Kathmandu University Journal of Science, Engineering and Technology, Vol. 20 No. 1, April 2026

Kathmandu University
Journal of Science, Engineering and Technology

Multi-facility allocation in network flow models: A case study

Swikar Jaiswal , Sabal Humagain , and Ram Chandra Dhungana*

Department of Mathematics, Kathmandu University, Dhulikhel, Kavre, Nepal.

Abstract

Optimizing network flow with facility allocation requires strategic placement and efficient traffic management. The considered model integrates the network flow model with location analysis. Allocation of facilities on arcs changes the capacity of the arcs, which reduces the optimality of the original network. The multi-facility allocation in the network flow model is the $\mathcal{NP}$-hard problem. An acceptable solution to the k-FlowLoc problem can be obtained via an auxiliary graph H that incorporates facility-location compatibility constraints and maximum-flow computation. This transformation ensured a one-to-one correspondence between feasible allocations and maximum flows of value k in H. Optimization objectives are guided by cost functions on H's edges, significantly influencing facility placement quality. Computational results with different cost functions demonstrate that this approach provides a scalable alternative to exact solutions. The solution with a dataset of Asan, Kathmandu, Nepal, validates its practical applicability in real-world applications and motivates further exploration of a new cost function.

Keywords: Network flow; Facility placement; Auxiliary network; Cost function; Flow maximization.

1. Introduction

The dynamic network flow theory of Ford and Fulkerson [1] provides the groundwork for time-dependent evacuation modeling, which has been extended and applied to address scenarios in buildings, urban areas, and large facilities (see [2]). A main challenge lies in mitigating congestion near shelters, where bottlenecks often delay movements. Thus, routing evacuees efficiently to safe destinations while respecting capacity and operational constraints is crucial. Among various mathematical approaches: simulation, fluid dynamics, control theory, and network flow designs remain the most computationally efficient and analytically robust. Fundamental flow models primarily aim to either maximize total flow or minimize overall cost. Dynamic variants of network flow includes the earliest arrival, quickest, lexicographically maximal, and quickest transshipment flows, each presenting different time-based and priority aspects.

The placement of facilities such as hospitals, warehouses, emergency service centers, and security units plays an important role in managing both emergency and disaster situations, as well as daily operations. In both contexts, location optimization is supported by mathematical models designed to balance coverage, approachability, and efficiency. Facility location approaches cover several mathematical models. Covering problem that selects sites to serve either all or the optimum possible number of demand points. p-median models that position facilities to minimize total or mean travel distance. p-center models that reduce the greatest distance between any demand point and its closest facility. These concepts were applied by Jia et al. [3] in facility location models. Under uncertain demand, the pick-up location models [4, 5] determine optimal collection points and evacuee assignments to minimize system costs or optimum travel time. Rescue transfer location models [4] optimize the placement of intermediate rescue centers to reduce total travel cost. Shelter location models [6, 7, 8] apply hierarchical and multi-objective formulations to minimize vehicle hours and total evacuation time, linking upper-level shelter decisions with lower-level traffic assignment problems.

FlowLoc models combine facility location and flow optimization, analyzing the impact of facility placement along road segments on the network [9, 10]. For a more realistic representation of evacuation scenarios, dynamic flow models account for time-variant as well as capacity limitations [11]. Network flow-based methods have established a strong capability in managing large-scale evacuation systems. Consequently, the allocation of facilities in evacuation networks directly influences flow optimality and the effectiveness of emergency responses. Contraflow ideas, where lanes are reversed to increase outbound capacity, have also been applied to FlowLoc and have shown improvements in flow and evacuation time [12, 13]. Prioritized FlowLoc for prioritized flow has been provided with polynomial time solution for single facility and heuristics for multi-facility [14]. FlowLoc has grown from a simple capacity reduction idea into a broader set of tools such as contraflow, time dependent behavior, and storage effects. Even with these extensions, the auxiliary graph approach remains the main practical method for placing multiple facilities in larger networks.

This study investigates the FlowLoc model for multi-facility allocation. To examine the influence of the facility, different cost settings in the auxiliary graph are considered. The corresponding reductions are compared on the basis of heuristics and the exact solution of the mixed integer formulation of the integrated model. For the experiments, a dataset of Asan Kathmandu, Nepal, is taken. Results are compared with twenty different cost functions. Their effects are observed on total flow, placement patterns, and the gap between heuristic and optimal results. The findings show that cost designs help to achieve more flow with reduced capacities, and modeling choices shape final placements. These results offer practical guidance for using FlowLoc methods in dense urban settings.

Section 2 describes the FlowLoc problem and its solution status. Section 3 explains the idea of constructing auxiliary graph and shows how feasibility checks and heuristic methods come from the

*Corresponding author. Email: ram.dhungana@ku.edu.np

minimum cost flow setup. Section 4 applies the model to a dataset of Asan, Kathmandu road network. Results are analyzed based on facility placements across all cost choices. Section 5 concludes the work.

2. Mathematical Formulation

Suppose a tuple $(G, \eta, \mathcal{F}, \sigma, \delta)$ where $G = (V, E)$ is a network with capacities $\eta : E \to \mathbb{N}$. The set $\mathcal{F}$ contains the facilities, each facility q with size $\sigma(q)$. The function $\delta(e)$ gives the maximum number of facilities that can be placed on an edge e. The set of candidate edges is $\mathcal{E} = \{e \in E : \delta(e) > 0\}$. The objective is to place all facilities on edges in $\mathcal{E}$ while maximizing the total flow between all possible source-sink pairs. The total number of facilities is denoted by $k = |\mathcal{F}|$ and the number of candidate edges by $|\mathcal{E}|$.

Let ϕ^{st} denote the flow between nodes s and t and let ϕ^{st}_{ij} be the corresponding flow on edge (i, j). A facility can only be placed on an edge whose capacity is at least as large as the facility size. If a facility is placed on an edge, the usable capacity becomes $\eta_e - \sigma_{q'}$, where q' is the largest facility placed on that edge. The binary variable ζ_{eq} is 1 if facility q is placed on edge e, and 0 otherwise. The mixed integer programming formulation of the problem presented in [10] is:

$$\max \sum_{(s,t) \in V \times V : s < t} \phi^{st}$$

Flow Conservation Constraints:

For each pair $(s, t) \in V \times V$, with $s < t$:

$$\textstyle\sum_{(i,t) \in E} \phi^{st}_{it} = \phi^{st} \tag{1}$$

$$\textstyle\sum_{i:(i,j) \in E} \phi^{st}_{ij} = 0 \quad \forall j \neq s, t \in V \tag{2}$$

$$\phi^{st}_e + \phi^{ts}_e = 0 \quad \forall e \in E \tag{3}$$

Facility Placement Constraints:

$$\textstyle\sum_{q \in \mathcal{F}} \zeta_{eq} \leq \delta_e \quad \forall e \in \mathcal{E} \tag{4}$$

$$\textstyle\sum_{e \in \mathcal{E}} \zeta_{eq} = 1 \quad \forall q \in \mathcal{F} \tag{5}$$

$$\zeta_{ijq} = \zeta_{jiq} \in \{0, 1\} \quad \forall (i, j) \in \mathcal{E}, q \in \mathcal{F} \tag{6}$$

Capacity Constraints:

$$\phi^{st}_e + \sigma_q \cdot \zeta_{eq} \leq \eta_e \quad \forall e \in \mathcal{E}, (s, t) \in V \times V : s < t, q \in \mathcal{F} \tag{7}$$

$$\phi^{st}_e \leq \eta_e \quad \forall e \in E \setminus \mathcal{E}, (s, t) \in V \times V : s < t \tag{8}$$

This model ensures that each facility is placed exactly once respecting capacity limits, and follows the standard FlowLoc setup. Even though this formulation is exact, it is computationally infeasible for larger networks as k-FlowLoc is $\mathcal{NP}$-hard. A more scalable method from Heller and Hamacher [10] uses Theorem 1. Building the auxiliary graph H and computing a minimum cost flow of value k can be done in polynomial time. This reduces the combinatorial difficultly of problem to designing acceptable cost functions on the arcs (p_i, e_j).

Let $H = (V_H, E_H)$ be the directed auxiliary graph with vertices V_H, arcs E_H, and capacities defined as:

$$\begin{aligned}
V_H &= \{x, y\} \cup \{p_i : i = 1, \ldots, k\} \cup \{e_j : e_j \in \mathcal{E}\} \\
E_H &= \{(x, p_i) : i = 1, \ldots, k\} \\
&\quad \cup \{(p_i, e_j) : \sigma(p_i) \leq \eta(e_j)\} \\
&\quad \cup \{(e_j, y) : e_j \in \mathcal{E}\} \\
\eta_H(x, p_i) &= 1, \quad \eta_H(p_i, e_j) = 1, \quad \eta_H(e_j, y) = \delta(e_j)
\end{aligned}$$

Theorem 1. *There exists a feasible k-FlowLoc placement if and only if there exists an $x - y$ flow of value k in H. Moreover, flows of value k are in one-to-one correspondence with feasible placements [10].*

A path $x \to p_i \to e_j \to y$ represents placing facility p_i on edge e_j. The arc (e_j, y) controls how many facilities can be allocated to edge e_j. A minimum cost flow of value k produces k such paths and therefore gives a placement. The cost function $\texttt{cost} : A_H \to \mathbb{R}_{\geq 0}$ is specified using the choices in Table 1. Any arc not listed in the table receives cost zero.

Table 1: Cost functions for heuristic solutions

$\texttt{cost}(p_i, e_j)$	a $-\eta(e_j)$	b $-\eta(e_j)$ $+\sigma(p_i)$	c $-\eta(e_j)$ $*\delta(e_j)$	d $-\delta(e_j) * ($ $\eta(e_j) - \sigma(q_i))$	e 0
$\texttt{cost}(e_j, y)$	i 1	ii $-\delta(e_j)$	iii $-\eta(e_j)$	iv 0	

3. Facility Placement Optimization

With the mathematical formulation discussed, the next step is to obtain an acceptable solution. This is achieved by constructing an auxiliary directed graph where every valid facility-edge allocation is represented as unique $x - y$ path. A FlowLoc instance is feasible exactly when an $x - y$ flow of value k exists in this graph. Any such flow gives a complete allocation. After choosing a cost rule for the arcs, an allocation is obtained by running a minimum cost flow in the auxiliary graph. A plain maximum flow can also be computed to check feasibility before applying the cost function.

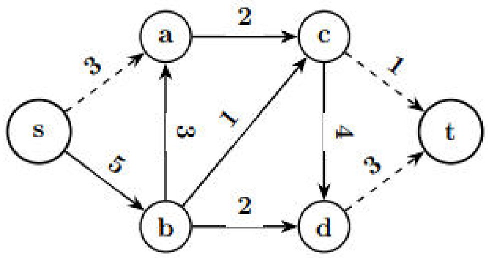
Algorithm 1 k-Facility Allocation on Network [10]

Require: Graph $G = (V, E)$; capacities $\eta : E \to \mathbb{N}$; feasible edges $\mathcal{E} \subseteq E$; facilities $\mathcal{F}$ with size $\sigma(q)$ for $q \in \mathcal{F}$; max placement count $\delta(e)$ for $e \in \mathcal{E}$; and a $\texttt{cost}$ function.
Ensure: Allocation ζ
1: Construct auxiliary graph $H = (V_H, E_H)$.
2: Compute ϕ, a minimum-cost (x, y)-flow in H of value k, with respect to $\texttt{cost}$.
3: **If** such a flow doesn't exist, **then** a complete allocation is infeasible; **terminate** the algorithm.
4: Derive allocation ζ as $\zeta_{eq} = \phi^{xy}_{qe}$ from flow ϕ
5: **return** ζ

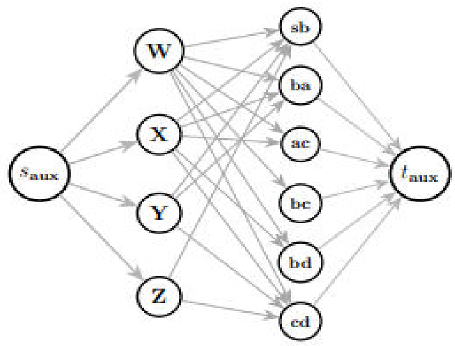
Algorithm 1 checks if all facilities can be placed and produces a feasible placement whenever one exists. The integral flow theorem states that a maximum-flow computation in H with integer capacities returns an integer flow. If the maximum (x, y)-flow in H has value $k = |\mathcal{F}|$, the flow splits into k arc-disjoint paths of the form $x \to q_i \to e_j \to y$. These paths define a feasible placement ζ that meets all constraints of the k-FlowLoc problem. Thus, the algorithm reports infeasibility when the flow is below k, otherwise returns a valid placement.

Let m, n denote the number of arcs and vertices in G while m_H, n_H denote the same for corresponding auxiliary graph H. The graph H has $O(k + |\mathcal{E}|)$ nodes and at most $k|\mathcal{E}| + k + |\mathcal{E}|$ arcs, so $m_H = O(km)$. Algorithm 1 calls a minimum-cost flow (MCF) computation on H.

Using any strongly polynomial successive-shortest-path or cost-scaling algorithm [15], the running time is $T_{\text{MCF}}(n_H, m_H, k) = O(k\, m_H \log n_H) = O\big(k^2 m \log(k+m)\big) = O\big(k^2 m \log(k+m)\big)$.

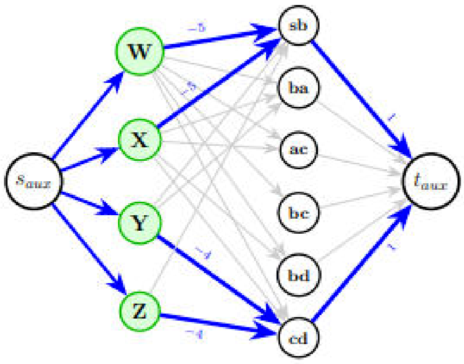


(a) Network with locations (solid edges)

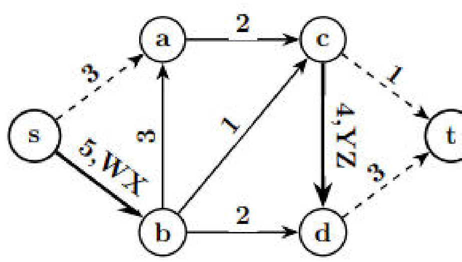


(b) Auxiliary graph H for network in Figure 1a

Figure 1: Network configuration and corresponding auxiliary graph construction



(a) Auxiliary graph for `cost` (a, i).



(b) Facility Allocation based on Figure 2a.

Figure 2: Cost function (a,i) prioritizes high-capacity edges for facility-location

Extracting the flow and deriving ζ takes linear time and does not change the final complexity. The main cost therefore comes from the minimum-cost flow step.

Table 2: Comparison of heuristic results under different cost-function formulations with optimal flow 1727. Best values appear in bold and the worst in italics.

Cost	Flow	Loss (%)	Food	Water	Vendor	Toilet
(a, i)	1500	13.14	(17, 21)	(16, 20)	(16, 20)	(11, 16)
(a, ii)	1487	13.90	(17, 21)	(6, 10)	(16, 20)	(16, 20)
(a, iii)	1500	13.14	(16, 20)	(16, 20)	(17, 21)	(11, 16)
(a, iv)	1500	13.14	(17, 21)	(16, 20)	(16, 20)	(11, 16)
(b, i)	1442	16.50	(17, 21)	(16, 20)	(16, 17)	(11, 16)
(b, ii)	1487	13.90	(17, 21)	(6, 10)	(16, 20)	(16, 20)
(b, iii)	1500	13.14	(16, 20)	(16, 20)	(17, 21)	(11, 16)
(b, iv)	1442	16.50	(17, 21)	(16, 20)	(16, 17)	(11, 16)
(c, i)	**1507**	**12.74**	(16, 20)	(16, 20)	(6, 10)	(6, 10)
(c, ii)	**1507**	**12.74**	(6, 10)	(6, 10)	(16, 20)	(16, 20)
(c, iii)	1486	13.95	(16, 20)	(16, 20)	(11, 16)	(17,21)
(c, iv)	**1507**	**12.74**	(16, 20)	(16, 20)	(6, 10)	(6, 10)
(d, i)	1471	14.82	(17, 21)	(11, 16)	(16, 20)	(16, 20)
(d, ii)	1487	13.90	(17, 21)	(6, 10)	(16, 20)	(16, 20)
(d, iii)	1471	14.82	(17, 21)	(16, 20)	(11, 16)	(16, 20)
(d, iv)	1471	14.82	(17, 21)	(11, 16)	(16, 20)	(16, 20)
(e, i)	***1409***	***18.41***	(16, 17)	(11, 16)	(6, 10)	(15, 19)
(e, ii)	1439	16.68	(13, 17)	(13, 17)	(6, 10)	(6, 10)
(e, iii)	1486	13.95	(16, 20)	(16, 20)	(11, 16)	(17, 21)
(e, iv)	***1409***	***18.41***	(16, 17)	(11, 16)	(6, 10)	(15, 19)

The method runs in polynomial time but still acts as a heuristic for the k-FlowLoc problem, which is $\mathcal{NP}$-hard. It guarantees an acceptable solution that with respect to the chosen cost function on H, but it does not guarantee a globally optimal solution.

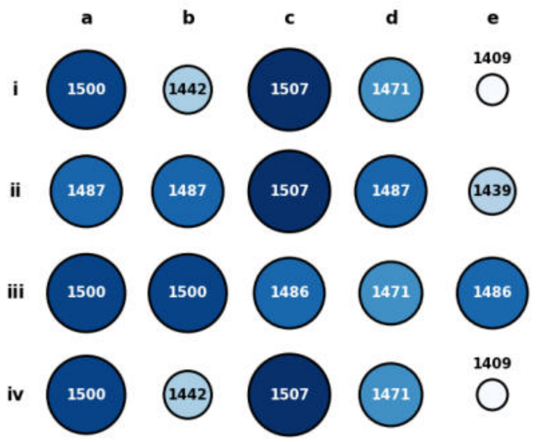
Example 2. Consider an undirected network $G = (V, E)$ with $V = \{s, a, b, c, d, t\}$. The network and its edge capacities are shown in Figure 1a. Facility placement is allowed only on the solid edges, which form the set $\mathcal{E}$. Each edge in $\mathcal{E}$ can hold up to $\delta(e) = 2$ facilities, except the edge (b, a), which allows $\delta(ba) = 4$ facilities. The facility set is $\mathcal{F} = \{W, X, Y, Z\}$ with sizes $\sigma(W) = 1$, $\sigma(X) = 2$, $\sigma(Y) = 3$, and $\sigma(Z) = 4$. The auxiliary graph shown in Figure 1b is constructed as mentioned in Step 1 of Algorithm 1. The colored arcs in Figure 2a represent the paths selected by the minimum cost flow. The final placement obtained from these paths is displayed in Figure 2b.

4. Case Study: Asan, Kathmandu Road Network

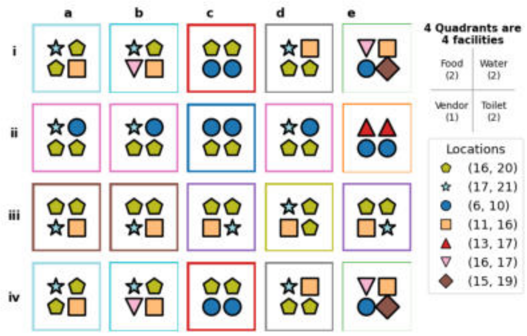
The central Asan area of Kathmandu has been modeled as an undirected network with 22 junctions and 33 road segments, as shown in Figure 4. Thirteen edges are chosen as candidate locations for placing facilities, and each edge has a placement limit as shown in Figure 5. Four types of facilities are assumed for the study: *Food Stall* (size 2), *Water ATM* (size 2), *Mobile Toilet* (size 2), and *Vendor* (size 1). After all facilities are placed, the multi-terminal maximum flow over all ordered pairs is recalculated. The aim is to pick locations for placement of facility that reduces the loss in total flow.

The auxiliary graph method i.e., Algorithm 1 was applied to all twenty cost function combinations listed in Table 1. The resulting flow values are shown in Figure 3a. Combinations `(e,i)` and `(e,iv)` represent the case where no cost is used on the auxiliary arcs and give a baseline flow of 1409. Since cost type `e` sets $\texttt{cost}(p_i, e_j) = 0$, the influence of $\texttt{cost}(e_j, y)$ can be seen by itself. Column `e` shows that using only the terms $-\eta(e_j)$ or $-\delta(e_j)$

increases the flow to 1439 and 1486 respectively. Both values are higher than the no cost baseline, which shows that even simple cost rules based on local capacity or placement limits can give noticeable improvement.



(a) Flow values across different cost combinations. Size of circles denotes relative flows.



(b) Facility allocations for all cost combinations, arranged in quadrants for Food, Vendor, Toilet, and Water.

Figure 3: Facility allocations for all cost combinations, arranged in quadrants for Food, Vendor, Toilet, and Water.

Across all cost pairs, the same pattern remain consistent. The terms $-\eta(e_j)$ (column `a`) and $-\delta(e_j)$ (part of column `d`) give strong results, while the product $-\eta(e_j)\delta(e_j)$ (column `c`) produces the best flows overall. Column `b`, which uses the cost $-\eta(e_j) + \sigma(p_i)$ and therefore favors arcs that keep more capacity after placement, also performs well. Cost family `d` gives better results than the no cost options but stays between the other families in quality. For this network, the top combinations are `(c,i)`, `(c,ii)`, and `(c,iv)`, each giving a residual flow of 1507.

Figure 3b shows the placement results for all cost pairs. Each square represents one pair and is divided into four parts for the four facilities. The symbols are used to mark the different candidate locations. It has been noticed that changing the cost structure leads to noticeably different placements. This means the cost design has a strong influence on facility allocation. Squares with the same color have the same flow value. Even within the best flow group (1507), there are two distinct allocations. High performing costs often place several facilities on the same arc, which reduces the total flow loss across the network. In contrast, the no cost cases `(e,i)` and `(e,iv)` spread facilities across many edges, causing larger flow reductions.

For benchmarking, the optimal placement was computed using PuLP's mixed integer solver. The optimal residual flow is 1727. This is achieved by placing all larger facilities on edges connected to node 6 (Keltol) near Seto Machindranath, and placing the vendor (size 1) on edge $(16, 20)$ near Ranjana Parking, as shown in Figure 5(d). Table 2 summarizes the results. Cost family `c` is the strongest overall, with loss values as low as 12.74%. The weakest cases are `(e,i)` and `(e,iv)`, both giving a flow of 1409 (loss 18.41%). These differences show how important cost choices are for guiding the heuristic toward better placements.

Figure 4: Aerial view of the case study area, Asan, Kathmandu.

On average, the optimal solution takes about 6 minutes to compute on an AMD Ryzen 5 5600H processor with 24GB RAM. A single run of the auxiliary graph heuristic takes only 0.1139 seconds. The heuristic therefore provides results more than 3000 × faster, making it suitable for real-time or large-scale use.

Overall, the results show that the auxiliary graph heuristic does not reach the true optimal solution but still performs well when the cost rules are chosen carefully. Costs based on $\eta(e_j)$ and $\delta(e_j)$ work reliably, and the product term gives the best outcomes in this network. These results point to the importance of cost design in getting good facility placements and show that the auxiliary graph method is practical for dense urban networks where fast computation is needed.

5. Conclusions

In this case study, the theoretical ground of flow models and location analysis on networks are studied. The FlowLoc framework combines flow routing with facility placement. It is shown to be $\mathcal{NP}$-hard for multi-facility allocation. Several heuristics with different cost structures in auxiliary-network were evaluated. Heuristics are validated using real data. Solutions provide a systematic approach for facility allocation and support practical decision-making. A dataset of congested urban area, Asan, Kathmandu is considered for the case study. It has been observed that roadside facilities reduce effective road capacity and restrict pedestrian movement. Results indicate that cost-guided auxiliary-graph heuristics offer an effective method for addressing the multi-facility allocation problem in network. Case study result suggests placing all larger facilities on edges connected to Keltol and placing the vendor near Ranjana Parking. Future extensions may include time-dependent capacities or demands and multi-commodity versions for flows such as pedestrians, emergency vehicles, or commercial traffic. Additional cost rules informed by measured traffic data, simulations, or data-driven models may further improve solution performance.

Acknowledgments

The authors thank the University Grant commission, Nepal, for the UGC Small RDI Grant(SDRIG-08/81-S &T-07) and the Department of Mathematics, School of Science, Kathmandu University.

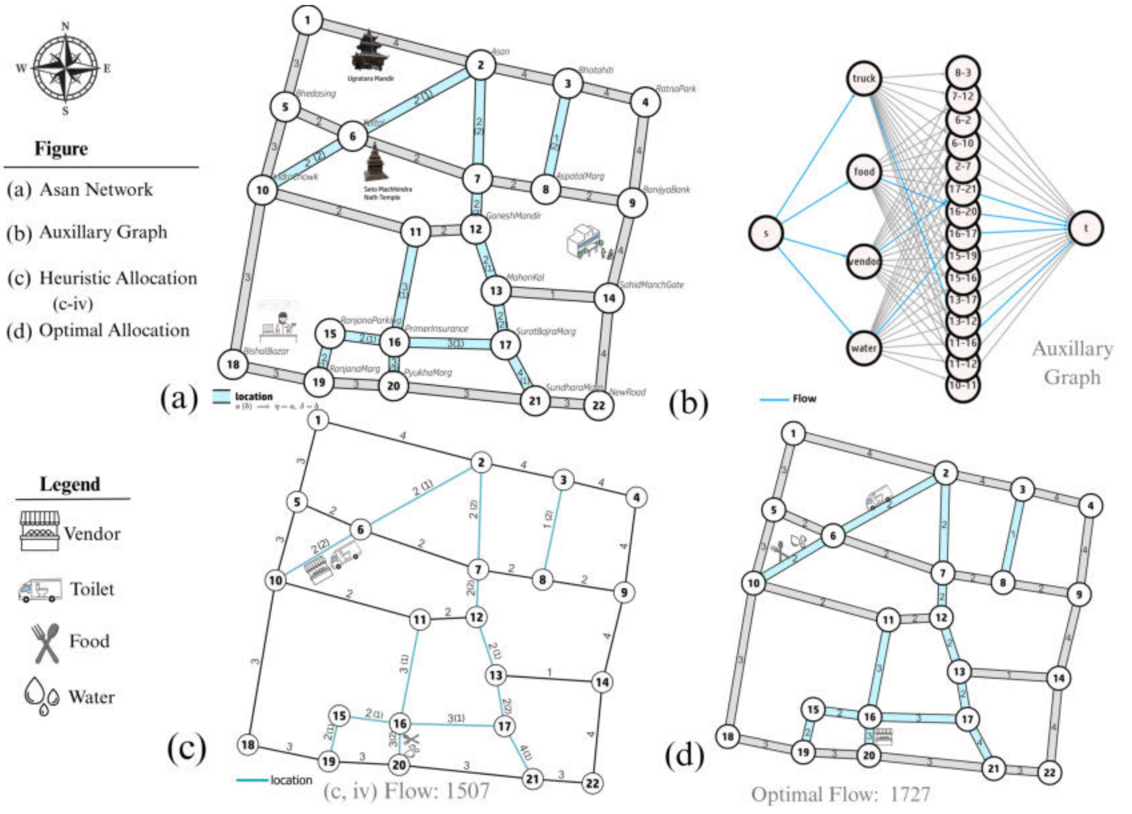


Figure 5: (a) Mapped network and candidate locations, (b) auxiliary graph representation, (c) heuristic allocation for cost type (c, iv), and (d) optimal allocation from PuLP solver.

References

[1] Ford L R & Fulkerson D R, Constructing maximal dynamic flows from static flows, *Operations Research*, 6(3) (1958) 419–433. ISSN 1526-5463. https://doi.org/10.1287/opre.6.3.419.

[2] Dhamala T N, Pyakurel U & Dempe S, A critical survey on the network optimization algorithms for evacuation planning problems, *International Journal of Operations Research*, 15(3) (2018) 101–133. ISSN 1813-713X. https://doi.org/10.6886/IJOR.201809_15(3).0002.

[3] Jia H, Ordóñez F & Dessouky M, A modeling framework for facility location of medical services for large-scale emergencies, *IIE Transactions*, 39(1) (2007) 41–55. ISSN 1545-8830. https://doi.org/10.1080/07408170500539113.

[4] An S, Cui N, Li X & Ouyang Y, Location planning for transit-based evacuation under the risk of service disruptions, *Transportation Research Part B: Methodological*, 54 (2013) 1–16. ISSN 0191-2615. https://doi.org/10.1016/j.trb.2013.03.002.

[5] Kulshrestha A, Lou Y & Yin Y, Pick-up locations and bus allocation for transit-based evacuation planning with demand uncertainty, *Journal of Advanced Transportation*, 48(7) (2014) 721–733. ISSN 2042-3195. https://doi.org/10.1002/atr.1221.

[6] KONGSOMSAKSAKUL S, YANG C & CHEN A, Shelter location-allocation model for flood evacuation planning, *Journal of the Eastern Asia Society for Transportation Studies*, 6 (2005) 4237–4252. https://doi.org/10.11175/easts.6.4237.

[7] Ng M, Park J & Waller S T, A hybrid bilevel model for the optimal shelter assignment in emergency evacuations, *Computer-Aided Civil and Infrastructure Engineering*, 25(8) (2010) 547–556. ISSN 1093-9687. https://doi.org/10.1111/j.1467-8667.2010.00669.x.

[8] Sherali H D, Carter T B & Hobeika A G, A location-allocation model and algorithm for evacuation planning under hurricane/flood conditions, *Transportation Research Part B: Methodological*, 25(6) (1991) 439–452. ISSN 0191-2615. https://doi.org/10.1016/0191-2615(91)90037-j.

[9] Hamacher H W, Heller S & Rupp B, Flow location (flowloc) problems: Dynamic network flows and location models for evacuation planning, *Annals of Operations Research*, 207(1) (2013) 161–180. ISSN 1572-9338. https://doi.org/10.1007/s10479-011-0953-9.

[10] Heller S & Hamacher H W. *The multi-terminal q-FlowLoc problem: A heuristic*. Springer Berlin Heidelberg (2011). ISBN 9783642215278, pp. 523–528. https://doi.org/10.1007/978-3-642-21527-8_57.

[11] Nath H N, Pyakurel U, Dhamala T N & Dempe S, Dynamic network flow location models and algorithms for quickest evacuation planning, *Journal of Industrial & Management Optimization*, 17(5) (2021) 2943. ISSN 1553-166X. https://doi.org/10.3934/jimo.2020102.

[12] Dhungana R C & Dhamala T N. Flowloc problems on evacuation network. In: *The 11th Triennial Conference of Association of Asia Pacific Operational Research Societies (APORS 2018, August 6-9)*. Operational Research Society, Nepal (2018), pp. 121–123. URL https://www.researchgate.net/publication/327097624_FlowLoc_Problems_on_Evacuation_Network.

[13] Dhungana R C & Dhamala T N, Maximum flowloc problems with network reconfiguration, *International Journal of Opera-*

tions Research, 16(1) (2019) 13–26. ISSN 1813-713X. https://doi.org/10.6886/IJOR.201903_16(1).0002.

[14] Wagle S, Pyakurel U & Dhamala T N, Prioritized max-flowloc for evacuation planning, *The Nepali Mathematical Sciences Report*, 40(1–2) (2023) 106–119. ISSN 2392-411X. https://doi.org/10.3126/nmsr.v40i1-2.61506.

[15] Ahuja R, Magnanti T & Orlin J. *Network Flows: Theory, Algorithms, and Applications*. Prentice Hall (1993). ISBN 9780136175490. URL https://books.google.com.np/books?id=WnZRAAAAMAAJ.

Appendix

Table 3: Summary of notations.

Notation	Description
$G = (V, E)$	Network with vertex set V and edge set E
$\delta(e)$	Maximum number of facilities allowed on edge e
ϕ^{st}	Flow between source s and sink t
ζ_{eq}	Binary variable indicating facility placement
ϕ^{st}_{ij}	Flow on edge (i, j) for source–sink pair (s, t)
$\sigma(q)$	Size of facility q
$\eta : E \to \mathbb{N}$	Capacity function
$\mathcal{F}$	Set of facilities
$k = \lvert\mathcal{F}\rvert$	Number of facilities
$H = (V_H, E_H)$	Auxiliary graph