DIMACS-DIMATIA International REU
Research Experience for Undergraduates 2017

May 30 – August 1 2017,

Piscataway, NJ, USA and Prague, Czech Republic

Jaroslav Hančl (ed.)

DIMACS/DIMATIA International REU 2017

Published by IUUK–ITI series 2017-666
Centre of Excellence – Institute for Theoretical Computer Science,
Faculty of Mathematics and Physics, Charles University
Malostranské náměstí 25, 118 00, Praha 1, Czech Republic

Prague 2017

Preface

Charles University in Prague and particularly Department of Applied Mathematics (KAM), Computer Science Institute of Charles University (IÚUK) and its international centre DIMATIA, are very happy to host one of the very few International REU programmes which were established ane are in long term supported by the National Science Foundation.

Repeatedly, since the establishment of DIMACS–DIMATIA REU in 1999, it has been awarded for its accomplishments and educational excellence.

On the Czech side, the programme was financed jointly by the RSJ foundation, by the School of Computer Science of Faculty of Mathematics and Physics of Charles University, in particular by KAM and IÚUK, and by our grants CE–ITI P202/12/G061, ERCCZ LL1201 and SVV 202–09/260452. We thank all the contributors and hope that the future will bring us a more stable support. Nevertheless all our institutions are proud sponsors of this unique activity.

This booklet reports just the programme in 2017. I thank to Jaroslav Hančl, the Czech mentor of this year, for a very good work both during the programme itself and after.

Prague, November 15, 2017
Jaroslav Nešetřil

The DIMACS/DIMATIA Exchange program has been going on for many years and is a valuable experience for all involved. This year, I was the graduate coordinator from Rutgers, and it was a pleasure interacting with all the participants involved. I can personally say I very much enjoyed the exchange program, and learned and developed a great deal from helping to coordinate the REU on the American side.

Of course, the exchange program is about the students, and all of the Americans expressed their satisfaction. In particular, they learned a lot and enjoyed the lectures delivered at Charles University to prepare them for the Midsummer Combinatorial Workshop. In combination with seeing a different country and experiencing a different culture, this was a very valuable experience for them.

I would like to express my gratitude to the DIMATIA staff and the Czech REU students involved in the exchange for being such welcoming and generous hosts. The stay would have been much less enjoyable had it not been for their time spent with us as guides to the Czech Republic.

Parker Hund,
Rutgers University

DIMACS/DIMATIA Research Experiences for Undergraduates (REU) is a joint program of the DIMATIA center, Charles University in Prague, The Czech Republic and DIMACS center, Rutgers University, The State University of New Jersey, NJ, USA.

This year's participants from Charles University were students Matěj Konečný, Jana Novotná, Jakub Pekárek, Václav Rozhoň, Jakub Svoboda and Štěpán Šimsa. We spent our time in Piscataway together with more than thirty students from universities all over the United States, Russia, Mexico and Thailand. We participated in the first part of the program at Rutgers University from May 30th to July 16th. This part of the program mainly consists of open mathematical problems being solved by students and led by their mentors. Our mentor was Periklis Papakounstantinou. Students attended several lectures, workshops and tutorials. By the end they also participated in a Field Trip to IBM Headquarters close to New York.

In addition to the scientific program, an important part of the REU is an intercultural experience. At the beginning, Czech students had time to present Czech Republic, its customs and culture. On the other side American students demonstrated culture and lifestyle of people in America. Moreover, the students together participated in many sport activities, hiking and several sightseeing trips.

Six American students were selected to join, together with their graduate coordinator, the Czech students in the second part of the REU which took place at Charles University in Prague from July 20st to August 1st. The US students were Yulia Alexandr, Kayla Cummings, Edgar Jaramillo-Rodriguez, Marina Knittel, Aaron Zhang and their graduate coordinator was Parker Hund.

In Prague, the students attended a series of lectures given by professors mainly from the Department of Applied Mathematics and the Computer Science Institute of Charles University and Midsummer Combinatorial Workshop XXII held from July 31th to August 4th.

Many results and thoughts from the last summer are still being improved and some of them are going to be submitted to international conferences. This booklet presents the results of the Czech students from the REU programme and reports of the American students about their lectures at Prague.

At the end, I would like to thank all the participating students, people at DIMACS and other organizers. Also an important role played people from the both our departments at Prague. I thank them for many helpful

comments, encouraging advices and an overal support. For me it was a perfect experience and I am glad that I could have been a part of this program.

Prague, Winter 2017
Jaroslav Hančl,
Charles University in Prague

List of sponsors

DIMACS
Center for Discrete Mathematics and Theoretical Computer Science
Founded as a National Science Foundation Science and Technology Center

RSJ

CSI
IÚUK

ITI

The participants of the Internation REU programme at Rutgers University.

The participants on the visit to IBM headquaers in US.

Midsummer Combinatorial Workshop XXII

Contents

Convexity and Helly's Theorem

Transcribed by Aaron Zhang from a lecture by Vít Jelínek

1 Introduction

The notion of convexity finds many applications in mathematics and computer science. We start by defining convex combinations and convex sets.

Definition 1.1. Let $x_1, ..., x_n$ be points in $\mathbb{R}^d$. A convex combination of $x_1, ..., x_n$ is a linear combination $a_1x_1 + ... + a_nx_n$ where each a_i is nonnegative and $\sum_{i=1}^{n} a_i = 1$.

For example, in $\mathbb{R}^2$, $(2, 3)$ can be written as a convex combination of $(1, 0)$ and $(4, 9)$ because $(2, 3) = \frac{2}{3}(1, 0) + \frac{1}{3}(4, 9)$. However, $(5, 12)$ is not a convex combination of $(1, 0)$ and $(4, 9)$.

Definition 1.2. A set $S \subseteq \mathbb{R}^d$ is convex if, for any two points $x_1, x_2 \in S$, every point on the line segment between x_1 and x_2 is also in S. In other words, for $0 \leq t \leq 1$, $tx_1 + (1-t)x_2 \in S$.

In terms of convex combinations, a set S is convex if it contains all convex combinations of points in S. Note that the empty set and a set with only one point satisfy the definition of convexity. Intuitively, a set is convex if it has no dents. For example, in $\mathbb{R}^2$, the semicircle of points that lie above the x-axis within distance 1 from the origin is a convex set. However, if we made a dent by removing a smaller semicircle around the origin, the set would no longer be convex, because a line segment between $(-1, 0)$ and $(1, 0)$ would not be entirely contained in the set. Now, we define the convex hull of a set of points.

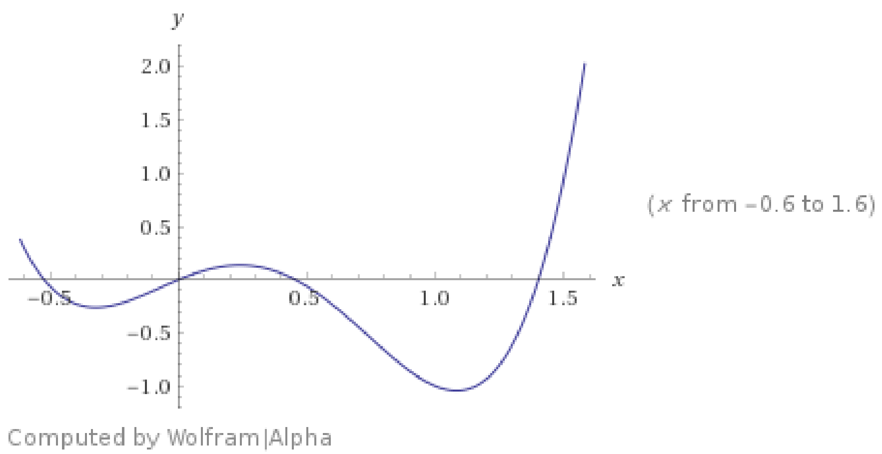
Definition 1.3. The convex hull of a set $S \subseteq \mathbb{R}^d$ is the smallest convex set containing S. Equivalently, the convex hull of S is the set of all convex combinations of points in S.

As the reader should verify, arbitrary intersections of convex sets are convex, so the smallest set containing S is well defined and is the intersection of all convex sets containing S. We have now defined convexity, but before we start proving properties of convex sets, let's see some examples of convexity in action.

2 Applications of convexity

Example 2.1. Suppose we have a function $f : \mathbb{R} \to \mathbb{R}$ and we want to find the value of x that minimizes $f(x)$. For example, $f(x)$ might represent the fuel consumption of a vehicle during a trip if it travels at a speed x. If f is differentiable, one reasonable approach to find the minimum is to start at some value of x and evaluate $f'(x)$. If $f'(x) < 0$, we know that nearby points to the right of x will decrease the value of f, so we can increase x by a small amount and repeat the process. Likewise, if $f'(x) > 0$, we can decrease x by a small amount and repeat the process. The generalization of this idea to higher dimensions is called *gradient descent* and is one of the most widely used optimization techniques in computer science.
In general, this algorithm might get stuck at a local minimum rather than finding the global minimum. For example, consider the plot below of $f(x) = 3x^4 - 4x^3 - x^2 + x$. If we start our algorithm at $x = -0.5$, we might get stuck at the local minimum near $x = -0.3$, which is far from the global minimum near $x = 1.1$.

However, suppose we have a different function f such that the region of



the plane above the graph of f is a convex set, for example $f(x) = x^2$. In this case, we call f a convex function. It turns out that, given a reasonable method of updating x based on $f'(x)$, gradient descent will converge to the global optimum of a convex function from any starting point. Convexity is useful in many other settings in optimization as well.

The study of convex functions and their properties is a well developed area of research in mathematics and computer science. One particularly useful result in probability is *Jensen's inequality*, which bounds the value of a convex function of the expectation of a random variable. Jensen's inequality and other properties of convex functions are often used to derive the fundamental results in the field of information theory.

Example 2.2. *Linear programming* is another optimization framework in computer science that is incredibly versatile and well studied. The goal is to find a setting of variables $x_1, ..., x_n$ that maximizes a given linear objective function $c_1x_1 + ... + c_nx_n$. The settings of the variables are subject to a collection of linear constraints, each of the form $a_1x_1 + ... + a_nx_n \leq b$ for some $a_1, ..., a_n, b$. For example, suppose a company uses wood and metal to produce chairs and tables. A chair requires 10 units of wood and 10 units of metal and generates a profit of \$20. A table requires 30 units of wood and 20 units of metal and generates a profit of \$40. Suppose the company has 500 units of wood and 400 units of metal and must determine the number of chairs c and tables t to produce to maximize the profit. This problem can be formulated as a linear program: maximize $20c + 40t$ subject to:

$$\begin{aligned} 10c + 30t &\leq 500, \\ 10c + 20t &\leq 400, \\ c, t &\geq 0. \end{aligned}$$

How do we solve a linear program? The *simplex* algorithm is one of two widely used methods and makes use of convexity. First, consider the set of points $(c, t) \in \mathbb{R}^2$ that satisfy the constraints above. This set is called the feasible region. The reader should prove that the feasible region is a convex set in $\mathbb{R}^2$, and indeed, that the feasible region of any linear program is a convex set in $\mathbb{R}^n$, where n is the number of variables. In this example, the feasible region is a convex polygon bounded by four sides that correspond to the four constraints (sketch a diagram to visualize this). This polygon is the convex hull of its four vertices. (The higher dimensional analogue of a convex polygon is a convex polytope, and if the feasible region is a bounded set, the convex polytope is again the convex hull of its vertices.) In fact, the choice of (c, t) that maximizes the objective function must be at one of the vertices. (Given that the polygon is the convex hull of its vertices, what can we say about the value of the objective function at any interior point?) The simplex algorithm starts at an arbitrary vertex of the feasible region and repeatedly moves to an adjacent vertex that increases the value of the objective function until it reaches a vertex whose value is higher than the value at all its neighbors. Think about why the value at such a vertex must be the maximum value of the objective function.

With these examples in mind, it should be clear that studying the properties of convexity is both worthwhile for its own sake, and to develop the theoretical foundations of many practical applications. Our focus here will be proving and making use of *Helly's theorem*, a classic result in discrete geometry and convex analysis.

3 Helly's theorem

Our main result will be

Theorem 3.1. (Helly's Theorem) Suppose $C_1, ..., C_n$ is a finite family of convex sets in $\mathbb{R}^d$ and $n \geq d+1$. Then if the intersection of every $d+1$ of these sets is nonempty, the intersection of all the C_i is nonempty.

In $\mathbb{R}^2$, try to come up with an example of a family of convex sets where each pair of sets intersect, but the intersection of all the sets is empty. This shows that $d+1$ cannot be replaced with d in the theorem. Helly's theorem says that if we require that each triplet of sets intersect, the intersection of all the sets must be nonempty. We will prove Helly's theorem using the following lemma, which is interesting in its own right.

Lemma 3.2. (Radon's Lemma) Let $P \subseteq \mathbb{R}^d$ be a set containing at least $d+2$ points. Then we can partition P into two sets P^+, P^- such that the convex hulls of P^+ and P^- intersect.

Proof. Without loss of generality, let $P = \{p_1, ..., p_{d+2}\}$ be a set of size $d+2$. Define $v_1, ..., v_{d+1}$ where $v_i = p_i - p_{d+2}$. Then $v_1, ..., v_{d+1}$ is a set of $d+1$ vectors in $\mathbb{R}^d$, and hence linearly dependent. Thus, we can find $a_1, ..., a_{d+1}$, not all 0, such that $a_1 v_1 + ... + a_{d+1} v_{d+1} = 0$. Expanding the formula for each v_i, we have

$$a_1 p_1 + ... + a_{d+1} p_{d+1} + (-a_1 - ... - a_{d+1}) p_{d+2} = 0 \tag{1}$$

Note that the sum of the coefficients in this linear combination is 0. Let P^+ be the set of points in P with positive coefficients, and let P^- be the rest of the points in P. Rearrange equation 1 by moving the terms containing points in P^- to the right side. Now the sum of coefficients on each side of the equation are equal, so divide both sides by the same amount so the sum of coefficients on each side is 1. This gives us a convex combination of points in P^+ that is also a convex combination of points in P^-. □

We can now prove Helly's theorem.

Proof. We proceed by induction on n. As a basis for the induction, note that Helly's theorem holds when $n = d + 1$. Now let $n \geq d + 2$, and let $C_1, ..., C_n$ be a family of convex sets satisfying the conditions in Helly's theorem. By the induction hypothesis, if we remove any one set from this family, the remaining sets have a point in common. Let p_i be the point in common of the remaining sets when C_i is removed. Because $p_1, ..., p_n$ is a set of at least $d + 2$ points in $\mathbb{R}^d$, Radon's Lemma gives us a partition P^+, P^- of these points such that the convex hulls of P^+ and P^- intersect. Let p be any point in this intersection; we claim that p belongs to all of $C_1, ..., C_n$. Indeed, for each C_i, the point p_i belongs to only one of the partitions P^+, P^-, and thus C_i contains all points in the other partition. Because C_i is convex, C_i also contains all points in the convex hull of the other partition, including p. □

Exercise 1. Prove the following variant of Helly's theorem. Let $C_1, ..., C_n$ be a family of convex sets in $\mathbb{R}^d$ with $n \geq d+1$. Now suppose the intersection of every $d + 1$ of these sets contains some ball of radius 1. Show that the intersection of all the sets contains some ball of radius 1.

4 Applications of Helly's theorem

Helly's theorem may sound like a very particular statement, but in fact finds many surprising applications in discrete geometry. Here we present two of them.

Definition 4.1. Let X be a finite set of n points in $\mathbb{R}^d$. For $\alpha \in [0, \frac{1}{2}]$, an α-centerpoint of X is a point $y \in \mathbb{R}^d$ such that every closed half-space with y on the boundary contains at least αn points of X.

Note that a centerpoint of X need not belong to X. For example, if X is a triangle in $\mathbb{R}^2$, any point in the triangle is a $\frac{1}{3}$-centerpoint of X. In fact, any finite set of points in $\mathbb{R}^2$ has a $\frac{1}{3}$-centerpoint. This is a consequence of the following result, whose proof uses Helly's theorem.

Theorem 4.2. If X is a finite set of points in $\mathbb{R}^d$, then X has a $\frac{1}{d+1}$-centerpoint.

Proof. Let n denote the size of X. We will prove the theorem in the case $n \geq d+2$; the other case should be verified by the reader. Let F be the family of subsets $Y \subseteq X$ such that Y contains more than $\frac{d}{d+1}n$ points of X. Note that any $d+1$ sets in Y intersect. We would like to apply Helly's theorem, so define a new family F' by taking the convex hull of each set in F. Now Helly's theorem guarantees a point p contained in all the sets in F'; we claim p is a $\frac{1}{d+1}$-centerpoint. If not, then there would be a hyperplane through p such that one of the sets in F lies entirely on one side, contradicting the fact that p is in the convex hull of each set in F. $\square$

The most interesting step in the last proof was finding an appropriate family satisfying the conditions of Helly's theorem. We will use this idea again in the proof of the next result.

Definition 4.3. Let $A, B \subseteq \mathbb{R}^d$. We say that a hyperplane separates A and B if A and B lie entirely in opposite open half-spaces determined by the hyperplane (the hyperplane cannot intersect A or B).

Theorem 4.4. (Kirchberger's Theorem) Let A and B be finite subsets of $\mathbb{R}^d$. Suppose $|A \cup B| \geq d+2$, and for every subset $X \subseteq A \cup B$ of size $d+2$, there is a hyperplane separating $X \cap A$ and $X \cap B$. Then there is a hyperplane separating A and B.

In other words, if any $d+2$ points of A and B are separable by a hyperplane, the entire sets A and B are separable by a hyperplane.

Proof. We will prove Kirchberger's theorem for $d = 2$; the reader should verify that the result generalizes to higher dimensions. For $a, b, c \in \mathbb{R}$ with a and b not both 0, let $L^+_{a,b,c}$ be the set of points (x, y) in $\mathbb{R}^2$ such that $ax + by > c$. Similarly, define $L^-_{a,b,c}$ to be those points (x, y) such that $ax + by < c$. $L^+_{a,b,c}$ and $L^-_{a,b,c}$ are the two open half-spaces separated by the line $ax + by = c$. Let $A = \{p_1, ..., p_n\}$, and let $B = \{q_1, ..., q_m\}$. For each $p_i \in A$, let $X_i = \{(a, b, c) \in \mathbb{R}^3 : p_i \in L^+_{a,b,c}\}$. Similarly, for each

$q_j \in B$, let $Y_j = \{(a,b,c) \in \mathbb{R}^3 : q_j \in L^-_{a,b,c}\}$. We wish to find a point $(a,b,c) \in \mathbb{R}^3$ that lies in the intersection of all the X_i and Y_j. Each X_i and Y_j is convex (verify this!), and the conditions of Kirchberger's theorem tell us that any 4 of these sets intersect. Helly's theorem guarantees a point in the intersection of all these sets. $\square$

5 Variants of Helly's theorem

Helly's theorem has inspired research toward a number of similar results. We start by stating an infinite version of Helly's theorem for compact convex sets.

Theorem 5.1. Let F be an infinite family of compact convex sets in $\mathbb{R}^d$. Then if every $d+1$ sets in F intersect, all the sets in F intersect.

The compactness condition is necessary. Try to find an example in $\mathbb{R}$ of an infinite family of convex sets that are closed but not bounded, such that any two sets intersect but there is no point that lies in all the sets. Likewise, try to find an example of an infinite family of convex sets that are bounded but not closed where the conclusion of the theorem fails. The proof of the infinite version of Helly's theorem, which we omit, follows from the finite version of Helly's theorem and some facts from analysis about compact sets. The following is a quantitative version of Helly's theorem:

Theorem 5.2. For each $d \in \mathbb{N}$, there exists $\epsilon_d > 0$ such that the following holds. Let $C_1, ..., C_n$ be a finite family of convex sets in $\mathbb{R}^d$. If $n \geq 2d$ and every $2d$ sets in the family have an intersection with volume at least 1, then the volume of the intersection of all the sets is at least ϵ_d.

We omit the proof, but encourage the reader to think about why we cannot replace $2d$ with $2d-1$ in the theorem. Try to find 4 convex sets in $\mathbb{R}^2$ such that any 3 sets have an intersection with infinite area, but the area of the intersection of all 4 sets can be arbitrarily small. Then generalize this to higher dimensions.
Finally, we encourage the reader to look up the notion of a Helly family, which pertains to a collection of Helly-type results beyond discrete geometry. We present one such example in graph theory.

Theorem 5.3. (Helly property of trees) If T is a tree and $T_1, ..., T_k$ are pairwise intersecting subtrees of T, then T has a vertex that belongs to all of the T_i.

Exercise 2. Prove the Helly property of trees. It may be helpful to use the same proof structure as the proof of Helly's theorem on convex sets: induction on the size of the family, in this case k.

Exercise 3. A useful result in mathematics and computer science is that, given any n-vertex tree, we can find a vertex whose deletion leaves subtrees each of size at most $\frac{n}{2}$. Many proofs of this result are known, and most are algorithmic: they specify a procedure to find such a vertex. Try to use the Helly property of trees to provide a short combinatorial proof of this result.

There is much more to explore in the study of convexity and discrete geometry. The speaker recommends the two series of lectures on discrete geometry by Matoušek in the references below.

References

[1] Douglas West. *Introduction to Graph Theory*. Prentice Hall, Upper Saddle River, New Jersey, 2001.

[2] Jiří Matoušek. *Lectures on Discrete Geometry*. Springer-Verlag, New York, New York, 2002.

[3] Jiří Matoušek. *Using the Borsuk-Ulam Theorem*. Springer-Verlag, Berlin, Germany, 2003.

[4] Sanjoy Dasgupta and Christos Papadimitriou and Umesh Vazirani. *Algorithms*. McGraw-Hill, New York, New York, 2008.

[5] Wolfram Alpha, `http://www.wolframalpha.com/`

Group isomorphism problem and nondeterministic space-bounded classes

Matěj Konečný, Jakub Pekárek, Václav Rozhoň, Štěpán Šimsa

1 Introduction

The graph isomorphism problem (GI) is the following problem: one is given two graphs and has to decide whether they are isomorphic or not. Although easy to state, the graph isomorphism problem is one of the few problems known to be in NP for which we do not know whether it is NP-complete or it is in P, although it is strongly believed that the problem is not NP-complete. One reason for this is the recent celebrated quasipolynomial algorithm by Babai [5], the other is a famous result from complexity theory stating that if GI is NP-complete, then the polynomial hierarchy collapses to the second level ($\mathrm{PH} = \Sigma_2$) [6]. The key component of the proof is an Arthur-Merlin interactive proof protocol [3, 7] for the graph nonisomorphism problem.

The group isomorphism problem (ΓI) is the problem of deciding whether two groups are isomorphic. The problem can be reduced by polynomial reduction to GI [8]. This means that also for ΓI one can show that if it is NP-complete, the polynomial hierarchy collapses to the second level. It is believed, though, that ΓI is strictly easier than GI. One reason for this is that $\Gamma\mathrm{I} \subseteq \mathrm{NSC}_2$, a class that is believed to be substantially smaller than NP.

We study the group isomorphism problem from the complexity-theory point of view. The tools we use are mainly the ones that were used to prove the above-mentioned complexity-theoretical result on GI and for brevity we usually do not give full proofs, mainly when they go along similar lines as the appropriate proofs of results concerning graph isomorphism (for details, see, e.g. [2] or [4]).

Besides ΓI we also study the quantifier hierarchy defined over space bounded classes $\mathrm{SC}_k = \mathrm{TIMESPACE}(poly(n), \log^k(n))$. The classes of our interest are $\mathrm{SC}_1 = L$ (which was already extensively studied and seems to be well-understood), and SC_2 that we study with respect to the group isomorphism problem. We use the notation NSC_k for the nondeterministic variant of SC_k.

The starting point of this enterprise is the following unpublished result due to Papakonstantinou.

Theorem 1.1. ΓI is in NSC_2.

Sketch of proof. Each group can be generated by $\mathcal{O}(\log(n))$ of its elements. Guess the generators of both the first and the second group (we need $\mathcal{O}(\log^2(n))$ bits to store them), close the both sets under the inverse operation. Further guess the isomorphism of these two generating sets. With a help of a deterministic algorithm for finding paths (Reingold's algorithm) we can deterministically express each element of a group as a product of the generators (path in the corresponding Cayley graph). Then we can easily check whether the guessed function is really an isomorphism. $\square$

For believing that a problem is easy, it is valuable to know that not only the problem itself lies in a class of relatively easy problems, but also its complement does. To this end we define space bounded analogues of so-called interactive proof systems in Section 2. This is done in a non-standard way as the standard approach would give classes that are too weak for our methods. We believe that our approach may be of its own interest, as it closely relates to the standard polynomial hierarchy. In the subsequent Sections 3 and 4 we then place the group non-isomorphism problem in classes defined by space-bounded interactive proofs with public and private coins, respectively.

2 Different models

While the following results are stated for the Σ hierarchy (starting with existential quantifier), one can extend the results to Π hierarchy in a straightforward manner.

Definition 2.1. (Strong and weak hierarchy) For class of problems $C = \mathrm{TIMESPACE}(f, g)$ we define two types of hierarchies. *Weak hierarchy* is the hierarchy defined by alternations and we will denote its levels Σ_k^C and Π_k^C. *Strong hierarchy* is defined in a similar manner and we use symbols $\overline{\Sigma_k^C}$ and $\overline{\Pi_k^C}$ to denote its levels.

We say that a multi-tape Turing machine M is nondeterministic-read-only, if it has one input tape that can be read in both directions, one one-way write-only output tape and several one-way read-only nondeterministic (or random) tapes (i.e. it reads the bits in the given order and each can be read only once). The nondeterministic tapes are not space-bounded.

Both hierarchies (where either $S = \Sigma_k^C$ or $S = \overline{\Sigma_k^C}$) can be defined as sets of languages such that $L \in S$ if and only if there exists a nondeterministic-read-only machine M with resources in C and

$$x \in L \Longleftrightarrow \exists a_1 \forall a_2, \ldots, Q a_k : M(x, a_1, a_2, \ldots, a_k) = 1.$$

The difference between the weak and strong hierarchies is how the machine M gets the nondeterministic bits. In the weak hierarchy they are all on one tape written in the same order as is the order of quantifiers[1], while in the strong hierarchy there is one tape for every a_i.

The following observations can be seen immeadiately from the definition:

Observation 2.2. For any $C \subseteq C'$ we have $\Sigma_k^C \subseteq \Sigma_k^{C'}$ and $\overline{\Sigma_k^C} \subseteq \overline{\Sigma_k^{C'}}$.

Moreover, $\Sigma_1^C = \overline{\Sigma_1^C}$ and $\Sigma_k^C \subseteq \overline{\Sigma_k^C}$.

We can also easily observe that the two notions are the same for $C = \mathrm{TIMESPACE}(poly, poly)$, because the machine can start by copying the content of its non-deterministic tape(s) on the worktape.

Observation 2.3.

$$\Sigma_k^{\mathrm{P}} = \overline{\Sigma_k^{\mathrm{P}}}.$$

In the following paragraphs we are going to use the interactive proof classes MA and AM. For precise definitions see [2]. Here, by $(\exists^+ x)\varphi(x)$ we mean that for at least $\frac{2}{3}$-fraction of feasible choices of x, the formula $\varphi(x)$ is satisfied. Using this notation, one can say that a language L is in MA if there is a polytime computable formula $\varphi(x, a, r)$ and $x \in L \leftrightarrow$

[1] this is the usual definition

$(\exists a)(\exists^+ r)\varphi(x,a,r)$ and $x \notin L \leftrightarrow (\forall a)(\exists^+ r)\neg\varphi(x,a,r)$, for AM the order of the quantifiers is reversed.

The classes MA and AM can be also defined in both the weak and the strong sense in a completely analogous manner to the SC_k hierarchy. We will use $\overline{\mathrm{MA}_{\mathrm{SC}_k}}$ is in $\overline{\mathrm{AM}_{\mathrm{SC}_k}}$ for the strong classes.

Note that due to the space restriction we are not able to amplify the probability of success in the weak notion of AM and MA by grouping more than constant number of rounds of the same protocol in one interaction.

Because of this, we choose to work with the strong notion with multiple tapes. In this notion we can amplify probabilistic protocols by their repeated application and by an argument completely analogous to the one that $\mathrm{MA}_P \in \mathrm{AM}_P$ [2] one can prove that $\overline{\mathrm{MA}_{\mathrm{SC}_k}}$ is in $\overline{\mathrm{AM}_{\mathrm{SC}_k}}$, as switching the order of quantifiers does not alter the arrangement of the tapes.

However, we show that this notion of hierarchy is very strong, as SAT $\in \overline{\mathrm{MA}_L}$:

Proposition 2.4. $\mathrm{NP} \subseteq \overline{\mathrm{MA}_L}$.

Proof. (sketch) We solve SAT by a machine from $\overline{\mathrm{MA}_L}$. Note that any problem from NP is reducible to SAT by L-reductions.

Nondeterministic tape of our machine contains an assignment of variables of the formula. We uniformly randomly choose one of m clauses from the formula and verify that the clause is satisfied by the non-deterministic assignment. If the assignment is not satisfactory, then there is at least one unsatisfied clause, and a probability at least $1/m$ of choosing it and hence answering correctly. We amplify the probability by repeating this protocol. □

Similarly, one can easily observe that $\mathrm{NP} \subseteq \overline{\Sigma_2^L}$. Note that the classical property of the weak hierarchy is that for every $C = \mathrm{TIMESPACE}(f,g)$ it holds that $\Sigma_1^C = \Pi_1^C$ implies $\bigcup_{i=1}^n \Sigma_i^C = \Sigma_1^C$ [2], this does not seem to hold in the strong notion of hierarchy. This is because for an L machine we actually know that that the premise is true (NL = coNL), but we will soon see that the consequence of an analogous result would mean that PH = NL.

Another nonstandard feature of the model is that derandomization of $\overline{\mathrm{MA}_L}$ (i.e., proof of $\overline{\mathrm{MA}_L} = \mathrm{NL}$) would imply that NP = NL, as SAT $\in \overline{\mathrm{MA}_L}$ (choose randomly a clause from input formula and verify that it is satisfied; this proccess can be repeated to amplify sufficiently the probability of rejecting wrong input).

We follow by clasifying the complete problems for both the weak and the strong notion of the hierarchy in the interesting case when the underlying complexity class C is equal to $\mathrm{SC}_k = \mathrm{TIMESPACE}(poly(n), \log^k(n))$.

2.1 Complete problems for the hierarchy

The complete problem for $\overline{\Sigma_k^{\mathrm{SC}_m}}$ is similar to the complete problem for the polynomial hierarchy. Its corresponding language contains formulas with their pathwidth bounded by $\log^m(n)$ satisfying:

$$\varphi \in L \iff (\exists a_1)(\forall a_2)\dots(Qa_k)\varphi(a_1, a_2, \dots, a_k) = 1.$$

For the proof of the case with $k = 1$, as well as the definition of pathwidth for formulas see [1, 9]. Note that the formula has to be given together with its path decomposition. The general case can be easily proved in the same way.

The complete problem for $\Sigma_k^{\mathrm{SC}_m}$ is the same as the complete problem for $\overline{\Sigma_k^{\mathrm{SC}_m}}$ with the additional restriction that in the pathwidth decomposition there are at first the bags with variables from a_1, then variables from a_2, etc. (variables from a_i and a_{i+1} can intersect in one bag). The proof is straightforward and similar to the previous case.

2.2 Strong hieararchy and PH

In this section we prove that there is a strong relation between the strong hierarchy over L and the standard PH hierarchy. Specifically, we prove the following theorem:

Theorem 2.5. For every $k \geq 2$:

$$\Sigma_{k-1}^{\mathrm{P}} \subseteq \overline{\Sigma_k^{\mathrm{L}}}$$

First recall that any problem in Σ_k^{P} is L-reducible to $\Sigma_k^{\mathrm{P}}\mathrm{SAT}$ (the k-quantifier satisfiability problem). Now we define an auxiliary model of the hierarchy $\widetilde{\Sigma_k^{\mathrm{L}}}$ that is defined as $\overline{\Sigma_k^{\mathrm{L}}}$ except that we do not restrict the machine to read the nondeterministic tapes only once.

Lemma 2.6. $\Sigma_k^{\mathrm{P}}\mathrm{SAT} \subseteq \widetilde{\Sigma_k^{\mathrm{L}}}$.

Proof. We propose a $\widetilde{\Sigma_k^{\mathrm{L}}}$ machine solving Σ_k^{P}SAT. Suppose the formula is already in the CNF form (otherwise we construct an equivalent CNF formula in an online manner). We interpret the nondeterministic tapes as the assignments of corresponding variables, take the clauses one by one and for each we check whether the assignment is correct (we scan the nondeterministic tape once each time we need a nondeterministic variable). $\square$

Now we are ready to prove Theorem 2.5. Although quite technical when written down, the idea behind the proof is very easy. We want to prove the inclusion $\widetilde{\Sigma_{k-1}^{\mathrm{L}}} \subseteq \overline{\Sigma_k^{\mathrm{L}}}$. This we will simply do by repeating the nondeterministic bits on each tape enough times so that we can simulate multiple reads. Then, of course, one has to check whether all the copies are the same. To do that for, say, $(\forall a)\varphi(x,a)$, one can devise new equi-satisfiable formula

$$(\forall a)(\exists i)\,(\text{some copies in } a \text{ differ on the } i\text{-th bit} \vee \varphi(x,a))\,.$$

If the copies differ, then the bits on tape a are not part of what we want to check, so we simply accept. Otherwise we can simulate multiple reads by having enough copies of the nondeterministic bits. For the existential quantifier, one universally (over index i) checks whether all the bits on position i are the same and the formula is satisfied at the same time.

The formal proof follows:

Proof of Theorem 2.5. By Lemma 2.6 we know that $\Sigma_{k-1}^{\mathrm{P}} \subseteq \widetilde{\Sigma_{k-1}^{\mathrm{L}}}$, so it is enough to show that $\widetilde{\Sigma_{k-1}^{\mathrm{L}}} \subseteq \overline{\Sigma_k^{\mathrm{L}}}$. Let M be a machine working in $\widetilde{\Sigma_{k-1}^{\mathrm{L}}}$ and $L(M)$ the corresponding language. We will construct a machine M' in $\overline{\Sigma_k^{\mathrm{L}}}$ such that $L(M') = L(M)$.

Denote the nondeterministic tapes of M resp. M' as $t_1, \dots, t_{k-1}$ resp. $t'_1, \dots, t'_k$ (in the order of the corresponding quantified variables). We know there exists a polynomial $P(n)$ such that M does at most $P(n)$ steps for $|x| = n$. We will treat every tape $t'_1, \dots, t'_k$ as having two sections. First of them will be long $\lceil \log(P(N)) \rceil = O(\log(n))$ bits and the second one will have $P(n)$ parts of $P(n)$ bits. The first tape does not have the first section and the last tape, t'_k, has only the first section. We will start by reading the first sections and storing them to our work tape (so for every tape $2 \leq i \leq k$ we have some number b_i). Now M' simulates M step-by-step, but after each step it jumps to the next part of the second section of each nondeerministic tape.

Whenever we read part of some tape i we remember its b_{i+1}-th bit and check if it is equal to the b_{i+1}-th bit of the previous part of same the tape (for any tape we remember at most two bits for every tape by throwing away old bits).

If we found two bits that were not equal on the first (existential) tape, we reject. Otherwise, if we found two bits that were not equal on the second (universal) tape, we accept. Generally, if there was no difference on tapes $t'_1, \ldots, t'_{l-1}$ and there is a difference on tape t'_l we reject if l as an existential tape and accept otherwise.

If we did not find any not matching bit on any of the tapes we accept/reject base on our simulation of M.

We now prove that $L(M) = L(M')$.

We know that

$$x \in L(M) \iff (\exists a_1)(\forall a_2)\ldots M(x, a_1, a_2, \ldots, a_{k-1}) = 1. \tag{2}$$

We have M' and the content of the first tape is a'_1, the content of the second tape is $b_2 a'_2$, third tape $b_3 a'_3$, and so on until the last tape contains only b_k.

First suppose that $x \in L(M)$. We want to show that $x \in L(M')$. We will choose a'_1 (the content of first tape of M') as $a_1^{P(n)}$ where a_1 is what we get from Equation 2 ($a_1^{P(n)}$ means $P(n)$ copies of a_1 padded with zeros to length $P(n)$). That means that no matter what b_2 is we will not find a difference between two parts on the first tape and so we will not reject because of this.

Then there are two possibilities for a'_2. Either $a'_2 = c^{P(n)}$ for some c or not. If not then we can choose b_3 such that we find a difference between two different parts of a'_3 and then we accept (as we wanted). Otherwise we take $a_2 = c$ and Equation 2 gives us a_3. Now we continue with choosing $a'_3 = a_3^{P(n)}$ and proceed the same as with a'_1.

When we constructed a'_{k-1} and b_k we either accepted because of some difference in a'_{2l} or all the tapes $a'_1, \ldots, a'_{k-1}$ contain $P(n)$ exact copies of $a_1, \ldots, a_{k-1}$ for which M accepts. But then our machine M' simulates the machine M with the access to the same inputs and because M accepted, M' will accept as well.

The second part of the proof $x \notin L(M) \implies x \notin L(M')$ can be done analogically. $\square$

Trivially, the second inclusion would be true for any class $\mathrm{L} \subseteq C \subseteq \mathrm{P}$ and then $\overline{\Sigma_k^{\mathrm{L}}} \subseteq \overline{\Sigma_k^{C}}$ so the Theorem 2.5 is true for any such class C.

Theorem 2.7. For every $k \geq 1$:

$$\overline{\Sigma_k^{\mathrm{L}}} \subseteq \Sigma_{k-1}^{\mathrm{P}}$$

Proof. For $k = 1$ we get the well-known theorem NL $\subseteq$ P. For general k the proof goes along the same lines. State of the machine at each step of computation can be described by the content of work tape, position of heads reading input/work/nondeterministic tapes and the transition state of the machine. All of this information can be described by $\mathcal{O}(\log(n))$ bits.

Now when simulating a $\overline{\Sigma_k^{\mathrm{L}}}$ machine M by $\Sigma_{k-1}^{\mathrm{P}}$ machine M' we at first guess the assignment of the first $k-1$ tapes of M on the $k-1$ corresponding tapes of M'. Then we build the state graph of the machine M of size $poly(n)$ (the same as in the proof NL $\subseteq$ P). Because the content of the first $k-1$ tapes is fixed now the only states with outdegree two in the state graph are those that correspond to reading the last nondeterministic tape while all of the other states have outdegree one. Now the existence of the content such that M accepts is clearly equivalent to existence of an appropriate path in the state graph that can be found in polynomial time. □

3 ΓNI is in $\mathrm{IP}_{\mathrm{SC}_2}$

In this section we give an interactive protocol for group nonisomorphism for machine in SC_2. The proof goes along the same lines as the classical proof of graph nonisomorphism from [2] (choose one group at random, permute its elements and send it to the prover asking him what group we have sent). However, as the memory of the verifier is limited, it cannot store the whole information of the chosen permutation of the group. We overcome this problem by not sampling uniformly from the set of all permutations, but rather we choose only from a subset of permutations that are generated from the partial permutation of the generating sets of size $\mathcal{O}(\log n)$. This set of generators can be stored in the limited memory of an SC_2 machine.

The reason why we can use the notion of generating sets is the following simple lemma (it can be viewed as the special case of the Alon-Roichmann theorem).

Lemma 3.1. Let Γ be a group with at least two elements. Then for each ε there is a c_ε such that at most ε fraction of subsets of Γ of size $c_\varepsilon \log(n)$ is not generating.

Sketch of proof. One by one, we will randomly pick elements into the set $A \subseteq \Gamma$. As the set generated by A is a subgroup of Γ, either $\langle A \rangle = \Gamma$ and we have a generator, or by Lagrange's theorem $|\langle A \rangle| \leq |\Gamma|/2$, hence the probability, that in the next step we pick an element outside of $\langle A \rangle$ is at least $\frac{1}{2}$ and thus enlarge the size of the generated subgroup at least by a factor of two (again by Lagrange's theorem). Then it is enoug to pick a suitable c_ε to get the probability of failure below ε. $\square$

As we have already stated, the crucial idea is to sample a permutation of group elements only from the limited set induced by a partial permutation of a small generating set. The heart of the argument is then to show that if our two groups are isomorphic, then the resulting distribution of the generated Cayley tables for the first group is the same as the one for the second group.

Definition 3.2. For a permutation $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ we define $\widetilde{\pi}$ to be the corresponding permutation matrix.

Theorem 3.3. There is a machine $M \in \mathrm{SC}_2$ such that

- M takes as an input a Cayley table C_Γ and an ordered set of indices $(s_i)_{i=1}^k$;
- M outputs *NO* if elements $(s_i)_{i=1}^k$ do not generate Γ;
- otherwise M outputs $\bar{C}_\Gamma = \widetilde{\pi} C_\Gamma$ for some permutation π, where the elements of $\bar{C}_\Gamma$ are canonically renumbered $0, 1, 2, \ldots$ in this order and the elements of s_i form an initial segment of the table following the order $s_1, \ldots, s_k$; and
- $\widetilde{\pi} M(C_\Gamma, (s_i)_{i=1}^k) = M(\widetilde{\pi} C_\Gamma, (\pi s_i)_{i=1}^k)$ holds for every permutation π of group elements.

Moreover, for π an injective function from the generating set $(s_i)_{i=1}^k$ to Γ it holds that $M(C_\Gamma, (s_i)_{i=1}^k) = M(C_\Gamma, (\pi s_i)_{i=1}^k)$ if and only if π (uniquely) extends to an automorphism of Γ.

To propose such an algorithm we use a Cayley graph representation of Γ. As a subroutine we use Reingold's algorithm. We will need that this algorithm satisfies a natural property that its output does not depend on the underlying representation of the graph it uses (i.e. on the order of the vertices).

Definition 3.4. Let G be a k-regular graph. We define $S(G)$ as a representation of G via an ordered set of ordered lists of neighbors of each vertex, i.e. $S(G)_i = (v_1, v_2, \ldots, v_k)$, where vertices $\{i, v_j\} \in E(G)$ for every $j \in \{1, \ldots, k\}$.

Definition 3.5. For permutation $\pi : \{1, \ldots, n\} \to \{1, \ldots, n\}$ we define $\pi(S(G))$ as permutation of vertices of G while preserving the order of edges incident to every vertex. So for every $i \in \{1, \ldots, n\}$:

$$\pi(S(G))_{\pi(i)} = (\pi(v_1), \pi(v_2), \ldots, \pi(v_k)),$$

where $S(G)_i = (v_1, \ldots, v_k)$.

Proposition 3.6 (s, t connectivity in small space)**.** There is a machine $M_{st} \in \mathrm{L}$ such that

- M_{st} takes as an input $S(G)$ and two vertices s and t of G,
- M_{st} outputs list of indices from $S(G)$ defining path from s to t or $false$ if no such path exists,
- $\pi(M_{st}(S(G), s, t)) = M_{st}(\pi(S(G)), \pi(s), \pi(t))$ for every permutation π of vertices of G

Proof. One can check that the Reingold's algorithm has all the desired properties. $\square$

Now we use this algorithm to prove Theorem 3.3.

Sketch of proof of Theorem 3.3. First we close the $(s_i)_{i=1}^k$ under taking inverse and add the unit element, such that the closure order the original elements of s_i go first, then the unit element (if not present in s_i already) and then all the missing inverses, again following the order s_i. We get an l-tuple of logarithmic size (from now on by s we mean $s = (s_i)_{i=1}^l$ containing these new elements). Now we can define a Cayley graph G from this l-tuple and represent it as $S(G)$, where the order of neighbours of each vertex is the same as the order in this l-tuple. We can use machine M_{st} from Proposition 3.6 to produce a path from every element in G to 1. If one of these paths does not exist than the graph is not connected and that is equivalent to the fact that the original k-tuple was not a generating set. In this case we return NO. Otherwise, for every element g in the group we define its position in the Cayley table we output as follows – if $g = s_i$ then it is i and otherwise it is $l + x$ where x is the number of elements among $\Gamma \setminus \{s_i\}$ that are smaller than g with respect to lexicographical order of the paths from 1 to g. To compare two elements with respect to this order we run two instances of M_{st} simultaneously. We return the Cayley table C_Γ element by element. To find the index stored at position (i, j) we find the two elements with indices i and j in our lexicograhical order, multiply them (we use the Cayley table from input tape) and return the index of the new element in our order.

Now the fourth property of the theorem follows from Proposition 3.6. The last property follows from the fact that the permutation of generators has a unique extension to the whole group. If the equality $M(C_\Gamma, (s_i)_{i=1}^k) = M(C_\Gamma, (\pi s_i)_{i=1}^k)$ holds, then the permutation of generators uniquely extends to an automorphism of Γ. On the other hand, no automorphism other than identity is identical on any set of generators. $\square$

Now we can easily build the sampling procedure with the desired property that if the two groups are isomorphic, then the distribution of groups we sample does not depend on the group we choose to sample from.

Theorem 3.7. ΓNI has an interactive protocol with private bits and verifier from SC_2.

Idea. Pick $p \in \{0, 1\}$ uniformly at random. Pick $c \cdot \log(n)$ different group element indices $s_i, s_2, \ldots, s_{c \log n}$ at random. Generate C_{Γ_p} permuted according to (s_i) one element at a time using the algorithm from 3.3 and send it to the prover. Prover replies with $q \in \{0, 1\}$ indicating which group on the input is isomorphic to the group defined by C_{Γ_p}. Accept if and only if $p = q$. This can be easily amplified in such a way that the probability of success can be any constant (combine several rounds of interaction in one). $\square$

4 ΓNI is in $\mathrm{AM}_{\mathrm{SC}_2}$

Definition 4.1. ($\mathrm{AM}_{\mathrm{SC}_2}$ model) Input tape (random access), random tape (read once), non-deterministic tape (read once), memory $log^2(n)$, run-time $poly(n)$.

As in the corresponding proof of GI we use the so-called lower bound method, where we choose a suitable set representing symmetries of our two groups and then argue that the set is substantially larger if the two groups are non-isomorphic. As in the case of the IP protocol from the previous section we choose only a suitable subset of symmetries that can be described in small space – in this we differ from the corresponding proof for GI where one works with the set of all permutations of the given graph.

The suitable set S is defined as a union of S_1 and S_2 where for $i \in \{1, 2\}$ we have $S_i = \{(\widetilde{\Gamma}, \pi) \mid \widetilde{\Gamma} = M(C_{\Gamma_i}, s_1, \ldots, s_k), (s_1, \ldots, s_k) \in \Gamma_i^k, \pi \in \mathrm{aut}(\widetilde{\Gamma})\}$ where M is the machine from Theorem 3.3.

Observe that

$$|S_i| = |\{s_1, \ldots, s_k \in \Gamma_i : s_1, \ldots, s_k \text{ are disjoint and generating}\}|.$$

This immediately follows from the last part of Theorem 3.3, as we know that $M(C_{\Gamma_i}, s_1, \ldots, s_k) = M(C_{\Gamma_i}, t_1, \ldots, t_k)$ iff the partial function mapping each s_i to t_i extends to an automorphism π of Γ_i. On the other hand, any automorphism of Γ also specifies uniquely a partial automorphism of its first k elements.

From Lemma 3.1 we can easily infer that there is a suitable c such that after defining $k = c \log n$ we have

$$\begin{aligned} 0.9 \cdot n(n-1) \cdots (n-k+1) \leq |\{s_1, \cdots, s_k \in \Gamma_i, \text{ disjoint and generating}\}| \\ \leq n(n-1) \cdots (n-k+1). \end{aligned}$$

In other words, most of the sets are generating.

Further observe that if $\Gamma_1 \simeq \Gamma_2$ then $|S| \leq n(n-1) \ldots (n-k+1)$. If $\Gamma_1 \not\simeq \Gamma_2$ then $|S| \geq 1.8 \cdot n(n-1) \ldots (n-k+1)$. Thus we are in the right situation for the set lower bound protocol. Note that any member of the disjoint union $S_1 \sqcup S_2$ can be represented by the sequence $s_1, \ldots, s_k$, π restricted to this set and one bit b stating whether the sequence refers to the element of the first or the second group. Thus, any member of S can be represented by $\mathcal{O}(\log^2 n)$ bits. For successful application of the set lower bound protocol we first need to verify that given such a representation $(s_1, \ldots, s_k), \pi, b$ we can verify that this is, indeed, a member of S. Then we need to find an appropriate 2-universal family of hash functions such that for given y we can check whether $h(s_1, \ldots, s_k, \pi, b) = y$. The latter problem is subject to the following proposition.

Proposition 4.2. There is a 2-universal family of hash functions $\mathcal{H}$ from $\{0,1\}^{p(n)}$ to $\{0,1\}^k$ with $p(n)$ being a polynomial in n and $k = \mathcal{O}(\log^2(n))$ such that:

- Computation of $h(x), h \in \mathcal{H}$ is in SC_2 if we have random access to bits of x.
- The random bits required to choose h from $\mathcal{H}$ can be read only once.

Sketch. Define $\mathcal{H}$ as the set of functions $h_{b_1,\ldots,b_k,\beta}(x)$ for all $b_i \in \{0,1\}^{p(n)}$, $\beta \in \{0,1\}^k$, where the result of each $h_{b_1,\ldots,b_k,\beta}$ is defined as a concatenation of k bits, each computed by scalar product $< b_i|x >$ for $x \neq 0$ or the $h_{b_1,\ldots,b_k,\beta}(x) = \beta$ for $x = 0$. Note that we can choose our function $h \in \mathcal{H}$ just by reading the strings $b_1, b_2, \ldots$ and compute each dot product in a streaming fashion, thus we do not need to store any bit from the b_i's. □

Theorem 4.3. $\Gamma\mathrm{NI} \in \mathrm{AM}_{\mathrm{SC}_2}$

Sketch. This is done by following the classical lower bound protocol.

- Arthur chooses y from a suitable range.
- Merlin sends a message containing $x = ((s_1, \ldots, s_k), \pi, b)$, where π is a permutation of $\{1, \ldots k\}$ and b is a bit representing which group to sample from.
- Arthur first verifies that x is valid by checking that

$$M(C_{\Gamma_b}, (s_i)_{i=1}^k) = M(C_{\Gamma_b}, (\pi(s_i))_{i=1}^k),$$

  where M is the machine from Theorem 3.3. Then he checks that $h\left((M(C_{\Gamma_b}, (s_i)_{i=1}^k), \pi)\right) = y$ following Proposition 4.2.

The protocol can be repeated to give an exponentially small probability of failure. □

References

[1] Allender, Eric and Chen, Shiteng and Lou, Tiancheng and Papakonstantinou, Periklis A. and Tang, Bangsheng *Width-Parametrized SAT: Time–Space Tradeoffs*, Theory of Computing, 10(12):297–339, 2014.

[2] Arora, Sanjeev and Barak, Boaz *Computational Complexity: A Modern Approach*, Cambridge University Press, New York, NY, USA, 1st edition, 2009.

[3] Babai, Laszlo *Trading Group Theory for Randomness*, Proceedings of the Seventeenth Annual ACM Symposium on Theory of Computing, STOC '85, pages 421–429, New York, NY, USA, 1985. ACM.

[4] Babai, Laszlo *E-mail and the unexpected power of interaction*, Structure in Complexity Theory Conference, 1990.

[5] Babai, Laszlo *Graph Isomorphism in Quasipolynomial Time*, http://arXiv:1512.03547, 2015.

[6] Boppana, Ravi B. and Hastad, Johan and Zachos, Stathis *Does co-NP have short interactive proofs?*, Information Processing Letters, 25(2):127–132, 1987.

[7] Goldwasser, S. and Sipser, M. *Private Coins Versus Public Coins in Interactive Proof Systems*, Proceedings of the Eighteenth Annual ACM Symposium on Theory of Computing, STOC '86, pages 59–68, New York, NY, USA, 1986. ACM.

[8] Miller, Gary L. *Graph isomorphism, general remarks*, Journal of Computer and System Sciences, 18(2): 128–142, 1979.

[9] Papakonstantinou, Periklis A. *A note on width-parameterized SAT: An exact machine-model characterization*, Information Processing Letters, 110(1):8–12, 2009.

Growth of Meandric Numbers

Transcribed by Yulia Alexandr, Kayla Cummings and Edgar Jaramillo Rodriguez from a lecture of Martin Klazar

1 Introduction and Early Results

This talk concerns the enumeration of meanders with $2n$ crossings. We begin with some relevant definitions.

Definition 1.1. A *meander* is a closed, simple curve in the plane crossing a line l at $2n$ points.

A meander M with $2n$ vertices can be thought of in terms of its upper and lower halves, where each half is a set of n non-intersecting edges connecting the $2n$ crossing points. We see that this formulation is similar to the definition of a non-crossing matching.

Definition 1.2. A *matching* on $[2n] = \{1, 2, \ldots, 2n\}$ is a 1-regular ordered graph $G = ([2n], E)$ such that each edge is vertex-disjoint. A matching N on $2n$ vertices is *non-crossing* if $N = ([2n], E)$ such that $1 \leq a < b < c < d \leq 2n$ implies $\{a, c\} \wedge \{b, d\} \notin E$. (see Figure III.1, below)

Of course we must be careful as not every pair of matchings in the upper and lower halves will produce a meander. For example, making the upper and lower halves both equal to the first example in Figure III.1 would produce two closed curves. However, we can use our knowledge of meanders to develop an upper bound for the number of meanders with $2n$ crossings. For this we must make use of the following lemma.

Lemma 1.3. The number of non-crossing matchings on $2n$ vertices is equal to the n^{th} Catalan number c_n.

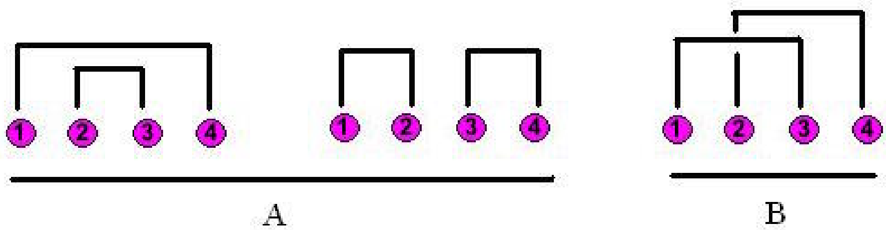


Figure III.1: (A) examples of a non-crossing matching on [4], (B) example of a crossing matching on [4]

Proof. It is a well known result that the n^{th} Catalan number c_n encodes the number of Dyck words of length $2n$. A Dyck word is a string consisting of n X's and n Y's such that no initial segment of the string has more Y's than X's. For example, the string: $XXYYXY$ is a Dyck word of length 6. Now let f be a function from the set of non-crossing matchings on $[2n]$ to the set of Dyck words of length $2n$. We define f such that for each for each edge $\{a, b\}$ in our matching (with $a < b$) we label a as X and label b as Y. This produces a string of X's and Y's of length $2n$. This string is clearly a Dyck word because the X's and Y's denote the left and right endpoints of each edge and it is impossible to see more right endpoints than left endpoints when reading the labels from left to right. It is left as an exercise to show that f is a bijection, which implies that the size of the two sets are equal as desired. $\square$

From this we see that the number of meanders on $2n$ vertices, m_n, is at most c_n^2, so

$$m_n \leq c_n^2 \leq (4^n)^2 = 16^n.$$

However, by exhaustive calculation the first few meandric numbers are $\{1, 2, 8, 47, 262, 1828, 13820, \dots\}$, so we see that our bound is not tight. The rest of this talk introduces one attempt to tighten this bound in the limit.

2 The Meandric Constant

Observe that an appropriate lower bound for m_n is simply the number of non-crossing matchings (number of "upper halves") which grows similarly as 4^n in the limit. Pairing this with our upper bound of 16^n, motivates us to believe that the growth of the meandric numbers should be of the form

μ^n where μ is some number between 4 and 16. This leads to the following definition.

Definition 2.1. Let the *meandric constant* μ be defined as follows.

$$\mu := \lim_{n\to\infty} m_n^{1/n}$$

To prove that μ exists, we will make use of supermultiplicative form of Fekete's Lemma, here provided without proof.

Lemma 2.2 (Fekete's Lemma, supermultiplicative form)**.** Let m_n be a sequence of real numbers such that $m_n \geq 1$ for all n. If $m_a m_b \leq m_{a+b}$ for all a, b, then

$$\lim_{n\to\infty} m_n^{1/n} = sup\{m_i^{1/i} : i \in \mathbb{N}\}.$$

We are now equipped to prove the existence of μ.

Proof. Let M_k denote the set of meanders with $2k$ crossings. Consider the map $f : M_a \times M_b \to M_{a+b}$ defined as follows. For M_1, M_2 meanders in M_a and M_b, respectively, first reindex M_2 by shifting each index by a. You now have two adjacent and disjoint meanders. Leave the bottom non-crossing matchings of each meander alone. From the upper non-crossing matching select the two adjacent ("middle") edges and connect them, then add another edge to the upper half connecting the two remaining crossing points (See Figure III.2). This map is clearly injective so it must be that $|M_a \times M_b| \leq |M_{a+b}|$. It follows from Fekete's Lemma and the upper bound developed earlier that μ exists. □

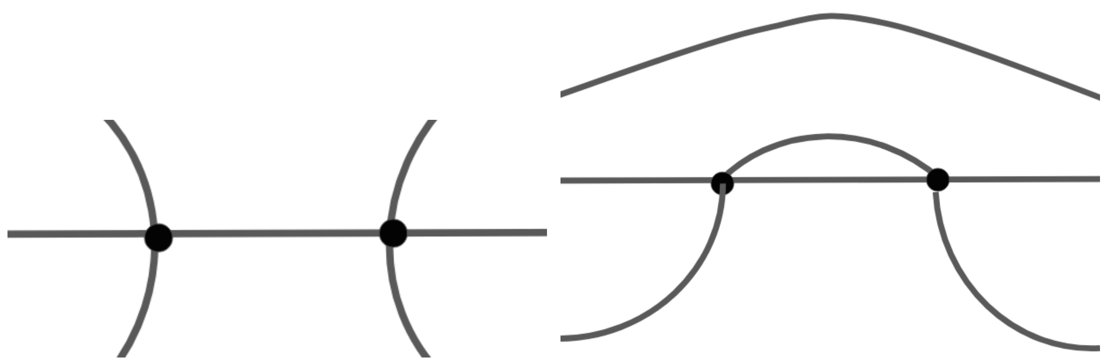


Figure III.2: "Rewire" adjacent edges

We have thus established the existence of μ. However, as of now this constant still has not been computed precisely. In fact, it is not even known whether the constant is computable. As of now, it has been shown that $11.38 \le \mu \le 12.901$. The problem of whether we can improve these bounds is still open.

References

[1] Michel Goossens, Frank Mittelbach, and Alexander Samarin. *The LaTeX Companion*. Addison-Wesley, Reading, Massachusetts, 1993.

[2] Knuth: Computers and Typesetting, `http://www-cs-faculty.stanford.edu/~uno/abcde.html`

Ramsey Number for Binary Matrices

Jaroslav Hančl Jr., Jana Novotná, Jakub Pekárek

1 Introduction

A *Ramsey number* $R(G)$ of a graph G is a well studied graph parameter describing the complexity of a graph. More recently, ordered Ramsey number [1] and [3] for ordered graphs is of a particular interest. For any ordered graph $\Gamma = (\Gamma, \prec)$ an *ordered Ramsey number* $\overline{R}(\Gamma)$ is a minimal n such that every ordered complete graph with n vertices and with edges colored by two colors contains a monochromatic copy of Γ. Clearly $R(G) \leq \overline{R}(\Gamma)$ for any ordering Γ of a graph G.

In this paper we study a similar structure for tables. Let T and T' be two tables with black and white cells. We say that T *contains* T', or T' *is a minor of* T, if T' can be obtained from T by deleting some rows and columns.

Let M be a $k \times k$ matrix and T be an $n \times n$ table. We say that T *covers* M if T contains a $k \times k$ subtable T' such that all ones in M are monochromatic in T'. A *Ramsey number* $R(M)$ of a matrix M is the smallest n such that any $n \times n$ table covers M. Throughout the paper we use the notation of Ramsey number only for matrices until specified otherwise.

Since any graph G can be ordered and characterised by its adjacency matrix M_G we have $R(G) \leq R(M_G)$. On the other hand, probabilistic argument shows that matrix J consisting only of ones has Ramsey number exponential in a size of J.

Permutation matrices is of special interest. Given permutation π on k elements, a *permutation matrix* M_π is a square binary matrix whose entry (i, j) is 1 whenever $\pi(i) = j$ and 0 otherwise. Balko, Jelínek and Valtr [2] remarked that $R(M_\pi) \leq k^2$ for any permutation matrix. On the other hand, they showed that almost any permutation asymptotically satisfy $R(M_\pi) \geq O(k^2/\log^2 k)$.

We present the bounds on a Ramsey number for a special classes, resp. particular matrices. In the second section we discuss methods using Dirichlet principle and derive bounds for lines, diagonals and L-shape matrices. Third section is devoted to the block scheme which gives us bounds to more general shapes. Section 4 deals with concatenation of permutation matrices and matrices that are formed via recursion. Finally, the last two sections shows tight bounds for small matrices, almost-diagonal matries and conjectures a connection to the shortest permutations paths.

Notation

For our purposes we use I_k the identity matrix of size k and J_k the $k \times k$ matrix with all entries 1. By diagonal (of length l) we mean any elements of a matrix, resp. table, in the form $(a+i, b+i)$, where a, b are positive integers and $i \in \{1, 2, \ldots, l\}$. In a notion of antidiagonal we just reverse one coordinate; that means its elements are in the form $(a+i, b-i)$.

Let $L_{k,t}$ denote horizontal line of width t and length k. That is a matrix of ones and dimensions $t \times k$.

2 Dirichlet's principle

Dirichlet's princeple tells us that in $2k-1$ elements of two colors, there is one dominating color shared among at least k elements.

2.1 Lines and diagonals

Consider diagonal matrix I_k of length k. Using the Dirichlet's principle, it is trivial to show that $R(I_k) = 2k-1$. Consider the diagonal of a table of size $2k-1$ there are at least k cells of one of the colors, giving a I_k minor. On the other hand, if we take a table of size $2k$, we may split it into two halves (either diagonally, horizontally or vertically) and color each half with a different color. Such table clearly does not contain I_k minor of either color. Essentially the same arguments apply when instead of I_k we consider L_k.

Corollary 2.1. $R(I_k) = R(L_k) = 2k-1$.

2.2 Thick lines and diagonals

The previous idea can be expanded to lines and diagonals of width greater than one.

To construct an upped bound for $R(L_{k,t})$ for a thick line, we use iterated Dirichlet's principle. Consider the first row of our table, there is one dominating color. Let us delete all columns where the first line has a cell of the non-dominating color. We obtained a subtable of at least half the width of the original table with monochromatic first row. We repeat this process on lower rows to obtain $2t-1$ consecutive monochromatic rows, each time obtaining a subtable of at least half width compared to the previous step. Finally, out of these $2t-1$ monochromatic rows at least t of them are of the same color, forming a $L_{k,t}$ minor.

From the construction we may see, that the width of the original table must be (roughly) at least $k * 2^{2t-1}$. Thus a constant-width line has a linear Ramsey number. While the constants are certainly not tight, the bound cannot be asymptotically improved in general, as for a line of width 1 it corresponds to the previous bound up to $O(1)$ additive error, and for linear-width line we have an exponential upper bound, which is complemented by an exponential lower bound which can be shown through standard probabilistic method argument.

Corollary 2.2. $R(L_{k,t}) \leq k \cdot 2^{2t-1}$ while also $R(L_{k,t}) \in \Omega(k \cdot 2^t)$

Consider the same idea for diagonals, let $I_{k,t}$ denote diagonal of width t and length k. We consider this diagonal to be composed of the t longest diagonals in a matrix $k \times k$, where for t even we choose either of the two possibilities. By the same argument as above, we may obtain an arbitrary amount of monochromatic parallel diagonals. The last step is however more complicated.

Suppose case for $I_{k,2}$. From the first part of the construction we have three monochromatic diagonals out of which two are of a dominant color. If the two diagonals are neighbouring, we get the minor trivially. If however the middle diagonal has a non-dominant color, we have to delete some rows and columns to bring two diagonals together. When they are together, each two consecutive elements of the top diagonal give a row and a column which intersect in a cell from the bottom diagonal. As the elements from the bottom diagonal are at distance two, we need to delete (roughly) every other row and column. Similarly, if the two diagonals we need to bringing together had j diagonals of the opposite color in between them, we would need to keep only every $(j+1)$-th row and column (deleting all the other).

In general case of $I_{k,t}$ we can assume that there are at most $t-1$ diagonals of the non-dominant color forming gaps (in some non-specific way) in between the diagonals of the dominant color. Omitting the details we may say that each gap composed of j non-dominantly colored diagonals reduces the number of rows and columns of our table to roughly $1/(j+1)$. The total decrease of size is the product of the reductions over all the gaps. As the sizes of the gaps sum up to at most $t-1$, it is easy to see that the maximal reduction occurs when all the gaps are of size exactly 1. Although in such a specific configuration we may design a more sophisticated approach, we get an general upper bound on the reduction is at most $1/2^{t-1}$.

Counting the number of rows and columns necessary for the final sub-table to have size at least k, we conclude that this construction requires $k \cdot 2^{2t-1} \cdot 2^{t-1} = k \cdot 2^{3t-2}$. As with the previous upper bound for $L_{k,t}$, while the bounds are not tight in terms of constants, they cannot be improved asymptotically in general case.

Corollary 2.3. $R(I_{k,t}) \leq k \cdot 2^{3t-2}$ while also $R(I_{k,t}) \in \Omega(k \cdot 2^t)$

2.3 Rows and columns with dominant colors: "L"-matrix

Sometimes we use the Dirichlet's principle to divide rows and columns into two groups depending on the dominant color in some small subsection. To illustrate this approach, together with a few other ideas, we present a specific upper bound. In this section, let L_k denote an L-shaped $k \times k$ matrix, in other words a matrix with the left-most column and the bottom row filled with ones. We will show that $R(\mathsf{L}_k) \leq 5k-5$.

Let us consider a table T colored by black and white colors of size $5k-4$. Let us assume for contradiction that it has no L minor. We cut the table once in each dimension into parts of size $3k-1$ and $2k-3$, resulting in four rectangles A, T, R, Z, where A is the left-bottom rectangle and has dimensions $3k-2 \times 3k-2$, Z is the right-top rectangle of dimensions $2k-3 \times 2k-3$, T is the rectangle on top of A and R is to rectangle to the right of A. We will not consider the contents of Z.

For every column of T there exists a dominant color with at least $k-1$ elements. The same applies for every row of R. Every cell in A corresponds to an intersection of one column from T and one row from R and vice versa. Whenever we consider one row from R and one column from T of the same dominant color, then the associated cell in A must have the opposite color, otherwise we would get an L minor. Therefore, whenever there are at least k rows of R and at least k columns of T with the same dominating color, we get a monochromatic subtable of A of size at least $k \times k$ in the opposite color. As there are $3k-1$ rows and columns in R and T, we conclude that R has at least $2k-1$ rows of one dominant color, and T has at least $2k-1$ columns of the opposite color. Let B be the subtable of A corresponding to intersections of these rows and columns.

The subtable B has size at least $2k-1 \times 2k-1$, above each row of B there are at least $k-1$ elements of some color a and to the right of every row of B there are at least $k-1$ elements of another color b. The subtable B itself must have a dominant color, without loss of generality, let it be the color b, the case for a is symmetric. Since b is dominant, then there is at least one column with dominant color b, and therefore at least k cells in color b. The left-most of them has at least $k-1$ cells of color b above it and so forms the bottom-left corner of an L minor.

On the other hand, consider a different table T of size $3k-3 \times 3k-3$ defined as following. We color each anti-diagonal (left-bottom to top-right) in one color. On the left edge of the matrix, the first $2k-3$ cells are white and the remaining k cells are black. On the bottom, the first $k-1$ cells are black and the remaining $2k-2$ cells are white. Thus the table consists of top and bottom white triangles and a black band in between. We observe that L_k minor has ones on all $2k-1$ of its anti-diagonals. There are not enough black diagonals. At the same time, both white triangles consist of only $2k-2$ white diagonals. Therefore, an L_k minor has to be white and take white cells from both triangles. We observe that the left-most white cell in the bottom triangle has only $k-2$ white cells above it and the same is true for any other cell to the right that lies under the top triangle. Symmetrical observation applies to the bottom white cell of the top triangle. We conclude that T contains no L_k minor.

Corollary 2.4. $3k-3 \leq R(\mathsf{L}_k) \leq 5k-5$.

3 Block scheme

Let $R(A, B)$ denote the minimum number n such that for each table of size $n \times n$ colored in two colors, there is either a minor A in the first color or a minor B in the other color. For the purposes of this section, let us call the first color black, and the other color white.

The base idea of this section is that every block of size $k \times k$ is either purely white or contains at least one black cell.

3.1 Permutations in $O(k^2)$

Fact 3.1. Let M be a permutation matrix $k \times k$. Then $R(M) \leq k^2 - O(k)$

This fact follows from a simple construction. Let us have a table of size $k^2 \times k^2$ and split it into blocks of size $k \times k$. Each block is either monochromatic, in which case it contains a monochromatic minor M, or contains at least one element of each color. Since in M there are no two elements in the same column or row, we now take the blocks corresponding to each one in M and delete all rows and columns incident with this block except one containing a cell of the desired color. This way we have constructed the minor M.

This simple idea allows minor improvements, for instance one of the blocks can be only of size 1×1 as we can switch the meaning of the colors so that is it the desired color without deleting any rows or columns. This reduces the total size of the table to $k^2 - k + 1$.

Note that this construction can be interpreted as an upper bound for $R(M, J_k)$.

3.2 Few ones off diagonal

Let M be a permutation matrix M with only a fixed amount of ones placed off the diagonal. Let j denote the number of ones off the diagonal.

First we use the Dirichlet's scheme to find a subtable with a monochromatic diagonal of length $k+2jk-j$. We can do this in every table of size at least $2k+4jk-2j-1$. To find the M minor in this subtable, we associate each element of M that is not on the diagonal with a $k \times k$ block. Each $k \times k$ block contains either a monochromatic J_k, and therefore a monochromatic M minor, or at least one element of each color. By deleting all rows and columns incident with this block except one with the cell of appropriate color. This shortened the diagonal by $2(k-1)$ element, while we found one element off the diagonal whose position (in respect to the remaining subtable) depends solely on our choice of the location of the block.

If we do this for all the j element, each time we shorten the diagonal by $2(k-1)$ element, which leaves us with a diagonal of length exactly k and all of the j elements. It is clear that given a long enough diagonal we may place all of the blocks appropriately in a non-incident manner, although detailed description is a bit technical as each block removes two pars of the diagonal simultaneously. Note that since M is permutation, we never choose any row or column into two blocks and so we never delete previously found elements.

Corollary 3.2. Let M be a permutation matrix $k \times k$ with at most j elements off its diagonal, or a union of I_k and at most j more one-elements with at most one in each row and column. Then $R(M) \leq 2k + 4jk - 2j - 1$.

Note that if there are asymptotically many one-elements placed only a small constant off the diagonal, we may use a Dirichlet's scheme to find a thick diagonal in the beginning and then find all the remaining elements with higher distance.

3.3 Lines of fixed color

Suppose we want to find a (horizontal) line of length l in a given color, or a monochromatic square matrix of size $m \times m$. In other words, we are looking for an upper bound to $R(L_l, J_m)$.

Let us take a large-enough table and consider only the first m columns. In this section of the table, at most $m - 1$ rows can be all-white, otherwise we have m rows of width m, forming a white J_m minor. We delete these at most $m - 1$ white rows. In the resulting submatrix, each row has at least one black cell in the first m columns. We iterate this process to each m-tuple of columns, each time increasing the minimal number of black cells in each row of the whole table and deleting at most $m - 1$ rows each time.

In order to reach an L_l black minor, we need to iterate l times and ensure that at least one row is remaining at the end of the process. The total width of the area we use is clearly exactly $l \cdot m$. The total number of deleted rows is at most $l \cdot (m - 1)$, so if we start with at least $l \cdot (m - 1) + 1$ rows, we must find one of the two minors in the prescribed color.

Corollary 3.3. Every rectangle table of size at least $l \cdot (m - 1) + 1 \times lm$ colored in black and white colors contains either l black cells in one row or a white J_m minor.

Since we expect every m columns to produce only one black cell in the worst case, we may generalize this construction. In particular, we may split the line of length l into an arbitrary number of parts with each gap of arbitrary width. We do this by simply adding the correct amount of gap-columns in between any two m-tuples of columns from the construction.

The construction is oblivious to the existence and positions of these extra columns and will conserve them while constructing the line.

Corollary 3.4. Let T be any table of size at least $(l\cdot(m-1)+1)\times(lm+g)$ colored in black and white colors with a given decomposition of the table into l m-tuples of columns and additional g columns in any order. Then T either has a white J_j minor or a row minor such that it contains exactly one black cell from every m-tuple and one cell from each of the additional g columns.

3.4 Matrices with column-wise limited ones

Claim 3.5. Let M be a matrix with at most c ones in each column. Then $R(M, J_j) \leq k \cdot j^c$.

The idea of this construction is based on the previous construction of lines of fixed color. We are looking for either a black M minor or a white J_j minor. First, we split the table into k vertical bands of the same width. Each band corresponds to a column of M. We now go through the rows of M one by one and process all the band of the table in parallel.

The high-level idea is that in each band we eventually find the corresponding column of M as a minor. However, in order to construct the whole M minor, these individual column minors have to use the same rows of the table. In each step we take one row of M and find an appropriate row in the table, such that it has a long black horizontal line in every band where M has a one. In doing so we delete all the other columns from these particular bands. This ensures that all the remaining columns have a black cell on the desired row. To find each line, we consider a group of the top non-processed rows (rows not previously used) of the table, all of which except one get deleted during the step.

We define a property called *width* for every one in M as j^i where i counts the number of ones under this one in the same column. So each bottom one has width 1 and each top one has width at most j^{c-1}. The width of each one corresponds to the width of the line we want to find in the corresponding band when processing this element in our construction. The bottom ones having width 1 means exactly that we only need one column left from every band in the end.

Note that we start with all bands of width j^c. From the Corollary 3.3 we know that in order to find a line of length l, it suffices to have an area of width lj. Thus each band is wide enough to find a black line of the width of the first one in the corresponding column, which is in turn long enough to allow us to find the next line and so on until we go through all the levels. So far we do not consider the height of the table needed.

However, for every row of M, we need to find all the lines for all the one elements simultaneously. This is where the Corollary 3.4 shows that we can ignore all the bands where we are not looking for a line and find all the lines simultaneously in all of the relevant bands, reducing the width of each band to exactly a fraction of $1/j$. In other words, the price of the higher complexity of the pattern is that we delete more rows, but we delete the same amount of columns from each band as if we were trying to find the lines individually.

It is clear that the width of the bands is sufficient. Let us analyze the number of rows used. According to Corollary 3.4 in each step we need to use block of $l \cdot (j-1)+1$ rows, where l denotes the sum of lengths of the lines we are looking for. All of these lines except one are deleted in the step. While the value of l varies greatly for each row of M, we can sum up the requirements over the whole matrix M (processed in k steps) as $L \cdot (j-1)+k$ where L denotes the sum of all lines we need to find during the whole construction. Thus L corresponds to the sum of widths of all ones in M, which can be estimated by sum over columns of M as $L \leq k \cdot (1+j+j^2+...+j^{c-1})$. Plugging this bound into the previous expression we obtain the bound on the number of used rows kj^c.

3.5 Unions of permutations

Let $M_1, M_2, ..., M_c$ be permutation matrices $k \times k$. We define the union of $M_1, ..., M_c$ as a matrix M of dimensions $k \times k$ so that each position of M is a one if and only if one of the permutation matrices has a one at the same position. From the previous section we have an immediate corollary.

Corollary 3.6. Let M be a union of c permutation matrices. Then $R(M) \leq k^{c+1}$.

3.6 Multiple colors

Let us consider a more general task when the table is colored using χ colors. We can use the previous construction to bound the Ramsey number even in this setting. The idea is that for any matrix M with at most c ones per column we use the result 3.5 in the following way. We pick one particular color and look for either M minor in this particular color or a J_w minor composed of only the remaining colors. We choose the constant w so that if we find the J_w minor, we may iterate the same process within the resulting subtable with a number of colors decreased by one.

Let $R_\chi(A, B)$ denote the minimum size of a table colored by χ colors needed to guarantee an existence of either a monochromatic A minor in the first color or a B minor in the union of all the other colors. Let $R_\chi(M)$ denote the minimum size of a table colored by χ colors needed to guarantee an existence of a monochromatic M minor in any color.

Let M be a matrix $k \times k$ with at most c ones per column. From the idea presented above we have the following simple relations.

$R_2(M) \leq k^{c+1}$

$R_3(M) \leq R_3(M, J_{R2(M)}) \leq k \cdot R_2(M)^c \leq k \cdot k^{c(c+1)} = k^{c^2+c+1}$

...

$R\chi(M) \leq R_\chi(M, J_{R_{\chi-1}(M)}) \leq k \cdot R_{(\chi-1)}(M)^c \leq= k^{c^{\chi-1}+c^{\chi-2}+\ldots+c+1} = k^{(c^\chi-1)/(c-1)}$

Corollary 3.7. Let M be a matrix with at most c ones in each column. Then $R_\chi(M) \leq k^{(c^\chi-1)/(c-1)}$.

Our multi-color construction reduces the number of colors by one at a time by dividing the colors into two groups one of which has only one element. It would seem that there could exist a similar idea where the groups of colors would be roughly the same size resulting in a recursion of only a logarithmic depth. This could intuitively reduce the bound (doubly-exponential in χ) to a bound simply exponential in χ. We remark that such result cannot be acquired in such a way. By division of colors into two groups (essentially treating them as two colors), we want to find either a J_w minor in one set of colors or the same J_w minor in the other set of colors. The symmetry is a crucial difference, as $R(J_w)$ is exponential in w for two colors. This would cause the bound to have a form of a tower function (of logarithmic height) rather than a simple exponential function.

4 Operations

Union of intersections is in block schemes.

4.1 Concatenation

A concatenation of two matrices M_1 and M_2 is a matrix

$$M = \begin{pmatrix} M_1 & 0 \\ 0 & M_2 \end{pmatrix}$$

We denote the oncatenation M as $M_1|M_2$. Recall definition of $R(A, B)$ as the minimum size of table necessary to always find A minor in the first color of B minor in the other.

Let $M_1, M_2, ..., M_m$ denote square matrices, $k_1, k_2, ..., k_m$ their sizes and $s_i = k_1 + k_2 + ...k_i$ the prefix sums of k_i.

Let us first consider the case when $M = M_1 = M_2 = ... = M_m$, then $R(M_1|...|M_m) \leq (2m-1)R(M)$. We simply split the table into blocks of size $R(M_1)$ each of which has to contain M_1 minor in one or the other color. Considering only the blocks on the diagonal and using Dirichlet's principle we get at least m monochromatic copies of M in concatenation.

In the general setting we consider the following construction. We find all the $M_1|...|M_m$ minor incrementally, finding the individual $M_1, ...M_m$ minors one by one from the top-left corner of the table. First we use block of the table of size $R(M_1)$ to find a monochromatic M_1 minor. If we consider only the columns and rows to to left and down from the used block, we again take a block the top-left corner of this area to find M_2 minor. The problem now is that using a block of size $R(M_2)$ might give us a M_2 minor in the opposite color than the previous M_1 minor. Instead we use a block of size $R(M_2, J_{s_2})$, thus we either find the M_2 minor in the correct color, or a big monochromatic subtable big enough to contain the whole $M_1|M_2$. We iterate this approach and in each iteration we either extend the previous partial minor or find a completely new extended partial minor in the opposite color. We obtain the following relations:

$$S(M_1|...|M_m) \leq S(M_1) + S(M_2, J_{s_s}) + ... + S(M_m, J_{s_m}) \leq \Sigma_{i=1}^{m} S(M_i, J_{s_i})$$

In case when $M_1, ..., M_m$ permutations, we have $R(M_i, J_{s_i}) \leq k_1 \cdot s_i$ from 3.5. The previous relation can then be simplified as follows:

$$S(M_1|...|M_m) \leq \Sigma_{i=1}^{m} k_i \cdot s_i = \Sigma_{i,j=1}^{m} k_i \cdot k_j$$

Further supposing that $k = k_1 = k_2 = ... = k_m$ (thus M_is are permutation matrices of the same size) we get:

$$S(M_1|...|M_m) \leq \Sigma_{i,j=1}^{m} k_i \cdot k_j \leq km(m+1)/2$$

These results can of course be generalized to matrices with a limited number of ones in colums or rows (independently) using the general statement of 3.5.

4.2 Recursion

Let π and σ be two permutations. We define their matrix composition $\pi \circ \sigma$ by its permutation matrix M in the following way. We start with a permutation matrix M_π, replace any position with one by the matrix M_σ and any position with zero by the appropriate zero matrix. Now we observe the following lemma

Lemma 4.1. Let π and σ be two permutations then

$$R(M_{\pi\circ\sigma}) \leq R(M_\pi)R(M_\sigma).$$

Proof. First we divide the table in the blocks of size $R(M_\sigma)$. Note that the number of such a blocks is $R(M_\pi)^2$. In each block there can be found a monochromatic copy of matrix M_σ as a submatrix, hence we mark those blocks as white or black (if both, we choose one color arbitrary). In this way we obtain the $R(M_\pi) \times R(M_\pi)$ block matrix. Finally, this block matrix contains a monochromatic copy of a matrix M_π, thus we have found a monochromatic copy of $M_{\pi\circ\sigma}$. $\square$

Particulary, consider M_π to be a general permutation matrix of size c and $M_\sigma = I_{k/c}$. Since block schemes implies $R(M_\pi) \leq c^2 - c + 1$ and $R(I_{k/c}) = 2k/c - 1$ then

$$R(M) \leq 2kc - c^2 - 2k + c + 2k/c - 1. \quad (3)$$

Example 4.2. Let M be a permutation matrix created by recursion from I_2 and an anti-diagonal matrix of size $k/2$ (we suppose k is even). Then $\frac{5}{2}k - 3 \leq R(M) \leq 3k - 3$. The upper-bound we obtain from the inequality (3) above and the lower-bound we obtain from the construction on Figure IV.1.

This process can be generalized. Let $i \in [t]$ and π_i be a permutation on set $\{1, 2, \ldots, v_i\}$ with permutation matrix M_i. Set $\pi = \pi_1 \circ \pi_2 \circ \cdots \circ \pi_t$ to be their matrix composition then

$$R(M_\pi) \leq R(M_{\pi_1}) \cdot R(M_{\pi_1}) \cdot \ldots \cdot R(M_{\pi_t}).$$

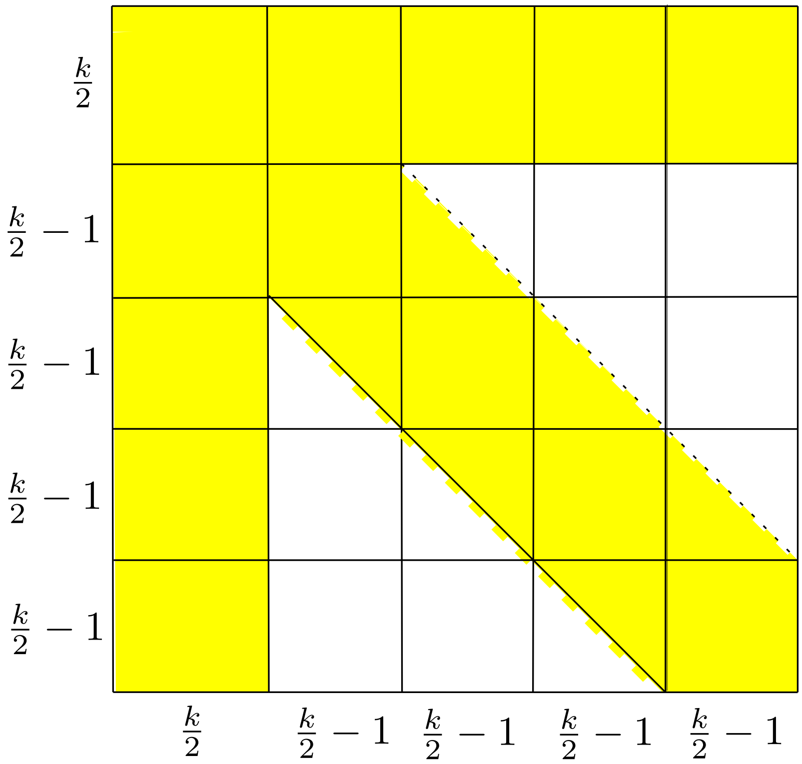


Figure IV.1: Lower bound

5 Bounds of specific matrices

5.1 Almost-diagonal precisely

Let A_k be a permutation matrix of a permutation $(1, 2, 3, \ldots, k-2, k, k-1)$, where only last two elements are switched. Although this matrix is highly similar to the identity I_k, its Ramsey number differs by a square root factor.

Claim 5.1. $2k + \sqrt{2k} - 3 < R(A_k) \leq 2k + \sqrt{2k}$

Proof. First we show the upper bound. Let T be a $(2k+\sqrt{2k}) \times (2k+\sqrt{2k})$ table and let s be the smallest number such that there is $k-2$ elements of the same color, say black, in the first $k-2+s$ diagonal elements. Then the last 3×3 table must contain an antidiagonal of length 2 in the white color.

First suppose that $s = 0$. Let $t = \sqrt{2k} + 2$. We focus on the number of black cells in the latter $(k+t) \times (k+t)$ table V. Any diagonal of V that starts on the element with coordinates $(1, j)$, resp. $(j, 1)$, should have at least $t - (j-1)$ black cells. Summing over all starting cells we obtain at least t^2 black cells. However, all black cells of V should avoid antidiagonal of length two, hence there should be maximum of $2(k+t) - 1$ black cells. Comparing those two lines we obtain $t^2 \leq 2(k+t) - 1$ which is not true.

Now assume that $s \neq 0$. The bound on the number of black cells are now $t^2 \leq 2(k+t-s)-1$, which is worst then the previous case.

Let $n = 2k + \sqrt{2k} - 3$. For the lower bound we construct an $n \times n$ table T' which avoids A_k in the following way. First $k-2$ rows and columns of T' are black. In the residual square table V' we color all cells white but the east-south path P from the upper leftmost cell. The path consists of first $\sqrt{2k}$ in the first row, then continues $2\sqrt{2k}-2$ cells down, then $2\sqrt{2k}-4$ cells right, $2\sqrt{2k}-6$ cells down, etc. So there is

$$\sqrt{2k} + \sum_{j=1}^{\sqrt{2k}} 2\sqrt{2k} - 2j = 2k$$

black sells meaning that P ends somewhere inside V'.

Matrix A_k is not covered by black colored cells of T' because the path P avoids antidiagonal of length 2. And finally we show that A_k is not covered by white colored cells of T'. For contrary we suppose that it is covered. Suppose that the sequence of chosen columns of this cover is lexicographically the smallest one. Then we did not choose the first row of V' because its first white cell is at distance less than k from the right boundary. So we choose the cell $(2,1)$ for the first one of A_k and continue choosing white cells on this diagonal till we hit path P again. We cannot choose the column of the path because it would mean we are too low (we would have not chosen $\sqrt{2k}$ rows of V'). So we skip this column of the path P and continue with the white cell $(\sqrt{2k}+1, \sqrt{2k}+1)$ on the main diagonal of V'. We continue hitting the path P and do not choose its rows and columns (justifying that there would not be enough space for the rest of A_k). In the end, we have crossed path P $2\sqrt{2k}$-times, did not choose any of its $\sqrt{2k}$ rows and $\sqrt{2k}$ columns hence we have only $k-1$ choices for rows and $k-1$ choices for columns. Contradiction. $\square$

Remark 4. Upper bound in the previous case can be improved to $2k+\sqrt{2k}-1$ when we reduce the last 3×3 table with a black antidiagonal of length 2 to the size 3×2.

5.2 Small matrices

First let us define small permutation matrices. Any permutation matrix of size $k < 5$ is isomorphic (its permutation matrix is symmetric) to one of those:

$$P_2^1 = \begin{pmatrix} \bullet & \\ & \bullet \end{pmatrix}, P_3^1 = \begin{pmatrix} \bullet & & \\ & \bullet & \\ & & \bullet \end{pmatrix}, P_3^2 = \begin{pmatrix} \bullet & & \\ & & \bullet \\ & \bullet & \end{pmatrix}, P_4^1 = \begin{pmatrix} \bullet & & & \\ & \bullet & & \\ & & \bullet & \\ & & & \bullet \end{pmatrix}$$

$$P_4^2 = \begin{pmatrix} \bullet & & & \\ & \bullet & & \\ & & & \bullet \\ & & \bullet & \end{pmatrix}, P_4^3 = \begin{pmatrix} \bullet & & & \\ & & & \bullet \\ & & \bullet & \\ & \bullet & & \end{pmatrix}, P_4^4 = \begin{pmatrix} \bullet & & & \\ & & \bullet & \\ & \bullet & & \\ & & & \bullet \end{pmatrix}$$

$$P_4^5 = \begin{pmatrix} & \bullet & & \\ \bullet & & & \\ & & & \bullet \\ & & \bullet & \end{pmatrix}, P_4^6 = \begin{pmatrix} \bullet & & & \\ & & & \bullet \\ & \bullet & & \\ & & \bullet & \end{pmatrix}, P_4^7 = \begin{pmatrix} & \bullet & & \\ & & & \bullet \\ \bullet & & & \\ & & \bullet & \end{pmatrix}$$

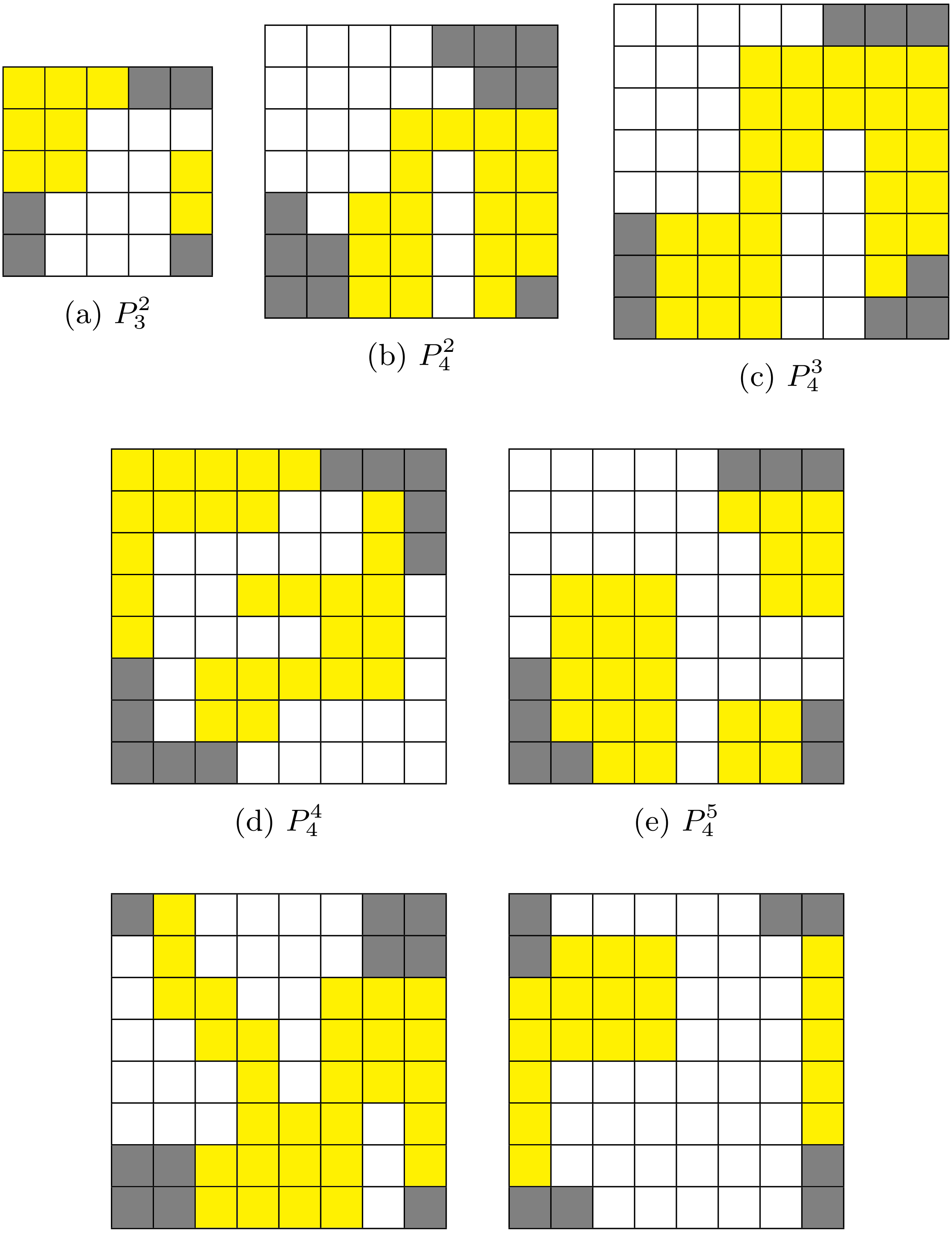
We have found the Ramsey numbers for all permutation matrices of size $k < 5$ except P_4^6 and P_4^7. All diagonal matrices of length k has Ramsey number $2k - 1$ due to 2.1, hence $R(P_1^1) = 1$, $R(P_2^1) = 3$, $R(P_3^1) = 5$ and $R(P_4^1) = 7$.

Lemma 5.2. Ramsey numbers for all small non-diagonal permutation matrices are $R(P_3^2) = 6$, $R(P_4^2) = 8$, $R(P_4^3) = R(P_4^4) = R(P_4^5) = 9$, $R(P_4^6) \in \{9, 10\}$ and $R(P_4^6) \in \{9, 13\}$.

Proof. All lower bounds can be checked in the Table 1. We deal with the upper bounds one by one but usually we use some kind of block argument.

1. $R(P_3^2) \leq 6$: The lowest rightmost 3×3 table contains an antidiagonal of length at least 2 in one color, say black. If there is a black cell in the top leftmost 3×3 table then with the black diagonal it forms P_3^2. If not, there is a block 3×3 of white color and we are done.

2. $R(P_4^2) \leq 8$: The lowest rightmost 6×6 table V covers P_3^2 enforcing the top leftmost 2×2 table to be monochromatic, say black. Then V cannot contain black antidiagonal of length 2. Hence in a division of V into two blocks of size 3, in any of these blocks you can find both white diagonal and white antidiagonal, which enforces (P_4^2 in white color.



(a) P_3^2 (b) P_4^2 (c) P_4^3

(d) P_4^4 (e) P_4^5

(f) P_4^6 (g) P_4^7

Table 1: Lower bounds for small non-diagonal permutation matrices

3. $R(P_4^3) \leq 9$: Same argumentation as in the case of P_3^2. There is a monochromatic antidiagonal of length 3 in the lowest rightmost 5×5 table meaning that the top leftmost 4×4 table either has one element

of this color enforcing P_4^3 or it is monochromatic and the conclusion is trivial.

4. $R(P_4^4) \leq 9$: The lowest rightmost 6×6 table V covers P_3^2 enforcing the top leftmost 3×3 table to be monochromatic, say black. Then there is either a black cell in V or V is white, both cases easily concludes monochromatic P_4^3.

5. $R(P_4^5) \leq 9$: We divide the 3×3 matrix into blocks of size 3. Any of the diagonal blocks have to contain a monochromatic antidiagonal of length 2 and at least two of those colors must be the same enforcing P_4^6.

6. $R(P_4^6) \leq 10$: There is a monochromatic P_3^2 in the lowest rightmost 6×6 table meaning that the top leftmost 4×4 table either has one element of this color enforcing P_4^6 or it is monochromatic and the conclusion is trivial.

7. $R(P_4^7) \leq 13$: It is a simple consequence of block scheme. We pick the color of the cell with coordinates $(1,5)$, say black. Then we consider 4×4 blocks on the "appropriate" positions given by matrix P_4^6 with no common rows and columns. Then there is either one black cell in all these blocks or the whole block is white, both cases enforcing P_4^7.

□

6 Shortest permutation paths

In order to find the Ramsey number for given matrices we have studied the several matrix parameters. Shortest permutation path is the most reasonable one.

Let M_π be a permutation matrix. The shortest permutation path (SPP) is the shortest path in the Manhattan metric which starts in an one element of M_π and visits all the one elements of M_π. We denote its length; that is the number of elements the SPP visits, by $SPP(M_\pi)$.

Trivially $SPP(I_k) = 2k-1$ and for most of small and well behaved permutations, the value of SPP can be computed easily. So there is a wise question on the correlation or connection of SPP and Ramsey number. On one hand, Ramsey number and length of SPP coincides for all permutations of length at most 4 except permutation P_4^7, where we were not able to find the Ramsey number and hence prove the equality.

On the other hand one can easily check that $SPP(A_k) = 2k$ but Claim 5.1 implies $R(A_k) = 2k + \sqrt{2k} + O(1)$. That means we cannot bound Ramsey number by SPP from below. However, that was expected since Balko, Jelínek and Valtr [2] remarked that $R(M_\pi) \geq O(k^2/\log^2 k)$ holds for almost any sufficiently large permutation and one can check that $SPP(M_\pi) = O(k\sqrt{k})$. Indeed, we divide M_π to the blocks of size $\sqrt{k}$ and define a path P which go through all ones such that it visits all ones in a single block and then moves to the consecutive block, visit all ones, move etc. All subpaths of path P between two consecutive ones are either inside one block or move to the adjacent block; in both cases the length is at most $3\sqrt{k}$. Hence the path P has length at most $3k\sqrt{k}$ implying $SPP(M_\pi) \leq 3k\sqrt{k}$.

There might still be a hope for an lower bound of Ramsey number using the value of SPP as there seems to be some connection between the two. We may predict that a matrix with large SPP could have big (superlinear) Ramsey number. We suspect this connection might give rise to constructive lower bounds for Ramsey numbers. As of now, we would be interested in the following $k \times k$ permutation matrix

$$Q = \left(\begin{array}{cccccccccccccccc}
\bullet&&&&&&&&&&&&&&& \\
&&&&\bullet&&&&&&&&&&& \\
&&&&&&&&\bullet&&&&&&& \\
&&&&&&&&&&&&\bullet&&& \\
&\bullet&&&&&&&&&&&&&& \\
&&&&&\bullet&&&&&&&&&& \\
&&&&&&&&&\bullet&&&&&& \\
&&&&&&&&&&&&&\bullet&& \\
&&\bullet&&&&&&&&&&&&& \\
&&&&&&\bullet&&&&&&&&& \\
&&&&&&&&&&\bullet&&&&& \\
&&&&&&&&&&&&&&\bullet& \\
&&&\bullet&&&&&&&&&&&& \\
&&&&&&&\bullet&&&&&&&& \\
&&&&&&&&&&&\bullet&&&& \\
&&&&&&&&&&&&&&&\bullet
\end{array}\right)$$

which is formed by dividing the matrix into a blocks $\sqrt{k} \times \sqrt{k}$ and in each (i, j)-th block there is exactly one one with the coordinates (with respect to the given block) (j, i). We know that $SPP(Q) = (k-1)(\sqrt{k}+1)$ which could imply at least some kind of non-trivial lower bound on $R(Q)$.

References

[1] Balko, M. and Cibulka, J. and Král, K. and Kynčl, J. *Ramsey numberes of ordered graphs.* https://arxiv.org/pdf/1606.05628.pdf, 2015.

[2] Balko, M. and Jelínek, V. and Valtr, P. *On Ordered Ramsey Numbers of Bounded-degree graphs*. https://arxiv.org/pdf/1606.05628.pdf, 2016.

[3] Conlon, D. and Fox, J. and Lee, C. and Sudakov, B. *Ordered Ramsey Numbers*. https://arxiv.org/pdf/1410.5292.pdf, 2016.

REU Log

Jakub Svoboda

Prologue

I was really happy when I heard that I was accepted to REU DIMS program.

Yeah, that was February. My life was pure and unspoiled with real science. Now, I am here sitting on my bed and thinking about future, and past. How I got here and what I will do.

I don't know if I can make it, so I am writing this log to future generation to avoid my mistakes.

Beginning

This week we travelled from cozy little Czech Republic to this place. We were welcomed warmly and with pizza.

But during the night, I realized. We need to publish, otherwise, we are doomed and we must put to the KAM-ITI series any filler we will find. Everyone was really friendly, but I recognized hidden threat behind their eyes. I woke up very early, partly because of jet lag, but also because of the fear of not publishing.

The next day we met Periklis. He was really friendly and presented three problems for us. When I was sitting in this conference room, I was feeling hope. Maybe, just maybe, we are not doomed after all. But then Periklis disappeared and with him, hope.

Then the nightmare started, bureaucracy. It is not their fault, they wouldn't do if they didn't have to. Even papers seemed harmless, but I can read between the lines. And as I read, I started sweating, with ice cold sweat.

From the beginning Radim had mental breakdown. He didn't think that he would publish paper. One day, he started talking nonsenses, something about gender studies. We thought that the next day, he will be better, but last time we have seen him running to the woods and licking Poison Ivy.

Even though we lost Radim we had really productive beginning. We were thinking about some sparse matrices and we had some results. Then we realized that we made a huge mistake, we were discussing the problem in the corridor. There was probably someone who could hear us and plagiarize our work. This feeling was multiplied by the fact, that in the corner there was person with cape taking notes.

Working days

Working days were all similar. We went to the CoRE building and stayed in freezing temperatures until 4 to 5 pm. We studied some books, talked about problems or wrote nonsenses on the board to confuse proof stealers.

During afternoon we relaxed. We did sports and insanity. Sometimes we went to the gym. It was nice. Every time they swiped our ID card in entrance, it said: "Rejected". It worked for five weeks, but after that they said that we cant go to the gym, because we haven't paid. We still don't know how they knew it...

Sometimes, in CoRE was some event. With food. This was our happy day. After the event we sneaked to the room with food and took care of it. We lived from it for another two days.

Once a week, there was lecture. The lecture was given by interesting people, but they wanted to popularize their stuff instead to tell us everything. So it wasn't every time so good.

Weekend trips

During working day we worked, learned and so on. But we came here also for the taste of America. So during weekends, we tried to visit as much we could. We had yet another reason for this, we took all the papers and moved randomly so no one can spy on us.

Appalachian trail

On weekends, Rutgers U is very dangerous place. So we decided to play a little trick on all paper stealers. We left two expendables (yes, Matěj was one of them) in Silvers and we went to Appalachian trail with all proofs and unpublished papers. We hoped that no stealer will suspect that.

We went to New York (dangerous, too much people, too hot, our papers almost self-ignited) and then to Garrison. Our plan was to sleep in a shelter near the rail road. This seemed like a good idea before we had seen that the shelter is actually bower for weddings and for endings of romantic movies my mom watches. But it was dark and we didn't have other place to sleep, so we settled there. The bower was like five meters (3.342210^{-11} AU if you want your crazy units) from Hudson River, so during night it got colder...

After a good night's sleep (if you count lying awake freezing in your sleeping bag) we hit the road. First day went well. We remains of the Fort Montgomery, famous place of American defeat. They tried to impress us, but we weren't impressed at all. In the Czech Republic, almost every hill or valley is famous place of Czech defeat, so for us it was quite boring.

After Fort Montgomery we climbed Bear Mountains (no bear seen). And then we sometimes happy, sometimes exhausted, continued to Fingerboard shelter next to Lake Tioraki (no bear seen). We went to sleep peacefully, only bit afraid of bears (no bear seen). Jardáč was the most frightened, but after I told him that by the bears, they only mean koala bears, he calmed a little bit.

When I woke up next morning our bags were ripped and someone ate our breakfast. Personally, I blame Jardáč, but he said that he had seen a bear eating it (one bear seen).

The rest of the trip was relatively boring. Only Štěpán was hungry and tired, but we were in the hurry, so we left him on some random train station. He will starve to the death soon to stop his suffering.

Boston

After previous weekends we decided that this week, the main group with all our proofs will be truly random.

We went to New York and wandered around. We visited Central park (Hey, New Yorkers reading this, you have very fat squirrels, you should do something about it.), then down town. The views was beautiful, but we wanted to see the Statue of Liberty. We kinda knew where Statue is, but nobody told us that it is on the island. We got to the down town Manhattan and suddenly, sea. That was a problem, but we weren't desperate. Yet.

There was some building with sign "Ferry" on it, so we, as good Czechs, asked about the price of the ticket. We were really excited, that ticked was free, so we boarded immediately.

Yea, we were pretty excited... Maybe too much.

Maybe, we should have asked where is ferry sailing before boarding.

We learned our lesson about ten minutes later, when we were passing by Liberty island.

If you are from New York, you probably know that we were sailing for about thirty minutes to Staten island.

Actually, you probably don't know this, because who in the right mind would go to the Staten island?

We were there, so we decided to be irrational and take a stroll. There was nothing much interesting around.

But we got hungry and returned back, we ate in the best restaurant of its kind in down town Manhattan. It was cheap thought... maybe because the kind of restaurant was McDonald and it was only 3.5 star rating among 3.3 star rating Macs.

Then we visited Brooklyn bridge and part of Brooklyn, because our tickets to Boston were bought for 3am, Sunday. We had some time in Brooklyn, so we decided to be lost for a while. The plan went really well. We made it till 3am only because we were running from Brooklyn bridge.

In 7am we were in morning Boston. It was nice, so we took a nap in a park.

We discovered freedom trail. It was really good. We learned many interesting things about Boston and generally about US history. Too bad for Janča, she refused to visit any "historical" site if it isn't at least 500 years old, yea, she is spoiled European.

MIT and Harvard was also nice but best was McDonald at the bus station. We loved it. It provided us shelter for the night until our bus departed.

The biggest challenge on Monday was not to fall asleep during the lecture. I failed it miserably... But with delight!

Washington D.C.

Another weekend we decided, that we will visit Washington (D.C., poor Janča, she confused that and flew to the west.). We had some good experiences with night rides from New York. So we bought our tickets to 3:30am on Sunday.

I wanted visit New York with Vašek, but I had to do my workout. I tried to do it as fast as possible, but Vašek left during workout and I had to go to the shower.

Running, in the middle of college ave I realized that the shower was pointless. I was wet as I got from the shower.

Starring at the empty New Brunswick platform, I realized that even running was pointless. Train and Vašek were gone. So I had 40 minutes to contemplate about my mistakes and how to find Vašek.

I tried to message him, but he didn't replied. So I decided to do some sightseeing in New York alone. During the day my main hobby was to imagine post apocalyptic New York. How would look the clans in New York if the society collapsed.

I really thought it through. My favourite clan would be Defenders of the High Line. My second favourite would be The Battery Park Battalion.

New York was nice, but truly interesting things happens after midnight. I gave up sightseeing and settled myself near Madison Square Garden and read.

While I was reading first chapter, some woman was changing clothes... mildly said. "OK, this is different culture, probably it is fine." And I continued reading.

While I was reading second chapter, some weird guy appeared. He didn't bother me, but went to other guys seated in front of me. He looked like a beggar, but weird. He was whispering something to other guys. They replied. Only thing I caught was "You want cocaine?". When he nodded and gave them money and received one white pill in return.

I haven't read third chapter there. I did quick computation and linear regression in my head. Only thing that could top the other would be shooting and I don't want to be there during shooting. So I left.

Only later somebody pointed out that there is more common sequence that goes: "sex, drugs and rock'n'roll". I regretted this, I would love to hear some rock'n'roll.

So I slowly walked to the Time square and played chess there. I have won and because of that world was beautiful. Even the guys in pose I would describe as typical New York pose were bearable. Typical New York pose is when you are lying on the pavement and have one hand in your pants (not protecting your wallet thought, probably protecting something different).

Washington was also nice, we only didn't understand one thing, why is

Capitol closed on Sunday. We spend a lot time in museums. They were free!

In Washington there are very nice and polite security guys. As we were falling asleep on various locations they only softly reminded us that we shloudn't sleep there. So if you weren't, you should go and sleep there in the park.

Conclusion

From previously mentioned follows that we liked our stay in Rutgers. We also learned a lot, especially about hardness and polynomial hierarchy.

Our surveys show that after the stay, we like computer science in the average 7% more. We are thinking about getting PhD about 11% more and we like Scarlet Knights 94665% more.

On the other hand, we like bureaucracy 17% less and average free food 0.4% less (but this can be attributed to abundance of the high quality free food.

References

[1] My blog (weekly log and primary source of informations):
`http://reu.dimacs.rutgers.edu/~jsvoboda/`

[2] How to contact guy from Madison Square Garden (Instructive video):
`https://www.youtube.com/watch?v=DLzxrzFCyOs`

The US participants on the trip to Český Krumlov.