \pmod 4$.

(*v*) *If* $m \geq 5$ *then the* 4*-tuple* $(N_{1,8}(f), N_{3,8}(f), N_{5,8}(f), N_{7,8}(f))$ *is congruent modulo* 4 *to one of the tuples shown in Table* 1.

(*vi*) *If* $\Phi_n(x) \mid f(x)$ *and* $n > 2$ *then* n *is not a prime power.*

TABLE 1. Allowed values of $(N_{1,8}(f), N_{3,8}(f), N_{5,8}(f), N_{7,8}(f))$ mod 4.

$m = 5$	$m \geq 6$
$(2,1,0,1)$	$(0,0,0,0)$
$(2,3,0,3)$	$(0,2,0,2)$
$(0,3,2,3)$	$(2,2,2,2)$
$(0,1,2,1)$	$(2,0,2,0)$
$(1,0,1,2)$	$(1,1,3,3)$
$(1,2,1,0)$	$(1,3,3,1)$
$(3,2,3,0)$	$(3,3,1,1)$
$(3,0,3,2)$	$(3,1,1,3)$

Proof. First, since $(x+1)^m \mid f(x)$ and $N(f) = f(1)$, clearly $2^m \mid N(f)$. Using (8), we have $\deg(f) < \frac{4}{3}(2^m - 1) < 2^{m+1} - 1$, so $N(f) < 2^{m+1}$, and hence $N(f) = 2^m$.

Second, let D denote the operation $D(f) = xf'(x)$, and let D^k denote its k-fold iterate. The m-fold zero of f at -1 implies that $D^k(f)(-1) = 0$ for $0 \leq k < m$, which yields (ii).

Third, from (i) and (ii), we have $|E_0(f)| = |E_1(f)| = 2^{m-1}$. Thus,

$$\sum_{s \in E_1(f)} s^2 \equiv |E_1(f)| \equiv 0 \pmod 8$$

since $m \geq 4$. Also, for $k \geq 1$, note that

$$\sum_{r \in E_0(f)} \left(\frac{r}{2}\right)^k \equiv N_{2,4}(f) \pmod 2,$$

and using (ii) to combine these facts we conclude that $N_{2,4}(f)$ is even.

Fourth, if $m \geq 5$, then using (ii) and (iii) we find

$$\sum_{s \in E_1(f)} s^4 \equiv N_{1,8}(f) + N_{7,8}(f) + 17(N_{3,8}(f) + N_{5,8}(f)) \equiv 0 \pmod{32}.$$

Since $|E_1(f)| = 2^{m-1}$, it follows that

$$16(N_{3,8}(f) + N_{5,8}(f)) + 2^{m-1} \equiv 0 \pmod{32}.$$

Consequently,

$$N_{3,8}(f) + N_{5,8}(f) \equiv N_{1,8}(f) + N_{7,8}(f) \equiv 2^{m-5} \pmod 2. \tag{10}$$

Next, reducing the identity in (ii) modulo 16 with $k = 2$, we see that

$$N_{1,8}(f) + N_{7,8}(f) + 9(N_{3,8}(f) + N_{5,8}(f)) \equiv 4N_{2,4}(f) \pmod{16},$$

so

$$8(N_{3,8}(f) + N_{5,8}(f)) \equiv 4N_{2,4}(f) \pmod{16}.$$

Combining this with (10), we conclude that $N_{2,4}(f) \equiv 2^{m-4} \bmod 4$. Since $|E_0(f)| = 2^{m-1}$, clearly the same congruence holds for $N_{0,4}(f)$.

Fifth, since

$$\sum_{s \in E_1(f)} s^3 \equiv N_{1,8}(f) + 3N_{3,8}(f) + 5N_{5,8}(f) + 7N_{7,8}(f) \equiv 0 \mod 8,$$

we have

$$N_{3,8}(f) + 2N_{5,8}(f) \equiv N_{7,8}(f) \mod 4.$$

Combining this observation with (10) and the fact that $|E_1(f)| \equiv 0 \bmod 4$ yields Table 1.

Last, from the proof of (i) we see that if g is an irreducible factor of f and $g(-1) \neq 0$, then $g(1) = \pm 1$. However, if p is a prime number and $k \geq 1$ then $\Phi_{p^k}(1) = p$. □

We use part (i) of the lemma to derive a lower bound on $d(m)$.

Corollary 1. *For every positive integer m, we have $d(m) \geq 2^m + (m-3)/2$.*

Proof. Suppose $f(x) \in \mathcal{N}$ has a zero of order m at $x = -1$ and that $\deg(f) = d(m)$. Let $r = \deg(f) + 1 - N(f) = d(m) + 1 - 2^m$, so r is the number of coefficients of $f(x)$ equal to 0. Let $h(x) = (x+1)f(-x)$. Then $N(h) \leq 2r + 2$, so by Descartes' rule of signs, the polynomial $h(x)$ has at most $2r + 1$ positive real zeros counting multiplicities. Hence $m \leq 2r + 1$, and the inequality follows. □

2.2. **Algorithm.** We use these properties to develop an algorithm to determine the exact value of $d(m)$ for several m. We first implement a simple procedure in Maple to test all Newman polynomials $f(x)$ of a particular degree having $|E_0(f)| = |E_1(f)| = 2^{m-1}$. Using this program we find that $d(m) = \deg(g_m)$ for $m \leq 4$, and that $g_m(x)$ is the unique solution in this range.

For $m \geq 5$ we implement a more sophisticated algorithm that uses more of the properties given by Lemma 1 to narrow the search by more than a factor of 1000. Parts (ii), (iv), and (v) of the lemma are particularly helpful. Part (vi) does not play a role in our method, but we include it in the lemma to contrast our problem with the similar problems studied by Boyd and the authors ([3], [4], [5]), where the extremal polynomials are known to be divisible by several cyclotomic polynomials with small index.

Algorithm 1. *Newman polynomials with low degree and prescribed vanishing at -1.*

Input. Positive integers m and d, with $m \geq 5$ and $2^m \leq d \leq \deg(g_m)$.
Output. All Newman polynomials with degree d and a zero of order m at -1.
Invariants. E is a set of nonnegative integers with $\min(E) = 0$ and $\max(E) = d$, $f(x) = \sum_{e \in E} x^e$, and for every k with $1 \leq k \leq m'$, the number $S_k = \sum_{e \in E} (-1)^e e^k$. Here, m' is a positive integer with $m' \leq m - 1$ whose value is selected to minimize the computation time. (We use $m' = 4$.)

Description.

Step 0. Initialize.
Set $E = \{0, \ldots, d\}$, $n_0 = \lceil (d+1)/2 \rceil - 2^{m-1}$, $n_1 = \lfloor (d+1)/2 \rfloor - 2^{m-1}$, and $B = \sum_{i=1}^{n_0} (2i)^{m'}$. The quantity n_0 is the number of even integers

we must eliminate from E in order to create a candidate Newman polynomial and n_1 is the number of odd integers.

Step 1. Select odd residue classes.
For each tuple (r_1, r_3, r_5, r_7) in Table 1 for the given m, do Step 2.

Step 2. Remove combinations of odd integers from E.
We enumerate all sets T of positive odd integers with $|T| = n_1$ and $\max(T) \le d-1$ such that $|\{e \in E \setminus T : e \equiv i \mod 8\}| \equiv r_i \bmod 4$ for $i \in \{1, 3, 5, 7\}$. We use a backtracking strategy with four levels to accomplish this. Each level handles a different congruence class mod 8, and has two main parts. An outer loop runs through the different possible sizes of sets of integers to select from this class for T so that the corresponding congruence condition is satisfied. An inner loop uses the revolving door algorithm [12] to enumerate all subsets from the congruence class of the desired size. Under this algorithm, each subset constructed differs from the previous one in a minimal way: One element is inserted and another is deleted. This allows us to update E, $f(x)$, and the S_k quickly each time we alter T.
For each combination of odd integers T removed from E, perform Step 3.

Step 3. Test impasse condition.
If $S_{m'} < B$, then clearly we cannot reduce $S_{m'}$ to zero by eliminating n_0 even integers from E. Thus, if this is the case, return to Step 2; otherwise continue with Step 4.

Step 4. Remove combinations of integers congruent to 2 mod 4 from E.
As in Step 2, an outer loop runs through the possible sizes $p \le n_0$ of sets of integers congruent to 2 mod 4 to remove from E in order for the congruence required in part (iv) of Lemma 1 to hold. An inner loop uses the revolving door algorithm to construct all such subsets of the given size p. Set $q = n_0 - p$.

Step 5. Test impasse conditions.
We test two necessary conditions. First, we ensure that $S_k \equiv 0 \bmod 4^k$ for each k, since subsequent changes to E affect only integers divisible by 4. Our enumeration strategy in fact guarantees this condition for $k = 1$ and 2, so we test this for $3 \le k \le m'$ only. Second, we check if it is possible to reduce the value of S_1 to 0 by eliminating q integers divisible by 4 from $E \setminus \{d\}$. We require $2q(q+1) \le S_1 \le 4tq - 2q(q-1)$, where $t = \lfloor (d-1)/4 \rfloor$. If both conditions are satisfied, continue with Step 6; otherwise, return to Step 4.

Step 6. Remove combinations of integers congruent to 0 mod 4 from E.
We enumerate all subsets V of size q from $\{4, 8, \ldots, 4t\}$ with sum S_1. Step 5 guarantees that there exists at least one such subset. To this end, note that if $\max(V) = 4r$ then $S_1 - 4r$ must lie between the sum of the $q-1$ smallest and $q-1$ largest elements of $\{4, 8, \ldots, 4r-4\}$. This yields $S_1/q + 2(q-1) \le 4r \le S_1 - 2q(q-1)$. We therefore obtain the following recursive algorithm: If $q = 1$ return $\{S_1\}$; otherwise, for each r satisfying the inequalities above, adjoin $4r$ to each subset of $\{4, 8, \ldots, 4r-4\}$ having

cardinality $q-1$ and sum S_1-4r. We update E, $f(x)$, and the S_k each time an r is selected, and invoke Step 7 on each polynomial constructed.

Step 7. Test $f(x)$.
If $S_k = 0$ for $2 \le k \le m'$ and $f^{(k)}(-1) = 0$ for $m'+1 \le k \le m-1$, then print $f(x)$.

2.3. **Results.** We implement this algorithm in C++ and use it to determine the value of $d(m)$ for $m = 5$ and $m = 6$. We find that the greedy construction is optimal for $m = 5$, although it is not unique. Another solution is given by the pure product $[1,5,7,9,19]$, and a third is the pure product $[1,3,5,7,11]$ multiplied by the noncyclotomic polynomial

$$h_5(x) = x^{14} + x^{12} - x^{11} - x^9 + x^8 - x^7 + x^6 - x^5 - x^3 + x^2 + 1.$$

The program requires about 0.2 seconds on a Sun UltraSPARC to complete this search, invoking Step 7 on 1578 polynomials.

We find that the greedy construction is not optimal for $m = 6$. The optimal Newman polynomial is unique and has degree 76, while $\deg(g_6) = 84$. The optimal polynomial is the pure product $[1,3,5,7,13,17]$ multiplied by the noncyclotomic polynomial

$$\begin{aligned} h_6(x) = {} & x^{30} - x^{27} + x^{26} - x^{25} + x^{24} - x^{23} + x^{22} - x^{21} + 2x^{20} \\ & - x^{19} + x^{18} - 2x^{17} + x^{16} - x^{15} + x^{14} - 2x^{13} + x^{12} \\ & - x^{11} + 2x^{10} - x^9 + x^8 - x^7 + x^6 - x^5 + x^4 - x^3 + 1. \end{aligned}$$

The program requires about 4.5 hours to determine this, invoking Step 7 on about 936 million polynomials.

We summarize our results in Table 2.

TABLE 2. Optimal Newman polynomials.

m	$d(m)$	$\deg(g_m)$	Optimal polynomials
1	1	1	$[1]$
2	4	4	$[1,3]$
3	9	9	$[1,3,5]$
4	20	20	$[1,3,5,11]$
5	41	41	$[1,3,5,11,21]$ $[1,5,7,9,19]$ $[1,3,5,7,11] \cdot h_5(x)$
6	76	84	$[1,3,5,7,13,17] \cdot h_6(x)$

We obtain an improved upper bound on $d(m)$ for $m \ge 7$ by analyzing the polynomials in this table. Let $\{a_1, \ldots, a_5\}$ denote the sequence of exponents in the pure product $[1,5,7,9,19]$. Because this polynomial has no x^2, x^3, or x^4 term, it is easy to see that we can extend this sequence so that for any m the polynomial $[a_1, \ldots, a_m] \in \mathcal{N}$ and has an m-fold zero at -1. We set

$$a_k = r_k + \sum_{i=1}^{k-1} a_i \tag{11}$$

for $k \geq 6$, where $r_k = -3$ if k is odd and $r_k = -4$ if k is even. Equivalently, $a_k = a_{k-1} + 2a_{k-2}$ for $k \geq 6$, and we calculate

$$d(m) \leq \sum_{k=1}^{m} a_k = \frac{7}{6} \cdot 2^m + \frac{7}{2} - \frac{(-1)^m}{6}$$

for $m \geq 7$.

Our solution for $m = 6$ yields a slightly better upper bound. Because this polynomial has no x^2 or x^3 term, we may extend the sequence $\{b_k\}$ of exponents appearing in the pure product by defining

$$b_k = r_k + \sum_{i=1}^{k-1} b_i$$

for $k \geq 6$, where $r_k = -3$ if k is odd and 2 if k is even. We find

$$d(m) \leq \deg(h_6) + \sum_{k=1}^{m} b_k = \frac{7}{6} \cdot 2^m + \frac{1}{2} + \frac{5}{6}(-1)^m$$

for $m \geq 7$. In the next section, we obtain substantially better bounds on $d(m)$ by studying $e(m)$.

3. Sum-distinct sets of odd integers

We develop an algorithm to determine sum-distinct sets of positive odd integers with minimal sum. Our results allow us to derive an improved bound on $d(m)$. Before describing our algorithm, we first investigate a property that often arises in the study of sum-distinct sets.

3.1. **Pseudo-sum-distinct sets.** A set S is *pseudo-sum-distinct* if nonempty, disjoint subsets of S having the same cardinality have distinct sums. Unlike sum-distinct sets, pseudo-sum-distinct sets are translation invariant, so when we define a quantity $p(m)$ for these sets analogous to our definition of $w(m)$ for sum-distinct sets, we fix the minimal element of the sets involved at 0:

$$p(m) = \min\{\max(S) : S \subset \mathbf{Z},\ |S| = m,\ \min(S) = 0, \text{ and } S \text{ is pseudo-sum-distinct}\}.$$

The values of $p(m)$ give us information on minimal gaps between elements in sum-distinct sets.

Lemma 2. *Suppose $\{e_1, \ldots, e_m\}$ is sum-distinct and $e_1 < \cdots < e_m$. Then for each i and j with $1 \leq i < j \leq m$, we have $e_j - e_i \geq p(j - i + 1)$.*

Proof. The set $\{e_i, \ldots, e_j\}$ is sum-distinct, so the set $\{0, e_{i+1} - e_i, \ldots, e_j - e_i\}$ is pseudo-sum-distinct, hence $e_j - e_i \geq p(j - i + 1)$. □

Some lower bounds for $p(m)$ are known. A simple combinatorial argument that considers subsets of size $\lfloor m/2 \rfloor$ yields

$$p(m) \geq \frac{\binom{m}{\lfloor m/2 \rfloor} - 1}{\lfloor m/2 \rfloor};$$

Elkies [7] uses an analytic method to obtain a bound that is asymptotically $\sqrt{2m}/\pi$ times better,

$$p(m) \geq \frac{2^{m+3}}{(2m+1)\pi^2} - 1.$$

We derive an upper bound for $p(m)$ in terms of the Conway-Guy sequence (6). We first require a few facts from Guy [8] about this sequence.

Lemma 3. *Let $\{u_k\}$ denote the Conway-Guy sequence.*

(*i*) *$\{u_k\}$ is strictly increasing.*
(*ii*) *For $n \geq 1$, $u_{n+1} > u_n + u_{n-1}$.*
(*iii*) *For $n \geq 1$, $\sum_{k=0}^{n-1} u_k < 2u_n$.*

Proof. For (i), since $u_1 > u_0$ and $u_{n+1} - u_n = u_n - u_{n-\langle\sqrt{2n}\rangle}$, the result follows by induction. For (ii), we first verify the inequality for $n = 1$ and 2. Suppose that $n \geq 3$ and $u_n > u_{n-1} + u_{n-2}$. Then $u_{n-2} \geq u_{n-\langle\sqrt{2n}\rangle}$ by (i), so $u_{n+1} = 2u_n - u_{n-\langle\sqrt{2n}\rangle} \geq 2u_n - u_{n-2} > u_n + u_{n-1}$. The inequality (iii) clearly holds for $n = 1$ and 2. Suppose that $n \geq 2$ and $\sum_{k=0}^{n-1} u_k < 2u_n$. Then, using (i) and (ii), we have $\sum_{k=0}^{n} u_k < 2u_n + u_n = u_{n+1} + u_{n-\langle\sqrt{2n}\rangle} + u_n < u_{n+1} + u_{n-1} + u_n < 2u_{n+1}$. □

We remark that Guy in fact proves a stronger version of this last inequality, showing that $\sum_{k=0}^{n-1} u_k < u_n + u_{n-3}$ for $n \geq 5$.

We say that a sequence is pseudo-sum-distinct if any finite collection of its terms has this property. The result of Bohman [1] shows that the Conway-Guy sequence is pseudo-sum-distinct, and we use this fact to derive an upper bound on $p(m)$.

Lemma 4. *Let $\{u_k\}$ denote the Conway-Guy sequence. Then $p(m) \leq 3u_{m-3} + 1$.*

Proof. Let $P = \{3u_0, \ldots, 3u_n\} \cup \{3u_n - 1, 3u_n + 1\}$, and suppose A and B are nonempty, disjoint subsets of P with the same cardinality and sum. Certainly $\{3u_n - 1, 3u_n + 1\} \subseteq A \cup B$ by Bohman's result, and further by considering the sums modulo 3 both $3u_n - 1$ and $3u_n + 1$ must be in the same set, say A. If $3u_n \notin B$, then using part (iii) of Lemma 3, we have $\sum B \leq 3\sum_{k=0}^{n-1} u_k < 6u_n \leq \sum A$, a contradiction. If $3u_n \in B$, then the hypothesis that $A = \{3a_1, \ldots, 3a_r, 3u_n - 1, 3u_n + 1\}$ and $B = \{3b_1, \ldots, 3b_{r+1}, 3u_n\}$ have equal sums implies that $\{a_1, \ldots, a_r, u_n\}$ and $\{b_1, \ldots, b_{r+1}\}$ have equal sums, again a contradiction. □

Maltby [11] employs a similar construction in the study of sum-distinct sets.

We next develop an algorithm to compute the exact value of $p(m)$.

Algorithm 2. *Optimal pseudo-sum-distinct sets.*

Input. Positive integers m and B, and the values of $p(k)$ for $k < m$.
Output. All pseudo-sum-distinct sets U with $|U| = m$, $\min(U) = 0$, and $\max(U) = B$.
Invariants. U is pseudo-sum-distinct, and for each k with $1 \leq k \leq \lfloor m/2 \rfloor$, the set $T_k = \{\sum A : A \subseteq U \text{ and } |A| = k\}$.
Description.

Step 0. Set $U = \{0, B\}$, $T_1 = \{0, B\}$, $T_2 = \{B\}$, and $T_k = \{\}$ for $3 \leq k \leq \lfloor m/2 \rfloor$.

Step 1. If $|U| = m$ then print U. Otherwise, let $e = \max(U \setminus \{B\})$. For each t satisfying $\max(e + 1, p(|U|)) \leq t \leq B - p(m - |U| + 1)$, perform Step 2.

Step 2. Test if $U \cup \{t\}$ is pseudo-sum-distinct: For each k with $1 \leq k < \lfloor m/2 \rfloor$, and for each $u \in T_k$, test if $u + t \in T_{k+1}$. If all tests are negative, continue with Step 3.

Step 3. Add t to U, and for each k from $\lfloor m/2 \rfloor - 1$ down to 1, then each $u \in T_k$, insert $u + t$ into T_{k+1}. Perform Step 1. Remove t from U, and for each k from 1 to $\lfloor m/2 \rfloor - 1$, then each $u \in T_k$, remove $u + t$ from T_{k+1}.

Proof. Suppose $|U| = k$ in Step 1. The lower bound on t in this step arises from choosing $i = 1$ then $i = k - 1$ together with $j = k$ in Lemma 2; the upper bound from $i = k$ and $j = m$. □

Implementation. The set U is implemented efficiently using a sorted array. Since we require fast insertions, searches, and removals on the sets T_k, we implement these sets using randomized binary search trees [14] so that all operations run in expected $O(\log n)$ time.

Results. We use Algorithm 2 to determine the value of $p(m)$ for $m \le 10$ and find that in fact $p(m) = 3u_{m-3}+1$ in this range. In most cases, there are exactly two extremal sets: The set P constructed in Lemma 4, and its complement $-P+\max(P)$. For example, for $m = 7$ we apply the construction of the lemma to $\{0, 1, 2, 4, 7\}$ to obtain $\{0, 3, 6, 12, 20, 21, 22\}$; its complement is $\{0, 1, 2, 10, 16, 19, 22\}$. Additional pairs appear for $m = 6$ by applying the same construction to $\{0, 2, 3, 4\}$ and for $m = 8$ by starting with $\{0, 1, 2, 7, 10, 13\}$.

We remark that we do not expect equality in Lemma 4 for larger m. Certainly any pseudo-sum-distinct sequence satisfying (iii) of Lemma 3 may be used in the construction of Lemma 4, and particular sequences may be used to improve this bound. For example, the sequences $\mathbf{w}^1$ and $\mathbf{w}^2$ recorded by Lunnon [10, p. 312] have the required properties, and using $\mathbf{w}^1$ improves our bound on $p(m)$ for $m \ge 15$; using $\mathbf{w}^2$ improves it further for $m \ge 16$.

3.2. **Sum-distinct sets.** We now describe an algorithm for computing $e(m)$.

Algorithm 3. *Sum-distinct sets of odd integers with minimal sum.*

Input. Positive integers m and B, and the values of $p(k)$ for $k \le m$.
Output. All sum-distinct sets U of positive odd integers with $\sum U \le B$.
Invariants. $U = \{e_1, \ldots, e_k\}$ is sum-distinct, $e_1 < \cdots < e_k$, $f(x) = \prod_{i=1}^{k}(x^{e_i} + 1)$ is a Newman polynomial.
Description.

Step 0. For each e_1 satisfying $1 \le e_1 \le (B - 2\sum_{i=1}^{m} p(i))/m$, set $U = \{e_1\}$ and $f(x) = x^{e_1} + 1$, and perform Step 1.

Step 1. If $|U| = m$ then print U. Otherwise, let $k = |U|$, and for each odd t satisfying

$$\max\{e_k + 2, e_1 + 2p(k+1)\} \le t \le \frac{B - \sum U - 2\sum_{i=1}^{m-k} p(i)}{m-k},$$

perform Step 2.

Step 2. If $(x^t + 1)f(x)$ is a Newman polynomial, set $e_{k+1} = t$, admit e_{k+1} to U, and multiply $f(x)$ by $x^{e_{k+1}} + 1$. Perform Step 1, then remove e_{k+1} from U and divide $f(x)$ by $x^t + 1$.

Proof. For the lower bound in Step 1, if $S = \{e_i, \ldots, e_k, t\}$ is a sum-distinct set of odd integers, then $\frac{1}{2}(S - e_i)$ is a pseudo-sum-distinct set of integers with minimal

element zero, and consequently $t - e_i \geq 2p(k - i + 2)$. For efficiency we select only $i = 1$ and $i = k$ when computing the lower bound.

For the upper bound in Steps 0 and 1, suppose that $|U| = k$, so that the elements $e_{k+1}, \ldots, e_m$ are not yet chosen. By the previous argument, we have $e_i - e_{k+1} \geq 2p(i - k)$ for $i > k$, and so

$$(m-k)e_{k+1} + 2\sum_{i=1}^{m-k} p(i) \leq \sum_{i=k+1}^{m} e_i \leq B - \sum U.$$

□

Implementation. We represent $f(x)$ as a bit vector so that we can operate on blocks of coefficients in a single step. This is particularly useful on a computer with a 64 bit word size. To test if $(x^t+1)f(x)$ is a Newman polynomial, we use bitwise "and" to test each block of coefficients against the bits t positions higher. The product is a Newman polynomial if each result is zero. Multiplication is similar, only we use bitwise "or" to set the new terms of $f(x)$, and we must take care to work from the high bits to the low bits to prevent newly set bits from propagating into successive blocks. For division, we use bitwise "xor" to clear bits, working from low bits to high. We may operate on blocks provided t is greater than the block size, and C++ allows one to address blocks of different sizes in a bit vector (often 8, 16, 32, and sometimes 64 bits), so only in the rare case when $t < 8$ must we operate on single bits.

Results. We use Algorithm 3 to determine $e(m)$ for $m \leq 11$. The results of Algorithm 2 provide the required values of $p(m)$ for $m \leq 10$; for $m = 11$, we use the lower bound $p(11) \geq p(10) + 1 = 134$. The optimal sets for each m are recorded in Table 3.

TABLE 3. Sum-distinct sets of odd integers with minimal sum.

m	$e(m)$	Optimal sets
1	1	$\{1\}$
2	4	$\{1,3\}$
3	9	$\{1,3,5\}$
4	20	$\{1,3,5,11\}$
5	41	$\{1,3,5,11,21\}$ $\{1,5,7,9,19\}$
6	78	$\{1,5,7,9,19,37\}$
7	153	$\{1,5,7,9,19,37,75\}$
8	302	$\{1,5,7,9,19,37,75,149\}$
9	601	$\{1,5,7,9,19,37,75,149,299\}$ $\{3,7,11,19,25,31,69,149,287\}$
10	1180	$\{3,5,11,17,27,37,71,145,287,577\}$
11	2313	$\{1,9,11,13,35,53,71,141,283,565,1131\}$

We find that the sequence $\{a_k\}$ discussed in Section 2 is optimal for $5 \leq m \leq 9$, but not for $m = 10$ and 11. We determine improved upper bounds on $d(m)$ by

analyzing these and other sequences detected by Algorithm 3. As in (11), each sequence $\{e_k\}$ may be extended using a recurrence relation of the form

$$e_k = r_k + \sum_{i=1}^{k-1} e_i$$

where the value of r_k depends only on the parity of k, so we obtain again the Jacobsthal recurrence $e_k = e_{k-1} + 2e_{k-2}$ for sufficiently large k. Thus, for each sequence $\{e_k\}$ we may compute c so that $\deg([e_1, \ldots, e_m]) \sim c \cdot 2^m$.

We compute this constant c for several sequences found by Algorithm 3, including all of the sequences in Table 3. These values are listed in Table 4.

TABLE 4. Asymptotic growth associated with sum-distinct sequences.

c	$e_1, e_2, e_3, \ldots$
4/3	$1, 3, \ldots$
7/6	$1, 5, 7, 9, 19, \ldots$
109/96	$3, 7, 11, 19, 25, 31, 69, 149, \ldots$
9/8	$3, 5, 11, 17, 27, 37, 71, \ldots$ $9, 11, 13, 31, 43, 49, 75, 141, \ldots$
107/96	$3, 5, 7, 11, 17, 73, 107, 141, 287, \ldots$
53/48	$1, 9, 11, 13, 35, 53, 71, 141, \ldots$ $7, 11, 15, 37, 39, 53, 67, 145, \ldots$
211/192	$1, 23, 39, 43, 45, 47, 55, 133, 289, \ldots$
421/384	$11, 27, 31, 35, 43, 75, 97, 119, 269, 573, \ldots$
35/32	$1, 9, 13, 17, 35, 67, 69, 147, \ldots$ $3, 5, 11, 17, 27, 71, 105, 139, 281, \ldots$
103/96	$1, 13, 15, 17, 43, 69, 103, 137, 275, \ldots$ $11, 13, 15, 37, 59, 73, 103, 133, 279, \ldots$

Among all sequences detected by our method, the first sequence with $c = 103/96$ in Table 4 produces the best polynomials for $m \geq 12$. Using this sequence we find that

$$d(m) \leq e(m) \leq \frac{103}{96} \cdot 2^m + \frac{247}{2} - \frac{(-1)^m}{6} \tag{12}$$

for $m \geq 12$.

We conjecture that for any $\epsilon > 0$ we have $d(m) < (1 + \epsilon) \cdot 2^m$ for sufficiently large m.

4. SUM-DISTINCT SETS WITH MINIMAL LARGEST ELEMENT

While it is straightforward to amend Algorithm 3 to compute sum-distinct sets with minimal largest element, we find that an algorithm incorporating some of the techniques of Lunnon [10] is significantly faster for computing $w(m)$. Like the previous algorithm, the following method uses $p(m)$ to determine bounds on the elements e_k selected. Unlike Algorithm 3, it selects the elements in descending order, and replaces the Newman polynomial $f(x)$ with another generating function that allows us to select successive elements more quickly.

Algorithm 4. *Sum-distinct sets with minimal largest element.*

Input. Positive integers m and B, and the values of $p(k)$ for $k \le m$.
Output. All sum-distinct sets U of positive integers with $\max(U) \le B$.
Invariants. $U = \{e_1, \ldots, e_k\}$ is sum-distinct and $e_1 > \cdots > e_k$.
Description.

Step 0. For each t from B down to $p(m)+1$, set $e_1 = t$, $U = \{e_1\}$, and $f_1(x) = 1 + x^{e_1} + x^{-e_1}$.

Step 1. Let $k = |U|$. If $k = m$ then print $\{e_1, \ldots, e_m\}$. Otherwise, for each t from $\min\{e_k - 1, e_1 - p(k+1)\}$ down to $1 + p(m-k)$ for which x^t does not appear in $f_k(x)$, set $e_{k+1} = t$, admit e_{k+1} to U, and set $f_{k+1}(x) = (1 + x^t + x^{-t}) f_k(x)$. Perform Step 1, then remove e_{k+1} from U.

Proof. If $\{e_1, \ldots, e_m\}$ is sum-distinct, then $e_1 - e_{k+1} \ge p(k+1)$ and $e_{k+1} - e_m \ge p(m-k)$. Combining these facts with $e_m \ge 1$ yields the bounds on t in Steps 0 and 1. Also, the term x^t appears in $f_k(x)$ if and only if there exist disjoint subsets A and B of $U = \{e_1, \ldots, e_k\}$ such that $\sum A - \sum B = t$, so x^t does not appear in $f_k(x)$ if and only if $U \cup \{t\}$ is sum-distinct. □

Implementation. We describe some features of our implementation, some of which are also employed by Lunnon. First, it is convenient to represent each $f(x)$ as a product of terms of the form $1 + x^{e_i} + x^{2e_i}$ so that bits are shifted in only one direction when multiplying. Second, we use 0 to represent a coefficient that is present in each $f_k(x)$ and 1 to indicate a coefficient that is absent. This simplifies the code used to search for a valid t in Step 1. We find it is efficient to check blocks of 16 coefficients at once in this search, and whenever a block having at least one set bit is detected, we first split the block into two 8-bit halves to narrow the search before checking bit by bit. Third, we often do not have to compute the entire polynomial $f_k(x)$ in Step 1. Rather, we need only compute those bits which are significant for the selection of the e_i for $i \ge k$. Since $e_i \le e_1 - p(i)$ for each i, this greatly reduces the effort needed to compute $f_k(x)$ for larger k, where the algorithm spends most of its time.

Our implementation requires about 300 lines of source code in C++.

Results. We use Algorithm 4 to compute $w(m)$ for $m \le 9$, extending Lunnon's computations by one step. The computations for $m \le 8$ require about two minutes using a Silicon Graphics computer with a MIPS R10000 processor; the case $m = 9$ requires almost 20 hours. We find that $w(m) = u_m$ for $m \le 9$, and that the set U_9 constructed from the Conway-Guy sequence is the unique optimal solution for $m = 9$. The optimal sets for $m \le 9$ are listed in Table 5, where the set U_m appears first for each m.

References

[1] T. Bohman, *A sum packing problem of Erdős and the Conway-Guy sequence*, Proc. Amer. Math. Soc. **12** (1996) 3627–3636.

[2] ———, *A construction for sets of integers with distinct subset sums*, Electron. J. Combin. **5** (1998), Research Paper 3, 14 pp. (electronic).

[3] P. Borwein and M. J. Mossinghoff, *Polynomials with height 1 and prescribed vanishing at 1*, Experiment. Math. **9** (2000), 425–433.

[4] D. W. Boyd, *On a problem of Byrnes concerning polynomials with restricted coefficients*, Math. Comp. **66** (1997), 1697–1703.

TABLE 5. Sum-distinct sets with minimal largest element.

m	$w(m)$	Optimal Sets
1	1	$\{1\}$
2	2	$\{1,2\}$
3	4	$\{2,3,4\}$
		$\{1,2,4\}$
4	7	$\{3,5,6,7\}$
5	13	$\{6,9,11,12,13\}$
		$\{3,6,11,12,13\}$
6	24	$\{11,17,20,22,23,24\}$
7	44	$\{20,31,37,40,42,43,44\}$
8	84	$\{40,60,71,77,80,82,83,84\}$
		$\{39,59,70,77,78,79,81,84\}$
		$\{20,40,71,77,80,82,83,84\}$
9	161	$\{77,117,137,148,154,157,159,160,161\}$

[5] ———, *On a problem of Byrnes concerning polynomials with restricted coefficients, II*, Math. Comp., posted on May 11, 2001, PII S 0025-5718(01)01360-6 (to appear in print).
[6] J. H. Conway and R. K. Guy, *Solution of a problem of P. Erdős*, Colloq. Math. **20** (1969), 307.
[7] N. D. Elkies, *An improved lower bound on the greatest element of a sum-distinct set of fixed order*, J. Combin. Theory Ser. A **41** (1986), 89–94.
[8] R. K. Guy, *Sets of integers whose subsets have distinct sums*, pp. 141–154 in *Theory and Practice of Combinatorics*, edited by A. Rosa, G. Sabidussi, and J. Turgeon, Ann. of Discrete Math. **12**, North-Holland, Amsterdam, 1982.
[9] R. K. Guy, *Unsolved Problems in Number Theory*, 2nd ed., Springer-Verlag, New York, 1994.
[10] A. F. Horadam, *Jacobsthal representation numbers*, Fibonacci Quart. **34** (1996), 40–54.
[11] W. F. Lunnon, *Integer sets with distinct subset-sums*, Math. Comp. **50** (1988), 297–320.
[12] R. Maltby, *Bigger and better subset-sum-distinct sets*, Mathematika **44** (1997), 56–60.
[13] A. Nijenhuis and H. S. Wilf, *Combinatorial Algorithms*, Academic Press, Orlando, 1978.
[14] A. M. Odlyzko and B. Poonen, *Zeros of polynomials with 0,1 coefficients*, Enseign. Math. (2) **39** (1993), 317–348.
[15] R. Sedgewick, *Algorithms in C++, Parts 1–4*, 3rd ed., Addison-Wesley, Reading, MA, 1998.

DEPARTMENT OF MATHEMATICS AND STATISTICS, SIMON FRASER UNIVERSITY, BURNABY, B.C., CANADA V5A 1S6
E-mail address: pborwein@cecm.sfu.ca

DEPARTMENT OF MATHEMATICS, UCLA, LOS ANGELES, CALIFORNIA 90095
E-mail address: mjm@math.ucla.edu