An Algorithmic Proof for the Induction of M-convex Functions through Networks

Akiyoshi SHIOURA*

July, 1996

Abstract

Quite recently, Murota introduced an M-convex function as a quantitative generalization of the set of integral points in a base polyhedron, as well as an extension of valuated matroid over base polyhedron. Just as a base polyhedron can be transformed through a network, an M-convex function can be induced through a network. This paper gives an algorithmic proof for the induction of an M-convex function. The proof is based on the correctness of a simple algorithm, which efficiently finds an exchangeable element with a novel operation called crossover. We also analyze a behavior of induced functions when they take the value $-\infty$.

Keywords: matroid, base polyhedron, submodular system, convex function.

1 Introduction

In 1990, Dress and Wenzel introduced a valuated matroid as a quantitative generalization of a matroid [1, 2]. A valuated matroid is a pair of a matroid $(E, \mathcal{B})$ and a function $\omega : \mathcal{B} \to \mathbf{R}$ which enjoys the following exchange property:

(VM) For any $B, \widetilde{B} \in \mathcal{B}$, and $u \in B - \widetilde{B}$, there exists $v \in \widetilde{B} - B$ such that $B - u + v \in \mathcal{B}$, $\widetilde{B} + u - v \in \mathcal{B}$, and

$$\omega(B) + \omega(\widetilde{B}) \leq \omega(B - u + v) + \omega(\widetilde{B} + u - v).$$

Such a function ω is called a valuation of $(E, \mathcal{B})$.

Quite recently, Murota introduced the concept of M-convex function [4, 5, 6], which is a quantitative generalization of integral points in a base polyhedron as well as an extension of (the negative of) matroid valuation over base polyhedron. It is known that the set of integral points in a base polyhedron $B \subseteq \mathbf{Z}^E$ has the following simultaneous exchange property:

(B-EXC) For any $x, \widetilde{x} \in B$ and $u \in E$ with $x(u) > \widetilde{x}(u)$, there exists $v \in E$ with $x(v) < \widetilde{x}(v)$ such that

$$x - \chi'_u + \chi'_v,\ \widetilde{x} + \chi'_u - \chi'_v \in B,$$

where $\chi'_u \in \{0, 1\}^E$ is the characteristic vector of $u \in E$, that is, $\chi'_u(w) = 1$ if $w = u$ and $\chi'_u(w) = 0$ otherwise. Following [4, 5, 6], we call $B \subseteq \mathbf{Z}^E$ an integral base set if it

*Department of Mathematical and Computing Sciences, Tokyo Institute of Technology, 2-12-1 Oh-okayama, Meguro-ku, Tokyo 152, Japan, `shioura@is.titech.ac.jp`.

satisfies (B-EXC). Compared with this, an M-convex function $f : \mathbf{Z}^E \to \mathbf{R} \cup \{+\infty\}$ satisfies, by definition, the following quantitative generalization of the simultaneous exchange property:

(M-EXC) For any $x, \widetilde{x} \in \mathrm{dom}_{\mathbf{Z}} f$ and $u \in E$ with $x(u) > \widetilde{x}(u)$, there exists $v \in E$ with $x(v) < \widetilde{x}(v)$ such that

$$f(x) + f(\widetilde{x}) \geq f(x - \chi'_u + \chi'_v) + f(\widetilde{x} + \chi'_u - \chi'_v),$$

where $\mathrm{dom}_{\mathbf{Z}} f = \{x \in \mathbf{Z}^E \mid f(x) < +\infty\}$. The property (M-EXC) implies that the effective domain $\mathrm{dom}_{\mathbf{Z}} f$ of f is an integral base set.

In the theory of matroid and polymatroid, there have been considered several operations such as reduction, contraction, truncation, union, and network induction (see [3] as a relevant reference). These operations apply to matroids and base polyhedra. Above all, the induction by networks is one of the most powerful operations for matroids and base polyhedra and includes other operations such as truncation, union, etc. as special cases. Recent works by Murota [4, 6] revealed that those operations mentioned above also apply to M-convex functions. In particular, an M-convex function can be also transformed into another M-convex function through a network.

Let $G = (V, A; V^+, V^-)$ be a directed graph with two specified vertex sets $V^+, V^- \subseteq V$ such that $V^+ \cap V^- = \emptyset$. The vertex sets V^+, V^- are called the entrance set and the exit set, respectively. We denote an upper capacity function by $\overline{c} : A \to \mathbf{Z} \cup \{+\infty\}$, a lower capacity function by $\underline{c} : A \to \mathbf{Z} \cup \{-\infty\}$, and a weight function by $\gamma : A \to \mathbf{R}$. Suppose we are given an M-convex function $f^+ : \mathbf{Z}^{V^+} \to \mathbf{R}$. A flow is a function $\varphi : A \to \mathbf{Z}$, and its boundary $\partial\varphi : V \to \mathbf{Z}$ is defined as

$$\partial\varphi(v) = \sum\{\varphi(a) \mid a \in \delta^+ v\} - \sum\{\varphi(a) \mid a \in \delta^- v\} \qquad (v \in V),$$

where $\delta^+ v$ $(\delta^- v)$ denotes the set of arcs leaving (entering) v. For any function $z : V \to \mathbf{R}$, we denote the restriction of z to V^+ and to V^- by $(z)^+$ and $(z)^-$, respectively. A flow φ is called feasible if it satisfies the following conditions:

$$\begin{aligned}
&\underline{c}(a) \leq \varphi(a) \leq \overline{c}(a) \qquad (a \in A), \\
&\partial\varphi(v) = 0 \qquad (v \in V - (V^+ \cup V^-)), \\
&(\partial\varphi)^+ \in \mathrm{dom}_{\mathbf{Z}} f^+.
\end{aligned}$$

We define a function $f : \mathbf{Z}^{V^-} \to \mathbf{R} \cup \{\pm\infty\}$ as follows:

$$f(x) = \inf\{\langle \gamma, \varphi \rangle_A + f^+((\partial\varphi)^+) \mid \varphi : \text{feasible flow}, (\partial\varphi)^- = x\},$$

where

$$\langle \gamma, \varphi \rangle_A = \sum\{\gamma(a)\varphi(a) \mid a \in A\}.$$

Note that $f(x) = +\infty$, by convention, if there is no feasible flow φ with $(\partial\varphi)^- = x$. We have the following theorem, which is proved by Murota [4, Theorem 7.2], [6, Theorem 4.14] based on a characterization of M-convexity by minimizers and on an optimality condition for the generalized submodular flow problem [5].

Theorem 1.1 *The function* f *is M-convex provided* $f : \mathbf{Z}^{V^-} \to \mathbf{R} \cup \{+\infty\}$, *i.e., it does not take the value* $-\infty$. ∎

The objective of this paper is to provide an alternative simpler proof of this theorem, and also to analyze a behavior of induced functions when they take the value $-\infty$. Our proof is fairly straightforward and algorithmic, by establishing directly the condition (M-EXC) for the induced function f. The essence of the proof lies in the correctness of a simple algorithm, which for any $x, \widetilde{x} \in \mathrm{dom}_{\mathbf{Z}} f$ and $u \in V^-$ with $x(u) > \widetilde{x}(u)$, finds a vertex $v \in V^-$ with $x(v) < \widetilde{x}(v)$ such that

$$f(x - (\chi_u)^- + (\chi_v)^-) + f(\widetilde{x} + (\chi_u)^- - (\chi_v)^-) \le f(x) + f(\widetilde{x}). \tag{1}$$

Here $\chi_u \in \{0,1\}^V$ is the characteristic vector of $u \in V$. More specifically, for given feasible flows φ and $\widetilde{\varphi}$ with $(\partial\varphi)^- = x$, $(\partial\widetilde{\varphi})^- = \widetilde{x}$ and a vertex $u \in V^-$ with $x(u) > \widetilde{x}(u)$, our algorithm finds feasible flows $\varphi', \widetilde{\varphi}'$ and a vertex $v \in V^-$ with $x(v) < \widetilde{x}(v)$ satisfying $(\partial\varphi')^- = x - (\chi_u)^- + (\chi_v)^-$, $(\partial\widetilde{\varphi}')^- = \widetilde{x} + (\chi_u)^- - (\chi_v)^-$, and

$$\begin{aligned} &\langle \gamma, \varphi' \rangle_A + f^+((\partial\varphi')^+) + \langle \gamma, \widetilde{\varphi}' \rangle_A + f^+((\partial\widetilde{\varphi}')^+) \\ &\qquad \le \langle \gamma, \varphi \rangle_A + f^+((\partial\varphi)^+) + \langle \gamma, \widetilde{\varphi} \rangle_A + f^+((\partial\widetilde{\varphi})^+). \end{aligned}$$

Our algorithm efficiently finds $v \in V^-$ with $x(v) < \widetilde{x}(v)$ which meets the condition (1). Such v can be simply obtained by computing the value $f(x - (\chi_u)^- + (\chi_w)^-) + f(\widetilde{x} + (\chi_u)^- - (\chi_w)^-)$ for each $w \in V^-$ with $x(w) < \widetilde{x}(w)$, and each value can be computed by solving a generalized submodular flow problem [5]. However, we have to solve generalized submodular flow problems at most $2|V^-|$ times in total, and it takes too much time. On the other hand, our algorithm finds $v \in V^-$ with $x(v) < \widetilde{x}(v)$ satisfying the inequality (1) with $\mathrm{O}(|V|^2 ||\varphi - \widetilde{\varphi}||_A)$ evaluations for f^+ and in $\mathrm{O}(|A|\, ||\varphi - \widetilde{\varphi}||_A)$ time, where φ, $\widetilde{\varphi}$ are given feasible flows which realize x, $\widetilde{x}$ respectively, and

$$||\varphi - \widetilde{\varphi}||_A \equiv \sum \{ |\varphi(a) - \widetilde{\varphi}(a)| \mid a \in A \}.$$

It searches for v by iteratively updating flows in a straightforward way. Furthermore, we use a novel operation called *crossover* to prevent the algorithm from falling into a loop.

We also discuss induced functions which take the value $-\infty$. For induced functions which do not take $-\infty$, a behavior is already given as Theorem 1.1, but for induced functions with the value $-\infty$ it is not known yet. The correctness of our algorithm also reveals a behavior of such functions.

This paper is organized as follows: Section 2 provides the proof for the network induction based on the correctness of our algorithm. Section 3 describes our algorithm INDUCTION. Section 4 shows the correctness and the time complexity of the algorithm.

2 The Proof for the Induction

This section gives the alternative proof of Theorem 1.1 and an analysis of induced functions that take the value $-\infty$.

We first assert a slightly stronger claim than Theorem 1.1. Note that the induced function f may take the value $-\infty$ while an M-convex function does not take the value $-\infty$ by definition.

Theorem 2.1 *For any $x, \widetilde{x} \in \mathrm{dom}_{\mathbf{Z}} f$ and $u \in V^-$ with $x(u) > \widetilde{x}(u)$, there exists $v \in V^-$ with $x(v) < \widetilde{x}(v)$ such that $x - (\chi_u)^- + (\chi_v)^- \in \mathrm{dom}_{\mathbf{Z}} f$, $\widetilde{x} + (\chi_u)^- - (\chi_v)^- \in \mathrm{dom}_{\mathbf{Z}} f$, and*

$$f(x) + f(\widetilde{x}) \ge f(x - (\chi_u)^- + (\chi_v)^-) + f(\widetilde{x} + (\chi_u)^- - (\chi_v)^-).$$

∎

If f does not take the value $-\infty$, this claim is nothing but (M-EXC), the M-convexity of the induced function f. The proof of Theorem 2.1 is based on the correctness of the algorithm INDUCTION, which has the following property:

Theorem 2.2 *Suppose we are given feasible flows* $\varphi, \widetilde{\varphi}$ *with* $(\partial\varphi)^- = x$, $(\partial\widetilde{\varphi})^- = \widetilde{x}$ *and a vertex* $u \in V^-$ *with* $x(u) > \widetilde{x}(u)$. *Then, the algorithm* INDUCTION *finds feasible flows* $\varphi', \widetilde{\varphi}'$ *and a vertex* $v \in V^-$ *with* $x(v) < \widetilde{x}(v)$ *satisfying* $(\partial\varphi')^- = x - (\chi_u)^- + (\chi_v)^-$, $(\partial\widetilde{\varphi}')^- = \widetilde{x} + (\chi_u)^- - (\chi_v)^-$, *and*

$$\begin{aligned} &\langle\gamma, \varphi'\rangle_A + f^+((\partial\varphi')^+) + \langle\gamma, \widetilde{\varphi}'\rangle_A + f^+((\partial\widetilde{\varphi}')^+) \\ &\qquad \leq \quad \langle\gamma, \varphi\rangle_A + f^+((\partial\varphi)^+) + \langle\gamma, \widetilde{\varphi}\rangle_A + f^+((\partial\widetilde{\varphi})^+). \end{aligned}$$

■

A proof of this theorem is given later in Section 4.

Proof of Theorem 2.1: Given any $x, \widetilde{x} \in \mathrm{dom}_{\mathbf{Z}} f$ and $u \in V^-$ with $x(u) > \widetilde{x}(u)$, let $\{\varphi_k \mid k \geq 1\}, \{\widetilde{\varphi}_k \mid k \geq 1\}$ be the sequences of feasible flows such that $(\partial\varphi_k)^- = x$, $(\partial\widetilde{\varphi}_k)^- = \widetilde{x}$, and that

$$\lim_{k\to\infty} (\langle\gamma, \varphi_k\rangle_A + f^+((\partial\varphi_k)^+)) \;=\; f(x), \qquad \lim_{k\to\infty} (\langle\gamma, \widetilde{\varphi}_k\rangle_A + f^+((\partial\widetilde{\varphi}_k)^+)) \;=\; f(\widetilde{x}).$$

Such sequences exist necessarily from the definition of f.

By Theorem 2.2, for each pair of flows φ_k and $\widetilde{\varphi}_k$ $(k \geq 1)$ there exist feasible flows $\varphi'_k, \widetilde{\varphi}'_k$ and a vertex $v_k \in V^-$ with $x(v_k) < \widetilde{x}(v_k)$ such that $(\partial\varphi'_k)^- = x - (\chi_u)^- + (\chi_{v_k})^-$, $(\partial\widetilde{\varphi}'_k)^- = \widetilde{x} + (\chi_u)^- - (\chi_{v_k})^-$, and

$$\begin{aligned} &\langle\gamma, \varphi'_k\rangle_A + f^+((\partial\varphi'_k)^+) + \langle\gamma, \widetilde{\varphi}'_k\rangle_A + f^+((\partial\widetilde{\varphi}'_k)^+) \\ &\qquad \leq \langle\gamma, \varphi_k\rangle_A + f^+((\partial\varphi_k)^+) + \langle\gamma, \widetilde{\varphi}_k\rangle_A + f^+((\partial\widetilde{\varphi}_k)^+). \end{aligned}$$

Since the vertex set V^- is finite, there is at least one vertex v which appears infinite number of times in the sequence $\{v_k \mid k \geq 1\}$. Thus, we have an inequality

$$\begin{aligned} & f(x - (\chi_u)^- + (\chi_v)^-) + f(\widetilde{x} + (\chi_u)^- - (\chi_v)^-) \\ \leq\;& \inf\{\langle\gamma, \varphi'_k\rangle_A + f^+((\partial\varphi'_k)^+) \mid k \geq 1, v_k = v\} + \inf\{\langle\gamma, \widetilde{\varphi}'_k\rangle_A + f^+((\partial\widetilde{\varphi}'_k)^+) \mid k \geq 1, v_k = v\} \\ \leq\;& \inf\{\langle\gamma, \varphi_k\rangle_A + f^+((\partial\varphi_k)^+) \mid k \geq 1, v_k = v\} + \inf\{\langle\gamma, \widetilde{\varphi}_k\rangle_A + f^+((\partial\widetilde{\varphi}_k)^+) \mid k \geq 1, v_k = v\} \\ =\;& f(x) + f(\widetilde{x}). \end{aligned}$$

■

We next analyze the induced function f when taking $-\infty$. Theorem 2.1 reads as follows when $f(x) = -\infty$ or $f(\widetilde{x}) = -\infty$:

Lemma 2.3 *Let* $x, \widetilde{x} \in \mathrm{dom}_{\mathbf{Z}} f$ *with either* $f(x) = -\infty$ *or* $f(\widetilde{x}) = -\infty$. *Then for any* $u \in V^-$ *with* $x(u) > \widetilde{x}(u)$, *there exists* $v \in V^-$ *with* $x(v) < \widetilde{x}(v)$ *such that* $x - \chi_u + \chi_v \in \mathrm{dom}_{\mathbf{Z}} f$, $\widetilde{x} + \chi_u - \chi_v \in \mathrm{dom}_{\mathbf{Z}} f$, *and either* $f(x - \chi_u + \chi_v) = -\infty$ *or* $f(\widetilde{x} + \chi_u - \chi_v) = -\infty$. ■

This lemma reveals a behavior of induced functions taking $-\infty$, which says that $f(x) = -\infty$ for any point x in the 'interior' of $\mathrm{dom}_{\mathbf{Z}} f$. Since $\mathrm{dom}_{\mathbf{Z}} f$ is an integral base set by Theorem 2.1, there exists a submodular function $\rho : 2^{V^-} \to \mathbf{Z} \cup \{+\infty\}$ such that

$$\{x \in \mathbf{Z}^{V^-} \mid x(X) \leq \rho(X) \ (\forall X \subseteq V^-),\ x(V^-) = \rho(V^-)\} = \mathrm{dom}_{\mathbf{Z}} f.$$

Compare the next theorem with the related result for the convolution operation [6, Theorem 5.8(2)], which is a special case of the network induction [4].

Theorem 2.4 *Suppose the function* f *takes the value* $-\infty$. *If* $f(x_0) = -\infty$ *for some* $x_0 \in \mathrm{dom}_{\mathbf{Z}} f$, *then*

$$f(x) = -\infty \text{ for all } x \in I(x_0) \equiv \left\{ x' \in \mathbf{Z}^{V^-} \;\middle|\; \begin{array}{ll} x'(X) \leq \rho(X) & (\forall X \subseteq V^-, x_0(X) = \rho(X)), \\ x'(X) < \rho(X) & (\forall X \subseteq V^-, x_0(X) < \rho(X)), \\ x'(V^-) = \rho(V^-) & \end{array} \right\}.$$

Proof. We show by induction on the integer k that for any $x_0 \in \mathrm{dom}_{\mathbf{Z}} f$ with $f(x_0) = -\infty$ and $x \in I(x_0)$ with $||x - x_0||_{V^-} \equiv \sum\{|x(w) - x_0(w)| \mid w \in V^-\} = 2k$, $f(x)$ is equal to $-\infty$. We assume that $k > 0$, otherwise the claim is obvious.

First we prove when $k = 1$, where $x = x_0 - \chi_s + \chi_t$ for some $s, t \in V^-, s \neq t$. We assert that $x' = x - \chi_s + \chi_t \in \mathrm{dom}_{\mathbf{Z}} f$. For any $X \subseteq V^-$, if $x(X) < \rho(X)$ then it is clear that $x'(X) \leq x(X) + 1 \leq \rho(X)$. When $x(X) = \rho(X)$, we have $x_0(X) = \rho(X)$ since $x \in I(x_0)$. Therefore $s \in X$ if and only if $t \in X$, and we again have $x'(X) = \rho(X)$. This concludes that $x' \in \mathrm{dom}_{\mathbf{Z}} f$. Thus, Lemma 2.3 applies to x_0 and x'. We observe that s is a unique element in V^- with $x_0(s) > x'(s)$ and t is a unique element in V^- with $x_0(t) < x'(t)$. Since $x = x_0 - \chi_s + \chi_t = x' + \chi_s - \chi_t$, Lemma 2.3 for x_0 and x' implies $f(x) = -\infty$.

Next we assume that the claim holds for any j less than $k (\geq 2)$. Apply Lemma 2.3 to x and x_0, and obtain either $f(x_0 - \chi_u + \chi_v) = -\infty$ or $f(x + \chi_u - \chi_v) = -\infty$ for some $u \in V^-$ with $x_0(u) > x(u)$ and $v \in V^-$ with $x_0(v) < x(v)$. Moreover, if $x(X) = \rho(X) (= x_0(X))$ then $(x_0 - \chi_u + \chi_v)(X) = (x + \chi_u - \chi_v)(X) = \rho(X)$, which yields $x \in I(x_0 - \chi_u + \chi_v) \cap I(x + \chi_u - \chi_v)$. Since $||x - (x_0 - \chi_u + \chi_v)||_{V^-} = 2(k-1)$ and $||x - (x + \chi_u - \chi_v)||_{V^-} = 2$, we obtain $f(x) = -\infty$ from the assumption. Hence the claim holds for k. ∎

3 An Algorithm

This section presents the algorithm Induction. Given feasible flows $\varphi, \widetilde{\varphi}$ with $(\partial\varphi)^- = x$, $(\partial\widetilde{\varphi})^- = \widetilde{x}$ and a vertex $u \in V^-$ with $x(u) > \widetilde{x}(u)$, the algorithm Induction finds feasible flows $\varphi', \widetilde{\varphi}'$ and a vertex $v \in V^-$ with $x(v) < \widetilde{x}(v)$ satisfying $(\partial\varphi')^- = x - (\chi_u)^- + (\chi_v)^-$, $(\partial\widetilde{\varphi}')^- = \widetilde{x} + (\chi_u)^- - (\chi_v)^-$, and

$$\begin{aligned} &\langle \gamma, \varphi' \rangle_A + f^+((\partial\varphi')^+) + \langle \gamma, \widetilde{\varphi}' \rangle_A + f^+((\partial\widetilde{\varphi}')^+) \\ &\qquad \leq \ \langle \gamma, \varphi \rangle_A + f^+((\partial\varphi)^+) + \langle \gamma, \widetilde{\varphi} \rangle_A + f^+((\partial\widetilde{\varphi})^+). \end{aligned}$$

For convenience, we assume without loss of generality that $V^+ \cup V^- = V$. If this assumption fails, extend the function $f^+ : \mathbf{Z}^{V^+} \to \mathbf{R}$ over $\mathbf{Z}^{V - V^-}$ as follows:

$$\overline{f}^+(x^+, x^0) = \begin{cases} f^+(x^+) & (x^0 = \mathbf{0}), \\ +\infty & (x^0 \neq \mathbf{0}), \end{cases} \qquad (x^+ \in \mathbf{Z}^{V^+}, x^0 \in \mathbf{Z}^{V - (V^+ \cup V^-)}),$$

and reset V^+ to $V - V^-$. Then we can fulfill the assumption $V^+ \cup V^- = V$.

The algorithm maintains a set of four functions $\psi, \widetilde{\psi} \in \mathbf{Z}^A$, $b, \widetilde{b} \in \mathbf{Z}^V$ and a vertex $s \in V$

satisfying the following condition (FBS):

$$
\textbf{(FBS)}\left\{
\begin{array}{l}
\underline{c}(a) \le \psi(a) \le \overline{c}(a),\ \underline{c}(a) \le \widetilde{\psi}(a) \le \overline{c}(a) \quad (a \in A) \\
\qquad\qquad\qquad\qquad \text{(flows } \psi, \widetilde{\psi} \text{ fulfill capacity constraints)}, \\
(b)^+,\ (\widetilde{b})^+ \in \mathrm{dom}_{\mathbf{Z}} f^+ \\
\qquad\qquad\qquad\qquad ((b)^+, (\widetilde{b})^+ \text{ are integral bases in } \mathrm{dom}_{\mathbf{Z}} f^+), \\
(b)^- = x - (\chi_u)^-,\ (\widetilde{b})^- = \widetilde{x} + (\chi_u)^- \\
\qquad\qquad\qquad\qquad ((b)^-, (\widetilde{b})^- \text{ are almost equal to } x, \widetilde{x}), \\
b = \partial\psi - \chi_s,\ \widetilde{b} = \partial\widetilde{\psi} + \chi_s \\
\qquad\qquad\qquad\qquad (b, \widetilde{b} \text{ are almost equal to boundaries } \partial\psi, \partial\widetilde{\psi}), \\
F^+(\psi, b) + F^+(\widetilde{\psi}, \widetilde{b}) \le F^+(\varphi, \partial\varphi) + F^+(\widetilde{\varphi}, \partial\widetilde{\varphi}) \\
\qquad\qquad\qquad\qquad \text{(tuple } (\psi, \widetilde{\psi}, b, \widetilde{b}, s) \text{ is `cheaper' than flows } \varphi, \widetilde{\varphi}).
\end{array}
\right.
$$

where $F^+(\psi, b) = \langle \gamma, \psi \rangle_A + f^+((b)^+)$ for any $\psi \in \mathbf{Z}^A, b \in \mathbf{Z}^V$. We refer to such a tuple $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ as a *flow-base set*. The functions b, $\widetilde{b}$ are called *base functions* and the vertex s an *imbalance vertex*. Note that flows $\psi, \widetilde{\psi}$ and an imbalance vertex s of some flow-base set $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ uniquely determine base functions b and $\widetilde{b}$.

At the beginning of the algorithm, we have the flow-base set $(\varphi, \widetilde{\varphi}, b, \widetilde{b}, u)$ with base functions $b, \widetilde{b}$ such as

$$
\left.
\begin{array}{l}
(b)^+ = (\partial\varphi)^+,\ (\widetilde{b})^+ = (\partial\widetilde{\varphi})^+, \\
(b)^- = x - (\chi_u)^-,\ (\widetilde{b})^- = \widetilde{x} + (\chi_u)^-.
\end{array}
\right\} \tag{2}
$$

On the other hand, suppose that a flow-base set $(\psi, \widetilde{\psi}, b, \widetilde{b}, v)$ satisfies $v \in V^-$. Then, ψ and $\widetilde{\psi}$ have the following properties:

$$
\begin{array}{l}
(\partial\psi)^+,\ (\partial\widetilde{\psi})^+ \in \mathrm{dom}_{\mathbf{Z}} f^+, \\
(\partial\psi)^- = x - (\chi_u)^- + (\chi_v)^-,\ (\partial\widetilde{\psi})^- = \widetilde{x} + (\chi_u)^- - (\chi_v)^-, \\
F^+(\psi, \partial\psi) + F^+(\widetilde{\psi}, \partial\widetilde{\psi}) \le F^+(\varphi, \partial\varphi) + F^+(\widetilde{\varphi}, \partial\widetilde{\varphi}).
\end{array}
$$

Therefore, our aim is to obtain a flow-base set $(\psi, \widetilde{\psi}, b, \widetilde{b}, v)$ such that $v \in V^-$ and $x(v) < \widetilde{x}(v)$.

Our algorithm uses three operations in each iteration to update the current flow-base set.

Suppose that we have a flow-base set $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$, and that there exists an arc $a^* \in A$ such that

$$
a^* = (s, s'),\ \psi(a^*) > \widetilde{\psi}(a^*) \quad \text{or} \quad a^* = (s', s),\ \psi(a^*) < \widetilde{\psi}(a^*) \tag{3}
$$

for some vertex $s' \in V$. Then, we update the flows ψ, $\widetilde{\psi}$ to ψ', $\widetilde{\psi}'$ as follows:
If $a^* = (s, s')$, $\psi(a^*) > \widetilde{\psi}(a^*)$ then

$$
\psi'(a^*) = \psi(a^*) - 1,\ \widetilde{\psi}'(a^*) = \widetilde{\psi}(a^*) + 1,
$$

and if $a^* = (s', s)$, $\psi(a^*) < \widetilde{\psi}(a^*)$ then

$$
\psi'(a^*) = \psi(a^*) + 1,\ \widetilde{\psi}'(a^*) = \widetilde{\psi}(a^*) - 1.
$$

The flows on other arcs remain unchanged. We call this operation *flow exchange*. The flow exchange on a^* corresponds to pushing a unit flow on the arc a^* for both ψ and $\widetilde{\psi}$ simultaneously. The following lemma is clear from the definition.

Lemma 3.1 *Given a flow-base set* $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ *and an arc* a^* *with (3) for some* $s' \in V$, *let* $\psi', \widetilde{\psi}'$ *be the flows obtained by the flow exchange on* a^*. *Then, the next properties hold:*

$$\begin{array}{l} (\psi', \widetilde{\psi}', b, \widetilde{b}, s') \text{ is a flow-base set,} \\ \min\{\psi(a), \widetilde{\psi}(a)\} \le \psi'(a) \le \max\{\psi(a), \widetilde{\psi}(a)\} \quad (a \in A), \\ \min\{\psi(a), \widetilde{\psi}(a)\} \le \widetilde{\psi}'(a) \le \max\{\psi(a), \widetilde{\psi}(a)\} \quad (a \in A), \\ \psi'(a) + \widetilde{\psi}'(a) = \psi(a) + \widetilde{\psi}(a) \quad (a \in A). \end{array}$$

■

It may possibly happen that an arc becomes a candidate for the flow exchange in two successive iterations. For example, suppose $a^* = (s, s')$ satisfies $\psi(a^*) = \widetilde{\psi}(a^*) + 1$. If we perform the flow exchange on the arc a^*, then s' is the new imbalance vertex and $\psi'(a^*) = \widetilde{\psi}'(a^*) - 1$ for the new flows $\psi', \widetilde{\psi}'$. Although a^* becomes a candidate for the flow exchange again, if we perform the flow exchange on a^* then the resulting flows are same as ψ, $\widetilde{\psi}$. Therefore, our algorithm prohibits the flow exchange on the same arc consecutively not to return to the previous flow-base set.

We can always perform the flow exchange at the beginning of the algorithm, because $\partial\varphi(u) > \partial\widetilde{\varphi}(u)$ and hence

$$\{a \in \delta^+ u \mid \varphi(a) > \widetilde{\varphi}(a)\} \cup \{a \in \delta^- u \mid \varphi(a) < \widetilde{\varphi}(a)\} \ne \emptyset.$$

The second operation uses the simultaneous exchange property (M-EXC) of the M-convex function f^+. Let $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ be a current flow-base set with $s \in V^+$. If $b(s) < \widetilde{b}(s)$, there exists a vertex $s' \in V^+$ with $b(s') > \widetilde{b}(s')$ such that

$$f^+((b)^+ + (\chi_s)^+ - (\chi_{s'})^+) + f^+((\widetilde{b})^+ - (\chi_s)^+ + (\chi_{s'})^+) \le f^+((b)^+) + f^+((\widetilde{b})^+).$$

Let $b' = b + \chi_s - \chi_{s'}$, $\widetilde{b}' = \widetilde{b} - \chi_s + \chi_{s'}$. Then, the tuple $(\psi, \widetilde{\psi}, b', \widetilde{b}', s')$ obviously fulfills (FBS). We call this operation for base functions *base exchange.*

Lemma 3.2 *Let* $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ *be a flow-base set and* $b' = b + \chi_s - \chi_{s'}$, $\widetilde{b}' = \widetilde{b} - \chi_s + \chi_{s'}$ *be the base functions obtained by the base exchange for* b *and* b'. *Then,* $(\psi, \widetilde{\psi}, b', \widetilde{b}', s')$ *is also a flow-base set.* ■

The algorithm prohibits two successive base exchanges not to return to the previous flow-base set, which may happen when $b(s) = \widetilde{b}(s) + 1$ and $b(s') = \widetilde{b}(s') - 1$.

The last operation, called *crossover,* plays the most important role in the algorithm. An imbalance vertex changes as our algorithm iterates, but the same vertex can be an imbalance vertex many times. Therefore, the algorithm may fall into a loop. The operation crossover is performed when a vertex becomes an imbalance vertex twice, and makes our algorithm to stop in finite iterations.

Given two flow-base sets $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ and $(\psi', \widetilde{\psi}', b', \widetilde{b}', s)$ with a same imbalance vertex $s \in V$, the crossover of $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ and $(\psi', \widetilde{\psi}', b', \widetilde{b}', s)$ yields another flow-base set $(\psi'', \widetilde{\psi}'', b'', \widetilde{b}'', s)$, which is defined as

$$(\psi'', \widetilde{\psi}'', b'', \widetilde{b}'', s) = \begin{cases} (\psi, \widetilde{\psi}', b, \widetilde{b}', s) & \text{if } F^+(\psi, b) + F^+(\widetilde{\psi}', \widetilde{b}') \le F^+(\psi', b') + F^+(\widetilde{\psi}, \widetilde{b}), \\ (\psi', \widetilde{\psi}, b', \widetilde{b}, s) & \text{if } F^+(\psi, b) + F^+(\widetilde{\psi}', \widetilde{b}') > F^+(\psi', b') + F^+(\widetilde{\psi}, \widetilde{b}). \end{cases}$$

The following property is simple but crucial in the algorithm.

Lemma 3.3 *For given two flow-base sets* $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ *and* $(\psi', \widetilde{\psi}', b', \widetilde{b}', s)$ *with a same imbalance vertex* $s \in V$, *we suppose the following conditions:*

$$\psi \neq \psi' \text{ and } \widetilde{\psi} \neq \widetilde{\psi}', \tag{4}$$
$$\min\{\psi(a), \widetilde{\psi}(a)\} \le \psi'(a) \le \max\{\psi(a), \widetilde{\psi}(a)\} \quad (a \in A), \tag{5}$$
$$\min\{\psi(a), \widetilde{\psi}(a)\} \le \widetilde{\psi}'(a) \le \max\{\psi(a), \widetilde{\psi}(a)\} \quad (a \in A), \tag{6}$$
$$\psi(a) + \widetilde{\psi}(a) = \psi'(a) + \widetilde{\psi}'(a) \quad (a \in A). \tag{7}$$

Then, the crossover of $(\psi, \widetilde{\psi}, b, \widetilde{b}, s)$ *and* $(\psi', \widetilde{\psi}', b', \widetilde{b}', s)$ *yields a flow-base set* $(\psi'', \widetilde{\psi}'', b'', \widetilde{b}'', s)$ *with* $||\psi'' - \widetilde{\psi}''||_A < ||\psi - \widetilde{\psi}||_A$.

Proof. It is obvious that $(\psi'', \widetilde{\psi}'', b'', \widetilde{b}'', s)$ satisfies the condition (FBS) other than $F^+(\psi'', b'') + F^+(\widetilde{\psi}'', \widetilde{b}'') \le F^+(\varphi, \partial\varphi) + F^+(\widetilde{\varphi}, \partial\widetilde{\varphi})$, which is given by

$$\begin{aligned} F^+(\psi'', b'') + F^+(\widetilde{\psi}'', \widetilde{b}'') &\le (1/2)(F^+(\psi, b) + F^+(\widetilde{\psi}, \widetilde{b}) + F^+(\psi', b') + F^+(\widetilde{\psi}', \widetilde{b}')) \\ &\le F^+(\varphi, \partial\varphi) + F^+(\widetilde{\varphi}, \partial\widetilde{\varphi}). \end{aligned}$$

The conditions (5), (6) imply that

$$|\psi''(a) - \widetilde{\psi}''(a)| \le |\psi(a) - \widetilde{\psi}(a)| \quad (a \in A).$$

Furthermore, there exists an arc $a^* \in A$ with $\psi(a^*) \neq \psi'(a^*)$ from (4), and we also have $\widetilde{\psi}(a^*) \neq \widetilde{\psi}'(a^*)$ from (7). Combined with the conditions (5) and (6), we obtain

$$|\psi(a^*) - \widetilde{\psi}'(a^*)| = |\psi'(a^*) - \widetilde{\psi}(a^*)| < |\psi(a^*) - \widetilde{\psi}(a^*)|,$$

which provides the relation $||\psi'' - \widetilde{\psi}''||_A < ||\psi - \widetilde{\psi}||_A$. ∎

The algorithm is described below.

Algorithm Induction
Step 0: Put $i = 1$, $\psi_1 = \varphi$, $\widetilde{\psi}_1 = \widetilde{\varphi}, s_1 = u$. Set $b_1, \widetilde{b}_1$ as (2) and go to Step 1.
Step 1: Execute the procedure Update with the input $(\psi_i, \widetilde{\psi}_i, b_i, \widetilde{b}_i, s_i)$ and obtain the output $(\psi_{i+1}, \widetilde{\psi}_{i+1}, b_{i+1}, \widetilde{b}_{i+1}, s_{i+1})$. Go to Step 2.
Step 2: If $s_{i+1} \in V^-$ and $x(s_{i+1}) < \widetilde{x}(s_{i+1})$, then output s_{i+1} and stop. Otherwise, set $i = i + 1$ and go to Step 1.

[End of Algorithm Induction]

Procedure Update
Input: a flow-base set $(\zeta, \widetilde{\zeta}, d, \widetilde{d}, t)$ with $t \in V - \{w \in V^- \mid x(w) < \widetilde{x}(w)\}$.
Output: a flow-base set $(\zeta', \widetilde{\zeta}', d', \widetilde{d}', t')$ satisfying either (U-1), (U-2) or (U-3) when $t \neq u$, and either (U-1) or (U-3) when $t = u$, where

(U-1) $t' \in \{w \in V^- \mid x(w) < \widetilde{x}(w)\}$,
(U-2) $t' = u$, $||\zeta' - \widetilde{\zeta}'||_A \le ||\zeta - \widetilde{\zeta}||_A$,
(U-3) $t' \in V - \{w \in V^- \mid x(w) < \widetilde{x}(w)\}$, $||\zeta' - \widetilde{\zeta}'||_A < ||\zeta - \widetilde{\zeta}||_A$.

(Note that u is the vertex given as the input of the algorithm Induction.)
Step U0: Set $(\zeta_1, \widetilde{\zeta}_1, d_1, \widetilde{d}_1, u_1) = (\zeta, \widetilde{\zeta}, d, \widetilde{d}, t)$, $k = 1$, and go to Step U1.
Step U1: Execute the following.
Case 1: If $u_k \in \{w \in V^- \mid x(w) < \widetilde{x}(w)\}$, then output $(\zeta_k, \widetilde{\zeta}_k, d_k, \widetilde{d}_k, u_k)$ and stop.
Case 2: If $t \neq u$ and $u_k = u$, then output $(\zeta_k, \widetilde{\zeta}_k, d_k, \widetilde{d}_k, u_k)$ and stop.
Case 3 [crossover]: If $k > 1$ and $u_k = u_j$ for some j with $1 \le j \le k - 1$, then perform the

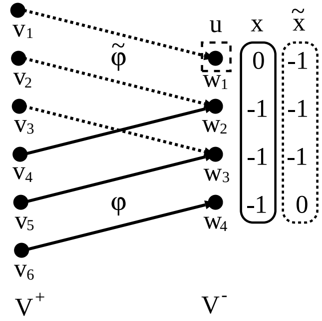

Figure 1: Initial flows

crossover of $(\zeta_j, \widetilde{\zeta}_j, d_j, \widetilde{d}_j, u_j)$ and $(\zeta_k, \widetilde{\zeta}_k, d_k, \widetilde{d}_k, u_k)$ to obtain $(\zeta'', \widetilde{\zeta}'', d'', \widetilde{d}'', u_k)$. Output $(\zeta'', \widetilde{\zeta}'', d'', \widetilde{d}'', u_k)$ and stop.
Case 4 [flow exchange]: If Cases 1 – 3 do not hold and

$$(\{a \in \delta^+ u_k \mid \zeta_k(a) > \widetilde{\zeta}_k(a)\} \cup \{a \in \delta^- u_k \mid \zeta_k(a) < \widetilde{\zeta}_k(a)\}) - \{e_{k-1} \text{ (if defined)}\} \neq \emptyset,$$

then select an arc e_k from this arc set and perform the flow exchange on e_k. Let $\zeta_{k+1}, \widetilde{\zeta}_{k+1}$ be the resulting flows, and u_{k+1} the vertex such that $e_k = (u_k, u_{k+1})$ or $e_k = (u_{k+1}, u_k)$. Put $d_{k+1} = d_k, \widetilde{d}_{k+1} = \widetilde{d}_k$, and $k = k + 1$. Repeat Step U1.
Case 5 [base exchange]: If Cases 1 – 4 do not hold, then it holds that $u_k \in V^+$ and $d_k(u_k) < \widetilde{d}_k(u_k)$ (Theorem 4.1 validates this claim). Perform the base exchange to find a vertex $u_{k+1} \in V^+$ with $d_k(u_{k+1}) > \widetilde{d}_k(u_{k+1})$ such that

$$f^+((d_k)^+ + (\chi_{u_k})^+ - (\chi_{u_{k+1}})^+) + f^+((\widetilde{d}_k)^+ - (\chi_{u_k})^+ + (\chi_{u_{k+1}})^+) \leq f^+((d_k)^+) + f^+((\widetilde{d}_k)^+).$$

Put $\zeta_{k+1} = \zeta_k$, $\widetilde{\zeta}_{k+1} = \widetilde{\zeta}_k$, $d_{k+1} = d_k + \chi_{u_k} - \chi_{u_{k+1}}$, $\widetilde{d}_{k+1} = \widetilde{d}_k - \chi_{u_k} + \chi_{u_{k+1}}$, and $k = k+1$. Repeat Step U1.

[End of Procedure Update]

To illustrate the algorithm Induction, we consider a numerical example. The underlying graph G has the vertex set

$$\begin{aligned} V &= V^+ \cup V^-, \\ & \quad V^+ = \{v_1, v_2, v_3, v_4, v_5, v_6\}, \quad V^- = \{w_1, w_2, w_3, w_4\}, \end{aligned}$$

and the arc set

$$A = \{(v_i, w_j) \mid i = 1, \cdots, 6,\ j = 1, \cdots, 4\},$$

where each arc has upper capacity 1, lower capacity 0, and weight 0. Figure 1 shows the input of the algorithm, i.e., two feasible flows $\varphi, \widetilde{\varphi}$ with $(\partial\varphi)^- = x$, $(\partial\widetilde{\varphi})^- = \widetilde{x}$ and a vertex $u = w_1 \in V^-$ with $x(u) > \widetilde{x}(u)$. Arcs with $\varphi(a) = 1$ and $\widetilde{\varphi}(a) = 1$ are drawn by solid and dotted lines, respectively, and others are missing.

Figures 2, 3, and 4 show changes of flow-base sets when the algorithm Induction is applied. Each figure corresponds to the first, second, or third iteration of Induction, respectively.

Figure 2 shows the sequence of flow-base sets produced by the procedure Update in the the first iteration of Induction. Before calling Update, we set b_1 and $\widetilde{b}_1$ as shown in Figure 2 (1-1) according to the equation (2). In the first iteration of Update, the arc $a = (v_1, w_1)$ entering the imbalance vertex w_1 satisfies $\zeta_1(a) < \widetilde{\zeta}_1(a)$. We perform flow exchange on this

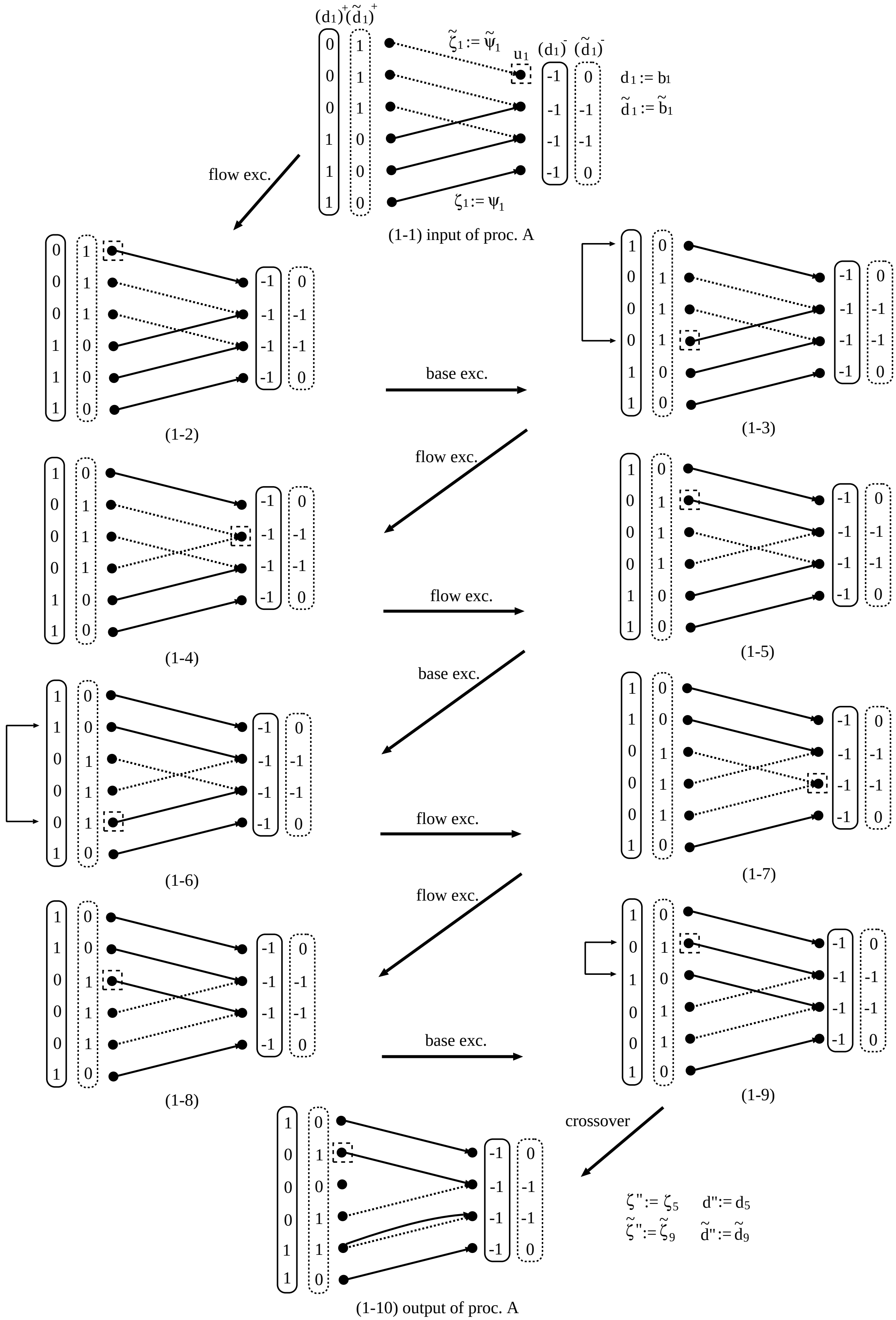

Figure 2: First iteration of INDUCTION

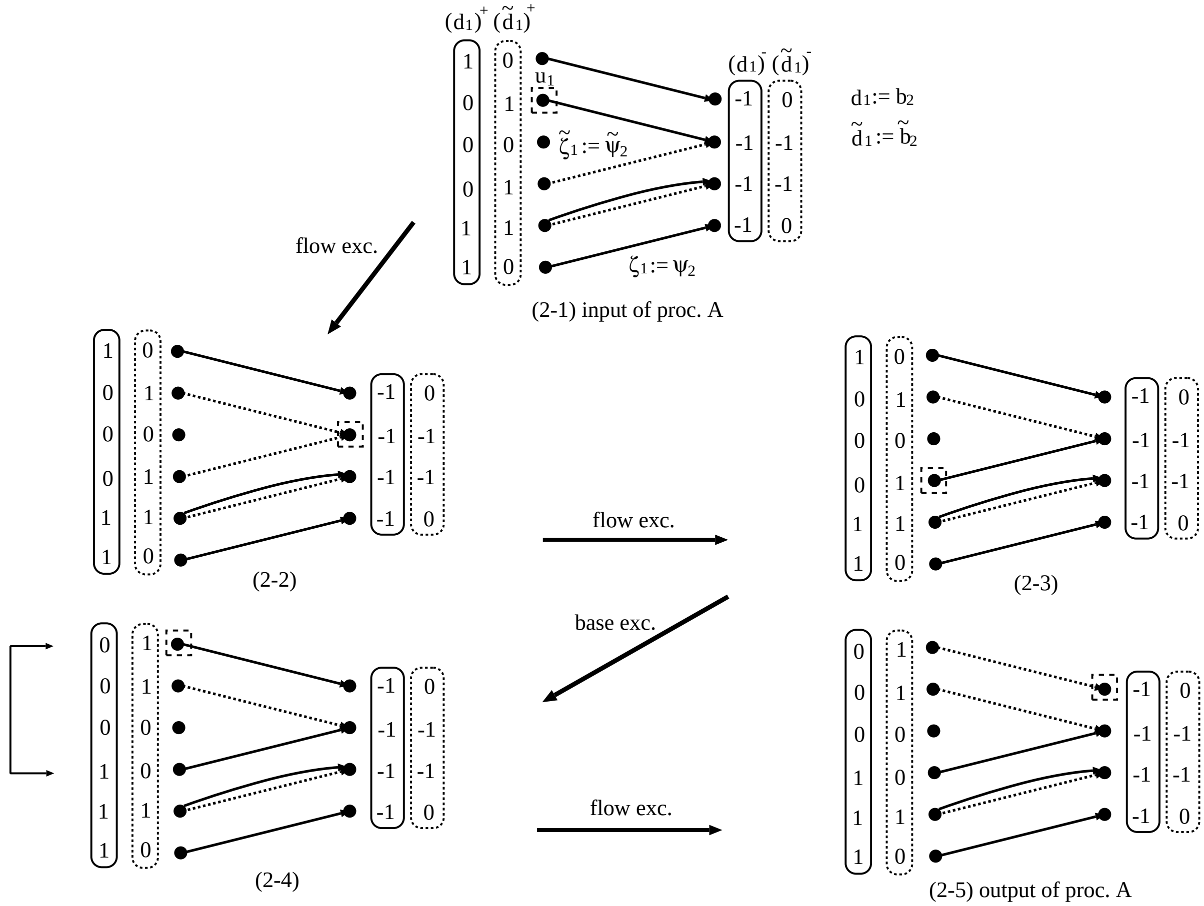

Figure 3: Second iteration of INDUCTION

arc and obtain a new flow-base set as Figure 2 (1-2). Then the imbalance vertex moves to v_1. Since

$$(\{a \in \delta^+ v_1 \mid \zeta_2(a) > \widetilde{\zeta}_2(a)\} \cup \{a \in \delta^- v_1 \mid \zeta_2(a) < \widetilde{\zeta}_2(a)\}) - \{(v_1, w_1)\} = \emptyset,$$

base exchange is applied to obtain the next flow-base set. The procedure UPDATE proceeds in this way while applying flow exchanges and base exchanges. After the eighth iteration, we get a flow-base set as shown in Figure 2 (1-9) which has the same imbalance vertex v_2 as in Figure 2 (1-5). Since the condition of Case 3 is fulfilled, crossover is performed at this point and then UPDATE terminates with output $(\zeta'', \widetilde{\zeta}'', d'', \widetilde{d}'', v_2)$. This output satisfies the condition (U-3), and the algorithm proceeds to the next iteration.

In the second iteration of INDUCTION, the procedure UPDATE is again called and yields a flow-base set as shown in Figure 3 (2-5) in its fourth iteration. Since the flow-base set meets the condition of Case 2, UPDATE terminates in the following iteration with the output $(\zeta_5, \widetilde{\zeta}_5, d_5, \widetilde{d}_5, w_1)$ satisfying (U-2).

The algorithm INDUCTION terminates at the third iteration as shown in Figure 4. The imbalance vertex is $w_4 \in V^-$ in the last iteration of UPDATE, which meets the condition of Case 1. Therefore, UPDATE outputs a flow-base set satisfying the condition (U-1) and hence INDUCTION finally stops with output w_4.

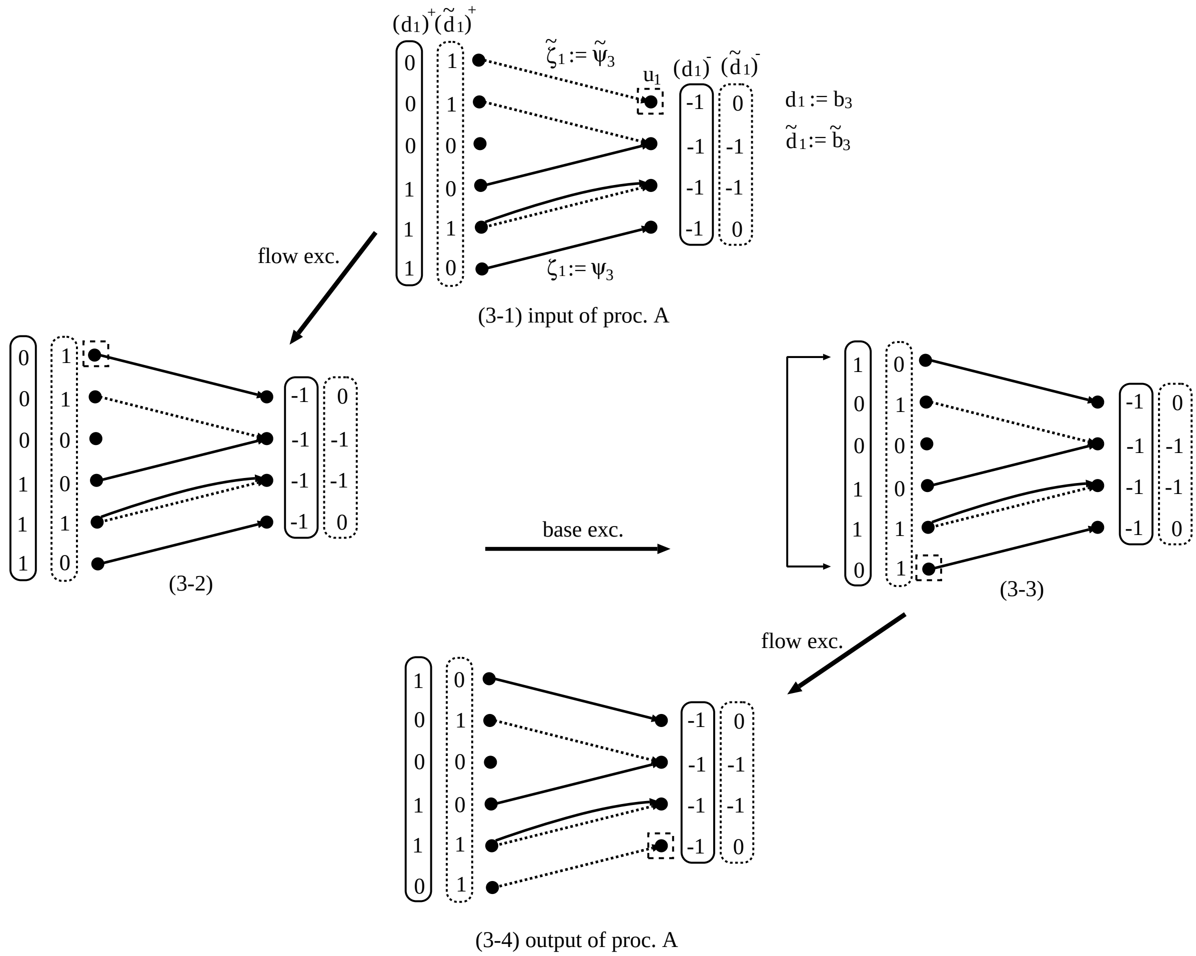

Figure 4: Third iteration of INDUCTION

4 Correctness and Time Complexity

This section proves Theorem 2.2, i.e., the correctness of the algorithm, and analyses the time complexity.

4.1 Correctness of the Procedure UPDATE

In this subsection, we show that the procedure UPDATE correctly finds the valid output.

First of all, we show the validity of Case 5 in Step U1.

Lemma 4.1 *Suppose that* $(\zeta_i, \widetilde{\zeta}_i, d_i, \widetilde{d}_i, u_i)$ *satisfies (FBS) for any* i $(1 \leq i \leq k)$*. If neither Case 1, 2, 3, nor 4 happens in the k-th iteration, then* $u_k \in V^+$ *and* $d_k(u_k) < \widetilde{d}_k(u_k)$*.*

Proof. Since Case 4 does not happen, it holds that

$$\{a \in \delta^+ u_k \mid \zeta_k(a) > \widetilde{\zeta}_k(a)\} \cup \{a \in \delta^- u_k \mid \zeta_k(a) < \widetilde{\zeta}_k(a)\} \subseteq \{e_{k-1} \text{ (if defined)}\}. \tag{8}$$

Here the LHS of (8) is equal to e_{k-1} if and only if Case 4 happens in the $(k-1)$-st iteration and $|\zeta_{k-1}(e_{k-1}) - \widetilde{\zeta}_{k-1}(e_{k-1})| = 1$ holds. Thus,

$$\partial\zeta_k(u_k) - \partial\widetilde{\zeta}_k(u_k) = \sum\{\zeta_k(a) - \widetilde{\zeta}_k(a) \mid a \in \delta^+ u_k\} - \sum\{\zeta_k(a) - \widetilde{\zeta}_k(a) \mid a \in \delta^- u_k\} \leq 1,$$

which, together with the equations

$$\partial\zeta_k(u_k) = d_k(u_k) + 1, \qquad \partial\widetilde{\zeta}_k(u_k) = \widetilde{d}_k(u_k) - 1,$$

provides $d_k(u_k) < \widetilde{d}_k(u_k)$. Furthermore, we see from the condition (FBS) that

$$(d_k)^- - (\widetilde{d}_k)^- = (x - \widetilde{x}) - 2(\chi_u)^-,$$

and therefore

$$u_k \in \{w \in V \mid d_k(w) < \widetilde{d}_k(w)\} \subseteq V^+ \cup \{w \in V^- \mid x(w) < \widetilde{x}(w)\} \cup \{u\}.$$

We have $u_k \notin \{w \in V^- \mid x(w) < \widetilde{x}(w)\}$ since Case 1 does not occur. Hence, we have only to show $u_k \neq u$. To the contrary assume that $u_k = u$. Then, we obtain that

$$(\partial\zeta_k)^- = (d_k)^- + (\chi_u)^- = x, \quad (\partial\widetilde{\zeta}_k)^- = (\widetilde{d}_k)^- - (\chi_u)^- = \widetilde{x}.$$

The inequality $\partial\zeta_k(u_k) = x(u) > \widetilde{x}(u) = \partial\widetilde{\zeta}_k(u_k)$ yields that

$$\{a \in \delta^+ u_k \mid \zeta_k(a) > \widetilde{\zeta}_k(a)\} \cup \{a \in \delta^- u_k \mid \zeta_k(a) < \widetilde{\zeta}_k(a)\} \neq \emptyset.$$

Therefore, either Case 2, 3, or 4 happens necessarily, which is a contradiction. ■

We require the following two lemmas to establish the validity of the output.

Lemma 4.2 *If Case 5 happens in some iteration, then it does not appear in the next iteration.*

Proof. Suppose that Case 5 happens in the $(k-1)$-st iteration. We see from $d_{k-1}(u_k) \geq \widetilde{d}_{k-1}(u_k) + 1$ that $d_k(u_k) \geq \widetilde{d}_k(u_k) - 1$. Thus,

$$\begin{aligned} & \sum\{\zeta_k(a) - \widetilde{\zeta}_k(a) \mid a \in \delta^+ u_k\} - \sum\{\zeta_k(a) - \widetilde{\zeta}_k(a) \mid a \in \delta^- u_k\} \\ = \; & \partial\zeta_k(u_k) - \partial\widetilde{\zeta}_k(u_k) \;=\; (d_k(u_k) + 1) - (\widetilde{d}_k(u_k) - 1) \;\geq\; 1. \end{aligned}$$

This means

$$\{a \in \delta^+ u_k \mid \zeta_k(a) > \widetilde{\zeta}_k(a)\} \cup \{a \in \delta^- u_k \mid \zeta_k(a) < \widetilde{\zeta}_k(a)\} \neq \emptyset.$$

Hence Case 5 does not appear in the k-th iteration. ■

Lemma 4.3 *All arcs in* $\{e_i \mid i \geq 1,$ *Case 4 happens in the* i*-th iteration*$\}$ *are distinct.*

Proof. To the contrary assume that there exists an arc e_k with $e_k = e_i$ for some $i < k$. From the definition of e_k, we have $i + 2 \leq k$. Since Case 3 does not happen in the k-th iteration, it holds that $u_k \notin \{u_i, u_{i+1}\}$. But it contradicts the assumption $e_k = e_i$. ■

We now prove the validity of the output of the procedure Update.

Lemma 4.4 *If either Case 1, 2, or 3 happens, then the output* $(\zeta', \widetilde{\zeta}', d', \widetilde{d}', s')$ *satisfies (U-1), (U-2), or (U-3), respectively.*

Proof. For Cases 1 and 2, the claim stands immediately from Lemmas 3.1 and 3.2.

When Case 3 happens, both $(\zeta_j, \widetilde{\zeta}_j, d_j, \widetilde{d}_j, u_j)$ and $(\zeta_k, \widetilde{\zeta}_k, d_k, \widetilde{d}_k, u_k)$ satisfy $u_k = u_j$ and $j + 2 \leq k$. From Lemmas 4.2 and 4.3, the arc set

$$F = \{e_i \mid j < i \leq k, \text{ Case 4 happens in the } i\text{-th iteration}\}$$

is nonempty and all arcs in F are distinct. We have $\zeta_k(a) \neq \zeta_j(a)$, $\widetilde{\zeta}_k(a) \neq \widetilde{\zeta}_j(a)$ for each $a \in F$, and $\zeta_k(a) = \zeta_j(a)$, $\widetilde{\zeta}_k(a) = \widetilde{\zeta}_j(a)$ for each $a \notin F$. Lemma 3.1 yields

$$\begin{aligned}
&\min\{\psi_j(a), \widetilde{\psi}_j(a)\} \leq \psi_k(a) \leq \max\{\psi_j(a), \widetilde{\psi}_j(a)\} \quad (a \in A),\\
&\min\{\psi_j(a), \widetilde{\psi}_j(a)\} \leq \widetilde{\psi}_k(a) \leq \max\{\psi_j(a), \widetilde{\psi}_j(a)\} \quad (a \in A),\\
&\psi_k(a) + \widetilde{\psi}_k(a) = \psi_j(a) + \widetilde{\psi}_j(a) \quad (a \in A).
\end{aligned}$$

Thus, we can apply Lemma 3.3, and obtain (FBS) and the relation $||\zeta'' - \widetilde{\zeta}''||_A < ||\zeta - \widetilde{\zeta}||_A$ for the output $(\zeta'', \widetilde{\zeta}'', d'', \widetilde{d}'', u_k)$ in Case 3. ■

The vertex set $\{u_1, \cdots, u_k\}$ increases by exactly one element as Step U1 repeats. Hence, the procedure UPDATE stops in at most $|V|+1$ iterations, which concludes that the procedure UPDATE finds a valid output in finite time.

4.2 Correctness of the Algorithm INDUCTION

From the correctness of the procedure UPDATE, it is obvious that we have a desired vertex v at the termination of INDUCTION, which happens in Step 2 with a vertex $s_{i+1} \in V^-$ such that $x(s_{i+1}) < \widetilde{x}(s_{i+1})$, i.e., (U-1) holds for $(\psi_{i+1}, \widetilde{\psi}_{i+1}, b_{i+1}, \widetilde{b}_{i+1}, s_{i+1})$. For the correctness of the algorithm, it is sufficient to derive that the algorithm stops in a finite number of iterations. When the condition (U-3) holds for the output $(\psi_{i+1}, \widetilde{\psi}_{i+1}, b_{i+1}, \widetilde{b}_{i+1}, s_{i+1})$ of the procedure UPDATE, we obtain the inequality

$$||\psi_{i+1} - \widetilde{\psi}_{i+1}||_A \leq ||\psi_i - \widetilde{\psi}_i||_A - 1.$$

Hence, (U-3) appears at most $||\varphi - \widetilde{\varphi}||_A$ times. If (U-2) holds, we have

$$||\psi_{i+1} - \widetilde{\psi}_{i+1}||_A \leq ||\psi_i - \widetilde{\psi}_i||_A$$

and (U-2) never happens in the next iteration since $s_{i+1} = u$. Thus, the number of iterations is at most $2||\varphi - \widetilde{\varphi}||_A$.

This concludes the proof of Theorem 2.2.

4.3 Time Complexity

We first analyze the time complexity of the procedure UPDATE. Let Ω be the time required for the evaluation of f^+.

The procedure maintains a current flow-base set $(\zeta_k, \widetilde{\zeta}_k, d_k, \widetilde{d}_k, u_k)$ and the arc set

$$E' = \{e_i \mid i \geq 1, \text{ Case 4 happens in the } i\text{-th iteration}\}.$$

It also uses a data structure for representing the vertex set $\{u_1, \cdots, u_k\}$ so that we can decide in constant time whether $w \in \{u_1, \cdots, u_k\}$ for any vertex $w \in V$. The values $f^+((d_i)^+)$ and $f^+((\widetilde{d}_i)^+)$ are restored for each $i = 1, \cdots, k$. Note that the size of the arc set E' and the vertex set $\{u_1, \cdots, u_k\}$ is at most the number of the iteration and therefore at most $|V|$.

By using the data structures above, Step U0 can be performed in $\mathrm{O}(|A|)$ time. Cases 1 and 2 of Step U1 can be done in constant time. Case 3 performs the crossover and finishes in $\mathrm{O}(|V|)$ time by using a current flow-base set $(\zeta_k, \widetilde{\zeta}_k, d_k, \widetilde{d}_k, u_k)$, the arc set E', and the restored values $f^+((d_j)^+)$ and $f^+((\widetilde{d}_j)^+)$ for some j. To execute Case 4 in constant time, we also compute the arc set

$$\{a \in \delta^+ w \mid \zeta_k(a) > \widetilde{\zeta}_k(a)\} \cup \{a \in \delta^- w \mid \zeta_k(a) < \widetilde{\zeta}_k(a)\}$$

for all vertex $w \in V$ in advance at the beginning of the procedure. This preprocessing requires $\mathrm{O}(|A|)$ time. Each iteration of the procedure checks whether

$$(\{a \in \delta^+ u_k \mid \zeta_k(a) > \widetilde{\zeta}_k(a)\} \cup \{a \in \delta^- u_k \mid \zeta_k(a) < \widetilde{\zeta}_k(a)\}) - \{e_{k-1} \text{ (if defined)}\} \neq \emptyset$$

and updates these arc sets in constant time. Case 5 takes $\mathrm{O}(|V|\Omega)$ time to find the vertex u_{k+1}. As mentioned in the previous subsection, the number of iterations is at most $|V|+1$, and the total time complexity of the procedure is $\mathrm{O}(|V|^2\Omega + |A|)$.

Next we analyze the time complexity of the algorithm Induction. The number of iterations is at most $2||\varphi - \widetilde{\varphi}||_A$ and each iteration requires the time for the procedure Update. Thus, the time complexity of the algorithm Induction is $\mathrm{O}((|V|^2\Omega + |A|)||\varphi - \widetilde{\varphi}||_A)$.

Theorem 4.5 *The algorithm* Induction *runs in* $\mathrm{O}((|V|^2\Omega + |A|)||\varphi - \widetilde{\varphi}||_A)$ *time.* ■

Acknowledgement

I am grateful to Kazuo Murota and Maiko Shigeno for discussions and their valuable comments on the manuscript.

References

[1] A. W. M. Dress and W. Wenzel, Valuated matroid: A new look at the greedy algorithm, Appl. Math. Lett. 3 (1990) 33–35.

[2] A. W. M. Dress and W. Wenzel, Valuated matroids, Adv. Math. 93 (1992), 214–250.

[3] S. Fujishige, Submodular functions and optimization, (Annals of Discrete Mathematics 47, North-Holland, Amsterdam, 1991).

[4] K. Murota, Convexity and Steinitz's exchange property, Advances in Mathematics, to appear. Extended abstract in: W. H. Cunningham, S. T. McCormick, M. Queyranne ed. Integer Programming and Combinatorial Optimization (LNCS 1084, Springer-Verlag, 1996) 260–274.

[5] K. Murota, Submodular flow problem with a nonseparable cost function, Report No. 95843-OR, Forschungsinstitut für Diskrete Mathematik, Universität Bonn (1995).

[6] K. Murota, Discrete convex analysis, RIMS preprint, No. 1065, Kyoto University (1996).