Theoretical Computer Science ••• (••••) •••–•••

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Note

Vertex 2-coloring without monochromatic cycles of fixed size is NP-complete

Michał Karpiński

Institute of Computer Science, University of Wrocław, Joliot-Curie 15, 50-383 Wrocław, Poland

ARTICLE INFO

Article history:
Received 28 September 2016
Received in revised form 17 October 2016
Accepted 17 October 2016
Available online xxxx
Communicated by J. Díaz

Keywords:
Graph coloring
Sat
Computational complexity
Monochromatic cycles

ABSTRACT

In this paper we study a problem of vertex two-coloring of an undirected graph such that there is no monochromatic cycle of the given length. We show that this problem is hard to solve. We give a proof by presenting a reduction from the variation of satisfiability (SAT) problem. We show the nice properties of coloring cliques with two colors which plays pivotal role in the reduction construction.

1. Introduction

Vertex coloring problems (VCP) have been studied extensively since the inception of graph theory. In classical form, problem of k-coloring a graph is stated like this: can we color vertices of a graph using k different colors, so that no neighbouring vertices have the same color? It is known that this problem is NP-complete in general case [10], but when $k = 2$, then the problem can be solved efficiently in linear time. VCPs have received much attention in the literature not only for their theoretical aspects and difficulty from the computational point of view, but also for their real world applications, for example in: scheduling [12], timetabling [7], register allocation [5], train platforming [2], frequency assignment [9], communication networks [17] and many other engineering fields.

In this paper we study a variation of the coloring problem. Using only two colors we want to color the vertices, so that there is no monochromatic cycle of the given length. There have been some research in solving a slightly different problem: is there a 2-coloring such that there exists no monochromatic cycles (of any length). This problem can be viewed as partitioning a graph into the two induces forests and it is known to be NP-complete [18] for the undirected graphs. Another result worth mentioning is by Nobinon et al. [13] where authors show that this problem is NP-complete even for oriented graphs. They also give implementation of three exact algorithms and some inapproximability results. The motivation to study this class of problems lies in the economics – 2-coloring without monochromatic cycles can be used in the study of rationality of consumption behavior (see [4,8] and [16]).

Many more papers have been written on the subject of acyclic coloring (or partitioning). Papers relevant to ours include (among many others): [3,15,1,11].

The rest of the paper is organized in the following way: in Section 2 we define notation used in this paper, we also give definitions of studied problems and we state the main theorem. In Section 3 and Section 4 we prove the hardness of

E-mail address: karp@cs.uni.wroc.pl.

http://dx.doi.org/10.1016/j.tcs.2016.10.011
0304-3975/© 2016 Elsevier B.V. All rights reserved.

our coloring problem for cycles of the small length (3 and 4). Later, in Section 5 we generalize ideas used in the previous sections to prove the main theorem. We end the paper with some conclusions and show perspectives for future work.

2. Preliminaries

The purpose of this section is to introduce the reader to the notation used in later sections to give definitions of the studied problems. Let $G = (V, E)$ be an undirected, unweighted graph. A *cycle* in G is a vertex disjoint, closed, simple path in G. We denote $\mathcal{C}_k$ to be a set of all cycles in G of length k. Let $c : V \to \{r, b\}$ be a mapping that for each vertex in V assigns one of two colors (*red* or *blue*). We will call any such c: a *coloring* of graph G. Furthermore, we will say that the given coloring c is *valid*, if a certain predicate $P(c)$ is true. Let K_n be a clique of size n, that is: a graph with n vertices in which every vertex is connected by an edge to any other vertex.

Let $(2, k)$-$\mathcal{COL}$ be the decision problem of whether there exists a valid 2-coloring for given graph. We give the validity predicate $P_k(c)$ below. It is true only if the coloring c does not contain any cycles of size k with vertices of the same color.

$$P_k(c) \equiv \forall Q \in \mathcal{C}_k \exists u, v \in Q \quad c(u) \neq c(v)$$

Formally, our problem can be expressed as:

$$(2, k)\text{-}\mathcal{COL} = \{G : \exists c\, P_k(c)\}$$

We are interested in knowing how hard is the question, whether given graph G belongs to $(2, k)$-$\mathcal{COL}$. In the next two sections we study the simplest variants, that is when $k = 3$ and $k = 4$. Cycle of the size three we call a *triangle*, and of the size four: a *square*.

Let $\mathcal{SAT}$ denote the classical boolean satisfiability problem, namely, it is the set of all boolean formulas in CNF (conjunctive normal form) for which there exists a truth assignment that satisfies it. It is known that this problem is NP-complete [6]. It is also known that a certain variation of $\mathcal{SAT}$ called $\mathcal{NAE}$-$\mathcal{SAT}$ (not-all-equal SAT) is NP-complete [14]. In this variation we impose additional constraint on the satisfying assignment: each clause has at least one literal that is true, and at least one that is false. We denote k-$\mathcal{SAT}$ and k-$\mathcal{NAE}$-$\mathcal{SAT}$ (for $k \geq 3$) to be subsets of $\mathcal{SAT}$ and $\mathcal{NAE}$-$\mathcal{SAT}$ where each clause in given formula has at most k literals (it's in kCNF). For $k < 3$ for both problems there exists polynomial time algorithms that solves them.

We are ready to state the main theorem:

Theorem 1. *For any integer $k \geq 3$, $(2, k)$-$\mathcal{COL}$ is NP-complete.*

In order to prove Theorem 1, we will prove the following theorem:

Theorem 2. *For any integer $k \geq 3$, there exists a polynomially computable function f, such that for any boolean formula ϕ, $\phi \in k$-$\mathcal{NAE}$-$\mathcal{SAT}$ if and only if $f(\phi) \in (2, k)$-$\mathcal{COL}$.*

For completeness we should also mention that $(2, k)$-$\mathcal{COL}$ is in NP. The certificate is just the coloring of graph. Knowing that k is constant we can simply use brute-force algorithm to check if there exists any cycle of length k in both subgraphs of the initial graph induced by two colors.

3. Two-coloring without monochromatic triangles

In this section we prove Theorem 2 for $k = 3$. Let ϕ be a boolean formula in 3CNF with n variables $x_1, \ldots, x_n$ and m clauses $C_1, \ldots, C_m$. We construct the desired graph G_ϕ in the following way. Let us begin by showing an abstract form of G_ϕ. The reduction consists of three gadgets: one for each variable, one for each clause, and one for each *super-edge*. The super-edge $\{u, v\}$ is an edge with a property, that any valid coloring c implies that $c(u) \neq c(v)$. For starters, assume that we already have such edges at our disposal. This is how we would construct G_ϕ: a gadget for variable x consists of two vertices labeled x and $\neg x$ connected by a super-edge. Gadget for clause $C = (u \vee v \vee w)$ consists of a triangle with vertices labeled u, v and w. We connect each literal from variable gadget to its every occurrence in clause gadgets using super-edges. Example is given in Fig. 1 for the formula $\phi = (x_1 \vee \neg x_1 \vee x_2) \wedge (\neg x_2 \vee x_3 \vee \neg x_3)$. Dashed lines represent super-edges. We prove that this is indeed the correct reduction.

Lemma 3. *For any given ϕ in 3CNF:*

$$\phi \in 3\text{-}\mathcal{NAE}\text{-}\mathcal{SAT} \iff G_\phi \in (2, 3)\text{-}\mathcal{COL}$$

Proof. First we assume that $\phi \in 3$-$\mathcal{NAE}$-$\mathcal{SAT}$ and let $\hat{\sigma}(x_1, \ldots, x_n)$ be the truth assignment that certify it. Each vertex with non-negated label x in vertex gadgets is colored red if $\sigma(x) = T$ and blue otherwise. Coloring of every other vertex is forced