Collapsing Words*

A. Cherubini[1], A. Kisielewicz[2]

[1] Politecnico di Milano, Dipartimento di Matematica, Piazza Leonardo da Vinci 32, 20133 Milano, Italy,
alessandra.cherubini@polimi.it

[2] University of Wrocław, Department of Mathematics and Computer Science, pl. Grunwaldzki 2, 50-384 Wrocław, Poland,
kisiel@math.uni.wroc.pl

Abstract. Given a word w over a finite alphabet Σ and a finite deterministic automaton $\mathcal{A} = \langle Q, \Sigma, \delta \rangle$, the inequality $|\delta(Q, w)| \leq |Q| - n$ means that under the natural action of the word w the image of the state set Q is reduced by at least n states. A word w is n-collapsing if this inequality holds for any deterministic finite automaton that satisfies such an inequality for at least one word. In this paper we prove that the problem of recognizing n-collapsing words is generally co-NP-complete, while restricted to 2-collapsing words over 2-element alphabet it belongs to P. This is connected with introducing a new approach to collapsing words, which is shown to be much more effective in solving various problems in the area. It leads to interesting connections with combinatorial problems concerning solving systems of permutation conditions on one hand, and coloring trees with distinguished nodes on the other hand.

1 Introduction

In this paper by an automaton $\mathcal{A} = \langle Q, \Sigma, \delta \rangle$ we mean a finite deterministic automaton with state set Q, input alphabet Σ, and transition function $\delta : Q \times \Sigma \to Q$. The action of Σ on Q given by δ will be denoted simply by concatenation: $qa = \delta(q, a)$. This action extends naturally on the action of the words of Σ^* on Q. Given a word $w \in \Sigma^*$, we are interested in the cardinality $|Qw|$ of the image of Q by w.

If $|Qw| = 1$, then w is called a *reset* word for $\mathcal{A}$, and $\mathcal{A}$ itself is called *synchronizing*. According to the famous Černý's conjecture, if $\mathcal{A}$ is synchronizing, then it has a reset word of length $\leq (m-1)^2$, where $m = |Q|$ is the number of states in $\mathcal{A}$. This conjecture was formulated in 1964, and it is probably the most longstanding open problem in the theory of finite automata. For interesting applications and recent results concerning this conjecture we refer the reader to [4] and references given therein. Problems with settling Černý's conjecture, on one hand, and its importance for the theory of finite automata and finite semigroups, on the other, suggest a need of more systematic approach to the

* This research was done with the support of GNSAGA and of ESF project Automatha.

problem. Such approach was initiated in papers [1, 8] basing on the earlier work [11].

Generally, we are now interested in how the set of states is reduced under the action of various words. Given $w \in \Sigma^*$, the difference of the cardinalities $|Q| - |Qw|$ is called the *deficiency* of the word w with respect to $\mathcal{A}$ and denoted $\mathrm{df}_{\mathcal{A}}(w)$. For $n \geq 1$, a word w is called *n-compressing* for $\mathcal{A}$, if $\mathrm{df}_{\mathcal{A}}(w) \geq n$. An automaton $\mathcal{A}$ is *n-compressible*, if there exists an n-compressing word for $\mathcal{A}$. A word $w \in \Sigma^*$ is *n-collapsing* (over Σ), if it is n-compressing for every n-compressible automaton with the input alphabet Σ.

It has been proved in [11] that n-collapsing words always exist, for any Σ and any $n \geq 1$. In [8] it is shown that, over a fixed alphabet Σ, each n-collapsing word is *n-full*, that is, it contains any word of length n among its subwords. Surveys of results and problems in this area are given in [4] and in [5]. In particular, a result showing that the problem of recognizing n-collapsing words is decidable (and is in the class co-NP) is proved in [9] and the question whether this problem is co-NP-complete is formulated ([4, Problem 1]).

In [1] a characterization of 2-compressing words was given by associating to every word a family of finitely generated subgroups in some finitely generated free groups; it was proved that the property of being 2-collapsing is connected with the indices of some subgroups in this context. A more geometric version of this idea has been developed in [2]. Some further results in this direction are contained in [3, 10]. Unfortunately, these characterizations did not allow neither to settle the complexity of the above problem nor to generalize to n-collapsing words (cf. remarks in [4]).

In this paper we apply another more combinatorial approach to collapsing words, which was introduced in [6]. It made possible to answer a number of open questions formulated in [1], and attack related problems concerning reset words ([7]). In this paper we present the full solution to the central problem of complexity which was announced and shortly described in [6] and [7].

Our main motivation in proving the NP-completeness result is preparing a good starting point for further research by showing that (in view of this result) certain characterizations are not available here. Our proof exhibits a very tight connection with two other computational problems: one is connected with solving certain conditions on permutations, a sort of systems of permutation equations (Section 3), and another one concerns coloring trees with distinguished nodes satisfying some natural conditions (Section 5). These problems seem interesting by themselves and are important from computational point of view.

First, a characterization of 2-collapsing words in terms of solving systems of permutation conditions is given in Section 3. This characterization is used both for directly designing an efficient algorithm for recognizing 2-collapsing words over a 2-element alphabet (Section 4), and for demonstrating that in other cases the problem is intractable. The proof of this fact occupies the remaining sections.

In our approach, we view an automaton $\mathcal{A} = \langle Q, \Sigma, \delta \rangle$ as a set of transformations labeled by letters of Σ rather than as a standard triple. By *transformations of $\mathcal{A}$* we mean those transformations of Q that are induced via δ by letters of

Σ. Note that to define an automaton it is enough to assign just to any letter of Σ a transformation of Q. The monoid (semigroup) generated by the transformations of $\mathcal{A}$ consists precisely of the transformations corresponding to words in Σ^* (Σ^+). Those transformations that are permutations, if any, generate a group called the *group of permutations of* $\mathcal{A}$.

For $a \in \Sigma$, $\mathrm{df}_{\mathcal{A}}(a) = 0$ if and only if the corresponding transformation is a permutation of Q. If $\mathrm{df}_{\mathcal{A}}(a) = 1$, then there is a uniquely determined state $z \in Q$, which does not belong to the image Qa, and two different states $x, y \in Q$ satisfying $xa = ya$; in such a case the corresponding transformation will be referred to as a *transformation of type* $\{x,y\}\backslash z$ (x, y identified, z missing). More generally a transformation a of $\mathcal{A}$ is called *of type* $I\backslash M$, for I, M subsets of Q, if I is the set of those $x \in Q$ for which there is $y \in Q, y \neq x$ such that $xa = ya$, and $M = Q \setminus Qa$. Our idea is that this is essentially all information we need to compute the deficiency of any word.

Note that cardinalities of I and M are related; in particular, $|I| \leq 2|M|$, and the equality holds whenever no three different elements of Q has the same image under a. Note also that the deficiency is a nondecreasing function of factor relation, in the sense that, if $w = v_1 u v_2$, then $\mathrm{df}_{\mathcal{A}}(u) \leq \mathrm{df}_{\mathcal{A}}(w)$. In particular, if $w = a_1 a_2 \dots a_n$, then

$$\mathrm{df}_{\mathcal{A}}(a_1) \leq \mathrm{df}_{\mathcal{A}}(a_1 a_2) \leq \dots \leq \mathrm{df}_{\mathcal{A}}(a_1 a_2 \dots a_n).$$

2 Decidability

For a fixed alphabet Σ, and fixed $n > 1$, let $\mathcal{C}_n$ denote the language of n-collapsing words. For some time it was not even clear that n-collapsing words can be recognized, i.e., that the language $\mathcal{C}_n$ is recursive. In fact, this is the main result announced in [4], where a large sketch of the proof of this fact is given. The full proof in [9] consists of several lemmas and occupies more than 10 pages. Our new approach makes possible to obtain a shorter elementary proof.

Theorem 1. *For each word $w \in \Sigma^*$ that fails to be n-collapsing there exists an n-compressible automaton $\mathcal{A}$ satisfying $\mathrm{df}_{\mathcal{A}}(w) < n$ whose number of states $|Q|$ is less than $5n|w|$.*

Proof. Suppose that w is not n-collapsing, and let $\mathcal{A} = \langle Q, \Sigma, \delta \rangle$ be an n-compressible automaton such that $\mathrm{df}_{\mathcal{A}}(w) < n$. Concerning the actual deficiency on word w, we may assume that $\mathrm{df}_{\mathcal{A}}(w) = n - 1$ (since in case of need one may add new states $q_0, q_1, \dots, q_k$ to Q, all transformed into q_0 by all the transformations, and in such a way suitably increase the deficiency on all the words). Our aim is to construct an automaton $\mathcal{A}' = \langle Q', \Sigma, \delta' \rangle$ over the same alphabet Σ, with $\mathrm{df}_{\mathcal{A}'}(w) = n - 1$, and $\mathrm{df}_{\mathcal{A}'}(v) \geq n$ for some $v \in \Sigma^*$, and such that $Q' \subseteq Q$ is small enough.

Let $w = \gamma_1 \gamma_2 \dots \gamma_t$ with $\gamma_i \in \Sigma$. We define the *partial deficiency sets* D_j of w as the sets of elements missing in the partial images $Q\gamma_1\gamma_2 \dots \gamma_{j-1}$. More precisely, we define $D_1 = \emptyset$, and for $1 \leq j \leq t$, if γ_j is a permutation, then $D_{j+1} = (D_j)\gamma_j$, and if γ_j is a non-permutation of type $I \backslash M$, then

$$D_{j+1} = M \cup D_j\gamma_j \setminus (I \setminus D_j)\gamma_j$$

(cf. Figure 1). Note that the sets D_j are fully determined by some partial information on transformations restricted to a certain subset Q' of Q.

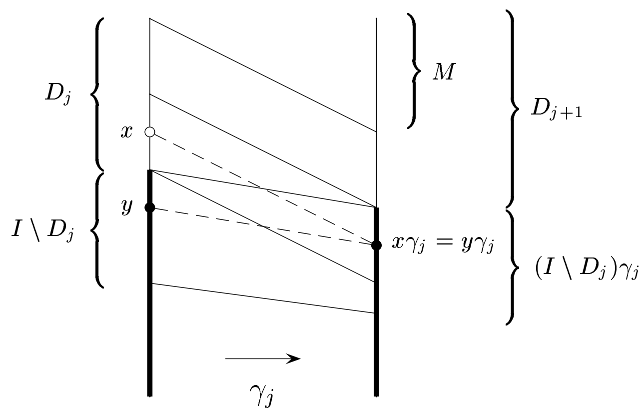

Fig. 1. Scheme of a non-permutation transformation.

Our idea is to keep the deficiency sets of w unchanged with respect to $\mathcal{A}' = \langle Q', \Sigma, \delta' \rangle$. To this end it is enough to put the following for every letter γ_j in w:

(i) γ_j acts on D_j in the same way in $\mathcal{A}'$ as in $\mathcal{A}$, that is $\delta'(x, \gamma_j) = y$ whenever $x \in D_j$ and $x\gamma_j = y$;
(ii) if γ_j is a non-permutation of type $I_j \backslash M_j$ in $\mathcal{A}$, then γ_j acts on I_j in the same way in $\mathcal{A}'$ as in $\mathcal{A}$, that is $\delta'(x, \gamma_j) = y$ whenever $x \in I_j$ and $x\gamma_j = y$;

We note that when we omit the pairs (x, y) given by (ii), then the remaining pairs $(x, x\gamma_j)$ forms a 1-1 correspondence between the sets $Q \setminus I_j$ and $Q \setminus (M_j \cup I_j\gamma_j)$. If we omit further pairs given by (i), for various occurrences of letter γ_j in w, then we have still a 1-1 correspondence between certain subsets of Q, and what are exactly the pairs in this 1-1 correspondence is irrelevant for the sets D_j. We make use of this fact to remove irrelevant elements from Q, and obtain an automaton $\mathcal{A}'$ with a state set $Q' \subseteq Q$ and with required properties.

Namely, let Q' consist of those elements x and y that occur in (i) or (ii) for any γ_j, $1 \leq j \leq t$. More precisely, we put

$$Q' = \bigcup_{1 \leq j \leq t} (D_{j+1} \cup I_j \cup I_j\gamma_j)$$

(of course, if γ_j is a permutation, we put $I_j = \emptyset$; note that we have $I_i = I_j$ whenever $\gamma_i = \gamma_j$).

We define a new action of γ_j on Q' as follows. For a fixed j, let $\gamma_{i_1} = \gamma_{i_2} = \ldots = \gamma_{i_s}$ represent all the occurrences of the letters γ_j in w (where $s = s(j)$ depends on j, and let $D_j^* = D_{i_1} \cup D_{i_2} \cup \ldots D_{i_s}$). We agree first that the new action of γ_j on the set $I_j \cup D_j^*$ is exactly the same as in the old action on Q, that is $\delta'(x, \gamma_j) = y$ whenever $x \in I_j \cup D_j^*$ and $\delta(x, \gamma_j) = y$. By the remark above the sets

$$Q \setminus (I_j \cup D_j^*) \ \text{ and } \ Q \setminus (M_j \cup (I_j \cup D_j^*)\gamma_j)$$

are equinumerous, and therefore the sets

$$Q' \setminus (I_j \cup D_j^*) \ \text{ and } \ Q' \setminus (M_j \cup (I_j \cup D_j^*)\gamma_j)$$

are equinumerous, as well (note that $M_j \subseteq D_{j+1} \subseteq Q'$). We complete the definition of the new action of γ_j on Q' by choosing any 1-1 correspondence ϕ_j between sets $Q' \setminus (I_j \cup D_j^*)$ and $Q' \setminus (M_j \cup (I_j \cup D_j^*)\gamma_j)$, and setting $\delta'(x, \gamma_j) = \phi_j(x)$ for $x \in Q' \setminus (I_j \cup D_j^*)$.

It should be now clear that the deficiency sets of w with respect to $\mathcal{A}' = \langle Q', \Sigma, \delta' \rangle$ are exactly the same at those with respect to $\mathcal{A}$. In particular, $\mathrm{df}_{\mathcal{A}'}(w) = |D_{t+1}| = \mathrm{df}_{\mathcal{A}}(w) < n$.

To estimate the size of Q', note that by assumption $|D_j| \leq n - 1$, and recall that $|I_j| \leq 2|M_j|$ and $M_j \subseteq D_{j+1}$, for all j. It follows that $|Q'| \leq 4(n-1)|w|$.

Yet, we still need to ensure that $\mathcal{A}'$ is n-compressible, and to this end we may need to enlarge Q' a little. By n-compressibility of $\mathcal{A}$, it follows that there exists a word u such that $\mathrm{df}_{\mathcal{A}}(u) \geq n$. Consequently, $\mathrm{df}_{\mathcal{A}}(wu) \geq n$, and we assume that u is the shortest word with this property.

Since $\mathrm{df}_{\mathcal{A}}(w) \leq n$, there are $x, y \in Q'w = Q' \setminus D_{t+1}$ such that $xu = yu$. For $u = \delta_1 \ldots \delta_s$, denote $x_1 = x$, and $x_{i+1} = x_i\delta_i$, and similarly, $y_1 = y$, and $y_{i+1} = y_i\delta_i$, for all $1 \leq i \leq s$. Note that $x_{s+1} = xu = yu = y_{s+1}$. If all $x_i, y_i \in Q'$, then the automaton $\mathcal{A}'$ defined above satisfies $\mathrm{df}_{\mathcal{A}'}(wu) \geq n$, and so it is as required.

Hence, we suppose first that all $x_i \in Q'$, but there is j such that $y_j \notin Q'$, and all $y_i \in Q'$ for all $i > j$. Note that $j < s$, since $y_s\tau_s = x_s\tau_s = y_{s+1}$ and therefore all $x_s, y_s, y_{s+1} \in Q'$. Further, since u is the shortest with $\mathrm{df}_{\mathcal{A}}(wu) \geq n$, all $x_1, \ldots, x_j$ are pairwise distinct and belong to D_{t+1}. In particular, $j \leq n - 1$. Now, we add j new states to Q', say, $Q'' = Q' \cup \{p_1', \ldots, p_j'\}$, and rather than completing the definition of the new action of γ_j on Q' we complete this definition on the set Q'' by choosing a suitable 1-1 correspondence between the sets $Q'' \setminus (I_j \cup D_j^*)$ and $Q'' \setminus (M_j \cup (I_j \cup D_j^*)\gamma_j)$ such that the following j independent conditions hold: $p_{i+1} = \delta'(p_i, \tau_i)$ for all $1 \leq i < j$, and $\delta'(p_j, \tau_i) = y_{j+1}$. Since $y_j \notin Q'$, such a 1-1 correspondence obviously exists. The resulting automaton on Q'' has the same deficiency sets on w as $\mathcal{A}$, and $p_1u = x_1u$. Consequently, $\mathrm{df}_{\mathcal{A}'}(wu) \geq n$, as required. The cardinality $|Q''| \leq 5(n-1)|w| + n - 1$, which (taking into account that for $|w| < n$ our statement is trivial) yields the required bound.

Finally, if there are j and k such that $y_j, x_k \notin Q'$, then assuming that $k \leq j$, we may simply take $Q' \cup \{x_k\}$ to get the previous case. Up to symmetry, this exhausts all the possibilities, thus completing the proof.

Theorem 1 obviously shows that, for each $n > 1$, the language of n-collapsing words over Σ is recursive (it is always enough to check a finite number of automata to see whether a word is n-collapsing or not). The bound in our theorem is linear both in n and in the length of w with a coefficient $C = 5$, but it is possible to improve it, observing that in fact we do not need all the states in Q', and we need only one y for each x with $x\gamma = y\gamma$, to ensure that the resulting automata have the same deficiency sets. With some additional effort, one could obtain a coefficient as low as $C = 2$, which is the one given by Petrov in [9]. Since the algorithm based on this idea is non-practical anyway, we leave this possible improvement to the reader. Another improvement is an observation that a more precise estimation in our proof yields, in fact, $|Q'| \le (n-1)(|w| + 3|\Sigma|)$.

3 New characterization of 2-collapsing words

The known fact that each collapsing word is n-full suggests the following definition. An n-compressible automaton $\mathcal{A}$ is called *proper* ([1]), if no word of length n is n-compressing for it. In order to decide whether a word $w \in \Sigma$ is n-collapsing it is enough to check whether it is n-full, and if so, whether it is n-compressing for each proper n-compressible automaton.

It is not difficult to see that the n-collapsing words over a one-element alphabet are simply the words of length larger or equal to n, while 1-collapsing words over any alphabet are simply 1-full words (i.e. those involving all the letters).

From now on our study is focused on 2-collapsing words over an finite alphabet Σ of cardinality greater than 1. We start from a classification of proper 2-compressible automata, which has been established in [1]. We rephrase it (together with the arguments) in the language of our approach.

Obviously, any n-compressible automaton $\mathcal{A}$ has at least one non-permutation transformation. Yet, for $\mathcal{A}$ to be proper 2-compressible, non-permutation transformations have to satisfy quite strong conditions. In order to formulate and prove a suitable result, first we note that a proper 2-compressible automaton cannot have any transformation with deficiency larger than 2. Indeed, in such a case a suitable single letter forms a 2-compressing word. Also no transformation of type $\{x, y\}\backslash z$ with $z \notin \{x, y\}$ is allowed. This is because composing such a transformation with itself yields the deficiency larger than one, which means that a suitable subword of the form α^2 is 2-compressing. Furthermore, if we have two transformations of type $\{x_1, y_1\}\backslash z_1$ and $\{x_2, y_2\}\backslash z_2$, then we may assume that $z_1 \in \{x_2, y_2\}$ (and $z_2 \in \{x_1, y_1\}$); otherwise a suitable word of the form $\alpha\beta$ is 2-compressing.

These remarks show that there are the following two possibilities for a proper 2-compressible automaton: either all non-permutation transformations are of the same type, and in this case there are x, y such that each non-permutation transformation is of type $\{x, y\}\backslash x$, or there are at least two non-permutation transformations of different types. In the latter case we have again two possibilities: either there is x such that each non-permutation transformation is of type $\{x, z\}\backslash x$ for some z, or there are x, y such that each non-permutation transformation is of type $\{x, y\}\backslash x$ or $\{x, y\}\backslash y$.

None of these conditions is sufficient for an automaton to be proper 2-compressible. In each case there have to be permutation transformations in order to form a transformation corresponding to a 2-compressing word. It follows, in particular, that a proper 2-compressible automaton has both non-permutation and permutation transformations. All the remarks above are rephrasing of the results established in Section 2 of [1]. We go a step further. Namely, it is not difficult to see that in each of these cases there is a suitable necessary condition on the group of permutations which makes the whole condition sufficient. This is quite obvious after the discussion above, so we simply formulate the result.

Proposition 1. *An automaton $\mathcal{A}$ is proper 2-compressible if and only if $\mathcal{A}$ satisfies one of the following conditions:*

(i) there are x, y such that all non-permutation transformations are of the same type $\{x, y\}\backslash x$, and the group of permutations fixes neither the element x nor the set $\{x, y\}$;
(ii) there is x such that each non-permutation transformation is of type $\{x, z\}\backslash x$ for some z, at least two different types occur, and the group of permutations does not fix x;
(iii) there are x, y such that each non-permutation transformation is of type $\{x, y\}\backslash x$ or $\{x, y\}\backslash y$, both the types occur, and the group of permutations does not fix the set $\{x, y\}$.

Let us note that in the classification in [1], the automata in cases (i) and (ii) are called MONO, and those satisfying (iii) are called STEREO.

Now, we wish to show that for a word $w \in \Sigma^*$ being 2-collapsing over an alphabet Σ is equivalent to the nonexistence of nontrivial solutions to certain systems of conditions on permutations. Consider partitions of Σ into blocks, where blocks are intended to represent types of transformations and closely correspond to the role assignments introduced in [1]. A nontrivial partition $\{P, B_2, \ldots, B_h\}$ of Σ with a distinguished block P will be called a *DB-partition* and will be denoted by (P, Υ), where $\Upsilon = \{B_2, \ldots, B_h\}$ is the induced partition of $\Sigma \setminus P$ ($h \geq 2$). Roughly speaking the letters in P are intended to represent permutation transformations and letters in B_i are intended to represent non-permutation transformation of the type $\{1, i\}\backslash 1$ for $1, i$ fixed states of Q. Let w be a 2-full word over Σ. To each factor of w of the form $\alpha v \beta$, where v is a nonempty word whose all letters belong to P (i.e. $v \in P^+$), while $\alpha \notin P$ and $\beta \in B_j$, we assign a permutation condition of the form

$$1v \in \{1, j\},$$

where the letters of P are treated as permutation variables. Thus, the condition means that the image of 1 under the product v of permutations belongs to the set $\{1, j\}$. The resulting set of permutation conditions (containing all conditions corresponding to factors of w with the properties described above) will be denoted $\Gamma_w(P,\Upsilon)$ and referred to as the system of permutation conditions determined by a word w and a DB-partition (P,Υ). Note that different orderings of blocks in $\{B_2,\dots,B_h\}$ lead to systems which are "equivalent" in the sense that one can be obtained from the other just by renaming the variables; so we do not care of the orderings of blocks.

We say that this system *has a solution* if there exists an assignment of permutations on a finite set $\{1,2,\dots,N\}$ to letters in P such that all the conditions in $\Gamma_w(P,\Upsilon)$ are satisfied. A *trivial* solution is one with all permutations fixing 1. Also, in the special case when Υ consists of a unique block B_2 (and in consequence, all j's on the right hand side of the conditions are equal 2), solutions with all permutations fixing the set $\{1,2\}$ are considered trivial. The remaining solutions are *nontrivial*.

A partition $(P,\{B_1,B_2\})$ of Σ (into exactly 3 blocks, with a distinguished block P) will be called a *3DB-partition*. Again the letters in P are intended to represent permutation transformations while letters in $B_i, i=1,2$ are intended to represent non-transformation permutations of the type $\{1,2\}\backslash i$ for $1,2$ fixed states of Q. For such partition, we define another system of permutation conditions as follows. To each factor of w of the form $\alpha v\beta$, with $\alpha\in B_i, \beta\in B_j$, $i,j\in\{1,2\}$, and $v\in P^+$, we assign a permutation condition of the form

$$iv\in\{1,2\}$$

(the image of i under v belongs to $\{1,2\}$). The resulting set of permutation conditions will be denoted by $\Gamma'_w(P,\{B_1,B_2\})$. For such a system, a solution in permutations is *nontrivial* if the image of the set $\{1,2\}$ does not remain fixed under all the permutations.

Theorem 2. *A word $w\in\Sigma^*$ is 2-collapsing if and only if it is 2-full and the following conditions holds:*

(i) $\Gamma_w(P,\Upsilon)$ has no nontrivial solution for any DB-partition (P,Υ) of Σ;
(ii) $\Gamma'_w(P,\{B_1,B_2\})$ has no nontrivial solution for any 3DB-partition $(P,\{B_1,B_2\})$ of Σ.

The reader may observe an explicit similarity with the characterization in [1, Theorem 3.3]. Yet while, indeed, there is a correspondence in the general structure, our approach is almost converse: rather then looking into an algebraic structure behind, we reduce the problem to the simplest conditions on permutations.

Proof. We prove the „only if" part. We know that if w is 2-collapsing, then it is 2-full. Suppose by contradiction that the system $\Gamma_w(P,\Upsilon)$ has a nontrivial solution for some DB-partition (P,Υ), and that this solution consists of permutations on a set $Q=\{1,2,\dots,N\}$. We build an automaton $\mathcal{A}$ over Σ with the set Q of states as follows: the letters in P act as the permutations given in the solution, and the letters in each of blocks $B_i\in\Upsilon$ act as (arbitrary) transformations of type $\{1,i\}\backslash 1$. Since the solution is nontrivial, the group of permutations does not fix 1, and in case when Υ consists of one block B_2, the group of permutations does not fix the set $\{1,2\}$, either. Thus, by Proposition 1, in each case $\mathcal{A}$ is a proper 2-compressible automaton (of type MONO). To get a contradiction we show that $\mathrm{df}_{\mathcal{A}}(w) = 1$ (which means that w is not 2-compressing for $\mathcal{A}$, and hence not 2-collapsing).

First, suppose that w has no factor of the form $\alpha v \beta$, with $v \in P^+$, $\alpha \in B_i$ and $\beta \in B_j$ (which means that the system $\Gamma_w(P,\Upsilon)$ is empty, and permutations in the solution are restricted only by the condition of being nontrivial). Then w is of the form $v\alpha_{i_1}\ldots\alpha_{i_m}u$ with $v,u \in P^*$ and $\alpha_{i_j} \notin P$, and we show that $\mathrm{df}_{\mathcal{A}}(w) = 1$. Indeed, for the first segment with permutation variables we have $\mathrm{df}_{\mathcal{A}}(v) = 0$, and next, $\mathrm{df}_{\mathcal{A}}(v\alpha_{i_1}) = 1$. Since the type of α_{i_1} is $\{1,i\}\backslash 1$ for some i, then 1 is missing in the image $Qv\alpha_{i_1}$. Since the type of α_{i_2} is $\{1,i\}\backslash 1$ for some i, α_{i_2} simply permutes the elements in $Qv\alpha_{i_1}$; the deficiency set is again $\{1\}$, and it is easy to see that the same happens at every step. Consequently, $\mathrm{df}_{\mathcal{A}}(w) = 1$, as claimed.

Now, let $\alpha v\beta$ be a factor of w with $v \in P^+$, $\alpha \in B_i$ and $\beta \in B_j$, and assume that it is the first factor of this form in w. It follows that $w = s\alpha v\beta t$, where $s,t \in \Sigma^*$, and by the previous argument $\mathrm{df}_{\mathcal{A}}(s\alpha) = 1$, and 1 is missing in the image $Qs\alpha$. Now, since v is nonempty, the permutation condition $1v \in \{1,j\}$ is in $\Gamma_w(P,\Upsilon)$. It means that 1 is moved into 1 or j by v, and consequently it is 1 or j that is missing in the image $Qs\alpha v$. Since β identifies 1 and j, $\mathrm{df}_{\mathcal{A}}(s\alpha v\beta) = 1$, and it is again 1 that is missing in the image $Qs\alpha v\beta$. Repeating this argument several times we see the deficiency does not decrease, and consequently $\mathrm{df}_{\mathcal{A}}(w) = 1$, which is the required contradiction.

As the second case we assume that the system $\Gamma'_w(P,\{B_1,B_2\})$ has a nontrivial solution for some 3DB-partition $(P,\{B_1,B_2\})$. The proof is analogous, and we only point out the differences. Here, we build an automaton $\mathcal{A}$ where letters in each of the two blocks B_i act as arbitrary transformations of type $\{1,2\}\backslash i$. Since the solution is nontrivial, the group of permutations does not fix the set $\{1,2\}$. Thus, in this case, $\mathcal{A}$ is a proper 2-compressible automaton (of type STEREO). If w has no factor of the form $\alpha v\beta$, with $v \in P^+$, $\alpha \in B_i$ and $\beta \in B_j$, then the same argument as before shows that $\mathrm{df}_{\mathcal{A}}(w) = 1$. If it has one, and $w = s\alpha v\beta t$ exhibits the first factor of this type, then $\mathrm{df}_{\mathcal{A}}(s\alpha) = i$ and i is missing in the image $Qs\alpha$. By the corresponding permutation condition $iv \in \{1,2\}$, and therefore $\mathrm{df}_{\mathcal{A}}(s\alpha v\beta) = 1$ with j missing in the image $Qs\alpha v\beta$. Since $i,j \in \{1,2\}$, we may continue this argument, to get that $\mathrm{df}_{\mathcal{A}}(w) = 1$, as required.

To prove the „if" part, assume by contradiction that w is not 2-collapsing. Then, since by assumption it is 2-full, there has to be a proper 2-compressible automaton $\mathcal{A}$ over Σ, with the set of states $Q = \{1,2,\ldots,N\}$, for which w is not 2-compressing. If $\mathcal{A}$ is of type MONO ((i) or (ii) in Proposition 1), then we consider the DB-partition of Σ, where P represents permutations of $\mathcal{A}$, and $B_2,\ldots,B_h$ represent transformations of types $\{1,2\}\backslash 1$, …, $\{1,h\}\backslash 1$, respectively (we assume without loss of generality that $x = 1$ is the distinguished state). The fact that w is not 2-compressing for $\mathcal{A}$ means that deficiency does not decrease, except for the first initial segment of the form $v\alpha$ with $v \in P^*$ and $\alpha \in B_i$, for some i. The only segments where the deficiency could decrease are those of the form $\alpha v \beta$, with $\alpha \in B_i, \beta \in B_j$ and $v \in P^+$. The fact that the deficiency does not decrease on these segments is equivalent to that the permutations satisfy corresponding conditions $1v \in \{1, j\}$, as required. The solution they form is nontrivial because of respective conditions (i) or (ii) in Proposition 1.

If $\mathcal{A}$ is of type STEREO (Proposition 1 (iii)), we consider the 3DB-partition of Σ, where P represents again permutations of $\mathcal{A}$, and B_1, B_2 represent transformations of types $\{1,2\}\backslash 1$ and $\{1,2\}\backslash 2$, respectively, and we use the same argument as before.

4 2-element alphabet

In this section we consider the simplest nontrivial case of $\Sigma = \{a, b\}$. In this case we have only two DB-partitions, those into two singletons. The corresponding two systems $\Gamma_w(P, \Upsilon)$ are each in one variable, and to present them in detail we introduce additional notation. Let us define

$$E_a(w) = \{k \geq 1 \, : \, ba^k b \text{ is a factor of } w\};$$
$$E_b(w) = \{k \geq 1 \, : \, ab^k a \text{ is a factor of } w\}.$$

Then, depending on whether $P = \{a\}$ or $P = \{b\}$, the system $\Gamma_w(P, \Upsilon) = \Gamma_a(w)$ is one of the following

$$\Gamma_a(w) = \{1a^k \in \{1,2\} \; : \; k \in E_a(w)\}$$
$$\Gamma_b(w) = \{1b^k \in \{1,2\} \; : \; k \in E_b(w)\}$$

By Theorem 2 a word $w \in \{a,b\}^*$ is 2-collapsing if and only if it is 2-full and none of the systems $\Gamma_a(w)$ or $\Gamma_b(w)$ has a nontrivial solution. Nontrivial solution is, in this case, a single permutation a or b, respectively, fixing neither 1 nor $\{1,2\}$. Whether such a solution exists or not depends only on the sets of integers $E_a(w)$ or $E_b(w)$ defined above, and the conditions have purely arithmetical form.

Theorem 3. *A word $w \in \{a,b\}^*$ is 2-collapsing if and only if it is 2-full and for all $E = E_a(w)$ or $E_b(w)$, $n \geq 3$, and $r < n$, the set E modulo n is not contained in $\{0, r\}$.*

Proof. In view of Theorem 2 it is enough to prove that one of the systems $\Gamma_a(w)$ or $\Gamma_b(w)$ has a nontrivial solution if and only if there are $n \geq 3$ and $r < n$ such that $E = E_a(w)$ or $E_b(w)$ is contained in $\{0, r\}$ modulo n.

Suppose first that the system $\Gamma_a(w)$ has a nontrivial solution. In this case the solution is a single permutation a, which we consider as a product of disjoint cycles. If 1 and 2 are in the same cycle, then this cycle is of length $n \geq 3$, since a does not fix the set $\{1,2\}$. We may assume that $a = (1, \ldots, 2, \ldots, n) \ldots$, with 2 standing on $(r+1)$-th place, $0 < r < n$. Then the conditions $1a^k \in \{1,2\}$ mean that either $k \equiv 0$ or $k \equiv r \pmod{n}$, for each $k \in E_a(w)$. It follows that

$E_a(w) \subseteq \{0, r\}$ modulo n, as required. If 1 and 2 are in different cycles of a, then the cycle containing 1 has length $n \geq 2$, since a does not fix 1. For $n = 2$, the conditions means that $k \equiv 0 \pmod 2$ for each $k \in E_a(w)$, which is equivalent to $E_a(w) \subseteq \{0, 2\}$ modulo 4. For $n > 2$, we obtain $E_a(w) \subseteq \{0\}$ modulo n, which completes the case of the system $\Gamma_a(w)$. For $\Gamma_b(w)$ the proof is analogous.

Conversely, if for instance $E_a(w) \subseteq \{0, r\}$ modulo n ($n \geq 3$, $r < n$), then the permutation $a = (1, \ldots, 2, \ldots, n) \ldots$, with 2 standing on $(r+1)$-th place is obviously a nontrivial solution of the system $\Gamma_a(w)$. The same argument applies to $E_b(w) \subseteq \{0, r\}$.

We note that the condition on n in the theorem can be restricted to n not exceeding the value of the second smallest element in E. This is so, because for larger n the two smallest elements are two different non-zero remainders modulo n, and therefore E cannot be contained modulo n in any set $\{0, r\}$. Taking this into account we have the following algorithm for checking whether a word $w \in \{a, b\}^*$ is 2-collapsing.

if w is not 2-full then **return** NO;
for all $E \leftarrow \{k \geq 1 : ba^k b \text{ is a factor of } w\}$
or $E \leftarrow \{k \geq 1 : ab^k a \text{ is a factor of } w\}$ do
    if $|E| > 1$ then
        $N \leftarrow$ the second smallest element in E
        for $n \leftarrow 1$ to N do
            $E \leftarrow E \bmod n$
            if $E \subseteq \{0, r\}$ for some $r < n$ then **return** NO
return YES

Since the sum of the elements in E is smaller than the length $|w|$ of w, we obtain the following

Corollary 1. *For a 2-element alphabet Σ, checking whether a word $w \in \Sigma^*$ is 2-collapsing may be done in polynomial time with respect to $|w|$.*

We note that another characterization of 2-collapsing words over a 2-element alphabet, based on the general result of [1], is given in [10, Proposition 3]. This also can be used to infer the corollary above. The fact that 2-collapsing words over a 2-element alphabet may be recognized in polynomial time was also obtained in [2] as a consequence of a general algorithm to check whether a word is 2-collapsing.

5 Related computational problems

We proceed to show that for the case $|\Sigma| = 3$ the situation is essentially different. In this case we have three types of DB-partitions and one type of 3DB-partition. In order to check whether a word $w \in \Sigma^*$ is 2-collapsing or not, we need to check all the corresponding systems $\Gamma_w(P, \Upsilon)$ and $\Gamma'_w(P, \{B_1, B_2\})$, according to

Theorem 2. The cases with $|P| = 1$ are still easy; they lead to systems in one variable, and one may prove that all these systems can be solved in polynomial time. The case $|P| = 2$ leads to the system $\Gamma_w(P, \Upsilon)$ in two variables of the form

$$\Gamma(u_1, \ldots, u_s) = \{1u_1, \ldots, 1u_s \in \{1,2\}\}, \qquad u_i \in \{\beta, \gamma\}^*.$$

and the following PERMUTATION CONDITIONS problem:

INSTANCE: A finite set of words $\{u_1, \ldots, u_s\}$ over a 2-element alphabet $\Sigma = \{\beta, \gamma\}$.
QUESTION: Does the corresponding system $\Gamma(u_1, \ldots, u_s)$ of permutation conditions in two variables have a nontrivial solution: i.e. are there permutations β and γ fixing neither $\{1\}$ nor $\{1, 2\}$ satisfying all the conditions of this system?

Our first aim is to show that this problem is NP-complete.

Permutations are generally not easy to visualize. Therefore we convert the above problem into a problem concerning coloring of a binary tree with distinguished nodes. Namely, we consider trees representing words $u \in \{\beta, \gamma\}^+$, assuming that edges going to the left child represent applying permutation β, while edges going to the right child represent applying permutation γ. To represent a set $U = \{u_1, \ldots, u_s\}$ of words we take the minimal binary tree in which all words $u_1, \ldots, u_s$ are represented. The nodes representing these words together with the root form the set of *distinguished nodes* of the tree. The resulting tree with distinguished nodes will be denoted $T(U) = T(u_1, \ldots, u_s)$.

A 1-2-*coloring* of a binary tree T with distinguished nodes is a coloring of the nodes with positive integers such that each distinguished node has color either 1 or 2. The root is always colored 1. The coloring is *nontrivial*, if there is a color different from 1 and 2. It is *coherent*, if for any two nodes s, t having the same color the following conditions hold: if both s and t are left (right) children then their parents have the same color; if both s and t have left (right) children, then these children have the same color. Nontrivial coherent 1-2-colorings are called briefly *nc-colorings*. We have the following

Lemma 1. *Let $u_1, \ldots, u_s$ be words over alphabet $\{\beta, \gamma\}$, such that any word of length 2 is a prefix of some u_i, $1 \leq i \leq s$. Then the tree $T = T(u_1, \ldots, u_s)$ has an nc-coloring if and only if the system $\Gamma(u_1, \ldots, u_s)$ has a nontrivial solution.*

Proof. For the „if" part, we assign simply the color $1u$ to the node s in T corresponding to the word u. Then, such a coloring is nontrivial due to the assumption that all nodes representing words of length less or equal 2 are in the tree. Indeed, the children of the root are colored 1β and 1γ. If any of them is different from 1 and 2, we are done; otherwise, at least one of them must be 2, since the permutations do not fix 1. It follows that both colors 2β and 2γ occur in the tree, and since the permutations do not fix $\{1, 2\}$, one of them is different from 1 and 2, as required. The coherency is by the fact that the action of β and γ is the same at each node.

Conversely, if $T = T(u_1, \ldots, u_s)$ has a nontrivial coherent 1-2-coloring, then it determines (at least partially) the action of β and γ on the colors (which is unique by coherency). As a result one can get partial representations of β and γ as products of cycles. One may easily add new colors, absent in the given 1-2-coloring, to form full representations, and thus get a nontrivial solution of the system $\Gamma(u_1, \ldots, u_s)$.

This leads us to the following NC-COLORINGS problem:

INSTANCE: A binary tree $T = T(u_1, \ldots, u_s)$ with distinguished nodes.
QUESTION: Does T have a nontrivial coherent 1-2-coloring?

We consider an example, which is the starting point of our construction. Let

$$W_0 = \{\beta, \beta^3, \beta^4, \beta\gamma^2, \beta\gamma^3, \beta^2\gamma^2\beta, \beta^2\gamma^2\beta^2, \gamma, \gamma^3, \gamma^4, \gamma\beta^2, \gamma\beta^3, \gamma^2\beta^2\gamma, \gamma^2\beta^2\gamma^2\}.$$

The tree $T(W_0)$ representing this set of words is pictured in Figure 2. The distinguished nodes are marked as black filled circles. The labels represent a nontrivial coherent 1-2-coloring. This coloring is *the most general* one in the sense that any other *nc*-coloring of $T(W_0)$ can be obtained from it by suitable identifications of colors. For example, we can identify $z = y$ obtaining another *nc*-coloring; note however that other identifications (for instance $x = y$) may lead to non-coherent or trivial colorings. We have

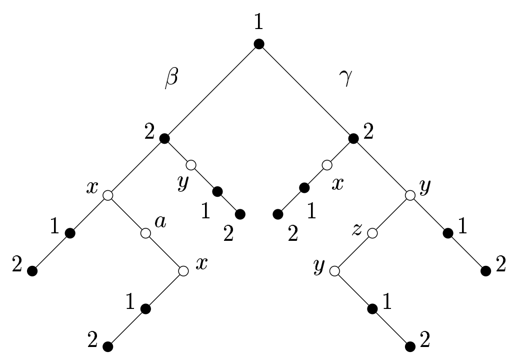

Fig. 2. The most general *nc*-coloring of $T(W_0)$.

Lemma 2. *The coloring in Figure 2 is the most general nc-coloring of* $T(W_0)$. *In each nc-coloring of this tree the colors* x *and* y *are different from 1 and 2.*

Proof. The proof of this lemma is routine. It requires checking a number of cases, using coherency and the fact that distinguished nodes have to be colored 1 or 2. We give only the beginning of the proof. First note that the nodes corresponding to β and γ (the children of the root) are distinguished, so they have to be colored 1 or 2. We consider possible cases. If both these nodes are colored 1, then since by assumption the root is colored 1, by coherency, both the left and the right children of any node colored 1 are colored the same, and consequently all the nodes in the tree are colored 1, which yields a trivial coloring.

As the second case we assume that β is colored 1 and γ is colored 2. Then, by coherency, nodes corresponding to β^2, β^3, and β^4 are all colored 1. Now $\beta\gamma$ is colored 2 (the same as γ^2), $\beta\gamma^2$ and γ^2 are colored the same, 1 or 2 (since $\beta\gamma^2$ is distinguished). Consequently, the nodes corresponding to γ^n are colored 1 if n is even, and 2 if n is odd, which by coherency yields again a trivial coloring. The third case is symmetrical, and it follows that in a nontrivial coherent coloring both the children of the root have to be colored 2. The next step is to exclude the cases that one of the nodes labeled x or y in Figure 2 is colored 1 or 2. This and the completion of the proof is left to the reader.

It follows that in any system of permutation conditions containing $\Gamma(W_0)$ the nontrivial solutions are all of the form

$$\beta = (12x)(yz)\ldots, \quad \gamma = (12y)(xa)\ldots$$

Further cycles involving further colors can occur in β and γ. We note that the action of β on a and the action of γ on z are not determined by the conditions so far. We make use of this fact in showing that 3SAT can be reduced to NC-COLORINGS.

6 General construction

For each instance of 3SAT consisting of a collection $\{C_1, \ldots, C_r\}$ of clauses on a finite set $(x_1, \ldots, x_n)$ (with $|C_j| = 3$) we construct a binary tree

$$\mathbf{T} = \mathbf{T}(C_1, \ldots, C_r, x_1, \ldots, x_n)$$

with distinguished nodes such that it has an *nc*-coloring if and only if there is a truth assignment satisfying all the clauses. The tree $\mathbf{T}$ consists of the main part $M(\mathbf{T})$, the path $P(\mathbf{T})$ attached to $M(\mathbf{T})$, and $3r$ paths attached to some distinguished nodes of $P(\mathbf{T})$. The $3r$ paths correspond to all occurrences of variables in the instance of 3SAT (cf. Figure 3). Each of these paths consists of 3 segments bounded and determined by distinguished nodes: the *variable segment*, the *negation segment*, and the *clause segment*. The nodes on $P(\mathbf{T})$ beginning variable segments will be referred to as the *starting nodes*.

The tree is constructed so that in any *nc*-coloring the starting nodes have always the same fixed color. The distinguished nodes finishing variable segments are called the *variable valuation nodes*; their color may be 1 or 2, and is intended to reflect the valuation, FALSE or TRUE, of variables in the 3SAT instance. The negation segment may be missing; it is present if and only if the corresponding occurrence of variable is negated in the clause it occurs. In such a case the

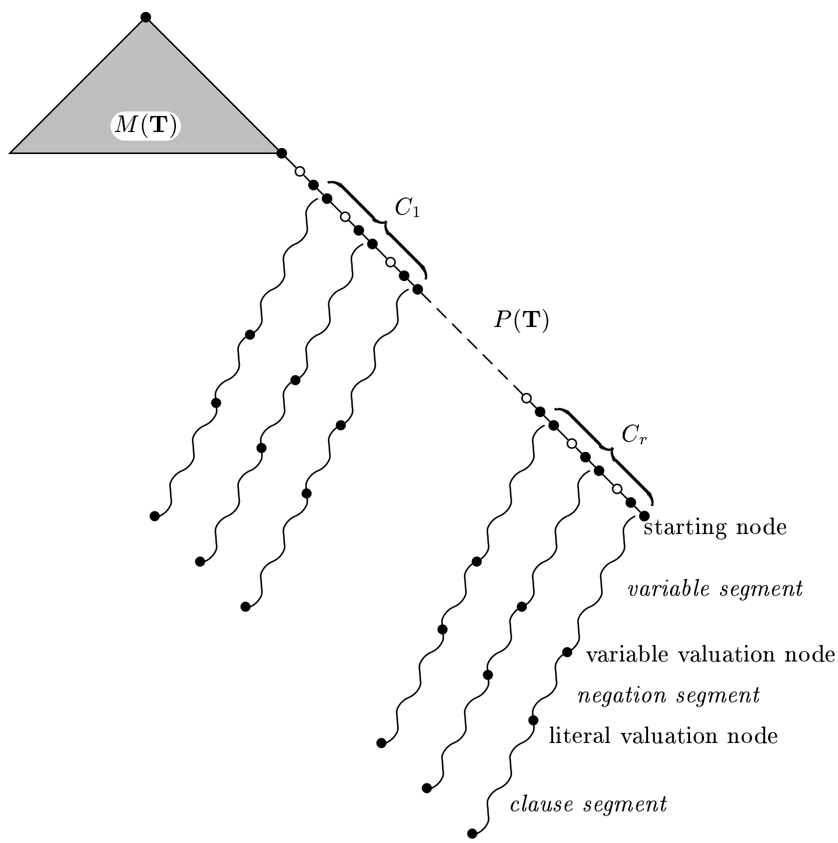

Fig. 3. General scheme of **T**.

distinguished node finishing the negation segment is called the *literal valuation node*, and its color, in any nc-coloring (due to properties of the segment) is always opposite to that of the preceding variable valuation node. In case of lacking the negation segment the literal valuation node is identified with the variable valuation node.

7 Structure of permutations

The main part $M(\mathbf{T})$ is an extension of $T(W_0)$ described in the previous section. This implies, in particular, that a nontrivial solution of the corresponding system of permutation conditions needs to be of the form $\beta = (12x)(yz)\ldots$, and $\gamma = (12y)(xa)\ldots$. We add further parts to $T(W_0)$ corresponding to further permutation conditions, which determine much more details in the structure of β and γ. Our aim is to make β in a nontrivial solution to be the product of cycles of the form

$$\beta = (12x)(yz)(aa_1 \ldots a_n a_1^1 a_2^1 a_3^1 \ldots a_1^r a_2^r a_3^r)(bb_1 \ldots b_n b_1^1 b_2^1 b_3^1 \ldots b_1^r b_2^r b_3^r) \ldots \qquad (1)$$

$$\ldots (A_1^j c_j d_j e_j) \ldots (B_1^j B_2^j B_3^j) \ldots (A_i X_i) \ldots (B_i Y_i) \ldots \ldots,$$

where $i = 1, \ldots, n$, $j = 1, 2, \ldots, r$ (with n and r given in the 3SAT instance). Thus, β is intended to consist of two cycles of length 3 and 2, respectively, two long cycles of length $N = 1+n+3r$ each, r cycles of length 4, r additional cycles of length 3, and n cycles $(A_i X_i)$ which may be of length 2 or 1 (since we will allow in this case that $X_i = A_i$). Generally, all the letters are intended to represent different elements (colors), except that $X_i, Y_i \in \{A_i, B_i\}$, and $\{A_1^j c_j d_j e_j\} = \{A_1^j A_2^j A_3^j A_0^j\}$, where A_0^j are extra colors, not mentioned yet. The latter condition means that the elements $\{A_1^j A_2^j A_3^j A_0^j\}$ are to be arranged in an arbitrary cycle; we did not mention cycles $(B_i Y_i)$ because they are either one-element or are supposed to coincide with a suitable cycle $(A_i B_i)$.

We intend γ to be of the form

$$\gamma = (12y)(xa)(zb) \ldots (a_i A_i) \ldots (b_i B_i) \ldots (a_t^j A_t^j) \ldots (b_t^j B_t^j) \ldots, \qquad (2)$$

where $i = 1, \ldots, n$, $j = 1, 2, \ldots, r$, and $t = 1, 2, 3$. Thus it is intended to consist of one cycle of length 3, and $2 + n + 3r$ cycles of the length 2.

To achieve this we add to the tree $T(W_0)$ additional binary branches corresponding to additional words. First we add to W_0 two additional words $\beta^2\gamma\beta^N\gamma\beta$ and $\beta^2\gamma\beta^N\gamma\beta^2$, where $N = 1+n+3r$. This results in attaching to $T(W_0)$ an additional path of length $N+3$ with 2 distinguished nodes; it is shown in Figure 4 as one starting in the node labeled a. This construction includes a trick we wish to describe now, since we apply it repeatedly while adding further branches.

Let us consider an nc-coloring of the tree constructed so far, pictured in Figure 4. The colors of the nodes in the part corresponding to $T(W_0)$ are already determined by earlier consideration and are copied from Figure 3. For the colors

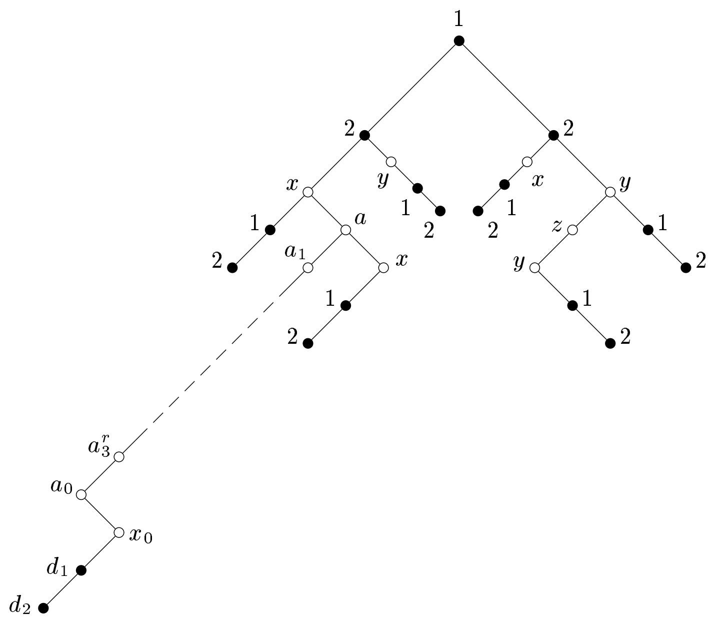

Fig. 4. The first additional path attached to $T(W_0)$.

of the nodes on the additional path we will use letters $a_1, \ldots, a_n$, $a_1^1, a_2^1, a_3^1, \ldots,$ a_1^r, a_2^r, a_3^r (corresponding to the intended long cycle in β), and a_0, x_0, d_1, d_2 for the last four nodes. We prove that the latter have to be, in fact, $a, x, 1$ and 2, respectively. Indeed, the last node is distinguished, so it has to be colored 1 or 2; by coherency, the parent of a node colored 1 is colored x, which by Lemma 2 is different from 1 and 2; hence, by coherency, it has to be $d_2 = 2$, and consequently, $d_1 = 1, x_0 = x$ and $a_0 = a$, as claimed.

Thus we have forced the color a_0 to be the same as a in any nc-coloring. Observe that this is achieved by attaching a path to the node labeled a_0, which is the same as that attached to the node labeled a, and whose coloring is determined completely by coherency. In the construction of $M(\mathbf{T})$, which is pictured in Figure 5, we apply this trick several times. In Figure 5 it is shown by arrows: an arrow attached to a node labeled ℓ symbolizes the path attached to the earlier node having the same label; the direction of the arrow shows the direction of the first edge in the path. In consequence, the colors of the two nodes have to be the same in any nc-coloring. Dashed lines in Figure 5 mean that the gadgets attached to all nodes a_i, a_t^j and b_i, b_t^j on these paths ($i = 1, \ldots, n$, $i = j, \ldots, r$, $t = 1, 2, 3$) are analogous. A careful analysis of this construction yields the following:

Lemma 3. *The coloring presented in Figure* 5 *is the most general nc-coloring of the pictured tree $M(\mathbf{T})$.*

We show how to obtain the nc-coloring corresponding to the permutations β and γ of the form given in (1) and (2). First we identify X_i and Y_i with A_i and B_i

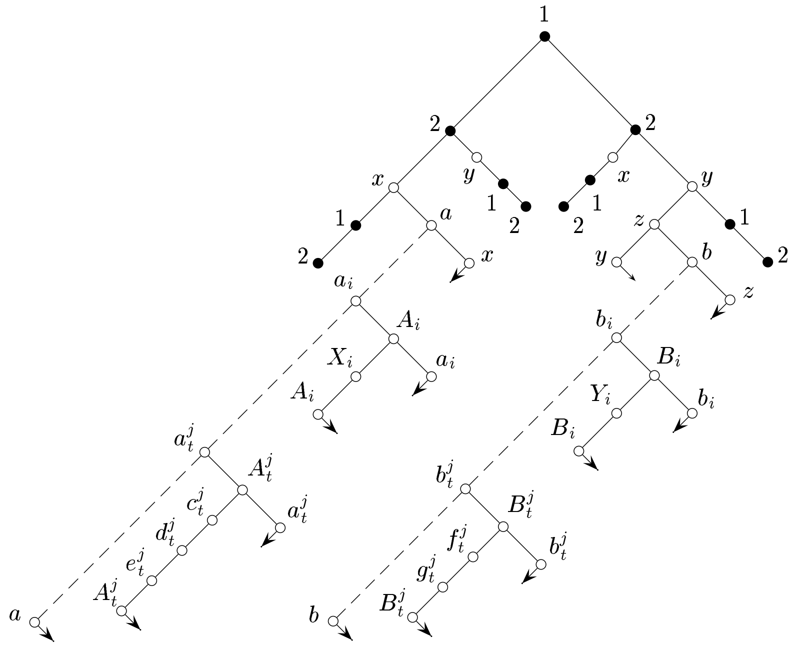

Fig. 5. The part $M(\mathbf{T})$ with the most general nc-coloring.

choosing for each i either the equalities $X_i = B_i$ and $Y_i = A_i$, which correspond to the cycle (A_iB_i) in β, or $X_i = A_i$ and $Y_i = B_i$, which correspond to the fix-points $(A_i)(B_i)$ in β. In the sequel we refer to these as to (A_iB_i)- and $(A_i)(B_i)$-*identifications*, respectively. Next, we identify colors $f_1^j = B_2^j$ and $g_1^j = B_3^j$, this identification by coherency uniquely determines identifications of colors $f_2^j = B_3^j$, $g_2^j = B_1^j$, $f_3^j = B_1^j$, and $g_3^j = B_2^j$, and corresponds to the cycle (B_1^j, B_2^j, B_3^j) in β. Similarly, if we choose any identification of colors $\{c_1^j, d_j^1, e_1^j,\}$ with $\{A_2^j, A_3^j, A_0^j\}$, then by coherency this determines uniquely identifications of colors c_t^1, d_t^1, e_t^1 for $t = 1, 2, 3$, and this is equivalent with arranging letters $\{A_1^j, A_2^j, A_3^j, A_0^j\}$ into an arbitrarily chosen cycle. Of course, we can do it independently for each $j = 1, \ldots, n$, and this corresponds to fixing n cycles of length 4 in β. In the sequel we refer to the above identifications as to (B_1^j, B_2^j, B_3^j)- and $\{A_1^j, A_2^j, A_3^j, A_0^j\}$-*identifications*, respectively.

8 Encoding truth assignments

The second part denoted $P(\mathbf{T})$ is just the path corresponding to set of words

$$\{\gamma^7, \gamma^8, \ldots, \gamma^{3r+4}, \gamma^{3r+5}\}.$$

This path has $2r$ distinguished nodes, and continues the pattern of distinguished nodes started in $M(\mathbf{T})$: two consecutive distinguished nodes follow one, which is not distinguished (see Figure 3). In consequence, in any nc-coloring the distinguished nodes on $P(\mathbf{T})$ have to be colored 1 and 2, alternately. The *starting nodes* on $P(\mathbf{T})$ are those labeled 2.

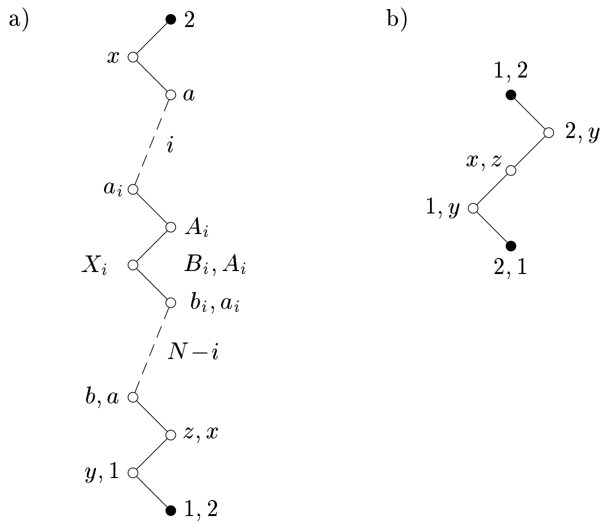

Fig. 6. Variable and negation segments.

As it was already mentioned, each path beginning in a starting node corresponds to an occurrence of a variable x_i in a clause C_j. It consists of three or two parts, depending on whether the variable occurs negated in C_j or not. The variable segment is always of the form

$$C(k) = T(\beta\gamma\beta^i\gamma\beta\gamma\beta^{N-i}\gamma\beta\gamma),$$

where i is the index of the variable x_i and $N = 1+n+3r$ as before (see Figure 6a). Since in any nc-coloring the color of the starting node is 2, further colors in these segments are also determined by the structure of $M(\mathbf{T})$ and coherency. Thus, the next two nodes have to be colored x, and a, respectively, and the next one, according to the most general coloring given in Figure 5, has to be colored a_i. The next node is colored A_i, and the next one X_i. We show that $X_i = A_i$ or B_i. Indeed, starting from the other end the distinguished variable valuation node has to be colored 1 or 2. It follows, by coherency, that the alternatives for preceding nodes are $(y, 1)$, (z, x), and (b, a), respectively, as shown in Figure 6a. Then, since the length of the dashed β-path is $m = N - i$, the color of the node beginning this path, according to the most general coloring given in Figure 5, is b_i or a_i, respectively. This implies that the color of the preceding node is B_i or A_i, as claimed.

Let us note that choosing the color 1 for the variable valuation node corresponds to identify $X_i = B_i$ (and consequently, $Y_i = A_i$), while choosing color 2 for this node corresponds to identify $X_i = A_i$. By coherency, the variable valuation nodes corresponding to occurrences of the same variable x_i must have the same color, since the variable segments for these occurrences are identical. So generally (A_iB_i)- and $(A_i)(B_i)$-identifications considered before correspond to choose 1 or 2, respectively, as color of vertex x_i.

For negated occurrences of variables we have the negated segment, which we put to be the path $T(\gamma\beta^2\gamma)$ drawn in Figure 6b. We leave to the reader to check that, due to the main part $M(\mathbf{T})$, it works exactly as assumed: in any nc-coloring of $\mathbf{T}$, the first node of this path is colored 1 if and only if the last node is colored 2.

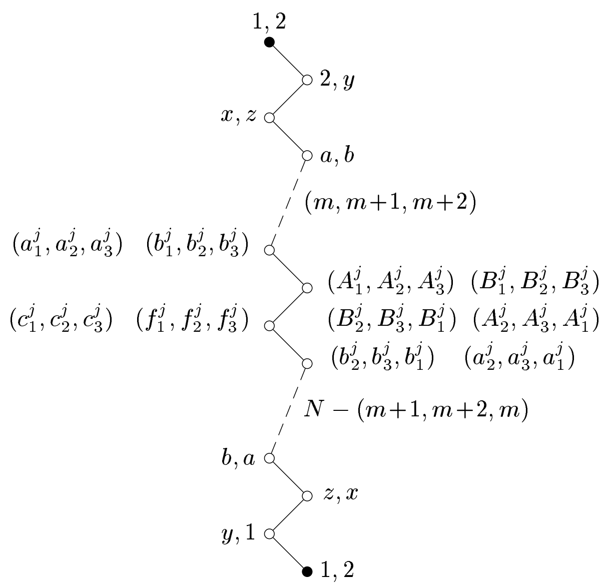

Fig. 7. Colors on the clause segment.

Finally, the clause segments corresponding to variables occurring in a clause C_j are similar to variable segments. They differ only in that they have an additional γ-edge at the beginning, and the lengths of long β-paths depend only on the clause the variable occurs in. Namely, all the three clause segments corresponding to the clause C_j have the form

$$C(k, \ell) = T(\gamma\beta\gamma\beta^k\gamma\beta\gamma\beta^{N-\ell}\gamma\beta\gamma),$$

where $N = 1+n+3r$, as before, and for the three successive variables occurring in C_j the clause segments are:

$$C(m, m+1), C(m+1, m+2), C(m+2, m),$$

respectively, where $m = 1+n+3(j-1)$. They are intended to allow on the corresponding literal valuation nodes any triple of colors from $\{1, 2\}$ except $(1, 1, 1)$ (which is the unique valuation corresponding to the false value of the clause!).

Similarly as for the variable segment, we consider possibilities for an *nc*-coloring of the clause segments; this is illustrated in Figure 7.

Similarly as for the variable segment, we consider possibilities for a nontrivial coherent 1-2-coloring of the clause segments (see Figure 7). For the first node, which is a literal valuation node, we have two possibilities 1 or 2. Since at the beginning the three segments are identical, the alternatives for successive nodes are the same: $(2, y), (x, z)$, and (a, b), respectively. Then, the segments differ slightly in the length of the $T(\beta^k)$-path, so we consider further alternatives in form of triples with entries corresponding to the values $k = m, m+1, m+2$ with $m = 1+n+3(j-1)$. From the most general coloring of $M(\mathbf{T})$ given in Figure 5, for the node ending the $T(\beta^k)$-path the alternative of colors is (a_1^j, a_2^j, a_3^j) or (b_1^j, b_2^j, b_3^j), respectively. Then, for the two next nodes the alternatives are (A_1^j, A_2^j, A_3^j) or (B_1^j, B_2^j, B_3^j), and (c_1^j, c_2^j, c_3^j) or (f_1^j, f_2^j, f_3^j), respectively. Since for now, we know nothing about the latter colors, we consider colors of other nodes from the other end of the segment.

The last node (as distinguished) may be colored 1 or 2, and the alternatives for preceding nodes are $(y, 1)$, (z, x), and (b, a), as before (cf. Figure 6 and Figure 7). Now, since the length of the $T(\beta^{N-\ell})$-path is $N-(m+1)$, $N-(m+2)$ or $N-m$, respectively, with $m = 1+n+3(j-1)$, the alternatives for the node starting the $T(\beta^{N-\ell})$-path are (b_2^j, b_3^j, b_1^j) or (a_2^j, a_3^j, a_1^j), respectively. Hence, the alternatives for the preceding node are just (B_2^j, B_3^j, B_1^j) or (A_2^j, A_3^j, A_1^j), respectively, which impose some conditions on the values of $c_1^j, c_2^j, c_3^j, f_1^j, f_2^j, f_3^j$, depending on valuation of literal valuation nodes.

For example, if the literal valuation node corresponding to the first occurrence of a variable in C_j is colored 1, then (in any nontrivial coherent 1-2 coloring) it has to be $c_1^j = B_2^j$ or A_2^j. We show that the first possibility is excluded. Indeed, if $c_1^j = B_2^j$, then by coherency, $d_1^j = f_2^j$, $e_1^j = g_2^j$, and $A_1^j = B_2^j$. It follows further that $a_1^j = b_2^j$, and $a = b_1$; and further, $x = B_1$, $2 = Y_1$, $1 = B_1$, which yields , the contradiction $x = 1$. A similar argument shows that if the literal valuation node corresponding to the second or third occurrence of a variable in C_j is colored 1, then $c_2^j = A_3^j$ or $c_3^j = A_1^j$, respectively.

In particular, if all the literal valuation nodes corresponding to the same clause C_j are colored 1, then it follows that the $T(\beta^4)$-paths going from A_t^j, $t = 1, 2, 3$, are all colored with the same color $A_1^j = A_2^j = A_3^j$. This leads to a contradiction as before (it follows successively: $a_1^j = a_2^j$, $a = a_1$, $x = A_1$, and $1 = A_1$). This proves the first statement of the following.

Lemma 4. *In any nc-coloring, the literal valuation nodes corresponding to the same clause C_j cannot be all colored* 1*; any other coloring of these nodes may occur.*

For the second statement, it is enough to apply first the (B_1^j, B_2^j, B_3^j)-identification, and then to choose a suitable $\{A_1^1, A_2^1, A_3^1, A_0^1\}$-identification. For example, if the successive literal valuation nodes corresponding to clause C_j are colored $(1, 2, 1)$, then in Figure 7 we have colors $(c_1^j, f_2^j, c_3^j) = (A_2^j, B_3^j, A_1^j)$. Here A_2^j has to be the color following A_1^j, B_3^j has to be the color following B_2^j (which is now the case), and A_1^j has to be the color following A_3^j. It is the $(A_1^1, A_2^1, A_0^1, A_3^1)$-identification that satisfies these conditions.

Summarizing, if we have given any nc-coloring of $\mathbf{T}$, then the colors of variable valuation nodes determine, in a coherent way, the colors assigned to variables $x_1, \ldots, x_n$, which are always 1 or 2. The colors of literal valuation nodes are reverse or the same depending on whether the variable is negated in a clause or not, and it never happens that the literal valuation nodes corresponding to the same clause have all color 1. It follows that if we treat 1 as the false value, and 2 as the true value, then the nc-coloring yields a truth assignment for $(x_1, \ldots, x_n)$ satisfying all the clauses.

Conversely, for any truth assignment for $(x_1, \ldots, x_n)$ satisfying all the clauses, there exists an nc-coloring of $\mathbf{T}$ corresponding to it. This is simply the most general coloring given in Figure 5 with suitable (A_iB_i)- and $(A_i)(B_i)$-identifications, the (B_1^j, B_2^j, B_3^j)-identification, and suitable $\{A_1^1, A_2^1, A_0^1, A_3^1\}$-identifications. The latter exists, as we have observed, for all clauses valuated true.

Since all those constructions may be done, obviously, in polynomial time, we obtain

Theorem 4. *The problems* PERMUTATION CONDITIONS *and* NC-COLORINGS *are both NP-complete.*

9 Application to collapsing words

By the result of Section 2, we know that if a word $w \in \Sigma^*$ is not n-collapsing then there is an n-compressible automaton $\mathcal{A} = \langle Q, \Sigma, \delta \rangle$ with $|Q| \leq 5n|w|$ such that $\mathrm{df}_{\mathcal{A}}(w) < n$. From this (combined with the fact that n-collapsing words are n-full) it follows that the following general problem of recognizing n-collapsing words is in co-NP.

INSTANCE: A finite alphabet Σ, a word $w \in \Sigma^+$, and an integer $n > 0$.
QUESTION: Is w n-collapsing over Σ?

We now show how 3SAT can be reduced in polynomial time to the above problem. First, we associate, with each instance of 3SAT the tree

$$\mathbf{T} = \mathbf{T}(C_1, \ldots, C_r, x_1, \ldots, x_n)$$

defined in the previous section, and next, the set of words $\{w_1, w_2, \ldots, w_s\} \subseteq \{\beta, \gamma\}^*$ determined by the distinguished nodes of $\mathbf{T}$. We wish to define the word $w = \phi(w_1, w_2, \ldots, w_s)$ over the alphabet $\Sigma = \{\alpha, \beta, \gamma\}$ so that it satisfies the following three conditions:

(i) $\alpha v \alpha$, where no α occurs in v, is a factor of w if and only if $v = w_i$ for some i;
(ii) w contains all factors of the form xyx and xy^2x, for all different $x, y \in \Sigma$, $x \neq \alpha$;
(iii) $\gamma\alpha\beta\gamma$, $\gamma\beta\alpha\gamma$ and $\beta\alpha\gamma\beta$, $\beta\gamma\alpha\beta$ are factors of w.

To this end we put $w = u_1u_2u_3u_4$, where the four segments are defined as follows:

$$u_1 = \alpha w_1 \alpha w_2 \alpha \ldots \alpha w_s \alpha;$$

note that by properties of $\mathbf{T}$, $\alpha\beta\alpha$ and $\alpha\gamma\alpha$ are among factors of u_1, while $\beta\alpha^2\beta$, $\gamma\alpha^2\gamma$ are not;

$$u_2 = \beta\alpha\beta\alpha\beta\alpha^2\beta\alpha\gamma\alpha\gamma\alpha\gamma\alpha^2\gamma;$$

these are simply factors $\beta\alpha\beta, \beta\alpha^2\beta\alpha$ and $\gamma\alpha\gamma, \gamma\alpha^2\gamma$ separated by letter α (the later makes sure that no new factor of the form $\alpha v \alpha$ arises);

$$u_3 = \alpha\gamma\alpha\beta\gamma^2\alpha\beta^2\gamma^2\beta\alpha\gamma\alpha\beta\alpha\gamma\beta^2\alpha\gamma^2\beta^2\gamma\alpha\beta\alpha;$$

i.e. the four words from condition (iii) above separated by other words in such a way that that no new factor of the form $\alpha v \alpha$ arises; note that factors $\alpha\beta\gamma^2\alpha$, $\alpha\beta^2\gamma^2\beta\alpha$, $\alpha\gamma\beta^2\alpha$, and $\alpha\gamma^2\beta^2\gamma\alpha$ occurring in u_3, are (by properties of $\mathbf{T}$) among factors of u_1;

$$u_4 = \beta\gamma\beta^2\gamma^2\beta\gamma;$$

this ensures that all the remaining words of the form xyx and xy^2x, which are those not involving α, are among factors of w.

Now, consider the conditions for $w = \phi(w_1, w_2, \ldots, w_s)$ to be 2-collapsing. To this end we need to consider systems $\Gamma_w(P, \Upsilon)$ and $\Gamma'_w(P, \{B_1, B_2\})$ described in Section 3. First, we observe that (due to properties of w) for the systems of the second type have only trivial solutions.

Indeed, if say $P = \{\alpha\}$, $B_1 = \{\beta\}$, $B_2 = \{\gamma\}$, then since $\beta\alpha\beta$ and $\gamma\alpha\gamma$ are factors of w, the conditions $1\alpha \in \{1, 2\}$ and $2\alpha \in \{1, 2\}$ are in $\Gamma'_w(P, \{B_1, B_2\})$, which means that α fixes the set $\{1, 2\}$. The same argument works for other cases here.

We consider systems of the first type. If $P = \{\alpha\}$ and, say, $\beta \in B_2$ then since $\beta\alpha\beta$ and $\beta\alpha^2\beta$ are factors of w, the conditions $1\alpha \in \{1, 2\}$ and $1\alpha^2 \in \{1, 2\}$ are in $\Gamma_w(P, \Upsilon)$, which means that α fixes either $\{1\}$ or $\{1, 2\}$. In case when B_2 is the unique block in Υ, we are done. Otherwise, $\gamma \in B_3$, and since $\gamma\alpha\gamma$ is a factor of w, $1\alpha \in \{1, 3\}$, which yields that α has to fix 1, as required. If $P = \{\beta\}$, then similar reasoning shows that since $\gamma\beta\gamma$ and $\gamma\beta^2\gamma$ are factors of w, β fixes either $\{1\}$ or $\{1, 2\}$, and since $\alpha\beta\alpha$ is a factor of w, β fixes 1, as required. In view of symmetry between β and γ this exhausts all the possibilities for $|P| = 1$.

If $P = \{\alpha, \beta\}$ then the factors of w, $\gamma\alpha\gamma$ and $\gamma\alpha^2\gamma$ yield that α fixes either $\{1\}$ or $\{1,2\}$, and factors $\gamma\beta\gamma$ and $\gamma\beta^2\gamma$ yield that also β fixes either $\{1\}$ or $\{1,2\}$. To make sure that either both fix $\{1\}$ or both fix $\{1,2\}$ we use the fact that $\gamma\beta\alpha\gamma$ and $\gamma\alpha\beta\gamma$ are factors of w. It follows that $1\beta\alpha, 1\alpha\beta \in \{1,2\}$ which makes impossible that one of them fixes 1, but not 2, while another one fixes $\{1,2\}$, but not 1 alone. (For example, if $1\alpha = 1$ and $1\beta = 2$, then $1\beta\alpha = 2\alpha$). The same argument works for $P = \{\alpha, \gamma\}$.

It follows that the only nontrivial case is that of $P = \{\beta, \gamma\}$. Consequently, $\phi(w_1, w_2, \ldots, w_s)$ is 2-collapsing if and only if $\Gamma(w_1, w_2, \ldots, w_s)$ has a nontrivial solution, which holds if and only if $\mathbf{T} = \mathbf{T}(C_1, \ldots, C_r, x_1, \ldots, x_n)$ has an nc-coloring, which holds if and only if there is a truth assignment for the corresponding 3SAT instance. Thus we have

Theorem 5. *The general problem of recognizing n-collapsing words defined above is co-NP-complete.*

Our proof gives also the result for the variant of the problem with a fixed alphabet on 3 letters. It can be easily modified to get the following

Theorem 6. *The problem of recognizing 2-collapsing words over a fixed alphabet Σ with more than 2 letters is co-NP-complete.*

Proof. The modification of the previous proof is the following. We wish to define the word $w' = \psi(w_1, w_2, \ldots, w_s)$ over alphabet $\Sigma = \{\alpha, \beta, \gamma, \delta, \ldots\}$ with more than 3 letters in such a way that it satisfies the following two conditions:

(i) $\alpha v \alpha$ is a factor of w' if and only if $v = w_i$ for some i (for any word v with no occurrence of letter α);
(ii) w contains all factors of the form xyx, xy^2, and $xyzx$, for $x, y, z \in \Sigma$ pairwise distinct, except for $\alpha\gamma^2\alpha$, $\alpha\beta^2\alpha$, $\alpha\beta\gamma\alpha$, and $\alpha\gamma\beta\alpha$ (unless they are among w_i);

To this end we put $w = v_1v_2v_3v_4v_5$, where $v_1 = u_1u_2u_3$ is the word over $\{\alpha, \beta, \gamma\}$ for U_1, u_2 and u_3 defined before; v_2 consists of all the words of the form $x\alpha x$ and $x\alpha^2 x$ separated by letter α, and finishing with α, for all $x \neq \alpha$; v_3 consists of all the words of the form $x\alpha yx$ and $xy\alpha x$ separated by letter α for all $x \neq \alpha$, $x \neq y$, and $\{x,y\} \neq \{\beta, \gamma\}$; v_4 consists of all the words of the form $\alpha xy\alpha$ for all $x \neq y$, and $\{x,y\} \neq \{\beta,\gamma\}$; and v_5 is the least word containing all the possible words on four letters other than α.

An essential property of w defined in this way is that, if $\alpha v \alpha$ is a factor of w and v has no occurrence of α, then either $v = w_i$ or there is at most one occurrence of β or γ in v. Now, consider the conditions for w being 2-collapsing. As before we prove that all the systems $\Gamma_w(P, \Upsilon)$ and $\Gamma'_w(P, \{B_1, B_2\})$, but one, have only trivial solutions.

For the systems of the second type the argument is the same as before. For the first type we assume first that there is a letter $x \neq \alpha$ not in P. Without loss of generality we may assume that $x \in B_2$. Then for any letter $y \in P$ we have factors xyx and xy^2x in w, which as before means that y fixes either 1 or $\{1,2\}$.

Moreover, if there are at least two blocks in Υ, and $z \in B_3$, then zyz is a factor of w (also if $z = \alpha$), and it follows, as before, that all $y \in P$ fix 1, as required. If there is the unique block B_2 in Υ, then we need to show that either all $y \in P$ fix 1 or all $y \in P$ fix $\{1, 2\}$. To this end we use the fact that $xyzx$ is a factor of w for all $x \neq \alpha$. If $y, z \in P$, then from the facts that each of them fixes either 1 or $\{1, 2\}$, and $1yz, 1zy \in \{1, 2\}$, we infer easily that either both fix 1 or both fix $\{1, 2\}$. This extends obviously to all the elements of P, proving that there are only trivial solutions in this case.

It remains the case when the only letter not in P is α. Then, for all $x \in P \setminus \{\beta, \gamma\}$, $\alpha x \alpha$ and $\alpha x^2 \alpha$ are factors of w, and it follows that x fixes either 1 or $\{1, 2\}$. If the system $\Gamma(w_1, w_2, \ldots, w_s)$ has only trivial solutions, then (because of the factor u_1 of w) the same holds also for β and γ. In such a case factors $\alpha x y \alpha$ of w with $\{x, y\} \neq \{\beta, \gamma\}$ guarantee, as in the previous case, that $\Gamma_w(P, \Upsilon)$ has only trivial solutions. If there is a nontrivial solution (β, γ) for $\Gamma(w_1, w_2, \ldots, w_s)$, then taking all $x \neq \beta, \gamma$ such that they fix both 1 and 2, we obtain a nontrivial solution of $\Gamma_w(P, \Upsilon)$. Indeed, by properties of u_1, $1\beta = 1\gamma = 2$, and by the essential property of w mentioned earlier, all the permutation conditions not corresponding to the factor u_1 involve at most one occurrence of β or γ. Hence, in view of our assumption on other permutations, all these conditions are trivially satisfied. Consequently, w is 2-collapsing if and only if $\Gamma(w_1, w_2, \ldots, w_s)$ has a nontrivial solution, which as before completes the proof.

References

1. D. S. Ananichev, A. Cherubini, and M. V. Volkov, *Image reducing words and subgroups of free groups*, Theor. Comput. Sci. 307, no.1 (2003), 77-92.
2. D. S. Ananichev, A. Cherubini, and M. V. Volkov, *An inverse automata algorithm for recognizing 2-collapsing words*, Lect. Notes Comp. Sci. 2450, Springer, Berlin 2003, 270-282.
3. D. S. Ananichev and I. V. Petrov, *Quest for short synchronizing words and short collapsing words*, WORDS. Proc. 4th Int. Conf., Univ. of Turku, Turku, 2003, 411-418.
4. D. S. Ananichev, I. V. Petrov, and M. V. Volkov, *Collapsing words: A Progress Report*, Lect. Notes Comp. Sci. 3572, Springer, Berlin 2005, 11-21.
5. A. Cherubini, *Synchronizing and Collapsing words*, Milan j.math. 75, (2007), 305-321.
6. A. Cherubini, P. Gawrychowski, A. Kisielewicz, B. Piochi, *A combinatorial approach to collapsing words*, Lect. Notes Comp. Sci. 4162, Springer, Berlin, 2006, 256-266.
7. A. Cherubini, A. Kisielewicz, *Recognizing collapsing words is NP-complete*, in: Proceedings of Descriptional Complexity of Formal Systems (Las Cruces, New Mexico, USA), 2006, 106-117.
8. S. W. Margolis, J.-E. Pin, and M. V. Volkov, *Words guaranteeing minimum image*, Internat. J. Foundations Comp. Sci. 15 (2004) 259-276.
9. I. V. Petrov, *An algorithm for recognition of n-collapsing words*, Theor. Comput. Sci. 391, no.1-2 (2008), 99-108.
10. E. V. Pribavkina, *On some properties of the language of 2-collapsing words*, Lect. Notes Comp. Sci. 3572, Springer, Berlin, 2005, 374-384.

11. N. Sauer and M. G. Stone, *Composing functions to reduce image size*, Ars Combinatoria 1 (1991) 171-176.