Journal of the Operations Research
Society of Japan
2004, Vol. 47, No. 1, 1-17

A POLYNOMIAL-TIME ALGORITHM FOR THE GENERALIZED INDEPENDENT-FLOW PROBLEM

Akinobu Eguchi
Panasonic Mobile Communications Co., Ltd.

Satoru Fujishige
Kyoto University

Takashi Takabatake
Kochi Gakuen College

(Received May 7, 2002; March 20, 2003)

Abstract We consider a compound problem of the generalized minimum-cost flow problem and the independent-flow problem, which we call the generalized independent-flow problem. The generalized minimum-cost flow problem is to find a minimum-cost flow in a capacitated network with gains, where each arc flow is multiplied by a gain factor when going through an arc. On the other hand, the independent-flow problem due to Fujishige is to find a minimum-cost flow in a multiple-source multiple-sink capacitated network with submodular constraints on the set of supply vectors on the source vertex set and on the set of demand vectors on the sink vertex set. We present a polynomial-time algorithm for the generalized independent-flow problem, based on Wayne's algorithm for generalized minimum-cost flows and Fujishige's algorithm for independent flows, which can be regarded as an extension of Wallacher and Zimmermann's submodular flow algorithm.

Keywords: Generalized flow, independent flow, submodular flow, polynomial-time algorithm

1. Introduction

The independent-flow problem originally developed by Fujishige [3] is a generalization of a network flow problem in a capacitated network with a source vertex set and a sink vertex set on which submodular constraints are imposed for supply vectors and demand vectors. Fujishige [3] also described several theorems that algorithmically characterize optimal solutions of this problem and proposed algorithms for solving the independent-flow problem.

It is known that the submodular flow problem due to Edmonds and Giles [2], and the polymatroidal flow problem due to Hassin [7], and Lawler and Martel [9] are equivalent to the independent-flow problem, and the class of these three problems is called the neoflow problem in [4].

On the other hand, the generalized minimum-cost flow problem has been considered in the literature (see, e.g., [12] and its references), which is a generalization of an ordinary network flow problem in a capacitated network. For each arc a we have a positive multiplier $\alpha(a)$ called a gain factor, and each unit of flow in an arc a leaving its initial-vertex reaches its terminal-vertex with $\alpha(a)$ units.

In the present paper we generalize the independent-flow problem by considering a compound problem of the independent-flow problem and the generalized minimum-cost flow problem. We call this compound problem the *generalized independent-flow problem.*

Wayne [12] recently proposed a combinatorial polynomial-time algorithm for the generalized minimum-cost flow problem. His algorithm consists of two phases, (1) the approximation phase and (2) the purification phase. The approximation phase repeatedly performs canceling scaled minimum-ratio circuits and eventually computes an approximately good generalized flow. Then, the purification phase, beginning with the obtained approximately good solution, finds a generalized minimum-cost flow by canceling negative cost circuits. Wayne's approach was further extended to linear programming by McCormick and Shioura [10].

Based on Wayne's algorithm for generalized minimum-cost flows, we propose a polynomial-time algorithm for generalized independent flows. Our algorithm also has the approximation phase and the purification phase. We cancel scaled minimum ratio circuits to find an approximately optimal solution and then transform it into an optimal one. We need to develop new techniques both in the approximation phase and the purification phase, which will be described in Section 4.

Our algorithm can also easily be adapted to get a polynomial-time algorithm for the generalized submodular flow problem, which can be regarded as an extension of Wallacher and Zimmermann's submodular flow algorithm [11].

The present paper is organized as follows. Section 2 gives definitions and notation concerned with submodular systems and generalized network flows, and introduces several preliminary results as lemmas. Section 3 describes the generalized independent-flow problem and defines residual network for solving the problem. In Section 4 we propose an algorithm for generalized independent flows and analyze its time-complexity.

2. Definitions and Preliminaries

In this section we give definitions and notation concerned with submodular systems and generalized network flows, and preliminary results to be used in the subsequent sections.

We denote the set of reals by $\mathbf{R}$ and the set of nonnegative reals by $\mathbf{R}_+$. For any finite set X we denote its cardinality by $|X|$.

2.1. Submodular systems

Let E be a nonempty finite set and $\mathcal{D}$ be a collection of subsets of E which forms a distributive lattice with set union and intersection as the lattice operations, join and meet, i.e., for each $X, Y \in \mathcal{D}$ we have $X \cup Y, X \cap Y \in \mathcal{D}$. For a distributive lattice $\mathcal{D} \subseteq 2^E$, a function $f : \mathcal{D} \to \mathbf{R}$ satisfying the following system of inequalities is called a *submodular function* on the distributive lattice $\mathcal{D}$.

$$\forall X, Y \in \mathcal{D} : f(X) + f(Y) \geq f(X \cup Y) + f(X \cap Y) \tag{2.1}$$

For a distributive lattice $\mathcal{D} \subseteq 2^E$ with $\emptyset, E \in \mathcal{D}$ and a submodular function $f : \mathcal{D} \to \mathbf{R}$ with $f(\emptyset) = 0$, we call the pair $(\mathcal{D}, f)$ a *submodular system on* E, where note that E is a unique maximal element of $\mathcal{D}$. For more details about submodular functions and submodular systems see [4].

For each nonempty $X \subseteq E$ and $x \in \mathbf{R}^E$ we define

$$x(X) = \sum_{e \in X} x(e) \tag{2.2}$$

and $x(\emptyset) = 0$. We also define a polyhedron

$$\mathrm{B}(f) = \{x \mid x \in \mathbf{R}^E,\ \forall X \in \mathcal{D} : x(X) \leq f(X),\ x(E) = f(E)\}, \tag{2.3}$$

which is called the *base polyhedron* associated with submodular system $(\mathcal{D}, f)$. Each $x \in \mathrm{B}(f)$ is called a *base* of $(\mathcal{D}, f)$. For any base x of $(\mathcal{D}, f)$ and any $e \in E$ we define

$$\mathcal{D}(x) = \{X \mid X \in \mathcal{D},\ x(X) = f(X)\}, \tag{2.4}$$

$$\mathcal{D}(x, e) = \{X \mid e \in X \in \mathcal{D}(x)\}, \tag{2.5}$$

which are sublattices of $\mathcal{D}$. Each $X \in \mathcal{D}(x)$ is called a *tight set for* x or an x*-tight set*. Denote the unique maximal element of $\mathcal{D}(x, e)$ by $\mathrm{dep}(x, e)$. The function $\mathrm{dep} : \mathrm{B}(f) \times E \to 2^E$ is called the *dependence function*. Note that

$$\mathrm{dep}(x, e) = \bigcap\{X \mid e \in X \in \mathcal{D},\ x(X) = f(X)\}, \tag{2.6}$$

which can be rewritten as

$$\mathrm{dep}(x, e) = \{e' \mid e' \in E,\ \exists\beta > 0 : x + \beta(\chi_e - \chi_{e'}) \in \mathrm{B}(f)\}, \tag{2.7}$$

where χ_e is the unit vector with $\chi_e(e) = 1$ and $\chi_e(e') = 0$ for $e' \in E \setminus \{e\}$. An ordered pair (e, e') such that $e' \in \mathrm{dep}(x, e) \setminus \{e\}$ is called an *exchangeable pair* associated with base x.

For any $x \in \mathrm{B}(f)$, $e \in E$, and $e' \in \mathrm{dep}(x, e) \setminus \{e\}$ define

$$\tilde{\mathrm{c}}(x, e, e') = \max\{\beta \mid \beta \in \mathbf{R},\ x + \beta(\chi_e - \chi_{e'}) \in \mathrm{B}(f)\}, \tag{2.8}$$

which is called the *exchange capacity* associated with x, e, and e'. Here, we define $\tilde{\mathrm{c}}(x, e, e') = +\infty$ if there exists an arbitrarily large β satisfying the condition in the right-hand side of (2.8).

The exchange capacity is also expressed as

$$\tilde{\mathrm{c}}(x, e, e') = \min\{f(X) - x(X) \mid e \in X \in \mathcal{D}, e' \notin X\}, \tag{2.9}$$

which is assumed to be equal to $+\infty$ if there does not exist any X satisfying the condition in the right-hand side. Note that for any $\beta \in \mathbf{R}$ such that $0 \le \beta \le \tilde{\mathrm{c}}(x, e, e')$ we have $x + \beta(\chi_e - \chi_{e'}) \in \mathrm{B}(f)$.

The following lemma is fundamental and will be used in the subsequent sections.

Lemma 2.1 ([11]): *Suppose* $x,\ y \in \mathrm{B}(f)$*. Consider a bipartite graph* $G^* = (X, Y; C^*)$ *with end-vertex sets*

$$X = E, \qquad Y = E' = \{e' \mid e \in E\} \tag{2.10}$$

and an arc set

$$C^* = \{(u, v') \mid u, v \in E, u \in \mathrm{dep}(x, v) \setminus \{v\}\} \subseteq E \times E', \tag{2.11}$$

where v' *denotes a copy of* v*. A capacity* $c(u, v')$ *of each arc* (u, v') *in* C^* *is defined to be equal to* $\tilde{\mathrm{c}}(x, v, u)$*. Then, there exists a flow* $\varphi : C^* \to \mathbf{R}_+$ *in network* $\mathcal{N} = (G^*, c)$ *satisfying the capacity constraints and*

$$\varphi(\delta^+ u) = \max\{0, x(u) - y(u)\} \quad (u \in X), \tag{2.12}$$

$$\varphi(\delta^- v') = \max\{0, y(v) - x(v)\} \quad (v' \in Y), \tag{2.13}$$

where $\delta^+ u = \{(u, v') \mid v' \in Y, (u, v') \in C^*\}$ *and* $\delta^- v' = \{(u, v') \mid u \in X, (u, v') \in C^*\}$*.*

A weaker version of Lemma 2.1 with infinite capacities for C^* was given in [3].

A sequence of monotone increasing elements

$$\mathcal{C} : S_0 \subset S_1 \subset \cdots \subset S_k \tag{2.14}$$

of $\mathcal{D}$ is called a *chain* of $\mathcal{D}$. If there exists no chain that contains $\mathcal{C}$ as a proper subsequence, then $\mathcal{C}$ is called a *maximal chain* of $\mathcal{D}$. Note that for any maximal chain $\mathcal{C}$ given by (2.14) we have $S_0 = \emptyset$ and $S_k = E$. We also have the following lemma.

Lemma 2.2: *Any maximal chain $\emptyset = S_0 \subset S_1 \subset \cdots \subset S_k = E$ of $\mathcal{D}$ gives us a partition*

$$\Pi(\mathcal{D}) = \{S_i \setminus S_{i-1} \mid i = 1, 2, \cdots, k\} \tag{2.15}$$

of E, which is independent of the choice of a maximal chain of $\mathcal{D}$.

Consider a submodular system $(\mathcal{D}, f)$ on E and a nonempty subset F of E. Regarding F as a new singleton e_F, define $E||F = (E \setminus F) \cup \{e_F\}$, a distributive lattice $\mathcal{D}||F$ by

$$\mathcal{D}||F = \{X \mid X \in \mathcal{D},\ X \cap F = \emptyset\} \cup \{(X \setminus F) \cup \{e_F\} \mid X \in \mathcal{D},\ X \supseteq F\}, \tag{2.16}$$

and a submodular function $f : \mathcal{D}||F \to \mathbf{R}$ by

$$(f||F)(X) = \begin{cases} f(X) & \text{if } e_F \notin X \\ f((X \setminus \{e_F\}) \cup F) & \text{if } e_F \in X \end{cases} \qquad (X \in \mathcal{D}||F). \tag{2.17}$$

Then $(\mathcal{D}||F, f||F)$ is a submodular system on $E||F$, which we call the *aggregation* of $(\mathcal{D}, f)$ by F. For any collection of disjoint subsets $F_1, F_2, \cdots, F_k$ of E we get a submodular system by repeated aggregations of $(\mathcal{D}, f)$ by $F_1, F_2, \cdots, F_k$, which is also called the aggregation of $(\mathcal{D}, f)$ by $\{F_1, F_2, \cdots, F_k\}$. It should be noted that the definition of aggregation given here is slightly different from the original one given in [4].

2.2. Generalized circulations

Let $G = (V, A)$ be a directed graph with a finite vertex set V and a finite arc set A. For any arc $a \in A$ we denote by $\partial^+ a$ and $\partial^- a$, respectively, the initial end-vertex and the terminal end-vertex of a. Also, for any vertex $v \in V$ denote by $\delta^+ v$ and $\delta^- v$, respectively, the set of arcs leaving v and the set of arcs entering v, i.e., $\delta^+ v = \{a \mid a \in A,\ \partial^+ a = v\}$ and $\delta^- v = \{a \mid a \in A,\ \partial^- a = v\}$. We sometimes express an arc a as the ordered pair $(\partial^+ a, \partial^- a)$ of its initial and terminal end-vertices when there is no possibility of confusion.

A *generalized network* $\mathcal{G} = (G = (V, A), c, \alpha)$ is a capacitated network with gains, where $G = (V, A)$ is its underlying directed graph, $c : A \to \mathbf{R}_+$ a *capacity function*, and $\alpha : A \to \mathbf{R}_+$ a positive *gain function*. Here, one unit of flow in an arc a leaving its initial end-vertex $\partial^+ a$ becomes $\alpha(a)$ units of flow when it reaches its terminal end-vertex $\partial^- a$.

A function $\varphi : A \to \mathbf{R}_+$ is called a *generalized circulation*, or simply a *circulation*, in $\mathcal{G}$ if it satisfies the following capacity constraints (2.18) and flow conservation law (2.19):

$$\forall a \in A : 0 \le \varphi(a) \le c(a), \tag{2.18}$$

$$\forall v \in V : \partial\varphi(v) = 0, \tag{2.19}$$

where the *boundary* $\partial\varphi : V \to \mathbf{R}$ of flow φ in $\mathcal{G}$ is defined as

$$\partial\varphi(v) = \sum_{a \in \delta^+ v} \varphi(a) - \sum_{a \in \delta^- v} \alpha(a)\varphi(a) \quad (v \in V). \tag{2.20}$$

The *gain of a cycle* C is defined as the product of gain factors $\alpha(a)$ of arcs a lying on cycle C, where a cycle is a directed elementary closed path in G. Denote the gain of cycle C by $\alpha(C)$. A cycle C with $\alpha(C) = 1$ is called a *unit-gain cycle*. A cycle C with $\alpha(C) > 1$ is called a *flow-generating cycle* and a cycle C with $\alpha(C) < 1$ a *flow-absorbing cycle*. A *bicycle* consists of a flow-generating cycle, a flow-absorbing cycle and a (possibly degenerate) path from the first cycle to the second, where the flow-generating and flow-absorbing cycles may have common arcs or vertices and we assume that the path connecting the two cycles does not have a common vertex with the cycles except for its initial and terminal vertices. A *circuit* is a circulation in $\mathcal{G}$ that takes on positive values only on arcs of a unit-gain cycle or a bicycle.

We have the following lemma (see Gondran and Minoux [6]) that any circulation is decomposed into a collection of a small number of circuits.

Lemma 2.3: *Let φ be a circulation in $\mathcal{G}$. Then φ can be decomposed into circuits ψ_1, $\psi_2, \cdots, \psi_k$ in $\mathcal{G}$ with $k \le m$ as $\varphi = \sum_{i=1}^{k} \psi_i$ such that for each $i = 1, 2, \cdots, k$ and $a \in A$, $\psi_i(a) > 0$ implies $\varphi(a) > 0$.*

3. The Generalized Independent-Flow Problem

In this section we describe the generalized independent-flow problem and give the definition of residual network associated with a generalized independent flow. We also show two fundamental theorems: one relates two generalized independent flows and the other characterizes optimal generalized independent flows, both in terms of residual network.

3.1. Problem description

Let $G = (V, A; S^+, S^-)$ be a graph with an n-vertex set V, an m-arc set A, a set $S^+ \subseteq V$ of *entrances* (or *sources*), and a set $S^- \subseteq V$ of *exits* (or *sinks*), where we assume that $S^+ \cap S^- = \emptyset$. Also, let $c : A \to \mathbf{R}_+$ be a *capacity function*, $\alpha : A \to \mathbf{R}_+$ a positive *gain function*, and $\gamma : A \to \mathbf{R}$ a *cost function*. Moreover, let $(\mathcal{D}^+, f^+)$ and $(\mathcal{D}^-, f^-)$ be submodular systems on S^+ and S^-, respectively. The dependence functions and exchange capacity functions for submodular systems $(\mathcal{D}^\pm, f^\pm)$ are denoted by $\tilde{\mathrm{c}}^\pm$ and $\mathrm{dep}^\pm$. Let us denote by $\mathcal{N}_{GI} = (G = (V, A; S^+, S^-), c, \gamma, \alpha, (\mathcal{D}^+, f^+), (\mathcal{D}^-, f^-))$ the network described above.

Now, consider the following flow problem in $\mathcal{N}_{GI}$.

$$
\begin{aligned}
P_{GI} : \text{Minimize} \quad & \sum_{a \in A} \gamma(a)\varphi(a) && (3.1)\\
\text{subject to} \quad & 0 \le \varphi(a) \le c(a) \quad (a \in A), && (3.2)\\
& \partial\varphi(v) = 0 \quad (v \in V - (S^+ \cup S^-)), && (3.3)\\
& \partial^+\varphi \in \mathrm{B}(f^+), && (3.4)\\
& \partial^-\varphi \in \mathrm{B}(f^-). && (3.5)
\end{aligned}
$$

Here $\partial^+\varphi$ is the restriction of $\partial\varphi$ to S^+ and $\partial^-\varphi$ is the restriction of $-\partial\varphi$ to S^-, where note that ∂ is the boundary operator in the underlying generalized network $\mathcal{G} = (G, c, \alpha)$ with gain function α. A function φ satisfying constraints (3.2)$\sim$(3.5) is called a *generalized independent flow*, or simply a *feasible flow*, in $\mathcal{N}_{GI}$. A generalized independent flow φ can be regarded as a flow in the underlying generalized flow network $\mathcal{G} = (G, c, \alpha)$ with entrance vertex set S^+ and exit vertex set S^- whose supply vector $\partial^+\varphi$ and demand vector $\partial^-\varphi$ are, respectively, bases of $(\mathcal{D}^+, f^+)$ and $(\mathcal{D}^-, f^-)$. We call Problem P_{GI} described above a *generalized independent-flow problem* and an optimal solution of P_{GI} an *optimal generalized independent flow*, or simply an *optimal flow*, in $\mathcal{N}_{GI}$.

We assume that we are given an initial feasible flow in $\mathcal{N}_{GI}$ (see Appendix A). Hence, without loss of generality we assume $\mathbf{0} \in \mathrm{B}(f^+)$ and $\mathbf{0} \in \mathrm{B}(f^-)$ so that $\varphi = \mathbf{0}$ is a feasible flow in $\mathcal{N}_{GI}$. We also assume that cost function γ, capacity function c, and submodular functions $f^\pm$ are integer-valued and that gain function α is rational-valued, each $\alpha(a)$ $(a \in A)$ being expressed as a ratio of positive integers. We denote by B the maximum absolute value of the integers taken on by these integer-valued functions $f^\pm, c$ and γ, and integers appearing as ratios of two integers for gain factors. We assume $B \ge 2$.

Given a feasible flow φ in $\mathcal{N}_{GI}$, we denote the objective function value of (3.1) for φ by $\gamma(\varphi)$. We also denote the minimum value of the objective function by γ^*. We call a feasible flow φ in $\mathcal{N}_{GI}$ *ϵ-optimal* if $\gamma(\varphi) \leq \gamma^* + \epsilon$, i.e., its cost value is within ϵ from the optimal one. Note that $\gamma^* \leq 0$ since the zero flow is feasible due to the assumption. The definition of ϵ-optimality given here is different from the ordinary relative approximate optimality as employed in [12]; readers will find it suitable for our purpose.

It should be noted that when $S^+ = V$ and $S^- = \emptyset$, the generalized independent-flow problem P_{GI} can be regarded as a compound problem of generalized flows and submodular flows, which we call the *generalized submodular-flow problem.*

3.2. Residual network

Given a feasible flow φ in $\mathcal{N}_{GI}$, define the *residual network* $\mathcal{N}_\varphi$=$(G_\varphi$=(V, A_φ), c_φ, $\gamma_\varphi, \alpha_\varphi)$ associated with φ as follows. The residual network is essential in our algorithm to find an ϵ-optimal flow. The vertex set of $\mathcal{N}_\varphi$ is V, the same as that of $\mathcal{N}_{GI}$, and the arc set A_φ is given by

$$A_\varphi = A_\varphi^+ \cup A_\varphi^- \cup A_\varphi^* \cup B_\varphi^*, \tag{3.6}$$

$$A_\varphi^+ = \{(u, v) \mid v \in S^+, u \in \mathrm{dep}^+(\partial^+\varphi, v) - \{v\}\}, \tag{3.7}$$

$$A_\varphi^- = \{(v, u) \mid v \in S^-, u \in \mathrm{dep}^-(\partial^-\varphi, v) - \{v\}\}, \tag{3.8}$$

$$A_\varphi^* = \{a \mid a \in A, \varphi(a) < c(a)\}, \tag{3.9}$$

$$B_\varphi^* = \{\bar{a} \mid a \in A, \varphi(a) > 0\} \ \ (\bar{a} : \text{a reorientation of arc } a). \tag{3.10}$$

Also, the residual capacity function $c_\varphi : A_\varphi \to \mathbf{R}_+$ is given by

$$c_\varphi(a) = \begin{cases} \tilde{c}^+(\partial^+\varphi, v, u) & (a = (u, v) \in A_\varphi^+) \\ \tilde{c}^-(\partial^-\varphi, v, u) & (a = (v, u) \in A_\varphi^-) \\ c(a) - \varphi(a) & (a \in A_\varphi^*) \\ \alpha(\bar{a})\varphi(\bar{a}) & (a \in B_\varphi^*) \end{cases} \qquad (a \in A_\varphi), \tag{3.11}$$

where $\tilde{c}^+$ and $\tilde{c}^-$ are exchange capacities associated with $(\mathcal{D}^+, f^+)$ and $(\mathcal{D}^-, f^-)$, respectively. Furthermore, the residual cost function $\gamma_\varphi : A_\varphi \to \mathbf{R}$ is defined by

$$\gamma_\varphi(a) = \begin{cases} \gamma(a) & (a \in A_\varphi^*) \\ -\gamma(\bar{a})/\alpha(\bar{a}) & (a \in B_\varphi^*) \\ 0 & (a \in A_\varphi^+ \cup A_\varphi^-) \end{cases} \qquad (a \in A_\varphi), \tag{3.12}$$

and the residual gain function $\alpha_\varphi : A_\varphi \to \mathbf{R}_+$ by

$$\alpha_\varphi(a) = \begin{cases} \alpha(a) & (a \in A_\varphi^*) \\ 1/\alpha(\bar{a}) & (a \in B_\varphi^*) \\ 1 & (a \in A_\varphi^+ \cup A_\varphi^-) \end{cases} \qquad (a \in A_\varphi). \tag{3.13}$$

The following theorem is concerned with the expression of a difference of two generalized independent flows in a residual network considered as a generalized flow network.

Theorem 3.1: *For two feasible flows φ and φ' in $\mathcal{N}_{GI}$ consider the residual network $\mathcal{N}_\varphi$ associated with φ and define a function $\psi : A_\varphi^* \cup B_\varphi^* \to \mathbf{R}_+$ by*

$$\psi(a) = \begin{cases} \varphi'(a) - \varphi(a) & \text{if } a \in A_\varphi^* \text{ and } \varphi'(a) > \varphi(a) \\ \alpha(\bar{a})(\varphi(\bar{a}) - \varphi'(\bar{a})) & \text{if } a \in B_\varphi^* \text{ and } \varphi'(\bar{a}) < \varphi(\bar{a}) \\ 0 & \text{otherwise} \end{cases} \tag{3.14}$$

for each $a \in A^*_\varphi \cup B^*_\varphi$*. Then* ψ *can be extended on arc set* A_φ *of* $\mathcal{N}_\varphi$ *to be a generalized circulation in* $\mathcal{N}_\varphi$*.*
(Proof) Let $\psi^+ : A^+_\varphi \to \mathbf{R}_+$ be a function such that

$$\psi^+(u,v) \le \tilde{c}^+(\partial^+\varphi, v, u) \qquad ((u,v) \in A^+_\varphi), \tag{3.15}$$

$$\sum_{v:\ (u,v)\in A^+_\varphi} \psi^+(u,v) = \max\{0, \partial^+\varphi(u) - \partial^+\varphi'(u)\} \qquad (u \in S^+), \tag{3.16}$$

$$\sum_{u:\ (u,v)\in A^+_\varphi} \psi^+(u,v) = \max\{0, \partial^+\varphi'(v) - \partial^+\varphi(v)\} \qquad (v \in S^+). \tag{3.17}$$

The existence of such a function ψ^+ follows from Lemma 2.1. Similarly we can define a function $\psi^- : A^-_\varphi \to \mathbf{R}_+$ such that the direct sum of ψ, ψ^+, and ψ^- is a desired extension of ψ that is a circulation in $\mathcal{N}_\varphi$. □

Suppose that we are given a feasible flow φ in $\mathcal{N}_{GI}$ and let ψ be a circuit in $\mathcal{N}_\varphi$ that takes on positive values only on a unit-gain cycle or bicycle Q. We change flow φ by using ψ to get a new φ' as follows. We denote by $A(Q)$ the set of arcs in Q.

$$\varphi'(a) = \begin{cases} \varphi(a) + \psi(a) & \text{if } a \in A^*_\varphi \text{ and } a \in A(Q) \\ \varphi(a) - \psi(\bar{a})/\alpha(a) & \text{if } \bar{a} \in B^*_\varphi,\ \bar{a} \in A(Q), \\ & \qquad \text{and } \bar{a} \text{ is a reorientation of } a \\ \varphi(a) & \text{otherwise} \end{cases} \tag{3.18}$$

for each $a \in A$. We call this operation *changing flow* φ *by circuit* ψ.

The following theorem characterizes optimal generalized independent flows in terms of residual network.

Theorem 3.2: *A feasible flow* φ *in* $\mathcal{N}_{GI}$ *is optimal if and only if there is no circuit of negative cost in residual network* $\mathcal{N}_\varphi$ *with respect to cost function* γ_φ*.*
(Proof) If there is a circuit ψ of negative cost in residual network $\mathcal{N}_\varphi$ with respect to cost function γ_φ, then for a sufficiently small $\beta > 0$ changing φ by circuit $\beta\psi$ yields a feasible flow in $\mathcal{N}_{GI}$ that has a cost smaller than that of φ. Hence φ is not optimal.

Conversely, suppose that there is no circuit of negative cost in residual network $\mathcal{N}_\varphi$. Let φ' be any feasible flow in $\mathcal{N}_{GI}$. Then let $\psi : A_\varphi \to \mathbf{R}_+$ be a generalized circulation in $\mathcal{N}_\varphi$ as in Theorem 3.1. From Lemma 2.3 ψ can be decomposed into circuits ψ_i $(i \in I)$ such that $\psi = \sum_{i\in I} \psi_i$. It follows from the assumption that

$$\gamma(\varphi') - \gamma(\varphi) = \gamma_\varphi(\psi) = \sum_{i\in I} \gamma_\varphi(\psi_i) \ge 0. \tag{3.19}$$

Hence φ is an optimal flow in $\mathcal{N}_{GI}$. □

4. Algorithms

We basically adopt Wayne's approach [12] to the generalized independent-flow problem by incorporating Wayne's generalized minimum-cost flow algorithm [12] with Fujishige's independent-flow algorithm [3]. Our algorithm can be regarded as an extension of Wallacher and Zimmermann's submodular flow algorithm [11]. As in Wayne's algorithm [12], our algorithm consists of two phases: an approximation phase and a purification phase. The approximation phase repeatedly modifies a current flow along circuits of negative cost to improve the objective function value, and the purification phase along circuits of nonpositive cost. In both phases we need new techniques that are not in [3, 11, 12].

4.1. A scaling algorithm for approximation

The following lemma is fundamental in getting a new feasible flow by changing a feasible flow φ by a circuit in its associated residual network $\mathcal{N}_\varphi$.

Lemma 4.1: *Let φ be a feasible flow in $\mathcal{N}_{GI}$ and ψ be a circuit in $\mathcal{N}_\varphi$ with its associated unit-gain cycle or bicycle Q. Define $k = \max\{1, |A(Q) \cap A_\varphi^+|, |A(Q) \cap A_\varphi^-|\}$, where $A(Q)$ denotes the set of arcs in Q. Then a new flow obtained by changing φ by $(1/k)\psi$ is also a feasible flow in $\mathcal{N}_{GI}$.*

(Proof) Let the arcs in $A(Q) \cap A_\varphi^+$ be given by (u_i, v_i) $(i \in I)$. Then, for any $\lambda_i \in \mathbf{R}$ such that $0 \leq \lambda_i \leq \tilde{c}^+(\partial^+\varphi, v_i, u_i)$ we have

$$\forall i \in I : \partial^+\varphi + \lambda_i(\chi_{v_i} - \chi_{u_i}) \in \mathrm{B}(f^+). \tag{4.1}$$

Since $\mathrm{B}(f^+)$ is a convex set, for any positive integer $l \geq |I|$ we have

$$\partial^+\varphi + \frac{1}{l}\sum_{i \in I} \lambda_i(\chi_{v_i} - \chi_{u_i}) \in \mathrm{B}(f^+). \tag{4.2}$$

A similar fact holds for arcs in $A(Q) \cap A_\varphi^-$. Hence, for $k = \max\{1, |A(Q) \cap A_\varphi^+|, |A(Q) \cap A_\varphi^-|\}$ the flow φ' obtained by changing φ by circuit $(1/k)\psi$ satisfies

$$\partial^+\varphi' \in \mathrm{B}(f^+), \qquad \partial^-\varphi' \in \mathrm{B}(f^-). \tag{4.3}$$

Because of the changing of φ by circuit $(1/k)\psi$ in $\mathcal{N}_\varphi$ the capacity constraints and the flow conservation law are satisfied by the new φ'. Hence φ' is a feasible flow in $\mathcal{N}_{GI}$. □

Based on this lemma, we carefully choose a circuit for changing a current flow to make our algorithm efficient. For a circuit ψ in $\mathcal{N}_\varphi$ we define *ratio* $\mu_\varphi(\psi)$ as

$$\mu_\varphi(\psi) = \sum_{a \in A_\varphi} \gamma_\varphi(a)\psi(a) \Big/ \sum_{a \in A_\varphi} t_\varphi(a)\psi(a), \tag{4.4}$$

where $t_\varphi : A_\varphi \to \mathbf{R}_+$ is defined as $t_\varphi(a) = 1/c_\varphi(a)$ for each $a \in A_\varphi$ and we write $t_\varphi(\psi) = \sum_{a \in A_\varphi} t_\varphi(a)\psi(a)$ in the sequel. Such a ratio was first introduced by Wallacher and Zimmermann [11] for submodular flows and adapted to generalized minimum-cost flows by Wayne [12].

Our scaling approximation algorithm is given as follows.

Algorithm_Approximation

Input: $\mathcal{N}_{GI}$ and ϵ (> 0).

Output: An ϵ-optimal flow φ in $\mathcal{N}_G$.

Step 1: Put $\varphi \leftarrow \mathbf{0}$, $\hat{\mu}^* = -B^2$, and $\hat{\mu} \leftarrow \hat{\mu}^*/2$.

Step 2: If $\hat{\mu} \geq -\epsilon/2(m + n^2)$, then φ is ϵ-optimal and return φ.

Step 3: While there is a circuit ψ such that $\mu_\varphi(\psi) < \hat{\mu}$, find such a circuit ψ, let Q be its underlying unit-gain cycle or bicycle, put $k \leftarrow \max\{1, |A(Q) \cap A_\varphi^+|, |A(Q) \cap A_\varphi^-|\}$, and change current φ by circuit $(1/k)\psi$.

Step 4: Put $\hat{\mu} \leftarrow \hat{\mu}/2$ and go to Step 2.

End

Step 3 with current $\hat{\mu}$ is called a *$\hat{\mu}$-scaling phase.*

We will describe how to perform Step 3 efficiently and show the correctness of our algorithm.

Lemma 4.2: *Let φ be a feasible flow in $\mathcal{N}_{GI}$ and ψ be a circuit in $\mathcal{N}_\varphi$ such that $\psi(a) = c_\varphi(a)$ for at least one arc $a \in A_\varphi$. Suppose that ψ is of negative-cost. If for a positive integer k we can change φ by circuit $(1/k)\psi$ to obtain a new feasible flow φ', then we have*

$$\gamma(\varphi') - \gamma(\varphi) \leq \mu_\varphi(\psi)/k. \tag{4.5}$$

Here, we can choose a positive integer k as in Step 3 of Algorithm_Approximation *and k is at most $2n$.*
(Proof) From the assumption,

$$\begin{aligned}\gamma(\varphi') - \gamma(\varphi) &= \gamma_\varphi((1/k)\psi) = \mu_\varphi((1/k)\psi)t_\varphi((1/k)\psi) \\ &= \mu_\varphi(\psi)t_\varphi((1/k)\psi) \leq \mu_\varphi(\psi)/k,\end{aligned} \tag{4.6}$$

where note that $\mu_\varphi((1/k)\psi) = \mu_\varphi(\psi)$, $\mu_\varphi(\psi) < 0$, and $t_\varphi((1/k)\psi) \geq 1/k$.

Moreover, from Lemma 4.1 we can choose a positive integer k such that $k \leq |\{a \mid a \in A_\varphi^+ \cup A_\varphi^-,\ \psi(a) > 0\}|$. Hence $k \leq 2n$. □

Lemma 4.3: *Let φ be a feasible flow in $\mathcal{N}_{GI}$, γ^* be the cost of an optimal flow in $\mathcal{N}_{GI}$ and μ_φ^* be the minimum ratio value in $\mathcal{N}_\varphi$. Then we have*

$$\mu_\varphi^* \leq \frac{\gamma^* - \gamma(\varphi)}{m + n^2}. \tag{4.7}$$

(Proof) Let φ^* be an optimal flow in $\mathcal{N}_{GI}$. Also let $\psi : A_\varphi \to \mathbf{R}_+$ be a circulation in $\mathcal{N}_\varphi$ appearing in Theorem 3.1 where φ' should be replaced by φ^*. As in Lemma 2.3, decompose ψ into circuits ψ_i $(i \in I)$ in $\mathcal{N}_\varphi$ such that $\psi = \sum_{i\in I} \psi_i$. Then we have

$$\mu_\varphi^* \leq \min\{\mu_\varphi(\psi_i) \mid i \in I\} \leq \frac{\gamma_\varphi(\psi)}{t_\varphi(\psi)} \leq \frac{\gamma_\varphi(\psi)}{m + n^2} = \frac{\gamma^* - \gamma(\varphi)}{m + n^2}, \tag{4.8}$$

where note that $t_\varphi(\psi) \leq |A_\varphi| \leq m + n^2$. □

Lemma 4.4: *There are at most $4n(m + n^2)$ changings by circuits in each $\hat{\mu}$-scaling phase of* Algorithm_Approximation*.*
(Proof) When $2\hat{\mu} = \hat{\mu}^*(= -B^2)$, at the beginning of the $\hat{\mu}$-scaling phase, the minimum ratio value $\mu_{\mathbf{0}}^*$ in $\mathcal{N}_{\mathbf{0}}$ satisfies

$$\mu_{\mathbf{0}}^* = \mu_{\mathbf{0}}(\psi) \geq \frac{\sum_{a\in A_{\mathbf{0}}}(-B)\psi(a)}{\sum_{a\in A_{\mathbf{0}}} t_{\mathbf{0}}(a)\psi(a)} \geq \frac{\sum_{a\in A_{\mathbf{0}}}(-B)\psi(a)}{\sum_{a\in A_{\mathbf{0}}}(1/B)\psi(a)} = -B^2 = \hat{\mu}^*, \tag{4.9}$$

where ψ is a circuit in $\mathcal{N}_{\mathbf{0}}$ that attains the minimum ratio value $\mu_{\mathbf{0}}^*$. It follows that there is no circuit whose ratio is less than $2\hat{\mu}$ for the initial value of $\hat{\mu}$. This also holds true after $\hat{\mu}$ is cut in half in Step 4, due to Step 3. Hence $2\hat{\mu} \leq \mu_\varphi^*$ holds for any current flow φ. It follows from Lemma 4.3 that

$$2\hat{\mu}(m + n^2) \leq \gamma^* - \gamma(\varphi) \leq 0. \tag{4.10}$$

In a $\hat{\mu}$-scaling phase we repeatedly change a current flow by a circuit of ratio less than $\hat{\mu}$. Lemma 4.2 implies that this improves the objective function value by at least $\hat{\mu}/2n$. It follows from (4.10) that the number of changings by circuits in the $\hat{\mu}$-scaling phase is at most $4n(m + n^2)$. □

We assume that an exchange capacity in an arbitrary base polyhedron can be computed in η time. We also use $\tilde{\mathrm{O}}(f)$ to denote $\mathrm{O}(f \log^{\mathrm{O}(1)}(m+n))$.

Lemma 4.5: *We can compute an ϵ-optimal flow in $\tilde{\mathrm{O}}(n^5(\eta+n^2)(\log(1/\epsilon)+\log B))$ time by* Algorithm_Approximation.

(Proof) Since $\hat{\mu} = -B^2/2$ in Step 1 and $\hat{\mu}$ is cut in half at the end of each scaling phase, our approximation algorithm carries out $\tilde{\mathrm{O}}(\log(1/\epsilon)+\log B)$ scaling phases until $\hat{\mu} \geq -\epsilon/2(m+n^2)$. Moreover, from Lemma 4.3, at the end of Algorithm_Approximation we have

$$-\epsilon \leq 2\hat{\mu}(m+n^2) \leq \mu^*_\varphi(m+n^2) \leq \gamma^* - \gamma(\varphi) \tag{4.11}$$

for a finally obtained flow φ. Hence, φ is ϵ-optimal.

As shown in Lemma 4.4, we repeat changing of a current flow by a circuit at most $4n(m+n^2)$ times in each scaling phase.

We can discern whether there exists a circuit whose ratio is less than $\hat{\mu}$ and if any exists, we can find such a circuit by checking feasibility of TVPI (two variables per inequality) systems as proposed by Wayne in [12]. Define a reduced cost $\gamma_{\hat{\mu}}(a) = \gamma(a) - \hat{\mu}t(a)$ for $a \in A_\varphi$. It is easy to see that a circuit has a ratio less than $\hat{\mu}$ if and only if there is a negative-cost circuit with respect to cost function $\gamma_{\hat{\mu}}$. Existence of such a circuit is equivalent to infeasibility of the following TVPI system:

$$\forall a \in A_\varphi : \gamma_{\hat{\mu}}(a) + \pi(\partial^+ a) - \alpha(a)\pi(\partial^- a) \geq 0 \tag{4.12}$$

due to the linear programming duality. This feasibility test can be done in $\tilde{\mathrm{O}}((m+n^2)n^2)$ [1, 8], and when the system is infeasible, as its by-product we obtain a unit-gain cycle or bicycle that gives a circuit of ratio smaller than $\hat{\mu}$.

Since it takes $\mathrm{O}(n^2\eta+m)$ time to construct each residual network, the total complexity of Algorithm_Approximation is $\tilde{\mathrm{O}}((\log(1/\epsilon)+\log B)4n(m+n^2)(n^2\eta+m+(m+n^2)n^2))$, i.e., $\tilde{\mathrm{O}}(n^5(\eta+n^2)(\log(1/\epsilon)+\log B))$. □

4.2. An algorithm for purification

Given a feasible flow φ, we transform φ to another feasible flow φ' such that φ' is an extreme-point flow of cost no more than $\gamma(\varphi)$. This transformation is called *purification* in linear programming. Our purification algorithm uses flow-type techniques instead of matrix computation. We will use this purification algorithm later to obtain an optimal flow from a $1/(B^{8m}+1)$-optimal flow in $\mathcal{N}_{GI}$.

First, we give further definitions to describe our purification algorithm. Let φ be a feasible flow in $\mathcal{N}_{GI}$ and consider the residual network $\mathcal{N}_\varphi = (G_\varphi = (V, A_\varphi), c_\varphi, \gamma_\varphi, \alpha_\varphi)$ associated with φ. We define $\hat{A}_\varphi$ to be the set of arcs in A_φ that have (positive) residual capacities in both directions, i.e.,

$$\hat{A}_\varphi = \{a \mid a \in A_\varphi,\ c_\varphi(a) > 0,\ c_\varphi(\bar{a}) > 0\}, \tag{4.13}$$

where $\bar{a}$ is a reorientation of a. Define a network $\hat{\mathcal{N}}_\varphi = (\hat{G}_\varphi = (V, \hat{A}_\varphi), \hat{c}_\varphi, \hat{\gamma}_\varphi, \hat{\alpha}_\varphi)$, where $\hat{c}_\varphi$, $\hat{\gamma}_\varphi$, and $\hat{\alpha}_\varphi$ are restrictions of c_φ, γ_φ, and α_φ to $\hat{A}_\varphi$.

We have partitions $\Pi(\mathcal{D}^+(\partial^+\varphi))$ of S^+ and $\Pi(\mathcal{D}^-(\partial^-\varphi))$ of S^-, where see (2.4) and (2.15) for the notation. Here, it should be noted that components of $\Pi(\mathcal{D}^+(\partial^+\varphi))$ are exactly strongly connected components (regarding as subsets of S^+) of the subgraph (S^+, A^+_φ) of G_φ, and similarly for $\Pi(\mathcal{D}^-(\partial^-\varphi))$. Let $\hat{G}'_\varphi = (V', \hat{A}'_\varphi)$ be the graph obtained from $\hat{G}$ by shrinking each component of $\Pi(\mathcal{D}^+(\partial^+\varphi))$ and $\Pi(\mathcal{D}^-(\partial^-\varphi))$ into a single vertex. We then define an ordinary generalized network $\hat{\mathcal{N}}'_\varphi = (\hat{G}'_\varphi = (V', \hat{A}'_\varphi), \hat{c}'_\varphi, \hat{\gamma}'_\varphi, \hat{\alpha}'_\varphi)$, where $\hat{c}'_\varphi$, $\hat{\gamma}'_\varphi$, and $\hat{\alpha}'_\varphi$ are restrictions of $\hat{c}_\varphi$, $\hat{\gamma}_\varphi$, and $\hat{\alpha}_\varphi$ to $\hat{A}'_\varphi$.

Lemma 4.6: *Given a unit-gain cycle or bicycle in* $\hat{\mathcal{N}}'_\varphi$*, we can construct a unit-gain cycle or bicycle* Q *in* $\mathcal{N}_\varphi$ *and cancel a circuit on* Q *after* $\mathrm{O}(n)$ *calls of an exchange capacity oracle.*
(Proof) Let Q' be a unit-gain cycle or bicycle in $\hat{\mathcal{N}}'_\varphi$. Denote by $\hat{A}(Q')$ the set of arcs in Q'. Then consider $\hat{A}(Q')$ as a subset of $\hat{A}_\varphi$ before the shrinking, and let $\hat{W}$ be the end-vertices of arcs in $\hat{A}(Q')$ with respect to $\hat{G}_\varphi$. Denote by $\hat{G}_\varphi(\hat{W})$ the subgraph of $\hat{G}_\varphi$ induced by $\hat{W}$.

For any pair of vertices u and v in a single component of partition $\Pi(\mathcal{D}^+(\partial^+\varphi))$ or $\Pi(\mathcal{D}^-(\partial^-\varphi))$ there exist both arcs (u,v) and (v,u) in G_φ. Hence for any pair of vertices in $\hat{G}_\varphi(\hat{W})$ that belong to a single component of $\Pi(\mathcal{D}^+(\partial^+\varphi))$ or $\Pi(\mathcal{D}^-(\partial^-\varphi))$ there exist both arcs (u,v) and (v,u) in G_φ.

Note that since every vertex v in a cycle has exactly two arcs incident to v in the cycle, at most four vertices in $\hat{W}$ are shrunk to a single vertex in Q' of $\hat{G}'_\varphi$.

We first construct a unit-gain cycle or bicycle Q contained in $\hat{G}_\varphi(\hat{W})$. We keep a set A_Q of arcs which will form a desired unit-gain cycle or bicycle Q. Initially we set $A_Q = \{a \mid a \in Q'\} \subseteq \hat{A}_\varphi$ and then add some arcs in $\hat{G}_\varphi(\hat{W})$ that are missing in $\hat{G}'_\varphi$ after shrinking.

If a vertex $v \in Q'$ corresponds, by shrinking, to two vertices v_1 and v_2 in $\hat{W}$, we assume without loss of generality that there are an arc of Q' entering v_1 and another arc of Q' leaving v_2. Then we add arc (v_1, v_2) to A_Q.

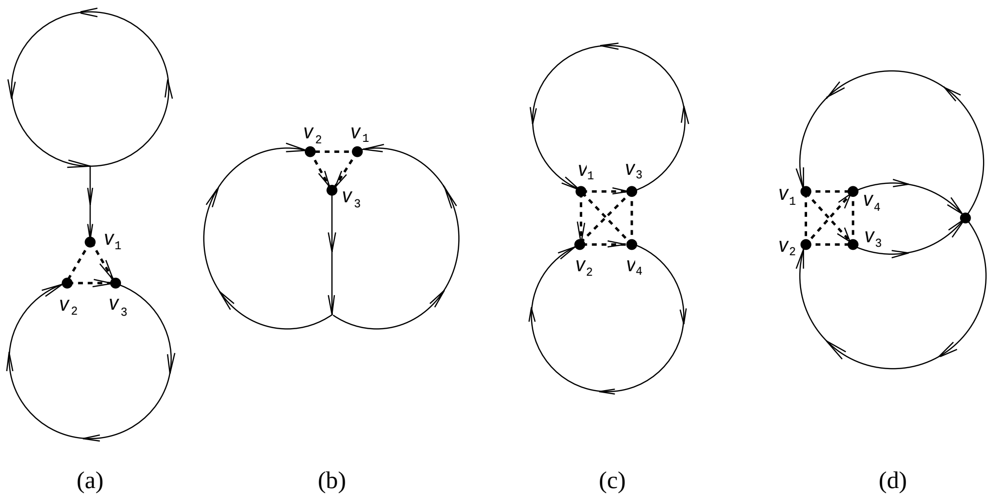

Figure 1: Examples of vertex sets expanded for bicycles.

If v corresponds to three vertices v_1, v_2, and v_3 (see Figure 1 (a) and (b)), then let us first assume that two vertices v_1 and v_2 are terminal end-vertices of two arcs of Q' and v_3 is an initial end-vertex of another arc of Q'. We then add arcs (v_1, v_3) and (v_2, v_3) to A_Q. When two of these three vertices are initial end-vertices of some two arcs of Q' and one is a terminal end-vertex of another arc of Q', we add to A_Q two arcs from the latter vertex to the former two vertices in a similar way.

If v corresponds to four vertices v_1, v_2, v_3, and v_4 (see Figure 1 (c) and (d)), then without loss of generality we assume the following (i) and (ii).

(i) There are an arc entering v_1 and another arc leaving v_3, and these two arcs belong to the flow-generating cycle in Q'.
(ii) There are an arc entering v_2 and another arc leaving v_4, and these two arcs belong to the flow-absorbing cycle in Q'.

We add arcs (v_1, v_3) and (v_2, v_4) to A_Q. If v is a specified degenerate path connecting the flow-generating cycle and the flow-absorbing cycle (Figure 1 (c)), then we also add arc

(v_1, v_2) to A_Q. Here, we assume that such a degenerate path is uniquely specified to each bicycle.

After adding arcs to A_Q for each vertex in Q' that corresponds to at least two vertices in $\hat{W}$, the arcs in A_Q form a unit-gain cycle or bicycle Q in $\mathcal{N}_\varphi$.

Now we consider a circuit ψ that takes on positive values only on Q. Such a circuit ψ is uniquely determined up to a positive multiple factor. We compute a maximum $\beta > 0$ such that changing the current φ by circuit $\beta\psi$ yields a feasible flow in $\mathcal{N}_{GI}$. We next show that we can compute such a maximum $\beta > 0$ by $\mathrm{O}(n)$ calls of an exchange capacity oracle.

Denote by φ_β the new flow obtained by changing current φ by circuit $\beta\psi$. Flow φ_β is feasible in $\mathcal{N}_{GI}$ if and only if the following (I) and (II) hold.

(I) For each $a \in A^*_\varphi \cup B^*_\varphi$ we have

$$\beta\psi(a) \le c_\varphi(a). \tag{4.14}$$

(II) For the entrance vertex set S^+ and the exit vertex set S^-,

$$\partial^+\varphi_\beta \in \mathrm{B}(f^+), \qquad \partial^-\varphi_\beta \in \mathrm{B}(f^-), \tag{4.15}$$

where $\partial^\pm$ is with respect to the original network $\mathcal{N}_{GI}$.

Condition (I) implies that φ_β satisfies the capacity constraints. Since $\beta\psi$ is a circuit, φ_β satisfies the flow conservation law. Condition (II) is exactly (3.4) and (3.5).

The maximum β satisfying Condition (I) is easily computed. Therefore, let us consider a problem of determining the maximum β that satisfies Condition (II). It should be noted that there is no arc in $A(Q) \cap (A^+_\varphi \cup A^-_\varphi)$ that connects distinct two components of partition $\Pi(\mathcal{D}^+(\partial^+\varphi))$ or $\Pi(\mathcal{D}^-(\partial^-\varphi))$. Hence, if Condition (II) holds, we have

$$\mathcal{D}^+(\partial^+\varphi) \subseteq \mathcal{D}^+(\partial^+\varphi_\beta), \qquad \mathcal{D}^-(\partial^-\varphi) \subseteq \mathcal{D}^-(\partial^-\varphi_\beta). \tag{4.16}$$

This means that we can determine the maximum β that satisfies Condition (II) by considering separate subproblems; each corresponds to a component of $\Pi(\mathcal{D}^+(\partial^+\varphi))$ or $\Pi(\mathcal{D}^-(\partial^-\varphi))$, as follows.

Let U^+_i $(i \in I^+)$ be components of $\Pi(\mathcal{D}^+(\partial^+\varphi))$ such that $\partial^+\varphi_\beta(u) \ne \partial^+\varphi(u)$ for some $u \in U^+_i$. Then $\partial^+\varphi_\beta$ is expressed as

$$\partial^+\varphi_\beta = \partial^+\varphi + \beta \sum_{i \in I^+} b^+_i, \tag{4.17}$$

where for each $i \in I^+$, b^+_i is a nonzero vector such that the size of its support $\mathrm{supp}(b^+_i)$ is at most four and $b^+_i(U^+_i) = 0$ due to the definition of Q.

Now, for each $i \in I^+$ consider

(II^+_i)

$$\partial^+\varphi + \beta b^+_i \in \mathrm{B}(f^+). \tag{4.18}$$

We can easily see from (4.16) and (4.17) that $\partial^+\varphi_\beta \in \mathrm{B}(f^+)$ if and only if (4.18) holds for all $i \in I^+$. Similarly, we consider Condition (II^+_i) $(i \in I^-)$ for the exit vertex set S^-.

For each $i \in I^+$ the maximum β that satisfies (4.18) is computed as follows. Divide $\mathrm{supp}(b^+_i)$ into two sets W_1 and W_2 such that $b^+_i(W_1) > 0$ and $b^+_i(W_2) < 0$. Let $(\partial^+\varphi)'$ and $(f^+)'$ be those obtained by the aggregation of $\partial^+\varphi$ and f^+ by W_1 and W_2, and let v_{W_1} and v_{W_2} be new vertices corresponding to W_1 and W_2. Define

$$\beta^+_i(W_1, W_2) = \tilde{\mathrm{c}}((\partial^+\varphi)', v_{W_1}, v_{W_2})/b^+_i(W_1), \tag{4.19}$$

where $\tilde{c}$ is the exchange capacity associated with the aggregation. Denote by $\hat{\beta}_i^+$ the minimum of values $\beta_i^+(W_1, W_2)$ for all such divisions (W_1, W_2). Similarly define $\hat{\beta}_i^-$ $(i \in I^-)$ for the exit S^-. Hence the required minimum β, denoted by $\hat{\beta}$, is given by

$$\hat{\beta} = \min\{\min\{c_\varphi(a)/\psi(a) \mid \psi(a) > 0, a \in A^*_\varphi \cup B^*_\varphi\}, \\ \min\{\hat{\beta}_i^+ \mid i \in I^+\}, \min\{\hat{\beta}_i^- \mid i \in I^-\}\}. \tag{4.20}$$

Since $|\mathrm{supp}(b_i^\pm)| \le 4$ for each $i \in I^\pm$, the number of possible such divisions (W_1, W_2) is at most seven, so that determining $\hat{\beta}$ requires $\mathrm{O}(n)$ calls of an exchange capacity oracle.

Moreover, after changing current φ by circuit $\hat{\beta}\psi$ we get a new flow $\varphi_{\hat{\beta}}$ and new residual network $\mathcal{N}_{\varphi_{\hat{\beta}}}$ in which Q is neither a unit-gain cycle nor a bicycle, i.e., circuit ψ is canceled, due to (4.20). □

It should be noted that if $\hat{\beta} = \min\{c_\varphi(a)/\psi(a) \mid \psi(a) > 0, a \in A^*_\varphi \cup B^*_\varphi\}$ and the minimum is attained by arc $\hat{a}$, then arc $\hat{a}$ disappears in new residual network $\mathcal{N}_{\varphi_{\hat{\beta}}}$ and its corresponding $\hat{\mathcal{N}}_{\varphi_{\hat{\beta}}}$. If $\hat{\beta} = \min\{\hat{\beta}_i^+ \mid i \in I^+\}$ and the minimum is attained by $i_0 \in I^+$, then let $X^+ \in \mathcal{D}$ be a new tight set for $\partial^+\varphi_{\hat{\beta}}$ that separates W_1 and W_2 given as above for $U_{i_0}^+$. Because of this tight set, arcs between W_1 and W_2 disappear in $\hat{\mathcal{N}}_{\varphi_{\hat{\beta}}}$. Furthermore, the size of partition $\Pi(\mathcal{D}^+(\partial^+\varphi_{\hat{\beta}}))$ is larger than that of previous $\Pi(\mathcal{D}^+(\partial^+\varphi))$, and similarly for the exit if $\hat{\beta} = \min\{\hat{\beta}_i^- \mid i \in I^-\}$. Hence,

Lemma 4.7: *Canceling a circuit as in Lemma* 4.6 *can be made* $\mathrm{O}(n+m)$ *times.*

The following lemma characterizes extreme points of the feasible flow polyhedron.

Lemma 4.8: *A feasible flow φ in $\mathcal{N}_{GI}$ is an extreme point of the feasible flow polyhedron for $\mathcal{N}_{GI}$ if and only if there does not exist a unit-gain cycle or a bicycle in network $\hat{\mathcal{N}}'_\varphi$ defined above.*

(Proof) We first prove the 'only if' part. Assume that there exists a unit-gain cycle or bicycle Q' in $\hat{\mathcal{N}}'_\varphi$. Let ψ be a circuit in $\hat{\mathcal{N}}_\varphi$ constructed from Q' as in Lemma 4.6. It follows from the definition of $\hat{\mathcal{N}}'_\varphi$ that for a sufficiently small $\beta > 0$ changings of φ by circuit $\beta\psi$ and by $-\beta\psi$ both yield feasible flows and hence φ is not an extreme point.

Next, we show the 'if' part. Suppose that there does not exist a unit-gain cycle or a bicycle in network $\hat{\mathcal{N}}'_\varphi$. Then consider a new cost function $\bar{\gamma} : A \to \mathbf{R}$ such that its associated $\bar{\gamma}'_\varphi : A'_\varphi \to \mathbf{R}$ instead of original γ_φ is a nonnegative function and satisfies

$$a \in A'_\varphi,\ \bar{a} \notin A'_\varphi \Longrightarrow \bar{\gamma}'_\varphi(a) > 0, \tag{4.21}$$

where $\bar{a}$ is a reorientation of a. Note that if $a, \bar{a} \in A'_\varphi$, then $\bar{\gamma}'_\varphi(a) = \bar{\gamma}'_\varphi(\bar{a}) = 0$. Now, for any feasible flow φ' in $\mathcal{N}_{GI}$ we have inequality (3.19) with γ replaced by $\bar{\gamma}$ and in particular (3.19) with γ replaced by $\bar{\gamma}$ holds with strict inequality since every unit-gain cycle or bicycle contains an arc a such that $\bar{\gamma}'_\varphi(a) > 0$. It follows that φ is an extreme point of the feasible flow polyhedron for $\mathcal{N}_{GI}$. □

Now we describe an algorithm for purification.

Algorithm_Purification

Input: $\mathcal{N}_{GI}$ and a feasible flow φ in $\mathcal{N}_{GI}$.

Output: A feasible flow $\hat{\varphi}$ that is an extreme point of the feasible flow polyhedron and has a cost not more than $\gamma(\varphi)$.

Step 1: Construct $\hat{\mathcal{N}}'_\varphi$.

Step 2: While there is a unit-gain cycle or a bicycle in $\hat{\mathcal{N}}'_\varphi$, cancel a circuit in $\mathcal{N}_\varphi$ associated with a unit-gain cycle or a bicycle in $\hat{\mathcal{N}}'_\varphi$ of nonpositive cost with respect to cost function $\hat{\gamma}'_\varphi$.

Put $\hat{\varphi} \leftarrow \varphi$ and return $\hat{\varphi}$.

End

We examine the complexity of Algorithm_Purification.

Lemma 4.9: Algorithm_Purification *runs in* $\mathrm{O}(mn(n\eta + m))$ *time.*

(Proof) Because of Lemma 4.7 we repeat canceling a circuit in Step 2 of Algorithm_Purification $\mathrm{O}(n+m)$ times. Constructing a residual network and its shrunken network $\hat{\mathcal{N}}'_\varphi$ requires $\mathrm{O}(n^2\eta + m)$ time, finding a unit-gain cycle or bicycle in $\hat{\mathcal{N}}'_\varphi$ $\mathrm{O}(mn)$ time, rescaling a circuit in $\mathcal{N}_\varphi$ $\mathrm{O}(n+n\eta)$ times, and canceling a unit-gain cycle or bicycle in $\mathcal{N}_\varphi$ $\mathrm{O}(m+n)$ time. Here, note that finding a unit-gain cycle or a bicycle is done in a network $\hat{\mathcal{N}}'_\varphi$ with $\mathrm{O}(m)$ arcs, which requires $\mathrm{O}(mn)$ time by the Bellman-Ford shortest-path algorithm (see Appendix B). Also note that any cycle or bicycle has $\mathrm{O}(n)$ arcs. Hence Algorithm_Purification runs in $\mathrm{O}(mn(n\eta + m))$ time. □

The following lemma shows a value of ϵ such that we find an optimal flow in $\mathcal{N}_{GI}$ in polynomial time by combining Algorithm_Approximation and Algorithm_Purification.

Lemma 4.10: *Suppose* $\epsilon = 1/(B^{8m}+1)$ *and let* φ_0 *be any feasible flow in* $\mathcal{N}_{GI}$ *that is an extreme point of the feasible flow polyhedron and has a cost not more than that of an* ϵ*-optimal flow. Then* φ_0 *is an optimal flow in* $\mathcal{N}_{GI}$.

(Proof) For a $1/(B^{8m}+1)$-optimal flow φ we have $\gamma(\varphi) < \gamma^* + B^{-8m}$. By Cramer's rule, a cost of any extreme-point flow is a rational number. We will show that costs of two extreme-point flows have a common denominator not greater than B^{8m}. It then follows that an extreme-point flow of cost not more than that of a $1/(B^{8m}+1)$-optimal flow is an optimal flow.

Now, let φ_0 be an extreme-point flow. Define $\bar{A} = \{a \mid a \in A,\ \varphi_0(a) = c(a)\}$ and $\underline{A} = \{a \mid a \in A,\ \varphi_0(a) = 0\}$. Then φ_0 is a unique solution of the following system of linear equations for φ:

$$\varphi(a) = c(a) \quad (a \in \bar{A}), \tag{4.22}$$
$$\varphi(a) = 0 \quad (a \in \underline{A}), \tag{4.23}$$
$$\partial\varphi(v) = 0 \quad (v \in V \setminus (S^+ \cup S^-)), \tag{4.24}$$
$$\partial^+\varphi(X) = f^+(X) \quad (X \in \mathcal{D}^+(\partial^+\varphi_0)), \tag{4.25}$$
$$\partial^-\varphi(X) = f^-(X) \quad (X \in \mathcal{D}^-(\partial^-\varphi_0)), \tag{4.26}$$

where $\mathcal{D}^\pm(\partial^\pm\varphi_0)$ are defined as in (2.4). We can easily see that choosing maximal chains $\mathcal{C}^\pm$ of $\mathcal{D}^\pm(\partial^\pm\varphi_0)$, the system of equations (4.25) and (4.26) is equivalent to the following

$$\partial^+\varphi(X) = f^+(X) \quad (X \in \mathcal{C}^+), \tag{4.27}$$
$$\partial^-\varphi(X) = f^-(X) \quad (X \in \mathcal{C}^-). \tag{4.28}$$

Letting $\mathcal{C}^+ = (S_0^+ = \emptyset, S_1^+, \cdots, S_{k^+}^+ = S^+)$ and $\mathcal{C}^- = (S_0^- = \emptyset, S_1^-, \cdots, S_{k^-}^- = S^-)$, it is further equivalent to

$$\partial^+\varphi(S_i^+ \setminus S_{i-1}^+) = f^+(S_i^+) - f^+(S_{i-1}^+) \quad (i = 1, 2, \cdots, k^+), \tag{4.29}$$
$$\partial^-\varphi(S_i^- \setminus S_{i-1}^-) = f^-(S_i^-) - f^-(S_{i-1}^-) \quad (i = 1, 2, \cdots, k^-). \tag{4.30}$$

Let $M\varphi = b$ be the system of linear equations formed by (4.22)∼(4.24), (4.29), and (4.30). Note that the number of rows of M is at most $m+n$. Let N be a nonsingular $m \times m$ submatrix of M, so that $\varphi_0 = N^{-1}b_N$ where b_N is a restriction of b on the components corresponding to rows of N.

For each $a \in A$ denote by Δ_a the determinant of a matrix obtained by replacing the column corresponding to a of N by b_N. By Cramer's rule $\varphi_0(a)$ is given as $\varphi_0(a) = \Delta_a/|N|$, where $|N|$ is the determinant of N. Recall that we have

$$|N| = \sum_{\sigma \in \mathcal{P}_m} \operatorname{sgn}(\sigma)\ N(1,\sigma(1))N(2,\sigma(2)) \cdots N(m,\sigma(m)), \tag{4.31}$$

where $\mathcal{P}_m$ is the set of all permutations of $\{1, 2, \cdots, m\}$ and $\operatorname{sgn}(\sigma) = 1$ or -1 according as σ is an even permutation or an odd permutation. Since each element of N is equal to zero, one, or $-\alpha(a)$ $(a \in A)$, the absolute value of each summand in (4.31) is equal to zero or product of some gain factors. It follows from (4.22)∼(4.24), (4.29), and (4.30) that there are at most three nonzero elements in each column. Hence the number of nonzero terms in (4.31) is at most 3^m. Denoting the denominator of $\alpha(a)$ by $\alpha_d(a)$ for each arc $a \in A$, we can assume that the denominator of $|N|$ is $\prod_{j=1}^{m} \alpha_d(a_j)$ and that the numerator is less than $3^m B^m$, which is further less than $B^{2m}B^m = B^{3m}$ when $B \geq 2$. Similarly, we can see that the denominator of Δ_a is at most $\prod_{j=1}^{m} \alpha_d(a_j)$. Since the denominator of $\varphi_0(a)$ is given by the product of the denominator of Δ_a and the numerator of $|N|$, the denominator of $\varphi_0(a)$ is at most B^{4m}. It follows that the denominator of the cost of any extreme-point flow is at most B^{4m} and that a common denominator of the difference of costs of any two extreme-point flows can be at most B^{8m}. This completes the proof of the present lemma. □

From the above argument we have the following theorem.

Theorem 4.11: *Combining* Algorithm_Approximation *with* $\epsilon = 1/(B^{8m}+1)$ *and* Algorithm_Purification, *we can find an optimal flow in* $\tilde{\mathrm{O}}(n^5(n^2+\eta)m\log B)$ *time.*

(Proof) We first use Algorithm_Approximation to obtain a $1/(B^{8m}+1)$-optimal flow and then find an optimal flow by Algorithm_Purification. The former requires $\tilde{\mathrm{O}}(n^5(n^2+\eta)m\log B)$ time and the latter $\tilde{\mathrm{O}}(mn(n\eta+m))$ time. □

5. Concluding Remarks

We have proposed the model of generalized independent-flow problem and a weakly polynomial-time algorithm for solving it. There is possibility of improving the complexity of our algorithm by incorporating recent development in algorithms for submodular flows (see, e.g., [5]), which is left for future research.

Acknowledgments

We are grateful to Akira Nakayama and two anonymous referees for their useful comments that improved the presentation of this paper. The present work was supported by a Grant-in-Aid of the ministry of Education, Culture, Sports, Science and Technology of Japan.

References

[1] E. Cohen and N. Megiddo: Improved algorithms for linear inequalities with two variables per inequality. *SIAM Journal on Computing*, **23** (1994) 1313–1347.

[2] J. Edmonds and R. Giles: A min-max relation for submodular functions on graphs. *Annals of Discrete Mathematics*, **1** (1977) 185–204.

[3] S. Fujishige: Algorithms for solving the independent-flow problems. *Journal of the Operations Research Society of Japan*, **21** (1978) 189–203.
[4] S. Fujishige: *Submodular functions and optimization* (North-Holland, 1991).
[5] S. Fujishige and S. Iwata: Algorithms for submodular flows. *IEICE Transactions on Information and Systems*, **E83-D** (2000) 322–329.
[6] M. Gondran and M. Minoux: *Graphs and Algorithms* (Wiley, 1984).
[7] R. Hassin: Minimum cost flow with set-constraints. *Networks*, **12** (1982) 1–21.
[8] D. S. Hochbaum and J. Naor: Simple and fast algorithms for linear and integer programs with two variables per inequality. *SIAM Journal on Computing*, **23** (1994) 1179–1192.
[9] E. L. Lawler and C. U. Martel: Computing maximal polymatroidal network flows. *Mathematics of Operations Research*, **7** (1982) 334–347.
[10] S. T. McCormick and A. Shioura: Minimum ratio canceling is oracle polynomial for linear programming, but not strongly polynomial, even for networks. *Operations Research Letters*, **27** (2000) 199–207.
[11] C. Wallacher and U. T. Zimmermann: A polynomial cycle canceling algorithm for submodular flows. *Mathematical Programming, Ser. A*, **86** (1999) 1–15.
[12] K. D. Wayne: A polynomial combinatorial algorithm for generalized minimum cost flow. *Proceedings of the 31st Annual ACM Symposium on Theory of Computing*, (1999) 11–18.

Appendix A

We describe an algorithm for finding an initial feasible flow in original network $\mathcal{N}_{GI}$.

First, we find extreme bases $b^+ \in \mathrm{B}(f^+)$ and $b^- \in \mathrm{B}(f^-)$ by the so-called greedy algorithm for base polyhedra.

Next, to each vertex $v \in S^+ \cup S^-$ we add a selfloop a_v and define a flow value $\varphi(a_v)$ and a gain value $\alpha(a_v)$ as follows: if $v \in S^+$ and $b^+(a_v) > 0$ (or if $v \in S^-$ and $b^-(a_v) < 0$), then put $\varphi(a_v) = 2b^+(a_v)$ and $\alpha(a_v) = 1/2$ (or $\varphi(a_v) = -2b^-(a_v)$ and $\alpha(a_v) = 1/2$), and if $v \in S^+$ and $b^+(a_v) < 0$ (or if $v \in S^-$ and $b^-(a_v) > 0$), then put $\varphi(a_v) = -b^+(a_v)$ and $\alpha(a_v) = 2$ (or $\varphi(a_v) = b^-(a_v)$ and $\alpha(a_v) = 2$). Moreover, we assume that capacities of the added selfloops are sufficiently large and that their costs are equal to a positive constant. We put $\varphi(a) = 0$ for each original arc $a \in A$.

Then φ is a feasible flow in the extended network, so that we can apply our proposed algorithm to find an optimal flow $\hat{\varphi}$. Note that $\hat{\varphi}(a_v) = 0$ for each added selfloop a_v if and only if there exists a feasible flow in the original network $\mathcal{N}_{GI}$, and that restricting $\hat{\varphi}$ on the original arc set A gives a feasible flow in the original network (if it is feasible).

Appendix B

A method for finding a unit-gain cycle or a bicycle was presented by Wayne [12]. We describe it here in detail for completeness.

Recall that a bicycle consists of a flow-generating cycle, a flow-absorbing cycle, and a path connecting the first cycle to the second. It is easy to see that the set of flow-absorbing cycles in $\mathcal{N}_\varphi$ is in one-to-one correspondence with that of negative length cycles with respect to length function $+\log \alpha_\varphi$ defined by $(+\log \alpha_\varphi)(a) = \log \alpha_\varphi(a)$ $(a \in A_\varphi)$, and similarly, that the set of flow-generating cycles is in one-to-one correspondence with that of negative length cycles with respect to length function $-\log \alpha_\varphi$ defined by $(-\log \alpha_\varphi)(a) = -\log \alpha_\varphi(a)$ $(a \in A_\varphi)$. The basic approach is to first try to find a bicycle if one exists and otherwise to try to find a unit-gain cycle.

Algorithm_Circuit

Input: A residual network $\mathcal{N}_\varphi$.

Output: A unit-gain cycle or bicycle Q in $\mathcal{N}_\varphi$ if one exists, or NONE if none exists.

Step 1: Put $G' \leftarrow G_\varphi$. Find a flow-generating cycle (a cycle of negative length) by using the Bellman-Ford shortest-path algorithm for G' with length function $-\log \alpha_\varphi$, where we add new vertex s and new arcs (s, v) of zero length for every vertex v in G'. If none exists, then put $G'_2 \leftarrow G'$, let G'_1 be an empty graph, and go to Step 3. Otherwise find a set W of vertices that participate in cycles of negative length (flow-generating cycles) or can be reached from a flow-generating cycle along a (directed) path, which can be done by adapting the previous Bellman-Ford shortest-path computation. Let G'' be the subgraph of G_φ induced by W.

Step 2: If a Bellman-Ford computation for G'' with length function $+\log \alpha_\varphi$ restricted on the arc set of G'' finds a flow-absorbing cycle in G'', then find a bicycle Q and return Q. Otherwise put $G'_1 \leftarrow G''$ and G'_2 be the subgraph of G_φ induced by $V \setminus W$. Let G' be the direct sum of G'_1 and G'_2, and go to Step 3.

Step 3: Construct a subgraph H of current graph G' that consists of arcs a satisfying $l_\pi(a) = l(a) + \pi(\partial^+ a) - \pi(\partial^- a) = 0$, where $l(a) = (+\log \alpha_\varphi)(a)$ for arc a in G'_1 and $l(a) = (-\log \alpha_\varphi)(a)$ for arc a in G'_2, and the vertex label $\pi : V \to \mathbf{R}_+$ is the distance label already computed by the last Bellman-Ford computation in Steps 1 and 2. Detect a cycle by using the depth-first search or the breadth-first search in H. If we find a cycle Q, it is a unit-gain cycle in $\mathcal{N}_\varphi$ and return Q; otherwise return NONE.

End

It should be noted that Algorithm_Circuit invokes the Bellman-Ford shortest-path algorithm twice.

Akinobu Eguchi
Panasonic Mobile Communications
Mobile Technology Solutions Center
Corporate Engineering Division
`eguchi.akinobu@jp.panasonic.com`

Satoru Fujishige
RIMS, Kyoto University
`fujishig@kurims.kyoto-u.ac.jp`

Takashi Takabatake
Department of Medical Technology
Kochi Gakuen College
`takabatake@kochi-gc.ac.jp`