Technische Universiteit Delft
Faculteit Elektrotechniek, Wiskunde en Informatica
Delft Institute of Applied Mathematics

”Hyperplane covering problems” (Nederlandse titel: ”Overdekkingsproblemen met hypervlakken”)

Verslag ten behoeve van het
Delft Institute of Applied Mathematics
als onderdeel ter verkrijging

van de graad van

BACHELOR OF SCIENCE
in
TECHNISCHE WISKUNDE

door

YVONNE DEN BAKKER

Delft, Nederland
Augustus, 2021

BSc verslag TECHNISCHE WISKUNDE

"Hyperplane covering problems"

(Nederlandse titel: "Overdekkingsproblemen met hypervlakken")

YVONNE DEN BAKKER

Technische Universiteit Delft

Begeleider

Dr. A. Bishnoi

Overige commissieleden

Dr. T.W.C. Vroegrijk

Augustus, 2021 Delft

Chapter 1

Introduction

How many lines do you need to cover a grid? How many hyperplanes do you need to cover the vertices of a hypercube? What happens if you're not allowed to cover a certain point, say, the origin? Does it matter which point you're missing? How do cover sizes behave when you're covering a subspace not once, but multiple times? All of these questions refer to the subspace covering problem. There are many different variants of this problem, ranging from very easy to intimidatingly difficult to solve. In my thesis, I will explore many such questions. To start, I will discuss the variants of the problem that I studied and will reappear later on.

The first of these is the binary hypercube $\{0,1\}^n$ contained in $\mathbb{F}^n$, for some field $\mathbb{F}$. If you were to cover every point of this hypercube, two parallel hyperplanes of the form $\{\vec{x} : x_1 = 0\}$ and $\{\vec{x} : x_1 = 1\}$ suffice and it can be easily shown that this is indeed the minimum number of hyperplanes needed. However, if we were to limit the set of covered points to all but the origin, an optimal cover becomes a lot more difficult to find. One way of covering such a space is by taking the n hyperplanes $\{\vec{x} : x_i = 1\}$ for $i \in \{1, 2, \ldots, n\}$. In the next chapter I will discuss whether or not this much larger cover is indeed optimal. We will also see how the choice of the field $\mathbb{F}$ can have an impact on the answer to some covering problems where the points are covered more than once.

The main problem that I am interested in is as follows. Let $S_1, \ldots, S_n$ be finite subsets of a field $\mathbb{F}$. We can create the set $S = S_1 \times \cdots \times S_n$, which I will refer to as a grid in $\mathbb{F}^n$. Let k be a positive integer and p a point in this grid. We want to find the smallest number of hyperplanes required to cover all points of this grid at least k times, except for the point p, which is not covered at all. We will also see some variants where the point p is covered up to $k-1$ times. We will generally denote this minimum by the function f, whose parameters will be the grid S, the integer k, and the underlying field $\mathbb{F}$. As translations don't have any impact on the cover size, we will assume that the origin is always contained in the grid.

This problem is very broad, as it allows variations in the field, multiplicity and rules regarding the point that is avoided by the cover. I will give a short overview of the variants I will study the most. All the variants I consider will have $k \geq 2$, which is in part because for $k = 1$, very strong results have been proven already. For higher multiplicities, there is much more to discover. Apart from that property I will mainly consider two fields. The first of which is $\mathbb{R}$, the second is $\mathbb{F}_q$. For grids in $\mathbb{R}^N$, I will sometimes consider non-origin points p to be the missing point, but mostly focus on cases where p is the origin. On these grids, this point cannot be covered. For $\mathbb{F}_q^n$, the missing point will always be equal to the origin, while it may be covered up to $k-1$ times.

1.1 Overview of the thesis

In this section, I will give an overview of the topics I will discuss in my thesis. The next chapter is dedicated to giving an overview of the most important developments in this field thus far. I will discuss celebrated results and hint to the direction that I will take with my own research.

In chapter 3 I will talk about different possible variants of this problem, and how much a seemingly small change can significantly alter the solution. I will not cover every possible variant of the subspace covering problem, there are simply too many possibilities. In chapter 2, I will mention a few of these variants. The variants I will consider here do, in one way or another, lead up to the variations of the problem I will cover in later chapters.

Chapter 4 is about grids in $\mathbb{R}^2$. In this chapter I discuss coverings with multiplicity of $n \times m$ grids in $\mathbb{R}^2$. My first result is a proof on the covering of a grid $n \times 2$ for any $n \geq 2$. I will show that for any such grid, if you want to cover all but one point, a minimal cover contains exactly $k(n-1)+1+\lfloor \frac{k-1}{n} \rfloor$ lines. After considering the simple case, I will discuss the Integer Programming model I created in order to generate values on general grids. Following the outcome of this program, I managed to prove an interesting theorem on general $n \times m$ grids. I showed that for $n \geq (k-1)(m-1)+1$, an inequality that can also be written as a bound for multiplicity k as $k \leq \frac{n-1}{m-1}+1$, the minimum cover size is exactly equal to $f(n,m,k) = k(n-1)+(m-1)$.

Chapter 5 can be seen as an extension of chapter 4, as in this chapter I discuss $n \times m \times l$ grids in $\mathbb{R}^3$. In order to see how the cover sizes behave in higher dimensions, I altered the Integer Programming model I used in the previous chapter. The data from the model made the relationship between the two- and three-dimensional grids clear, which led me to find an upper bound for three-dimensional grids. It even turned out that this could be applied to the construction I created in the previous chapter, which together prove the exact minimum cover size. I also managed to generalize the first of these two results, closing the chapter with an upper bound for $N+1$-dimensional grids.

In chapter 6 I will discuss grids in $\mathbb{F}_q^n$. The main result of this chapter is a generalisation of a theorem in the paper on covering binary hypercubes with multiplicity by Bishnoi, Boyadzhiyska, Das and Mészáros [5]. They proved how big the minimal cover of those hypercubes should be, for certain values of k and n. It turns out that this result does not only hold for binary hypercubes, but can, when phrased slightly differently, be applied to any subspace $\mathbb{F}_q^n$. In the next part of this chapter I will discuss the Integer Programming model that calculates the minimal cover size of a space $\mathbb{F}_q^n \setminus \{0\}$. Unlike the model in $\mathbb{R}^2$, I was not able to prove any result following the outcome.

Chapter 2

Background

Before I started studying this problem, it piqued the interest of plenty of researchers, just as it did mine. Many different variations of this problem have been studied, see for example [5, 11, 1, 3]. One of the most recent papers on this topic, written by my supervisor, Anurag Bishnoi, together with Boyadzhiyska, Das and Mészáros [5], was only posted on arXiv earlier this year. To fully understand this problem, it's necessary to know about some of the history and results leading up to what is known today. The following description of the background is for a large part based on the paper of Bishnoi, Boyadzhiyska, Das and Mészáros.

2.1 Early stages

The first important result was proven by Jamison [15] in the late 1970s, as a response to a question asked by Jean Doyen during a conference. Consider the following problem: How many affine subspaces of codimension d (where a subspace with $d = 1$ is a hyperplane) do you need to cover all nonzero points in $\mathbb{F}_q^n$ while avoiding the origin? He found that the minimum number of affine subspaces needed was equal to $q^d - 1 + (n - d)(q - 1)$. This result proves to be very valuable immediately. When you substitute $q = 2$ and $d = 1$, you get that the minimum number of hyperplanes needed to cover the binary hypercube $\mathbb{F}_2^n$ except for the origin, is equal to n. If we compare this bound to the cover described in the introduction, we conclude that this cover is indeed optimal. Jamison wasn't the only one who managed to prove this statement. Brouwer and Schrijver [8] proved the same theorem, restricted to $d = 1$, but now with a much shorter proof. Instances like these show that there is not just one set way of proving results of this type. We will see that there are plenty of different tools that can be used, ranging from the polynomial method to the probabilistic method.

An even more general way of describing the subspace covering problem is as follows. Taking the Cartesian product of finite subsets $S_1, S_2, \dots, S_n$ of an arbitrary field $\mathbb{F}$, Alon and Füredi [3] proved that if you were to cover all but one point of $S_1 \times S_2 \times \cdots \times S_n$, the minimum number of hyperplanes needed is equal to $\sum_i (|S_i| - 1)$. This bound is tight, as shown by the following collection of hyperplanes: If $(a_1, \dots, a_n)$ is the point missed, then all hyperplanes of the form $x_i = s$, for $1 \leq i \leq n$ and $s \in S_i \setminus \{a_i\}$, form such a cover of size $\sum_{i=1}^{n} (|S_i| - 1)$. The proof method of Alon-Furedi is similar to the one by Brouwer-Schrijver, and it was an important milestone in the development of the so-called polynomial method [9] which has led to the resolution of several important problems in various branches of mathematics.

Apart from applications in finite geometry and Ramsey theory [16], the main stimulant for working on these problems were the proof methods used. The most important of these was the polynomial method. This method gained a lot of traction after it was used to solve a major open problem concerning the Kakeya conjecture [12], and more recently, the cap-set problem

[13]. While the subset covering problem is by no means new, after increased interest in the polynomial method, more researchers started looking into this problem again. Due to earlier results of Jamison, the polynomial method is sometimes referred to as the Jamison method in finite geometry [9]. The idea of this method is the following:

Straight lines in a two-dimensional grid can be described by the equation $f(x, y) = ax+by+c$, where this equation is equal to zero at all the points that lie on this line. For any point not on the line, the polynomial will not be equal to zero. A similar thing holds for hyperplanes in higher dimensional spaces, which can be described by multi-variate polynomials. A hyperplane defined as $H_i = \{\vec{x} : \vec{x} \cdot \vec{a_i} = c_i\}$ for a normal vector $\vec{a_i} \in \mathbb{F}_n$ and a constant $c_i \in \mathbb{F}$, corresponds to a polynomial $f(\vec{x}) = \prod_i(\vec{x} \cdot \vec{a_i} - c_i)$. Again, $f(\vec{x}) = 0$ if and only if $\vec{x}$ is covered by the hyperplane H_i. A hyperplane covering can therefore be described as the product of a bunch of polynomials, creating a polynomial with higher degree (this degree is exactly equal to the amount of hyperplanes in the cover). This new polynomial clearly satisfies the criteria that it vanishes with certain multiplicity on each point that needs to be covered, while it does not vanish on the origin or another missing point. If we find lower bounds for the degrees for polynomials of this type, it also implies a lower bound on the minimal cover size.

2.2 Other variants

Until now, the problem was still relatively straightforward. The goal was to find the minimal size of a single cover of all but one points of relatively simple subspace. The previous few results gave very strong results for this problem, so the most logical thing to do next was to extend it. There are three main ways this can be done. The first way is by considering variations over different subspaces. One possibility is to consider variants over rings, like Kós, Mészáros and Rónyai [17] and Bishnoi, Clark, Potukuchi and Schmitt [6] did. Blokhuis, Brouwer and Szõnyi [7] studied this problem for quadratic surfaces and Hermitian varieties in projective and affine spaces over $\mathbb{F}_q$. Aaronson, Groenland, Grzesik, Johnston and Lomiej Kielaka [1] went an entirely different route. In the setting of covering a binary hypercube $\{0, 1\}^n \subseteq \mathbb{R}^n$, they considered a variant where the cover avoided up to four points, instead of just one. A final way to extend this problem is by covering with multiplicity, meaning that every point, except for the origin, will be covered at least k times. This is also the direction on which I will focus. When covering with multiplicities, there are again two different versions of the problem. One option is to completely omit the origin, the other is to cover the origin at least one time fewer than the rest of the points.

2.3 Variants with higher multiplicity

The first setting, in which the origin isn't covered at all and every other point is covered with certain multiplicity, was studied by Bruen [10] over finite fields and by Ball and Serra [4] over finite grids in arbitrary fields. While they did not always find tight bounds, some results turned out to be rather important. The Punctured Combinatorial Nullstellensatz proven by Ball and Serra, which is a variation on Alon's Combinatorial Nullstellensatz [2], is especially useful. In Theorem 4.1 in their paper, they proved the following result:

Theorem 2.1. *Let f be the size of the cover that covers all but one point of the grid $S_1 \times S_2 \times \cdots \times S_n$ k times, while not covering one point. Then $f \geq (k-1)\ max_j(|S_j|-1)+\sum_{i=1}^{n}(|S_i|-1)$.*

The work of Clifton and Huang [11] renewed interest in these problems. They studied the situation of covering all nonzero points of $\{0, 1\}^n \subseteq \mathbb{R}^n$ at least k times while avoiding the origin.

They introduced three different bounds, each for varying values of k. To show the strength of these results, I will compare them to an easily found lower bound following from the Alon-Füredi theorem. This bound is the following: If you take any k-cover in the same setting as before, it's possible to arbitrarily remove $k-1$ hyperplanes and be left with at least a 1-cover. It follows from the Alon-Füredi theorem that such a cover consists of at least n hyperplanes. Therefore, any k-cover will not be smaller than $n+(k-1)$. In the situation where $k=2$, the minimum cover size will thus be equal to $n+1$. This bound even turns out to be optimal, as a cover of that exact size can be found. It's simply equal to the n hyperplanes denoted by $\{\vec{x} : x_i = 1\}$ for $i \in [n]$ together with the hyperplane $\vec{x} = \vec{1}$. For higher values of k there is however plenty of room for improvement. Clifton and Huang showed that for $k=3$ and $n \geq 2$, the answer is equal to $n+3$, which differs by exactly 1 from the previous result. For $k \geq 4$ and $n \geq 3$, the difference is even larger. For these values, they conclude that the answer must lie somewhere between $n+k+1$ and $n+\binom{k}{2}$. Once k becomes very large with respect to n fixed, their answer changes significantly and they conclude with a tight lower bound of size $(c_n + o(1))k$, for c_n the nth term in the harmonic series. These bounds are a great improvement compared to what we found previously.

Even greater progress was achieved after the work of Sauermann and Widgerson [18]. Using the polynomial method, they concluded that for $k \geq 2$ and $n \geq 2k-3$, the lower bound can be improved to $n+2k-3$. For $k=3$, this lower bound is the same as the one Clifton and Huang found. The strength of this new result lies in the bounds for higher multiplicities. Apart from an improved lower bound, the theorem they proved also implied that applying the polynomial method in a simple manner like this will not result in improved bounds.

What Sauermann and Widgersons theorem implies, is that $n+2k-3$ is also an upper bound to the polynomial problem. To be exact, they showed that there exists a polynomial satisfying all criteria with degree $n+2k-3$ no matter how often (as long as it's less than $k-1$)[1] the origin is covered. It follows that, using this version of the method, it's impossible to find a better bound than this. There is however reason to believe that the actual answer for the hyperplane covering problem is different. Clifton and Huang have conjectured that the answer should be equal to $n+\binom{k}{2}$, which follows from one of their constructions. It is also the best upper bound they could think of. One might imagine a version of the polynomial method where the added assumption that the polynomial is a product of linear factors is able to give an improved bound. While that might be possible, no one has managed to prove something like it. This can be considered a setback, but by no means does it suggest the end of looking for ways to find better bounds. The result Sauermann and Widgerson proved only holds for fields with characteristic zero, and besides, the polynomial method isn't the only tool available to solve these types of problems.

Another way to continue advancing in this field is to abstain from using the polynomial method completely. Bishnoi, Boyadzhiyska, Das and Mészáros [5] approached the problem more directly with combinatorial techniques. They studied the variant where they covered every point of the binary hypercube $\mathbb{F}_2^n$ k times, while covering the origin up to $k-1$ times, using affine subspaces of codimension d. Just like Clifton and Huang, they concluded that the minimal cover size depends on how large k is with respect to n. In their paper, they showed (for $d=1$) that for $k \geq 2^{n-2}$, the minimal cover size is exactly equal to $2k - \lfloor \frac{k}{2^{n-1}} \rfloor$. When $n > 2^k$, they found that the minimal cover size has value $n+2k-3$. This can be compared to what Sauermann and Widgerson proved in their paper, which is that there exists a polynomial of degree $n+2k-4$. This illustrates the gaps between the polynomial and the hyperplane problem.

[1] If the origin is covered exactly $k-1$ times, such a polynomial cannot exist. In that case the answer to polynomial problem as well as the hyperplane problem turns out to be $n+2k-2$.

All of the developments described in this chapter brought the subspace covering problem to where it is now, and how I will look at it. Some of the past results, in particular those by Alon-Füredi and Ball-Serra, proved very useful to me. In this thesis, I will attempt to advance the development of this problem even further.

Chapter 3

Difference between variants

The problem I'm studying has plenty of variants. It's possible to vary what kind of subsets you cover, how often you want to cover each point, if you allow for the missing to be covered and which point you want your cover to avoid. The most important of these factors to me is the multiplicity. The only overlapping aspect of all the versions I studied is that for each problem, I considered higher multiplicities and thus covered the points more than once. In this chapter I will illustrate the difference between a few different variants of the problem, using three different 3×3 grids where different criteria are applied to the missing point. As this chapter is mainly here to serve illustrative purposes, I will not prove every statement I make. Instead, I will rely on the minimal cover sizes that follow from the models I programmed and will discuss in later chapters. Before I start with the comparisons, I would like to acknowledge Alexander Clifton, who told my supervisor about the example $\{0, 1, 2\}^2$ for $k = 3$ where the cover sizes can differ depending on which point is missing.

3.1 Version 1: $\{0, 1, 2\}^2 \in \mathbb{R}^2$

The first grid we will explore is a simple 3×3 grid in $\mathbb{R}^2$ where the distance between each point is equal. In this variant of the problem, we will see that which point is left uncovered does matter.

When the origin remains uncovered, the choice of the point that is missing is indeed relevant for the minimal cover size. This is because the point that is missing determines which lines are not allowed to be a part of the cover. The most interesting of these lines are lines that cover three points, as those cover the grid most efficiently. As there are always two vertical and two horizontal lines that do not go through any missing point, what remains are the two diagonal lines. When a corner point is missing we cannot draw one of the two diagonals. When a middle point is missing we cannot draw either diagonal, and when a side point is missing we can draw both. This turns out to be the reason the minimal cover size depends on the placement of the missing point. In the table below, the cover sizes for these grids and k up to 5 are shown for when the missing point remains completely uncovered. Indeed, for $k = 3$ and $k = 5$, if we miss the middle point of this grid, the cover size is larger than if we miss any other point.

Missing point uncovered				
$\{0, 1, 2\}^2$	$k = 2$	$k = 3$	$k = 4$	$k = 5$
Corner	6	9	12	15
Side	6	9	12	15
Middle	6	10	12	16

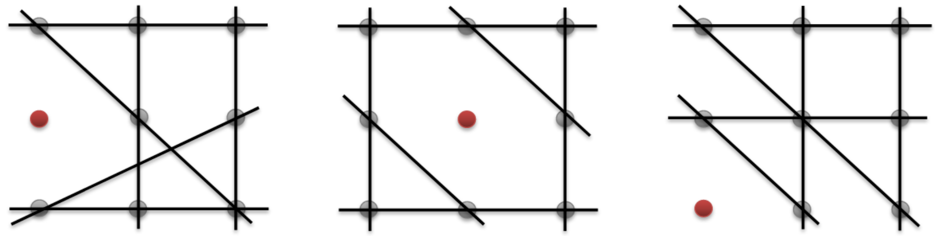

Figure 3.1: A 2-cover of the 3×3 grid for each missing point.

When we do allow the missing point to be covered up to $k-1$ times, we find it does not matter which point is missing, as can be seen in the table below. The explanation for this is simple. When the missing point p is allowed to be covered up to $k-1$ times, a simple construction that would create a cover of the desired size is the following: Take the optimal cover for $k=2$ and then, for $k>2$, add $k-2$ sets of three different parallel lines. The optimal cover for $k=2$ consists of 6 lines. For each of the possible missing points, a possible cover is depicted in Figure 3.1. This results in a k-cover, which can be made for every $k>2$ and any missing point.

Missing point covered up to $k-1$ times				
$\{0,1,2\}^2$	$k=2$	$k=3$	$k=4$	$k=5$
Corner	6	9	12	15
Side	6	9	12	15
Middle	6	9	12	15

3.2 Version 2: $\{a,b,c\} \times \{d,e,f\} \in \mathbb{R}^2$

The second grid is like the first, except there is no information about the distance between each point. Therefore some lines that could be drawn on the $\{0,1,2\}^2$ grid can't be drawn here. Take for example the line $x+y=2$ or equivalently $x+y=b+e$. The first of these also covers the middle point $(1,1)$, while the same can only be said for the second line in very specific situations. In fact, that holds if and only if $e=f-\frac{f-d}{c-a}(b-a)$. Similarly, the equivalent of the line $x-y=2$ in $\{0,1,2\}^2$ can only be drawn when $e=d+\frac{f-d}{c-a}(b-a)$ holds. This does impact the cover size when the missing point cannot be covered. When a middle point is missing, the cover size is the same as for $\{0,1,2\}^2$, as those two lines I just mentioned can't be drawn anyways. When a corner point is missing, either one of those two lines cannot be used, so we do see that the cover size for certain k increases compared to $\{0,1,2\}^2$. If a side point is missing however, while there are two lines that now cannot be drawn, it's possible to cover this with the same amount of lines as was done in $\{0,1,2\}^2$. In order to see what the minimal cover size is for any 3×3 grid, no matter the spacing between the points, I used the grid $\{1,4,9\}^2$. One can check that the previously mentioned specific situations don't apply here, thus making it it a good example. The results are summarized in the table below, giving the minimal cover sizes for this grid when we don't allow the missing point to be covered at all.

Missing point uncovered				
$\{a,b,c\}\times\{d,e,f\}$	$k=2$	$k=3$	$k=4$	$k=5$
Corner	6	10	12	16
Side	6	9	12	15
Middle	6	10	12	16

When we do allow the missing point to be covered $k-1$ times, we find that the same cover as in the $\{0,1,2\}^2$ variant holds here as well. This is because it only relies on lines that are either horizontal, vertical or only need to go through two points. This means that which point is missing is irrelevant, as depicted in the table below.

Missing point covered up to $k-1$ times				
$\{a,b,c\}\times\{d,e,f\}$	$k=2$	$k=3$	$k=4$	$k=5$
Corner	6	9	12	15
Side	6	9	12	15
Middle	6	9	12	15

3.3 Version 3: $\mathbb{F}_3^2$

The final grid we will use to compare in this chapter is $\mathbb{F}_3^2$. Due to the different field, this problem is fundamentally different. Here, it is irrelevant which point is missing, as all lines in this grid cover exactly three points. But just like the variations over $\mathbb{R}$, the cover size can depend on whether or not the missing point may be covered $k-1$ times or not at all. Unlike the grids in $\mathbb{R}^2$, this is not mainly a matter of how many points a line covers, but which configurations are possible.

$\{a,b,c\}\times\{d,e,f\}$	$k=2$	$k=3$	$k=4$	$k=5$
Missing point uncovered	6	8	12	14
Missing point covered up to $k-1$ times	6	8	11	14

The difference in size between covers which may or may not include lines that go through the missing point is depicted in the table above. It's interesting to note that for $k=5$ the value is different, but from that point on it's (temporarily) equal again.

3.4 Comparison

For each of the three versions, the answer to the question if the minimal cover size depends on the point that is not covered, is different. For $\{0,1,2\}^2$ it matters; the results differ between the missing point being in the middle or not. We saw that in the case of $\mathbb{F}_3^2$ it did not matter at all, and that for $\{a,b,c\}\times\{d,e,f\}$ it doesn't matter most of the time, but when specific criteria are met, it does. Another thing to conclude is the difference between allowing the missing point to be covered up to $k-1$ times, or not at all. In all the different versions, for at least one multiplicity k the cover size was higher when we did not allow this point to be covered. But we also saw that for many versions and multiplicities, it actually didn't matter. The difference between covering a grid while avoiding a point at least once, or completely, is not that big after all, when k is small. The takeaway from this chapter is that there are many versions of this problem, and they all differ slightly. Studying this difference might make us better understand results that hold in more general situations, like the Ball-Serra bound, Theorem 2.1.

Chapter 4

Grids in $\mathbb{R}^2$

4.1 Simple grids

My journey into the territory of $\mathbb{R}^2$ only really began after studying a 3×2 grid. I convinced myself that the minimal cover size for covering all but one point of this grid was equal to $2k + 1 + \lfloor \frac{k-1}{3} \rfloor$, a statement which I later managed to prove. I also noticed that unlike most grids, it seems it does not matter which point is missing. This cover is illustrated in Figure 4.1 for both possible missing points.

Figure 4.1: A 3-cover for the 2×3 grid. The black lines create a 1-cover, together with the red line it forms a 2-cover and including the blue lines it becomes the 3-cover.

Following this example, the question arose if something similar would hold in a more general situation, say, a $n \times 2$ grid. In order to show that in these grids it did not matter which point was missing for the minimal cover size, I proved the following result over $n \times 2$ grids in $\mathbb{R}^2$:

Theorem 4.1. *Let $S = \{0, \ldots, n-1\} \times \{0, 1\}$ and $n \geq 2$. Then the minimal amount of lines required to cover every point except for one, which is not be covered at all, is equal to $k(n-1) + 1 + \lfloor \frac{k-1}{n} \rfloor$.*

Proof. We will first prove the upper bound by giving a construction of the desired cover size. This construction contains two different categories of lines, each of which can be part of the cover. All of them are lines going through at least two points of the grid and hence must be one of the following:

- Horizontal lines, going through a whole row of n points;

- Vertical and diagonal lines (vd-lines), which cover two points each, one on the lower and one on the upper row.

The next pieces are the types of points, which have the following types:

- The point p that is left uncovered, which without loss of generality can be assumed to have coordinates $(i_p, 1)$ for some $0 \leq i_p \leq n-1$;
- Points in the same row as the missing point (upper points). There are $n-1$ of these, where each of them can exclusively be covered by n different vd-lines;
- Points in the other row (lower points). Each of these points can be covered with $n-1$ different vd-lines, and all of the points together can be covered using a horizontal line.

The optimal cover contains $1 + \lfloor \frac{k-1}{n} \rfloor$ copies of the line $y = 0$ and the following $k(n-1)$ vd-lines: for every $i \in \{0, 1, \ldots, n-1\} \setminus \{i_p\}$, and $w \in \{1, 2, \ldots, k\}$ we add the line joining $(i, 1)$ and $(i + w - 1 \mod n, 0)$. Note that if $k > n$, then these vd-lines will repeat, and we add them with the required multiplicity.

This construction covers every upper point k times: They all have coordinates equal to $(i, 1)$ for $i \in \{0, 1, \ldots, n-1\} \setminus \{i_p\}$ and we add k lines through all of these points, corresponding to the k different values of w. In order to prove that this cover indeed covers each of the bottom points at least k times, we will consider two different categories of values for k.

First assume that $k = an$ for a a positive integer. We defined the vd-lines to be the lines through $(i, 1)$ and $(i + w - 1 \mod n, 0)$ for every w. When $w \mod n = 1$, the lower points that are covered are of the type $(i, 0)$ for $i \in \{0, 1, \ldots, n-1\} \setminus \{i_p\}$. This excludes the point $(i_p, 0)$. For the next value of w, the point $(i_p + 1 \mod n, 0)$ is excluded. This continues until $w \mod n = n$ and the covered lower points are equal to $(i - 1 \mod n, 0)$ which excludes the point $(i_p - 1, 0)$. Thus, when considering all the lines in the cover for w up to k, for k a multiple of n, every lower point is excluded from a set of lines corresponding to some value of w exactly the same amount of times, each by a different value of $w \mod n$. When $k = an$, this implies that the lower points are each covered $k - a$ times, requiring a copies of the horizontal line to finish the k-cover. As $a = \frac{k}{n} = 1 + \lfloor \frac{k-1}{n} \rfloor$ for these value of k, the given amount of horizontal lines is sufficient to create a k-cover.

For the second case, take $k = an + r$ where a is a non-negative integer and $0 < r < n$ is a positive integer. We will claim that there is always at least one lower point, to be specific $(i_p, 0)$, that is covered only $a(n-1)+r-1$ times. That would mean that in order to create a k-cover, we need to add $a+1$ copies of the horizontal line. For these values of k, $a+1 = \frac{k-r}{n} + 1 = 1 + \lfloor \frac{k-1}{n} \rfloor$, which is indeed the amount of horizontal lines described in the cover. To prove this claim we will show that every lower point is covered at least $a(n-1) + r - 1$ times. For w up to and including an we found before that every lower point is covered exactly $a(n-1)$ times. We're left to show that for $w > an$, every lower point is covered at least $r - 1$ times. This follows from the following. For these vales of w, there are r different sets of $n-1$ lines. Each of these cover all but one of the lower points, which is different for each of these w. Hence, there are r different points that are missed once and are covered only $r - 1$ times. The remaining points are covered r times. This proves the claim.

To sum it up, for all values of k, the described cover indeed covers every point k times, proving the upper bound.

The lower bound can be proven by counting the incidences (p, l), where p is a point on the grid, not being the missing point, and l is any line that could be part of the cover. Note that the minimal cover of any grid $n \times 2$ must contain $k(n-1)$ vd-lines. This holds as the only way of covering an upper point is by using a vd-line. Each vd-line covers each of the

$(n-1)$ upper points exactly once, so $k(n-1)$ vd-lines are required for a k-cover. Denote the number of vd-lines (excluding the required $k(n-1)$ vd-lines) by a, and the number of horizontal lines by b. As a vd-line will cover exactly 2 points and a horizontal line will cover exactly n points, the number of incident pairs is equal to $2(k(n-1)+a)+nb = 2(n-1)k+2a+nb$. As the upper bound for the minimal cover size is equal to $k(n-1)+1+\lfloor\frac{k-1}{n}\rfloor$, we know that $k(n-1)+a+b \leq k(n-1)+1+\lfloor\frac{k-1}{n}\rfloor$. To show that the minimal cover size cannot be smaller than this upper bound, we will assume that $a+b < \lfloor\frac{k-1}{n}\rfloor$. We can now say the following: $2(n-1)k+2a+nb < 2(n-1)k+n(a+b) < 2(n-1)k+n\lfloor\frac{k-1}{n}\rfloor \leq 2(n-1)k+(k-1) = k(2n-1)-1$. This results in a contradiction. The number of incidences should be greater or equal to $k(2n-1)$, as otherwise each of the $2n-1$ points on this grid cannot be covered k times. This implies that the cover cannot be smaller than the upper bound, thus proving the lower bound.

Notice that we never put any restrictions on which point is left uncovered. Therefore, the missing point is indeed irrelevant.

□

4.2 Generating values

As the $n \times 2$ grid is able to illustrate perfectly, figuring out what the minimal cover size is of a grid, with certainty, can be rather hard and laborious. In order to know more about the way these minimal cover sizes behave, I wanted to generate these values using a computer. To be able to do that, I used the model described below. But before I get into that, there is one assumption I need to prove:

Lemma 4.2. *There is always an optimal k-cover in which every line contains at least two points of the grid.*

Proof. Assume that we have a k-cover of a grid. If a line covers only one point, then you can replace it by a line covering that point along with an extra one. After replacement we still have a k-cover of the same size. □

(**Integer Programming Formulation**) Let $\mathcal{H}$ be the list of all lines in $\mathbb{R}^2$ that go through at least two points of an $n \times m$ grid $G \subset \mathbb{R}^2$. Let $\vec{p}$ be a point in G, which we consider to be the missing point. For a point $\vec{v} \in G$, let $\chi^{\vec{v}}$ be the vector of length $|\mathcal{H}| = N$, whose i'th coordinate $\chi_i^{\vec{v}}$ is 1 if the i'th element of $\mathcal{H}$ contains $\vec{v}$ and 0 otherwise.

Let $w_1, \ldots, w_N \in \mathbb{R}$. We need to find out the smallest value of $\sum_{i=1}^N w_i$ subject to the following constraints:

- $w_i \in \mathbb{N}$ for all i.
- for every $\vec{v} \in G \setminus \{\vec{p}\}$, $\sum_{i=1}^N \chi_i^{\vec{v}} w_i \geq k$.
- $\sum_{i=1}^N \chi_i^{\vec{p}} w_i = 0$.

Remark 4.3. In this model, I allowed variation on which point of the grid to miss. This choice was made in light of the findings in chapter 3, which showed that the point that's missing from the grid may impact the optimal cover size.

To program this model, I used the programming language Python in addition to Gurobi [14], a mathematical optimization solver. The code I created can be found in appendix A. In short, the code sets up a grid and determines which points it consists of and which lines go through each set of points. Having created a collection of all possible lines that can make up the cover, the vectors $\chi^{\vec{v}}$ are formed, where each value is 1 if and only if the i'th element of $\mathcal{H}$ covers the point $\vec{v}$. Then, Gurobi finds the minimal size for a cover determined by the three given criteria: The cover size is an integer, every point except for the missing point is covered k times or more and the missing point is not covered at all.

Using this model, I had a lot of freedom on choosing the grids and the missing point. I generated data on evenly spaced grids, quadratically spaced grids and both corner and side points. While this data overlapped for a big part, there were some notable differences. However, I chose to focus on the evenly spaced grids while missing the origin.

From these values, I noticed the following pattern. For certain grid sizes and values of k, the generated minimal cover size was exactly equal to the lower bound given by Ball and Serra's Punctured Combinatorial Nullstellensatz: $k(n-1)+(m-1)$ (see Theorem 2.1). In order to find out why this was happening, I sought to find a construction to create a cover of that exact size. This will be described in the following section. The exact values for which this holds can be found in Figure 4.2 below.

m \ n	2	3	4	5	6	7	8	9	10	11	12
2	5	7	10	13	16	19	22	25	28	31	34
3		9	12	14	17	20	23	26	29	32	35
4			14	16	19	21	24	27	30	33	36
5				18	21	23	26	28	31	34	37
6					23	25	28	30	33	35	38
7						27	30	32	35	37	40
8							32	34	37	39	42
9								36	39	41	44
10									41	43	46

0
1
2
3
4
5
6
7
8
9

m \ n	2	3	4	5	6	7	8	9	10	11	12
2	6	10	13	17	21	25	29	33	37	41	45
3		12	15	19	23	26	30	34	38	42	46
4			18	21	25	28	32	36	39	43	47
5				24	27	30	34	38	41	45	49
6					30	33	36	40	43	47	51
7						36	39	42	45	49	53
8							42	45	48	51	55
9								48	51	54	57
10									54	57	60

Figure 4.2: Minimal cover sizes for the grids $n \times m$. From top to bottom, $k = 3$ and $k = 4$. The colours show the difference between the Ball-Serra lower bound and the optimal cover size for each grid.

It can easily be checked that the values that do not differ from the lower bound satisfy the condition that $n \geq (k-1)(m-1)+1$.

4.2.1 Construction for the upper bound

A construction resulting in a cover exactly as large as the Ball-Serra bound turned out to exist, and it applies to all grids and multiplicities that satisfy $n \geq (k-1)(m-1)+1$, where the missing point is the origin. Assume we have an evenly spaced grid consisting of m points on the vertical axis and n points on the horizontal axis. Without loss of generality we can assume this to be equal to $S = \{0, \ldots, n-1\} \times \{0, \ldots, m-1\}$. Then, the construction is the following:

I will begin by describing the cover for $k = 2$. We need to cover every point twice, except for the origin, which we cannot cover. To do this, we take $m-1$ horizontal lines, given by $y = 1$,

$\ldots$, $y = m-1$ and $n-1$ vertical lines, given by $x = 1, \ldots, x = n-1$. This covered all points with one coordinate equal to zero exactly once and all other points exactly twice. To complete the cover, we add $n-1$ diagonal lines. These can be described as $x+y = 1, \ldots, x+y = n-1$ and cover all points with a single coordinate equal to zero. Hence, this cover is a 2-cover.

For any $k > 2$ we need to add extra sets of lines. This set of lines for each additional value $w = 3, \ldots, k$ is the following: A set of $(w-2)(m-1)$ vertical lines, to be described as $x = n-(w-1)(m-1), x = n-(w-1)(m-1), \ldots, x = n$, and a set of diagonal lines to be described as $x+y = 1, \ldots, x+y = n-(w-1)(m-1)-2, x+y = n-(w-1)(m-1)-1$. Note that this total set of lines has size $n-1$. The 2-cover has size $(m-1)+2(n-1)$. If you add these for every value of w up to k, you get $k(n-1)+(m-1)$ which is exactly the cover size promised.

To illustrate this construction, I've added three pictures showing the required lines to cover a 7×3 grid for $k = 2, 3$ and 4.

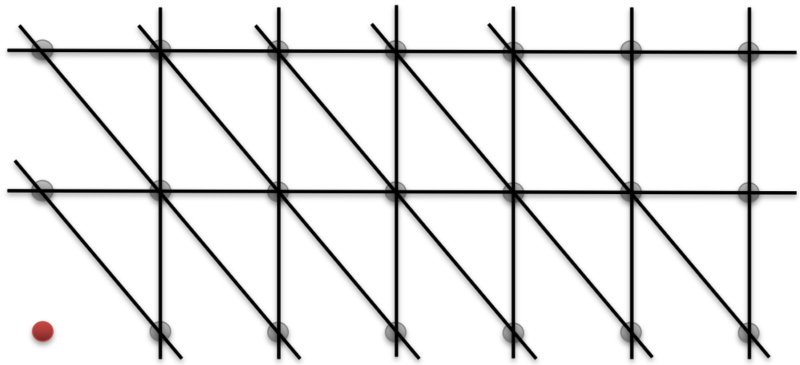

Figure 4.3: A 2-cover for the 7×3 grid.

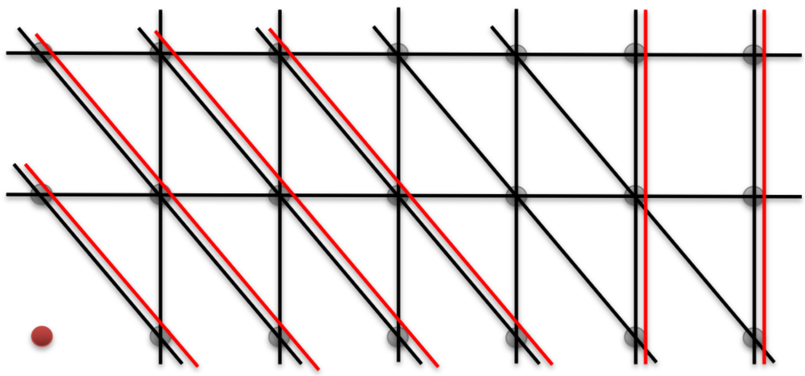

Figure 4.4: A 3-cover for the 7×3 grid. The red lines are the ones added on top of the previous 2-cover.

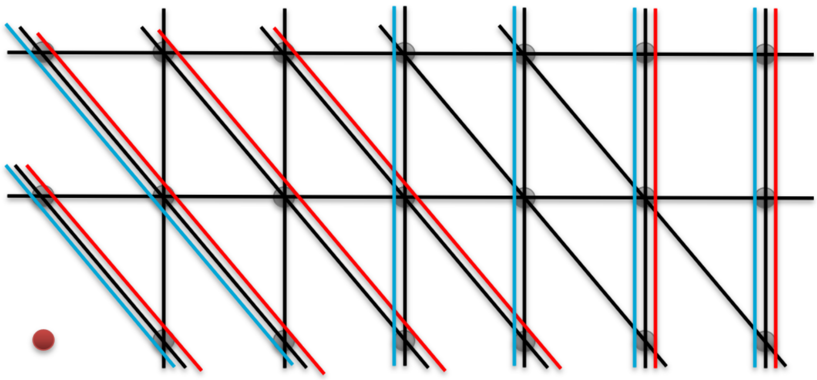

Figure 4.5: A 4-cover for the 7×3 grid. The blue lines are the ones added on top of the previous 3-cover.

I would like to remark that in the depicted grid, $k = 4$ is the largest multiplicity for which this construction holds. It's not hard to see why this holds.

While this construction might make the following conclusion seem rather straightforward, it by no means suffices as a formal proof. In the following theorem I will prove that this construction gives an optimal k-cover. I will denote the minimal cover size, i.e., the minimum number of lines required to cover every non origin point at least k times, by the function $f(n, m, k)$.

Theorem 4.4. *Let $S = \{0, \dots, n-1\} \times \{0, \dots, m-1\}$ and $n \geq m$. Then, for $n \geq (k-1)(m-1)+1$, it holds that $f(n, m, k) = k(n-1) + (m-1)$.*

Proof. The lower bound of this function follows from Theorem 2.1. To prove the upper bound, we will show that the construction I described earlier indeed creates a cover of size $k(n-1)+(m-1)$. The lines in this cover can be described as follows:

- 1 copy of $y = i$ for every $1 \leq i \leq m-1$;
- $k-t$ copies of $x = i$ for all $1 \leq t \leq k-1$ and $n - t(m-1) \leq i \leq n-(t-1)(m-1)-1$, and 1 copy of $x = i$ for each $1 \leq i \leq n-(k-1)(m-1)-1$;
- t copies of $x + y = i$ for each $1 \leq t \leq k-1$, $n - t(m-1) \leq i \leq n-(t-1)(m-1)-1$, and $k-1$ copies for $x + y = i$ for each $1 \leq i \leq n-(k-1)(m-1)-1$.

The total number of lines sums up to be $(m-1)+(m-1)\frac{k(k-1)}{2} + (n-(k-1)(m-1)-1) + (m-1)\frac{k(k-1)}{2} + (k-1)(n-(k-1)(m-1)-1)$, which simplifies to $k(n-1)+(m-1)$ which is indeed the desired size.

We can write $\{1, \dots, n-1\} = B_1 \cup \dots \cup B_{k-1} \cup B_k$ where for $1 \leq t \leq k-1$, $B_t = [n-t(m-1), n-(t-1)(m-1)-1] \cap \mathbb{Z}$ and $B_k = [1, n-(k-1)(m-1)-1] \cap \mathbb{Z}$. Then for any point (a, b) in the grid with no coordinate equal to 0 there is a unique t such that $(a, b) \in B_t \times \{1, \dots, m-1\}$. As this point is covered once by $y = b$, we just need to show that it is covered at least $k-1$ times by the vertical and the diagonal lines.

Let $i = a + b$. Note that $n - t(m-1) \leq a \leq n-(t-1)(m-1)-1$ and $1 \leq b \leq m-1$, which implies that $n - t(m-1) + 1 \leq i \leq n-(t-2)(m-1)-1$, i.e., $i \in B_t \cup B_{t-1}$.

If $t = k$, then $i \leq n-(k-2)(m-1)-1$ and hence the point is covered $k-1$ times by the line $x+y=i$. Therefore, assume that $t \leq k-1$, and now consider two cases $i \geq n-(t-1)(m-1)$ and $i \leq n-(t-1)(m-1)-1$. In the first case the point is covered $t-1$ times by $x+y=i$ and in the second it is covered by t times by such a line. As this point is also covered $k-t$ times by the line $x = a$, we see that the point is covered at least $k-1$ times by either vertical or diagonal lines.

Finally, we can handle the points of the form $(a, 0)$ and $(0, b)$. Points of the form $(0, b)$ are covered once by $y = i$ and $k-1$ times by $x+y = i$ for each $1 \leq i \leq (m-1) \leq n-(k-2)(m-1)-1$. (remember that $n \geq (k-1)(m-1)+1$ by assumption). For $1 \leq i \leq n-(k-1)(m-1)-1$, the points $(0, i)$ are covered once by $x = i$ and $k-1$ times by $x+y = i$. For $n-(k-1)(m-1) \leq i \leq n-1$, $(i, 0)$ is covered by $k-t$ copies of $x = i$ and t copies of $x + y = i$. Therefore, every point of the form $(a, 0)$ and $(0, b)$ is covered exactly k times, resulting in a k-cover. $\square$

Chapter 5

Grids in $\mathbb{R}^3$ and higher dimensions

5.1 Generating values

After seeing the success that followed from the generated values for grids in $\mathbb{R}^2$, I wondered if I would be able to adapt this program to the three-dimensional situation. The model required little alteration. Similarly to the two-dimensional model, I need to justify one of the assumptions I made.

Lemma 5.1. *There is always an optimal k-cover in which every plane contains at least three points of the grid.*

Proof. Assume that we have a k-cover of a grid. If a plane covers only one or two points, then you can replace it by a plane covering the same point(s) along with one or two extra ones. After replacement we still have a k-cover of the same size. □

(**Integer Programming Formulation**) Let $\mathcal{H}$ be the list of all planes in $\mathbb{R}^3$ that go through at least three points of an $n \times m \times l$ grid $G \subset \mathbb{R}^3$. For a point $\vec{v} \in G$, let $\chi^{\vec{v}}$ be the vector of length $|\mathcal{H}| = N$, whose i'th coordinate $\chi_i^{\vec{v}}$ is 1 if the i'th element of $\mathcal{H}$ contains $\vec{v}$ and 0 otherwise.

Let $w_1, \ldots, w_N \in \mathbb{R}$. We need to find out the smallest value of $\sum_{i=1}^{N} w_i$ subject to the following constraints:

- $w_i \in \mathbb{N}$ for all i.
- for every $\vec{v} \in G \setminus \{\vec{0}\}$, $\sum_{i=1}^{N} \chi_i^{\vec{v}} w_i \geq k$.
- $\sum_{i=1}^{N} \chi_i^{\vec{0}} w_i = 0$.

This code was programmed in the same way as the previous one, using Python and Gurobi. The main difference was having to generate all planes going through three or more points, instead of all lines going through at least two. Apart from that, the code is pretty much the same and can be found in appendix A. Generating all the planes in a large three-dimensional space, and then finding a minimal cover does require a lot of computing power, a resource that is (for me) somewhat limited. This caused me to be able to values using smaller dimensions than I did in the two-dimensional situation.

Due to limited computing power, I chose to only generate minimal cover sizes of grids where the third and smallest dimension was less or equal to four. There were two natural comparisons to be made. The first is to compare the data to the Ball-Serra lower bound. These pictures look similar to when the $n \times m$ grids were compared to the Ball-Serra bound. When n was large in comparison to m, l and k, the minimal cover sizes were equal to the bound. The more similar

m \ n	2	3	4	5	6	7
2	6	8	11	14	17	20
3		10	12	15	18	21
4			14	17	19	22
5				19	21	
6					23	

Figure 5.1: Minimal cover sizes for the grids $n \times m \times 2$ for $k = 3$. The yellow values are identical to the related $n \times m$ grid with $k = 3$, while the light green values are one larger than those.

m \ n	2	3	4	5	6	7
2	8	10	12	15	18	21
3		11	14	16	19	22
4			16	18	20	23
5				20	22	24
6					24	

Figure 5.2: Minimal cover sizes for the grids $n \times m \times 3$ for $k = 3$. The colours indicate the difference between the minimal cover size of this grid and the related $n \times m$ grid with $k = 3$. The difference is one for the lightest values, two for the middle values and three for the darkest values.

those values were to each other, the more the minimal cover size deviated from the Ball-Serra bound. These pictures can be found in appendix B. Another way of analysing this data is to compare it to the values of the $n \times m$ grids. That way, it can be seen by how much the cover size increases when the grid that will be covered is extended by one or more planes. Examples of this can be found in Figure 5.1 and Figure 5.2, which depict the values of the grids $n \times m \times l$ for $k = 3$ and l equal to 2 or 3, respectively. For $k = 2$ and $k = 4$, these tables can be found in appendix B.

From these pictures it looks like starting from certain small grid sizes, the minimal cover size of an $n \times m \times l$ grid increased by one, every time l increased by one. This started from $l = 1$, which is equivalent to the grid in $\mathbb{R}^2$.

To prove such a statement, I came up with a construction that would turn any cover of an $n \times m$ grid into a cover of an $n \times m \times l$ grid, including the necessary conditions under which a certain minimal cover size can be achieved. Analogous to the way I defined $f(n, m, k)$ in the previous chapter, I will define $f(n, m, l, k)$ to be the minimal number of planes required to cover every non origin point of an evenly spaced $n \times m \times l$ grid k times. I managed to prove the following result:

Theorem 5.2. *Let $S = \{0, \dots, n-1\} \times \{0, \dots, m-1\}$, $n \geq m$ and $w \leq k$. If a k-cover of S of size T can be partitioned into a w-cover and at least $w(l-1)$ additional lines, where l is a positive integer, then $f(n, m, l, k) \leq T + (k-w)(l-1)$.*

Before proving this, I will remark on how I formulated this theorem. The observation that inspired this theorem assumed $w = k - 1$, as in that case the bound is the following: $f(n, m, l, k) \leq T + (l - 1)$. This is also the best version of the theorem, as it increases the upper bound for the three-dimensional grid as little as possible. It is also the version that I will apply later, to prove equivalence under certain conditions.

Proof. Let $\mathcal{H} = \{H_i : i \in \{1, \dots, f(n, m, k)\}\}$ for $H_i = \{(x, y) \cdot \vec{a_i} = c_i\}$ be a k-cover of the grid $n \times m$. Define $\mathcal{K} \subseteq \mathcal{H}$ to be the largest multiset of lines in $\mathcal{H}$ such that $\mathcal{H} \setminus \mathcal{K}$ is a w-cover and $|\mathcal{K}| = w(l - 1)$. To create the k-cover of $n \times m \times l$, let's call this $\mathcal{G}$, first add $(k - w)$ copies of the planes $\{z = i : 1 < i \leq l\}$. Now, all points with z-coordinate not-equal to zero have been covered $(k - w)$ times. In $\mathbb{R}^3$, the multiset of lines $\mathcal{H}$ translates to a multiset of planes $\mathcal{H}_+ = \{H_i : i \in \{1, \dots, f(n, m, l, k)\}\}$ for $H_{i+} = \{(x, y, z) \cdot \vec{a_{i+}} = c_{i+}\}$. Where a line in $\mathcal{H}$ would cover a point (x_1, y_1), the corresponding plane in $\mathcal{H}_+$ covers all the points (x_1, y_1, z) for every value of z on the grid. We define $\mathcal{K}_+$ to be the planes corresponding in the same way to the lines in $\mathcal{K}$. Add the planes from $\mathcal{H}_+ \setminus \mathcal{K}_+$ to $\mathcal{G}$. By assumption, the multiset $\mathcal{H}_+ \setminus \mathcal{K}_+$ still forms a w-cover. Therefore all points such that $z \neq 0$ and $(x, y) \neq (0, 0)$ are covered k times. The points $(0, 0, z \neq 0)$ are still covered $(k - w)$ times.

To finish the cover, for each point $(0, 0, r)$, for $1 \leq r \leq l - 1$, we pick w lines from $\mathcal{K}$ and then add the w planes spanned by these lines and the point $(0, 0, r)$ to our multiset $\mathcal{G}$. This way, each of these points is now covered k times. This cover has size $T + (k - w)(l - 1)$, which implies that this is an upper bound.

□

While this result turns out to be rather strong in some cases, as shown by the following theorem, it doesn't mean that it's always the best possible bound. I noticed that for certain values of n and m, when $k \geq 3$ and $l = 2$, the minimal cover size for the grid $n \times m \times 2$ is equal to the minimal cover size for the grid $n \times m$. One example of this is the grid $4 \times 3 \times 2$ when $k = 3$, as shown in Figure 5.3. The minimal 3-cover contains 12 planes, which is the same amount as the 12 lines need to cover a 4×3 grid.

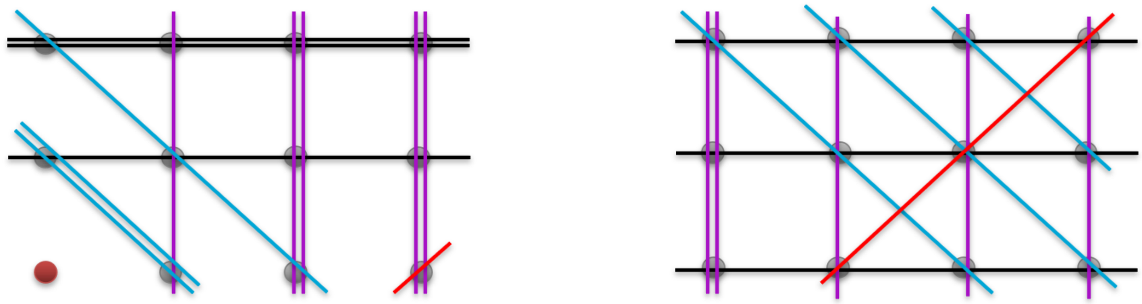

Figure 5.3: The left picture depicts the $(z = 0)$-plane while the right one depicts the $(z = 1)$-plane.

There are multiple ways to cover this grid, some of which, if not all, are represented by Figure 5.3. A line of any colour is a plane together with any other line of that same colour. This cover isn't a sole example, and from Figure 5.1, one might hypothesise that when some restrictions on the grid and multiplicity apply, such a construction can always be made. However, I wasn't able to replicate such a delicate configuration of lines and will leave this problem to an interested reader.

In the next theorem we combine the results from Theorem 5.2 and Theorem 4.4. It turns out that when the same restrictions apply to n,m and k as in Theorem 4.4, we know not only the cover size of the grid $n \times m$, but that of the $n \times m \times l$ grid as well.

Theorem 5.3. *Let $S = \{0, \ldots, n-1\} \times \{0, \ldots, m-1\} \times \{0, \ldots, l-1\}$, $n \geq m \geq l$ and $n \geq (k-1)(m-1)+1$. Then $f(n, m, l, k) = k(n-1) + (m-1) + (l-1)$.*

Proof. The lower bound follows from Theorem 2.1. To find the upper bound, remember that from Theorem 4.4 it follows that $f(n, m, 1, k) = k(n-1) + (m-1)$. We know that this cover can be split into a $(k-1)$-cover and $n-1$ separate lines. That's because the minimal cover size of a $(k-1)$-cover $f(n, m, 1, k-1) = (k-1)(n-1) + (m-1)$ is equal to $f(n, m, 1, k) + (n-1) = k(n-1) + (m-1) + (n-1)$, in addition to knowing what this $(k-1)$-cover might look like, according to Theorem 4.4 Thus, when $n \geq (k-1)(l-1)+1$, which always holds because $m \geq l$, we satisfied the assumptions from Theorem 5.2 for $w = k-1$. Applying this theorem and considering $f(n, m, 1, k) = k(n-1) + (m-1)$, it follows that $f(n, m, l, k) \leq k(n-1) + (m-1) + (l-1)$. Therefore the equality holds. □

5.2 Higher dimensions

After proving these last two theorems, I started to wonder if similar results would hold for higher dimensions. Especially the proof of Theorem 5.3 seemed to imply that to be possible, if we're able to prove a result similar to Theorem 5.2 which would give an upper bound for the cover size of any grid in $\mathbb{R}^{N+1}$ compared to the cover size of almost the same grid in $\mathbb{R}^N$. A generalisation of Theorem 5.2 was indeed possible, which is given and proven below. Generalizing Theorem 5.3 turned out to be much more complicated than expected and I didn't manage to prove it. This generalisation can be found in the conclusion as a conjecture.

Theorem 5.4. *Let w be an integer smaller than or equal to k and $n_1 \geq n_2 \geq \cdots \geq n_N$. If a k-cover of the grid $n_1 \times n_2 \times \cdots \times n_N$ in $\mathbb{R}^N$ of size T can be partitioned into a w-cover and at least $w(n_{N+1} - 1)$ other hyperplanes, then $f(n_1, \ldots, n_N, n_{N+1}, k) \leq T + (k-w)(n_{N+1} - 1)$.*

Proof. Let $\mathcal{H} = \{H_i : i \in \{1, \ldots, f(n_1, \ldots, n_N, k)\}\}$ for $H_i = \{(x_1, \ldots, x_N) \cdot \vec{a_i} = c_i\}$ be a k-cover of the grid $n_1 \times n_2 \times \cdots \times n_N$. Define $\mathcal{K} \subseteq \mathcal{H}$ to be the largest multiset of hyperplanes in $\mathcal{H}$ such that $\mathcal{H} \backslash \mathcal{K}$ is a w-cover and $|\mathcal{K}| = w(n_{N+1}-1)$. To create the k-cover of $n_1 \times n_2 \times \cdots \times n_N \times n_{N+1}$, let's call this $\mathcal{G}$, first add $(k-w)$ copies of the hyperplanes $\{x_{N+1} = i : 1 < i \leq n_{N+1}\}$. In $\mathbb{R}^{N+1}$, the multiset of hyperplanes $\mathcal{H}$ translates to a multiset of hyperplanes $\mathcal{H}_+ = \{H_i : i \in \{1, \ldots, f(n_1, \ldots, n_{N+1}, k)\}\}$ for $H_{i+} = \{(x_1, \ldots, x_{N+1}) \cdot \vec{a_{i+}} = c_{i+}\}$. Where a line in $\mathcal{H}$ would cover a point $(x_{1,1}, x_{2,1}, ..., x_{N,1})$, the corresponding plane in $\mathcal{H}_+$ covers all the points $(x_{1,1}, x_{2,1}, ..., x_{N,1}, x_{N+1})$ for every value of x_{N+1} on the grid. We define $\mathcal{K}_+$ to be the hyperplanes corresponding in the same way to the hyperplanes in $\mathcal{K}$. Now, all points with x_{N+1}-coordinate not equal to zero have been covered $(k-w)$ times. Add the hyperplanes from $\mathcal{H}_+ \setminus \mathcal{K}_+$ to $\mathcal{G}$. By assumption, the multiset $\mathcal{H}_+ \setminus \mathcal{K}_+$ still forms a w-cover. Therefore all points such that $x_{N+1} \neq 0$ and $(x_1, \ldots, x_N) \neq \{0\}^N$ are covered k times. The points $(0, \ldots, 0, x_{N+1} \neq 0)$ are still covered $(k-w)$ times.

To finish the cover, for each point $(0, 0, \ldots, r)$, for $1 \leq r \leq n_{N+1} - 1$, we pick w hyperplanes from $\mathcal{K}$ and then add the w hyperplanes spanned by these hyperplanes and the point $(0, 0, \ldots, r)$ to our multiset $\mathcal{G}$. This way, each of these points is now covered k times. This cover has size $T + (k-w)(n_{N+1} - 1)$, which implies that this is an upper bound. □

Chapter 6

Finite fields

In this chapter I will talk about grids in finite fields, as opposed to grids in the infinite field $\mathbb{R}$. I will refer to the minimal cover sizes in the following way. The minimum number of hyperplanes required to cover every non origin vertex on a hypercube $\mathbb{F}_q^n$ k times, while covering the origin exactly s times, will be denoted by $g_q(n, k; s)$. The function $f_q(n, k)$ is defined as $\min_{s<k} g_q(n, k; s)$, which just means the minimum number of hyperplanes required to cover every non origin point k times, while the origin may be covered up to $k - 1$ times.

6.1 Generalisation

In their paper, Bishnoi, Boyadzhiyska, Das and Mészáros [5] studied variants of the subspace covering problem over binary hypercubes. One of their main results was that for $k \geq 2^{n-2}$, the minimal cover size is exactly equal to $2k - \lfloor \frac{k}{2^{n-1}} \rfloor$. It turns out that this result can be generalized to $\mathbb{F}_q^n$ for any prime power q. Before I'm able to prove this, I need a different result. It's a very direct generalization of Lemma 2.1 from the same paper.

Lemma 6.1. *Let n, k, s be integers such that $n \geq 1$ and $k > s \geq 0$ and let q be a prime power. Then*

$$g_q(n, k; s) \geq qk - \lfloor \frac{k - s}{2^{n-1}} \rfloor.$$

In particular, $f_q(n, k) \geq qk - \lfloor \frac{k}{q^{n-1}} \rfloor$

Proof. Let $\mathcal{H}$ be an optimal $(k; s)$-cover of $\mathbb{F}_q^n$, so that we have $g(n, k; s) = |\mathcal{H}|$. We double-count the pairs $(\vec{x}, S)$ with $\vec{x} \in \mathbb{F}_q^n$. On the one hand, every affine subspace $S \in \mathcal{H}$ contains q^{n-1} points, and so there are $q^{n-1}|\mathcal{H}|$ such pairs. On the other hand, since every nonzero point is covered at least k times and the origin is covered s times, there are at least $(q - 1)k + s$ such pairs. Thus $(q - 1)k + s \leq q^{n-1}|\mathcal{H}|$, and the claimed lower bound follows from solving for $|\mathcal{H}|$ and observing that $g_q(n, k; s)$ is an integer. The bound on $f_q(n, k)$ is obtained by noticing that our lower bound on $g_q(n, k; s)$ is increasing in s, which implies that $g_q(n, k; s) \geq qk - \lfloor \frac{k}{q^{n-1}} \rfloor$ for all s. As $f_q(n, k) = \min_s g_q(n, k, s)$, we get the same lower bound on $f_q(n, k)$. □

Using this result, it's possible to prove the following theorem.

Theorem 6.2. *Let $k \geq 1$, $n \geq 1$ and q a prime power. If $k \geq q^{n-1} - q^{n-2}$, then $f_q(n, k) = g_q(n, k; 0) = qk - \lfloor \frac{k}{q^{n-1}} \rfloor$.*

Proof. The lower bound follows from Lemma 6.1.

For the upper bound, first take $k = q^{n-1} - q^{n-2}$. We will show that the family of hyperplanes $\mathcal{H}_0 = \{H_{\vec{u}} : \vec{u} \in \mathbb{F}_q^n, u_n \neq 0\}$ where we define $H_{\vec{u}} = \{\vec{x} : \vec{x} \cdot \vec{u} = 1\}$, forms a k-cover. Note that $|\mathcal{H}_0| = q^n - q^{n-1} = kq = qk - \lfloor \frac{k}{q^{n-1}} \rfloor$ where none of those hyperplanes cover the origin. Let $\vec{x'}, \vec{u'} \in \mathbb{F}_q^{n-1}$, $x \in \mathbb{F}_q$ and $u \in \mathbb{F}_q^n \setminus \{\vec{0}\}$. Take $\vec{x} = (\vec{x'}, x)$ and $\vec{u} = (\vec{u'}, u)$. We have that $\vec{x} \cdot \vec{u} = 1$ iff $\vec{x'} \cdot \vec{u'} = 1 - xu$. If $\vec{x'} \neq \vec{0}$, there are q^{n-2} possible solutions to this for each of the $q - 1$ values of u, giving us a total of $(q-1)q^{n-2}$ hyperplanes from $\mathcal{H}_0$ that are covering $\vec{(x)}$. If $\vec{x'} = \vec{0}$ and thus $x \neq 0$, there exists exactly one value of u so that the equation $ux = 1$ is satisfied. As we're free to choose each of the other $n - 1$ coordinates freely, we get that $H_{\vec{u}}$ covers $\vec{x}$ for q^{n-1} different values of $\vec{u}$. Therefore every point is covered $q^{n-1} - q^{n-2}$ times, thus making a $(q^{n-1} - q^{n-2})$-cover of size $|\mathcal{H}_0| = qk - \lfloor \frac{k}{q^{n-1}} \rfloor$

Now take $q^{n-1} - q^{n-2} \leq k < q^{n-1}$. Add an arbitrary choice of $k - (q^{n-1} - q^{n-2})$ q-tuples of parallel hyperplanes. Now we have $q^n - q^{n-1} + q(k - (q^{n-1} - q^{n-2})) = qk = qk - \lfloor \frac{k}{q^{n-1}} \rfloor$ elements where every nonzero point is covered k times and the origin is covered $k - q^{n-2} < k$ times.

Finally take $k \geq q^{n-1}$. We can then write $k = aq^{n-1} + b$ for some $a \geq 1$ and $0 \leq b < q^{n-1}$. Let $\mathcal{H}_1 = \{H_{\vec{u}} : \vec{u} \in \mathbb{F}_q^n \setminus \{\vec{0}\}\}$ be the set of all affine hyperplanes avoiding the origin, of which there are $q^n - 1$. Moreover, for each nonzero $\vec{x}$, there are exactly q^{n-1} vectors $\vec{u}$ s.t. $\vec{x} \cdot \vec{u} = 1$. So, each such point is covered over q^{n-1} times by the hyperplanes in $\mathcal{H}_1$. Let $\mathcal{H}$ be the multiset of hyperplanes obtained by taking a copies of $\mathcal{H}_0$ and appending an arbitrary choice of b q-tuples of parallel hyperplanes. Each nonzero point is then covered at least $aq^{n-1} + b = k$ times, the origin only by $b < q^{n-1} < k$ times. So, $\mathcal{H}$ is a k-cover. □

6.2 Generating values

Just like with the variants over the field $\mathbb{R}$, finding the minimal value of a cover is no easy task. In their paper, Bishnoi, Boyadzhiyska, Das and Mészáros also used a model to generate values for minimal cover sizes for grids over $\mathbb{F}_2^n$, complementing their own results. As they managed to explain and prove many of these values, the most interesting thing for me was to attempt to generalize this model for larger fields $\mathbb{F}_q^n$. This resulted in the following model formulation:

(**Integer Programming Formulation**) Let $\mathcal{H}$ be the list of all hyperplanes in $\mathbb{F}_q^n$. For a point $\vec{v} \in \mathbb{F}_q^n$, let $\chi^{\vec{v}}$ be the q-ary vector of length $|\mathcal{H}| = q(q^n - 1)/(q - 1) = N$, whose i'th coordinate $\chi_i^{\vec{v}}$ is 1 if the i'th element of $\mathcal{H}$ contains $\vec{v}$ and 0 otherwise.

Let $w_1, \ldots, w_N \in \mathbb{R}$. We need to find out the smallest value of $\sum_{i=1}^N w_i$ subject to the following constraints:

- $w_i \in \mathbb{N}$ for all i.
- for every $\vec{v} \in \mathbb{F}_q^n \setminus \{\vec{0}\}$, $\sum_{i=1}^N \chi_i^{\vec{v}} w_i \geq k$.
- $\sum_{i=1}^N \chi_i^{\vec{o}} w_i \leq k - 1$.

To program this model, instead of Python, I used Sagemath [19] together with Gurobi. I made this choice because Sagemath supports finite prime fields, while Python does not allow me to do calculations over those fields in an intuitive way. The code I used can be found in appendix A. This code works very similarly to the ones over $\mathbb{R}$. The main difference is just that the available tools in Sagemath make it very easy to find the subspaces required for covering. Just like the version of the model for grids in $\mathbb{R}^3$, running this code demands a lot of computing power. Because of that, I have not been able to generate as many values as I would have liked.

The amount of data I was able to generate was a little disappointing. Especially as the number of dimensions increased, it quickly became very difficult for the computer to finish calculating the minimal cover. This can be seen very clearly in Figure 6.1, as these include the highest values I was reasonably able to calculate.

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	4	6	8	11	14	16	19	22	24	27	30	32	35	38
3	6	8	10	13	16	18	21	24	26	29	32	35	38	41
4	8	10	12	15	18	20	23	26	29	31	34	37	40	43
5	10	12	14	17										
6	12													

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	6	9	12	15	19	23	27	30	34	38	42	45	49	53
3	9	12	15	18	22	26	30	34	37	41	45	48	52	56
4	12	15												

n \ k	1	2	3	4	5	6	7	8
2	8	12	16	20	24	29	34	39
3	12	16	20	24	28	33	38	

n \ k	1	2	3	4	5	6	7	8
2	12	18	24	30	36	42	48	55

Figure 6.1: Minimal cover sizes for subspaces $\mathbb{F}_q^n$. From top to bottom, $q = 3$, $q = 4$, $q = 5$, $q = 7$. The coloured values are the ones to which Theorem 6.1 can be applied.

In this table, I coloured the values to which Theorem 6.1 could be applied. This is, sadly, the only pattern I was able to find in this data. The cover sizes for grids in $\mathbb{F}_q^n$ that do not satisfy the conditions for Theorem 6.1, do not appear to increase regularly when k increases. When considering a hyperspace $\mathbb{F}_q^n$ for k fixed, one thing stood out to me is that often, when n increased by 1, the minimal cover size increased by $q - 1$. But, due to a few seemingly arbitrary exceptions, I have not been able to come up with any arguments explaining this behaviour.

Chapter 7

Conclusion

In my thesis I investigated multiple variants of the subspace covering problem. The first variant I really focused on was trying to find the minimal amount of lines or (hyper)planes needed to cover every nonzero point of a grid $S = \{0, \dots, n_1 - 1\} \times \{0, \dots, n_2 - 1\} \times \cdots \times \{0, \dots, n_N - 1\}$ over $\mathbb{R}^N$ at least k times, while leaving the origin uncovered. When $N = 2$ and $n_2 = 2$, I managed to determine the cover size exactly in Theorem 4.1. I also showed that for any dimension N, that when n_1 is large with respect to k and n_2 up to n_N, the minimal cover size is bound from above by the cover size of a related $(N-1)$-dimensional grid plus a constant. In the other variant I studied, I tried to find the minimal amount of hyperplanes needed to cover all nonzero points of $\mathbb{F}_q^n$ at least k times, while the origin may be covered up to $k-1$ times. I was able to prove that when k was large with respect to n, the answer could be determined exactly. To finish this thesis, I will give some conjectures. They seem to be suggested by the data I generated, but I wasn't yet able to prove that they are true.

Improving Ball-Serra

In the beginning of chapter 4 I proved that for a $n \times 2$ grid in $\mathbb{R}^2$, the minimal cover size was equal to $k(n-1) + 1 + \lfloor \frac{k-1}{n} \rfloor$. On these grids, for most values of k, the cover size is higher than the lower bound given by Ball-Serra, which here is equal to $1 + k(n-1)$. This conclusion prompted me to try to find specific formulations to quantify this difference in a broader collection of grids. Looking at the cover sizes of $n \times 3$ grids, I formed the following hypothesis from the data I generated:

Conjecture 7.1. *Let $n \geq 5$. The minimum number of lines you need to cover a $n \times 3$ grid k times when missing any point is equal to $k(n-1) + 2 + \lfloor \frac{2k}{n+3} \rfloor + \lfloor \frac{k}{n+1} \rfloor + \lfloor \frac{k}{n+2} \rfloor$*

Just like the situation with $n \times 2$ grids, for k high enough, if this conjecture were to be true, this minimal cover size is higher than the Ball-Serra bound of $k(n-1) + 2$.

Looking back at Figure 4.2, it appears as if all the covers that satisfy the Ball-Serra lower bound are taken care of by Theorem 4.4. There is however much more that can be learned from that data. We saw that for $n \geq (k-1)(m-1) + 1$, the Ball-Serra lower bound holds as equality. We might also notice that for the values $\frac{1}{2}(k-1)(m-1) + 1 \leq n < (k-1)(m-1) + 1$, the minimal cover size increases when compared to the Ball-Serra lower bound in a regular pattern. The minimal cover size increases exactly by one every time m increases by one, when m satisfies the previously given inequalities. If the lower bound of the following conjecture were to be proven, then that would be a significant improvement for the lower bound when considering larger grids.

Conjecture 7.2. *Let* $S = \{0, \dots, n-1\} \times \{0, \dots, m-1\}$ *and* $n \geq m$ *Then, for* $(k-1)\lfloor \frac{1}{2}(m-1) \rfloor + 1 \leq n < (k-1)(m-1)+1$,

$$f(n,m,k) = k(n-1) + (m-1) + m - \left\lfloor \frac{n-1}{k-1} + 1 \right\rfloor.$$

Comparing to different versions in $\mathbb{R}^2$

As I mentioned after introducing my code in chapter 4, I also generated data for grids which were spaced quadratically (meaning that the grids were of the type $(0, 1^2, 2^2, \dots) \times (0, 1^2, 2^2, \dots) \times \cdots$, and grids where a side point was missing. These can be found in appendix B. Comparing these results to the evenly spaced grids where the origin was missing proved rather insightful. All of these grids had the same minimal cover size when $n \geq (k-1)(m-1)+1$, and for the grids missing the side point this held for even larger n. This made me believe there should be a construction that covers every grid in the same amount of lines as Theorem 4.4 shows. This can be rephrased as follows in the following conjectures:

Conjecture 7.3. *For any* $n \times m$ *grid in* $\mathbb{R}^2$, *when* $n \geq (k-1)(m-1)+1$ $f(n,m,k) = k(n-1) + (m-1)$.

Conjecture 7.4. *Let* $S = \{0, \dots, n-1\} \times \{0, \dots, m-1\}$, $n \geq m$ *and let* $f(n,m,k)$ *be the minimum number of lines required to cover every point except for a point* $p = (0,j)$ *or* $p = (i,0)$ *for* $0 < i < n$ *and* $0 < j < m$ *at least* k *times while missing* p. *Then, for* $n \geq (k-1)(m-1)$ *for* $m > 1$ *and* $n \geq (k-1)$ *when m=1,* $f(n,m,k) = k(n-1) + (m-1)$.

Generalizing to higher dimensions

In chapter 5 I managed to prove the minimal cover size of three-dimensional grids, and found an upper bound for grids of any dimension, each when specific conditions are met. Trying to prove the minimal cover size for grids of any dimension was however a little too ambitious. The following conjecture is exactly what I was trying to prove.

Conjecture 7.5. *Let* $n_1 \geq n_2 \geq \cdots \geq n_{N+1}$ *and* $n_1 \geq (k-1)(n_2-1)+1$. *Then* $f(n_1, \dots, n_{N+1}, k) = k(n_1-1) + \sum_{i=2}^{N+1}(n_i - 1)$.

I do think this conjecture is true, but the proof might need a different approach. Directly proving the upper bound, in the same way as I did in Theorem 4.4 and even using a similar construction, might be more successful.

Finite field version

In chapter 6 I compared the generated data to the results of Theorem 6.1. There is one more comparison that can be made. In Figure 7.1, I compared this data to the Ball-Serra bound discussed earlier. One should note however that this bound is a lower bound that holds when the origin is not allowed to be covered at all. Because in this variation of the problem I did allow the origin to be covered up to $k-1$ times, the comparison should be considered with care.

What stands out to me are the values where this lower bound is equal to the generated result. The minimal cover sizes when the origin is left uncovered form an upper bound for minimal cover sizes when the origin may be covered up to $k-1$ times. This implies that the covers for these subspaces are optimal when the origin is not covered. This can be summarized by the following conjecture:

Conjecture 7.6. *If* $k \leq q$, *then* $f_q(n,k) = g_q(n,k;0)$.

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	4	6	8	11	14	16	19	22	24	27	30	32	35	38
3	6	8	10	13	16	18	21	24	26	29	32	35	38	41
4	8	10	12	15	18	20	23	26	29	31	34	37	40	43
5	10	12	14	17										
6	12													

0
1
2
3
4
5
6
7
8
9

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	6	9	12	15	19	23	27	30	34	38	42	45	49	53
3	9	12	15	18	22	26	30	34	37	41	45	48	52	56
4	12	15												

n \ k	1	2	3	4	5	6	7	8
2	8	12	16	20	24	29	34	39
3	12	16	20	24	28	33	38	

n \ k	1	2	3	4	5	6	7	8
2	12	18	24	30	36	42	48	55

Figure 7.1: Minimal cover sizes for subspaces $\mathbb{F}_q^n$. From top to bottom, $q = 3$, $q = 4$, $q = 5$, $q = 7$. The colours show the difference between the Ball-Serra lower bound for each grid

Appendices

Appendix A

Code

A.1 Grids in $\mathbb{R}^2$

```
import numpy as np
import gurobipy as gp
from gurobipy import GRB

## FUNCTIONS TO DETERMINE LINES

# Returns a collection of all vertical and horizontal lines in X x Y,
# where each line ax+by=c is represented by [a,b,c]
def hvlines(X,Y):
    linecoords = [0,0,0]
    total = np.array([])
    for i in range(len(X)):
        linecoords[0] = 1
        linecoords[1] = 0
        linecoords[2] = X[i]
        total = np.append(total, linecoords)
    for j in range(len(Y)):
        linecoords[0] = 0
        linecoords[1] = 1
        linecoords[2] = Y[j]
        total = np.append(total,linecoords)
    total2 = [[0,0,0] for _ in range(len(X)+len(Y))]
    for k in range(len(X)+len(Y)):
        total2[k] = total[3*k:3*k+3]
    return total2

# Returns the line through point a and point b
def linethr(a,b):
    linecoords=[0,0,0]
    if (a[1]< b[1] and a[0]<b[0]) or(a[1]> b[1] and a[0]>b[0]):
    #if (a_y < b_y and a_x < b_x) or (a_y > b_y and a_x > b_x), the line is increasing.
        linecoords[1] = 1
        linecoords[0] = -abs(a[1]-b[1])*(abs(a[0]-b[0]))**(-1)
        linecoords[2] = min(a[1],b[1])-- min(a[0],b[0])*linecoords[0]
```

```
        return linecoords
    if (a[1]< b[1] and a[0]>b[0]) or (a[1]> b[1] and a[0]<b[0]):
    #if (a_y < b_y and a_x > b_x) or (a_y > b_y and a_x < b_x), the line is decreasing.
        linecoords[1] = 1
        linecoords[0] = abs(a[1]-b[1])*(abs(a[0]-b[0]))**(-1)
        linecoords[2] =  max(a[1],b[1])+ min(a[0],b[0])*linecoords[0]
        return linecoords

# Returns a list containing all horizontal, vertical, increasing and decreasing
# lines through every point in the grid X x Y
def createlines(X,Y): #Where X and Y are the n resp. m elements of the grid
    total = hvlines(X,Y)
    lines =  set(tuple(i) for i in total)
    for i in range(len(X)):
        for j in range(len(Y)):
            for k in range(len(X)):
                for l in range(len(Y)):
                    if X[i]!=X[k] and Y[j]!=Y[l]:
                        lines.add(tuple(linethr([X[i],Y[j]],[X[k],Y[l]])))
    return list(lines)


## DETERMINE AND CREATE GRID

n = int(input("Enter n: "))
m = int(input("Enter m: "))
k = int(input("Enter k: "))

# Make a linear grid
X = list(np.linspace(0,n-1,n))
Y = list(np.linspace(0,m-1,m))

# Create a list of all lines going through at least two points in the grid
Lines = createlines(X,Y)

# Create a list of all points in the grid
pts = []
N = n*m
M = len(Lines)
for Xi in X:
    for Yj in Y:
        pts.append([Xi,Yj])

# Which point do we want to avoid?
# Default is (0,0) but can be altered.
MPx = 0
MPy = 0
if not (MPx == MPy  and MPx == 0):
    pts[0],pts[n*MPx + MPy] = pts[n*MPx + MPy], pts[0]
```

```
#
def ksi(i,j): #i is the i'th point and j in the j'th line
    if pts[i][0]*Lines[j][0] + pts[i][1]*Lines[j][1] == Lines[j][2]:
        return 1
    else:
        return 0

## MODEL
p = gp.Model("R2 lines")
w = p.addMVar(shape = M, vtype = GRB.INTEGER, name="w")

    ## Set objective
obj = np.array([1]*M)
p.setObjective(obj @ w, GRB.MINIMIZE)

    ## First constraint
for i in range(0,M):
    p.addConstr(w[i] >= 0, "Constraint 1")

    ## Second constraint
j = 0
obj2 = np.zeros((N-1,M))
for l in range(1,N):
    for i in range(M):
        obj2[j,i] = ksi(l,i)
    j += 1

for j in range(0,N-1):
    p.addConstr(obj2[j] @ w >= k, "Constraint 2")

    ## Third constraint
obj3 = np.array([1]*M)
for i in range(0,M):
    obj3[i] = ksi(0,i)

p.addConstr(obj3 @ w == 0, "Constraint 3")


    ## Solve
p.optimize()
for v in p.getVars():
    print('%s %g' % (v.varName, v.x))
    print("k = ", k, "\n\n")
    print('Obj: %g' % p.objVal)


for i in range(0,len(Lines)):
    print(i, Lines[i],"\n")
```

A.2 Grids in $\mathbb{R}^3$

```
import numpy as np
import gurobipy as gp
from gurobipy import GRB

## FUNCTIONS TO DETERMINE PLANES

# Returns a collection of all vertical and horizontal lines in X x Y x Z,
# where each line ax+by+dz=c is represented by [a,b,d,c]
def hvlines(X,Y,Z):
    linecoords = [0,0,0,0]
    total = np.array([])
    for i in range(len(X)):
        linecoords[0] = 1
        linecoords[1] = 0
        linecoords[2] = 0
        linecoords[3] = X[i]
        total = np.append(total, linecoords)
    for j in range(len(Y)):
        linecoords[0] = 0
        linecoords[1] = 1
        linecoords[2] = 0
        linecoords[3] = Y[j]
        total = np.append(total,linecoords)
    for k in range(len(Z)):
        linecoords[0] = 0
        linecoords[1] = 0
        linecoords[2] = 1
        linecoords[3] = Z[k]
        total = np.append(total,linecoords)
    total2 = [[0,0,0,0] for _ in range(len(X)+len(Y)+len(Z))]
    for i in range(len(X)+len(Y)+len(Z)):
        total2[i] = total[4*i:4*i+4]
    return total2

# Returns the plane going through the entered points
def planethr(p_0,p_1,p_2):
    planecoords = [0,0,0,0] #x,y,z,c
    p_01 = [p_0[0]-p_1[0], p_0[1]-p_1[1], p_0[2]-p_1[2]]
    p_02 = [p_0[0]-p_2[0], p_0[1]-p_2[1], p_0[2]-p_2[2]]
    cross =  np.cross(p_01,p_02)
    planecoords[0] = cross[0]
    planecoords[1] = cross[1]
    planecoords[2] = cross[2]
    planecoords[3] = p_0[0]*cross[0] + p_0[1]*cross[1] + p_0[2]*cross[2]
    return planecoords
```

```
# Returns a list containing all lines through every point in the grid X x Y x
↪  Z
def createplanes(X,Y,Z): #Where X, Y and Z are the n resp. m resp. l elements
↪  of the grid
    planes =  set() #tuple(i) for i in total)
    for i in range(len(X)):
        for j in range(len(Y)):
            for o in range(len(Z)):
                for k in range(len(X)):
                    for l in range(len(Y)):
                        for p in range(len(Z)):
                            for q in range(len(X)):
                                for r in range(len(Y)):
                                    for w in range(len(Z)):
                                        if
                                        ↪  [X[i],Y[j],Z[o]]!=[X[k],Y[l],Z[p]]!=[X[q],Y[r
                                            plane = tuple(planethr(
                                            ↪  [X[i],Y[j],Z[o]],[X[k],Y[l],Z[p]],[X[q],Y
                                            if plane[3]!=0:
                                                planes.add(plane)
    Planes = list(planes)
    # Gets rid of doubles, as -ax-by-dz=-c is equal to ax+by+dz+c
    count = 0
    while count < len(Planes):
        i = 0
        while i < len(Planes):
            if (Planes[count][0]==-Planes[i][0] and
            ↪  Planes[count][1]==-Planes[i][1] and
            ↪  Planes[count][2]==-Planes[i][2] and
            ↪  Planes[count][3]==-Planes[i][3]):
                Planes.pop(i)
                i = len(Planes)
            i += 1
        count += 1
    return Planes

## DETERMINE AND CREATE GRID

n = int(input("Enter n: "))
m = int(input("Enter m: "))
l = int(input("Enter l: "))
k = int(input("Enter k: "))

# make a nice grid
X = list(np.linspace(0,n-1,n))
Y = list(np.linspace(0,m-1,m))
Z = list(np.linspace(0,l-1,l))

Lines = createplanes(X,Y,Z)
```

```
pts = []
N = n*m*l
M = len(Lines)
for Xi in X:
    for Yj in Y:
        for Zk in Z:
            pts.append([Xi,Yj,Zk])


def ksi(i,j): #i is the i'th point and j in the j'th plane
    if pts[i][0]*Lines[j][0] + pts[i][1]*Lines[j][1] + pts[i][2]*Lines[j][2] ==
    ↪  Lines[j][3]:
        return 1
    else:
        return 0

## MODEL
p = gp.Model("R2 lines")
w = p.addMVar(shape = M, vtype = GRB.INTEGER, name="w")

    ## Set objective
obj = np.array([1]*M)
p.setObjective(obj @ w, GRB.MINIMIZE)

    ## First constraint
for i in range(0,M):
    p.addConstr(w[i] >= 0, "Constraint 1")

    ## Second constraint
j = 0
obj2 = np.zeros((N-1,M))
for l in range(1,N):
    for i in range(M):
        obj2[j,i] = ksi(l,i)
    j += 1

for j in range(0,N-1):
    p.addConstr(obj2[j] @ w >= k, "Constraint 2")

    ## Third constraint
obj3 = np.array([1]*M)
for i in range(0,M):
    obj3[i] = ksi(0,i)

p.addConstr(obj3 @ w == 0, "Constraint 3")


    ## Solve
p.optimize()
```

```
for v in p.getVars():
   print('%s %g' % (v.varName, v.x))
print("k = ", k, "\n\n")
print('Obj: %g' % p.objVal)


for i in range(0,len(Lines)):
    print(i, Lines[i],"\n")
```

A.3 $\mathbb{F}_q^n$

```
import numpy as np

q = 3
n = 4

## CREATE FIELD
V = VectorSpace(GF(q),n)
K = np.array(V)
K = np.delete(K,0,0)
H2 = K

V = VectorSpace(GF(q),n+1)
K = []
Q = q*sum(q^i for i in range(0,n))
for U in V.subspaces(1):
    K.append(list(U)[1])

# We go through all the one-dimensional subspaces and then
# pick one nonzero vector from each one of them.

H=K
for x in H:
    if all(x[i] == 0 for i in range(len(x) - 1)):
        H.remove(x)

# We go through all the one-dimensional subspaces and then
# pick one nonzero vector from each one of them.


def ksi(v,i): #where v and i are as defined in the function ksi in the model
    hp = list(H[i])
    c = hp[-1]
    hp.pop()
    hp = np.array(hp)
    v = np.array(v)
    if np.dot(hp,v) == c:
        return 1
    else:
```

```
        return 0

## MODEL

def gurobiloop(k_min,k_max):
    for k in range(k_min,k_max+1):
        p = MixedIntegerLinearProgram(solver = 'GLPK', maximization = False)
        w = p.new_variable(integer=True)

        ## Set objective
        p.set_objective(p.sum(w[i] for i in range(0,Q)))

        ## First constraint
        for i in range(0,Q):
            p.add_constraint(w[i] >= 0)

        ## Second constraint
        for v in H2:
            p.add_constraint(p.sum(w[i]*ksi(v,i)
                for i in range(0,Q)) >= k)

        ## Third constraint
        p.add_constraint(p.sum(w[i]*ksi([0]*n,i)
            for i in range(0,Q)) <= k-1)

        ## Write
        for v in p.getVars():
            print('%s %g' % (v.varName, v.x))
        temp = r"C:\Users\ymden\Desktop\LPform\LP" + str(k) + ".lp"
        p.write_lp(temp)

gurobiloop(1,14)
```

Appendix B

Generated Values

B.1 $\mathbb{R}^2$

m \ n	2	3	4	5	6	7	8	9	10
2	3	5	7	9	11	13	15	17	19
3		6	8	10	12	14	16	18	20
4			9	11	13	15	17	19	21
5				12	14	16	18	20	22
6					15	17	19	21	23
7						18	20	22	24
8							21	23	25
9								24	26
10									27

0
1
2
3
4
5
6
7
8
9

m \ n	2	3	4	5	6	7	8	9	10	11	12
2	5	7	10	13	16	19	22	25	28	31	34
3		9	12	14	17	20	23	26	29	32	35
4			14	16	19	21	24	27	30	33	36
5				18	21	23	26	28	31	34	37
6					23	25	28	30	33	35	38
7						27	30	32	35	37	40
8							32	34	37	39	42
9								36	39	41	44
10									41	43	46

m \ n	2	3	4	5	6	7	8	9	10	11	12
2	6	10	13	17	21	25	29	33	37	41	45
3		12	15	19	23	26	30	34	38	42	46
4			18	21	25	28	32	36	39	43	47
5				24	27	30	34	38	41	45	49
6					30	33	36	40	43	47	51
7						36	39	42	45	49	53
8							42	45	48	51	55
9								48	51	54	57
10									54	57	60

Figure B.1: Minimal cover sizes for the grids $n \times m$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the Ball-Serra lower bound for each grid.

B.2 Variations on $\mathbb{R}^2$

m \ n	2	3	4	5	6	7	8	9	10
2	3	5	7	9	11	13	15	17	19
3		6	8	10	12	14	16	18	20
4			9	11	13	15	17	19	21
5				12	14	16	18	20	22
6					15	17	19	21	23
7						18	20	22	24
8							21	23	25
9								24	26
10									27

0
1
2
3
4
5
6
7
8
9

m \ n	2	3	4	5	6	7	8	9	10
2	5	7	10	13	16	19	22	25	28
3		9	11	14	17	20	23	26	29
4			13	16	18	21	24	27	30
5				18	20	23	25	28	31
6					23	25	27	30	32
7						27	29	32	34
8							32	34	37
9								36	39
10									41

m \ n	2	3	4	5	6	7	8	9	10
2	6	10	13	17	21	25	29	33	37
3		12	15	19	22	26	30	34	38
4			18	21	24	28	32	35	39
5				24	27	31	34	37	41
6					30	33	37	40	43
7						36	39	42	45
8							42	45	48
9								48	51
10									54

Figure B.2: Minimal cover sizes for the grids $n \times m$ when instead of the origin, the point $(0, 1)$ is missing. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the Ball-Serra lower bound for each grid.

m \ n	2	3	4	5	6	7	8	9	10
2	3	5	7	9	11	13	15	17	19
3		6	8	10	12	14	16	18	20
4			9	11	13	15	17	19	21
5				12	14	16	18	20	22
6					15	17	19	21	23
7						18	20	22	24
8							21	23	25
9								24	26
10									27

0
1
2
3
4
5
6
7
8
9

m \ n	2	3	4	5	6	7	8	9	10
2	5	7	10	13	16	19	22	25	28
3		10	12	14	17	20	23	26	29
4			15	17	19	21	24	27	30
5				20	22	24	26	28	31
6					24	27	29	31	33
7						29	32	34	36
8							34	37	39
9								39	42
10									44

m \ n	2	3	4	5	6	7	8	9	10
2	6	10	13	17	21	25	29	33	37
3		12	15	19	22	26	30	34	38
4			18	21	24	28	32	35	39
5				24	27	31	34	37	41
6					30	33	37	40	43
7						36	39	42	45
8							42	45	48
9								48	51
10									54

Figure B.3: Minimal cover sizes for the quadratically spaced grids $(1, 2^2, \ldots, n^2) \times (1, 2^2, \ldots, m^2)$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the Ball-Serra lower bound for each grid.

B.3 $\mathbb{R}^3$

B.3.1 Compared to Ball-Serra bound

m \ n	2	3	4	5	6	7
2	4	6	8	10	12	14
3		7	9	11	13	15
4			10	12	14	16
5				13	15	
6					16	

0
1
2
3
4
5
6

m \ n	2	3	4	5	6	7
2	6	8	11	14	17	20
3		10	12	15	18	21
4			14	17	19	22
5				19	21	
6					23	

m \ n	2	3	4	5	6	7
2	8	11	14	18	22	26
3		13	16	19	23	27
4			19	22	25	28
5				24	28	
6					30	

Figure B.4: Minimal cover sizes for the grids $n \times m \times 2$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the Ball-Serra lower bound for each grid.

B.3.2 Compared to $\mathbb{R}^2$

m \ n	2	3	4	5	6	7
2	6	7	9	11	13	15
3		8	10	12	14	16
4			11	13	15	17
5				14	16	18
6					17	

0
1
2
3
4
5
6

m \ n	2	3	4	5	6	7
2	8	10	12	15	18	21
3		11	14	16	19	22
4			16	18	20	23
5				20	22	24
6					24	

m \ n	2	3	4	5	6	7
2	11	13	16	19	23	27
3		15	18	21	25	28
4			20	23	27	30
5				26	29	32
6					31	

Figure B.5: Minimal cover sizes for the grids $n \times m \times 3$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the Ball-Serra lower bound for each grid.

m \ n	2	3	4	5	6	7
2	8	9	10	12	14	16
3		10	11	13	15	17
4			12	14	16	
5				15	17	

0
1
2
3
4
5
6

m \ n	2	3	4	5	6	7
2	11	12	14	17	19	22
3		14	16	18	20	23
4			17	19	22	
5				21	24	

m \ n	2	3	4	5	6	7
2	14	16	19	22	25	28
3		18	20	23	27	30
4			23	25	28	
5				28	30	

Figure B.6: Minimal cover sizes for the grids $n \times m \times 4$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the Ball-Serra lower bound for each grid.

m \ n	2	3	4	5	6	7
2	4	6	8	10	12	14
3		7	9	11	13	15
4			10	12	14	16
5				13	15	
6					16	

0
1
2
3
4
5

m \ n	2	3	4	5	6	7
2	6	8	11	14	17	20
3		10	12	15	18	21
4			14	17	19	22
5				19	21	
6					23	

m \ n	2	3	4	5	6	7
2	8	11	14	18	22	26
3		13	16	19	23	27
4			19	22	25	28
5				24	28	
6					30	

Figure B.7: Minimal cover sizes for the grids $n \times m \times 2$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the minimal cover size of the $n \times m$ grid.

m \ n	2	3	4	5	6	7
2	6	7	9	11	13	15
3		8	10	12	14	16
4			11	13	15	17
5				14	16	18
6					17	

0
1
2
3
4
5

m \ n	2	3	4	5	6	7
2	8	10	12	15	18	21
3		11	14	16	19	22
4			16	18	20	23
5				20	22	24
6					24	

m \ n	2	3	4	5	6	7
2	11	13	16	19	23	27
3		15	18	21	25	28
4			20	23	27	30
5				26	29	32
6					31	

Figure B.8: Minimal cover sizes for the grids $n \times m \times 3$. From top to bottom, $k = 2$, $k = 3$, $k = 4$. The colours show the difference between the minimal cover size of the $n \times m$ grid.

B.4 $\mathbb{F}_q^n$

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	4	6	8	11	14	16	19	22	24	27	30	32	35	38
3	6	8	10	13	16	18	21	24	26	29	32	35	38	41
4	8	10	12	15	18	20	23	26	29	31	34	37	40	43
5	10	12	14	17										
6	12													

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	6	9	12	15	19	23	27	30	34	38	42	45	49	53
3	9	12	15	18	22	26	30	34	37	41	45	48	52	56
4	12	15												

n \ k	1	2	3	4	5	6	7	8
2	8	12	16	20	24	29	34	39
3	12	16	20	24	28	33	38	

n \ k	1	2	3	4	5	6	7	8
2	12	18	24	30	36	42	48	55

Figure B.9: Minimal cover sizes for subspaces $\mathbb{F}_q^n$. From top to bottom, $q = 3$, $q = 4$, $q = 5$, $q = 7$. The coloured values are the ones to which theorem 6.1 can be applied.

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	4	6	8	11	14	16	19	22	24	27	30	32	35	38
3	6	8	10	13	16	18	21	24	26	29	32	35	38	41
4	8	10	12	15	18	20	23	26	29	31	34	37	40	43
5	10	12	14	17										
6	12													

n \ k	1	2	3	4	5	6	7	8	9	10	11	12	13	14
2	6	9	12	15	19	23	27	30	34	38	42	45	49	53
3	9	12	15	18	22	26	30	34	37	41	45	48	52	56
4	12	15												

n \ k	1	2	3	4	5	6	7	8
2	8	12	16	20	24	29	34	39
3	12	16	20	24	28	33	38	

n \ k	1	2	3	4	5	6	7	8
2	12	18	24	30	36	42	48	55

0
1
2
3
4
5
6
7
8
9

Figure B.10: Minimal cover sizes for subspaces $\mathbb{F}_q^n$. From top to bottom, $q = 3$, $q = 4$, $q = 5$, $q = 7$. The colours show the difference between the Ball-Serra lower bound for each grid

Bibliography

[1] James Aaronson, Carla Groenland, Andrzej Grzesik, Tom Johnston, and Bartłomiej Kielak. Exact hyperplane covers for subsets of the hypercube, 2021.

[2] Noga Alon. Combinatorial nullstellensatz. *Combinatorics, Probability and Computing*, 8(1-2):7–29, 1999.

[3] Noga Alon and Zoltán Füredi. Covering the cube by affine hyperplanes. *European Journal of Combinatorics*, 14(2):79–83, 1993.

[4] Simeon Ball and Oriol Serra. Erratum to: Punctured combinatorial nullstellensätze. *Combinatorica*, 29:511–522, 09 2009.

[5] Anurag Bishnoi, Simona Boyadzhiyska, Shagnik Das, and Tamás Mészáros. Subspace coverings with multiplicities, 2021.

[6] Anurag Bishnoi, Pete L. Clark, Aditya Potukuchi, and John R. Schmitt. On zeros of a polynomial in a finite grid. *Combinatorics, Probability and Computing*, 27(3):310–333, 2018.

[7] Aart Blokhuis, A. Brouwer, and Tamás Szőnyi. Covering all points except one. *Journal of Algebraic Combinatorics*, 32:59–66, 08 2010.

[8] A.E Brouwer and A Schrijver. The blocking number of an affine space. *Journal of Combinatorial Theory, Series A*, 24(2):251–253, 1978.

[9] A. A. Bruen and J. C. Fisher. The jamison method in galois geometries. *Designs, Codes and Cryptography*, 1(3):199–205, 1991.

[10] A.A Bruen. Polynomial multiplicities over finite fields and intersection sets. *Journal of Combinatorial Theory, Series A*, 60(1):19–33, 1992.

[11] Alexander Clifton and Hao Huang. On almost k-covers of hypercubes. *Combinatorica*, 40, 04 2020.

[12] Zeev Dvir. On the size of kakeya sets in finite fields. *Journal of the American Mathematical Society*, 22(4):1093–1097, Jun 2008.

[13] Jordan Ellenberg and Dion Gijswijt. On large subsets of f_q^n with no three-term arithmetic progression. *Annals of Mathematics*, 185, 05 2016.

[14] Gurobi Optimization, LLC. Gurobi Optimizer Reference Manual, 2021.

[15] Robert E Jamison. Covering finite fields with cosets of subspaces. *Journal of Combinatorial Theory, Series A*, 22(3):253–266, 1977.

[16] P. Komjáth. Partitions of vector spaces. *Periodica Mathematica Hungarica*, 28(3):187–193, 1994.

[17] Géza Kós, Tamás Mészáros, and Lajos Rónyai. Some extensions of Alon's nullstellensatz. *Publicationes Mathematicae Debrecen*, pages 507–519, 2011.

[18] Lisa Sauermann and Yuval Wigderson. Polynomials that vanish to high order on most of the hypercube, 2020.

[19] The Sage Developers. *SageMath, the Sage Mathematics Software System (Version x.y.z)*, 2021. `https://www.sagemath.org`.