Paul Tabatabai, Bsc

On sequences covering subsets of a finite set

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme: Technical Mathematics, Operations Research and Statistics

submitted to

Graz University of Technology

Supervisor

Assoc.Prof. Dipl.-Math. Dr.rer.nat.habil. Christian Elsholtz

Institut für Analysis und Zahlentheorie

Graz, January 2018

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master‘s thesis.

Date	Signature

Acknowledgements

I would like to express my gratitude towards my supervisor Professor Elsholtz for his support during the research and writing of this thesis and for introducing me to many interesting mathematical topics, which will stay on my mind for a long time even after my departure from university.

I would like to thank my friends Leonardo Alese and Stefan Lendl for their invaluable cooperation, during which we managed to prove the results given in Chapter 2, and for the many fun afternoons of interesting discussions on problems discussed in and related to this thesis.

I am very grateful to my friend Manfred Scheucher for his many insightful comments and suggestions.

My sincere thanks to my parents Brigitta Tabatabai-Stocker and Behnam Tabatabai for their continuous support and encouragement throughout my studies.

Special thanks to my friends Jakob Saqri, Isabella Pototschnig, Benjamin Marko, Michael Cano, Leonardo Alese, Stefan Lendl and Manfred Scheucher and to my parents Brigitta Tabatabai-Stocker and Behnam Tabatabai for their help with proofreading this thesis.

Abstract

Let $s(n)$ denote the smallest positive integer with the property that there exists a sequence S of length $s(n)$ over the alphabet $\{1, \ldots, n\}$ such that S contains every subset of $\{1, \ldots, n\}$ as a block of consecutive elements. We provide the previously unknown values $s(6) = 24$ and $s(7) = 40$ by means of a backtracking algorithm, utilizing an efficient early-pruning condition. Further, we give an integer programming formulation for calculating values of $s(n)$. We introduce a probabilistic heuristic algorithm, which provides the currently smallest known upper bounds for the values $s(n)$ for $n = 8, \ldots, 20$. Finally we analyze constructions by Jukna and Lipski—the latter giving the currently smallest known asymptotic upper bound for $s(n)$. We introduce a simple greedy algorithm, outperforming Lipski's construction for all values of n where computation is feasible, which indicates that the bound obtained from Lipski's construction may not be asymptotically tight.

Let $a(n)$ denote the smallest positive integer with the property that there exists a colouring f of $\{1, \ldots, a(n)\}$ such that for every subset $R \subseteq \{1, \ldots, n\}$ there exists an arithmetic $|R|$-progression A in $\{1, \ldots, a(n)\}$ with $\{f(a) : a \in A\} = R$.
Further, let $a(n, k)$ denote the smallest positive integer with the property that there exists a colouring f of $\{1, \ldots, a(n, k)\}$ such that for every k-subset $R \subseteq \{1, \ldots, n\}$ there exists an arithmetic k-progression A in $\{1, \ldots, a(n, k)\}$ with $\{f(a) : a \in A\} = R$.
Determining the behaviour of the functions $a(n)$ and $a(n, k)$ is a previously unstudied problem. Using a genetic algorithm, we calculate upper bounds for $a(n)$ for small values of n. In joint work with Leonardo Alese and Stefan Lendl, we use the first moment method to give an asymptotic upper bound for $a(n, k)$ for the case where $k = o(n^{1/6})$.

We introduce the following problem: In a fixed class of graphs we want to find a graph G with the least possible number of vertices that can be vertex-coloured in such a way that every subset of $\{1, \ldots, n\}$ appears as the vertex colours of a connected subgraph of G. We give examples for several classes of graphs.

Contents

List of Algorithms

1 Sequences covering all subsets of a finite set

1.1 Introduction

In 1977 Witold Lipski [14] studied the combinatorial problem of finding short sequences containing every subset of a finite set as a block of consecutive elements.

Let S be a finite sequence. We call a collection of consecutive elements of S a *block*. If a block consists of k elements, we may call the block a k-*block*. Let $\binom{[n]}{k}$ denote the family of all k-subsets of $[n] = \{1, \ldots, n\}$ and let $Y \in \binom{[n]}{k}$ be a k-subset of $[n]$. We say a k-block $B = (b_1, \ldots, b_k)$ *covers* Y if the elements of the block B are pairwise distinct and $\{b_1, \ldots, b_k\} = Y$. We say S *covers* Y if there exists a block of S that covers Y.

Example. The sequence $(1, 2, 3, 4, 1, 2)$ covers the set $\{1, 3, 4\}$ because it contains the block $(3, 4, 1)$. $\triangle$

Let $n \in \mathbb{N}$ and let $\mathcal{P}$ be a family of subsets of $[n]$. We say a sequence S over the alphabet $[n]$ is a $\mathcal{P}$-covering sequence (or S *covers* $\mathcal{P}$) if S covers every set in $\mathcal{P}$. The problem of finding a shortest $\mathcal{P}$-covering sequence is P-complete if we allow $\mathcal{P}$ to be an arbitrary family of subsets of $[n]$. This was shown in 1977 by L. T. Kou [12]. For $n \in \mathbb{N}$, let $P(n)$ denote the powerset of $[n]$ except for the empty set, i.e,

$$P(n) = 2^{[n]} \setminus \{\emptyset\},$$

and for pairwise distinct positive integers $k_1, \ldots, k_r \leq n$ define

$$P_{k_1,\ldots,k_r}(n) = \binom{[n]}{k_1} \cup \binom{[n]}{k_2} \cup \cdots \cup \binom{[n]}{k_r},$$

the family of all subsets $X \subseteq [n]$ with $|X| \in \{k_1, \ldots, k_r\}$. We are interested in the cases $\mathcal{P} = P(n)$ and $\mathcal{P} = P_{k_1,\ldots,k_r}(n)$. We define $s(n)$ to be the length of a shortest $P(n)$-covering sequence and $s_{k_1,\ldots,k_r}(n)$ to be the length of a shortest $P_{k_1,\ldots,k_r}(n)$-covering sequence.

Example. The sequence $(1, 2, 3, 1, 4, 2, 3, 4)$ is a $P(4)$-covering sequence because it covers all subsets of $\{1, \ldots, 4\}$:

$$\begin{array}{rl}
\{1,2\}: & (\mathbf{1}, \mathbf{2}, 3, 1, 4, 2, 3, 4) \\
\{1,3\}: & (1, 2, \mathbf{3}, \mathbf{1}, 4, 2, 3, 4) \\
\{1,4\}: & (1, 2, 3, \mathbf{1}, \mathbf{4}, 2, 3, 4) \\
\{2,3\}: & (1, \mathbf{2}, \mathbf{3}, 1, 4, 2, 3, 4) \\
\{2,4\}: & (1, 2, 3, 1, \mathbf{4}, \mathbf{2}, 3, 4) \\
\{3,4\}: & (1, 2, 3, 1, 4, 2, \mathbf{3}, \mathbf{4}) \\
\{1,2,3\}: & (\mathbf{1}, \mathbf{2}, \mathbf{3}, 1, 4, 2, 3, 4) \\
\{1,2,4\}: & (1, 2, 3, \mathbf{1}, \mathbf{4}, \mathbf{2}, 3, 4) \\
\{1,3,4\}: & (1, 2, \mathbf{3}, \mathbf{1}, \mathbf{4}, 2, 3, 4) \\
\{2,3,4\}: & (1, 2, 3, 1, \mathbf{4}, \mathbf{2}, \mathbf{3}, 4) \\
\{1,2,3,4\}: & (1, 2, 3, \mathbf{1}, \mathbf{4}, \mathbf{2}, \mathbf{3}, 4)
\end{array}$$

△

Note that $s_k(n) \leq s(n)$ holds for all $k \leq n$. Lipski [14] used Propositions 1.1 and 1.2 to give lower bounds for $s(n)$.

Proposition 1.1 (Lipski [14])**.** *For $n, k \in \mathbb{N}$ (where $k \leq n$) we have*

$$s_k(n) \geq n \cdot \left\lceil \binom{n-1}{k-1} / k \right\rceil .$$

Proof. For each fixed $i \in [n]$ there are $\binom{n-1}{k-1}$ k-subsets of $[n]$ containing i. Let S be a sequence covering all k-subsets of $[n]$. Each element of S appears in at most k k-blocks of S (elements of S that appear close to the beginning or the end of S are contained in less than k k-blocks). Hence, every $i \in [n]$ must appear at least $\lceil \binom{n-1}{k-1} / k \rceil$ times in S, which implies

$$|S| \geq n \cdot \left\lceil \binom{n-1}{k-1} / k \right\rceil .$$

□

Proposition 1.2 (Lipski [14])**.** *For $n, k \in \mathbb{N}$ (where $k \leq n$), we have*

$$s_k(n) \geq k + \binom{n}{k} - 1.$$

Proof. Since there are $\binom{n}{k}$ k-subsets of $[n]$, a sequence covering all k-subsets of $[n]$ must consist of at least $\binom{n}{k}$ k-blocks. A sequence consisting of exactly $\binom{n}{k}$ k-blocks has length $k + \binom{n}{k} - 1$; the first k elements form a single k-block, and each of the remaining $\binom{n}{k} - 1$ elements induces another k-block. □

Setting $k = \lceil n/2 \rceil$ in the above proposition, Lipski obtained the lower bound

$$s(n) \geq \lceil n/2 \rceil + \binom{n}{\lceil n/2 \rceil} - 1 \quad \text{(for all } n \geq 1\text{)}.$$

Since $\binom{n}{\lceil n/2 \rceil} = 2^n \sqrt{\frac{2}{\pi n}}\,(1 + o(1))$ as n tends to infinity, we obtain the following corollary:

Corollary (Lipski [14])**.** *There exists a real-valued function $\psi_1 : \mathbb{N} \mapsto \mathbb{R}$ with $\psi_1(n) = o(1)$ as n tends to infinity, such that for all sufficiently large $n \in \mathbb{N}$*

$$s(n) \geq 2^n \sqrt{\frac{2}{\pi n}}(1 + \psi_1(n))$$

holds.

In Section 1.3.3 we present an upper bound for $s(n)$, which was obtained by Lipski [14] using a clever construction:

Theorem (Lipski [14])**.** *There exists a real-valued function $\psi_2 : \mathbb{N} \mapsto \mathbb{R}$ with $\psi_2(n) = o(1)$ as n tends to infinity, such that for all sufficiently large $n \in \mathbb{N}$*

$$s(n) \leq 2^n \frac{2}{\pi}(1 + \psi_2(n))$$

holds.

The true asymptotic behaviour of $s(n)$ is not known. The closely related problem of determining $s_k(n)$ is well studied, but in a slightly different context:

Definition 1.3 (Universal sequences and universal cycles)**.** A sequence S over the alphabet $[n]$ is called *(n, k)-universal sequence* if S consists of exactly $\binom{n}{k}$ k-blocks, each covering a *unique* k-subset of $[n]$. A cycle is a sequence where we extend the definition of consecutive elements by allowing *wrap-around* along the ends of the sequence (for example, the 3-blocks of the cycle $C = (1, 2, 3, 4)$ are $(1, 2, 3), (2, 3, 4), (4, 1, 2)$ and $(3, 4, 1)$). A cycle C is called *(n, k)-universal cycle* if C consists of exactly $\binom{n}{k}$ k-blocks, each covering a *unique* k-subset of $[n]$. $\triangle$

Equivalently, a sequence S is an *(n, k)-universal sequence* if it covers all k-subsets of $[n]$ and has length $k + \binom{n}{k} - 1$. A cycle C is an *(n, k)-universal sequence* if and only if it covers all k-subsets of $[n]$ and has length $\binom{n}{k}$.
The following conjecture by Chung, Diaconis and Graham [4] has been the main focus of research on this subject.

Conjecture 1.4 (Chung, Diaconis, Graham [4]). *Let $k \in \mathbb{N}$. There exists a positive integer n_k such that for all $n \geq n_k$, there exists an (n,k)-universal cycle if and only if*

$$n \mid \binom{n}{k},$$

or equivalently $k \mid \binom{n-1}{k-1}$.

Because of symmetry, all elements $1, \ldots, n$ must appear an equal number of times in an (n,k)-universal cycle. Thus, the condition in Conjecture 1.4 is necessary for all $k, n \in \mathbb{N}$ (where $k \leq n$).
For $k = 2$, the condition $2 \mid \binom{n-1}{1}$ is equivalent to n being odd. An $(n,2)$-universal cycle corresponds to an Eulerian tour in the complete graph K_n on n vertices, and since the graph K_n is Eulerian if and only if n is odd, Conjecture 1.4 is true for the case $k = 2$ (observed by Chung, Diaconis, Graham [4], and by D. Curtis *et al.* [5]).
For $k = 3$, Jackson [10] proved that Conjecture 1.4 holds (with $n_3 = 8$) and further gave a partial positive result for the case $k = 4$ (missing the case where $n \equiv 2 \mod 8$, which remains unresolved to this date). Glenn Hurlbert [9] proved Conjecture 1.4 for $k = 6$ for the case where n is relatively prime to k.

For n and k where an (n,k)-universal cycle does not exist, it is natural to consider the problem of finding a shortest cycle covering all k-subsets of $[n]$. Let $c_k(n)$ denote the length of such a shortest cycle.

In 2016, Michał Dębski and Zbigniew Lonc [6] gave the following asymptotic results:

Theorem (Dębski and Lonc [6]). *For fixed $k \in \mathbb{N}$, as n tends to infinity,*

$$c_k(n) = \binom{n}{k} + O(n^{\lceil k/2 \rceil})$$

holds. Let $0 < \alpha \leq \frac{1}{3}$ be fixed. Let $k = k(n) \leq n^\alpha$ for all $n \in \mathbb{N}$. As $n \to \infty$, we have

$$c_k(n) = \binom{n}{k} + o\left(\binom{n}{k}^{\beta}\right),$$

where $\beta = \frac{1+\alpha}{2-2\alpha}$.

Note that since $c_k(n) \leq s_k(n) \leq c_k(n) + n$ for all $k, n \in \mathbb{N}$ (where $k \leq n$), the asymptotic results in the above theorem also apply to $s_k(n)$. Lipski conjectured that asymptotically, $s(n) \sim s_{\lfloor n/2 \rfloor}(n)$. Unfortunately, the asymptotic behaviour of $c_k(n)$ seems to be unknown in the case where $k = k(n)$ is a *linear* function of n.

Determining $s_k(n)$ for $k = 2$ and $k = n - 1$ is easy. In Propositions 1.5 and 1.6 we give the corresponding results.

Proposition 1.5. *For all $n \geq 2$*

$$s_{n-1}(n) = 2n - 2$$

holds.

Proof. Proposition 1.2 implies $s_{n-1}(n) \geq 2n - 2$. We claim that

$$S_{n-1}(n) = (1, 2, \ldots, n, 1, 2, \ldots, n-2)$$

is a $P_{n-1}(n)$-covering sequence, showing $s_{n-1}(n) \leq 2n - 2$.

The sets in $P_{n-1}(n)$ are exactly the sets of the form $\{1, \ldots, n\} \setminus \{j\}$ for $1 \leq j \leq n$. The first n elements of $S_{n-1}(n)$ contain the two $(n-1)$-blocks $(1, \ldots, n-1)$ and $(2, \ldots, n)$, covering the sets $\{1, \ldots, n\} \setminus \{n\}$ and $\{1, \ldots, n\} \setminus \{1\}$, respectively. For each $i \in \{1, \ldots, n-2\}$ the sequence $S(n)$ contains the block $(i+2, i+3, \ldots, n, 1, \ldots, i)$, which covers the set $\{1, \ldots, n\} \setminus \{i+1\}$. □

Proposition 1.6. *For all $n \geq 2$,*

$$s_2(n) = \begin{cases} \binom{n}{2} + 1 & \textit{if } n \textit{ is odd} \\ \binom{n}{2} + \frac{n}{2} & \textit{if } n \textit{ is even} \end{cases}$$

holds.

Proof. Let K_n be the complete (undirected) graph on n vertices. Each edge in K_n corresponds to a 2-subset of $[n]$. A walk $v_1, \ldots, v_r$ in K_n that traverses every edge exactly once thus corresponds to a shortest $P_2(n)$-covering sequence. If n is odd, K_n is Eulerian and there exists a walk that traverses every edge exactly once, corresponding to a $P_2(n)$-covering sequence of length $\binom{n}{2} + 1$.

For even n, Proposition 1.1 implies $s_2(n) \geq \binom{n}{2} + \frac{n}{2}$. We construct a $P_2(n)$-covering sequence of length $\binom{n}{2} + \frac{n}{2}$. In general, a connected graph has an Eulerian walk if and only if at most two of its vertices have odd degree. For even n, all vertices of K_n have odd degree. By adding $\frac{n-2}{2}$ mutually disjoint edges to K_n, we obtain a graph K_n^*, in which all but two vertices have odd degree. The $P_2(n)$-covering sequence corresponding to an Eulerian walk in K_n^* has length $\binom{n}{2} + \frac{n-2}{2} + 1 = \binom{n}{2} + \frac{n}{2}$.

□

1.2 Exact values and bounds for small n

Until now, exact values of $s(n)$ were known only up to $n = 5$. The values listed in Table 1 were already known to Lipski [14].

n	$s(n)$
1	1
2	2
3	4
4	8
5	13

Table 1: Previously known values of $s(n)$.

In Section 1.2.1 we provide a backtracking algorithm, which allows us to prove $s(6) = 24$ and $s(7) = 40$. Showing $s(6) = 24$ can be done by other means as well; the lower bound in Proposition 1.1 implies $s(6) \geq s_3(6) \geq 24$, and a $P(6)$-covering sequence of length 24 can easily be found by the randomized heuristic algorithms introduced in Section 1.2.2; one such sequence is (commas omitted):

$$S_6^* = (123456125362415364136254).$$

For $n = 7$, the backtracking approach was necessary; the lower bound in Proposition 1.2 only implies $s(7) \geq 38$, and running the heuristic algorithms from Section 1.2.2 a large number of times, we were only able to find sequences implying $s(7) \leq 42$. Using the backtracking algorithm, we showed $s(7) > 39$ and found the following $P(7)$-covering sequence of length 40 (commas omitted):

$$S_7^* = (1237612531467254173526347563124651724356).$$

We use the heuristic algorithms from Section 1.2.2 to give new upper bounds for $s(n)$ for $n = 8, 9, \ldots, 20$, where the backtracking approach was not computationally feasible.

Lipski [14] left it as an exercise to the reader to prove $s(5) = 13$. It is easy to calculate $s(5)$ by exhaustive search, but we give a combinatorial proof here.

Proposition 1.7. *We have $s(5) = s_3(5) = 13$.*

Proof. The sequence

$$(1,2,3,4,5,1,2,4,1,3,5,2,4)$$

covers all subsets of $\{1,\ldots,5\}$ and hence $s_3(5) \leq s(5) \leq 13$.

To show $s_3(5) > 12$, assume that there is a sequence $S = (a_1,\ldots,a_{12})$ of length 12 that covers all 3-subsets of $\{1,\ldots,5\}$. Without loss of generality, we can assume that the first three elements of S are $1,2,3$:

$$S = (1,2,3,a_4,a_5,a_6,a_7,a_8,a_9,a_{10},a_{11},a_{12}).$$

Note that in a sequence of length 12, there are exactly $\binom{5}{3} = 10$ 3-blocks. Therefore, each 3-block of S must cover a unique 3-subset of $\{1,\ldots,5\}$. For each fixed $i \in \{1,\ldots,5\}$, the number of 3-sets containing the element i is $\binom{4}{2} = 6$. Thus, for each $i \in \{1,\ldots,5\}$, there must be exactly six 3-blocks in S that contain i exactly once. Since every occurrence of an element i can be part of at most three 3-blocks, every $i \in \{1,\ldots,5\}$ has to appear at least twice in S. The elements 1 and 2 have to appear at least three times:
Since a_1 $(= 1)$ appears in only a single 3-block (the block (a_1,a_2,a_3)), the sequence S must contain at least two more occurrences of the element 1, otherwise S could contain at most four 3-blocks containing the element 1. Similarly, a_2 $(= 2)$ appears only in two 3-blocks (the blocks (a_1,a_2,a_3) and (a_2,a_3,a_4)) and S thus has to contain at least two more occurrences of the element 2, otherwise S could contain at most five 3-blocks containing the element 2.
Since $|S| = 12$, we have thus accounted for all elements; S contains exactly two occurrences of the elements $3,4,5$ and exactly three occurrences of the elements 1 and 2.

Note that s_{12} is only part of one 3-block in S and that s_{11} is only part of two 3-blocks in S. We show that $a_{12} = 2$ and $a_{11} = 1$ must hold. Assume that $a_{12} \in \{3,4,5\}$. Then, arguing similarly to before, S would have to contain more than two occurrences of some element $j \in \{3,4,5\}$, contradicting the fact that only the elements 1 and 2 appear three times in S. Thus $a_{12} \in \{1,2\}$ and similarly, $a_{11} \in \{1,2\}$. Assume that $a_{12} = 1$. Then the two occurrences a_1 and a_{12} of the element 1 each appear in only a single 3-block of S, and S would have to contain two more occurrences of the element 1, contradicting the fact that there must be exactly three occurrences of the element 1 in S. Hence the only possible configuration is the following:

$$S = (1,2,3,a_4,a_5,a_6,a_7,a_8,a_9,a_{10},1,2).$$

Further, $a_{10} \notin \{1,2,3\}$. For each of these choices for a_{10}, S contains a block that does not cover a unique 3-subset of $\{1,\ldots,5\}$:

for $s_{10} = 1$, the sequence S contains the block $(1, 1, 2)$, for $s_{10} = 2$, the sequence S contains the block $(2, 1, 2)$, and for $s_{10} = 1$, the sequence S contains the blocks $(1, 2, 3)$ and $(3, 1, 2)$, both covering the same subset. Without loss of generality, $a_{10} = 4$:

$$S = (1, 2, 3, a_4, a_5, a_6, a_7, a_8, a_9, 4, 1, 2).$$

The set $\{1, 2, 5\}$ has to be covered by a block in which the elements 1 and 2 are separated: $(1, 5, 2)$ or $(2, 5, 1)$. All other $\{1, 2, 5\}$-covering blocks place 1 and 2 next to each other, forcing a *fourth* block in S that contains both 1 and 2, which can not cover a unique 3-set. There are four possibilities:

$$\begin{aligned}
\text{a) } S &= (1, 2, 3, a_4, 1, 5, 2, a_8, a_9, 4, 1, 2).\\
\text{b) } S &= (1, 2, 3, a_4, a_5, 1, 5, 2, a_9, 4, 1, 2).\\
\text{c) } S &= (1, 2, 3, a_4, 2, 5, 1, a_8, a_9, 4, 1, 2).\\
\text{d) } S &= (1, 2, 3, a_4, a_5, 2, 5, 1, a_9, 4, 1, 2).
\end{aligned}$$

All other possible configurations (for example $(1, 2, 3, 1, 5, 2, a_7, a_8, a_9, 4, 1, 2)$) contain a block that does not cover a unique 3-set. In the above example the set $\{1, 2, 3\}$ is covered by both of the blocks $(1, 2, 3)$ and $(2, 3, 1)$.

We analyze case a). The sequence S must consist of exactly three occurrences of the elements 3, 4 and 5, and thus $\{a_4, a_8, a_9\} = \{3, 4, 5\}$ must hold. We have $a_4 = 4$; for $a_4 = 3$, the sequence S contains the block $(2, 3, 3)$ and for $a_4 = 5$, the sequence S contains the block $(5, 1, 5)$. Arguing similarly, we have $a_8 = 3$. It follows that $a_9 = 5$ and

$$S = (1, 2, 3, 4, 1, 5, 2, 3, 5, 4, 1, 2).$$

The above sequence does not cover the set $\{2, 4, 5\}$. Using similar arguments, in each of the remaining cases b), c) and d) we also end up with a sequence not covering all of the 3-subsets of $\{1, \dots, 5\}$. This contradicts the existence of a sequence of length 12 covering all 3-subsets of $\{1, \dots, 5\}$. $\square$

1.2.1 Backtracking Algorithm

Backtracking algorithms have been successfully used to calculate exact values of combinatorially defined functions. For example, backtracking algorithms for calculating small van der Waerden Numbers are described in Landman's and Robertson's book *Ramsey Theory on the Integers* [13]. In this section we present a similar approach for calculating values of $s(n)$ and $s_{k_1,\dots,k_r}(n)$.

For given $n \in \mathbb{N}$ and $N \in \mathbb{N}$ we want to find a $P(n)$-covering sequence of length N or prove that no such sequence exists. This can be done by searching the set of all sequences of length N over the alphabet $[n]$. To show that $s(7) = 40$ one would have to partially search a space of size 7^{40} (to find a suitable sequence) and exhaust a search space of size $7^{39} \approx 9 \times 10^{32}$ (to show that no $P(7)$-covering sequence of length 39 exists). There are two observations that immediately help us reduce the search space.

1. We only need to generate sequences up to isomorphism; a sequence S is a $P(n)$-covering sequence if and only if any permutation of the element labels of S again results in a $P(n)$-covering sequence.

2. We do not need to generate sequences where an element appears in two consecutive positions.

Further, sometimes it suffices to look at the initial part of a sequence to recognize that it can not be a $P(n)$-covering sequence of length N. Consider the following example. Say we want want to check whether the following sequence can be extended to become a $P(5)$-covering sequence of length 13:

$$S = (1, 2, 3, 1, 3).$$

The set $\{1, 2, 3\}$ is covered twice by S. It is covered by both the block $(1, 2, 3)$ and the block $(2, 3, 1)$. Further the block $(3, 1, 3)$ does not cover any 3-subset. Hence two 3-blocks do not cover a new 3-subset, and arguing similarly as in Proposition 1.2, an extension S' of S that covers all 3-subsets of $\{1, \ldots, 5\}$ must consist of $2 + \binom{5}{3} = 12$ or more 3-blocks, implying $|S'| \geq 14$.

We call a sequence *bad* if it can not be extended to become a $P(n)$-covering sequence of length N. Whenever we detect such a *bad* sequence, we know we do not have to check any extensions of that sequence, further reducing the search space.

In Proposition 1.8 we give a sufficient condition by which we can recognize bad sequences. Let S be a sequence over the alphabet $[n]$. For $k \leq n$ and $i \in [n]$ define $a_{k,i}$ to be the number of k-subsets of $[n]$ containing i that are *not* covered by S. Further we define l_i to be the number of elements in S that appear after the *last* occurrence of i.

Proposition 1.8. *Let S be a sequence over the alphabet $[n]$. If for some $k \leq n$ the inequality*

$$|S| + \sum_{i \in [n]} \left\lceil \frac{a_{k,i} - \max(k - l_i - 1, 0)}{k} \right\rceil > N$$

holds, then S is a bad sequence, i.e., S can not be extended to become a $P(n)$-covering sequence of length N.

Proof. We show that each term

$$\left\lceil \frac{a_{k,i} - \max(k - l_i - 1, 0)}{k} \right\rceil$$

is a lower bound for the number of additional occurrences of the element i that have to be added to S in order for S to cover all k-subsets of $[n]$ containing i. If there are less than $k-1$ elements trailing the last occurrence of i in S, then new k-blocks containing i can be created by appending elements of $[n] \setminus \{i\}$ to S. In this case, the number $k - l_i - 1$ counts how many new k-blocks can be created this way. If there are $k-1$ or more elements trailing the last occurrence of i in S, appending elements of $[n] \setminus \{i\}$ to S can not create any new k-blocks containing i. Note that in this case $k - l_i - 1 < 0$.

Each newly generated block containing i can *at most* cover one previously not covered k-set containing i. Thus, after potentially appending some number of elements of $[n] \setminus \{i\}$ to S, the number of k-sets containing i that are not covered by S is at *least*

$$a_{k,i} - \max(k - l_i - 1, 0).$$

Since each occurrence of i appears in at most k k-blocks—each covering at most one k-set—we need to add at least

$$\left\lceil \frac{a_{k,i} - \max(k - l_i - 1, 0)}{k} \right\rceil$$

additional occurrences of the element i to S in order for S to cover all of the remaining k-sets containing i. □

Definition 1.9. We call a sequence *invalid* if the inequality from Proposition 1.8 holds for some $k \leq n$. We call a sequence *valid* if it is *not invalid.* △

Note that while an invalid sequence is always a bad sequence, a valid sequence may or may not be bad.

Without loss of generality, we can fix the first three elements of our sequence to be $1, 2, 3$.
In rough terms, the algorithm works as follows. We start with $S = (1, 2, 3)$. We keep *extending* S with the smallest possible element (1 if the last element of S is not 1, and 2 otherwise), until S becomes invalid or until S has reached the goal length N. If S has length N and is a $P(n)$-covering sequence, we return S and terminate. Otherwise we call the procedure *makeValid*, which—by potentially *backtracking*—finds the *next* valid sequence; first we iteratively increment the last element of S. If S does not become valid this way, we

remove the last element from S (backtracking) and again iteratively increment the last element and repeat this procedure. If we end up with the sequence $S = (1, 2, 3)$, we have exhausted the whole search space. If we end up with a valid sequence (which is not of goal length N), we go back to the extending-phase and repeat this whole process.

In summary, we keep alternatingly calling the procedures *extend* and *makeValid*, and whenever we reach goal length, we check whether we have found a $P(n)$-covering sequence.

Algorithm 1.10 Backtracking Algorithm

Input: $n, N \in \mathbb{N}$.

Output: A $P(n)$-covering sequence of length N, or proof that no such sequence exists.

$S = (1, 2, 3)$.
while True **do**
    extend(S).
    if $|S| = N$ **then**
        if S is a $P(n)$-covering sequence of length N **then**
            return S.
    makeValid(S).
    if $|S| = 3$ **then**
        return *No covering of length N exists.*
    if $|S| = N$ **then**
        if S is a $P(n)$-covering sequence of length N **then**
            return S.

By $S[-1]$ and $S[-2]$ we denote the last and second-to-last elements of S, respectively. The *extend*-procedure is very simple.

Algorithm 1.11 Procedure: extend(S)

while $|S| < N$ and isValid(S) **do**
    if $S[-1] = 1$ **then**
        Append the element 2 to S.
    else
        Append the element 1 to S.

The *makeValid*-procedure is where we make sure to generate sequences only up to isomorphism. Consider this example. Let $S = (1, 2, 3, 4, 5, 3)$ be a sequence over the alphabet $\{1, \ldots, 8\}$. Iteratively incrementing the last element of S gives the following sequences:

$$\begin{gathered}(1,2,3,4,5,4),\\(1,2,3,4,5,6),\\(1,2,3,4,5,7),\\(1,2,3,4,5,8).\end{gathered}$$

Note that we do not consider the incremented sequence $(1,2,3,4,5,5)$ since it contains the element 5 in two consecutive positions. Further, the sequences $(1,2,3,4,5,6),(1,2,3,4,5,7),(1,2,3,4,5,8)$ are pairwise isomorphic to each other; the trailing elements $6,7$ and 8 occur exactly once in their respective sequences. In general, let R_1 be the set of elements that do not appear in our current sequence S. Whenever we want to increment the last element of S, we only need to consider the elements that do appear in S and a single representative for the set R_1. If R_1 is nonempty, we choose the largest element in R_1 as a representative for R_1.

Algorithm 1.12 Procedure: makeValid(S)

while True **do**
    if ($S[-1]=n$) or ($S[-1]=n-1$ and $S[-2]=n$) **then**
        Remove last element from S.
        if $|S|=3$ **then**
            return. (Search space exhausted.)
        if isValid(S) **then**
            return.
    $R_1=\{i\in[n]: i\notin S\}$.
    $R_2=\{i\in[n]: i\in S \ \wedge\ i\neq S[-2] \ \wedge\ i>S[-1]\}$.
    $R=\{\max(R_1)\}\cup R_2$.
    Sort R in ascending order.
    for $r\in R$ **do**
        $S[-1]=r$.
        if isValid(S) **then**
            return.

Algorithm 1.10 can also be used to compute values of $s_{k_1,\ldots,k_r}(n)$. In this case we define a sequence to be *valid* if the inequality from Proposition 1.8 holds for some $k\in\{k_1\ldots,k_r\}$. Further we do validity-checking only for values included in $\{k_1\ldots,k_r\}$ and check whether S is a $P_{k_1,\ldots,k_r}(n)$-covering sequence whenever S is of goal length N.

Implementation details. During the execution of the backtracking algorithm, S is changed often, and we need to keep track of the sets that are covered by the current sequence S. Further, for all $i \in [n]$ we need to keep track of the number of sets containing i that are covered by S. This can be done efficiently because whenever we change S, we need to consider at most the n trailing elements of S to check for newly covered sets or sets that are not covered anymore.

It is a good strategy to do validity-checking by only considering values of k that are close to $n/2$, since usually the invalidity condition is fulfilled for these values first. Not checking small and large values of k leads to a few unnecessarily generated sequences, but the time lost by that is outweighed by having to keep track of covered k-sets for fewer values of k.

Results. Using Algorithm 1.10 we calculated $s(6)$, $s(7)$ and $s_k(n)$ for various values of k and n. The algorithm generated 126704677 sequences to prove that no $P(7)$-covering sequence of length 39 exists, i.e., $s(7) > 39$. Table 2 lists the values of $s(n)$ and $s_k(n)$ we managed to calculate.

n	$s(n)$	$s_2(n)$	$s_3(n)$	$s_4(n)$	$s_5(n)$	$s_6(n)$	$s_7(n)$	$s_8(n)$
1	1	-	-	-	-	-	-	-
2	2	2	-	-	-	-	-	-
3	4	4	3	-	-	-	-	-
4	8	8	6	4	-	-	-	-
5	13	11	13	8	5	-	-	-
6	24	18	24	20	10	6	-	-
7	40	22	37	38	28	12	7	-

Table 2: Known values of $s(n)$ and $s_k(n)$.

Below, we list the values of $s_{k_1,\dots,k_r}(n)$ we calculated. Since $s_3(5) = s(5)$ and $s_3(6) = s(6)$, we have $s_{k_1,\dots,k_r}(n) = s(n)$ for $n \in \{5, 6\}$ and $3 \in \{k_1, \dots, k_r\}$. Further, we calculated $s_{3,4}(7) = s(7)$ and hence $s_{k_1,\dots,k_r}(7) = s(7)$ for $3, 4 \in \{k_1, \dots, k_r\}$.

- $s_{3,4}(4) = 6$
- $s_{2,4}(5) = 11$
- $s_{2,5}(5) = 11$
- $s_{5,6}(6) = 10$
- $s_{2,6}(6) = 18$
- $s_{4,5}(6) = 20$
- $s_{4,6}(6) = 20$
- $s_{2,4}(6) = 21$
- $s_{2,5}(6) = 28$
- $s_{2,5,6}(6) = 18$
- $s_{4,5,6}(6) = 20$
- $s_{2,4,5}(6) = 21$
- $s_{2,4,6}(6) = 21$
- $s_{2,4,5,6}(6) = 21$
- $s_{2,3}(7) = 37$

- $s_{2,4}(7) = 38$
- $s_{2,5}(7) = 29$
- $s_{2,6}(7) = 22$
- $s_{2,7}(7) = 22$
- $s_{3,4}(7) = 40$
- $s_{3,5}(7) = 37$
- $s_{3,6}(7) = 37$
- $s_{4,5}(7) = 38$
- $s_{5,6}(7) = 28$
- $s_{6,7}(7) = 12$
- $s_{2,4,6}(7) = 38$
- $s_{5,6,7}(7) = 28$
- $s_{2,5,6,7}(7) = 29$
- $s_{2,3,5,6,7}(7) = 37$
- $s_{2,4,5,6,7}(7) = 38$

One can ask which subset-sizes $k_1, \ldots, k_r \leq n$ are the ones "responsible" for the value of $s(n)$, i.e., what integers $k_1, \ldots, k_r \leq n$ have the property $s_{k_1,\ldots,k_r}(n) = s(n)$. Lipski [14] conjectured, that as n tends to infinity, $s(n) \sim s_{\lfloor \frac{n}{2} \rfloor}(n)$ holds. For $n = 3, \ldots, 6$ we have $s(n) = s_{\lceil \frac{n}{2} \rceil}(n)$:

$$
\begin{aligned}
&s_2(3) = s(3) = 5, \\
&s_2(4) = s(4) = 8, \\
&s_3(5) = s(5) = 13 \text{ and} \\
&s_3(6) = s(6) = 24.
\end{aligned}
$$

For $n = 7$, we calculated $s_4(7) = 38 < 40 = s(7)$, which breaks the above pattern, but we can see that $s_{3,4}(7) = s(7) = 40$, giving rise to the optimistic conjecture that for all odd n

$$s_{\lfloor \frac{n}{2} \rfloor, \lceil \frac{n}{2} \rceil}(n) = s(n),$$

and for all even n either

$$s_{\frac{n}{2}-1, \frac{n}{2}, \frac{n}{2}+1}(n) = s(n),$$

or—even more optimistically—$s_{\frac{n}{2}}(n) = s(n)$ holds. Unfortunately calculating $s(n)$ for $n \geq 8$ is out of reach of our backtrackig algorithm, so we were not able to check whether this pattern continues for $n \geq 8$.

On the number of solutions. We are interested in the number of solutions to our problem, i.e., the number of shortest $P(n)$-covering sequences. We say two sequences S_1, S_2 over the alphabet $[n]$ are *isomorphic* to each other ($S_1 \backsim S_2$) if there exists a permutation $\pi : [n] \to [n]$ such that $(\pi(s) : s \in S_1) = S_2$. The backtracking algorithm described in this section can easily be modified to generate *all* $P(n)$-covering sequences. While it creates solutions only up to isomorphism, it generates some solutions twice; note that if a sequence S is a shortest $P(n)$-covering sequence, then the reverse of the sequence, $\overleftarrow{S}$ is a also a shortest $P(n)$-covering sequence, and in general $S \not\backsim \overleftarrow{S}$. In this case our algorithm will output both S and $\overleftarrow{S}$. If a solution is isomorphic to its own reverse, our algorithm generates it only once. We are interested in the number of *different* solutions. To make this precise, we define $v(n)$ to be the size of a maximal family of shortest $P(n)$-covering sequences $\{S_1, \ldots, S_{v(n)}\}$ such that for all $i < j$ the sequences S_i and S_j fulfill $S_i \not\sim S_j$ and $S_i \not\sim \overleftarrow{S_j}$. Further we define $w(n)$ to be the total number of shortest $P(n)$-covering sequences (up to isomorphism) that are isomorphic to their own reverse.

Table 3 lists the values of $v(n)$ and $w(n)$ for $n = 1, \ldots, 7$. Interestingly, there are many shortest $P(6)$-covering sequences, but only a single shortest $P(7)$-covering sequence exists (commas omitted):

$$S_7^* = (1237612531467254173526347563124651724356).$$

n	$v(n)$	$w(n)$
1	1	1
2	1	1
3	1	1
4	3	1
5	2	0
6	57	1
7	1	0

Table 3: Known values of $v(n)$ and $w(n)$.

1.2.2 Heuristic approaches

The backtracking algorithm described in the previous section was only computationally feasible for up to $n = 7$. In this section we describe three heuristic approaches to generate short $P(n)$-covering sequences for larger values of n. To our knowledge, the sequences found by Algorithm 1.17 give the smallest known upper bounds for $s(n)$ for $n = 8, \ldots, 20$.

The first heuristic algorithm we describe, *Element Greedy*, is very simple. We start with the empty sequence $S = ()$ and in each step we append an element to S that maximizes the number of newly covered sets. In case of a tie between candidate elements, we pick one candidate uniformly at random from the set of candidates.

Definition 1.13. Let S and T be sequences over the alphabet $[n]$. We define $\xi(S, T)$ to be the set of subsets of $[n]$ that are covered by the concatenation ST of S and T, but not by S. In the case where $T = (i)$ (for some $i \in [n]$) consists of a single element, we write $\xi(S, i)$ instead of $\xi(S, (i))$. $\triangle$

Algorithm 1.14 Element Greedy

Input: $n \in \mathbb{N}$.
Output: A $P(n)$-covering sequence.

$S = ()$.
$Q = P(n)$.
while $Q \neq \emptyset$ **do**
    $Z = \{i \in [n] \setminus \{S[-1]\} : |\xi(S, i)| \text{ is maximal}\}$.
    Choose z uniformly at random from Z.
    $Q = Q \setminus \xi(S, z)$.
    Append the element z to S.
return S.

Running this algorithm a large number of times for $n = 6$, the shortest $P(6)$-covering sequence we found was of length 25 (while in fact $s(6) = 24$). This led to the idea of adding another random element to the algorithm. In each step, with some small probability ρ, instead of choosing an element that maximizes the number of newly covered sets, we add an element to S that is chosen uniformly at random from the set of *all* elements, except for the trailing element of S.

Algorithm 1.15 Randomized Element Greedy

Input: $n \in \mathbb{N}, \rho \in (0, 1)$.
Output: A $P(n)$-covering sequence.

$S = ()$.
$Q = P(n)$.
while $Q \neq \emptyset$ **do**
    with probability ρ **do**
        Choose z uniformly at random from $\{1, \ldots, n\} \setminus \{S[-1]\}$.
    otherwise do
        $Z = \{i \in [n] \setminus \{S[-1]\} : |\xi(S, i)| \text{ is maximal}\}$.
        Choose z uniformly at random from Z.
    $Q = Q \setminus \xi(S, z)$.
    Append the element z to S.
return S.

Running Algorithm 1.15 for $n = 6$ and $\rho = 0.01$, we find a $P(6)$-covering sequence of length 24 on average after about 2618 runs of the algorithm—which takes about one second with our implementation of the algorithm.

It is in fact impossible for Algorithm 1.14 to find a $P(6)$-covering sequence of length 24. It is easy to check whether a given sequence $S = (a_1, \ldots, a_r)$

could have been the output of Algorithm 1.14; for each $i \in \{2, \dots, r\}$ check whether a_i maximizes $|\xi\left((a_1, \dots, a_{i-1}), \cdot\right)|$. In Section 1.2.1 we used a backtracking algorithm to generate all $P(6)$-covering sequences of length 24. For each shortest $P(6)$-covering sequence $(a_1, \dots, a_{24})$ there exists some index d such that a_d does *not* maximize $|\xi\left((a_1, \dots, a_{d-1}), \cdot\right)|$.

The main issue with Algorithms 1.14 and 1.15 is the following; if at some point during the execution of the algorithm, no possible extension of the current sequence S results in a newly covered set, i.e., if

$$|\xi(S, i)| = 0 \text{ for all } i \in [n],$$

the algorithm picks an element at random. If only few subsets are not covered by S, the algorithm will append random elements to S for many iterations, until—by chance—S covers one of the rare not-yet covered subsets. To counteract this issue, we implemented a variant of the heuristic, where whenever no possible extension of the current sequence results in a previously not covered set, we append the minimum amount of elements to the sequence such that at least one new set is covered.

Definition 1.16 (Overlap and non-overlap)**.** Let S be a sequence over the alphabet $[n]$ and let $X \subset [n]$. Let $\mathrm{OL}(S, X)$ be the elements of the maximal final part of S such that all elements of that part are pairwise distinct and included in X. We define the non-overlap between S and X by

$$\mathrm{NOL}(S, X) = X \setminus \mathrm{OL}(S, X).$$

$\triangle$

Example. Let $S = (1, 2, 5, 4, 2, 1, 3, 1)$ and let $X = \{1, 2, 3, 5\}$. Then the overlap $\mathrm{OL}(S, X)$ is $\{1, 3\}$ and the non-overlap $\mathrm{NOL}(S, X)$ is $\{2, 5\}$. $\triangle$

Algorithm 1.17 Heuristic Algorithm

Input: $n \in \mathbb{N}$.
Output: A $P(n)$-covering sequence.

$S = ()$.
$Q = P(n)$.
while $Q \neq \emptyset$ **do**
  if $|\xi(S,i)| = 0$ for all $i \in [n]$ **then**
    $\mathcal{K} = \{X \in Q : \ |\mathrm{NOL}(S,X)| \text{ is minimized}\}$.
    Choose Z uniformly at random from $\mathcal{K}$.
    Let R be a random ordering of $\mathrm{NOL}(S,Z)$.
    $Q = Q \setminus \xi(S,R)$.
    Append the elements of R to S.
  else
    $Z = \{i \in [n] \setminus \{S[-1]\} : |\xi(S,i)| \text{ is maximal}\}$.
    Choose z uniformly at random from Z.
    $Q = Q \setminus \xi(S,z)$.
    Append the element z to S.
return S.

In Table 4 we list the lengths of shortest $P(n)$-covering sequences found by Algorithm 1.17 for the values $n = 8, \ldots, 20$, where the backtracking approach from Section 1.2.1 was not computationally feasible. We also list the number of times we ran the algorithm, and the time it took to find the corresponding sequences. Note that we found a $P(8)$-covering sequence of length 81 within 28 minutes. This seems to have been a very lucky find; afterwards, during many more hours of running Algorithm 1.17, we did not find another such sequence of length 81 or less.

n	$s(n)$	iterations	time
8	≤ 81	554642	28 min
9	≤ 164	1131135	2 h 36 min
10	≤ 331	340095	1 h 47 min
11	≤ 652	103733	1 h 24 min
12	≤ 1287	94963	3 h 39 min
13	≤ 2522	5657	42 min
14	≤ 4913	14274	4 h 2 min
15	≤ 9579	2480	2h 34 min
16	≤ 18628	151	58 min
17	≤ 36384	66	1 h 40 min
18	≤ 70803	51	6 h 42 min
19	≤ 138327	3	53 min
20	≤ 270156	16	16 h 10 min

Table 4: Upper bounds for $s(n)$ obtained by Algorithm 1.17.

The $P(8)$-covering sequence of length 81 we found is given below (commas omitted):

$$\begin{aligned} S_8^* = (&123456781325684173264875126 \\ &387245167358241635841357268 \\ &153742657412834671285743861). \end{aligned}$$

Algorithm 1.17 (and all other algorithms described in this thesis) have been implemented in Python 2.7 and were run with PyPy (`pypy.org`), which features a Just-in-Time compiler, often allowing for faster execution of Python code. The calculations were done on a desktop computer using an Intel i3-4130T dual core processor and 16 gigabytes of memory.

1.2.3 Integer Programming

For given $n, k, N \in \mathbb{N}$, we want to check by computer whether a $P_k(n)$-covering sequence of length N exists.

Below we describe a formulation using binary variables and *linear* constraints. We can then use commercial solvers like *Gurobi* or open-source solvers like *GLPK* (GNU Linear Programming Kit) to find a $P_k(n)$-covering sequence of length N or to prove that no such sequence exists. In order to extend this formulation to $P(n)$-covering sequences, the constraints given below have to be added to the program for all $k \in \{2, \ldots, n\}$.

The number of variables and constraints of the binary integer program formulation we describe is very large, and therefore this approach is only feasible for very small values of n. A binary integer program formulation is still interesting, since many meta-heuristics exist for solving such formulations. Finding the right meta-heuristic might make this approach feasible for larger n.

Let $S = (a_1, \ldots, a_N)$ be a sequence of length N. We define

$$\mathcal{W}_k = \{(i, i+1, \ldots, i+k-1) : 0 \leq i \leq N-k+1\}$$

to be the set of the indices of all k-blocks of S. We call $\mathcal{W}_k$ the set of k-windows of S. We say a k-window $W \in \mathcal{W}_k$ *covers* a k-subset $Q \subseteq [n]$ if the k-block corresponding to W covers Q.

Example. For $N = 8$ and $k = 4$, we have

$$\mathcal{W}_k = \{(1,2,3,4), (2,3,4,5), (3,4,5,6), (4,5,6,7), (5,6,7,8)\}.$$

$\triangle$

For every element i of the alphabet $[n]$ and every position $p \in [N]$ we introduce a binary variable $x_{i,p}$. The variable $x_{i,p}$ will equal 1 if and only if $a_p = i$.

For every set $Q \in P_k(n)$ and every k-window W we introduce a binary variable $y_{W,Q}$. The variable $y_{W,Q}$ will equal 1 if and only if W covers Q.

To make sure we get a valid sequence, the elements of our sequence must be well defined. To this end, we add the following family of constraints:

$$\sum_{i \in [n]} x_{i,p} = 1 \quad \forall p \in [N].$$

Since we want our sequence to cover all sets in $P_k(n)$, we add the following family of constraints enforcing every set Q to be covered at least once:

$$\sum_{W \in \mathcal{W}_k} y_{B,Q} \geq 1 \quad \forall Q \in P_k(n).$$

A k-window can cover at most one k-set. We model this fact by adding the following family of constraints:

$$\sum_{Q \in P_k(n)} y_{W,Q} \leq 1 \quad \forall W \in \mathcal{W}_k.$$

We have to make sure that our x variables and y variables do not contradict each other, i.e., if a y variable encodes that a subset Q is covered by a window W, then the block corresponding to W must be some permutation of the set Q. In the language of our variables, for every set $Q \in P_k(n)$ and every k-window W, we need the following implication:

$$y_{W,Q} = 1 \Rightarrow (\sum_{p \in W} x_{i,p} = 1 \quad \forall i \in Q).$$

Note that the above implication can be written as a family of k implications:

$$(y_{W,Q} = 1 \Rightarrow \sum_{p \in W} x_{i,p} = 1) \quad \forall i \in Q.$$

We express each of these k implications by two *linear* constraints (due to Manfred Scheucher in personal communication):

$$\sum_{p \in W} x_{i,p} - y_{W,Q} \geq 0, \text{ and}$$

$$\sum_{p \in W} x_{i,p} + (|Q| - 1) y_{W,Q} \leq |Q|.$$

The first linear constraint is equivalent to

$$(y_{W,Q} = 1 \Rightarrow \sum_{p \in W} x_{i,p} \geq 1) \quad \forall i \in Q,$$

and the second is equivalent to

$$(y_{W,Q} = 1 \Rightarrow \sum_{p \in W} x_{i,p} \leq 1) \quad \forall i \in Q.$$

We only care about feasibility in our problem. We minimize over the constant 0 to ensure we have a program in correct syntax. The final binary integer programming formulation is the following.

$$
\begin{aligned}
&\max 0 \\
&\text{such that:} \\
&\sum_{i\in[n]} x_{i,p} &&= 1 && \forall p \in [N]. \\
&\sum_{W\in W_k} y_{W,Q} &&\geq 1 && \forall Q \in P_k(n). \\
&\sum_{Q\in P_k(n)} y_{W,Q} &&\leq 1 && \forall W \in \mathcal{W}_k. \\
&\sum_{p\in W} x_{i,p} - y_{W,Q} &&\geq 0 && \forall Q \in P_k(n)\ \forall i \in Q\ \forall W \in \mathcal{W}_k. \\
&\sum_{p\in W} x_{i,p} + (|Q|-1)y_{W,Q} - |Q| &&\leq 0 && \forall Q \in P_k(n)\ \forall i \in Q\ \forall W \in \mathcal{W}_k. \\
&x_{i,p} &&\in \{0,1\} && \forall i \in [n]\ \forall p \in [N]. \\
&y_{W,Q} &&\in \{0,1\} && \forall Q \in P_k(n)\ \forall W \in \mathcal{W}_k.
\end{aligned}
$$

1.3 Asymptotic bounds

A trivial upper bound for $s(n)$ is the following. The sequence created by concatenation of the elements of each subset of $\{1, \ldots, n\}$ (in any order) has length

$$\sum_{i=2}^{n} \binom{n}{i} i = n2^{n-1} - n,$$

and thus $s(n) \leq n2^{n-1} - n$ for all $n \geq 1$. To this date, the shortest known construction for general $n \in \mathbb{N}$ is due to Lipski [14]. Lipski gave a clever construction, using the fact that the powerset of an n-element set can be partitioned into $\binom{n}{\lfloor \frac{n}{2} \rfloor}$ mutually disjoint symmetric chains.

Jukna [11] gave a simpler construction, which is inspired by the technique Lipski used. While both constructions are asymptotically of size

$$\frac{2}{\pi} 2^n (1 + o(1)),$$

Lipski's construction is shorter for all values of n, and for certain values of n, Lipski's construction implies $s(n) < \frac{2}{\pi} 2^n$.

Before we present both constructions, we discuss how to find a partition of the powerset of a finite set into symmetric chains, which is used by both constructions.

Definition 1.18. Let X be a set. The powerset 2^X of X together with the set-inclusion relation $\subseteq$ forms the poset $(2^X, \subseteq)$.
A sequence $(X_1 \subsetneq X_2 \subsetneq \cdots \subsetneq X_k)$, where $X_i \subseteq X$ for all $i \in \{1, \ldots, k\}$ is called a *chain*. If $|X_1| + |X_k| = |X|$ and $|X_{i+1}| = |X_i| + 1$ for all $i \leq k - 1$, the chain is called *symmetric*.
A family of sets in 2^X in which no set is a subset of any of the other sets is called an *anti-chain*. $\triangle$

Sperner's [17] theorem states that the size of a largest anti-chain in the powerset of an n-element set is $\binom{n}{\lfloor n/2 \rfloor}$. From this it follows by Dilworth's [7] theorem that the powerset of an n-element set can be partitioned into $\binom{n}{\lfloor n/2 \rfloor}$ chains. In fact, one can construct such a partition into $\binom{n}{\lfloor n/2 \rfloor}$ *symmetric* chains. This was (in a slightly different context) already known to De Bruijn, Tengenberg and Kruyswijk [2] and others.

We say a collection

$$\begin{aligned} C_1 =&(X_1^{(1)} \subsetneq X_2^{(1)} \subsetneq \cdots \subsetneq X_{r_1}^{(1)}) \\ C_2 =&(X_1^{(2)} \subsetneq X_2^{(2)} \subsetneq \cdots \subsetneq X_{r_2}^{(2)}) \\ &\cdots \\ C_m =&(X_1^{(m)} \subsetneq X_2^{(m)} \subsetneq \cdots \subsetneq X_{r_m}^{(m)}) \end{aligned}$$

of chains is a *partition* of the powerset 2^X of some set X if

$$2^X = \bigcup_{i=1}^{r} \{X_j^{(i)} : 1 \leq j \leq r_i\}.$$

In this case, we also write

$$2^X = \bigcup_{i=1}^{m} C_i.$$

Lemma 1.19 (De Bruijn, Tengenberg and Kruyswijk [2])**.** *Let X be a set consisting of n elements. Then 2^X can be partitioned into $\binom{n}{\lfloor n/2 \rfloor}$ pairwise disjoint symmetric chains.*

Algorithm 1.20 was described in the following formulation by Lipksi [14] and is due to Greene and Kleitman [8]. Lemma 1.19 follows from the proof of correctness of Algorithm 1.20.

Algorithm 1.20 Symmetric Chain Partition (Greene and Kleitman [8])

Input: $n \in \mathbb{N}$.
Output: A partition $\mathcal{C}$ of $2^{[n]}$ into $\binom{n}{\lceil n/2 \rceil}$ mutually disjoint symmetric chains.

```
C = {(∅, {1})}.
for i ∈ {2, ..., n} do
    N = ∅.
    for C = (X_1, ..., X_r) ∈ C do
        Add (X_1, ..., X_r, X_r ∪ {i}) to N.
        if r ≥ 2 then
            Add (X_1 ∪ {i}, ..., X_{r−1} ∪ {i}) to N.
    Set C = N.
Return C.
```

We give a detailed proof for the correctness of Algorithm 1.20.

Proposition 1.21. *Algorithm 1.20 outputs a partition of* $2^{[n]}$ *into* $\binom{n}{\lceil n/2 \rceil}$ *mutually disjoint symmetric chains.*

Proof. We prove this fact by induction. Initially, $\mathcal{C} = \{(\emptyset, \{1\})\}$ is a partition of $2^{[1]} = \{\emptyset, \{1\}\}$ into one symmetric chain. Now let $\mathcal{C}$ be a partition of $2^{[k]}$ into $\binom{k}{\lceil k/2 \rceil}$ mutually disjoint symmetric chains. We claim that after the execution of the for-loop for $i = k+1$, the set $\mathcal{N}$ is a partition of $2^{[k+1]}$ into $\binom{k+1}{\lceil (k+1)/2 \rceil}$ mutually disjoint symmetric chains. First we show that each generated chain is in fact a *symmetric* chain. Let $C = (X_1, \ldots, X_r)$ be a symmetric chain of sets in $2^{[k]}$.
If $r = 1$, k must be even (for odd k, each symmetric chain contains one set of size $\lfloor k/2 \rfloor$ and one set of size $\lceil k/2 \rceil$). Thus, $|X_1| = k/2$, and $(X_1, X_1 \cup \{k+1\})$ is in fact symmetric:

$$|X_1| + |X_1 \cup \{k+1\}| = k+1, \text{ and}$$
$$|X_1 \cup \{k+1\}| = |X_1| + 1.$$

If $r \geq 2$, we have to verify that both chains $(X_1, \ldots, X_r, X_r \cup \{k+1\})$ and $(X_1 \cup \{k+1\}, \ldots, X_{r-1} \cup \{k+1\})$ are symmetric. Obviously both chains fulfill the property that the sizes of their sets increase in steps of exactly one. Since $|X_1| + |X_r| = k$ and $|X_{r-1}| = |X_r| - 1$, we have

$$|X_1| + |X_r \cup \{k+1\}| = k+1, \text{and}$$
$$|X_1 \cup \{k+1\}| + |X_{r-1} \cup \{k+1\}| = (k+1) + (k-1+1) = k+1.$$

We now show that exactly $\binom{k+1}{\lceil (k+1)/2 \rceil}$ symmetric chains are generated. First consider the case where k is odd. Then every symmetric chain in $\mathcal{C}$ consists of at least two sets, and thus for every $C \in \mathcal{C}$, two new chains are created. Thus, for odd k,

$$|\mathcal{N}| = 2\binom{k}{\lceil k/2 \rceil} = \binom{k+1}{\lceil k/2 \rceil + 1} = \binom{k+1}{\lceil \frac{k+1}{2} \rceil}.$$

For even k, there are exactly $\binom{k}{k/2} - \binom{k}{k/2+1}$ chains of size 1 in $\mathcal{C}$; there are $\binom{k}{k/2+1}$ chains in $\mathcal{C}$ that contain a set of size $\frac{k}{2} + 1$, and thus—because of symmetry—also a set of size $\frac{k}{2} - 1$. The remaining $\binom{k}{k/2} - \binom{k}{k/2+1}$ chains must thus be of size 1. For every chain of size 1 in $\mathcal{C}$, only one (instead of two) new chain is created. Thus, for even k,

$$\begin{aligned}|\mathcal{N}| &= \left(\binom{k}{k/2} - \binom{k}{k/2+1}\right) + 2 \cdot \binom{k}{k/2+1} \\ &= \binom{k}{k/2} + \binom{k}{k/2+1} = \binom{k+1}{\lceil \frac{k+1}{2} \rceil}.\end{aligned}$$

We show that the chains in $\mathcal{N}$ form a partition of $2^{[k+1]}$. Each set in $2^{[k+1]}$ that does not contain the element $k+1$ is part of exactly one chain $(X_1, \ldots, X_r, X_r \cup \{i\})$, generated in line 5 of the algorithm. Let $Z \in 2^{[k+1]}$ be a set containing the element $k+1$. There exists a unique chain $C_{Z'} \in \mathcal{C}$ such that $Z' = Z \setminus \{k+1\} \in C_{Z'}$. Consider the code within the for-loop for $C = C_{Z'}$. If Z' is the maximal set of $C_{Z'}$ then Z is contained as the maximal element of the sequence generated in line 5 of the algorithm. If Z' is not the maximal set of $C_{Z'}$ (this implies in particular that $C_{Z'}$ consists of at least two sets), then Z is contained in the chain generated in line 7 of the algorithm. □

It follows from Definition 1.18 that a symmetric chain in the powerset of an n-element set contains exactly one set of size $\frac{n}{2}$ if n is even, and exactly one set of size $\lfloor \frac{n}{2} \rfloor$ and exactly one set of size $\lceil \frac{n}{2} \rceil$ if n is odd. It follows further that in every symmetric chain the number of sets of size less than $\lfloor \frac{n}{2} \rfloor$ is equal to the number of sets of size larger than $\lceil \frac{n}{2} \rceil$.

Example. For $n = 5$, Algorithm 1.20 yields the following partition of the powerset of $\{1, \ldots, 5\}$ into $\binom{5}{2} = 10$ mutually disjoint symmetric chains $C_1, \ldots, C_{10}$.

$$
\begin{aligned}
C_1 &= (\emptyset \subset \{1\} \subset \{1,2\} \subset \{1,2,3\} \subset \{1,2,3,4\} \subset \{1,2,3,4,5\}).\\
C_2 &= (\{5\} \subset \{1,5\} \subset \{1,2,5\} \subset \{1,2,3,5\}).\\
C_3 &= (\{4\} \subset \{1,4\} \subset \{1,2,4\} \subset \{1,2,4,5\}).\\
C_4 &= (\{4,5\} \subset \{1,4,5\}).\\
C_5 &= (\{3\} \subset \{1,3\} \subset \{1,3,4\} \subset \{1,3,4,5\}).\\
C_6 &= (\{3,5\} \subset \{1,3,5\}).\\
C_7 &= (\{3,4\} \subset \{3,4,5\}).\\
C_8 &= (\{2\} \subset \{2,3\} \subset \{2,3,4\} \subset \{2,3,4,5\}).\\
C_9 &= (\{2,5\} \subset \{2,3,5\}).\\
C_{10} &= (\{2,4\} \subset \{2,4,5\}).
\end{aligned}
$$

Note that each chain is of even size and contains exactly one set of size 2 and exactly one set of size 3. △

Definition 1.22. Let $n \in \mathbb{N}$ and let $C = (X_1 \subsetneq \cdots \subsetneq X_r)$ be a chain in $2^{[n]}$. Let $\textsc{Seq}(C)$ denote a sequence obtained from the following construction. Initially let $\textsc{Seq}(C)$ consist of the elements of X_1 in any order. Then, for $i = 2, \ldots, r$ iteratively append to $\textsc{Seq}(C)$ the elements of $X_i \setminus X_{i-1}$ in any order. △

The sequence $\textsc{Seq}(C)$ covers all sets in C. In fact, $\textsc{Seq}(C)$ contains all of the sets in C as an *initial part*, i.e., for every $X \in C$, the block consisting of the first $|X|$ elements of $\textsc{Seq}(C)$ covers X. Note further that $\textsc{Seq}(C)$ consists of exactly $\max\{|X| : X \in C\} = |X_r|$ elements.

Example. Consider the chain

$$C = (\{1,2,3\} \subsetneq \{1,2,3,5,7\} \subsetneq \{1,2,3,5,6,7\} \subsetneq \{1,2,3,4,5,6,7\}).$$

The associated sequence is $\textsc{Seq}(C) = (1,2,3,5,7,6,4)$. △

We are now ready to describe the constructions given by Jukna [11] and Lipski [14].

1.3.1 Jukna's construction

Let $n \in \mathbb{N}$. If n is even, let $k_1 = k_2 = \frac{n}{2}$. If n is odd, let $k_1 = \lfloor \frac{n}{2} \rfloor$ and $k_2 = \lceil \frac{n}{2} \rceil$. We split $[n]$ into two parts by setting

$$S = \{1, 2 \ldots, k_1\} \text{ and}$$
$$T = \{k_1 + 1, \ldots, n\}.$$

Jukna applies Lemma 1.19 to both S and T, obtaining a partition into pairwise disjoint symmetric chains of the powersets of each S and T, respectively:

$$2^S = \bigcup_{i=1}^{m_1} C_i, \text{ where } m_1 = \binom{k_1}{\lfloor \frac{k_1}{2} \rfloor},$$
$$2^T = \bigcup_{j=1}^{m_2} D_j, \text{ where } m_2 = \binom{k_2}{\lfloor \frac{k_2}{2} \rfloor}.$$

Jukna associates a sequence $S_i = \textsc{Seq}(C_i)$ to every chain C_i and a sequence $T_j = \textsc{Seq}(D_j)$ to every chain D_j.

Every subset $R \subseteq [n]$ can be written as $R = E \cup F$, where $E \subseteq S$ and $F \subseteq T$. From the remark after Definition 1.22 we know that there exist two indices $i \in \{1, \ldots, m_1\}$ and $j \in \{1, \ldots, m_2\}$ such that E appears as the initial part of the sequence S_i and that F appears as the initial part of the sequence T_j. We define $\overleftarrow{T_j}$ to be the reverse of the sequence T_j. Note that F appears as the *final* part of $\overleftarrow{T_j}$. Since E appears as initial part of S_j and F as final part of $\overleftarrow{T_j}$, the sequence $\overleftarrow{T_j} S_i$ covers R.
The sequence

$$J(n) = \overleftarrow{T_1} S_1 \overleftarrow{T_1} S_2 \ldots \overleftarrow{T_1} S_{m_1} \overleftarrow{T_2} S_1 \ldots \overleftarrow{T_2} S_{m_1} \ldots \overleftarrow{T_{m_2}} S_1 \ldots \overleftarrow{T_{m_2}} S_{m_1}$$

contains the sequence $\overleftarrow{T_j} S_i$ for all $i \in \{1, \ldots, m_1\}$ and $j \in \{1, \ldots, m_2\}$ and thus covers all subsets of $[n]$.

1.3.2 The length of Jukna's construction

As first noticed by Markus Hartmair, Jukna [11] made a small miscalculation when he attempted to show that $|J(n)| \sim 2^n \frac{2}{\pi}$ as n tends to infinity. In this section we give an exact formula for the length of $J(n)$, from which we then derive the desired asymptotic behaviour of $|J(n)|$.

For every $j \in \{1, \ldots, m_2\}$, the sequence J contains exactly m_1 copies of the sequence $\overleftarrow{T_j}$ and for every $i \in \{1, \ldots, m_1\}$, the sequence J contains exactly m_2 copies of the sequence S_i. Thus,

$$|J(n)| = m_1 \left(|T_1| + \cdots + |T_{m_2}|\right) + m_2 \left(|S_1| + \cdots + |S_{m_1}|\right). \tag{1}$$

Definition 1.23. For $k \in \mathbb{N}$ let $\mathcal{C} = \{C_1, \ldots, C_m\}$ be a partition of $2^{[k]}$ into $m = \binom{k}{\lfloor \frac{k}{2} \rfloor}$ symmetric, mutually disjoint chains. Let $b(k)$ denote the sum of the lengths of the sequences $\textsc{Seq}(C_i)$, i.e., let

$$b(k) = \sum_{i=1}^{m} |\textsc{Seq}(C_i)|.$$

$\triangle$

Note that $|T_1| + \cdots + |T_{m_2}| = b(k_2)$ and $|S_1| + \cdots + |S_{m_1}| = b(k_1)$. Lemma 1.24 gives an exact formula for $b(k)$.

Lemma 1.24. *We have*

$$b(k) = k + \sum_{j=\lceil k/2 \rceil}^{k-1} j \left(\binom{k}{j} - \binom{k}{j+1} \right). \tag{2}$$

Proof. Let $\mathcal{C} = \{C_1, \ldots C_m\}$ be a partition of $2^{[k]}$ into mutually disjoint symmetric chains. Since the length of each sequence $\textsc{Seq}(C_i)$ is equal to the size of the largest set in C_i, we have

$$b(k) = \sum_{C \in \mathcal{C}} \max\left\{|X| : \ X \in C\right\}.$$

Since $\mathcal{C}$ is a partition of the powerset of $[k]$, there is exactly one chain in $\mathcal{C}$ that contains the set $\{1, \ldots, k\}$. The length of the corresponding sequence contributes the term k to the sum in equation (2).

We claim that for each $j \in \{\lceil k/2 \rceil, \ldots, k-1\}$, the number of chains $C \in \mathcal{C}$ with $\max\{|X| : X \in C\} = j$ is

$$\binom{k}{j} - \binom{k}{j+1}.$$

There are $\binom{k}{j+1}$ subsets of $[k]$ of size $j+1$. For each subset of size $j+1$ there is exactly one chain in $\mathcal{F}$ that contains this subset. Each of these $\binom{k}{j+1}$ chains also contains exactly one subset of $[k]$ of size j. Therefore, the remaining $\binom{k}{j} - \binom{k}{j+1}$ subsets of size j each appear in a chain whose largest set has size j. Since $\mathcal{C}$ is a partition of the powerset of $[k]$, there are *exactly* $\binom{k}{j} - \binom{k}{j+1}$ chains $C \in \mathcal{C}$ with $\max\{|X| : X \in C\} = j$.

For every such chain C, the corresponding sequence $\textsc{Seq}(C)$ has length j, in total contributing the term $j\left(\binom{k}{j} - \binom{k}{j+1}\right)$ to the sum in equation (2). This concludes the proof. $\square$

Using the representation for $b(k)$ from Lemma 1.24, the sequence $(b(k))_{k\geq 1}$ can be seen to be equal to the integer sequence *A014314* in Sloane's *OEIS* [16]. V. Kotěšovec [16] used the generating function of that sequence to give the following simplified formula for $b(k)$.

Lemma 1.25 (V. Kotěšovec)**.** *For all $k \in \mathbb{N}$,*

$$b(k) = 2^{k-1} + \begin{cases} (k-1)\binom{k-1}{k/2} & \text{if } k \text{ is even,} \\ \frac{k-1}{2}\binom{k}{\lfloor k/2 \rfloor} & \text{if } k \text{ is odd.} \end{cases}$$

Since for even k, we have $\binom{k}{k/2} = 2\binom{k-1}{k/2}$, this simplifies to

$$b(k) = 2^{k-1} + \frac{k-1}{2}\binom{k}{\lfloor k/2 \rfloor}.$$

Using the formula given in Lemma 1.25, we obtain an exact formula for $|J(n)|$ from equation (1):

$$\begin{aligned} |J(n)| = &\binom{k_1}{\lfloor k_1/2 \rfloor}\left(2^{k_2-1} + \frac{k_2-1}{2}\binom{k_2}{\lfloor k_2/2 \rfloor}\right) \\ &+\binom{k_2}{\lfloor k_2/2 \rfloor}\left(2^{k_1-1} + \frac{k_1-1}{2}\binom{k_1}{\lfloor k_1/2 \rfloor}\right), \end{aligned} \tag{3}$$

where $k_1 = k_2 = \frac{n}{2}$ if n is even, and $k_1 = \lfloor \frac{n}{2} \rfloor$ and $k_2 = \lceil \frac{n}{2} \rceil$ if n is odd.
As $k \to \infty$,

$$\binom{k}{\lfloor k/2 \rfloor} = 2^k \sqrt{\frac{2}{\pi k}}\,(1 + o(1)),$$

and we can thus easily obtain the asymptotic behaviour of $|J(n)|$ from (3).

As $n \to \infty$, we have

$$\begin{aligned}|J(n)| &\sim 2^{k_1}\sqrt{\frac{2}{\pi k_1}} \cdot 2^{k_2-1}\sqrt{\frac{2k_2}{\pi}} + 2^{k_2}\sqrt{\frac{2}{\pi k_2}} \cdot 2^{k_1-1}\sqrt{\frac{2k_1}{\pi}} \\ &= 2^n\sqrt{\frac{k_2}{k_1}}\frac{1}{\pi} + 2^n\sqrt{\frac{k_1}{k_2}}\frac{1}{\pi} \\ &= 2^n\frac{1}{\pi}\left(\sqrt{\frac{k_2}{k_1}} + \sqrt{\frac{k_1}{k_2}}\right).\end{aligned}$$

Since $\sqrt{\frac{k_2}{k_1}} + \sqrt{\frac{k_1}{k_2}} = 2 + o(1)$ as $n \to \infty$, we have $|J(n)| \sim 2^n\frac{2}{\pi}$.

Below we give some values of $|J(n)|$ and $|J(n)| - 2^n\frac{2}{\pi}$, showing that in general $|J(n)| > 2^n\frac{2}{\pi}$. This holds in fact for all $n \geq 4$, but requires a lot of calculation to show.

n	$\lvert J(n)\rvert$	$\lvert J(n)\rvert - 2^n\frac{2}{\pi}$	$\frac{\lvert J(n)\rvert - 2^n\frac{2}{\pi}}{\lvert J(n)\rvert}$
4	12	1.81	0.15
5	23	2.63	0.11
6	42	1.26	0.03
7	93	11.51	0.12
8	204	41.03	0.20
9	387	60.05	0.16
100	$9.25 \cdot 10^{29}$	$1.18 \cdot 10^{29}$	0.13
250	$1.26 \cdot 10^{75}$	$1.06 \cdot 10^{74}$	0.08
500	$2.24 \cdot 10^{150}$	$1.53 \cdot 10^{150}$	0.07
1000	$7.18 \cdot 10^{300}$	$3.62 \cdot 10^{299}$	0.05

Table 5: Approximate values of $|J(n)|$, $|J(n)| - 2^n\frac{2}{\pi}$ and the ratio $\frac{|J(n)| - 2^n\frac{2}{\pi}}{|J(n)|}$ for $n = 4, \ldots, 9, 100, 250, 500, 1000$.

1.3.3 Lipski's construction

Lipski uses the partition of the powerset of $[n]$ into mutually disjoint symmetric chains as an intermediate building block to construct what he calls a *special collection of permutations*.

Definition 1.26. Let $k, t \in \mathbb{N}$ and let $R_1, \ldots, R_t$ be sequences, each encoding a permutation of the set $[k]$, i.e., the sequences R_i are each of length k and contain every element $j \in [k]$ exactly once.

The collection $R_1, \dots, R_t$ is called a *special collection of k-permutations* if every subset of $[k]$ appears as an initial or final part of at least one of the sequences $R_1, \dots, R_t$. $\triangle$

Using the symmetric chain partition of $2^{[k]}$, Lipski shows the following Lemma.

Lemma 1.27. *Let $k \in \mathbb{N}$. There exists a special collection of k-permutations $R_1, \dots, R_t$, where*

$$t = t(k) = \begin{cases} \frac{1}{2}\binom{k}{k/2} & \text{if } k \text{ is even,} \\ \frac{1}{2}\binom{k}{\lfloor k/2 \rfloor}\left(1 + \frac{1}{k}\right) & \text{if } k \text{ is odd.} \end{cases}$$

Proof. First, let k be even. We start by partitioning $2^{[k]}$ into mutually disjoint symmetric chains,

$$2^{[k]} = \bigcup_{i=1}^{m} C_i, \text{ where } m = \binom{k}{\frac{k}{2}}.$$

For each $i \in \{1, \dots, m\}$ we set $S_i = \textsc{Seq}(C_i)$. Since every chain C_i contains exactly one set of size $\frac{k}{2}$, there exists a bijection

$$\Phi : \{S_i : \ i \in \{1, \dots, m\}\} \to \binom{[k]}{\frac{k}{2}}, \tag{4}$$

mapping each sequence S_i to the $\frac{k}{2}$-subset that is covered by the initial part of S_i. Each of the sequences S_i consists of at most k elements. We *extend* each sequence S_i of length less than k to a sequence of k elements, by appending to S_i (in arbitrary order) the elements of $\{1, \dots, k\}$ not appearing in S_i. Note that every sequence S_i now contains every element in $\{1, \dots, k\}$ exactly once. By extending the sequences S_i, the first $k/2$ elements remain unchanged, and thus the mapping (4) is still a well defined bijection. Since k is even, for a fixed element $\eta \in [k]$, exactly half of all $\frac{k}{2}$-subsets of $[k]$ contain η. Without loss of generality, the initial part of the first $t = \frac{1}{2}\binom{k}{k/2}$ chains contains the element 1; for all $i \in \{1, \dots, t\}$ we have $1 \in \Phi(S_i)$.

The complement of each set $\Phi(S_i)$ in $[k]$ is also of size $k/2$, and thus for every $i \in \{1, \dots, t\}$ there exists a unique index $j(i) \in \{t+1, \dots, 2t\}$, such that

$$\Phi(S_{j(i)}) = \{1, \dots, k\} \setminus \Phi(S_i).$$

For each $i \in \{1, \dots, t\}$ let R_i consist of the first $k/2$ elements of S_i followed by the first $k/2$ elements of $S_{j(i)}$ in reverse order.

We claim that the resulting sequences $R_1, \dots, R_t$ form a special collection of k-permutations. Let $X \subseteq [k]$. If $|X| \le k/2$ and $1 \in X$, then X appears as initial part of one of the sequences R_i. If $|X| \leq k/2$ and $1 \notin X$, then X appears as final part of one of the sequences R_i. Thus, each subset $X \subset [n]$ with $|X| \leq k/2$ appears as final or initial part in one of the sequences R_i. Let $Y \subseteq X$ with $|Y| > k/2$. Since $X = [k] \setminus Y$ appears in some R_j as initial (final) part, the set Y appears in the same R_j as final (initial) part.

Let now k be even. Then $k-1$ is odd and we use the above construction to obtain a special collection of $(k-1)$-permutations $R_1, \dots, R_{t(k-1)}$ of size $t(k-1) = \frac{1}{2}\binom{k-1}{(k-1)/2}$.
For each $i \in \{1, \dots, t(k-1)\}$ replace R_i by the two sequences

$$R_i^{+k} = R_i + (k) \text{ and}$$
$$R_i^{-k} = (k) + R_i.$$

We claim that the resulting collection of sequences is a special collection of k-permutations. If a subset $X \subset \{1, \dots, k-1\}$ appears as *initial* part of some sequence R_i, then X appears as initial part of R_i^{+k}. Similarly, if X appears as *final* part of R_i, then X appears as final part of R_i^{-k}. Let $Y \subseteq \{1, \dots, k\}$ with $k \in Y$. We write $Y = X \cup \{k\}$. Since X appears as initial (final) part of some R_i, the set Y appears as initial (final) part of R_i^{-k} (of R_i^{+k}). Thus, the sequences $R_1^{-k}, R_1^{+k}, \dots, R_{t(k-1)}^{-k}, R_{t(k-1)}^{+k}$ form a special collection of k-permutations. The number of sequences in this collection is

$$t(k) = 2t(k-1) = \frac{1}{2}\binom{k}{\lfloor k/2 \rfloor}\left(1 + \frac{1}{k}\right).$$

□

Remark. Note that in the previous Lemma, for the case where k is even, $t(k)$ is the smallest possible size of a special collection of k-permutations. To see this, note that there are $\binom{k}{k/2}$ $\frac{k}{2}$-subsets of $[k]$. Each sequence in a special collection of k-permutations can contain at most two of these subsets as initial or final part. Thus a special collection of k-permutations must consist of at least $\frac{1}{2}\binom{k}{k/2}$ sequences. Lipski asked whether for all odd k a special collection of k-permutations of size $\left\lceil \frac{1}{2}\binom{k}{\lceil k/2 \rceil} \right\rceil$ exists, and gave an example for $k = 5$, where a special collection of k-permutations of size $\left\lceil \frac{1}{2}\binom{5}{3} \right\rceil = 5$ in fact exists:

$$(1, 2, 3, 4, 5)$$
$$(2, 3, 5, 1, 4)$$
$$(3, 4, 2, 1, 5)$$
$$(1, 3, 4, 2, 5)$$
$$(2, 4, 1, 3, 5)$$

For $k = 7$, using a randomized search algorithm, we found a special collection of k-permutations of size $\left\lceil \frac{1}{2}\binom{7}{4} \right\rceil = 18$:

$$\begin{array}{ll} (5,4,1,2,3,7,6) & (2,4,7,3,1,6,5) \\ (6,4,1,5,7,2,3) & (7,3,1,4,6,2,5) \\ (4,5,3,7,2,6,1) & (6,4,2,3,7,5,1) \\ (1,3,5,4,7,2,6) & (1,2,4,6,3,7,5) \\ (6,4,5,3,7,2,1) & (1,5,2,6,4,7,3) \\ (2,7,5,1,3,4,6) & (3,1,2,4,5,7,6) \\ (4,5,7,1,2,3,6) & (3,4,2,5,6,7,1) \\ (6,7,4,1,2,3,5) & (5,2,4,7,6,3,1) \\ (3,6,5,2,1,7,4) & (5,2,6,7,3,1,4) \end{array}$$

$\triangle$

Lipski now constructs a $P(n)$-covering sequence from the special collection of k-permutations. First consider the case where $n = 2k$ is even. Let $\mathcal{R}$ be a special collection of k-permutations and let $\mathcal{Q}$ be the collection of sequences obtained from $\mathcal{R}$ by incrementing every element in each sequence in $\mathcal{R}$ by k:

$$\mathcal{Q} = \{(a_1 + k, \ldots, a_k + k) : (a_1, \ldots, a_k) \in \mathcal{R}\}.$$

We write $\mathcal{R} = \{R_1, \ldots, R_{t(k)}\}$ appending $\mathcal{Q} = \{Q_1, \ldots, Q_{t(k)}\}$. Note that every subset of $\{k+1, \ldots, 2k\}$ appears as final or initial part in at last one sequence in $\mathcal{Q}$. Let X be a subset of $\{1, \ldots, n\} = \{1, \ldots, 2k\}$. Then $X = E \cup F$, where $E \subset \{1, \ldots, k\}$ and $F \subset \{k+1, \ldots, 2k\}$. The set E appears as initial or final part of some sequence R_i, and F appears as initial or final part of some sequence Q_j. Thus, one of the sequences

$$\begin{array}{c} R_i Q_j \\ Q_j R_i \\ R_i \overleftarrow{Q_j} \\ \overleftarrow{Q_j} R_i \end{array}$$

must cover X. The goal is to create a sequence that contains each of the sequences $R_i Q_j$, $Q_j R_i$, $R_i \overleftarrow{Q_j}$, $\overleftarrow{Q_j} R_i$ as subsequence for all $i, j \in \{1, \ldots, t(k)\}$. For $0 \leq r \leq t - 1$, Lipski defines

$$\begin{aligned} A_r &= R_1 Q_{\mathrm{shift}_r(1)} R_2 Q_{\mathrm{shift}_r(2)} \ldots R_t Q_{\mathrm{shift}_r(t)}, \text{ and} \\ B_r &= R_1 \overleftarrow{Q}_{\mathrm{shift}_r(1)} R_2 \overleftarrow{Q}_{\mathrm{shift}_r(2)} \ldots R_t \overleftarrow{Q}_{\mathrm{shift}_r(t)}, \end{aligned}$$

where shift_r denotes the function performing a cyclic shift along the indices $1, \ldots, t$ by r units, i.e., for $i \in \{1, \ldots, t\}$

$$\mathrm{shift}_r(i) = 1 + (i + r - 1 \bmod t).$$

Example. For $t = 4$, we have

$$\begin{aligned} A_0 &= R_1Q_1R_2Q_2R_3Q_3R_4Q_4, \\ A_1 &= R_1Q_2R_2Q_3R_3Q_4R_4Q_1, \\ A_2 &= R_1Q_3R_2Q_4R_3Q_1R_4Q_2, \\ A_3 &= R_1Q_4R_2Q_1R_3Q_2R_4Q_3. \end{aligned}$$

$\triangle$

It is easy to see that for each pair $i, j \in \{1, \ldots, t\}$, the sequence R_iQ_j is contained in one of the sequences A_r. Further, for each $j \in \{1, \ldots, t\}$ and each $i \in \{2, \ldots, t\}$, the sequence Q_jR_i is contained in one of the sequences A_r. Note that for each $j \in \{1, \ldots, t\}$, the sequence R_jQ_1 does *not* appear in any sequence A_r, but *does* appear in the the concatenation of the sequences $A_0, \ldots, A_{t-1}$. To see this, note that for $i \in \{0, \ldots, t-2\}$, the sequence A_iA_{i+1} contains the sequence Q_iR_1 (let here $Q_0 = Q_t$) because Q_i is the final block of A_i and R_1 is the first block of A_{i+1}. The only sequence of the form Q_jR_i that is not contained in $A_0 \ldots A_{t-1}$ is $Q_{t-1}R_1$. Since the sequence A_{t-1} ends with the sequence Q_{t-1}, this is fixed by adding R_1 to the concatenation of the sequences. The sequence $A_0A_1 \ldots A_{t-1}R_1$ thus contains all of the sequences R_iQ_j and Q_jR_i. Similarly, $B_0B_1 \ldots B_{t-1}R_1$ contains all of the sequences $R_i\overleftarrow{Q_j}$ and $\overleftarrow{Q_j}R_i$.
Since the sequence B_0 starts with R_1, the sequence

$$L(n) = L(2k) = A_0A_1 \ldots A_{t-1}B_0B_1 \ldots B_{t-1}R_1$$

contains both the sequences $A_0A_1 \ldots A_{t-1}R_1$ and $B_0B_1 \ldots B_{t-1}R_1$ and thus covers all subsets of $[n]$.

In the case where $n = 2k + 1$ is odd, for $t = t(k)$, let the sequences A_r and B_r be defined as before. Further, for each $r \in \{0, \ldots, t-1\}$, we define sequences obtained from A_r and B_r by inserting the element n after each occurrence of one of the sequences $R_1, \ldots, R_t$:

$$\begin{aligned} A_r^* &= R_1nQ_{\mathrm{shift}_r(1)}R_2nQ_{\mathrm{shift}_r(2)} \ldots R_tnQ_{\mathrm{shift}_r(t)}, \text{ and} \\ B_r^* &= R_1n\overleftarrow{Q}_{\mathrm{shift}_r(1)}R_2n\overleftarrow{Q}_{\mathrm{shift}_r(2)} \ldots R_tn\overleftarrow{Q}_{\mathrm{shift}_r(t)}. \end{aligned}$$

The sequence $L(2k) = A_0A_1 \ldots A_{t-1}B_0B_1 \ldots B_{t-1}R_1$ covers all subsets of $[2k+1]$ not containing the element $n = 2k + 1$, and the sequence

$$L(2k)^* = A_0^*A_2^* \ldots A_{t-1}^*B_0^*B_2^* \ldots B_{t-1}^*nR_1$$

covers all subsets of $[2k+1]$ containing the element $n = 2k+1$. Since $L(2k)$ ends with the sequence R_1, and $L(2k)^*$ starts with the same sequence R_1, we can omit one occurrence of R_1 in their concatenation and thus the sequence

$$L(2k+1) = A_0 A_1 \ldots A_{t-1} B_0 B_1 \ldots B_{t-1} A_0^* A_2^* \ldots A_{t-1}^* B_0^* B_2^* \ldots B_{t-1}^* n R_1$$

covers all subsets of $[n]$.

1.3.4 The length of Lipski's construction.

In the case where $n = 2k$ is even, $L(2k)$ consists of $t = t(k)$ sequences A_i and B_i—each consisting of $2t(k)k$ elements—plus the trailing sequence R_1 (which is of length k). In total,

$$|L(2k)| = 2t \cdot 2tk + k = 4t^2 k + k.$$

In the case where $n = 2k+1$ is odd, $L(2k+1)$ consists of $t = t(k)$ sequences A_i and B_i, each consisting of $2tk$ elements, and t sequences A_i^* and B_i^*—each consisting of $2tk + t$ elements—plus the trailing sequence nR_1 (which is of length $k+1$). In total,

$$|L(2k+1)| = 4t^2 k + 2t(2tk + t) + k + 1 = 8t^2 k + 2t^2 + k + 1.$$

We show that $|L(n)| \sim 2^n \frac{2}{\pi}$. Since $\binom{k}{\lfloor k/2 \rfloor} = 2^k \sqrt{\frac{2}{\pi k}}\,(1 + o(1))$ as k tends to infinity, we have

$$\begin{aligned} t(k) &= \begin{cases} \frac{1}{2}\binom{k}{k/2} & \text{if } k \text{ is even} \\ \frac{1}{2}\binom{k}{\lfloor k/2 \rfloor}\left(1 + \frac{1}{k}\right) & \text{if } k \text{ is odd} \end{cases} \\ &\sim \frac{1}{2} 2^k \sqrt{\frac{2}{\pi k}}. \end{aligned}$$

For even $n = 2k$, we thus have

$$|L(n)| \sim 4t(k)^2 k = 2^{2k} \frac{2}{\pi k} k = 2^n \frac{2}{\pi}.$$

For odd $n = 2k+1$, we have

$$|L(n)| \sim 8t(k)^2 k = 2 \cdot 2^{2k} \frac{2}{\pi k} k = 2^n \frac{2}{\pi}.$$

In fact, whenever k is even and $n = 2k$ or $n = 2k+1$ (which holds for $n = 8, 9, 12, 13, 16, 17, \ldots$), we have

$$|L(n)| < 2^n \frac{2}{\pi}.$$

It follows from a result mentioned by Banakh *et al.* [1] that for all even $k \geq 4$,

$$\binom{k}{k/2} \leq 2^k \sqrt{\frac{2}{\pi k}} \left(1 - \frac{2}{9k}\right),$$

and thus

$$\begin{aligned} |L(2k)| = 4t^2k + k &\leq 2^n \frac{2}{\pi} \left(1 - \frac{2}{9k}\right)^2 + k \\ &= 2^n \frac{2}{\pi} - 2^n \frac{2}{\pi} \frac{2}{9k} + 2^n \frac{2}{\pi} \frac{4}{36k^2} + k. \end{aligned}$$

For $n \geq 8$,

$$-2^n \frac{2}{\pi} \frac{2}{9k} + 2^n \frac{2}{\pi} \frac{4}{36k^2} + k < 0,$$

proving the claim.
For the case where $n = 2k + 1$ (with k even),

$$\begin{aligned} |L(2k+1)| &= 8t^2k + 2t^2 + k + 1 \\ &\leq 2^n \frac{2}{\pi} \left(1 - \frac{1}{4.5k}\right)^2 + \frac{1}{4} 2^n \frac{2}{\pi k} \left(1 - \frac{1}{4.5k}\right)^2 + k + 1. \end{aligned}$$

Similar to before, this upper bound can be shown to be less than $2^n \frac{2}{\pi}$ for all $n \geq 9$.

Although Lipski's construction is currently the shortest known construction for general $n \in \mathbb{N}$, it seems to be far from optimal. To demonstrate this, we introduce a simple greedy algorithm of deterministic nature.

1.3.5 Overlap-Greedy Algorithm

In this section we introduce a very simple greedy algorithm. While this algorithm is outperformed by the heuristic approach described in Section 1.2.2, the greedy algorithm is completely deterministic, and one might thus have a chance to analyze the lengths of the $P(n)$-covering sequences it creates. Evidence suggests that the $P(n)$-covering sequences generated by this algorithm are of length strictly less than $\frac{2}{\pi} 2^n$ for all n.

Algorithm 1.29 starts with the empty sequence $S = ()$ and iteratively adds the least amount of elements needed such that S covers a previously not covered set. These elements are added to S in increasing order. If the choice of elements is not unique, the algorithm picks the elements to be added in such a way that the newly covered set is minimal with respect to the size-lexicographic order, which is defined below.

Definition 1.28 (Size-lexicographic order)**.** Let $\mathcal{P}$ be a family of subsets of $[n]$. We define the relation $\leq_{sl}$ on $\mathcal{P}$ as follows. For $X, Y \in \mathcal{P}$:

$$X \leq_{sl} Y \iff |X| < |Y| \text{ or } (|X| = |Y| \text{ and } X \leq_{lex} Y).$$

$\triangle$

Example. The size-lexicographic order on $P(3)$ is the following:

$$\{1\} \leq_{sl} \{2\} \leq_{sl} \{3\} \leq_{sl} \{1,2\} \leq_{sl} \{1,3\} \leq_{sl} \{2,3\} \leq_{sl} \{1,2,3\}.$$

$\triangle$

We repeat the definition for the overlapping, and non-overlapping part between a sequence and a set.

Definition (Overlap and non-overlap)**.** Let S be a sequence over the alphabet $[n]$ and let $X \subset [n]$. Let $\mathrm{OL}(S, X)$ be the elements of the maximal final part of S such that all elements of that part are pairwise distinct and included in X. We define the non-overlap between S and X by $\mathrm{NOL}(S, X) = X \setminus \mathrm{OL}(S, X)$.

Example. Let $S = (1, 2, 5, 4, 2, 1, 3, 1)$ and let $X = \{1, 2, 3, 5\}$. Then the overlap $\mathrm{OL}(S, X)$ is $\{1, 3\}$ and the non-overlap $\mathrm{NOL}(S, X)$ is $\{2, 5\}$. $\triangle$

Algorithm 1.29 Overlap-Greedy

Input: $n \in \mathbb{N}$.

Output: A $P(n)$-covering sequence.

$S = ()$.
$P = P(n)$.
while $P \neq \emptyset$ **do**
    $Z = \{\ Y \in P :\ |\mathrm{NOL}(S, Y)| \text{ is minimized}\}$.
    $X = min_{sl} Z$.
    Append the elements of $\mathrm{NOL}(X)$ to S in increasing order.
    $P = P \setminus \{X\}$.
return S.

Note that by adding elements to the current sequence S, the updated sequence S might cover more than one previously not covered set. Such sets Y will be detected in the subsequent iterations of the while-loop, where the non-overlap $\mathrm{NOL}(S, Y)$ will be the empty set.

For $n \in \mathbb{N}$, let G_n denote the length of the $P(n)$-covering sequence generated by Algorithm 1.29, and let L_n denote the length of Lipski's construction. Table 6 compares G_n to L_n and to the quantity $2^n - 1$ for $n = 3, \ldots, 20$.

n	L_n	G_n	G_n/L_n	$G_n/(2^n-1)$
3	12	4	0.3333	0.5714
4	10	9	0.9000	0.6000
5	21	15	0.7143	0.4839
6	51	33	0.6471	0.5238
7	108	60	0.5556	0.4724
8	148	123	0.8311	0.4824
9	311	230	0.7395	0.4501
10	725	481	0.6634	0.4702
11	1518	909	0.5988	0.4441
12	2406	1790	0.7440	0.4371
13	5007	3470	0.6930	0.4236
14	11207	6714	0.5991	0.4098
15	23208	13161	0.5671	0.4017
16	39208	25686	0.6551	0.3919
17	80859	50317	0.6223	0.3839
18	176409	98553	0.5587	0.3760
19	362610	193994	0.5350	0.3700
20	635050	382160	0.6018	0.3645

Table 6: Comparison of the length of Lipski's construction to the length of the sequences generated by the Overlap-Greedy Algorithm for small values of n.

Note that for $n \geq 10$, the ratio $G_n/(2^n - 1)$ seems to be strictly decreasing. Figure 1 plots $G_n/(2^n - 1)$ for $n = 3, \ldots 20$. The red line plots the value $\frac{2}{\pi}$. The family $P(n)$ consists of $2^n - 1$ sets and thus the ratio $G_n/(2^n - 1)$ can be interpreted as the average size of $\mathrm{NOL}(S, X)$ (taken over all steps of Algorithm 1.29). Showing that the mean length of the non-overlap tends to zero as $n \to \infty$, would prove $G_n = o(2^n)$ and thus $s(n) = o(2^n)$, which would be an asymptotic improvement over Lipski's result. Even showing that for sufficiently large n, the average size of $\mathrm{NOL}(S, X)$ is bounded from above by some constant $c < \frac{2}{\pi}$ would imply $s(n) \leq c2^n$ (for large n).

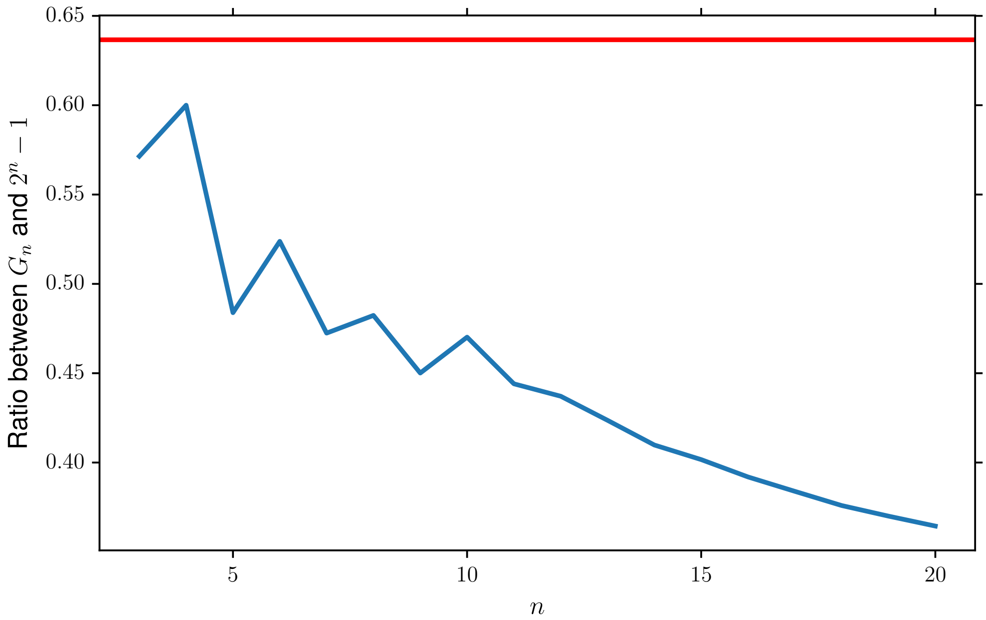


Figure 1: Plot of the ratio $G_n/(2^n - 1)$ for $n = 3, \ldots, 20$. The red line corresponds to the ratio $2/\pi$.

2 Generalization to arithmetic progressions

2.1 Introduction

A natural generalization of the problem described in Chapter 1 is to ask for the length of a shortest sequence over the alphabet $[n] = \{1, \ldots, n\}$, covering each nonempty subset of $\{1, \ldots, n\}$ by an arithmetic progression. We make this precise in Definitions 2.2 and 2.3.

Definition 2.1. Let $a, k, d \in \mathbb{N}$. The set $A = \{a, a+d, a+2d, \ldots, a+(k-1)d\}$ is called an (arithmetic) k-progression. We say A has *common distance* d. $\triangle$

In Chapter 1 we studied the length of a shortest sequence $S = (a_1, \ldots, a_{|S|})$ over the alphabet $[n]$ such that for all $1 \leq k \leq n$, every k-subset of $\{1, \ldots, n\}$ is covered by a k-block $(a_r, a_{r+1}, \ldots, a_{r+k-1})$; that means we required the index set $\{r, \ldots, r+k-1\}$ to be a k-progression with common distance 1. The generalization discussed in this chapter is equivalent to asking the same question as before, but allowing sets to be covered by arithmetic progressions of *any* common distance.

If $S = (a_1, a_2, \ldots, a_N)$ is a sequence over the alphabet $[n]$, we can represent S by the n-colouring f of $[N]$ defined by

$$f(1) = a_1, f(2) = a_2, \ldots, f(N) = a_N,$$

and vice-versa. In problems concerning arithmetic progressions, most authors talk about n-colourings of an integer-interval, rather than about sequences over the alphabet $[n]$. We follow this convention and state the problem in the language of colourings.

Definition 2.2. Let $n, N \in \mathbb{N}$ $(n \leq N)$ and let $f : [N] \to [n]$ be an n-colouring of $[N]$. Let $R \in \binom{[n]}{k}$ be a k-subset of $[n]$. We say a k-progression A in $[N]$ is *R-coloured* if $\{f(a) : a \in A\} = R$. We say f *covers* R if there is a k-progression that is R-coloured. If $\mathcal{P}$ is a family of subsets of $[n]$, we say f *covers* $\mathcal{P}$, if f covers all sets in $\mathcal{P}$.

Remark. Note that since A consists of k elements and $|R| = k$, the condition $\{f(a) : a \in A\} = R$ implies that no two elements of A are coloured with the same colour.

$\triangle$

Example. The 6-colouring

$$f = (6, 6, 5, 2, 1, 3, 4, 3, 2, 5, 6, 1, 3, 4)$$

of the interval $\{1, 2, \dots, 14\}$ covers $P(6)$ because for every subset $R \subseteq \{1, \dots, 6\}$ there is a progression in $\{1, 2, \dots, 14\}$ that is R-coloured; we give examples for some subsets:

$$\begin{aligned}
\{1,4,6\}: &\quad (6, \mathbf{6}, 5, 2, 1, 3, \mathbf{4}, 3, 2, 5, 6, \mathbf{1}, 3, 4)\\
\{1,3,5\}: &\quad (6, 6, 5, 2, 1, 3, 4, \mathbf{3}, 2, \mathbf{5}, 6, \mathbf{1}, 3, 4)\\
\{1,2,3,4\}: &\quad (6, 6, 5, \mathbf{2}, \mathbf{1}, \mathbf{3}, \mathbf{4}, 3, 2, 5, 6, 1, 3, 4)\\
\{1,2,4,6\}: &\quad (6, 6, 5, 2, \mathbf{1}, 3, \mathbf{4}, 3, \mathbf{2}, 5, \mathbf{6}, 1, 3, 4)\\
\{1,2,3,4,5,6\}: &\quad (6, \mathbf{6}, \mathbf{5}, \mathbf{2}, \mathbf{1}, \mathbf{3}, \mathbf{4}, 3, 2, 5, 6, 1, 3, 4)
\end{aligned}$$

△

Definition 2.3. For $n \in \mathbb{N}$, let $a(n)$ denote the smallest positive integer such that there exists an n-colouring f of $[a(n)] = \{1, 2, \dots, a(n)\}$ that covers $P(n) = 2^{[n]} \setminus \{\emptyset\}$, i.e., for every $R \subseteq [n]$ there exists an arithmetic $|R|$-progression in $[a(n)]$ that is R-coloured.

For $n, k \in \mathbb{N}$ (where $k \leq n$), let $a(n, k)$ denote the smallest positive integer such that there exists an n-colouring f of $[a(n, k)] = \{1, 2, \dots, a(n, k)\}$ covering $P_k(n) = \binom{[n]}{k}$, i.e., for every $R \in \binom{[n]}{k}$ there exists a k-progression in $[a(n, k)]$ that is R-coloured.

△

Remark. Anti Van-der-Waerden Numbers.
An arithmetic k-progression whose elements are coloured with k distinct colours is called a *rainbow k-progression.* A well-studied problem in Ramsey Theory concerning rainbow progressions is the following:
The Anti-Van-Der-Waerden number $\mathrm{aw}([N], k)$ is defined to be the smallest positive integer r such that every surjective r-colouring of $[N]$ contains at least one rainbow-progression. In their 2016 paper, Butler *et al.* [3] calculate exact values of $\mathrm{aw}([N], k)$ for small values of N and k and gave the following important asymptotic result.

Theorem (Butler *et al.* [3])**.** *There exists a positive integer N_0 and positive real numbers c_1, c_2 such that*

$$c_1 \log N \leq aw([N], 3) \leq c_2 \log N$$

for all $N \geq N_0$. For fixed $k \geq 4$, as N tends to infinity,

$$aw([N], k) = N^{1-o(1)}$$

holds.

The problem of studying Anti Van-der-Waerden numbers is about finding colourings avoiding all rainbow k-progressions. Conversely, the problem studied in this chapter is about finding colourings that *do not* avoid *any* rainbow-progressions.

In Section 2.2.1 we use a genetic algorithm to find colourings giving upper bounds for $a(n)$ for some small values of n.

We are also interested in the asymptotic behaviour of $a(n, k)$, in particular for the case where $k = k(n)$ is a function growing in n. In Section 2.3 we analyze the asymptotic behaviour of $a(n, k)$ for the case where $k = o(n^{1/6})$.

2.2 Exact values and bounds for small n

To find upper bounds for $a(n)$ for small n, we used a genetic algorithm to find n-colourings of short integer intervals covering all subsets of $[n]$.

The technique presented in the following Section 2.2.1 is described well in Melanie Mitchell's book *An Introduction to Genetic Algorithms* [15] and our implementation follows that description.

2.2.1 Genetic Algorithm and Backtracking approach

The general idea of genetic algorithms is the following. Let $\mathcal{S}$ be a set and let $\text{fit} : \mathcal{S} \to \mathbb{R}$. Consider the optimization problem

$$\text{find } x \in \text{argmax}\{\text{fit}(x) : x \in \mathcal{S}\}.$$

If it is not clear how to traverse the feasible set $\mathcal{S}$, one can use heuristic ideas motivated by biology to find points in $\mathcal{S}$ that have a high fitness-value. This idea is applicable if the set $\mathcal{S}$ has the property that—roughly speaking—modifying and combining elements in $\mathcal{S}$ results in elements that themselves are members of $\mathcal{S}$. In every iteration i, one has a multiset (list) $\mathcal{F}_i \subset \mathcal{S}$ (the multiset $\mathcal{F}_0$ is usually created by picking elements from $\mathcal{S}$ at random), called *current population*. From $\mathcal{F}_i$ pairs of elements (*parent* elements) with high fitness-values are selected to generate new elements in $\mathcal{S}$, called *offspring*. This is motivated by the idea from biology that in a breeding population, fit individuals will be more likely to breed. Afterwards, each element in the offspring might—with some probability—be slightly modified by what is called a *mutation*. This is motivated by the idea that in a population of organisms, external factors might cause slight changes in the DNA of individuals. From the offspring of $\mathcal{F}_i$, the fittest elements are chosen to form the population of the next generation, $\mathcal{F}_{i+1}$ (survival of the fittest).

In our case $\mathcal{S}$ will be the set of all n-colourings of the interval $[N]$. The fitness value $\mathrm{fit} : \mathcal{S} \to \mathbb{R}$ is designed such that colourings that cover many subsets of $[n]$ have a large fitness-value.

Definition 2.4. Let f be an n-colouring of the interval $[N]$ (where $n \leq N$). We define the fitness of f, $\mathrm{fit}(f)$, to be the number of subsets of $[n]$ that are covered by the colouring f. Different fitness functions can be defined by assigning weights to the covered subsets, according to their size. Reasonable options include:

$$\mathrm{fit}(f) = \sum_{R \subseteq [n]:\ R \text{ is covered by } f} |R|,$$

and

$$\mathrm{fit}(f) = \sum_{R \subseteq [n]:\ R \text{ is covered by } f} \left(\left\lceil \frac{n}{2} \right\rceil - \left| \left\lceil \frac{n}{2} \right\rceil - |R| \right| \right).$$

The second option assigns a high fitness-value to colourings that cover many subsets of $[n]$ that have size close to $\lceil \frac{n}{2} \rceil$. △

For any of the above choices, an n-colouring of $[N]$ covers all subsets of $[n]$ if and only if it has the maximum possible fitness-value.

We first describe the main procedure in Algorithm 2.5. The operations SelectParent, Crossover, InsertionMutate, ReversionMutate and ChangeMutate are described afterwards.

Algorithm 2.5 Genetic Algorithm

Input: $n, N, \mathrm{POP}, \mathrm{GEN}, \mathrm{OFF} \in \mathbb{N},\ \gamma_1, \gamma_2, \gamma_3, \tau \in [0,1]$.

$i = 0$.
$\mathcal{F}_0 =$ Randomly created multiset of n-colourings of $[N]$, $|\mathcal{F}_0| = \mathrm{POP}$.
while $i \leq \mathrm{GEN}$ **do**
  Offspring = [ ].
  while |Offspring| < OFF **do**
    $\mathrm{Parent}_1 = \textsc{SelectParent}(\mathcal{F}_i)$.
    $\mathrm{Parent}_2 = \textsc{SelectParent}(\mathcal{F}_i)$.
    with probability τ **do**
      $\mathrm{Child}_1, \mathrm{Child}_2 = \textsc{Crossover}(\mathrm{Parent}_1, \mathrm{Parent}_2)$.
    otherwise do
      $\mathrm{Child}_1 = \mathrm{Parent}_1$.
      $\mathrm{Child}_2 = \mathrm{Parent}_2$.
    Add Child_1 and Child_2 to Offspring.
    for $C \in$ Offspring **do**
      with probability γ_1 **do**
        $C = \textsc{InsertionMutate}(C)$, (and update C in Offspring).
      with probability γ_2 **do**
        $C = \textsc{ReversionMutate}(C)$, (and upd. C in Offspring).
      with probability γ_3 **do**
        $C = \textsc{ChangeMutate}(C)$, (and update C in Offspring).
  Set $\mathcal{F}_{i+1}$ to be the POP fittest colourings in Offspring.
  If $\mathcal{F}_{i+1}$ contains a colouring of maximum possible fitness, return that colouring and terminate.
  i = i+1.

We describe the process for selecting parent colourings.

Definition 2.6 ($\textsc{SelectParent}(\mathcal{F})$)**.** Let $B = [\mathrm{fit}(f_1), \ldots, \mathrm{fit}(f_{|\mathcal{F}|})]$ be the list of fitnesses of the colourings in $\mathcal{F}$. We define

$$w_i = \mathrm{fit}(f_i) - \min B,$$

giving a list of nonnegative weights $W = (w_1, \ldots, w_{|\mathcal{F}|})$, where the weight w_i corresponds to the deviation of the fitness of f_i from the fitness of the member of $\mathcal{F}$ with the worst fitness.
We return a colouring that is chosen at random from $\mathcal{F}$ with probability according to the list of weights W, i.e., the colouring $f_k \in \mathcal{F}$ is chosen with

probability

$$\frac{w_k}{\sum_{i=1}^{|\mathcal{F}|} w_i}.$$

△

Remember that an n-colouring f of $[N]$ can be represented as a sequence of length N over the alphabet $[n]$ by setting $S_f = (a_1, a_2, \ldots, a_N)$, where we write $a_i = f(i)$ for $i \in \{1, \ldots, N\}$. For the following definitions, let the input-colouring f be given by such a sequence.

The operation responsible for creating offspring is CROSSOVER.

Definition 2.7 (CROSSOVER)**.** The function CROSSOVER takes as input two colourings $(a_1, \ldots, a_N)$ and $(b_1, \ldots, b_N)$, picks an index $i \in \{1, \ldots, N-1\}$ uniformly at random and returns the colourings

$$\begin{aligned}\text{Child}_1 &= (a_1, \ldots, a_i, b_{i+1}, \ldots, b_N),\\ \text{Child}_2 &= (b_1, \ldots, b_i, a_{i+1}, \ldots, a_N).\end{aligned}$$

△

Whenever offspring is created, each colouring in the offspring is—with some probability—slightly changed (*mutated*) by one or multiple of the mutation operations INSERTIONMUTATE, CHANGEMUTATE and REVERSIONMUTATE.

Definition 2.8 (INSERTIONMUTATE)**.** The operation INSERTIONMUTATE picks two distinct indices $i, j \in \{1, \ldots, N\}$ uniformly at random, deletes the element a_i from the input sequence, and re-inserts it such that a_i is now at position j of the sequence.

INSERTIONMUTATE :
$(a_1, \ldots, a_N) \mapsto (a_1, \ldots, a_{j-1}, a_i, a_{j+1}, \ldots, a_{i-1}, a_{i+1}, \ldots, a_N)$
(for $j < i,$ and similarly for $j > i$).

△

Definition 2.9 (CHANGEMUTATE)**.** The operation CHANGEMUTATE picks an index $i \in \{1, \ldots, N\}$ and an element $c \in \{1, \ldots, n\}$ uniformly at random and changes the value of a_i to c.

CHANGEMUTATE :
$(a_1, \ldots, a_N) \mapsto (a_1, \ldots, a_{i-1},\ c,\ a_{i+1}, \ldots, a_N).$

△

Definition 2.10 (ReversionMutate)**.** The function ReversionMutate picks two indices $i, j \in \{1, \ldots, N\}$ (where $i < j$) uniformly at random and replaces the subsequence $(a_i, a_{i+1} \ldots, a_{j-1}, a_j)$ by its reverse $(a_j, a_{j-1}, \ldots, a_{i+1}, a_i)$.

ReversionMutate :

$(a_1, \ldots, a_i, a_{i+1} \ldots, a_{j-1}, a_j, \ldots, a_N) \mapsto (a_1, \ldots, a_j, a_{j-1}, \ldots, a_{i+1}, a_i, \ldots, a_N)$.

△

Results. For n up to 6, we managed to calculate $a(n)$ by means of exhaustive search, i.e., checking every possible n-colouring of $[a(n) - 1]$ to see that no such colouring is a solution. Using the genetic algorithm described in this section, we managed to find colourings giving upper bounds for $a(7), a(8)$, and $a(9)$, which are listed in Table 7.

n	$a(n)$
1	1
2	2
3	3
4	6
5	9
6	14
7	≤ 23
8	≤ 39
9	≤ 78

Table 7: Known values and bounds for $a(n)$ for small values of n.

We conjecture 23 to be the true value of $a(7)$, while we believe the bounds for $n = 8, 9$ not to be tight.

A note on a possible backtracking approach. Unfortunately, we did not manage to prove optimality for the colourings we found for $n \geq 7$. In order to prove $a(7) = 23$, we implemented a backtracking algorithm similar to the one used in Section 1.2.1, but failed to find an early-pruning criterion that helped narrow down the search space sufficiently. We describe the progress we made with this approach.
The most natural pruning mechanism is the following. Let f be a partial n-colouring (some elements may not be coloured) of the integer interval $[N]$ and

let K_f be the number of fully coloured progressions in $[N]$ plus the number of progressions in $[N]$ that are not fully coloured but that contain at least two elements that are coloured with the same colour. If the number of k-subsets that are not covered by f is larger than $|\mathrm{AP}_k(N)| - K_f$, the current colouring can not be extended to become a $P(n)$-covering colouring.

The order in which the elements of $[N]$ are coloured during backtracking is important. Instead of backtracking through all possible n-colourings of $[N]$ by colouring the elements in their natural ordering $1, 2, \ldots, N$ (like we did in Section 1.2.1), we can fix a permutation of $[N]$ and backtrack in the order according to that permutation. We chose an ordering of $[N]$ such that in every step the newly coloured element is part of *many* arithmetic progressions that have the property that at least one of their elements is already coloured. This idea helped to reduce the search space considerably, but we still were not able to calculate $a(7)$ with this approach.

2.3 Asymptotic bounds

In this section we give two results about the asymptotic behaviour of $a(n, k)$. The proof for Theorem 2.11 was found in cooperation with Leonardo Alese and Stefan Lendl and is given in the following Section 2.3.1.

Theorem 2.11. *For fixed $k \in \mathbb{N}$ we have*

$$a(n, k) = \mathcal{O}\left(\log n \cdot n^{k/2}\right)$$

as $n \to \infty$. If $k = k(n) = o(n^{1/6})$ tends to infinity as $n \to \infty$, we have

$$a(n, k) = \mathcal{O}\left(\log n \cdot e^{\frac{k}{2}} \cdot k^{\frac{-k}{2}+\frac{5}{4}} \cdot n^{\frac{k}{2}}\right).$$

Proposition 2.12. *As $n \to \infty$,*

$$a(n, k) = \Omega\left(\sqrt{k\binom{n}{k}}\right)$$

holds. If in particular $k = o(n^{\frac{1}{2}})$ tends to infinity as $n \to \infty$, we have

$$a(n, k) = \Omega\left(k^{\frac{-k}{2}+\frac{1}{4}} \cdot n^{\frac{k}{2}} \cdot e^{\frac{k}{2}}\right).$$

Comparing the asymptotic upper and lower bound for the case where $k = o(n^{1/6})$ tends to infinity, we see that the bounds only differ by a factor of $k \log n$.

2.3.1 Asymptotic upper bound

In the following pages, when using the term *random colouring,* we always refer to a colouring that colours each element with a colour chosen uniformly at random from the given set of colours.

Definition 2.13. Let $k, n, N \in \mathbb{N}$ (where $k \leq n \leq N$) and let f be a random n-colouring of $[N]$. For each $R \in \binom{[n]}{k}$ let $X_R(f)$ be the indicator variable of the random event *"There exists an R-coloured k-progression in $[N]$"*. For each k-progression A in $[N]$, let $Y_{A,R}(f)$ be the event *"The progression A is R-coloured"*.
We further define

$$X(f) = \sum_{R \in \binom{[n]}{k}} X_R(f),$$

a random variable counting the number of k-subsets of $[n]$ that are covered by f. △

Let $\mathrm{AP}_k(N)$ denote the set of all k-progressions in $[N]$. Note that $X_R(f)$ is the indicator variable of the event $\bigcup_{A \in \mathrm{AP}_k(N)} Y_{A,R}(f)$.

Lemma 2.14 states the Bonferroni inequality, which can be obtained by truncating the sum in the inclusion-exclusion principle such that only the intersections of up to two events are considered. We will use Lemma 2.14 to give a lower bound on the expectations $\mathbb{E}X_R(f)$. Then, by using linearity of expectation, we will obtain a lower bound for $\mathbb{E}X(f)$.

Lemma 2.14 (Bonferroni inequality)**.** *Let $A_1, \dots, A_k$ be random events. Then*

$$\mathbb{P}(A_1 \cup \dots \cup A_k) \geq \sum_{i=1}^{k} \mathbb{P}(A_i) - \sum_{i<j} \mathbb{P}(A_i \cap A_j).$$

Let $\mathcal{H}_k(N) = \binom{\mathrm{AP}_k(N)}{2}$ denote the set of all unordered pairs of k-progressions in $[N]$. Applying the Bonferroni inequality to our setting, we directly obtain Lemma 2.15.

Lemma 2.15. *Let $k, n, N \in \mathbb{N}$ (where $k \leq n \leq N$) and let f be a random n-colouring of $[N]$. For every k-subset R of $[n]$, the following holds.*

$$\begin{aligned}
&\mathbb{E}X_R(f) = \mathbb{P}(X_R(f) = 1) = \mathbb{P}\left(\bigcup_{A \in AP_k(N)} Y_{A,R}(f) \right) \\
&\geq \sum_{A \in AP_k(N)} \mathbb{P}(Y_{A,R}(f)) - \sum_{\{A,B\} \in \mathcal{H}_k(N)} \mathbb{P}\left(Y_{A,R}(f) \cap Y_{B,R}(f)\right) \\
&= \sum_{A \in AP_k(N)} \mathbb{P}(Y_{A,R}(f)) - \sum_{i=0}^{k-1} \sum_{\substack{\{A,B\} \in \mathcal{H}_k(N) \\ |A \cap B| = i}} \mathbb{P}(Y_{A,R}(f) \cap Y_{B,R}(f)).
\end{aligned}$$

□

To evaluate the lower bound from Lemma 2.15, we need to count the number of k-progressions in $[N]$ and the number of i-intersecting pairs of k-progressions in $[N]$. Asymptotic formulas are given in Lemma 2.17. Further we need to calculate the probabilities $\mathbb{P}(Y_{A,R}(f))$ and $\mathbb{P}(Y_{A,R}(f) \cap Y_{B,R}(f))$. The exact formulas are given in Lemma 2.18.

Definition 2.16. We define $h(N, k) = |\mathrm{AP}_k(N)|$ to be the number of k-progressions in $[N]$, and for $i = 0, 1, \dots, k-1$ we define $h_i(N, k)$ to be the number of pairs of i-intersecting k-progressions in $[N]$, i.e.,

$$h_i(N, k) = \left|\{\ \{A, B\} \in \mathcal{H}_k(N) : |A \cap B| = i\ \}\right|.$$

△

Lemma 2.17. *As N tends to infinity, the following asymptotic formulas hold for both the case where $k = k(N) \le N$ tends to infinity, and for the case where k is constant.*

- $h(N,k) = \frac{N^2}{2k-2} + \mathcal{O}(N)$.
- $h_0(N,k) \le \frac{N^4}{8(k-1)^2} + e_0(N,k)$, *where* $e_0(N,k) = \mathcal{O}(N^2/k)$.
- $h_1(N,k) = \mathcal{O}\left(N^3 k\right)$.
- $h_j(N,k) = \mathcal{O}\left(N^2 k^4\right)$ *for* $j \ge 2$.

Proof. Every k-progression in $[N]$ is uniquely determined by its first element s and its common distance d. Such a pair (s,d) encodes a valid k-progression if and only if $s + (k-1)d \le N$. For a given first element s, the largest possible value of d is thus $\left\lfloor \frac{N-s}{k-1} \right\rfloor$. It follows that

$$h(N,k) = \sum_{s=1}^{N-k+1} \left\lfloor \frac{N-s}{k-1} \right\rfloor.$$

By omitting the floor function, we obtain the following upper bound:

$$h(N,k) \le \frac{N^2 - N - k^2 + 3k - 2}{2k-2} \le \frac{N^2}{2k-2} + \frac{3k}{2k-2}.$$

Conversely, by omitting the floor function and subtracting 1 for every term in the sum, we get the following lower bound:

$$h(N,k) \ge \frac{N^2 - N - k^2 + 3k - 2}{2k-2} - (N - k + 1).$$

Thus $h(N,k) = \frac{N^2}{2k-2} + \mathcal{O}(N)$ holds. The bound for $h_0(N)$ is trivial; we just bound $h_0(N)$ by the total number of pairs of k-progressions.

$$h_0(N,k) \le \binom{h(N,k)}{2} \le \binom{\left\lfloor \frac{N^2}{2k-2} + \frac{3k}{2k-2} \right\rfloor}{2} = \frac{N^4}{8(k-1)^2} + e_0(N,k),$$

where $e_0(N,k) = \mathcal{O}(N^2/k)$.

Next we show that $h_1(N,k) = \mathcal{O}(N^3 k)$. For each k-progression $A = (a_1, \dots, a_k)$ in $[N]$ and each $a_j \in A$ there are at most $k \cdot N$ k-progressions $B = (b_1, \dots, b_k)$ such that $a_j \in B$; indeed, if $b_i = a_j$ for some $i \in [k]$, then there are at most N valid choices for b_{i+1} (or b_{i-1} if $i = k$), each choice uniquely determining B. Thus,

$$h_1(N,k) \le h(N,k) k^2 N \le \left(\frac{N^2}{2k-2} + \frac{3k}{2k-2} \right) k^2 N = \mathcal{O}(N^3 k).$$

Finally, we show that $h_i(N,k) = \mathcal{O}(N^2k^4)$ holds for $i \geq 2$. We want to count pairs of k-progressions that intersect in at least two points. For each pair of distinct points $x_1, x_2 \in [N]$ there are at most $\binom{k}{2}$ k-progressions containing both x_1 and x_2; for each pair of distinct indices $j_1, j_2 \in [k]$ there is at most one progression $A = (a_1, \dots, a_k)$ with $a_{j_1} = x_1$ and $a_{j_2} = x_2$. Thus there are at most $\binom{\binom{k}{2}}{2}$ pairs of k-progressions both containing x_1 and x_2. There are $\binom{N}{2}$ choices for the pair x_1, x_2 and we thus obtain

$$h_i(N,k) \leq \binom{N}{2}\binom{\binom{k}{2}}{2} = \mathcal{O}(N^2k^4).$$

□

The following Lemma is straight-forward to prove.

Lemma 2.18. *Let $k \leq n \leq N$ be positive integers. Let $f : [N] \to [n]$ be a random n-colouring of $[N]$. Then, for any k-progression A in $[N]$ and any k-subset R of $[n]$*

$$\mathbb{P}\left(Y_{A,R}(f)\right) = \frac{k!}{n^k}$$

holds.
Further, for any pair A, B of k-progressions in $[N]$ such that $|A \cap B| = i$, where $i \in \{0, \dots, k-1\}$:

$$\mathbb{P}\left(Y_{A,R}(f) \cap Y_{B,R}(f)\right) = \frac{(k-i)!k!}{n^{2k-i}}.$$

Proof. Let $R \in \binom{[n]}{k}$. The total number of possible n-colourings of a given k-progression is n^k. The number of possible ways to colour a given k-progression with k distinct colours is $k!$, thus $\mathbb{P}(Y_{A,R}(f)) = \frac{k!}{n^k}$.
Let A and B be two k-progressions intersecting in i positions. A and B are in total made up of $2k-i$ distinct elements in $[N]$, thus the total number of possible n-colourings of A and B is n^{2k-i}. Let $\{a_1, \dots, a_{k-i}\}$ be the elements in $[N]$ that appear in A but not in B and let $B = (b_1, \dots, b_k)$. Similarly to before, there are $k!$ ways to colour B with the k colours in R. Each such colouring of B determines the colours of the intersection between A and B. Thus the remaining $k-i$ colours in R must be used for the colouring of the elements $\{a_1, \dots, a_{k-i}\}$. There are $(k-i)!$ ways to colour $k-i$ elements with $k-i$ distinct colours, proving the second claim. □

We are ready to evaluate the lower bound from Lemma 2.15.

Lemma 2.19. *Let $k = k(n) = o(n^{1/6})$ and let $N = N(n) = \left\lceil \sqrt{2}\sqrt{\frac{k-1}{k!}} \cdot n^{k/2} \right\rceil$. There exists a function $\phi : \mathbb{N} \to \mathbb{R}$ with $\phi(n) = o(1)$ as n tends to infinity such that the following property holds for all $n \in \mathbb{N}$:*
Let f_n be a random n-colouring of $[N]$. Then, for every $R \in \binom{[n]}{k}$ the inequality

$$\mathbb{E}X_R(f_n) \geq \frac{1}{2} + \phi(n)$$

holds.

Proof. In Lemma 2.15 we established a bound for $\mathbb{E}X_R(f_n)$:

$$\mathbb{E}X_R(f_n) \geq \sum_{A \in \mathrm{AP}_k(N)} \mathbb{P}(Y_{A,R}(f_n)) - \sum_{i=0}^{k-1} \sum_{\substack{\{A,B\} \in \mathcal{H}_k(N) \\ |A \cap B| = i}} \mathbb{P}(Y_{A,R}(f_n) \cap Y_{B,R}(f_n)).$$

Using the counting functions from Definition 2.16 and the probabilities from Lemma 2.18, we can rewrite the above bound as

$$\begin{aligned}
&\mathbb{E}X_R(f_n) \\
&\geq h(N)\frac{k!}{n^k} - h_0(N)\frac{k!k!}{n^{2k}} - h_1(N)\frac{k(k-1)!}{n^{2k-1}} - \sum_{i=2}^{k-1} h_i(N)\frac{(k-i)!k!}{n^{2k-i}} \\
&\geq h(N)\frac{k!}{n^k} - \left(\frac{N^4}{8(k-1)^2} + e_0(N,k)\right)\frac{k!k!}{n^{2k}} \\
&- h_1(N)\frac{k(k-1)!}{n^{2k-1}} - \sum_{i=2}^{k-1} h_i(N)\frac{(k-i)!k!}{n^{2k-i}} \\
&=: L(n).
\end{aligned}$$

In the last inequality we used the upper bound for h_0 from Lemma 2.17. We now show that the lower bound $L(n)$ is equal to $1/2 + \phi(n)$ for some function $\phi(n) = o(1)$.

Plugging in the asymptotic formulas for h and h_i ($i = 0, \ldots, k-1$) we get

$$\begin{aligned}
L(n) = &\left(\frac{N^2}{2k-2} + \mathcal{O}(N)\right)\frac{k!}{n^k} - \left(\frac{N^4}{8(k-1)^2} + e_0(N,k)\right)\frac{k!k!}{n^{2k}} \\
&+\mathcal{O}(N^3k)\frac{(k-1)!k!}{n^{2k-1}} + \mathcal{O}(N^2k^4)\sum_{i=2}^{k-1}\frac{(k-i)!k!}{n^{2k-i}}.
\end{aligned}$$

We show that only the terms $\frac{N^2}{2k-2}\frac{k!}{n^k}$ and $\frac{N^4}{8(k-1)^2}\frac{k!k!}{n^{2k}}$ are asymptotically relevant. To this end, we show that all other terms vanish asymptotically;

(i) $\mathcal{O}(N)\frac{k!}{n^k} = o(1)$,

(ii) $e_0(N,k)\frac{k!k!}{n^{2k}} = \mathcal{O}(N^2/k)\frac{k!k!}{n^{2k}} = o(1)$,

(iii) $\mathcal{O}(N^3k)\frac{(k-1)!k!}{n^{2k-1}} = o(1)$, and

(iv) $\mathcal{O}(N^2k^4)\sum_{i=2}^{k-1}\frac{(k-i)!k!}{n^{2k-i}} = o(1)$.

In the case where k is constant, we have $N = \mathcal{O}\left(n^{k/2}\right)$ and i) - iv) hold trivially. Below we prove i) - iv) for the case where $k = o(n^{1/6})$ tends to infinity. Using Stirling's formula, we obtain

$$N = \left\lceil \sqrt{2}\sqrt{\frac{k-1}{k!}}n^{k/2} \right\rceil \sim \sqrt{2k}(2\pi k)^{-1/4}\left(\frac{en}{k}\right)^{k/2}. \tag{5}$$

Using (5) and Stirling's formula we see

$$\begin{aligned} N\frac{k!}{n^k} &\sim \sqrt{2k}(2\pi k)^{-1/4}\left(\frac{en}{k}\right)^{k/2}\sqrt{2\pi k}\left(\frac{k}{e}\right)^k\frac{1}{n^k} \\ &= \mathcal{O}\left(k^{3/4}e^{-k/2}\cdot k^{k/2}n^{-k/2}\right) = o(1), \end{aligned}$$

proving (i). Similarly, we have

$$\begin{aligned} N^3\frac{k!k!}{n^{2k-1}} &\sim (2k)^{3/2}(2\pi k)^{-3/4}\left(\frac{en}{k}\right)^{3k/2}(2\pi k)\left(\frac{k}{e}\right)^{2k}\frac{1}{n^{2k-1}} \\ &= \mathcal{O}\left(e^{-k/2}k^{k/2+\frac{7}{4}}n^{-k/2+1}\right) = o(1), \end{aligned}$$

implying both (ii) and (iii). Note that the terms of the sum in (iv) are monotonically increasing. We can thus use the bound

$$\sum_{i=2}^{k-1}\frac{(k-i)!k!}{n^{2k-i}} \le k\frac{k!}{n^{k+1}}$$

before applying (5) and Stirling's formula; obtaining

$$\begin{aligned} &N^2k^4\sum_{i=2}^{k-1}\frac{(k-i)!k!}{n^{2k-i}} \le N^2k^5\frac{k!}{n^{k+1}} \\ &\sim 2k(2\pi k)^{-1/2}\left(\frac{en}{k}\right)^k k^5\sqrt{2\pi k}\left(\frac{k}{e}\right)^k\frac{1}{n^{k+1}} \\ &= \mathcal{O}\left(\frac{k^6}{n}\right) = o(1), \end{aligned}$$

proving (iv). Note that in the last line we use the assumption $k = o(n^{1/6})$.

We are thus left with the following representation of $L(n)$:

$$L(n) = \frac{N^2}{2k-2}\frac{k!}{n^k} - \frac{N^4}{8(k-1)^2}\frac{k!k!}{n^{2k}} + o(1).$$

Since $N = \left\lceil \sqrt{2}\sqrt{\frac{k-1}{k!}} n^{k/2} \right\rceil$, the above expression simplifies to

$$L(n) = \frac{1}{2} + o(1).$$

Thus, there exists a function $\phi : \mathbb{N} \to \mathbb{R}$ such that for all $n \in \mathbb{N}$

$$\mathbb{E}X_R(f_n) \geq L(n) = \frac{1}{2} + \phi(n),$$

where $\phi(n) = o(1)$. $\square$

From Lemma 2.19 we can easily derive the following fact.

Lemma 2.20. *Let $k = k(n) = o(n^{1/6})$ and let $N = N(n) = \left\lceil \sqrt{2}\sqrt{\frac{k-1}{k!}} \cdot n^{k/2} \right\rceil$. There exists a function $\phi(n) = o(1)$ such that the following property holds for all $n \in \mathbb{N}$:*
Let $\mathcal{F}_n \subseteq \binom{[n]}{k}$ be a family of k-subsets of $[n]$. There exists an n-colouring f_n^* of $[N]$ such that the number of sets in $\mathcal{F}_n$ that are covered by f_n^* is at least $|\mathcal{F}_n| \left(\frac{1}{2} + \phi(n)\right)$.

Proof. For each $n \in \mathbb{N}$ let f_n be a random n-colouring of $[N]$. From Lemma 2.19 we know that there exists a function $\phi(n) = o(1)$ such that for all $n \in \mathbb{N}$ and $R \in \binom{[n]}{k}$ the inequality

$$\mathbb{E}X_R(f_n) \geq \left(\frac{1}{2} + \phi(n)\right)$$

holds. By linearity of expectation we can compute a lower bound for the expected number of sets in $\mathcal{F}_n$ that are covered by f_n:

$$\mathbb{E}X(f_n) = \mathbb{E}\left(\sum_{R \in \mathcal{F}_n} X_R(f_n)\right) = \sum_{R \in \mathcal{F}_n} \mathbb{E}X_R(f_n) \geq |\mathcal{F}_n| \left(\frac{1}{2} + \phi(n)\right).$$

For each $n \in \mathbb{N}$, there exists an n-colouring f_n^* of $[N]$ that covers at least the expected number of covered sets in $\mathcal{F}_n$, completing the proof. $\square$

We are ready to prove the main result.

Proof of Theorem 2.11.

Proof. For each $n \in \mathbb{N}$ let $N = \left\lceil \sqrt{2}\sqrt{\frac{k-1}{k!}} \cdot n^{k/2} \right\rceil$. By Lemma 2.20 we know that there exists an n-colouring $g_n^{(0)}$ of $[N]$ that covers at least $\binom{n}{k}\left(\frac{1}{2} + \phi(n)\right)$ of the sets in $\mathcal{F}_0 := \binom{[n]}{k}$, where ϕ is an asymptotically vanishing function.

Let $\mathcal{F}_1$ be the family of sets in $\mathcal{F}_0$ that have not been covered by $g_n^{(0)}$. We apply Lemma 2.20 again, yielding an n-colouring $g_n^{(1)}$ of $[N]$ covering at least $|\mathcal{F}_1|\left(\frac{1}{2} + \phi(n)\right)$ of the sets in $\mathcal{F}_1$. We repeat this process r times, by defining $\mathcal{F}_i$ to be the family of k-subsets of $[n]$ not yet covered by any of the colourings $g_n^{(0)} \dots, g_n^{(i-1)}$.

After r iterations, the number of k-subsets of $[n]$ that are not covered by any of the constructed colourings is at most $|\mathcal{F}_0|\left(\frac{1}{2} - \phi(n)\right)^r$. Setting $r = r(n,k) = \lceil \alpha \cdot k \log n \rceil$, where $\alpha > \frac{1}{\log(2)}$, we get (see Proposition 2.21)

$$|\mathcal{F}_0|\left(\frac{1}{2} - \phi(n)\right)^{r(n,k)} = \binom{n}{k}\left(\frac{1}{2} - \phi(n)\right)^{r(n,k)} \to 0 \text{ as } n \to \infty.$$

Thus, for sufficiently large n, after $r(n,k)$ iterations, every k-subset of $[n]$ is covered by at least one of the colourings

$$g_n^{(0)}, g_n^{(1)}, \dots, g_n^{(r(n,k)-1)}.$$

From the colourings $g_n^{(0)}, g_n^{(1)}, \dots, g_n^{(r(n,k)-1)}$ we construct an n-colouring g of $S := \{1, 2, \dots, r(n,k) \cdot N\}$. We split S into $r(n,k)$ intervals of length N and colour each of these intervals with the corresponding colouring $g_n^{(i)}$. Formally, we set

$$g\left(i \cdot N + s\right) = g_n^{(i)}(s) \quad i \in \{0, \dots, r(n,k) - 1\},\ s \in \{1, \dots, N\}.$$

The colouring g is an n-colouring of $S = \left\lceil \lceil \alpha \cdot k \log n \rceil \cdot \left\lceil \sqrt{2}\sqrt{\frac{k-1}{k!}} \cdot n^{k/2} \right\rceil \right\rceil$ that covers all k-subsets of $[n]$. It follows that

$$a(n,k) = \mathcal{O}\left(k \cdot \log n \cdot \sqrt{\frac{k-1}{k!}} \cdot n^{k/2}\right).$$

If $k = o(n^{1/6})$ tends to infinity as $n \to \infty$,

$$a(n,k) = \mathcal{O}\left(\log n \cdot e^{\frac{k}{2}} \cdot k^{\frac{-k}{2} + \frac{5}{4}} \cdot n^{k/2}\right)$$

holds. If k is constant, we have

$$a(n,k) = \mathcal{O}\left(\log n \cdot n^{k/2}\right).$$

□

It only remains to verify the following calculation, which was needed for the proof of Theorem 2.11.

Proposition 2.21. *Let $k = k(n) = o(n^{1/6})$ and let $r(n,k) = \lceil \alpha \cdot k \log n \rceil$, where $\alpha > \frac{1}{\log(2)}$. Let ϕ be a real-valued function such that $\phi(n) = o(1)$ as n tends to infinity. Then*

$$\binom{n}{k}\left(\frac{1}{2} - \phi(n)\right)^{r(n,k)} \to 0 \text{ as } n \to \infty.$$

Proof. For every $\epsilon \in (0, \frac{1}{2})$, there exists a positive integer n_0 such that for all $n \geq n_0$,

$$\binom{n}{k}\left(\frac{1}{2} - \phi(n)\right)^{r(n,k)} \leq \binom{n}{k}\left(\frac{1}{2} + \epsilon\right)^{r(n,k)}.$$

If $k = o(n^{1/6}) = o(n^{1/2})$ tends to infinity as $n \to \infty$, we have

$$\begin{aligned}
&\binom{n}{k}\left(\frac{1}{2} + \epsilon\right)^{r(n,k)} \sim \sqrt{\frac{1}{2\pi k}}\left(\frac{ne}{k}\right)^{k} \cdot \left(\frac{1}{2} + \epsilon\right)^{r(n,k)} \\
&= \exp\left(\log\sqrt{\frac{1}{2\pi k}} + k\log n - k\log k + k + r(n,k)\log\left(\frac{1}{2} + \epsilon\right)\right).
\end{aligned}$$

If k is constant, we have

$$\binom{n}{k}\left(\frac{1}{2} + \epsilon\right)^{r(n,k)} \sim \exp\left(k\log n + r(n,k)\log\left(\frac{1}{2} + \epsilon\right)\right).$$

We define $r(n,k)$ such that in both of the above expression, the argument of the exponential function goes to $-\infty$ as $n \to \infty$. Since the the terms $\log\sqrt{\frac{1}{2\pi k}}$, $k\log k$ and k are $o(k\log n)$, this is achieved by setting $r(n,k)$ such that

$$k\log n + r(n,k)\log\left(\frac{1}{2} + \epsilon\right) \to -\infty \text{ as } n \to \infty,$$

which is done by setting $r(n,k) = \lceil \alpha \cdot k\log n \rceil$, where $\alpha > -\frac{1}{\log\left(\frac{1}{2}+\epsilon\right)}$. Since ϵ can be chosen arbitrarily small, we can choose any $\alpha > \frac{1}{\log(2)}$. □

2.3.2 Asymptotic lower bound

We conclude this chapter with the proof of Proposition 2.12.

Proof of Proposition 2.12.

Proof. Let $N \in \mathbb{N}$. If we want to colour $[N]$ with n colours such that for each $R \in \binom{[n]}{k}$ there is a k-progression in $[N]$ that is R-coloured, we require in particular that there are at least $\binom{n}{k}$ arithmetic k-progressions in $[N]$.
We know from the proof of Lemma 2.17 that the number of k-progressions in $[N]$ is

$$h(N,k) = \sum_{s=1}^{N-k+1} \left\lfloor \frac{N-s}{k-1} \right\rfloor .$$

By omitting the floor function for every term in the sum, we get the following upper bound:

$$h(N,k) \leq \frac{N^2 - N - k^2 + 3k - 2}{2k-2} \leq \frac{N^2 + 3k}{2k-2}.$$

Thus, if our requirement $h(N,k) \geq \binom{n}{k}$ holds, then

$$\frac{N^2+3k}{2k-2} \geq \binom{n}{k} \tag{6}$$

also holds. Solving (6) for N, we obtain

$$\begin{aligned} N &\geq \sqrt{2(k-1)\binom{n}{k} - 3k} \\ &= \Omega\left(\sqrt{k\binom{n}{k}}\right). \end{aligned}$$

If $k = o(n^{\frac{1}{2}})$ tends to infinity as $n \to \infty$, we have

$$\binom{n}{k} = \frac{1}{\sqrt{2\pi k}} \left(\frac{ne}{k}\right)^k (1 + o(1)),$$

and thus in this case

$$N = \Omega\left(k^{\frac{-k}{2}+\frac{1}{4}} \cdot n^{\frac{k}{2}} \cdot e^{\frac{k}{2}}\right)$$

holds. $\square$

3 Generalization to graphs

We describe an interesting generalization of the problems described in Chapters 1 and 2. Let $\mathcal{F}$ be a fixed family of graphs and let $\mathcal{P}$ be a family of subsets of $[n]$. We want to find a graph $G \in \mathcal{F}$ with the least possible number of vertices such that the vertices of the graph can be coloured such that for every set $R \in \mathcal{P}$, there exists a connected (induced) subgraph of G—consisting of $|R|$ vertices—whose vertex colours are exactly the elements of R.

Definition 3.1. Let $n \in \mathbb{N}$ and let G be a graph. Let f be a vertex colouring $V(G) \to [n]$. For a family of subsets $\mathcal{P}$ of $[n]$ we say f *covers* $\mathcal{P}$ if for every $X \in \mathcal{P}$ there exists a connected subgraph H of G such that $|V(H)| = |X|$ and $\{f(v) : v \in V(H)\} = X$.
We say a graph $G \in \mathcal{F}$ *can cover* $\mathcal{P}$ if there exists a colouring $f : V(G) \mapsto [n]$, such that f covers $\mathcal{P}$. △

We are interested in the cases $\mathcal{P} = P(n)$ and $\mathcal{P} = P_{k_1,\dots,k_r}(n)$.

Definition 3.2. Let $\mathcal{F}$ be a family of graphs. We define $g(\mathcal{F}, n)$ to be the least possible number of vertices of a graph in $\mathcal{F}$ that can cover $P(n)$, and $g_{k_1,\dots,k_r}(\mathcal{F}, n)$ to be the least possible number of vertices of a graph in $\mathcal{F}$ that can cover $P_{k_1,\dots,k_r}(n)$. △

Example. Let $\mathcal{K} = \{K_1, K_2, \dots\}$ denote the class of all complete graphs. Since any collection of vertices of the complete graph K_n induces a connected subgraph, the graph K_n can cover $P(n)$ and thus

$$g(\mathcal{K}, n) = n$$

for all $n \in \mathbb{N}$. △

The class $\mathcal{T}$ of all trees is more interesting. The graph T_n given in the proof of the following example was found during a discussion with Stefan Lendl and Leonardo Alese.

Example. We have $g(\mathcal{T}, n) = g_2(\mathcal{T}, n) = \binom{n}{2} + 1$.

Proof. In a graph, connected subgraphs with two vertices are exactly the edges of the graph. Thus, a tree that can cover $P_2(n)$ must consist of at least $\binom{n}{2}$ edges. A tree with $\binom{n}{2}$ edges has $\binom{n}{2} + 1$ vertices and thus we have

$$g_n(\mathcal{T}) \geq g_{n,2}(\mathcal{T}) \geq \binom{n}{2} + 1.$$

We construct a tree T_n with $\binom{n}{2}+1$ vertices and a corresponding colouring f that covers $P(n)$. This shows $g_{n,2}(\mathcal{T}) \leq g_n(\mathcal{T}) \leq \binom{n}{2}+1$.

T_n consists of three layers.

$$V(T_n) = \{r\} \cup \{x_2, \dots, x_n\} \cup \{y_{i,j} : \ 2 \leq i \leq n, \ i+1 \leq j \leq n\}.$$

There is an edge between the root vertex r and each of the vertices x_i. Further for each vertex x_i, there is an edge between x_i and each of the vertices $y_{i,j}$, where $j \in \{i+1, \dots, n\}$. The colouring f is defined as follows:

$$\begin{aligned} &f(r) = 1, \\ &f(x_i) = i \text{ for } i \in \{2, \dots, n\}, \\ &f(y_{i,j}) = j \text{ for } i \in \{2, \dots, n\}, j \in \{i+1, \dots, n\}. \end{aligned}$$

Figure 2 shows T_n and f.

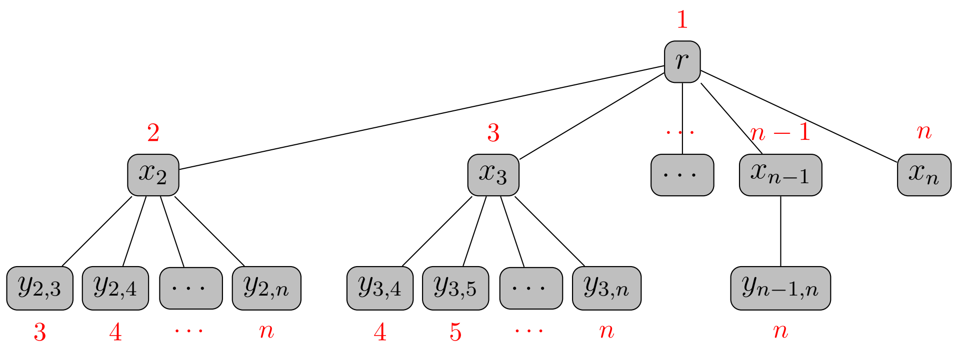


Figure 2: The graph T_n. The numbers in red correspond to the values of the colouring f.

We show that every subset $A \subseteq [n]$ appears as the set of vertex-colours of an connected subgraph of size $|A|$ in T_n. Let $a_1, \dots, a_r$ denote the elements of A in increasing order. If $a_1 = 1$, set

$$H = \{r, x_{a_2}\} \cup \{y_{a_2,a_j} : \ j = 3, \dots, r\},$$

and if $a_1 \neq 1$, set

$$H = \{x_{a_1}\} \cup \{y_{a_1,a_j} : j = 2, \dots, r\}.$$

In both cases the subgraph induced by H is connected and the colours of the vertices of H are exactly $a_1, \dots, a_r$.

□

Example (Hypercubes)**.** Another interesting graph class to consider is the class of all hypercubes $\mathcal{H} = \{Q_1, Q_2, Q_3, \dots\}$, where the hypercube Q_k of dimension k is defined as the graph on the vertex set

$$V(Q_k) = \{(x_1, \dots, x_k) : \; x_i \in \{0,1\}, 1 \le i \le k\},$$

where two vertices $x, y \in V(Q_k)$ are connected by an edge if and only if x differs from y in exactly one position. Note that Q_k has 2^k vertices and $k2^{k-1}$ edges. The hypercube Q_k can also be constructed by taking two copies of Q_{k-1} and adding an edge between each vertex and its copy. We define $h(n) = \log_2 g(\mathcal{H}, n)$, the dimension of the smallest hypercube that can cover $P(n)$. It is easy to verify that $h(3) = 2$ and $h(4) = 3$. Corresponding colourings are given in Figures 3 and 4.

A cube that can cover $P(n)$ can in particular cover $P_2(n)$ and must therefore consist of at least $\binom{n}{2}$ edges. Thus, if Q_k can cover $P(n)$, then

$$k2^{k-1} \ge \binom{k}{2}$$

must hold. It follows that $h(6) \ge 4$ and $h(8) \ge 5$. The bound $h(5) \ge 4$ was shown using an exhaustive computer search. Using a randomized search algorithm, we found a 7-colouring of Q_4, covering every subset of $\{1, \dots, 7\}$ (given in Figure 5) and a 9-colouring of Q_5, covering every subset of $\{1, \dots, 9\}$ (given in Figure 6). This implies $h(9) \le 5$ and $h(7) \le 4$. This gives the following values of $h(n)$.

n	$h(n)$
2	1
3	2
4	3
5	4
6	4
7	4
8	5
9	5

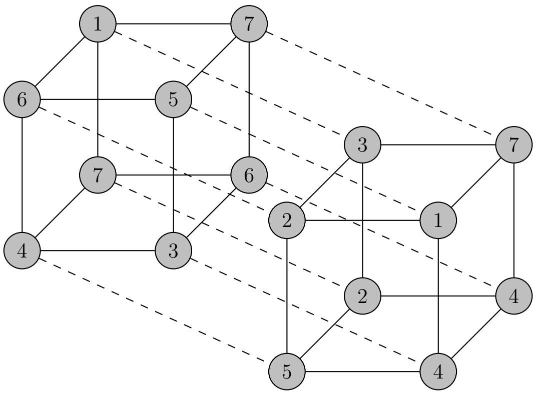
Table 8: Known values of $h(n)$ for small n.

– (0,0): 1 – (0,1): 2 – (1,0): 3 – (1,1): 1

Figure 3: A colouring of Q_2 with 3 colours. Every subset of $\{1,2,3\}$ appears as the colours of the vertices of a connected subgraph in Q_2.

– (0, 0, 0): 1
– (1, 0, 0): 3
– (0, 1, 0): 1
– (0, 0, 1): 4
– (1, 1, 0): 4
– (0, 1, 1): 2
– (1, 0, 1): 3
– (1, 1, 1): 3

Figure 4: A colouring of Q_3 with 4 colours. Every subset of $\{1,\ldots,4\}$ appears as the colours of the vertices of a connected subgraph in Q_3.



– (0, 0, 0, 0): 4
– (1, 0, 0, 0): 5
– (0, 1, 0, 0): 3
– (0, 0, 1, 0): 6
– (0, 0, 0, 1): 7
– (1, 1, 0, 0): 4
– (0, 1, 1, 0): 5
– (1, 0, 1, 0): 2
– (0, 1, 0, 1): 6
– (0, 0, 1, 1): 1
– (1, 0, 0, 1): 2
– (1, 1, 1, 0): 1
– (1, 1, 0, 1): 4
– (1, 0, 1, 1): 3
– (0, 1, 1, 1): 7
– (1, 1, 1, 1): 7

Figure 5: The graph Q_4 and a corresponding vertex colouring with 7 colours. Every subset of $\{1,\ldots,7\}$ appears as the colours of the vertices of a connected subgraph in Q_4.

– (0, 0, 0, 0, 0): 9	– (0, 1, 0, 1, 0): 3	– (1, 0, 1, 1, 1): 9
– (0, 0, 0, 0, 1): 8	– (1, 0, 1, 0, 0): 6	– (1, 1, 1, 0, 0): 3
– (0, 0, 0, 1, 0): 3	– (1, 0, 0, 1, 0): 1	– (1, 1, 1, 1, 0): 5
– (0, 0, 1, 0, 0): 7	– (0, 0, 1, 1, 1): 3	– (1, 0, 1, 0, 1): 5
– (0, 1, 0, 0, 0): 5	– (0, 1, 1, 1, 1): 8	– (1, 0, 0, 0, 1): 5
– (1, 0, 0, 0, 0): 4	– (1, 0, 1, 1, 0): 2	– (1, 1, 0, 0, 1): 2
– (0, 0, 0, 1, 1): 2	– (0, 1, 0, 0, 1): 5	– (1, 0, 0, 1, 1): 7
– (0, 0, 1, 1, 0): 4	– (1, 1, 0, 1, 1): 2	– (0, 1, 0, 1, 1): 6
– (0, 1, 1, 0, 0): 4	– (1, 1, 0, 1, 0): 6	– (0, 1, 1, 1, 0): 7
– (1, 1, 0, 0, 0): 8	– (0, 1, 1, 0, 1): 9	– (1, 1, 1, 1, 1): 1
– (0, 0, 1, 0, 1): 3	– (1, 1, 1, 0, 1): 6	

Figure 6: A colouring of Q_5 with 9 colours. Every subset of $\{1, \ldots, 9\}$ appears as the colours of the vertices of a connected subgraph in Q_5.

For all $n \in \mathbb{N}$ we have $h(n+1) \leq h(n) + 1$; the hypercube of dimension $h(n) + 1$ can be constructed by taking two copies Q, Q^* of the hypercube of dimension $h(n)$ (and connecting each vertex to its copy). Colour the vertices of Q with n colours such that every subset of $\{1, \ldots, n\}$ appears as the colours of the vertices of a connected subgraph in Q and colour all vertices of Q^* with the colour $n+1$. The resulting colouring covers all subsets of $\{1, \ldots, n+1\}$. From this it follows that $5 \leq h(10) \leq 6$. $\triangle$

The generalization we introduced in this chapter offers a wealth of problems to study. Graph classes that might be of particular interest for further research include the class of all binary trees, the class of all caterpillar graphs, the class of all r-regular graphs (for some fixed positive integer r) and the class of all triangulations.

4 References

[1] Iryna Banakh, Taras Banakh, Pavel Trisch, and Myroslava Vovk. Toehold purchase problem: A comparative analysis of two strategies. 2012.

[2] N.G. De Bruijn, C. van E. Tengbergen, and D. Kruyswijk. On the set of divisors of a number. *Nieuw Archief voor Wiskunde*, 23(2):191–193, 1951.

[3] Steve Butler, Craig Erickson, Leslie Hogben, Kirsten Hogenson, Lucas Kramer, Richard Kramer, Jephian C.-H Lin, Ryan Martin, Derrick Stolee, Nathan Warnberg, and Michael Young. Rainbow arithmetic progressions. *Journal of Combinatorics*, 7:595 – 626, 08 2016.

[4] Fan Chung, Persi Diaconis, and Ron Graham. Universal cycles for combinatorial structures. *Discrete Mathematics*, 110(1):43 – 59, 1992.

[5] Dawn Curtis, Taylor Hines, Glenn Hurlbert, and Tatiana Moyer. Near-universal cycles for subsets exist. *SIAM Journal on Discrete Mathematics*, 23(3):1441–1449, 2009.

[6] Michał Dębski and Zbigniew Lonc. Universal cycle packings and coverings for k-subsets of an n-set. *Graphs and Combinatorics*, 32(6):2323–2337, Nov 2016.

[7] R. P. Dilworth. A decomposition theorem for partially ordered sets. *Annals of Mathematics*, 51(1):161–166, 1950.

[8] Curtis Greene and Daniel J. Kleitman. Strong versions of Sperner's theorem. *Journal of Combinatorial Theory, Series A*, 20(1):80 – 88, 1976.

[9] Glenn Hurlbert. On universal cycles for k-subsets of an n-set. *SIAM Journal on Discrete Mathematics*, 7(4):598–604, 1994.

[10] B.W. Jackson. Universal cycles of k-subsets and k-permutations. *Discrete Mathematics*, 117(1):141 – 150, 1993.

[11] Stasys Jukna. *Extremal Combinatorics: With Applications in Computer Science*. Springer Publishing Company, Incorporated, 2nd edition, 2011.

[12] Lawrence T. Kou. Polynomial complete consecutive information retrieval problems. *SIAM J. Comput.*, 6:67–75, 1977.

[13] B.M. Landman and A. Robertson. *Ramsey Theory on the Integers: Second Edition*. Student Mathematical Library. American Mathematical Society, 2014.

[14] Witold Lipski. On strings containing all subsets as substrings. *Discrete Mathematics*, 21(3):253 – 259, 1978.

[15] Melanie Mitchell. *An Introduction to Genetic Algorithms*. MIT Press, Cambridge, MA, USA, 1996.

[16] N. J. A. Sloane. A014314. *The On-Line Encyclopedia of Integer Sequences*.

[17] Emanuel Sperner. Ein Satz über Untermengen einer endlichen Menge. *Mathematische Zeitschrift*, 27(1):544–548, 1928.