Urmila Pyakurel, Stephan Dempe, and Tanka Nath Dhamala

Efficient Algorithms for Flow over Time Evacuation Planning Problems with Lane Reversal Strategy

Urmila Pyakurel, Stephan Dempe, and Tanka Nath Dhamala

Efficient Algorithms for Flow over Time Evacuation Planning Problems with Lane Reversal Strategy

TU Bergakademie Freiberg

Fakultät für Mathematik und Informatik

Prüferstraße 9

09599 FREIBERG

http://tu-freiberg.de/fakult1

EFFICIENT ALGORITHMS FOR FLOW OVER TIME EVACUATION PLANNING PROBLEMS WITH LANE REVERSAL STRATEGY

URMILA PYAKUREL[1], STEPHAN DEMPE[2],TANKA NATH DHAMALA[3]

ABSTRACT. The contraflow techniques have widely been effective in evacuation planning research. We present efficient algorithms to solve the evacuation network flow problems, namely, the maximum, earliest arrival, quickest and lex-maximum dynamic contraflow problems having constant attributes and their generalizations with partial contraflow reconfiguration. Moreover, the contraflow models with inflow dependent and load dependent transit times are introduced and presented strongly polynomial time algorithms to compute approximation solutions of the corresponding quickest contraflow problems on two terminal networks with partial reversals of arc capacities. Our results on partial lane reversals should be quite relevant for reducing evacuation time and supporting logistics in emergencies.

1. INTRODUCTION

An existence of quite many predictable and unpredictable large scale disasters worldwide, regardless of natural, various discoveries and urbanization or terrorist created, an efficient, implementable and reliable evacuation planning is most essential to save life and to support humanitarian relief with optimal use and equitable distribution of available resources. Among the earthquakes, volcanic eruptions, landslides, floods, tsunamis, hurricanes, typhoons, chemical explosions and terrorist attacks, the most remarkable losses are noticed by: April 2015 Nepal, March 2011 Japan, Haiti, Chichi, Bam, Kashmir and Chile earthquakes, the tsunami in Japan and the Indian Ocean, the major hurricanes Katrina, Rita and Sandy in USA, and the September 11 attacks in USA. Unfortunately, 1255 Nepal earthquake already killed 25-33 percentage population of its capital city and around 8 Richter scale earthquake of 1934 in its territory was the most devastating earthquake. There are threatened for the greater earthquakes more than 8.4 in Richter scale around the region in future too, Pyakurel [42] . However, a reasonable emergency plan is still lacking around the world including Nepal.

Along with diversified fields of mathematical research in resolving these planning issues, operations research methodologies are the most efficient and unavoidable. An evacuation optimizer looks after a plan on evacuation network for an efficient

transfer of maximum evacuees from the dangerous (sources) to safer (sinks) locations as quickly as possible. An optimal shelter location and support of humanitarian logistics within these emergency scenarios are equally demanding but challenging issues. Our interest is to look on the transportation planning that has strong mathematical based model with wide real world applications not only in the emergency situation but also in the rush hours of highly populated city as well. The comprehensive explanation on diversified theories and applications can be found in the survey papers of Hamacher and Tjandra [18], Cova and Johnson [7], Altay and Green III [1], Pascoal et al. [35], Moriarty et al. [34], Chen and Miller-Hooks [6], Yusoff et al. [52], Dhamala [8] and Kotsireas et al. [27], and the literatures therein.

The evacuation network is represented as a dynamic network that corresponds to a region (or a building, a shopping mall) to be evacuated in which the street-intersections (or rooms) represent the nodes and the connections between these parts (i.e., streets in region, or doors between rooms) denote the edges. The initial unsafe locations of evacuees are the source nodes and the locations at safety regions are the sink nodes. The nodes and edges are bounded by capacities. Each arc has a transit time or a cost function. The group of evacuees that passes through the network over time is modeled as a flow. An evacuation plan is heavily dependent upon the number of sources, sinks, parameters on the arcs and nodes, like constant, time-dependent or flow dependent capacities or transit times as well as additional constraints. The time may be discrete or continuous. The discrete time steps approximate the computationally heavy continuous models at the cost of solution approximations. Also the constant time probably approximated by free flow speeds or certain queuing rules and constant capacity settings mostly realize the evacuation problems to be linear, at least more tractable, in contract to the more general and realistic flow-dependent real-world evacuation scenarios.

Following the pioneer foundation of Ford and Fulkerson [13, 14] with an objective of maximizing maximum flow from a source to a sink at the end of given discrete time period, Gale [15] shows an existence of the maximum flow from the very beginning in discrete time setting. Two pseudo-polynomial time algorithms are presented by Wilkinson [51] and Minieka [33] for the latter problem with constant arc transit times. An upward scaling approximation algorithm of Hoppe [20] polynomially solves this problem within a factor of $1 + \epsilon$, for every $\epsilon > 0$. For the special serial-parallel networks, Ruzika et al. solve this problem applying a minimum cost circulation flow algorithm by exploiting a property that every cycle in its residual network has non-negative cycle length. Minieka [33], and Hoppe and Tardos [21] maximize the flow in priority ordering which is important in some scenarios of evacuation planning. Burkard et al. [5], and Hoppe and Tardos [21] present efficient algorithms for shifting the already fixed evacuees in minimum time. However, the general multi-terminal evacuation problems with variable number of evacuees at sources are computationally hard even with constant attributes on arcs. Likewise, the earliest arrival transshipment solutions that fulfill the specific demands at sinks by the specific supplies from sources maximizing at each point of time are also not solved in polynomial time yet. The multi-source single-sink (cf. Baumann and Skutella [4]) and also zero transit times either one source or one sink (cf. Fleischer [10]) earliest arrival transshipment problems have polynomial time solutions. Gross et al. [16] and Kappmeier [23] propose efficient algorithms that compute the approximate earliest arrival flows on arbitrary networks. Several works have been

done with continuous time settings as well (see, Pyakurel and Dhamala [41] for the references). Several polynomial time algorithms by natural transformations are obtained by Fleischer and Tardos [12].

During or after disastrous situations, the evacuation planer discourage people to move towards risk areas from safer places because of which the corresponding road lanes are unoccupied. However, the lanes outwards from sources become more congested due to large number of evacuees and vehicles on the streets. The optimal lane reversal strategy plans make the traffic systematic and smooth by removing the traffic jams caused by different large scale natural and man-made disasters, busy office hours, special events and street demonstrations. The contraflow reconfiguration, by means of various operational research models, heuristics, optimization and simulation techniques, reverses the idle direction of empty lanes towards sinks satisfying the given constraints that increase the flow value, decrease the average evacuation time and save some lanes with excess capacities for an use of emergency vehicles and logistic supports needed to move towards the sources. However, an efficient and universally acceptable solution approach that meets the macroscopic and microscopic behavioral characteristics is still lacking. A tradeoff between computational costs and solution quality should be a compromise.

Arguing on the fact that computational costs of exact mathematical solutions for general contraflow techniques being quite high, a series of heuristic procedures are approached in literature that are computationally manageable. Kim et al. [26] present two greedy and bottleneck heuristics for possible numerical approximate solutions to the quickest contraflow problem and show that at least 40% evacuation time can be reduced by reverting at most 30% arcs. They model the problem of lane reversals mathematically as an integer programming problem by means of flows on network and also prove that it is $\mathcal{NP}$-hard. All contraflow evacuation problems are at least harder than the corresponding problems without contraflow. We recommend Dhamala [8] and the references therein for partial approaches of contraflow heuristics.

Though comparatively less, recent interest also includes analytical techniques after Rebennack et al. [45] solved the single-source single-sink maximum contraflow and quickest contraflow problems optimally in polynomial times. The earliest arrival and the maximum contraflow problems are solved with the temporally repeated solutions in Dhamala and Pyakurel [9, 37]. Its continuous solution is obtained in [39]. In [38], authors solved the earliest arrival contraflow on single-source single-sink network in pseudo-polynomial time. They also introduced the lex-maximum dynamic contraflow problem in which flow is maximized in given priority ordering and solved it in polynomial time complexity. These problems are also solved in continuous time by using nice property of natural transformation in [39, 41]. With given supplies and demands, the earliest arrival transshipment contraflow problem is modeled in discrete time [40] and solved it on multi-source network with polynomial algorithm. Taking zero transit times on each arc, the problem is also solved on multi-sink network in polynomial time complexity. For the multi terminal network, they presented approximation algorithms to solve the earliest arrival transshipment contraflow problem. The discrete solutions are extended into continuous time in [41, 42].

Although these restrictions violate the first-in-first-out (FIFO) property and do not capture the more realistic behavior of the flow over time with flow-dependent

times, we give approximate results with polynomial time algorithms comparable to those on quickest flows over time with load dependent transit times. Our proposed algorithms also summarize the earlier results on contraflow in compact form. The main strategy of this work is to save the unused capacities of arcs using partial contraflow techniques which can be used for logistic supports in emergencies and locate the possible facilitates on arcs.

The organization of this paper is as follows. Section 2 presents the basic terminology necessary to the paper and different flow models. All the previously solved dynamic contraflow problems with constant transit time are extended in the partial contraflow configuration and solved with efficient algorithms in Section 3. Section 4 introduces the partial contraflow approach to solve the dynamic flow problems with inflow dependent and load dependent transit times presenting efficient algorithms. The paper is concluded with Section 5.

2. Basic terminology

An evacuation network is represented either in nodes-arcs form or in elements-paths form. The first model is represented by $\mathcal{N} = (V, A, b, \tau, S, D, T)$, where V is the set of n nodes, A is the set of m arcs with sources nodes $s \in S$ and sink nodes $d \in D$. However the latter model, so-called an abstract dynamic network, is represented by $\mathcal{N} = (A, \mathcal{P}, b, \tau, S, D, T)$, where A denotes the set of elements and the set of paths $\mathcal{P}$ contains both s-d and d-s paths. The nodes may be equipped with the initial occupancy $o : V \rightarrow R^+$. The predefined parameter T denotes a permissible time window within which the whole evacuation process has to be completed. It may be discretized into discrete time steps $\mathbf{T} = \{0, 1, \ldots, T\}$ or can be considered a continuous one as $\mathbf{T} = [0, T]$. We assume that no arcs enter the source node and no arcs exist the sink node so that $B_s = \emptyset$ and $A_d = \emptyset$ holds, where $B_v = \{e \mid e = (u, v) \in A\}$ and $A_v = \{e \mid e = (v, u) \in A\}$.

The upper capacity (bound) function $b : A \times T \rightarrow R^+$ on arcs limits the flow rate passing along the arcs for each point in time. The transit time function $\tau : A \times T \rightarrow R^+$ measures the amount of time it takes for the flow to travel along the arcs. We frame this work with constant, inflow-dependent and load-dependent transit times on arcs. With inflow-dependent transit times, the transit time $\tau_e(x_e(\theta))$ is a function of inflow rate $x_e(\theta)$ on the arc e at given time point θ, so that at a time flow units enter an arc with the uniform speed and remains the uniform speed through traveling this arc. Whereas with load-dependent transit times, the transit time $\tau_e(l_e(\theta))$ depends on the total amount of flow on an arc e at a given time θ, i.e., the load $l_e(\theta)$ on e. In this approach all units of flow on an arc enjoy the same speed and the transit time of an arc varies with each unit of flow entering or leaving the arc with the continuous changes of flow on this arc.

The flow rate function is defined by $x : A \times T \rightarrow R^+$, where $x_e(\theta)$ denotes the flow rate on e at time θ. It may be taken as an inflow, outflow and intermediate flow rates, that measure the flow at entry, exit and intermediate points on an arc, respectively. For $\theta \in \{0, 1, \ldots, T\}$ and constant function τ, the amount of flow sent at time θ into e arrives to its end at time $\theta + \tau_e$. Whereas for continuous time $\theta \in [0, T]$ and constant function τ, the amount of flow per time unit enters at this rate e at time θ and proceed continuously.

We introduce an additional parameter $\lambda_e \in R^+$ on arc e, say a gain factor, in order to model a generalized dynamic flow when only λ_e units of flow leave from w

at time $\theta + \tau_e$ by entering a unit of flow on $e = (v, w)$ at time θ. If the flow is only lost but never gained along all arcs, then $\lambda_e \leq 1$ holds for each arc $e \in A$, and we call the network as lossy. A a lossy network is denoted by $\mathcal{N} = (V, A, b, \tau, \lambda, S, D, T)$.
Flow models. Let a non-negative function $y : A \rightarrow \mathcal{R}^+$ represents the static flow. Neglecting the cost on each arc, a static s-d flow y of value $val(y)$ in objective (2.1) satisfies the flow conservation and capacity constraints (2.2) and (2.3), respectively.

$$(2.1) \qquad val(y) = \sum_{e \in B_d} y_e = \sum_{e \in A_s} y_e$$

$$(2.2) \qquad \sum_{e \in B_v} y_e - \sum_{e \in A_v} y_e = 0, \qquad \forall\, v \in V \backslash \{s, d\}$$

$$(2.3) \qquad 0 \leq y_e \leq b_e, \qquad \forall\, e \in A$$

The maximum static flow (MSF) maximizes objective (2.1). If the flow conservation also satisfies at the terminals of a network, then the network flow is called circulation. Any flow y with value $val(y)$ can be transformed into a static flow with value zero by adding an extra arc (d, s) with value $val(y)$ through it. Such a flow with value zero is a zero circulation. If the cost is included, then we have to minimize the cost. Fixing the flow value $val(y)$, a minimum cost static flow problem seeks to shift the given flow with minimum cost $\sum_{e \in A} c_e y_e$. A minimum cost static flow problem with zero circulation turns into a minimum cost circulation flow (MCCF) problem. We denote the maximum static flow value by $val_{max}(y)$.

Let us assume that the arc travel times and capacities are constant over the time. With an amount of inflow $x_e(\theta)$ on arc e at discrete time $\theta = 0, 1, \ldots, T$ that may change over the planning horizon T, the generalized dynamic flow (GDF) $x : A \times T \rightarrow R^+$ for given time T satisfies the constraints (2.4-2.6).

$$(2.4) \qquad \sum_{\sigma = \tau_e}^{T} \sum_{e \in B_v} \lambda_e x_e(\sigma - \tau_e) = \sum_{\sigma = 0}^{T} \sum_{e \in A_v} x_e(\sigma), \; \forall\, v \notin \{s, d\}$$

$$(2.5) \qquad \sum_{\sigma = \tau_e}^{\theta} \sum_{e \in B_v} \lambda_e x_e(\sigma - \tau_e) \geq \sum_{\sigma = 0}^{\theta} \sum_{e \in A_v} x_e(\sigma), \; \forall\, v \notin \{s, d\}, \; \theta \in \mathbf{T}$$

$$(2.6) \qquad b_e(\theta) \geq \lambda_e x_e(\theta) \geq 0, \; \forall\, e \in A, \; \theta \in \mathbf{T}$$

The generalized earliest arrival flow problem (GEAFP) is to find a GDF of maximum value $\text{val}_{\max}(x_e, \theta)$ for all time units $\theta \in \{0, 1, \ldots, T\}$ defined by objective function (2.7). It is defined as a generalized maximum dynamic flow problem (GMDFP) if the maximization is considered for $\theta = T$.

$$(2.7) \qquad \text{val}(x, \theta) = \sum_{e \in B_d} \sum_{\sigma = \tau_e}^{\theta} \lambda_e x_e(\sigma - \tau_e), \; \theta = 0, 1, \ldots, T$$

Note that no flow remains in the dynamic network at time T. It is ensured by assuming that $x_e(\theta) = 0$ for all $\theta \geq T - \tau_e$.

For the following models we make an assumption that the gain factor $\lambda = 1$. Then the GDF reduces to the dynamic flow and the GEAFP problem reduces to the earliest arrival flow problem (EAFP) with objective function (2.8). We denote the maximum dynamic flow problem by MDFP.

$$(2.8) \qquad \text{val}(x, \theta) = \sum_{\sigma = 0}^{\theta} \sum_{e \in A(s)} x_e(\sigma) = \sum_{\sigma = \tau_e}^{\theta} \sum_{e \in B(d)} x_e(\sigma - \tau_e)$$

For a given value Q_0, the quickest flow problem (QFP) looks for the minimal time $\min T = T(Q_0)$ such that the flow value is at least Q_0 satisfying the constraints (2.4-2.6) with equality in (2.5) and $\lambda = 1$. For a given time horizon T and an ordered set of multi-terminals, the lex-maximum dynamic flow problem (LMDFP) finds a feasible flow that lexicographically maximizes the amount leaving (entering) each terminal in the given priority order.

Let $\mathcal{N} = (V, A, b, \tau, S, D, \mu(s), \mu(d))$ be a multi-terminal network with a source-supply and sink-demand vectors $\mu(s)$ and $\mu(d)$, respectively, such that $\mu(S \cup D) = \sum_{v \in S \cup D} \mu(v) = 0$. The multi-terminal EAFP sends the total supply $\mu(S) = \sum_{s \in S} \mu(s)$ from S to the total demand $\mu(D) = \sum_{d \in D} \mu(d)$ in D with maximum value at each point of time $\theta > 0$. If all demands be fulfilled with supplies by shifted them within given time T, then the problem turns into the transshipment problem. The earliest arrival transshipment problem (EATP) maximizes $val(x, \theta)$ in objective function (2.9) satisfying the multi-terminal constraints (2.4-2.6) for all time points $\theta \in \{0, 1, \ldots, T\}$ with $\lambda = 1$.

$$(2.9) \qquad \mathrm{val}(x, \theta) = \sum_{\sigma=0}^{\theta} \sum_{e \in A(s):s \in S} x_e(\sigma) = \sum_{\sigma=\tau_e}^{\theta} \sum_{e \in B(d):d \in D} x_e(\sigma - \tau_e)$$

If all supplies are transshipped from S to D satisfying its demands in minimum time $\min T = T(\mu(S \cup D))$, then the problem is called quickest transshipment problem (QTP).

With continuous time settings as $\mathbf{T} = [0, T]$ all of the above models can be remodeled with replacing the summation over time with respective integrals. The amount of flows entry x_e on arcs considered above in discrete models naturally transfer to the entry flow rates in this continuous approach.

Natural transformation. Fleischer and Tardos [12] connect the continuous and discrete flow models by the following natural transformation, as defined in (2.10), that deals with the same computational complexity to both.

$$(2.10) \qquad x_e^c(\psi) = x_e(\theta), \text{ for all } \theta \text{ and } \psi \text{ with } \theta \leq \psi < \theta + 1$$

where $x_e(\theta)$ is the amount of discrete dynamic flow that enters arc e at time $\theta = 0, 1, \ldots, T$ with constant capacities on the arcs. For static flow y_e on arc e, the amount of discrete dynamic flow with travel time τ_e on arc e is

$$(2.11) \qquad x_e(\theta) = \sum_{\sigma=0}^{\tau(e)-1} y_e(\theta - \sigma), \text{ for all } \theta = 0, 1, \ldots, T-1$$

Notice that the flow which enters an arc e at time $\theta - \tau_e$ arrives at the head node at time θ in discrete time, but at time $[\theta + 1)$ in continuous time. Then the flow x^c is feasible and the amount of source-sink flow at any integer time interval $[\theta, \theta + k)$, for $\theta = 0, 1, \ldots, T$, $k \in N$, will be the same for both settings.

3. Lane reversals with constant attributes

This section deals with the concepts of dynamic, generalized dynamic and abstract dynamic contraflow techniques and propose some efficient algorithms for their solutions with constant travel times and capacities on arcs.

Models for arc reversals. Let the reversal of an arc $e = (v, w)$ be $e' = (w, v)$. Then, accounting the capacities and transit times with reversal of arcs on given network $\mathcal{N} = (V, A, b, \tau, S, D, T)$ creates a reconfigured network $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, S, D, T)$ with modified arc capacities $b_{\overline{e}}$ and transit times $\tau_{\overline{e}}$ as

$$b_{\overline{e}} = b_e + b_{e'}, \quad \text{and} \quad \tau_{\overline{e}} = \begin{cases} \tau_e, & \text{if } e \in A \\ \tau_{e'}, & \text{otherwise} \end{cases}$$

where, an edge $\overline{e} \in E$ in $\overline{\mathcal{N}}$ if $e \vee e' \in A$ in $\mathcal{N}$. The remaining graph structure and data are unaltered. By discarding the time factor, a satic contraflow configuration will be defined analogously.

Example 1. The reconfigured network of Figure 1(a) is Figure 1(b) which is obtained by adding two way arc capacities and taking the same transit times. Arc (s, y) has capacity 7 and transit time 4. If we assume that a time unit is 5 minutes and a flow unit is 10, it takes 20 minutes to travel (s, y) and maximum 70 flow can simultaneously travel through. Here, it is randomly fixed in one direction and checked whether the flow increases or not as in Figures 1 (b) and 1(c). Either direction may yield time dependent results. A decision of arc orientations at intersecting paths makes the problem complicated and eventually $\mathcal{NP}$-hard, [45].
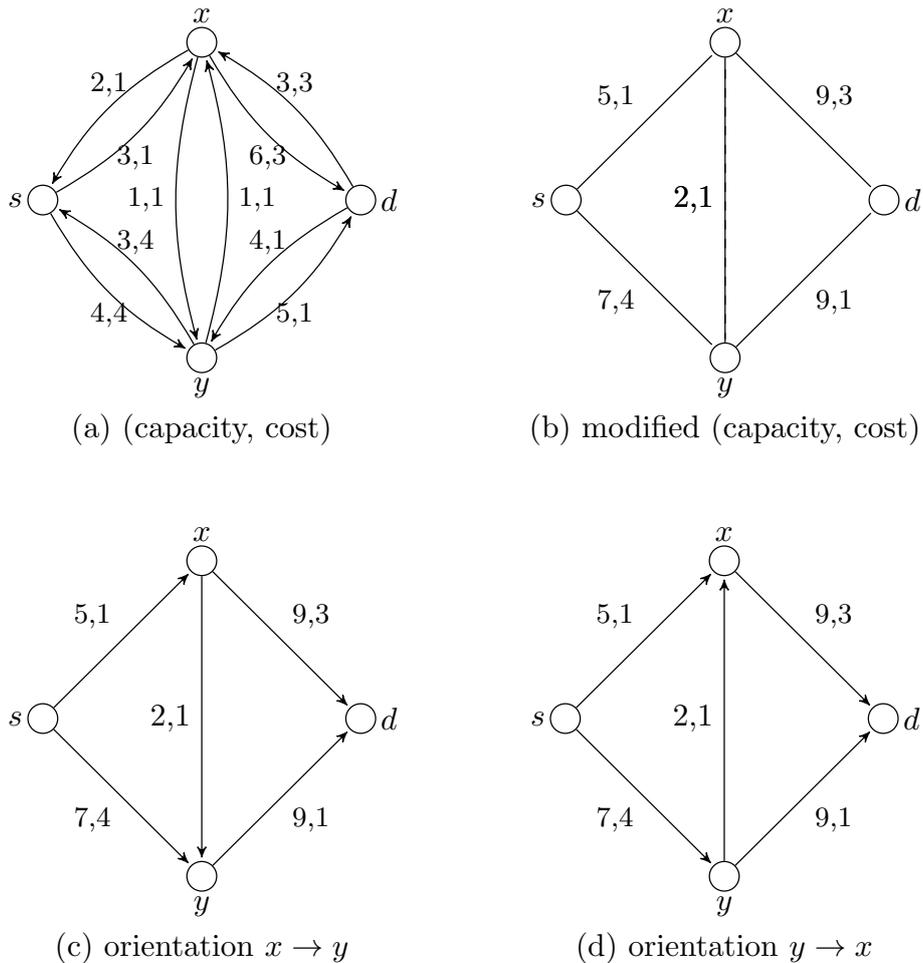

(a) (capacity, cost)

(b) modified (capacity, cost)

(c) orientation $x \to y$

(d) orientation $y \to x$

FIGURE 1. (b) Reconfigured network and the orientations towards (c) node y, and (d) node x

3.1. **Dynamic contraflow.**

Problem 1. *For given network $\mathcal{N} = (V, A, b, \tau, S, D, T)$ with integer inputs (additional supplies and demands), the dynamic contraflow Problem (DCFP) $\boldsymbol{P}$ is to find a dynamic (respectively, transshipment) S-D flow for all time θ, $0 \leq \theta \leq T$ with arc reversal capability.*

As there is no algorithm to find the temporally repeated flows on general S-D dynamic network, an exact optimal dynamic contraflow solution on general network has not been found polynomially yet. In particular, on the networks like two-terminals s-d, priority based S-D, transshipments S-d, s-D and S-D networks with different constraints, the optimal static flow is decomposable into chains (paths) that are temporally repeated over given time horizon T, yielding the dynamic flow. Thus, on reconfigured network $\overline{\mathcal{N}}$, any technique computing a temporally repeated flow is applicable to find an optimal solution to the dynamic contraflow problem (DCFP) $\boldsymbol{P}$ for original network $\mathcal{N}$. This can be achieved by applying Step 3 of Algorithm 1. Notice that the DCFP $\boldsymbol{P}$ may represent one of the maximum dynamic contraflow (MDCF), earliest arrival contraflow (EACF) and lex-maximum dynamic contraflow (LMDCF) problems. For given supplies and demands on sources and sinks, the DCFP $\boldsymbol{P}$ also represents the transshipment contraflow problem. Once a dynamic flow is obtained, it should be subtracted from reconfigured capacities of arcs to record the maximum arc capacities that are not used to increase flow value as in Step 4. During the computation of temporally repeated flows, remove the cycle flows as in Step 5, if there exist, so that the simultaneous flows in both directions are not possible. Thus, a flow is either along arc e or arc e' in Step 6 and its value is not greater than the reversed capacities on all arcs at all time units. Hence, the condition of feasibility is satisfied by the algorithm.

Algorithm 1. Dynamic contraflow algorithm (DCFA)

 (1) **Input.** A Dynamic network $\mathcal{N} = (V, A, b, \tau, s, d, T)$ with constant and symmetric transit time, i.e., $\tau_e = \tau_{e'}$.
 (2) Construct the reconfigured network $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, s, d, T)$ of $\mathcal{N}$ using contraflow configuration.
 (3) Use a temporally repeated flow algorithm to solve DCFP $\boldsymbol{P}$ on reconfigured network $\overline{\mathcal{N}}$ with capacity $b_{\overline{e}}$ and transit time $\tau_{\overline{e}}$ for each $\overline{e} \in E$.
 (4) Record the arc capacities as $\tilde{b}_{e \vee e'} = b_{\overline{e}} - x_{\overline{e}}$ for all $e \in A$.
 (5) Decompose the flow into different paths and removable cycles.
 (6) Reverse arc $e' \in A$ if and only if $b_e < x_e$ along the arc $e \in A$ or if $x_e > 0$ along the arc $e \notin A$.
 (7) **Output.** A dynamic contraflow x with the partial arc reversals.

From the series of literature on analytical contraflow approach [36, 37, 38, 40, 39, 41, 42, 43, 44, 45], it is conformed that the optimal DCF solution for the given network $\mathcal{N} = (V, E, b, \tau, s, d, T)$ is equivalent to the optimal dynamic flow solution on corresponding reconfigured network $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, s, d, T)$. In the notation, the input network may be a super-source super-sink added extended network if one is considering a multi-terminal network with additional restrictions whenever a temporally repeated solution is possible. Our DCF algorithm does not change the optimal DCF solution but it saves maximum arc capacities simultaneously. For example, by considering only a single $s - x - d$ path in Figure 1 (a), a maximum

possible flow alone arc (s, d) is 5 units at a time unit θ with contraflow configuration but its original capacity is 6 units without reversal of arc (d, x). Our algorithm records the exceeding one unit capacity alone arc (x, d) together with 3 units capacity of (d, x) and allows it for logistic and emergency supports for evacuees. The complexity of this algorithm is similar to the applied ones. It can be summarized as

Theorem 3.1. *Any dynamic contraflow on network* $\mathcal{N} = (V, A, b, \tau, s, d, T)$ *is equivalent to a dynamic flow on its reconfigured network* $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, s, d, T)$.

3.1.1. *Maximum dynamic contraflow.* If the flow has to be maximized for given time period T without considering earlier periods, then Problem 1 with the arcs permissible to be reversible only at time zero is the MDCFP. Polynomial time algorithms to solve the *S-D* MDCFP has not been investigated yet. However, it can be solved with pseudo-polynomial time by reducing it into extended *s-d* network of its time expanded network. The *s-d* MDCFP is polynomially solvable in discrete time, [45] and in continuous time, [39]. Recall that, their contraflow configuration is complete so that there may be unnecessary arc reversals at the end. For example, arc e' has 3 lanes and one needs only single lane to be reversed for optimality, but authors in [45, 39] reverse all three lanes. In this work, we obtain the optimal *s-d* MDCF solution and record the capacities of unused lanes simultaneously in polynomial time. Based on the result of [45, 39], we use the temporally repeated flow algorithm of [14] in Step 3 of Algorithm 1 that gives the MDF solution on reconfigured network $\overline{\mathcal{N}}$ which is equivalent to the MDCF solution on $\mathcal{N}$ and records unused arc capacities using Step 4. Thus, we state Theorem 3.2.

Theorem 3.2. *The s-d MDCFP saving the unused arc capacities can be solved in* $O(h_1(n, m) + h_2(n, m))$ *time in which flow decomposition and minimum cost flow problems are solved in* $h_1(n, m) = O(n.m)$ *and* $h_2(n, m) = O(n^2.m^3.\log n)$ *times, respectively.*

Proof. Theorem 3.2 is proved in two steps. First we show that Algorithm 1, using MDF algorithm of [14] in Step 3 is feasible. Steps 2-5 of the algorithm is feasble. With flow decomposition, as there is no any cycles carrying positive flow, there is either a flow along arc e or e' but never in both arcs in Step 6. Thus, flow is not greater than the reversed capacities on all the arcs at all time units.

Second, we show the optimality. Feasibility of the algorithm implies that every feasible MDF in the reconfigured network $\overline{\mathcal{N}}$ is feasible to the MDCF solution in the original network $\mathcal{N}$, i.e.,

$$(3.1) \qquad [\mathcal{N}]_{MDCF_{opt}} \geq [\overline{\mathcal{N}}]_{MDF_{opt}}.$$

According to [45, 41], we have

$$[\mathcal{N}]_{MDCF_{opt}} \leq [\mathcal{N}(T)]_{MSCF_{opt}}.$$

Moreover, construct the reconfigured network of $\mathcal{N}(T)$ as $\overline{\mathcal{N}}(T) = (V_T, \overline{E}_T)$ with arc set $\overline{E}_T$ defined as $\tilde{e} \in \overline{E}_T$, if $\overline{e} \in E_T$ or $\overline{e}' \in E_T$, the capacity function $b(\tilde{e})$ is given by $b^{\overline{\tau}}(\tilde{e}) = b^{\overline{\tau}}(\overline{e}) + b^{\overline{\tau}}(\overline{e}')$. Then, we have

$$[\mathcal{N}(T)]_{MSCF_{opt}} = [\overline{\mathcal{N}}(T)]_{MSF_{opt}}.$$

The minimum cost circulation flow algorithm of [14] determines the MDF x in $\overline{\mathcal{N}}$ with a temporally repeated flow of obtained maximum static flow in $\overline{\mathcal{N}}(T)$.

After computing the MDF x on $\overline{\mathcal{N}}$, we save the unused arc capacities of arcs as $\tilde{b}_{e\vee e'} = b_{\overline{e}} - x_{\overline{e}}$ for all $e \in A$, and thus, it does not effect the optimal solution for the MDCFP on $\mathcal{N}$. This implies that

$$[\mathcal{N}]_{MDCF_{opt}} \leq [\overline{\mathcal{N}}(T)]_{MSF_{opt}} = [\overline{\mathcal{N}}]_{MDF_{opt}}.$$

Combining inequality (3.1) with the preceding result proves the optimality of the algorithm.

At last, the complexity of the algorithm is dominated by the complexity of minimum cost flow computation in Step 3 and flow decomposition in Step 5. This completes the proof.   □

Example 2. Figure 2(a) represents the residual network of reconfigured network in Figure 1(a) with given static flow 12, where, there exists no augmenting path yielding the static flow value more than 12. Then the static paths $P_1 = s - x - d$ and $P_2 = s - y - d$ with flows 5 and 7, respectively, are temporally repeated to get the MDF on directed reconfigured network as follows. The maximum static flow (MSF) alone arcs (s, x) and (x, d) is 5, and alone arcs (s, y) and (y, d) is 7. By subtracting the static flows from the capacities of respective arcs, we get the unused arc capacities as $(x, d) = 9 - 5 = 4$ and $(y, d) = 9 - 7 = 2$ which are allowed to flow alone opposite direction (d, x) and (d, y) whenever required for other purposes. As, arc (x, y) and (y, x) are not used in the static paths, their capacities are unturned as in Figure 2(b).

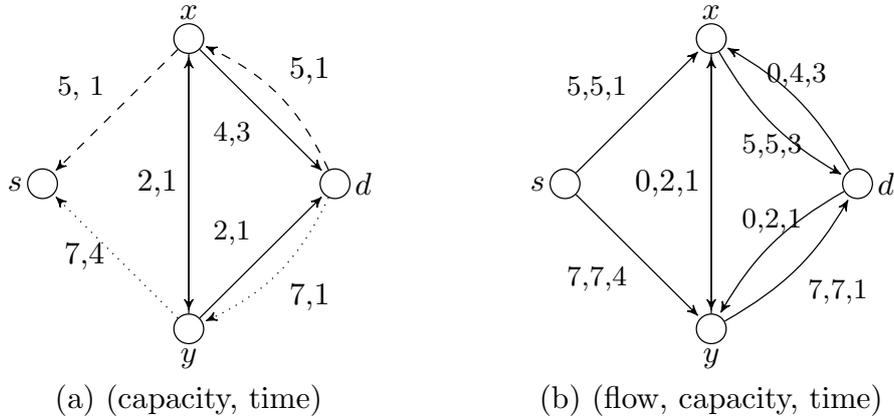(a) (capacity, time)   (b) (flow, capacity, time)

FIGURE 2. (a) The residual network of Figure 1(a), and (b) A maximum contraflow solution

Notice that the residual of any undirected network is a directed network Schroeder et al. [50]. We discussed an algorithm based on residual network that gives the solution for the partial contraflow configuration. The algorithm works as follows. First, construct the residual network of reconfigured network $\overline{\mathcal{N}}$ assuming the transit time as cost on each arc. Second, find all augmenting paths from source to sink. At last, when there does not exist any augmenting paths for s-d flows in residual network, record the remaining residual capacities on all arcs that are unused arc capacities to increase the flow value and can be used for emergency support. Thus, we can reverse only the arc capacities contained on s-d augmenting paths of residual network, and this gives the partial contraflow configuration. This can be verified from Figure 2 (a) in which only two augmenting paths $P_1 = s - x - d$ and $P_2 = s - y - d$ are

constructed. Then, the remaining residual capacities of arcs (x, d), (y, d) and (x, y) or (y, x) are 4, 2 and 2, respectively which are not necessary to reverse towards the sink. Moreover, saving these arc capacities does not increase the complexity of the algorithm.

We claim another algorithm for the partial contraflow configuration that is based on the minimum cut problem. The formation of reconfigured network is from weighted undirected network (two-ways network). On the weighted undirected network, a minimum s-d cut can be obtained in $O(n^2 \log^3 n)$ time complexity that is independent with maximum flows computations, Karger and Stein [25]. The partial contraflow configuration can be achieved by reversing only the capacities of arcs that are equal to the minimum cut capacities. However, there are some complications in which it is difficult to identify the used capacities of other arcs that are not contained on minimum cut set. We can say that maximum flow is equal to the minimum cut but there is no any technique developed to decompose the maximum flow value in different s-d paths in undirected network given a minimum cut solution. If any exists, then the partial contraflow configuration is not harder than the distribution of maximum flow value in different arcs and paths of the network.

3.1.2. *Earliest arrival contraflow.* The earliest arrival contraflow problem (EACFP) on dynamic network $\mathcal{N} = (V, A, b, \tau, S, D, T)$ is to find a dynamic S-D flow which is maximum for all time steps $\theta = 0, 1, \ldots, T$, with arc reversal capability.

As the S-D MDCFP is NP-hard, the S-D EACFP is also NP-hard. Even the s-d EACFP is not solvable polynomially with arc reversals at time zero, however, it is possible on particular s-d series parallel network, [9, 37, 39]. In this particular case, the earliest arrival property is satisfied by temporally repeated MDCF solutions. The s-d EACFP should find the shortest distance paths at all successive time points and each paths should be repeated with its flows yielding the optimal solution in polynomial time. Thus, a flow through the arc (x, y) is essential due to the shortest path $s - x - y - d$ in Figure 2 (a) at the beginning with s-d maximum flow value 2 at time 3. However, it is not yet guaranteed whether this orientation remains valid for the latter time points and the path repeats at each time point. A complication for the s-d EACF solution arises because of the flipping requirements of such intermediate arcs with respect to the time. However, as in [36, 39], an approximate s-d EACF within $(1 - \epsilon)$ of the optimal EACF has been obtained in polynomial time. For this we run the fully polynomial time approximation algorithm of [20] on Step 1 of Algorithm 1 and obtain the approximate s-d EACF solution in $O(m\epsilon^{-1}(m + n\log n)\log U)$ time, where $U = max_{\overline{e} \in E} b(\overline{e})$ and also save unused arc capacities using Step 4.

When the network $\mathcal{N}$ in Algorithm 1 is reduced to s-d series parallel network and apply the MCCF algorithm of [46] in Step 3, the strongly polynomial solution with giving temporally repeated flow and saving unused arc capacities is obtained for the s-d EACFP. The main advantage in series-parallel graphs is that each cycle in the residual network has nonnegative cycle length.

Theorem 3.3. *On s-d series parallel network, the EACF solution can be computed in $O(nm + m\log m)$ time complexity by reversing only partial arc capacities at time zero.*

Recall that the EAFP continues the already obtained flows in earlier steps to forthcoming flows in forward steps, the final solution may change the direction of arcs and obeys the backward flow laws in its processing. Thus, with the relaxation of arc reversal time to time, the *s-d* EACF can be computed in pseudo-polynomial time, [38, 41]. Algorithm 1 not only computes the same result in which the EAF is computed in Step 3 using chain decomposable flow of Hoppe [20] that is similar to the successive shortest path algorithm of Wilkinson [51] and Minieka [33] but also saves maximum arc capacities that are not used to increase the flow value.

3.1.3. *Lexicographically maximum dynamic contraflow.* With given priority ordering at terminals on network $\mathcal{N} = (V, A, b, \tau, S, D, T)$, the LMDCFP is to find a feasible dynamic flow at each priority terminal with arc reversal capability at any time. Fixing the supplies and demands at sources and sinks, the LMDCFP problem has been solved with polynomial time complexity [38, 41]. However, it is also solvable for unknown supplies and demands on terminals in the same complexity because there is a priority in terminals not in supplies and demands. In the reconfigured network $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, S, D, T)$ of Algorithm 1, if we calculate the minimum cost flow in Step 3 at each iteration as in [21], the LMDCF solution is obtained after δ iterations within time horizon T by saving unused arc capacities. Thus,

Theorem 3.4. *The LMDCFP with partial reversals of arc capacities can be solved in $O(\delta \times MCF(m, n))$ time, where $MCF(m, n) = O(m \log n(m + n \log n))$ complexity of minimum cost flow solution.*

3.1.4. *Earliest arrival transshipment contraflow.* If the supplies and demands are fixed on the sources and sinks, respectively, and the supplies should be shifted within given time horizon T by sending maximum amount at every time point from the beginning with arc reversals capability, then the problem is earliest arrival transshipment contraflow problem (EATCFP).

Authors in [36, 40, 41, 42] investigate the EATCFP and present polynomial time algorithms on multi-source or multi sink networks for specific arc transit times. Moreover, a pseudo-polynomial time algorithm has been presented and its approximation solution is computed for arbitrary transit times on each arcs. If the transit time is zero, i.e., if arc capacities restrict the quantity of flow that can be sent at any time, then, the approximation solution is obtained in polynomial time. For an urban evacuation scenarios including life boats or pick-up bus stations, the concept of zero transit time is very importation and applicable.

Based on the previous results from literature, the EATCFP can be solved in different conditions using Algorithm 1. Also using the same algorithm, arc capacities can be saved in maximum number that are not used in contraflow configuration. However, this problem is not solved on the *S-D* network yet. For the *S-d* network $\mathcal{N} = (V, A, b, \tau, S, d, T)$ with arbitrary transit times on each arc, the EATCFP can be solved in polynomial time complexity with arc reversal capability at any time whenever necessary. Moreover, the algorithm records all unused arc capacities. By constructing extended network of reconfigured *S-d* network, we can compute the s^*-d minimum cost flow circulations according to [4] and can save arcs capacity using Step 4 of our algorithm. If the network is *s-D* with arbitrary transit times, its solution does not exist because there is always conflict which *s-D* path should be used first to make earliest possible flows. To solve the problem, the transit time

is assumed to be zero and thus every s-D paths have same length yielding optimal EATCF in polynomial time. Different networks can be categorized as in [48] wherein the s-D network EATCFP can be solved polynomially using Algorithm 1 with reversing the partial capacities of arcs.

Theorem 3.5. *The EATCF problem is solved polynomially on multi-sources or multi-sinks network saving the unused arc capacities.*

Even with the zero transit times, the S-D EATCF solution is not possible. Consider a network $\mathcal{N}$ with two sources s_1, s_2, two sinks d_1, d_2 and arcs (s_1, d_1), (s_1, d_2) and (s_2, d_2). Each source and sink have supply 2 and demand 2, and each arc has unit capacity. If we use all paths at time zero, we can transship 3 units of flows. But leaving the path $s_1 - d_2$ empty, we can transship only 2 units of flows at time zero violating the maximality at every time point.

Thus, we can investigate the approximate solution for the S-D network EATCFP. For the solution, the reconfigured S-D transshipment network is transformed into time expanded network. Then the extended time expanded network is constructed adding supper source s^* and super sink d^* with enough time bound T in which we can apply the algorithm of Gross *et al.* [16] that computes value-approximate EAT solution. The optimal EAT solution is bounded by the 2 times approximation EAT solution which is called 2-value approximate EAT. It is equivalent to the 2-value approximate EATCF on given network with arbitrary transit time on each arc. As it works on time expanded network, its complexity is pseudo-polynomial. However, if the transit time is reduced to be zero, polynomial time 2-value approximation EATCF can be obtained using algorithm of [23] on Step 3 of Algorithm 1.

Theorem 3.6. *The 2-value approximated EATCFP on S-D network can be solved efficiently by saving all unused arc capacities for emergency supports.*

3.1.5. *Quickest contraflow.* Our understanding of the time minimization problem is to minimize the total time required to transship the fixed flow amount to the sinks with arc reversal capability, i.e., the quickest contraflow problem (QCFP). For given network $\mathcal{N} = (V, A, b, \tau, S, D, Q_0)$ with integer inputs, the DCFP **P** is to find a minimum time T to transship Q_0 flow to D on S-D networks with arc reversal capability.

Authors in [45, 36, 39, 41, 43] investigated the s-d network QCFP and S-D network QTCFP and presented efficient algorithms to solve it. On the S-D network, the QCFP is harder than 3-SAT and PARTITION [45]. However, for s-d network, a strongly polynomial time algorithm has been presented to solve the problem in discrete time settings. Their algorithm has based on the parametric search algorithms of Megiddo [32] and Burkard *et al.* [5]). Computing $s - d$ paths, they first obtained an upper bound on the quickest time in polynomial time and applied a binary search repeatedly to compute MDCF along the path until all given flow values at the source are sent to the sink. It can be solved in $O(m^2(\log n)^3(m + n \log n))$ time. In same complexity, the QCFP is solved in continuous times by Pyakurel and Dhamala [39]. Pyakurel et al. [43] presented the first minimum cost flow algorithm to solve the s-d QCF problem. The s-d QCF solution has been computed by solving the minimum cost (static) flow problems using cost scaling algorithm of Lin and Jaillet [30]. It takes $O(nm \log(n^2/m) \log(n\tau_0))$ time to solve it where τ_0 is the maximum transit time. All the algorithms are presented with complete contraflow configuration. However, unnecessary reversals of arcs are not

relevant during evacuation process. Here, we not only improve the complexity of algorithm to solve the s-d QCFP but also save all unused arc capacities.

Instead of cost scaling algorithm of Lin and Jaillet [30], we use cancel and tighten algorithm of Saho and Shigeno [47] in Step 3 of Algorithm 1 that computes the quickest flow by solving the minimum cost flow problem in strongly polynomial time complexity. Based on the results, we state Lemma 3.7 and Theorem 3.8 without their detailed proofs.

Lemma 3.7. *The minimum cost flow solution can be computed on the* $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, s, d)$ *optimally.*

Theorem 3.8. *The QCFP on s-d network can be solved in $O(nm^2(\log n)^2)$ time complexity with partial reversals of arc capacities.*

3.1.6. *Quickest transshipment contraflow.* If we have given equal supplies and demands on S and D, respectively, and we have to transship the given supplies from S by satisfying the demands at D in minimum time T, then the QCFP is the quickest transshipment contraflow problem (QTCFP). Authors in [39, 41] investigated the S-D network QTCFP and presented efficient algorithms to solve it.

3.2. **Generalization of dynamic contraflow.**

Problem 2. *Given a lossy network $\mathcal{N} = (V, A, b, \tau, \lambda, S, D, T)$ with integer inputs, the generalized earliest arrival contraflow problem (GEACF) is to find a generalized maximum flow at every time point from the begging with arc reversal capability at time zero.*

If flow is maximized for a given time horizon T, then Problem 2 is the generalized maximum dynamic contraflow problem (GMDCFP). As the MDCFP is NP-hard on S-D network, additional gain factor on each arc makes the GMDCFP as well as GEACFP NP-hard on S-D lossy network. Considering s-d lossy network, Problem 2 can be solved computing GMDCF and GEACF in pseudo-polynomial time complexity, [44, 36]. However, in the same complexity, we can save some unused arc capacities for other purpose with Algorithm 2.

There are different factors that make the flow lost during evacuation process. However, we consider a special case in which we assume that in each time unit the same percentage of the remaining flow value is lost. Thus we consider a special case $\lambda \equiv 2^{c.\tau}$ for some constant $c < 0$.

Algorithm 2. Generalized dynamic contraflow algorithm (GDCFA)
  (1) Given a lossy network $\mathcal{N} = (V, A, b, \tau, \lambda, s, d)$ with integer inputs.
  (2) Compute the GMDCF and GEACF on $\overline{\mathcal{N}} = (V, E, \overline{b}, \overline{\tau}, \lambda, s, d)$ with respective capacities and transit time $b_{\overline{e}}$ and $\tau(\overline{e})$ as in Step 1 of Algorithm 1, and with additional gain factor $\lambda_{\overline{e}} \equiv 2^{c.\tau_{\overline{e}}}, c < 0$ using algorithm of [17].
  (3) Reverse arc $e' \in A$ if and only if $b_e < x_e$ along the arc $e \in A$ or if $x_e > 0$ along the arc $e \notin A$.
  (4) Record the arc capacities as $\tilde{b}_{e \vee e'} = b_{\overline{e}} - x_{\overline{e}}$ for all $e \in A$.
  (5) The resulting flow is GMDCF and GEACF with the arc reversals by saving unused arc capacities on $\mathcal{N}$.

In the reconfigured network, we compute a maximum flow along shortest s-d paths of residual network, augment this flow and repeat this process until no s-

d path exists in the residual network. Then, the augmented maximum flow constructs an optimal solution as the temporally repeated flow technique for standard maximum dynamic flow.

3.3. Abstract dynamic contraflow.

Model of path reversals. In abstract dynamic network $\mathcal{N} = (A, \mathcal{P}, b, \tau, S, D, T)$, path set $\mathcal{P}$ contains both s-d and d-s paths. As the movement of flow along d-s paths are stopped during evacuation process, the reversals of such paths by adding the capacities with s-d paths increase the flow value entering the sinks [42]. We improve the solution in which the contraflow process not only increases the flow value but also record maximum unused capacities of elements contained in the paths and keep them unturned for logistic supports and emergency vehicles moving towards the sources. With arc reversals, a transformed network $\overline{\mathcal{N}} = (E, \overline{\mathcal{P}}, \overline{b}, \tilde{b}_{e \vee e'}, \overline{\tau}, \overline{S}, \overline{D})$ is obtained, where new capacity and transit time are $b(\overline{P}) = \min_{\overline{e} \in \overline{P}} b_{\overline{e}}$ and $\tau(\overline{P}) = \sum_{\overline{e} \in \overline{P}} \tau_{\overline{e}}$, respectively, for $\overline{P} \in \overline{\mathcal{P}}$. As path $\overline{P}$ uses only minimum capacities of elements contained in it, we can save maximum unused capacities of the elements as $\tilde{b}_{e \vee e'} = b_{\overline{e}} - \sum_{\overline{P} \in \overline{\mathcal{P}} : \overline{e} \in \overline{P}} b(\overline{P}), \forall\, e \vee e' \in A$. By saving $\tilde{b}_{e \vee e'}$ capacities, reconfigured abstract network is obtained as $\overline{\mathcal{N}}_E = (E, \overline{\mathcal{P}}, b(\overline{P}), \tau(\overline{P}), \overline{S}, \overline{D})$. This process of contraflow is the partial contraflow configuration.

The abstract dynamic contraflow problem on multi-terminal network has not been investigated and thus, its solution status is unknown yet. However, on two terminal network, an algorithm has been presented (cf. Algorithm 3) to solve it with partial path reversal capability.

Algorithm 3. Abstract dynamic contraflow algorithm (ADCFA)

(1) **Input:** Given a abstract dynamic network $\mathcal{N} = (A, \mathcal{P}, b, \tau, s, d, T)$.
(2) Obtain a transformed abstract network $\overline{\mathcal{N}} = (E, \overline{\mathcal{P}}, \overline{b}, \tilde{b}_{e \vee e'}, \overline{\tau}, \overline{s}, \overline{d})$.
(3) Obtain the reconfigured abstract network $\overline{\mathcal{N}}_E = (E, \overline{\mathcal{P}}, b(\overline{P}), \tau(\overline{P}), \overline{s}, \overline{d}, T)$.
(4) Compute the abstract dynamic flow on $\overline{\mathcal{N}}_E$.
(5) An d-s path with its capacity is reversed if and only if the flow along s-d path P is greater than its capacity or if there is a nonnegative flow along s-d path $P \notin \mathcal{P}$.
(6) Record the maximum unused capacities $\tilde{b}_{e \vee e'}$ of elements $e \vee e' in A$ contained in s-d or d-s paths.
(7) **Output:** An abstract dynamic partial contraflow for network $\mathcal{N}$.

3.3.1. *Abstract maximum dynamic contraflow.* On network $\mathcal{N} = (A, \mathcal{P}, b, \tau, s, d, T)$, the abstract maximum dynamic contraflow problem (AMDCFP) is to find a abstract maximum dynamic flow (AMDF) that can be sent from s to d in time T and saving unused capacities of elements by reversing the direction of paths at time zero. Authors in [42] solve the AMDCFP in polynomial time. Their algorithm computes the abstract maximum dynamic flow on reconfigured network $\overline{\mathcal{N}}_E = (E, \overline{\mathcal{P}}, b(\overline{P}), \tau(\overline{P}), \overline{s}, \overline{d})$ which is equivalent to the AMDCF solution on given network. Algorithm 3 is similar to their algorithm, however, we cam save maximum unused element capacities with additional Step 6 and can use it for logistic and emergency supports to the evacuees. In the reconfigured network

$\overline{\mathcal{N}}_E = (E, \overline{\mathcal{P}}, b(\overline{P}), \tau(\overline{P}), \overline{s}, \overline{d}, T)$, an abstract dynamic flow $x : \overline{\mathcal{P}}_T \to \mathcal{R}^+$ is computed where

$$\overline{\mathcal{P}}_T = \left\{ \overline{P}_\theta : \overline{P} \in \overline{\mathcal{P}}, \theta \in \mathcal{T}, \beta + \sum_{\overline{p} \in \overline{P}} \tau(\overline{p}) \leq T \right\}$$

with the set of all temporal paths $\overline{P}_\theta$ defined as

$$\overline{P}_\theta = \left\{ (\overline{e}, \beta) \in A_T : \overline{e} \in \overline{P}, \beta = \theta + \sum_{\overline{p} \in (\overline{P}, \overline{e})} \tau(\overline{p}) \right\}.$$

Here the flow along path $\overline{P}_\theta$ enters element $\overline{e}$ at time $\theta + \sum_{\overline{p} \in (\overline{P}, \overline{e})} \tau(\overline{p})$ and reaches the sink at time $\theta + \sum_{\overline{e} \in \overline{P}} \tau(\overline{e})$. Then the maximum dynamic abstract flow maximizes objective (3.2) by respecting the capacity and feasibility constraints (3.3-3.4), Kappmeier et al. [24].

$$(3.2) \qquad\qquad \max \qquad \sum_{\overline{P}_\theta \in \overline{\mathcal{P}}_T} x(\overline{P}_\theta)$$

$$(3.3) \qquad \text{such that} \sum_{\overline{P}_\theta \in \overline{\mathcal{P}}_T : (\overline{e}, \beta) \in \overline{P}_\theta} x(\overline{P}_\theta) \quad \leq \quad \overline{b}_{\mathcal{A}}(\overline{e}), \forall \, \overline{e} \in \mathcal{A}, \, \theta \in \mathcal{T}$$

$$(3.4) \qquad\qquad\qquad\qquad x(\overline{P}_\theta) \quad \geq \quad 0, \, \forall \, \overline{P}_\theta \in \overline{\mathcal{P}}_T$$

Theorem 3.9. *The s-d AMDCFP can be solved in polynomial time with partial reversals of element capacities contained in the s-d or d-s paths.*

Moreover, using Algorithm 3, the abstract earliest arrival contraflow (AEACF) on s-d abstract network and its approximation solution on S-D network with fixed supplies/demands can be solved as follows.

3.3.2. *Abstract earliest arrival contraflow problem.* The S-D network AMDCFP and the S-D network AEACFP have not been studied yet. It may be interesting to find an efficient solution technique for the S-D AEAPCFP. We consider the problem on s-d network. The s-d AEACF on $\mathcal{N} = (\mathcal{A}, \mathcal{P}, b_{\mathcal{A}}, \tau_{\mathcal{A}}, s, d, T)$ is the s-d MDF at every time period from the beginning by reversing the partial capacities of paths. The solution can be computed with Algorithm 3 in which the s-d AEAF at Step 4 is obtained according to [23]. The s-d abstract dynamic network is transformed into time expanded abstract network and computed the static flow in the priority ordering using abstract lex-maximum flow algorithm of [23]. The static solution is equal to the AEAF on $\overline{\mathcal{N}}_E = (E, \overline{\mathcal{P}}, b(\overline{P}), \tau(\overline{P}), \overline{s}, \overline{d})$ and thus equivalent to the AEACF on $\mathcal{N} = (A, \mathcal{P}, b, \tau, s, d)$. As the solution is directly dependent on time expanded abstract network, its complexity if pseudo-polynomial. Due to Step 6 of algorithm, partial reversals of path capacities is possible.

Theorem 3.10. *The s-d AEACFP can be solved in pseudo-polynomial time complexity with partial reversals of path capacities.*

If the supplies and demands are fixed, then the AEACFP can be investigated in S-D network and a MDF at every $\theta \in \{1, 2, \ldots, T\}$ needs to be obtained through s-d paths satisfying the supplies and demands at S and D, respectively. As there is no efficient technique to find optimal solution of the problem, it is still possible to

compute its near optimal solution. In Step 4 of Algorithm 3, if we find the 2-value-approximate AEAF using algorithm of [23], it gives the near optimal solution to the AEACF on given network. However, the approximation algorithm is used on time expanded abstract network of reconfigured network $\overline{\mathcal{N}}_E = (E, \overline{\mathcal{P}}, b(\overline{P}), \tau(\overline{P}), \overline{S}, \overline{D})$. The resulting S-D time expanded abstract network is converted into s-d network by constructing extended time expanded abstract network that also satisfies the switching property (i.e., if two paths P_1 and P_2 share an element, there exists another path P_3 that is contained by the beginning part of P_1 and the end part of P_2).

Then, the AMSF for weighted case is solved in the extended time expanded abstract network using RAMFMC algorithm of [31]. The obtained solution on extended time expanded abstract network is 2-value-approximate AEAF on reconfigured network and thus the 2-value-approximate AEACF on original network with given supplies and demands. Due to the complexity of time expanded abstract network, we get the near optimal solution on pseudo-polynomial time on S-D network.

4. Lane reversals with variable attributes

In this section, we consider problems with variable transit times, i.e., transit times may vary due to the current situation of flow in arcs. If transit times depend upon the density, speed and flow rate along the arc, then it is considered as flow dependent transit times. If the transit time $\tau_e(x_e(\theta))$ is a function of inflow rate $x_e(\theta)$ on the arc e at given time point θ, so that at a time flow units enter an arc with the uniform speed and remains the uniform speed through traveling this arc, it is a inflow dependent transit time, as a relaxation of flow-dependent transit times. Another relaxation is load dependent transit times in which the transit time $\tau_e(l_e(\theta))$ depends on the total amount of flow on an arc e at a given time θ, i.e., the load $l_e(\theta)$ on e. In this approach all units of flow on an arc enjoy the same speed and the transit time of an arc varies with each unit of flow entering or leaving the arc with the continuous changes of flow on this arc. There exist various ways of considering the transit time attribute, however, we represent them by the same denotation τ_e as for the constant transit times on the arc $e \in A$ to be understood with any specific context chosen. An accurate estimation of link travel times is quite complicated as it is nonlinear function of probable congestion. We recommend Smith and Cruz [49] for an overview and various approaches of travel times estimation on arterial links, free and high ways.

4.1. **Contraflow with inflow dependent transit times.** In this section, we investigate the quickest contraflow problem with an additional restriction on the transit time function which completely depends on the current rate of inflow into that arc at any point of time. As network has two way arcs e and e', in which the inflow rates are $x_e(\theta)$ and $x_{e'}(\theta)$, respectively. During evacuation process, as d-s flows are not allowed, the inflow alone arc e' is zero, i.e., $x_{e'}(\theta) = 0, \theta = 0, \ldots, T$. Due to the bounded capacity b_e, the inflow $x_e(\theta)$ alone e gets congested. With contraflow configuration, the direction of the empty arc e' is changed and its capacity is added to the capacity of e so that new arc $\bar{e}$ is formed with new capacity $b_{\bar{e}}$. With increased capacity of arcs, the inflow rates may increase, i.e., $x_{\bar{e}}(\theta) \geq x_e(\theta)$, where $x_{\bar{e}}(\theta)$ is the inflow rate on the reconfigured network. Depending upon the inflow rates, the transit times also change at each point of time. As arc e' is reversed, the capacity $b_{e'}$ of the arc is used to shift flows with maximum possible

inflow rate $x_{e'}(\theta)$ alone the reversed direction and its transit time is $\tau_{e'}(x_{e'}(\theta))$. As in Köhler et al. [28], we assume that the arc-wise entering flows on reconfigured network impose the pace of every unit of flow and it remains fixed throughout.

In oder to model the inflow dependent flow over time problem, we assume that at any moment of time, the transit time function on an arc is given as a piecewise constant, non-decreasing and left-continuous function of inflow rate. Note that the transit time function can be restricted to be only integral values as it can be easily relaxed to allow arbitrary rational values by scaling the time with a proper way. Moreover, any general non-negative, non-decreasing and left-continuous function has been approximated by a step function within arbitrary precision. The transit time of reconfigured network is $\tau_{\bar{e}}(x_{\bar{e}}(\theta))$ that may be different than the transit time $\tau_e(x_e(\theta))$ or $\tau_{e'}(x_{e'}(\theta))$ depending upon the inflow value. For example, if the inflow rate increases with increasing capacity, the transit time also increases due to congestion. If the inflow rate decreases, the transit time also decreases because flow can move with high speed. For the constant inflow rate, the transit time is equal to the time without contraflow, i.e., $\tau_e(x_e(\theta))$ or $\tau_{e'}(x_{e'}(\theta))$. For the simplicity, we assume that the transit time is the same as without contraflow alone the arc e, i.e., $\tau_{\bar{e}}(x_{\bar{e}}(\theta)) = \tau_e(x_e(\theta))$ or $\tau_{\bar{e}}(x_{\bar{e}}(\theta)) = \tau_{e'}(x_{e'}(\theta))$ for the reversal of the arc e'. We present Algorithm 4 to solve the QCFP with inflow dependent transit times.

Algorithm 4. QCF with inflow dependent transit times (QIFDTA)
  (1) **Input.** Given network $\mathcal{N} = (V, A, b, \tau, s, d)$ with given flow value Q_0.
  (2) Compute the quickest flow on reconfigured network $\overline{\mathcal{N}} = (V, E, \bar{b}, \bar{\tau}, s, d, Q_0)$ using algorithm of Köhler et al. [28] with respective capacities $b_{\bar{e}}$ and inflow dependent transit times $\tau_{\bar{e}} = \tau_{\bar{e}}(x_{\bar{e}}(\theta))$ defined as

$$
\begin{array}{rcl}
b_{\bar{e}} & = & b_e + b_{e'} \\
\tau_{\bar{e}}(x_{\bar{e}}(\theta)) & = & \left\{ \begin{array}{ll} \tau_e(x_e(\theta)), & \text{if } e \in A \\ \tau_{e'}(x_{e'}(\theta)), & \text{otherwise} \end{array} \right.
\end{array}
$$

  for all periods $\theta = 0, \ldots, T$ and edges $\bar{e} \in E$.
  (3) Reverse the arc $e' \in A$ if and only if $b_e < x_e$ along the arc $e \in A$ or if $x_e > 0$ along the arc $e \notin A$.
  (4) Record the arc capacities as $\tilde{b}_{e \vee e'} = b_{\bar{e}} - x_{\bar{e}}$ for all $e \in A$.
  (5) **Output.** QCF with inflow dependent transit times and partial reversals of arc capacities for original network $\mathcal{N}$.

Algorithm 4 is feasible. Before we prove the correctness of the algorithm, we prove that the temporally repeated flow with inflow dependent transit times can be computed using fan and bow networks in Step 1 as in Köhler et al. [28]. The Fan network of reconfigured network $\overline{\mathcal{N}}$ is defined as $\overline{\mathcal{N}}^F(T) = (V_T^F, E_M^F \cup E_H^F)$ where where $V_T^F = \{v_\theta \mid v \in V, \theta = 0, 1, \ldots, T\}$ denotes the set of nodes. The set of holdover arcs $E_H^F = \{(v_\theta, v_{\theta+1}) \mid \theta = 0, 1, \ldots, T-1\}$ with infinite capacity allows a possibility of holding a flow at node v. The set of fan arcs $E_M^F = \{(v_\theta, w_{\theta+\tau_{\bar{e}}^s(\theta)}) \mid \bar{e} = (v, w) \in E, \theta = 0, 1, \ldots, T-1\}$, where $\tau_{\bar{e}}^s$ is the travel time on $\bar{e}$ at time θ. Here $\overline{\tau^s}$ is assumed to be non-decreasing left-continuous inflow dependent transit time step function with only integer values.

Consider an example as in Figure 3 of an arc of time expanded fan network with inflow rates 2 and 4, and $T = 6$. The capacitate horizontal fan arcs that control the flow distribution are continued with uncapacitate fan arcs pointing downwards

that give the different possible transit times. For the flow rate $y_{\bar{e}}$ sent into the fan, transit time to the slowest portion of this flow is at least $\tau_{\bar{e}}^s(y_{\bar{e}})$. However, the fan network representation only realizes a relaxation of the real inflow dependent transit times settings. As in Figure 3, every flow units should travel $\bar{e}$ with $\tau_{\bar{e}}^s(z_2) = 4$. But only z_2-z_1 units travel with time 4 as z_1 may already travel with time 2.
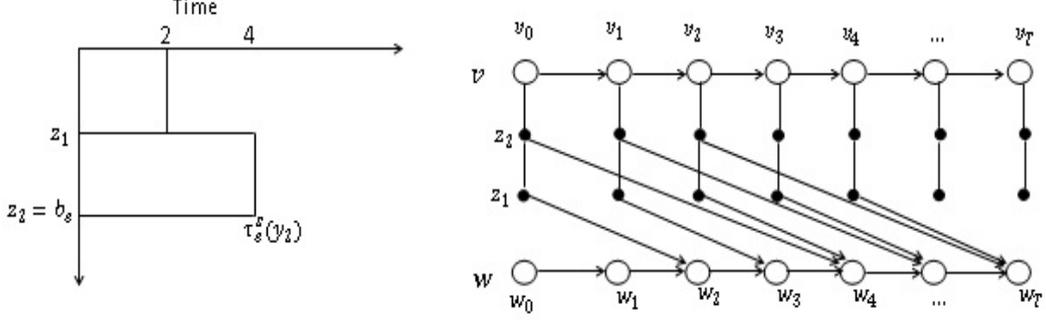

FIGURE 3. The expansion of a single arc $\bar{e} = (v, w)$ in the fan network with given piecewise travel times 2 and 4, for at most z_1 and z_2 flow units, respectively. All arcs except the regulating are of infinite capacities.

The size of fan network increases the size of time-expanded network by its step functions very largely. Therefore, it is quite hard to handle the flow over time with inflow dependent transit times even with approximate settings in the fan. In order to deal with a certain class of flows over time with inflow dependent transit times, the bow network of the reconfigured network is considered as in Köhler et al. [28] as follows.

Again define the transit times given by the step functions $\tau_{\bar{e}}^s$ on arc $\bar{e}$, and consider the breakpoints of flow rates $0 < z_1 < z_2 \cdots < z_k = b_{\bar{e}}$ with the corresponding transit times $\bar{\tau}^1 < \bar{\tau}^2 < \cdots < \bar{\tau}^k$, respectively. Then the bow network is defined by $\overline{\mathcal{N}}^B = (V^B, E^B)$, where $V^B = V$ and $\bar{e} \in E^B$ consists of two classes of arcs, namely, the bow arcs b^i and the regulating arcs r_i, $i = 1, 2, \ldots, k$. The former are uncapacitated with transit times $\bar{\tau}^i$, whereas the latter with zero transit time and capacity z_i. The size of this expansion is linear in the number of arcs. Clearly, the fan-network is a time expansion of the bow network.
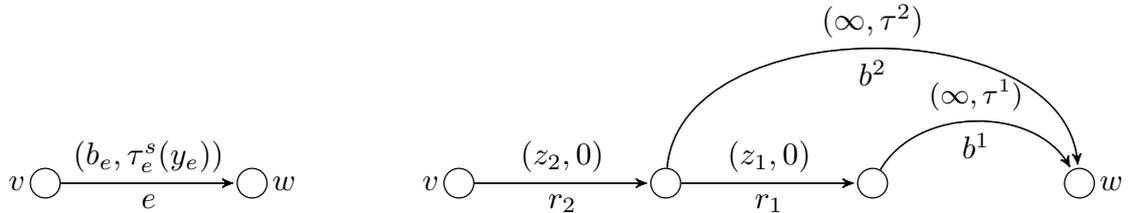

FIGURE 4. Expansion of a single arc $e = (v, w)$ in the bow network with travel times 2 and 4, for at most z_1 and z_2 flow units, respectively.

For given flow over time x in the reconfigured network $\overline{\mathcal{N}}$ with inflow dependent transit times step functions $\tau_{\bar{e}}^s$ and time horizon T given by the flow rates $x_{\bar{e}}(\theta)$:

$E \times T \to R^+$, the flow over time x^B on the bow expansion of $\bar{e}$ is considered as in Definition (4.1).

$$(4.1) \qquad x_{\bar{e}}^B(\theta) = \begin{cases} x_{\bar{e}}(\theta), & \text{if } \bar{e} = b^i \text{ or } \bar{e} = r_j \text{ with } j \geq i \\ 0, & \text{otherwise} \end{cases}$$

With this, every flow over time with inflow dependent transit times in $\overline{\mathcal{N}}$ can be considered as a flow over time with constant transit times in $\overline{\mathcal{N}}^B$, but not conversely. The problem on bow network is certainly a relaxation of the original with inflow dependent transit times flow over time problem. Lemma 4.1 assumes inflow dependent non-decreasing piecewise constant transit time functions.

Lemma 4.1. *For given dynamic s-d flow with inflow dependent transit times sending Q_0 units in reconfigured network $\overline{\mathcal{N}}$ withing time $T^* = \min T(Q_0)$, a temporally repeated flow with inflow dependent transit time can be computed in strongly polynomial time that sends the same amount of s-d flow within time at most $2T^*$.*

Proof. On reconfigured network $\overline{\mathcal{N}}$ with a piecewise constant, non-decreasing and left-continuous transit time functions, we use the cancel-and-tighten algorithm of [47] to determine a quickest flow in the bow graph $\overline{\mathcal{N}}^B$ of $\overline{\mathcal{N}}$. The minimal time horizon obtained by cancel-and-tighten algorithm to transship Q_0 flow is T^B and the static flow is y^B in $\overline{\mathcal{N}}^B$ such that the value of resulting temporally repeated flow x^B as in [14] is

$$(4.2) \qquad val(x^B) = T^B val(y^B) - \sum_{\bar{e} \in E^B} \tau_{\bar{e}} y_{\bar{e}}^B = Q_0$$

Here the time horizon T^B in $\overline{\mathcal{N}}^B$ is a lower bound on the optimal time horizon T^* in $\overline{\mathcal{N}}$. The dynamic flow x^B in $\overline{\mathcal{N}}^B$ may not be a feasible dynamic flow in $\overline{\mathcal{N}}$, [28]. However, we overcome from the difficulty by rerouting the static flow y^B, i.e., flow can be pushed from fast bow arcs up to the slowest flow carrying bow arcs so that the transit time of bow arcs is fixed same as an arc $\bar{e} \in E$. The temporally repeated flow as in equation (4.2) can be computed with the modified static flow $\bar{y}^B$ and is denoted by $\bar{x}^B$ in $\overline{\mathcal{N}}^B$ for any time horizon $T \geq T^B$. The resulting transit time of bow arc $b^{\bar{e}}$ is $\tau_{b^{\bar{e}}} = \tau_{\bar{e}}^s(\bar{y}_{b^{\bar{e}}}^B)$ for each arc $\bar{e} \in E$. As $\bar{x}^B$ is an increasing function of T, the value of T is bounded from above by $2T^B$ [28]. Thus, $\bar{x}^B$ can be interpreted as a dynamic flow with inflow dependent transit time in $\overline{\mathcal{N}}$ and we have $x_{\bar{e}}(\theta) = \bar{x}_{b^{\bar{e}}}^B(\theta)$ for any time θ.

Recall that in configured network $\overline{\mathcal{N}}$, the flow entering an arc $\bar{e}$ at time θ in dynamic flow x arrives at its head after $\tau_{\bar{e}}^s(x_{\bar{e}}(\theta))$ time, and thus $\tau_{\bar{e}}^s(x_{\bar{e}}(\theta)) \leq \tau_{b^{\bar{e}}}$ for all θ [28]. As a result, the dynamic flow x in $\overline{\mathcal{N}}$ is feasible and satisfies the flow conservation constraints allowing the intermediate storage, i.e., a flow arriving via arc $\bar{e}$ at node $v \in V$ at time $\theta + \tau_{\bar{e}}^s(x_{\bar{e}}(\theta))$ waits there for $\tau_{b^{\bar{e}}} - \tau_{\bar{e}}^s(x_{\bar{e}}(\theta))$ time units until the corresponding flow in $\overline{x}^B$ has also arrived. Thus obtained dynamic flow x in $\overline{\mathcal{N}}$ is called a temporally repeated flow with inflow-dependent transit times where as the corresponding static flow y in $\overline{\mathcal{N}}$ is obtained by $y_{\bar{e}} = \bar{y}_{b^{\bar{e}}}^B$.

Moreover, the temporally repeated flow x can be obtained in strongly polynomial time and it is satisfied that $val(x) = val(\bar{x}^B) = Q_0$, and the time horizon of x is equal to T, a time horizon of $\bar{x}^B$. As time horizon T is bounded from above by $2T^B$, we come to the conclusion with $T \leq 2T^B \leq 2T^*$. $\qquad \square$

Theorem 4.2. *An approximated solution to the QCFP with inflow dependent transit times can be obtained in strongly polynomial time by reversing only partial arc capacities.*

Proof. We prove the theorem in two steps. First we show the feasibility and then we prove the correctness of Algorithm 4. Steps 2 and 4 are feasible. As there is no any cycle flows in step 3, there is either a flow along arc e or (e') but never in both arcs in Step 3. This proves that the flow is not greater than the reversed capacities on all the arcs at all time units. Thus, Step 3 is also feasible that proves the feasibility of the algorithm.

Now, we show the optimality. On reconfigured network $\overline{\mathcal{N}}$, we can compute the temporally repeated flow with inflow dependent transit times using algorithm of Köhler et al. [28] that gives the approximate quickest flow as in Lemma 4.1. From the feasibility of our algorithm, we directly conclude that every feasible quickest flow solution on the reconfigured network $\overline{\mathcal{N}}$ is equivalent to the QCF solution in the original network $\mathcal{N}$ as in constant transit times [45, 39, 40]. Thus, the obtained approximate quickest flow on $\overline{\mathcal{N}}$ is the approximate QCF for network $\mathcal{N}$ with inflow dependent transit times with is obtained in polynomial time complexity. Moreover, by reducing the dynamic flow $x_{\bar{e}}$ from arc capacity $b_{\bar{e}}$ for each arc $\bar{e} \in E$, we can save all the unused arc capacity of the network in the same complexity. □

4.2. **Contraflow with load dependent transit times.** In real application, the transit times on each arc is not fixed. It changes automatically depending upon the movement of flow on the arcs. In order to deal flows over time with the load dependent transit times, we assume as in [29] that at each moment of time, the entire flow on an arc travels with uniform speed that depends only on the current load (the amount of flow) of that arc. However, the independence of flows on different arcs weakens this model yielding only approximation solutions to the reality. For the load l_e and flow rate y_e on arc e, the relation $l_e = y_e \tau_e(y_e)$ holds for a static flow. With an assumption that the function $y_e \to y_e \tau_e(y_e)$ is non-negative, strictly increasing and convex, we also denote the transit time as a function of the load $\hat{\tau}_e(l_e)$. Satisfying the former relation by the flow rate and the load, we note that $\tau_e(y_e) = \hat{\tau}_e(l_e)$. Thus the load dependent transit times model relies on the fact that the speed of flow along an arc e is proportional to the inverse of current transit time $\hat{\tau}_e(l_e(\theta))$ at any moment of time θ. For the simplicity we write the transit time $\hat{\tau}_e(l_e(\theta))$ as $\tau_e(l_e(\theta))$ for load dependent transit time in the rest of the paper. For an inverse an arc e', its load dependent transit times is $\tau_{e'}(l_{e'}(\theta))$.

Here, we investigate the QCFP with load dependent transit times on each arc. With complete contraflow technique, Pyakurel et al. [43] proposed an approximation algorithm for the QCFP with load-dependent transit times, i.e., they do not care the unused arc capacities. By saving unused capacities of all arcs, we modify their algorithm in this work. We consider s-d network for the investigation of the QCFP. It is still unknown whether one can find an efficient technique to solve the QCFP with load dependent transit times on each arc for S-D network.

Recall that arc e' with load $l_{e'}(\theta)$ is being empty during evacuation process, i.e., $l_{e'}(\theta)) = 0, \forall \theta = 0, \ldots, T$. With contraflow configuration, the empty arc e' is reversed to opposite direction and its capacity $b_{e'}$ is added to the capacity b_e of arc e, if there exits, to form new arc $\bar{e}$ with new capacity $b_{\bar{e}}$. On the reconfigured arc $\bar{e}$, the entire flow travels with uniform speed that depends only on the current load of that arc as in without contraflow. Thus let $l_{\bar{e}}(\theta)$ be the load on arc $\bar{e}$ at

each point of time, i.e., $l_{\bar{e}}(\theta) \geq l_e(\theta)$. The transit time of the reconfigured arc $\bar{e}$ dependents on the current load $l_{\bar{e}}(\theta)$, i.e., $\tau_{\bar{e}}(l_{\bar{e}}(\theta)) \geq \tau_e(l_e(\theta))$, however, we assume $\tau_{\bar{e}}(l_{\bar{e}}(\theta)) = \tau_e(l_e(\theta))$ for the simplicity. If arc e' is only on the network then its reversed capacity $b_{\bar{e}}$ is equal to the original capacity $b_{e'}$ with maximum possible load $l_{e'}(\theta)$, i.e., $l_{\bar{e}}(\theta) = l_{e'}(\theta)$ and its transit time $\tau_{\bar{e}}(l_{\bar{e}}(\theta))$ is equal to $\tau_{e'}(l_{e'}(\theta))$.

We present Algorithm 5 to solve the QCF problem with load dependent transit times. The algorithm of [29] in Step 2 computes the quickest flow with temporally repeated flow but it is not optimal. Thus, our algorithm computes the approximate QCF solution with load dependent transit times and saves all unused capacities of arcs for other purpose.

Algorithm 5. QCF algorithm with load dependent transit times (QLDTA)
  (1) **Input.** Given network $\mathcal{N} = (V, A, b, \tau, s, d)$ with given flow value Q_0.
  (2) Computes the Quickest flow using algorithm of [29] on $\overline{\mathcal{N}} = (V, E, \bar{b}, \bar{\tau}, s, d)$ with respective capacities $b_{\bar{e}}$ and load dependent transit times $\tau_{\bar{e}} = \tau_{\bar{e}}(l_{\bar{e}}(\theta))$ defined as follows.
$$\tau_{\bar{e}}(l_{\bar{e}}(\theta)) = \begin{cases} \tau_e(l_e(\theta)) & \text{if } e \in \mathcal{A} \\ \tau_{e'}(l_{e'}(\theta)) & \text{otherwise} \end{cases}$$
  for all $\bar{e} \in E, \theta = 0, \ldots, T$.
  (3) Reverse arc $e' \in A$ if and only if $b_e < x_e$ along the arc $e \in A$ or if $x_e > 0$ along the arc $e \notin A$.
  (4) Record the arc capacities as $\tilde{b}_{e \vee e'} = b_{\bar{e}} - x_{\bar{e}}$ for all $e \in A$.
  (5) **Output.** QCF with load-dependent transit times and partial reversals of arc capacities for original network.

We prove the correctness of the algorithm. Before that we adopt the result of [29] on the reconfigured network $\overline{\mathcal{N}}$ as the following lemma.

Lemma 4.3. *Suppose that there exists a flow over time with load dependent transit times on $\overline{\mathcal{N}}$ that sends Q_0 units of s-d flow within time period T. Then a flow over time satisfying the same demand within time horizon at least $2T$ exists in the class of temporally repeated flows. Also, this solution can be computed polynomially within time $(2 + \epsilon)T$ for any given $\epsilon > 0$.*

The flow over time problem on $\overline{\mathcal{N}}$ has been approximately solved in polynomial time by exploiting the property of temporarily repeated solution of the corresponding average static flow rate $y_e = \frac{1}{T} \int_0^T x_e(\theta) d\theta$ as in [29]. As finding quickest flow over time with load dependent transit time is strongly $\mathcal{NP}$-hard, same result can be transformed in the reconfigured network and there does not exist an FPTAS unless $\mathcal{P} = \mathcal{NP}$ holds. The reductions are based on the $\mathcal{NP}$-complete PARTITION and SATISFIABILITY problems.

Theorem 4.4. *An approximation solution to the QCFP with load dependent transit times can be obtained in polynomial time by partial reversals of arc capacities.*

Proof. Algorithm 5 is feasible and the proof of the correctness is similar as in [43]. However, this algorithm saves all unused arc capacities without violating the complexity of the algorithm. The complexity is dominated by the calculation of approximate quickest flow with load dependent transit times in Step 2 using algorithm of [29] and its complexity is polynomial time. Moreover, the obtained approximate quickest flow on reconfigured network is equivalent to the approximate

QCF solution for given network with load dependent transit times. This completes
the proof. □

5. Conclusions

The evacuation planning problem with complete contraflow configuration has
been studied and analyzed deeply. Highlighting the overall pros and cons of com-
plete contraflow models and algorithms, a new and more relevant approach, i.e., a
partial lane reversals, has been introduced. In this approach, we have not only saved
maximum evacuees but also saved all unused capacities of the lanes for other emer-
gency and logistic supports for the evacuees with partial reversals of lane capacities.
The dynamic contraflow problems including, the maximum dynamic, the earliest
arrival, quickest, lex-maximum dynamic and transshipment contraflow problems
have been solved with efficient algorithms. The maximum dynamic and earliest ar-
rival contraflow problems are generalized on lossy networks with partial contraflow
reconfiguration. An abstract contraflow approach with partial lane reversals has
been developed and solved dynamic flow problems with efficient algorithms. These
problems has been solved in constant transit times on each arc or path.

Moreover, the partial contraflow models with variable transit times on each arc
have been introduced. For the inflow dependent transit times on each arc, a strongly
polynomial time algorithm has been presented that computes an approximation
solution to the two terminal quickest contraflow problem with partial lane reversals.
Considering the transit times depending on the current load of an arc, the two
terminal quickest contraflow problem has been solved with polynomial algorithm
and computed approximation solution by reversing only the partial capacity of
lanes.

To the best of our knowledge, these problems we introduced are for the first time
in the partial contraflow approach. Moreover, we are interested to extend these
contraflow models and algorithms to solve other dynamic network flow problems
with variable attributes that are more applicable and relevant. In addition, we
intend to implement the results for supporting logistics in emergencies using the
partial contraflow techniques.

References

1. N. Altay and III-W. G. Green (2006). OR/MS research in disaster operations management. *European Journal of Operational Research*, **175**, 475–493.
2. E.J. Anderson (1989). Extreme-points for continuous network problems with arc delays, Journal Information and Optimization sciences, 10: 45-52.
3. N. Baumann and E. Kóhler (2007). Approximating earliest arrival flows with flow-dependent transit times, *Discrete Applied Mathematics*, **155**, 161-171.
4. N. Baumann and M. Skutella (2009). Earliest arrival flows with multiple sources. *Mathematics of Operations Research*, **34**, 499-512.
5. R.E. Burkard, K. Dlaska and B. Klinz (1993). The quickest flow problem. *ZOR-Methods and Models of Operations Research*, 37:31–58.
6. L. Chen, and E. Miller-Hooks (2008). The building evacuation problem with shared information. Naval Research Logistics, **55**, 363–376.
7. T. Cova and J.P. Johnson (2003). A network flow model for lane-based evacuation routing. *Transportation Research Part A: Policy and Practice*, **37**, 579–604.
8. T.N. Dhamala (2015). A survey on models and algorithms for discrete evacuation planning network problems. *Journal of Industrial and Management Optimization*, **11**, 265-289.
9. T.N. Dhamala and U. Pyakurel (2013). Earliest arrival contraflow problem on series-parallel graphs. *International Journal of Operations Research*, **10**, 1-13

10. L. Fleischer (2001). Universally maximum flow with piecewise-constant capacities. Networks, **38**, 115-125.
11. L. Fleischer and M. Skutella (2007). Quickest flows over time, *SIAM Journal of Computing* **36**, 1600-1630.
12. L.K. Fleischer and E. Tardos (1998) Efficient continuous-time dynamic network flow algorithms. *Operations Research Letters*, **23**, 71-80.
13. F.R. Ford and D.R. Fulkerson (1958). Constructing maximal dynamic flows from static flows. Operations Research, **6**, 419-433.
14. L.R. Ford and D.R. Fulkerson (1962). *Flows in networks*, Princeton University Press, Princeton, New Jersey.
15. D. Gale (1959) Transient flows in networks. *Michigan Mathematical Journal*, 59-63.
16. M. Groß, J-P. W. Kappmeier, D.R. Schmidt and M. Schmidt (2012). Approximating earliest arrival flows in arbitrary networks. *L. Epstein and P. Ferragina (Eds.): ESA 2012, LNCS*, **7501**, 551-562.
17. M. Gross and M. Skutella (2012). Generalized maximum flows over time. *Approximation and Online Algorithms Lecture Notes in Computer Science*, **7164**, 247–260.
18. H.W. Hamacher and S.A. Tjandra (2002). Mathematical modeling of evacuation problems: A state of the art. In: *M. Schreckenberger and S.D. Sharma (Eds.), Pedestrain and Evacuation Dynamics*, Springer, 227-266.
19. A. Hall, K. Langkau and M. Skutella (2003). An FPTAS for quickest multicommodity flows with inflow-dependent transit times, in: S. Arora, K. Jansen, J.D. Rolim, A. Sahai (Eds.), Approximation, Randomization, and Combinatorial Optimization (APPROX03), *Lecture Notes in Computer Science*, **2764**, Springer, Berlin, 71-82.
20. B.E. Hoppe (1995). *Efficient dynamic network flow algorithms.* PhD thesis, Cornell University.
21. B. Hoppe and E. Tardos (2000). The quickest transshipment problem. Mathematics of Operations Research, **25**, 36-62.
22. E. Hoppe and E. Tardos (1994). *Polynomial time algorithms for some evacuation problems*, in: Fifth Annual ACMSIAM Symposium on Discrete Algorithms (SODA94), 433441.
23. J-P. W. Kappmeier (2015). *Generalizations of flows over time with application in evacuation optimization.* PhD Thesis, Technical University, Berlin, Germany.
24. J-P. W. Kappmeier, J. Matuschke and B. Peis (2014). Abstract flows over time: A first step towards solving dynamic packing problems. *Theoretical Computer Sciences*, **544**, 74-83.
25. D.R. Karger and C. Stein (1996). A new approach to the minimum cut problem. *Journal of the ACM*, **43**(4), 601-640.
26. S. Kim, S. Shekhar and M. Min (2008). Contraflow transportation network reconfiguration for evacuation route planning. *IEEE Transactions on Knowledge and Data Engineering*, **20**, 1-15.
27. I.S. Kotsireas, A. Nagurney and P.M. Pardalos (Eds.) (2015). *Dynamics of disasters - key concepts, models, algorithms, and insights.* Springer Proceedings in Mathematics & Statistics.
28. E. Köhler, K. Langkau and M. Skutella (2002). Time expanded graphs for flow depended transit times. *R. Möhring and R. Raman (Eds.): ESA 2002, LNCS* 2461, Springer-Verlag, 599-611.
29. E. Köhler and M. Skutella (2005). Flows over time with load-dependent transit times. *SIAM Journal of Optimization*, 15(4): 1185–1202.
30. M. Lin and P. Jaillet (2015). On the quickest flow problem in dynamic networks− a parametric min-cost flow approach. *Proceedings of the Twenty-Sixth Annual ACM-SIAM Symposium on Discrete Algorithms*, 1343-1356.
31. M. Martens and S.T. McCormick (2008). A polynomial algorithm for weighted abstract flow. Proceedings of the 13th Conference on Integer Programming and Combinatorial Optimization. Volume 5035 of Lecture Notes in Computer Science, Springer, 97-111.
32. N. Megiddo (1979). Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4:414-424.
33. E. Minieka (1973). Maximal, lexicographic, and dynamic network flows. *Operations Research*, **21**, 517-527.
34. K.D. Moriarty, D. Ni and J. Collura (2007). Modeling traffic flow under emergency evacuation situations: Current practice and future directions. *86th Transportation Research Board Annual Meeting*, Washington, D.C. $http://works.bepress.com/daiheng_ni/24$.

35. M.M.B. Pascoal, M.E.V. Captivo and J.C.N. Climaco (2006). A comprehensive survey on the quickest path problem. Annals of Operations Research, 147(1), 5-21.
36. U. Pyakurel (2016). *Evacuation planning problem with contraflow approach.* PhD Thesis, IOST, Tribhuvan University, Nepal.
37. U. Pyakurel and T.N. Dhamala (2014). Earliest arrival contraflow model for evacuation planning. *N*eural, Parallel, and Scientific Computations [CNLS-2013], **22**, 287-294.
38. U. Pyakurel and T.N. Dhamala (2015). Models and algorithms on contraflow evacuation planning network problems. *I*nternational Journal of Operations Research, **12**, 36-46.
39. U. Pyakurel and T.N. Dhamala (2016). Continuous time dynamic contraflow models and algorithms. *Advances in Operations Research - Hindawi*; Article ID 368587, 1-7.
40. U. Pyakurel and T.N. Dhamala (2017a). Evacuation planning by earliest arrival contraflow, *Journal of Industrial and Management Optimization* **13**, 489-503.
41. U. Pyakurel and T.N. Dhamala (2017b). Continuous dynamic contraflow approach for evacuation planning. *Annals of Operation Research*, **253**, 573-598.
42. U. Pyakurel, T.N. Dhamala and S. Dempe (2017a). Efficient continuous contraflow algorithms for evacuation planning problems. *Annals of Operations Research (ANOR)*, **254**, 335-364.
43. U. Pyakurel, H.N. Nath, and T.N. Dhamala (2017b). Efficient contraflow algorithms for quickest evacuation planning. *Science China Mathematics*, DOI 10.1007/s11425-017-9264-3.
44. U. Pyakurel, H.W. Hamacher and T.N. Dhamala (2014). Generalized maximum dynamic contraflow on lossy network. *International Journal of Operations Research Nepal*, **3**, 27-44.
45. S. Rebennack, A. Arulselvan, L. Elefteriadou and P.M. Pardalos (2010). Complexity analysis for maximum flow problems with arc reversals. *Journal of Combinatorial Optimization*, **19**, 200-216.
46. S. Ruzika, H. Sperber and M. Steiner (2011). Earliest arrival flows on series- parallel graphs. *Networks*, 10, 169-173.
47. M. Saho and M. Shigeno (2017). Cancel-and-tighten algorithm for quickest flow problems. *Network*, DOI 10.1002/net.
48. M. Schmidt and M. Skutella (2010). Earliest arrival flows in networks with multiple sinks. Electronics Notes in Discrete Mathematics, **36**, 607-614.
49. J.M.G. Smith and E.R.B. Cruz (2014). $M\backslash G\backslash c\backslash c$ state dependent travel time models and properties, *Physica A*, 395, 560-579.
50. J. Schroeder, A. Guedes and E.P. Duarte (2004). *Computing the minimum cut and maximum flow of undirected graphs.* Technical Report Federal University of Paran.
51. W.L. Wilkinson (1971). An algorithm for universal maximal dynamic flows in a network. *Operations Research*, **19**: 1602-1612.
52. M. Yusoff, J. Ariffin and A. Mohamed (2008). Optimization approaches for macroscopic emergency evacuation planning: a survey. *Information Technology, ITSim, International Symposium, IEEE*, **3**, **26-28**, 1–7.

[1,3] CENTRAL DEPARTMENT OF MATHEMATICS, TRIBHUVAN UNIVERSITY, P.O.BOX 13143, KATH-MANDU, NEPAL; [1,2] TU BERGAKADEMIE, FAKULTÄT FÜR MATHEMATIK UND INFORMATIK, 09596 FREIBERG, GERMANY

E-mail address: [1] urmilapyakurel@gmail.com, [2] dempe@math.tu-freiberg.de
E-mail address: [3]amb.dhamala@daadindia.org