Journal of Institute of Science and Technology, 30(1), 243-254 (2025)
ISSN: 2467-9062 (print), e-ISSN: 2467-9240
https://doi.org/10.3126/jist.v30i1.74913

Research Article

MAXIMUM FLOW LOCATION (FLOWLOC) MODELING WITHOUT SIZE RESTRICTIONS ON FACILITIES

Hari Nandan Nath
Bhaktapur Multiple Campus, Tribhuvan University, Bhaktapur, Nepal
Correspondence: hari.nath@bkmc.tu.edu.np

(Received: *February 02, 2025;* **Final Revision:** *June 27, 2025;* **Accepted:** *June 28, 2025***)**

ABSTRACT

There are situations when traffic flow is obstructed because of facility placement in the road segments. FlowLoc models deal with such a situation in which a set of facilities with given sizes (capacities) has to be allocated with a given set of arcs of a directed network. The existing models assume that the size of each facility does not exceed the capacity of each of the arc in the given set of arcs and that the number of given facilities does not exceed the total number of available locations on arcs. In this work, we extend the FlowLoc modeling in which facilities can have arbitrary sizes and can have arbitrary numbers. We construct mixed-integer programming models based on static and dynamic network flow modeling. Realizing NP-hardness of problems, we design polynomial time heuristic algorithms to solve the constructed models. The performance of the algorithms is tested in a road network of Bhaktapur city Nepal.

Keywords: Dynamic FlowLoc, Heuristics, Network flow, Static FlowLoc

2020 Mathematics Subject Classification: Primary: 90B10, 90C27, 68Q25; Secondary: 90B06, 90B20

INTRODUCTION

Modern civilization cannot be imagined without transportation facilities. Motor vehicles and road networks have become integral parts of human life. But because of the limited capacity of road networks and the growing number of vehicles, day by day, traffic jams have become a common problem for everyone in urban cities.

Growth of number of vehicles and road traffic congestion, as a consequence, has become one of the characteristics of urban life in developing countries. There are several consequences of traffic congestion, including human capital hour loss and increased fuel consumption.

Transportation planning is one of the major components of urban planning. A well managed transportation plan is important not only to deal with traffic congestion but also to build an equitable and inclusive city that ensures accessibility, safety, and reduces the environmental health risks of its residents.

In spite of scientific and technological advancements, the world is facing natural and human created hazards leading to large-scale disasters (Laporte *et al.*, 2019). All countries, whether they are developing or developed, face hazards like earthquakes, floods, landslides, fire, windstorm, hailstorm, epidemics, and avalanches periodically. Facilities allocation and traffic management are among the challenging issues in such situations. Not only in emergency situations, traffic management is challenging in day-to-day life also.

As in many applications, the use of mathematical modeling is growing day by day for transportation management and route guidance. The common modeling approaches in the area are based on network flow theory, traffic simulation, fluid dynamics, control theory, and variational inequalities. There are pros and cons of each approach. As the transportation networks are very large, the network flow modeling is the most promising approach (Köhler *et al.*, 2002).

The early network flow problems arise as transportation problems in late 1930s to 1940s (Ahuja *et al.*, 1993) . The development of the early contributions can be found in (Ford & Fulkerson, 1962). Mathematically, the problems are defined on graphs that consist of as set of nodes and set of arcs (consisting of pairs of nodes) with some attributes like capacity, cost , etc. The major network flow problems (but not limited to) are: shortest path problems, maximum static/dynamic flow problems, quickest flow problems and minimum cost flow problems. For a comprehensive study of the problems, their variations and applications, we refer to (Ahuja *et al.*, 1993).

One of the flourishing research areas, as an application of network flows, is emergency transportation planning. For the review of such kinds of problems, we refer to (Hamacher & Tjandra, 2001; Cova & Johnson, 2003; Kotsireas *et al.*, 2015; Akter & Wamba, 2019; Dhamala *et al.*, 2018). Current research in the area also considers contraflow problems dealing with the re- versal of the necessary arcs during the evacuation process (Kim *et al.*, 2008). One of the flourishing research areas, as an application of network flows, is emergency transportation planning. For the review of such kind of problems, we refer to (Hamacher & Tjandra, 2001 ; Cova & Johnson, 2003; Kotsireas, Nagurney, & Pardalos, 2015; Akter & Wamba, 2019; Dhamala, Pyakurel , & Dempe, 2018). Current research in the area also considers contraflow problems dealing with the reversal of the necessary arcs during the evacuation process (Kim *et al.*, 2008; Rebennack *et al.*, 2010; Pyakurel & Dhamala, 2017; Pyakurel *et al.*, 2018) , and flows with storage in intermediate nodes (Pyakurel & Dempe, 2019, 2020; Bhandari *et al.*, 2020; Dhamala *et al.*, 2023; Dhamala *et al.*, 2024).

One of the flourishing research areas, as an application of network flows, is emergency transportation planning. For the review of such kind of problems, we refer to (Hamacher & Tjandra, 2001; Cova & Johnson, 2003; Kotsireas *et al.*, 2015; Dhamala *et al.*, 2018; Akter & Wamba, 2019) . Current research in the area also considers contraflow problems dealing with the reversal of the necessary arcs during the evacuation process (Kim, *et al.*, 2008). Locating the facilities on networks or otherwise are studied under facility location problems which consist of choosing the "best" location for one or several facilities to satisfy demands from a given set of points (Laporte *et al.*, 2019) . The problems can be classified as continuous or discrete as the optimal facilities can be placed at any point in a given space, or at a point from among a given set of points in a space. If the facilities are to be placed on nodes or edges of a network, the problem is known as a network location problem. The problems are broadly classified as median problems (Hakimi, 1965), center problems (Hakimi, 1965), covering problems (Berge, 1957; Hakimi, 1965), dispersion problems (Kuby, 1987), and location routing problems (Balakrishnan *et al.*, 1987).

In normal, as well as in emergency situations, the traffic has to be blocked in some road segments because of placement of required facilities or using road segments for other purposes (parking, etc.) reducing the capacity of the corresponding segment. The allocation of facilities or the blockage of the arc capacities has to be done carefully. The combination of network flow modeling with location modeling can be crucial to deal with traffic management in such situations.

Combining network flow decisions with facilities allocation is one of the growing areas of research. As a relatively new area of modeling, the related problems consider identification of facility locations on the basis of the optimal network flow decisions. Hamacher *et al.* (2013) introduce FlowLoc modeling to identify the optimal facility locations on arcs of a network so as to maximize the value of the static and dynamic flow. They formulate the problems as mixed integer linear programming problems and devise exact algorithms for single facility cases and efficient heuristics for multiple facility cases. In similar settings, (Nath *et al.*, 2021) formulate the problem to minimize the time horizon of the flow and devise the exact algorithms and efficient heuristics with and without arc reversals. (Heßler & Hamacher, 2016; Nath & Dhamala, 2018) consider the identification of optimal shelters in evacuation planning along with the flow optimization. Considering the fact that the paths from the safe areas to the danger zones may be blocked because of the traffic flow directed towards the safe areas allowing reversal of the usual traffic flow, (Nath *et al.*, 2021; Nath *et al.*, 2024) consider the problem of identification of an optimal path for facility movement towards the source along with source-sink flow optimization.

In the FlowLoc models, a directed network is taken with given arc-capacities, and with or without transit time on arcs. A set P of facilities with given sizes and a set of arcs L on which the facilities are to be placed are also given. The problem is to identify the optimal allocation of P to L that maximizes the amount of flow (Hamacher *et al.*, 2013) or minimizes the time horizon of the given amount of flow (Nath *et al.*, 2021). In the models considered, (i) all the facilities are exhaustibly allocated to the arcs, and (ii) each facility can be placed on any of the arcs in L. This is possible only when the capacity of the largest facility does not exceed the capacity of each of the arcs in L and the number of facilities does not exceed the total number of facility locations. The main purpose of this work is to extend the maximum static and maximum dynamic FlowLoc models to the cases with arbitrary number of facilities in F with arbitrary sizes. This results in the modification of the objective functions and modification/addition of constraints in the existing FlowLoc models.

The rest of the text is organized as follows. In Section 2, we recapitulate the basic models on which the present work is based upon. In Section 3, we develop maximum static/dynamic FlowLoc models with arbitrary facilities and their solution procedures. In Section 4 we present the computational performance of the developed solution algorithms, and in Section 5, we conclude the paper.

BASIC MODELS

Let us consider a directed network with set of vertices V and set of arcs $A \subseteq V \times V$. Associated with each arc $(i, j) \in A$ is the capacity $u_{ij} \geq 0$. We write the vector $(u_{ij})_{(i,j)\in A} = u$. In network flows, we consider problems in which something called flow that is considered to move from node i to node j of each arc $(i, j) \in A$. The amount of flow that moves from i to j is denoted by x_{ij}. The flow is considered to be generated at the source node $s \in V$ and moves towards the sink node $t \in V$. We denote such a network as $G = (V, A, s, t, u)$. In many applications, associated with each arc (i, j) is the travel time τ_{ij} also. In such cases, x_{ij} denotes the rate of flow per unit time. We denote such a network as $G = (V, A, s, t, u, \tau)$ where $\tau = (\tau_{ij})_{(i,j)\in A}$. We define

$$V_i^+ = \{j \in V | (i, j) \in A\}, V_i^- = \{j \in V | \ (j, i) \in A\}$$

which are the sets of arcs directed out from $i \in V$, and directed towards $i \in V$, respectively.

Our modeling approach depends, mainly, on two network flow problems: the maximum static flow problem (also known as the maximum flow problem) and the maximum dynamic flow problem (also known as the maximum flow over time problem).

Maximum Static and Dynamic Flow Models

Given $G = (V, A, s, t, u)$, the maximum static flow problem can be stated as

$$\max v \tag{1}$$

subject to

$$\sum_{j \in V_i^+} x_{ij} - \sum_{j \in V_i^-} x_{ji} = \begin{cases} v, & i = s \\ 0, & i \notin V - \{s, t\} \\ -v, & i = t \end{cases} \tag{2}$$

$$0 \leq x_{ij} \leq u_{ij} \tag{3}$$

where v is called the value of x. Various algorithmic approaches including those running in strongly polynomial time of are available in literature (Ahuja *et al.*, 1993; Orlin, 2013).

Given a network $G = (V, A, s, t, u, \tau)$, a time horizon T, if X_{ij}^{θ} is a dynamic flow on the arc (i, j) at time $\theta \in [0, T)$, then the excess of the flow X at a node $i \in V$ at time θ is given by

$$excess_X(i, \theta) = \sum_{j \in V^-} \int_0^{\theta - \tau_{ij}} X_{ij}^{\xi}\, d\xi - \sum_{j \in V_i^+} \int_0^{\theta} X_{ji}^{\xi}\, d\xi \tag{4}$$

The value of the maximum dynamic flow X over the time horizon T is $excess_X(t, T)$ and the maximum dynamic flow problem can be stated as

$$excess_X(t, T) = \sum_{j \in V^-} \int_0^{\theta - \tau_{ij}} X_{ij}^{\xi}\, d\xi - \sum_{j \in V_i^+} \int_0^{\theta} X_{ji}^{\xi}\, d\xi \tag{5}$$

subject to

$$excess_X(i, \theta) \geq 0\ \forall i \in V - \{s\}, \forall \theta \in [0, T) \tag{6}$$

$$excess_X(i, T) = 0\ \forall i \in V - \{s, t\} \tag{7}$$

$$0 \leq X_{ij}^{\theta} \leq u_{ij}\ \forall (i, j) \in A, \forall \theta \in [0, T) \tag{8}$$

However, given a static flow rate xij with value v, it has been proved that the amount of flow accumulated at t from s is $Tv - \sum_{(i,j) \in A} \tau_{ij} x_{ij}$ (Ford & Fulkerson, 1962; Skutella, 2009) and hence the maximum dynamic flow problem can be stated as

$$\max Tv - \sum_{(i,j) \in A} \tau_{ij} x_{ij} \tag{9}$$

The problem is equivalent to the minimum cost circulation problem (Goldberg & Tarjan, 1989) in a network with added infinite capacity arc (t, s) in G with cost τ_{ij} for each arc $(i, j) \in A$ and $-T$ for (t, s). Note that if θ is considered to take discrete values $0, 1, \cdots, T$, then T in (9) has to be replaced by $T + 1$.
From constraint (2), we have

$$v = \sum_{j \in V_s^+} x_{sj} - \sum_{j \in V_s^-} x_{js}$$

and adding all the constraints (2) having 0 on the right hand side, we get

$$\sum_{j \in V_s^+} x_{sj} - \sum_{j \in V_s^-} x_{js} = - \sum_{j \in V_t^+} x_{sj} + \sum_{j \in V_t^-} x_{jt}.$$

So, we can state the maximum flow problem as

$$\max \sum_{j \in V_s^+} x_{sj} - \sum_{j \in V_s^-} x_{js} \tag{10}$$

subject to

$$\sum_{j \in V_s^+} x_{ij} - \sum_{j \in V_s^-} x_{ji} = 0\ \forall i \in V - \{s, t\} \tag{11}$$

alongwith the constraint (3). Likewise, we can state the maximum flow problem as

$$\max T\left(\sum_{j \in V_s^+} x_{sj} - \sum_{j \in V_s^-} x_{js} \right) - \sum_{(i,j) \in A} \tau_{ij} x_{ij} \tag{12}$$

subject to the constraints (3) and (11).

Maximum Static and Maximum Dynamic FlowLoc Models

Consider the situation in which facilities are to be allocated to the arcs so that the capacity of the arc which hosts the facility is reduced by the size of the facility. This will reduce the capacity of the arcs that host the facilities, and this may result in a reduction in the value of the flow (static or dynamic). The facilities are to be allocated in such a way that the reduction in the value of the flow is the least.

Let $\mathbb{P}$ be the set of facilities with size r_p of the facility $p \in P$ and $L \subseteq A$ be the set of arcs that can host the facilities such that $(i, j) \in L$ can host η_{ij} facilities. If facilities $p_{i_1}, p_{i_2}, \cdots, p_{i_k}$ are assigned to an arc $(i, j) \in L$, the capacity of (i, j) is reduced to $u_{ij} - \max\{p_{i_1}, \cdots, p_{i_k}\}$ which may reduce the value of the maximum static or dynamic flow. The maximum static $|\mathbb{P}|$-FlowLoc model is stated as follows (Hamacher *et al.*, 2013).

$$\max \sum_{j \in V_s^+} x_{sj} - \sum_{j \in V_s^-} x_{js} \tag{13}$$

subject to

$$\sum_{j\in V_s^+} x_{ij} - \sum_{j\in V_s^-} x_{ji} = 0 \; \forall i \in V - \{s,t\} \tag{14}$$

$$0 \le x_{ij} \le u_{ij} \; \forall (i,j) \in A \tag{15}$$

$$x_{ij} + r_p y_{ijp} \le u_{ij} \; \forall (i,j) \in L, p \in \mathbb{P} \tag{16}$$

$$\sum_{p\in\mathbb{P}} y_{ijp} \le \eta_{ij} \; \forall (i,j) \in L \tag{17}$$

$$\sum_{p\in\mathbb{P}} y_{ijp} = 1 \; \forall p \in \mathbb{P} \tag{18}$$

$$y_{ijp} \in \{0,1\} \; \forall (i,j) \in L \; \forall p \in \mathbb{P} \tag{19}$$

Given the time horizon T, the dynamic FlowLoc model maximizes

$$T\left(\sum_{j\in V_s^+} x_{sj} - \sum_{j\in V_s^-} x_{js}\right) - \sum_{(i,j)\in A} \tau_{ij} x_{ij}$$

along with the constraints (14) – (19).

The single facility case of the problems with $|\mathrm{P}| = 1$ can be solved in strongly polynomial time. This can be realized with the help of Algorithm 1 adapted from (Hamacher *et al.*, 2013) which can be used to solve the single-facility maximum static FlowLoc problem and that a maximum static flow problem can be solved in strongly polynomial time (Orlin, 2013).

Algorithm 1: Single-Facility Maximum FlowLoc Algorithm

Input: A directed network *G = (V, A, s, t, u)*, set of possible locations L and a single facility p with size r_p.
Output: Location *loc*(p) to place the facility p.

Calculate the maximum static flow x in G
if $u_{ij} - x_{ij} \ge r_p$ for some $(i,j) \in L$ **then**
  $loc(p) := (i,j)$
  return $loc(p)$
end if
$loc(p) := None, maxv = -1$
for $(i,j) \in L$ **if** $r_p \le u_{ij}$ **do**
  $u_{ij} := u_{ij} - r_p$
  $v :=$ value of the maximum static flow in G
  if $v > maxv$ **then**
    $loc(p) := (i,j)$
    $maxv := v$
  end if
  $u_{ij} := u_{ij} + r_p$
end for return $loc(p)$

If maximum flow calculations in Algorithm 1 are replaced by the maximum dynamic flow calculations, we get the solution of the single-facility maximum dynamic FlowLoc problem. Since the maximum flow calculations and maximum dynamic flow calculations (using minimum cost flow) can be done in strongly polynomial time (Orlin, 1993), and they are to be performed $|L| \le |A|$ times in the worst case, Algorithm 1 has strongly polynomial running time, and we have the following result.

Theorem 1. *The single-facility maximum static/dynamic FlowLoc problem can be solved in strongly polynomial time.*
However, both the maximum static and maximum dynamic FlowLoc problems are NP-hard for $|\mathbb{P}| > 1.$ Moreover, the following theorem rules out the possibility of polynomial-time approximation algorithms for the problems.

Theorem 2 (Hamacher *et al.*, 2013). There is no polynomial time α-approximation algorithm for maximum static $|\mathbb{P}|$-FlowLoc problem with a finite constant α, unless $P = NP$ for $|\mathbb{P}| > 1$.

The proofs of NP-hardness and Theorem 2 are given in (Hamacher *et al.*, 2013). Since the maximum static FlowLoc problem is a special case of a dynamic FlowLoc problem with $T = 1$, $\tau_{ij} = 0 \; \forall (i,j) \in A$ which can be realized by putting $T = 1$ and $\tau_{ij} = 0$ in (9) ($T = 0$ if $[0, T)$ has to be discretized into $0, 1, \cdots, T$), the same is true for maximum dynamic $|\mathbb{P}|$-FlowLoc problem also.

The above-mentioned FlowLoc models inherently assume the following:

1. The number of facilities in $\mathbb{P}$ is not more than the total number of locations available, i.e., $|\mathbb{P}| \le \sum_{(i,j)\in L} \eta_{ij}$ which is reflected in the constraint (18) which necessitates each facility to be placed in exactly one of the arcs in L.
2. Each facility in $\mathbb{P}$ can be placed in any of the arcs in L as stated by (16). This is only possible when $\max_{p\in\mathbb{P}} r_p \le u_{ij} \; \forall (i,j) \in L$.

In this work, we develop the maximum static and dynamic FlowLoc models relaxing the above-mentioned two conditions.

The MAXIMUM FLOWLOC PROBLEMS WITH FACILITIES OF ARBITRARY SIZES

In this section, we extend the existing maximum FlowLoc models to address the issues raised in the last paragraph of Section 2. The notations introduced in Section 2 for modeling are listed below.

Sets

V	set of nodes
A	set of arcs
V_i^+	set of nodes succeeding the node i
V_i^-	set of nodes preceding the node i
L	set of arcs which can host the facilities
$\mathbb{P}$	set if facilities

Parameters

s	the source node
t	the sink node
u_{ij} $(i,j) \in A$	capacity of arc (i,j)
τ_{ij} $(i,j) \in A$	travel time of from i to j
η_{ij} $(i,j) \in L$	number of facilities that $(i,j) \in L$ can host
r_p $p \in \mathbb{P}$	size of the facility p
T	time horizon
M	a sufficiently big number

Variables

x_{ij} $(i,j) \in A$	static flow (rate) on arc
y_{ijp} $(i,j) \in L, p \in \mathbb{P}$	binary variable, $y_{ijp} = 1$ if facility p is assigned to (i,j)

Maximum Static FlowLoc Model with Facilities of Arbitrary Sizes

Example 1 Consider a dummy network as shown in Fig. 1. The labels against the arcs are the capacities of the respective arcs. Let $L = \{(a,b),(s,a),(s,b)\}$ with number of locations $\eta_{ab} = \eta_{sa} = 2, \eta_{sb} = 1$. Let $\mathbb{P} = \{p_1, p_2, p_3, p_4, p_5\}$ be the set of facilities with sizes $r_{p_1} = 2, r_{p_2} = r_{p_3} = 3, r_{p_4} = 4, r_{p_5} = 5$.

Considering the number of facilities, and the number of locations on L to place the facilities, we can assign at most five facilities. However, since the capacity of (a,b) is only 2, we can only assign p_1 to it. So, the maximum number of facilities that can be assigned reduces to four. If we assign p_1 to (a,b), p_2, p_3 to (s,a), and p_4 to (s,b), the maximum flow value is 6. On the other hand, if we assign p_1 to (a,b), p_2 to (s,b), and p_3, p_4 to (s,a), the maximum flow value is 7. Moreover, if we assign p_1, p_2 to (s,a), p_3 to (s,b), we can assign only three facilities because the size of p_4 is more than the capacity of (a,b).

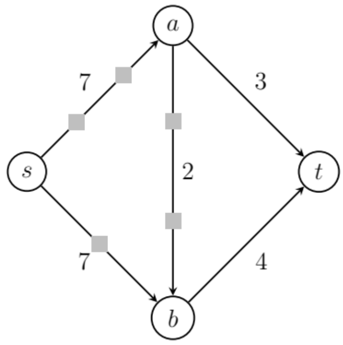

Figure 1. A Dummy Network

To model the problem to maximize the flow with the maximum number of facilities allocations, in the static FlowLoc model (13)–(19), we modify the objective function, add a constraint (23) and replace (18) by (26) which does not necessitate all the facilities to be exhausted. Since the allocation of facilities to arcs reduces the capacities of the arcs, there may be a reduction in the maximum flow. The lack of strict equality in (26) may give a solution that does not assign any facility to the arcs to maximize the flow if the objective function is kept as (13). To address the issue, we add $M \sum_{(i,j)\in L} \sum_{p\in\mathbb{P}} y_{ijp}$ to the objective of the maximum flow model so that appropriate value of M would guarantee the assignment of maximum number of facilities so that the objective maximizes the flow and maximizes the number of assigned facilities simultaneously. The model, which is a mixed integer linear program(MILP), is presented below.

Objective

$$\max \sum_{j\in V_s^+} x_{sj} - \sum_{j\in V_s^-} x_{js} + M \sum_{(i,j)\in L} \sum_{p\in\mathbb{P}} y_{ijp} \tag{20}$$

Constraints

$$\sum_{j\in V_i^+} x_{ij} - \sum_{j\in V_i^-} x_{ji} = 0 \ \forall i \in V - \{s,t\} \tag{21}$$

$$0 \le x_{ij} \le u_{ij} \ \forall (i,j) \in A \tag{22}$$

$$r_p y_{ijp} \le u_{ij} \ \forall (i,j) \in L \ \forall p \in \mathbb{P} \tag{23}$$

$$x_{ij} + r_p y_{ijp} \le u_{ij} \ \forall (i,j) \in L \ \forall p \in \mathbb{P} \tag{24}$$

$$\sum_{p\in\mathbb{P}} y_{ijp} \le \eta_{ij} \ \forall (i,j) \in L \tag{25}$$

$$\sum_{(i,j)\in L} y_{ijp} \leq 1 \;\forall p \in \mathbb{P} \tag{26}$$

$$y_{ijp} \in \{0, 1\} \;\forall (i,j) \in L \;\forall p \in \mathbb{P} \tag{27}$$

The objective (20) maximizes the value of the static s–t flow after assigning the maximum possible facilities to the arcs in L (if M is chosen to be sufficiently large, as specified in Theorem 3). Constraint (21) stipulates that no flow is conserved on the intermediate nodes. Constraint (22) restricts the non-negative flow on each arc not to exceed the capacity of the arc. Constraint (23) makes sure that the size of a facility does not exceed the capacity of the arc to which it is assigned. Constraint (25) ensures that the number of facilities assigned to an arc does not exceed the number of facilities that the arc can host. Constraint (26) guarantees that each facility is assigned to at most one arc in L. Constraint (27) makes each y_{ijp} a binary variable. The value $y_{ijp} = 1$ means that the facility $p \in P$ is assigned to $(i,j) \in L$. If we choose M to be more than the value of the maximum flow without facilities assignment, we claim that the maximum number of possible facilities assigned by the model. This is proved in the following theorem.

Theorem 3. Let v^* be the value of maximum static s–t flow without facility assignment. If $M > v^*$, then objective (20) maximizes the number of facilities and the value of the static flow after facilities assignment.

Proof: Let

$$v = \sum_{j\in V_s^+} x_{sj} - \sum_{j\in V_s^-} x_{js}\,, Y = \sum_{(i,j)\in L}\sum_{p\in\mathbb{P}} y_{ijp}\,, \text{and } Z = v + MY.$$

Suppose that Y can be increased to $Y_1 = Y + \delta$ which results in v changing to v_1. So, $\delta \geq 1$ and since the facility assignment does not increase the value of the flow, $v_1 \leq v$. Let $Z_1 = v_1 + MY_1$.

Now

$$\begin{aligned} Z_1 - Z &= v_1 + MY_1 - v - MY \\ &= v_1 - v + M\delta \end{aligned}$$

Since v^* is the maximum value of the s–t flow, if $M > v^*$,

$$Z_1 - Z > v_1 - v^* + v^*\delta = v_1 + (\delta - 1)v^* \geq 0$$

showing that $Z_1 > Z$. So, if we maximize $v + MY$ with $M > v^*$, both Y and v will be maximized.

Since the maximum static FlowLoc problem is a special case of the corresponding problem with arbitrary facilities, analogous to Theorem 2, we have the following theorem.

Theorem 4. *There is no polynomial time α-approximation algorithm to solve maximum static $|\mathbb{P}|$-FlowLoc problem with arbitrary facilities for a finite constant α, unless $P = NP$ for $|\mathbb{P}| > 1$.*

Since there does not exist an approximation algorithm that runs in polynomial time, we construct a polynomial time heuristic (Algorithm 3) to solve the maximum static FlowLoc problem with arbitrary facilities. Note that given a set of facilities P with arbitrary sizes, each facility may not fit in each of facility locations in L. First, we present a procedure in Algorithm 2 that finds a maximal subset of facilities with minimum possible sizes.

Algorithm 2: Feasible Facilities Assignment

Input: Set of feasible locations L, set of facilities $\mathbb{P}$

Output: Set of feasible facilities F with its assignment to L

```
F := ∅
i := 1, j := 1
L := {l_1, l_2, ···, l_|L|} such that u_{l_1} ≤ u_{l_2} ≤ ··· ≤ u_{l_|L|}
P := {p_1, p_2, ···, p_|P|} such that r_{p_1} ≤ r_{p_2} ≤ ··· ≤ r_{p_|P|}
while i ≤ |L| and j ≤ |P| do
   for k := 1 to η_{l_i}
     if j ≤ |P| and r_{p_j} ≤ u_{l_i} then
          Assign p_j to l_i
          F := F ∪ {p_j}
          j := j + 1
     else
          break
     end if
```

end for
$i \coloneqq i + 1$
end while

Example 2. For the problem given in Example 1, we apply Algorithm 2 to write $L = \{(a,b),(s,a),(s,b)\}$ with the non-decreasing order of their capacities, $\mathbb{P} = \{p_1, p_2, p_3, p_4, p_5\}$ with non-decreasing order of their sizes. Now the algorithm assigns p_1 to (a,b), p_2, p_3 to (s,a) and p_4 to (s,b). In this way $F = \{p_1, p_2, p_3, p_4\}$.

Algorithm 3 starts by ordering the set of feasible facilities F from $\mathbb{P}$ (found using Algorithm 2) in the decreasing order of their sizes and finding the maximum s–t flow in the network. If there is an arc $l^* \in L$ that has enough capacity left $(u_l^* - x_l^*)$ to hold a facility p_f with the maximum size r_{p_f} among the unassigned facilities, the facility p_f is assigned to l^*, otherwise, l^* is identified by solving the single-facility maximum static FlowLoc problem with facility size r_{p_f}. Then η_{l^*} facilities are assigned to l^*, the capacity of l^* is reduced by the facility size r_{p_f}, and it is removed from L. The procedure is continued until there are unassigned facilities in F. The computational experiments presented in Section 4 show that Algorithm 3 performs very well in practical road networks.

Algorithm 3: Maximum static FlowLoc Heuristic

Input: Network $G = (V, A, u, s, t)$, set of possible locations $L \subseteq A$ with number of locations η_{ij} $\forall (i,j) \in L$, set of facilities $\mathbb{P}$ with size of the facility r_p $\forall p \in P$
Output: The facilities assignment loc: $\mathbb{P} \to L$
Find a set of feasible facilities F using Algorithm 2
Order the facilities in F such that $r_{p_1} \geq r_{p_2} \geq \cdots \geq r_{p_{|F|}}$
$f \coloneqq 1, l^* \coloneqq None$
while $f \leq |F|$ **do**
  $x \coloneqq$ maximum static flow in G
  if $r_{p_f} \geq \max\{u_l - x_l : l \in L\}$ **then**
    $l^* \coloneqq \arg\max\{u_l - x_l : l \in L\}$
  else
    $l^* \coloneqq$ single-facility flow location with facility size r_{p_f}
  end if
  for $i = 1$ to η_{l^*} **do**
    loc$(p_f) \coloneqq l^*$
    $f \coloneqq f + 1$
  end for
  $L \coloneqq L - \{l^*\}$
  $u_{l^*} \coloneqq u_{l^*} - r_{p_f}$
end while

Maximum Dynamic FlowLoc Model with Facilities of Arbitrary Sizes
Given the network $G = (V, A, s, t, u, \tau)$ and a time horizon T, the set of facilities $\mathbb{P}$ with size r_p for each $p \in \mathbb{P}$, and the set of possible locations L with number η_{ij} of facilities that $(i,j) \in L$ can host, the maxmum dynamic FlowLoc problem can be set as

$$\max\ T\left(\sum_{j \in V_s^+} x_{sj} - \sum_{j \in V_s^-} x_{js}\right) - \sum_{(i,j) \in A} \tau_{ij} x_{ij} + M \sum_{(i,j) \in L} \sum_{p \in \mathbb{P}} y_{ijp} \tag{28}$$

subject to the constraints (21) – (27).

With the similar arguments given in Theorem 3, we have the following.

Theorem 5. *Let v^* be value of the maximum dynamic s–t flow without facility assignment. If $M > v^*$, then objective (28) maximizes the number of facilities and the value of the dynamic flow after facilities assignment.*

Likewise, we cannot find an efficient approximation algorithm to solve the maximum dynamic FlowLoc problem with arbitrary facilities. Analogous to Theorem 4, we have the following result.

Theorem 6. There is no polynomial time α-approximation algorithm to solve maximum dynamic $|\mathbb{P}|$-FlowLoc problem with arbitrary facilities for a finite constant α,unless $P = NP$ for $|P| > 1$.

Proof. Taking $T = 1$ and $\tau_{ij} = 0\ \forall(i,j) \in A$, the objective (28) becomes (20) and the maximum dynamic $|\mathbb{P}|$-FlowLoc problem reduces to the maximum static $|\mathbb{P}|$-FlowLoc problem. Now, using Theorem 4, the result is immediate.

As in earlier cases, we cannot construct efficient exact or approximate algorithms to solve the maximum dynamic FlowLoc problem. However, we can easily translate Algorithm 3 to the dynamic flow case that performs very well in practical situations. The procedure is given in Algorithm 4.

Algorithm 4: Maximum dynamic FlowLoc Heuristic

Input: Network $G = (V, A, u, s, t, \tau)$, time horizon T, set of possible locations $L \subseteq A$ with number of locations $\eta_{ij}\ \forall(i,j) \in L$, set of facilities $\mathbb{P}$ with size of the facility $r_p\ \forall p \in P$

Output: The facilities assignment loc: $\mathbb{P} \rightarrow L$

Find a set of feasible facilities F using Algorithm 2
Order the facilities in F such that $r_{p_1} \geq r_{p_2} \geq \cdots \geq r_{p_{|F|}}$
$f := 1, l^* := None$
while $f \leq |F|$ **do**
    $x :=$ static flow corresponding to the maximum dynamic flow with time horizon T in G
    if $r_{p_f} \geq \max\{u_l - x_l : l \in L\}$ **then**
        $l^* := \arg\max\{u_l - x_l : l \in L\}$
    else
        $l^* :=$ single-facility flow location with facility size r_{p_f}
    end if
    for $i = 1$ to η_{l^*} **do**
        loc$(p_f) := l^*$
        $f := f + 1$
    end for
    $L := L - \{l^*\}$
    $u_{l^*} := u_{l^*} - r_{p_f}$
end while

COMPUTATIONAL EXPERIMENTS

We chose the area nearby Dattatreya temple as the source and Sallaghari area as the sink (Fig. 2). The capacity of a road segment is taken as the width of the road which ranges from 3 to 13. The set of facilities $\mathbb{P}$ containing various numbers of facilities as shown in Table 1. The sizes of the facilities are chosen randomly between 1 and 13. The set of possible facilities L is chosen at random. The number of facilities each arc in L can host is chosen at random between 1 to 5. For each $|L|$ and $|\mathbb{P}|$ shown in Table 1, we consider 10 instances, and the running times of MILP and Algorithm 3 are compared. For solving MILP, we have used HiGHS solver (Huangfu & Hall, 2018) which is one of the high performance solvers for solving large-scale mixed integer programming models. Setting the solver time of 10 minutes, we the solver is unable to find the optimal solution for 2 instances with $|L|$ = 100, $|\mathbb{P}|$ = 150, and 2 instance of $|L|$ = 100, $|\mathbb{P}|$ = 250.

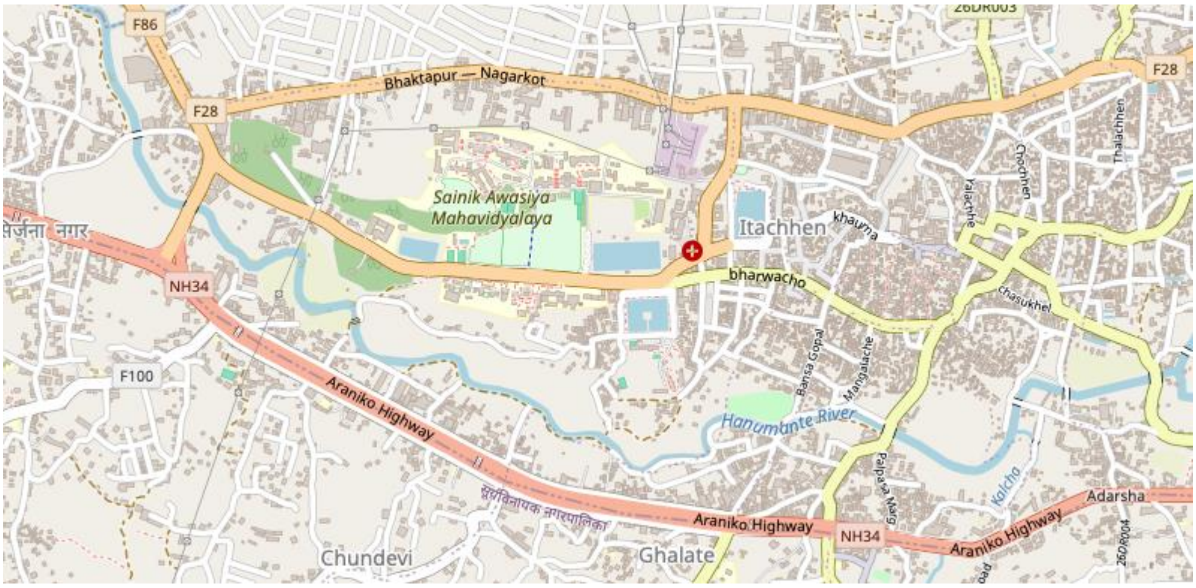

Figure 2. Map of Bhaktapur City Nepal for Computational Experiments

For computational experiments, we consider a road network of Bhaktapur city Nepal. The network data is obtained from OpenStreetMap using OSMnx (Boeing, 2024). The network has 502 vertices and 1354 arcs (Fig. 3).

Figure 3. Bhaktapur Road Network, $|V| = 502, |A| = 1354$

It is found that Algorithm 3 outperforms the HiGHS solver in the running time. With $|L|$ as large as 100 and $|P|$ as large as 250, the median running time is less than half a second. Out of 160 instances, only 18 instances have an optimality gap (between the flow values obtained in by the solver HiGHS and by Algorithm 3) which is 9.19% on average with a maximum of 14.29%.

In similar settings, the results for the maximum dynamic FlowLoc are presented in Table 2. Out of 160 cases, the HiGHS solver cannot terminate within the set time of 10 minutes in 15 cases with $|L| = 100$, $|\mathbb{P}| \geq 100$ while Algorithm 4 terminates with a maximum median time of 3.9 seconds. Among the remaining cases, there is a optimality gap between the dynamic flow values given by HiGHS solver and Algorithm 4 in 20 cases with a maximum gap of 22% and average gap of just 6.9%. The coding of the computational experiments is done mainly in Julia programming language. The network data are obtained using Python programming in MacOS with 1.8 GHz Dual-Core Intel Core i5 processor and 8 GB RAM.

Table 1. Running Time (in seconds) Comparisons (Maximum static FlowLoc)

		MILP HiGHS solver running time			Algorithm 3 running time		
$\lvert L \rvert$	$\lvert \mathbb{P} \rvert$	Maximum	Median	Minimum	Maximum	Median	Minimum
10	20	1.9992	0.9711	0.871	0.0845	0.0585	0.0053
10	40	1.2789	1.0525	0.9745	0.1486	0.0867	0.0686
10	50	3.2776	1.1172	0.9598	0.1447	0.1002	0.0698
20	20	1.8309	1.0499	0.9688	0.0829	0.0748	0.0599
20	50	3.9827	1.4273	1.0274	0.1169	0.0895	0.073
20	80	7.3981	2.4078	1.2695	0.1985	0.1274	0.0728
20	100	6.4552	1.6316	1.1181	0.193	0.1329	0.086
40	60	12.92	4.8959	1.294	1.2841	0.1658	0.1264
40	80	36.4384	5.6216	1.9459	0.2552	0.1599	0.1052
40	100	29.2741	10.6303	1.8837	0.2445	0.1441	0.1058
50	50	6.0078	2.7629	1.695	0.2872	0.1422	0.1111
50	75	42.6668	10.6992	2.6358	0.2116	0.1534	0.1259
50	125	89.8906	19.0245	7.1632	0.2694	0.1744	0.1283
100	100	308.5865	21.3646	10.3396	0.418	0.2506	0.2164
100	150	-	342.4617	47.5277	1.1781	0.3593	0.2472
100	250	-	426.5246	77.4677	1.8281	0.4066	0.1814

Table 2. Running Time (in seconds) Comparisons (Maximum dynamic FlowLoc)

		MILP HiGHS solver running time			Algorithm 3 running time		
$\lvert L \rvert$	$\lvert \mathbb{P} \rvert$	Maximum	Median	Minimum	Maximum	Median	Minimum
10	20	2.1382	0.4708	0.3779	0.5905	0.1862	0.0724
10	40	3.5833	0.8727	0.4516	0.3966	0.2426	0.1012
10	50	29.8164	1.3771	0.3992	0.881	0.2499	0.1098
20	20	1.4383	0.6188	0.3654	0.4399	0.263	0.1334
20	50	13.6587	2.2117	0.4874	0.8997	0.4395	0.111
20	80	10.6863	3.2288	0.6961	0.4495	0.3441	0.11
20	100	8.6209	2.3724	0.2232	1.082	0.4805	0.2461
40	60	52.6433	8.2008	1.1132	1.7058	0.4895	0.1073
40	80	38.9494	14.5926	4.6544	1.6964	0.887	0.3815
40	100	103.378	13.1534	3.3732	1.7977	0.7166	0.4309
50	50	32.5465	4.6223	0.5566	0.9037	0.3611	0.0955
50	75	40.5402	16.3165	2.4597	1.3676	0.6177	0.1581
50	125	79.4816	17.7182	1.855	2.1923	0.8985	0.5929
100	100	-	20.8028	13.8726	1.9458	0.7531	0.3216
100	150	-	-	24.8462	3.831	2.2406	1.3239
100	250	-	-	27.9191	7.0353	3.9418	2.5819

CONCLUSIONS

In this work, we extend the existing multi-facility maximum flow location modeling to the cases where the facilities can have arbitrary number and arbitrary size. The maximum static FlowLoc model and the maximum dynamic FlowLoc model with arbitrary facilities are set to choose the maximum number of possible facilities to maximize the static/dynamic flow after assigning the facilities to the given set of arcs.

We model the problems as mixed integer programming problems. The problems are found to be NP-hard with no polynomial time approximation algorithms. To find practically good solutions, we design polynomial time heuristic algorithms. Such an algorithm for the maximum static flow location case has been tested in the network of Bhaktpur city Nepal with 502 vertices and 1354 arcs. Taking varying number of facilities and possible facility locations, the algorithm is found to obtain the optimal solution in more than 88% instances out of 160 instances, considering giving average optimality gap of 9% in the remaining instances.

The considered models are macroscopic in nature, i.e., they do not consider the individual behaviors of the flow units. That means when considering the parameters and interpreting the results, the law of averages has to be kept in mind. Although the heuristic algorithms for multi-facility maximum FlowLoc cases give optimal solutions or near optimal solutions in the considered case study, there is no theoretical guarantee of the quality of the solutions in other cases, especially in larger networks. The algorithms are particularly useful in emergency planning because such cases demand a well-organized plan instead of a perfectly optimal plan. Additionally, the results of heuristic algorithms can be good starting solutions in the algorithmic frames, e.g. genetic algorithms, which continuously improve the solutions until better solutions are obtained for the available time and other computational resources. As flow location modeling is a relatively new area of research, there can be several extensions of this modeling. The following are some of the immediate future directions.

- As this work considers maximum static/dynamic flow location modeling with single-source-single sink cases only considering all the flow units as a single commodity, the modeling can be extended to cases with multiple sources, multiple sinks and multiple commodities.
- In the cases considered in this work, any facility can be placed at the available locations allowed by the capacity. The consideration of different categories of facilities in the modeling can also be a useful extension.
- The dynamic flow location modeling focusses on maximizing the flow within the given time horizon. An important and potential problem can be finding the minimum time horizon after assigning facilities, given the number of flow units at sources.

ACKNOWLEDGEMENTS
The author would like to acknowledge the support of University Grants Commission Nepal under Small Research, Development and Innovation Grants (SRDIG) to carry out the research.

CONFLICT OF INTERESTS
The authors declare no conflict of interests.

DATA AVAILABILITY STATEMENT
The data that support the findings of this study are available from the author, upon reasonable request.

REFERENCES
Ahuja, R.K., Magnanti, T.L., & Orlin, J.B. (1993). *Network flows: theory, algorithms, and applications*. Prentice hall.

Akter, S., & Wamba, S.F. (2019). Big data and disaster management: a systematic review and agenda for future research. *Annals of Operations Research, 283*(1-2), 939–959.

Balakrishnan, A., Ward, J.E., & Wong, R.T. (1987). Integrated facility location and vehicle routing models: Recent work and future prospects. *American Journal of Mathematical and Management Sciences, 7*(1-2), 35–61.

Berge, C. (1957). Two theorems in graph theory. *Proceedings of the National Academy of Sciences of the United States of America, 43*(9), 842.

Bhandari, P.P., Khadka, S.R., Ruzika, S., & Schäfer, L.E. (2020). Lexicographically maximum dynamic flow with vertex capacities. *Journal of Mathematics and Statistics, 16*(1), 142–147.

Boeing, G. (2024). Modeling and analyzing urban networks and amenities with osmnx [Working paper]. Retrieved January 14, 2025 from https:// geoffboeing.com/publications/ osmnx-paper/

Cova, T.J., & Johnson, J.P. (2003). A network flow model for lane-based evacuation routing. *Transportation Research Part A: Policy and Practice, 37*(7), 579–604.

Dhamala, T.N., Adhikari, M.C., Khanal, D.P., & Pyakurel, U. (2024). Generalized maximum flow over time with intermediate storage. *Annals of Operations Research, 335*(1), 111–134.

Dhamala, T.N., Pyakurel, U., & Dempe, S. (2018). A critical survey on the net work optimization algorithms for evacuation planning problems. *International Journal of Operations Research, 15*(3), 101–133.

Dhamala, T.N., Wagle, S., & Pyakurel, U. (2023). Flowloc problems with maximum excess flow. *Journal of Industrial and Management Optimization, 19*(12), 8851–8870.

Ford, L.R., & Fulkerson, D.R. (1962). *Flows in networks*. New Jersey: Princeton University Press.

Goldberg, A.V., & Tarjan, R.E. (1989). Finding minimum-cost circulations by canceling negative cycles. *Journal of the ACM (JACM), 36*(4), 873–886.

Hakimi, S.L. (1965). Optimum distribution of switching centers in a communication network and some related graph theoretic problems. *Operations Research, 13*(3), 462–475.

Hamacher, H.W., Heller, S., & Rupp, B. (2013). Flow location (FlowLoc) problems: dynamic network flow and location models for evacuation planning. *Annals of Operations Research, 207*(1), 161–180.

Hamacher, H.W., & Tjandra, S.A. (2001). *Mathematical modelling of evacuation problems: A state of art*. Fraunhofer-Institut für Techno-und Wirtschaftsmathematik, Fraunhofer (ITWM).

Heßler, P., & Hamacher, H.W. (2016). Sink location to find optimal shelters in evacuation planning. *EURO Journal on Computational Optimization, 4*(3-4), 325–347.

Huangfu, Q., & Hall, J.J. (2018). Parallelizing the dual revised simplex method. *Mathematical Programming Computation, 10*(1), 119–142.

Kim, S., Shekhar, S., & Min, M. (2008). Contraflow transportation network reconfiguration for evacuation route planning. *IEEE Transactions on Knowledge and Data Engineering, 20*(8), 1115–1129.

Köhler, E., Langkau, K., & Skutella, M. (2002). Time-expanded graphs for flow dependent transit times. In *European symposium on algorithms* (pp. 599–611).

Kotsireas, I.S., Nagurney, A., & Pardalos, P.M. (2015). *Dynamics of disasters-key concepts, models, algorithms, and insights*. Springer Proceedings in Mathematics and Statistics.

Kuby, M.J. (1987). Programming models for facility dispersion: The p-dispersion and maxisum dispersion problems. *Geographical Analysis, 19*(4), 315–329.

Laporte, G., Nickel, S., & Saldanha-da-Gama, F. (Eds.). (2019). *Location science*. Switzerland: Springer.

Nath, H.N., Dempe, S., & Dhamala, T.N. (2021). A bicriteria approach for saving a path maximizing dynamic contraflow. *Asia-Pacific Journal of Operational Research*, 39(3), 2150027.

Nath, H.N., & Dhamala, T.N. (2018). Network flow approach for locating optimal sink in evacuation planning. *International Journal of Operations Research, 15*(4), 175–185.

Nath, H.N., Dhamala, T.N., & Dempe, S. (2024). Saving a path minimizing egress time of a dynamic contraflow: a bi-objective programming approach. *OPSEARCH, 61*(1), 98–120.

Nath, H.N., Pyakurel, U., Dhamala, T.N., & Dempe, S. (2021). Dynamic network flow location models and algorithms for quickest evacuation planning. *Journal of Industrial and Management Optimization, 17*(5), 2925–2941.

Orlin, J.B. (2013). Max flows in O(nm) time, or better. In *Proceedings of the forty-fifth annual acm symposium on theory of computing* (pp. 765–774).

Pyakurel, U., & Dempe, S. (2019). *Network flow with intermediate storage: Models and algorithms*. Preprint, TU Bergakademie, Freiberg.

Pyakurel, U., & Dempe, S. (2020). Network flow with intermediate storage: models and algorithms. *SN Operations Research Forum, 1*, 37.

Pyakurel, U., & Dhamala, T.N. (2017). Continuous dynamic contraflow approach for evacuation planning. *Annals of Operations Research, 253*(1), 573–598.

Pyakurel, U., Nath, H.N., & Dhamala, T.N. (2018). Efficient contraflow algorithms for quickest evacuation planning. *Science China Mathematics, 61*(11), 2079–2100.

Rebennack, S., Arulselvan, A., Elefteriadou, L., & Pardalos, P.M. (2010). Complexity analysis for maximum flow problems with arc reversals. *Journal of Combinatorial Optimization, 19*(2), 200–216.

Skutella, M. (2009). An introduction to network flows over time. In Cook, W., Lovász, L., & Vygen, J. (Eds.), *Research trends in combinatorial optimization* (pp. 451-482). Springer.