

Set Intersection with Minimal Support

The SIMS-Problem

By Jonas Seiler and Cecilia Knäbchen

Contents

1. Problem definition	3
2. Best known bounds	4
3. Upper Bounds	5
4. Lower Bounds	7
5. Normal Form	8
6. Combinatorial approach	9
7. Linear Programming approach	10
8. Addendum	11

1. Problem definition

Given a number $n \in \mathbb{N}$, find the minimal number $k \in \mathbb{N}$ such that there are n sets A_1, \dots, A_n containing numbers in $[k]$, i.e. $A_i \subseteq \{1, \dots, k\}$ satisfying:

$$|A_i \cap A_j| = |i - j| \text{ for all } 1 \leq i < j \leq n$$

For example, for $n = 4$, the answer would be $k = 5$, with which we could pick the 4 sets as:

$$A_1 = \{1, 2, 3, 4\}$$

$$A_2 = \{1, 5\}$$

$$A_3 = \{1, 2\}$$

$$A_4 = \{1, 3, 4, 5\}$$

or a more visual alternative:

$$\begin{array}{llll} A_1: & 1 & 2 & 3 & 4 \\ A_2: & 1 & & & 5 \\ \text{identity for } \sum_{d|n} \phi(d) = n & & & & \\ A_3: & 1 & 2 & & \\ A_4: & 1 & & 3 & 4 & 5 \end{array}$$

You can try to find sets which only use the numbers 1 to 4 but will hopefully be convinced that $k = 5$ is optimal.

2. Best known bounds

In the following table, we record our best known values for k .

n	optimal value with combinatorial solver	optimal value with LP solver
0	0	0
1	0	0
2	1	1
3	2	2
4	5	5
5	9	9
6	16	16
7	24	24
8		36
9		50
10		70
11		91
12		120
13		150
14		189
15		231
16		280
17		336
18		398
19		468
20		547
21		630
22		728
23		≤ 827
24		≤ 944
25		≤ 1064
26		≤ 1198
27		≤ 1341
28		≤ 1493
29		≤ 1661
30		≤ 1838
31		≤ 2027
32		≤ 2232
33		≤ 2442
34		≤ 2680
35		≤ 2918
36		≤ 3179

Our strategy in solving this problem combinatorically will be explained in Section 6. Our formulation of this problem as an (I-)LP will be explained in Section 7

3. Upper Bounds

Our current best upper bound comes from an explicit construction. This bound is not tight, there are more optimal solutions starting from $n = 6$, but we have not yet been able to derive a pattern out of those.

Our construction works as follows:

For each set-distance $i \in \{1, \dots, n-1\}$, and for each coset representative $a \in \{0, \dots, i-1\}$, if there are atleast 2 two set indices $b, c \in \{1, \dots, n\}$ with $b \bmod i = c \bmod i = a$, i.e. they lie in the same coset, then define $\varphi(i)$ new unused numbers to add to all sets with indices in that coset and increase k by $\varphi(i)$, the number of newly added and used numbers.

Remember that $\varphi(i)$ is Euler's totient function, the number of relatively prime integers up to i .

For example, for $n = 6$, this constructions yields the following:

	i=1	i=2		i=3		i=4		i=5
A ₁ :	1	2		4 5		10 11		14 15 16 17
A ₂ :	1		3		6 7		12 13	
A ₃ :	1	2			8 9			
A ₄ :	1		3	4 5				
A ₅ :	1	2			6 7	10 11		
A ₆ :	1		3		8 9		12 13	14 15 16 17

From this we can also see that this is not optimal, since for $n = 6$, there is a solution with $k = 16$.

EXPLICITCONSTRUCTION(n):

```

1   $A_1, \dots, A_n = \{\}$ 
2   $k = 0$ 
3  For  $i \in \{1, \dots, n-1\}$ 
4      For  $a \in \{1, \dots, \min(i, n-i)\}$            // cosets with atleast two set-indices
5          For  $b \in \{1, \dots, \varphi(i)\}$            // number of elements to add
6              For  $j \in \{0, 1, \dots, (n \div i) - 1\}$  // set-indices to add to
7                   $A_{a+ji} = A_{a+ji} \cup \{k+b\}$ 
8              End
9              If
10                  End
11                   $k += \varphi(i)$ 
12          End
13  End
14  return  $A_1, \dots, A_n$ 
```

We first prove that this construction indeed yields a solution and then calculate it's size.

For sets A_a and A_b with $|a - b| = k$, we look at divisors d of k . For each divisor, they share $\varphi(d)$ elements. Together they therefore share $\sum_{d|k} \varphi(d) = k$ elements as wanted. This last identity is due to Euler. A proof can be found in.

To see that they share exactly those elements, see that they only share $\varphi(i)$ elements for every $i < n$ such that they are in the same coset of $\mathbb{Z}/i\mathbb{Z}$, i.e.

$$\begin{aligned} a \bmod i &= b \bmod i = c \\ \Leftrightarrow a &= ui + c \wedge b = vi + c \\ \Leftrightarrow k &= a - b = i(u - v) \\ \Leftrightarrow i &\mid k \end{aligned}$$

For $i > k$, a and b are never in the same coset, and therefore a and b share exactly $\varphi(i)$ elements for every $i \leq k$ with $i \mid k$ as stated.

Now we want to estimate k in dependency of n . Using the terms for the

4. Lower Bounds

For a lower bound on k , we first calculate a lower bound on the number of elements needed in A_i that cannot be contained in A_j for $j < i$. We call the set of these elements A_i^{new} .

For this, we first calculate some lower bound on $|A_i|$. By using a weaker form of the inclusion-exclusion principle, we see that

$$\begin{aligned}
 |A_i| &= \left| \bigcup_{\substack{j=1 \\ j \neq i}}^n A_i \cap A_j \right| \\
 \text{inclusion-exclusion} &\geq \sum_{\substack{j=1 \\ j \neq i}}^n |A_i \cap A_j| - \sum_{\substack{1 \leq j < k \leq n \\ j, k, i \text{ distinct}}}^n |(A_i \cap A_j) \cap (A_i \cap A_k)| \\
 &\geq \sum_{\substack{j=1 \\ j \neq i}}^n |A_i \cap A_j| - \sum_{\substack{1 \leq j < k \leq n \\ j, k, i \text{ distinct}}}^n |(A_i \cap A_j) \cap (A_i \cap A_k)|
 \end{aligned}$$

Without any additional knowledge, we do not know anything about $A_i \cap A_j \cap A_k$, so we cannot use a stronger version of the inclusion-exclusion principle. Next, we see that by our problem definition, $|A_i \cap A_j| = |i - j|$ and

5. Normal Form

For each solution, we can rename the elements to obtain another solution. Since such solutions are “the same” in some sense, we would like some way to check if two solutions are “the same” or if they really differ in a significant way. While we could count how many elements each set uses, there are solutions which use the same number of elements while still not being able to be transformed into each other via a renaming of used elements.

In this section we therefore want to introduce a normal form for solutions which is the same for all solutions arising from another via renaming of elements.

Each element is either contained in a set A_i or not. We can therefore associate an element k with a n -dimensional vector v^k with $v_i^k = 1$ if $k \in A_i$ or 0 otherwise.

For example, in the following solution for $n = 4$

```
A1:  1  2  3  4
A2:  1           5
A3:  1  2
A4:  1       3  4  5
```

we have $v^1 = (1, 1, 1, 1)$ or $v^2 = (1, 0, 1, 0)$. Writing our solution in the format above, the vectors are just the columns with 0s filled in the blank spots.

We therefore transform our solution into one, where the elements are renamed in such a way, that the vectors are sorted lexicographically.

The exemplary solution above is already in this normal form while the following solution isn't:

```
A1:  1  2       4  5
A2:  1       3
A3:  1  2
A4:  1       3  4  5
```

Here, $v^3 = (0, 1, 0, 1)$ is sorted before $v^4 = (1, 0, 0, 1) = v^5$.

It is clear that this normal form is unique by the uniqueness property of sorted lists up to equivalent elements, which would be elements that are in the exact same set in our case, which can be renamed at will anyway.

It is also clear that each solution has a normal form, since we can always define these column vectors and sort them.

Another way to create a solution out of another, is to switch the sets A_i with $A_{n-(i-1)}$, i.e. A_1 becomes A_n , A_2 becomes A_{n-1} and so on. Our normal form is not resistant to this renaming and we have not found a “cheap” normal form that is invariant under both transformations though we welcome all suggestions.

6. Combinatorial approach

7. Linear Programming approach

8. Addendum

Euler phi function identity and citation in Chapter 3