MATHEMATICAL ENGINEERING
TECHNICAL REPORTS

Finding a Shortest Non-Zero Path in Group-Labeled Graphs

Yusuke KOBAYASHI and Sho TOYOOKA

(Communicated by Hiroshi HIRAI)

METR 2015–19 May 2015

DEPARTMENT OF MATHEMATICAL INFORMATICS
GRADUATE SCHOOL OF INFORMATION SCIENCE AND TECHNOLOGY
THE UNIVERSITY OF TOKYO
BUNKYO-KU, TOKYO 113-8656, JAPAN

WWW page: http://www.keisu.t.u-tokyo.ac.jp/research/techrep/index.html

Finding a Shortest Non-Zero Path in Group-Labeled Graphs*

Yusuke KOBAYASHI† Sho TOYOOKA‡

May 2015

Abstract: A group-labeled graph is a directed graph with each arc labeled by a group element, and the label of a path is defined as the sum of the labels of the traversed arcs. In this paper, we propose a polynomial time randomized algorithm for the problem of finding a shortest s-t path with a non-zero label in a given group-labeled graph (which we call the Shortest Non-Zero Path Problem). This problem generalizes the problem of finding a shortest path with an odd number of edges, which is known to be solvable in polynomial time by using matching algorithms. In our algorithm for the Shortest Non-Zero Path Problem, we reduce it to the computation of the permanent of a polynomial matrix modulo two. Furthermore, by devising an algorithm for computing the permanent of a polynomial matrix modulo 2^r, we extend our result to the problem of packing internally-disjoint s-t paths.

Keywords: Group-labeled graphs, non-zero shortest path, permanent.

1 Introduction

The shortest path problem is one of the most well-studied problems in combinatorial optimization. In the problem, the objective is to find a shortest path connecting two specified vertices s and t in a given graph, and it can be done easily by the breadth first search if each edge has a unit length. For the shortest path problem in undirected (or directed) graphs with non-negative edge lengths, many polynomial time algorithms are proposed, such as Dijkstra's algorithm [3] and Bellman-Ford algorithm [1]. As an extension of the shortest path problem, we can consider the problem with a parity constraint: given an undirected graph $G = (V, E)$, two specified vertices s and t, and a non-negative length $l(e)$ of each edge $e \in E$, find a shortest odd (or even) s-t path. Here, a path is said to be *odd* (resp. *even*) if it contains odd (resp. even) number of edges. Actually, this problem can be reduced to the weighted matching problem (see e.g. [10, Section 29.11e] and [6]), and hence it can be solved in polynomial time with the aid of weighted matching algorithms. Note that the directed variant is much harder than the undirected case, namely, it is NP-hard to test whether a given directed graph contains an odd (or even) directed path from s to t [8]. We also note that we can easily find a shortest odd (or even) s-t *walk* in a given (directed) graph by a standard dynamic programming.

*Research is supported by JST, ERATO, Kawarabayashi Large Graph Project, and by KAKENHI Grant Number 24106002, 24700004.

†University of Tsukuba, Japan. E-mail: kobayashi@sk.tsukuba.ac.jp

‡University of Tokyo, Japan. E-mail: sho_toyooka@mist.i.u-tokyo.ac.jp

As a generalization of the parity constraints, group-labeled graphs have been investigated [7], where a group-labeled graph is a directed graph with each arc labeled by a group element. In a group-labeled graph, the label of a path is defined as the sum of the labels of the traversed arcs, where each arc can be traversed in the converse direction and then the label is inversed. Group labeled graphs are also called *gain graphs* or *voltage graphs*, and they were originally introduced in the field of topological graph theory with an application to construct graph embeddings in surfaces (see [4, 5, 12]). In this paper, we consider only abelian groups, and hence the group operation is denoted by addition and the identity is denoted by 0. We now introduce the *Shortest Non-Zero Path Problem*, which is described as follows: given a group-labeled graph with two specified vertices s and t, find an s-t path with a non-zero label that contains minimum number of arcs. This generalizes the shortest odd s-t path problem in undirected graphs with unit length edges, because odd s-t paths in an undirected graph G are corresponding to non-zero s-t paths in the $\mathbb{Z}_2$-labeled graph obtained from G by orienting each edge arbitrarily and by setting the label of each arc as 1. In this paper, we propose a polynomial time randomized algorithm for the Shortest Non-Zero Path Problem.

In order to state our result formally, we now give some notations. For an abelian group Γ, a Γ-*labeled graph* is a pair (G, ψ) of a directed graph $G = (V, E)$ and a mapping $\psi : E \to \Gamma$ (called a *label function*). A *walk* in a Γ-labeled graph (G, ψ) is a sequence $W = (v_0, e_1, v_1, e_2, v_2, \dots, e_l, v_l)$ of vertices v_i and arcs e_i in G such that either $e_i = v_{i-1}v_i$ or $e_i = v_i v_{i-1}$ for $i = 1, \dots, l$. A *path* is a walk whose vertices are distinct from one another. The *label of a walk* $W = (v_0, e_1, v_1, \dots, e_l, v_l)$ is defined as $\psi(W) = \tilde{\psi}(e_1) + \tilde{\psi}(e_2) + \cdots + \tilde{\psi}(e_l)$, where $\tilde{\psi}(e_i) = \psi(e_i)$ if $e_i = v_{i-1}v_i$ and $\tilde{\psi}(e_i) = -\psi(e_i)$ if $e_i = v_i v_{i-1}$. The arc set of a walk W is denoted by $E(W)$. With these notations, the Shortest Non-Zero Path Problem and our result are described as follows.

> Shortest Non-Zero Path Problem in Γ-labeled Graphs
>
> **Input:** a Γ-labeled graph (G, ψ) with two specified vertices $s, t \in V$.
>
> **Find:** an s-t path P with $\psi(P) \neq 0$ that contains a minimum number of arcs (if exists).

Theorem 1. *Let Γ be a fixed finite abelian group. There is a polynomial time randomized algorithm for the Shortest Non-Zero Path Problem in Γ-labeled Graphs.*

In what follows, by subdividing all arcs and assigning appropriate labels if necessary, we assume that the input graph contains neither self-loops nor parallel arcs without loss of generality.

Note that we can easily find a shortest s-t walk with a non-zero label in polynomial time by a standard dynamic programming. However, the same argument cannot be applied to the Shortest Non-Zero Path problem.

The rest of this paper is organized as follows. In Section 2, we give an algebraic algorithm for the Shortest Non-Zero Path Problem and prove Theorem 1. In Section 3, we extend our result to a kind of path packing problem, which we call the Shortest Non-Zero k Disjoint Paths Problem. In the algorithm for the Shortest Non-Zero k Disjoint Paths Problem, we use an algorithm for computing the permanent of a polynomial matrix modulo 2^r, which is given in Section 4.

2 Algebraic Approach to the Problem

In this section, we give a proof of Theorem 1, namely, we propose an algebraic approach to the Shortest Non-Zero Path Problem, in which we use the permanent of a polynomial matrix. The *permanent* of an $n \times n$ matrix $A = (a_{ij})$ is defined as

$$\mathrm{per}\, A = \sum_{\sigma \in S_n} \prod_{i=1}^{n} a_{i\sigma(i)},$$

where S_n is the set of all permutations on n elements. By the definition, the permanent of the adjacency matrix of a directed graph is corresponding to the number of cycle covers in this directed graph, where a cycle cover is a set of arcs in which each vertex has exactly one incoming arc and exactly one outgoing arc. More generally, we can easily see the following.

Lemma 2. *Let $G = (V, E)$ be a directed graph which has no multiple arcs and may have self-loops. Let $A = (a_{ij})$ be a matrix whose rows and columns are indexed by V such that $a_{ij} = 0$ holds for any $i, j \in V$ with $ij \notin E$. Then, we have*

$$\mathrm{per}\, A = \sum_{F \in \mathcal{C}(G)} \prod_{ij \in F} a_{ij},$$

where $\mathcal{C}(G)$ is the set of all cycle covers in G.

To prove Theorem 1, we first deal with the case of $\Gamma = \mathbb{Z}_p (:= \mathbb{Z}/p\mathbb{Z})$ for some p. We extend this case to the general case by using the fundamental theorem of finite abelian groups.

Suppose that we are given an instance of the Shortest Non-Zero Path Problem, that is, we are given a $\mathbb{Z}_p$-labeled graph $(G = (V, E), \psi)$ with two specified vertices $s, t \in V$. By identifying $\mathbb{Z}_p$ with $\{0, 1, 2, \dots, p-1\} \subseteq \mathbb{Z}$, for each $ij \in E$, we regard $\psi(ij)$ as an integer with $0 \le \psi(ij) \le p-1$. We define a matrix $A = (a_{ij})$ over $\mathbb{Z}[x, y]$ whose rows and columns are indexed by V as follows:

$$a_{ij} = \begin{cases} xy^{\psi(ij)} & \text{if } ij \in E,\ i \neq t, \text{ and } j \neq s; \\ xy^{p-\psi(ji)} & \text{if } ji \in E,\ i \neq t, \text{ and } j \neq s; \\ 1 & \text{if } i = j \in V \setminus \{s, t\}; \\ 1 & \text{if } (i, j) = (t, s); \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

Note that since G has neither self-loops nor parallel arcs, $ij \in E$ implies that $i \neq j$ and $ji \notin E$, which ensures that a_{ij} is well-defined. Since the maximum degree of y in $\mathrm{per}\, A$ is at most $|V|p$, $\mathrm{per}\, A$ can be uniquely expressed as

$$\mathrm{per}\, A = \sum_{l=0}^{|V|p} q_l(x) y^l,$$

where $q_l(x)$ is a polynomial in x with integer coefficients. With these polynomials, we define $Q(x)$ as the polynomial with coefficients in $\{0, 1\}$ such that

$$Q(x) \equiv \sum_{l \not\equiv 0 \pmod{p}} q_l(x) \pmod{2}, \tag{2}$$

where we denote $\sum_i b_i x^i \equiv \sum_i c_i x^i \pmod{2}$ if $b_i \equiv c_i \pmod{2}$ for every i.

Lemma 3. *For a $\mathbb{Z}_p$-labeled graph (G, ψ) with two vertices s and t, $Q(x)$ defined above can be computed in polynomial time.*

Proof. In order to compute $Q(x)$, we only need the value of $\operatorname{per} A$ modulo two, which can be computed as follows:

1. replace y with x^N to obtain an one-variable polynomial matrix A', where N is greater than the maximum degree of x in $\operatorname{per} A$ (e.g., $N := n + 1$),
2. compute $\operatorname{per} A'$ modulo two, and
3. replace x^{aN+b} with $x^b y^a$ in $\operatorname{per} A'$ to obtain $\operatorname{per} A$ modulo two.

Since we can compute the permanent of one-variable polynomial matrix modulo two in polynomial time (see [2] or Section 4), this algorithm runs in polynomial time. □

The following proposition shows a relationship between $Q(x)$ and the Shortest Non-Zero Path Problem.

Proposition 4. *Suppose that we are given a $\mathbb{Z}_p$-labeled graph (G, ψ) with two vertices s and t, which is an instance of the Shortest Non-Zero Path Problem. Assume that it has a unique optimal solution. Then, the optimal value of this instance is equal to the minimum degree of $Q(x)$ defined as above.*

Proof. For an instance $(G = (V, E), \psi, s, t)$ of the Shortest Non-Zero Path Problem, we construct a new directed graph $G' = (V', E')$ with labels from G as follows.

- For each arc $ij \in E$, add a new arc ji with the label $-\psi(ij)$.
- For each vertex $v \in V \setminus \{s, t\}$, add a self-loop incident to v with the label 0.
- Remove all arcs entering s and leaving t.
- Add a new arc ts with the label 0.

By abusing notation, the label function of G' is also denoted by ψ. Since G' and the matrix A defined as (1) satisfy the condition in Lemma 2, i.e., $ij \notin E'$ implies that $a_{ij} = 0$, we obtain

$$\operatorname{per} A = \sum_{F \in \mathcal{C}(G')} \prod_{ij \in F} a_{ij}, \tag{3}$$

where $\mathcal{C}(G')$ is the set of all cycle covers in G'. We observe that a cycle cover $F \in \mathcal{C}(G')$ must contain the arc ts, and hence F also contains a path P from s to t. We now divide $\mathcal{C}(G')$ into two parts: one is the set $\mathcal{C}_1$ of all cycle covers containing an s-t path P with $\psi(P) \neq 0$, and the other is the set $\mathcal{C}_2$ of all cycle covers containing an s-t path P with $\psi(P) = 0$. By (3), for each cycle cover $F \in \mathcal{C}(G')$, we can naturally define the contribution of F to $Q(x)$, say $Q_F(x)$. That is, $Q_F(x) = 0$ if $\sum_{e \in F} \psi(e) \equiv 0 \pmod{p}$, and $Q_F(x) = x^{c_F}$ otherwise, where c_F is the number of arcs ij in F such that $i \neq j$ and $(i, j) \neq (t, s)$. Then, we have $Q(x) \equiv \sum_{F \in \mathcal{C}(G')} Q_F(x) \pmod{2}$ by the definition. In what follows, we consider $\sum_{F \in \mathcal{C}_1} Q_F(x)$ and $\sum_{F \in \mathcal{C}_2} Q_F(x)$, separately.

First, we consider $\sum_{F \in \mathcal{C}_1} Q_F(x)$. For an s-t path P, let A_P be the matrix obtained from A by eliminating the rows and the columns corresponding to the vertices in P. Since each $F \in \mathcal{C}_1$ contains a non-zero s-t path, we have

$$\begin{aligned} \sum_{F \in \mathcal{C}_1} \prod_{ij \in F} a_{ij} &= \sum_{P:\text{ non-zero } s\text{-}t \text{ path}} \left(\prod_{ij \in E(P)} a_{ij} \right) \operatorname{per} A_P \\ &= \sum_{P:\text{ non-zero } s\text{-}t \text{ path}} x^{|E(P)|} y^{\psi'(P)} \operatorname{per} A_P, \end{aligned} \tag{4}$$

where $\psi'(P)$ is some integer with $\psi'(P) \equiv \psi(P) \pmod p$. Consider the cycle cover $F_0 \in \mathcal{C}_1$ consisting of the unique optimal solution (the shortest non-zero path) P_0 of the original problem, arc ts, and self-loops incident to vertices in $G' - P$. Then, $Q_{F_0}(x) = x^{|E(P_0)|}$. By the uniqueness of the optimal solution and (4), we can see that $x^{|E(P_0)|}$ is the minimum degree term in $\sum_{F \in \mathcal{C}_1} Q_F(x)$ and its coefficient is 1.

Next, we show $\sum_{F \in \mathcal{C}_2} Q_F(x) \equiv 0 \pmod 2$. Let $F \in \mathcal{C}_2$ be a cycle cover satisfying that $Q_F(x) \neq 0$. By the definition of $Q_F(x)$, $\sum_{e \in F} \psi(e) \not\equiv 0 \pmod p$ and $Q_F(x) = x^{c_F}$, where c_F is the number of arcs ij in F such that $i \neq j$ and $(i,j) \neq (t,s)$. Let P be the s-t path with the label zero in F. We consider the cycle cover $F' \in \mathcal{C}_2$ obtained from F by reversing all arcs in $F - E(P) - ts$. Since $\sum_{e \in F'} \psi(e) \equiv -\sum_{e \in F} \psi(e) \not\equiv 0 \pmod p$, we have $Q_{F'}(x) = x^{c_F}$, and hence $Q_F(x) + Q_{F'}(x) \equiv 0 \pmod 2$. Note that $F \neq F'$, because $F - E(P) - ts$ contains at least one arc that is not a self-loop. In this way, all cycle covers F in $\mathcal{C}_2$ with $Q_F(x) \neq 0$ can be put into pairs so that the total contribution of each pair to $Q(x)$ is zero modulo two. Therefore, we obtain $\sum_{F \in \mathcal{C}_2} Q_F(x) \equiv 0 \pmod 2$.

By the above analyses of $\sum_{F \in \mathcal{C}_1} Q_F(x)$ and $\sum_{F \in \mathcal{C}_2} Q_F(x)$, the minimum degree of $Q(x)$ is equal to the minimum length of the non-zero s-t path. □

By combining Lemma 3 and Proposition 4, we obtain a deterministic polynomial time algorithm for the Shortest Non-Zero Path Problem under the assumption that the instance has a unique optimal solution. Even when a given instance has more than one optimal solutions, we can convert it to the case with a unique optimal solution by perturbing the lengths of the arcs.

Proposition 5. *Suppose that we are given a Γ-labeled graph (G, ψ) with two vertices s and t, which is an instance of the Shortest Non-Zero Path Problem. We construct a new instance by replacing each arc e with a path of length $w(e)$, where $w(e)$ is chosen independently and uniformly at random from $W := \{2|V||E|, 2|V||E|+1, \dots, 2|V||E|+2|E|-1\}$. Here, the labels of the new arcs are chosen so that the label of the path is equal to $\psi(e)$. Then, the obtained instance has a unique optimal solution with probability at least $\frac{1}{2}$ (if the original instance has a feasible solution).*

Proof. The validity of this proposition is based on the following *isolation lemma* [9]:

Lemma 6. *Let S be a finite set, $\mathcal{F}$ be a family of subsets of S, and W be a set of integers different from each other. Suppose that the weight of each element in S is chosen from W independently and uniformly at random, then with probability at least $1 - \frac{|S|}{|W|}$, there is a unique set in $\mathcal{F}$ of minimum total weight.*

We apply this lemma, in which $S = E$, $W = \{2|V||E|, 2|V||E|+1, \dots, 2|V||E|+2|E|-1\}$, and $\mathcal{F}$ is the family of all subsets of E belonging to each s-t non-zero path in G. Then, with probability at least $1 - \frac{|E|}{2|E|} = \frac{1}{2}$, there is a unique s-t non-zero path of minimum weight in G. Since the weight of an s-t path in G is equal to the length of the corresponding path in the new instance, the obtained instance has a unique optimal solution with probability at least $\frac{1}{2}$. □

Since an optimal solution in the instance obtained in Proposition 5 is corresponding to an optimal solution in the original instance, by Lemma 3 and Propositions 4 and 5, we obtain Theorem 1 under the assumption that $\Gamma = \mathbb{Z}_p$ for some integer p.

We now consider the case when Γ is a finite abelian group. In this case, we apply the fundamental theorem of finite abelian groups and decompose Γ as $\Gamma = \mathbb{Z}_{p_1} \oplus \mathbb{Z}_{p_2} \oplus \cdots \oplus \mathbb{Z}_{p_r}$ where $p_1, \dots, p_r$ are some integers. By using this decomposition, for an instance (G, ψ, s, t) of the Shortest Non-Zero Path Problem in Γ-labeled Graphs, define a new label function $\psi_i : E \to \mathbb{Z}_{p_i}$ for $i = 1, 2, \dots, r$ such that $\psi(e) = \psi_1(e) \oplus \psi_2(e) \oplus \cdots \oplus \psi_r(e)$ for $e \in E$. For $i = 1, 2, \dots, r$, let P_i be a shortest non-zero

s-t path in (G, ψ_i). Then, a shorest one among $\{P_1, P_2, \ldots, P_r\}$ is a shortest non-zero s-t path in (G, ψ), because a path P satisfies that $\psi(P) \neq 0$ if and only if $\psi_i(P) \neq 0$ for some $i \in \{1, 2, \ldots, r\}$. Therefore, by solving the Shortest Non-Zero Path Problem in $\mathbb{Z}_{p_i}$-labeled Graphs for $i = 1, 2, \ldots, r$, we obtain an optimal solution of the original problem in (G, ψ), which shows Theorem 1.

3 Extension to Packing Disjoint s-t Paths

In this section, we generalize the Shortest Non-Zero Path Problem to the problem of finding k internally-disjoint s-t paths of shortest total length under the condition that the sum of their labels is not zero. The problem is formally described as follows, where k is a positive integer and Γ is a finite abelian group.

Shortest Non-Zero k Disjoint Paths Problem in Γ-labeled Graphs

Input: a Γ-labeled graph (G, ψ) with two specified vertices $s, t \in V$.

Find: k internally-disjoint s-t paths $P_1, \ldots, P_k$ minimizing the total number of arcs contained in them subject to $\sum_{i=1}^{k} \psi(P_i) \neq 0$ (if exist).

We can easily see that the case of $k = 1$ is corresponding to the Shortest Non-Zero Path Problem. The objective of this section is to extend Theorem 1 to the following theorem.

Theorem 7. *Let k be a fixed positive integer and Γ be a fixed finite abelian group. There is a polynomial time randomized algorithm for the Shortest Non-Zero k Disjoint Paths Problem in Γ-labeled Graphs.*

Proof. By subdividing all arcs and assigning appropriate labels if necessary, we may assume that the input graph contains neither self-loops nor parallel arcs and there is no arc connecting s and t without loss of generality. By using the same argument as the previous section, if suffices to discuss the case of $\Gamma = \mathbb{Z}_p$. Suppose that we are given an instance of the Shortest Non-Zero k Disjoint Paths Problem. We construct a new graph $G' = (V', E')$ from G by replacing s with its k copies $s_1, s_2, \ldots, s_k$ and by replacing t with its k copies $t_1, t_2, \ldots, t_k$. Note that each arc incident to s (resp. t) is also replaced with its k copies incident to s_i (resp. t_i), and the label function ψ on E is naturally extended to E'. Define $S = \{s_1, s_2, \ldots, s_k\}$ and $T = \{t_1, t_2, \ldots, t_k\}$.

Recall that, for each $ij \in E'$, we can regard $\psi(ij)$ as an integer with $0 \leq \psi(ij) \leq p - 1$. We define a matrix $A' = (a'_{ij})$ over $\mathbb{Z}[x, y]$ whose rows and columns are indexed by V' as follows:

$$a'_{ij} = \begin{cases} xy^{\psi(ij)} & \text{if } ij \in E',\ i \notin T, \text{ and } j \notin S; \\ xy^{p-\psi(ji)} & \text{if } ji \in E',\ i \notin T, \text{ and } j \notin S; \\ 1 & \text{if } i = j \in V' \setminus (S \cup T); \\ 1 & \text{if } (i, j) = (t_l, s_l) \text{ for some } l \in \{1, 2, \ldots, k\}; \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

In a similar way to (2), we express per A' as

$$\operatorname{per} A' = \sum_{l=0}^{|V'|p} q'_l(x) y^l$$

and define $Q'(x)$ as the polynomial with coefficients in $\{0, 1, 2, \dots, 2^r - 1\}$ such that

$$Q'(x) \equiv \sum_{l \not\equiv 0 \pmod p} q'_l(x) \pmod{2^r}.$$

Here, r is the minimal integer such that $(k!)^2/2^r$ is not an integer. In a similar way to Proposition 4, we can obtain the following proposition.

Proposition 8. *Let k be a positive integer. Suppose that we are given a $\mathbb{Z}_p$-labeled graph $G = (V, E)$ with two vertices s and t, which is an instance of the Shortest Non-Zero k Disjoint Paths Problem. Assume that it has a unique optimal solution. Then, the optimal value of this instance is equal to the minimum degree of $Q'(x)$ defined as above.*

Proof of Proposition 8. We construct a new graph $G'' = (V'', E'')$ with labels from $G' = (V', E')$ as follows.

- For each arc $ij \in E$, add a new arc ji with the label $-\psi(ij)$.
- For each vertex $v \in V \setminus (S \cup T)$, add a self-loop incident to v with the label 0.
- Remove all arcs entering S and leaving T.
- Add a new arc $t_l s_l$ with the label 0 for $l = 1, 2, \dots, k$.

By abusing notation, the label function of G'' is also denoted by ψ.

Since G'' and the matrix A' defined as (5) satisfy the condition in Lemma 2, i.e., $ij \notin E''$ implies that $a'_{ij} = 0$, we obtain

$$\operatorname{per} A' = \sum_{F \in \mathcal{C}(G'')} \prod_{ij \in F} a'_{ij},$$

where $\mathcal{C}(G'')$ is the set of all cycle covers in G''. We observe that a cycle cover $F \in \mathcal{C}(G'')$ must contain the arc $t_l s_l$ for $l = 1, 2, \dots, k$, and hence F also contains k (fully) disjoint paths from S to T, which we call *S-T paths*. We now divide $\mathcal{C}(G'')$ into two parts: one is the set $\mathcal{C}'_1$ of all cycle covers containing S-T paths whose sum of the labels is non-zero (*non-zero S-T paths*), and the other is the set $\mathcal{C}'_2$ of all cycle covers containing S-T paths whose sum of the labels is zero (*zero S-T paths*). In the same way as the proof of Proposition 4, for each cycle cover $F \in \mathcal{C}(G'')$, we can naturally define the contribution of F to $Q'(x)$, say $Q'_F(x)$. In what follows, we consider $\sum_{F \in \mathcal{C}'_1} Q'_F(x)$ and $\sum_{F \in \mathcal{C}'_2} Q'_F(x)$, separately.

First, we consider $\sum_{F \in \mathcal{C}'_1} Q'_F(x)$. For a set $\mathcal{P}$ of paths, let $E(\mathcal{P})$ be the set of the arcs contained in the paths in $\mathcal{P}$, define $\psi(\mathcal{P}) := \sum_{e \in E(\mathcal{P})} \psi(e)$, and let $A'_{\mathcal{P}}$ be the matrix obtained from A' by eliminating the rows and the columns corresponding to the vertices in the paths in $\mathcal{P}$. Since each $F \in \mathcal{C}'_1$ contains non-zero S-T paths, we have

$$\sum_{F \in \mathcal{C}'_1} \prod_{ij \in F} a'_{ij} = \sum_{\mathcal{P}\text{: non-zero } S\text{-}T \text{ paths}} x^{|E(\mathcal{P})|} y^{\psi'(\mathcal{P})} \operatorname{per} A'_{\mathcal{P}}, \tag{6}$$

where $\psi'(\mathcal{P})$ is some integer with $\psi'(\mathcal{P}) \equiv \psi(\mathcal{P}) \pmod p$. Let $\mathcal{P}_0$ be the unique optimal solution of the Shortest Non-Zero k Disjoint Paths Problem. Consider a cycle cover $F_0 \in \mathcal{C}'_1$ in G'' that consists of non-zero S-T paths $\mathcal{P}$ corresponding to $\mathcal{P}_0$, arcs $t_l s_l$ $(l = 1, \dots, k)$, and self-loops incident to vertices not contained in $\mathcal{P}$. Then, $Q'_{F_0}(x) = x^{|E(\mathcal{P}_0)|}$. Since we have $(k!)^2$ choices of F_0 with this condition, by the uniqueness of the optimal solution and (6), $(k!)^2 x^{|E(\mathcal{P}_0)|}$ is the minimum degree term in $\sum_{F\in\mathcal{C}'_1} Q'_F(x)$. Note that $(k!)^2 \not\equiv 0 \pmod{2^r}$ by the definition of r.

Next, we show $\sum_{F\in\mathcal{C}'_2} Q'_F(x) \equiv 0 \pmod{2^r}$. Let $F \in \mathcal{C}'_2$ be a cycle cover satisfying that $Q'_F(x) \neq 0$. Then, $\sum_{e\in F}\psi(e) \not\equiv 0 \pmod p$ and $Q'_F(x) = x^{c_F}$, where c_F is the number of arcs ij in F such that $i \neq j$ and $(i,j) \neq (t_l, s_l)$. By changing the indices of $\{s_1,\dots,s_k\}$ and $\{t_1,\dots,t_k\}$ in F, we obtain $(k!)^2$ cycle covers $F_1(=F), F_2, \dots, F_{(k!)^2} \in \mathcal{C}'_2$ such that $Q'_{F_i}(x) = x^{c_F}$ for $i = 1,2,\dots,(k!)^2$. Note that these cycle covers are distinct since the original graph has no arc connecting s and t. Let $\mathcal{P}$ be the zero S-T paths in F, and consider the cycle cover $F' \in \mathcal{C}'_2$ obtained from F by reversing all arcs in $F - E(\mathcal{P}) - \{t_1 s_1, \dots, t_k s_k\}$. Since $\sum_{e\in F'}\psi(e) \equiv -\sum_{e\in F}\psi(e) \not\equiv 0 \pmod p$, we have $Q_{F'}(x) = x^{c_F}$. Again, by changing the indices of $\{s_1,\dots,s_k\}$ and $\{t_1,\dots,t_k\}$ in F', we have $(k!)^2$ cycle covers $F'_1(=F'), F'_2, \dots, F'_{(k!)^2} \in \mathcal{C}'_2$ such that $Q'_{F'_i}(x) = x^{c_F}$ for $i = 1,2,\dots,(k!)^2$. Therefore, $\sum_{i=1}^{(k!)^2}(Q_{F_i}(x) + Q_{F'_i}(x)) \equiv 0 \pmod{2^r}$ by the definition of r. In this way, all cycle covers F in $\mathcal{C}'_2$ with $Q'_F(x) \neq 0$ can be divided into sets of $2(k!)^2$ cycle covers so that the total contribution of each set to $Q'(x)$ is zero modulo 2^r. Therefore, we obtain $\sum_{F\in\mathcal{C}'_2} Q'_F(x) \equiv 0 \pmod{2^r}$.

By the above analyses of $\sum_{F\in\mathcal{C}'_1} Q'_F(x)$ and $\sum_{F\in\mathcal{C}'_2} Q'_F(x)$, the minimum degree of $Q'(x)$ is equal to the optimal value of the Shortest Non-Zero k Disjoint Paths Problem. □

We can compute $Q'(x)$ modulo 2^r in polynomial time as we will see in the next section. Therefore, by Proposition 8 and the perturbation technique used in Section 2, we obtain Theorem 7. □

4 Computing the Permanent Modulo 2^r

For the computation of $Q'(x)$, we propose an algorithm for computing the permanent of polynomial matrices modulo 2^r, which we believe is of independent interest.

Although computing the permanent of integer matrices is NP-hard [11], Valiant [11] gave a polynomial time algorithm for computing the permanent of matrices whose entries are in $\mathbb{Z}_{2^r}$, where r is a fixed constant. By using a similar technique to [11], Björklund [2] gave a polynomial time algorithm for computing the permanent of matrices whose entries are in $\mathbb{Z}_4[x]$, that is, each entry is a polynomial in x with coefficients in $\mathbb{Z}_4$. Our contribution is to generalize this result to the case of $\mathbb{Z}_{2^r}[x]$, where r is a fixed constant. For a matrix A whose entries are in $\mathbb{Z}[x]$ and for a positive integer r, let $\mathrm{per}_{2^r} A$ be the permanent of A modulo 2^r, i.e., the polynomial with coefficients in $\{0,1,2,\dots,2^r-1\}$ such that

$$\mathrm{per}_{2^r} A \equiv \mathrm{per}\, A \pmod{2^r}.$$

Our result is stated as follows.

Theorem 9. *Let r be a fixed nonnegative integer and A be an $n\times n$ matrix whose entries are in $\mathbb{Z}[x]$. Suppose that we are given an integer N which is greater than the maximum degree of* $\mathrm{per}_{2^r} A$. *Then,* $\mathrm{per}_{2^r} A$ *can be computed in polynomial time in n and N.*

Proof. Our proof is based on ideas in [2]. Let E_N denote $\mathbb{Z}[x]/(x^N)$, which is a quotient ring divided by the ideal generated by x^N. Roughly, E_N is the set of polynomials obtained from $\mathbb{Z}[x]$ by ignoring the terms whose degrees are at least N. Since the maximum degree of $\mathrm{per}_{2^r} A$ is at most $N-1$, to compute $\mathrm{per}_{2^r} A$, we may identify $\mathbb{Z}[x]$ with E_N by ignoring the terms whose degrees are at least N. Let $\mathrm{M}_n(E_N)$ be the set of all $n\times n$ matrices whose entries are in E_N. We say that a polynomial $a \in E_N$ is *even* if all coefficients of a are even and *odd* if a is not even. For an odd polynomial a, let $m(a)$ be the index of the lowest order term of a whose coefficient is odd.

For a given matrix $A = (a_{ij}) \in \mathrm{M}_n(E_N)$, our algorithm for computing $\mathrm{per}_{2^r} A$ is described as follows. Note that all the computation in the algorithm is done on E_N, that is, we remove all terms whose degrees are at least N.
Algorithm PERMANENT(r, A)

A1. If $n = 1$, return a_{11} modulo 2^r. If $r = 0$, return 0.

A2. Choose $i \in \{1, 2, \ldots, n\}$ such that a_{i1} is odd and $m(a_{i1})$ is minimum (if exists). Then, exchange rows 1 and i.

A3. If a_{i1} is even for $i = 2, 3 \ldots, n$, then compute $\mathrm{per}_{2^r} A$ by Lemma 10 and return it. Otherwise, take an index $i \in \{2, 3, \ldots, n\}$ such that a_{i1} is odd, and compute a polynomial $c \in E_N$ such that $a_{i1} + ca_{11} \in E_N$ is even by Lemma 11.

A4. Let $A[i, 1]$ be the matrix obtained from A by replacing the ith row with the first row. Compute $\mathrm{per}_{2^r}(A + cA[i, 1])$ by using Algorithm PERMANENT$(r, A + cA[i, 1])$ recursively and compute $c\mathrm{per}_{2^r} A[i, 1]$ by Lemma 12. Then, compute $\mathrm{per}_{2^r} A$ by

$$\mathrm{per}_{2^r} A \equiv \mathrm{per}_{2^r}(A + cA[i, 1]) - c\mathrm{per}_{2^r} A[i, 1] \pmod{2^r},$$

and return it.

For integers $n \geq 1$, $r \geq 0$, and $k \geq 0$, let $T_N(n, r, k)$ be the worst case running time of the algorithm for computing $\mathrm{per}_{2^r} A$ for a matrix $A = (a_{ij}) \in \mathrm{M}_n(E_N)$ such that $|\{i \in \{1, 2, \ldots n\} \mid a_{i1} \text{ is odd}\}|$ is at most k. Note that T_N is monotone, that is, $T_N(n, r, k) \geq T_N(n', r', k')$ if $n \geq n'$, $r \geq r'$, and $k \geq k'$. For each n and each r, let $T^*_N(n, r) := \max_k T_N(n, r, k) (= T_N(n, r, n))$. In what follows, we prove that $T^*_N(n, r)$ is bounded by a polynomial in n and N for fixed r. Let $\mathrm{poly}(n, N)$ denote some polynomial in n and N. Note that when $\mathrm{poly}(n, N)$ appears more than once, they might denote distinct polynomials.

The following lemmas are used in Algorithm PERMANENT(r, A).

Lemma 10. *Let $n \geq 2$ and $r \geq 1$ be integers and $A = (a_{ij})$ be a matrix in $\mathrm{M}_n(E_N)$. If a_{i1} is even for $i = 2, 3 \ldots, n$, then we can compute $\mathrm{per}_{2^r} A$ in $T^*_N(n-1, r) + (n-1)T^*_N(n-1, r-1) + \mathrm{poly}(n, N)$ time. That is, $T_N(n, r, 1) \leq T^*_N(n-1, r) + (n-1)T^*_N(n-1, r-1) + \mathrm{poly}(n, N)$ for $n \geq 2$ and $r \geq 1$.*

Proof. By expanding $\mathrm{per} A$ along the first column, we have

$$\mathrm{per}_{2^r} A \equiv a_{11}\mathrm{per}_{2^r} A_{11} + \sum_{i=2}^{n} a_{i1}\mathrm{per}_{2^r} A_{i1} \pmod{2^r}, \tag{7}$$

where A_{i1} is the matrix obtained from A by removing row i and column 1. For $i = 2, 3, \ldots, n$, since a_{i1} is even, we have

$$a_{i1}\mathrm{per}_{2^r} A_{i1} \equiv a_{i1}\mathrm{per}_{2^{r-1}} A_{i1} \pmod{2^r}.$$

This shows that we can compute (7) in $T^*_N(n-1, r) + (n-1)T^*_N(n-1, r-1) + \mathrm{poly}(n, N)$ time. $\square$

Lemma 11. *For odd polynomials $a \in E_N$ and $b \in E_N$ with $m(a) \leq m(b)$, we can compute a polynomial $c \in E_N$ such that $b + ca \in E_N$ is even in polynomial time in N.*

Proof. Such a c can be computed by the following algorithm.

B1. Set $l = 0$ and $c^{(0)} = 0 \in E_N$.

B2. While $b + c^{(l)}a$ is not even, set $c^{(l+1)} = c^{(l)} + x^{m(b+c^{(l)}a)-m(a)}$ and increment l.

B3. Return $c^{(l)}$.

Since each iteration in Step B2 increases $m(b+c^{(l)}a)$ by at least one and this value is at most $N-1$, this algorithm runs in polynomial time in N. $\square$

Lemma 12. *Let $n \geq 2$ and $r \geq 1$ be integers and $A = (a_{ij})$ be a matrix in $\mathrm{M}_n(E_N)$ whose first and second rows are identical. Then,*

$$\mathrm{per}_{2^r} A \equiv 2 \sum_{1 \leq i < j \leq n} a_{1i} a_{2j} \mathrm{per}_{2^{r-1}} A_{1i,2j} \pmod{2^r},$$

where $A_{1i,2j}$ is the matrix obtained from A by removing rows 1 and 2 and columns i and j. Furthermore, $\mathrm{per}_{2^r} A$ can be computed in $\frac{1}{2}n(n-1)T^*_N(n-2, r-1) + \mathrm{poly}(n, N)$ time, where $T^*_N(0, r-1)$ is regarded as a constant.

Proof. By expanding per$A$ along the first and second rows,

$$\begin{aligned}
\mathrm{per}_{2^r} A &\equiv \sum_{i \neq j} a_{1i} a_{2j} \mathrm{per}_{2^r} A_{1i,2j} \\
&\equiv 2 \sum_{1 \leq i < j \leq n} a_{1i} a_{2j} \mathrm{per}_{2^r} A_{1i,2j} \\
&\equiv 2 \sum_{1 \leq i < j \leq n} a_{1i} a_{2j} \mathrm{per}_{2^{r-1}} A_{1i,2j} \pmod{2^r},
\end{aligned}$$

where the last equality is derived from the fact that $2a \equiv 2a' \pmod{2^r}$ if and only if $a \equiv a' \pmod{2^{r-1}}$ for $a, a' \in E_N$. Since $\mathrm{per}_{2^{r-1}} A_{1i,2j}$ can be computed in $T^*_N(n-2, r-1)$ time, $\mathrm{per}_{2^r} A$ can be computed in $\frac{1}{2}n(n-1)T^*_N(n-2, r-1) + \mathrm{poly}(n, N)$ time. $\square$

Now we are ready to evaluate $T_N(n, r, k)$ and prove Theorem 9. For $k \geq 2$ and $r \geq 1$, by Step A4 of Algorithm Permanent(r, A) and Lemma 12, we obtain

$$T_N(n, r, k) \leq T_N(n, r, k-1) + \frac{1}{2}n(n-1)T^*_N(n-2, r-1) + \mathrm{poly}(n, N).$$

By using this inequality repeatedly, it holds that

$$\begin{aligned}
T_N(n, r, k) &\leq T_N(n, r, 1) + \frac{k-1}{2}n(n-1)T^*_N(n-2, r-1) + \mathrm{poly}(n, N) \\
&\leq T_N(n, r, 1) + \frac{n^3}{2}T^*_N(n-2, r-1) + \mathrm{poly}(n, N). \qquad (8)
\end{aligned}$$

Note that this inequality holds also for $k = 0, 1$. By combining (8) with Lemma 10, we have

$$\begin{aligned}
T_N(n, r, k) &\leq T^*_N(n-1, r) + (n-1)T^*_N(n-1, r-1) + \frac{n^3}{2}T^*_N(n-2, r-1) + \mathrm{poly}(n, N) \\
&\leq T^*_N(n-1, r) + n^3 T^*_N(n, r-1) + \mathrm{poly}(n, N),
\end{aligned}$$

where we use the monotonicity of T^*_N in the second inequality. Since this inequality holds for any $k \geq 0$, we have

$$T^*_N(n, r) \leq T^*_N(n-1, r) + n^3 T^*_N(n, r-1) + \mathrm{poly}(n, N) \qquad (9)$$

for any n and r. By using (9) repeatedly (by changing n), we obtain

$$T_N^*(n,r) \le n^4 T_N^*(n,r-1) + \text{poly}(n,N). \tag{10}$$

Furthremore, by using (10) repeatedly (by changing r), we obtain $T_N^*(n,r) = (\text{poly}(n,N))^{O(r)}$. This shows that Algorithm Permanent(r,A) runs in polynomial time in n and N for fixed r. (End of the proof of Theorem 9) □

References

[1] R.E. Bellman, On a routing problem, *Quarterly of Applied Mathematics* (1958) **16**, pp. 87–90.

[2] A. Björklund and T. Husfeldt, Shortest two disjoint paths in polynomial time, *Proceedings of the 41st International Colloquium on Automata, Languages and Programming, Part I. LNCS 8572* (2014), pp. 211–222.

[3] E.W. Dijkstra, A note on two problems in connexion with graphs, *Numerische Mathematik* (1959) **1**, pp. 269–271.

[4] J.L Gross and T.W. Tucker, Generating all graph coverings by permutation voltage assignments, *Discrete Mathematics* (1977) **18**, pp. 273–283.

[5] J.L Gross and T.W. Tucker, *Topological Graph Theory*, Wiley Interscience, 1987.

[6] M. Grötschel and W.R. Pulleyblank, Weakly bipartite graphs and the max-cut problem, *Operations Research Letters* (1981) **1**, pp. 23–27.

[7] T. Huynh, *The Linkage Problem for Group-Labelled Graphs* PhD. Thesis, Department of Combinatorics and Optimization, University of Waterloo, Ontario, 2009.

[8] A.S. LaPaugh and C.H. Papadimitriou, The even-path problem for graphs and digraphs, *Networks* (1984) **14**, pp. 507–513.

[9] K. Mulmuley, U.V. Vazirani, and V.V. Vazirani, Matching is as easy as matrix inversion, *Combinatorica* (1987) **7**, pp. 105–113.

[10] A. Schrijver, *Combinatorial Optimization. Polyhedra and Efficiency*, Springer-Verlag, 2003.

[11] L.G. Valiant, The complexity of computing the permanent, *Theoretical Computer Science* (1979) **8**, pp. 189–201.

[12] T. Zaslavsky, Biased graph. I. bias, balance, and gains, *Journal of Combinatorial Theory, Series B* (1989) **47**, pp. 32–52.