2012 IEEE 53rd Annual Symposium on Foundations of Computer Science

Concave Generalized Flows with Applications to Market Equilibria

László A. Végh
Department of Management
London School of Economics
London, UK
Email: L.Vegh@lse.ac.uk

***Abstract*—We consider a nonlinear extension of the generalized network flow model, with the flow leaving an arc being an increasing concave function of the flow entering it, as proposed by Truemper [1] and Shigeno [2]. We give a polynomial time combinatorial algorithm for solving corresponding flow maximization problems, finding an ε-approximate solution in $O(m(m + \log n)\log(MUm/\varepsilon))$ arithmetic operations and value oracle queries, where M and U are upper bounds on simple parameters. This also gives a new algorithm for linear generalized flows, an efficient, purely scaling variant of the Fat-Path algorithm by Goldberg, Plotkin and Tardos [3], not using any cycle cancellations.**

We show that this general convex programming model serves as a common framework for several market equilibrium problems, including the linear Fisher market model and its various extensions. Our result immediately provides combinatorial algorithms for various extensions of these market models. This includes nonsymmetric Arrow-Debreu Nash bargaining, settling an open question by Vazirani [4].

***Keywords*-network flow algorithms; generalized flows; convex programming; market equilibrium.**

I. Introduction

A classical extension of network flows is the *generalized network flow model*, with a gain factor $\gamma_e > 0$ associated with each arc e so that if α units of flow enter arc e, then $\gamma_e\alpha$ units leave it. Since first studied by Kantorovich [5], Dantzig [6] and Jewell [7], the problem has found many applications including financial analysis, transportation, management science, see [8, Chapter 15].

In this paper, we consider a nonlinear extension, *concave generalized flows*, studied by Truemper [1] in 1978, and by Shigeno [2] in 2006. For each arc e we are given a concave, monotone increasing function Γ_e such that if α units enter e then $\Gamma_e(\alpha)$ units leave it. We give a combinatorial algorithm for corresponding flow maximization problems, with running time polynomial in the network data and some simple parameters.

Generalized flows are linear programs and thus can be solved efficiently by general linear programming techniques, the currently most efficient such algorithm being the interior-point method by Kapoor and Vaidya [9]. Combinatorial approaches have been used since the sixties (e.g. [7], [10], [11]), yet the first polynomial-time combinatorial algorithms were given only in 1991 by Goldberg, Plotkin and Tardos [3]. This inspired a line of research to develop further polynomial-time combinatorial algorithms, e.g. [12]–[21]; for a survey on combinatorial generalized flow algorithms, see [22]. Despite the vast literature, no strongly polynomial algorithm is known so far. Our algorithm for this special case derives from the Fat-Path algorithm in [3], with the remarkable difference that no cycle cancellations are needed.

Nonlinear extensions of generalized flows have also been studied, e.g. in [23], [24], minimizing a separable convex cost function for generalized flows. However, these frameworks do not contain our problem, which involves nonlinear convex constraints.

Concave generalized flows being nonlinear convex programs, they can also be solved by the ellipsoid method, yet no practically efficient methods are known for this problem. Hence finding a combinatorial algorithm is also a matter of running time efficiency. Shigeno [2] gave the first combinatorial algorithm that runs in polynomial time for some restricted classes of functions Γ_e, including piecewise linear. It is also an extension of the Fat-Path algorithm in [3]. In spite of this development, it has remained an open problem to find a combinatorial polynomial-time algorithm for arbitrary concave increasing gain functions.

Our result settles this question by allowing arbitrary increasing concave gain functions provided via value oracle access. The running time bounds for this general problem are reasonably close to the most efficient linear generalized flow algorithms. Concave gain functions extend the applicability range of the classical generalized flow model, as they can describe e.g. deminishing marginal utilities. We show that the model is a general framework containing multiple convex programs for market equilibrium settings, for which combinatorial algorithms have been developed over the last decade. As an application, we get a combinatorial algorithm for nonsymmetric Arrow-Debreu Nash bargaining, resolving an open question by Vazirani [4]. We can also extend existing results to more general settings.

The concave optimization problem might have irrational optimal solutions: in general, we give a fully polynomial-time approximation scheme, with running time dependent on $\log(\frac{1}{\varepsilon})$ for finding an ε-approximate solution. In the market

This work was done in the College of Computing, Georgia Institute of Technology, supported by NSF Grant CCF-0914732.

DOI 10.1109/FOCS.2012.33

equilibrium applications we have rational convex programs (as in [4]): the existence of a rational optimal solution is guaranteed. We show a general technique to transform a sufficiently good approximation delivered by our algorithm to an exact optimal solution under certain circumstances.

The rest of the paper is organized as follows. In Section II, we give the precise definitions of the problems considered. Thereby we introduce a new, equivalent variant of the problem, called the *symmetric formulation*, providing a more flexible algorithmic framework. Section III shows the applications for market equilibrium problems. Section IV explores the background of minimum-cost circulation and generalized flow algorithms. Section V gives the algorithm for symmetric concave generalized flows. Section VI shows how the algorithm can be applied for the more standard sink formulation. Section VII describes a general method for finding the optimal solutions for rational convex programs, in particular, to the nonsymmetric Arrow-Debreu Nash bargaining problem.

II. Problem definitions

We define two closely related variants of the concave generalized flow problem. The first is essentially the problem proposed by Truemper [1] and Shigeno [2]. Let $G = (V, E)$ be a directed graph. Let $n = |V|$, $m = |E|$, and for each node $i \in V$, let d_i be the total number of incoming and outgoing arcs incident to i.

We are given lower and upper arc capacities $\ell, u : E \to \mathbb{R}$ and a monotone increasing concave function $\Gamma_{ij} : [\ell_{ij}, u_{ij}] \to \mathbb{R} \cup \{-\infty\}$ on each arc and node demands $b : V \to \mathbb{R}$. By a *pseudoflow* we mean a function $f : E \to \mathbb{R}$ with $\ell \le f \le u$. Given the pseudoflow f, let

$$e_i := \sum_{j:ji\in E} \Gamma_{ji}(f_{ji}) - \sum_{j:ij\in E} f_{ij} - b_i.$$

In the first variant of the problem, called the *sink formulation*, there is a distinguished *sink* node $t \in V$. The pseudoflow f is *feasible*, if $e_i \ge 0$ for all $i \in V - t$ and $e_t > -\infty$. The objective is to maximize e_t for feasible pseudoflows.

Shigeno [2] defines this problem with $e_i = 0$ if $i \in V - t$, and $b \equiv 0$ and without explicit capacity constraints. She also discusses the version with $e_i \ge 0$, and gives a reduction from the original version to this one. Whereas capacity constraints can be simulated by the functions Γ_e, we impose them explicitly as they will be included in the running time bounds. The formulation with $e_i \ge 0$ seems more natural as it gives a convex optimization problem, which is not the case for $e_i = 0$.

In the sink formulation, the node t plays a distinguished role. It turns out to be more convenient to handle all nodes equally. For this reason, we introduce another, seemingly more general version, called the *symmetric formulation*. Ideally, we would like to find a pseudoflow satisfying $e_i \ge 0$ for every $i \in V$. The formulation will be a relaxation of this feasibility problem, allowing violation of the constraints, penalized by possibly different rates at different nodes.

For each node $i \in V$ we are given a penalty factor $M_i > 0$ and an auxiliary variable $\kappa_i \ge 0$. The objective is to minimize $\kappa_f = \sum_{i\in V} M_i\kappa_i$ for a pseudoflow f subject to $e_i + \kappa_i \ge 0$ for each $i \in V$. The objective κ_f is called the *excess discrepancy*. $\kappa_f = 0$ means $e_i \ge 0$ for each $i \in V$. These conditions might be violated, but we have to pay penalty M_i per unit violation at i.

The sink version fits into this framework with $M_i = \infty$ for $i \ne t$ and $M_t = 1$. However, it can be shown that setting finite, polynomially bounded M_i values, the symmetric version returns an optimal (or sufficiently close approximate) solution to the sink version. While the symmetric formulation could seem more general than the sink version, it can indeed be reduced to it. For an instance of the symmetric version with graph $G = (V, E)$, let us add a new node t with an arc from t to every node $i \in V$ with gain function $\Gamma_{ti}(\alpha) = \alpha/M_i$. The reason for introducing the symmetric formulation is its pertinence to our algorithmic purposes.

A. Complexity model

From a complexity perspective, the description of the functions might be infinite. To handle this difficulty, following the approach of Hochbaum and Shantikumar [25], we assume oracle access to the Γ_{ij}'s: our running time estimation will give a bound on the number of necessary oracle calls. Two kinds of oracles are needed: *(i)* value oracle, returning $\Gamma_{ij}(\alpha)$ for any $\alpha \in [\ell_{ij}, u_{ij}]$; and *(ii)* inverse value oracle, returning a value β with $\alpha = \Gamma_{ij}(\beta)$ for any $\alpha \in [\Gamma_{ij}(\ell_{ij}), \Gamma_{ij}(u_{ij})]$.

We assume that both oracles return the exact (possibly irrational) solution, and any oracle query is done in $O(1)$ time. Also, we assume any basic arithmetic operation is performed in $O(1)$ time, regardless to size and representation of the possibly irrational numbers. We expect that our results naturally extend to the setting with only approximate oracles and computational capacities in a straightforward manner. Notice that in an approximate sense, an inverse value oracle can be simulated by a value oracle.

By an ε*-approximate solution* to the symmetric concave generalized flow problem we mean a feasible solution with the excess discrepancy larger than the optimum by at most ε. An ε*-approximate solution* to the sink version means a pseudoflow with the objective value e_t at most ε less than the optimum, and the total violation of the inequalities $e_i \ge 0$ for $i \in V - t$ is also at most ε. (Note that an ε-approximate solution is thus not necessarily feasible.)

Let us assume that all M_i values are positive integers, and let M denote their maximum.

In the complexity estimation, we will have U as an upper bound on the absolute values on the b_i's, the capacities ℓ_{ij}, u_{ij} and the $\Gamma_{ij}(\ell_{ij})$, $\Gamma_{ij}(u_{ij})$ values. For each arc ij, let us define $r_{ij} = |\Gamma_{ij}(\ell_{ij})|$ whenever $\Gamma_{ij}(\ell_{ij}) > -\infty$ and $r_{ij} = 0$ otherwise. Let

$$U = \max\{\,\max\{|b_i| : i \in V\},\\ \max\{|\ell_{ij}|, |u_{ij}|, |\Gamma_{ij}(u_{ij})|, r_{ij} : ij \in E\}\}.$$

For the sink version, we need to introduce one further complexity parameter U^* due to difficulties arising if $\Gamma_{ij}(\ell_{ij}) = -\infty$ for certain arcs. Such arcs do appear in the market applications where we have logarithmic gain functions. Let U^* satisfy $U \le U^*$, and that $e_t \le U^*$ for any pseudoflow (it is easy to see that $U^* = d_t U$ always satisfies this property). We also require that whenever there exists a feasible solution to the problem (that is, $e_i \ge 0$ for each $i \in V - t$ and $e_t > -\infty$), there exists one with $e_t \ge -U^*$. If $\Gamma_{jt}(\ell_{jt}) > -\infty$ for each arc $jt \in E$, then $U^* = d_t U$ satisfies this property as well.

The main result is as follows:

Theorem 1. *For the symmetric formulation of the concave generalized flow problem, there exists a combinatorial algorithm that finds an ε-approximate solution in $O(m(m+n\log n)\log(MUm/\varepsilon))$. For the sink formulation, there exists a combinatorial algorithm that finds an ε-approximate solution in $O(m(m+n\log n)\log(U^*m/\varepsilon))$. In both cases, the running time bound is on the number of arithmetic operations and oracle queries.*

For linear generalized flows, we are interested in finding exact solutions and therefore we use a different complexity model. We assume all ℓ, u and b are given as integers and γ as rational numbers; let B be the largest integer used in their descriptions. We obtain a running time bound $O(m^2(m\log B + \log M)\log n)$ for the symmetric and $O(m^2(m+n\log n)\log B)$ for the sink formulation (see the full version). This is the same as the complexity bound of the highest gain augmenting path algorithm [14]. The best current running time bounds are $O(m^{1.5}n^2\log B)$ using an interior point approach [9], and $\tilde{O}(m^2 n\log B)$ [20], an enhanced version of [14].

The starting point of our investigation is the FAT-PATH algorithm [3]. The first important idea is using the symmetric formulation. This is a more flexible framework, and thus we will be able to entirely avoid cycle cancellation and use excess transportation phases only. Our result gives the first generalized flow algorithm that uses a pure scaling technique, without any cycle cancellation. The key new idea here is the way 'Δ-positive' and 'Δ-negative' nodes are defined, maintaining a 'security reserve' in each node that compensates for adjustments when moving from the Δ-scaling phase to the $\Delta/2$-phase.

We extend the linear algorithm to the concave setting using a local linear approximation of the gain functions, following Shigeno [2]. This approximation is motivated by the technique of Minoux [26] and Hochbaum and Shantikumar [25] for minimum cost flows with separable convex objectives.

III. Applications to market equilibrium problems

Intensive research has been pursued over the last decade to develop polynomial-time combinatorial algorithms for certain market equilibrium problems. The starting point is the algorithm for computing market clearing prices in Fisher's model with linear utilities by Devanur et al. [27], followed by a study of several variations and extensions of this model. For a survey, see [28, Chapter 5] or [4].

In the *linear Fisher market model*, we are given a set B of buyers and a set G of goods. Buyer i has a budget m_i, and there is one divisible unit of each good to be sold. For each buyer $i \in B$ and good $j \in G$, $U_{ij} \ge 0$ is the utility accrued by buyer i for one unit of good j. Let $n = |B| + |G|$ and m be the number of pairs ij with $U_{ij} > 0$. We assume there is such an edge incident to every buyer and to every good. Let $U_{\max} = \max\{U_{ij} : i \in B, j \in G\}$ and $R = \max\{m_i : i \in B\}$. An equilibrium solution consist of prices p_i on the goods and an allocation x_{ij}, so that *(i)* all goods are sold, *(ii)* all money of the buyers is spent, and *(iii)* each buyers i buys a best bundle of goods, that is, goods j maximizing U_{ij}/p_j.

The equilibrium solutions for linear Fisher markets were described via a convex program by Eisenberg and Gale [29] in 1959; the combinatorial algorithms for this problem and other models rely on the KKT-conditions for the corresponding convex programs. Exact optimal solutions can be found, since these problems admit rational optimal solutions.

$$\begin{aligned} &\max \sum_{i\in B} m_i \log z_i \\ &z_i \le \sum_{j\in G} U_{ij} x_{ij} \quad \forall i \in B \\ &\sum_{i\in B} x_{ij} \le 1 \quad \forall j \in G \\ &z, x \ge 0 \end{aligned} \qquad \text{(EG)}$$

We show that the Eisenberg-Gale convex program, along with all extensions studied so far, falls into the broader class of concave generalized flows. Moreover, in all these extension we may replace linear or piecewise linear concave functions by arbitrary concave ones, still solvable approximately by our algorithm.

For the Eisenberg-Gale program, let us define the graph (V, E) with $V = B \cup G \cup \{t\}$. Let $ji \in E$ whenever $j \in G$, $i \in B$, $U_{ij} > 0$, and set $\Gamma_{ji}(\alpha) = U_{ij}\alpha$ as a linear gain function. Also, let $it \in E$ for every $i \in B$ with $\Gamma_{it}(\alpha) = m_i \log \alpha$. Finally, set $b_j = -1$ for $j \in G$, and $b_i = 0$ for $i \in B$. The above program describes exactly the sink version of this concave generalized flow instance with $f_{ji} = x_{ij}$ for

$i \in B$, $j \in G$ and $f_{it} = z_i$. (To formally fit into the model, we may add upper capacities $u_{ji} = 1$ and $u_{it} = \sum_{j \in G} U_{ji}$ without changing the set of feasible solutions.) Our general algorithm gives an ε-approximation for this problem. For a sufficiently small ε, this can be transformed to an exact optimal solution.

The flexibility of the concave generalized flow model enables various extensions. For example, we can replace each linear function $U_{ji}\alpha$ by an arbitrary concave increasing function, obtaining the perfect price discrimination model of Goel and Vazirani [30]. They studied piecewise linear utility functions; our model enables arbitrary functions (although the optimal solution may be irrational).

In the *Arrow-Debreu Nash bargaining (ADNB)* defined by Vazirani [4], traders arrive to the market with initial endowments of goods, giving utility c_i for player i. They want to redistribute the goods to obtain higher utilities using Nash bargaining. The disagreement point is when everyone keeps the initial endowment, guaranteeing her $c_i \geq 0$ utility. In an optimal Nash bargaining solution we maximize $\sum_{i \in B} \log(z_i - c_i)$ over the constraint set in (EG). Unlike for the linear Fisher model, equilibrium prices may not exist, corresponding to a disagreement solution. A sophisticated two phase algorithm is given in [4], first for deciding feasibility, then for finding the equilibrium solution.

The convex program for nonsymmetric ADNB can be obtained from the Eisenberg-Gale program by modifying the first set of inequalities to $z_i \leq \sum_{j \in G} U_{ij} x_{ij} - c_i$. In the formulation as a concave generalized flow, this corresponds to modifying the $b_i = 0$ values for $i \in B$ to $b_i = c_i$. Hence this problem also fits into our framework. From this general perspective, it does not seem more difficult than the linear Fisher model.

Nonsymmetric Nash-bargaining was defined by Kalai [31]. For ADNB, it corresponds to maximizing $\sum_{i \in B} m_i \log(z_i - c_i)$ over the constraint set in (EG), for some positive coefficients m_i. The algorithm in [4] heavily relies on the assumption $m_i = 1$, and does not extend to this more general setting, called *nonsymmetric ADNB*. Finding a combinatorial algorithm for this latter problem was left open in [4]. Another open question in [4] is to devise a combinatorial algorithm for (nonsymmetric) ADNB with piecewise linear, concave utility functions. Our result generalizes even further, for arbitrary concave utility functions, since the linear functions $U_{ij}\alpha$ can be replaced by arbitrary concave functions.

Let $C = \max c_i$. Our algorithm can be used to find an exact solution to the nonsymmetric ADNB problem in time $O(m(m + n \log n)(n \log(nU_{\max} R) + \log C))$. The running time bound in [4] for symmetric ADNB ($R = 1$) is $O(n^8 \log U_{\max} + n^4 \log C)$.

Let us also remark that an alternative convex program for the linear Fisher market, given by Shmyrev [32], shows that it also fits into the framework of minimum-cost circulations with a separable convex cost function, and thus can be solved by the algorithms of Hochbaum and Shantikumar [25] or Karzanov and McCormick [33]. Recently, [34] gave a strongly polynomial algorithm for a class of these problems, which includes Fisher's market with linear and with spending constraint utilities. However, this does not seem to capture perfect price discrimination or ADNB, where no alternative formulations analogous to [32] are known.

As further applications of the concave generalized flow model, we can take single-source multiple-sink markets by Jain and Vazirani [35], or concave cost matchings studied by Devanur and Jain [36].

A distinct characteristic of the Eisenberg-Gale program and its extensions is that they are rational convex programs. We may loose this property when changing to general concave spending constraint utilities. However, for the case when the existence of a rational solution is guaranteed, one would prefer finding an exact optimal solution. Section VII addresses the question of rationality. Theorem 12 shows that under certain technical conditions, our approximation algorithm can be turned into a polynomial time algorithm for finding an exact optimal solution.

IV. Previous algorithms for flow problems

We refer the reader to [8] for a background on flow problems; the full version also gives a more detailed overview. The fundamental problem for our investigations is the **minimum-cost circulation** problem.[1] Algorithms are built on two main algorithmic paradigms: *cycle cancelling* (see e.g. [8, Chapter 9.6]) and *successive shortest paths* (see e.g. [8, Chapter 9.7]). Neither of these basic algorithms are polynomial, but both can be modified to run in strongly polynomial time (e.g. [37]–[39]). For the successive shortest path framework, the first (weakly) polynomial running time was obtained by the scaling method of Edmonds and Karp [40]. This serves as a starting point for a significant part of algorithms for various flow models, including our concave generalized flow algorithm.

Generalized flow algorithms are based on methods for minimum-cost circulations. The key notion is *flow generating cycle*, where the product of the gain factors γ_e is greater than 1. This corresponds to a negative cycle with respect to the cost $c_e = -\log \gamma_e$. A solution is optimal, if there exists no flow generating cycle in the residual graph, connected to the sink by a path. We may cancel all flow generating cycles by directly adapting algorithms for minimum-cost circulations. Onaga [41] showed that if after cancelling all flow generating cycles, we only use highest gain augmenting paths for excess transportation, no new flow generating cycle is created. This is analogous to the successive shortest

[1] We shall use the term 'circulation' to distinguish form other flow problems in the paper.

paths algorithm and is also not polynomial. The FAT-PATH algorithm by Goldberg, Plotkin and Tardos [3] uses a method analogous to the Edmonds-Karp capacity scaling. In the Δ-phase, instead of using highest gain paths, Δ-fat paths are used, that are able to transport Δ-units of excess. This may create new flow generating cycles, which should be canceled in the next phase.

The basic framework of [41] and of FAT-PATH, namely using the different paradigms for eliminating flow-generating cycles and for transporting excess to the sink has been adopted by most subsequent algorithms, e.g. [13]–[16], [20].

Minimum-cost circulations with separable convex costs. A natural and well-studied nonlinear extension of minimum-cost circulations is replacing each arc cost c_e by a convex function C_e. This is a widely applicable framework, see [8, Chapter 14]. Both the minimum mean cycle cancellation and the capacity scaling algorithms can be naturally extended to this problem with polynomial running time bounds: cycle cancellation was adapted by Karzanov and McCormick [33], while capacity scaling by Minoux [26] and by Hochbaum and Shanthikumar [25]. The two frameworks are based on fundamentally different relaxations of the KKT-conditions. [33] directly uses the (right) derivative values of the C_e's, while [26] and [25] use a gradually refined linear approximation.

Concave generalized flows. Shigeno's [2] approach was to extend the FAT-PATH algorithm of [3] to the concave setting. However, [2] obtains polynomial running time bounds only for restricted classes of gain functions. The algorithm consists of two procedures applied alternately, similarly to FAT-PATH: a cycle cancellation phase to generate excess on cycles with positive gains, and a path augmentation phase to transport new excess to the sink in chunks of Δ. For both phases, previous methods naturally extend: cycle cancelling is performed analogously to [33], whereas path augmentation to [25]. Unfortunately, this yields polynomial running time only under certain restrictions. The main reason for this is that the different relaxations cannot be fit smoothly into a unified framework.

V. CONCAVE GENERALIZED FLOWS ALGORITHM

Our algorithm for the case of linear gains does excess transportation similarly to FAT-PATH, however, the cycle-cancelling steps are completely eliminated and we use a purely scaling framework. The successive shortest paths algorithms for minimum cost circulations start with an infeasible pseudoflow, having both positive and negative nodes. To use an analogous method for generalized flows, we have to give up the standard framework of algorithms where $e_i \geq 0$ is always maintained for all $i \in V - t$. This is the reason why we use the more flexible symmetric model: we start with possibly several nodes having $e_i < 0$, and our aim is to eliminate them. An important property of the algorithm is that we always have to maintain $\mu_i = 1/M_i$ for $e_i < 0$; for this reason we shall avoid creating new negative nodes.

Similarly to FAT-PATH, we use a scaling algorithm. In the Δ-phase, we consider the residual graph restricted to Δ-fat arcs, arcs that may participate in a highest gain Δ-fat-path, and maintain a conservative labeling μ with $\gamma^{\mu}_{ij} \leq 1$ on Δ-fat arcs. When moving to the $\Delta/2$-phase, this condition may get violated due to $\Delta/2$-fat arcs that were not Δ-fat. Analogously to the Edmonds-Karp algorithm, we modify the flow by saturating each violated arc and thereby restitute dual feasibility. However, these changes may create new negative nodes and thus violate the condition $\mu_i = 1/M_i$ for $e_i < 0$.

We resolve this difficulty by maintaing a 'security reserve' of $d_i\Delta\mu_i$ in each node i (d_i is the number of incident arcs). This gives an upper bound on the total change caused by restoring feasibility of incident arcs in all subsequent phases. We call a node Δ-positive if $e_i > d_i\Delta\mu_i$, Δ-negative if $e_i < d_i\Delta\mu_i$ and Δ-neutral if $e_i = d_i\Delta\mu_i$. Δ-negative nodes may become negative ($e_i < 0$) at a later phase, and therefore we maintain $\mu_i = 1/M_i$ for them. We send flow from Δ-positive nodes to Δ-negative and Δ-neutral ones. Thereby we treat some nodes with $e_i > 0$ as sinks and increase their excess further; however, as Δ decreases, such nodes may gradually become sources.

This linear generalized flow algorithm smoothly extends to concave generalized flows. We use the local linearization $\theta^{\mu}_{\Delta}(ij)$ of Γ_{ij} used by Shigeno, analogously to [25]. In the Δ-phase, we consider the graph of Δ-fat arcs, and maintain $\theta^{\mu}_{\Delta}(ij) \leq 1$ on them.

When moving from a Δ-phase to a $\Delta/2$-phase in the linear algorithm, the only reason for infeasibility is due to $\Delta/2$-fat arcs that were not Δ-fat. In contrast, feasibility can be violated on Δ-fat arcs as well, as $\theta^{\mu}_{\Delta}(ij) \leq 1 < \theta^{\mu}_{\Delta/2}(ij)$ may happen due to the finer linear approximation of the gain functions in the $\Delta/2$-phase. Fortunately, feasibility can be restored in this case as well, by changing the flow on each arc by a small amount.

A. Optimality conditions

The characterization of optimality was given in [2]; we have to modify them slightly as we use the symmetric formulation. The problem can be easily transformed to an equivalent instance with *(i)* $\ell \equiv 0$ and $\Gamma_{ij}(0) = 0$ for every arc with $\Gamma_{ij}(0) > -\infty$; and *(ii)* every gain function Γ_{ij} is strictly monotone increasing on $[0, u_{ij}]$. We shall assume these properties in the sequel.

The concavity of Γ_{ij} implies that for each $0 \leq \alpha$, there exists the right derivative, denoted by $\Gamma^{+}_{ij}(\alpha)$, and for $0 < \alpha$, there exists the left derivative $\Gamma^{-}_{ij}(\alpha)$. If $0 \leq \alpha < \alpha'$, then $\Gamma^{+}_{ij}(\alpha') \leq \Gamma^{-}_{ij}(\alpha') \leq \Gamma^{+}_{ij}(\alpha) \leq \Gamma^{-}_{ij}(\alpha)$.

For a pseudoflow $f : E \rightarrow \mathbb{R}$, we define the residual network by $ij \in E_f$ if $ij \in E$ or $ji \in E$ and $f_{ji} > 0$.

For notational convenience, we define $f_{ji} = -\Gamma_{ij}(f_{ij})$ on backward arcs. We also define the function $\Gamma_{ji}(\alpha) : [-\Gamma_{ij}(u_{ij}), \Gamma_{ij}(0)] \to [-u_{ij}, 0]$ by $\Gamma_{ji}(\alpha) = -\Gamma_{ij}^{-1}(-\alpha)$. Hence $\Gamma_{ji}(f_{ji}) = -f_{ij}$.

In the concave setting, we call a cycle C in E_f a *flow generating cycle*, if $\Gamma^+(C) = \Pi_{ij\in C}\Gamma^+_{ij}(f_{ij}) > 1$. For such a C, it can be shown that positive flow can be generated in any node of C by sending flow around the cycle. The pair (C, P) is called a *generalized augmenting path (GAP)* in the following cases: **(a)** C is a flow generating cycle, $i \in V(C)$, $t \in V$ is a node with $e_t < 0$, and P is a path in E_f from i to t ($i = t$, $P = \emptyset$ is possible); **(b)** $C = \emptyset$, and P is a path in E_f between two nodes s and t with $e_s > 0$, $e_t < 0$; **(c)** $C = \emptyset$, and P is a path in E_f between s and t with $e_s \le 0$, $e_t < 0$ and $\Gamma_f^+(P) > M_s/M_t$. It is easy to verify the following.

Lemma 2. *If f is an optimal solution, then no GAP exists.*

Relabeling is a standard technique for generalized flows. Given $\mu : V \to \mathbb{R}_{>0} \cup \{\infty\}$, let us define $f^\mu_{ij} = f_{ij}/\mu_i$ for each arc $ij \in E$. We get problems equivalent to the original with relabeled functions $\Gamma^\mu_{ij}(\alpha) = \Gamma_{ij}(\mu_i\alpha)/\mu_j$. Accordingly, the relabeled demands, excesses, and capacities are $b^\mu_i = b_i/\mu_i$, $e^\mu_i = e_i/\mu_i$, and $u^\mu_{ij} = u_{ij}/\mu_i$. A relabeling is *conservative*, if for any residual arc $ij \in E_f$, $\Gamma^{\mu+}_{ij}(f^\mu_{ij}) \le 1$, that is, no edge may increase the relabeled flow. Furthermore we require $\mu_i \ge 1/M_i$ for every $i \in V$ and equality whenever $e_i < 0$.

If $\mu_i = \infty$, we define $b^\mu_i = e^\mu_i = 0$, $u^\mu_{ij} = 0$ for $ij \in E$, and furthermore $\Gamma^{\mu+}_{ji}(f^\mu_{ji}) = 0$ for all arcs $ji \in E_f$. Finally, if $ij \in E_f$ with $\mu_i = \infty$, $\mu_j < \infty$, then $\Gamma^{\mu+}_{ij}(f^\mu_{ij}) = \infty$. The following theorem can be derived using the Karush-Kuhn-Tucker conditions.

Theorem 3 ([2]). *Let $f \in \mathbb{R}^E$ satisfy $0 \le f \le \tilde{u}$. Then the following are equivalent. **(i)** f is an optimal solution to the symmetric version. **(ii)** E_f contains no generalized augmenting paths. **(iii)** There exists a conservative labeling μ with $e_i = 0$ whenever $1/M_i < \mu_i < \infty$.* □

B. Δ-conservative and Δ-canonical labelings

Let us define the *fatness* of $ij \in E_f$ by $s_f(ij) = \Gamma_{ij}(u_{ij}) - \Gamma_{ij}(f_{ij})$ (if ij is a backward arc, this is equivalent to $s_f(ij) = f_{ji}$.) The fatness expresses the maximum possible flow increase in j if saturating ij. This notion enables us to identify arcs that can participite in fat paths during the algorithm. In accordance with the other variables, the relabeled fatness is defined as $s^\mu_f(ij) = s_f(ij)/\mu_j$.

Consider a scaling parameter $\Delta > 0$. The *Δ-fat graph* $E^\mu_f(\Delta)$ is the set of residual arcs of relabeled fatness at least Δ:

$$E^\mu_f(\Delta) = \{ij : ij \in E_f : s^\mu_f(ij) \ge \Delta\}.$$

Arcs in $E^\mu_f(\Delta)$ will be called *Δ-fat arcs*. As in [2], we use the following linearization on Δ-fat arcs in chunks of Δ.

$$\theta^\mu_\Delta(ij) := \frac{\Delta\mu_i}{\Gamma^{-1}_{ij}(\Gamma_{ij}(f_{ij}) + \Delta\mu_j) - f_{ij}} \quad ij \in E^\mu_f(\Delta). \quad (1)$$

This is well-defined since $\Gamma_{ij}(f_{ij}) + \Delta\mu_j \le \Gamma_{ij}(u_{ij})$ for Δ-fat arcs. Note that if $\Gamma_{ij}(.)$ is linear, i.e. $\Gamma_{ij}(\alpha) = \gamma_{ij}\alpha$, then $\theta^\mu_\Delta(ij) = \gamma_{ij}$. Also, if the reverse arc ji Δ-fat, than using $\Gamma_{ji}(f_{ji}) = -f_{ij}$ and $\Gamma^{-1}_{ji} = -\Gamma_{ij}(-\alpha)$, we get

$$\theta^\mu_\Delta(ji) = \frac{\Delta\mu_j}{\Gamma_{ij}(f_{ij}) - \Gamma_{ij}(f_{ij} - \Delta\mu_i)}. \quad (2)$$

Consider a label function $\mu : V \to \mathbb{R}_{>0} \cup \{\infty\}$; recall that d_i is the total number of arcs incident to i. A node $i \in V$ is called *Δ-negative* if $e^\mu_i < d_i\Delta$, *Δ-neutral* if $e^\mu_i = d_i\Delta$ and *Δ-positive* if $e^\mu_i > d_i\Delta$. The labeling μ *Δ-conservative*, if $\theta^\mu_\Delta(ij) \le 1$ holds for every $ij \in E^\mu_f(\Delta)$. Furthermore, we require $\mu_i \ge 1/M_i$ for all $i \in V$, with equality for every Δ-negative node i.

Note that a Δ-conservative labeling cannot have any nodes with $\mu_i = \infty$. Using the convexity of Γ^{-1}, it can be shown that if μ is a Δ-conservative labeling then it is Δ′-conservative for all $\Delta' \ge \Delta$. Let $Ex^\mu(f) = \sum_{i\in V}\max\{e^\mu_i, 0\}$ and $Ex^\mu_\Delta(f) = \sum_{i\in V}\max\{e^\mu_i - d_i\Delta, 0\}$ denote the total relabeled excess of positive and Δ-positive nodes, respecitively.

The key importance of Δ-conservativity is that it is maintained when sending Δ units of flow on arcs with $\theta^\mu_\Delta(ij) = 1$. This is formulated in the next simple lemma.

Lemma 4. *Assume μ is Δ-conservative, and let $ij \in E^\mu_f(\Delta)$ be an arc with $\theta^\mu_\Delta(ij) = 1$. If we increase f^μ_{ij} by Δ, then $\Gamma(f^\mu_{ij})$ also increases by Δ, and Δ-conservativity is maintained.*

Proof: $\theta^\mu_\Delta(ij) = 1$ is equivalent to $\Gamma^\mu_{ij}(f^\mu_{ij} + \Delta) = \Gamma^\mu_j(f^\mu_{ij}) + \Delta$, showing the first part. Let $\bar{f}_{ij} = f_{ij} + \Delta\mu_i$ be the modified flow. For the second part, $\theta^\mu_\Delta(ij) \le 1$ for $\bar{f}_{ij}$ easily follows from convexity. Further, observe (2) shows that we get $\theta^\mu_\Delta(ji) = 1$ for $\bar{f}_{ij}$. This gives Δ-conservativity for the modified flow as all other arcs are left unchanged. ■

The next lemma shows how a Δ-conservative labeling can be transformed to a Δ/2-conservative one. Analogous claims are proved in [26] and [25].

Lemma 5. *Let f be a pseudoflow with a Δ-conservative labeling μ. Then there exists a flow $\bar{f}$ such that μ is Δ/2-conservative for $\bar{f}$ and $Ex^\mu_{\Delta/2}(\bar{f}) \le Ex^\mu_\Delta(f) + \frac{3}{2}m\Delta$.*

Proof: Consider a Δ/2-fat arc ij with $\theta^\mu_{\Delta/2}(ij) > 1$ for f, that is,

$$\Gamma^{-1}_{ij}\left(\Gamma_{ij}(f_{ij}) + \frac{\Delta}{2}\mu_j\right) - f_{ij} < \frac{\Delta}{2}\mu_i. \quad (3)$$

There are two possible scenarios: *(a)* ij was not Δ-fat, that is,

$$\frac{\Delta}{2}\mu_j \le \Gamma_{ij}(u_{ij}) - \Gamma_{ij}(f_{ij}) < \Delta\mu_j, \tag{4}$$

or *(b)* ij was also a Δ-fat arc. Then by Δ-conservativity,

$$\Gamma_{ij}^{-1}(\Gamma_{ij}(f_{ij}) + \Delta\mu_j) - f_{ij} \ge \Delta\mu_i. \tag{5}$$

In both cases, let us define

$$\bar{f}_{ij} = \Gamma_{ij}^{-1}\left(\Gamma_{ij}(f_{ij}) + \frac{\Delta}{2}\mu_j\right).$$

$\Delta/2$-fatness of ij guarantees that this is well-defined. In case *(a)*, we claim that ij is not $\Delta/2$-fat for $\bar{f}$. Indeed,

$$\Gamma_{ij}(u_{ij}) - \Gamma_{ij}(\bar{f}_{ij}) = \Gamma_{ij}(u_{ij}) - \left(\Gamma_{ij}(f_{ij}) + \frac{\Delta}{2}\mu_j\right) < \frac{\Delta}{2}\mu_j.$$

The last inequality follows by the second part of (4). In case *(b)*, we claim that if ij is a $\Delta/2$-fat arc for $\bar{f}$ then $\theta^{\mu}_{\Delta/2}(ij) \le 1$ must hold for $\bar{f}$. Indeed, if we subtract (3) from (5), we get

$$\Gamma_{ij}^{-1}\left(\Gamma_{ij}(f_{ij}) + \Delta\mu_j\right) - \Gamma_{ij}^{-1}\left(\Gamma_{ij}(f_{ij}) + \frac{\Delta}{2}\mu_j\right) > \frac{\Delta}{2}\mu_i,$$

and by substituting $\bar{f}_{ij}$, it follows that $\theta^{\mu}_{\Delta/2}(ij) < 1$ for $\bar{f}$.

If if ji is also a $\Delta/2$-fat arc for $\bar{f}$, then $\theta^{\mu}_{\Delta/2}(ji) \le 1$ holds for $\bar{f}$. This can be derived using (2) and (3).

We define $\bar{f}_{ij}$ the above way whenever ij is a $\Delta/2$-fat arc with $\theta_{\mu}(ij) > 1$. (As a simple consequence of concavity, this cannot be the case for both ij and ji.) If this does not hold for neither ij nor ji, then let $\bar{f}_{ij} = f_{ij}$. The next simple claim compares f_{ij} and $\Gamma(f_{ij})$ to $\bar{f}_{ij}$ and $\Gamma(\bar{f}_{ij})$.

Claim 6. $|\bar{f}^{\mu}_{ij} - f^{\mu}_{ij}| \le \frac{\Delta}{2}$ *and* $|\Gamma^{\mu}_{ij}(\bar{f}^{\mu}_{ij}) - \Gamma^{\mu}_{ij}(f^{\mu}_{ij})| \le \frac{\Delta}{2}$.

For $\Delta/2$-conservativity, we also need to show that $\bar{f}$ has no $\Delta/2$-negative nodes with $\mu_i > 1/M_i$. By the above claim, the total possible change of relabeled flow on arcs incident to i is $d_i\Delta/2$. A node is nonnegative for Δ if $e^{\mu}_i \ge d_i\Delta$ and for $\Delta/2$ if $e^{\mu}_i \ge d_i\Delta/2$. Consequently, a Δ-nonnegative node cannot become $\Delta/2$-negative.

Finally, $Ex^{\mu}_{\Delta/2}(f) \le Ex^{\mu}_{\Delta}(f) + \sum_{i\in V} d_i\Delta/2$, and each arc is responsible for creating at most $\Delta/2$ units of new excess. This gives $Ex^{\mu}_{\Delta/2}(\bar{f}) \le Ex^{\mu}_{\Delta}(f) + \frac{m}{2}\Delta$, as required. ■

The subroutine ADJUST(Δ) performs the simple modifications described in the proof

Given a pseudoflow f and a Δ-conservative labeling μ, the arc $ij \in E^{\mu}_f(\Delta)$ is called *tight* if $\theta^{\mu}_{\Delta}(ij) = 1$. A directed path in $E^{\mu}_f(\Delta)$ is called tight if it consists of tight arcs. μ is a Δ-*canonical* labeling, if from each node i there exists a tight path to a Δ-negative or to a Δ-neutral node. Such a path is approximately a highest gain Δ-fat augmenting path. The subroutine TIGHTEN-LABEL(f, μ, Δ) returns a Δ-canonical label $\mu' \ge \mu$ for a Δ-conservative label μ. This is a multiplicative variant of Dijsktra's algorithm (see the full version for details). In every iteration, let S be the set of nodes from which there exists a tight path to a Δ-negative or Δ-neutral node. We increase μ_i for every node in $V \setminus S$ at the same rate, until either a new arc becomes tight or a Δ-positive node becomes Δ-neutral. In both cases, S is extended.

C. The main algorithm

```
Algorithm SYMMETRIC CONCAVE FAT-PATH
for i ∈ V do μ_i ← 1/M_i;
for ij ∈ E do f_ij ← u_ij;
Δ ← MU + 1;
while (2n + 3m)Δ ≥ ε do
    do
        TIGHTEN-LABEL(f, μ, Δ);
        D ← {i ∈ V : e_i > (d_i + 1)Δ};
        N_0 ← {i ∈ V : e_i ≤ d_iΔ};
        pick s ∈ D, t ∈ N connected by a tight path P;
        send Δ units of flow along P;
    while D ≠ ∅;
    ADJUST(Δ);
    Δ ← Δ/2;
return ε-approximate optimal solution f.
```

Figure 1. The algorithm for symmetric concave generalized flows

Let us initialize $\mu_i = 1/M_i$ for every $i \in V$, and $f_{ij} = u_{ij}$ for every $ij \in E$ and pick $\Delta = MU + 1$.

The algorithm consists of Δ-phases, and terminates with an ε-approximate solution if $(2n + 3m)\Delta < \varepsilon$. During the Δ-phase, we maintain a pseudoflow f and a Δ-conservative labeling μ. The μ_i values may only increase. Let D denote the set of nodes i with $e^{\mu}_i > (d_i+1)\Delta$. The Δ-phase consists of iterations, and terminates whenever D becomes empty. In each iteration, we update μ to a canonical labeling by calling TIGHTEN-LABEL(f, μ, Δ). If $D \ne \emptyset$ still holds, send Δ units of relabeled flow on a tight path from some $s \in D$ to a Δ-neutral or Δ-negative node t.

D. Analysis

Claim 7. *The initial μ is Δ-conservative, and Δ-conservativity is maintained during the entire Δ-phase.*

Proof: Initially, $f \equiv u$ and hence E_f is the set of backward arcs. For an arc $ij \in E$, $s^{\mu}_{\Delta}(ji) = u_{ij}/\mu_i \le MU < \Delta$, and hence $E^{\mu}_f(\Delta) = \emptyset$. Also, $\mu_i = 1/M_i$ holds for every node i. TIGHTEN-LABEL(f, μ, Δ) clearly maintains Δ-conservativity. We use only tight arcs to send flow, and Lemma 4 guarantees that this preserves Δ-conservativity. At the end of the Δ-phase, ADJUST(Δ) transforms f to a $\Delta/2$-conservative pseudoflow. ■

Claim 8. *The* Δ*-phase starts with* $Ex^{\mu}_{\Delta}(f) \le (2n+3m)\Delta$.

Proof: For the initial solution, $Ex^{\mu}_{\Delta}(f) \le M(\sum_{i\in V}|b_i| + mU) \le (m+n)MU$. The claim follows, since $\Delta = MU + 1$. Once we finish all iterations in the Δ-phase, $D = \emptyset$ implies $Ex^{\mu}_{\Delta}(f) \le n\Delta$. By Lemma 5, ADJUST($\Delta$) transfroms f to a $\Delta/2$-conservative solution by increasing the excess by at most $\frac{3}{2}m\Delta$. Hence the $\Delta/2$ phase starts with $Ex^{\mu}_{\Delta}(f) \le (2n+3m)\Delta/2$, proving the claim. ∎

Lemma 9. *A* Δ*-phase consists of at most* $2n+3m$ *iterations.*

Proof: Let $\Psi(i) = \lfloor e^{\mu}_i/\Delta\rfloor - d_i$ if $e^{\mu}_i > (d_i+1)\Delta$ and $\Psi(i) = 0$ otherwise. Consider the potential function $\Psi = \sum_{i\in V}\Psi(i)$. By Claim 8, $\Psi \le 2n+3m$ holds at the beginning. In the relabeling steps, Ψ may only decrease, and in every path augmentation, it decreases by exactly 1. ∎

Recall that $\kappa_f = \sum_{i\in V} M_i\kappa_i = \sum_{i\in V} M_i \min\{-e_i, 0\}$ denotes the excess discrepancy. For a Δ-conservative μ, $M_i\kappa_i = e^{\mu}_i =$ holds for every node i with $e_i < 0$, because of $\mu_i = 1/M_i$. Consequently, κ_f is the total relabeled deficiency of the negative nodes. The next theorem shows that if $\Delta < \varepsilon/(2n+3m)$, then we have an ε-optimal solution at the end of the Δ-phase.

Theorem 10. *At the end of phase* Δ*, the actual* f *is* $(2n+3m)\Delta$*-optimal.*

Proof: Let us keep running the algorithm forever unless it finds a 0-discrepancy solution at some phase. First, consider the case when for some $\Delta' = \Delta/2^k$, we terminate with a 0-discrepancy solution. In all phases between Δ and Δ' , the total decrease of excess discrepancy is bounded by $(2n+3m)(\Delta/2 + \Delta/4 + \ldots + \Delta/2^k) < (2n+3m)\Delta$. Since in the Δ'-phase we have a 0-discrepancy solution, the total discrepancy at the end of the Δ-phase is at most $(2n+3m)\Delta$, proving the theorem.

Assume now the procedure runs forever. For each $i \in V$, κ_i is decreasing and thus converges to some limit κ^*_i. Let $\kappa^* = \sum_{i\in V} M_i\kappa^*_i$. As above, the total decrease of the excess discrepancy after phase Δ is bounded by $(2n+3m)\Delta$, hence $\kappa_f \le \kappa^* + (2n+3m)\Delta$. The proof finishes by constructing an optimal pseudoflow f^* with discrepancy κ^*.

Let $f^{(t)}$ denote the flow at time t, for $\Delta^{(t)} = \Delta_0/2^t$, with labels $\mu^{(t)}_i$. For each node i, $\mu^{(t)}_i$ is increasing; let $\mu^*_i = \lim_{t\to\infty}\mu^{(t)}_i$. For every $ij \in E$, $f^{(t)}_{ij}$ is a bounded sequence $(0 \le f^{(t)}_{ij} \le u_{ij})$. Consequently, we can choose an infinite set $T' \subseteq \mathbb{N}$ so that restricted to $t \in T'$, all sequences $f^{(t)}_{ij}$ converge; let f^*_{ij} denote the limits. We shall prove that f^* is an optimal pseudoflow with optimal labeling μ^*_i, completing the proof.

Let $V_\infty = \{i : \mu^*_i = \infty\}$. It can be shown that $V - V_\infty \ne \emptyset$ as otherwise we would terminate with a 0-discrepancy solution in a finite number of steps.

Let e^*_i denote the excesses of f^*. If $e^*_i < 0$, then clearly $i \in N^*$ and $\mu^*_i = 1/M_i$. If $e^*_i > 0$, we shall prove $\mu^*_i = \infty$. For a contradiction, assume $\mu^*_i < \infty$. Then for sufficiently large $t \in T'$, $(d_i + 2n + 3m)\Delta^{(t)}\mu^{(t)}_i < e^{(t)}_i$ and thus $Ex^{\mu}_{\Delta^{(t)}}(f) > (2n+3m)\Delta^{(t)}$, a contradiction.

We have to prove $\Gamma^{\mu^*+}_{ij}(f^{*\mu^*}_{ij}) \le 1$ whenever $ij \in E_{f^*}$. If $\mu^*_j = \infty$, then $\Gamma^{\mu^*+}_{ij}(f^{*\mu^*}_{ij}) = 0$. If $\mu^*_j < \infty$, then the definition (1) gives

$$1 \ge \theta^{\mu^{(t)}}_{\Delta^{(t)}}(ij) = \frac{\Delta^{(t)}\mu^{(t)}_j}{\Gamma^{-1}_{ij}(\Gamma_{ij}(f^{(t)}_{ij}) + \Delta^{(t)}\mu^{(t)}_j) - f^{(t)}_{ij}} \cdot \frac{\mu^{(t)}_i}{\mu^{(t)}_j}.$$

Then $\Delta^{(t)}\mu^{(t)}_j \to 0$ and hence the first fraction converges to $\Gamma^{+}_{ij}(f^*_{ij}) = 1/\{\Gamma^{-1}_{ij}\}^{+}(\Gamma^{+}_{ij}(f^*_{ij}))$, while the second to μ^*_i/μ^*_j. ∎

Theorem 11. *The algorithm finds an* ε*-approximate solution to the symmetric concave generalized flow problem in* $O(m(m+n\log n)\log(MUm/\varepsilon))$ *oracle calls.*

Proof: The initial value of Δ is $MU+1$, and we terminate if $\Delta < \varepsilon/(2n+3m)$. Hence the total number of scaling phases is $O(\log(MUm/\varepsilon))$. The number of iterations in a phase is $O(m)$ by Lemma 9, and the running time of an iteration is dominated by TIGHTEN-LABEL, a slightly modified version of Dijkstra's algorithm that can be implemented in $O(m+n\log n)$ time using Fibonacci heaps as in [42]. ∎

VI. SINK VERSION OF THE PROBLEM

Let us now show how the algorithm for the symmetric version can be used to solve the sink version. An ε-approximate solution to the sink version means a pseudoflow f with $\sum_{i\in V-t}\max\{0,-e_i\} \le \varepsilon$ and e_t being at least the optimum value minus ε.

Let us set $b_t = U^*+1$, a strict upper bound on $\sum_{j:jt\in E}\Gamma_{jt}(f_{jt}) - \sum_{j:tj\in E} f_{tj}$ (we defined U^* in Section II-A). For every pseudoflow, $e_t < 0$ is guaranteed. Let us set $M_i = \lceil(2U^*+1)/\varepsilon\rceil + 1$ if $i \in V-t$ and $M_t = 1$. Let us run the algorithm for the symmetric formulation to obtain an ε-optimal solution f.

If $\kappa_f > 2U^*+1+\varepsilon$, then no feasible solution may exist. Indeed, by the definition of U^*, if there is a feasible solution f', then there exists one with $e_t \ge -U^*$. If f' is such a feasible solution for the sink formulation, then its excess discrepancy for the symmetric formulation is at most $\kappa_{f'} \le b_i + U^* \le 2U^*+1$, a contradiction as f was ε-optimal for the symmetric formulation.

If $\kappa_f \le 2U^*+1+\varepsilon$, then

$$\sum_{i\in V-t}\max\{0,-e_i\} = \frac{1}{\lceil(2U^*+1)/\varepsilon\rceil+1}\sum_{i\in V-t}M_i\kappa_i \le \frac{\kappa_f}{\lceil(2U^*+1)/\varepsilon\rceil+1} \le \varepsilon.$$

Also κ_t cannot be further than ε from the optimum value of e_t for the sink formulation. Indeed, let f' be the optimal solution to the sink formulation with objective value e'_t. Then $\kappa_{f'} = b_t - e'_t$. The claim follows by

$$b_t - e'_t + \varepsilon = \kappa_{f'} + \varepsilon \geq \kappa_f \geq \kappa_t = b_t - e_t,$$

and thus $e_t \geq e'_t - \varepsilon$. In the first inequality, we use that f is ε-optimal for the sink formulation. This gives a running time bound $O(m(m + n \log n) \log(U^* m/\varepsilon))$.

VII. Finding the optimal solution for rational convex programs

In this section, we give a general theorem which shows how an approximate solution to the sink version can be converted to an exact optimal solution, given that one exists. These properties are not difficult to verify for the linear Fisher market, or the more general nonsymmetric Arrow-Debreu Nash bargaining (ADNB) problem. Unlike the linear Fisher model, ADNB might be infeasible. However, it can be shown that if the problem is infeasible, then for appropriate (polynomially small) ε, the ε-approximate version is also infeasible.

Theorem 12. *Let problem $\mathcal{P}$ be given by the sink formulation with n nodes and m arcs, and complexity parameters U, U^*. Assume $\mathcal{P}$ is guaranteed to have a rational optimal solution, and the following conditions hold for some values ε, T and a function $\tau(n, m, U^*)$.*

- (P1) *Consider the algorithm for the sink version for an ε-approximation. Then either there exists no feasible solution, or $\mu_i \leq T$ holds for any $i \in V$, even if running the algorithm for an arbitrary number of phases.*
- (P2) *A subroutine is provided for finding an optimal solution $\tilde{f}$ in $\tau(n, m, U^*)$ time, if the following assumptions hold. Assume that for each $ij \in E$, we are given an interval $I_{ij} \subseteq [\ell_{ij}, u_{ij}]$ with $|I_{ij}| \leq 2T\varepsilon$, with the guarantee that there exists an optimal solution f^* with $f^*_{ij} \in I_{ij}$ for all $ij \in E$.*

Then there exist an algorithm for finding the exact optimal solution or proving that the problem is infeasible in $O(m(m + n \log n) \log(U^* m/\varepsilon)) + \tau(n, m, U^*)$.

We remark that in *(P2)*, $\tilde{f} = f^*$ is not required. To ensure property *(P2)*, a useful method is to enforce the existence of a unique optimal solution by perturbing the input data, as done by Orlin [43] for linear Fisher markets. If there is a unique rational optimal solution f^* with all entries having denominator at most Q, then setting $2T\varepsilon < 1/Q$ enables us to identify the set of arcs with $f^*_{ij} > 0$. This can be already enough to compute f^* efficiently.

Using the notation of Section III, the following can be verified for the nonsymmetric ADNB problem.

Theorem 13. *Let $K = nRU_{\max}$. Setting $T = U^* = \max\{C, nK \log K\}$, $\varepsilon = 1/(2K^n U^*)$ satisfies the requirements on U^* in Section II-A and (P1) and (P2) in Theorem 12. Our algorithm delivers an optimal solution in running time $O(m(m + n \log n)(n \log(nRU_{\max}) + \log C))$ for nonsymmetric ADNB.*

If we apply this algorithm to linear Fisher markets ($c \equiv 0$), the algorithm runs in a fundamentally different way as [27] or [43]. While both these algorithms increase the prices, ours works the other way around: it starts with the highest possible prices, and decreases them.

VIII. Discussion

We have given the first polynomial time combinatorial algorithms for both the symmetric and the sink formulation of the concave generalized flow problem. Our algorithm is not strongly polynomial. In fact, no such algorithm is known already for the linear case: it is a fundamental open question to find a strongly polynomial algorithm for linear generalized flows. If resolved, a natural question could be to devise a strongly polynomial algorithm for some class of convex generalized flow problems, analogously to the recent result [34], desirably including the market and Nash bargaining applications.

Linear Fisher market is also captured by [34]. A natural question is if there is any direct connection between our model and the convex minimum cost flow model studied in [34]. Despite certain similarities, no reduction is known in any direction. Indeed, no such reduction is known even between the linear special cases, that is, generalized flows and minimum-cost circulations. In fact, the only known market setting captured by both is linear Fisher. Perfect price discrimination and ADNB are not known to be reducible to flows with convex objective. In contrast, spending constraint utilities [44] are not known to be captured by our model, although they are captured by the other.

Acknowledgments

The author is grateful to Vijay Vazirani for several fruitful discussions on market equilibrium problems.

References

[1] K. Truemper, "Optimal flows in nonlinear gain networks," *Networks*, vol. 8, no. 1, pp. 17–36, 1978.

[2] M. Shigeno, "Maximum network flows with concave gains," *Mathematical Programming*, vol. 107, no. 3, pp. 439–459, 2006. [Online]. Available: http://dx.doi.org/10.1007/s10107-005-0608-1

[3] A. V. Goldberg, S. A. Plotkin, and É. Tardos, "Combinatorial algorithms for the generalized circulation problem," *Mathematics of Operations Research*, vol. 16, no. 2, p. 351, 1991.

[4] V. V. Vazirani, "The notion of a rational convex program, and an algorithm for the Arrow-Debreu Nash bargaining game," *Journal of ACM (JACM)*, vol. 59, no. 2, 2012.

[5] L. V. Kantorovich, "Mathematical methods of organizing and planning production," *Publication House of the Leningrad State University*, p. 68, 1939, English translation in *Management Science* 6(4):366-422, 1960.

[6] G. B. Dantzig, *Linear Programming and Extensions*. Princeton University Press, Princeton, 1963.

[7] W. S. Jewell, "Optimal flow through networks," *Operations Research*, vol. 10, pp. 476–499, 1962.

[8] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., feb 1993.

[9] S. Kapoor and P. M. Vaidya, "Speeding up Karmarkar's algorithm for multicommodity flows," *Mathematical Programming*, vol. 73, no. 1, pp. 111–127, 1996.

[10] K. Onaga, "Dynamic programming of optimum flows in lossy communication nets," *IEEE Transactions on Circuit Theory*, vol. 13, pp. 308–327, 1966.

[11] K. Truemper, "On max flows with gains and pure min-cost flows," *SIAM Journal on Applied Mathematics*, vol. 32, no. 2, pp. 450–456, 1977.

[12] E. Cohen and N. Megiddo, "New algorithms for generalized network flows," *Mathematical Programming*, vol. 64, no. 1, pp. 325–336, 1994.

[13] D. Goldfarb and Z. Jin, "A faster combinatorial algorithm for the generalized circulation problem," *Mathematics of Operations Research*, vol. 21, no. 3, pp. 529–539, 1996.

[14] D. Goldfarb, Z. Jin, and J. B. Orlin, "Polynomial-time highest-gain augmenting path algorithms for the generalized circulation problem," *Mathematics of Operations Research*, vol. 22, no. 4, pp. 793–802, 1997.

[15] É. Tardos and K. D. Wayne, "Simple maximum flow algorithms in lossy networks," in *Proceedings of IPCO, Lecture Notes in Computer Science*, vol. 1412, 1998, pp. 310–324.

[16] L. K. Fleischer and K. D. Wayne, "Fast and simple approximation schemes for generalized flow," *Mathematical Programming*, vol. 91, no. 2, pp. 215–238, 2002.

[17] D. Goldfarb, Z. Jin, and Y. Lin, "A polynomial dual simplex algorithm for the generalized circulation problem," *Mathematical Programming*, vol. 91, no. 2, pp. 271–288, 2002.

[18] D. Goldfarb and Y. Lin, "Combinatorial interior point methods for generalized network flow problems," *Mathematical Programming*, vol. 93, no. 2, pp. 227–246, 2002.

[19] K. D. Wayne, "A polynomial combinatorial algorithm for generalized minimum cost flow," *Mathematics of Operations Research*, pp. 445–459, 2002.

[20] T. Radzik, "Improving time bounds on maximum generalised flow computations by contracting the network," *Theoretical Computer Science*, vol. 312, no. 1, pp. 75–97, 2004.

[21] M. Restrepo and D. P. Williamson, "A simple GAP-canceling algorithm for the generalized maximum flow problem," *Mathematical Programming*, vol. 118, no. 1, pp. 47–74, 2009.

[22] M. Shigeno, "A survey of combinatorial maximum flow algorithms on a network with gains," *Journal of the Operations Research Society of Japan*, vol. 47, pp. 244–264, 2004.

[23] D. P. Ahlfeld, J. M. Mulvey, R. S. Dembo, and S. A. Zenios, "Nonlinear programming on generalized networks," *ACM Transactions on Mathematical Software (TOMS)*, vol. 13, no. 4, pp. 350–367, 1987.

[24] D. P. Bertsekas, L. C. Polymenakos, and P. Tseng, "An ε-relaxation method for separable convex cost network flow problems," *SIAM Journal on Optimization*, vol. 7, no. 3, pp. 853–870, 1997.

[25] D. S. Hochbaum and J. G. Shanthikumar, "Convex separable optimization is not much harder than linear optimization," *Journal of the ACM (JACM)*, vol. 37, no. 4, pp. 843–862, 1990.

[26] M. Minoux, "Solving integer minimum cost flows with separable convex cost objective polynomially," *Mathematical Programming Study*, vol. 25, p. 237, 1985.

[27] N. R. Devanur, C. H. Papadimitriou, A. Saberi, and V. V. Vazirani, "Market equilibrium via a primal–dual algorithm for a convex program," *Journal of the ACM (JACM)*, vol. 55, no. 5, p. 22, 2008.

[28] N. Nisan, T. Roughgarden, E. Tardos, and V. Vazirani, *Algorithmic Game Theory*. Cambridge University Press New York, NY, USA, 2007.

[29] E. Eisenberg and D. Gale, "Consensus of subjective probabilities: The pari-mutuel method," *The Annals of Mathematical Statistics*, vol. 30, no. 1, pp. 165–168, 1959.

[30] G. Goel and V. V. Vazirani, "A perfect price discrimination market model with production, and a (rational) convex program for it," *Mathematics of Operations Research*, vol. 36, pp. 762–782, 2011.

[31] E. Kalai, "Nonsymmetric Nash solutions and replications of 2-person bargaining," *International Journal of Game Theory*, vol. 6, no. 3, pp. 129–133, 1977.

[32] V. I. Shmyrev, "An algorithm for finding equilibrium in the linear exchange model with fixed budgets," *Journal of Applied and Industrial Mathematics*, vol. 3, no. 4, pp. 505–518, 2009.

[33] A. V. Karzanov and S. T. McCormick, "Polynomial methods for separable convex optimization in unimodular linear spaces with applications," *SIAM J. Comput.*, vol. 26, no. 4, pp. 1245–1275, 1997.

[34] L. A. Végh, "Strongly polynomial algorithm for a class of minimum-cost flow problems with separable convex objectives," in *Proceedings of STOC*. ACM, 2012, pp. 27–40.

[35] K. Jain and V. V. Vazirani, "Eisenberg-gale markets: Algorithms and game-theoretic properties," *Games and Economic Behavior*, vol. 70, no. 1, pp. 84–106, 2010.

[36] N. Devanur and K. Jain, "Online matching with concave returns," in *Proceedings of the 44th Symposium on Theory of Computing (STOC)*. ACM, 2012, pp. 137–144.

[37] É. Tardos, "A strongly polynomial minimum cost circulation algorithm," *Combinatorica*, vol. 5, no. 3, pp. 247–255, 1985.

[38] A. V. Goldberg and R. E. Tarjan, "Finding minimum-cost circulations by canceling negative cycles," *Journal of the ACM (JACM)*, vol. 36, no. 4, pp. 873–886, 1989.

[39] J. B. Orlin, "A faster strongly polynomial minimum cost flow algorithm," *Operations Research*, vol. 41, no. 2, pp. 338–350, 1993.

[40] J. Edmonds and R. M. Karp, "Theoretical improvements in algorithmic efficiency for network flow problems," *Journal of the ACM (JACM)*, vol. 19, no. 2, pp. 248–264, 1972.

[41] K. Onaga, "Optimum flows in general communication networks," *Journal of the Franklin Institute*, vol. 283, no. 4, pp. 308–327, 1967.

[42] M. L. Fredman and R. E. Tarjan, "Fibonacci heaps and their uses in improved network optimization algorithms," *Journal of the ACM (JACM)*, vol. 34, no. 3, pp. 596–615, 1987.

[43] J. B. Orlin, "Improved algorithms for computing fisher's market clearing prices," in *Proceedings of the 42nd ACM Symposium on Theory of Computing (STOC)*. ACM, 2010, pp. 291–300.

[44] V. V. Vazirani, "Spending constraint utilities with applications to the Adwords market," *Mathematics of Operations Research*, vol. 35, no. 2, pp. 458–478, 2010.