Journal of Graph Algorithms and Applications
http://jgaa.info/ vol. 16, no. 2, pp. 225–241 (2012)

Minimizing the Number of Label Transitions Around a Nonseparating Vertex of a Planar Graph

Bojan Mohar[1,2] *Petr Škoda*[1]

[1]Department of Mathematics
Simon Fraser University
Burnaby, BC, V5A 1S6, Canada.
[2]On leave from:
IMFM & FMF, Department of Mathematics,
University of Ljubljana, Ljubljana, Slovenia.

Abstract

We study the minimum number of label transitions around a given vertex v_0 in a planar multigraph G, in which the edges incident with v_0 are labelled with integers $1, \dots, l$, and the minimum is taken over all embeddings of G in the plane. For a fixed number of labels, a linear-time fixed-parameter tractable algorithm that computes the minimum number of label transitions around v_0 is presented. If the number of labels is unconstrained, then the problem of deciding whether the minimum number of label transitions is at most k is NP-complete.

Submitted:	Reviewed:	Revised:	Reviewed:
July 2011	October 2011	November 2011	January 2012
Revised:	Accepted:	Final:	Published:
January 2012	January 2012	January 2012	January 2012
Article type:		Communicated by:	
Regular paper		C. D. Tóth	

Supported in part by an NSERC Discovery Grant (Canada), by the Canada Research Chair program, and by the Research Grant P1–0297 of ARRS (Slovenia).

An extended abstract of this paper appeared in the proceedings of IWOCA 2011.

E-mail addresses: mohar@sfu.ca (Bojan Mohar) pskoda@sfu.ca (Petr Škoda)

1 Introduction

Let G be a planar multigraph. Suppose that the edges incident with a vertex $v_0 \in V(G)$ are labelled by integers $1, \ldots, l$. We are interested in finding an embedding of G in the plane such that the number of label transitions around v_0 is minimized. By a *label transition* we mean two edges that are consecutive in the local rotation around v_0 and whose labels are different. The motivation for this problem comes from investigations of minimum genus embeddings of graphs with small separations. In particular, to compute the genus of a 2-sum of two graphs [12], see also [6, 7, 11], it is necessary to know if a graph admits a planar embedding with only four label transitions (where $l = 2$). The problem of minimizing the number of transitions may also be of interest in bioinformatics. Namely, problems arising in genome sequencing and in relation to phylogenetic trees involve notions very close to the minimization of transitions. Our solution in this paper answers a question posed by Cedric Chauve [3] in relation to a generalization of the Consecutive Ones Property of matrices.

By deleting the vertex v_0 from G and putting all edge labels onto vertices incident with the deleted edges, we obtain an equivalent formulation of the same problem. We may assume that the edges incident with the same vertex $v \neq v_0$ have different labels because such edges can always be drawn next to each other without increasing the number of label transitions. Both representations are useful and will be treated in this paper. Let H be the graph obtained from G by deleting v_0. For each $v \in V(H)$, let $\lambda(v)$ be the set of all labels of edges joining v and v_0 in G. If v is not a neighbor of v_0, then $\lambda(v) = \emptyset$. The pair (H, λ) carries the whole information about G and the labels of edges around v_0 (assuming that the edges incident with the same vertex $v \neq v_0$ have different labels).

Let $\mathcal{L}$ be a set of labels. The graph H together with the labelling $\lambda : V(H) \to 2^{\mathcal{L}}$ is a *labelled graph*. Let $\widehat{H}$ be the graph obtained from a labelled graph H by adding a vertex v_0 to H and joining it to each vertex v by $|\lambda(v)|$ edges and labelling these edges by elements of $\lambda(v)$. The vertex v_0 is called the *center* of $\widehat{H}$. If the graph $\widehat{H}$ is planar (which can be checked in linear time, see [9]), we are back to an instance of the original problem.

Given (H, λ) or $\widehat{H}$, v_0, and the labelling of edges incident with v_0, consider an embedding Π of $\widehat{H}$ in the plane (all embeddings in this paper are into the plane). Define the *label sequence* $Q = Q(\Pi)$ of Π to be the cyclic sequence of labels of edges emanating from v_0 in the clockwise order of the local rotation around v_0 in Π. The *origin* of a label $L \in Q$ that came from an edge vv_0 is the vertex v. A *label transition* in Q is a pair of (cyclically) consecutive labels A, B in Q such that $A \neq B$. The *number of transitions* $\tau(Q)$ of Q is the number of label transitions in Q. The *number of transitions* $\tau(\widehat{H})$ of $\widehat{H}$ is the minimum $\tau(Q(\Pi))$ taken over all planar embeddings Π of $\widehat{H}$. When considering label transitions, the graphs H and $\widehat{H}$ are used interchangeably, i.e., we may write $\tau(H)$ instead of $\tau(\widehat{H})$.

A *cutvertex* in a graph G is a vertex of G whose removal increases the number of components of G. A vertex is *nonseparating* if it is not a cutvertex. The following problem will be of our main interest:

MIN-TRANS. Given a planar multigraph G with edges incident to a fixed vertex v_0 labelled by $1, \dots, l$ and an integer k, determine if $\tau(G) \le k$.

In the following, we show that MIN-TRANS can be solved in linear time when v_0 is nonseparating and the number of labels l is fixed.

Theorem 1 *For every fixed integer l, there is a linear-time algorithm that determines the minimum number of transitions $\tau(G)$ of a given planar multigraph G with edges incident to a fixed nonseparating vertex v_0 labelled by at most l different labels. In particular, the* MIN-TRANS *problem when v_0 is nonseparating is fixed-parameter tractable for the parameter l.*

The proof of Theorem 1 is given in Sections 3 and 4. Let us observe that the time complexity of our algorithm remains near-linear $O(n^{1+\varepsilon})$ (for any $\varepsilon > 0$) as long as $l = o(\log \log \log n)$ where n is the size of the input.

If we allow v_0 to be a cutvertex, then the problem changes dramatically. Though the formalism developed in this paper is not sufficient to deal with this general case, we believe that it can also be solved in linear time for every fixed l.

We also show that our algorithmic result is best possible in the sense that MIN-TRANS becomes NP-complete when l is part of the input.

Theorem 2 MIN-TRANS *is NP-complete if the number of labels l is unconstrained. The problem remains NP-complete even when v_0 is nonseparating and each label occurs precisely twice.*

The proof of Theorem 2 is deferred to Section 5.

2 Outline

The main ideas of our algorithm are described by the following steps. First, we simplify the input graph G so that every 2-connected component of $G - v_0$ is either an edge or a cycle bounding a face. We call the resulting graph a *cactus*.

Having obtained the simplified graph, we consider the tree-structure of its blocks using a dynamic programming approach. Herefrom we consider a cactus with a distinguished vertex (*root*). For each pair A, B of labels, we keep information about the minimum number of transitions $\rho[A, B]$ assuming that the label sequence starts and ends at the root and the sequence is prepended and appended with A and B, respectively. After establishing Lemma 6 that covers concatenations of optimal label sequences, the recursive procedure is rather straightforward, except for the case when the root has big degree. This is the hard part and the details are presented in Section 4.

The main observation is that we only have to distinguish, for each block incident with the root, the l^2 values $\rho[A, B]$ (where l is the fixed number of labels) and that the important information is only how the values differ from $\nu = \min_{A,B} \rho[A, B]$. The difference $\rho[A, B] - \nu$ is either $0, 1$, or 2. We can store the complete information about this difference in a variable taking at most 3^{l^2} different values. These values are called *types*. The blocks with the same type need not be distinguished.

In order to determine an optimal sequence of blocks incident with the root, we transform the problem into a problem of finding an optimum closed walk in a multigraph K on l vertices (corresponding to the labels) in which each pair of vertices is joined by 3^{l^2} edges (one for each possible type). In this closed walk, the number of edges corresponding to any particular type p must be equal to the number of blocks incident with the root whose type is equal to p. The edge of K corresponding to the type p between vertices A and B is given weight equal to the value of p at A, B. Although the number of closed walks is exponential, we show that one can restrict himself to consider only a constant number of walks in K. The details are given in Section 4.

Let W be an optimal closed walk in K. We consider W as a multiset of edges, the multiplicities corresponding to the number of times each edge appears in W. The walk W can be decomposed into two parts, a closed walk S and an eulerian multiset T of edges, where S has constant size and traverses each edges traversed by W at least once. Moreover, T contains only edges that have weight 0 and for each type p, the number n_p of edges in T (counted with appropriate multiplicities of the multiset) that corresponds to p is even. An important property of such a decomposition S, T is that there exists a multiset T' of edges of K such that T' also satisfies the properties above and uses n_p edges of type p. The set T' can be constructed in the following way. For each type p, we pick an edge e_p of S with weight 0 that corresponds to p. The multiset T' consists of the edge e_p with multiplicity n_p for each type p. It is not hard to check that S and T' form a decomposition of a closed walk in K that has the same total weight as the original walk. Therefore, S, T' also decomposes an optimal closed walk. There are bounded number of closed walks S to be considered and for each such closed walk, we define T' and check if the pair S, T' satisfies the required conditions. This yields a constant-time algorithm with linear-time preprocessing that determines the types of the blocks incident with the root.

3 Bounded Number of Labels

In this section we develop most of the formalism needed to prove Theorem 1. In particular, it is observed that we can restrict our attention to a special class of cactus graphs; also, the basic structure of the algorithm is presented.

Let H and G be labelled graphs with labellings λ and μ, respectively. If every label sequence of (H, λ) is also a label sequence of (G, μ), and vice versa, then H and G are said to be *equivalent*.

A connected graph G is called a *cactus* if every block of G is either an edge or a cycle. (A *block* in a connected graph G is either a cutedge or a maximal 2-connected subgraph of G). A labelled cactus G is *leaf-labelled* if every endblock of G is an edge, every vertex of G has at most one label, and a vertex of G is labelled if and only if it is a leaf.

The following lemma shows that it suffices to prove Theorem 1 for the case when H is a leaf-labelled cactus. To avoid trivialities, we shall assume that there are at least two vertices whose label set $\lambda(v)$ is non-empty.

Lemma 1 *Let H be a connected labelled graph. If $\widehat{H}$ is planar, then there exists a leaf-labelled cactus G which is equivalent to H. Furthermore, G can be constructed in linear time.*

Proof: We construct G from H in the following series of steps. First, we move the labels onto leaves. After that we remove unlabelled parts of the graph and finally we remove the insides of cycles.

Construct H' from H in the following manner. For each labelled vertex $v \in V(H)$, attach $|\lambda(v)|$ new vertices v_L, $L \in \lambda(v)$ to v and then remove all labels from v. Label each v_L with the label L. A planar embedding of $\widehat{H'}$ can be transformed to an embedding of $\widehat{H}$ with the same label sequence by contracting the edges vv_L, $v \in V(H)$, $L \in \lambda(v)$, in H'. Conversely, a planar embedding of $\widehat{H}$ can be transformed to an embedding of $\widehat{H'}$ with the same label sequence by subdividing the edges incident to the center of $\widehat{H}$. Hence H and H' are equivalent.

Suppose that v is a cutvertex of H' and a component B' of $H' - v$ contains no labels. Since every planar embedding of $\widehat{H'} - B' = \widehat{H' - B'}$ can be extended to an embedding of $\widehat{H'}$ with the same label sequence by embedding B' in one of the faces around v, the labelled graphs H' and $H' - B'$ are equivalent. For each cutvertex v, remove all unlabelled components of $H' - v$ from H' to obtain H'' that is equivalent to H'. It follows that every endblock of H'' contains a label. The subgraph H'' can be easily constructed in linear time by cutting off the appropriate endblocks of H'.

Let Π'' be a planar embedding of $\widehat{H''}$ and v_0 the center of $\widehat{H''}$. Let G be the subgraph of H'' that is formed by the vertices and edges of the facial walk in $\widehat{H''} - v_0$ corresponding to the face in which v_0 was embedded. Note that G contains all labelled vertices of H''. We claim that G is equivalent to H''. Since $\widehat{G}$ is a subgraph of $\widehat{H''}$, every embedding of $\widehat{H''}$ gives an embedding of $\widehat{G}$ with the same label sequence. By construction, every block of G is either an edge or a cycle. Since every endblock of H'' contains a label, also every endblock of G contains a label. Thus, in every planar embedding of $\widehat{G}$, every cycle of G is a facial cycle. Hence, an embedding of $\widehat{G}$ can be extended to an embedding of $\widehat{H''}$ with the same label sequence by embedding the rest of H'' into the facial cycles as given in Π''. Hence G and H'' are equivalent.

Since a planar embedding of $\widehat{H''}$ can be obtained in linear time (see for example [4]), G can be constructed in linear time. It is not difficult to check that G is a leaf-labelled cactus as required. □

In our algorithm, we use a rooted version of graphs. A *root* r in a leaf-labelled cactus H can be any vertex of H. The root is marked by a special label

$L_r \notin \mathcal{L}$. We then speak of a *rooted leaf-labelled cactus*, or simply a cactus (H, r). The restriction on labels in a rooted leaf-labelled cactus is slightly relaxed, every leaf still has precisely one label (possibly L_r) and a non-leaf vertex is labelled only if it is the root. When a label sequence Q of H is cut at the label L_r (and L_r is deleted), we obtain a linear sequence called a *rooted label sequence* of H. Let $\mathcal{Q}(H)$ denote the set of all rooted label sequences of H. Similarly to the unrooted graphs, two rooted graphs are *equivalent* if they admit the same rooted label sequences.

There exists a tree-like structure, called a PC-tree (see [13]), that captures all embeddings of a cactus in the plane. PC-trees and their rooted version, PQ-trees, are used in testing planarity [1], see also [4]. We note that Min-Trans (with v_0 nonseparating) reduces to the problem of minimizing the number of label transitions over all cyclic permutations of the leaves of a PC-tree that are compatible with the PC-tree.

For every embedding Π of $\widehat{H}$, there is the *flipped* embedding Π' of $\widehat{H}$ where each clockwise rotation in Π is a counter-clockwise rotation in Π'. The following lemma formulates this for a rooted label sequence of H. For a linear sequence Q, let Q^R denote the sequence obtained by reversing Q.

Lemma 2 *Let (H, r) be a rooted leaf-labelled cactus. If Q is a rooted label sequence of H, then the reversed sequence Q^R is also a rooted label sequence of H.*

The following lemmas establish a recursive construction of rooted label sequences. Let us recall that for a cutvertex v of H, a *v-bridge* in H is a subgraph of H consisting of a connected component of $H - v$ together with all edges joining this component and v.

Lemma 3 *Let (H, r) be a rooted leaf-labelled cactus where r is a leaf. Let u be the neighbor of r. If u is labelled, then H is of order 2 and has a unique rooted label sequence $Q = \lambda(u)$. Otherwise, (H, r) is equivalent to $(H - r, u)$.*

Proof: If u is labelled, then u is a leaf and H contains precisely one label $\lambda(u)$ and therefore $\lambda(u)$ is the unique rooted label sequence of H. Otherwise, take an embedding of $\widehat{H - r}$ in the plane. Recall that u as the root of $H - r$ is given a special label L_u and thus there is an edge connecting u and the center of $\widehat{H - r}$. Subdividing this edge gives a planar embedding of $\widehat{H}$ with the same rooted label sequence. Similarly, one can obtain an embedding of $\widehat{H - r}$ from an embedding of $\widehat{H}$ with the same rooted label sequence. □

Lemma 4 *Let (H, r) be a rooted leaf-labelled cactus such that r is a cutvertex and let $B_1, \ldots, B_k$ be the r-bridges in H. Every rooted label sequence of H can be partitioned into k consecutive parts where each of the k parts is a rooted label sequence of one of (B_i, r). Conversely, if Q_i is a rooted label sequence of (B_i, r) $(1 \le i \le k)$ and $(i_1, \ldots, i_k)$ is a permutation of $(1, \ldots, k)$, then the concatenation $Q_{i_1} Q_{i_2} \cdots Q_{i_k}$ is a rooted label sequence of H.*

Proof: Suppose for a contradiction that there is a rooted label sequence Q of H with a cyclic subsequence $L_1L_2L_3L_4$ (in this order) such that L_1 and L_3 have origins in B_1 and L_2, L_4 have origins outside B_1. Let Π be an embedding of $\widehat{H}$ that corresponds to Q and $v_1, \ldots, v_4$ the origins of $L_1, \ldots, L_4$. Let G be the graph obtained from $\widehat{H}$ by deleting the center v_0 and adding an edge uv for every two consecutive edges uv_0, vv_0 in the local rotation around v_0. The embedding Π can be extended to a planar embedding Π' of G such that the added edges form a facial cycle. Since v_1 and v_3 are in B_1, there is a path P in $B_1 - r$ joining v_1 and v_3. Similarly, there is a path Q in $H - (B_1 - r)$ joining v_2 and v_4. Since P and Q are disjoint and both are embedded inside C, their endvertices cannot interlace on C. This contradiction proves the claim and implies that all labels in each B_i appear consecutively in every rooted label sequence of H. This proves the first part of the lemma.

The second part is an easy consequence of the fact that arbitrary embeddings of $\widehat{B_i}$ ($1 \le i \le k$) can be combined into an embedding of $\widehat{H}$ so that the cyclic order of r-bridges around r is $B_{i_1}, B_{i_2}, \ldots, B_{i_k}$. □

Let C be a cycle of G. For $v \in V(C)$, let $D_v(C)$ be the union of v-bridges in G that do not contain C.

Lemma 5 *Let (H, r) be a rooted leaf-labelled cactus with r in a cycle C of length k. If $D_r(C)$ is empty, then every rooted label sequence Q of H can be partitioned into $k-1$ (possibly empty) consecutive parts P_v, $v \in V(C) \setminus \{r\}$, where P_v is a rooted label sequence of $(D_v(C), v)$ and P_v appear in Q in one of the two orders corresponding to the two orientations of C.*

Proof: Let Q be a rooted label sequence of H such that the conclusion of the lemma is not true. If labels contained in one of the subgraphs $D_v(C)$ do not form a consecutive subsequence of Q, we obtain a contradiction as in the proof of Lemma 4. Suppose now that Q contains a cyclic subsequence $L_1L_3L_2L_4$ of labels (in this order) such that the origins $u_1, \ldots, u_4$ of $L_1, \ldots, L_4$ are in $D_{v_1}(C), \ldots, D_{v_4}(C)$ and v_1, v_2, v_3, v_4 appear on C in this order. Let Π be an embedding of $\widehat{H}$ that corresponds to Q and let G be the graph obtained from $\widehat{H}$ by deleting the center v_0 and adding an edge uv for every two consecutive edges uv_0, vv_0 in the local rotation around v_0. The embedding Π can be easily modified to a planar embedding Π' of G. It is easy to check that $u_1, \ldots, u_4, v_1, v_3$ are the branch-vertices of a subdivision of $K_{3,3}$ in G, a contradiction with G being planar. □

We are interested in rooted label sequences that have the minimum number of transitions. But to combine them later on, it is important to specify the first and the last label in the rooted label sequence. This motivates the following definition. Let $\mathcal{Q}$ be a set of (linear) label sequences. We say that a sequence $Q \in \mathcal{Q}$ is *AB-minimal* for labels $A, B \in \mathcal{L}$, if

$$\tau(AQB) = \min\{\tau(ASB) \mid S \in \mathcal{Q}\}$$

where AQB is the sequence obtained from Q by adding labels A and B at the beginning and at the end of Q, respectively. A rooted label sequence Q of (H, r) is *AB-minimal* if Q is AB-minimal in $\mathcal{Q}(H)$. Minimal sequences are composed of minimal sequences as the following lemma shows. This allows us to restrict our attention to minimal sequences.

Lemma 6 *Let $\mathcal{Q}$ be the set of all sequences that are concatenations of a sequence in $\mathcal{Q}_1$ and a sequence in $\mathcal{Q}_2$ (in this order). Then for $A, B \in \mathcal{L}$, every AB-minimal sequence Q in $\mathcal{Q}$ is a concatenation of an AC-minimal sequence in $\mathcal{Q}_1$ and a CB-minimal sequence in $\mathcal{Q}_2$ for some label $C \in \mathcal{L}$.*

Proof: Suppose that the lemma is not true for labels A, B. Let $Q = Q_1Q_2$ be an AB-minimal sequence in $\mathcal{Q}$ where $Q_i \in \mathcal{Q}_i$, $i \in \{1, 2\}$. Let C be the first label of Q_2. By assumption, either Q_1 is not AC-minimal in $\mathcal{Q}_1$ or Q_2 is not CB-minimal in $\mathcal{Q}_2$. If Q_1 is not AC-minimal, let Q_1' be an AC-minimal sequence in $\mathcal{Q}_1$. Then $\tau(AQ_1'C) < \tau(AQ_1C)$. It follows that

$$\tau(AQ_1'Q_2B) < \tau(AQ_1Q_2B) = \tau(AQB),$$

a contradiction with the choice of Q. Thus, Q_2 is not CB-minimal. Consequently, if Q_2' is a CB-minimal sequence in $\mathcal{Q}_2$, then $\tau(CQ_2'B) < \tau(CQ_2B) = \tau(Q_2B)$. Hence, we have

$$\tau(AQ_1Q_2'B) \leq \tau(AQ_1CQ_2'B) < \tau(AQ_1CQ_2B) = \tau(AQ_1Q_2B) = \tau(AQB),$$

again a contradiction with the choice of Q. □

Let (H, r) be a rooted leaf-labelled cactus. We can describe "optimal" embeddings of $\widehat{H}$ in the plane by a set of AB-minimal rooted label sequences of H, one for each pair of labels $A, B \in \mathcal{L}$. Let $\rho_H[A, B]$ be the minimum number of label transitions in an AB-minimal rooted label sequence of H. Note that the values of ρ_H differ by at most 2 since adding labels A and B to a sequence increases the number of label transitions by at most 2. Hence we can represent ρ_H by the minimum $\rho_H[A, B]$ over all labels A, B and by the individual differences from this minimum. Let n_H be the *minimum number of label transitions* in a rooted label sequence of H and let $p_H[A, B] = \rho_H[A, B] - n_H$. As noted above, $p_H[A, B] \in \{0, 1, 2\}$. The function $p_H : \mathcal{L} \times \mathcal{L} \to \{0, 1, 2\}$ is called the *type* of H. It is convenient to view the type as a number between 1 and $t := 3^{l^2}$, whose digits in the ternary system correspond to the particular values $p_H[A, B]$ (for some linear ordering of all pairs $(A, B) \in \mathcal{L} \times \mathcal{L}$). Note that the number t of different types is a constant when the number of labels is fixed. We will see in Lemma 8 that rooted cacti of the same type are interchangeable in any ordering around a cutvertex. We call the pair (p_H, n_H) the *descriptor* of H. For simplicity, we also call the function ρ_H the descriptor of H since it is easy to compute ρ_H from (p_H, n_H), and vice versa.

Note that the "unrooted" number of transitions $\tau(H)$ can be obtained from the descriptor of H as

$$\tau(H) = \min_{A \in \mathcal{L}} \rho_H[A, A].$$

To see this, suppose first that Q is a cyclic label sequence of H (containing L_r) and A is a label next to L_r in Q. Let R be the cyclic sequence obtained from Q by replacing L_r with two labels A. Then R has the same number of transitions as the cyclic sequence $Q - L_r$. By splitting R between the two new labels A, we obtain a linear sequence ASA, where S is a rooted label sequence of H, with the same number of transitions as in R. This shows that $\tau(H) \geq \min_{A \in \mathcal{L}} \rho_H[A, A]$. For the other inequality, note that from a linear sequence ASA, where S is a rooted label sequence of H, we can obtain a cyclic label sequence of H with smaller or equal number of label transitions by joining the ends of ASA and then deleting the two labels A.

Next, we consider how a descriptor of a rooted leaf-labelled cactus can be computed from descriptors of its subcacti. The first non-trivial case is when the root lies on a cycle.

Lemma 7 *Let (H, r) be a rooted leaf-labelled cactus such that r is a vertex of degree 2 in a cycle C of length k in H. Then the descriptor of H can be computed from descriptors of $(D_v(C), v)$, $v \in V(C)$, in time $\mathcal{O}(l^3 k)$.*

Proof: In contrast with rooting at a cutvertex, the order of subgraphs $D_v(C)$ around C is fixed. Let us take an embedding of C and let $r, v_1, \ldots, v_{k-1}$ be the vertices of C in the clockwise order. By Lemma 5, every sequence in $\mathcal{Q}(H)$ is a concatenation of $k-1$ sequences from $\mathcal{Q}(D_{v_1}(C)), \ldots, \mathcal{Q}(D_{v_{k-1}}(C))$ in this or the reverse order. Let $\mathcal{P}_i$ be the set of sequences that are concatenation of $k-i$ sequences from $\mathcal{Q}(D_{v_i}(C)), \ldots, \mathcal{Q}(D_{v_{k-1}})(C)$ in this order. By Lemma 6, every AB-minimal sequence in $\mathcal{P}_i$ is obtained as a concatenation of an AL-minimal sequence in $\mathcal{Q}(D_{v_i}(C))$ and an LB-minimal sequence in $\mathcal{P}_{i+1}$ for some $L \in \mathcal{L}$. Let $q_i[A, B]$ be the number of transitions in an AB-minimal sequence of $\mathcal{P}_i$. Since $\mathcal{P}_{k-1} = \mathcal{Q}(D_{v_{k-1}}(C))$, $q_{k-1}[A, B] = \rho_{D_{v_{k-1}}(C)}[A, B]$. Lemma 6 gives that

$$q_i[A, B] = \min_{L \in \mathcal{L}} \{\rho_{D_{v_i}(C)}[A, L] + q_{i+1}[L, B]\}, \text{ for } 1 \leq i < k-1.$$

Note that q_1 stores the number of transitions of all AB-minimal rooted label sequences of H when the order of C is fixed. To allow for flipping of C, we note that an AB-minimal rooted label sequence of H is a BA-minimal rooted sequence of H in the flipped embedding of H. Hence

$$\rho_H[A, B] = \min\{q_1[A, B], q_1[B, A]\}.$$

For each $1 \leq i \leq k-1$, we compute each of the l^2 values of q_i in time $\mathcal{O}(l)$. That gives the overall time complexity $\mathcal{O}(l^3 k)$. □

Computing the descriptor of a leaf-labelled cactus rooted at a cutvertex turns out to be the crux. Let (H, r) be a rooted leaf-labelled cactus where r is a cutvertex of H and let $B_1, \ldots, B_k$ be the r-bridges in H. Let $b_H(i)$ be the number of r-bridges in H of type i, $i = 1, \ldots, t$. We view b_H as an integer vector in $\mathbb{Z}^t$ with $\sum_{i=1}^t b_H(i) = k$. A non-negative integer vector $b \in \mathbb{Z}^t$ is called a *bridge vector* and $\mathrm{sum}(b) = \sum_{i=1}^t b(i)$ the *sum* of b. Note that there are at most $\mathcal{O}(k^{t+1})$ different non-negative integer vectors b in $\mathbb{Z}^t$ with the sum at most k.

Each bridge vector b describes a problem to be solved: How to order k, $k = \mathrm{sum}(b)$, bridges of types given by b around a vertex so that the number of label transitions on the boundaries between bridges is minimized. We will see that each ordered sequence of types in b gives a sequence of labels with each type being a connection to the next label. Although this sequence of labels is not unique, it is a useful concept that will be heavily used in Section 4. For fixed labels A, B, let $\mathcal{R}_{AB}(n)$ be the set of sequences of $n+1$ labels $(L_0, \dots, L_n)$ such that $L_0 = A$ and $L_n = B$. Let $P = (p_1, \dots, p_k)$ be a sequence containing all types occurring in b (with appropriate multiplicities). For such an *ordering of types in* b and a sequence $R \in \mathcal{R}_{AB}(k)$, $R = (L_0, \dots, L_k)$, let $m(P, R) = \sum_{i=1}^{k} p_i[L_{i-1}, L_i]$. Let $m_b[A, B]$ be the the minimum $m(P, R)$ taken over all orderings P of b and all sequences $R \in \mathcal{R}_{AB}(k)$, $k = \mathrm{sum}(b)$. This minimum depends only on the types of the bridges, not on the minimum number of label transitions of the bridges.

The fact that computation of m_b is a solution to the posed problem and that it gives a way to compute the descriptor of a leaf-labelled cactus rooted at a cutvertex is made precise in the following lemma.

Lemma 8 *Let* (H, r) *be a rooted leaf-labelled cactus and let* $B_1, \dots, B_k$ *be the* r*-bridges in* H*. Then*

$$\rho_H[A, B] = m_{b_H}[A, B] + \sum_{i=1}^{k} n_{B_i}.$$

Proof: Let Q be an AB-minimal rooted label sequence of H. By Lemma 4 and repeated application of Lemma 6, the sequence Q is a concatenation of sequences $Q_{j_1} Q_{j_2} \cdots Q_{j_k}$ where Q_i $(i = 1, \dots, k)$ is an $L_{i-1}L_i$-minimal rooted label sequence of (B_i, r) for some labels L_i such that $L_0 = A$ and $L_k = B$, and $(j_1, \dots, j_k)$ is a permutation of $(1, \dots, k)$. For $i = 1, \dots, k$, let p_i be the type of B_{j_i}. Let $R = (L_0, \dots, L_k)$ and $P = (p_1, \dots, p_k)$. Then $m(P, R) = \sum_{i=1}^{k} p_i[L_{i-1}, L_i]$ and $\rho_H[A, B] = m(P, R) + \sum_{i=1}^{k} n_{B_i}$.

Conversely, let $P = (p_1, \dots, p_k)$ be a sequence of k types such that there is a permutation $(j_1, \dots, j_k)$ of $(1, \dots, k)$ such that the type of B_{j_i} is p_i. Let $R = (L_0, \dots, L_k)$ be a sequence of $k+1$ labels and let Q_i be an $L_{i-1}L_i$-minimal rooted label sequence in (B_{j_i}, r). By Lemma 4, $Q = Q_1, Q_2, \dots, Q_k$ is a rooted label sequence of H with $\tau(Q) = m(P, R) + \sum_{i=1}^{k} n_{B_i}$. This completes the proof. □

This gives rise to the following dynamic program. Given a non-zero bridge vector b, there are only t possibilities for the type p of the first bridge whose label sequence starts a minimal label sequence of H (the existence of such a bridge follows from Lemma 4). By deleting the type p from b, we obtain a smaller bridge vector b_p. The value m_{b_p} is computed recursively and then combined with p to obtain m_b. However, using this approach would yield a polynomial-time algorithm that is not fixed parameter tractable (since there are $\Theta(n^t)$ bridge vectors of sum at most n). In the next section, we sidestep this problem and present a linear-time algorithm for computing m_b.

Finally, let us outline an algorithm for Min-Trans that, as we show in the next section, yields Theorem 1. We assume that the input graph has at least three labels to avoid trivialities.

Algorithm 1:

Input: a labelled graph G
Output: the minimum number of transitions $\tau(G)$

Construct the leaf-labelled cactus H that is equivalent to G (Lemma 1).
Root H at an arbitrary unlabelled vertex r.
$\rho_H \longleftarrow$ `Descriptor`(H, r).
Compute $\tau(G)$ from ρ_H.
return $\tau(G)$.

Function Descriptor(H, r)

Input: a rooted leaf-labelled cactus (H, r)
Output: the descriptor ρ_H of H

switch *according to the role of* r **do**
- **case** r *is a leaf and its neighbor* u *is labelled*
  - Note that F has just two vertices r and u.
  - The descriptor ρ_H corresponds to the single-label sequence $\lambda(u)$.
- **case** r *is a leaf and its neighbor* u *is not labelled*
  - $\rho_H \longleftarrow$ `Descriptor`$(H - r, u)$.
- **case** r *is in a cycle* C *and is of degree two*
  - **foreach** $v \in V(C) \setminus \{r\}$ **do**
    - Let B_v be the union of all v-bridges that do not contain C.
    - $\rho_{B_v} \longleftarrow$ `Descriptor`(B_v, v).
  - Use Lemma 7 to compute ρ_H from ρ_{B_v}.
- **case** r *is a cutvertex*
  - **foreach** r*-bridge* B_i **do**
    - $\rho_{B_i} \longleftarrow$ `Descriptor`(B_i, r).
  - Construct the bridge vector b from ρ_{B_i}.
  - Compute m_b.
  - Use Lemma 8 to compute ρ_H from m_b and ρ_{B_i}.

return ρ_H.

Note that throughout Algorithm 1, each vertex of H appears as a root in `Descriptor` at most twice; once as a cutvertex and once either as a leaf or on a cycle. Therefore, each of the cases can happen at most n times, $n = |V(H)|$, and the basic recursion runs in linear time. By Lemma 7, the case when the root is in a cycle takes time $\mathcal{O}(l^3 n)$ since the sum of lengths of all cycles is bounded by n. If we can compute m_b for a bridge vector b in constant time, then Algorithm 1 runs in linear time. This is the goal of the next section.

4 Dealing with Bridge Vectors

In the previous section we have sketched an algorithm for computing the minimum number of label transitions in a planar multigraph when v_0 is nonseparating. In this section we outline an algorithm for computing m_b of a bridge vector b in constant time (Lemma 13), the last ingredient for the proof of Theorem 1. We start by observing that m_b is bounded independently of the bridge vector b. Let us recall that $t = 3^{l^2}$ and t is the number of types.

Lemma 9 *Let b be a bridge vector. Then for every $A, B \in \mathcal{L}$,*

$$m_b[A,B] \leq 2t + 2.$$

Proof: For every type $p \in \{1, \ldots, t\}$, there are labels A_p, B_p such that $p[A_p, B_p] = 0$. By Lemma 2, $p[B_p, A_p] = 0$ as well. Let $k = \mathrm{sum}(b)$ and let $P = (p_1, \ldots, p_k)$ be the sequence of types in b in the increasing order. Let $R = (L_0, \ldots, L_k)$ be the sequence of labels such that $L_0 = A$, $L_k = B$, and for $i = 1, \ldots, k-1$, $L_i = A_{p_i}$ if i is odd and $L_i = B_{p_i}$ if i is even. Note that for $i = 2, \ldots, k-1$, if $p_{i-1} = p_i$, then either $p_i[L_{i-1}, L_i] = p_i[A_{p_i}, B_{p_i}]$ or $p_i[L_{i-1}, L_i] = p_i[B_{p_i}, A_{p_i}]$ and so $p_i[L_{i-1}, L_i] = 0$. Since $p_i[A', B'] \leq 2$ for all labels A', B' and there are at most $t-1$ transitions between different types,

$$m(P,R) = \sum_{i=1}^{k} p_i[L_{i-1}, L_i] \leq 2(t-1) + 4.$$

Thus, $m_b[A,B] \leq m(P,R) \leq 2t+2$. □

We show next that each ordering of b given by an ordering P of types in b and a label sequence $R \in \mathcal{R}_{A,B}(\mathrm{sum}(b))$ corresponds to a walk in a particular multigraph of constant size. Let K be the complete edge-colored and edge-weighted multigraph on vertex set $\mathcal{L}$ where two vertices $A, B \in \mathcal{L}$ are joined by t edges such that the pth edge is colored by p and has weight $p[A,B]$. Note that there are t loops at every vertex of K. For a walk W in K, the *weight* $w(W)$ of W is the sum of weights of edges in W. Let $P = (p_1, \ldots, p_k)$ be an ordering of types in a bridge vector b and $R = (L_0, \ldots, L_k)$ be a sequence of labels. The sequences P, R generate a walk W in K of length k where in the ith step the edge $L_{i-1}L_i$ with color p_i is used. The weight of W is $m(P,R)$. The walk W uses $b(p)$ edges of color p. The converse statement also holds: A walk W that uses $b(p)$ edges of color p gives an ordering P of types in b and a label sequence R such that $m(P,R) = w(W)$. This gives the following lemma.

Lemma 10 *Let b be a bridge vector, A, B labels, and w an integer. There is an AB-walk W of weight w in K such that W uses $b(p)$ edges of color p if and* *only if there is an ordering P of types in b and a label sequence $R = (A, \dots, B)$ such that $m(P, R) = w$.*

In the rest of this section, we will work with multisets, that is, sets where we remember the multiplicity of the elements in the multiset. If an element is not in the multiset, we also say that its multiplicity is 0. In the union A of two multisets A_1 and A_2, $A = A_1 \cup A_2$, the multiplicity of an element a in A is the sum of the multiplicities of a in A_1 and A_2. Similarly, we define the difference of two multisets, denoted $A_1 - A_2$. A multiset A is a submultiset of a multiset B, $A \subseteq B$, if, for every $a \in A$, the multiplicity of a in A is at most the multiplicity of a in B.

For a multigraph G and a set (multiset) S of edges in G, the subgraph $G(S)$ *induced* by S is the graph with edge-set S (the set of edges in S without their multiplicity) and whose vertices are all those vertices of G that are incident with an edge in S. A multiset S of edges of G is *Eulerian* if every vertex in G is incident with an even number of edges in S (counting multiplicities if S is a multiset). We shall use the following well known fact: A multiset S of edges of a multigraph G are edges of some closed walk in G (that uses each edge in S the same number of times as its multiplicity) if and only if S is Eulerian and $G(S)$ is connected.

If a multiset of edges S in G can be extended to an Eulerian multiset by adding some edges from a multiset T, then the number of edges from T that has to be added to S is bounded by $|V(G)|^2$ as is shown in the following lemma.

Lemma 11 *Let S, T be multisets of edges of a multigraph G on k vertices. If S can be extended to an Eulerian multiset by adding some edges in T, then this can be done with at most k^2 edges.*

Proof: Let T' be a subset of T such that $S \cup T'$ is Eulerian. Let T'' be the subset of edges in T' constructed as follows. For every two vertices u and v of G such that T' contains an odd number of edges joining u and v, put into T'' an arbitrary edge from T' connecting u and v. Since, for a vertex $v \in V(G)$, the degree of v in $G(T'')$ and the degree of v in $G(T')$ have the same parity, we have that $S \cup T''$ is also Eulerian. Since there are at most k^2 pairs of vertices of G, we have that $|T''| \leq k^2$. □

In the proof of Lemma 13, we will use the following technical lemma that says that we can always find a small number of cycles in an Eulerian multiset S of edges of K such that the rest of S contains an even number of edges of every color.

Lemma 12 *Let S be an Eulerian multiset of edges in K. Let R be the set of colors p such that there is an odd number of edges in S with color p. Then there is a collection of $k \leq t$ cycles $C_1, \dots, C_k$ in K such that $\cup_{i=1}^{k} E(C) = T \subseteq S$ (where the union is a union of multisets) and there is an odd number of edges of color p in T if and only if $p \in R$.*

Proof: Let $C_1, \ldots, C_m$ be a cycle decomposition of S. For $i = 1, \ldots, m$, let $x_i \in \mathbb{Z}_2^t$ be the binary vector whose pth entry counts the number of edges of color p in C_i modulo 2. Then $\sum_{i=1}^m x_i = x$ where x is the characteristic vector of R, i.e., $x(p) = 1$ if and only if $p \in R$.

The proof proceeds by induction on m. If $m \leq t$, then we are done. Thus $m > t$. Since the dimension of the vector space $\mathbb{Z}_2^t$ is t, there are linearly dependent vectors $x_{i_1}, x_{i_2}, \ldots, x_{i_s}$ such that $\sum_{i=1}^s x_{j_i} = 0$. Remove the edges of the cycles $C_{i_1}, \ldots, C_{i_s}$ from S to obtain an Eulerian multiset S', $S' = S - \cup_{j=1}^s E(C_{i_j})$, with cycle decomposition of length $m - s$. Since $x_{i_1}, \ldots, x_{i_s}$ were linearly dependent, the set R' of colors p such that there is an odd number of edges in S' with color p is equal to R. By the induction hypothesis, there are k cycles $C'_1, \ldots, C'_k$ in K such that $\cup_{i=1}^k E(C'_i) \subseteq S'$ and that have the required parity property. Since $S' \subseteq S$, the edges of these cycles form also a submultiset of S, yielding the result. □

The following lemma shows that, for a bridge vector b, the value m_b can be computed in constant time.

Lemma 13 *Let b be a bridge vector. Then for labels A, B, $m_b[A, B]$ can be computed in time $\mathcal{O}((l^2 t)^{4l^2 t+1})$.*

Proof: By Lemma 10, there is an AB-walk W in K that uses $b(p)$ edges of color p and such that $w(W) = m_b[A, B]$. The walk W can be viewed as a multiset of edges of K (where the multiplicity of each edge equals the number of times the edge appears in W) since every AB-walk using these edges the same number of times has the same weight. We will partition W into two multisets of edges, S and T, such that S contains all the "important" edges in W and the size of S is bounded by a constant.

Let S_1 be the multiset of edges in W that have positive weight. By Lemma 9, $|S_1| \leq w(W) \leq 2t+2$. Let S_2 be the set of all edges of K of weight 0 that appear in W. Note that $|S_2| \leq l^2 t$ since K has l vertices and there are t loops at each vertex and t edges joining each pair of vertices. Let $S' = S_1 \cup S_2$ and let e be an AB-edge in K. By Lemma 11, $S' \cup \{e\}$ can be extended to an Eulerian multiset by adding a set S_3 of at most l^2 edges of $W - S'$. Let $S'' = S' \cup S_3$. Since $S'' \cup \{e\}$ is connected and Eulerian, $S'' \cup \{e\}$ is the edge set of a closed walk in G (by the fact stated before Lemma 11). Thus S'' is the multiset of edges of an AB-walk and $W - S''$ is an Eulerian multiset of edges. By Lemma 12, there is a multiset S_4 of at most lt edges in $W - S''$ such that there is an even number of edges of each color in $W - S'' - S_4$. Let $S = S'' \cup S_4$. Then $|S| \leq 4l^2 t$ since $|S_i| \leq l^2 t$ for each $i = 1, \ldots, 4$. So W can be split into two multisets S and T such that

(C1) $S \cup \{e\}$ is Eulerian and has at most $4l^2 t$ edges,

(C2) $K(S)$ is connected,

(C3) every edge in T is also present in S, (C4) the number of edges of color p in T is even for every color p,

(C5) all edges in T have weight 0.

Let S and T be a decomposition satisfying (C1)–(C5). Let b_T be the bridge vector of T. Now, we will show that, given S and b_T, we can construct a multiset T' with edges given by the same bridge vector b_T as T such that S and T' satisfy (C1)–(C5). A color p is *present* in T' if $b_T(p) > 0$. By (C4), $b_T \equiv 0 \mod 2$. Let $G = K(S)$ be the graph induced by S. By (C3) and (C5), for each color p present in T, G has an edge e_p of color p and weight 0.

Take T' that consists of $b_T(p)$ copies of the edge e_p for each color p present in T. Then S and T' satisfy conditions (C1)–(C5). Since $S \cup T' \cup \{e\}$ is an Eulerian multiset of edges and $K(S \cup T')$ is connected, we obtain that $S \cup T'$ is a multiset of edges of an AB-walk W' in K. Since $w(W') = w(S) = w(W)$, we have that $m_b[A, B] \leq w(W')$ by Lemma 10.

The algorithm generates all possible multisets S and then determines if there is a multiset T satisfying (C1)–(C5) and extending S to an AB-walk. There are l^2t edges in K, so there are at most $\mathcal{O}((l^2t)^{4l^2t})$ choices for a multiset S of at most $4l^2t$ edges. The conditions (C1) and (C2) can be checked in $\mathcal{O}(l^2t)$ time. The bridge vector b_T is computed from S and b in time $\mathcal{O}(t)$. The condition (C4) can hold only if $b_T \equiv 0 \mod 2$ and this can be verified in time $\mathcal{O}(t)$. The conditions (C3) and (C5) can hold only if, for each color p present in T, there is an edge of color p and weight 0 in $K(S)$. This can be verified in time $\mathcal{O}(l^2t)$. Then there exists T such that the decomposition S, T satisfies (C1)–(C5). Thus $m_b[A, B] \leq w(S)$. Since there exists a decomposition S and T with $w(S) = m_b[A, B]$ that satisfies (C1)–(C5), the exhaustive search will eventually find such a set S. Hence, the total time is $\mathcal{O}(l^2t(l^2t)^{4l^2t})$. □

It is likely that a fixed-parameter tractable solution can also be described by the use of min-max algebra for shortest paths, see [5] and [2], [8].

Finally, let us conclude the section by completing the proof of Theorem 1.

Proof: (Proof of Theorem 1.) The proof of the correctness of Algorithm 1 consists of several lemmas. Lemma 1 shows that any input graph can be transformed to an equivalent leaf-labelled cactus. Lemmas 3, 4, and 5 justify our recursive approach for computing the descriptors of rooted cacti.

The linearity of Algorithm 1 was established at the end of Section 3 provided that we can compute m_b in constant time. The cornerstone of the argument was that Lemma 8 allows us to deal with bridge vectors instead with collections of bridges. By Lemma 13, we can compute m_b for a bridge vector in time $\mathcal{O}(l^4t(l^2t)^{4l^2t})$ (applying the lemma for each pair of labels). Since there are at most n cut-vertices in the graph, the algorithm runs in time $\mathcal{O}(l^4t(l^2t)^{4l^2t}n)$. □

5 NP-Completeness

When the number of labels is not bounded, Min-Trans becomes harder. In this section we give a proof of Theorem 2 by providing a polynomial-time reduction from the Hamiltonian Cycle Problem (see [10]).

Ham-Cycle. For a graph G, determine if G contains a hamiltonian cycle.

Proof: (Proof of Theorem 2.) An embedding of G with at most k transitions is a certificate for Min-Trans which asserts that Min-Trans is in NP. To show that Min-Trans is NP-complete, we give a polynomial-time reduction from Ham-Cycle. Let G be a graph of order n. Let H be the graph whose vertex set is $V(H) = \{w\} \cup V(G) \cup (E(G) \times \{0,1\})$. We connect w to each vertex in $V(G)$ and for each edge $uv \in E(G)$ we join one of $(uv, 0)$ and $(uv, 1)$ with u, and the other one with v. Only the leaves of H are labelled. Vertex (e, i) is labelled e. Thus, the number of labels is $|E(G)|$ and each label occurs precisely twice. It is immediate that H can be constructed in polynomial time in $|V(G)|$.

We ask if the number of transitions $\tau(H)$ is smaller or equal to k for

$$k = \sum_{v \in V(G)} (\deg(v) - 1) = 2|E(G)| - |V(G)|.$$

In the affirmative, there is a planar embedding Π of $\widehat{H}$ with $\tau(\Pi) \leq k$. The local rotation around w gives a cyclic order π of vertices of G. Root H at w. By Lemma 4, every label sequence of H is a concatenation of sequences Q_v, $v \in V(G)$, such that Q_v consists of labels on leaves of H attached to v. Since labels in Q_v are the edges adjacent to v, they are different and thus $\tau(Q_v) = \deg(v) - 1$. Hence,

$$\tau(H) \geq \sum_{v \in V(G)} (\deg(v) - 1) = k. \tag{1}$$

To get an equality here, we need that there are no more label transitions between neighboring sequences Q_v.

Let $e_1(v)$ and $e_2(v)$ be the first and the last label in Q_v. We have an equality in (1) if and only if for every two consecutive vertices u, v in π, $e_1(u) = e_2(v)$. This gives a cyclic sequence C of n edges that visits every vertex precisely once. Hence C is a hamiltonian cycle in G.

On the other hand, a hamiltonian cycle C in G gives a cyclic order on vertices of G. This and the cyclic order of the edges of C give a construction of an embedding of $\widehat{H}$ with $\tau(H) = k$. □

References

[1] K. S. Booth and G. S. Lueker. Testing for the consecutive ones property, interval graphs, and graph planarity using PQ-tree algorithms. *J. Comput. System Sci.*, 13:335–379, 1976.

[2] P. Butkovič and R. A. Cuninghame-Green. On matrix powers in max-algebra. *Linear Algebra Appl.*, 421:370–381, 2007.

[3] Cedric Chauve. Private communication.

[4] N. Chiba, T. Nishizeki, S. Abe, and T. Ozawa. A linear algorithm for embedding planar graphs using PQ-trees. *J. Comput. System Sci.*, 30:54–76, February 1985.

[5] R. Cuninghame-Green. *Minimax Algebra*, volume 166. Springer-Verlag Berlin and Heidelberg GmbH & Co. K, New York, Inc, 1979.

[6] R. W. Decker, H. H. Glover, and J. P. Huneke. The genus of the 2-amalgamations of graphs. *J. Graph Theory*, 5(1):95–102, 1981.

[7] R. W. Decker, H. H. Glover, and J. P. Huneke. Computing the genus of the 2-amalgamations of graphs. *Combinatorica*, 5:271–282, September 1985.

[8] M. Gavalec. Linear matrix period in max-plus algebra. *Linear Algebra Appl.*, 307(1-3):167–182, 2000.

[9] J. Hopcroft and R. Tarjan. Efficient planarity testing. *J. ACM*, 21:549–568, October 1974.

[10] R. Karp. Reducibility among combinatorial problems. In R. Miller and J. Thatcher, editors, *Complexity of Computer Computations*, pages 85–103. Plenum Press, 1972.

[11] B. Mohar and C. Thomassen. *Graphs on Surfaces*. Johns Hopkins Univ. Press, Baltimore, MD, 2001.

[12] B. Mohar and P. Škoda. Obstructions for xy-alternating embeddings of graphs in surfaces. *In preparation.*

[13] W.-K. Shih and W.-L. Hsu. A new planarity test. *Theoret. Comput. Sci.*, 223:179–191, July 1999.