LEHRSTUHL FÜR MATHEMATISCHE
GRUNDLAGEN DER INFORMATIK

P. S. Bonsma, Th. Epping, W. Hochstättler:

Complexity results on restricted instances of a paint shop problem for words

Technical Report btu-lsgdi-005.08

Contact: `bonsma@math.utwente.nl`, {`epping,wh`}`@math.tu-cottbus.de`

Brandenburgische Technische Universität Cottbus
Institut für Mathematik
Lehrstuhl für Mathematische Grundlagen der Informatik
Postfach 10 13 44
D – 03013 Cottbus

2000 Mathematics Subject Classification: 05B35, 68Q17, 90B35
Keywords: APX-hardness, Binary matroids, MaxFlow-MinCut, NP-completeness, Paint shop, Sequencing

Complexity Results on Restricted Instances of a Paint Shop Problem for Words

P. S. Bonsma [a] Th. Epping [b] W. Hochstättler [b]

[a] *Department of Applied Mathematics, University of Twente, The Netherlands*
[b] *Department of Mathematics, BTU Cottbus, Germany*

Abstract

We study the following problem: an instance is a word with every letter occurring twice. A solution is a 2-coloring of its letters such that the two occurrences of every letter are colored with different colors. The goal is to minimize the number of color changes between adjacent letters.

This is a special case of the paint shop problem for words, which was previously shown to be $\mathcal{NP}$-complete. We show that this special case is also $\mathcal{NP}$-complete and even $\mathcal{APX}$-hard. However, we can transform it into matroid theory to solve specific instances within polynomial time and apply MaxFlow-MinCut duality.

Key words: APX-hardness, Binary matroids, MaxFlow-MinCut, NP-completeness, Paint shop, Sequencing
1991 MSC: 05B35, 68Q17, 90B35

1 Introduction

An *alphabet* Σ is a set of letters. A *word* w is an ordered sequence of letters from an alphabet. The same letter can appear at multiple positions in a word. In [4], the following problem is studied: an instance consists of a word w, a set of colors $C = \{1, \ldots, c\}$, and a color requirement $r_{\sigma i} \in \mathbb{N}_{\geq 0}$ for every letter $\sigma \in \Sigma$ in the word and every color $i \in C$ which states that exactly $r_{\sigma i}$ occurrences of letter σ must be colored with color i. A feasible solution is a coloring of the letters satisfying the color requirement. If the letters σ and σ'

Email addresses: `bonsma@math.utwente.nl` (P. S. Bonsma), `epping@math.tu-cottbus.de` (Th. Epping), `hochstaettler@math.tu-cottbus.de` (W. Hochstättler).

Preprint submitted to Elsevier Science *29 September 2003*

appearing at position i and $i+1$ are colored with different colors, we say there is a *color change* between position i and $i+1$. If it is unambiguous, we will also say that there is a color change between the consecutive letters $\sigma\sigma'$. The goal is to find a feasible solution that minimizes the number of color changes. This problem is called the *paint shop problem for words (PPW)* and is motivated by an application in car manufacturing, where costly enamel color changes in the paint shop should be minimized without a perturbation of the (fixed) car body production sequence (see [4] for details).

In [4], it is shown that the PPW is $\mathcal{NP}$-complete even when restricted to instances where only two different letters occur in the word, and also when restricted to instances where only two different colors are used. The main result of the present paper is the $\mathcal{APX}$-hardness of the PPW for an even smaller subset of its instances: instances with $|C| = 2$ and $r_{\sigma i} = 1$ for every $\sigma \in \Sigma$ and $i \in C$ (called *1-regular 2-colored instances*). This answers one of the problems stated in [4]. Moreover, we show that the 1R2C-PPW is equivalent to the problem of finding a shortest circuit in a certain class of binary matroids. Consequences of this approach are polynomial time solution algorithms for specific instances and a duality theorem.

Problem 1 (1R2C-PPW) *Let w be a word of length $n = 2|\Sigma|$ in which every letter of Σ occurs exactly twice, and let C be a set of colors with $|C| = 2$. Find an assignment of the colors in C to the letters in w such that every letter of Σ is colored exactly once in each color of C and the number of color changes within w is minimized.*

2 Preliminaries

There is an insightful way (first introduced in [3]) to represent instances and solutions of the 1R2C-PPW, which is shown in Figure 1. If the two occurrences of a letter $\sigma \in \Sigma$ appear at position x_σ and y_σ in the word, represent this letter pair with the interval $I_\sigma := [x_\sigma, y_\sigma]$ (drawn as a horizontal line segment). The set of all letter pair intervals of the instance is denoted by $I = \{I_\sigma : \sigma \in \Sigma\}$. A solution is a set of real numbers corresponding to the color changes. Observe that a solution for the 1R2C-PPW is feasible if and only if there is an odd number of color changes within every interval I_σ.

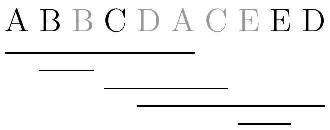


Fig. 1. The interval representation of a 1R2C-PPW instance and a solution.

In the remainder of this section, we collect the main definitions that are used within this paper. Our notation, however, is fairly standard.

2.1 Graphs

Let $G = (V, E)$ denote a simple, undirected graph. The *degree of a vertex* $v \in V$ is denoted by $d(v)$, and the *maximum degree of* G is defined by $\max\{d(v) : v \in V\}$. A *cubic graph* is a graph in which every vertex has degree 3. The edge incident with vertex i and j is denoted by ij or ji. Edges with a direction assigned to them are called *arcs*. An arc from i to j is denoted by (i, j). In this case, i is an *in-neighbor* of j and j is an *out-neighbor* of i. Directed graphs are denoted by $G = (V, A)$. If $G = (V, A)$ is a directed graph, $H = (V, E)$ is an undirected graph, $|A| = |E|$ and $(i, j) \in A \Rightarrow ij \in E$ then A is called an *orientation* of H. $G - v$ denotes the graph obtained from G by removing vertex v and all edges incident with v. $G + uv$ denotes the graph obtained by adding an edge between vertices $u, v \in V$. A *vertex cover* of a graph G is a set $S \subseteq V$ such that every edge is incident with at least one vertex of S. A *stable set* is a set $\tilde{S} \subseteq V$ such that no two vertices in $\tilde{S}$ are adjacent. If I is a set of intervals on the real line, then the *interval graph* $G_I = (I, E)$ is defined such that $I_\sigma I_\tau \in E :\Leftrightarrow I_\sigma \cap I_\tau \neq \emptyset$.

2.2 Approximation algorithms

An algorithm for a minimization problem is called a *ρ-approximation algorithm* if for every instance with optimal solution value k it gives a solution with value at most ρk. Here, $\rho = 1 + \epsilon$ with $\epsilon > 0$ (for maximization problems the definition is similar). A *PTAS* for a minimization problem is a scheme to find polynomial time $(1 + \epsilon)$-approximation algorithms for *every* $\epsilon > 0$. The problem class $\mathcal{APX}$ is the class of problems for which a polynomial time ρ-approximation algorithm exists for some $\rho > 1$ ($0 < \rho < 1$ for maximization problems). A problem is called $\mathcal{APX}$-hard if the existence of a PTAS for this problem would imply that for every problem in $\mathcal{APX}$ a PTAS exists. Moreover, Arora, Lund, Motwani, Sudan and Szegedy [2] have shown that:

Theorem 2 *If there exists a PTAS for an $\mathcal{APX}$-hard problem, then $\mathcal{P}$=$\mathcal{NP}$,*

which is quite unlikely.

2.3 Matroids

We assume basic familiarity with matroid theory, see e.g. [8,11]. In particular we will need the following concepts. Let $M = M[A]$ denote a binary vector matroid represented by a matrix A over GF(2). M is called *regular* if the entries of A can be signed so that A is represented by a totally unimodular matrix over $\mathbb{R}$. The matroid M is called a *MaxFlow-MinCut (MFMC)* matroid or is said to have the *MFMC property* if it does not contain a Fano (F_7) dual minor. By a well-known result of Tutte (see e.g. [8]), a binary matroid is *regular* if and only if it does not contain an F_7 minor or its dual.

3 Complexity results

In this section, the $\mathcal{APX}$-hardness of the 1R2C-PPW is proved using an L-reduction as introduced in [9]. We give a reduction from the following problem:

Problem 3 (3GVC) *Let $G = (V, E)$ be a cubic graph. Find a vertex cover $S \subseteq V$ of minimum cardinality.*

In [1], this problem is shown to be $\mathcal{APX}$-hard. First we show that graphs with maximum degree 3 have an ordering of the vertices and an orientation of the edges satisfying the following criterion:

Criterion 4 *Let $G = (V, A)$ be a directed graph with a complete order $\prec$ on the vertices. We say that G and $\prec$ satisfy Criterion 4 if every vertex $v \in V$ has at most one in-neighbor u with $u \prec v$, and at most one in-neighbor w with $v \prec w$, and every vertex $v \in V$ with degree 2 has exactly one in-neighbor.*

Lemma 5 *Every graph G with maximum degree 3 has an ordering of the vertices and an orientation of the edges that satisfies Criterion 4. This vertex order and orientation can be found in polynomial time.*

PROOF. We can assume $G = (V, E)$ to be connected. We proceed by induction on $|V|$. The assertion is true when $|V| = 1$. Consider four cases:

(1) G has a leaf. This case is trivial.

(2) There is a degree 2 vertex v with neighbors u, w and $uw \notin E$. Now $G' = G - v + uw$ has an orientation and a vertex order as claimed. Assume wlog. $u \prec w$. Use this orientation and order for G, with v added in this order such that $u \prec v \prec w$, and the edges uv and vw oriented as a subdivision of the arc between u and w in G'. Now for u and w the situation in G is the same as in G' and v has one in-neighbor and degree 2, so this order and orientation satisfies Criterion 4.

(3) There is a degree 2 vertex v with neighbors u, w and $uw \in E$. The graph $G' = G - v$ has an orientation and a vertex order as claimed. Assume (u, w) is oriented this way. Then u has at most one in-neighbor x in G'. If there is such an x and $x \prec u$, insert v after u in this order, otherwise insert v before u in this order. Using the orientation of G' and adding (w, v) and (v, u) gives an orientation for G. Now w has one in-neighbor.

u has two in-neighbors if and only if $d(u) = 3$, and in this case one in-neighbor appears before u in the order and one after. Otherwise u has one in-neighbor. v has one in-neighbor and degree 2. Therefore this order and orientation satisfies Criterion 4.

(4) All vertices have degree three. Choose a vertex v. Consider $G' = G - v$ and denote the neighbors of v by x, y and z. G' has an orientation and vertex order as required. In G', x, y and z all have degree 2. Let u be the in-neighbor of x and w be the in-neighbor of y. Observe that wlog. we can assume that either $u \prec x$ and $w \prec y$ or $x \prec u$ and $y \prec w$. In the first case, insert v in the order after x and y, and in the second case insert v before x and y. Using the orientation of G' and adding (v, x), (v, y) and (z, v) gives an orientation for G. Now x (y) has two in-neighbors, one ordered before x (y) and one after, and z and v both have one in-neighbor, so this order and orientation satisfies Criterion 4.

This describes a polynomial time algorithm to find a vertex order and orientation satisfying Criterion 4 for every graph with maximum degree 3. □

Theorem 6 *The 1R2C-PPW is $\mathcal{APX}$-hard.*

PROOF. Let G be an instance of the 3GVC. Find an orientation of the edges and an order $\prec$ on the vertices that satisfies Criterion 4 (see Lemma 5). The resulting directed graph will also be denoted by $G = (V, A)$ with $V = \{1, \ldots, n\}$ and $u < v \Leftrightarrow u \prec v$. We use G to construct an instance for the 1R2C-PPW. This instance will consist of n blocks of letters, one block for every vertex. For every $i \in V$ we will introduce letters A_i, B_i and C_i that only appear in block i. For every arc $a \in A$ we will introduce letter E_a that appears once in both of the blocks corresponding to the end vertices. For every vertex pair $i, j \in V$ with $i < j$ we will introduce letter D_{ij}, that appears once in block i and once in block j.

Observe that because G satisfies Criterion 4, for every vertex v we can find a complete order $\prec_A$ on the arcs incident with v such that:

- if $a = (u, v)$ and $u < v$ then $a \prec_A b$ for every other arc b incident with v.
- if $a = (u, v)$ and $v < u$ then $b \prec_A a$ for every other arc b incident with v.
- if $a \in A$ is incident with u and v and $b \in A$ is incident with v and w and $u < v < w$ then $a \prec_A b$.

If vertex i is incident with arcs a, b and c, and the order $a \prec_A b \prec_A c$ satisfies the above properties, then we introduce the following block for v:

$$A_i D_{1i} D_{2i} \ldots D_{(i-1)i} E_a E_b E_c B_i B_i D_{i(i+1)} D_{i(i+2)} \ldots D_{in} A_i C_i C_i$$

Now order the blocks left to right from 1 to n. Taken together this gives the word w for the 1R2C-PPW. See Figure 2 for an example. In this example, in block 1, $E_{(1,3)}$ is placed before $E_{(1,2)}$. This is an arbitrary choice, both options give a valid construction. The same is true for the placement of $E_{(4,1)}$ and $E_{(4,3)}$ in block 4.

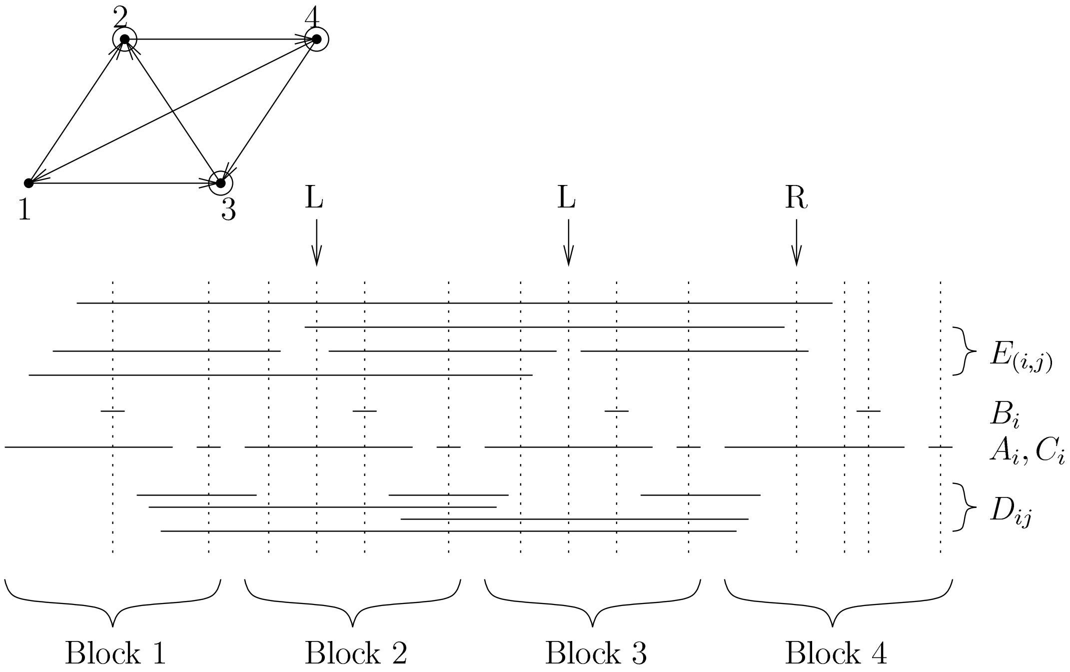


Fig. 2. A 3GVC instance and solution and the corresponding 1R2C-PPW instance and solution.

Claim 7 *If the 1R2C-PPW instance w has a solution with at most $2|V| + 2k$ color changes, then G has a vertex cover using at most k vertices.*

To prove this, we make the following observations that are true for every feasible solution of instance w.

(1) Because of the place of the B_i pair and the C_i pair in block i, in block i there are at least two color changes, for every i.
(2) If there is an arc $a \in A$ between i and j, then in any optimal solution of the 1R2C-PPW instance, either in block i or in block j there are at least four color changes.

We prove this last observation by contradiction. Suppose a is an arc between i and j, $i < j$ and in block i and in block j there are at most three color changes. Since one of the color changes in block i must be between C_iC_i, there are at most two color changes between the A_i pair. Since the number of color changes between a letter pair must be odd, there is only one color change between the A_i pair. The same holds for the A_j pair. This color change between the A_i (A_j) pair must be between the B_i (B_j) pair. In block i, E_a appears before the B_i pair, whereas D_{ij} appears after it. In block j, E_a and D_{ij} both appear before the B_j pair. This means that there is an odd number of color changes between the E_a pair if and only if there is an even number of color changes between the D_{ij} pair, a contradiction.

From the two observations above it follows that if there is a solution with at most $2|V| + 2k$ color changes, then G has a vertex cover with at most k vertices.

Claim 8 *If G has a vertex cover of cardinality k, there is a solution for the 1R2C-PPW instance w with $2|V| + 2k$ color changes.*

Let S be a vertex cover of G. If $i \notin S$, we apply only two color changes in block i: one between B_iB_i and one between C_iC_i. If $i \in S$, we use four color changes in block i: between B_iB_i and C_iC_i but also between the consecutive letters $D_{(i-1)i}E_a$ and between the consecutive letters E_cB_i. If there is an arc $a \in A$ with end vertices $i \in S$ and $j \in S$ $(i < j)$, we have to move one of the latter two color changes. If a is directed towards i, then E_aB_i is a consecutive letter combination in w and we move the color change one position to the left (between E_bE_a for some b). If a is directed towards j, then $D_{(j-1)j}E_a$ is a consecutive letter combination in w, and we move the color change one position to the right (between E_aE_b for some b). This ensures that for every a between i and j with $i < j$, there is either one color change in block i between E_a and B_i and no color change in block j between $D_{(j-1)j}$ and E_a or there is no color change in block i between E_a and B_i and one color change in block j between $D_{(j-1)j}$ and E_a. In Figure 2 an example of such a 1R2C-PPW solution corresponding to a vertex cover is shown. The color changes marked with 'L' ('R') are the color changes moved to the left (right) in this last step. We prove that this method gives a feasible solution:

(1) Let $i < j$. In block i, there is one color change to the right of D_{ij}: this is the color change between C_iC_i. In block j, there is no color change to the left of D_{ij}. In every block between i and j there is an even number of color changes. There are no color changes between blocks. So there is an odd number of color changes between the two D_{ij} letters for every $i < j$.

(2) Let a be an arc between i and j, $i < j$. We know that either there is a color change in block i between E_a and the first B_i, or there is a color change in block j between D_{ij} and E_a, but not both. In addition, there is exactly one color change in block i between the first B_i and D_{ij} (this is the color change between B_iB_i). So there is an odd number of color changes between the E_a pair if there is an odd number of color changes between the D_{ij} pair. This was shown to be true, so for every a there is an odd number of color changes between the two E_a letters.

(3) Between every A_i pair, there are either one or three color changes.

(4) Between B_iB_i and between C_iC_i there is a color change.

We conclude that this is a feasible solution with exactly $2|V| + 2|S|$ color changes.

Now suppose a $(1+\epsilon)$-approximation algorithm for the 1R2C-PPW exists. We use this to construct a $(1+3\epsilon)$-approximation algorithm for the 3GVC. Let $G = (V, E)$ be a 3GVC instance with minimum vertex cover S. We use the above transformation to construct a 1R2C-PPW instance w, with minimum number of color changes $m = 2|S| + 2|V|$. Use the approximation algorithm to find a solution with $k \leq (1+\epsilon)m$ color changes. This gives a vertex cover of cardinality $k' \leq k/2 - |V|$. Observe that since G is cubic, $|S| \geq \frac{1}{3}|E| = \frac{1}{2}|V|$, so $3|S| \geq |V| + |S|$. Therefore:

$$\frac{k' - |S|}{|S|} \leq \frac{k/2 - |V| - (m/2 - |V|)}{|S|} \leq 3\frac{k/2 - m/2}{|V| + |S|} = 3\frac{k - m}{m} \leq 3\epsilon.$$

So a PTAS for the 1R2C-PPW would give a PTAS for the 3GVC. Therefore, the 1R2C-PPW is also $\mathcal{APX}$-hard. □

Corollary 9 *The 1R2C-PPW is $\mathcal{NP}$-complete.*

PROOF. Use the same reduction as for the proof of Theorem 6. □

4 Lower and upper bounds

Observe that the maximum number of pairwise disjoint intervals in the interval set I of a 1R2C-PPW instance w corresponds to a maximum stable set in the interval graph G_I and yields a straightforward lower bound on the minimum number of color changes for w. We can generalize this idea to improve on the bound. Therefore, we use the partial ordering on I given by proper containment, and the corresponding comparability graph $G = (I, A)$, where $(I_\sigma, I_\tau) \in A :\Leftrightarrow I_\tau \subset I_\sigma$.

Now, we apply the algorithm depicted in Figure 3, which passes through G in a bottom-up fashion. A lower bound on the minimum number of color changes for a vertex $I_\sigma \in I$ is given by the value of a maximum weighted stable set in the subgraph of G induced by the out-neighbors of I_σ (we assume that this value is computed by the method `maxWSS(`I_σ`)`). Each I_σ must contain an odd number of color changes. A vertex $I_\sigma \in I$ is called *processed* if it is already weighted with a lower bound. It is called *processable* if all out-neighbors of I_σ are processed.

```
INPUT: The interval set $I$ of a 1R2C-PPW instance $w$
OUTPUT: A lower bound on the minimum number of color changes for $w$

Construct the comparability graph $G = (I, A)$ of $I$
Assign a weight $w(I_\sigma) = 1$ to all vertices $I_\sigma \in I$ with no out-neighbor

While there exists an unprocessed vertex in $G$
  Choose a processable vertex $I_\sigma \in I$ and set $w(I_\sigma) = \texttt{maxWSS}(I_\sigma)$
  If $w(I_\sigma)$ is even, set $w(I_\sigma) \leftarrow w(I_\sigma) + 1$

Add an extra vertex $r$ to $I$ and arcs $\{(r, I_\sigma) : I_\sigma \in I\}$ to $A$
Return maxWSS($r$)
```

Fig. 3. An algorithm for the computation of a lower bound.

Theorem 10 *The algorithm in Figure 3 computes a lower bound on the minimum number of color changes for a 1R2C-PPW instance in polynomial time.*

PROOF. By induction it is immediate, that $w(I_\sigma)$ is a lower bound on the number of color changes in I_σ for each $\sigma \in \Sigma$. In the computation of this lower bound, the subgraph of G_I induced by the out-neighbors of a vertex $I_\sigma \in I$ corresponds to a subset of I and, therefore, is an interval graph. It is well-known that the maximum weighted stable set problem is $\mathcal{NP}$-complete in general, but can be solved in polynomial time for an interval graph by computing a maximum weighted clique in its complement graph (see [5]). $\square$

By linear algebra we get a general upper bound of $|\Sigma|$ considering the following matrix:

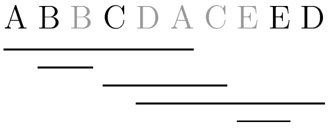
Definition 11 *Let w be a 1R2C-PPW instance. The $(|\Sigma| \times (n-1))$-matrix $A = (a_{\sigma i})$ is defined with respect to the interval set I of w by*

$$a_{\sigma i} := \begin{cases} 1 \text{ , } \textit{if } x_\sigma \leq i < y_\sigma \\ 0 \text{ , } \textit{otherwise, where } \sigma \in \Sigma. \end{cases}$$

This way, every column of A corresponds to a possible position of a color change in w (see Figure 4 for an example).

Recall that there must be an odd number of color changes between the two occurrences of any letter $\sigma \in \Sigma$ in w. Thus, any solution $x \in \{0,1\}^n$ of $(A, \vec{1})x \equiv \vec{0} \bmod 2$ with $x_n = 1$ corresponds to a feasible solution of the 1R2C-PPW (we denote by $\vec{1}$ ($\vec{0}$) the vector of all ones (zeros) of appropriate dimension).

Theorem 12 *Each 1R2C-PPW instance using an alphabet Σ has a solution with at most $|\Sigma|$ color changes.*



$$\left(\begin{array}{ccccccccc}
1 & \mathbf{1} & \mathbf{1} & \mathbf{1} & 1 & 0 & 0 & \mathbf{0} & 0 \\
0 & \mathbf{1} & \mathbf{0} & \mathbf{0} & 0 & 0 & 0 & \mathbf{0} & 0 \\
0 & \mathbf{0} & \mathbf{0} & \mathbf{1} & 1 & 1 & 0 & \mathbf{0} & 0 \\
0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & 1 & 1 & 1 & \mathbf{1} & 1 \\
0 & \mathbf{0} & \mathbf{0} & \mathbf{0} & 0 & 0 & 0 & \mathbf{1} & 0
\end{array}\right)\begin{array}{c}
\mathrm{A} \\ \mathrm{B} \\ \mathrm{C} \\ \mathrm{D} \\ \mathrm{E}
\end{array}$$

Fig. 4. A 1R2C-PPW instance and the associated matrix A.

PROOF. By induction on $|\Sigma|$ and since we can assume that the first column of A is the unit vector e_1, it is easy to see that the rank of A is $|\Sigma|$. Any solution of the 1R2C-PPW corresponds to a linear combination of columns of A yielding $\vec{1}$ over GF(2). If there exists such a linear combination then clearly there also is one using at most $|\Sigma|$ columns. $\square$

$|\Sigma|$ color changes are required for words $w = \sigma_1\sigma_1\sigma_2\sigma_2\ldots\sigma_{|\Sigma|}\sigma_{|\Sigma|}$ which contain the two occurrences of each letter $\sigma_i \in \Sigma$ consecutively. Thus, there are cases where the bound is strict.

5 Solvable instances

Despite the complexity results of Section 3, some instances of the 1R2C-PPW can be solved in polynomial time if we translate the problem into matroid theory. Again we consider the binary matrix A introduced in the last section.

Interpreting each column of $(A, \vec{1})$ as an element of a binary vector matroid $M[A, \vec{1}]$, we are faced with the problem of finding a shortest circuit in $M[A, \vec{1}]$ that contains the element $\vec{1}$.

Problem 13 *Let* $M = (E, \mathcal{I})$ *be a connected binary matroid with a special element* l *and a distance function* $d : E \setminus l \to \mathbb{N}_{\geq 0}$. *Find a circuit* C *of* M *containing* l *such that* $\sum_{e \in C} d(e)$ *is minimized.*

Problem 13 is called *binary clutter problem* and can be shown to be $\mathcal{NP}$-hard (see [11]). However, it is well-known that, for regular matroids, it can be solved by linear programming. For MFMC matroids and their duals, it can be solved by matroid decomposition. The existence of an F_7 minor or its dual within a matroid can be checked in polynomial time. A practical and useful implementation of a more general minor-checking algorithm is available as a part of the MACEK project (see [7]).

Theorem 14 ([11]) *Problem 13, and therefore also the 1R2C-PPW, is solv-*

able in polynomial time if $M[A, \vec{1}]$ is a regular matroid, or an MFMC matroid, or the dual of an MFMC matroid.

Note that A is totally unimodular, because A has the consecutive ones property for rows (see [6]). Thus, $M[A, \vec{1}]$ is a one-point-extension of a regular matroid. If A is graphic, then the binary clutter problem is a *T-join problem* and, thus, solvable in polynomial time. For cographic A, the binary clutter problem contains the $\mathcal{APX}$-complete *MAX-CUT problem* as a special case. In view of the fact that any regular matroid can be decomposed from graphic, cographic and a particular extra matroid one might hope that some day a constant factor approximation algorithm for the 1R2C-PPW is revealed.

5.1 MaxFlow-MinCut duality

In order to derive a duality result we need the following notion:

Definition 15 *Let $M = (E, \mathcal{I})$ be a matroid with a specific element $l \in E$ and a distance function $d : E \to \mathbb{N}_{\geq 0}$. We call a set $S = \{C_1, \ldots, C_k\}$ of cocircuits of M a* coflow *through l of value k, if both (a) $C_i \cap C_j = \{l\}$ for all $C_i, C_j \in S$ and $C_i \neq C_j$, and (b) $|\{C_i : e \in C_i\}| \leq d(e)$ for all $e \in E$.*

Informally speaking, a coflow is a disjoint packing of cocircuits if we disregard the element l. We cite the dualized version of Seymour's famous MaxFlow-MinCut theorem for binary matroids.

Theorem 16 ([10]) *Let $M = (E, \mathcal{I})$ be a binary matroid with a specific element $l \in E$. Then, M has no F_7 minor that contains l if and only if the minimum length of a circuit through l equals the maximum value of a coflow through l for all distance functions $d : E \to \mathbb{N}_{\geq 0}$.*

In terms of Problem 13, we have $M = M[A, \vec{1}]$, $l = \vec{1}$, and a distance function $d \equiv 1$ except for $d(\vec{1}) = 0$. The disjoint packing of cocircuits consists of GF(2)-sums (i. e., symmetric differences) of rows of A. These have to be odd, as each cocircuit must contain the element $\vec{1}$. Thus, we get the following.

Theorem 17 *If $M[A, \vec{1}]$ has no F_7 minor that contains $\vec{1}$, the minimum number of color changes for a 1R2C-PPW instance equals the maximum value of a disjoint odd row sum packing of rows of the corresponding matrix A.*

Example 18 *The minimum number of color changes for the instance shown in Figure 4 is 4, and $M[A, \vec{1}]$ contains no F_7 minor. A maximum disjoint odd row sum packing is given by $\{A\Delta B\Delta C\}$, $\{C\Delta D\Delta E\}$, $\{B\}$, and $\{E\}$.*

6 Summary and open problems

We have answered one open problem stated in [4]. It turns out that the 1R2C-PPW, a very restricted version of the PPW, is still $\mathcal{NP}$-complete, and even $\mathcal{APX}$-hard. This brings the next open question in [4] into focus: whether there is a constant factor approximation algorithm for the 1R2C-PPW. One possibility to answer this question to the affirmative would be if one could show, that the lower bound from Theorem 10 yields at least a constant fraction of the optimum value.

Furthermore, we have shown that the 1R2C-PPW is equivalent to the problem of finding a shortest circuit in a certain class of binary matroids, which allows the solution of specific instances within polynomial time. In this context it would be interesting if, in addition to Theorem 17, not only the value of a disjoint odd row sum packing, but also the disjoint odd row sum packing itself can be computed efficiently. We do not know how to do this even for regular matroids $M[A, \vec{1}]$.

References

[1] P. Alimonti and V. Kann. Some $\mathcal{APX}$-completeness results for cubic graphs. *Theoretical Computer Science*, 237(1–2):123–134, 2000.

[2] S. Arora, C. Lund, R. Motwani, M. Sudan, and M. Szegedy. Proof verification and the hardness of approximation problems. *Journal of the ACM*, 45(3):501–555, 1998.

[3] Th. Epping, W. Hochstättler, and M. E. Lübbecke. Maxflow-mincut duality for a paint shop problem. In U. Leopold-Wildburger, F. Rendl, and G. Wäscher, editors, *Operations Research Proceedings 2002*, pages 377–382, Berlin, 2003. Springer.

[4] Th. Epping, W. Hochstättler, and P. Oertel. Complexity results on a paint shop problem. To appear in: Discrete Applied Mathematics.

[5] M. C. Golumbic. *Algorithmic graph theory and perfect graphs.* Academic Press, 1980.

[6] R. L. Graham, M. Grötschel, and L. Lovasz, editors. *Handbook of Combinatorics (Volume II).* Elsevier, 1995.

[7] P. Hlineny. The macek project (matroids also computed efficiently kit). `http://www.fpv.umb.sk/~hlineny/macek/`.

[8] J. G. Oxley. *Matroid theory.* Oxford University Press, 1997.

[9] C. H. Papadimitriou and M. Yannakakis. Optimization, approximation and complexity classes. *Journal of Computer and System Sciences*, 43(3):425–440, 1991.

[10] P. D. Seymour. The matroids with the max-flow min-cut property. *Journal of Combinatorial Theory*, B(23):189–222, 1977.

[11] K. Truemper. *Matroid decomposition*. Academic Press, 1992. `http://emis.math.metu.edu.tr/monographs/md/`.