APPROXIMATE ALGORITHMS FOR CONTINUOUS-TIME QUICKEST MULTI-COMMODITY CONTRAFLOW PROBLEM

SHIVA PRAKASH GUPTA[1], DURGA PRASAD KHANAL[2], URMILA PYAKUREL[3] AND TANKA NATH DHAMALA[4]

[1] *Tri-Chandra Multiple Campus, Tribhuvan University, Kathmandu, Nepal*
[2] *Saraswati Multiple Campus, Tribhuvan University, Kathmandu, Nepal*
[3,4] *Central Department of Mathematics, Tribhuvan University, Kathmandu, Nepal*

Corresponding author: Urmila Pyakurel, email: urmilapyakurel@gmail.com

Abstract: Multi-commodity flow problem appears when several distinct commodities are shipped from supply nodes to the demand nodes through a network without violating the capacity constraints. The quickest multi-commodity flow problem deals with the minimization of time satisfying given demand. In general, the quickest multi-commodity flow problems are computationally hard. The outbound lane capacities can be increased through reverting the orientation of lanes towards the demand nodes. We present two approximation algorithms by introducing partial contraflow technique in the continuous-time quickest multi-commodity flow problem: one polynomial-time with the help of length-bounded flow and another FPTAS by using Δ-condensed time-expanded graph. Both algorithms reverse only necessary arc capacities to get the optimal solutions and save unused arc capacities which may be used for other purposes.

Key Words: Multi-commodity flow, continuous-time, quickest contraflow, length bound, Δ-condense.

2010 Mathematics Subject Classification. *Primary: 90B10, 90C27, 68Q25; Secondary: 90B06, 90B20.*

1. INTRODUCTION

Routing of several distinct commodities from specific supply nodes to corresponding demand nodes is essential in day to day life. The transshipment of commodities is a network routing problem in which the sum of flow of commodities on each arc is bounded by its capacity. The problems modeled as a multi-commodity flow problem are message routing in telecommunication, railway network, vehicle routine in transportation, production planning, and logistics, etc. For more details, we refer to [1, 3, 14, 29, 31].

The road traffic network is considered as a network. The source points, destination points, and the junction of roads represent nodes and link from one node to another node is an arc. The arcs are assigned with capacity and travel time (cost) in the network. The chain from a specific source to its corresponding destination is a source-sink path. Commodities that are transshipped through the network along the path is considered as a flow.

Generally, network flow problems can be classified into two ways: static flow and dynamic flow (also known as flow over time) problems. The problem in which supply and demand on the origin and destination nodes are given, and wish to find the minimum possible time to fulfill the demand is called the quickest flow problem. By applying a binary search to the maximum flow solution of Ford and Fulkerson [10], Burkard et al. [4] provided the first polynomial-time-bound for this problem. Furthermore, a strongly polynomial-time-bound by incorporating a parametric approach to the minimum cost flow problem is obtained by the same authors.

The problem is extended to the case of a multi-terminal network. Hoppe and Tardos [13] presented a polynomial-time solution as one of such extensions, in which the vector of supplies and demands at the terminals are given, where the task is to find a flow over time that satisfies all supplies and demands within minimum possible time, is the quickest transshipment problem.

Fleischer and Skutella [8] generalized the dynamic flow problem introduced by Ford and Fulkerson [10] to the case of a multi-commodity network. The static multi-commodity flow problem is solved in polynomial-time by using the ellipsoid or interior point method. However, the multi-commodity dynamic flow problem is $\mathcal{NP}$-hard, [12]. Authors in [8] have investigated approximate solutions of the quickest multi-commodity flow (QMCF) problem. They developed two algorithms: one polynomial-time algorithm by using T-length bounded approximation and another fully polynomial-time approximation scheme (FPTAS) by using Δ-condensed time-expanded network.

Contraflow strategy increases the flow value by expanding the capacity of the lanes through reversing the orientation of lanes towards the demand nodes. Rebennack et al. [28] developed the models and provided analytical solutions for the two-terminal maximum and quickest flow problems. Furthermore, they developed strongly polynomial-time algorithms, where lane reversals are made at time zero and kept fixed afterward.

The optimal solution to the earliest arrival flow in a two-terminal network for a discrete-time setting is provided by Pyakurel and Dhamala in [19, 22]. By using the natural transformation, this discrete solution has been transformed into a continuous-time setting by the same authors in [20, 21]. Pyakurel et al. [23] designed the approximation algorithms for the earliest arrival transshipment problem with arbitrary and zero transit times on each arc. Furthermore, authors in [25] presented a strongly polynomial-time algorithm for the quickest flow problem and developed an approximation algorithm with load-dependent transit times. Dhungana and Dhamala [7] presented an algorithm for the maximum dynamic flow improvement problem with lane reversals having budget-constrained. Nath et al. [18] investigated the quickest contra-flow-loc problem with an objective of minimum increment of optimal time.

By reverting only necessary arc capacities to increase the flow value, Pyakurel et al. [27] developed a partial contraflow approach. This approach saves unused arc capacities that can be used for the logistic supports and facility location in emergency periods. Authors in [26] introduced partial contraflow with path reversals. Recently, Dhamala et al. [5]

presented two approximation algorithms for the QMCF problem with a partial contraflow in the discrete-time setting.

In this paper, we apply a partial contraflow approach in the QMCF problem and introduce the quickest multi-commodity contraflow (QMCCF) problem in a continuous-time setting. By reducing the multi-commodity flow problem into the single-commodity flow problems and decomposing the flow along the paths, we present two approximate algorithms to solve the problem. The first algorithm approximates it in polynomial-time by using a T-length bounded approach and second by using Δ-condensed time-expanded network in fully polynomial-time. The major significance of this study is the reduction of the delivery time of commodities by applying partial contraflow technique.

The rest of the paper is organized as follows. Section 2 provides some basic terminologies and models used in the article. The quickest multi-commodity flow problem with partial contraflow is introduced in Section 3. In this section, we present two algorithms for approximate solutions to this problem. The paper is concluded in Section 4.

2. Basic Terminologies and Flow Models

The transmission of more than one different commodities from respective sources to the corresponding sinks through a network concerns with the multi-commodity flow problem. In this section, we present the mathematical formulation of the flow models for multi-commodity flow.

2.1. **Flow models:** Consider a static network topology $\mathbb{N} = (V, E, K, b, d_i, S, D)$ with node set V, arc set $E \subseteq V \times V$, and the set of commodities $K = \{1, 2, ..., k\}$, where $|V| = n$ and $|E| = m$. S and D represent the set of sources and sinks respectively. For each commodity $i \in K$, d_i represents the demand of commodity i that is to be routed through a unique source-sink pair (s_i, t_i), where $s_i \in S \subset V$ and $t_i \in D \subset V$. The capacity function $b : E \to \mathcal{R}^+$ restricts the flow of commodities on each arc $e = (v, w)$. We denote $E^+(v) = \{(v, w) \mid w \in V\}$ and $E^-(v) = \{(w, v) \mid w \in V\}$ as the sets of out going arcs from node v and incoming arcs to node v, respectively, such that $E^+(D) = \emptyset$ and $E^-(S) = \emptyset$ except in contraflow network.

Network with the temporal dimension, denoted by $\mathbb{N} = (V, E, K, b, \tau, d_i, S, D, T)$, is a dynamic network, where a non-negative transit time function $\tau : E \to \mathcal{R}^+$ measures the time to transship the flow from the tail (v) to the head (w) of arc $e = (v, w)$. The time horizon T given in network $\mathbb{N}$ is denoted by $\mathbb{T} = \{0, 1, ..., T\}$ in discrete-time settings and $\mathbb{T} = [0, T)$ in continuous-time settings.

Static multi-commodity flow. A static multi-commodity flow g for the given static network $\mathbb{N}$ without temporal dimension is a sum of all non-negative static flows g^i defined by

the functions $g^i : E \to \mathcal{R}^+$ for each commodity i satisfying

(2.1)
$$\sum_{e \in E^+(v)} g_e^i - \sum_{e \in E^-(v)} g_e^i = \begin{cases} d_i & \text{if } v = s_i \\ -d_i & \text{if } v = t_i \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in K$$

(2.2)
$$0 \le \sum_{i \in K} g_e^i \le b_e \ \forall e \in E.$$

The first and second conditions of constraint in (2.1) represent the supply and demand at sources and sinks, respectively, whereas the third condition represents flow conservation for each commodity at intermediate nodes. The constraints in (2.2) are bundle constraints bounded by arc capacities. The cost of static flow g is defined as

(2.3)
$$c(g) = \sum_{e \in E} \sum_{i \in K} c_e^i g_e^i$$

where, c_e^i represents the cost coefficient associated with arc $e \in E$ and commodity $i \in K$.

Dynamic multi-commodity flow. For a given network $\mathbb{N}$, a dynamic multi-commodity flow f with constant transit time τ on arcs is a sum of flows defined by $f^i : E \times \mathbb{T} \to \mathcal{R}^+$ satisfying the constraints (2.4 - 2.6).

(2.4)
$$\sum_{e \in E^+(v)} \int_0^T f_e^i(\theta) d\theta - \sum_{e \in E^-(v)} \int_{\tau_e}^T f_e^i(\theta - \tau_e) d\theta = \begin{cases} d_i & \text{if } v = s_i \\ -d_i & \text{if } v = t_i \\ 0 & \text{otherwise} \end{cases} \quad \forall i \in K$$

(2.5)
$$\sum_{e \in E^+(v)} \int_0^\delta f_e^i(\theta) d\theta - \sum_{e \in E^-(v)} \int_{\tau_e}^\delta f_e^i(\theta - \tau_e) d\theta \le 0 \ \forall \ v \notin \{s_i, t_i\}, \ i \in K, \ \delta \in \mathbb{T}$$

(2.6)
$$0 \le f_e(\delta) = \sum_{i \in K} f_e^i(\delta) \le b_e \ \forall e \in E, \ i \in K, \ \delta \in \mathbb{T}$$

Here, the third condition of the constraints in (2.4) are flow conservation constraints at time horizon T. The strict inequality in (2.5) represents weak flow conservation constraints that allow to store the flow at intermediate nodes, where as equality in (2.5) represents flow conservation at intermediate nodes for all times δ without intermediate storage. Similarly, the bundle constraints in (2.6) are bounded above by the capacities. The goal is to transship the given amount of flow in order to satisfy the demand d_i of each commodity i from s_i to t_i, which is stated in first two conditions of (2.4). The cost of dynamic flow f is defined by

(2.7)
$$c(f) = \sum_{e \in E} \sum_{i \in K} c_e^i \int_0^T f_e^i(\theta) d\theta.$$

The bound of cost for every single commodity i is calculated as

(2.8)
$$\sum_{e \in E} c_e^i \int_0^T f_e^i(\theta) d\theta \le C_i,$$

and the budget constraints for dynamic flow f, stated in (2.7), is bounded by C.

2.2. **Natural Transformation:** Natural transformation is a technique to transform the solution of discrete dynamic flow problem to the continuous parameters. Fleischer and Tardos [9] presented the relations between discrete and continuous flow models with the notion of natural transformation by defining $f_e[\theta, \theta + 1) = \phi_e(\theta)$, where f represents the continuous dynamic flow for unit time interval $[\theta, \theta + 1)$ and ϕ represents the discrete dynamic flow entering arc e at time $\theta \in \{0, 1, ..., T\}$, (see also [6]). With the help of this natural transformation, multi-commodity flow can be extended as follows: any discrete dynamic flow $\phi_e^i(\theta)$ with integral time horizon T is equivalent to the continuous dynamic flow $f_e^i[\theta, \theta + 1)$ by incorporating the flow $\phi_e^i(\theta)$ entering arc e at time step $\theta \leq T - \tau_e$ as a constant flow rate on arc e during the unit time interval $[\theta, \theta + 1)$.

2.3. **Auxiliary Network:** For a given two-way network $\mathbb{N}$, the corresponding auxiliary network is denoted by $\mathbb{N}^a = (V, E^a, K, b^a, \tau^a, d_i, S, D, T)$, with undirected edges in $E^a = \{(v, w) : (v, w) \text{ or } (w, v) \in E\}$, where $\overleftarrow{e} = (w, v)$ is the reversed arc of $e = (v, w)$. The capacity of auxiliary arc is the sum of capacities of arcs e and $\overleftarrow{e}$ such that $b_e^a = b_e + b_{\overleftarrow{e}}$, where $b_e = 0$ if $e \notin E$. The transit time of auxiliary arc is

$$\tau_e^a = \begin{cases} \tau_e & \text{if } e \in E \\ \tau_{\overleftarrow{e}} & \text{otherwise.} \end{cases}$$

All others parameters are same.

Contraflow means the flipping of the orientation of arcs to increase the flow and reduce the time horizon by increasing the arc capacities as defined above. In partial contraflow, only necessary arc capacities are reversed to increase the flow value. The main idea behind the partial contraflow for a static network is as follows.

- Arc $\overleftarrow{e} = (w, v)$ is reversed if and only if either flow along the arc $e = (v, w)$ is greater than its capacity or there is nonnegative flow along the arc $e = (v, w) \notin E$. If $g_e > b_e$ and $b_e^a > g_e$, where $b_e^a = b_e + b_{\overleftarrow{e}}$, then the arc $\overleftarrow{e}$ is reversed partially and unused capacity of arc $\overleftarrow{e}$ is saved.
- If $g_e > b_e$ and $b_e^a = g_e$, then we have to reverse the arc $\overleftarrow{e}$ completely. Thus no capacity is saved.
- If $g_e < b_e$ neither arc e nor arc $\overleftarrow{e}$ is reversed, then remaining capacity of arc e and all capacity of arc $\overleftarrow{e}$ are saved.

Dynamic multi-commodity contraflow model. Consider a pre-defined two-way network $\mathbb{N}$, a dynamic multi-commodity flow f with constant transit time τ on arcs is a sum of flows defined by $f^i : E \times \mathbb{T} \to \mathcal{R}^+$ satisfying the constraints (2.4-2.5) with

$$(2.9) \qquad 0 \leq f_e(\delta) = \sum_{i \in K} f_e^i(\delta) \leq b_e + b_{\overleftarrow{e}} \quad \forall e \in E, \ i \in K, \ \delta \in \mathbb{T}.$$

The constraints in 2.9 represent the flows on the arcs which are bounded above by the contraflow capacities.

Example 2.1. Consider a two-commodity network with capacity and transit time on the arcs as shown in Figure 1. Flows are to be transshipped from s_1 to t_1 for Commodity-1

and s_2 to t_2 for Commodity-2. In Figure 1, (b) represents the auxiliary network of (a). Figure 1 (c) illustrates a feasible multi-commodity static flow and (d) shows the saving of unused arcs using partial contraflow. In Figure 1 arcs (u, t_2) and (v, t_1) represent fly over arcs without intersections. To reallocate the capacity of bundle constraint on arc (x, y), we use resource-directive decomposition method. This method solves the problem by reducing multi-commodity flow problem into the independent single-commodity flow problems by reallocating capacity of each arc to each commodity in such a way that optimal solution is obtained.

(a) capacity, transit time

(b) capacity, transit time

(c) flow/capacity, transit time

(d) flow, saved capacity
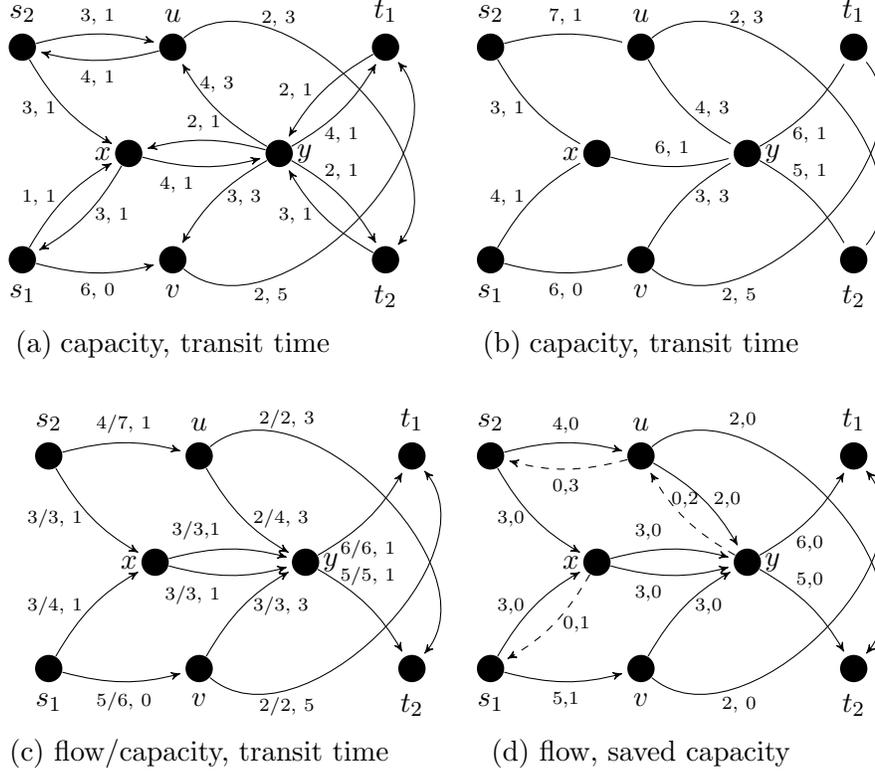
FIGURE 1. (b) Represents the auxiliary network of the given network (a). (c) Represents the static multi-commodity flow and (d) is the network having flow and saved capacity on the arcs.

2.4. **Temporally repeated flow:** Ford and Fulkerson [10] introduced the concept of temporally repeated flow to calculate maximum dynamic flow by temporally repeating the feasible static flow without intermediate storage. Let $\mathbf{P}_i$ be the collection of $s_i - t_i$ paths that carries the flow of commodity $i \in K$ such that $\cup_{i=1}^{k}\mathbf{P}_i = \mathcal{P}$, where $\mathcal{P}$ is the set of all multi-commodity paths. Let g^i be a feasible static $s_i - t_i$ flow. Then the path decomposition of the flow g^i is defined by

$$g_e^i = \sum_{P \in \mathbf{P}_i : e \in P} g_p^i.$$

Let τ_P be the transit time of every path $P \in \mathbf{P}_i$ which is bounded from above by T. Then for the feasible static flow g^i in $\mathbb{N}$ with path decomposition $(g_P^i)_{P \in \mathbf{P}_i}$, the temporally repeated flow sends flow g_P^i at constant rate in to path $P \in \mathbf{P}_i$ starting from time 0 to $T - \tau_P$.

For such flow g^i with decomposition $(g_P^i)_{P \in \mathbf{P}_i}$ such that $g_P^i = 0$ for all $P \in \mathbf{P}_i$ with $\tau_P > T$, the value of corresponding temporally repeated dynamic flow f^i for commodity i is

$$(2.10) \qquad |f^i| = \sum_{P \in \mathbf{P}_i} (T - \tau_P)g_P^i = T|g^i| - \sum_{e \in E} \tau_e g_e^i.$$

2.5. **Approximation scheme:** In multi-commodity flow problems, the linear programming techniques provide polynomial-time solutions. However, in many applications, these techniques can take a long time to solve the problems. So, approximation algorithms that provide solutions close to the optimal solution are the better options rather to find exact solutions. As a consequence, an intense attempt was made to obtain an efficient approximate algorithm for the multi-commodity flow problem.

Let the optimal solution of the objective function be denoted by $OPT(I)$, where $I \in \mathcal{X}$ is an instance of a minimization (or a maximization) problem $\mathcal{X}$. Suppose $\epsilon > 0$, an algorithm $\mathcal{A}$ is called a $(1 + \epsilon)$ (or $(1 - \epsilon)$) approximation algorithm for problem $\mathcal{X}$, if for each instances I of $\mathcal{X}$ it produces a feasible solution with objective value $\mathcal{A}(I)$ such that

$$|\mathcal{A}(I) - OPT(I)| \leq \epsilon \, OPT(I)$$

- For a problem $\mathcal{X}$, a polynomial-time approximation scheme (PTAS) is an approximation scheme having time complexity polynomial in the input size of the problem.
- For a problem $\mathcal{X}$, a fully polynomial-time approximation scheme (FPTAS) is an approximation scheme having time complexity polynomial in the input size of the problem and also polynomial in $1/\epsilon$.

3. Continuous-time QMCF with Partial Contraflow

We introduce the contraflow strategy to the continuous-time QMCF problem in this section. The continuous-time QMCCF problem shifts given amount of commodities from s_i to t_i in minimum time by reversing the direction of arcs towards the demand nodes without any processing cost. We also develop an efficient algorithm for an approximate solution to this problem. Moreover, our algorithm saves the capacity of lanes that are not necessary to reverse for reducing the time into quickest one. This extends the network flow models introduced in [8] and [12] into partial contraflow framework introduced in [5] and [27].

Problem 3.1. *Consider a network* $\mathbb{N} = (V, E, K, b, \tau, d_i, S, D, T)$. *The continuous-time QMCCF problem is to find the minimum possible time satisfying the constraints 2.4, 2.5, and 2.9 that transship given amount of commodities d_i from s_i to t_i with bounded cost for each $i \in K$ by reversing the direction of necessary arcs at time zero.*

$\mathcal{NP}$-hardness of the multi-commodity flow over time problem with or without intermediate node storage is shown by Hall et al. [12] even in case of series-parallel graphs or having only two commodities.

Theorem 3.2. *The continuous-time QMCCF problem with bounded cost is $\mathcal{NP}$-hard.*

Proof. The maximum dynamic multi-commodity flow problem is $\mathcal{NP}$-hard, [12]. It can be solved as a static flow problem using time-expanded graph in pseudo-polynomial-time complexity without restriction on intermediate node storage [24]. The quickest temporally repeated flow with bounded cost is strongly $\mathcal{NP}$-hard and does not allow FPTAS unless $\mathcal{P} = \mathcal{NP}$. As in [12], the quickest multi-commodity flow without intermediate node storage and simple flow paths holds the same hardness. The contraflow problem is $\mathcal{NP}$-complete which is shown in [16] by using 3-SAT. It follows the $\mathcal{NP}$-hardness of continuous-time QMCCF problem with bounded cost. $\square$

As continuous-time QMCF problem is $\mathcal{NP}$-hard, an approximate solution to this problem is presented in [8] by using two approaches: a length-bounded approach and condensed time-expanded network. We introduce the concept of partial contraflow in these two approaches and provide approximate algorithms to solve the Problem 3.1.

3.1. **Length bounded flow:** The multi-commodity path flow g^i satisfying demands and supplies d_i at terminals $S \cup D$ is a T-length bounded flow if the flow on each path $P \in \mathbf{P}_i$ can be decomposed into the sum of flows g^i_P, i.e., $g^i = \sum_{P \in \mathbf{P}_i} g^i_P$ with $g^i_P > 0$ such that the length $\tau_P = \sum_{e \in P} \tau_e \leq T$. The set of all T-length bounded paths is denoted by $\mathbf{P}^T_i = \{P \in \mathbf{P}_i : \tau_P \leq T\} \subseteq \mathbf{P}_i$. As T-length bounded static flow problem satisfying multi-commodity demands is $\mathcal{NP}$-hard, an approximate solution to this problem is presented in [8] within polynomial-time complexity. As in [15], we present Algorithm 1 to calculate approximate solution of continuous-time QMCCF problem.

Algorithm 1: Approximate Continuous-time QMCCF Algorithm

Input : Given multi-commodity flow network $\mathbb{N} = (V, E, K, b, \tau, d_i, S, D, T)$

Output: The continuous-time QMCCF

(1) The given network is transformed to auxiliary network by adding two-way capacities as $\mathbb{N}^a = (V, E^a, K, b^a, \tau^a, d_i, S, D, T)$ as

$$b^a_e = b_e + b_{\overleftarrow{e}}$$

$$\tau^a_e := \begin{cases} \tau_e & \text{if } e \in E \\ \tau_{\overleftarrow{e}} & \text{otherwise.} \end{cases}$$

(2) Compute the continuous-time QMCF on the transformed network $\mathbb{N}^a$ by using approximate length-bounded algorithm of [8] for quickest multi-commodity flow problem with bounded cost on auxiliary network.

(3) Decompose the flow along the $s_i - t_i$ paths and cycles and remove the cycles $\forall i$.

(4) Reverse $\overleftarrow{e} \in E$ up to the capacity $g_e - b_e$ iff $g_e > b_e$, b_e replaced by 0 whenever $e \notin E$, $\forall i$, where $g_e = \sum_{i=1}^k g^i_e$.

(5) For each $e \in E$, if $\overleftarrow{e}$ is reversed, $s_c(\overleftarrow{e}) = b^a_e - g_e$ and $s_c(e) = 0$. If neither e nor $\overleftarrow{e}$ is reversed, $s_c(e) = u_e - g_e > 0$, where $s_c(e)$ is the saved capacity of e.

(6) Transform the solution to the original network.

First, we present Lemma 3.3 to recall the solution procedure of the QMCF problem with bounded cost in Step 2 of Algorithm 1. Next for the correctness of Algorithm 1, we prove Theorem 3.4.

Lemma 3.3. *The T-length bounded approximate continuous-time QMCF with bounded cost can be computed on an auxiliary network in polynomial-time complexity.*

Proof: Consider an auxiliary network $\mathbb{N}^a$ with predetermined time horizon T, bounded cost C and precision $\epsilon > 0$. Then, a static flow g is computed as $g_e^i = \frac{1}{T} \int_0^T f_e^i(\theta) d\theta$ for all $e \in E^a$ and $i \in K$ by averaging the feasible dynamic flow f on each arc. The feasibility of g can be shown as follows.

- Capacity constraints:

$$\sum_{i \in K} g_e^i = \frac{1}{T} \int_0^T \left(\sum_{i \in K} f_e^i(\theta) \right) d\theta \leq \frac{1}{T} \int_0^T b_e^a d\theta = b_e^a$$

- Flow conservation:

$$\sum_{e \in E^+(v)} g_e^i - \sum_{e \in E^-(v)} g_e^i$$

$$= \frac{1}{T} \int_0^T \left(\sum_{e \in E^+(v)} f_e^i(\theta) - \sum_{e \in E^-(v)} f_e^i(\theta) \right) d\theta$$

$$= 0 \quad \forall v \in V \setminus \{s_i, t_i\} \text{ and } i \in K$$

Furthermore, flow g satisfies the following three conditions: it is (i) $(1+\epsilon)T$-length bounded, (ii) satisfies a fraction $\frac{1}{T}$ of the supplies and demands and (iii) $c(g) = \frac{c(f)}{T}$. We iterate the process until such flow exists. First condition follows from Dhamala et al. [5]. As $|g^i| = \frac{1}{T} \int_0^T |f^i(\theta)| d\theta \geq \frac{1}{T} d_i$ for all i, second condition is at hand and third condition is as similar to second one.

Now, flow is send into each path P with flow rate g_P^i by using T-length bounded path decomposition of g. The flow is pushed for next T time units so that total flow will reach at destination in at most $2T$ time. Thus, arbitrary feasible static flow g satisfying above requirements can be turned into feasible dynamic flow f with time horizon $2T$ meeting the same requirements as f.

By setting the cost $c_e = \tau_e$ on static flow g_e^i, we have

$$\sum_{e \in E^a} \tau_e \, g_e^i = \frac{1}{T} \sum_{e \in E^a} \tau_e \left(\int_0^T f_e^i(\theta) \right) d\theta = \frac{1}{T} c(f^i).$$

Due to T-length boundedness, flow can travel along the path with cost at most T and total cost is bounded above by $T|f^i|$. The temporally repeated flow with time horizon $2T$ can be calculated as

$$2T|g^i| - \sum_{e \in E^a} \tau_e \, g_e^i = 2|f^i| - \frac{1}{T} c(f^i) \geq |f^i| \geq d_i, \quad \forall i \in K,$$

where $|g^i| = \frac{|f^i|}{T}$. Thus, static flow g^i satisfies the demand d_i for all $i \in K$ within time $2T$.

By incorporating length boundedness into binary search, the optimal make-span T^* yields $T^* \leq T \leq (1 + \epsilon'/4)T^*$ for any $\epsilon' > 0$ to obtain T. Setting $\epsilon = \epsilon'/4$, we have dynamic flow with make-span $(2 + \epsilon)T \leq (2 + \epsilon')T^*$. $\square$

Theorem 3.4. *T-length bound approximate solution to the continuous-time QMCCF problem with bounded cost can be obtained by using Algorithm 1.*

Proof: The theorem will be proved in two parts, in the first part we show the feasibility and in the next part, we show the optimality of the algorithm. To show the feasibility of Algorithm 1, it is enough to show that Step 4 is well defined because Step 1-3 are feasible. By removing flow in cycles in Step 3, flow is either in direction $e = (v, w)$ or in $\overleftarrow{e} = (w, v)$ but not in both directions. Hence, the flow obtained from Step 4 is feasible flow with arc reversal in the network $\mathbb{N}$. The capacity of unused arcs are saved in Step 5 by reversing only necessary arcs in Step 4. Since the obtained flow obeys capacity constraint on each arc in the auxiliary network, the algorithm is feasible. Now we show the optimality. Depending on the feasibility, we conclude that an optimal solution to an approximate continuous-time QMCCF problem with bounded cost on two-terminal network $\mathbb{N}$ is also a feasible solution to the approximate continuous-time QMCF problem with bounded cost on the corresponding auxiliary network $\mathbb{N}^a$.

The solution of maximum contraflow problem with single source-sink can be computed in strongly polynomial-time complexity $O(mn + n^2m^3 \log n)$, [28]. By incorporating a parametric search of [4], a strongly polynomial-time bound for the quickest contraflow problem can be found. This problem can be solved with partial lane reversals and saves unused lane capacities in time $O(nm^2(\log n)^2)$, [27]. On the other hand, by separating the paths $\mathbf{P}_i$ from $s_i - t_i$ for each commodity $i \in K$, the multi-commodity flow problem can be reduced to a single commodity flow problem and can be solved as a single commodity flow problem. An approximate QMCF solution can be obtained optimally on the auxiliary network $\mathbb{N}^a$ by using Lemma 3.3. Moreover, any optimal solution on $\mathbb{N}^a$ is equivalent to a feasible solution to given network $\mathbb{N}$.

The unused capacities of the arcs are saved by using partial contraflow as described in Step 5 of Algorithm 1. Thus, an approximate QMCCF with bounded cost on each arc of the network $\mathbb{N}$ can be computed optimally. $\square$

Corollary 3.5. *An approximate solution of continuous-time QMCCF problem can be computed by using T-length bound in polynomial-time complexity.*

Proof: The complexity of Algorithm 1 is dominated by Steps 2 and 3. Step 3 is solved in $O(mn)$ time. According to [8], Step 2 is solved in polynomial-time. Since other remaining steps can be solved in linear time $O(m)$, the problem can be computed in polynomial-time complexity. $\square$

3.2. An FPTAS for the continuous-time QMCCF problem: Let us consider a multi-terminal network $\mathbb{N} = (V, E, K, b, \tau, d_i, S, D, T)$, where all parameters are integers. By rescaling the time by large time steps instead of the unit time step, a polynomial-time bound

is obtained. For $\Delta > 0$, If all transit times on arcs are multiple of Δ, then the condensed time-expanded network can be obtained by rescaling the time as $\mathbb{N}_T^\Delta = (V_T^\Delta, E_M^\Delta \cup E_H^\Delta)$ such that $\lceil T/\Delta \rceil$ is bounded by a polynomial in the input size, where the sets of nodes and arcs are defined by

$$
\begin{aligned}
V_T^\Delta &= \{v_{\alpha\Delta} : v \in V,\ \alpha = 0, 1, 2, ..., \lceil T/\Delta \rceil - 1\} \\
E_M^\Delta &= \{(v_{\alpha\Delta}, w_{\alpha\Delta + \tau_e}) : e = (v, w) \in E,\ \alpha = 0, 1, ..., \lceil (T - \tau_e)/\Delta \rceil - 1\} \\
E_H^\Delta &= \{(v_{\alpha\Delta}, v_{\alpha\Delta+1}) : e = (v, w) \in E,\ \alpha = 0, 1, ..., \lceil T/\Delta \rceil - 2\}.
\end{aligned}
$$

The copies of V_T^Δ corresponds to V in time $\mathbb{T} = \{\alpha\Delta\}$ for discrete-time and $\mathbb{T} = [\alpha\Delta, (\alpha + 1)\Delta)$ for continuous-time, where $\alpha = \{0, 1, 2, ..., \lceil T/\Delta \rceil - 1\}$. In this setting, capacities are rescaled by Δb_e. If T and τ_e are multiple of Δ for all $e \in E$, then any dynamic multi-commodity flow f in $\mathbb{N}$ that completes by time horizon T corresponds to a static multi-commodity flow g of equal value and cost in $\mathbb{N}_T^\Delta$ and vice-versa, [8]. The Δ-condense time-expanded network transforms into the classical time-expanded network by taking $\Delta = 1$. If arc transit times are not multiple of Δ, then transit times are scaled to multiple of Δ by $\tau_e' = \lceil \tau_e/\Delta \rceil \Delta$, for all arcs $e \in E$ with $0 \le \tau_e' - \tau_e < \Delta$.

We present Algorithm 2 as in [11] to solve Problem 3.1 by using Δ-condensed time-expanded network. A Δ-condensed auxiliary network is constructed by scaling capacities of arcs Δ times the sum of capacities of arcs e and $\overleftarrow{e}$ of the given network. Flows are sent through the transformed Δ-condensed auxiliary network $\mathbb{N}^{\Delta a}$ having transit time $\tau_e'^a$. By using FPTAS-Core of [8] and the lane reversal technique of [27], we solve the QMCCF problem.

Algorithm 2: FPTAS for continuous-time QMCCF problem

Input : Given multi-commodity flow network $\mathbb{N} = (V, E, K, b, \tau, d_i, S, D, T)$

Output: The continuous-time QMCCF

(1) The given network is transformed to auxiliary network by adding two-way capacities as $\mathbb{N}^{\Delta a} = (V, E^a, K, b^a, \tau'^a, d_i, S, D, T)$ as

$$b_e^a = \Delta(b_e + b_{\overleftarrow{e}})$$

$$\tau_e'^a := \begin{cases} \lceil \tau_e/\Delta \rceil \Delta & \text{if } e \in E \\ \lceil \tau_{\overleftarrow{e}}/\Delta \rceil \Delta & \text{otherwise.} \end{cases}$$

(2) Compute the continuous-time QMCF on Δ-condensed auxiliary network by using FPTAS Core of [8] for QMCF problem with bounded cost on auxiliary network.

(3) Decompose the flow along the $s_i - t_i$ paths and cycles and remove the cycles $\forall i$.

(4) Reverse $\overleftarrow{e} \in E$ up to the capacity $g_e - b_e$ iff $g_e > b_e$, b_e replaced by 0 whenever $e \notin E$, $\forall i$, where $g_e = \sum_{i=1}^k g_e^i$.

(5) For each $e \in E$, if $\overleftarrow{e}$ is reversed, $s_c(\overleftarrow{e}) = b_e^a - g_e$ and $s_c(e) = 0$. If neither e nor $\overleftarrow{e}$ is reversed, $s_c(e) = b_e - g_e > 0$, where $s_c(e)$ is the saved capacity of e.

(6) Transform the solution to the original network.

Lemma 3.6. *An FPTAS for continuous-time QMCF problem with bounded cost can be computed on an auxiliary network in fully polynomial-time.*

Proof: Let $\mathbb{N}$ be a single commodity network with demand d as input with tentative time horizon T and precision $\epsilon > 0$. To calculate bounds on optimal make-span T^*, a standard binary search framework is used, [8]. Based on lower and upper bound, an approximate T with $T^* \leq T \leq (1 + O(\epsilon))T^*$ is calculated. Assuming $0 \leq \tau_e' - \tau_e \leq \Delta$, and choosing appropriate $\Delta = \epsilon^2 T/n$ with $T' = \lceil (1 + \epsilon)^3 T/\Delta \rceil \Delta$, there exists a static flow g in the Δ-condensed time-expanded network $\mathbb{N}_{T'}^\Delta$ satisfying demand $(1 + \epsilon)d$. A dynamic flow f can be calculated in the network $\mathbb{N}$ with time horizon at most $(1 + \epsilon)T'$ satisfying the demand d with the help of g.

The process to find a static flow g in Δ-condensed time-expanded network satisfying demand $(1 + \epsilon)d$ at cost $(1 + \epsilon)C$ for the correctness of Step 2 of the algorithm is as follows. As τ'^a and T' are multiple of Δ and flow over time that completes by time T corresponds to a static flow of equal cost in $\mathbb{N}_{T'}^\Delta$. So, it suffices to show that, there exists a flow over time $\bar{f}$ with time horizon T' satisfying demand $(1 + \epsilon)d$ at cost $(1 + \epsilon)C$.

The storage of flow at intermediate nodes is not necessary for single-commodity flow problems with the cost. So, any flow f' on arc e at time θ can be written in the path flow form as

(3.1) $$f_e'(\theta) = \sum_{P \in \mathbf{P}: e \in P} f_P'(\theta - \tau_e(P)),$$

where $\tau_e(P)$ is the length of subpath obtained by removing arc e and its successors. This flow is reduced to a smooth dynamic flow by

(3.2) $$f_P^{sm}(\theta) = \frac{1}{\epsilon T} \int_{\theta - \epsilon T}^{\theta} f_P'(\zeta)d\zeta$$

for all $\theta \in [0, (1+\epsilon)^2 T + \epsilon T)$. This flow f_P^{sm} has a cost $c(f^{sm}) = c(f') \leq (1+\epsilon)^2 C$ and satisfies demands $(1+\epsilon)^2 D$. Since each path $P \in \mathbf{P}$ is simple, therefore $0 \leq \tau_P' - \tau_P \leq n\Delta = \epsilon^2 T$, and the flow is feasible for all $\theta \in [0, (1+\epsilon)^2 T + \epsilon T + \epsilon^2 T)$. Furthermore, taking $f^{sc} = f^{sm}/(1+\epsilon)$, flow f^{sc} is required flow satisfying demand $(1 + \epsilon)d$ at cost $(1 + \epsilon)C$ and time at most T' which corresponds to the static flow g. Similarly, for a given flow g, we can compute a flow over time f in $\mathbb{N}$ satisfying demand D cost bounded by C and time horizon bounded by $(1 + \epsilon)T'$. Hence, from $O(\log \frac{1}{\epsilon})$ static min cost flow computations in $\mathbb{N}_T^\Delta$ with $O(\frac{n^2}{\epsilon^2})$ nodes and $O(\frac{mn}{\epsilon^2})$ arcs, an approximate solution of quickest flow problem with bounded cost can be obtained. Now, the result is generalized to the case of multi-commodity flow. In this case, a flow that contains non-simple paths can be reduced to a simple path by delaying for some time σ using intermediate storage.

A path on network $\mathbb{N}$ having nodes $(v_0, v_1, \ldots, v_q) \in P^\sigma$ in the path based model is a path P^σ with delays $\sigma_j, j = 1, 2, \ldots, q$. To store flow at node v_j while moving forward from node v_j to node v_{j+1} on path P^σ, amount of time required is particularized by delay $\sigma_j, j = 1, 2, ..., q$. The dynamic flow f_{P^σ} is the decomposition of dynamic flow f on paths with the delay P^σ by

(3.3)
$$f_e(\theta) = \sum_{P^\sigma : e \in P} f_{P^\sigma}(\theta - \tau_e(P^\sigma)),$$

where, $\tau_e(P^\sigma)$ is the time with delays on path P^σ to reach the arc e.

Finally, after smoothing and scaling the flow function with delay we get,

(3.4)
$$f_{P^\sigma}^{sc+sm,i}(\theta) = \frac{1}{(1+\epsilon)\epsilon T'} \int_{\theta - \epsilon T}^{\theta} f_{P^\sigma}^i(\zeta)d\zeta.$$

We check whether $\mathbb{N}_T^\Delta$ contains continuous-time QMCF satisfying all demands d_i having bounded cost with intermediate node storage in each search step of the algorithm, and update the value of T. By using constant factor approximation algorithm, the lower bound can be computed in polynomial-time. Applying geometric mean binary search steps for static multi-commodity flow computations with bounded cost having the same number of nodes and arcs, the estimated T' can be found within $O(\log \frac{1}{\epsilon})$. $\square$

Theorem 3.7. *An FPTAS provides an approximate solution to the continuous-time QM-CCF problem with bounded cost by using Algorithm 2.*

Proof: The theorem will be proved if we can show the feasibility and optimality of the Algorithm 2. Step 1 and Step 6 are feasible as shown in Theorem 3.4. The feasibility of Step 2 is shown in Lemma 3.6 and the feasibility of Step 4 is assured by Step 3, which cancels the positive flow on cycles, so flow is in only one direction. Step 5 saves the unused condensed capacity of the arc obtained by Step 4. Thus, Algorithm 2 gives a feasible solution.

An optimal solution on two-terminal network $\mathbb{N}$ to Problem 3.1 is also a feasible solution to the approximate continuous-time QMCF on the corresponding auxiliary network $\mathbb{N}^{\Delta a}$ having the same bound on cost. By reducing dynamic multi-commodity flow to a static flow problem on the time-expanded network, the solution is computed in pseudo-polynomial time complexity, [30], whereas Δ-condensed time-expanded network provides approximate polynomial-time bound. On auxiliary network $\mathbb{N}^{\Delta a}$, an approximate quickest flow solution can be obtained optimally by Lemma 3.6. As in Theorem 3.4, we can reduce multi-commodity flow problem to single-commodity and relation of flow over time to the quickest flow problem. Moreover, any optimal solution on $\mathbb{N}^{\Delta a}$ is equivalent to the feasible solution to given network $\mathbb{N}$. Thus, an approximate continuous-time QMCCF solution with bounded cost can be computed optimally on given network $\mathbb{N}$. $\square$

Corollary 3.8. *An approximate continuous-time QMCCF can be computed in fully polynomial-time complexity.*

Proof: Since all other Steps except Steps 2 and 3 are computed in linear time, so the complexity of Algorithm 2 is dominated by Steps 2 and 3. Step 3 is solved in $O(mn)$ time. A $(1 + \epsilon)$ approximate solution of static multi-commodity flow problem with bounded cost can be obtained by $O(\log \frac{1}{\epsilon})$ computation in a $\mathbb{N}_T^\Delta$, [8]. Since there are $(\frac{n}{\epsilon^2})$ layers with nodes $(\frac{n^2}{\epsilon^2})$ and arcs $(\frac{mn}{\epsilon^2})$ in $\mathbb{N}_T^\Delta$, the problem can be computed in fully polynomial-time. $\square$

Example 3.9. Consider a two-way network represented in Figure (1)(a) having two commodities Commodity-1 and Commodity-2 which are to be transshipped from source-sink pairs $s_1 - t_1$ and $s_2 - t_2$, respectively. Let the demands for Commodity-1 and Commodity-2 be $d_1 = 10$ and $d_2 = 17$ units, respectively. We calculate the quickest time without contraflow from Figure (1)(a) in which the quickest time to satisfy the demand of Commodity-1 is $T = [7, 8)$ and that for Commodity-2 is $T = [6, 7)$. So the quickest time to satisfy both demands is $T = 8$. Similarly, the quickest time with contraflow (cf. Figure (1)(c)) for Commodity-1 and Commodity-2 are $T = [5, 6)$ and $T = [6, 7)$ so that the minimum time to satisfy both demands is $T = 7$. This shows that due to contraflow, time is improved by 12.5 percent.

When we use length-bounded approximation without contraflow (cf. Figure (1)(a)) for Commodity-1, paths with 5-length bound are essential (because, if we take 4-length bounded paths and push the flow up to next 4 time units, the demands will not be satisfied) and flow is pushed for the next 4 time intervals ($T = [0, 1), [1, 2), [2, 3)$ and $[3, 4)$) so that demand will satisfy within time $T = [8, 9)$. For Commodity-2, paths of 4-length bound are essential and flow is pushed for the next 4 time intervals so that demand will satisfy within the time interval $[7,8)$. So minimum time to satisfy both demands without contraflow is $T = 9$. Similarly, with contraflow configuration (cf. Figure (1)(c)), Commodity-1 needs 4-length bounded paths and flow is pushed for 2-time intervals so that the quickest time is $T = [5, 6)$. For Commodity-2, paths of 4-length bound are essential and flow is pushed for next 4 time intervals so that the quickest time is $T = [7, 8)$. The minimum time to satisfy both demands with contraflow is $T = 8$. Due to contraflow, 11.1 percent of the time is saved.

To solve the problem by using Δ-condensed network with $\Delta = 2$, we first present the network for Commodity-1 with condensed capacity and transit time in Figure (2)(b) and present condensed time-expanded network in Figure (2)(c). The process for Commodity-2 is similar.

Here, using Δ-condensed time-expanded network with $\Delta = 2$ with contraflow configuration, quickest time to fulfill demand d_1 is $T = [6, 8)$ (cf. Figure 2(c)). Similarly, we can calculate the quickest time for d_2 which takes the time $T = [8, 10)$ and the minimum time to satisfy both demands is $T = 10$. We calculate flow in Δ-condensed time-extended network without contraflow from Figure 1(a), the time $T = [8, 10)$ and $T = [10, 12)$ are required for Commodity-1 and Commodity-2, respectively. So minimum time to satisfy both demands is $T = 12$. It shows that due to contraflow configuration, the quickest time is improved by 16.66 percent.

We summarize the Example 3.9 from the following table.

4. Conclusion

In day to day life, transporting several commodities from one place to another is essential. One of the major problems in operations research is the minimization of time (cost) for the transshipment of these commodities from the supply nodes to the demand nodes. The quickest flow problem has been investigated to fulfill the demand in minimum possible

(a) capacity, transit time

(b) condensed capacity, transit time

(c) condensed time-expanded network, taking $\Delta = 2$.
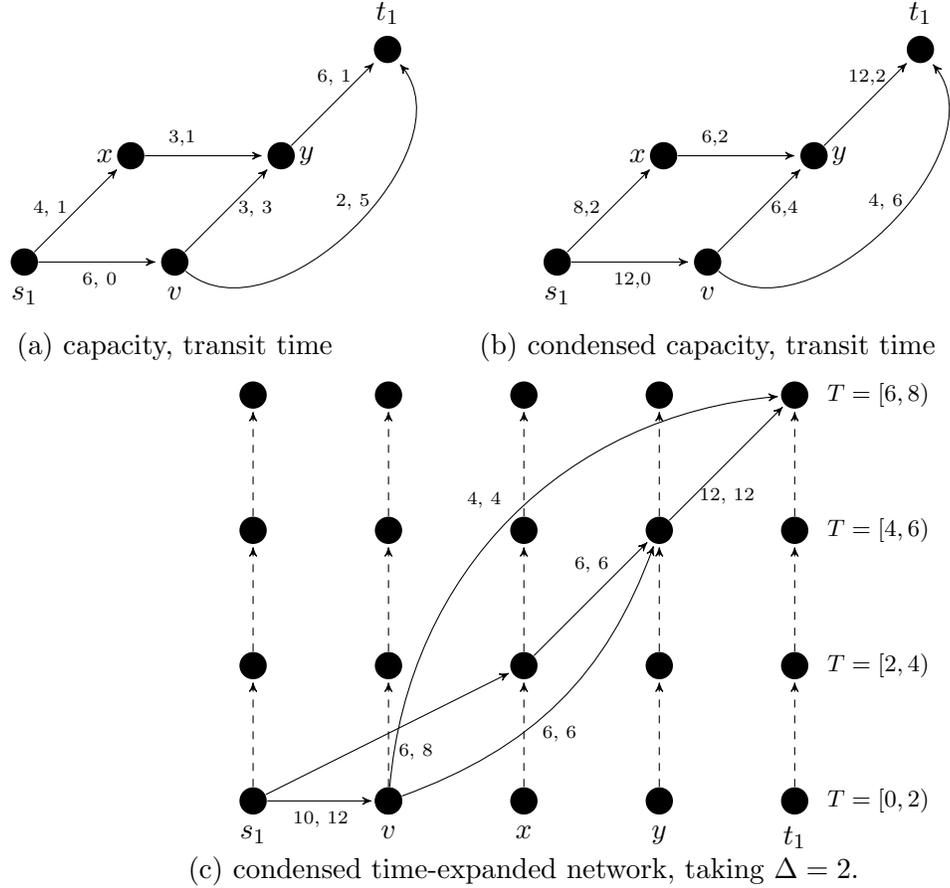
FIGURE 2. (a) Network for Commodity-1 after contraflow. (b) Δ-condensed network with condensed capacity and transit times on the arcs. (c) Δ-condensed time-expanded network with $\Delta = 2$ for Commodity-1.

TABLE 1. Quickest time with and without lane reversals

case	without LR	with LR	%Change
Normal	8	7	12.5
Length bound	9	8	11.1
Δ-condensed	12	10	16.6

LR = Lane Reversals

time. In case of a single-commodity, efficient algorithms to solve quickest flow problems are presented by the different researchers but the multi-commodity flow problem is $\mathcal{NP}$-hard. Moreover, a polynomial-time approximation algorithm by length-bounded function and an FPTAS-Core by using Δ-condensed time-expanded network have been obtained.

Partial contraflow strategy is an important approach to reduce the time horizon by expanding the capacity of the lanes in which only necessary arc capacities are reversed towards demand nodes. It saves the capacity of unused arcs that can be used for logistic supports in emergency periods.

In this research, we investigated the continuous-time quickest multi-commodity flow problem with partial contraflow. We introduced its mathematical model and presented two algorithms, one a polynomial-time approximation and another an FPTAS. As we have investigated the problem with constant transit time, our further interest is to extend these approaches for flow-dependent, time-dependent and load-dependent attributes.

5. Acknowledgements

References

[1] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network flows, theory algorithm and applications*, Prentice Hall, Englewood Cliffs, 1993.

[2] A. Ali, R. Helgason, J. Kennington, and H. Lall, Computational comparison among three multi-commodity network flow algorithms, *Operations Research*, Vol. 28(4), pp 995-1000, 1980.

[3] A. Assad, Multi-commodity network flows a survey, *Networks*, Vol. 8(1), pp 37-91, 1978.

[4] R. E. Burkard, K. Dlaska, and B. Klinz, The quickest flow problem, *ZOR-Methods and Models of Operations Research*, Vol. 37, pp 31-58, 1993.

[5] T. N. Dhamala, S. P. Gupta, D. P. Khanal, and U. Pyakurel, Quickest multi-commodity flow over time with partial lane reversals, *Journal of Mathematics and Statistics*, Vol. 16, pp 198-211, 2020.

[6] 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(3), pp 101-133, 2018.

[7] R. C. Dhungana and T. N. Dhamala, Flow improvement in evacuation planning with budget constrained switching costs, *International Journal of Mathematics and Mathematical Sciences-Hindawi*, Vol. 2020, Article ID 1605806, https://doi.org/10.1155/2020/1605806.

[8] L. Fleischer and M. Skutella, Quickest flows over time, *SIAM Journal on Computing*, Vol. 36(6), pp 1600-1630, 2007.

[9] L. Fleischer and E. Tardos, Efficient continuous-time dynamic network flow algorithms, *Oper. Res. Letter*, Vol. 23, pp 71-80, 1998.

[10] L. R. Ford and D. R. Fulkerson, *Flows in networks*, Princeton University Press, Princeton, New jersey, 1962.

[11] S. P. Gupta, U. Pyakurel, and T. N. Dhamala, An FPTAS for continuous-time QMCF problem with partial lane reversals, submitted to proceedings of International Webinar RAPAM-2020, Kurseong College, Darjeeling, India.

[12] A. Hall, S. Hippler, and M. Skutella, Multi-commodity flows over time: efficient algorithms and complexity, *Science Direct*, Vol. 379, pp 387-404, 2007.

[13] B. Hoppe and E. Tardos, The quickest transshipment problem, *Mathematics of Operations Research*, Vol. 25, pp 36-62, 2000.

[14] J. Kennington, A survey of linear cost multi-commodity network flows, *Operations Research*, Vol. 26(2), pp 209-236, 1978.

[15] D. P. Khanal, U. Pyakurel, and T. N. Dhamala, Approximation to Quickest Multi-commodity contraflow over time with length bound, submitted to proceedings of International Webinar RAPAM-2020, Kurseong College, Darjeeling, India.

[16] S. Kim, S. Shekhar, and M. Min, Contraflow transportation network reconfiguration for evacuation route planning, *IEEE Trans. Knowl. Data Eng.*, Vol. 20, pp 1115-1129, 2008.

[17] K. Langkau, Flow over time with flow-dependent transit times, Ph D thesis, T.U. Berlin, 2003.

[18] H. N. Nath, U. Pyakurel, T. N. Dhamala, and S. Dempe, Dynamic network flow location models and algorithms for evacuation planning, *Journal of Industrial and Management Optimization, 2020*, doi:10.3934/jimo.2020102.

[19] U. Pyakurel and T. N. Dhamala, Models and algorithms on contraflow evacuation planning network problems, *International Journal of Operations Research*, Vol. 12, pp 36-46, 2015.

[20] U. Pyakurel and T. N. Dhamala, Continuous time dynamic contraflow models and algorithms, *Advances in Operations Research-Hindawi*, Article ID 368587, pp 1-7, 2016.

[21] U. Pyakurel and T. N. Dhamala, Continuous dynamic contraflow approach for evacuation planning, *Annals of Operations Research*, Vol. 253(1), pp 573-598, 2017a.

[22] U. Pyakurel and T. N. Dhamala, Evacuation planning by earliest arrival contraflow, *Journal of Industrial and Management Optimization*, Vol. 13, pp 487-501, 2017b.

[23] U. Pyakurel, T. N. Dhamala, and S. Dempe, Efficient continuous contraflow algorithms for evacuation planning problems, *Annals of Operations Research (ANOR)*, Vol. 254, pp 335-364, 2017c.

[24] U. Pyakurel, S. P. Gupta, D. P. Khanal, and T. N. Dhamala, Efficient algorithms on multi-commodity flow over time problems with partial lane reversals, *International Journal of Mathematics and Mathematical Sciences, Hindawi*, Vol. 2020, Article ID 2676378, 13 pages https://doi.org/10.1155/2020/2676378.

[25] U. Pyakurel, H. N. Nath, and T. N. Dhamala, Efficient contraflow algorithms for quickest evacuation planning, *Science China Mathematics*, Vol. 61, pp 2079-2100, 2018.

[26] U. Pyakurel, H. N. Nath, and T. N. Dhamala, Partial contraflow with path reversals for evacuation planning, *Annals of Operations Research (ANOR)*, Vol. 283, pp 591-612, 2019a.

[27] U. Pyakurel, H. N. Nath, S. Dempe, and T. N. Dhamala, Efficient dynamic flow algorithms for evacuation planning problems with partial lane reversal, *Mathematics*, Vol. 7, pp 1-29, 2019b.

[28] 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, pp 200-216, 2010.

[29] K. Salimifard and S. Bigharaz, The multi-commodity network flow problem: state of the art classification, applications, and solution methods, *Springer*, pp 1-47, 2020.

[30] M. Skutella, An introduction to network flows over time. *In: Research Trends in Combinatorial Optimization, Springer*, pp 451-482, 2009.

[31] I.-L. Wang, Multi-commodity network flows: A survey, part I: Applications and formulations, *International Journal of Operations Research*, Vol. 15(4), pp 145-153, 2018.