A-025

Finding Reconfigurations between List Edge-Colorings of a Graph

Takehiro Ito* Tohoku University
Marcin Kamiński† Université Libre de Bruxelles
Erik D. Demaine‡ MIT

1 Introduction

Reconfiguration problems arise when we wish to find a step-by-step transformation between two feasible solutions of a problem such that all intermediate results are also feasible. Recently, Ito *et al.* [5] proposed a framework of reconfiguration problems, and gave complexity and approximability results for reconfiguration problems derived from several well-known problems, such as INDEPENDENT SET, CLIQUE, MATCHING, etc. In this paper, we study a reconfiguration problem for list edge-colorings of a graph.

An (ordinary) *edge-coloring* of a graph G is an assignment of colors from a color set C to each edge of G so that every two adjacent edges receive different colors. In *list edge-coloring*, each edge e of G has a set $L(e)$ of colors, called the *list* of e. Then, an edge-coloring f of G is called an *L-edge-coloring* of G if $f(e) \in L(e)$ for each edge e, where $f(e)$ denotes the color assigned to e by f. Fig. 1 illustrates three L-edge-colorings of the same graph with the same list L; the color assigned to each edge is surrounded by a box in the list. Clearly, an edge-coloring is an L-edge-coloring for which $L(e)$ is the same color set C for every edge e of G, and hence list edge-coloring is a generalization of edge-coloring.

Suppose now that we are given *two* L-edge-colorings of a graph G (e.g., the ones in Fig. 1(a) and (c)), and we are asked whether we can transform one into the other via L-edge-colorings of G such that each differs from the previous one in only one edge color assignment. We call this problem the LIST EDGE-COLORING RECONFIGURATION problem. For the particular instance of Fig. 1, the answer is "yes," as illustrated in Fig. 1, where the edge whose color assignment was changed from the previous one is depicted by a thick line. One can imagine a variety of practical scenarios where an edge-coloring (e.g., representing a feasible schedule) needs to be changed (to use a newly found better solution or to satisfy new side constraints) by individual color changes (preventing the need for any coordination) while maintaining feasibility (so that nothing breaks during the transformation). Reconfiguration problems are also interesting in general because they provide a new perspective and deeper understanding of the solution space and of heuristics that navigate that space.

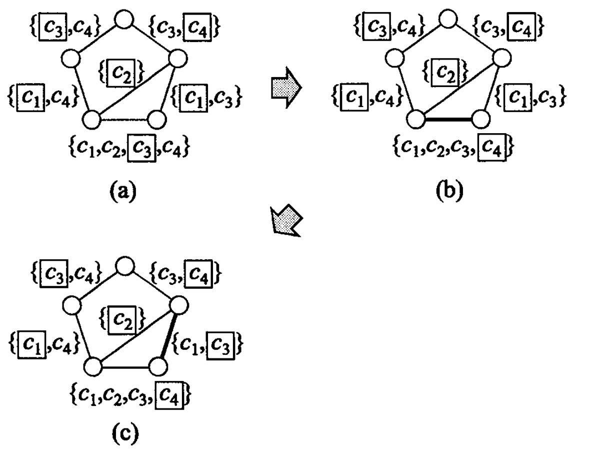

Fig. 1: A sequence of L-edge-colorings of a graph.

Reconfiguration problems have been studied extensively in recent literature [1, 2, 3, 4, 5], in particular for (ordinary) vertex-colorings. For a positive integer k, a *k-vertex-coloring* of a graph is an assignment of colors from $\{c_1, c_2, \ldots, c_k\}$ to each vertex so that every two adjacent vertices receive different colors. Then, the k-VERTEX-COLORING RECONFIGURATION problem is defined analogously. Bonsma and Cereceda [1] proved that k-VERTEX-COLORING RECONFIGURATION is PSPACE-complete for $k \geq 4$, while Cereceda *et al.* [2] proved that k-VERTEX-COLORING RECONFIGURATION is solvable in polynomial time for $1 \leq k \leq 3$. Edge-coloring in a graph G can be reduced to vertex-coloring in the "line graph" of G. However, by this reduction, we can solve only a few instances of LIST EDGE-COLORING RECONFIGURATION; all edges e of G must have the same list $L(e) = C$ of size $|C| \leq 3$ although any edge-coloring of G requires at least $\Delta(G)$ colors, where $\Delta(G)$ is the maximum degree of G. Furthermore, the reduction does not work the other way, so we do not obtain any complexity results.

In this paper, we give two results for LIST EDGE-COLORING RECONFIGURATION. The first is to show that the problem is PSPACE-complete, even for planar graphs of maximum degree 3 and just six colors. The second is to give a sufficient condition for which there exists a transformation between any two L-edge-colorings of a tree. Specifically, for a tree T, we prove that any two L-edge-colorings of T can be transformed into each other if $|L(e)| \geq \max\{d(v), d(w)\} + 1$ for each edge $e = vw$ of T, where $d(v)$ and $d(w)$ are the degrees of the endpoints v and w of e, respectively. Our proof for the sufficient condition yields a polynomial-time algorithm that finds a transformation between given two L-edge-colorings of T via $O(n^2)$ intermediate L-edge-colorings, where n is the number of vertices in T.

We remark that our sufficient condition is best possible in some sense. Consider a star $K_{1,n-1}$ in which each edge e has the same list $L(e) = C$ of size $|C| = n - 1$. Then, $|L(e)| = \max\{d(v), d(w)\}$ for all edges $e = vw$, and it is easy to see that there is no transformation between any two L-edge-colorings of the star.

*takehiro@ecei.tohoku.ac.jp
†marcin.kaminski@ulb.ac.be
‡edemaine@mit.edu

2 PSPACE-completeness

We first introduce some terms and define the problem more formally. In the Introduction, we have defined an L-edge-coloring of a graph $G = (V, E)$ with a list L. We say that two L-edge-colorings f and f' of G are *adjacent* if $|\{e \in E : f(e) \neq f'(e)\}| = 1$, that is, f' can be obtained from f by changing the color assignment of a single edge e; the edge e is said to be *recolored* between f and f'. A *reconfiguration sequence* between two L-edge-colorings f_0 and f_t of G is a sequence of L-edge-colorings $f_0, f_1, \dots, f_t$ of G such that f_{i-1} and f_i are adjacent for $i = 1, 2, \dots, t$. We also say that two L-edge-colorings f and f' are *connected* if there exists a reconfiguration sequence between f and f'. Clearly, any two adjacent L-edge-colorings are connected. Then, the LIST EDGE-COLORING RECONFIGURATION problem is to determine whether given two L-edge-colorings of a graph G are connected. The *length* of a reconfiguration sequence is the number of L-edge-colorings in the sequence, and hence the length of the reconfiguration sequence in Fig. 1 is 3.

We have the following theorem.

Theorem 1 LIST EDGE-COLORING RECONFIGURATION *is PSPACE-complete for planar graphs of maximum degree 3 whose lists are chosen from six colors.*

As a proof of Theorem 1, we give a reduction from Nondeterministic Constraint Logic (NCL) [4]. However, we omit the details due to the page limitation.

3 Trees

Since LIST EDGE-COLORING RECONFIGURATION is PSPACE-complete, it is very unlikely that the problem can be solved in polynomial time for general graphs. However, in this section, we give a sufficient condition for which any two L-edge-colorings of a tree T are connected; our sufficient condition can be checked in polynomial time. Moreover, our proof yields a polynomial-time algorithm that finds a reconfiguration sequence of length $O(n^2)$ between given two L-edge-colorings, where n is the number of vertices in T.

Theorem 2 *For a tree T with n vertices, any two L-edge-colorings f and f' of T are connected if $|L(e)| \geq \max\{d(v), d(w)\}+1$ for each edge $e = vw$ of T. Moreover, there is a reconfiguration sequence of length $O(n^2)$ between f and f'.*

Since $\Delta(T) \geq \max\{d(v), d(w)\}$ for all edges vw of a tree T, Theorem 2 immediately implies the following sufficient condition for which any two (ordinary) edge-colorings of T are connected. Note that, for a positive integer k, a *k-edge-coloring* of a tree T is an L-edge-coloring of T for which all edges e have the same list $L(e) = \{c_1, c_2, \dots, c_k\}$.

Corollary 1 *For a tree T with n vertices, any two k-edge-colorings f and f' of T are connected if $k \geq \Delta(T) + 1$. Moreover, there is a reconfiguration sequence of length $O(n^2)$ between f and f'.*

It is obvious that the sufficient condition of Corollary 1 is also best possible in some sense; consider a star $K_{1,n-1}$ in the Introduction.

As a proof of Theorem 2, we give a polynomial-time algorithm that finds a reconfiguration sequence of length $O(n^2)$ between given two L-edge-colorings f_0 and f_t of a tree T if our sufficient condition holds. However, due to the page limitation, we only give an outline of our algorithm.

By the breadth-first search starting from an arbitrary vertex r of degree 1, we order all edges $e_1, e_2, \dots, e_{n-1}$ of a tree T. At the ith step, $1 \leq i \leq n-1$, the algorithm recolors e_i from the current color to its target color $f_t(e_i)$, as follows. From the current L-edge-coloring f, we first obtain an L-edge-coloring f' of T such that

(i) there is no edge which is adjacent with e_i and is colored with $f_t(e_i)$; and
(ii) there exists a reconfiguration sequence between f and f' in which none of the edges $e_1, e_2, \dots, e_{i-1}$ is recolored.

Then, we recolor e_i to $f_t(e_i)$. Therefore, e_i is never recolored after the ith step, while e_i may be recolored before the ith step even if e_i is colored with $f_t(e_i)$. We can show that every edge of T can be recolored in such a way, and hence we eventually obtain the target L-edge-coloring f_t. Since the algorithm recolors each edge e_j with $j \geq i$ at most once in the ith step, we can recolor e_i by recoloring at most $n - i$ edges. Our algorithm thus finds a reconfiguration sequence of total length $\sum_{i=1}^{n-1}(n-i) = O(n^2)$.

References

[1] Bonsma, P., Cereceda, L.: Finding paths between graph colourings: PSPACE-completeness and superpolynomial distances. In: Kučera, L., Kučera, A. (eds.) MFCS 2007. LNCS, vol. 4708, pp. 738–749. Springer, Heidelberg (2007)

[2] Cereceda, L., van den Heuvel, J., Johnson, M.: Finding paths between 3-colourings. In: Proc. of IWOCA 2008, pp. 182–196 (2008)

[3] Gopalan, P., Kolaitis, P.G., Maneva, E.N., Papadimitriou, C.H.: The connectivity of Boolean satisfiability: computational and structural dichotomies. In: Bugliesi, M., Preneel, B., Sassone, V., Wegener, I. (eds.) ICALP 2006. LNCS, vol. 4051, pp. 346–357. Springer, Heidelberg (2006)

[4] Hearn, R.A., Demaine, E.D.: PSPACE-completeness of sliding-block puzzles and other problems through the nondeterministic constraint logic model of computation. Theoretical Computer Science 343, 72–96 (2005)

[5] Ito, T., Demaine, E.D., Harvey, N.J.A., Papadimitriou, C.H., Sideri, M., Uehara, R., Uno, Y.: On the complexity of reconfiguration problems. In: Hong, S., Nagamochi, H., Fukunaga, T. (eds.) ISAAC 2008. LNCS, vol. 5369, pp. 28–39. Springer, Heidelberg (2008)