The number of pessimistic guesses in Generalized Black-peg Mastermind

Gerold Jäger[a], Marcin Peczarski[b]

[a] Institute for Applied Stochastics and Operations Research, Technical University of Clausthal, D-38678 Clausthal, Germany
[b] Institute of Informatics, University of Warsaw, ul. Banacha 2, PL-02-097 Warszawa, Poland

Abstract

Mastermind is a famous two-player game, where the codemaker has to choose a secret code and the codebreaker has to guess it in as few questions as possible using information he receives from the codemaker after each guess. In Generalized Black-peg Mastermind for given arbitrary numbers p, c, the secret code consists of p pegs each having one of c colors, and the received information consists only of a number of black pegs, where this number equals the number of pegs matching in the corresponding question and the secret code. Let $b(p,c)$ be the pessimistic number of questions for Generalized Black-peg Mastermind. By a computer program we compute several values $b(p,c)$. By introducing some auxiliary games and combining this program with theoretical methods, for arbitrary c we obtain exact formulas for $b(2,c)$, $b(3,c)$ and $b(4,c)$ and give upper and lower bounds for $b(5,c)$ and a lower bound for $b(6,c)$. Furthermore, for arbitrary p, we present upper bounds for $b(p,2)$, $b(p,3)$ and $b(p,4)$. Finally, we give bounds for the general case $b(p,c)$. In particular, we improve an upper bound recently proved by Goodrich.

Keywords: Combinatorial problems, Algorithms, Mastermind, Logic game, Computer aided proof

1. Introduction

Mastermind is a two-player game invented by Mordecai Meirowitz in 1970. The first player, called *codemaker*, chooses a secret code. The secret consists of 4 pegs, each of which in one of 6 colors. The second player, called *codebreaker*, asks questions to guess the secret. A question also consists of 4 pegs in 6 colors. Each question is answered by the codemaker with black and white pegs. A black peg means that one peg of the guess is correct in position and color, but does not inform which one. A white peg means that one peg of the guess is correct only in color, but does not inform which one as well. The game ends when the codebreaker receives the answer containing 4 black pegs. The goal of the codebreaker is to discover the secret in as few questions as possible.

Mastermind is widely considered in the algorithmic literature. Miscellaneous approaches have been proposed to solve the game. Whereas most work uses classic combinatorial methods [2, 4, 5, 6, 8, 13, 14, 18], in [1, 12, 16] evolutionary or genetic algorithms are proposed. The $\mathcal{NP}$-completeness of the Mastermind game was proved in [17]. *Static Mastermind* [6, 7, 10] and *AB Game* [3] are further variants of this game.

In [11] we considered *Generalized Mastermind*, further denoted by $G(p,c)$, where an arbitrary number p of pegs and an arbitrary number c of colors are given. Goodrich considered in [9] a version of Generalized Mastermind, where whites pegs are ignored in answers, i.e., only black pegs are allowed. We call this game *Generalized Black-peg Mastermind* and denote it by $GB(p,c)$. Goodrich [9] proved that the game $GB(p,c)$ is also $\mathcal{NP}$-complete and gave the following upper bound for the pessimistic number of questions

$$p\lceil \log_2 c\rceil + \lceil (2 - 1/c)p\rceil + c. \qquad (1)$$

In this paper we apply the approach from [11] to solve the game $GB(p,c)$. The paper is organized as follows. Section 2 introduces two new auxiliary games used extensively throughout the paper. Section 3 contains auxiliary results needed to prove the results in the subsequent sections. Section 4 presents values obtained by computer search. Section 5 presents how to obtain tight lower and upper bounds for a fixed number of pegs. In particular, we completely solve the cases of 2, 3 and 4 pegs, give tight lower bounds for 5 and 6 pegs, and a tight upper bound for 5 pegs. Section 6 presents upper bounds for a fixed number of colors $c \leq 4$. In Section 8

Email addresses: gerold.jaeger@tu-clausthal.de (Gerold Jäger), marpe@mimuw.edu.pl (Marcin Peczarski)

we improve the upper bound (1) of Goodrich [9] and we prove lower bounds for an arbitrary number of pegs and colors. The paper closes with some open problems in Section 9.

2. Preliminaries

Let $f(p, c)$ be the pessimistic number of questions in the game $G(p, c)$, and $b(p, c)$ the pessimistic number of questions in the game $GB(p, c)$. To prove lower and upper bounds, we introduce two new auxiliary games.

Let $b_*(p, c)$ be the pessimistic number of questions in a game $GB_*(p, c)$, in which (besides the *real* colors) we can use one additional *filler* color in questions, i.e., totally $c + 1$ colors, but this filler color does not appear in the secret.

Let $b^*(p, c)$ be the pessimistic number of questions in a game $GB^*(p, c)$, when we begin with $c - 1$ questions of the form $xxx \dots x$, each with different color x.

As each strategy for $GB(p, c)$ is also a strategy for $GB_*(p, c)$, and as each strategy for $GB^*(p, c)$ is also a strategy for $GB(p, c)$, the following lemma is obvious.

Lemma 1. *For $p \geq 1$ and $c \geq 1$ it holds*

$$b_*(p, c) \quad \leq \quad b(p, c) \quad \leq \quad b^*(p, c).$$

Note that Lemma 1 justifies the introduction of the games $GB_*(p, c)$ and $GB^*(p, c)$.

3. Auxiliary results

At the beginning, we prove some results for $GB(p, c)$, $GB_*(p, c)$ and $GB^*(p, c)$, which we will extensively use in the rest of the paper.

It can be shown that all real colors are equivalent in the first question. For our purpose, it suffices to prove the following weaker result that we can freely choose only one real color in the first question.

Lemma 2. *Let $p \geq 1$, $c \geq 1$, and $x_1, x_2, \dots, x_c$ be the real colors of the games $GB(p, c)$ and $GB_*(p, c)$. Furthermore denote with y the filler color of the game $GB_*(p, c)$, and let an arbitrary $x \in \{x_1, x_2, \dots, x_c\}$ be given. Then it holds:*

(a) If there is a strategy for $GB(p, c)$ using at most q questions, then there is also a strategy using at most q questions and starting with the question $xxx \dots x$.

(b) If there is a strategy for $GB_*(p, c)$ using at most q questions, then there is also a strategy using at most q questions and starting with a question $z_1 z_2 z_3 \dots z_p$, where $z_i \in \{x, y\}$ for $i = 1, 2, \dots, p$.

Proof. Let an arbitrary strategy S for the game $GB(p, c)$ or $GB_*(p, c)$ be given using at most q questions. Let the first question of S be $z_1 z_2 \dots z_p$, and let T be the set of all possible questions in $GB(p, c)$ or $GB_*(p, c)$, respectively. Consider the function $\pi : T \to T$ defined as follows. For $1 \leq i \leq p$, if $z_i \neq x$ and $z_i \neq y$, π exchanges at peg i the colors z_i and x.

Observe that if X is a secret, i.e., contains only colors x_1, x_2, $\dots$, x_c, then $\pi(X)$ is also a secret. Now let $\pi(S)$ denote the strategy, where we apply π to all questions in S.

Let X be a secret. If S asks a question Q and receives B blacks, then $\pi(S)$ asking $\pi(Q)$ about the secret $\pi(X)$ receives also B blacks. Hence, if S chooses the next question Q', then $\pi(S)$ chooses $\pi(Q')$. Thus for both, $GB(p, c)$ or $GB_*(p, c)$, it holds that if S solves a secret X in q questions, then $\pi(S)$ solves the secret $\pi(X)$ in q questions. As for every secret Y a secret X exists with $Y = \pi(X)$, the strategy $\pi(S)$ solves the game $GB(p, c)$ or $GB_*(p, c)$ using pessimistically also q questions. □

Lemma 3. *For $p \geq 1$ and $c \geq 1$ it holds*

$$b_*(p, c) + 1 \quad \leq \quad b_*(p, c + 1).$$

Proof. Consider an arbitrary strategy S for $GB_*(p, c+1)$ with $q = b_*(p, c + 1)$ questions. Let x_1, x_2, $\dots$, x_{c+1} be the colors used in the secret, and let x_{c+2} be the filler color used only in questions. By Lemma 2(b), w.l.o.g. we can assume that the first question of S has only pegs of colors x_{c+1} and x_{c+2}.

Obviously, the strategy S works also for p pegs and c colors, where the colors used in the secret are x_1, x_2, $\dots$, x_c. In this case the first answer is always 0 blacks. Hence, we do not need to ask it and we can start the game from the second question. Therefore, to complete the game we need at most $q - 1$ questions. Moreover, as the colors x_{c+1} and x_{c+2} are not used in the secret, we can replace x_{c+2} by x_{c+1} in all questions. We obtain a strategy for $GB_*(p, c)$ using at most $q - 1$ questions. We conclude that $b_*(p, c) \leq b_*(p, c + 1) - 1$. □

Lemma 4. *For $p \geq 1$ and $c_1 \geq c_0 \geq 1$ it holds*

$$b(p, c_1) \quad \geq \quad b_*(p, c_0) + (c_1 - c_0).$$

Proof. This follows from Lemma 1 and Lemma 3 by induction on $c_1 - c_0$. □

Lemma 5. *For $c_1, c_0 > p \geq 1$ it holds*

$$b^*(p, c_1) \quad = \quad b^*(p, c_0) + (c_1 - c_0).$$

Proof. Observe that the state, when the first $c_0 - 1$ questions have been asked in the game $GB^*(p, c_0)$, and the state, when the first $c_1 - 1$ questions have been asked in the game $GB^*(p, c_1)$, are the same, because in both cases we know the number of each color in the secret. At most p different colors are possible and we can use a filler color in the rest of the game. It follows

$$b^*(p, c_1) - (c_1 - 1) \quad = \quad b^*(p, c_0) - (c_0 - 1)$$

which is equivalent to the assertion of Lemma 5. □

Lemma 6. *For $c_1 \geq c_0 \geq p \geq 1$ it holds*

$$b(p, c_1) \quad \leq \quad b^*(p, c_0) + (c_1 - c_0).$$

Proof. The result holds for $c_1 = c_0$ and $c_1 > c_0 > p$ by Lemma 1 and Lemma 5. For $c_1 > c_0 = p$ we claim that

$$b^*(p, c_1) - (c_1 - 1) \quad \leq \quad b^*(p, c_0) - (c_0 - 1).$$

The proof is the same as for Lemma 5 with the only difference as follows. Whereas on the left side after the first $c_1 - 1$ questions we have a filler color unused in the secret which we can use, on the right side we have not such a filler color. Finally, by Lemma 1, it holds $b(p, c_1) \leq b^*(p, c_1)$. □

We close this section with an interesting result. An analogue theorem for $f(p, c)$ is still an open question.

Theorem 7. *For $p \geq 1$ and $c \geq 1$ it holds*

$$b(p, c) \quad \leq \quad b(p + 1, c).$$

Proof. Consider a strategy for $p + 1$ pegs and c colors. We want to use it for p pegs. Let the secret be $x_1x_2x_3 \ldots x_p$. For our $p + 1$ pegs strategy we consider the secret $x_1x_2x_3 \ldots x_pz$, where z is a known fixed color. Now if the strategy requires to ask the question $y_1y_2y_3 \ldots y_py_{p+1}$, we ask the question $y_1y_2y_3 \ldots y_p$. Assume we have received the answer B blacks. If $y_{p+1} = z$, we change the answer to $B + 1$ blacks. If $y_{p+1} \neq z$, we proceed with B blacks. Then the next question for the p pegs strategy comes from the $(p + 1)$-pegs strategy. □

4. Computed values

We have adapted the program developed in [11] for $f(p, c)$ to compute values $b(p, c)$, $b_*(p, c)$ and $b^*(p, c)$. Again this program is based on the *nauty* package for generating families of graphs without isomorphisms [15, 19]. By Lemma 2 and by permuting the peg positions up to isomorphism, we need to consider only one first question in $GB(p, c)$ and only $p + 1$ different first questions in $GB^*(p, c)$. The computation of all presented values needed a few hours on a Core 2 Duo 2.13 GHz or a Core i5 2.4 GHz processor. The source code of our program is publicly available [20].

Table 1: Computed values $b(p, c)$ for $p \leq 8$ and $c \leq 10$

		c									
		1	2	3	4	5	6	7	8	9	10
p	1	1	2	3	4	5	6	7	8	9	10
	2	1	3	4	5	6	7	8	9	10	11
	3	1	4	5	6	7	8	**9**	**10**	**11**	**12**
	4	1	5	5	6	8	**9**	**10**	**11**	**12**	**13**
	5	1	5	6	7	≤ 9					
	6	1	6	6	8	≤ 10					
	7	1	6	7							
	8	1	7	7							

Table 2: Computed values $b_*(p, c)$ for $p \leq 8$ and $c \leq 10$

		c									
		1	2	3	4	5	6	7	8	9	10
p	1	1	2	3	4	5	6	7	8	9	10
	2	1	3	4	5	6	7	8	9	10	11
	3	1	3	5	6	7	8	**9**	**10**	**11**	**12**
	4	1	4	5	6	8	**9**	**10**	**11**	**12**	**13**
	5	1	4	6	7						
	6	1	5	6							
	7	1	5	7							
	8	1	5	7							

The values $b(p, c)$, $b_*(p, c)$, $b^*(p, c)$ are presented in Tables 1, 2, and 3, respectively. Note that the values preceded by "≤" are only upper bounds. They are found by restricting the program to search for the Knuth–Greedy strategy [13] only, i.e., we always choose a question minimizing the maximum number of possible secrets.

Note that the value $b_*(4, 5)$ has been received as follows. The computer program returned that $b_*(4, 5) > 7$. From Lemma 1 it follows $b_*(4, 5) \leq b(4, 5) = 8$, which means that $b_*(4, 5) = 8$.

Similarly, the values $b_*(8, 3)$ and $b(8, 3)$ have been received as follows. By Theorem 7 and by Table 1, we conclude $b(8, 3) \geq b(7, 3) = 7$. By Lemma 1 and by Table 3 it follows $b_*(8, 3) \leq b(8, 3) \leq b^*(8, 3) = 7$. As the computer program returned that $b_*(8, 3) > 6$, we received $b_*(8, 3) = b(8, 3) = 7$.

The additional values found by theory are in bold face. The values in Table 1 have been obtained using

Table 3: Computed values $b^*(p, c)$ for $p \leq 8$ and $c \leq 10$

		c									
		1	2	3	4	5	6	7	8	9	10
p	1	1	2	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
	2	1	3	4	**5**	**6**	**7**	**8**	**9**	**10**	**11**
	3	1	4	5	6	**7**	**8**	**9**	**10**	**11**	**12**
	4	1	5	5	7	8	**9**	**10**	**11**	**12**	**13**
	5	1	5	6	8	9	10	**11**	**12**	**13**	**14**
	6	1	6	7	8	10					
	7	1	6	7	9						
	8	1	7	7							

Theorem 8(h) (see Section 5). The values in Table 2 have been obtained using Lemma 1, Lemma 3 and Table 1, and the values in Table 3 using Lemma 5.

5. Fixed number of pegs

In this section we present results for a fixed number of pegs and an arbitrary number of colors.

Theorem 8. *It holds:*

(a) $b(1, c) \geq c$ *for* $c \geq 1$,

(b) $b(2, c) \geq c + 1$ *for* $c \geq 2$,

(c) $b(3, c) \geq c + 2$ *for* $c \geq 2$,

(d) $b(4, c) \geq c + 3$ *for* $c \geq 5$,

(e) $b(5, c) \geq c + 3$ *for* $c \geq 2$,

(f) $b(6, c) \geq c + 3$ *for* $c \geq 2$,

(g) $b(p, c) \leq c + p - 1$ *for* $1 \leq p \leq 5$ *and* $c \geq 1$,

(h) $b(p, c) = c + p - 1$ *for* $1 \leq p \leq 4$ *and* $c > p$.

Proof.

Case (a)
We set $c_0 = 1$ and $c_1 = c$ in Lemma 4 and we use the value $b_*(1, 1) = 1$ from Table 2.

Case (b)
We set $c_0 = 2$ and $c_1 = c$ in Lemma 4 and we use the value $b_*(2, 2) = 3$ from Table 2.

Case (c)
For $c \geq 3$ we set $c_0 = 3$ and $c_1 = c$ in Lemma 4 and we use the value $b_*(3, 3) = 5$ from Table 2. For $c = 2$ we use the value $b(3, 2) = 4$ from Table 1.

Case (d)
We set $c_0 = 5$ and $c_1 = c$ in Lemma 4 and we use the value $b_*(4, 5) = 8$ from Table 2.

Case (e)
For $c \geq 3$ we set $c_0 = 3$ and $c_1 = c$ in Lemma 4 and we use the value $b_*(5, 3) = 6$ from Table 2. For $c = 2$ we use the value $b(5, 2) = 5$ from Table 1.

Case (f)
We set $c_0 = 2$ and $c_1 = c$ in Lemma 4 and we use the value $b_*(6, 2) = 5$ from Table 2.

Case (g)
If $c \geq p$, this follows from Lemma 6 for $c_0 = p$ and $c_1 = c$, where we use that it holds by Table 3:

$$b^*(p, p) = 2p - 1$$

for $1 \leq p \leq 5$. For $c < p$ all needed values appear in Table 1.

Case (h)
This claim follows directly from (a), (b), (c), (d) and (g). □

Note that the method presented in this section would also be applicable to obtain tight bounds or even exact formulas for $b(p, c)$ with $p > 4$, if we would be able to compute values $b_*(p, c)$ for $c \leq p + 1$ and $b^*(p, p)$.

6. One, two or three colors

In this section we present results for at most three colors and an arbitrary number of pegs. We postpone the case of four colors to Section 7. If we compare the results for the game $G(p, c)$ in [11] with the results for the game $GB(p, c)$ in this section, we observe that white pegs in answers are important, in particular when the number of colors increases. The following theorem states that white pegs are not very helpful for two colors.

Theorem 9. *For* $p \geq 1$ *it holds*

$$f(p, 2) \leq b(p, 2) \leq f(p, 2) + 1.$$

Proof. The left inequality follows from the observation that each strategy for $GB(p, c)$ is also a strategy for $G(p, c)$.

For proving the right inequality let an optimal strategy for $G(p, 2)$ be given with $f(p, 2)$ questions in the worst case. We change this strategy to a strategy for $GB(p, 2)$ with $f(p, 2) + 1$ questions in the worst case. The only difference is that we start with the question $xxx \ldots xx$, and then we ask the questions of the optimal strategy for $G(p, 2)$. To use the strategy for $G(p, 2)$

we need to know the number W of white pegs in answers. But after the first question the number W is uniquely determined by the number B of black pegs as $W = p - B - |k - k'|$, where k, k' is the number of x in the secret and in the considered question, respectively (see [11, eq. (8)]). Thus this strategy solves $GB(p, 2)$ in at most $f(p, 2) + 1$ questions. □

Theorem 10. *For $p \geq 1$ and $1 \leq c \leq 3$ it holds*

$$b(p, c) \quad \leq \quad c + p - 1.$$

Proof. We consider each number of colors separately.

Case $c = 1$

This case is clear, as $b(p, 1) = 1 \leq 1 + p - 1$.

Case $c = 2$

Let the colors be x, y. First, we ask $xxx \dots xx$. We receive k_0 blacks. Next we ask the $p - 1$ questions: $yxx \dots xx$, $xyx \dots xx$, $xxyx \dots x$, $\dots$, $xx \dots xyx$. We receive $k_1, k_2, \dots, k_{p-1}$ blacks, respectively. For each $1 \leq i \leq p - 1$ we consider the following two cases. If $k_0 > k_i$, then peg i has color x. If $k_0 < k_i$, then peg i has color y. The secret contains k_0 pegs of color x and $p - k_0$ pegs of color y. Hence, the color at peg p can be deduced. Finally, we ask the last question to receive p blacks. Hence, we have the upper bound $b(p, 2) \leq p+1$.

Case $c = 3$

Let the colors be x, y, z. First, we ask the question $xxx \dots xx$. We receive k_0 blacks. Next we ask the p questions: $yxx \dots xx$, $xyx \dots xx$, $xxyx \dots x$, $\dots$, $xx \dots xyx$, $xxx \dots xy$. We receive $k_1, k_2, \dots, k_p$ blacks, respectively. For each $1 \leq i \leq p$ we consider the following three cases. If $k_0 > k_i$, then peg i has color x. If $k_0 < k_i$, then peg i has color y. If $k_0 = k_i$, then peg i has color z. Finally, we ask the last question to receive p blacks. Hence, we have the upper bound $b(p, 3) \leq p+2$. □

7. Four colors

We devote a separate section to the case of four colors, because it concerns the original motivation of Goodrich's work, see [9, Sect. 1.2, the first paragraph]. The problem of finding genomic data, stated there, is in fact the game $GB(p, 4)$.

Theorem 11. *For $p \geq 1$ it holds*

$$b(p, 4) \quad \leq \quad p + 3.$$

Proof. Let $p = 4s + t$, where $s \in \mathbb{N}_0$ and $t \in \{1, 2, 3, 4\}$. Let the colors be w, x, y, z. We begin with three questions: $www \dots w$, $xxx \dots x$, $yyy \dots y$. Then we know how many times the colors w, x and y appear in the secret. The number of pegs of the color z can be deduced. Next we apply Procedure I for the first $4s$ pegs (see below), requiring 4s questions. After that we know the colors of the first $4s$ pegs and how many times each color appears in the last t pegs. As explained in the Procedures II, III, IV, V (see below), we can guess these last pegs in at most $t - 1$ questions. Finally, we ask the last question. Totally, we have at most

$$3 + 4s + (t - 1) + 1 \quad = \quad p + 3$$

questions.

Procedure I: for the first $4s$ pegs

We define a new auxiliary game as follows. Consider the game $GB(p, c)$ and $p_0 < p$. Assume that in $GB(p, c)$ we begin with $c - 1$ questions of the form $x_i x_i x_i \dots x_i$, each with different color x_i for $i = 1, 2, \dots, c - 1$. Let the answers be k_1, k_2, $\dots$, k_{c-1}. After that we know that color x_i appears k_i times in the secret for $i = 1$, 2, $\dots$, $c - 1$. Note that

$$k_c \quad := \quad p - \sum_{i=1}^{c-1} k_i$$

is the number, how often the color x_c appears in the secret. Then $GB'(p_0, c)$ is defined as the game of finding the first p_0 pegs of the secret of $GB(p, c)$, where only questions are used with a fixed color x_i in the last $p - p_0$ pegs for $i = 1, 2, \dots, c$. If such a question receives the answer l blacks, the difference $l - k_i$ gives information about the first p_0 pegs. In the game $GB'(p_0, c)$ we consider as answers only the differences $l - k_i$, and not the absolute numbers of blacks l. By $b'(p_0, c)$ we denote the pessimistic number of questions in this game excluding the final question, which gives the correct answer for the first p_0 pegs.

The idea behind this game is that we can repeat this strategy $\lfloor p/p_0 \rfloor$ times, where each repetition computes the next p_0 pegs by setting in all questions as previous pegs the correct ones.

We adapted the computer program to search for strategies in games $GB'(p_0, c)$. The computation returned that $b'(4, 4) = 4$, which means that each 4 pegs can be guessed in 4 questions. Observe that $b'(4, 4) < b(4, 4) - 1$, which is reasonable, as in the game GB' we have $2p+1$ possible answers to each question and in GB only $p + 1$ ones. The first two questions of this strategy are:

$$wwww\ xxx \ldots x, \tag{2}$$
$$wwxx\ \ yyy \ldots y. \tag{3}$$

If the answer to (2) is 4 or −4, the second question is not needed, and we know that the first four pegs are *wwww* or *xxxx*, respectively. In the other case we ask (3). If the combination of answers to (2) and (3) is $(0,4)$, $(0,-4)$, $(2,2)$, $(2,-2)$, $(-2,2)$, $(-2,-2)$ then the first four pegs are *wwxx*, *yyyy*, *wwzz*, *yyww*, *zzxx*, *xxyy*, respectively. In the other cases we need to ask the third or even the fourth question. Unfortunately, almost each combination of answers needs an individual treatment. Note that the whole strategy, being too complicated to be presented here, has been made publicly available [20].

Procedure II: for the last 1 *peg, if* $t = 1$

We do not need to ask any question. The color of the last peg can be deduced.

Procedure III: for the last 2 *pegs, if* $t = 2$

If both pegs have the same color, we know this color, and we are done. If the pegs have different colors, say w and x, we ask the question

$$\underbrace{v_1 \ldots v_{p-2}}_{\text{known pegs}} wx. \tag{4}$$

This question distinguishes two possible combinations, as shown in Table 4.

Table 4: Strategy for $t = 2$

The last two pegs	The answer to (4)
wx	p
xw	$p-2$

Procedure IV: for the last 3 *pegs, if* $t = 3$

If all three pegs have the same color, we are done. In the other case we ask the two questions:

$$\underbrace{v_1 \ldots v_{p-3}}_{\text{known pegs}} wxy, \tag{5}$$
$$\underbrace{v_1 \ldots v_{p-3}}_{\text{known pegs}} wwx. \tag{6}$$

If there are only two colors, say w appears two times and x appears one time, there are three possible combinations of the last three pegs. As shown in Table 5, all combinations are distinguished after (5).

If we have three colors, say w, x, y, there are six possibilities. As shown in Table 6, all combinations are distinguished after the questions (5) and (6). Note that "—" indicates that the second question is not required.

Table 5: First strategy for $t = 3$

The last three pegs	The answer to (5)
wwx	$p-2$
wxw	$p-1$
xww	$p-3$

Table 6: Second strategy for $t = 3$

The last three pegs	The answer to (5)	The answer to (6)
wxy	p	—
xyw	$p-3$	$p-3$
ywx	$p-3$	$p-1$
yxw	$p-2$	$p-3$
xwy	$p-2$	$p-2$
wyx	$p-2$	$p-1$

Procedure V: for the last 4 *pegs, if* $t = 4$

If all four pegs have the same color, we are done. In the other case we ask the two questions:

$$\underbrace{v_1 \ldots v_{p-4}}_{\text{known pegs}} wxyz, \tag{7}$$
$$\underbrace{v_1 \ldots v_{p-4}}_{\text{known pegs}} wwxy. \tag{8}$$

If there are only two colors with cardinalities 3 and 1, say w appears three times and x appears one time, there are four possible combinations of the last four pegs. As shown in Table 7, all combinations are distinguished after the questions (7) and (8).

If there are only two colors with cardinalities 2 and 2, say w and x appear two times, there are six combinations. As shown in Table 8, all combinations are distinguished after the questions (7) and (8).

Table 7: First strategy for $t = 4$

The last four pegs	The answer to (7)	The answer to (8)
wwwx	$p-3$	$p-2$
wwxw	$p-3$	$p-1$
wxww	$p-2$	—
xwww	$p-4$	—

Table 8: Second strategy for $t = 4$

The last four pegs	The answer to (7)	The answer to (8)
wwxx	$p-3$	$p-1$
xxww	$p-3$	$p-4$
wxwx	$p-2$	$p-3$
wxxw	$p-2$	$p-2$
xwwx	$p-4$	$p-3$
xwxw	$p-4$	$p-2$

If there are only three colors, say w appears two times and x and y appear one time, and after (7) and (8) the secret is not known, we ask the third question

$$\underbrace{v_1 \dots v_{p-4}}_{\text{known pegs}} xxzz. \tag{9}$$

The answers for all possible 12 combinations are shown in Table 9.

Table 9: Third strategy for $t = 4$

The last four pegs	The answer to (7)	The answer to (8)	The answer to (9)
wwxy	$p-3$	p	—
wwyx	$p-2$	$p-2$	$p-4$
wxwy	$p-2$	$p-2$	$p-3$
wywx	$p-3$	$p-3$	$p-4$
wxyw	$p-1$	—	—
wyxw	$p-3$	$p-2$	—
xwyw	$p-3$	$p-3$	$p-3$
ywxw	$p-4$	$p-2$	$p-4$
xwwy	$p-4$	$p-2$	$p-3$
ywwx	$p-4$	$p-3$	—
xyww	$p-4$	$p-4$	—
yxww	$p-3$	$p-4$	—

If all pegs have different colors, there are 24 possible combinations. The answers to the questions (7) and (8) for all combinations are shown in Table 10. If these answers do not distinguish all possible secrets, we ask a third question, which is given in the last column of Table 10. It can easily be checked that the corresponding answers distinguish all combinations. □

Note that if we would be able to compute values $b'(c, c)$ with $c > 4$ of the auxiliary game, we could generalize Theorem 10 and Theorem 11 to larger c.

Table 10: Fourth strategy for $t = 4$

The last four pegs	The answer to (7)	The answer to (8)	The third question
wxyz	p	—	—
wxzy	$p-2$	$p-2$	*wxzy*
wyxz	$p-2$	$p-2$	*wxzy*
wyzx	$p-3$	$p-3$	*wyzx*
wzxy	$p-3$	$p-1$	—
wzyx	$p-2$	$p-3$	*wzyx*
xwyz	$p-2$	$p-3$	*wzyx*
xwzy	$p-4$	$p-2$	—
xywz	$p-3$	$p-4$	*xywz*
xyzw	$p-4$	$p-4$	*xyzw*
xzwy	$p-4$	$p-3$	*xzww*
xzyw	$p-3$	$p-4$	*xywz*
ywxz	$p-3$	$p-2$	—
ywzx	$p-4$	$p-3$	*xzww*
yxwz	$p-2$	$p-4$	*yxwz*
yxzw	$p-3$	$p-4$	*xywz*
yzwx	$p-4$	$p-4$	*xyzw*
yzxw	$p-4$	$p-3$	*xzww*
zwxy	$p-4$	$p-1$	—
zwyx	$p-3$	$p-3$	*wyzx*
zxwy	$p-3$	$p-3$	*wyzx*
zxyw	$p-2$	$p-4$	*yxwz*
zywx	$p-4$	$p-4$	*xyzw*
zyxw	$p-4$	$p-3$	*xzww*

8. The general case

First, we improve the upper bound (1) shown by Goodrich [9]. In our proof we also use the following procedure of Goodrich's work.

Let $1 \le l \le m < r \le p$. We assume that we know how many times each color appears in the range $[l, \dots, r]$ and outside of the range. After performing the procedure we know how many times each color appears in the range $[l, \dots, m]$ and thus also how many times in the the range $[m+1, \dots, r]$. Let $x_0, x_1, \dots, x_k$ be the colors appearing in the range $[l \dots r]$, where $k \le r - l$ and $k < c$. The procedure asks k questions, where in the range $[l \dots r]$

$$\underbrace{x_i \dots x_i}_{m-l+1} \underbrace{x_0 \dots x_0}_{r-m}$$

for $i = 1, 2, \dots, k$ is used, and at the other positions an arbitrary fixed color is used.

Theorem 12. *For $c \geq 1$ and $p \geq 1$ it holds*

$$b(p,c) \quad \leq \quad \begin{cases} c + p\lceil \log_2 p \rceil - p + 1 & \text{for } c > p, \\ c + p\lceil \log_2 c \rceil & \text{for } c \leq p. \end{cases}$$

Proof. We prove each case separately.

Case $c > p$

We prove that

$$b(p,c) \quad \leq \quad (c-1) + (p\lceil \log_2 p \rceil - p + 1) + 1.$$

We ask $c - 1$ questions of the form $xxx \ldots x$, each with different color x. This is the first term in the above inequality. As the number of the last color can be deduced, we know how many times each color appears in the secret. Then we apply the binary search [11, p. 640, last two lines, p. 641, the first paragraph]. This is the second term in the inequality. Now we can ask the final question – the third term.

Case $c \leq p$

Let $p = cs + t$, where $s, t \in \mathbb{N}_0$ with $1 \leq t \leq c$. Let $L := \lceil \log_2 c \rceil$. The game consists of four phases.

The first phase

We ask $c - 1$ questions, each using a different color and containing p pegs of that color. After that we know how many times each color appears in the secret.

The second phase

Using the Goodrich procedure, we can guess which colors appear at the first c positions by using $c - 1$ questions. Similarly we can find the colors at the next c positions, etc. The colors at the last t positions can be deduced. The second phase needs

$$(c-1)s \quad = \quad p - s - t$$

questions.

The third phase

We consider each c pegs separately. At the end of this phase we consider the last t pegs. We use the Goodrich procedure to partition each c (or the last t) pegs into two groups, then into four groups, into eight groups, etc. Finally, we obtain c (or t) groups, each containing one peg. Formally, a step is the partition of c (or t) pegs, where each group containing at least two pegs is partitioned into two smaller groups. We need at most L such steps. The following example shows this more clearly.

Let $c = 9$. We have $L = \lceil \log_2 9 \rceil = 4$ steps. After the 1st step we have a partition into two groups containing 4 and 5 pegs, respectively. We denote this by 4|5. This requires $8 = c - 1$ questions. After the 2nd step we have a partition 2|2|2|3. This requires $(4-1) + (5-1) = c-2$ questions. After the 3rd step we have 1|1|1|1|1|1|1|2. This requires $(2-1) + (2-1) + (2-1) + (3-1) = c - 4$ questions. After the 4th step we have 1|1|1|1|1|1|1|1|1. This requires $2 - 1 \leq c - 1$ questions.

Each step requires at most c questions minus the number of calls to the Goodrich procedure, which was called exactly $c - 1$ times. Hence, we have totally $cL - c + 1$ questions. We repeat the above for each subsequent c pegs and for the last t pegs. Hence, this phase requires at most

$$s(cL - c + 1) + (tL - t + 1) \quad = \quad pL - p + s + 1$$

questions.

The fourth phase

We ask the final question.

Adding questions in all phases and taking into account that $t \geq 1$ we conclude

$$\begin{aligned} b(p,c) \quad & \leq \quad (c-1) + (p - s - t) \\ & \qquad + (pL - p + s + 1) + 1 \\ & = \quad c + pL - t + 1 \\ & \leq \quad c + pL \\ & = \quad c + p\lceil \log_2 c \rceil. \end{aligned}$$

□

Finally, we consider the lower bound.

Theorem 13. *For $p \geq 2$ and $c \geq 2$ it holds*

$$b(p,c) \quad \geq \quad c + \frac{p}{\log_2 p} - 2.$$

Proof. By Lemma 4 we have

$$b(p,c) \quad \geq \quad b_*(p,2) + c - 2$$

It is sufficient to show that

$$b_*(p,c) \quad \geq \quad \frac{p \log_2 c}{\log_2 p}.$$

We use the information-theoretic bound (see [11, Sect. 2.2]). Let $q = b_*(p,c)$. The number of possible secrets is c^p. There are $p + 1$ possible answers, where exactly one answer ends the game. Let $T(p,q)$ be the maximum number of leaves in a rooted tree of height q, where each node has at most $p + 1$ children exactly one of which is a leaf. We have $T(p,q) = \sum_{i=0}^{q-1} p^i \leq p^q$. It must hold $c^p \leq T(p,q)$. Hence, we have $q \geq \frac{p \log_2 c}{\log_2 p}$. □

Note that Theorems 12 and 13 are rather useless to get new results for Table 1. For small p and c, better bounds can be obtained using already knows values and the results for fixed number of pegs and colors.

9. Open problems

The obtained results leave many interesting open questions. The comparison of Table 1 and Table 2 immediately suggests the following conjecture.

Conjecture 14. *For $p \geq 1$ and $c \geq 3$ it holds*

$$b(p,c) \quad = \quad b_*(p,c).$$

From Table 2 we conjecture the following.

Conjecture 15. *For $p \geq 1$ it holds*

$$b_*(p,p+1) \quad = \quad 2p.$$

If this would be true, we could apply Lemma 4 for $c_0 = p+1$ and $c_1 = c$ leading to $b(p,c) \geq c+p-1$ for $c > p$.

Similarly, from Table 3 we conjecture the following (see also the proof of Theorem 8(g)).

Conjecture 16. *For $p \geq 1$ it holds*

$$b^*(p,p) \quad = \quad 2p-1.$$

If this would be true, we could apply Lemma 6 for $c_0 = p$ and $c_1 = c$ leading to $b(p,c) \leq c+p-1$ for $c \geq p$.

Finally, we think that the results obtained in this paper lead strong credence to the following conjecture, which would be a generalization of Theorem 8(h), Theorem 10 and Theorem 11.

Conjecture 17. *(a) For $c > p \geq 1$ it holds*

$$b(p,c) \quad = \quad c+p-1.$$

(b) For $p \geq c \geq 1$ it holds

$$b(p,c) \quad \leq \quad c+p-1.$$

References

[1] L. Bento, L. Pereira, A.C. Rosa, Mastermind by evolutionary algorithms. in: Proc. of ACM Symp. Applied Computing, ACM Press (1999) 307–311.

[2] Z. Chen, C. Cunha, S. Homer, Finding a hidden code by asking questions, in: Proc. of Second Annual International Conference on Computing and Combinatorics (COCOON), Lecture Notes in Comput. Sci. 1090 (1996) 50–55.

[3] S.T. Chen, S.S. Lin, Optimal algorithms for $2 \times n$ AB games – a graph-partition approach, J. Inform. Sci. and Engineering 20(1) (2004) 105–126.

[4] S.T. Chen, S.S. Lin, L.T. Huang, Optimal algorithms for $2 \times n$ Mastermind games – a graph partition approach, Comput. J. 47(5) (2004) 602–611.

[5] S.T. Chen, S.S. Lin, L.T. Huang, A two-phase optimization algorithm for Mastermind, Comput. J. 50(4) (2007) 435–443.

[6] V. Chvátal, Mastermind, Combinatorica 3 (1983) 325–329.

[7] W. Goddard, Static Mastermind, J. Combin. Math. Combin. Comput. 47 (2003) 225–236.

[8] W. Goddard, Mastermind revisited, J. Combin. Math. Combin. Comput. 51 (2004) 215–220.

[9] M.T. Goodrich, On the algorithmic complexity of the Mastermind game with black-peg results, Inform. Process. Lett. 109 (2009) 675–678.

[10] D.L. Greenwell, Mastermind, J. Recr. Math. 30(3) (1999–2000) 191–192.

[11] G. Jäger, M. Peczarski, The number of pessimistic guesses in Generalized Mastermind, Inform. Process. Lett. 109 (2009) 635–641.

[12] T. Kalisker, D. Camens, Solving Mastermind using genetic algorithms, in: Proc. of Genetic and Evolutionary Computation Conference (GECCO), Lecture Notes in Comput. Sci. 2724 (2003) 1590–1591.

[13] D.E. Knuth, The computer as Mastermind, J. Recr. Math. 9(1) (1976–77) 1–6.

[14] K. Koyama, T.W. Lai, An optimal Mastermind strategy, J. Recr. Math. 25(4) (1993) 251–256.

[15] B.D. McKay, Isomorph-free exhaustive generation, J. Algorithms 26 (1998) 306–324.

[16] J.J. Merelo-Guervós, P. Castillo, V.M. Rivas, Finding a needle in a haystack using hints and evolutionary computation: the case of evolutionary Mastermind, Appl. S. Comput. 6 (2006) 170–179.

[17] J. Stuckman, G.Q. Zhang, Mastermind is NP-Complete, INFOCOMP J. Comput. Sci. 5 (2006) 25–28.

[18] P.F. Swaszek, The Mastermind novice, J. Recr. Math. 30(3) (1999–2000) 193–198.

[19] Source code of [15]. Available: `http://cs.anu.edu.au/people/bdm/nauty`.

[20] Source code and $GB'(4,4)$ strategy. Available: `http://mimuw.edu.pl/~marpe/research/src/gbpmm.zip`.