Reconfiguration of Maximum-Weight b-Matchings in a Graph

Ito, Takehiro
Graduate School of Information Sciences, Tohoku University

Kakimura, Naonori
Department of Mathematics, Keio University

Kamiyama, Naoyuki
Institute of Mathematics for Industry, Kyushu University

Kobayashi, Yusuke
Faculty of Engineering, Information and Systems, University of Tsukuba

他

https://hdl.handle.net/2324/4362392

出版情報：Journal of Combinatorial Optimization. 37 (2), pp.454-464, 2018-04-27. Springer
バージョン：
権利関係：

Reconfiguration of Maximum-Weight *b*-Matchings in a Graph[⋆]

Takehiro Ito[1,⋆⋆], Naonori Kakimura[2,⋆⋆⋆], Naoyuki Kamiyama[3,†],
Yusuke Kobayashi[4,‡], and Yoshio Okamoto[5,6,§]

[1] Graduate School of Information Sciences, Tohoku University, Japan
`takehiro@ecei.tohoku.ac.jp`
[2] Department of Mathematics, Keio University, Japan
`kakimura@math.keio.ac.jp`
[3] Institute of Mathematics for Industry, Kyushu University, Japan
`kamiyama@imi.kyushu-u.ac.jp`
[4] Faculty of Engineering, Information and Systems, University of Tsukuba, Japan
`kobayashi@sk.tsukuba.ac.jp`
[5] Department of Communication Engineering and Informatics, University of Electro-Communications, Japan
`okamotoy@uec.ac.jp`
[6] RIKEN Center for Advanced Intelligence Project

Abstract. Consider a graph such that each vertex has a nonnegative integer capacity and each edge has a positive integer weight. Then, a b-matching in the graph is a multi-set of edges (represented by an integer vector on edges) such that the total number of edges incident to each vertex is at most the capacity of the vertex. In this paper, we study a reconfiguration variant for maximum-weight b-matchings: For two given maximum-weight b-matchings in a graph, we are asked to determine whether there exists a sequence of maximum-weight b-matchings in the graph between them, with subsequent b-matchings obtained by removing one edge and adding another. We show that this reconfiguration problem is solvable in polynomial time for instances with no integrality gap. Such instances include bipartite graphs with any capacity function on vertices, and 2-matchings in general graphs. Thus, our result implies that the reconfiguration problem for maximum-weight matchings can be solved in polynomial time for bipartite graphs.

keywords: combinatorial reconfiguration, graph algorithm, b-matching

1 Introduction

Recently, *reconfiguration problems* [11] have attracted much attention in the field of theoretical computer science. These problems arise when we wish to find a step-by-step transformation between two feasible solutions of a combinatorial problem such that all intermediate results are also feasible and each step conforms to a fixed reconfiguration rule (i.e., an adjacency relation defined on feasible solutions of the original combinatorial problem). For example, in the (cardinality) MATCHING RECONFIGURATION problem, feasible solutions are matchings in a graph having the same cardinality and one of the studied reconfiguration rules is to exchange an edge in the current matching with an edge which is not contained

---

[⋆] A preliminary version of this paper has been presented at COCOON 2017 [12].

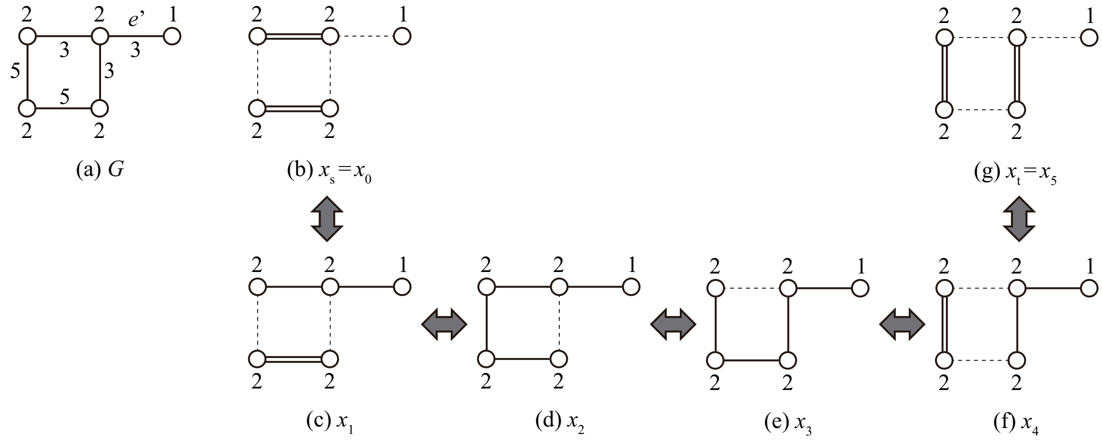
[⋆⋆] Supported by JST CREST Grant Number JPMJCR1402, and by JSPS KAKENHI Grant Number JP16K00004, Japan.

[⋆⋆⋆] Supported by JST ERATO Grant Number JPMJER1201, and by JSPS KAKENHI Grant Number JP17K00028, Japan.

[†] Supported by JST PRESTO Grant Number JPMJPR14E1, Japan.

[‡] Supported by JST ERATO Grant Number JPMJER1201, and by JSPS KAKENHI Grant Numbers JP16K16010 and JP16H03118, Japan.

[§] Supported by Kayamori Foundation of Informational Science Advancement, JST CREST Grant Number JP-MJCR1402, and JSPS KAKENHI Grant Numbers JP24106005, JP24700008, JP24220003 and JP15K00009, Japan.



(a) G (b) $x_s = x_0$ (c) x_1 (d) x_2 (e) x_3 (f) x_4 (g) $x_t = x_5$

Fig. 1. (a) Graph G with vertex-capacities and edge-weights, and (b)–(g) a sequence $\langle x_\mathrm{s} = x_0, x_1, \ldots, x_5 = x_\mathrm{t}\rangle$ of maximum-weight b-matchings in G, where each $x_i(e)$, $0 \le i \le 5$, is represented as the number of parallel edges between the endpoints of the edge e.

in the matching. This kind of reconfiguration problems has been studied extensively for several well-known combinatorial problems, including SATISFIABILITY [6, 17, 19], INDEPENDENT SET [4, 5, 15], VERTEX COVER [13, 20], CLIQUE [14], DOMINATING SET [7, 8], VERTEX-COLORING [1, 3, 9], and so on. (See also a survey [10].)

1.1 Our problem

In this paper, we generalize (cardinality) MATCHING RECONFIGURATION, and study a reconfiguration problem for MAXIMUM-WEIGHT b-MATCHING defined as follows.

For a graph G, we denote by $V(G)$ and $E(G)$ the vertex set and edge set of G, respectively. Let $b\colon V(G) \to \mathbb{Z}_{\ge 0}$ be a capacity function on vertices, where $\mathbb{Z}_{\ge 0}$ is the set of all nonnegative integers. Then, a vector $x \in \mathbb{Z}_{\ge 0}^{E(G)}$ is called a *b-matching* in G if $\sum_{e\in\delta(v)} x(e) \le b(v)$ holds for each vertex $v \in V(G)$, where $\delta(v)$ denotes the set of all edges incident to the vertex v. For example, Fig. 1(b)–(g) illustrate six b-matchings in the graph G of Fig. 1(a). Note that an ordinary matching in a graph G is a b-matching in G such that $b\colon V(G) \to \{1\}$. The *cardinality* of a b-matching x in G is defined as $\sum_{e\in E(G)} x(e)$. Let $w\colon E(G) \to \mathbb{Z}_+$ be a weight function on edges, where $\mathbb{Z}_+$ is the set of all positive integers. Then, the *weight* of a b-matching x in G is defined as $\sum_{e\in E(G)} w(e)x(e)$.

For two b-matchings x and x' in a graph G, we write $x \leftrightarrow x'$ if there exists a pair of edges e and f in G such that $x(e) - x'(e) = x'(f) - x(f) = 1$ and $x(g) = x(g)$ for all edges $g \in E(G) \setminus \{e, f\}$. Thus, both x and x' have the same cardinality. (See any two consecutive b-matchings in Fig. 1(b)–(g) as examples.) For two maximum-weight b-matchings x and x' in G, we write $x \overset{w}{\leftrightsquigarrow} x'$ if there exists a sequence $\langle x_0, x_1, \ldots, x_\ell\rangle$ of b-matchings in G such that

(i) $x_0 = x$ and $x_\ell = x'$;
(ii) all b-matchings $x_0, x_1, \ldots, x_\ell$ have the maximum weight in G; and
(iii) $x_{i-1} \leftrightarrow x_i$ holds for each $i \in \{1, 2, \ldots, \ell\}$.

Then, the MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION problem is defined as follows:

Input: A graph G, a capacity function $b\colon V(G) \to \mathbb{Z}_{\ge 0}$ on vertices, a weight function $w\colon E(G) \to \mathbb{Z}_+$ on edges, and two maximum-weight b-matchings x_s and x_t in G
Question: Determine whether $x_\mathrm{s} \overset{w}{\leftrightsquigarrow} x_\mathrm{t}$ or not.

We denote by a 5-tuple $(G, b, w, x_\mathrm{s}, x_\mathrm{t})$ an instance of MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION. Note that this is a decision problem and hence it does not ask for an actual sequence of maximum-weight b-matchings. For the particular instance of Fig. 1, it has a desired sequence $\langle x_\mathrm{s} = x_0, x_1, \ldots, x_5 = x_\mathrm{t} \rangle$ as illustrated in the figure, and hence the answer is yes.

1.2 Known and related results

Ito et al. [11, Proposition 2] studied (cardinality) MATCHING RECONFIGURATION, and gave a polynomial-time algorithm to solve the problem for any graph. Mühlenthaler [21] generalized MATCHING RECONFIGURATION to the reconfiguration problem for degree-constrained subgraphs in a graph G, where a degree-constrained subgraph is a subgraph of G such that the degree of each vertex satisfies both lower and upper bounds of the vertex. This generalized reconfiguration problem is also solvable in polynomial time for any graph [21].[7]

In the reconfiguration problem of Mühlenthaler [21], each edge can be chosen at most once in a degree-constrained subgraph. However, the algorithm of [21] can be easily extended so that it works correctly and runs in polynomial time even if multiplicities on edges are allowed. By setting the lower bound equal to zero and the upper bound equal to $b(v)$ for each vertex v in a graph G, b-matchings (and hence ordinary matchings) in G can be seen as degree-constrained subgraphs of G. Consider MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION when restricted to identical edge-weight. Then, each maximum-weight b-matching in a graph G is simply a maximum-cardinality b-matching in G. Thus, the result by Mühlenthaler [21] implies the following proposition.

Proposition 1 ([21]). MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION *is solvable in polynomial time when restricted to identical edge-weight.*

As far as we know, reconfiguration problems have been studied mostly for unweighted instances. Note that SHORTEST PATH RECONFIGURATION [2] and STEINER TREE RECONFIGURATION [18] are defined on unweighted graphs, and hence they are cardinality variants. MATROID RECONFIGURATION [11, Proposition 1] is the only example in the reconfiguration framework which admits a polynomial-time algorithm for weighted instances. However, matchings do not form matroid bases.

1.3 Our contribution

In this paper, we show that MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION is solvable in polynomial time for instances with no integrality gap. Such instances include bipartite graphs with any capacity function b on vertices, and general graphs G with the capacity function $b\colon V(G) \to \{2\}$. Thus, our result yields that the reconfiguration problem for maximum-weight (ordinary) matchings can be solved in polynomial time for bipartite graphs.

Our idea is to use the structure of maximum-weight b-matchings in a graph with no integrality gap. As an intuitive example, the edge e' in Fig. 1 would be "useless" if $w(e') \leq 2$ because edges in two given maximum-weight b-matchings have weights at least three; indeed, it becomes a no-instance if $w(e') \leq 2$. In Section 2, we formulate the problem of finding a maximum-weight b-matching in a graph as an integer program, and show that the complementary slackness condition gives a characterization of b-matchings that have the maximum weight (Lemma 1). Then, in Section 3, we will make use of Lemma 1, and reduce the problem of asking the existence of a desired sequence of maximum-*weight* b-matchings to the problem of asking that of maximum-*cardinality* b-matchings; recall that the cardinality variant is solvable in polynomial time (Proposition 1).

2 Maximum-Weight b-Matchings

In this section, we give a characterization of maximum-weight b-matchings which will play an important role in our algorithm in Section 3.

---

[7] Properly speaking, both Ito et al. [11] and Mühlenthaler [21] studied their reconfiguration problems under a more generalized reconfiguration rule, called the TAR (Token Addition and Removal) rule. Their results hold also under the reconfiguration rule of this paper, which is called the TJ (Token Jumping) rule.

Let G be a simple graph, and let $b\colon V(G) \to \mathbb{Z}_{\geq 0}$ and $w\colon E(G) \to \mathbb{Z}_+$ be capacity and weight functions, respectively. We can formulate the problem of finding a maximum-weight b-matching in G as the following integer program **IP**:

$$\begin{aligned} \text{max.} \quad & \sum_{e \in E(G)} w(e)x(e) & \\ \text{s.t.} \quad & \sum_{e \in \delta(v)} x(e) \leq b(v) & (\forall v \in V(G)) \\ & x(e) \in \mathbb{Z}_{\geq 0} & (\forall e \in E(G)). \end{aligned}$$

We denote by a triple (G, b, w) an input to **IP**. Let **LP** be the following linear programming relaxation of **IP**:

$$\begin{aligned} \text{max.} \quad & \sum_{e \in E(G)} w(e)x(e) & \\ \text{s.t.} \quad & \sum_{e \in \delta(v)} x(e) \leq b(v) & (\forall v \in V(G)) \\ & x(e) \geq 0 & (\forall e \in E(G)). \end{aligned}$$

The dual program **DP** of **LP** can be described as follows:

$$\begin{aligned} \text{min.} \quad & \sum_{v \in V(G)} b(v)y(v) & \\ \text{s.t.} \quad & y(u) + y(v) \geq w(e) & (\forall e = \{u, v\} \in E(G)) \\ & y(v) \geq 0 & (\forall v \in V(G)). \end{aligned}$$

The complementary slackness condition (see, e.g., [16, Corollary 3.23]) implies the following theorem.

Theorem 1. *Suppose that x and y are feasible solutions of* **LP** *and* **DP**, *respectively. Then, the following two statements* (1) *and* (2) *are equivalent.*

(1) *x and y are optimal solutions of* **LP** *and* **DP**, *respectively.*

(2) *x and y satisfy the following* (i) *and* (ii):

(i) *$y(u) + y(v) = w(e)$ for every edge $e = \{u, v\} \in E(G)$ with $x(e) > 0$; and*

(ii) $\displaystyle\sum_{e \in \delta(v)} x(e) = b(v)$ *for every vertex $v \in V(G)$ with $y(v) > 0$.*

For each feasible solution y of **DP**, let $V_y = \{v \in V(G) \mid y(v) > 0\}$ and $E_y = \{e = \{u, v\} \in E(G) \mid y(u) + y(v) = w(e)\}$. Then, Theorem 1 implies the following corollary.

Corollary 1. *Assume that the optimal value of* **IP** *for (G, b, w) is equal to that of* **LP**. *Let y be an optimal solution of* **DP**. *Then, a b-matching $x \in \mathbb{Z}_{\geq 0}^{E(G)}$ in G has the maximum weight if and only if $\{e \in E(G) \mid x(e) > 0\} \subseteq E_y$ and $\sum_{e \in \delta(v)} x(e) = b(v)$ for every vertex $v \in V_y$.*

Proof. We first prove the only-if direction. Suppose that x is a maximum-weight b-matching in G. Then, because the optimal value of **LP** is assumed to be equal to that of **IP**, x is an optimal solution of **LP**. Since y is an optimal solution of **DP**, x and y satisfy Theorem 1(2). Therefore, $\{e \in E(G) \mid x(e) > 0\} \subseteq E_y$ holds. For every vertex $v \in V_y$, we have $y(v) > 0$ and hence Theorem 1(2)-(ii) yields $\sum_{e \in \delta(v)} x(e) = b(v)$.

We then prove the if direction. Suppose that a b-matching x in G satisfies $\{e \in E(G) \mid x(e) > 0\} \subseteq E_y$ and $\sum_{e \in \delta(v)} x(e) = b(v)$ for every vertex $v \in V_y$. Since each edge $e = \{u, v\} \in E(G)$ with $x(e) > 0$ is contained in E_y, we have $y(u) + y(v) = w(e)$. Therefore, Theorem 1(2)-(i) holds. We then claim that x and y satisfy Theorem 1(2)-(ii). Consider any vertex $v \in V(G)$ such that $y(v) > 0$. Then, we have $v \in V_y$, and hence $\sum_{e \in \delta(v)} x(e) = b(v)$ holds; thus, Theorem 1(2)-(ii) holds. In this way, x and y satisfy Theorem 1(2). Then, Theorem 1(1) yields that x is an optimal solution of **LP**. Since the optimal value of **IP** is assumed to be equal to that of **LP**, x is a maximum-weight b-matching in G. □

We now rephrase Corollary 1 so that it can be easily applied to our algorithm in the next section. For a graph G and its edge subset $E' \subseteq E(G)$, we denote by $G[E']$ the subgraph of G induced by E', that is, the vertex set of $G[E']$ is $\{u, v \in V(G) \mid \{u, v\} \in E'\}$ and the edge set of $G[E']$ is E'. For a vertex subset $C \subseteq V(G)$, we say that a b-matching $x \in \mathbb{Z}_{\geq 0}^{E(G)}$ in G is *C-saturated* if $\sum_{e \in \delta(v)} x(e) = b(v)$ holds for every vertex $v \in C$. Then, Corollary 1 can be rephrased as the following lemma; recall that a *vertex cover* of a graph G is a vertex subset of G which contains at least one of the endpoints of every edge in G.

Lemma 1. *Assume that the optimal value of* **IP** *for* (G, b, w) *is equal to that of* **LP**. *Then, there exist a vertex subset* $C \subseteq V(G)$ *and an edge subset* $E' \subseteq E(G)$ *such that*

(*a*) *C is a vertex cover of $G[E']$; and*

(*b*) *a b-matching $x \in \mathbb{Z}_{\geq 0}^{E(G)}$ in G has the maximum weight if and only if $\{e \in E(G) \mid x(e) > 0\} \subseteq E'$ and x is C-saturated.*

Furthermore, such a pair of C and E' can be found in polynomial time.

Proof. Because an optimal solution y of **DP** can be computed in polynomial time, we can obtain V_y and E_y in polynomial time. Let $C = V_y$ and $E' = E_y$. Then, Condition (b) follows immediately from Corollary 1.

We now verify Condition (a). Consider any edge $e = \{u, v\} \in E' = E_y$. Then, $y(u) + y(v) = w(e)$ holds. Since $w(e) > 0$, we have $y(u) > 0$ or $y(v) > 0$. Therefore, $u \in V_y$ or $v \in V_y$, that is, at least one of the endpoints of e is contained in V_y. In this way, $C = V_y$ forms a vertex cover of $G[E']$. □

Note that we use the assumption of (nonzero) positive edge-weights only in the proof of Lemma 1(a). Theorem 1 and Corollary 1 hold even for nonnegative edge-weights, that is, $w(e) \geq 0$ for all edges $e \in E(G)$.

3 Algorithm

In this section, we give the main result of the paper as the following theorem.

Theorem 2. Maximum-weight b-matching reconfiguration *can be solved in polynomial time for any instance* $(G, b, w, x_{\mathrm{s}}, x_{\mathrm{t}})$ *such that the optimal value of* **IP** *for* (G, b, w) *is equal to that of* **LP**.

It is known that the optimal value of **IP** for (G, b, w) is equal to that of **LP** if G is bipartite [22, Theorem 21.2], or $b\colon V(G) \to \{2\}$ [22, Corollary 30.2a]. Then, we have the following corollary.

Corollary 2. Maximum-weight b-matching reconfiguration *can be solved in polynomial time for bipartite graphs, or* $b\colon V(G) \to \{2\}$.

In the remainder of this section, we prove Theorem 2 by giving such an algorithm. As we mentioned in Introduction, we will reduce the problem of asking the existence of a desired sequence of maximum-*weight* b-matchings to the problem of asking that of maximum-*cardinality* b-matchings, by using the characterization of maximum-weight b-matchings (Lemma 1).

Let $(G, b, w, x_{\mathrm{s}}, x_{\mathrm{t}})$ be an instance of maximum-weight b-matching reconfiguration such that the optimal value of **IP** for (G, b, w) is equal to that of **LP**. Let $C \subseteq V(G)$ and $E' \subseteq E(G)$ be the pair obtained by Lemma 1. By Lemma 1(b), any maximum-weight b-matching $x \in \mathbb{Z}_{\geq 0}^{E(G)}$ in G satisfies $x(e) = 0$ for all edges $e \in E(G) \setminus E'$. Therefore, it suffices to consider only C-saturated b-matchings in the induced subgraph $G[E']$. Note that both x_{s} and x_{t} are C-saturated b-matchings in $G[E']$, because they are maximum-weight b-matchings in G.

For two C-saturated b-matchings $x, x' \in \mathbb{Z}_{\geq 0}^{E'}$ in $G[E']$, we write $x \overset{C,E'}{\leftrightsquigarrow} x'$ if there exists a sequence $\langle x_0, x_1, \ldots, x_\ell \rangle$ of b-matchings in $G[E']$ such that

(i) $x_0 = x$ and $x_\ell = x'$;

(ii) all b-matchings $x_0, x_1, \ldots, x_\ell$ are C-saturated; and

(iii) $x_{i-1} \leftrightarrow x_i$ holds for each $i \in \{1, 2, \ldots, \ell\}$.

By Lemma 1(b) we then have the following proposition.

Algorithm 1 Polynomial-time algorithm for MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION

```
Input: An instance $(G, b, w, x_{\mathrm{s}}, x_{\mathrm{t}})$ of MAXIMUM-WEIGHT $b$-MATCHING RECONFIGURATION such that the optimal value of IP for $(G, b, w)$ is equal to that of LP
Output: yes/no
  Step 1. Obtain a vertex subset $C \subseteq V(G)$ and an edge subset $E' \subseteq E(G)$ satisfying Conditions (a) and (b) of Lemma 1. Let $E_C$ be the set of all edges $e$ in $G[E']$ such that both endpoints of $e$ belong to $C$.
  Step 2. If there exists an edge $e \in E_C$ such that $x_{\mathrm{s}}(e) \neq x_{\mathrm{t}}(e)$, then return no.
  Step 3. Let $b'(v) = b(v) - \sum_{e \in \delta(v) \cap E_C} x_{\mathrm{s}}(e)$ for each vertex $v$ in $G[E']$. Delete all edges in $E_C$ from $G[E']$; let $G'$ be the resulting graph. Let $x'_{\mathrm{s}}$ and $x'_{\mathrm{t}}$ be two $b'$-matchings in $G'$ such that $x'_{\mathrm{s}}(e) = x_{\mathrm{s}}(e)$ and $x'_{\mathrm{t}}(e) = x_{\mathrm{t}}(e)$, respectively, for all edges $e$ in $G'$.
  Step 4. Apply Proposition 1 to the instance $(G', b', w_{\mathsf{id}}, x'_{\mathrm{s}}, x'_{\mathrm{t}})$ where $w_{\mathsf{id}} \colon E(G') \to \{1\}$, and return its answer.
```

Proposition 2. *$x_{\mathrm{s}} \overset{C,E'}{\leftrightsquigarrow} x_{\mathrm{t}}$ if and only if $x_{\mathrm{s}} \overset{w}{\leftrightsquigarrow} x_{\mathrm{t}}$.*

Therefore, for proving Theorem 2, it suffices to determine whether $x_{\mathrm{s}} \overset{C,E'}{\leftrightsquigarrow} x_{\mathrm{t}}$ or not, in polynomial time. Our algorithm can be outlined as Algorithm 1. By Lemma 1 and Proposition 1, Algorithm 1 runs in polynomial time. Thus, we will prove its correctness in the remainder of this section.

We first show the correctness of Step 2 of Algorithm 1. To show this, we note that no edge in E_C can be touched by any transformation of C-saturated b-matchings, as in the following lemma.

Lemma 2. *Let x and x' be C-saturated b-matchings in $G[E']$ such that $x \overset{C,E'}{\leftrightsquigarrow} x'$. Then, $x(e) = x'(e)$ holds for each edge $e \in E_C$.*

Proof. Suppose for a contradiction that there exists a C-saturated b-matching x' in $G[E']$ such that $x \overset{C,E'}{\leftrightsquigarrow} x'$ and $x'(e^*) \neq x(e^*)$ for some edge $e^* \in E_C$. Since $x \overset{C,E'}{\leftrightsquigarrow} x'$ holds, there exists a sequence $\langle x = x_0, x_1, \ldots, x_\ell = x' \rangle$ of C-saturated b-matchings in $G[E']$. Let $i \in \{1, 2, \ldots, \ell\}$ be an index such that $x_{i-1}(e^*) \neq x_i(e^*)$; such an index i exists because $x(e^*) \neq x'(e^*)$ holds. Since $|x_i(e^*) - x_{i-1}(e^*)| = 1$, by changing the roles of x and x' if necessary, we may assume that $x_i(e^*) = x_{i-1}(e^*) - 1$. Then, since $x_{i-1} \leftrightarrow x_i$, there is exactly one edge e^+ in $E' \setminus \{e^*\}$ such that $x_i(e^+) = x_{i-1}(e^+) + 1$. Since $e^+ \neq e^*$, the edge e^+ is not incident to at least one endpoint of e^*, say v^*. This implies that

$$\sum_{e \in \delta(v^*)} x_i(e) = \sum_{e \in \delta(v^*)} x_{i-1}(e) - 1 = b(v^*) - 1,$$

which contradicts that x_i is C-saturated, because $e^* \in E_C$ and hence v^* is in C. ☐

We then show that the graph G' obtained by Step 3 of Algorithm 1 satisfies the following lemma.

Lemma 3. *The graph G' obtained by Algorithm 1 is a bipartite graph with bipartition C and $V(G') \setminus C$.*

Proof. Since all edges in E_C have been deleted, there is no edge joining two vertices in C. On the other hand, by Lemma 1(a) at least one of the endpoints of each edge in $G[E']$ is contained in C. Thus, there is no edge joining two vertices in $V(G') \setminus C = V(G[E']) \setminus C$. Therefore, G' is a bipartite graph with bipartition C and $V(G') \setminus C$. ☐

Finally, the correctness of Step 4 of Algorithm 1 can be verified by combining the following lemma with Lemma 2.

Lemma 4. *A b'-matching x in G' is C-saturated if and only if x has the maximum cardinality in G'.*

Proof. We first prove the only-if direction. Suppose that a b'-matching x in G' is C-saturated. Then, $\sum_{e \in E(G')} x(e) \geq \sum_{v \in C} b'(v)$ holds. Since G' is a bipartite graph whose one side of the bipartition is

C, any b'-matching in G' is of cardinality at most $\sum_{v\in C} b'(v)$. Therefore, x is a maximum-cardinality b'-matching in G'.

We then prove the if direction. Suppose that a b'-matching x in G' has the maximum cardinality in G'. It suffices to prove

$$\sum_{e\in E(G')} x(e) \geq \sum_{e\in E(G')} x'_{\mathrm{s}}(e) \geq \sum_{v\in C} b'(v);$$

then, all vertices in C must be saturated by x, because G' is a bipartite graph with bipartition C and $V(G') \setminus C$. The first inequality holds because x is a maximum-cardinality b'-matching in G'. We thus prove the second inequality, as follows. Since x_{s} is a maximum-weight b-matching in G, by Lemma 1(b) it satisfies $\{e \in E(G) \mid x_{\mathrm{s}}(e) > 0\} \subseteq E'$ and is C-saturated. Therefore, we have

$$\sum_{e\in E(G')} x'_{\mathrm{s}}(e) = \sum_{e\in E'\setminus E_C} x_{\mathrm{s}}(e) = \sum_{e\in E(G)\setminus E_C} x_{\mathrm{s}}(e) \geq \sum_{v\in C} b(v) - \sum_{e\in E_C} x_{\mathrm{s}}(e) \geq \sum_{v\in C} b'(v),$$

as claimed. □

In this way, Algorithm 1 correctly solves MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION in polynomial time. This completes our proof of Theorem 2.

4 Concluding Remarks

In this paper, we have shown that MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION is solvable in polynomial time for instances with no integrality gap. We emphasize again that such instances include b-matchings (and hence ordinary matchings) in bipartite graphs and 2-matchings in general graphs.

As we have mentioned in Section 2, we use the assumption of (nonzero) positive edge-weights only in the proof of Lemma 1(a). Indeed, Theorem 1 and Corollary 1 hold even for nonnegative edge-weights, that is, $w(e) \geq 0$ for all edges $e \in E(G)$. The complexity status of MAXIMUM-WEIGHT b-MATCHING RECONFIGURATION remains open for nonnegative edges-weights.

As another (more general) open question, we recall that both Ito et al. [11] and Mühlenthaler [21] studied their reconfiguration problems under a more generalized reconfiguration rule, called the TAR rule. In the WEIGHTED b-MATCHING RECONFIGURATION problem under the TAR rule, we are given two b-matchings (which do not necessarily have the maximum weight) together with an integer threshold $k \in \mathbb{Z}_{\geq 0}$, and asked the existence of a sequence of b-matchings between them, obtained by either adding or deleting one edge at a time, with keeping weights at least k. It remains open to clarify the complexity status for WEIGHTED b-MATCHING RECONFIGURATION under the TAR rule; this open question was originally posed by Ito et al. [11] for WEIGHTED MATCHING RECONFIGURATION.

References

1. Bonamy, M., Johnson, M., Lignos, I., Patel, V., Paulusma, D.: Reconfiguration graphs for vertex colourings of chordal and chordal bipartite graphs. J. Combinatorial Optimization 27, pp. 132–143 (2014)
2. Bonsma, P.: Rerouting shortest paths in planar graphs. Discrete Applied Mathematics 231, pp. 95–112 (2017)
3. Bonsma, P., Cereceda, L.: Finding paths between graph colourings: PSPACE-completeness and superpolynomial distances. Theoretical Computer Science 410, pp. 5215–5226 (2009)
4. Bonsma, P., Kamiński, M., Wrochna, M.: Reconfiguring independent sets in claw-free graphs. Proc. of the 14th Scandinavian Symposium and Workshops on Algorithm Theory (SWAT 2014), Lecture Notes in Computer Science 8503, pp. 86–97 (2014)
5. Demaine, E.D., Demaine, M.L., Fox-Epstein, E., Hoang, D.A., Ito, T., Ono, H., Otachi, Y., Uehara, R., Yamada, T.: Linear-time algorithm for sliding tokens on trees. Theoretical Computer Science 600, pp. 132–142 (2015)
6. Gopalan, P., Kolaitis, P.G., Maneva, E.N., Papadimitriou, C.H.: The connectivity of Boolean satisfiability: computational and structural dichotomies. SIAM J. Computing 38, pp. 2330–2355 (2009)

7. Haas, R., Seyffarth, K.: The k-dominating graph. Graphs and Combinatorics 30, pp. 609–617 (2014)
8. Haddadan, A., Ito, T., Mouawad, A.E., Nishimura, N., Ono, H., Suzuki, A., Tebbal, Y.: The complexity of dominating set reconfiguration. Theoretical Computer Science 651, pp. 37–49 (2016)
9. Hatanaka, T., Ito, T., Zhou, X.: The list coloring reconfiguration problem for bounded pathwidth graphs. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E98-A, pp. 1168–1178 (2015)
10. van den Heuvel, J.: The complexity of change. Surveys in Combinatorics 2013, London Mathematical Society Lecture Notes Series 409 (2013)
11. Ito, T., Demaine, E.D., Harvey, N.J.A., Papadimitriou, C.H., Sideri, M., Uehara, R., Uno, Y.: On the complexity of reconfiguration problems. Theoretical Computer Science 412, pp. 1054–1065 (2011)
12. Ito, T., Kakimura, N., Kamiyama, N., Kobayashi, Y., Okamoto, Y.: Reconfiguration of maximum-weight b-matchings in a graph. Proc. of the 23rd Annual International Computing and Combinatorics Conference (COCOON 2017), Lecture Notes in Computer Science 10392, pp. 287–296 (2017)
13. Ito, T., Nooka, H., Zhou, X.: Reconfiguration of vertex covers in a graph. IEICE Transactions on Information and Systems E99-D, pp. 598–606 (2016)
14. Ito, T., Ono, H., and Otachi, Y.: Reconfiguration of cliques in a graph. Proc. of the 12th Annual Conference on Theory and Applications of Models of Computation (TAMC 2015), Lecture Notes in Computer Science 9076, pp. 212–223 (2015)
15. Kamiński, M., Medvedev, P., Milanič, M.: Complexity of independent set reconfigurability problems. Theoretical Computer Science 439, pp. 9–15 (2012)
16. Korte, B., Vygen, J.: Combinatorial Optimization: Theory and Algorithms. 5th edn. Springer (2012)
17. Makino, K., Tamaki, S., Yamamoto, M.: An exact algorithm for the Boolean connectivity problem for k-CNF. Theoretical Computer Science 412, pp. 4613–4618 (2011)
18. Mizuta, H., Ito, T., Zhou, X.: Reconfiguration of Steiner trees in an unweighted graph. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences E100-A(7), pp. 1532–1540 (2017)
19. Mouawad, A.E., Nishimura, N., Pathak, V., Raman, V.: Shortest reconfiguration paths in the solution space of Boolean formulas. SIAM J. Discrete Mathematics 31, pp. 2185–2200 (2017)
20. Mouawad, A.E., Nishimura, N., Raman, V.: Vertex cover reconfiguration and beyond. Proc. of the 25th International Symposium on Algorithms and Computation (ISAAC 2014), Lecture Notes in Computer Science 8889, pp. 452–463 (2014)
21. Mühlenthaler, M.: Degree-constrained subgraph reconfiguration is in P. Proc. of the 40th International Symposium on Mathematical Foundations of Computer Science (MFCS 2015), Lecture Notes in Computer Science 9235, pp. 505–516 (2015)
22. Schrijver A.: Combinatorial Optimization: Polyhedra and Efficiency. Springer-Verlag, Berlin (2003)