Maximizing the number of integer pairs summing to powers of 2 via graph labeling and solving restricted systems of linear (in)equations

Max Alekseyev

Department of Mathematics

THE GEORGE WASHINGTON UNIVERSITY

WASHINGTON, DC

2025

Outline

Problem of the Week # 1321

In April 2021, Dan Ullman and Stan Wagon published *Problem # 1321* in the *Macalester College Problem of the Week* series, where they introduced:

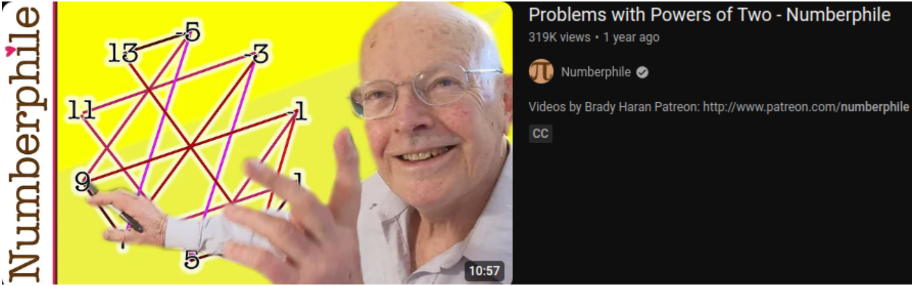
- a function $f(A)$ of a finite set A of integers equal the number of 2-element subsets of A that sum to a power of 2.
  For example, $f(\{-1,3,5\}) = 3$ since $-1+3=2^1$, $-1+5=2^2$, and $3+5=2^3$.
- a function $g(n)$ as

$$g(n) := \max_{A:|A|=n} f(A).$$

The Ullman and Wagon problem asked for a proof that $g(10) \geq 14$, which was quickly improved to $g(10) \geq 15$ by the readers.

It further inspired a harder question of determining the exact value of $g(n)$ for a given n.

In September 2022, the problem of determining values of $g(n)$ received much attention after Neil Sloane presented it on the popular *Numberphile* Youtube channel:

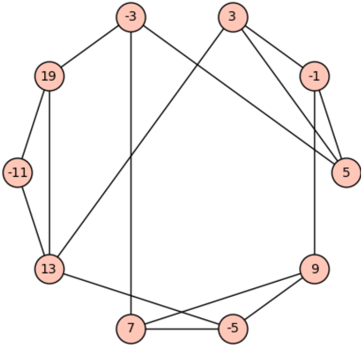


Computing $g(n)$ via Labeled Graphs

It was quickly noticed by many people that computing $g(n)$ has a natural interpretation as finding a maximum graph of order n, where

- the vertices are labeled with pairwise distinct integers; and
- the sum of the endpoint labels for each edge is a power of 2.

Example of a maximum graph of order $n = 10$ having size $g(10) = 15$:



In March 2022, M. S. Smith proved that such a graph cannot contain a cycle C_4, limiting the candidate graphs to well-studied squarefree graphs.

It led to establishing the values of $g(n)$ for all $n \leq 9$, and creation of the sequence `A352178` in the Online Encyclopedia of Integer Sequences:

A352178	Let S = {t_1, t_2, ..., t_n} be a set of n distinct integers and consider the sums t_i + t_j (i<j); a(n) is the maximum number of such sums that are powers of 2, over all choices for S.

0, 1, 3, 4, 6, 7, 9, 11, 13, 15, 17

Value $g(10) = 15$ was independently obtained by R. Pratt, M. Bolan, and F. Melaih, who also established that $g(11) = 17$. Computing $g(n)$ for $12 \leq n \leq 21$ is the subject of the present work.

Lower Bounds

At the same time, two computationally "easy" methods were proposed for obtaining lower bounds for $g(n)$:

A347301	Let S be a set of n distinct integers in the range -n-3 to n+3, and consider the sums s+t of pairs of distinct elements of S; a(n) is the maximum number of such sums that are powers of 2, over all choices for S.

0, 1, 3, 4, 5, 7, 9, 11, 13, 15, 17, 19, 21, 24, 26, 29, 31, 34, 36, 39, 41, 44, 46, 49, 51, 54, 56, 59, 61, 65, 67, 71, 73, 77, 79, 83, 85, 89, 91, 95, 97, 101, 103, 107, 109, 113, 115, 119, 121, 125, 127, 131, 133, 137, 139, 143, 146, 149, 152, 155, 159, 162, 166, 169, 173, 176

A357574	a(n) is the maximum number of pairs that sum to a power of 2 in a set of n consecutive odd numbers.

0, 1, 2, 4, 5, 7, 9, 11, 13, 15, 17, 19, 21, 24, 26, 29, 31, 34, 36, 39, 41, 44, 46, 49, 51, 54, 56, 59, 62, 65, 68, 71, 74, 77, 80, 83, 86, 89, 92, 95, 98, 101, 104, 107, 110, 113, 116, 119, 122, 125, 128, 131, 134, 137, 140, 143, 146, 150, 153, 157, 160, 164, 167

The two bounds coincide in the range of interest $n \in [12, 21]$, and let $\ell(n)$ denote the corresponding lower bound.

In fact, we have proved that $g(n) = \ell(n)$ for all $n \in [12, 21]$.

An unlabeled graph is admissible if its vertices can be labeled with pairwise distinct integers such that the sum of the endpoint labels for each edge is a power of 2.

We developed an algorithm for testing graph admissibility, which then was used to

- find all minimal forbidden subgraphs (like C_4) of orders ≤ 11;
- establish values $g(n)$ for $12 \leq n \leq 21$; and
- find all maximum admissible graphs for $n \leq 20$.

Outline

A Bit of Linear Algebra

Given a matrix M:

- Right kernel K_r:
  $M \cdot K_r = \mathbf{0}$; and for any vector v with $M \cdot v^T = 0$ we have $v^T = K_r \cdot u^T$ for some vector u.
- Left kernel K_l:
  $K_l \cdot M = \mathbf{0}$; and for any vector v with $v \cdot M = 0$ we have $v = u \cdot K_l$ for some vector u.
- Rouche–Frobenius–Capelli theorem:
  given a vector w, solution v to $M \cdot v^T = w^T$ exists if and only if $K_l \cdot w^T = \mathbf{0}^T$.
- All solutions to $M \cdot v^T = w^T$ are given by

$$v = v_0 + K_r \cdot u^T$$

for any particular solution v_0 (i.e. $M \cdot v_0^T = w^T$) and any vector u (of suitable size).

Matrix Equation

A given graph G on n vertices with m edges is admissible if and only if the following matrix equation is soluble:

$$M \cdot L^T = X^T. \tag{1}$$

such that

- M is the $m \times n$ incidence matrix of G with rows and columns indexed by the edges and vertices of G, and so M is a $\{0, 1\}$-matrix with each row containing exactly two 1's;
- $L = (l_1, l_2, \dots, l_n)$ is a vector of pairwise distinct integer vertex labels;
- $X = (x_1, x_2, \dots, x_m)$ is a vector formed by powers of 2 (possibly equal) representing the sums of edges' endpoint labels.

Both L and X are unknown and have to be determined.

Matrix Equation

A given graph G on n vertices with m edges is admissible if and only if the following matrix equation is soluble:

$$M \cdot L^T = X^T. \quad (1)$$

We start with solving (1) for L in terms of X, that is, we compute a particular solution $L_0 = (p_1, \dots, p_n)$, where $p_i = p_i(x_1, \dots, x_m)$ are linear polynomials with rational coefficients.

By Rouche–Frobenius–Capelli theorem, such a solution exists if and only if $K_I \cdot X = 0$. Let E be the set of elements of $K_I \cdot X$, which are linear homogeneous polynomials with integer coefficients representing linear equations in $x_1, \dots, x_m$.

For practical efficiency, K_I should be sparse, which (to some extent) can be achieved by LLL reduction.

Matrix Equation

A given graph G on n vertices with m edges is admissible if and only if the following matrix equation is soluble:

$$M \cdot L^T = X^T. \quad (1)$$

All solutions L for a fixed X can be obtained as

$$L^T = L_0^T + K_r u^T$$

for any vector u (of suitable size, of course).

However, we additionally require that L has pairwise distict elements.

The following theorem shows that with an appropriate choice of u, pairwise distinct rows in K_r can make the corresponding elements of L also pairwise distinct.

Pair of Distinct Rows in K_r

Theorem 1

Let v be an integer vector of size $k \geq 0$, and A be a $k \times s$ integer matrix with pairwise distinct rows. Then there exists an integer linear combination of the columns of A such that adding it to v results in a vector with pairwise distinct elements.

Proof.

If $s = 0$, then with necessity we have $k = 1$, and thus v already has pairwise distinct elements.

Let us prove the statement for $s = 1$. In this case, A represents a column-vector with pairwise distinct elements. Let t be the difference between the largest and the smallest elements of v. It is easy to see that vector $v + A^T \cdot (t+1)$ has pairwise distinct elements.

In the case of $s > 1$, let d be the difference between the largest and the smallest elements of A. Then the $k \times 1$ matrix $A' := A \cdot (1, (d+1), (d+1)^2, \ldots, (d+1)^{s-1})^T$ has pairwise distinct elements, thus reducing the problem to the case $s = 1$ considered above. □

Pairs of Equal Rows in K_r

Theorem 1 implies that we need to enforce non-equality only for the pairs of elements of $L = (l_1, \dots, l_n)$ that correspond to the equal rows of K_r.

For each pair of equal rows in K_r with indices $i < j$, we compute

$$q_{ij}(x_1, \dots, x_m) := (p_i(x_1, \dots, x_m) - p_j(x_1, \dots, x_m))c,$$

where c is a positive integer factor making all coefficients of q integer.

- If q_{ij} is zero polynomial, then the condition $l_i \neq l_j$ is unattainable, and thus the graph G is inadmissible.
- If q_{ij} consists of just a single term with a nonzero coefficient, then the condition $l_i \neq l_j$ always holds, and we ignore such q_{ij}.
- If q_{ij} contains two or more terms with nonzero coefficient, we add q_{ij} to the set N.

The resulting set N consists of polynomials in $x_1, \dots, x_m$ that have to take nonzero values on a solution X to (1). We will refer to such polynomials as inequations.

System of Equations and Inequations

For given finite sets E and N of nonzero linear polynomials in $x_1, x_2, \ldots, x_m$, we need to find all tuples of nonnegative integers $(y_1, y_2, \ldots, y_m)$ such that

$$\forall p \in E : \quad p(2^{y_1}, 2^{y_2}, \ldots, 2^{y_m}) = 0,$$
$$\forall p \in N : \quad p(2^{y_1}, 2^{y_2}, \ldots, 2^{y_m}) \neq 0.$$

Technically it is convenient to represent each parametric family of solutions as a map s from the set of variables $Y := \{y_1, y_2, \ldots, y_m\}$ to linear polynomials in these variables. Namely, s sends every variable from Y to

- either itself when it is a free variable; or
- to a linear polynomial in the free variables.

For example, the map $\{y_1 \mapsto y_4 + 1,\ y_2 \mapsto y_2,\ y_3 \mapsto y_2 + y_4 + 3, y_4 \mapsto y_4\}$ corresponds to the solution $(x_1, x_2, x_3, x_4) = (2^{y_4+1}, 2^{y_2}, 2^{y_2+y_4+3}, 2^{y_4})$, where y_2 and y_4 are free variables (taking nonnegative integer values).

GraphSolve Algorithm

function GraphSolve(G)
    Set $E := \emptyset$ and $N := \emptyset$
    Construct the incidence matrix M of G with rows and columns indexed by edges and vertices of G
    Compute a LLL-reduced basis K_l of the left kernel of M. ▷ We have $K_l \cdot M = 0$.
    for each row r in K_l **do**
        Add polynomial $r \cdot X$ to E
    end for
    Solve $ML^T = X^T$ for L in terms of $X = (x_1, \dots, x_m)$, let $(p_1, \dots, p_n)$ be any particular solution.
    Compute K_r whose columns form a basis of the right kernel of M. ▷ We have $M \cdot K_r = 0$.
    for each $\{i, j\} \subset \{1, 2, \dots, n\}$ **do**
        if ith and jth rows of K_r are not equal **then**
            continue to next subset $\{i, j\}$ ▷ Per Theorem 1.
        end if
        Set q equal to a multiple of $p_i - p_j$ with integer coefficients
        if $q = 0$ **then**
            return $\emptyset$ ▷ No solutions with $l_i \neq l_j$.
        end if
        if q contains two or more terms **then**
            Add q to N.
        end if
    end for
    $S := \emptyset$
    for each s in SolveInPowers(E, N) **do** ▷ s is a map from Y to linear polynomials in Y
        Set $x_i := 2^{s[y_i]}$ for each component x_i in X
        Solve $ML^T = (x_1, \dots, x_m)^T$ for L composed of pairwise distinct integers, and add the solution to S.
    end for ▷ Invoke Theorem 1 as needed.
    return S
end function

Outline

System of Equations and Inequations

- **Given:** a system (E, N) with finite sets E and N of nonzero linear polynomials in $x_1, x_2, \dots, x_m$.
- **Find:** all tuples of nonnegative integers $(y_1, y_2, \dots, y_m)$ such that

$$\forall p \in E: \quad p(2^{y_1}, 2^{y_2}, \dots, 2^{y_m}) = 0,$$
$$\forall p \in N: \quad p(2^{y_1}, 2^{y_2}, \dots, 2^{y_m}) \neq 0.$$

Foundational Theorem

As simple as it sounds, the following theorem provides a foundation for our algorithm.

Theorem 2

In any nonempty multiset of nonzero integers summing to 0*, there exist two elements with equal* 2*-adic valuations.*

Proof.

Let

- S be a nonempty multiset of nonzero integers summing to 0, and
- k be an element of S with the smallest 2-adic valuation, say $q := \nu_2(k) < \infty$.

If every other element of S has valuation greater than q, then the sum of all elements (which is 0) has valuation q, which is impossible. Hence, there exist at least two elements in S having 2-adic valuation equal q. □

Recursive Solution via Indeterminates Reduction

Applying Theorem 2 to an equation $c_1x_1 + \cdots + c_mx_m \in E$, we conclude that

- if only one of the coefficients $c_1, c_2, \ldots, c_m$ is nonzero, then the system (E, N) is inadmissible.
- otherwise, if there are two or more nonzero coefficients among $c_1, c_2, \ldots, c_m$, then there exists a pair of indices $i < j$ such that $c_i \neq 0$, $c_j \neq 0$, and $\nu_2(c_ix_i) = \nu_2(c_jx_j)$.

In the latter case, we make a substitution $x_i \leftarrow 2^{\nu_2(c_j)-\nu_2(c_i)}x_j$ or $x_j \leftarrow 2^{\nu_2(c_i)-\nu_2(c_j)}x_i$ (we pick one with integer coefficients) in both E and N, thus reducing the number of indeterminates. Then, if it does not make any elements of N evaluate to zero, we proceed with solving the reduced system recursively.

After the pair (i, j) is explored, we add a new inequation $2^{\nu_2(c_i)}x_i - 2^{\nu_2(c_j)}x_j$ to N (preventing getting the same solutions again in future), and proceed to a next pair of indices.

SolveInPowers Algorithm

function SolveInPowers($E,\ N$)
    if $E = \emptyset$ **then**
        return {the identity map: $y_i \mapsto y_i$} ▷ Every variable in Y is free.
    end if
    Pick $c_1 x_1 + \cdots + c_m x_m \in E$ with the smallest number of nonzero coefficients.
    Let $I := \{i \mid 1 \leq i \leq m,\ c_i \neq 0\}$ be the set of indices of nonzero coefficients.
    if $|I| = 1$ **then**
        return $\emptyset$ ▷ Such equation has no solutions.
    end if
    Set $S := \emptyset$ ▷ We accumulate solutions in S.
    for each $\{i, j\} \subseteq I$ **do** ▷ We iterate over all 2-element subsets of I.
        Possibly exchanging the values of i and j, ensure that $d := \nu_2(c_i) - \nu_2(c_j) \geq 0$.
        Compute N' from N by substituting $x_j \leftarrow 2^d x_i$ and excluding nonzero constants.
        if $0 \in N'$ **then**
            continue to the next pair $\{i, j\}$.
        end if
        Add $x_j - 2^d x_i$ to N. ▷ For future we disallow the equality $c_j x_j = c_i x_i$.
        Compute E' from E by substituting $x_j \leftarrow 2^d x_i$ and excluding zero polynomials.
        for each s in SolveInPowers($E',\ N'$) **do**
            Redefine $s[y_j] := s[y_i] + d$.
            Add s to set S.
        end for
    end for
    return S
end function

Outline

Minimal forbidden graphs

We used our algorithms to find minimal forbidden subgraphs (MFS) of small order, i.e., inadmissible graphs in which every proper subgraphs is admissible.

It is easy to see that each MFS must be connected. It is further almost trivial task to verify that C_4 is the smallest MFS, and it is the only one on 4 vertices. Therefore, for $n > 4$ we can restrict our attention to connected squarefree graphs as candidates.

To generate them we employed `SageMath` software (`sagemath.org`) with the function `nauty_geng()` based on the `nauty` tool supporting generation of both connected (option `-c`) and squarefree (option `-f`) graphs.

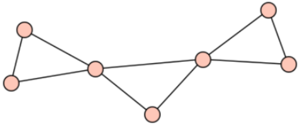
FindMFS Algorithm

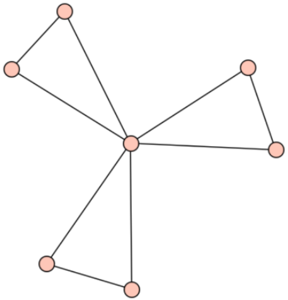
An algorithm for iterative computing minimal forbidden subgraphs, other than C_4, of order up to u:

```
function FindMFS(u)
    S := ∅
    for n = 5, ..., u do
        for each connected squarefree graph G of order n do
            if G contains any graph H from S as a subgraph then
                continue to next G
            end if
            if GraphSolve(G) is empty then
                Add G to the set S.
            end if
        end for
    end for
    return S
end function
```

Computed MFSs

We confirmed that the smallest MFS is the cycle C_4 as it was originally proved by Smith, and the next two MFSs have order 7:





In fact, the former graph was previously proved inadmissible by Bolan while showing that $g(10) = 15$.

There are no MFSs of order 8 or 9, but there are 15 MFSs of order 10, and there are 77 MFSs of order 11.

We use MFSs of order ≤ 10 for quick filtering of some inadmissible graphs.

Computing values of $g(n)$

We use the proposed algorithms for computing values of $g(n)$ for n in $[12, 21]$, relying on the known values $g(n)$ for $n \le 11$ and the lower bound $\ell(n)$ for $n \ge 12$:

n	1	2	3	4	5	6	7	8	9	10	11
$g(n)$	0	1	3	4	6	7	9	11	13	15	17

n	12	13	14	15	16	17	18	19	20	21
$\ell(n)$	19	21	24	26	29	31	34	36	39	41

Theorem 3

For each integer $n \in [12, 21]$, we have $g(n) = \ell(n)$.

Our goal is to show that for each $n \in [12, 21]$ we have $g(n) \le \ell(n)$, which is equivalent to showing that every graph of order n and size $\ell(n) + 1$ is inadmissible.

Useful Theorem

Theorem 4

For any integer $n > 2$:

(I) *if an admissible graph G of order n and size e exists, then its minimum degree $\geq e - g(n-1)$;*

(II)

$$g(n) \leq \left\lfloor \frac{n \cdot g(n-1)}{n-2} \right\rfloor;$$

(III) *if*

$$\frac{g(n-1)}{n-1} = \max_{k \in \{1,2,\dots,n-1\}} \frac{g(k)}{k},$$

then any admissible graph of order n and size $e > \frac{n}{n-1} g(n-1)$ is connected.

Proof of (I). Let G be an admissible graph of order n with e edges. If G has a vertex of degree $< e - g(n-1)$, then removing it from G results in an admissible graph of order $n-1$ and size $> g(n-1)$, a contradiction. □

Useful Theorem

Theorem 4

For any integer $n > 2$:

(I) *if an admissible graph G of order n and size e exists, then its minimum degree $\geq e - g(n-1)$;*

(II)

$$g(n) \leq \left\lfloor \frac{n \cdot g(n-1)}{n-2} \right\rfloor;$$

(III) *if*

$$\frac{g(n-1)}{n-1} = \max_{k \in \{1,2,\ldots,n-1\}} \frac{g(k)}{k},$$

then any admissible graph of order n and size $e > \frac{n}{n-1} g(n-1)$ is connected.

Proof of (II). Since minimum degree $\geq e - g(n-1)$, we have that size of G is at least $\frac{n(e-g(n-1))}{2}$, that is, $e \geq \frac{n(e-g(n-1))}{2}$, implying that $e \leq \left\lfloor \frac{n \cdot g(n-1)}{n-2} \right\rfloor$. For an admissible graph of order n and size $e = g(n)$, it implies $g(n) \leq \left\lfloor \frac{n \cdot g(n-1)}{n-2} \right\rfloor$. □

Useful Theorem

Theorem 4

For any integer $n > 2$:

(III) *if*

$$\frac{g(n-1)}{n-1} = \max_{k\in\{1,2,\dots,n-1\}} \frac{g(k)}{k},$$

then any admissible graph of order n and size $e > \frac{n}{n-1}g(n-1)$ is connected.

Proof of (III). Suppose that we have an admissible graph G of order n and size $e > \frac{n}{n-1}g(n-1)$. Let $s_1 \le \dots \le s_t$ be the orders of its connected components. If $t \ge 2$, then $s_t \le n-1$. Clearly, we have $s_1 + \dots + s_t = n$ and $e \le g(s_1) + \dots + g(s_t)$ and thus

$$\frac{g(s_1)+\dots+g(s_t)}{s_1+\dots+s_t} \ge \frac{e}{n} > \frac{g(n-1)}{n-1}.$$

On the other hand, since $\frac{g(s_1)+\dots+g(s_t)}{s_1+\dots+s_t}$ is the mediant fraction of $\frac{g(s_1)}{s_1}, \dots, \frac{g(s_t)}{s_t}$, we have a contradiction:

$$\frac{g(s_1)+\dots+g(s_t)}{s_1+\dots+s_t} \le \max_{1\le i\le t}\frac{g(s_i)}{s_i} \le \max_{k\in\{1,2,\dots,n-1\}}\frac{g(k)}{k} = \frac{g(n-1)}{n-1}. \quad \square$$

Useful Theorem

Theorem 4

For any integer $n > 2$:

(I) *if an admissible graph G of order n and size e exists, then its minimum degree $\geq e - g(n-1)$;*

(II)
$$g(n) \leq \left\lfloor \frac{n \cdot g(n-1)}{n-2} \right\rfloor;$$

(III) *if*
$$\frac{g(n-1)}{n-1} = \max_{k \in \{1,2,\dots,n-1\}} \frac{g(k)}{k},$$
then any admissible graph of order n and size $e > \frac{n}{n-1} g(n-1)$ is connected.

Iteratively for each $n \in [12, 21]$, Theorem 4(III) implies that any admissible graph of order n and size $\geq \ell(n)$ is connected.

That is, in the venue of proving Theorem 3 we can focus on the connected squarefree graphs only, which we refer to as candidate graphs.

Computing $g(n)$ for $n \in [12, 16]$

To prove Theorem 3 for $n = 12$, we generate all candidate graphs of order 12, size $\ell(12) + 1 = 20$, and minimum degree $\geq 20 - g(11) = 3$, which we then test for admissibility. There are 18 candidate graphs and none are admissible, which proves $g(12) = 19$.

Similarly, for $n = 13$, we generate all candidate graphs of order 13, size $\ell(13) + 1 = 22$, and minimum degree $\geq 22 - g(12) = 3$. There are 173 such candidate graphs, none of which are admissible. Thus, $g(13) = 21$.

From $g(13) = 21$, Theorem 4(II) implies that $g(14) \leq 24$, which matches the lower bound. Therefore, we obtain $g(14) = 24$ without any computation.

For order $n = 15$, we test if there is any admissible graph of size $\ell(15) + 1 = 27$. By Theorem 4(I) such a graph should have minimum degree ≥ 3. We generate $8,280$ such candidate graphs, but our check shows that all of them are inadmissible. Hence, $g(15) = 26$.

For order $n = 16$, Theorem 4(II) implies $g(16) \leq 29 = \ell(16)$, and thus $g(16) = 29$.

Computing $g(n)$ for $n \in [17, 18]$

For order $n = 17$, Theorem 4(I) implies that any admissible graph of size $\ell(17) + 1 = 32$ should have minimum degree ≥ 3. In fact, minimum degree should be exactly 3, since otherwise the size would be at least $17 \cdot 4/2 = 34$. There are $1,023,100$ such candidate graphs, which, in principle, are possible to inspect directly, although it would be quite time consuming.

Instead, we approached this problem from another angle: noticing that the removal of a vertex of degree 3 from an admissible graph of order 17 and size 32 results in a maximum admissible graph (MAG) of order 16. We constructed all MAGs of order 16 (as explained below) and established that none of them extends to an admissible graph of order 17 and size 32, thus proving that $g(17) = 31$.

For order $n = 18$, Theorem 4(II) implies $g(18) \leq 34 = \ell(18)$ and thus $g(18) = 34$.

Computing $g(n)$ for $n \in [19, 20]$

For order $n = 19$, Theorem 4(I) implies that any admissible graph of size $\ell(19) + 1 = 37$ should have minimum degree ≥ 3. With necessity such a graph has a vertex of degree 3 and we proceed similarly to the case $n = 17$. We construct MAGs of order 18 and show that none of them can be extended to an admissible graph of order 19 and size 37. It follows that $g(19) = 36$.

For order $n = 20$, Theorem 4(I) implies that any admissible graph of size $\ell(20) + 1 = 40$ should have minimum degree ≥ 4, implying that this graph is regular of degree 4. There are $15,156$ such candidate graphs, all of which are inadmissible. Hence, $g(20) = 39$.

Computing $g(21)$

For order $n = 21$, Theorem 4(I) implies that any admissible graph G of size $\ell(21) + 1 = 42$ should have the minimum degree $d \geq 3$. Also, $21d/2 \leq 42$ implies that $d \leq 4$.

- $d = 3$: We construct MAGs of order 20 and show that none of them can be extended to an admissible graph by adding a vertex of degree 3.
- $d = 4$: Such graph G would be 4-regular. Since G is squarefree, its girth g is either 3 or ≥ 5.
  - $g = 3$: Removing from G vertices forming a 3-cycle results in an admissible graph G' of order 18 and size 31, composed of 3 vertices of degree 2 and 15 vertices of degree 4. Further removing from G' two degree-2 vertices we obtain a MAG of order 16, which however cannot contain a vertex of degree 2.
  - $g \geq 5$: There exist 8 such graphs as computed by tool `GENREG`, which are listed at `https://mathe2.uni-bayreuth.de/markus/reggraphs.html` Our test shows that they all are inadmissible.

It follows that $g(21) = 41$.

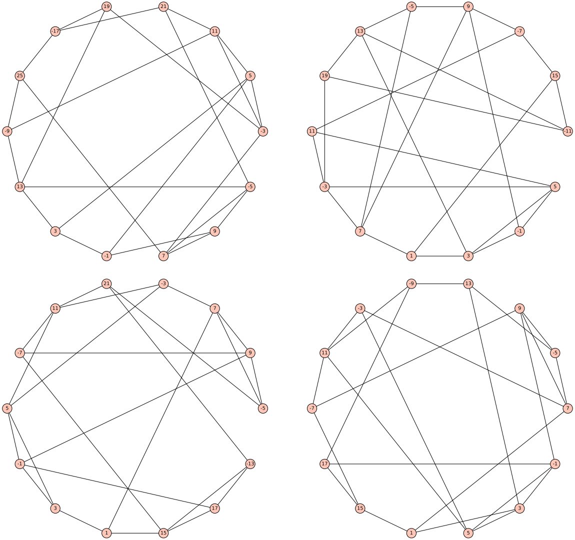
This concludes our proof of Theorem 3 (modulo the announced MAGs computations).

MAGs of order 14

The maximum admissible graphs of each order $n \leq 14$ can be obtained directly from the candidate graphs generated by `nauty`.

In particular, for $n = 14$ we can restrict our attention to the $2,184$ connected squarefree graphs of minimum degree 3, among which we identified only 4 MAGs:

MAGs of order 14



MAGs of order $n \in [15, 18]$

To construct MAGs of order $n \in [15, 18]$, which by Theorem 4(I) have minimum degree $\geq g(n) - g(n-1)$, we generate and test for admissibility candidate graphs of two types:

- *extended* graphs resulted from adding a vertex of degree $g(n) - g(n-1)$ to a MAG of order $n-1$; and
- *denovo* graphs with minimum degree $\geq g(n) - g(n-1) + 1$ generated by `nauty`. Clearly, denovo candidate graphs may exist only if $n(g(n) - g(n-1) + 1) \leq 2g(n)$, i.e., $g(n) \leq \frac{n}{n-2}(g(n-1) - 1)$.

For $n = 15$, there are 124 extended and 33,608 denovo candidate graphs, among which we identified $20 + 8 = 28$ MAGs.
For $n = 16$, there are 243 extended and no denovo candidate graphs, delivering just 2 MAGs.
For $n = 17$, there are 82 extended and 5,847,706 denovo candidate graphs, delivering $11 + 7 = 18$ MAGs.
For $n = 18$, there are 287 extended and no denovo candidate graphs, delivering just 2 MAGs.

MAGs of order 19

For MAGs of order 19, we obtain 15 MAGs of minimum degree 2 from the extended graphs. However, the number of denovo candidate graphs appears to be prohibitively large.

MAGs of order 19

We therefore take a different route by first computing *sub-maximum addmissible graphs* (sub-MAGs) that have size just one less than that of MAGs. Namely, we compute:

- sub-MAGs of order 14 composed of 400 graphs of minimum degree 2 extended from MAGs of order 13 and 34 denovo graphs of minimum degree 3;
- sub-MAGs of order 15 composed of 54 graphs of minimum degree 1 extended from MAGs of order 14, 1597 graphs of minimum degree 2 extended from sub-MAGs of order 14, and 70 denovo graphs of minimum degree 1;
- sub-MAGs of order 16 composed of 144 graphs of minimum degree 2 extended from MAGs of order 15 and 36 denovo graphs of minimum degree 3;
- sub-MAGs of order 17 composed of 32 graphs of minimum degree 1 extended from MAGs of order 16, 909 graphs of minimum degree 2 extended from sub-MAGs of order 16, 0 denovo graphs of minimum degree 3 without a connected pair of degree-3 vertices,[1] and 131 graphs extended from sub-MAGs of order 15 with a connected pair of degree-3 vertices;
- sub-MAGs of order 18 composed of 124 graphs of minimum degree 2 extended from MAGs of order 17 and 50 graphs of minimum degree 3 extended from sub-MAGs of order 17.

Then we get MAGs of minimum degree 3 by extending sub-MAGs of order 18, resulting in 7 such MAGs. So, we have $15 + 7 = 22$ MAGs of order 19.

MAGs of order 20

MAGs of order 20 have the minimum degree 3 and are the result of an extension of MAGs of order 19, which gives just 2 MAGs.

Outline

Computational Complexity and Benchmarks

While it seems difficult to give an accurate estimate for the computational complexity of the proposed algorithms, the obtained practical results show that they are quite efficient in practice:

Performance benchmarks for testing (inadmissible) graphs of order n, size $\ell(n) + 1$, and minimum degree ≥ 3, for $n \in \{12, 13, 15, 20\}$:

Graph order	12	13	15	20
# candidate graphs	18	173	8280	15,156
# graphs with an MFS	16	159	8252	14,591
# graphs tested w. GraphSolve	2	14	28	565
Average test time (sec.)	6.5	17.5	24.5	247.9

Performance benchmarks for identifying maximum admissible graphs:

Graph order	14	15	16	17	18
# candidate graphs	2,184	33,732	243	5,847,788	287
# graphs with an MFS	1,976	29,251	215	5,734,238	257
# graphs tested w. GraphSolve	208	4,481	28	113,550	30
Average test time (sec.)	30.9	44.2	51.0	81.4	79.1

Concluding Remarks

- Algorithm SolveInPowers can be easily extended to powers of other primes, and further to powers of arbitrary positive integer b by introducing independent variables for powers of each prime dividing b.
- While we applied algorithm SolveInPowers to *linear* equations and inequations, the linearity appears to be inessential and the algorithm should work equally well for *polynomial* (in)equations.
- Performance of algorithm SolveInPowers is sensitive to the number of nonzero coefficients in equations, which in algorithm GraphSolve are derived from the left kernel K_l of the given graph incidence matrix. We found that upfront LLL reduction of K_l helps to reduce the number of nonzero coefficients and greatly improves the performance.
- We have also explored the idea to perform LLL reduction of the equations in E after each substitution, but it seems(?) to not provide much benefit.
- Preprint is available at `https://arxiv.org/abs/2303.02872`, soon will appear in *Journal of Computer and System Sciences*.