GREEDY COLORINGS FOR THE BINARY PAINTSHOP PROBLEM

HADIS AMINI, FRÉDÉRIC MEUNIER, HÉLOÏSE MICHEL, AND ATEFEH MOHAJERI

Abstract. Cars have to be painted in two colors in a sequence where each car occurs twice; assign the two colors to the two occurrences of each car so as to minimize the number of color changes. This problem is denoted by $\mathrm{PPW}(2,1)$. This version and a more general version – with an arbitrary multiset of colors for each car – were proposed and studied for the first time in 2004 by Epping, Hochstättler and Oertel. Since then, other results have been obtained: for instance, Meunier and Sebő have found a class of $\mathrm{PPW}(2,1)$ instances for which the greedy algorithm is optimal. In the present paper, we focus on $\mathrm{PPW}(2,1)$ and find a larger class of instances for which the greedy algorithm is still optimal. Moreover, we show that when one draws uniformly at random an instance w of $\mathrm{PPW}(2,1)$, the greedy algorithm needs at most $1/3$ of the length of w color changes. We conjecture that asymptotically the true factor is not $1/3$ but $1/4$. Other open questions are emphasized.

Introduction

In [2], T.Epping, W.Hochstättler and P.Oertel introduced the following problem. Given a sequence of cars where repetition can occur, and for each car a multiset of colors where the sum of the multiplicities is equal to the number of repetitions of the car in the sequence, decide the color to be applied for each occurrence of each car so that each color occurs with the multiplicity that has been assigned. The goal is to minimize the number of color changes in the sequence. In the present note, we are interested in the case $\mathrm{PPW}(2,1)$: each car occurs twice and has to be painted in two colors; assign the two colors to the two occurrences of each car so as to minimize the number of color changes. We call this problem *the binary paintshop problem.* If cars are considered to be letters in an alphabet, the following is a formalization.

$\mathbf{PPW(2,1)}$ [Binary Paint Shop Problem] Given a finite alphabet Σ of cardinality n, whose elements are called *letters*, a word $w=(w_1,\ldots,w_{2n})\in\Sigma^{2n}$ where each letter appears twice, find a *coloring* $f_1,\ldots,f_{2n}\in\{\mathbf{red},\mathbf{blue}\}$ such that

$$\text{for all } i,j\in\{1,\ldots,2n\},\qquad w_i=w_j \text{ and } i\neq j \quad\Rightarrow\quad f_i\neq f_j$$

and the number of color changes within $(f_1,\ldots,f_{2n})$ is minimized.

One has a *color change* in f whenever $f_i\neq f_{i+1}$. The minimum of the number of color changes is denoted $\gamma=\gamma(w)=\gamma(w;f)$.

$\mathrm{PPW}(2,1)$ is known to be APX-hard [1]. Even with a constant ratio, no approximation algorithm is known. In the paper [3], some polynomial algorithms are given for restricted instances of $\mathrm{PPW}(2,1)$. In particular, if the instances are fifo, it is shown that the greedy algorithm is optimal. The *greedy algorithm* for an instance $\mathrm{PPW}(2,1)$ consists in the coloration of the letters in the given order so as to change the current color only at the second occurrences of letters, and only if necessary. A $\mathrm{PPW}(2,1)$ problem is said to be *fifo* if for any two letters the order of the first occurrences is the same as that of the second occurrences. In other words, in the car manufacturing model, the car that is proceeded first is also finished first. Equivalently, the instance w has no subword of the form $abba$. For example, the instance $ABACBC$ is fifo.

We extend this result as follows.

Theorem 1. *The instances of* $\mathrm{PPW}(2,1)$ *having neither a subword of the form $xyxzzy$ nor a subword of the form $xyyzxz$ are solved optimally by the greedy algorithm.*

Key words and phrases. greedy algorithm; paintshop problem; probabilistic bound.

Note that the two excluded subwords are the same up to a mirror symmetry. Moreover, as $xyxzzy$ and $xyyzxz$ have both subwords of the form $abba$, this theorem contains the former one.

A second theorem concerning the greedy algorithm is shown.

Theorem 2. *Let $n \geq 2$ be a fixed integer. When the instances of* PPW(2, 1) *are chosen uniformly at random among the instances of fixed size* $2n$ *(with n distinct letters), then one has*

$$\mathbb{E}_n(g) \leq \frac{2}{3}n,$$

where g is the number of color changes when one applies the greedy algorithm.

It shows that the greedy algorithm provides good solutions in general (compare with the 3/4 factor of the paper [3]).

The plan will simply be the following. In the first section, we introduce notations and basic tools concerning the binary paintshop problem. In the second section, Theorem 1 is proved and several connected open questions are emphasized. In the last one, one proves Theorem 2.

1. Notation and basic tools

Define for each input $w = (w_1, \ldots, w_{2n})$ of the PPW(2, 1) problem an hypergraph on the set $\{1, \ldots, 2n-1\}$ defined as

$$\mathcal{I}(w) := \{\{i, i+1, \ldots, j-1\} : 1 \leq i < j \leq 2n,\ w_i = w_j\}.$$

The hyperedges are intervals. In terms of the paintshop problem one can think of the elements of $\{1, \ldots, 2n-1\}$ as possible moments for color change: if moment i ($i = 1, \ldots, 2n-1$) is chosen, that means changing the color in our machine right after the occurrence of w_i (before the occurrence of w_{i+1}). See Figure 1 for an illustration.

The paintshop problem PPW(2, 1) consists in *designing a minimum number of color changes so that each hyperedge – each interval – of $\mathcal{I}(w)$ contains an odd number of them.* Less formally, suppose given a finite collection of intervals on the real line, PPW(2, 1) aims to find the minimum number of points such that each interval contains an odd number of them.

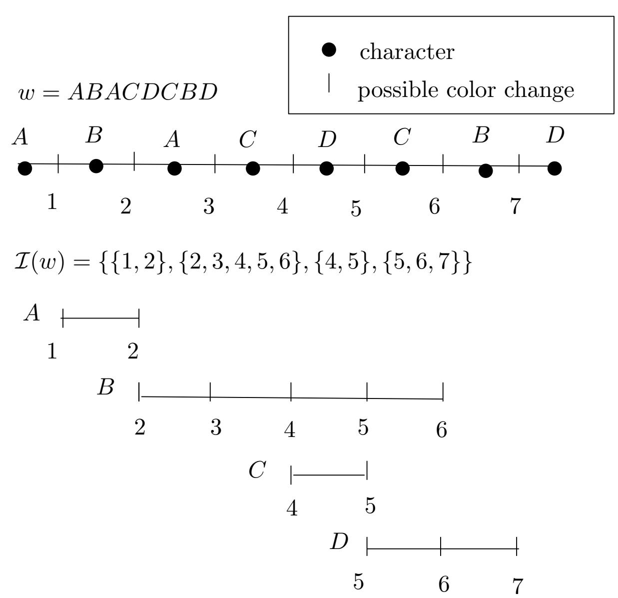

Figure 1. Illustration of the definition of $\mathcal{I}(w)$: an instance of PPW(2, 1) can be seen as a collection of intervals.

A word $w' = (w'_1, \ldots, w'_{l'})$ is *subword* of a word $w = (w_1, \ldots, w_l)$ if there is a strictly increasing map $\eta : \{1, \ldots, l'\} \to \{1, \ldots, l\}$ such that $w'_i = w_{\eta(i)}$. Less formally, starting with a word w, one gets a subword w' by deleting some of the characters w_i of w.

A collection $\mathcal{C}$ of subsets of a finite set V is said to be *laminar* if for all $A, B \in \mathcal{C}$, one has

$$A \subseteq B \quad \text{or} \quad B \subseteq A \quad \text{or} \quad A \cap B = \emptyset.$$

The notion of laminar collection is a very classical one in combinatorial optimization. In the case of binary paintshop problem, one will need a more specific notion. A collection $\mathcal{C}$ is *evenly laminar* if

- it is laminar, and
- for all $A \in \mathcal{C}$, there is an even number of subsets of $\mathcal{C}$ contained in A and distinct from A: the set $\{B \in \mathcal{C} : B \subsetneq A\}$ has an even cardinality.

2. Optimality of the greedy algorithm

Let $w = (w_1, \ldots, w_{2n})$ be an input of PPW$(2,1)$. We define the *greedy intervals* as the intervals of $\mathcal{I}(w)$ that terminate at color changes when one applies the greedy algorithm. We denote their set by $\mathcal{G}(w)$, which is then a subset of $\mathcal{I}(w)$. In the proof of Theorem 1, we use two lemmas, which together lead simply to the proof of the theorem. The first one shows that the greedy intervals are evenly laminar. The second – already proved in [1], but with a different proof – shows that any evenly laminar subcollection of intervals of $\mathcal{I}(w)$ provides a lower bound for the binary paintshop problem on w.

Lemma 1. *Let w be an input of* PPW$(2,1)$*. If w has neither a subword of the form $xyxzzy$ nor a subword of the form $xyyzxz$, then the set of greedy intervals $\mathcal{G}(w)$ is evenly laminar.*

Proof. Let us simply check that the set of greedy intervals $\mathcal{G}(w)$ is laminar. The fact that it is then evenly laminar is a straightforward consequence of the fact that the number of color changes (that is the number of right endpoints of greedy intervals) on any interval must be odd.

For a contradiction, suppose that there are at least two greedy intervals $[a_1, a_2]$ and $[b_1, b_2]$ such that $a_1 < b_1 \leq a_2 < b_2$, and choose them in such a way that a_2 is minimal, and, for this a_2, such that b_2 is minimal. Recall that the elements of these intervals are possible moments of color change; they are located between the pairs of consecutive characters.

Since the number of color changes on $[a_1, a_2]$ and $[b_1, b_2]$ must be both odd, we know that there is at least one color change in $[a_1, b_1 - 1] \cup [a_2 + 1, b_2 - 1]$.

Suppose first that there is a color change in $[a_2 + 1, b_2 - 1]$ (and hence assume that $b_2 - 1 \geq a_2 + 1$). It means that there is a greedy interval $[c_1, c_2]$ such that $c_2 \in [a_2 + 1, b_2 - 1]$. Choose c_2 minimal. By minimality of b_2, the left endpoint c_1 is either $> a_2$ or $< a_1$. But if $c_1 > a_2$, then one has a subword $xyxzzy$ with $x := a$, $y := b$ and $z := c$. Hence $c_1 < a_1$. Define the number of color changes counted modulo 2:

- r color changes in $[c_1, a_1 - 1]$,
- s color changes in $[a_1, b_1 - 1]$,
- t color changes in $[b_1, a_2]$,
- 1 color change in $[a_2 + 1, c_2]$ (by minimality of c_2), and
- u color changes in $[c_2 + 1, b_2]$.

One has: $r + s + t + 1 = 1 \bmod 2$ and $s + t = 1 \bmod 2$, hence $r = 1 \bmod 2$. There is a color change in $[c_1, a_1 - 1]$, and, by minimality of a_2, there must be a greedy interval $[d_1, d_2]$ with $c_1 < d_1 \leq d_2 < a_1 - 1$, which is impossible, otherwise $xyyzxz$ would be a subword with $x := c$, $y := d$ and $z := b$. Therefore, there is no color change in $[a_2 + 1, b_2 - 1]$.

Suppose now that there is a color change in $[a_1, b_1 - 1]$, and no color change in $[a_2 + 1, b_2 - 1]$. There is a greedy interval $[c_1, c_2]$ such that $a_1 \leq c_2 \leq b_1 - 1$. By minimality of a_2, one has $c_1 > a_1$, and thus a subword $xyyzxz$ with $x := a$, $y := c$ and $z := b$, which is impossible.

Therefore, it is not possible to find two greedy intervals $[a_1, a_2]$ and $[b_1, b_2]$ such that $a_1 < b_1 \leq a_2 < b_2$: the set $\mathcal{G}(w)$ is laminar. By the remark above, it is evenly laminar. □

Lemma 2. *Let w be an input of* PPW$(2,1)$ *and let $\mathcal{B} \subseteq \mathcal{I}(w)$. If $\mathcal{B}$ is evenly laminar, then one has $\gamma(w) \geq |\mathcal{B}|$. In other words, the cardinality of any evenly laminar subcollection of $\mathcal{I}(w)$ is a lower bound for the binary paintshop problem.*

Proof. Take any admissible solution of PPW$(2,1)$, which provides a set T of integers (color changes) such that $I \cap T$ is odd for any $I \in \mathcal{B}$. We prove by induction on the cardinality of $\mathcal{B}$ the following assertion:

Call $I_1, I_2, \ldots, I_{|\mathcal{B}|}$ the intervals of $\mathcal{B}$. Then there exist $x_1, x_2, \ldots, x_{|\mathcal{B}|} \in T$, all distinct, such that for each i, one has $x_i \in I_i$.

For $|\mathcal{B}| = 1$, the assertion is trivially true. Hence suppose that the cardinality is at least 2. Since $\mathcal{B}$ is laminar, one can choose a maximal interval J for inclusion, delete it, and obtain an evenly laminar subcollection $\mathcal{B}' = \{I_1, I_2, \ldots, I_{|\mathcal{B}'|}\}$. By induction, there exist $x_1, x_2, \ldots, x_{|\mathcal{B}'|} \in T$, all distinct, such that for all $i = 1, \ldots, |\mathcal{B}'|$, one has $x_i \in I_i$. Each interval I of $\mathcal{B}'$ such that $I \subseteq J$ provides a distinct x_i of T contained in J. By definition of an evenly laminar collection, the cardinality of $\{I \in \mathcal{B}' : I \subseteq J\}$ is even. Since $J \cap T$ is odd, there is an $x \in J \cap T$ that is not provided by any of the $I \subseteq J$ of $\mathcal{B}'$. This x cannot be in another interval I of $\mathcal{B}'$ since by definition of a laminar collection and by maximality of J, one has necessarily $I \subseteq J$. Hence, one has found an x in $J \cap T$, which is distinct of the x_i, $i = 1, \ldots, |\mathcal{B}'|$. This achieves the induction, and hence, the proof. □

Given a collection of intervals, in [1] (Section 4), a polynomial algorithm that finds the largest evenly laminar subcollection is given. Note that for NP-hard minimization problem it is always nice to have good and polynomially computable lower bounds.

With these two lemmas, it is easy to prove Theorem 1.

Proof of Theorem 1. Let w be an instance of PPW$(2, 1)$. Denote by $g(w)$ the number of color changes when the greedy algorithm is applied on w. One has

$$g(w) \geq \gamma(w) \geq |\mathcal{G}(w)| \geq g(w). \tag{1}$$

The first inequality tells simply that $\gamma(w)$ is the minimal number of color changes for an admissible coloring. The second inequality is a consequence of Lemma 1, which tells that $\mathcal{G}(w)$ is an evenly laminar subcollection of intervals of $\mathcal{I}(w)$, and of Lemma 2, which tells that $|\mathcal{G}(w)|$ is hence a lower bound for $\gamma(w)$. The last inequality is true because by definition one has $|\mathcal{G}(w)| = g(w)$.

The left and the right terms of the chain of inequalities (1) are equal, therefore all inequalities are equalities, and thus $g(w) = \gamma(w)$. □

In the statement of Theorem 1, one has a sufficient condition. The word $w = ABCBDDCA$ shows that it is not a necessary condition: the greedy algorithm provides the optimum, although w has a subword of the form $xyxzzy$ (take $x := B$, $y := C$ and $z := D$). We do not know whether it is possible to write a polynomially checkable necessary and sufficient condition for the optimality of the greedy algorithm.

An other open question is the following one: the condition of Theorem 1 is polynomially checkable: just repeat $O(|\Sigma|^3) = O(n^3)$ times the longest common subsequence algorithm (that works by dynamic programming). Is it possible to avoid the $O(n^3)$ repetitions? The answer is probably no, because this problem is the so-called *pattern matching problem*, which is in general an NP-hard problem (see the paper written by Vialette [4]). Note that a way for using Theorem 1 – or rather Lemma 2 – consists in applying the greedy algorithm and then in checking if the greedy intervals are indeed evenly laminar, which can be done in $O(n)$.

3. Expected value of the number of color changes in the greedy approach

The proof of Theorem 2, which is the main purpose of the present section, works by induction. The theorem tells us that, when an input of PPW$(2, 1)$ of fixed size $2n$ is drawn uniformly at random (hence with n distinct letters), the greedy algorithm makes in average at most $2/3n$ color changes. The unique counter-example is when $n = 1$, since in this case, one needs always 1 color change, which is greater that 2/3. First, we check that the theorem is true for $n = 2$ (Claim 1), for $n = 3$ (Claim 2) and for $n = 4$ (Claim 3). The proof consists then simply of showing that if it is true for n, then it is also true for $n + 3$.

Claim 1. *If $n = 2$, the greedy algorithm provides in average $4/3$ color changes.*

Indeed, up to permutations of the alphabet, one has 3 $\left(= \frac{1}{2!}\binom{4}{2}\right)$ types of words (we keep only them with the second occurrences appearing in the order AB) and a total of $2+1+1 = 4$ color changes. In the following array, we give the 3 types, and the number of color changes provided by the greedy algorithm.

$AABB$	2
$ABAB$	1
$BAAB$	1

Claim 2. *If $n = 3$, the greedy algorithm provides in average $27/15 < 2/3 \times 3$ color changes.*

Indeed, up to permutations of the alphabet, one has 15 $\left(= \frac{1}{3!}\binom{6}{2}\binom{4}{2}\right)$ types of words (we keep only them with the second occurrences appearing in the order ABC) and a total of $3 \times 3 + 6 \times 2 + 6 \times 1 = 27$ color changes. In the following array, we give the 15 types, and the number of color changes provided by the greedy algorithm.

$AABBCC$	3	$AABCBC$	2	$AACBBC$	2
$ABABCC$	2	$ABACBC$	2	$ABCABC$	1
$ACABBC$	3	$ACBABC$	1	$BAABCC$	2
$BAACBC$	2	$BACABC$	1	$BCAABC$	1
$CAABBC$	3	$CABABC$	1	$CBAABC$	1

Claim 3. *If $n = 4$, the greedy algorithm provides in average $240/105 < 2/3 \times 4$ color changes.*

Indeed, up to permutations of the alphabet, one has 105 $\left(= \frac{1}{4!}\binom{8}{2}\binom{6}{2}\binom{4}{2}\right)$ types of words (we keep only them with the second occurrences appearing in the order $ABCD$) and a total of $9 \times 4 + 36 \times 3 + 36 \times 2 + 24 \times 1 = 240$ color changes. In the following array, we give the 105 types, and the number of color changes provided by the greedy algorithm (they are ordered in the lexicographic order).

$AABBCCDD$	4	$AABBCDCD$	3	$AABBDCCD$	3	$AABCBCDD$	3	$AABCBDCD$	3
$AABCDBCD$	2	$AABDBCCD$	4	$AABDCBCD$	2	$AACBBCDD$	3	$AACBBDCD$	3
$AACBDBCD$	2	$AACDBBCD$	2	$AADBBCCD$	4	$AADBCBCD$	2	$AADCBBCD$	2
$ABABCCDD$	3	$ABABCDCD$	2	$ABABDCCD$	2	$ABACBCDD$	3	$ABACBDCD$	2
$ABACDBCD$	2	$ABADBCCD$	2	$ABADCBCD$	2	$ABCABCDD$	2	$ABCABDCD$	2
$ABCADBCD$	2	$ABCDABCD$	1	$ABDABCCD$	3	$ABDACBCD$	3	$ABDCABCD$	1
$ACABBCDD$	4	$ACABBDCD$	3	$ACABDBCD$	4	$ACADBBCD$	4	$ACBABCDD$	2
$ACBABDCD$	2	$ACBADBCD$	2	$ACBDABCD$	1	$ACDABBCD$	3	$ACDBABCD$	1
$ADABBCCD$	3	$ADABCBCD$	3	$ADACBBCD$	3	$ADBABCCD$	3	$ADBACBCD$	3
$ADBCABCD$	1	$ADCABBCD$	3	$ADCBABCD$	1	$BAABCCDD$	3	$BAABCDCD$	2
$BAABDCCD$	2	$BAACBCDD$	3	$BAACBDCD$	2	$BAACDBCD$	2	$BAADBCCD$	2
$BAADCBCD$	2	$BACABCDD$	2	$BACABDCD$	2	$BACADBCD$	2	$BACDABCD$	1
$BADABCCD$	3	$BADACBCD$	3	$BADCABCD$	1	$BCAABCDD$	2	$BCAABDCD$	2
$BCAADBCD$	2	$BCADABCD$	1	$BCDAABCD$	1	$BDAABCCD$	3	$BDAACBCD$	3
$BDACABCD$	1	$BDCAABCD$	1	$CAABBCDD$	4	$CAABBDCD$	3	$CAABDBCD$	4
$CAADBBCD$	4	$CABABCDD$	2	$CABABDCD$	2	$CABADBCD$	2	$CABDABCD$	1
$CADABBCD$	3	$CADBABCD$	1	$CBAABCDD$	2	$CBAABDCD$	2	$CBAADBCD$	2
$CBADABCD$	1	$CBDAABCD$	1	$CDAABBCD$	3	$CDABABCD$	1	$CDBAABCD$	1
$DAABBCCD$	3	$DAABCBCD$	3	$DAACBBCD$	3	$DABABCCD$	3	$DABACBCD$	3
$DABCABCD$	1	$DACABBCD$	3	$DACBABCD$	1	$DBAABCCD$	3	$DBAACBCD$	3
$DBACABCD$	1	$DBCAABCD$	1	$DCAABBCD$	3	$DCABABCD$	1	$DCBAABCD$	1

With these three claims starting the induction, we can write down the whole proof.

Proof of Theorem 2. Let w' be a PPW(2, 1) instance of length $2(n+3)$, that is, with $n+3$ distinct letters, each of them appearing twice. Color w' with the greedy algorithm, and call the three last letters that have been colored x, y and z. Now, denote w the word obtained from w' when all occurrences of x, y and z are deleted from it. The strategy of the proof consists in showing that if the greedy algorithm needs three new color changes when x, y, z are added to w in order to obtain w', then there is another word obtained from w by adding x, y and z that need exactly one supplementary change and that compensate for the three supplementary changes of w'. This correspondence is injective. Since the average number of color changes for words of length $2(n+3)$ is equal to the average number of color changes for words of length $2n$ plus the average number of color changes added by the last three letters xyz, this implies that the average number increases at most by $2/3$ when three new letters are added.

The difficulty comes when passing from w to w' leads to adding 3 new color changes. Call such a word w' a *bad word*.

Suppose first that the end of w' is of the form xyz. A bad word w' that terminates with xyz can only occur if the greedy algorithm provides simultaneously

- an even number of color changes between the two occurrences of x
- an odd number of color changes between the first occurrence of y and the second one of x
- an even number of color changes between the first occurrence of z and the second one of x.

$$w' = \left.\begin{array}{cc} x & \text{—even—} \\ y & \text{—odd—} \\ z & \text{—even—} \end{array}\right\} xyz$$

Indeed, this situation needs three more changes: one just before the second x, then one just before the second y, and then one just before the second z.

By exchanging the positions of the first occurrences of x and y, we get

$$\left.\begin{array}{cc} x & \text{—odd—} \\ y & \text{—even—} \\ z & \text{—even—} \end{array}\right\} xyz$$

that needs only one change: one just before the second y.

Hence, it compensates correctly for the three changes.

It remains to check the case when a bad word w' does not finish with xyz but finishes by something like $xyyz$, $xyzz$, $xyyzz$, $xzyz$, $xyzyz$, and $xzyyz$ (note that if we are only interested by an asymptotic behavior, these cases are pointless, since they are unlikely to occur when n becomes large).

$\underline{xyyz}$**:** this case is problematic if one has something like

$$w' = \left.\begin{array}{cc} x & \text{—even—} \\ z & \text{—even—} \end{array}\right\} xyyz,$$

since one needs to change the color just before the second x, then just before the second y and then just before the second z – three more changes than for w. But by exchanging the first occurrences of x and y one gets

$$\left.\begin{array}{cc} y & \text{—even—} \\ z & \text{—even—} \end{array}\right\} xxyz,$$

which needs only one more color change: just before the second x. This second type of words ends with xyz but was not encountered above, since x, y and z have the same parity, hence it compensates correctly for the three changes.

$\underline{xyzz}$**:** this case is problematic if one has something like

$$w' = \left.\begin{array}{cc} x & \text{—even—} \\ y & \text{—odd—} \end{array}\right\} xyzz,$$

since one needs to change the color just before the second x, then just before the second y and then just before the second z – three more changes than for w. But by inverting the first occurrences of x and y, and by moving the first occurrence of z one position to the left, one gets

$$\left.\begin{array}{cc} y & \text{—even—} \\ x & \text{—odd—} \end{array}\right\} xzyz,$$

which needs only one more color change: just before the second x. This second type of words ends was not encountered above, hence it compensates correctly for the three changes.

$\underline{xyyzz}$**:** this case is problematic if one has something like $w' = x$—even—$xyyzz$. The word $w' = x$—even—$yzxyz$ needs only one more change than for w, and was not encountered above, since x, y and z have the same parity, hence it compensates correctly for the three changes.

$\underline{xzyz}$**:** this case is never problematic since if there is a color change just before the second occurrence of y, there is no color change just before the second occurrence of z.

$\underline{xyzyz}$**:** this case is never problematic since there is never a color change just before the second occurrence of z.

$\underline{xzyyz}$**:** this case is never problematic since there is never a color change just before the second occurrence of z.

All in all, when three new letters are added, one needs in average at most two more color changes. With Claim 1, Claim 2 and Claim 3, the induction concludes the proof. ☐

We finish the note with a conjecture, supported by experimental evidences:

Conjecture 1. *When the instances of* $\mathrm{PPW}(2,1)$ *are chosen uniformly at random among the instances of fixed size* $2n$ *(with* n *distinct letters), then one has*

$$\lim_{n\to\infty}\frac{1}{n}\mathbb{E}_n(g)=\frac{1}{2},$$

where g *is the number of color changes when the greedy algorithm is applied.*

References

1. P. S. Bonsma, T. Epping, and W. Hochstättler, *Complexity results on restricted instances of a paint shop problem for words*, Discrete Appl. Math. **154** (2006), 1335–1343.
2. T. Epping, W. Hochstättler, and P. Oertel, *Complexity results on a paint shop problem*, Discrete Appl. Math. **136** (2004), 217–226.
3. F. Meunier and A. Sebő, *Paint shop, odd cycles and splitting necklace*, Discrete Appl. Math. (Accepted for publication).
4. S. Vialette, *On the computational complexity of 2-interval pattern matching problems*, Theoret. Comput. Sci. **312** (2002), 241–263.

Ecole Polytechnique, 91128 Palaiseau Cedex
E-mail address: {hadis.amini,heloise.michel,atefeh.mohajeri}@polytechnique.edu

Université Paris Est, LVMT, ENPC, 6-8 avenue Blaise Pascal, Cité Descartes Champs-sur-Marne, 77455 Marne-la-Vallée cedex 2, France.
E-mail address: frederic.meunier@enpc.fr