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

Contents lists available at ScienceDirect

Theoretical Computer Science

www.elsevier.com/locate/tcs

Better lower and upper bounds for the minimum rainbow subgraph problem ☆

Alexandru Popa

Faculty of Informatics, Masaryk University, Brno, Czech Republic

ARTICLE INFO

Article history:
Received 29 April 2012
Received in revised form 22 April 2014
Accepted 3 May 2014
Available online xxxx
Communicated by T. Erlebach

Keywords:
Approximation algorithms
Combinatorial problems
Minimum rainbow subgraph

ABSTRACT

In this paper we study the *minimum rainbow subgraph problem*, motivated by applications in bioinformatics. The input of the problem consists of an undirected graph with n vertices where each edge is colored with one of the p possible colors. The goal is to find a subgraph of minimum order (i.e. minimum number of vertices) which has precisely one edge from each color class.

In this paper we show a randomized $\max(\sqrt{2n}, \sqrt{\Delta}(1+\sqrt{\ln \Delta/2}))$-approximation algorithm using LP rounding, where Δ is the maximum degree in the input graph. On the other hand we prove that there exists a constant c such that the minimum rainbow subgraph problem does not have a $c \ln \Delta$-approximation, unless $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{O(\log\log n)})$.

1. Introduction

1.1. Motivation

An important problem in computational biology is the *pure parsimony haplotyping problem (PPH)*, introduced by Gusfield in 2003 [9]. The problem input consists of a set $\mathcal{G}$ of p genotypes (i.e. vectors with entries in $\{0, 1, 2\}$) corresponding to individuals in a population. A genotype g is explained by two haplotypes (i.e. vectors with entries in $\{0, 1\}$) h_1 and h_2 if for each entry i, either $g[i] = h_1[i] = h_2[i]$ or $g[i] = 2$ and $h_1[i] \neq h_2[i]$. For example, the genotype $g = 012$ is explained by the haplotypes $h_1 = 010$ and $h_2 = 011$ as $h_1[1] = h_2[1] = g[1] = 0$, $h_1[2] = h_2[2] = g[2] = 1$, $h_1[3] \neq h_2[3]$ and $g[3] = 2$. The goal is to find a set of haplotypes of minimum cardinality which explains the set $\mathcal{G}$ of genotypes. The positions where $g[i] = 2$ are named *ambiguous positions*. If the number of ambiguous positions in each genotype is at most k, then the problem is termed *PPH*(k).

Camacho et al. [5] show that the *PPH*(k) problem for $k \leq O(\log p)$ can be reduced in polynomial time to the *minimum rainbow subgraph (MRS)* problem which we describe next. The input consists of an undirected graph G where each edge is colored with one of the p possible colors. A rainbow subgraph $F \subseteq G$ contains precisely one edge from each color class. The goal of the problem is to find a rainbow subgraph of G which has a minimum number of vertices.

1.2. Previous work

Pure parsimony haplotyping The pure parsimony haplotyping problem was introduced by Gusfield [9]. Hubbell shows that the PPH problem is **NP**-hard [13]. Lancia et al. [16] show that the PPH(k) problem is **APX**-hard for $k \geq 3$ and present a

☆ A preliminary version of the paper was accepted at COCOON 2012.
E-mail address: popa@fi.muni.cz.

http://dx.doi.org/10.1016/j.tcs.2014.05.008
0304-3975/© 2014 Elsevier B.V. All rights reserved.

2^{k-1}-approximation algorithm. In the same paper [16] they also show a $\sqrt{p}$-approximation for the PPH problem. *PPH*(k) is fixed parameter tractable and is solvable in polynomial time for $k \le 2$ [17]. In [12] the PPH problem is called the *optimal haplotype inference*. Huang et al. [12] present an approximation algorithm based on semidefinite programming which, with high probability, stops after $O(\log p)$ iterations and is a $O(\log p)$-approximation. The PPH problem is extensively studied in literature and several heuristics and approaches based on integer programming were proposed (see [6] for a survey).

Minimum rainbow subgraph Rainbow subgraphs are fundamental in combinatorics and have been extensively studied (e.g. [7,18,1,10,19]). In general, combinatorists study the existence of a rainbow subgraph under various conditions. However, from the algorithmic perspective, the problem did not receive much attention until recently. Camacho et al. [5] give an approximation algorithm with a ratio of $\frac{5}{6}\Delta$, where Δ is the maximum degree of the input graph. This was later improved to $\frac{1}{2} + (\frac{1}{2} + \epsilon)\Delta$ for arbitrarily small ϵ [4]. Katrenič and Schiermeyer [4] also prove that the MRS problem is **APX**-hard on graphs with maximum degree 2 (notice that the **APX**-hardness of MRS in the general case follows from the **APX**-hardness of PPH(k)) and present an exact algorithm with time complexity $O(2^{(p+2p\log\Delta)}n^{O(1)})$. Koch et al. [15] show that a natural greedy algorithm achieves a ratio of $\frac{\Delta}{2} + \frac{\ln\Delta+1}{2}$ (if the average degree of the minimum rainbow subgraph is d, then the greedy algorithm achieves a ratio of $\frac{d}{2} + \frac{\ln\lceil d\rceil+1}{2}$). Notice that the best approximation ratio is still $O(n)$ in the worst case.

Other related problems If we do not consider the coloring of the edges, the MRS problem is known as the $k - f(k)$ dense subgraph problem introduced by Asahiro et al. [2]. The $k - f(k)$ dense subgraph problem is **NP**-hard as it is a special case of the maximum clique problem when $f(k) = k(k-1)/2$.

The MRS problem is a special case of the minimum k-colored subgraph problem (MkCSP) introduced by Hajiaghayi et al. [11]. MkCSP is defined as follows: given an undirected graph G, a color function that assigns to each edge one or more of p given colors, and an integer $k \le p$, find a minimum set of vertices of G inducing edges of at least k colors. As shown in [11], this problem has a surprising connection to the $k - f(k)$ dense subgraph problem and it is a generalization of the PPH problem. An important case of MkCSP occurs when $k = p$.

1.3. Our results

In this paper we decrease the gap between the approximation lower and upper bounds of the minimum rainbow subgraph problem. First, we show a $\max(\sqrt{2n}, \sqrt{\Delta}(1+\sqrt{\ln\Delta/2}))$-approximation algorithm, where n is the number of vertices in the input graph.

The algorithm is based on randomized linear programming (LP) rounding. The first step of the algorithm is to solve the LP relaxation of an integer program for the MRS problem. Then, we add each vertex in the solution with a probability proportional to the corresponding variable of the LP (multiplied by a certain factor). We show that the subgraph constructed in this way contains "most" of the p colors. Thus, we can apply the following naive algorithm for the remaining colors w: pick an arbitrary edge colored with w and add both its endvertices in the subgraph.

On the inapproximability side we show that the MRS is hard to approximate within a factor of $c\ln\Delta$, for some $c > 0$, unless $\mathbf{NP} \subseteq \mathbf{DTIME}(n^{O(\log\log n)})$. The hardness result is obtained via a gap-preserving reduction from the set cover problem. Given a set cover instance with n elements we create an instance of the MRS problem such that $OPT_{MRS} = n(OPT_{SC} + 1)$, where OPT_{MRS} and OPT_{SC} are the values of the optimal solutions of the MRS problem and, respectively, the set cover. Feige shows [8] that it is not possible, assuming $\mathbf{NP} \not\subseteq \mathbf{DTIME}(n^{O(\log\log n)})$, to decide in polynomial time if a set cover instance has a solution using k sets or the number of sets in the optimal solution is greater than $k\ln n$. Combining Feige's result with our reduction, we obtain the claimed hardness result.

The rest of the paper is organized as follows. In Section 2 we give preliminary definitions. In Section 3 we present the approximation algorithm and in Section 4 we show the hardness result. Section 5 is reserved for conclusions and open problems.

2. Preliminaries

In this section we introduce notation and give preliminary definitions. We start with the definition of the minimum rainbow subgraph problem.

Problem 1 *(Minimum rainbow subgraph). The input of the problem consists of an undirected graph $G = (V, E)$, and a function $col : E \to \{1, 2, \dots, p\}$. A rainbow subgraph of G is a graph $G' = (V', E')$ with $V' \subseteq V$ and $E' \subseteq E$ such that for any $i \in \{1, 2, \dots, p\}$ there is exactly one edge $e \in E'$ with $col(e) = i$. The goal is to find a rainbow subgraph of minimum order (i.e. $|V'|$ is minimized).*

Example 1. Consider the complete graph on the vertex set $V = \{a, b, c, d\}$ shown in Fig. 1. There exists a minimum rainbow subgraph with 3 vertices. One such subgraph is, for example, the graph induced by the vertex set $V' = \{a, b, c\}$.

In the rest of the paper we use the following notation. Let $\{1, 2, \dots, n\}$ be the vertex set of the input graph, m be the number of edges in G and Δ be the maximum degree in G. We say that a color w is *covered* by a subgraph $G' = (V', E')$

Please cite this article in press as: A. Popa, Better lower and upper bounds for the minimum rainbow subgraph problem, Theor. Comput. Sci. (2014), http://dx.doi.org/10.1016/j.tcs.2014.05.008