# A polynomial-time algorithm for a class of minimum concave cost flow problems

Qie He*[1], Shabbir Ahmed†[2] and George L. Nemhauser‡[2]

[1]Department of Industrial & Systems Engineering, University of Minnesota
[2]H. Milton Stewart School of Industrial & Systems Engineering, Georgia Institute of Technology

November 17, 2013

**Abstract**

We study the minimum concave cost flow problem over a two-dimensional grid network (CFG), where one dimension represents time ($1 \leq t \leq T$) and the other dimension represents echelons ($1 \leq l \leq L$). The concave function over each arc is given by a value oracle. We give a polynomial-time algorithm for finding the optimal solution when the network has a fixed number of echelons and all sources lie at one echelon. We also give an $O(T^4)$-time algorithm for finding the optimal solution in a capacitated grid network with two echelons and constant capacity on certain arcs. Both algorithms generalize the complexity results for many variants of the lot-sizing problem in terms of cost functions, number of echelons, intermediate demands, backlogging, and production and inventory capacities. We also show that CFG is NP-hard when the number of echelons is an input parameter or upward arcs are present. Our results resolve many of the complexity issues for CFG.

## 1 Introduction

We study the minimum concave cost flow problem over a grid network (CFG). Let $[n] = \{1, 2, \dots, n\}$ for any $n \in \mathbb{N}$. A *grid network* $\mathcal{N} = (V, A, b)$ is a directed acyclic graph with the node set

$$V = \{v_{l,t} | l \in [L], t \in [T]\},$$

the arc set

$$A = \{(v_{l,t}, v_{l,t+1}) | l \in [L], t \in [T-1]\} \cup \{(v_{l,t}, v_{l+1,t}) | t \in [T], l \in [L-1]\}$$

and the supply function $b : V \to \mathbb{R}$ which determines whether each node is a source ($b(v) > 0$), a sink ($b(v) < 0$) or a transshipment node ($b(v) = 0$), as shown in Figure 1. We refer to the two subscripts $l$ and $t$ as the indices of *echelon* and *period*, respectively, so the grid network has $L$ echelons and $T$ periods. Define arcs $(v_{l,t}, v_{l,t+1})$, $(v_{l,t+1}, v_{l,t})$, $(v_{l,t}, v_{l+1,t})$ and $(v_{l+1,t}, v_{l,t})$ to be *forward arc*, *backward arc*, *downward arc* and *upward arc* respectively for any $l$ and $t$. For now we assume that the grid network has only forward arcs and downward arcs and no arc capacities.

*qhe@umn.edu
†sahmed@isye.gatech.edu
‡george.nemhauser@isye.gatech.edu

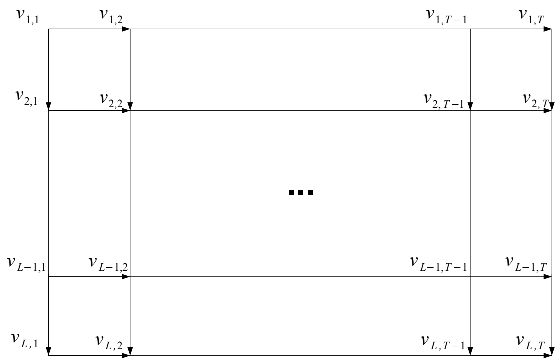



Figure 1: The grid network

Given a grid network $\mathcal{N} = (V, A, b)$, CFG is to find a flow $x \in \mathbb{R}^{|A|}$ to

$$\begin{array}{lll} \text{minimize} & \sum_{a\in A} c_a(x_a) & \\ \text{s.t.} & \sum_{a\in\delta^+(v)} x_a - \sum_{a\in\delta^-(v)} x_a = b(v), & \forall v \in V, \\ & x_a \geq 0, & \forall a \in A, \end{array} \tag{1}$$

where $c_a$ is the cost function for arc $a$, and $\delta^+(v)$ and $\delta^-(v)$ are the set of outgoing and incoming arcs at node $v$, respectively. We assume that the cost function $c_a$ is a general nonnegative concave function represented by a value oracle for each $a \in A$. Since the feasibility of CFG can be checked by solving a maximum flow problem in polynomial time, we assume that the problem is always feasible in the rest of the paper.

The minimum concave cost network flow problem (MCCNFP) is NP-hard in general, as shown by a reduction from the partition problem [7]. Since we are mainly interested in the grid network over which MCCNFP can be efficiently solved, our review is not intended to be exclusive and focuses mainly on polynomially solvable special cases, most of which fall into the category of lot-sizing problems. The classical uncapacitated lot-sizing problem (ULS) is to find an optimal production schedule given a sequence of non-stationary demands over $T$ periods to minimize the total production and inventory holding costs. ULS assumes that the production cost has a fixed-charge structure (which can be seen as a special concave cost function) and the inventory holding cost is linear. ULS was first solved in $O(T^2)$ by dynamic programming (DP) [17], and the complexity was improved to $O(T \ln T)$ later [1, 5, 16]. Variants of ULS have been studied extensively, including problems with more general cost structures for production and inventory holding, multiple echelons of production and demands at intermediate echelons, backlogging, and production capacities and inventory bounds. The common underlying network structure of these variants is a grid network with a single source. The multi-echelon ULS with demands at the last echelon was solved by Zangwill [18] in an $O(LT^4)$-time DP algorithm. The lot-sizing problem with constant production capacity was solved in $O(T^3)$ [6, 14]. Other polynomially solvable cases include ULS with inventory upper bounds and fixed-charge costs [2, 3], a multi-echelon ULS with constant production capacities at the first echelon [15], and a two-echelon ULS with intermediate demands [19]. Pochet and Wolsey [10] provides a detailed study of lot-sizing models that can be solved in polynomial time. Besides the lot-sizing problem, other polynomially solvable cases of MCCNFP include the pure remanufacturing problem [13], which can be formulated as a two-echelon CFG with concave cost on downward arcs, a single-source concave network flow problem with a single nonlinear arc cost [8], the network flow problem with a fixed number of sources and nonlinear arc costs [11], and a production-transportation network flow problem where the concave cost function is defined on only a constant number of variables [12]. The only general network structure we are aware of over which MCCNFP can be solved in polynomial time is a planar graph with sources and sinks lying on a constant number of faces of the graph, studied by Erickson et al. [4]. The grid network is a planar graph, but in general the sources and sinks in CFG are not on a constant number of faces. Recently, He et al. [9] showed that CFG with one echelon of sources and two echelons of sinks can be solved in polynomial time.

Two questions left open in [9] are: (1) Is CFG with one echelon of sources and $L$ echelons of sinks polynomially solvable for general fixed $L$? (2) If there are arc capacities, when can CFG be solved efficiently? In this paper, we answer the first question affirmatively, therefore generalizing the result in [9], and give conditions under which CFG with arc capacities is polynomially solvable. In addition, we present several NP-hard cases for CFG. Our main results can be summarized below.

1. If all sources lie at one echelon and the number of echelons $L$ is fixed, CFG (with backward arcs) can be solved in time polynomial in $T$ and the number of queries of the value oracle.
2. CFG (with backward arcs) can be solved in $O(T^4)$ time for $L = 2$, constant capacity on downward arcs and general capacities on other arcs.
3. If the number of echelons $L$ is an input parameter, CFG is NP-hard.
4. If the grid network contain upward arcs, CFG is NP-hard for fixed $L \geq 3$.

Our polynomial-time algorithm generalizes the complexity results for many variants of lot-sizing problems in terms of arbitrary concave production and inventory holding costs, multiple echelons with intermediate demands, backlogging, and presence of production and inventory capacities. Moreover, we can extend our algorithm to CFG with diagonal downward arcs $(v_{l,t}, v_{l+1,t+1})$, and show that CFG is NP-hard with diagonal upward arcs $(v_{l,t}, v_{l+1,t+1})$ or arcs $(v_{l,t}, v_{l+k,t+1})$ crossing $k$ echelons ($k \geq 2$). Our results resolve many of the complexity issues for CFG. The only remaining open questions are multi-echelon sources for the uncapacitated case and multi-echelon CFG with constant capacity on downward arcs.

To prove the polynomial solvability of different CFG cases, we formulate the problem as a $(T+1)$-stage DP and show that the cardinality of the state space at each stage is polynomial in $T$. The proposed DP formulation is different from that in [9]. The main advantage of the new formulation is that it can easily deal with backward arcs while the DP formulation in [9] cannot. To prove the polynomial size of the state space at each stage, we use a *flow decomposition* technique for CFG with multiple echelons of sinks and a *regeneration interval* technique for two-echelon CFG with constant capacity on downward arcs. Each technique has its own advantage and disadvantage. More precisely flow decomposition is more suitable for multi-echelon case without arc capacity and the regeneration interval technique is more suitable for capacitated case with a very small number of echelons. Note that CFG with $L$ echelons of sinks for general $L$ is much more difficult to prove than the two-echelon case in [9], since the structure of the optimal flows are much more complicated for general $L$.

The rest of the paper is organized as follows. In section 2, we give a polynomial-time algorithm for CFG with a fixed number of echelons and show that CFG is NP-hard with a varying number of echelons or with upward arcs. We then show that two-echelon CFG with constant capacity on downward arcs can be solved in $O(T^4)$ time in section 3. We conclude in section 4.

# 2 CFG without arc capacities

We first give a DP formulation for CFG, which is also applicable to the capacitated case in Section 3. The elements of the DP formulation are as follows:

1. Decision stages. There are $T+1$ stages corresponding to time period $t = 1, \dots, T$ with stage 0 for the dummy period 0.

2. States. Define the state $\boldsymbol{s}^t$ at stage $t$ to be a $L$-dimensional vector whose component $s_l^t$ denotes the flow over the forward arc $(v_{l,t}, v_{l,t+1})$. We assume that each component of $\boldsymbol{s}^0$ and $\boldsymbol{s}^T$ is 0. Note that the dimension of $\boldsymbol{s}^t$ can be reduced by one since the summation of the components of $\boldsymbol{s}^t$ is always $\sum_{l=1}^{L} b(v_{l,t})$ by flow balance constraints. When backward arcs are present, we augment the state vector $\boldsymbol{s}^t$ to include the flow over the backward arc $(v_{l,t+1}, v_{l,t})$ for each $l$.

3. Decision variables. The decision variable $\boldsymbol{u}^t$ at stage $t$ is a $(L-1)$-dimensional vector whose component $u_l^t$ denotes the flow over the downward arc $(v_{l,t+1}, v_{l+1,t+1})$.

4. The system equations. The state $\boldsymbol{s}^{t+1}$ at stage $t+1$ can be easily calculated by the flow balance constraints of the nodes at time period $t+1$. Let the system equations be $\boldsymbol{s}^{t+1} = H_t(\boldsymbol{s}^t, \boldsymbol{u}^t)$, where $H_t$ is the affine function representing the flow balance constraints for nodes at stage $t+1$.

5. The cost function. The cost at stage $t$ is the sum of all costs incurred by the downward arcs and forward arcs (and backward arcs) at that stage.

This DP formulation is difficult to solve directly, since the state space at each stage is an uncountable set. However from (1), we know that CFG is essentially to minimize a concave function over a flow polyhedron $P_F$. Then the optimum must be attained at an extreme point of $P_F$ (we call it extreme flow in the rest of the paper) if the problem is feasible. Since there are only a finite number of extreme flows for a given $P_F$, it suffices to consider a finite set of states corresponding to those extreme flows in the DP formulation.

## 2.1 Polynomial solvable cases

Our main result in this section is Theorem 1.

**Theorem 1.** *If all sources lie at one echelon and the number of echelons $L$ is fixed, CFG (with backward arcs) can be solved in time polynomial in $T$ and the number of queries of the value oracle.*

*Proof.* Given the proposed DP formulation, since the dimension of the state vector at each stage is $L-1$ (or $2(L-1)$ with backward arcs) with $L$ fixed, we only need to show each component of the state vector, namely the flow over each forward arc (or backward arc) under all extreme flows of $P_F$, can take on a set of values whose cardinality is polynomial in $T$. Theorem 1 follows directly from Proposition 1 below and backward induction for the DP formulation.

□

**Proposition 1.** *If all sources lie at one echelon and the number of echelons $L$ is fixed, the set of values that each arc can attain is polynomial in $T$ under all extreme flows of CFG.*

Proposition 1 presents a stronger result than what we need. It shows that each arc (not only the forward or backward arc in the state vector) can only take a polynomial number of values under all extreme flows of CFG. The technique we use to prove Proposition 1 is flow decomposition.

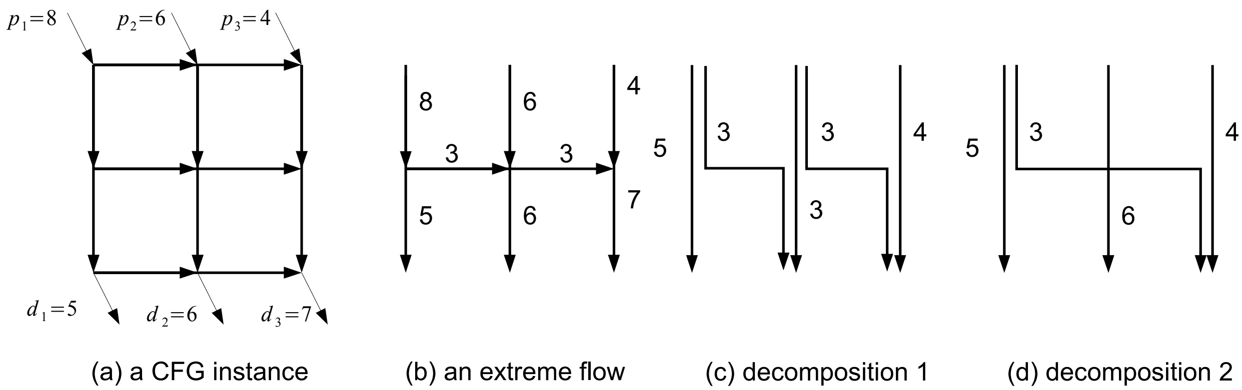



Figure 2: Two flow decompositions for an extreme flow of a 3-echelon 3-period CFG

Proposition 2 to Proposition 5 below present a series of structural results of decomposing extreme flows in a grid network. We introduce some notation first. Let $\boldsymbol{f}$ be an extreme flow of $P_F$. Let $G_f$ be the graph induced by the arcs carrying nonzero flow in $\boldsymbol{f}$. Then it is known that the underlying undirected graph of $G_f$ must be acyclic. Let $x_a^f$ be the amount of flow over arc $a$ under $\boldsymbol{f}$. Since we assume that all sources lie at the first echelon and all sinks lie below the first echelon, let the supply at source $v_{1,t}$ be $p_t$ and the demand at sink $v_{l,t}$ be $d_{l,t}$, for $t \in [T]$ and $l \in \{2, \ldots, L\}$. Proposition 2 below states that each extreme flow can be decomposed into flow over paths.

**Proposition 2** ([9])**.** *Each extreme flow $\boldsymbol{f}$ in $G$ can be decomposed into flows along paths each of which starts at one source and ends at one sink. In such a decomposition, there is at most one path with positive flow between each source-sink pair.*

Figure 2 shows two flow decompositions for an extreme flow in a CFG instance. Note that the flow decomposition for each extreme flow $\boldsymbol{f}$ is not unique. We can use a vector $\boldsymbol{\lambda}^f = (\lambda^f(P))$ to represent a flow decomposition for a given $\boldsymbol{f}$, where each component $\lambda(P)$ is the amount of flow sent along certain path $P$ connecting a source and a sink in $G_f$. Since we assume that all sources lie at the first echelon and all sinks lie below the first echelon, each path $P$ can be represented by a $(i, j, l)$-tuple, denoting that $P$ starts from source $v_{1,i}$ and ends at sink $v_{l,j}$. Then the flow decomposition $\boldsymbol{\lambda}^f$ is a vector containing $O((L-1)T^2)$ components:

$$\begin{aligned}\boldsymbol{\lambda}^f = \; & (\lambda^f(1,1,L), \lambda^f(1,1,L-1), \ldots, \lambda^f(1,1,2), \\ & \lambda^f(1,2,L), \lambda^f(1,2,L-1), \ldots, \lambda^f(1,2,2), \ldots, \\ & \lambda^f(1,T,L), \lambda^f(1,T,L-1), \ldots, \lambda^f(1,T,2), \ldots\ldots \lambda^f(T,T,2)),\end{aligned}$$

where $\lambda^f(i,j,l)$ denotes the amount of flow sent along the path connecting the source $v_{1,i}$ and sink $v_{l,j}$ in $G_f$. Then the amount of flow over arc $a$ can be calculated by summing up the amount of flow over those paths that contain the arc $a$ given the flow decomposition $\boldsymbol{\lambda}^f$:

$$x_a^f = \sum_{P:P \ni a} \lambda_P^f = \sum_{l=2}^{L} \sum_{(i,j):\text{the path from } v_{1,i} \text{ to } v_{l,j} \text{ contains arc } a} \lambda^f(i,j,l). \tag{2}$$

Since the flow decomposition is not unique for any extreme flow, we will choose a particular one for each $\boldsymbol{f}$ such that the right hand side of (2) has a concise form.

**Definition 1.** *Given an extreme flow $\boldsymbol{f}$, define $\boldsymbol{\chi}^f$ to be the lexicographically largest vector among all flow decomposition vectors for $\boldsymbol{f}$.*

Such a vector $\boldsymbol{\chi}^f$ must exist, since all flow decomposition vectors for $\boldsymbol{f}$ form a closed bounded set. The flow decomposition corresponding to $\boldsymbol{\chi}^f$ is the one we choose in computing $x_a^f$ with (2). The flow decomposition $\boldsymbol{\chi}^f$ has some nice properties, given by Proposition 3 and Proposition 4.

**Proposition 3.**

1. *For any $i_1 < i_2$, $j_1 < j_2$ and $l \in \{2, \ldots, L\}$, $\chi^f(i_1, j_2, l) \cdot \chi^f(i_2, j_1, l) = 0$.*
2. *If $\chi^f(i_1, j_1, l) > 0$ and $\chi^f(i_1, j_2, l) > 0$ with $j_1 < j_2 - 1$, then $\chi^f(i_1, j, l) = d_{l,j}$ for any $j \in \{j_1 + 1, \ldots, j_2 - 1\}$ and $l \in \{2, \ldots, L\}$.*

*Proof.* See appendix. □

Proposition 3 shows that under this particular flow decomposition $\boldsymbol{\chi}^f$, supply at each period is decomposed to satisfy demand from consecutive periods (statement 2), and demand at an early period is always served as much as possible by supply at an early period (follows from statement 1). The following proposition is a major technical result in this paper.

**Proposition 4.** *For fixed $L$, $\sum_{(i,j)\preceq(i',j')} \chi^f(i,j,l)$ can only attain a polynomial number of values in $T$ under all extreme flows for any $i', j' \in [T]$, $l \in \{2, \ldots, L\}$.*

*Proof.* See appendix. □

We need one more structural result of the extreme flow in a grid network before proving Proposition 1.

**Proposition 5.** *Given an extreme flow $\boldsymbol{f}$, let $\mathcal{P}_1$ be a path from $v_{l_1,t_1}$ to $v_{l_2,t_3}$ and $\mathcal{P}_2$ be a path from $v_{l_1,t_2}$ to $v_{l_2,t_4}$ in $G_f$ with $l_1 < l_2, t_1 \le t_2$ and $t_3 \le t_4$. If $\mathcal{P}_1$ and $\mathcal{P}_2$ both contain arc $a$, then any path from $v_{l_1,i}$ to $v_{l_2,j}$ in $G_f$ with $t_1 \le i \le t_2$ and $t_3 \le j \le t_4$ also contains the arc $a$.*

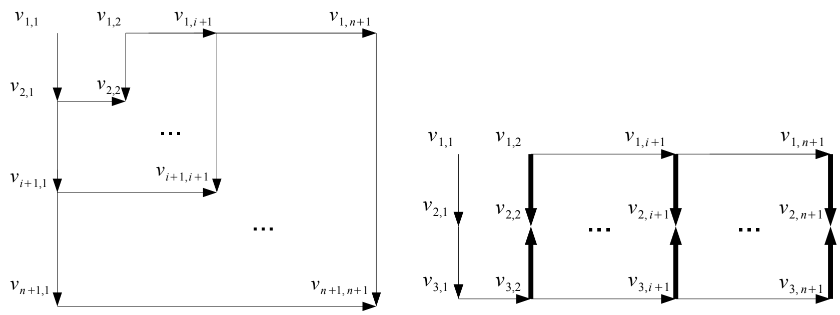
*Proof.* See appendix. □

Now we prove Proposition 1 which states that the set of values that each arc can attain is polynomial in $T$ under all extreme flows of CFG for fixed $L$.

*Proof of Proposition 1.* Given any arc $a$ and extreme flow $\boldsymbol{f}$, consider the corresponding lexicographically largest flow decomposition vector $\boldsymbol{\chi}^f$, let $(i_b^l, j_b^l)$ and $(i_e^l, j_e^l)$ be the lexicographically smallest and largest $(i,j)$ pairs such that $\chi^f(i,j,l) > 0$ and the path from source $v_{1,i}$ to sink $v_{l,j}$ contains arc $a$ in $G_f$, respectively. We claim that for any $(i,j)$ pair such that $(i_b^l, j_b^l) \preceq (i,j) \preceq (i_e^l, j_e^l)$, either $\chi^f(i,j,l) = 0$ or the path from $v_{1,i}$ to $v_{l,j}$ in $G_f$ also contains arc $a$. Since $(i_b^l, j_b^l) \preceq (i_e^l, j_e^l)$, then $i_b^l \le i_e^l$. Since $\chi^f(i_b^l, j_b^l) > 0$, $\chi^f(i_e^l, j_e^l) > 0$ and $i_b^l \le i_e^l$, we have $j_b^l \le j_e^l$ by statement 1 of Proposition 3. For any $(i,j)$ pair such that $(i_b^l, j_b^l) \prec (i,j) \prec (i_e^l, j_e^l)$, either $j < j_b^l$, $j > j_e^l$ or $j_b^l \le j \le j_e^l$. If $j < j_b^l$, then $\chi^f(i_b^l, j_b^l, l) \cdot \chi^f(i,j,l) = 0$ by statement 1 of Proposition 3, so $\chi^f(i,j,l) = 0$. If $j > j_e^l$, $\chi^f(i,j,l) = 0$ by the similar argument for the case $j < j_b^l$. For $j_b^l \le j \le j_e^l$, since both the path from $v_{1,i_b^l}$ to $v_{l,j_b^l}$ and the path from $v_{1,i_e^l}$ to $v_{l,j_e^l}$ contain the arc $a$, the path from $v_{1,i}$ to $v_{l,j}$ in $G_f$ should also contain arc $a$ by Proposition 5. Therefore given an arc $a$, an extreme flow $\boldsymbol{f}$ and the corresponding flow decomposition $\boldsymbol{\chi}^f$, by (2)

$$
\begin{aligned}
x_a^f &= \sum_{l=2}^{L} \sum_{(i,j):\text{the path from } v_{1,i} \text{ to } v_{l,j} \text{ contains arc } a} \chi^f(i,j,l) \\
&= \sum_{l=2}^{L} \sum_{(i_b^l, j_b^l)\preceq(i,j)\preceq(i_e^l, j_e^l)} \chi^f(i,j,l) \\
&= \sum_{l=2}^{L} \left[ \sum_{(i,j)\preceq(i_e^l, j_e^l)} \chi^f(i,j,l) - \sum_{(i,j)\prec(i_b^l, j_b^l)} \chi^f(i,j,l) \right].
\end{aligned}
\tag{3}
$$

The last term in (3) is a linear combination of $2(L-1)$ terms, each of which can take a polynomial number of values under all extreme flows by Proposition 4. Therefore, $x_a^f$ can only take a polynomial number of values under all extreme flows for any arc $a$. □




(a) CFG with variable $L$ (b) CFG with upward arcs

Figure 3: NP-hard CFG instances.

### 2.2 NP-hard cases

Theorem 2 demonstrates that a fixed number of echelons and no upward arcs are both critical conditions for CFG to be polynomially solvable.

**Theorem 2.**

1. *If the number of echelons $L$ is an input parameter, CFG is NP-hard.*

2. *If the grid network contains upward arcs, CFG is NP-hard for fixed $L \geq 3$.*

*Proof.* Part 1. We provide a reduction from the partition problem. An instance of the partition problem asks that given a set $S$ of integers $y_1, \ldots, y_n$ whether there exists a partition of $S$ such that the sum of the numbers in each partition is equal to $\sum_{i=1}^{2} y_i/2$. We construct an instance of CFG and show that the minimum cost of that instance is $n$ if and only if the partition instance is a yes instance and $n+1$ otherwise. The construction works as follows: consider a grid network with $L = n+1$ echelons, $T = n+1$ periods, two sources $v_{1,1}$ and $v_{1,2}$ with $b(v_{1,1}) = b(v_{1,2}) = \sum_{i=1}^{n} y_i/2$ and $n$ sinks $v_{i+1,i+1}$'s with $b(v_{i+1,i+1}) = -y_i$ for $i \in [n]$, as shown in Figure 3(a); the cost over each incoming arc for each sink is always 1 for nonzero flow and 0 otherwise, the cost over each of the rest arcs in Figure 3(a) is always 0, and the cost over each arc not in Figure 3(a) is always $2n$.
Part 2. We construct an instance of CFG with upward arcs and show that the minimum cost of that instance is $n$ if and only if the partition instance is a yes instance and $n+1$ otherwise. Construct a grid network with $L$ echelons ($L \geq 3$), $T = n+1$ periods, two sources $v_{1,1}$ and $v_{1,2}$ with $b(v_{1,1}) = b(v_{1,2}) = \sum_{i=1}^{n} y_i/2$ and $n$ sinks $v_{2,i+1}$'s with $b(v_{2,i+1}) = -y_i$ for $i \in [n]$, as shown in Figure 3(b). The cost over each incoming arc for each sink is always 1 with nonzero flow and 0 otherwise, the cost over each of other arcs in Figure 3(b) is always 0, and the cost over other arcs not present is $2n$. □

With a similar reduction as in the proof of Theorem 2, we can also show that CFG with diagonal upward arcs $(v_{l,t}, v_{l+1,t+1})$ or arcs $(v_{l,t}, v_{l+k,t+1})$ crossing $k \geq 2$ echelons is NP-hard.

# 3 CFG with arc capacities

The problem is NP-hard even if the network has only two echelons, a single source, and arc capacities on downward arcs, based on a reduction from the general capacitated lot-sizing problem [10]. Therefore to find polynomial solvable cases, we have to make some restriction on the arc capacities.

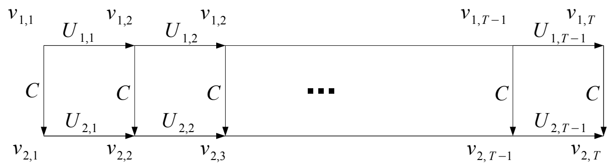



Figure 4: Two-echelon CFG with constant capacity on downward arcs.

Motivated by the fact that the constant capacitated lot-sizing problem can be in $O(T^3)$ time [14], we assume that the capacity on each downward arc in the grid network is constant. Our main result is that capacitated CFG with two echelons and constant capacity on downward arcs (CFG-2CD) can be solved in $O(T^4)$ time. Our result generalizes the constant capacity lot-sizing problem with arbitrary concave production and holding costs, backlogging, and inventory upper bounds. Below we formally define the problem CFG-2CD.

**Definition 2.** *Given a grid network $\mathcal{N} = (V, A, b)$, CFG-2CD is to find a flow $x \in \mathbb{R}^{|A|}$ to*

$$\begin{array}{lll} \text{minimize} & \sum_{a\in A} c_a(x_a) & \\ s.t. & \sum_{a\in\delta^+(v)} x_a - \sum_{a\in\delta^-(v)} x_a = b(v), & \forall v \in V, \\ & 0 \le x_a \le C, & a = (v_{1,t}, v_{2,t}), \forall t \in [T], \\ & 0 \le x_a \le U_{l,t}, & a = (v_{l,t}, v_{l,t+1}), \forall t \in [T-1], l = 1, 2, \end{array} \tag{4}$$

*where $U_{l,t}$ is the capacity on forward arc $a = (v_{l,t}, v_{l,t+1})$, and $C$ is the constant capacity on each downward arc, as shown in Figure 4.*

Given an extreme flow $\boldsymbol{f}$ of CFG-2CD, we redefine $G_f$ to be the graph induced by the arcs carrying flow strictly greater than 0 and less than the arc capacity. Then the underlying undirected graph of $G_f$ should be acyclic if $\boldsymbol{f}$ is an extreme flow. Our main technical result is Proposition 6.

**Proposition 6.** *The set of values that each forward or backward arc can attain is $O(T^3)$ under all extreme flows of CFG-2CD.*

*Proof.* To illustrate the idea of proof, we first consider a simple case where there is no backward arc and no arc capacity on any forward arc, i.e. $U_{l,t} = +\infty$ for each forward arc $a = (v_{l,t}, v_{l,t+1})$. Let $B^l_{i,j} = \sum_{t=i}^{j} b(v_{l,t})$ be the aggregated supplies and demands from period $i$ to period $j$ at echelon $l$ $(l = 1, 2)$. Let $a_{l,t}$ denote the forward arc $(v_{l,t}, v_{l,t+1})$ for each $t \in [T-1]$ and $l = 1, 2$. Since $x^f_{a_{1,t}} + x^f_{a_{2,t}} = B^1_{1,t} + B^l_{2,t}$ for all extreme flows, we only need to show that $x^f_{a_{2,t}}$ can take $O(T^3)$ values under all extreme flows.

Given an extreme flow $\boldsymbol{f}$ and a forward arc $a_{2,t}$, either $x^f_{a_{2,t}=0}$ or $x^f_{a_{2,t}} > 0$. If $x^f_{a_{2,t}} > 0$, consider the latest period $i$ before $t$ and the earliest period $j$ after $t$ when at least one of the forward arc at that period carries zero flow, i.e.

$$\begin{aligned} i &= \arg\max\{k \in [T] | k < t, x^f_{a_{1,k}} = 0 \text{ or } x^f_{a_{2,k}} = 0\}, \\ j &= \arg\min\{k \in [T] | k > t, x^f_{a_{1,k}} = 0 \text{ or } x^f_{a_{2,k}} = 0\}. \end{aligned}$$

The periods between period $i$ and period $j$ are usually called a *regeneration interval* in the lot-sizing literature [10]. Fix a pair $(i, j)$, there are four different cases regarding the position of zero-flow arcs: (1) $x^f_{a_{2,i}} = 0$ and $x^f_{a_{2,j}} = 0$; (2) $x^f_{a_{2,i}} = 0$ and $x^f_{a_{1,j}} = 0$; (3) $x^f_{a_{1,i}} = 0$ and $x^f_{a_{1,j}} = 0$; (4) $x^f_{a_{1,i}} = 0$ and $x^f_{a_{2,j}} = 0$. It suffices to show that $x^f_{a_{2,t}}$ can take $O(T^3)$ values in each of the cases.

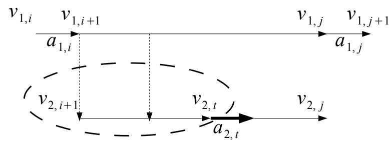



Figure 5: Case 1 $x^f_{a_{2,i}} = x^f_{a_{2,j}} = 0$.

Consider the first case $x^f_{a_{2,i}} = 0$ and $x^f_{a_{2,j}} = 0$. Since any forward arc between period $i$ and period $j$ carries nonzero flow and the underlying undirected graph of $G_f$ is acyclic, then there is at most one downward arc within period $i+1$ and period $j$ sending nonzero flow less than the capacity (otherwise there will be a cycle). Call this arc a *fractional arc*. Each of the other downward arcs between period $i$ and period $j$ either carries zero flow or flow at full capacity $C$. By the flow balance constraints for nodes $v_{2,i+1}, v_{2,i+2}, \ldots, v_{2,j}$, the total flow sent through all downward arcs within period $i+1$ and period $j$ is $B^2_{i+1,j}$. Therefore, the flow sent along the fractional arc is exactly $r_{ij} = B^2_{i+1,j} - C\lfloor \frac{B^2_{i+1,j}}{C} \rfloor$. As shown in Figure 3, if this fractional arc is before period $t$, then $x^f_{a_{2,t}} = kC + r_{ij} - B^2_{i+1,t}$ for some $0 \le k \le \lfloor \frac{B^2_{i+1,j}}{C} \rfloor$ by the flow balance constraints for nodes $v_{2,i+1}, v_{2,i+2}, \ldots, v_{2,t}$; otherwise $x^f_{a_{2,t}} = kC - B^2_{i+1,t}$ for some $0 \le k \le \lfloor \frac{B^2_{i+1,j}}{C} \rfloor$. Given each $(i,j)$ pair, $x^f_{a_{2,t}}$ can take $O(T)$ different values under all extreme flows. Therefore, $x^f_{a_{2,t}}$ can take $O(T^3)$ values in case 1 under all extreme flows. For the other three cases, $x^f_{a_{2,t}}$ can also take $O(T^3)$ values by a similar argument.

When there is arc capacity over each forward arc, we only need to modify the definition of a regeneration interval. Given an extreme flow $\boldsymbol{f}$, either $x^f_{a_{2,t}} = 0$, $x^f_{a_{2,t}} = U_{2,t}$ or $0 < x^f_{a_{2,t}} < U_{2,t}$. When $0 < x^f_{a_{2,t}} < U_{2,t}$, consider the latest period $i$ before $t$ and the earliest period $j$ after $t$ when at least one of the forward arc at that period carries zero flow or flow at full capacity, i.e.

$$
\begin{aligned}
i &= \arg\max\{k \in [T] | k < t, x^f_{a_{1,k}} = 0 \text{ or } x^f_{a_{1,k}} = U_{1,k} \text{ or } x^f_{a_{2,k}} = 0 \text{ or } x^f_{a_{2,k}} = U_{2,k}\} \\
j &= \arg\min\{k \in [T] | k > t, x^f_{a_{1,k}} = 0 \text{ or } x^f_{a_{1,k}} = U_{1,k} \text{ or } x^f_{a_{2,k}} = 0 \text{ or } x^f_{a_{2,k}} = U_{1,k}\}.
\end{aligned}
$$

For an $(i,j)$ pair, there are sixteen cases in total. We can show that $x^f_{a_{2,t}}$ takes $O(T^3)$ values under all extreme flows in each case by a similar argument when there is no capacity on forward arcs. Therefore, $x^f_{a_{2,t}}$ takes $O(T^3)$ values under all extreme flows for each forward $a_{2,t}$. Similarly when there are backward arcs, we only need to consider a constant number of more cases for each $(i,j)$ pair, and $x^f_{a_{2,t}}$ can still take $O(T^3)$ values under all extreme flows.

□

Our main result in this section is Theorem 3.

**Theorem 3.** *CFG (with backward arcs) and arc capacities can be solved in $O(T^4)$ time for $L = 2$ and constant capacity on downward arcs.*

*Proof.* CFG-2CD (with backward arcs) can be formulated as a $(T+1)$-stage DP. The state vector at each stage $t$ contains two components, the flow over forward arc $(v_{2,t}, v_{2,t+1})$ and the flow over backward arc $(v_{2,t+1}, v_{2,t})$. Under all extreme flows, one component is always zero or at full capacity, and the other component can take $O(T^3)$ different values by Proposition 6. Therefore, CFG-2CD can be solved in $O(T^4)$ time by backward induction.

□

## 4 Conclusion and future work

In this paper, we gave a polynomial-time algorithm for a class of polynomially solvable minimum concave cost flow problems, with or without arc capacities, and presented several NP-hard CFG instances. One possible direction worth pursuing is to eliminate the technical condition in Theorem 1 that all sources are at one echelon. We believe that this condition is not crucial for the polynomial solvability of CFG with a fixed number of echelons. Another problem we are currently working on is to generalize Theorem 3 to the multi-echelon CFG with constant capacity on downward arcs. The flow decomposition or regeneration interval technique alone is not sufficient to deal with both multiple echelons and arc capacities. It will be interesting to investigate whether the two techniques can be combined.

## References


[1] Alok Aggarwal and James K. Park. Improved algorithms for economic lot size problems. *Operations Research*, 41(3):549–571, 1993.

[2] Alper Atamtürk and Simge Küçükyavuz. Lot sizing with inventory bounds and fixed costs: Polyhedral study and computation. *Operations Research*, 53(4):711–730, 2005.

[3] Alper Atamtürk and Simge Küçükyavuz. An $O(n^2)$ algorithm for lot sizing with inventory bounds and fixed costs. *Operations Research Letters*, 36(3):297–299, 2008.

[4] Ranel E. Erickson, Clyde L. Monma, and Arthur F. Veinott, Jr. Send-and-split method for minimum-concave-cost network flows. *Mathematics of Operations Research*, 12(4):634–664, 1987.

[5] Awi Federgruen and Michal Tzur. A simple forward algorithm to solve general dynamic lot sizing models with $n$ periods in $O(n \log n)$ or $O(n)$ time. *Management Science*, 37(8):909–925, 1991.

[6] Michael Florian and Morton Klein. Deterministic production planning with concave costs and capacity constraints. *Management Science*, 18(1):12–20, 1971.

[7] Geoffrey M. Guisewite and Panos M. Pardalos. Minimum concave-cost network flow problems: applications, complexity, and algorithms. *Annals of Operations Research*, 25(1):75–99, 1990.

[8] Geoffrey M. Guisewite and Panos M. Pardalos. A polynomial time solvable concave network flow problem. *Networks*, 23(2):143–147, 1993.

[9] Qie He, Shabbir Ahmed, and George L. Nemhauser. Minimum concave cost flow over a grid network. *http://www.optimization-online.org/DB_HTML/2012/10/3637.html*.

[10] Yves Pochet and Laurence A. Wolsey. *Production planning by mixed integer programming*. Springer Verlag, New York, 2006.

[11] Hoang Tuy, Saied Ghannadan, Athanasios Migdalas, and Peter Värbrand. The minimum concave cost network flow problem with fixed numbers of sources and nonlinear arc costs. *Journal of Global Optimization*, 6(2):135–151, 1995.

[12] Hoang Tuy, Saied Ghannadan, Athanasios Migdalas, and Peter Värbrand. A strongly polynomial algorithm for a concave production-transportation problem with a fixed number of nonlinear variables. *Mathematical Programming*, 72(3):229–258, 1996.

[13] Wilco van den Heuvel and Albert P.M. Wagelmans. Four equivalent lot-sizing models. *Operations Research Letters*, 36(4):465–470, 2008.

[14] CPM Van Hoesel and Albert Peter Marie Wagelmans. An o $(t^3)$ algorithm for the economic lot-sizing problem with constant capacities. *Management Science*, 42(1):142–150, 1996.

[15] Stan van Hoesel, H. Edwin Romeijn, Dolores Romero Morales, and Albert P.M. Wagelmans. Integrated lot sizing in serial supply chains with production capacities. *Management science*, 51(11):1706–1719, 2005.

[16] Albert Wagelmans, Stan van Hoesel, and Antoon Kolen. Economic lot sizing: an $O(n \log n)$ algorithm that runs in linear time in the Wagner-Whitin case. *Operations Research*, 40(1):145–156, 1992.

[17] Harvey M. Wagner and Thomson M. Whitin. Dynamic version of the economic lot size model. *Management science*, 5(1):89–96, 1958.

[18] Willard I. Zangwill. A backlogging model and a multi-echelon model of a dynamic economic lot size production system-a network approach. *Management Science*, 15(9):506–527, 1969.

[19] Minjiao Zhang, Simge Küçükyavuz, and Hande Yaman. A polyhedral study of multi-echelon lot sizing with intermediate demands. *Operations Research*, 60(4):918–935, 2012.

## A Proof of Proposition 3

*Proof.*

1. Suppose that there exist $i_1 < i_2, j_1 < j_2$ and some $l_1 \in \{2, \ldots, L\}$ such that $\chi^f(i_1, j_2, l_1) \cdot \chi^f(i_2, j_1, l_1) > 0$. The path from $v_{1,i_1}$ to $v_{l_1,j_2}$ must intersect with the path from $v_{1,i_2}$ to $v_{l_1,j_1}$. If $\chi^f(i_1, j_2, l_1) \geq \chi^f(i_2, j_1, l_1)$, create a vector $\tilde{\boldsymbol{\chi}}^f$ in the following way: $\tilde{\chi}^f(i_1, j_1, l_1) = \chi^f(i_1, j_1, l_1) + \chi^f(i_2, j_1, l_1)$, $\tilde{\chi}^f(i_1, j_2, l_1) = \chi_f(i_1, j_2, l_1) - \chi_f(i_2, j_1, l_1)$, $\tilde{\chi}^f(i_2, j_1, l_1) = 0$, $\tilde{\chi}^f(i_2, j_2, l_1) = \chi^f(i_2, j_2, l_1) + \chi^f(i_2, j_1, l_1)$ and $\tilde{\chi}^f(i, j, l) = \chi^f(i, j, l)$ for other $(i, j, l)$ tuples. The vector $\tilde{\boldsymbol{\chi}}^f$ represents another flow decomposition of $\boldsymbol{f}$ and $\tilde{\boldsymbol{\chi}}^f \succ \boldsymbol{\chi}^f$, a contradiction to the fact that $\boldsymbol{\chi}^f$ is the lexicographically largest flow decomposition vector for $\boldsymbol{f}$. Similarly there is a contradiction when $\chi^f(i_2, j_1, l_1) \geq \chi^f(i_1, j_2, l_1)$.

2. Since $\chi^f(i_1, j_1, l) > 0$, by statement 1 $\chi^f(i, j, l) = 0$ for any $i < i_1$ and $j > j_1$. Since $\chi^f(i_1, j_2, l) > 0$, by statement 1 $\chi^f(i, j, l) = 0$ for any $i > i_1$ and $j < j_2$. Thus $\chi^f(i, j, l) = 0$ for each $j \in \{j_1+1, \ldots, j_2-1\}$ and $i \neq i_1$. Then $\chi^f(i_1, j, l) = d_{l,j}$ for any $j \in \{j_1+1, \ldots, j_2-1\}$ by the flow balance constraints.

□

## B Proof of Proposition 4

*Proof.* The proof is based on induction, first on time period $i'$ and then on echelon $l$.

For the base case $i' = 1$, $\sum_{(i,j) \preceq (i',j')} \chi^f(i, j, l) = \sum_{j=1}^{j'} \chi^f(1, j, l)$. We show that $\sum_{j=1}^{j'} \chi^f(1, j, l)$ can only attain a polynomial number of values based on induction on $l$. Consider the base case $l = L$. Let $P_t$ denote the cumulative supply up to period $t$ and $D_{l,t}$ denote the cumulative demand up to period $t$ at echelon $l$ for $t \in [T]$ and $l \in \{2, 3, \dots, L\}$. Let $S_{1,L} = \{D_{L,t} | t \in [T]\} \cup \{P_u - \sum_{l=2}^{L-1} D_{l,t_l} | u, t_1, t_2, \dots, t_{L-1} \in [T]\}$. We claim that

$$\sum_{j=1}^{j'} \chi^f(1, j, L) \in S_{1,L} \tag{5}$$

for any $j'$ under all extreme flows. Since the cardinality of the set $S_{1,L}$ is $O(T^L)$, we then prove the base case $l = L$. To prove (5), fix the extreme flow $\boldsymbol{f}$, let

$$j_{1,L} = \arg\max\{j | \chi^f(1, j, L) > 0\}$$

to be the latest time period of sinks on echelon $L$ that source $v_{1,1}$ contributes to. Note that $0 < \chi^f(1, j_{1,L}, L) \le d_{L,j_{1,L}}$. By statement 2 of Proposition 3, $\chi^f(1, j, L) = d_{L,j}$ for $j < j_{1,L}$. Then

$$\sum_{j=1}^{j'} \chi_f(1, j, L) = \begin{cases} D_{L,j'}, & j' < j_{1,L} \\ \sum_{j=1}^{j_{1,L}} \chi^f(1, j, L), & j' \ge j_{1,L}. \end{cases}$$

If $\chi^f(1, j_{1,L}, L) = d_{L,j_{1,L}}$, then $\sum_{j=1}^{j'} \chi^f(1, j, L) = D_{L,j_{1,L}}$ for all $j' \ge j_{1,L}$, and (5) holds. If $\chi_f(1, j_{1,L}, L) < d_{L,j_{1,L}}$, the demand $d_{L,j_{1,L}}$ must be partially satisfied by sources from later time periods. As shown in Figure 6, let $v_{1,r}$ be the first such source, so no source between $v_{1,1}$ and $v_{1,r}$ contributes to $d_{L,j_{1,L}}$, i.e. $\chi^f(i, j_{1,L}, L) = 0$ for any $1 < i < i_r$. Then by Proposition 3, the sources $v_{1,q}$ with $1 < q < r$ have no contribution to any sink at echelon $L$. We will show that

$$\sum_{j=1}^{j_{1,L}} \chi^f(1, j, L) = P_u - \sum_{l=2}^{L-1} D_{l,t_l} \tag{6}$$

holds for some $u < r$ and $t_2, \dots, t_{L-1} \in [T]$. Let $j_{i,l} = \arg\max\{j | \chi^f(i, j, l) > 0\}$ be the largest time period $j$ such that $\chi^f(i, j, l) > 0$ (set $j_{i,l} = 0$ if $\chi^f(i, j, l) = 0$ for each $j$). Since sources $v_{1,q}$ with $1 < q < r$ only contribute to sinks at echelon 2 to echelon $L - 1$, by summing up the flow decomposition components in $\boldsymbol{\chi}^f$ for sources $v_{0,1}, v_{1,2}, \dots, v_{1,q}$ we have $\sum_{j=1}^{j_{1,L}} \chi^f(1, j, L) + \sum_{l=2}^{L-1} \sum_{(i,j) \preceq (q, j_{q,l})} \chi^f(i, j, l) = P_q$ for each $q < r$. Then

$$\sum_{j=1}^{j_{1,L}} \chi^f(1, j, L) = P_q - \sum_{l=2}^{L-1} \sum_{(i,j) \preceq (q, j_{q,l})} \chi^f(i, j, l). \tag{7}$$

We then prove (6) by contradiction. Suppose that (6) does not hold, then according to (7), for each $q < r$ there must exist at least one $j_{q,l}$ for some $l$ such that $D_{l,j_{q,l}-1} < \sum_{(i,j) \preceq (q, j_{q,l})} \chi^f(i, j, l) < D_{l,j_{q,l}}$. In this case we can show that the underlying undirected graph of $G_f$ must contain a cycle, thus a contradiction. The cycle consists of two distinct paths between $v_{1,1}$ and $v_{1,r}$ ("path" refers to undirected path in the underlying graph of $G_f$ in the rest of the proof). The first path between $v_{1,1}$ and $v_{1,r}$ contains node $v_{1,j_{1,L}}$ at echelon $L$ due to the fact $\chi_f(1, j_{1,L}, L) > 0$ and $\chi^f(r, j_{1,L}, L) > 0$. The second path between $v_{1,1}$ and $v_{1,r}$ only contain nodes above echelon $L$, thus different from the first one. We now show the existence of the second path, which is a concatenation of shorter paths between many source-sink pairs.

Since for source $v_{1,1}$, there exists at least one $j_{1,l_1}$ such that $D_{l_1,j_{1,l_1}-1} < \sum_{(i,j) \preceq (1, j_{1,l_1})} \chi^f(i, j, l_1) < D_{l_1,j_{1,l_1}}$. Choose such $j_{1,l_1}$ with $l_1$ being the largest. There is a path from $v_{1,1}$ to $v_{l_1,j_{1,l_1}}$ due to that $\chi^f(1, j_{1,l_1}, l_1) > 0$, as shown in Figure 6. Meanwhile there exists a source $v_{1,i_2}$ ($i_2 > 1$) such that $\chi^f(i_2, j_{1,l_1}, l_1) > 0$ due to the assumption that $\sum_{(i,j)\preceq(1,j_{1,l_1})} \chi^f(i, j, l_1) < D_{l_1,j_{1,l_1}}$, so there is a path between $v_{1,i_2}$ and $v_{1,j_{1,l_1}}$. Then there is a path between $v_{1,1}$ and $v_{1,i_2}$ through $v_{1,j_{1,l_1}}$ only containing nodes above echelon $(l_1+1)$. In addition, we choose such $v_{1,i_2}$ with $i_2$ being the smallest.

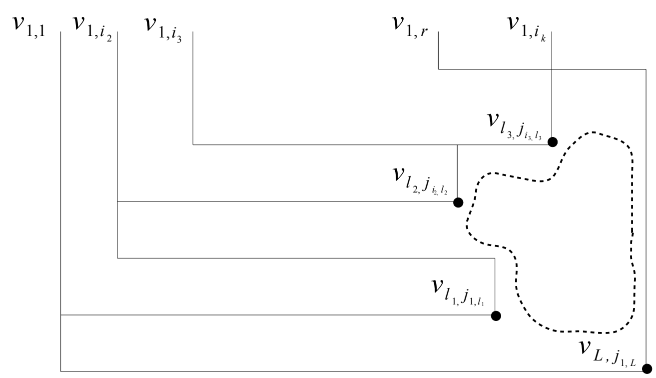



Figure 6: The circle that traverses nodes $v_{L,j_{1,L}}$,$v_{l_1,j_{1,l_2}}$, $v_{l_2,j_{i_2,l_2}}$ and $v_{l_3,j_{i_3,l_3}}$.

If $i_2 \geq r$, the path from $v_{1,1}$ to $v_{1,i_2}$ must intersect with the path from $v_{1,r}$ to $v_{L,j_{1,L}}$ at a node above echelon $L$, so there is a path from $v_{1,1}$ to $v_{1,r}$ containing nodes only above echelon $L$ (more precisely above echelon $(l_1 + 1)$). If $i_2 < r$, by assumption there must exist at least one $j_{i_2,l_2}$ such that $D_{l_2,j_{i_2,l_2}-1} < \sum_{(i,j)\preceq(i_2,j_{i_2,l_2})} \chi^f(i, j, l_2) < D_{l_2,j_{i_2,l_2}}$. Choose such $j_{i_2,l_2}$ with $l_2$ being the largest. We will show that there is path from $v_{1,i_2}$ to $v_{l_2,j_{i_2,l_2}}$ as well.

1. Case $l_1 \leq l_2$. For source $v_{1,1}$, $\sum_{(i,j)\preceq(1,j_{1,l_2})} \chi^f(1, j, l_2)$ equals to certain cumulative demand at echelon $l_2$ due to the choice of $l_1$. Sources $v_{1,2}, \ldots, v_{1,i_2-1}$ make no contribution to the sinks at echelon $l_2$. For source $v_{1,i_2}$, $D_{l_2,j_{i_2,l_2}-1} < \sum_{(i,j)\preceq(i_2,j_{i_2,l_2})} \chi^f(i, j, l_2) < D_{l_2,j_{i_2,l_2}}$. Then $\chi^f(i_2, j_{i_2,l_2}, l_2) > 0$, and there is a path from $v_{1,i_2}$ to $v_{l_2,j_{i_2,l_2}}$.

2. Case $l_1 > l_2$. If $\chi_f(i_2, j_{i_2,l_2}, l_2) = 0$, since $\sum_{(i,j)\preceq(i_2,j_{i_2,l_2})} \chi^f(i, j, l_2) < D_{l_2,j_{i_2,l_2}}$, there has to be two sources $v_{1,x}, v_{1,y}$ with $x < i_2 < y$, $\chi^f(x, j_{i_2,l_2}, l_2) > 0$ and $\chi^f(y, j_{i_2,l_2}, l_2) > 0$. Then the path between $v_{1,x}$ and $v_{1,y}$ through the sink $v_{l_2,j_{i_2,l_2}}$ must intersect with the path between $v_{1,1}$ and $v_{1,i_2}$. Therefore, $v_{1,i_2}$ and $v_{l_2,j_{i_2,l_2}}$ are connected by a path.

Since the demand at the sink $v_{l_2,j_{i_2,l_2}}$ is not fulfilled by sources up to $v_{1,i_2}$, there is another source $v_{1,i_3}$ with $i_3 > i_2$ such that $\chi^f(i_3, j_{i_2,l_2}, l_2) > 0$. Then there is a path from $v_{1,1}$ to $v_{1,i_3}$ through $v_{1,i_2}$ and $v_{l_2,j_{i_2,l_2}}$ only containing nodes above echelon $L$. We can continue this argument until we find a source $v_{1,i_k}$ connected to $v_{1,1}$ with $i_k \geq r$. Then the path between $v_{1,1}$ and $v_{1,i_k}$ must intersect with the path between $v_{1,r}$ and $v_{L,j_{1,L}}$. Then we find a path from $v_{1,1}$ to $v_{1,r}$ only containing nodes above echelon $L$.

We have finished the proof for the base case $l = L$ when $i' = 1$, and now proceed to the induction step when $i' = 1$. Suppose that the set of values $\sum_{j=1}^{j'} \chi^f(1, j, l)$ can attain is polynomial in $T$ for $j' \in [T]$ and $l > l'$, we would like to show the result also holds for $l = l'$.

1. If $\chi^f(1, j_{1,l'}, l') = d_{l',j_{1,l'}}$, then

$$\sum_{j=1}^{j'} \chi^f(1, j, l') = \begin{cases} D_{l',j'}, & j' < j_{1,l'} \\ D_{l',j_{1,l'}}, & j' \geq j_{1,l'}. \end{cases}$$

The term $\sum_{j=1}^{j'} \chi^f(1,j,l')$ can attain $T$ values under all extreme flows.

2. If $0 < \chi^f(1, j_{1,l'}, l') < d_{l',j_{1,l'}}$, then

$$\sum_{j=1}^{j'} \chi^f(1,j,l') = \begin{cases} D_{l',j'}, & j' < j_{1,l'} \\ \sum_{j=1}^{j_{1,l'}} \chi^f(1,j,l'), & j' \geq j_{1,l'}. \end{cases}$$

Let $r'$ be the smallest index $i > 1$ such that $\chi^f(i, j_{1,l'}, l') > 0$. Then by a similar argument in proving the validity of (6), there must exist some $u < r'$ and $t_2, \ldots, t_{l'-1} \in [T]$ such that

$$\sum_{j=1}^{j_{1,l'}} \chi^f(1,j,l') = P_u - \sum_{l=2}^{l'-1} D_{l,t_l} - \sum_{l=l'+1}^{L} \sum_{j=1}^{j_{1,l}} \chi^f(1,j,l). \tag{8}$$

By the induction hypothesis, the number of values that $\sum_{j=1}^{j_{1,l}} \chi^f(1,j,l)$ can attain is polynomial in $T$ for any $l > l'$. Then the result also holds for $l = l'$ according to (8).

We have finished the proof for the base case $i' = 1$ and now proceed to the induction step. Suppose that the number of values that $\sum_{(i,j)\preceq(i',j')} \chi^f(i,j,l)$ can attain is polynomial in $T$ for all $j', l$ and $1 \leq i' < k$, we would like to show the result holds for $i' = k$. The proof is also based on induction on echelon index $l$, similar to the proof for the base case $i' = 1$. We first prove the base case $l = L$. Given an extreme flow $\boldsymbol{f}$, the result holds in all three cases below.

1. $\chi^f(k,j,L) = 0$ for all $j$, i.e. the source $v_{0,k}$ makes no contribution to sinks on echelon $L$ under $\boldsymbol{\chi}^f$. Then $\sum_{(i,j)\preceq(k,j')} \chi^f(i,j,L) = \sum_{(i,j)\preceq(k-1,T)} \chi^f(i,j,L)$, and the result follows from the induction hypothesis.

2. $\chi^f(k, j_{k,L}, L) > 0$ and $\sum_{(i,j)\preceq(k,j_{k,L})} \chi^f(i,j,L) = D_{L,j_{k,L}}$. Then

$$\sum_{(i,j)\preceq(k,j')} \chi^f(i,j,L) = \begin{cases} \sum_{(i,j)\preceq(k-1,T)} \chi^f(i,j,L), & j' < j_{k,L}, \chi^f(k,j',L) = 0 \\ D_{L,j'}, & j' < j_{k,L}, \chi^f(k,j',L) > 0 \\ D_{L,j_{k,L}}, & j' \geq j_{k,L}. \end{cases}$$

Then the result follows from the induction hypothesis.

3. $\chi^f(k, j_{k,L}, L) > 0$ and $\sum_{(i,j)\preceq(k,j_{k,L})} \chi^f(i,j,L) < D_{L,j_{k,L}}$.

$$\sum_{(i,j)\preceq(k,j')} \chi^f(i,j,L) = \begin{cases} \sum_{(i,j)\preceq(k-1,T)} \chi^f(i,j,L), & j' < j_{k,L}, \chi^f(k,j',L) = 0 \\ D_{L,j'}, & j' < j_{k,L}, \chi^f(k,j',L) > 0 \\ \sum_{(i,j)\preceq(k,j_{k,L})} \chi^f(i,j,L), & j' \geq j_{k,L}. \end{cases}$$

By a similar argument in proving the validity of (6), there exists some $u, t_2, \ldots, t_{L-1} \in [T]$, such that

$$\sum_{(i,j)\preceq(k,j_{k,L})} \chi^f(i,j,L) = P_u - \sum_{l=2}^{L-1} D_{l,t_l}.$$

Finally, the induction step for case $i' = k$ is similar to the proof of the induction step for the base case $i' = 1$.

□

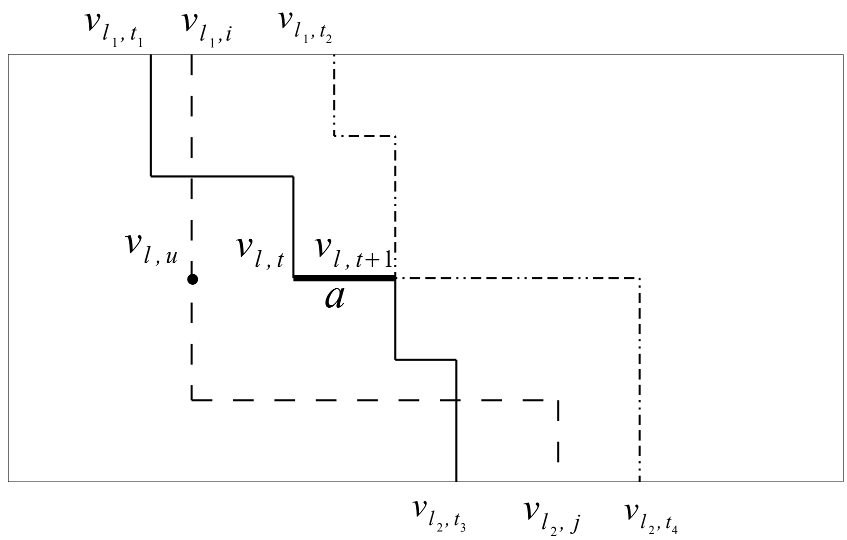



Figure 7: The cycle created in $G_f$ if the path from $v_{l_1,i}$ to $v_{l_2,j}$ bypasses the arc $a$.

# C Proof of Proposition 5

*Proof.* Proof by contradiction. As shown in Figure 7, the arc $a = (v_{l,t}, v_{l,t+1})$ is contained in the path from $v_{l_1,t_1}$ to $v_{l_2,t_3}$ and the path from $v_{l_1,t_2}$ to $v_{l_2,t_4}$ in $G_f$. Suppose that there exists some pair $(i, j)$ with $t_1 \leq i \leq t_2$ and $t_3 \leq j \leq t_4$ such that the path from $v_{l_1,i}$ to $v_{l_2,j}$ bypasses the arc $a$. Then the path must contain some node $v_{l,u}$ with either $u \leq t$ or $u \geq t+1$. If $u \leq t$, the path from $v_{l_1,t_1}$ to $v_{l,t}$ must intersect with the path from $v_{l_1,i}$ to $v_{l,u}$, and the path from $v_{l,t+1}$ to $v_{l_2,t_3}$ must intersect with the path $v_{l,u}$ to $v_{l_2,j}$ in $G_f$. The two intersections create a cycle in the underlying undirected graph of $G_f$, a contradiction. The argument is essentially the same if $u \geq t+1$.

□