JAGIELLONIAN UNIVERSITY
FACULTY OF MATHEMATICS AND COMPUTER SCIENCE
THEORETICAL COMPUTER SCIENCE DEPARTMENT

Michał Farnik

A hat guessing game

Ph.D. thesis

Advisor:
prof. Jarosław Grytczuk
Auxiliary advisor:
dr. Bartłomiej Bosek

Kraków 2015

I would like to thank my advisors, Jarosław Grytczuk and Bartłomiej Bosek for introducing me to the topic and for many helpful discussions.

I am also grateful to my family, especially my wife, for the support they have been giving to me.

Contents

Introduction

In our thesis we study a variant of a well known hat guessing game. In this variant players are given hats of different colors. Each player can see the hats of several other players but not his own, he guesses the color of the hat he is wearing. The players play as a team and their goal is to ensure that at least one of them guesses correctly.

We start the first chapter with a detailed description of the origin of the hat guessing game. We recall several versions of the game, explain why the game has become so popular and present interesting results and applications. Afterwards we give a complete description of the game we focus on. We also introduce the notation that will be used throughout the thesis. We conclude the first chapter with examples providing some insight into the problem.

In the second chapter we show that the number of colors of hats that allows the players to form a winning strategy is bounded by $O(\Delta(G))$. We introduce here two useful tools. The first one is the famous Lovász local lemma [**EL**] – by now a 40 year old classic with plentiful versions optimized for different purposes. The second is entropy compression [**Mos**] – a relatively newly acquired tool which was developed while seeking a derandomized version of Lovász local lemma. It is yet another example showing that interdisciplinary reasonings yield the most ingenious solutions, in this case using information theory in a problem of graph theory and combinatorics.

The third chapter is devoted to showing the $O(\mathrm{col}(G))$ bound. This is where we believe the true potential of the hat guessing game lies. The $O(\Delta(G))$ bound is interesting by itself and shows a nice application of the Lovász local lemma and entropy compression. The proof of it shows how heavily those methods rely on the parameter $\Delta(G)$. We seek for a method that would allow to refine the $O(\Delta(G))$ bound to a $O(\mathrm{col}(G))$ bound.

In the fourth chapter we introduce a variant of the game where each player can guess multiple times and incorporate the number of guesses into the bound for the number of colors. This may seem to be only a slight alteration of the game however allowing the players even two guesses on graphs as simple as stars gives them quite unexpected possibilities.

We conclude the thesis with the fifth chapter containing a summary of the obtained results and directions for future research.

CHAPTER 1

The Game

1.1. The origin

The earliest hat guessing game we found a reference to is from 1961 ([**Gar**]) however the one that inspired us and many others was formulated by T. Ebert in his Ph.D. Thesis ([**Ebe**]) in 1998. The game was very nicely advertised by S. Robinson in The New York Times ([**Rob**]): "The reason this problem is so captivating, mathematicians say, is that it is not just a recreational puzzle to be solved and put away. Rather, it has deep and unexpected connections to coding theory, an active area of mathematical research with broad applications in telecommunications and computer science. In their attempts to devise a complete solution to the problem, researchers are proving new theorems in coding theory that may have applications well beyond mathematical puzzles."

In the game there are n players (originally seven prisoners), each player gets either a red or a blue hat placed on his head. The color of each hat is chosen randomly, both colors are equally probable and the choices are independent. The players may agree on a strategy before the game begins however once the hats are placed no communication is allowed. Every player can see the hats of all of the other players but not his own. Once the players have seen each other's hats, each player must simultaneously guess the color of his own hat or pass. The team wins if at least one player guesses his hat color correctly and no one guesses his hat color wrong, otherwise the team loses. The aim is to devise a strategy maximizing the probability of winning.

Let us briefly analyze the simplest case nontrivial case – when $n = 3$. The naive strategy for any number of players is that one player says always "blue" and the others pass. This gives a $1/2$ win ratio. But we can give a better strategy: if the two hats a player sees are of the same color then he says the other color, if the two hats have distinct colors then he passes. This strategy fails only if all three hats have the same color and thus has $3/4$ win ratio. This is in fact the best ratio we can obtain. Note that for every hat placement where a player answers correctly there is a placement where he answers incorrectly – the placements differ only by the color of his hat. With this in mind we see that the best possible strategy is when for each placement either all players guess incorrectly or one guesses correctly and other pass. In this situation we can obtain a win ratio of $n/(n+1)$, as it is in the example above.

To solve the problem in general one could use the "deep and unexpected connections to coding theory". In [**LS**] H. Lenstra and G. Seroussi showed that strategies for the game with n players are equivalent to binary covering codes of length n and radius one. Optimal strategies are equivalent to minimal binary covering codes. For $n = 2^k - 1$ the optimal solution was obtained in [**EMV**] via Hamming Codes. The solution is also nicely presented in [**BM**] and related to the Kirkman's fifteen schoolgirls problem. For $n = 2^k$ the optimal solution via extended Hamming codes is described in [**CHLL**].

The hat guessing game is not only related to coding theory. In [**BGK**] the authors describe its impact on genetic programming. In [**AFGHIS**] and [**Imm**] the authors use a version of the game for derandomization of auctions. In [**GKRT**] the authors point out a possible application to examining DNA.

There are two obvious generalizations of Ebert's game. One is that we allow the hats to be in K colors instead 2. The other is that we introduce a visibility graph describing which hats each player can see. In the original problem the visibility graph was a clique. Various results were obtained by M. Krzywkowski in a series of papers ([**Krz1**]-[**Krz5**]) and summed up in his Ph.D. Thesis [**Krz6**].

Another generalization of Ebert's game is to require for a win that at least k of the players guess correctly while other pass. This was studied in [**MSY**].

An interesting variant of the game was presented by S. Riis in [**Rii**]. The players were not allowed to pass and to win all had to guess correctly. Again the aim was to find a strategy maximizing the probability of winning. The game is used to approach problems in Circuit Complexity and Network Coding. This subject is further studied in [**WCR**].

In [**Win**] P. Winkler presented a game where each player guessed and the goal was to maximize the minimal number of correct guesses. This version was further studied in [**Doe**].

There are also several interesting versions of the hat guessing game we do not mention here. They can be found in [**Fei1**], [**Fei2**], [**BHKL**] and [**HT**].

1.2. The variant we study

We consider the following variant of the hat guessing game: there are n players and an adversary. The adversary places a hat of one of K colors on the head of each player. The players are placed in vertices of a loopless graph G, two players can see each other's hats if and only if they are placed in vertices adjacent in G. No other interaction is allowed. Each player makes a private guess what hat he is wearing. The goal of the players is to ensure that at least one of them guesses correctly. To this end they are allowed to meet before the hats are placed and determine a public deterministic strategy (public means that everyone, including the adversary, knows the strategy and deterministic means that the guess of each player is determined by the hats he sees on his neighbors). The players know the graph G and their placement before determining their strategy.

We say that the players have a winning strategy for graph G and K colors if they have a strategy such that for every hat placement there is at least one player who guesses correctly. Otherwise, i.e. when for each strategy there is a hat placement where none of the players guesses correctly, we say that the adversary has a winning strategy.

1.3. Notation

We denote the graph hosting the players by G and call it the *visibility graph*. We will identify a player and the corresponding vertex. We will also identify a distribution of hats with a coloring of the vertices. Thus the statement "vertex v guesses it's color correctly" is the same as "the player corresponding to the vertex v guesses the color of his hat correctly".

Furthermore we use the following notions:

- $V = V(G)$ — the set of vertices of G. Throughout the thesis we will assume that the vertices of G are labeled with numbers $1, \dots, n$.
- $E = E(G)$ — the set of edges of G
- $n = n(G) = |V(G)|$ — the number of vertices of G, i.e. the number of players

- $d(v)$ — the degree of a vertex $v \in V$
- $\Delta = \Delta(G)$ — the maximal degree of G, i.e. $\max\{d(v)|\ v \in V\}$
- $N(v)$ — the neighborhood of a vertex $v \in V$, i.e. $\{w \in V|\ vw \in E\}$. Note that since we assume that G is loopless we have $v \notin N(v)$.

  If the set of vertices is ordered then we also consider the positive and negative neighborhoods: $N_+(v) = \{w \in V|\ vw \in E\ v < w\}$ and $N_-(v) = \{w \in V | vw \in E\ v > w\}$.
- col $=$ col(G) — the coloring number of G, i.e. the minimum over all orderings of V of $\max\{|N_-(v)| :\ v \in V\}$.

  Whenever we make use of the parameter col(G) we assume that for the ordering of $V(G)$ induced by the labeling we actually have col$(G) = \max\{|N_-(v)| :\ v \in V\}$.
- $[x]$ — the set of integers $\{1, \dots, x\}$. We use $[x]_0$ to denote the set $\{0, \dots, x\}$.
- K — the number of colors of hats. We represent the set of colors by $[K]$.
- ψ_v — the guessing strategy of the player corresponding to the vertex $v \in V$.

  The strategy is a priori a mapping $\psi'_v : [K]^{N(v)} \to [K]$, however it extends to a mapping $\psi_v : [K]^n \to [K]$ by composing ψ'_v with the projection $[K]^n \to [K]^{N(v)}$. Obviously ψ_v is invariant with respect to all coordinates which are not in $N(v)$, i.e. if $x_u = x'_u$ for every $u \in N(v)$ then $\psi_v(x_1, \dots, x_n) = \psi_v(x'_1, \dots, x'_n)$.
- ψ — the guessing strategy of the players as a team. It is the n-tuple of individual strategies $\psi := (\psi_1, \dots, \psi_n)$.
- HG $=$ HG(G) — the hat guessing number of a graph G. It is the largest integer K such that the players have a winning strategy for graph G and K colors.
- HG(s) $=$ HG(s, G) — the s-hat guessing number of a graph G, where s is a positive integer. It is the largest integer K such that the players have a winning strategy for graph G and K colors in a game where each player can guess s times. Obviously HG$(G) =$ HG$(1, G)$ and $s \leqslant$ HG$(s, G) <$ HG$(s + 1, G)$.

- $\binom{[K]}{s}$ — the set of subsets of $[K]$ of cardinality s, i.e. the set of possible answers of a player in the game with s guesses.
- ϕ — a coloring of a graph G. The coloring can be seen either as a mapping $V \to [K]$ or, since the vertices are labeled, as a n-tuple in $[K]^n$.

  We also consider partial colorings, i.e. mappings $V \to [K]_0$, where 0 denotes the blank color, that is the color of uncolored vertices.
- $\phi_{|W}$ — the coloring ϕ restricted to a subset of the set of vertices $W \subset V$.

1.4. Simple observations and known facts

In this section we make a few simple observations to allow the reader to get familiar with the hat guessing game. Afterwards we quote some known results.

Remark 1.1. If $K > ns$, i.e. the number of colors is greater than the product of the number of vertices and the number of guesses then the adversary has a winning strategy. In particular $\mathrm{HG}(s, G)$ is well defined and $\mathrm{HG}(s, G) \leqslant ns$.

Proof. Fix a player v. For every coloring of the hats of other players player v has a fixed guess. His answer is correct for s of the K colors his hat may have. Thus among the K^n possible hat distributions there are sK^{n-1} such that player v guesses correctly. Thus there are nsK^{n-1} correct guesses among all players and hat distributions. If $K > ns$ then there must be a hat distribution where no player guesses correctly. □

Remark 1.2. If $K > K'$ and the players have a winning strategy for graph G and K colors then they also have a winning strategy for graph G and K' colors.

If G' is a subgraph of G then $\mathrm{HG}(s, G') \leqslant \mathrm{HG}(s, G)$.

We omit the proof of the remark above since it is quite obvious.

Remark 1.3. $\mathrm{HG}(s, K_n) = ns$.

Proof. From Remark 1.1 we know that $\mathrm{HG}(s, K_n) \leqslant ns$. The following strategy shows that $\mathrm{HG}(s, K_n) \geqslant ns$:

Let ϕ be the coloring of the hats. The i-th player will assume that the value $t_0 := \sum_{v \in V} \phi(v) \pmod{ns}$ belongs to the set $\{s(i-1), \ldots, si-1\}$. Obviously precisely one player will make a correct assumption. Since the visibility graph is a clique the i-th player can calculate the value $t_i := \sum_{v \neq v_i} \phi(v) \pmod{ns}$ and thus obtain $\phi(v_i) = t_0 - t_i \pmod{ns}$. □

As a direct consequence of Remarks 1.2 and 1.3 we obtain:

Remark 1.4. If G contains an edge then $\mathrm{HG}(s, 2) \geqslant 2s$.

One could think that calculating the hat guessing number of a cycle should be quite simple. It turns out that even for small cycles this is not a trivial matter. This problem has been recently solved by W. Szczechla in [**Szcz**], he obtained the following:

Theorem 1.5.

$$\mathrm{HG}(C_n) = \begin{cases} 3 & \textit{if } 3|n \textit{ or } n = 4, \\ 2 & \textit{otherwise.} \end{cases}$$

The theorem above is a combination of the following: Theorem 1 in [**Szcz**] saying that $\mathrm{HG}(C_n) \geqslant 3$ if and only if $3|n$ or $n = 4$, Corollary 8 in [**Szcz**] saying that $\mathrm{HG}(C_n) \leqslant 3$ and the trivial Remark 1.4 giving $\mathrm{HG}(C_n) \geqslant 2$.

Another class of graphs for which we know a precise answer are trees. In [**BHKL**] there is the following:

Theorem 1.6 ([**BHKL**], Lemma 9)**.** *If T is a tree then $\mathrm{HG}(T) \leqslant 2$. Moreover for every guessing strategy for $K \geqslant 3$ colors, every vertex $v \in T$ and colors c_1, c_2 there is a coloring of the hats with K colors such that none of the players guesses correctly and the color of v is either c_1 or c_2.*

We state a slightly stronger version of this theorem in Theorem 3.10. The difference is of algorithmic nature: in [**BHKL**] one must find multiple colorings of $T \setminus \{v\}$ before one can decide whether c_1 or c_2 should be the color of v if none of the players is to guess correctly. In practice this leads to a gruesome recursion. In our method the color which may not be used to color v is identified by examining the strategy of the corresponding player, thus we can easily find the coloring using a tail recursion.

In Theorem 4.8 we generalize the result to the case of s guesses.

CHAPTER 2

The $O(\Delta(G))$ bound

In this chapter we will use entropy compression and Lovász local lemma to show that $\mathrm{HG}(G) \in O(\Delta(G))$. This result is folklore, we recall it to show how the problem has been approached.

2.1. Entropy compression

In this method we assume that the hats are in one of $K \leqslant \mathrm{HG}(G)$ colors. We take a long sequence of colors $l \in [K]^N$ and compress the information it provides using a winning strategy of the players ψ. After compression we receive a partial coloring of the graph $\phi \in [K]_0^n$, where the 0-th color is the blank. We also receive two sequences: $l^1 \in [\Delta]_0^N$ and $l^2 \in \{0,1\}^{2N}$. The key point is that the compression is lossless and the process is reversible. This gives an injection $[K]^N \to [K]_0^n \times [\Delta]_0^N \times \{0,1\}^{2N}$ and consequently $K^N \leqslant (K+1)^n(\Delta+1)^N 2^{2N}$. So $K \leqslant (K+1)^{n/N} 4(\Delta+1)$. Since the list may be arbitrarily long we may pass with N to infinity and obtain $K \leqslant 4\Delta + 4$.

We will now proceed with a formal proof.

Theorem 2.1. *The largest integer K such that the players have a winning strategy for graph G and K colors is not larger than $4\Delta(G) + 4$.*

Proof. Let K be the number of colors of hats. Assume that $K \leqslant \mathrm{HG}(G)$ and that ψ is a winning strategy for the players. Let N be an integer and $l \in [K]^N$ a sequence of colors of length N.

Given a partial coloring $\phi \in [K]_0^n$ of G we say that a vertex $v \in V$ is *colored* if $\phi(v) \in [K]$ or equivalently $\phi(v) \neq 0$. We say that k vertices are colored if and only if $k = |\phi^{-1}([K])| = n - |\phi^{-1}(\{0\})|$. We say that a vertex $v \in V$ *has to be recolored* if all vertices in $N(v) \cup \{v\}$ are colored and $\psi\big(\phi_{|_{N(v)}}\big) = \phi(v)$, i.e. if using by the adversary a coloring extending ϕ would allow player v to guess the color of his hat.

We compress l by executing N iterations of Algorithm 1. At each step $t = 0, \dots, N$ we have a partial coloring ϕ_t and three sequences: l_t, l_t^1 and l_t^2. Additionally let k_t be the number of vertices colored by ϕ_t. The partial coloring ϕ_0 is empty, i.e. $\phi_0(v) = 0$ for all $v \in V$ and $k_0 = 0$. The sequence l_0 is equal to the given sequence of colors l. The two sequences l_0^1 and l_0^2 are empty.

The t-th iteration of the algorithm is as follows:

Algorithm 1.

(1) Let c be the first color in the sequence l_{t-1}. Let l_t be equal to l_{t-1} without the first element.

(2) Let v be the vertex with smallest label among all uncolored vertices of ϕ_{t-1}, we will show later that not all vertices of ϕ_{t-1} are colored. Set $\phi_t(w) := \phi_{t-1}(w)$ for all $w \in V \setminus v$ and $\phi_t(v) := c$.

(3) Set $l_t^1 := l_{t-1}^1$ and $l_t^2 := l_{t-1}^2$. Add 0 at the end of l_t^2.

(4) If v has to be recolored in ϕ_t then set $\phi_t(v) := 0$, add 0 at the end of l_t^1 and add 1 at the end of l_t^2.

(5) For i:=1 to $d(v)$ do: Let v_i be the i-th vertex in $N(v)$, if v_i has to be recolored in ϕ_t then set $\phi_t(v_i) := 0$, add i at the end of l_t^1 and add 1 at the end of l_t^2.

First notice that in all partial colorings ϕ_t there are no vertices that have to be recolored. This can be shown by induction. Indeed in ϕ_0 there are no colored vertices. Moreover the only vertex that is colored in ϕ_t and has a different color (or rather no color) in ϕ_{t-1} is the vertex v defined in step (2). Thus the only vertices that may have to be recolored in ϕ_t are in $N(v) \cup \{v\}$. However steps (4) and (5) ensure that none of these vertices has to be recolored.

It is worth noticing that it may occur that a vertex $w \in N(v)$ has to be recolored after step (2) but either v or a vertex in $N(v) \cap N(w)$ with lower label than w becomes uncolored before w is checked in step (5). In this case w does not have to be recolored when checked in step (5) and consequently $\phi_t(w) = \phi_{t-1}(w)$.

In step (2) we claim that not all vertices are colored in ϕ_{t-1}. Indeed, since ψ is a winning strategy for the players if all vertices were colored in ϕ_{t-1} then at least one player would guess the color of his hat. By definition the corresponding vertex would have to be recolored. But we already established that in ϕ_{t-1} there are no vertices that have to be recolored.

Note that in each iteration of Algorithm 1 we delete one element from the sequence l. So the length of l_t is $N-t$. We also add one 0 to the sequence l_t^2. Moreover the number of 1's added to l_t^2 is equal to the number of terms added to l_t^1 and the number of vertices that were uncolored in steps (4) and (5). Since at each iteration we color precisely one vertex we obtain that the length of l_t^1 is equal to $t-k_t$. Consequently the length of l_t^2 is $2t-k_t$.

Thus after N iterations we obtain a partial coloring ϕ_N, an empty sequence l_N, a sequence l_N^1 of length $N-k_N$ and a sequence l_N^2 of length $2N-k_N$. Let l^1 be equal to l_N^1 with 0 added at the end k_N times and let l^2 be equal to l_N^2 with 0 added at the end k_N times.

We have constructed a mapping

$$[K]^N \ni l \mapsto (\phi_N, l^1, l^2) \in [K]_0^n \times [\Delta]_0^N \times \{0,1\}^{2N}.$$

Now we will show that this mapping is injective. To this end we will show how to recover l using the triple (ϕ_N, l^1, l^2) and the strategy ψ.

Let $\phi'_t = \operatorname{sgn} \circ \phi_t$, so that ϕ'_t is a partial coloring which gives color 1 to those vertices of G which are colored in ϕ_t. We will recover from l^1 and l^2 all the partial colorings ϕ'_t and all sequences l_t^1 and l_t^2. We know that ϕ'_0 is an empty coloring and that l_0^1 and l_0^2 are empty sequences. We execute for $t = 1, \ldots, N$ the following algorithm:

Algorithm 2.

(1) Let v be the vertex with the smallest label among all uncolored vertices of ϕ'_{t-1}. Set $\phi'_t(w) := \phi'_{t-1}(w)$ for all $w \in V \setminus v$ and $\phi'_t(v) := 1$.

(2) Set $l_t^1 := l_{t-1}^1$ and $l_t^2 := l_{t-1}^2$. Add 0 at the end of l_t^2.

(3) While the $(|l_t^2|+1)$-th element of l^2 is equal to 1 do:

  (a) Add 1 at the end of l_t^2.

  (b) Let i be the $(|l_t^1|+1)$-th element of l^1. Add i at the end of l_t^1.

  (c) If $i=0$ then set $\phi'_t(v) := 0$ else let v_i be the i-th vertex in $N(v)$ and set $\phi'_t(v_i) := 0$.

Note that Algorithm 2 is very similar to Algorithm 1. The first difference is that it does not read a color from the l_{t-1} and instead uses 1 to color the vertex with the smallest label among all uncolored vertices. The second difference is that in steps (4) and (5) Algorithm 1 uncolors certain vertices, writes the number of those vertices in l^2_t (it is the number of 1's between the t-th and $(t+1)$-th zero in l^2) and writes the labels of those vertices in l^1_t. In step (3) Algorithm 2 uses the relevant parts of l^2 and l^1 to identify the vertices uncolored by Algorithm 1 in ϕ_t and uncolors them as well in ϕ'_t. Hence by the construction $\phi'_t = \mathrm{sgn} \circ \phi_t$.

Now we are able to recover ϕ_t and l using ϕ_N, ϕ'_t and l^1_t. We reverse the process used to construct the colorings ϕ_t and execute the following algorithm for t ranging from N down to 1.

Algorithm 3.

(1) Set $\phi_{t-1} := \phi_t$. Let v be the vertex with the smallest label among all uncolored vertices of ϕ'_{t-1}.

(2) For $j := |l^1_t|$ downto $|l^1_{t-1}| + 1$ do:

 (a) Let i be the j-th element of l^1_t.

 (b) If $i > 0$ let v_i be the i-th vertex in $N(v)$ else set $v_i = v$.

 (c) We claim that in ϕ_{t-1} the vertex v_i is uncolored and all the vertices in $N(v_i)$ are colored. Set $\phi_{t-1}(v_i) = \psi\big(\phi_{|N(v_i)}\big)$.

(3) Let the t-th element of l be $\phi_{t-1}(v)$. Set $\phi_{t-1}(v) = 0$.

Note that in step (2) of t-th iteration of Algorithm 1 we set $\phi_t(w) := \phi_{t-1}(w)$ for all $w \in V \setminus v$ and $\phi_t(v) := c$ and subsequently uncolored several (possibly none) vertices in ϕ_t in steps (4) and (5). So to obtain ϕ_{t-1} from $\phi_t(w)$ we may reverse the process by assigning $\phi_{t-1} := \phi_t$, coloring in ϕ_{t-1} the vertices uncolored in steps (4) and (5) and uncoloring the vertex v. This is precisely what Algorithm 3 does. Moreover in step (3) it removes from v the color c from steps (1) and (2) of Algorithm 1.

In step (2c) we claim that in ϕ_{t-1} the vertex v_i is uncolored and all the vertices in $N(v_i)$ are colored. This is because at this point of Algorithm 3 the coloring ϕ_{t-1} is equal to the coloring ϕ_t obtained in Algorithm 1 after i-th execution of the loop in step (5). The vertex v_i is uncolored because it must have been uncolored since it was written down in l^1_t. The vertex v_i had to be recolored, so by definition all the vertices in $N(v_i)$ have been colored and moreover $\psi\big(\phi_{|_{N(v_i)}}\big)$ is the color that was removed.

Thus we have shown that the mapping

$$[K]^N \ni l \mapsto (\phi_N, l^1, l^2) \in [K]_0^n \times [\Delta]_0^N \times \{0,1\}^{2N}$$

constructed earlier has an inverse. So $K^N \leqslant (K+1)^n(\Delta+1)^N 2^{2N}$ and $K \leqslant (K+1)^{n/N}4(\Delta+1)$. Since N was chosen arbitrarily we may pass with N to infinity and obtain $K \leqslant 4\Delta + 4$. □

2.2. Lovász local lemma

We begin by recalling a symmetric version of the famous Lemma:

Lemma 2.2. *Let $\mathcal{A} = \{A_1, \ldots, A_n\}$ be a finite set of events in a probability space. For $A \in \mathcal{A}$ let $N(A)$ denote the smallest subset of $\mathcal{A}$ such that A is independent from $\mathcal{A} \setminus (N(A) \cup \{A\})$. If for all $A \in \mathcal{A}$ we have $\mathrm{P}(A) \leqslant p$ and $|N(A)| \leqslant d$ and moreover $ep(d+1) \leqslant 1$ then there is a nonzero probability that none of the events in $\mathcal{A}$ occur.*

For the sake of completeness we will provide a proof of Lemma 2.2. However to do this we will require an asymmetric version of the Lovász local lemma:

Lemma 2.3. *Let $\mathcal{A} = \{A_1, \ldots, A_n\}$ be a finite set of events in a probability space. For $A \in \mathcal{A}$ let $N(A)$ denote the smallest subset of $\mathcal{A}$ such that A is independent from $\mathcal{A} \setminus (N(A) \cup \{A\})$. If $x : \mathcal{A} \to (0,1)$ is an assignment such that for all $A \in \mathcal{A}$ we have $\mathrm{P}(A) \leqslant x(A) \prod_{B \in N(A)}(1 - x(B))$ then*

$$\mathrm{P}\left(\overline{A_1} \wedge \ldots \wedge \overline{A_n}\right) \geqslant \prod_{A \in \mathcal{A}} (1 - x(A)).$$

Proof. We prove the theorem using induction on $n = |\mathcal{A}|$. The case $n = 1$ is obvious. For $n > 1$ we claim that for all $A \in \mathcal{A}$ and $S \subset \mathcal{A}$ such that $a \notin S$ we have $\mathrm{P}(A | \bigwedge_{B \in S} \overline{B}) \leqslant x(A)$. Note that we may use conditional probability since $|S| < n$ and we have $\mathrm{P}(\bigwedge_{B \in S} \overline{B}) > 0$. Using the claim we obtain $\mathrm{P}\left(\overline{A_1} \wedge \ldots \wedge \overline{A_n}\right) = \mathrm{P}\left(A_1 | \overline{A_2} \wedge \ldots \wedge \overline{A_n}\right) \cdot \left(A_2 | \overline{A_3} \wedge \ldots \wedge \overline{A_n}\right) \cdot \ldots \cdot \mathrm{P}\left(\overline{A_n}\right) \geqslant (1 - x(A_1)) \cdot (1 - x(A_2)) \cdot \ldots \cdot (1 - x(A_n))$.

Thus all we need to do is prove the claim. We will do it using induction on the cardinality of S.

For $S = \emptyset$ we have $\mathrm{P}(A) \leqslant x(A)$ by assumption.

For the inductive step let $S_1 = S \cap N(A)$ and $S_2 = S \setminus N(A)$. Let $C_1 = \bigwedge_{B \in S_1} \overline{B}$ and $C_2 = \bigwedge_{B \in S_2} \overline{B}$. We have $\mathrm{P}(A \wedge C_1 \wedge C_2) = \mathrm{P}(A|C_1 \wedge C_2) \cdot \mathrm{P}(C_1|C_2) \cdot \mathrm{P}(C_2)$. On the other hand $\mathrm{P}(A \wedge C_1 \wedge C_2) = \mathrm{P}(A \wedge C_1|C_2) \cdot \mathrm{P}(C_2)$. Combining the two equalities we obtain $\mathrm{P}(A|C_1 \wedge C_2) = \mathrm{P}(A \wedge C_1|C_2)/\,\mathrm{P}(C_1|C_2)$.

Observe that $\mathrm{P}(A \wedge C_1|C_2) \leqslant \mathrm{P}(A|C_2) = \mathrm{P}(A) \leqslant x(A) \prod_{B \in N(A)}(1 - x(B))$. If $S_1 = \emptyset$ then $\mathrm{P}(C_1|C_2) = 1$ and we are done. Otherwise if $S_1 = \{B_1, \ldots, B_k\}$ then $\mathrm{P}(C_1|C_2) = \mathrm{P}\left(B_1|\overline{B_2} \wedge \ldots \wedge \overline{B_k} \wedge C_2\right) \cdot \left(B_2|\overline{B_3} \wedge \ldots \wedge \overline{B_k} \wedge C_2\right) \cdot \ldots \cdot \mathrm{P}\left(\overline{B_k|C_2}\right) \geqslant (1 - x(B_1)) \cdot (1 - x(B_2)) \cdot \ldots \cdot (1 - x(B_k))$, where the last inequality follows from $|S_2| < |S|$ and the inductive assumption.

Combining the inequalities from above we obtain

$$\mathrm{P}(A|C_1 \wedge C_2) \leqslant x(A) \prod_{B \in N(A)} (1 - x(B))/ \prod_{B \in S_1} (1 - x(B)) \leqslant x(A),$$

which completes the proof. □

We will now use the asymmetric version of Lovász local lemma to prove the symmetric one:

Proof of Lemma 2.2. Put $x(A) = \frac{1}{d+1}$ for all $A \in \mathcal{A}$. We have

$$\mathrm{P}(A) \leqslant p \leqslant \frac{1}{e(d+1)} < \frac{1}{d+1} \cdot \left(1 + \frac{1}{d}\right)^{-d} = \frac{1}{d+1} \cdot \left(\frac{d+1}{d}\right)^{-d} =$$

$$\frac{1}{d+1} \cdot \left(\frac{d}{d+1}\right)^{d} = \frac{1}{d+1} \cdot \left(1 - \frac{1}{d+1}\right)^{d} \leqslant x(A) \prod_{B \in N(A)} (1 - x(B)).$$

Thus from Lemma 2.3 there is a nonzero probability that none of the events in $\mathcal{A}$ occur. □

We proceed with applying Lovász local lemma to the hat game. To this end we will assign random colors to the hats.

Let Ω denote the probability space. Elementary events are colorings of the graph G, they all have equal probability which is K^{-n}. Let A_i denote the event "player i has guessed the color of his hat". Note that we will not change the probability space if we first assign random colors to all vertices except i, then allow player i to guess and conclude with assigning random color to vertex i. The probability that we assign to vertex i the color that player i has guessed is $\mathrm{P}(A_i) = K^{-1}$. Similarly if vertices i and j are independent in G then we may first assign random colors to all vertices except i and j, then allow players i and j to guess and conclude with assigning random colors to vertices i and j. This shows that $\mathrm{P}(A_i \wedge A_j) = K^{-2} = \mathrm{P}(A_i) \cdot \mathrm{P}(A_j)$. Thus if i and j are vertices independent in G then A_i and A_j are independent in Ω, i.e. if $A_j \in N(A_i)$ then $v_j \in N(v_i)$.

This shows that we may apply Lemma 2.2 with $p = K^{-1}$ and $d = \Delta$. We obtain the following:

Theorem 2.4. *The largest integer K such that the players have a winning strategy for graph G and K colors is smaller than $e(\Delta(G)+1)$.*

PROOF. Take $K = \lceil e(\Delta(G)+1) \rceil$. We have

$$pe(d+1) = K^{-1} e(\Delta(G)+1) < 1.$$

Thus the adversary has a winning strategy. □

Remark 2.5. Shearer proved in [**She**] a slightly stronger version of the Lovász local lemma. One can easily derive from his result that it suffices to assume $epd \leqslant 1$ in Lemma 2.2. As a consequence we obtain that the largest integer K such that the players have a winning strategy for graph G and K colors is smaller than $e\Delta(G)$.

CHAPTER 3

The $O(\mathrm{col}(G))$ bound

3.1. Bipolar strategies

In this section we will handle a special kind of strategies of the players – the bipolar strategies. We will show that if one restricts the strategies that the players can adopt to bipolar strategies then one has $\mathrm{HG}(G) \leqslant \mathrm{col}(G)+1$. In this chapter we assume that G is labeled with numbers from $[N]$. The labeling induces an order on $V(G)$, we assume that for this order we have $\mathrm{col}(G) = \max\{|N_{-}(v)| : \ v \in V\}$. All strategies of the players in this section are assumed to be bipolar.

Definition 3.1. We call a strategy ψ *bipolar* if for all vertices v_i, all $j \geqslant i$ and all partial colorings $(x_1, \ldots, x_{j-1}) \in [K]^{j-1}$ we have: either for all $x_j \in [K]$ the set $\psi_{v_i}(\{(x_1, \ldots, x_{j-1}, x_j)\} \times [K]^{n-j})$ is equal $[K]$ or for all $x_j \in [K]$ the set $\psi_{v_i}(\{(x_1, \ldots, x_{j-1}, x_j)\} \times [K]^{n-j})$ is a singleton.

Let us briefly explain the idea behind bipolar strategies. If the players adopt a bipolar strategy then each player watches the hats of his neighbors one after the other. At first the information he receives does not indicate what his final answer will be; in fact all answers are equally probable. The partial information affects however how he will react after seeing the following hats. At a certain point the player decides to guess and we obtain full information about his final answer.

A simple example of a bipolar strategy is the “sum modulo K” strategy from Remark 1.3. After seeing the hats of all but one of his neighbors the player has no indication on his final answer, all are equally possible. However the information he has provides a bijection between the colors of the last hat he will see and his final answers.

Now we will introduce the notion of a terminal vertex. The idea is to find the moment at which it is decided whether a player will guesses correctly.

Definition 3.2. We say that a vertex v_i is *terminal* for v_i and a partial coloring $(x_1, \ldots, x_{i-1}) \in [K]^{i-1}$ if for all $x_i \in [K]$ the set $\psi_{v_i}(\{(x_1, \ldots, x_{i-1}, x_i)\} \times [K]^{n-i})$ is a singleton.

We say that a vertex v_j is *terminal* for v_i and a partial coloring $(x_1, \ldots, x_{j-1}) \in [K]^{j-1}$ if $j > i$, for all $x_j \in [K]$ the set $\psi_{v_i}(\{(x_1, \ldots, x_{j-1}, x_j)\} \times [K]^{n-j})$ is a singleton and the set $\psi_{v_i}(\{(x_1, \ldots, x_{j-1})\} \times [K]^{n-j-1})$ is not a singleton.

We say that a vertex v_j is *terminal* for v_i and a coloring $\phi \in [K]^n$ if v_j is terminal for v_i and $\phi_{|\{v_1,\ldots,v_{j-1}\}}$.

We state a few simple remarks regarding terminal vertices:

Remark 3.3. If v_i is terminal for v_i then since ψ_{v_i} does not depend on the i-th coordinate all the singletons have to be the same and are in fact equal to $\psi_{v_i}(\{(x_1, \ldots, x_{i-1})\} \times [K]^{n-i-1})$. In other words player v_i decides what will be his guess before he receives a hat. So there is precisely one $t_i \in [K]$ such that $\psi_{v_i}(\{(x_1, \ldots, x_{i-1}, t_i)\} \times [K]^{n-i}) = \{t_i\}$ and player v_i guesses correctly.

Remark 3.4. If v_j is terminal for v_i for $j > i$ then from bipolarity of ψ we obtain that $\psi_{v_i}(\{(x_1, \ldots, x_{j-1})\} \times [K]^{n-j-1}) = [K]$. So the map $[K] \ni x_j \mapsto c(x_j) \in [K]$ such that $\psi_{v_i}(\{(x_1, \ldots, x_{j-1}, x_j)\} \times [K]^{n-j}) = \{c(x_j)\}$ is surjective. Consequently it is also bijective and there is precisely one $t_i \in [K]$ such that $\psi_{v_i}(\{(x_1, \ldots, x_{j-1}, t_i)\} \times [K]^{n-j}) = \{x_i\}$ and player v_i guesses correctly.

Remark 3.5. For fixed v_i and ϕ there is precisely one terminal vertex v_{j_0}. The index j_0 is the smallest among $j \geqslant i$ such that $\psi_{v_i}(\{(\phi(v_1), \ldots, \phi(v_j))\} \times [K]^{n-j})$ is a singleton. Furthermore v_{j_0} is either v_i or belongs to $N_+(v_i)$. Consequently for fixed ϕ a vertex v_j can be terminal for itself and vertices in $N_-(v_i)$, so for at most $\mathrm{col}(G) + 1$ vertices.

We can now prove the following theorem:

Theorem 3.6. *If ψ is a bipolar strategy of the players for a graph G and $K > \mathrm{col}(G) + 1$ colors then there is a hat placement where none of the players guesses correctly.*

Proof. We will construct the coloring inductively by extending a partial coloring of the first k vertices.

Let $(x_1, \dots, x_k)$ be the partial coloring of the first k vertices such that players v_i for $i = 1, \dots, k$ either cannot guess yet because $\psi_{v_i}(\{(x_1, \dots, x_k)\} \times [K]^{n-k}) = [K]$ or guess incorrectly because $\psi_{v_i}(\{(x_1, \dots, x_k)\} \times [K]^{n-k}) = \{c\}$ for some color $c \neq x_i$.

By Remark 3.5 there are at most $\mathrm{col}(G)+1$ vertices v_i, where $i \in \{1, \dots, k+1\}$, such that v_{k+1} is terminal for v_i and $(x_1, \dots, x_k)$. By Remarks 3.3 and 3.4 for each v_i there is one t_i such that $\psi_{v_i}(\{(x_1, \dots, x_k, t_i)\} \times [K]^{n-j}) = \{x_i\}$. Since $K > \mathrm{col}(G) + 1$ we pick x_{k+1} to be the color with the smallest number which is not equal to any of the t_i.

Thus we obtain a partial coloring $(x_1, \dots, x_{k+1})$. If a player guessed incorrectly for the coloring of k vertices then he still guesses incorrectly for the coloring of $k+1$ vertices. So we only need to check that a player v_i who could not guess for $(x_1, \dots, x_k)$ and guessed for $(x_1, \dots, x_{k+1})$ does not guess correctly. Note that v_{k+1} is the terminal vertex of v_i and we picked x_{k+1} so that none of the players v_i such that v_{k+1} is terminal for v_i guesses correctly. □

Based on Theorem 3.6 we make the following:

Conjecture 3.7. $\mathrm{HG}(G) \leqslant \mathrm{col}(G) + 1$.

Obviously to prove this conjecture one would "only" need to show that every winning guessing strategy can be used to obtain a winning bipolar strategy. However at the end of this chapter we will present Example 3.13 which shows that this may not always be the case.

3.2. Trees

The simplest interesting field to test Conjecture 3.7 is the case $\mathrm{col}(G) = 1$, i.e. when G is a forest. The hat guessing game may be led separately on each connected component thus without loss of generality we may assume that G is a tree. We will show that if G is a tree then $\mathrm{HG}(G) \leqslant 2$. We start by introducing the following concept:

Definition 3.8. Let T be a tree with root r. Let $[K]$ be the set of colors and $\psi_r : [K]^n \to [K]$ be the strategy of player r. We say that a color $c \in [K]$ is *dominant* for (T, r, ψ_r) if the set $\psi_r^{-1}(c)$ contains a cube of size $[K-1]^n$.

The usefulness of dominant colors comes from the following fact:

Remark 3.9. If $K > 2$ then (T, r, ψ_r) may have at most one dominant color. Indeed let c_1 and c_2 be dominant colors and let $\prod_{i=1}^{n}\{x_1^i, \ldots, x_{K-1}^i\}$ and $\prod_{i=1}^{n}\{y_1^i, \ldots, y_{K-1}^i\}$ be the cubes provided by the definition of a dominant color. Since for each i we have $|\{x_1^i, \ldots, x_{K-1}^i\} \cap \{y_1^i, \ldots, y_{K-1}^i\}| \geq K - 2 > 0$ the intersection of those cubes is nonempty. So $\psi_r^{-1}(c_1) \cap \psi_r^{-1}(c_2) \neq \emptyset$, which implies $c_1 = c_2$.

We will now prove the following:

Theorem 3.10. *Let T be a tree with root r. Let $[K]$ be the set of colors and ψ the strategy of the players. If $K > 2$ then for each color c which is not dominant there is a coloring ϕ such that $\phi(r) = c$ and none of the players guesses correctly.*

Proof. We will use induction on the size of T.

If $|T| = 1$ then ψ_r is constant and player r will not guess any color different form the dominant color, which is the value of ψ_r.

For the inductive step, let $N(r) = \{r_1, \ldots, r_t\}$ and let $T_1, \ldots, T_t$ be the trees who are the connected components of $T \setminus r$. Choose r_i to be the root of T_i. Let us introduce the strategy ψ_i on T_i. Note that the guess of a vertex T_i depends only on the colors assigned to vertices in T_i and c. Thus we may take ψ_i to be the restriction of ψ to T_i with the additional condition that r is colored with c. Let c_i be the dominant color of (T_i, r_i, ψ_i) or let $c_i = 1$ if (T_i, r_i, ψ_i) does not have a dominant color. Since c is not dominant there is a tuple $(x_1, \ldots, x_t) \in \prod_{i=1}^{t}[K] \setminus \{c_i\}$ such that $\psi_r(x_1, \ldots, x_t) \neq c$. From the inductive hypothesis we obtain colorings ϕ_i of T_i such that $\phi_i(r_i) = x_i$ and none of the players associated to vertices in T_i guesses correctly. Now we define the coloring ϕ by taking $\phi(r) = c$ and $\phi(v) = \phi_i(v)$ for $v \in T_i$. Note that neither the player associated with r nor the players associated to vertices in any of the T_i guess correctly. □

3.3. Complete bipartite graphs

In this section we will show that for every K there exists a complete bipartite graph G with $\mathrm{HG}(G) \geqslant K$. The example is taken from Theorem 7 in [**BHKL**], we modified it slightly so it would better relate to our considerations. Before we give the example we will briefly explore the concept of a separating set.

Definition 3.11. Let $\mathcal{A} \subset [K]^{[K]^n}$ be a set of functions from $[K]^n$ to $[K]$. We say that $\mathcal{A}$ is *separating* if for every subset $S \subset [K]^n$ of cardinality K there is a function $f \in \mathcal{A}$ such that $f(S) = [K]$.

We denote by $\operatorname{sep}(K)$ the size of the smallest separating set for $n = K - 1$.

Obviously $[K]^{[K]^n}$ is a trivial example of a separating set. To construct a slightly less trivial separating set take a K-tuple $P = (p_1, \ldots, p_K)$ of permutations of $[K]$. Define $g_P : [K]^2 \to [K]$ by setting $g_P(x_1, x_2) = p_{x_1}(x_2)$. Note that for $S \subset [K]^2$ of cardinality K one can easily find a tuple P such that $g_P(S) = [K]$. Similarly having K-tuples $P_1, \ldots, P_{n-1}$ we define $f_i : [K]^{i+1} \to [K]^i$ by setting $f_i(x_1, \ldots, x_i, x_{i+1}) = (x_1, \ldots, x_{i-1}, g_{P_i}(x_i, x_{i+1}))$. Then we take $f = f_1 \circ \ldots \circ f_{n-1}$. Again, for a set $S \subset [K]^n$ of cardinality K we can choose such $P_1, \ldots, P_{n-1}$ that $f(S) = [K]$. Thus we obtain the following:

Remark 3.12. The set $\mathcal{A}_{K,n}$ of all functions f that can be obtained using $K(n-1)$ permutations as described above is separating. In particular $\operatorname{sep}(K) \leqslant (K!)^{K(K-2)} < K^{K^3}$.

In fact $\operatorname{sep}(K)$ is even smaller than stated above. For instance one does not need to take all permutations. The aim of the construction above was not to find the best possible bound for $\operatorname{sep}(K)$ but rather to find a separating set related to bipolar functions. We will describe the relation in detail after we show the following example:

Example 3.13. Let G be the complete bipartite graph $K_{K-1,\operatorname{sep}(K)}$. We have $\operatorname{HG}(G) \geqslant K$.

Proof. Let $n = K - 1$ and let $\mathcal{A} \subset [K]^{[K]^n}$ be a separating set of cardinality $\operatorname{sep}(K)$. We identify the vertices from the right side of G with elements of $\mathcal{A}$ and for every $f \in \mathcal{A}$ we let $f : [K]^n \to [K]$ be the strategy of the player identified with f.

We claim that for every coloring $\phi_R \in [K]^{\operatorname{sep}(K)}$ of the right side of G there are at most n distinct colorings $\phi_L \in [K]^n$ of the left side of G such that for the combined coloring (ϕ_L, ϕ_R) of G all the vertices on the right side guess incorrectly. Indeed suppose that S is the set of all such colorings ϕ_L. If the cardinality of S is at least K then there is $f \in \mathcal{A}$ such that $f(S) = [K]$. But then $\phi_R(f) = f(\phi_L)$ for some $\phi_L \in S$ so that player f guesses correctly for the coloring (ϕ_L, ϕ_R) – a contradiction.

Thus there is a map $\Phi : [K]^{\mathcal{A}} \to ([K]^n)^n$ such that if all the vertices on the right side of G guess incorrectly for a coloring (ϕ_L, ϕ_R) then $\phi_L = (\Phi(\phi_R))(i)$ for some $i \in [n]$. We define the strategy ψ_i of the i-th player on the left side of G by setting $\psi_i(\phi_R) = ((\Phi(\phi_R))(i))(i)$.

Let (ϕ_L, ϕ_R) be a coloring of G. Either one of the vertices on the right side of G guesses correctly or $\phi_L = (\Phi(\phi_R))(i)$ for some $i \in [n]$. If the latter is true then $\psi_i(\phi_R) = \phi_L(i)$ so the i-th vertex on the left side of G guesses correctly. Thus we have defined a winning guessing strategy. □

Now we will briefly justify why we believe Example 3.13 is interesting. The first reason is that it shows that the hat guessing number cannot be bounded from above by a function of the clique number.

The second is that we believe that $K_{K-1,\mathrm{sep}(K)}$ has no bipolar winning strategy. Instead of a formal proof we only present an intuitive argument since bipolar strategies are only a means of handling Conjecture 3.7. The set $\mathcal{A}_{K,n}$ from Remark 3.12 was constructed in such a way that adopting the strategies $f \in \mathcal{A}_{K,n}$ in the proof of Example 3.13 will give a bipolar strategy for vertices on the right side of the graph. Of course $|\mathcal{A}_{K,n}| > \mathrm{sep}(K)$ but this problem could possibly be overcome. The real issue lies in the vertices on the left side of the graph and in the way they obtain their information. They start with the set $[K]^n$ of possible colorings of the left side of G. Each color observed on the right side of G allows to subtract a set from $[K]^n$, possibly entirely contained in previously subtracted sets. In the end they are left with a set of cardinality at most $n = K - 1$ and each of them "eliminates" one of the points that could give the adversary a winning coloring. The problem is that the set of remaining possible colorings can easily become unbalanced. The players could for example obtain the information that the coloring is in fact from the set $[K-1]^n$, i.e. none of their hats has color K.

There are ways one could try to fix this problem but after some consideration we came to the conclusion that they are doomed to fail.

Thus one could say that Example 3.13 undermines Conjecture 3.7. To counter this we present the following:

Example 3.14. Let G be the complete bipartite graph $K_{n,m}$. We have $\mathrm{HG}(G) < n + 2$.

Proof. Let ψ be the guessing strategy for $n+2$ colors. Let $V(G) = L \cup R$ be the partition of vertices with $|L| = n$. Let $\phi_1, \ldots, \phi_{n+1}$ be the colorings of L such that $\phi_i(v) = i$ for all $v \in L$. For any $v \in R$ the set $S_v = \{\psi_v(\phi_1), \ldots, \psi_v(\phi_{n+1})\}$ has cardinality at most $n + 1$, we set $\phi_R(v)$ to be the smallest color not in S_v. The set $\{\psi_v(\phi_R) \mid v \in L\}$ has cardinality at most n so there is $i \in [n + 1]$ not contained in this set. Note that for ψ the coloring (ϕ_i, ϕ_R) is a winning coloring for the adversary. □

Together Examples 3.13 and 3.14 give

$$\mathrm{HG}(K_{K-1,\mathrm{sep}(K)}) = K = \mathrm{col}(K_{K-1,\mathrm{sep}(K)}) + 1$$

which in our opinion makes Conjecture 3.7 even more viable and interesting.

We conclude this section by repeating a question from [**BHKL**]: is there a bipartite graph G satisfying $\mathrm{HG}(G) \geqslant K$ whose size is polynomial in K? To this we add a question of our own: is $\mathrm{sep}(K)$ polynomial in K?

CHAPTER 4

Multiple guesses

4.1. Generalized results

In this chapter we will analyze a variant of the game in which each player may guess s times. In this variant the values of the strategies ψ_i are subsets of $[K]$ of cardinality s instead of elements of $[K]$. We say that the player i guesses correctly for the coloring ϕ if $\phi(v_i) \in \psi_i(\phi)$. In this section we show that the methods involving entropy compression, Lovász local lemma or bipolar strategies generalize well and provide a bound s times larger than the original one. We start with entropy compression:

Theorem 4.1. *The largest integer K such that the players have a winning strategy for graph G, K colors and s guesses is not greater than $4s(\Delta(G)+1)$.*

Proof. We use a strategy very similar to the one from the proof of Theorem 2.1. Recall that in the proof of Theorem 2.1 we said that a vertex $v \in V$ has to be recolored if using by the adversary a coloring extending ϕ would allow player v to guess the color of his hat. In the current context it means that $\phi(v) \in \psi\big(\phi_{|_{N(v)}}\big)$. In Algorithm 1 each time a vertex had to be recolored the string l_t^1 was augmented by an element of $[\Delta]_0$. This allowed to recover the order of recoloring vertices in Algorithm 2. Then in Algorithm 3 the color $\psi\big(\phi_{|_{N(v)}}\big)$ was used to restore the color removed from the vertex that has been recolored. To cope with the increased number of guesses we can store in l_t^1 pairs from $[\Delta]_0 \times [s]$. The second element indicates which element of $\psi\big(\phi_{|_{N(v)}}\big)$ is actually equal to $\phi(v)$.

After this change we obtain an injection

$$[K]^N \to [K]_0^n \times ([\Delta]_0 \times [s])^N \times \{0,1\}^{2N}$$

which yields the bound $K \leqslant 4s(\Delta+1)$. □

Now we will generalize the result using Lovász local lemma:

Theorem 4.2. *The largest integer K such that the players have a winning strategy for graph G, K colors and s guesses is smaller than $es(\Delta(G)+1)$.*

Proof. Similarly to the proof of Theorem 2.4 we use Lemma 2.2. The only difference is that now the probability of a player guessing correctly is s/K. Thus for $K > es(\Delta(G)+1)$ we have

$$pe(d+1) = K^{-1}se(\Delta(G)+1) < 1,$$

so the adversary has a winning strategy. □

To generalize the result using bipolar strategies we first have to generalize the notion of bipolar strategy. There are several ways to transfer the idea to the variant with multiple guesses. We have chosen the one we believe is the simplest, the most natural and yet fairly general.

Definition 4.3. We call a strategy $\psi : [K]^n \to \mathcal{P}([K])$ s-bipolar if there are bipolar strategies $\psi^1, \ldots, \psi^s$ such that for all $\phi \in [K]^n$ we have $\psi(\phi) \subset \{\psi^1(\phi), \ldots, \psi^s(\phi)\}$.

Now we can prove the following generalization of Theorem 3.6

Theorem 4.4. *If ψ is an s-bipolar strategy of the players for a graph G and $K > s(\mathrm{col}(G)+1)$ colors then there is a hat placement where none of the players guesses correctly.*

Proof. Fix bipolar strategies $\psi^1, \ldots, \psi^s$ as in Definition 4.3. The construction of the coloring will be very similar to the one from the proof of Theorem 3.6.

Let $(x_1, \ldots, x_k)$ be the partial coloring of the first k vertices. For each bipolar strategy ψ^j there are at most $\mathrm{col}(G)+1$ vertices v_i such that v_{k+1} is terminal for v_i and $(x_1, \ldots, x_k)$. Hence there are at most $\mathrm{col}(G)+1$ colors $c \in K$ such that there is a coloring $(x_1, \ldots, x_k, c, x_{k+2}, \ldots, x_n)$ for which the strategy ψ^j allows a player to guess correctly. Pick x_{k+1} to be the color with the smallest number which is not among those at most $\mathrm{col}(G)+1$ colors for any $j \in [s]$. We may do so since $K > s(\mathrm{col}(G)+1)$. □

Theorem 4.4 could suggest that a generalization of Conjecture 3.7 is true, namely that $\mathrm{HG}(s, G) \leqslant s(\mathrm{col}(G) + 1)$. However the generalized version is not true even for $\mathrm{col}(G) = 1$ and $s = 2$, as we will show in Example 4.5.

4.2. Trees

In this section we will analyze the game with s-guesses on trees. We start with an example showing that unlike for one guess there are trees for which players with $s > 2$ guesses can achieve more than players on K_2.

Example 4.5. Let $G = K_{1,5}$ be the star with 5 leafs. We have $\mathrm{HG}(2, G) \geq 6$, i.e. the players have a winning strategy on G for 6 colors and 2 guesses.

Proof. We represent the set of possible colorings as $[6]^6 = [6] \times [6]^5$. On the first coordinate we put the color of the hat of the player corresponding to the vertex of degree 5. Similarly an element of $[6]^5$ corresponds to a coloring of the hats of players corresponding to vertices of degree 1.

Thus the strategy of the first player is represented by a mapping $\psi_1 \circ \pi_1$ where $\pi_1 : [6]^6 \to [6]^5$ is the projection forgetting the first coordinate and $\psi_1 : [6]^5 \to \binom{[6]}{2}$ is a mapping whose values are sets of cardinality at most 2. Similarly for $i = 2, \ldots, 6$ the strategy of the i-th player is represented by a mapping $\psi_i \circ \pi_i$ where $\pi_i : [6]^6 \to [6]$ is the projection on the first coordinate and $\psi_i : [6] \to \binom{[6]}{2}$.

We slightly abuse the terminology and call the set

$$A_i = \big\{(x_1, \ldots, x_6) \in [6]^6 \mid x_i \in \psi_i \circ \pi_i(x_1, \ldots, x_6)\big\}$$

the graph of $\psi_i \circ \pi_i$. By definition the i-th player guesses correctly if and only if the point in $[6]^6$ corresponding to the coloring belongs to A_i. Thus to show that the players have a winning strategy we must choose the mappings ψ_i in such way that the graphs of $\psi_i \circ \pi_i$ will cover the whole set $[6]^6$.

For $i = 2, \ldots, 6$ we set

$$\psi_i(j) = \begin{cases} \{5, 6\} & \text{for } i > j, \\ \{3, 4\} & \text{for } i = j, \\ \{1, 2\} & \text{for } i < j. \end{cases}$$

For $j = 1, \ldots, 6$ let

$$B_j = \{j\} \times [6]^5 \cap \left([6]^6 \setminus \bigcup_{i=2}^{6} A_i \right).$$

Notice that $B_1 = \{1\} \times \{1,2,3,4\}^5$ and for $j = 2, \ldots, 6$ we have $B_j = \{j\} \times \{3,4,5,6\}^{j-2} \times \{1,2,5,6\} \times \{1,2,3,4\}^{6-j}$.

Now we set

$$\psi_1(x_2, \ldots, x_6) = \{j \in [6] \mid (j, x_2, \ldots, x_6) \in B_j\}.$$

We have

$$\bigcup_{i=1}^{6} A_i = \bigcup_{i=2}^{6} A_i \cup \bigcup_{j=1}^{6} B_j = \bigcup_{i=2}^{6} A_i \cup \left([6]^6 \setminus \bigcup_{i=2}^{6} A_i \right) = [6]^6$$

thus the players have a winning strategy, provided that ψ_1 is a valid strategy, i.e. its values have cardinality at most 2.

Assume that there are $x_2, \ldots, x_6$ and $j_1 < j_2 < j_3$ such that $\{j_1, j_2, j_3\} \subset \psi_1(x_2, \ldots, x_6)$. We have $\psi_{j_1}(j_2) = \{1,2\}$ so $\{j_2\} \times [6]^{j_2-2} \times \{1,2\} \times [6]^{6-j_2} \subset A_{j_1}$. Furthermore $(j_2, x_2, \ldots, x_6) \in B_{j_2}$ and $A_{j_1} \cap B_{j_2} = \emptyset$. Thus $x_{j_2} \notin \{1,2\}$. Similarly from $\psi_{j_2}(j_2) = \{3,4\}$ follows $x_{j_2} \notin \{3,4\}$ and from $\psi_{j_3}(j_2) = \{5,6\}$ follows $x_{j_2} \notin \{5,6\}$. This is a contradiction since $x_{j_2} \in [6]$. □

We will show in Theorem 4.8 that in fact $\mathrm{HG}(2, K_{1,5}) = 6$.

Example 4.5 can be generalized for arbitrary s: if we can fit K cubes of size $(K-s)^n$ into the cube K^n in such way that none $s+1$ overlap at one point then the players have a winning strategy on $K_{1,n}$ for K colors and s guesses. On the other hand for $K_{1,n}$ every possible strategy of the players corresponding to leafs can be encoded by embedding K cubes of size $(K-s)^n$ into K^n. If at least $s+1$ cubes overlap at one point then we can obtain a coloring such that none of the players guesses correctly.

Before we can give a bound for $\mathrm{HG}(s, G)$ for trees we need to generalize the definition of dominant color to suit the context of s guesses.

Definition 4.6. Let T be a tree with root r. Let $[K]$ be the set of colors and $\psi_r : [K]^n \to \mathcal{P}([K])$ be the strategy of player r. We say that a color $c \in [K]$ is *s-dominant* for (T, r, ψ_r) if the set $\{x \in [K]^n \mid c \in \psi_r(x)\}$ contains a cube of size $[K-s]^n$.

The motivation for s-dominant colors comes from the following fact:

Remark 4.7. If $K > s(s+1)$ and the player corresponding to the vertex r has at most s guesses then (T, r, ψ_r) may have at most s s-dominant colors. Indeed let $c_1, \ldots, c_{s+1}$ be s-dominant colors and let $\prod_{i=1}^{n} A_i^j$ for $j = 1, \ldots, s+1$ be the cubes provided by the definition of an s-dominant color. For each i we have

$$|\bigcap_{j=1}^{s+1} A_i^j| \geqslant K - \sum_{j=1}^{s+1} K - |A_i^j| = K - (s+1)s > 0$$

so there is a point x in the intersection of those $s+1$ cubes. Hence $\{c_1, \ldots, c_{s+1}\} \subset \psi_r(x)$, which implies that among $c_1, \ldots, c_{s+1}$ are at most s distinct colors.

Now we can prove the main theorem of this section:

Theorem 4.8. *Let T be a tree with root r. Let $[K]$ be the set of colors, s the number of guesses and ψ the strategy of the players. If $K > s(s+1)$ then for each color c which is not s-dominant there is a coloring ϕ such that $\phi(r) = c$ and none of the players guesses correctly.*

Proof. The proof is similar to the proof of Theorem 3.10 – we use induction on the size of T.

If $|T| = 1$ then ψ_r is constant and player r will not guess any color which is not dominant, i.e. does not belong to ψ_r.

For the inductive step let $T_1, \ldots, T_t$ be the trees which are the connected components of $T \setminus r$ and let $N(r) = \{r_1, \ldots, r_t\}$ be the set of corresponding roots. As in the proof of Theorem 3.10 we set $\phi(r) = c$ and obtain strategies ψ_i as restrictions of ψ to T_i. Let C_i be the set of dominant colors of (T_i, r_i, ψ_i). Since c is not dominant there is a tuple $(x_1, \ldots, x_t) \in \prod_{i=1}^{t} [K] \setminus C_i$ such that $\psi_r(x_1, \ldots, x_t) \neq c$. From the inductive hypothesis we obtain colorings ϕ_i of T_i such that $\phi_i(r_i) = x_i$ and none of the players associated to vertices in T_i guesses correctly. By the choice of $(x_1, \ldots, x_t)$ player r also guesses incorrectly. ☐

Example 4.5 shows that we cannot generalize Conjecture 3.7 in the simplest way; it is not true that $\mathrm{HG}(s, G) \leqslant s(\mathrm{col}(G) + 1)$. We therefore propose a very cautious generalization:

Conjecture 4.9. *The value* $\mathrm{HG}(s, G)$ *is bounded by a function of* s *and* $\mathrm{col}(G)$*.*

We would like to augment this conjecture with the following example:

Example 4.10. For every n there is $m_0(n)$ such that for all $m \geqslant m_0(n)$ we have $\mathrm{HG}(s, K_{n,m}) = ns^2 + s$.

Proof. This is example is a generalization of Examples 3.13 and 3.14, the proof will also be similar. Let $V(G) = L \cup R$ be the partition of vertices.

To show that $\mathrm{HG}(s, K_{n,m}) \geqslant ns^2 + s$ assume that there are $K = ns^2 + s$ colors. For simplicity we will not generalize the notion of separating sets and simply assume that the vertices in R adopt every possible guessing strategy $[K]^n \to \binom{[K]}{s}$.

We claim that for every coloring ϕ_R of R there are at most ns distinct colorings of L such that none of the vertices in R guesses correctly. Indeed, if there were at least $ns+1$ of them then there would be a strategy ψ_v that would give $s(ns+1) = ns^2 + s = K$ distinct answers for those colorings. But that would mean that for one of them ψ_v would guess correctly, a contradiction.

We let each of the n vertices in L guess accordingly to s of the at most ns colorings that do not allow vertices in R to guess correctly. So similarly as in the proof of Example 3.13 if none of the vertices in R guesses correctly then one of the vertices in L guesses correctly. So we have obtained a winning guessing strategy.

To show that $\mathrm{HG}(s, K_{n,m}) \leqslant ns^2 + s$ assume that there are $K = ns^2 + s + 1$ colors and ψ is the guessing strategy. Let $\phi_1, \ldots, \phi_{ns+1}$ be the colorings of L such that $\phi_i(v) = i$ for all $v \in L$. For any $v \in R$ the set

$$S_v = \bigcup_{i=1}^{ns+1} \psi_v(\phi_i)$$

has cardinality at most $s(ns+1) = K-1$, we set $\phi_R(v)$ to be the smallest color not in S_v. The set $\bigcup_{x \in L} \psi_v(\phi_R)$ has cardinality at most ns so there is $i \in [ns+1]$ not contained in this set. Note that for ψ the coloring (ϕ_i, ϕ_R) is a winning coloring for the adversary. □

This example together with Theorem 4.8 encourages us to propose a very bold generalization of Conjecture 3.7:

Conjecture 4.11. $\mathrm{HG}(s, G) \leqslant s^2 \operatorname{col}(G) + s$.

Obviously for large s and small $\Delta(G)$ the bound conjectured above is much worse than the one obtained in Theorem 4.2. Thus we also state the following:

Conjecture 4.12. $\mathrm{HG}(s, G) \leqslant s(\Delta(G) + 1)$.

CHAPTER 5

Summary

In this chapter we will give a summary of the most interesting results, conjectures and questions contained in the thesis.

We will begin with the relation between $\mathrm{HG}(s,G)$ and $\Delta(G)$:

Theorem. $\mathrm{HG}(s,G) \leqslant es(\Delta(G)+1)$, *moreover* $\mathrm{HG}(s,K_n) = s(\Delta(K_n)+1)$.

This is Theorem 4.2 combined with Remark 1.3. We also stated the following conjecture:

Conjecture. $\mathrm{HG}(s,G) \leqslant s(\Delta(G)+1)$.

We believe that there is a strong relation between the values $\mathrm{HG}(s,G)$ and $\mathrm{col}(G)$. We expressed it in the form of two conjectures:

Conjecture. *The value* $\mathrm{HG}(s,G)$ *is bounded by a function of* s *and* $\mathrm{col}(G)$.

Conjecture. $\mathrm{HG}(s,G) \leqslant s^2\,\mathrm{col}(G) + s$.

To support those conjectures we have proved Theorem 4.8 implying in particular that:

Theorem. *If* $\mathrm{col}(G) = 1$, *i.e.* G *is a forest then* $\mathrm{HG}(s,G) \leqslant s^2 + s$.

The conjectures are also supported by Example 4.10:

Example. For every n there is $m_0(n)$ such that for all $m \geqslant m_0(n)$ we have $\mathrm{HG}(s,K_{n,m}) = ns^2 + s$.

We introduced the notion of s-bipolar strategy and proved Theorem 4.4:

Theorem. *For every* s*-bipolar guessing strategy for a graph* G *and* $K > s(\mathrm{col}(G)+1)$ *colors the adversary has a winning hat placement.*

We repeated a question from [**BHKL**]:

Question. *Is there a bipartite graph G satisfying* $\mathrm{HG}(G) \geqslant K$ *whose size is polynomial in K?*

We introduced the notion of separating sets and obtained that there is a graph of exponential size, which is a slight improvement over the double exponential size from [**BHKL**]. This also created the question:

Question. *Is* $\mathrm{sep}(K)$ *polynomial in K?*

Bibliography

[AFGHIS] G. Aggarwal, A. Fiat, A. Goldberg, J. Hartline, N. Immorlica, M. Sudan, *Derandomization of auctions*, Proceedings of the 37th Annual ACM Symposium on Theory of Computing, 619–625, ACM, New York, 2005.

[BM] E. Brown, K. Mellinger, *Kirkmans schoolgirls wearing hats and walking through fields of numbers*, Math. Magazine 82 (2009), no. 1, 3–15.

[BGK] E. Burke, S. Gustafson, G. Kendall, *A Puzzle to challenge genetic programming*, Genetic Programming, 136–147, Lecture Notes in Computer Science, Springer, 2002.

[BHKL] S. Butler, M. Hajianghayi, R. Kleinberg, T. Leighton, *Hat guessing games*, SIAM J. Discrete Math. 22 (2008), 592–605.

[CHLL] G. Cohen, I. Honkala, S. Litsyn, A. Lobstein, *Covering Codes*, North Holland, 1997.

[Doe] B. Doerr, *Winkler's Hat Guessing Game: Better Results for Imbalanced Hat Distributions*, arXiv:1303.7205 [math.CO], 2013.

[Ebe] T. Ebert, *Applications of recursive operators to randomness and complexity*, Ph.D. Thesis, University of California at Santa Barbara, 1998.

[EMV] T. Ebert, W. Merkle, H. Vollmer, *On the autoreducibility of random sequences*, SIAM J. Comput. 32 (2003), 1542–1569.

[EL] P. Erdős, L. Lovász, *Problems and results on 3-chromatic hypergraphs and some related questions*. In: A. Hajnal, R. Rado and V.T. Sós, eds., Infinite and Finite Sets, North-Holland 2009, II, 609–627.

[Fei1] U. Feige, *You can leave your hat on (if you guess its color)*, Technical Report MCS04-03, Computer Science and Applied Mathematics, The Weizmann Institute of Science, 2004, 10 pp.

[Fei2] U. Feige, *On optimal strategies for a hat game on graphs*, SIAM J. Discrete Math. 24 (2010), 782–791.

[Gar] M. Gardner, *The 2nd Scientific American Book of Mathematical Puzzles & Diversions*, Simon and Schuster, New York, 1961.

[GKRT] W. Guo, S. Kasala, M. Rao, B. Tucker, *The hat problem and some variations*, Advances in distribution theory, order statistics, and inference, 459–479, Statistics for Industry and Technology, Birkhäuser, Boston, 2007.

[HT] C. Hardin, A. Taylor, *An introduction to infinite hat problems*, The Mathematical Intelligencer September 2008, Vol. 30, Issue 4, 20–25.

[Imm] N. Immorlica, *Computing with strategic agents*, Ph.D. Thesis, Massachusetts Institute of Technology, 2005.

[Krz1] M. Krzywkowski, *Hat problem on a graph*, Mathematica Pannonica 21 (2010), 1–19.

[Krz2] M. Krzywkowski, *Hat problem on the cycle C_4*, International Mathematical Forum 5 (2010), 205–212.

[Krz3] M. Krzywkowski, *A modified hat problem*, Commentationes Math. Vol. 50, No. 2 (2010), 121–126.

[Krz4] M. Krzywkowski, *Hat problem on odd cycles*, Houston Journal of Mathematics 37 (2011), 1063–1069.

[Krz5] M. Krzywkowski, *A more colorful hat problem*, Ann. Univ. Paedagog. Crac. Stud. Math. Vol. 10 (2011), 67–77.

[Krz6] M. Krzywkowski, *Hat problem on a graph*, Ph.D. Thesis, University of Exeter, 2012.

[LS] H. Lenstra, G. Seroussi, *On hats and other covers*, IEEE International Symposium on Information Theory, Lausanne, 2002.

[Mos] R.A. Moser, *A constructive proof of the Lovász Local Lemma*, Proceedings of the 41st Annual ACM Symposium on Theory of Computing, 343–350.

[MSY] T. Ma, X. Sun, H. Yu, *A New Variation of Hat Guessing Games*, 17th International Computing and Combinatorics Conference (COCOON), 616–626.

[Rii] S. Riis, *Information flows, graphs and their guessing number*, Electron. J. Combin. 14 (2007), Research Paper 44.

[Rob] S. Robinson, *Why mathematicians now care about their hat color*, The New York Times, Science Times Section, page D5, April 10, 2001.

[She] J. Shearer, *On a problem of Spencer*, Combinatorica 5 (1985), 241–245.

[Szcz] W. Szczechla, *The three-colour hat guessing game on the cycle graphs*, arXiv:1412.3435 [math.CO], 2014.

[Win] P. Winkler, *Games people dont play.* In: Puzzlers Tribute, D. Wolfe, T. Rodgers, eds., A.K. Peters Ltd., 2001.

[WCR] T. Wu, P. Cameron, S. Riis, *On the guessing number of shift graphs*, J. Discrete Algorithms 7 (2009), 220–226.