Hindawi
International Journal of Mathematics and Mathematical Sciences
Volume 2021, Article ID 6651135, 7 pages
https://doi.org/10.1155/2021/6651135

Research Article

Lexicographically Maximum Contraflow Problem with Vertex Capacities

Phanindra Prasad Bhandari and Shree Ram Khadka

Central Department of Mathematics, Tribhuvan University Kathmandu, Kirtipur, Nepal

Correspondence should be addressed to Shree Ram Khadka; shree.khadka@cdmath.tu.edu.np

Received 26 November 2020; Revised 22 January 2021; Accepted 1 February 2021; Published 13 February 2021

Academic Editor: Chin-Chia Wu

The contraflow approach has been extensively considered in the literature for modeling evacuations and has been claimed, due to its lane-direction-reversal capability, as an efficient idea to speed up the evacuation process. This paper considers the contraflow evacuation model on network with prioritized capacitated vertices that allows evacuees to be held at intermediate spots too, respecting their capacities and priority order. In particular, it studies the maximum flow evacuation planning problem and proposes polynomial and pseudo-polynomial time solution algorithms for static network and dynamic multinetwork, respectively. A real dataset of Kathmandu road network with evacuation spaces is considered to implement the algorithm designed for dynamic multinetwork and to observe its computational performance.

1. Introduction

The contraflow approach, which refers to the reversibility of direction of traffic flow in one or more lanes of roadways for fixed time period, re-configures the road network, identifying ideal direction and reallocating available capacity for each arc. The approach, due to its lane-direction-reversal capability, can be taken as a potential remedy to mitigate congestion during emergencies. It significantly reduces the total evacuation time and/or increases the number of evacuees sent from the risk zone to safety. Studies show that reversing one lane of a four-lane dual highway increases the evacuation road capacity by approximately 30%, and by reversing all the inbound lanes, it increases by 67% [1]. The contraflow approach is primarily important for emergency evacuations; nonetheless, its applications are not limited to these. This is commonly used for accommodating directionally imbalanced traffic associated with daily commuter in big cities as well as consequences due to religious gathering, arrangement of concerts or tournaments, etc. However, there is limited implementation of it in real emergency evacuations due to difficulty in using commonly employed methods to duplicate traffic conditions of real contraflow lane during an emergency [1].

The first mathematical optimization model for the contraflow problem was proposed by Rebennack et al. [2] that relies on the basis of the network flow model in [3]. They have investigated analytical solutions for the maximum static contraflow (MSCF) problem and maximum dynamic contraflow (MDCF) problem with polynomial time complexities. The solution idea is based on transformation of input network into a new network for which existing network flow algorithms are applicable. The authors in [4] studied the continuous time maximum dynamic contraflow evacuation problem and proposed a polynomial time solution using the notion of natural transformation of flows suggested in [5].

Other variants that are closely related to the MDCF problem are the quickest contraflow (QCF) problem and earliest arrival contraflow (EACF) problem. The QCF problem on single-source-single-sink network has been solved polynomially in [2]. The EACF problem for the two-terminal series-parallel (TTSP) network has been studied and a polynomial time solution for this has been proposed in [6]. Maximum as well as earliest version of evacuation contraflow

problems in network with not necessarily equal transit time on anti-parallel lanes have been studied in [7]. Network reconstruction-based solution procedures have also been proposed for these problems modeled with discrete as well as continuous time setting. The authors in [8] studied these problems for multinetwork setup and proposed polynomial time solutions for both discrete as well as continuous time models. However, the solution procedures for earliest version of the problems work only for TTSP network. The contraflow approach has been incorporated in the network flow model to study facility location problem in [9], and the notion of abstract flow has been applied to network contraflow problems in [10]. The partial contraflow approach over the abstract network setting has been introduced in [11]. We refer to the survey articles [12, 13] for broader insight into dynamic network flow problems and evacuation planning problems.

This paper introduces a new aspect of the evacuation model designed on network with capacitated vertices by imposing the contraflow approach on it. The new model has arc reversal capability and is capable of holding evacuees at temporary shelters at intermediate vertices of given priority. The flow model adopted here is based on weak-conservation constraints given in [14] (cf. [15]). Based on this aspect, the maximum static contraflow problem on ordinary network and maximum dynamic contraflow problem on multinetwork are studied, and solution algorithms for them are proposed. It is crucial, in case of uneven road architecture, for example, to take contraflow models on multinetwork into account for preparing evacuation tasks [8]. Multinetworks capture the situation of road topology with parallel lanes of different transit time and anti-parallel lanes of unequal to and fro transit time. It is considered that the transit time parameter behaves symmetrically during the reversal of arc direction in the case of the dynamic contraflow problem.

The evacuation flow model introduced in [14] is revisited and the lexicographically maximum contraflow problem on network with capacitated vertices is introduced in Section 2. The solution procedures to the problems for static and dynamic cases are proposed in Sections 3.1 and 3.2, respectively. A case illustration with a real dataset is made in Section 4. Section 5 concludes the paper.

2. Model Description

Consider a directed multigraph $G = (V, A)$ with vertex set V and arc set A, both to be finite, such that $n: = |V|$ and $m: = |A|$. Represent the source and the sink by s and d, respectively, and assume a terminal set $S \subset V$ with $\mathcal{S}: = \{v_1, \ldots, v_r\}$ prioritized from higher to lower priority, i.e., $d = v_1 \succ \cdots \succ v_r$, to be given. Then, the corresponding two-terminal evacuation network for time horizon T is represented as $\mathcal{N} = (G, l(a), u(a), \tau(a), k(v), s, d, T)$. Here, $l: A \longrightarrow \mathbb{N}_0: = \mathbb{N} \cup \{0\}$ and $u: A \longrightarrow \mathbb{N}_0$ represent the lower and upper arc capacity functions which bound the number of flow units on each arc $a \in A$ at each time step from below and from above, respectively. Similarly, the vertex capacity function $k: \mathcal{S} \longrightarrow \mathbb{N}_0$ delimits the total number of flow units, which may be held in each of the vertices $v \in \mathcal{S}$. Moreover, the transit time function $\tau: A \longrightarrow \mathbb{N}$ specifies the time needed by a flow unit to traverse an arc. Treat time parameter in a discrete manner, i.e., $\mathcal{T}: = \{0, 1, \ldots, T\}$.

The non-negative flow variables $f(a, t)$ defined by $f: A \times \mathcal{T} \longrightarrow \mathbb{N}_0$ that specify the flow over time in the network $\mathcal{N}$ are the number of flow units entering arc a at time step t. The number of flow units entering arc a at time step t is assumed to be bounded by the capacity of an arc, i.e., $f(a, t)$ satisfies the capacity constraints for all $a \in A$ and for all $t \in \mathcal{T}$. That is,

$$0 \leq f(a, t) \leq u(a), \quad \forall a \in A, \forall t \in \mathcal{T}. \tag{1}$$

Moreover, $f(a, t)$ has to be equal to zero for all $t > T - \tau(a)$ and for all $a \in A$. The excess flow at vertex $v \in V$ at time $t \in \mathcal{T}$, denoted by $\mathrm{ex}_f(v, t)$, is defined as

$$0 \leq \mathrm{ex}_f(v, t) := \sum_{a \in \delta^-(v)} \sum_{\xi=0}^{t-\tau(a)} f(a, \xi) - \sum_{a \in \delta^+(v)} \sum_{\xi=0}^{t} f(a, \xi), \tag{2}$$

where $\delta^-(v): = \{a \in A: a = (w, v) \text{for some vertex } w \in V\}$ and $\delta^+(v): = \{a \in A: a = (v, w) \text{for some vertex } w \in V\}$ denote the set of arcs entering and leaving vertex $v \in V$, respectively.

Further, we need to ensure that the excess flow at each vertex $v \in \mathcal{S}$ over time horizon T is to be bounded by the capacity $k(v)$, i.e.,

$$\mathrm{ex}_f(v, T) \leq k(v), \quad \text{for all } v \in \mathcal{S}. \tag{3}$$

Consequently, the total flow of evacuees leaving the source s is equal to the total flow of the evacuees held at vertices $v \in \mathcal{S}$ over the time horizon T, i.e.,

$$\sum_{a \in \delta^+(s)} \sum_{\xi=0}^{T} f(a, \xi) - \sum_{a \in \delta^-(s)} \sum_{\xi=0}^{T} f(a, \xi) = \sum_{v \in \mathcal{S}} \mathrm{ex}_f(v, T). \tag{4}$$

An arc $a = (v, w) \in A$ in which the flow could travel from vertex v to vertex w is replaced by the arc (w, v) for contraflow purpose. The important feature of the considered dynamic network $\mathcal{N}$ is that the capacities and the transit time on anti-parallel arcs could be unequal, and it is allowed to have parallel arcs with different transit time only. Thus, the static network $\mathcal{N}$, we consider here, is not a multinetwork. To this end, the objective of maximum contraflow evacuation planning problem is to lexicographically maximize the vector $(\mathrm{ex}_f(v_1, T), \ldots, \mathrm{ex}_f(v_r, T))^\top$ such that $\mathrm{ex}_f(v_i, T) \leq k(v_i)$ for $i = 1, \ldots, r$, if the direction of arcs on $\mathcal{N}$ is allowed to reverse. The network flow problem with this objective is termed as *lexicographically maximum dynamic contraflow problem* and abbreviated as LexMDCF problem. The maximum contraflow problem with above objective for static network $\mathcal{N} = (G, l(a), u(a), k(v), s, d)$ is termed as

lexicographically maximum static contraflow problem and is abbreviated as LexMSCF problem.

3. Solution Discussion

Rebennack et al. [2] proposed polynomial time analytical solutions to the MSCF problem and the MDCF problem for the first time. They considered the problems in ordinary network that do not have capability of holding flows at intermediate vertices. Their solution idea is based on the reconstruction of input network into a new one, in which the existing network flow algorithms are applicable. This section discusses the solution procedures to LexMSCF problem for ordinary network and LexMDCF problem for multinetwork based on network reconstruction idea.

3.1. Lexicographically Maximum Static Contraflow Problem. Consider a static network $\mathcal{N} = (V, A, l(a), u(a), \tau(a), k(v), s, d)$ with terminal set $\mathcal{S} \subset V$ as described in Section 2. Moreover, consider that $k(d) = \infty$, and consider $k(v)$ to be finite for all $v \in S \backslash \{d\}$. The lexicographically maximum static flow (LexMSF) problem that lexicographically maximizes the amount of flow entering a set of terminals in $\mathcal{S}$ with respect to a given prioritization and given vertex capacities has been solved polynomially in [14]. Here, the objective of the LexMSCF problem is to solve the LexMSF problem on $\mathcal{N}$, if direction of arcs on $\mathcal{N}$ can be reversed.

We modify the solution idea of Rebennack et al. [2] that solves MSCF problem to solve the LexMSCF problem. Their idea is based on modification of input network into a new network by summing the capacities on arcs (v, w) and (w, v) such that MSCF problem reduces to MSF problem on it. In particular, the procedure has following steps. At first, given static network $\mathcal{N} = (V, A, u(a))$ is transformed into its auxiliary network $\widetilde{\mathcal{N}} = (V, \widetilde{A}, u(\widetilde{a}))$ where the arc set $\widetilde{A}$ contains undirected arc (v, w), if (v, w) and/or (w, v) belong to original arc set A with capacity $u(\widetilde{v,w}) = u(v, w) + u(w, v)$. Secondly, a maximum static $s - d$ flow is computed on so-formed undirected network $\widetilde{\mathcal{N}}$ by using any known algorithm. In our case, to ensure the intermediate holding capability in the solution and to respect the vertex capacities, the lexicographically maximum flow is computed by using LexMSF flow computation idea given in [14] instead of computing ordinary maximum flow. The modified procedure that solves LexMSCF problem is given in Algorithm 1.

We state the following lemma that shows equivalence between the optimal flow on $\widetilde{\mathcal{N}}$ and the optimal contraflow on input network $\mathcal{N}$, which turns out to be useful in optimality proof of algorithms designed for contraflow problems in this paper.

Lemma 1 (see [2]). *The maximum static contraflow on a static network $\mathcal{N}$ is equivalent to the maximum static flow on the corresponding transformed network $\widetilde{\mathcal{N}}$.*

Theorem 1. *Given a static network $\mathcal{N} = (V, A, l(a), u(a), k(v), s, d)$, source s and terminal set $\mathcal{S} = \{v_1, \ldots, v_r\} \subset V$ with $d = v_1 \succ \cdots \succ v_r$, and $l(a) = 0$ for all $a \in A$. Then, Algorithm 1 computes a lexicographically maximum static contraflow on $\mathcal{N}$ optimally in strongly polynomial time.*

Proof. The LexMSF Algorithm optimally computes a static flow for each terminals $v \in \mathcal{S}$ as sinks on reduced network $\widetilde{\mathcal{N}}$ (see [14]). Moreover, Lemma 1 shows that these flows are equivalent to the maximum static contraflows on the input network $\mathcal{N}$.

The computational complexity of the algorithm depends on time complexity of the solution procedure on the reduced network $\widetilde{\mathcal{N}}$. This is dominated by the time complexity of the solution procedure of LexMSF problem on $\widetilde{\mathcal{N}}$ since the flow decomposition in each iteration and network transformation can be done only in $O(mn)$, see [16], and $O(m)$ time, respectively. Note that the LexMSF problem can be solved in strongly polynomial time [14]. □

3.2. Lexicographically Maximum Dynamic Contraflow Problem for Multinetwork. Multinetworks capture the evacuation situation with anti-parallel lanes of unequal to and fro transit time as well as parallel lanes of unequal transit time. Maximum dynamic contraflow problems modeled on these class of network without capacitated vertices have been studied in [8]. For given dynamic multinetwork $\mathcal{N} = (V, A, l(a), u(a), \tau(a), k(a), s, d, T)$ and terminal set $\mathcal{S} := \{v_1, \ldots, v_r\}$ of capacitated vertices with $d = v_1 \succ \cdots \succ v_r$, the aim is to solve the LexMDCF problem, if the arc reversibility is permitted only once at time zero. In the following, the solution procedure (Algorithm 2) that solves the MDCF problem for multinetwork is modified to solve LexMDCF problem.

Solving MDCF problem, the arc $(w, v) \in A$ is reversed, if the flow along arc (v, w) exceeds $u(v, w)$ for $\tau(v, w) \leq \tau(w, v)$, or $\tau(w, v) < \tau(v, w)$. This can be viewed, alternatively, as follows: for $(v, w), (w, v) \in A$ such that $\tau(v, w) = \tau(w, v)$, the flow value at arc $\widetilde{a} = (v, w) \in \widetilde{A}$ greater than the capacity $u(v, w)$ of the corresponding arc $(v, w) \in A$ means there is flipping of the direction of arc $(w, v) \in A$. Similarly, in the case with unequal transit time, we can see the sense of flipping the direction of arc $a = (w, v) \in A$, if there is some positive flow on the corresponding arc $\widetilde{a} = (v, w)$. The minimum cost flow (MCF) algorithm applied to generate a dynamic temporally repeated flow ensures that there is flow along the arc (v, w) with less or equal transit time in comparison to the transit time of corresponding anti-parallel arc (w, v), regardless of whether the arc (w, v) is saturated. The parallel arcs $(v, w) \in \widetilde{N}$ have been labeled as $(v, w)_i$ such that $\tau(v, w)_i < \tau(v, w)_{i+1}$, for $i = 1, 2, \ldots, q; q \leq m$, to avoid obstruction on the multinetwork while applying MCF algorithm.

Following theorems (Theorems 2 and 3) show that the Algorithm 2 solves maximum dynamic contraflow problem for multinetwork optimally in strongly polynomial time.

Theorem 2. *Given a dynamic multinetwork $\mathcal{N} = (V, A, u(a), \tau(a), s, d, T)$ with integer inputs. Then, maximum dynamic flow on $\widetilde{\mathcal{N}}$ is equivalent to a maximum dynamic contraflow on $\mathcal{N}$.*

Proof. The auxiliary network $\widetilde{\mathcal{N}}$ of the original network $\mathcal{N}$ obtained in step 2 is an undirected multinetwork. The maximum dynamic contraflow problem on $\mathcal{N}$ can be viewed as a maximum dynamic flow problem on $\widetilde{\mathcal{N}}$. While solving the latter problem on $\widetilde{\mathcal{N}}$, the network is to be further transformed by replacing each undirected arc by two oppositely directed arcs with capacities and transit times of both arcs equal to that of original arc. This allows us to send flow on either direction of the arc. However, the flow direction, once chosen, remains fixed throughout the procedure. That is, there is only a flow on one direction of any arc, and never in both directions at the same time as well as at different time periods. However, there could be a flow along arc (v, w) and (w, v) such that $\tau(v, w) \neq \tau(w, v)$ for $(v, w), (w, v) \in A$ at the same time or at different time periods. The latter situation does not make the flow on $\widetilde{\mathcal{N}}$ an infeasible since, in fact, arcs (v, w) and (w, v) are physically different arcs for $\tau(v, w) \neq \tau(w, v)$, due to the labeling of arcs in step 3. Thus, the flow constructed by Algorithm 2 is feasible.

Since every feasible flow of the maximum dynamic flow problem on the transformed network $\widetilde{\mathcal{N}}$ is feasible to the maximum dynamic contraflow problem on network $\mathcal{N}$, the maximum dynamic flow on $\widetilde{\mathcal{N}}$ is not greater than the maximum dynamic contraflow on $\mathcal{N}$. On the other hand, since maximum dynamic flow on network $\mathcal{N}$ does not exceed maximum flow for the corresponding time expanded network $\mathcal{N}^T$ [3], the maximum dynamic contraflow on $\mathcal{N}$ is not greater than the maximum static contraflow in time expanded network $\mathcal{N}^T$. This static contraflow is equivalent to the optimal static flow in $\widetilde{\mathcal{N}}^T$ due to the fact that any maximum static contraflow on network N has equivalent maximum flow in the corresponding transformed network $\widetilde{\mathcal{N}}$ [2]. Again, since there exists a temporally repeated flow which is maximal over the time horizon T [3], the optimal static flow in $\widetilde{\mathcal{N}}^T$ is equivalent to the temporally repeated flow on $\widetilde{\mathcal{N}}$. Thus, the optimal dynamic contraflow on $\mathcal{N}$ is not greater than the optimal dynamic flow on $\widetilde{\mathcal{N}}$. □

Theorem 3. *For dynamic multinetwork $\mathcal{N} = (V, A, u(a), \tau(a), s, d, T)$ with integer inputs, Algorithm 2 runs in strongly polynomial time.*

Proof. Construction of auxiliary network in step 2 and labeling parallel arcs in step 3 require only linear time on m. The running time of Algorithm 2 is dominated by computation of a maximum dynamic flow in step 3. It is computed with the help of temporally repeated flow on $\widetilde{\mathcal{N}}$. Finding a temporally repeated flow is equivalent to solving a minimum cost flow problem. The minimum mean cycle-canceling algorithm of [17], for instance, requires $O(n^2 m^3 \log n)$ time for solving this problem. Next effort is to decompose the maximum static flow which requires $O(mn)$ time [16]. Thus, Algorithm 2 runs in a strongly polynomial time for dynamic multinetwork $\mathcal{N}$.

For given dynamic network $\mathcal{N}$ and terminal set $\mathcal{S}$ as described in Section 2, a lexicographically maximum dynamic flow (LexMDF) problem that lexicographically maximizes the amount of flow entering a set of terminals in $\mathcal{S}$ with respect to a given prioritization and given vertex capacities has been studied in [14]. The solution procedure to solve this problem is based on the notion of time expanded network introduced in [3]. Since the objective of LexMDCF problem is to respect the vertex capacities on the prioritized terminals, it is not sufficient to compute a maximum dynamic flow by the means of ordinary temporally repeated flows as in Algorithm 2. Instead, to solve the LexMDCF problem, a lexicographically maximum dynamic flow computation technique proposed in [14] can be applied. The solution procedure to solve LexMDCF problem for multinetwork has been summarized in Algorithm 3. □

Theorem 4. *Given a multinetwork $\mathcal{N} = (V, A, l(a), u(a), \tau(a), k(a), s, d, T)$, source s, and terminal set $\mathcal{S} = \{v_1, \ldots, v_r\}$ with $d = v_1 \succ \cdots \succ v_r$, and $l(a) = 0$ for all $a \in A$. Then, Algorithm 3 computes a lexicographically maximum dynamic contraflow on $\mathcal{N}$ in pseudo-polynomial time.*

Proof. The LexMDF algorithm of [14] is applied in Algorithm 3 that computes dynamic flows on transformed network $\widetilde{\mathcal{N}}$ iteratively for each vertex $v \in \mathcal{S}$ in priority order optimally. These flows have equivalent maximum dynamic contraflows on input network $\mathcal{N}$ for each iteration due to Theorem 2. Also, the application of LexMDF algorithm is dominating step with run time depending upon parameter T. Rest of the steps can be performed in strongly polynomial time. Thus, Algorithm 3 computes a lexicographically maximum dynamic contraflow on multinetwork $\mathcal{N}$ in pseudo-polynomial time. □

4. Case Illustration

A case illustration is made by considering Kathmandu road network within and on the Ring Road (see Figure 1). Two scenarios: Scenario I with 38 vertices, 118 arcs, and 13 intermediate shelters and Scenario II (including minor road segments) with 52 vertices, 180 arcs, and 14 intermediate shelters, are examined. New Road area, a highly congested business hub with narrow streets, is taken as the source. Evacuation spaces identified in [18] are taken as the sink (Tribhuvan University (TU) area and Bagmati Corridor near Balkhu) and other intermediate shelters. Standard area (population per 45 square meters sphere) has been considered for the holding capacity of each intermediate evacuation space. However, the sink is assumed to have sufficient capacity. Tables 1 and 2 show the name of evacuation spaces together with their corresponding holding capacities and priority order. For this case illustration, shelters (except sink) are prioritized in random selection (however, it can be done with respect to their capacities, distance from the source or available facilities, and so on). Being a discrete time auto-based evacuation planning model, it followed the "two second rule," considering each minute as

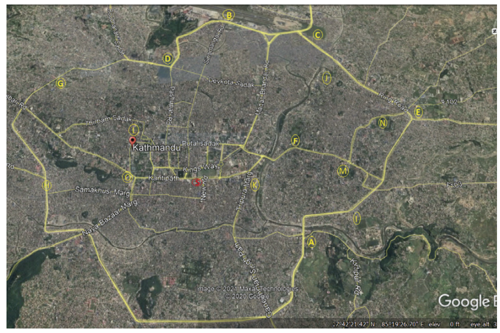

Figure 1: Road network for case illustration taken from Google Earth Pro where S denotes source (New Road) and letters from A to O are sink (Balkhu) and other prioritized intermediate vertices as mentioned in Table 2. Outer boundary traced yellow is Ring Road.

(1) Given a static network $\mathcal{N} = (V, A, l(a), u(a), k(v), s, d)$, $\mathcal{S}\colon = \{v_1, \ldots, v_r\} \subset V$ with $d = v_1 \succ \cdots \succ v_r$, $l(a) = 0$ for all $a \in A$ and integer inputs.
(2) Transform $\mathcal{N}$ into network $\widetilde{\mathcal{N}} = (V, \widetilde{A}, l(\widetilde{a}), u(\widetilde{a}), k(v), s, d)$ as in [2] and set $l(\widetilde{a}) = 0$, $\forall \widetilde{a} \in \widetilde{A}$.
(3) Solve LexMSF problem on network $\widetilde{\mathcal{N}}$ using algorithm in [14].
(4) Perform flow decomposition into path and cycle flows of maximum flows obtained from step 3 and remove all cycle flows.
(5) Arc $(w, v) \in A$ is reversed if and only if the flow along arc $(v, w) \in A$ is greater than $u(v, w)$ or if there is non-negative flow along arc $a \notin A$.
(6) Obtain LexMSCF solution on $\mathcal{N}$.

Algorithm 1: LexMSCF algorithm.

(1) Given a multinetwork $\mathcal{N} = (V, A, u(a), \tau(a), s, d, T)$ with single source s, single sink d, and integer inputs.
(2) Transform $\mathcal{N}$ into undirected multinetwork $\widetilde{\mathcal{N}} = (V, \widetilde{A}, u(\widetilde{a}), \tau(\widetilde{a}), s, d, T)$ where $\widetilde{a} = (v, w) \in \widetilde{A}$, if $(v, w), (w, v) \in A$ such that $\tau(v, w) = \tau(w, v)$, with $u(\widetilde{a}) = u(v, w) + u(w, v)$ and $\tau(\widetilde{a}) = \tau(v, w)$, and $\widetilde{a} = (v, w) \in \widetilde{A}$, if $(v, w) \in A$ and $(w, v) \notin A$ such that $\tau(v, w) = \tau(w, v)$, with $u(\widetilde{a}) = u(v, w)$ and $\tau(\widetilde{a}) = \tau(v, w)$.
(3) Label parallel arcs $(v, w) \in \widetilde{\mathcal{N}}$ as $(v, w)_i$ such that $\tau(v, w)_i < \tau(v, w)_{i+1}$, for $i = 1, 2, \ldots, q; q \leq m$.
(4) Generate a dynamic, temporally repeated flow on network $\widetilde{\mathcal{N}}$.
(5) Perform flow decomposition into path and cycle flows of the flow resulting from step 4. Remove the cycle flows.
(6) Arc $(w, v) \in A$ is reversed, if and only if the flow along arc (v, w) is greater than $u(v, w)$, or if there is a non-negative flow along arc $(v, w) \in A$.
(7) Get a maximum dynamic contraflow on $\mathcal{N}$.

Algorithm 2: MDCF algorithm [8].

a unit of time. It is considered that the average speed of cars is 550 meters per minute that highly matches with the transit time to travel the segment provided by Google Maps data during normal traffic. Considering the time horizon of 60, 90, and 100 minutes and sufficient sink capacity, the results for Scenario I (Table 1) and Scenario II (Table 2) before and after the application of the contraflow approach are demonstrated.

The results show that the total maximum flow could be increased, while the contraflow approach is applied, by up to approximately 109%, justifying the importance of the approach in the evacuation planning problem. For the real-world problem, it is necessary to restrict the sink also by its actual evacuation space capacity (24090 evacuees [18]). In this case, evacuees that cannot reach sink due to its capacity are distributed among intermediate shelters respecting corresponding capacities, and more than 58 thousand (only nearly 4 thousand less) evacuees in total can be evacuated in 100 minutes for Scenario II, if contraflow is applied. Additional evacuees, about 34 thousand in number for Scenario II with actual sink capacity, can be saved due to consideration of intermediate holding of flows in the evacuation model. Detailed result discussion for this case has been omitted here.

The algorithm has been coded into Python with version 3.9.1 and was run on the computer having Windows 10 operating system with 64 GB RAM and 3.60 GHZ Intel Core i9-9900k processor. It took around 1.5, 6, and 9 minutes to run the program while computing lexicographically maximum flows for $T = 60$, $T = 90$, and $T = 100$, respectively, for Scenario I. Similarly, for Scenario II, it took around 4, 16, and 22 minutes for $T = 60$, $T = 90$, and $T = 100$, respectively. These significantly different running times for different values of parameter T and network input size justify the assertion about time complexity (pseudo-polynomial) of Algorithm 3 made in Theorem 4. The finer discretization of time (controlling parameter in algorithm), instead of considering a

(1) Given a dynamic multinetwork $\mathcal{N} = (V, A, l(a), u(a), \tau(a), k(a), s, d, T)$, $\mathcal{S} = \{v_1, \ldots, v_r\}$ with $d = v_1 \succ \cdots \succ v_r$, $l(a) = 0$ for all $a \in A$ and integer inputs.
(2) Transform $\mathcal{N}$ into undirected multinetwork $\widetilde{\mathcal{N}} = (V, \widetilde{A}, l(\widetilde{a}), u(\widetilde{a}), \tau(\widetilde{a}), k(v), s, d, T)$ as in Algorithm 2 and set $l(\widetilde{a}) = 0$, $\forall \widetilde{a} \in \widetilde{A}$.
(3) Label each parallel arcs $(v, w) \in \widetilde{\mathcal{N}}$ as $(v, w)_i$ such that $\tau(v, w)_i < \tau(v, w)_{i+1}$ for $i = 1, 2, \ldots, q$; $q < m$.
(4) Compute LexMDF on network $\widetilde{\mathcal{N}}$ using algorithm in [14].
(5) Perform flow decomposition into path and cycle flows of maximum flows obtained from step 4 and remove all cycle flows.
(6) Arc $(w, v) \in A$ is reversed if and only if the flow along arc $(v, w) \in A$ is greater than $u(v, w)$ or if there is non-negative flow along arc $a \notin A$.
(7) Obtain LexMDCF solution for multinetwork $\mathcal{N}$.

Algorithm 3: LexMDCF algorithm for multinetwork.

Table 1: Maximum flow values (evacuees) at sink (Balkhu) and intermediate vertices for Scenario I.

SN	Evacuation spaces (vertices)	Vertex capacity	Priority order	Flow/contraflow for $T = 60$	Flow/contraflow for $T = 90$	Flow/contraflow for $T = 100$
1	Balkhu (A)	∞	1	10170/13980	17370/22980	19770/25980
2	Airport (B)	5640	2	2880/5640	3780/5640	4080/5640
3	Koteshor (C)	3270	3	150/3270	150/3270	150/3270
4	Gaushala (D)	2220	4	360/2220	360/2220	360/2220
5	Satdobato (E)	7920	5	0/1710	0/7110	0/7920
6	Pulchowk (F)	2760	6	240/600	240/600	240/1590
7	Shankha Park (G)	240	7	240/240	240/240	240/240
8	Oxygenation Park (H)	3270	8	0/0	0/0	0/0
9	Balkumari (I)	4740	9	0/0	0/0	0/0
10	Teku (J)	2400	10	240/540	240/540	240/540
11	Naxal (K)	900	11	690/900	690/900	690/900
12	Jawalakhel (L)	1440	12	0/0	0/0	0/0
13	Lagankhel (M)	330	13	0/0	0/0	0/0
14	Lainchaur (N)	2130	14	330/2130	330/2130	330/2130

Table 2: Maximum flow values (evacuees) at sink (Balkhu) and intermediate vertices for Scenario II.

SN	Evacuation spaces (vertices)	Vertex capacity	Priority order	Flow/contraflow for $T = 60$	Flow/contraflow for $T = 90$	Flow/contraflow for $T = 100$
1	Balkhu (A)	∞	1	13620/23010	22620/37410	25620/42210
2	Airport (B)	5640	2	2730/5640	3630/5640	3930/5640
3	Koteshor (C)	3270	3	120/2190	120/3270	120/3270
4	Gaushala (D)	2220	4	360/2220	360/2220	360/2220
5	Satdobato (E)	7920	5	0/0	0/2520	0/3660
6	Pulchowk (F)	2760	6	150/480	150/480	150/540
7	Shankha Park (G)	240	7	240/240	240/240	240/240
8	Ring Road (Balaju NG Chowk) (H)	3270	8	690/1470	690/1470	690/1470
9	Oxygenation Park (I)	1470	9	0/0	0/0	0/0
10	Chyasal (J)	4740	10	0/0	0/0	0/0
11	Teku (K)	2400	11	210/420	210/420	210/420
12	Naxal (L)	900	12	210/900	210/900	210/900
13	Jawalakhel (M)	1440	13	0/0	0/0	0/0
14	Lagankhel (N)	330	14	0/0	0/0	0/0
15	Lainchaur (O)	2130	15	180/2130	180/2130	180/2130

minute as a unit of time, would minimize the errors associated with transit time of road segments. However, it leads to higher time complexity to run the program.

5. Conclusion

Importance and applicability of the idea of contraflow especially in the evacuation planning problem has been increasing. Existing network contraflow models fail to capture the situation where it is possible to send evacuees out from the risk zone to even an intermediate capacitated spot if they cannot reach the destination. This paper considered contraflow evacuation planning problems adopting the weak-conservation constraints that allow holding of flow units at prioritized intermediate vertices of given capacities. In particular, it proposed solution algorithms for lexicographically maximum static contraflow problem and lexicographically maximum (discrete) dynamic contraflow problem for multinetwork.

The continuous time dynamic flow model with weak-conservation constraint can also be defined in similar way as given in Section 2 by integrating the flow units at each time point $t \in [0, T)$ instead of adding flow over time for each time step $t \in \mathcal{T}$. Dynamic version of lexicographically contraflow problems studied in this paper can be extended to one with continuous time setting too. These problems can be solved with computational time complexity equal to that of discrete time setting by applying notion of natural transformation of flows suggested in [5].

The limitation of the solution algorithm proposed for dynamic version of the problem is that it leads to a pseudo-polynomial time complexity since the size of underlying network strongly depends on T. Searching of polynomial time solution algorithm for this problem as well as studying the quickest and earliest versions of the problem would be future research areas. One could incorporate the idea of intermediate holding of flows in the multicommodity flow model that better reflects the real-world vehicle distribution scenario of most underdeveloped cities.

Data Availability

The data used for the case illustration are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

Phanindra Prasad Bhandari would like to thank the University Grants Commission, Nepal, for providing partial financial support through a PhD Fellowship Award (2016) (Award No. PhD-72/73-S&T-01) to conduct this research.

References

[1] L. Wei, J. Xu, T. Lei, M. Li, X. Liu, and H. Li, "Simulation and experimental analyses of microscopic traffic characteristics under a contraflow strategy," *Applied Sciences*, vol. 9, no. 13, p. 2651, 2019.

[2] S. Rebennack, A. Arulselvan, L. Elefteriadou, and P. M. Pardalos, "Complexity analysis for maximum flow problems with arc reversals," *Journal of Combinatorial Optimization*, vol. 19, no. 2, pp. 200–216, 2010.

[3] L. R. Ford and D. R. Fulkerson, "Constructing maximal dynamic flows from static flows," *Operations Research*, vol. 6, no. 3, pp. 419–433, 1958.

[4] S. R. Khadka and P. P. Bhandari, "Dynamic network contraflow evacuation planning problem with continuous time approach," *International Journal of Operations Research*, vol. 14, no. 1, pp. 27–34, 2017.

[5] L. Fleischer and É. Tardos, "Efficient continuous-time dynamic network flow algorithms," *Operations Research Letters*, vol. 23, no. 3-5, pp. 71–80, 1998.

[6] T. N. Dhamala and U. Pyakurel, "Earliest arrival contraflow problem on series-parallel graphs," *International Journal of Operations Research*, vol. 10, no. 1, pp. 1–13, 2013.

[7] P. P. Bhandari and S. R. Khadka, "Evacuation contraflow problems with not necessarily equal transit time on anti-parallel arcs," *American Journal of Applied Mathematics*, vol. 8, no. 4, pp. 230–235, 2020.

[8] P. P. Bhandari and S. R. Khadka, *Maximum Contraflow Evacuation Planning Problems on Multi-Network*, (Submitted), 2020.

[9] R. C. Dhungana and T. N. Dhamala, "Maximum flowloc problems with network reconfiguration," *International Journal of Operations Research*, vol. 16, no. 1, pp. 13–26, 2019.

[10] R. C. Dhungana, U. Pyakurel, and T. N. Dhamala, "Abstract contraflow models and solution procedures for evacuation planning," *Journal of Mathematics Research*, vol. 10, no. 4, pp. 89–100, 2018.

[11] U. Pyakurel, H. N. Nath, and T. N. Dhamala, "Partial contraflow with path reversals for evacuation planning," *Annals of Operations Research*, vol. 283, no. 1-2, pp. 591–612, 2019.

[12] T. N. Dhamala, U. Pyakurel, and S. Dempe, "A critical survey on the network optimization algorithms for evacuation planning problems," *International Journal of Operations Research*, vol. 15, no. 3, pp. 101–133, 2018.

[13] M. Skutella, "An introduction to network flows over time," in *Research Trends in Combinatorial Optimization*Springer, Berlin, Heidelberg, 2009.

[14] P. P. Bhandari, S. R. Khadka, S. Ruzika, and L. E. Schäfer, "Lexicographically maximum dynamic flow with vertex capacities," *Journal of Mathematics and Statistics*, vol. 16, no. 1, pp. 142–147, 2020.

[15] P. P. Bhandari and S. R. Khadka, "Maximum flow evacuation planning problem with non-conservation flow constraints at the intermediate nodes," in *Proceedings of International Conference on Mathematical Optimization*, Beijing, China, April 2019.

[16] R. K. Ahuja, T. L. Magnati, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, Upper Saddle River, NJ, USA, 1993.

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

[18] International Organization for Migration, *Report on Identification of Open Spaces for Humanitarian Purposes in Kathmandu Valley*, https://reliefweb.int/report/nepal/report-identification-open-spaces-humanitarian-purposes-kathmandu-valley, International Organization for Migration, Grand-Saconnex, Switzerland, 2011.