Information Processing Letters 114 (2014) 446–449

Contents lists available at ScienceDirect

Information Processing Letters

www.elsevier.com/locate/ipl

Parameterized algorithms for load coloring problem

Gregory Gutin *, Mark Jones

Department of Computer Science, Royal Holloway University of London, Egham, Surrey TW20 0EX, UK

ARTICLE INFO

Article history:
Received 8 August 2013
Received in revised form 10 March 2014
Accepted 19 March 2014
Available online 24 March 2014
Communicated by J. Torán

Keywords:
Design of algorithms
Fixed-parameter tractable
Kernel
Load coloring

ABSTRACT

One way to state the Load Coloring Problem (LCP) is as follows. Let $G = (V, E)$ be graph and let $f : V \to \{\text{red}, \text{blue}\}$ be a 2-coloring. An edge $e \in E$ is called red (blue) if both end-vertices of e are red (blue). For a 2-coloring f, let r'_f and b'_f be the number of red and blue edges and let $\mu_f(G) = \min\{r'_f, b'_f\}$. Let $\mu(G)$ be the maximum of $\mu_f(G)$ over all 2-colorings.

We introduce the parameterized problem k-LCP of deciding whether $\mu(G) \geqslant k$, where k is the parameter. We prove that this problem admits a kernel with at most $7k$. Ahuja et al. (2007) proved that one can find an optimal 2-coloring on trees in polynomial time. We generalize this by showing that an optimal 2-coloring on graphs with tree decomposition of width t can be found in time $O^*(2^t)$. We also show that either G is a Yes-instance of k-LCP or the treewidth of G is at most $2k$. Thus, k-LCP can be solved in time $O^*(4^k)$.

1. Introduction

For a graph $G = (V, E)$ with n vertices, m edges and maximum vertex degree Δ, the load distribution of a 2-coloring $f : V \to \{\text{red}, \text{blue}\}$ is a pair (r_f, b_f), where r_f is the number of edges with at least one end-vertex colored red and b_f is the number of edges with at least one end-vertex colored blue. We wish to find a coloring f such that the function $\lambda_f(G) := \max\{r_f, b_f\}$ is minimized. We will denote this minimum by $\lambda(G)$ and call this problem Load Coloring Problem (LCP). The LCP arises in Wavelength Division Multiplexing, the technology used for constructing optical communication networks [1,9]. Ahuja et al. [1] proved that the problem is NP-hard and gave a polynomial time algorithm for optimal colorings of trees. For graphs G with genus $g > 0$, Ahuja et al. [1] showed that a 2-coloring f such that $\lambda_f(G) \leqslant \lambda(G)(1 + o(1))$ can be computed in $O(n + g \log n)$-time, if the maximum degree satisfies $\Delta = o(\frac{m^2}{ng})$ and an embedding is given.

For a 2-coloring $f : V \to \{\text{red}, \text{blue}\}$, let r'_f and b'_f be the number of edges whose end-vertices are both red and blue, respectively (we call such edges *red* and *blue*, respectively). Let $\mu_f(G) := \min\{r'_f, b'_f\}$ and let $\mu(G)$ be the maximum of $\mu_f(G)$ over all 2-colorings of V. It is not hard to see (and it is proved in Remark 1.1 of [1]) that $\lambda(G) = m - \mu(G)$ and so the LCP is equivalent to maximizing $\mu_f(G)$ over all 2-colorings of V.

In this paper we introduce and study the following parameterization of LCP.

> k-Load Coloring Problem (k-LCP)
> *Input:* A graph $G = (V, E)$ and an integer k.
> *Parameter:* k
> *Question:* Is $\mu(G) \geqslant k$? (Equivalently, is $\lambda(G) \leqslant m - k$?)

We provide basics on parameterized complexity and tree decompositions of graphs in the next section. In Section 3, we show that k-LCP admits a kernel with at most $7k$ vertices. Interestingly, to achieve this linear bound, only two simple reduction rules are used. In Section 4, we generalize the result of Ahuja et al. [1] on trees by showing

* Corresponding author.
E-mail addresses: gutin@cs.rhul.ac.uk (G. Gutin), markj@cs.rhul.ac.uk (M. Jones).

http://dx.doi.org/10.1016/j.ipl.2014.03.008
0020-0190/

that an optimal 2-coloring for graphs with tree decomposition of width t can be obtained in time $2^t n^{O(1)}$. We also show that either G is a Yes-instance of k-LCP or the treewidth of G is at most $2k$. As a result, k-LCP can be solved in time $4^k n^{O(1)}$. We conclude the paper in Section 5 by stating some open problems.

2. Basics on fixed-parameter tractability, kernelization and tree decompositions

A *parameterized problem* is a subset $L \subseteq \Sigma^* \times \mathbb{N}$ over a finite alphabet Σ. L is *fixed-parameter tractable* if the membership of an instance (x,k) in $\Sigma^* \times \mathbb{N}$ can be decided in time $f(k)|x|^{O(1)}$, where f is a function of the parameter k only. It is customary in parameterized algorithms to often write only the exponential part of $f(k)$: $O^*(t(k)) := O(t(k)(kn)^{O(1)})$.

Given a parameterized problem L, a *kernelization of* L is a polynomial-time algorithm that maps an instance (x,k) to an instance (x',k') (the *kernel*) such that (i) $(x,k) \in L$ if and only if $(x',k') \in L$, (ii) $k' \leqslant g(k)$, and (iii) $|x'| \leqslant g(k)$ for some function g. The function $g(k)$ is called the *size* of the kernel.

It is well-known that a parameterized problem L is fixed-parameter tractable if and only if it is decidable and admits a kernelization. Due to applications, low degree polynomial size kernels are of main interest. Unfortunately, many fixed-parameter tractable problems do not have kernels of polynomial size unless the polynomial hierarchy collapses to the third level, see, e.g., [2–4]. For further background and terminology on parameterized complexity we refer the reader to the monographs [5,6,8].

Definition 1. A *tree decomposition* of a graph $G = (V,E)$ is a pair $(\mathcal{X}, \mathcal{T})$, where $\mathcal{T} = (I,F)$ is a tree and $\mathcal{X} = \{X_i\colon i \in I\}$ is a collection of subsets of V called *bags*, such that:

1. $\bigcup_{i \in I} X_i = V$;
2. For every edge $xy \in E$, there exists $i \in I$ such that $\{x,y\} \subseteq X_i$;
3. For every $x \in V$, the set $\{i\colon x \in X_i\}$ induces a connected subgraph of $\mathcal{T}$.

The *width* of $(\mathcal{T}, \mathcal{X})$ is $\max_{i \in I} |X_i| - 1$. The *treewidth* of a graph G is the minimum width of all tree decompositions of G.

To distinguish between vertices of G and $\mathcal{T}$, we call vertices of $\mathcal{T}$ *nodes*. We will often speak of a bag X_i interchangeably with the node i to which it corresponds in $\mathcal{T}$. Thus, for example, we might say two bags are *neighbors* if they correspond to nodes in $\mathcal{T}$ which are neighbors. We define the *descendants* of a bag X_i as follows: every child of X_i is a descendant of X_i, and every child of a descendant of X_i is a descendant of X_i.

Definition 2. A *nice tree decomposition* of a graph $G = (V,E)$ is a tree decomposition $(\mathcal{X}, \mathcal{T})$ such that $\mathcal{T}$ is a rooted tree, and each node i falls under one of the following classes:

- i **is a Leaf node:** Then i has no children;
- i **is an Introduce node:** Then i has a single child j, and there exists a vertex $v \notin X_j$ such that $X_i = X_j \cup \{v\}$;
- i **is a Forget node:** Then i has a single child j, and there exists a vertex $v \in X_j$ such that $X_i = X_j \setminus \{v\}$;
- i **is a Join node:** Then i has two children h and j, and $X_i = X_h = X_j$.

It is known that any tree decomposition of a graph can be transformed into a nice tree decomposition of the same width.

Lemma 1. *(See [7].) Given a tree decomposition with $O(n)$ nodes of a graph G with n vertices, we can construct, in time $O(n)$, a nice tree decomposition of G of the same width and with at most $4n$ nodes.*

3. Linear kernel

For a vertex v of a graph $G = (V,E)$ and set $X \subseteq V$, let $\deg_X(v)$ denote the number of neighbors of v in X. If $X = V$, we will write $\deg(v)$ instead of $\deg_V(v)$.

Lemma 2. *Let $G = (V,E)$ be a graph with no isolated vertices, with maximum degree $\Delta \geqslant 2$ and let $|V| \geqslant 5k$. If $|V| \geqslant 4k + \Delta$, then (G,k) is a Yes-instance of k-LCP.*

Proof. Suppose that $|V| \geqslant 4k + \Delta$, but (G,k) is a No-instance of k-LCP.

Let M be a maximum matching in G and let Y be the set of vertices which are not end-vertices of edges in M. If M has at least $2k$ edges, then we may color half of them blue and half of them red, so we conclude that $|M| < 2k$.

For an edge $e = uv$ in M, let $\deg_Y(e) = \deg_Y(u) + \deg_Y(v)$, that is the number of edges between a vertex in Y and a vertex of e.

Claim. *For any e in M, $\deg_Y(e) \leqslant \max\{\Delta - 1, 2\}$.*

Proof of Claim. Suppose that $\deg_Y(e) \geqslant \Delta$ and let $e = uv$. As u and v are adjacent, $d_Y(u)$ and $d_Y(v)$ are each less than Δ. But as $\deg_Y(u) + \deg_Y(v) = \deg_Y(e) \geqslant \Delta$, it follows that $\deg_Y(u) \geqslant 1$ and $\deg_Y(v) \geqslant 1$. Then either u and v have only one neighbor in Y, which is adjacent to both of them (in which case $\deg_Y(e) = 2$), or there exist vertices $x \neq y \in Y$ such that x is adjacent to u and y is adjacent to v. Then M is not a maximum matching, as $xuvy$ is an augmenting path, which proves the claim. □

Now let M' be a subset of edges of M such that

$$\sum_{e' \in M'} \deg_Y(e') \geqslant k - |M'|, \tag{1}$$

and

$$\left[\sum_{e' \in M'} \deg_Y(e')\right] - \deg_Y(e) < k - |M'|, \quad \text{for any } e \in M'. \tag{2}$$