Int. J. Contemp. Math. Sciences, Vol. 5, 2010, no. 43, 2137 - 2148

On the Hat Problem on the Cycle C_7

Marcin Krzywkowski

Faculty of Applied Physics and Mathematics
Gdańsk University of Technology
Narutowicza 11/12, 80–952 Gdańsk, Poland
fevernova@wp.pl

Abstract

The topic is the hat problem in which each of n players is randomly fitted with a blue or red hat. Then everybody can try to guess simultaneously his own hat color by looking at the hat colors of the other players. 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 maximize the probability of a win. In this version every player can see everybody excluding himself. We consider such a problem on a graph, where vertices correspond to players, and a player can see each player to whom he is connected by an edge. The solution of the hat problem on a graph is known for trees and for cycles on four or at least nine vertices. We consider the problem on the cycle on seven vertices. We prove that if in a strategy for this graph some vertex guesses its color with probability at least one by two, then the chance of success is at most one by two.

Mathematics Subject Classification: 05C38, 05C99, 91A12

Keywords: hat problem, graph, cycle

1 Introduction

In the hat problem, a team of n players enters a room and a blue or red hat is randomly placed on the head of each player. Each player can see the hats of all of the other players but not his own. No communication of any sort is allowed, except for an initial strategy session before the game begins. Once they have had a chance to look at the other 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 maximize the probability of winning.

The hat problem with seven players, called the "seven prisoners puzzle", was formulated by T. Ebert in his Ph.D. Thesis [13]. The hat problem was also the subject of articles in The New York Times [26], Die Zeit [7], and abcNews [25]. It is also one of the Berkeley Riddles [5].

The hat problem with $2^k - 1$ players was solved in [15], and for 2^k players in [12]. The problem with n players was investigated in [8]. The hat problem and Hamming codes were the subject of [9]. The generalized hat problem with n people and q colors was investigated in [24].

There are many known variations of the hat problem (for a comprehensive list, see [22]). For example in the papers [2, 11, 19] there was considered a variation in which passing is not allowed, thus everybody has to guess his hat color. The aim is to maximize the number of correct guesses. The authors of [17] investigated several variations of the hat problem in which the aim is to design a strategy guaranteeing a desired number of correct guesses. In [18] there was considered a variation in which the probabilities of getting hats of each colors do not have to be equal. The authors of [3] investigated a problem similar to the hat problem, in that paper there are n players which have random bits on foreheads, and they have to vote on the parity of the n bits.

The hat problem and its variations have many applications and connections to different areas of science, for example: information technology [6], linear programming [17], genetic programming [10], economics [2, 19], biology [18], approximating Boolean functions [3], and autoreducibility of random sequences [4, 13-16]. Therefore, it is hoped that the hat problem on a graph considered in this paper is worth exploring as a natural generalization, and may also have many applications.

We consider the hat problem on a graph, where vertices correspond to players and a player can see each player to whom he is connected by an edge. This variation of the hat problem was first considered in [20]. There were proven some general theorems about the hat problem on a graph, and the problem was solved on trees. Additionally, there was considered the hat problem on a graph such that the only known information are degrees of vertices. In [21] the problem was solved on the cycle C_4. The problem on cycles on at least nine vertices was solved in [23].

We consider the problem on the cycle on seven vertices. We prove that if in a strategy for this graph some vertex guesses its color with probability at least one by two, then the chance of success is at most one by two.

2 Preliminaries

For a graph G, the set of vertices and the set of edges we denote by $V(G)$ and $E(G)$, respectively. Let $v \in V(G)$. The open neighborhood of v, that is $\{x \in V(G) \colon vx \in E(G)\}$, we denote by $N_G(v)$. The closed neighborhood of v, that is $N_G(v) \cup \{v\}$, we denote by $N_G[v]$.

The degree of vertex v, that is, the number of its neighbors, we denote by $d_G(v)$. Thus $d_G(v) = |N_G(v)|$. The path (cycle, respectively) with n vertices we denote by P_n (C_n, respectively).

Let $V(G) = \{v_1, v_2, \ldots, v_n\}$. By $Sc = \{1, 2\}$ we denote the set of colors, where 1 corresponds to the blue color, and 2 corresponds to the red color.

By a case for a graph G we mean a function $c\colon V(G) \to \{1, 2\}$, where $c(v_i)$ means color of vertex v_i. The set of all cases for the graph G we denote by $C(G)$, of course $|C(G)| = 2^{|V(G)|}$. If $c \in C(G)$, then to simplify notation, we write $c = c(v_1)c(v_2)\ldots c(v_n)$ instead of $c = \{(v_1, c(v_1)), (v_2, c(v_2)), \ldots, (v_n, c(v_n))\}$. For example, if a case $c \in C(C_7)$ is such that $c(v_1) = 2$, $c(v_2) = 1$, $c(v_3) = 1$, $c(v_4) = 2$, $c(v_5) = 1$, $c(v_6) = 2$, and $c(v_7) = 2$, then we write $c = 2112122$.

By a situation of a vertex v_i we mean a function $s_i\colon V(G) \to Sc \cup \{0\} = \{0, 1, 2\}$, where $s_i(v_j) \in Sc$ if v_i and v_j are adjacent, and 0 otherwise. The set of all possible situations of v_i in the graph G we denote by $St_i(G)$, of course $|St_i(G)| = 2^{d_G(v_i)}$. If $s_i \in St_i(G)$, then for simplicity of notation, we write $s_i = s_i(v_1)s_i(v_2)\ldots s_i(v_n)$ instead of $s_i = \{(v_1, s_i(v_1)), (v_2, s_i(v_2)), \ldots, (v_n, s_i(v_n))\}$. For example, if $s_3 \in St_3(C_7)$ is such that $s_3(v_2) = 2$ and $s_3(v_4) = 1$, then we write $s_3 = 0201000$.

We say that a case c for the graph G corresponds to a situation s_i of vertex v_i if $c(v_j) = s_i(v_j)$, for every v_j adjacent to v_i. This implies that a case corresponds to a situation of v_i if every vertex adjacent to v_i in that case has the same color as in that situation. Of course, to every situation of the vertex v_i correspond exactly $2^{|V(G)| - d_G(v_i)}$ cases.

By a guessing instruction of a vertex $v_i \in V(G)$ we mean a function $g_i\colon St_i(G) \to Sc \cup \{0\} = \{0, 1, 2\}$, which for a given situation gives the color v_i guesses it is, or 0 if v_i passes. Thus guessing instruction is a rule determining behavior of a vertex in every situation. We say that v_i never guesses its color if v_i passes in every situation, that is $g_i \equiv p$. We say that v_i always guesses its color if v_i guesses its color in every situation, that is, for every $s_i \in St_i(G)$ we have $g_i(s_i) \in \{1, 2\}$ ($g_i(s_i) \neq 0$, equivalently).

Let c be a case, and let s_i be the situation (of vertex v_i) corresponding to that case. The guess of v_i in the case c is correct (wrong, respectively) if $g_i(s_i) = c(v_i)$ ($0 \neq g_i(s_i) \neq c(v_i)$, respectively). By result of the case c we mean a win if at least one vertex guesses its color correctly, and no vertex guesses its color wrong, that is, $g_i(s_i) = c(v_i)$ (for some i) and there is no j such that $0 \neq g_j(s_j) \neq c(v_j)$. Otherwise the result of the case c is a loss.

By a strategy for the graph G we mean a sequence $(g_1, g_2, \ldots, g_n)$, where g_i is the guessing instruction of vertex v_i. The family of all strategies for a graph G we denote by $\mathcal{F}(G)$.

If $S \in \mathcal{F}(G)$, then the set of cases for the graph G for which the team wins using the strategy S we denote by $W(S)$. Consequently, by the chance of success of the strategy S we mean the number $p(S) = |W(S)|/|C(G)|$. By the hat number of the graph G we mean the number $h(G) = \max\{p(S): S \in \mathcal{F}(G)\}$. We say that a strategy S is optimal for the graph G if $p(S) = h(G)$. The family of all optimal strategies for the graph G we denote by $\mathcal{F}^0(G)$.

By solving the hat problem on a graph G we mean finding the number $h(G)$.

Since for every graph we can apply a strategy in which one vertex always guesses it has, let us say, the first color, and the other vertices never guess their colors, we immediately get the following lower bound on the hat number of a graph.

Fact 1 *For every graph G we have $h(G) \geq 1/2$.*

The following two results are from [20]. The first of them is a sufficient condition for deleting a vertex of a graph without changing its hat number.

Theorem 2 *Let G be a graph, and let v be a vertex of G. If there exists a strategy $S \in \mathcal{F}^0(G)$ such that v never guesses its color, then $h(G) = h(G-v)$.*

The next theorem is the solution of the hat problem on trees.

Theorem 3 *For every tree T we have $h(T) = 1/2$.*

The following results are from [23]. Let us consider strategies such that every vertex guesses its color in exactly one situation. The next lemma gives such strategy for which the number of cases in which some vertex guesses its color wrong is as small as possible.

Lemma 4 *Let us consider the family of all strategies for C_n such that every vertex guesses its color in exactly one situation. The number of cases in which some vertex guesses its color wrong is minimal for the strategy such that every vertex guesses it has the second color when its neighbors have the first color.*

If $n \geq 3$ is an integer, then let

$$A_n = \{c \in C(C_n): c(v_{i-1}) = c(v_i) = c(v_{i+1}) = 1, \text{ for some } i \in \{2, 3, \ldots, n-1\}\},$$

that is, A_n is the set of cases for C_n such that there are three vertices of the first color the indices of which are consecutive integers. Let the sequence $\{a_n\}_{n=1}^{\infty}$ be such that $a_n = |A_n|$ $(n \geq 3)$, and also $a_1 = a_2 = 0$.

Now there is a recursive formula for a_n (with $n \geq 4$).

Lemma 5 *For every integer $n \geq 4$ we have $a_n = 2^{n-3} + a_{n-3} + a_{n-2} + a_{n-1}$.*

If n is an integer such that $n \geq 3$, then let

$$B_n = \{c \in C(C_n)\colon c(v_{i-1}) = c(v_i) = c(v_{i+1}) = 1 \text{ (for some } i \in \{2, 3, \ldots, n-1\}) \text{ or } c(v_{n-1}) = c(v_n) = c(v_1) = 1 \text{ or } c(v_n) = c(v_1) = c(v_2) = 1\},$$

that is, B_n is the set of cases for C_n such that there are three consecutive vertices of the first color. Let the sequence $\{b_n\}_{n=3}^{\infty}$ be such that $b_n = |B_n|$.

We have the following relation between the number b_n (with $n \geq 6$), and the elements of the sequence $\{a_n\}_{n=1}^{\infty}$.

Lemma 6 *If $n \geq 6$ is an integer, then $b_n = 5 \cdot 2^{n-6} + a_n - 2a_{n-5} - a_{n-6}$.*

3 Results

Let us consider strategies for the hat problem on the cycle on seven vertices in which some vertex guesses its color with probability at least one by two.

Theorem 7 *If S is a strategy for the cycle C_7 such that some vertex guesses its color with probability at least one by two, then $p(S) \leq 1/2$.*

Proof. First assume that some vertex, say v_i, never guesses its color. From the proof of Theorem 2 we know that $p(S) \leq h(C_7 - v_i)$. Since $C_7 - v_i = P_6$ and $h(P_6) = 1/2$ (by Theorem 3), we get $p(S) \leq 1/2$. Now assume that every vertex guesses its color, that is, every vertex guesses its color in at least one situation. Without loss of generality we assume that v_4 guesses its color with probability at least one by two. Every vertex of the cycle has exactly two neighbors, thus there are exactly $2^2 = 4$ possible situations of each one of them. Therefore guessing with probability at least one by two means guessing in at least two situations. We are interested in the possibility when the number of cases for which the team loses is as small as possible. We assume that v_4 guesses its color in exactly two situations, and every one of the remaining vertices guesses its color in exactly one situation. We prove that these guesses suffice to cause the loss of the team in at least half of all cases. Let v_i be any vertex of C_7. Any guess made by v_i in any situation is wrong in exactly $2^{7-3} = 2^4 = 16$ cases. We want to minimize the number of cases in which some vertex guesses its color wrong. Therefore we want the number of cases in which v_i guesses its color wrong, and at the same time also another vertex guesses its color wrong to be as great as possible. We distinguish between the following four possibilities about the behavior of the vertex v_4: (v_4:1) in both situations v_4 guesses it has the same color, and in both of them v_3 has different colors, and in both of them v_5 has different colors; (v_4:2) in both situations v_4

guesses it has the same color, and in both of them some neighbor of v_4 has the same color; (v_4:3) in both situations v_4 guesses it has different colors, in both of them v_3 has different colors, and in both of them v_5 has different colors; (v_4:4) in both situations v_4 guesses it has different colors, and in both of them some neighbor of v_4 has the same color;

(v_4:1) Without loss of generality we assume that v_4 guesses it has the second color in the situations 0010100 and 0020200. From Lemma 4 we know that if every vertex guesses its color in exactly one situation, then we may assume that every vertex guesses it has the second color when both its neighbors have the first color. Since $N_{C_7}[v_1] \cap N_{C_7}[v_4] = \emptyset$ and $N_{C_7}[v_7] \cap N_{C_7}[v_4] = \emptyset$, we may assume that the vertices v_1 and v_7 guess their colors when their neighbors have the first color, and then they guess they have the second color. Moreover, since $v_1, v_2 \notin N_{C_7}[v_4]$, we may assume that v_2 guesses its color in a situation in which v_1 has the first color, and in this situation it guesses it has the second color. In one situation in which v_4 guesses its color the vertex v_3 has the first color, and in the another it has the second color. Thus independently from that which color has the vertex v_3 in the situation in which v_2 guesses its color, the intersection of the set of cases in which v_2 guesses its color wrong with the set of cases in which v_4 guesses its color wrong has the same cardinality. Without loss of generality we assume that in the situation in which v_2 guesses its color the vertex v_3 has the first color. Thus v_2 guesses it has the second color in the situation 1010000. Similarly we may assume that v_6 guesses it has the second color in the situation 0000101. Now let us consider the behavior of the vertex v_3. Since $v_2 \notin N_{C_7}[v_4]$, we may assume that in the situation in which v_3 guesses its color the vertex v_2 has the first color. In one situation in which v_4 guesses its color v_3 has the first color, and in the another it has the second color. Thus independently from that which color v_3 guesses, the intersection of the set of cases in which v_3 guesses its color wrong with the set of cases in which v_4 guesses its color wrong has the same cardinality. Since v_2 guesses its color when v_3 has the first color, we may assume that v_3 guesses its color wrong when it has the first color, that is, v_3 guesses the second color. Similarly we can assume that v_6 guesses its color in a situation in which v_5 has the first color, and then it guesses it has the second color. Now it remains to analyze which color has v_4 when v_3 guesses its color, and which color it has when v_5 guesses its color. Since in every case in which v_4 guesses its color wrong it has the first color (as it guesses it has the second color), we may assume that v_3 and v_5 guess their colors when v_4 has the first color. Now we conclude that every vertex excluding v_4 guesses it has the second color when both its neighbors have the first color.

Now let us count the cases in which some vertex guesses its color wrong. First let us consider the set of cases in which some vertex other than v_4 guesses its color wrong, or v_4 guesses its color wrong in the situation 0010100. This set consists of cases such that there are three consecutive vertices of the first color. By the definition of the sequence $\{b_n\}_{n=3}^{\infty}$, there are b_7 such cases. Now let us consider the set of cases in which v_4 guesses its color wrong in the situation 0020200, while at the same time no other vertex guesses its color wrong. Thus this set consists of the cases such that $c(v_3) = 2$, $c(v_4) = 1$, and $c(v_5) = 2$, while at the same time there are no three consecutive vertices from the set $\{v_1, v_2, v_6, v_7\}$ which have the first color. It follows from the definition of the sequence $\{a_n\}_{n=1}^{\infty}$ that this set has $2^4 - a_4 = 16 - a_4$ elements. Now we conclude that the number of cases in which some vertex guesses its color wrong is equal to $b_7 + 16 - a_4$. Using Lemma 5, the definition of the sequence $\{a_n\}_{n=1}^{\infty}$, and the fact that $a_3 = 1$ (as 111 is the only one such case), we get

$$\begin{aligned}
a_1 &= 0,\\
a_2 &= 0,\\
a_3 &= 1,\\
a_4 &= 2 + a_1 + a_2 + a_3 = 2 + 0 + 0 + 1 = 3,\\
a_5 &= 2^2 + a_2 + a_3 + a_4 = 4 + 0 + 1 + 3 = 8,\\
a_6 &= 2^3 + a_3 + a_4 + a_5 = 8 + 1 + 3 + 8 = 20,\\
a_7 &= 2^4 + a_4 + a_5 + a_6 = 16 + 3 + 8 + 20 = 47.
\end{aligned}$$

By Lemma 6 we get

$$\begin{aligned}
b_7 &= 5 \cdot 2 + a_7 - 2a_2 - a_1\\
&= 10 + 47 - 2 \cdot 0 - 0\\
&= 57.
\end{aligned}$$

Now we get $b_7 + 16 - a_4 = 57 + 16 - 3 = 70$. This implies that the team wins for at most 58 cases. Consequently, $p(S) \le 58/128 < 64/128 = 1/2$.

(v_4:2) Without loss of generality we assume that v_4 guesses it has the second color in the situations 0010100 and 0010200. The only one difference about the behavior of v_4 comparing to the possibility (v_4:1) is that now v_4 guesses it has the second color in the situation 0010200 instead of 0020200. The only two vertices besides v_4 the closed neighborhood of which contains the vertex v_3 are v_2 and v_3. Since in every case in which v_3 has the first color the vertex v_4 guesses its color, and in no case in which v_3 has the second color the vertex v_4 guesses its color, we may assume that v_2 guesses its color in a situation in which v_3 has the first color. Thus we assume that v_2 guesses it has the second color in the situation 1010000. In every case in which v_3 has the first color the vertex v_4 guesses its color, and in no case in which v_3 has the second color the vertex v_4 guesses its color. Moreover, in some cases in which v_3 has the first color the vertex v_2 guesses its color, and in no case in which v_3 has the second color the vertex v_2 guesses its color. Therefore we may assume that v_3 guesses its color wrong when it has the first color. Thus v_3 guesses it has the second color in the situation 0101000.

Now let us count the cases in which some vertex guesses its color wrong. First let us consider the set of cases in which some vertex other than v_4 guesses its color wrong, or v_4 guesses its color wrong in the situation 0010100. In the same way as in (v_4:1) we conclude that this set has b_7 elements. Now let us consider the set of cases in which v_4 guesses its color wrong in the situation 0010200, while at the same time no other vertex guesses its color wrong. Let us consider any case which belongs to this set. Since v_4 guesses its color wrong, we have $c(v_3) = c(v_4) = 1$ and $c(v_5) = 2$. Only v_4 guesses its color wrong, so particularly v_3 does not guess its color wrong. Therefore $c(v_2) \neq 1$, that is $c(v_2) = 2$. To avoid a wrong guess of v_7 we cannot have $c(v_6) = c(v_7) = c(v_1) = 1$. Thus $c(v_6) = 2$ or $c(v_7) = 2$ or $c(v_1) = 2$. Then the only one vertex which guesses its color wrong is v_4. Let us observe that the considered set has 7 elements as there are seven possible colorings of the vertices v_1, v_6, and v_7 excluding this in which all these vertices have the first color. Now we conclude that some vertex guesses its color wrong in $b_7 + 7 = 57 + 7 = 64$ cases. Thus the team wins for at most 64 cases. Consequently, $p(S) \leq 64/128 = 1/2$.

(v_4:3) Without loss of generality we assume that in the situation 0010100 the vertex v_4 guesses it has the second color, and in the situation 0020200 it guesses it has the first color. The only one difference between the behavior of v_4 comparing to the possibility (v_1:1) is that in the situation 0020200 it guesses it has the first color instead of the second color. The only two vertices, besides v_4, the closed neighborhood of which contain the vertex v_4 are v_3 and v_5. Similarly as in the possibility (v_4:1) we assume that in some situation v_3 guesses it has the second color. Since in some cases in which v_3 and v_4 have the first color the vertex v_4 guesses its color wrong, we may assume that v_3 guesses its color in a situation in which v_4 has the first color. Similarly as in the possibility (v_4:1) we assume that v_4 guesses it has the second color in the situation 0101000. Similarly we can assume that v_5 guesses it has the second color in the situation 0001010.

Now let us count the cases in which some vertex guesses its color wrong. First let us consider the set of cases in which some vertex other than v_4 guesses its color wrong, or v_4 guesses its color wrong in the situation 0010100. In the same way as in the previous possibilities we get that this set has b_7 elements. Now let us consider the set of cases in which v_4 guesses its color wrong in the situation 0020200, while at the same time no other vertex guesses its color wrong. Let us consider any case which belongs to this set. Since v_4 guesses its color wrong, we have $c(v_3) = c(v_4) = c(v_5) = 2$. Thus no one of the vertices v_2, v_3, v_5, and v_6 guesses its color wrong. To avoid that v_1 or v_7 guesses its color wrong, we cannot have three consecutive vertices from the set $\{v_1, v_2, v_6, v_7\}$ which have the first color. In the same way as in the possibility (v_4:1) we get that there are $16 - a_4$ such cases. Now we conclude that in $b_7 + 16 - a_4 = 70$

cases some vertex guesses its color wrong. Thus the team wins for at most 58 cases. Consequently, $p(S) \leq 58/128 < 64/128 = 1/2$, a contradiction.

(v_4:4) Without loss of generality we assume that in the situation 0010100 the vertex v_4 guesses it has the second color, and in the situation 0010200 it guesses it has the first color. The only one difference between the behavior of v_4 comparing to the possibility (v_4:2) is that in the situation 0010200 the vertex v_4 guesses it has the first color instead of the second color. The only two vertices besides v_4, the closed neighborhood of which contain the vertex v_4 are v_3 and v_5. Similarly as in the possibility (v_4:2) we assume that the vertex v_3 guesses its has the second color in the situation 0101000, and that in some situation the vertex v_5 guesses it has the second color. Thus in every case in which v_5 guesses its color wrong it has the first color. The vertex v_4 guesses its color wrong in some cases in which both vertices v_4 and v_5 have the first color, and in some cases in which both vertices v_4 and v_5 have the second color. The vertex v_3 guesses its color in a situation in which v_4 has the first color. Therefore we may assume that v_5 guesses its color in a situation in which v_4 has the first color. Similarly as in the possibility (v_4:2) we assume that v_5 guesses it has the second color in the situation 0001010.

Now let us count the cases in which some vertex guesses its color wrong. First let us consider the set of cases in which some vertex other than v_4 guesses its color wrong, or v_4 guesses its color wrong in the situation 0010100. Similarly as in the previous possibilities we get that this set has b_7 elements. Now let us consider the set of cases in which v_4 guesses its color wrong in the situation 0010200, while at the same time no other vertex guesses its color wrong. Let us consider any case which belongs to this set. Since v_4 guesses its color wrong, we have $c(v_3) = 1$ and $c(v_4) = c(v_5) = 2$. If $c(v_2) = 1$, then $c(v_1) = 2$, otherwise v_2 guesses its color wrong. There are 4 such cases as there are four possible colorings of the vertices v_6 and v_7. Then only v_4 guesses its color wrong. Now assume that $c(v_2) = 2$. To avoid that v_7 guesses its color wrong, we cannot have $c(v_6) = c(v_7) = c(v_1) = 1$. Thus $c(v_6) = 2$ or $c(v_7) = 2$ or $c(v_1) = 2$. There are 7 such cases. Then only v_4 guesses its color wrong. Now we conclude that some vertex guesses its color wrong in $b_7 + 7 + 4 = 68$ cases. Thus the team wins for at most 60 cases. Consequently, $p(S) \leq 60/128 < 64/128 = 1/2$, a contradiction. ■

Corollary 8 *If $h(C_7) > 1/2$, then $h(C_7) = \max\{p(S) \colon S \in \mathcal{F}(C_7) \colon$ every vertex guesses its color in exactly one situation$\}$.*

Proof. By Fact 1 we have $h(C_7) \geq 1/2$. Let S_1 be an optimal strategy for the graph C_7. If some vertex, say v_i, never guesses its color, then by Theorem 2 we have $h(C_7) = h(C_7 - v_i) = h(P_6) = 1/2$. Since by Theorem 3 we have $h(P_6) = 1/2$, we get $h(C_7) = 1/2$, a contradiction to the assumption that $h(C_7) > 1/2$. Henceforth, in the strategy S_1 every vertex guesses its color in some situation. If some vertex guesses its color in at least two situations, then by Theorem 7 we have $p(S) \leq 1/2$. By the definition of an optimal strategy we get $h(C_7) = p(S_1) \leq 1/2$. This is a contradiction to the assumption that $h(C_7) > 1/2$. Thus in the strategy S_1 every vertex guesses its color in exactly one situation. This implies that $S_1 \in \{S \in \mathcal{F}(C_7)\colon$ every vertex guesses its color in exactly one situation$\}$. Therefore $p(S_1) \leq \max\{p(S)\colon S \in \mathcal{F}(C_7)\colon$ every vertex guesses its color in exactly one situation$\}$. By the definition of an optimal strategy we get $h(C_7) = p(S) \leq \max\{p(S)\colon S \in \mathcal{F}(C_7)\colon$ every vertex guesses its color in exactly one situation$\}$.

On the other hand, by definition we have $h(C_7) = \max\{p(S)\colon S \in \mathcal{F}(C_7)\}$. Since the set $\{S \in \mathcal{F}(C_7)\colon$ every vertex guesses its color in exactly one situation $\}$ is a subset of $\mathcal{F}(C_7)$, we have $\max\{p(S)\colon S \in \mathcal{F}(C_7)\colon$ every vertex guesses its color in exactly one situation$\} \leq \max\{p(S)\colon S \in \mathcal{F}(C_7)\}$. ■

References

[1] N. Alon, *Problems and results in extremal combinatorics – II,* Discrete Mathematics 308 (2008), 4460–4472.

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

[3] J. Aspnes, R. Beigel, M. Furst, and S. Rudich, *The expressive power of voting polynomials,* Combinatorica 14 (1994), 135–148.

[4] R. Beigel, L. Fortnow, and F. Stephan, *Infinitely-often autoreducible sets,* SIAM Journal on Computing 36 (2006), 595–608.

[5] *Berkeley Riddles,* www.ocf.berkeley.edu/ wwu/riddles/hard.shtml.

[6] M. Bernstein, *The hat problem and Hamming codes,* MAA Focus, November, 2001, 4–6.

[7] W. Blum, *Denksport für Hutträger,* Die Zeit, May 3, 2001.

[8] M. Breit, D. Deshommes, and A. Falden, *Hats required: perfect and imperfect strategies for the hat problem,* manuscript.

[9] E. Brown, K. Mellinger, *Kirkman's schoolgirls wearing hats and walking through fields of numbers,* Mathematics Magazine 82 (2009), no. 1, 3–15.

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

[11] S. Butler, M. Hajianghayi, R. Kleinberg, and T. Leighton, *Hat guessing games,* SIAM Journal on Discrete Mathematics 22 (2008), 592–605.

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

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

[14] T. Ebert and W. Merkle, *Autoreducibility of random sets: a sharp bound on the density of guessed bits,* Mathematical foundations of computer science 2002, 221–233, Lecture Notes in Computer Science, 2420, Springer, Berlin, 2002.

[15] T. Ebert, W. Merkle, and H. Vollmer, *On the autoreducibility of random sequences,* SIAM Journal on Computing 32 (2003), 1542–1569.

[16] T. Ebert and H. Vollmer, *On the autoreducibility of random sequences,* Mathematical foundations of computer science 2000 (Bratislava), 333–342, Lecture Notes in Computer Science, 1893, Springer, Berlin, 2000.

[17] 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.

[18] W. Guo, S. Kasala, M. Rao, and 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.

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

[20] M. Krzywkowski, *Hat problem on a graph,* Mathematica Pannonica 21 (2010), 3–21.

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

[22] M. Krzywkowski, *On the hat problem, its variations, and their applications,* Annales Universitatis Paedagogicae Cracoviensis Studia Mathematica 9 (2010), 55–67.

[23] M. Krzywkowski, *The hat problem on cycles with at least nine vertices,* submitted.

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

[25] J. Poulos, *Could you solve this $1 million hat trick?,* abcNews, November 29, 2001.

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

Received: May, 2010